THE CONTINUOUS ADJOINT METHOD FOR MULTI-FIDELITY HYPERSONIC INLET DESIGN

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF AERONAUTICS & ASTRONAUTICS
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Heather L. Kline
April 2017

Abstract

Hypersonic air-breathing propulsion systems offer greater efficiency through use of atmospheric oxygen rather than on-board compressed oxidizer tanks, for applications including access to space and hypersonic cruise but require significant further development. The high cost of wind tunnel and flight testing motivates the increased use of computer simulations and motivates advancements to these simulation techniques. A number of the challenges remaining in aerospacecraft design and simulation-based design techniques including gradient-based optimization and uncertainty quantification can be addressed by improving on methods of gradient computation. This dissertation focuses on developments towards generalizing a particular method of evaluating sensitivities and gradients (the continuous adjoint method) to a wider range of functions and multi-fidelity flowpath models. The primary contributions of this work are:

- The development of the continuous adjoint for a generalized outflow-based functional.
- A framework that utilizes this generalized functional to find the surface sensitivity for objectives defined external to the CFD volume, enabling multi-fidelity flowpath design.
- A multi-objective adjoint implementation that utilizes the principle of superposition to combine already-implemented functionals.
- Optimization studies utilizing these methods on a hypersonic inlet modeled using a multi-fidelity flowpath, including a three-dimensional viscous inlet, showing relatively large performance changes for small changes to geometry.

Acknowledgements

This work was supported by a NASA Space Technology Research Fellowship, under grant number NNX12AN23H. The Extreme Science and Engineering Discover Environment (XSEDE), which is supported by National Science Foundation grant number ACI-1053575,[1] was used for some of the results included in this work.

Professor Alonso has been a wonderful adviser, who has provided much insight and direction. I would also like to acknowledge the support of the Aerospace Design Lab and the SU2 development team, without which I would not have been able to complete the research contained in this thesis, never mind have fun while doing so. In particular, I would like to thank Dr. T.D. Economon and Dr. F. Palacios for their leadership in the development of SU2 many fruitful discussions.

The Hypersonic Airbreathing Propulsion Branch at NASA Langley has also been instrumental to my success, both through several works cited in this dissertation and through the advice and direction provided by Shelly Ferlemann, Aaron Auslender, Robert Baurle, Paul Ferlemann, and many others in this branch.

Last but not least, I want to thank my family, who have been endlessly supportive and encouraging throughout my education. Growing up around the doctors, linguists, engineers, nurses, teachers, world travelers, librarians, musicians, and mountain climbers who I am lucky to be related to, I am very privileged to have never doubted that I could, and should, do what I was passionate about - in part because of encouraging words, but more significantly because of the stellar examples I could see around me.

Contents

List of Tables

List of Figures

Nomenclature

Abbreviations

AD	Automatic Differentiation
CFD	Computational Fluid Dynamics
CFL	Courant-Friedrichs-Lewy number
DES	Detached Eddy Simulation
DNS	Direct Numerical Simulation
FVM	Finite Volume Method
HTOL	Horizontal Take-Off and Landing
IHF	Integrated Heat Flux
LES	Large Eddy Simulation
NASP	National AeroSpace Plane
PDE	Partial Differential Equation
RANS	Reynolds-Averaged Navier-Stokes
RBCC	Rocket-Based Combined Cycle
REST	Rectangular-to-Elliptical Shape Transition
SSTO	Single-Stage-To-Orbit
TBCC	Turbine-Based Combined Cycle
TPS	Thermal Protection System
TSTO	Two-Stage-To-Orbit

Physical Quantities

Kn	Knudsen number
M	Mach Number
Pr	Prandtl number
Re	Reynolds number
γ	ratio of specific heats
μ	viscosity $[kg\,m^{-1}\,s^{-1}]$
ρ	density $[kg\,m^{-3}]$
c	speed of sound $[m\,s^{-1}]$
c_p	constant-pressure specific heat $[J\,K^{-1}]$
E	internal energy $[J]$
h	enthalpy $[J]$
k	thermal conductivity $[W\,m^{-1}\,K^{-1}]$
P	pressure $[Pa]$
q	dynamic pressure $[Pa]$
R	specific gas constant $[J\,kg^{-1}\,K^{-1}]$
T	temperature $[K]$
t	time $[s]$
κ, H_m	surface curvature, mean surface curvature $[m^{-1}]$
$\vec{f}$	force vector $[N]$
$\vec{v} = (u, v, w)$	velocity vector $[m\,s^{-1}]$
x, y, z	Cartesian coordinates $[m]$
$\mathcal{A}$	cross-sectional area $[m^2]$
f	fuel fraction
h_{pr}	fuel products enthalpy $[J]$

$\dot{m}$ mass flow rate $[kg\,s^{-1}]$

η_{KE} kinetic energy efficiency

η_o overall efficiency

$\frac{\mathcal{F}_{un}-D_{est}}{\dot{m}}$ specific installed thrust $[m\,s^{-1}]$

$\mathcal{F}_{un}$ uninstalled (stream) thrust $[N]$

τ total temperature ratio

D_{est} drag estimate $[N]$

I_{sp} specific impulse $[s]$

P_{tr} total pressure recovery, $\frac{P_{t2}}{P_{t0}}$

Boundary, Volume Symbols

Γ boundary of computational volume

Ω computational volume

S solid wall boundary

Subscripts

0 upstream of inlet

2 inlet end, isolator entrance

3 isolator end, combustor entrance

4 combustor end, nozzle entrance

∞ freestream

e exit, or outflow

in inflow boundary

t total value

w solid wall boundary

Mathematical Symbols

$\mathcal{J}$ augmented objective function; Lagrangian

$\mathcal{Q}$	source term
$\mathcal{R}(U)$	residual
Ψ	vector of adjoint variables
$\vec{A}$	Jacobian matrices, $\vec{A} = (\partial\vec{F}/\partial U) = \{A_x, A_y, A_z\}$
$\vec{F}$	convective flux vector
$\vec{n}$	outward normal vector
$\vec{x}$	vector of design variables
$c(\vec{x})$	constraint function
j	arbitrary function
$J(\vec{x}),\ f(\vec{x})$	objective function
L	transformation matrix from U to W
U	Vector of flow conservative variables
V	vector of flow primitive variables
W	vector of flow conservative variables

Chapter 1

Introduction

What if spacecraft were completely reusable vehicles that could land or take off from a conventional airport? What if any point on earth was accessible within a couple of hours? What if small satellite launches were less expensive? Hypersonic air-breathing propulsion systems, which offer greater efficiency through use of atmospheric oxygen rather than on-board compressed oxidizer tanks, have the potential to make these ideas a reality, but require significant further development. Supersonic combustion ramjets, or scramjets, are airbreathing engines that operate in the hypersonic regime above Mach 5. Tests of scramjets have had partial success in flight tests such as the Hyper-X,[2] X-51,[3] and HIFiRE[4] projects. Sub-orbital horizontal takeoff and landing (HTOL) flights have been achieved by Scaled Composites, and much work has been accomplished towards reusable rockets by companies such as Space-X and Blue Origin. Research into hypersonic air-breathing propulsion and aerospacecraft design has been conducted in the United States and other countries, but progress has been limited in part by the high cost and risk of flight and wind tunnel tests. This motivates the increased use of computer simulations, and motivates advancements to simulation techniques. A number of the challenges of aerospacecraft design and simulation-based design techniques can be addressed by improving on methods of gradient computation, which facilitates gradient-based design and uncertainty quantification. This dissertation focuses on generalizing a particular method of evaluating sensitivities and gradients to a wider range of functions and multi-fidelity flowpath models, used specifically for design of hypersonic inlets and applicable to other problems.

The development of hypersonic engines is built on a long history of technological progress, from the development and growth of computers, to the first supersonic flights and the development of rocket fuels capable of taking payloads into orbit. Testing, simulation, and design techniques have also progressed. Early methods, many still used today, rely on high-cost experimental results to form empirical relations.[5] Analytical methods were developed, as the understanding of the underlying processes improved. Fluid dynamics simulation techniques were at first limited to potential flow, and then to inviscid flow, and more recently to Reynolds-Averaged Navier Stokes (RANS) flow, as

the power and utility of computers and maturity of computational methods has increased. Even higher fidelity simulations are now possible, including Large Eddy Simulation (LES),[6] Detached Eddy Simulation (DES),[7] Direct Numerical Simulation (DNS),[8] and simulations including plasma effects.[9] For engineering applications where large complex vehicles and a large number of flight conditions must be simulated, these more expensive simulations are often out of reach as a practical tool. It is expected that they will be used in the future, as the availability of computational resources increases and the algorithms become more efficient.

Computational Fluid Dynamics (CFD) simulations including Euler through DNS have been increasingly used in recent decades to reduce required wind tunnel and flight tests for aerospace applications including aircraft development, turbine engine development, and design of access-to-space and re-entry vehicles.[5] In the case of access-to-space and re-entry vehicles, there are sometimes no ground facilities capable of producing the conditions of interest.[10] Physics-based simulations also produce greater detail of the phenomena of interest, whereas experiments must contend with limits on the quantity and type of measurements that can be taken.[5] In general, simulation-based techniques reduce the cost of design. Between the computational costs of accurate simulation, the limitations of experimental facilities, and the complex design challenges of these engines, the design of hypersonic airbreathing propulsion systems stretches the current abilities of aerospace technology in several areas.

Improving upon simulation techniques and sensitivity evaluations is motivated by the challenges that have arisen in the use of CFD and the design of hypersonic engines. A variety of performance metrics should be considered, as well as the computational cost of running the simulations needed to evaluate these metrics as well as their gradients. In the realms of optimization, uncertainty quantification, mesh adaptation, and error estimation, it is not just the prediction of the relevant quantities, but also their gradients with respect to the design variables of the problem. Ideally, we would like to obtain gradients with respect to multiple objectives, and incorporating information from multiple models, at the highest efficiency possible. The computation of gradients, especially in problems with large numbers of design variables and quantities of interest, can be a large cost to the optimization process, potentially requiring hundreds of additional simulations for a single design step. By contrast, the continuous adjoint method provides gradients in a computationally efficient manner, while it suffers from a more limited range of objectives and limited system complexity. Expanding that range of objectives, and their complexity, for the continuous adjoint method is the subject of this thesis. The tools developed in this dissertation are applied to the shape optimization of a scramjet inlet.

Hypersonic airbreathing propulsion for access to space has been researched in various forms since the 1950s.[11] Several design, ground test, and flight test programs have been conducted in recent decades. The X-30, also known as the Orient Express, the HySTP program, or the National AeroSpace Plane (NASP) was one such project.[12] The X-30 was designed as a single stage to orbit

(SSTO) airbreathing vehicle, and as a potential replacement for the Space Shuttle. The NASP program office existed from 1986 to 1995. It was preceded by studies in the 60s for Single- and Two-Stage To Orbit (TSTO) aerospace planes. Studies including air-breathing propulsion as a potential replacement for the Space Shuttle began in 1975.[12] In the 90s the NASP program was restructured to include more incremental testing of technology.

Several conceptual design studies have been conducted, including studies investigating the trade-offs between different propulsion systems[13,14] as well as those investigating the system design with either a rocket-based combined cycle[15–18] (RBCC) or a turbine-based combined cycle[14,19] (TBCC). Professor Smart's group at the University of Queensland has done ample work both in testing and design of advanced scramjet engines.[20–23] This includes design of the REST-class inlets,[24] which will be discussed further in Section 2.2.1. The work in this dissertation is centered around a REST-class inlet[25] that was designed for conditions similar to HIFiRE Flight 2, a Mach 7, 1730 psf (82.832 kPa) flight.[4]

Flight and wind-tunnel test programs have built on and complemented conceptual design studies. The HyShot program[20] consisted of a double-wedge intake with supersonic combustion. The second flight test, in 2001, successfully reached the target altitude at Mach 7.5, as well as supersonic combustion for approximately 3 seconds. The X-43A, or Hyper-X program aimed to demonstrate hypersonic airbreathing propulsion systems that would enable the use of this technology for access to space.[26] It was hoped that the development of airbreathing propulsion would increase the safety and decrease the cost of space access vehicles. The X-43A employed a rectangular cross-section. The Hyper-X flight 2 at Mach 7 was successfully conducted in March 2004.[26] The Hypersonic Collaborative Australia/United States Experiment (HyCAUSE) flight test program[27] employed an inward-turning inlet with an elliptical cross-section. Inward-turning inlets can exceed the Kantrowitz limit, and this test employed moveable inlet geometry to facilitate starting. The flight and ground tests of this engine were conducted at Mach 10, dynamic pressure of 50-1600 psf (2.4-76.6 kPa). Both the X-43A and HyCAUSE projects were hydrogen-fueled. The HyCAUSE program began in 2004 with ground tests and CFD analyses, and the flight experiment was conducted in June 2007. The HIFiRE project[4] built on the HYCAUSE and HyShot programs. This engine uses a hydrocarbon fuel, and is intended to investigate acceleration performance as well as transition from dual-mode to scram-mode operation. The X-51A Scramjet Engine Demonstrator-WaveRider (SED) flight demonstrator was part of the Air Force Research Lab (AFRL) HyTech program. The X-51 was fueled by JP-7, and was intended to reach flight Mach numbers between Mach 6 and 6.5.[3] Its fourth flight test was successful at Mach 5.1,[28] with 210 seconds of scramjet combustion in May 2013.

Hypersonic airbreathing propulsion has been advanced from conceptual designs to flight tests over the last 50 years, yet despite the significant investments in this technology these engines still face difficulties in a number of areas including achieving thrust greater than drag, avoiding unstart,

and maintaining supersonic combustion. Moving from the current state of the art to using these engines for TSTO launch vehicles will require including complex physical phenomena and design trade-offs at an early stage in the design process. The work in this dissertation opens the door to doing just that—through increasing the complexity of problems that can be addressed by the continuous adjoint method, an efficient method of evaluating gradients that can now be applied to a wider range of gradient-based optimization problems.

1.1 Research Scope & Contributions

This section will enumerate the contributions of this dissertation to gradient evaluations and scramjet design as well as limitations in the scope. The continuous adjoint method is used to evaluate gradients, which are employed in gradient-based optimization for scramjet inlet shape design problems. While a large number of challenges are faced by scramjet design and the continuous adjoint method, only a small number of these challenges will be addressed by this dissertation.

The primary contributions of this dissertation are:

- The development of the continuous adjoint for a generalized outflow-based functional.
- A framework that utilizes this generalized functional to find the surface sensitivity for objectives defined external to the CFD volume, enabling multi-fidelity flowpath design.
- A multi-objective adjoint implementation that utilizes the principle of superposition to combine already-implemented functionals.
- Optimization studies utilizing these methods on a hypersonic inlet modeled using a multi-fidelity flowpath, including a three-dimensional viscous inlet, showing relatively large performance changes for small changes to geometry.

These contributions are tied together by the theme of generalizing the adjoint method to a wider range and complexity of objectives than previously thought possible, and by reducing the development time necessary to address new objectives with the adjoint method. Previously-available methods for the continuous adjoint require lengthy derivation for new functionals, and additional code implementation as well as verification of that implementation prior to use for real analysis. This has also been the case for combinations of objectives that have been treated in the past as a single objective, requiring separate derivation from their component parts. More significantly, however, these developments allow the use of the continuous adjoint to address functionals that would have not only required complex and unwieldy derivation and implementation, but would have been impossible to address due to the requirement that the functional be defined within the CFD volume and boundaries.

The scope of this dissertation covers the adjoint method for inviscid and viscous compressible flow, under the assumption of negligible viscous variations at the outflow boundary where the objective is evaluated. This assumption is consistent with the assumptions usually used for the continuous adjoint. A number of assumptions are made about the physical effects within the CFD volume that additionally limit the scope. Real gas effects (i.e., changing specific heats) are neglected within the CFD, although for the application studied these effects may be important. These effects are included in a one-dimensional external model, but neglected in the CFD volume. The adjoint for CFD with real gas effects is possible, and is the subject of much research,[29] however it is outside the scope of this work. Additional hypersonic effects that are not expected to be significant for the particular design case, including chemical reactions, ionization, and radiative heat transfer are also outside the scope and are neglected. In summary, the scope of this dissertation is limited to developments for sensitivity evaluation of existing physical models, and does not include modifications to those physical models.

Some of the methodology in this dissertation and preliminary results were included in previous work in applying the continuous adjoint method to multi-objective optimization,[30] deriving the continuous adjoint boundary conditions for an objective of mass flow rate,[31] using finite difference methods to evaluate the change in performance for a thermostatically deformed inlet.[32]

1.2 Scramjet Design

The tools developed in this dissertation are applied to the shape optimization of a scramjet inlet. This section will introduce the scramjet flowpath and discuss existing literature on scramjets and challenges relevant to their design. A scramjet is an air-breathing engine that uses the compression of air over the forebody and inlet to achieve the conditions necessary for supersonic combustion, using no mechanical compressor, as illustrated in Figure 1.1. The methods used to simulate a multi-fidelity flowpath are also summarized in this figure, and will be described further in Section 2.2 and Section 4.2. Some of the contributions of this dissertation are focused on the transfer of information between models that occurs at station 3 in Figure 1.1.

Scramjet engines operate in hypersonic conditions, ranging from Mach 5 to Mach 10 at current levels of technology. Above Mach 5, it becomes more efficient to utilize supersonic combustion rather than decellerating the air to subsonic speeds as is done in ramjets, which operate up to around Mach 5.5. A comparison between ramjet and scramjet efficiency is shown in Figure 1.2. Scramjets have the potential to facilitate more efficient trans-atmospheric flight and airplane-like operations of launch vehicles as well as military applications. Launch vehicle conceptual designs that incorporate scramjets include the U.S.-based National Aerospace Plane (NASP),[33] and the European SpaceLiner concept.[34]

The propulsion systems for scramjet-based launch vehicles must transition between different

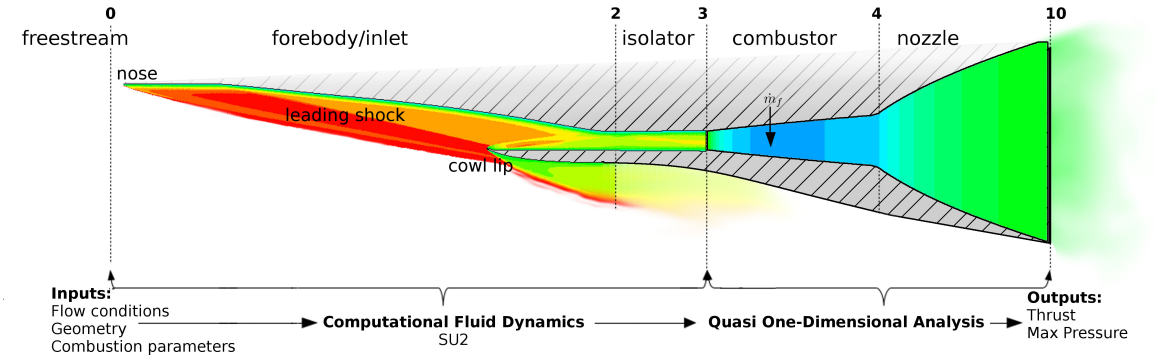


Figure 1.1: Scramjet flow path with station numbers indicated along top border of the image. Methods used to simulate each portion of the flow path are indicated along the bottom edge of the image. Contour levels are of Mach number.

propulsion techniques because scramjets only become efficient around Mach 5, and gradually lose thrust as altitude increases and the availability of atmospheric oxygen decreases. Two broad categories are identified in literature: Rocket-Based Combined-Cycle (RBCC) and Turbine-Based Combined-Cycle (TBCC). Two-Stage-To-Orbit (TSTO) system designs are more common, however Single-Stage-To-Orbit (SSTO) systems have also been considered. Conceptual design studies have been performed to evaluate the trade-offs between these systems,[13,17,18] with varying assumptions and conclusions. Hunt[15,16] details a NASA Dual-Fuel Airbreathing Hypersonic Vehicle study, including the trade-offs of hydrogen or hydrocarbon fuels: hydrogen-based fuel provides greater cooling capacity and greater range than hydrocarbon fuels for the same Mach number, however its low density leads to a larger required vehicle volume and corresponding aerodynamic drag and structural weight. This vehicle concept was intended to operate either as a Mach 10 cruise vehicle or as a Two-Stage-To-Orbit (TSTO) launch system. Scuderi et al.,[19] further analyze this concept, comparing TBCC versus RBCC and fueling variants, analyzing both the Mach 10 and space launch missions and finding that a hydrogen-fueled TBCC variant results in lower take-off gross weight for the TSTO mission. Investigations comparing staging Mach number for transition between propulsion types[14] indicate that Mach 7 may be a reasonable point to transition from first-stage to second-stage propulsion. Flight tests at or close to Mach 7 have also been conducted including HIFiRE 2,[4] HyShot 2,[20] and Hyper-X.[2] The performance at Mach 7 is therefore important to a successful design, and is the operating point chosen in this work.

Figure 1.2 illustrates the specific impulse I_{sp} versus Mach number for a variety of engines, showing both the relative efficiency of airbreathing and rocket engines as well as the decreasing efficiency of airbreathing engines as the freestream Mach number increases. Scramjet performance measures

including I_{sp} are discussed further in Section 2.2.2. This figure also plots RBCC and TBCC performances. Specific impulse can be larger for airbreathing engines because they are not carrying oxidizer tanks onboard. The trade-off is that the airbreathing engines must sufficiently compress the air entering the engine without too large a loss in efficiency. The inlet case investigated in this dissertation, and discussed in further detail in Chapter 6, is designed to operate at Mach 7 using hydrogen fuel, within the range shown in Figure 1.2 where scramjets have higher I_{sp} than ramjets.

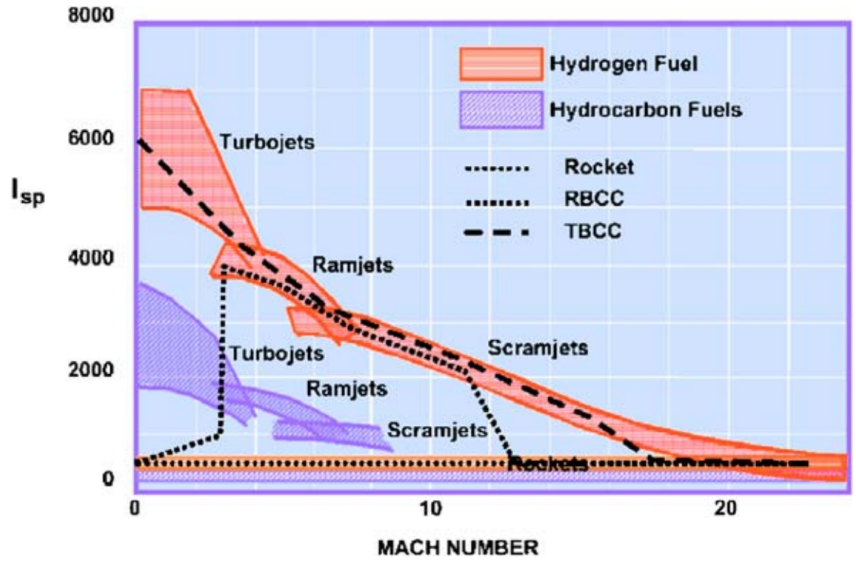


Figure 1.2: Comparison of airbreathing and rocket-based propulsion cycles and combined systems reproduced from Moses et al.[26]

Scramjet flight tests including the HyShot,[20] HIFiRE,[4,35] X-51[3] and X-43A[2] projects have had success in achieving positive thrust, while also highlighting the difficulties of designing hypersonic air-breathing engines. The challenging and multi-faceted design considerations of these engines, as well as the high expense and risk of flight testing, motivate the use of more complex and multi-fidelity objective functions in shape design. Despite the challenges, a more efficient launch system may be possible, due to the higher specific impulse from airbreathing engines. However, advanced design tools are needed to overcome the challenges of these engines.

In most literature, once the design moves past the basic conceptual stage where overall sizing and propulsion types are chosen, the components of the engine (inlet, isolator, combustor, and nozzle) are studied in isolation. This is done to simplify the problem and limit the computational expense, but higher performance can be achieved by including the entire flowpath as shown by Smart[22] in work aimed at determining reasonable compression ratios for scramjet inlets. In that work, the inlet was modeled with an empirical method based on previous inlet designs. Various design techniques for scramjet inlets will be discussed in further detail in Section 2.2.1. The methodology developed in

this dissertation is used to improve upon the performance of an inlet designed with these methods.

Idealized scramjet inlets designed in isolation result in very long engine lengths in order to achieve high efficiency, however such long lengths introduce issues with integrating the engine into the larger vehicle, structural weight and compliance, and efficient operation only at a single design point. When incorporating the engine into the larger vehicle, the inlet shape is coupled with the outer mold line of the fuselage. Active cooling may be required where fuel or another coolant is circulated next to the surfaces of the engine being heated by the friction of air. A small thrust margin where the difference of thrust and drag is small relative to the magnitude of drag, in addition to structural deformations, manufacturing tolerances, and uncertain freestream flow properties contribute to a highly sensitive design. This is just a selection of factors that lead to a need to move away from or make modifications to idealized designs, discussed further in Section 2.2.1.

There are also a number of design challenges that a scramjet engine must overcome to operate at all, much less operate efficiently. The most dramatic of these is the phenomenon of "unstart", where the shock system moves forward on the vehicle, significantly increasing drag and decreasing thrust as the flow in the combustor suddenly shifts from supersonic to subsonic flow. This phenomenon will be discussed in further detail in Section 2.1.2. It is important for scramjet designs to take the phenomena of unstart into consideration. Additional design challenges include ensuring that the simulations take into account the relevant physics that may include phenomena not usually encountered at lower speeds. These considerations will be discussed in further detail in Section 2.1.1. In some situations this adds an additional simulation burden, and the particular physical phenomena encountered can change the resulting design. A need to reduce drag results in a sharp nose for many scramjet vehicle designs. This narrow geometry introduces issues of structural compliance in a hot and high-pressure flow, where aerothermoelastic considerations may need to be taken into account.

Modern aerospace design has trended towards including multidisciplinary considerations as the designs of the engines, fuselage, lifting surfaces, and control surfaces are tightly coupled in aerospace design. This coupling is only more pronounced for scramjet vehicle designs. An external compression region labeled as "inlet/forebody" in Figure 1.1 results in designs where the outer mold line of the fuselage is coincident with the surface providing a majority of the compression required by the engine. In addition, the difference of thrust and drag (installed thrust) is small relative to the magnitude of the drag, meaning that small changes in either quantity results in failure to accelerate the vehicle.

Increasing the current capabilities to evaluate sensitivities is required to address the challenges of simultaneously meeting performance requirements, vehicle integration, and improving efficiency. The complex and multi-faceted design problem would benefit from efficiently computed sensitivities for a broader range of functions. Sensitivity analysis also contributes to the evaluation of uncertainty. Simulating a full flowpath at high fidelity is computationally expensive and unfeasible in many situations; this dissertation introduces a method of incorporating information from a multi-fidelity flowpath to the continuous adjoint method, which is a tool used for high-fidelity sensitivity

evaluation.

Based on previous airbreathing hypersonic propulsion and vehicle concept studies, a Mach 7 hydrogen-fueled scramjet geometry is investigated in this work. This operating point is important to consider for system design, since as a potential staging Mach number the vehicle would be required to produce thrust greater than drag but may be close to the operating limits of the first stage airbreathing propulsion system. The specific inlet geometry is drawn from NASA studies of Ferlemann and Gollan[25,36] at or around Mach 7 and intended to produce high-efficiency three-dimensional inlets for flight conditions similar to the HIFiRE 2^4 trajectory. The aim is to improve the performance of this inlet by developing a method to evaluate high-fidelity surface sensitivity to multi-fidelity flowpath performance. However, the methods developed in pursuit of this goal can also be applied to other problems.

The available literature on the type of inlet studied in this work, the REST-class inlet,[21,24,25,36] reveals both that these inlet have great potential as well as some remaining problems. These inlets can meet design requirements of being easily incorporated into the vehicle, approaching desired pressure ratios, and providing an elliptical combustor entrance. However, while the method of creating the inlet incorporates a boundary layer correction, the boundary layer growth predicted by CFD far exceeds the flat-plate correlation due in part to "roll-up" of the boundary layer in the corners of the inlet geometry. This leads to a discrepancy between the desired pressure ratio as well as a significant level of flow distortion at the outflow that would detrimentally effect combustor performance. In work by Gollan and Ferlemann,[25] a regression surface was created using several automatically-generated REST-class designs in order to improve on the performance of factors such as the boundary layer thickness and compression ratio. They were able to use this methodology to produce a design that improved on an objective function that combined multiple quantities of interest, however the authors noted that this objective was an arbitrary combination of a selection of quantities of interest, and not necessarily an indication of overall performance. This can be compared to work by Smart[22] that was able to incorporate the entire flowpath performance, but which did not address the precise inlet geometry and relied on an empirical fit based on the small data set of experimental results that exists for scramjet engines.

Given that a relatively small number of experimental data points are available for scramjets, that more experimental results come at a high cost, and that CFD results on these inlets are comparatively expensive due to the high degree of mesh refinement required, there is a need to provide methods that can efficiently improve on scramjet inlet designs incorporating the entire flowpath performance rather than isolated inlet performance, and requiring a smaller number of simulations. Aiming to provide this capability led to the developments within this dissertation, which are applicable to a wider range of problems beyond the scramjet inlet case addressed in this work. Scramjet design is discussed further in Section 2.2.1.

1.3 Gradient-Based Optimization

Gradient-based design optimization utilizes simulation techniques, gradient calculations, and optimization algorithms to iteratively improve upon a design. This is a subset of simulation-based design, which also includes gradient-free methods such as genetic algorithms and other gradient-free methods.[37,38] Gradient-based methods include steepest descent and sequential quadratic programming (SQP). The first SQP methodology is attributed to Wilson,[39] with further foundational work in this area by Powell[40,41] and Han,[42] and there is ample further literature on the topic.[43] The continuous adjoint is one option for evaluating the gradients needed for this design technique.

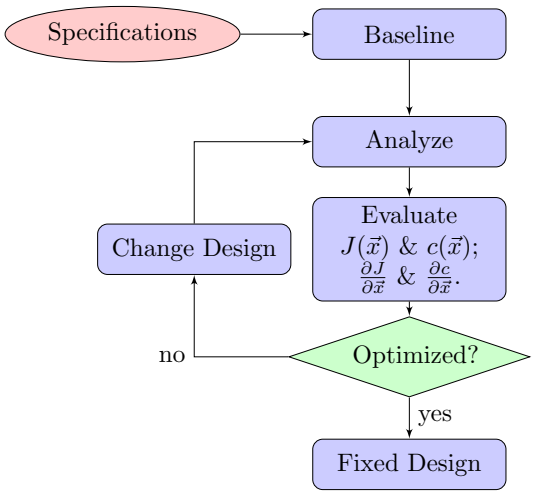


Figure 1.3: Optimization flow chart. J and c refer to the objective and constraints.

There are many optimization algorithms available, which vary the way that the design variables are altered at each step and how the gradients and constraints are applied. For this work, standard optimization algorithms are used. The developments of this dissertation focus on improvements to the gradient computation, and optimization is used as a demonstration of their application. While optimization routines will be discussed to provide background information no developments have been made in this dissertation to modify the optimization algorithms themselves.

Figure 1.3 illustrates a generic optimization process, starting from the specifications, along with the objective function J and constraints c that define the optimization problem. A baseline design is created with low fidelity or designer-intuition based methods. This baseline design is the initial point of the optimization and is usually chosen to meet the design constraints. The system is then analyzed with a mathematical model. The objective, constraints, and gradients of these values with respect to design variables $\vec{x}$ are found, and used to determine whether the design has been optimized (or if a maximum number of iterations or other criteria has been reached). If not, the design is changed by perturbing $\vec{x}$ based on gradient values, and the optimization process continues

until the exit criteria are met.

Categories of optimization frameworks include multi-disciplinary, multi-point, multi-objective, and multi-fidelity optimization. Multi-disciplinary optimization refers to the use of multiple models, each evaluating different physical systems, in order to optimize including the effects of and constraints coming from different disciplines. Multi-point optimization evaluates performance at multiple operating conditions, in contrast to single-point optimization. Multi-objective optimization refers to balancing multiple performance parameters, often through applying constraints to some of them. Multi-fidelity optimization can refer either to a system where different components are simulated at different levels of fidelity (in a way similar to using different discipline-based models), or it can refer to using a single model with different levels of fidelity as the optimization progresses. For example, one might start the optimization process at low fidelity and use high fidelity methods later on, or sampling techniques might be used to populate a response surface used for intermediate optimization steps.

In this dissertation, the term "multi-fidelity", or "multi-fidelity flowpath" refers to the use of multiple models within the evaluation of a single design. The flowpath of the engine is modeled with a combination of high-fidelity and low-fidelity simulations to compromise between computational cost and detail. This should not be confused with the practice of changing fidelity level of the overall model during the optimization process, which includes methods such as response surface generation,[44] and multi-fidelity methods to reduce uncertainty.[45] These methods are likely to be useful in design of hypersonic engines, however they will not be included in this work. In other words, a multi-fidelity flowpath model is used as the mathematical model that produces J and c. This means that J and c in the flow chart illustrated in Figure 1.3 are evaluated using a combination of models coupled together. A single-point optimization is conducted using a multi-fidelity flowpath model that incorporates multiple design considerations. Further background of optimization techniques is included in Section 2.2.3.

1.4 The Adjoint Method

There are several methods available to compute gradients. The adjoint method is a particularly efficient method due to its independence from the number of design variables and step sizes. This generates a new Partial Differential Equation (PDE) problem that uses information from the direct, or flow solution, to produce sensitivity of a function with respect to small normal deformations of the surface. This method depends on the choice of objective function, usually chosen as an integral over one of the boundaries of the computational domain. The details of this method will be discussed further in Section 2.3.3. Further details of this method are presented in Section 3.1, and modifications to this method introduced with this work are detailed in Section 3.2 through Section 3.4.

There are two main categories of the adjoint method: continuous, and discrete. The continuous and discrete adjoint methods provide surface sensitivities at the approximate cost of a single additional flow solution, as compared to the finite difference method that perturbs each design variable and re-evaluates the objective. The continuous adjoint method solves the discretization of the linearized fluid problem, while the discrete adjoint method solves the linearization of the discretized fluid problem. In other words, the continuous adjoint method solves a problem defined by careful implementation of a PDE defined by taking the linearization of the original continuous fluid problem. The discrete adjoint method solves a problem defined by taking the adjoint of the already-discretized fluid problem. A flowchart comparing the process to arrive at sensitivities via these two methodologies is illustrated in Figure 1.4. Another way of phrasing the difference between the continuous and discrete adjoint is that the continuous adjoint method finds the inexact derivative of an exact functional while the discrete adjoint method finds the exact derivative of an inexact functional. Nadarajah and Jameson[46] thoroughly discuss the trade-offs between continuous and discrete methods. As the mesh spacing approaches zero, the discrete and continuous formulations approach one another in terms of the mathematical expressions, and in practice as the mesh size increases the results become more accurate relative to finite difference gradients and each other. Since the discrete adjoint method is the exact derivative of the inexact functional, it has generally better agreement with finite difference gradients. The cost of deriving the discrete adjoint has been considered greater,[46] however it has been made significantly more accessible in recent years thanks to the advent of automatic differentiation (AD).[47]

The adjoint method can be traced back to the application of control theory to PDE problems by Lions[48] and Pironneau.[49,50] The continuous adjoint method was first developed for aerodynamic optimization by Jameson,[51] and since then much work has been presented in the literature in the development of solution methods, applications, and derivations to include additional objectives and physical phenomena. Further work in the continuous adjoint method by Reuther et al.[52] extended this method to multi-block codes and full aircraft simulations, and Jameson & Martinelli[53] provide further work in adjoint-based optimization of transonic aircraft. Giles & Pierce[54] derived the adjoint equations for viscous compressible flow and examined solution behavior, showed that the adjoint variables have a logarithmic singularity at sonic points, and that the adjoint variables are continuous across shocks. Castro[55] et al. developed an adjoint method for unstructured grids, which expands the grid-generation choices. Papadimitriou & Giannakoglou[56] derived the adjoint for a total pressure objective function at an outlet. Othmer[57] developed the continuous adjoint for a function of velocity in an incompressible flow, for topology optimization of duct flow. Hayashi[58] et al. detail characteristic-based boundary conditions for open boundaries of the adjoint for surface-based functionals. Palacios et al.[59] implemented many of these capabilities in an open-source CFD solver, SU2, which facilitated further development and is used in this work. Working in the SU2 framework, Economon[60] developed an adjoint formulation for unsteady problems, and Copeland[61]

developed an adjoint formulation for heat flux objectives in chemically reacting non-equilibrium flow. Arian & Salas[62] have also explored expanding the limits of what objectives can be addressed by the continuous adjoint method, focusing on solid wall boundaries.

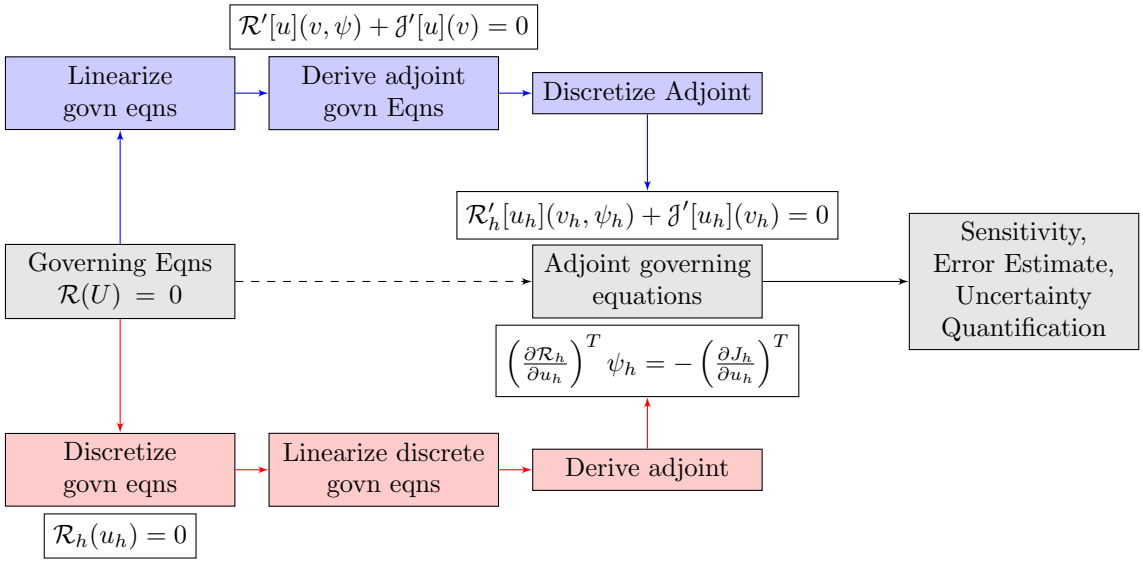


Figure 1.4: Continuous vs Discrete Adjoint Formulations

The open-source multiphysics simulation suite SU2[63] is used and modified in this dissertation, and includes both continuous and discrete formulations for the adjoint method. The discrete adjoint method as implemented in SU2 utilizes automatic differentiation of the initial CFD code to find an exact derivative of the inexact functional. More detail about AD methods is available in work by Gauger et al.[64] and Zhou et al.[65] This method has some advantages in obtaining results that are generally much closer to finite difference results, however there are greater memory requirements due to the larger amount of information needed during calculation. The continuous adjoint method, by contrast, is sometimes more removed from finite difference and discrete adjoint result, however it benefits from lower memory usage and, particularly relevant to this dissertation, the boundary conditions are relatively accessible within the implementation. An additional benefit of the continuous adjoint method is the possibility of exploiting numerical methods to improve convergence or address numerical instabilities in ways that are not available to the discrete adjoint method.

Mader et al.[47] and Albring et al.[66] have worked at making the discrete adjoint method more efficient and expanding its capabilities. The work contained in this dissertation approaches the problem from the other direction - by making the boundary conditions of the continuous adjoint method more general such that the numerical efficiency of the continuous adjoint method can be utilized with a reduced limitation on the objective function. Discrete adjoints via the automatic differentiation (AD) approach to entire tool chains has been studied, for example work by Gauger

et al.[64] that applied this method to a transonic airfoil. By contrast, in this work the adjoint method is applied to only one component, allowing a more flexible range of models to be used while exercising the benefits of the continuous adjoint method. The discrete adjoint is a useful tool, and it may be possible to apply a similar methodology as presented in this dissertation to the discrete adjoint, however that task requires significant further development and is outside of the scope of this dissertation.

1.4.1 Estimated Computational and Monetary Benefit

The benefits of, and need for, these developments can be shown through examining the number of PDE solutions required as compared to the finite difference method. In this section, the term "adjoint" is used to refer to both the discrete and continuous forms, which would require the same number of PDE solutions. The trade-offs between the discrete and continuous adjoint are described in detail in Section 2.3.3. The finite difference method is less efficient as it requires a much larger number of function evaluations as the number of design variables grow, however it has the benefit of not requiring significant further implementation to be applied to new functionals. The developments in this work decrease the amount of work required to apply the continuous adjoint to new functionals, making this more efficient method more accessible. The discrete adjoint for combinations of functionals was also implemented in this work, however the generalized outflow functional developed in this work is currently only available with the continuous adjoint.

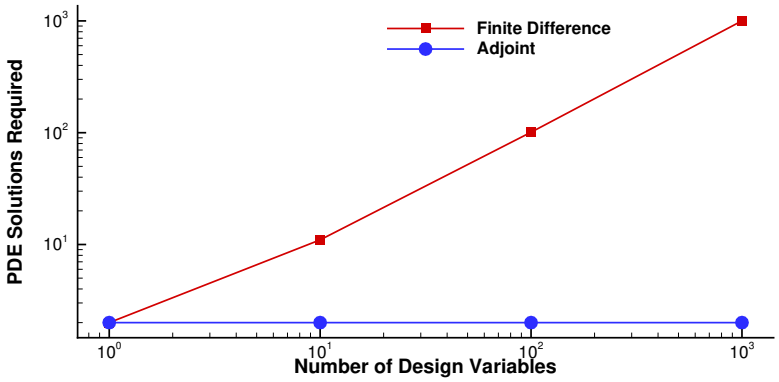


Figure 1.5: PDE solutions required for gradient calculation relative to number of design variables, including both direct and adjoint solutions.

Figure 1.5 compares the number of PDE solutions required to compute the gradient of a single function. The adjoint method, whether using the discrete or continuous form, projects surface sensitivity onto the design variables, and so has a cost independent of the number of design variables. Figure 1.6 compares the number of PDE solutions expected to complete an optimization

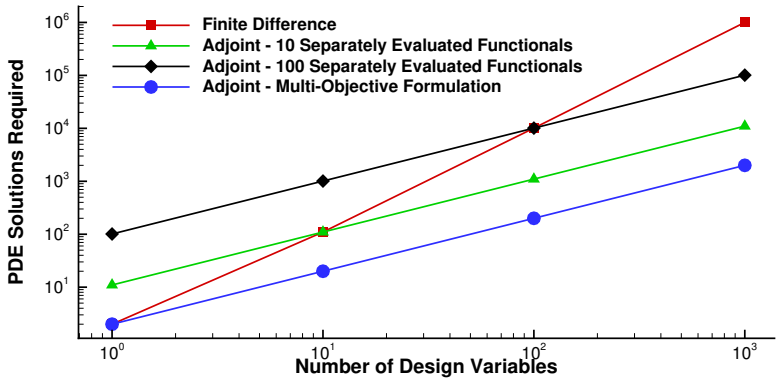


Figure 1.6: Number of PDE solutions required for optimization, assuming optimizer iterations ≈ number of design variables and no line searches.

problem. This plot also includes the number of PDE solutions required if each functional composing a multi-objective problem requires a separate adjoint formulation, in order to illustrate the benefit of including a multi-objective adjoint formulation. In these plots, the number of PDE solutions required is estimated as the number of design variables times the evaluations required for a gradient evaluation. This assumes that the gradient evaluation is able to re-use the solution produced in the evaluation of the functional, and that no line-searches requiring additional function evaluations are needed.

A realistic industry design problem may require hundreds if not thousands of design variables, and the number of major optimizer iterations required is generally proportional to the number of design variables.[67] Using a mere 100 design variables as an example, the evaluation of the gradient requires 99 fewer PDE solutions when using the adjoint. In the optimization problem with the same number of variables, 9,900 fewer PDE solutions are estimated to be needed. These savings are decreased if a multi-objective problem is addressed where the adjoint needs to be evaluated separately for each component of the objective. Estimating the cost at $0.10 per CPU hour and assuming 5,000 CPU hours per PDE evaluation leads to an estimate of saving $49,500 per gradient evaluation, or $4.95 million per optimization, for a problem with 100 design variables when using the adjoint compared to the finite difference, and assuming no line searches (or a single gradient evaluation per optimizer step). In this dissertation, these dramatic savings in computational and monetary cost are made available to a wider range and complexity of functionals than previously possible. This facilitates efficient simulation-based design for more realistic problems that may depend on multiple quantities of interest and use multiple interconnected simulations.

1.5 Dissertation Layout

This dissertation will proceed with a discussion of the relevant physical phenomena and existing techniques used for scramjet design and gradient-based optimization in Chapter 2. Following this background information, the details of the continuous adjoint method for a generalized outflow-based functional and other contributions summarized in Section 1.1 are described and derived in Chapter 3. Implementation details are included in Chapter 4, and verification results in Chapter 5. Chapter 6 details the specific design problem addressed and the results of various optimization studies that utilize the methodology developed in this dissertation. Conclusions and final remarks are included in Chapter 7.

Chapter 2

Relevant Physical Phenomena & Existing Methodology

This chapter will begin with a discussion of the physical phenomena relevant to hypersonic flow in Section 2.1, followed by a discussion of the current techniques used to design scramjet inlets in Section 2.2.1 and of optimization methods in Section 2.2.3. Section 2.3 discusses the methods used to model the set of phenomena that are included within the scope of this work and the adjoint method. More detailed discussion and mathematical formulation of the continuous adjoint method is reserved for Chapter 3. This chapter will close with a summary of the various challenges that are inherent to the scramjet engine design problem or arise from the limitations of existing methodologies highlighted in the individual sections.

2.1 Hypersonic Flow

Hypersonic flow describes a wide range of flow conditions and physical phenomena, not all of which will be relevant to this work. Section 2.1.1 will discuss hypersonic flow in general, summarizing the phenomena associated with hypersonic flow. Table 2.1 provides a convenient reference for the various flow phenomena that could be relevant to hypersonic flow, and the subset that are relevant to the particular cases in this work. Unstart is a phenomenon that deserves its detailed discussion, included in Section 2.1.2, although unstart is included only through approximate methods in the results presented in Chapter 6.

2.1.1 Flow Phenomena

When the governing equations of fluid flow are non-dimensionalized, the Mach number ($M = \frac{|\vec{v}|}{c}$) and Reynolds number ($Re = \frac{\rho u_\infty D}{\mu}$) appear as non-dimensional coefficients. These governing equations,

the Navier Stokes equations, are discussed in more detail in Section 2.3.2. These non-dimensional values can be used to divide fluid flow into regimes that categorize the fluid by what phenomena dominate its behavior, and in some cases similarity solutions are possible. The Mach number relates the flow velocity to the speed of sound, which can also be described as the rate that information travels through the fluid. The flow regimes defined by the Mach number can be described as subsonic ($M < 1$), transonic ($M \approx 1$), supersonic ($M > 1$), and hypersonic ($M \gg 1$).

There is a clear physical limit that defines the transition from subsonic to supersonic flow - once past the speed of sound, the fluid is moving faster than the speed at which pressure waves travel through that fluid. The transition from supersonic to hypersonic, by contrast, is not defined as a single discrete change in Mach number. Instead, it is characterized by the gradually increasing importance of various phenomena that are either not present or not significant at lower speeds. The significance of these phenomena is determined by a combination of the Mach number, temperature, and density of the flow around a vehicle. A general rule of thumb is that hypersonic flow begins around Mach 5, which also happens to be the Mach number around which scramjet propulsion is more efficient than ramjets due to the costs of deccelerating air outweighing the difficulties of combustion at supersonic speeds. Table 2.1 summarizes the major phenomena associated with hypersonic flow, and notes which phenomena are relevant and/or included in the analyses of this work. The following paragraphs will elaborate on the details of hypersonic phenomena.

Oblique shocks are discontinuities that occur in supersonic flow when it is turned at a high enough angle for the characteristics, or Mach lines, intersect. In other words, a discontinuity that forms when a body in the flow forces the fluid to change direction faster than can be achieved continuously. As the Mach number increases, the angle of the discontinuity relative to the solid body decreases, bringing this discontinuity very close to the physical body. The area between the shock and the body is referred to as the "shock layer". Depending on the Reynolds number, the shock layer may be so thin as to overlap with and interact heavily with the viscous boundary layer. The thin shock layer is one aspect of hypersonic flow that becomes more significant as the Mach number increases.

When flow passes through a shock, there is a sharp increase in entropy. When a blunt body is in supersonic flow, a bow shock forms such that the entropy change very close to the nose is very high, while further away from the body downstream the shock angle is higher and has a lower entropy change. This results in high entropy gradients within the flow, effecting the growth of boundary layers. The "entropy layer" is another phenomenon that occurs in hypersonic flow. The interaction of the entropy layer and the boundary layer is called the "vorticity interaction" due to the strong vorticity. This complicates the analysis of boundary layers as it is no longer clear how to define the outer edge of the boundary layer.

The Reynolds number is the ratio of the inertial to viscous forces, and its value is correlated with the behavior of the boundary layer, a region close to solid walls where velocity gradients form due to viscous effects. The behavior of boundary layers change as the Mach number of the flow

Table 2.1: Summary of Hypersonic Phenomena.

Phenomenon	Inclusion in this work	Effect	Limits/conditions when significant
Shock Layer	**Included** through Reynolds Averaged Navier Stokes (RANS)	Oblique shock lies close to the solid body. May interact with boundary layer at lower Reynolds numbers.	Increasingly significant as Mach increases.
Entropy Layer	**Included** through RANS	Large entropy gradients interact with boundary layers.	Near blunt edges/noses and regions with interacting shocks.
Viscous Interaction	**Included** through RANS. Viscosity changes **neglected** in adjoint formulation.	Interactions with high temperature that increase the viscosity and decrease the density to make boundary layers grow much faster.	Low(er) Reynolds Number.
Convective Heat Flux	**Included** in RANS and quasi 1-D models with an **assumed isothermal** wall.	Dominates design of hypersonic vehicles by structural effects, and effect on boundary layers.	Any hypersonic flow.
Real Gas Effects	**Neglected** in CFD, **included** in quasi 1-D flow.	Non-constant specific heats.	$T > 800°K$.
Chemically reacting flow	**Neglected**: small regions of $T > 2000°K$ exist in CFD, likely with **insignificant** effect. Chemical equilibrium assumed in Quasi 1-D model.	Changing composition of gases, non-equilibrium flow, non-ideal flow.	Near ablating surfaces, and/or when temperature high enough: O2 dissociation begins around $T > 2000°K - 4000°K$, and N2 dissociation begins around $T > 9000°K$.
Ionized flow	**Not significant**: temperature too low	May encounter plasma effects, communication black-out.	$T > 9000°K$.
Radiative Heat Flux	**Not significant**: temperature too low	Effects bow shock behavior, gas temperature, viscous effects.	$T > 10,000°K$.
Rarefied gas dynamics	**Not significant**: $Kn \ll 1$	velocity-slip, temperature-slip conditions. As density decreases further, non-continuum (ie, kinetic theory and the free molecular regime)	Knudsen number $\mathcal{O}(1)$, generally at altitudes > 92 km.

increases. Viscous dissipation transforms the kinetic energy of the freestream flow into internal energy, increasing the temperature of the gas. In hypersonic flow, the effects of this temperature increase begin to dominate the boundary layer characteristics. As the temperature increases, the density of the gas decreases, forcing the boundary layer to be larger in order to accommodate the required mas flow rate. The viscosity of the gas increases with the temperature, which also acts to increase the thickness of the boundary layer. These two effects together make hypersonic boundary layers grow more rapidly than at lower speeds. This behavior is described as the "viscous interaction". When the Reynolds number is low enough that the boundary layer intersects the shock layer, the shock layer must be treated as viscous. In hypersonic flows, the boundary layer thickness approaches a limit dependent only on the Mach and Reynolds number:

$$\delta \propto \frac{M_\infty^2}{\sqrt{Re_x}}. \tag{2.1}$$

High temperature flow is another phenomenon that becomes more significant in hypersonic flow. There are several effects associated with high temperature, which are grouped into "high-temperature effects". When the temperature increases enough to cause dissociation of the gas, the boundary layer becomes chemically reacting. Ablation of the vehicle surface also creates a chemically reacting layer. Ablation is the erosion of the vehicle surface whether by friction forces or by chemical reactions with the heat shield material. The effects of chemically reacting gas may extend throughout the shock layer. This affects the specific heats of the flow, and the gas no longer follows the "ideal" relations that are used for lower-speed flow. The effect of non-constant specific heat is held to be significant above a temperature of around $800°K$. At 1 atm, oxygen begins to dissociate around $2000°K$, and is totally dissociated around $4000°K$. gaseous Nitrogen dissociates around $9000°K$, and ions begin to form. When the gas begins to be ionized, plasma effects may be important, depending on charge concentrations, temperature, and density of the fluid. The effects of chemical changes and specific heats are sometimes also referred to as "real gas effects". These effects change the behavior of the fluid around the vehicle. The most significant high-temperature effect in terms of vehicle design, however, is the heat transfer into to surface of the vehicle, which becomes significant well before chemical changes to the gas come into play. Aerodynamic heating dominates the design of hypersonic vehicles. This comes in the form of convective heating due to the contact of the fluid against the surface of the vehicle, and at high enough temperatures radiative heating may also be significant. At reentry speeds, the high temperature effects cause "communications blackout" due to the ionization of gas that electromagnetically shields the vehicle from radio frequencies.

Hypersonic vehicles designed to fly at high altitudes must also contend with rarefied gas dynamics where the density of the flow is low enough to significantly change the way that the flow must be modeled. These effects are influenced by the mean free path, λ, which is defined by the mean distance a molecule travels before impacting another molecule. When λ is small relative to the size of fluid

property gradients (represented by the Knudsen number, $Kn = \lambda/L$), the flow can be treated as a continuum. As λ increases with decreased density, the assumption of continuum flow becomes less valid. As the Knudsen number approaches 1 at low densities (92 km, 300,000 ft) although the bulk flow can be treated as continuum, the flow at the surface of the vehicle can no longer be treated the same way as at higher densities. Rather than a "no-slip" condition where the fluid velocity goes to zero at the solid wall, there is a "velocity-slip condition". The temperature at the wall is also no longer equal to the temperature of the wall, called the "temperature slip condition". As the altitude continues to increase, and the flow is no longer treated as a continuum, kinetic theory must be used in the "free molecular regime".

Hypersonic flow introduces a challenge of determining whether and how to simulate each of the phenomena discussed in this section. This choice depends not just on the freestream flow conditions but also on the vehicle geometry, which determines whether high-temperature effects are relevant. What values are desired at a higher accuracy also determine what type of simulation is required. The computational cost and simulation complexity may be counted as an additional challenge to be addressed. Several effects that become significant in hypersonic flow have been described in this section. However, not all of them are significant in this work, and not all of them will be modeled. Whether each of the effects described in this section were included in the analyses used in this work is summarized in the second column of Table 2.1.

Scramjet engines at current levels of technology operate in the region of Mach 5 - Mach 10. Tests of these engines have occurred around 30 km altitudes, and as the engine requires oxygen from the atmosphere to operate, there is an upper limit to the altitude at which these engines can operate. Under these conditions, temperatures are not high enough to expect significant radiative heat flux, and temperatures above which real gas effects would be expected only occur in some areas of the flowpath. The Knudsen number is low enough to assume continuum flow, and viscous effects are expected to be significant. For these reasons, Reynolds-Averaged Navier-Stokes flow is used for the flow over the inlet. A one-dimensional channel flow model is used in the combustor, where changing specific heats are included due to the higher temperature occurring there. Section 6.2.1 includes a discussion of whether these assumptions are appropriate based on the detailed flow solutions computed.

For greater accuracy Large Eddy Simulation (LES) or Direct Numerical Simulation (DNS) could be used, which would more accurately resolve turbulence. Unsteady effects are also neglected. These tools exist, however they require greater computational resources, and are outside the scope of this work. It will likely be possible to use these more advanced methods for design, rather than just for analysis, however at the present time this is out of reach due to the computational expense. In this work, RANS simulations are used, as it is sufficient to model steady flow in many circumstances, although more accurate modeling of turbulent effects is desirable.

Table 2.2: Parameters related to unstart

Metric	Description	Relevant Equation(s)
Rayleigh limit	Thermodynamically choked channel, leading to subsonic flow and unstart.	Equation 2.3
Korkegi limit[73]	Flow separation occurs in boundary layer, leading to unstart.	Equation 2.4
Kantrowitz limit[74]	Minimum area ratio that can "self-start".	Equation 2.5
Isentropic compression	Maximum compression possible for isentropic flow; idealized maximum compression.	Equation 2.6

2.1.2 Unstart

Unstart is a phenomenon experienced by supersonic engines that can cause catastrophic failure. A "started" flow is one that has achieved supersonic speeds throughout a channel, and is associated with the maximum mass flow rate through the channel under given inflow conditions. A flow is "choked" when the flow is sonic at the throat or minimum area of the channel. An "unstart" occurs when a started flow transitions suddenly to subsonic, when a pressure disturbance downstream forces a shock to travel upstream, through the throat of the channel, and finally moving the shock system forward of the inlet. This results in subsonic flow and reduced mass flow through the engine. The thrust of an engine is reduced, and the new shock structure creates additional drag through increased pressure on the exterior surfaces and through the deceleration of a greater mass of air across the shocks, in the worst case across a detached bow shock.

Studies of unstart included experimental work investigating bleeding and moveable geometry as methods to address unstart[68] as well as the control systems to restart.[69] Waltrup and Billig[70,71] conducted extensive work investigating the interactions between inlets and combustors, and conducted much early work in developing isolator geometries. An isolator is a channel separating the inlet from the combustor, which serves to introduce a pressure rise that prevents disturbances from the combustor from traveling upstream and causing an unstart. Although theoretical work has since been conducted to predict unstart and its behavior, empirical methods based on experimental results[72] are still among the methods commonly used.

Predicting unstart is a difficult and computationally intensive task because it depends not only on the freestream conditions and engine geometry, but also on hysteresis, turbulent effects, and unsteady flow phenomena. The result of unstart simulations is often not a determination of whether or not the engine will unstart, but rather as a probability of unstart. Theoretic understanding of the physical processes that control or lead to unstart as well as the ability to simulate this process is currently an open area of research with much further work. Some of the parameters or limits that are used to describe unstart are summarized in Table 2.2, and will be described in further detail in the following paragraphs.

Jang, Nichols & Moin[75] detail a stability analysis of unstart. There are two categories of unstart,

referred to as Rayleigh and Korkegi limits. The Rayleigh limit is defined as the point where the bulk flow reaches the sonic condition due to thermal choking. In other words, heat addition to the flow causes deceleration leading to a sonic condition. At the point where a single point in the flow (the throat) reaches the sonic condition, two possible solutions exist: either the flow will return smoothly to supersonic, or the flow will continue to decelerate to subsonic flow. This is a steady phenomenon, and can be predicted with one-dimensional flow. The Korkegi limit[73] on the other hand is by nature an unsteady phenomenon: when regions of separated flow form high pressure downstream is able to push the flow upstream despite the bulk flow being supersonic. Predictions of this limit require detailed unsteady analysis.

To prevent unstart via the Rayleigh limit, the minimum Mach number or the total temperature ratio of the flow path can be taken into account, in either case increasing the margin to thermal choking. The total temperature ratio at which the flow would become sonic can be found starting from the differential equation describing Rayleigh flow, which is non-adiabatic flow through a frictionless constant-area channel with no mass addition:

$$\frac{dM^2}{M^2} = \frac{1+\gamma M^2}{1-M^2}\left(1+\frac{\gamma-1}{2}M^2\right)\frac{dT_t}{T_t}\,. \tag{2.2}$$

Integrating between the initial Mach number M and 1 results in the following relationship for the relative total temperature at the sonic condition.

$$\frac{T_t}{T_{t,M=1}} = \frac{2(\gamma+1)M^2}{(1+\gamma M^2)^2}\left(1+\frac{\gamma-1}{2}M^2\right). \tag{2.3}$$

Equation 2.3 provides an estimate for the total temperature ratio at thermal choking, which could be used to inform reasonable constraints to provide a safety margin to unstart. As this equation was found by assuming constant area, frictionless walls, and no mass addition, the actual total temperature ratio where the flow chokes will differ in a real engine, or in an engine model that takes these other effects into account.

Preventing unstart via the Korkegi limit requires introducing a pressure rise through the isolator that increases the pressure disturbance from the combustor that would cause unstart. In this work, the Rayleigh limit is taken into account within the combustor analysis as the one-dimensional flow model is able to predict the thermal choking point. The script has been written to always take the subsonic branch, resulting in low thrust values such that this condition is avoided in the optimization process. This limit is also taken into account by applying a penalty to the total temperature ratio, which is associated with increasing the margin to the Rayleigh limit.

The Korkegi limit is more difficult to account for as it requires an unsteady simulation. However, empirical correlations exist that can be exploited here. An engineering solution developed to reduce the risk of unstart is a constant-area duct that serves to isolate the inlet from the combustor, referred to as the isolator. It acts to increase the static pressure entering the combustor through a

series of reflected oblique shocks. These shocks increase the static pressure at the end of the duct, reducing the ability of a pressure rise in the combustor to travel upstream and unstart the inlet. Through experimentation, empirical correlations have been developed by Waltrup & Billig[70] as well as Sullins[72] to determine the minimum length of an isolator given the back pressure applied at the end of the constant-area channel.

Sullins[72] provides a correlation for the pressure rise through rectangular ducts, modified here for notation consistent with the station numbers in Figure 1.1:

$$\frac{(M_2^2-1)(S/H)Re_\theta^{0.3}}{\sqrt{\theta/H}} = 50\left(\frac{P}{P_2}-1\right)+170\left(\frac{P}{P_2}-1\right)^2. \tag{2.4}$$

In Equation 2.4, S is the length along the isolator from station 2, H is the duct height, $\theta = \int \frac{\rho(y)u(y)}{\rho_{ref}u_{ref}}\left(1-\frac{u(y)}{u_{ref}}\right)dy$ is the momentum thickness, and Re_θ is the Reynolds number based on θ. Given quantities at station 2, Equation 2.4 predicts the pressure rise that would occur over a back-pressured isolator of a given length. The quantities in this correlation were not used directly in the objective functions, and isolators were assumed to be un-back-pressured, however this correlation can be used to check the designs for feasibility.

A related metric of scramjet inlet performance is self-starting capability. This means that the minimum area throat has a Mach number greater than 1 even when a normal shock exists ahead of the inlet, and so the conditions downstream of the throat will be supersonic regardless of the upstream shock structure. This is contrasted with an inlet that has sonic conditions at the throat, which will have either subsonic or supersonic flow downstream depending on the geometry and backpressure. In order to be self-started, the contraction ratio must be greater than the ratio that would result in the sonic condition. This can be estimated with the Kantrowitz limit.[74] A normal shock wave is assumed at the beginning of the internal compression section, decelerating the air from freestream. The area that would produce sonic flow is then calculated assuming a one-dimensional isentropic flow and a perfect gas;

$$\left(\frac{\mathcal{A}_2}{\mathcal{A}_4}\right) = \frac{1}{M_2}\left(\frac{(\gamma+1)M_2^2}{(\gamma-1)M_2^2+2}\right)^{\frac{\gamma}{\gamma-1}}\left(\frac{\gamma+1}{2\gamma M_2^2-(\gamma-1)}\right)^{\frac{1}{\gamma-1}}\left(\frac{1+\frac{\gamma-1}{2}M_2^2}{\frac{\gamma+1}{2}}\right)^{\frac{\gamma+1}{2(\gamma-1)}}. \tag{2.5}$$

The maximum isentropic contraction ratio can also be found by solving for the area associated with decelerating the flow isentropically to the sonic condition:

$$\left(\frac{\mathcal{A}_4}{\mathcal{A}_0}\right) = M_0\left(\frac{\gamma+1}{2}\right)^{\frac{\gamma+1}{2(\gamma-1)}}\left(1+\frac{\gamma-1}{2}M_0^2\right)^{\frac{\gamma+1}{2(\gamma-1)}}. \tag{2.6}$$

The ratio from Equation 2.6 represents the largest contraction ratio possible while maintaining choked flow. A larger contraction ratio will result in unchoked flow and decreased mass flow rate even under the most ideal isentropic and inviscid conditions.

The Kantrowitz limit is a more conservative limit, which may not be possible to meet under some conditions or other design requirements that lead to choosing an inlet that is not self-starting and that therefore has some risk of un-starting. In some cases the Kantrowitz limit can be exceeded with bleeding holes and bypasses that facilitate starting the inlet. As the freestream Mach number increases, it becomes less feasible to require a self-starting inlet, both because the limited contraction ratio may not be able to provide the necessary compression, and because the normal shock at the internal section is less likely to occur. The isentropic limit from Equation 2.6, on the other hand, is a stricter limit: a channel that exceeds this limit will not only fail to self-start, but will be choked and un-started.

Unstart is a complex phenomenon with multiple causes whose prediction and simulation is still an open area of research. Within this work, consideration of unstart is limited to quantities that are correlated with reducing the risk of encountering unstart, and detailed prediction or modeling of this phenomenon is outside the scope of this work. An comparison of designs with respect to Equation 2.4 is included in Section 6.2.3, Table 6.3.

2.2 Design Techniques & Optimization

This section describes existing techniques for designing scramjet inlets, quantities of interest, and optimization methods. The design techniques described in Section 2.2.1 produce the initial design point used in optimization studies in Chapter 6, which use the methods described in Section 2.2.3 to improve a selection of the quantities discussed in Section 2.2.2.

2.2.1 Existing Scramjet Design Techniques

Scramjet engines and a review of literature related to their design and operation was presented in Section 1.2. This section will review existing methods of scramjet inlet design and scramjet flowpath design to provide context for the new methodologies we have developed in this work and for the optimization studies conducted. Engine components are often designed without knowledge of the full flowpath performance, however the operation of the entire engine cycle can be included at the conceptual design level. Work by Smart[22] has examined desirable inlet compression ratios through analysis of the full flowpath utilizing a mix of one-dimensional and empirical methods. This method yields information about what properties an inlet should have considering full flowpath performance, however details of the geometry that would produce that performance and is subject to the limitations of using empirical models to predict inlet performance.

There are trade-offs between inlet efficiency and the performance of the full engine cycle. If the compression of the inlet is not sufficient, the combustion process will not operate well, and too much compression will challenge the structural limits of the vehicle as well as requiring flow bleed or variable geometry to avoid unstart.

The two main categories of hypersonic inlets are "two-dimensional" rectangular cross-sectioned designs based on straight ramps (see Figure 2.2) that benefit from flow and manufacturing simplicity, and "three-dimensional" or "inward-turning" designs based on streamline tracing methodologies. Scramjets with rectangular inlets have included the X-43[2] and related NASA conceptual access to space design studies,[15] as well as HIFiRE flight 2[4] and the X-51[3] program. There are benefits from a simplicity of design and integration, however three-dimensional "inward-turning" inlets are thought to have higher efficiency. The HYCAUSE[27] program tested both inward-turning and two-dimensional inlets. The inward-turning inlets are expected to result in advantages for vehicle integration, reduced structural weight and reduced cooled surface area.

Billig[76] introduced streamline tracing methodologies, which start from an ideal desired flowfield and trace a cross section through that flow to arrive at a three-dimensional inlet shape. When the cross-section is circular, these inlets are similar to the Busemann inlet.[77] An example is shown in Figure 2.3. A Busemann inlet[77] is illustrated in Figure 2.1, which uses Mach waves to design a surface that isentropically compresses the flow up to a conical shock that returns the flow to horizontal. These inlets have theoretically high performance, but they neglect viscous effects and often generate long inlet lengths and complex geometries that are sensitive to small changes in flow conditions.

Further work by Smart[24] built on streamline tracing techniques to develop Rectangular-to-Elliptical-Shape-Transition (REST) inlets that trace multiple cross-sections through the desired flow field and blend them to arrive at designs with rectangular entrances and elliptical exits for ease of integration with a flat underside of a vehicle and an elliptical cross-sectioned combustion chamber. A pictoral representation is shown in Figure 2.4. These REST inlets include a boundary-layer correction, however analysis of the flow using CFD reveals that there is significant boundary-layer growth[36] and the inlets may not achieve the desired performance. This motivates the use of high fidelity optimization where those complex interactions can be taken into account in the design process. Due to the high computational cost of gradient-based design optimization, an inverse design should be used as the initial point to reduce the number of iterations required by the optimizer.

2.2.2 Quantities of Interest

Design challenges relevant to hypersonic vehicles include shape uncertainty, the risk of unstart that causes a sudden and sometimes catastrophic loss of thrust, and the requirement to operate at a wide range of conditions as the vehicle accelerates to an altitude where rocket propulsion takes over. The aerodynamic performance, heat flux and associated Thermal Protective System (TPS) thickness, combustion, and mixing are all also relevant factors to consider in the design of a scramjet and its inlet. There are several quantities that could be chosen for the objective function, some of which are in conflict, and the choice of objective will strongly determine the outcome. Heiser & Pratt[80] provide much further detail on hypersonic air-breathing design and performance. When the inlet is

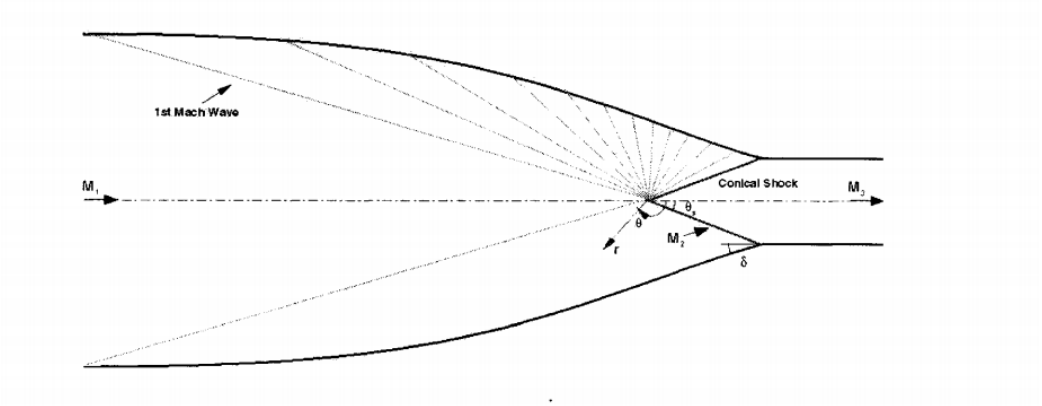


Figure 2.1: Flowfield of a full Busemann inlet.[78]

considered in isolation from the rest of the cycle, the *kinetic energy efficiency*,

$$\eta_{KE} = \frac{h_{t2} - h'_2}{h_{t0} - h_0} = \frac{u'^2_2}{u^2_0}\,, \tag{2.7}$$

or the *total pressure recovery*,

$$P_{tr} = P_{t2}/P_{t0}\,, \tag{2.8}$$

are often used. η_{KE} compares kinetic energy that the gas would have if it were expanded isentropically, to the kinetic energy of the freestream flow. P_{tr} indicates how much work the gas would do if expanded isentropically, relative to the freestream. While these measures rely only on the inlet, they both assume that the flow will behave isentropically downstream of the inlet. This is generally not the case, especially for a scramjet engine where a shocks, boundary layers, and heat transfer are expected to be significant. Therefore, it may be beneficial to consider the engine cycle performance during the optimization process.

When the behavior of the combustor and nozzle are known or assumed, the stream thrust can be evaluated:

$$\mathcal{F}_{un} = \dot{m}_0 c_0 M_0 \left((1+f)\frac{M_{10}}{M_0}\sqrt{\frac{T_{10}}{T_0}} - 1 \right) + \frac{\mathcal{A}_{10}}{\mathcal{A}_0}\left(\frac{P_{10}}{P_0} - 1 \right). \tag{2.9}$$

In this equation M is the Mach number, P is the static pressure, T is the static temperature, f is the mass flow fraction of fuel:air, c indicates the speed of sound, $\mathcal{A}$ is the cross-sectional area, and $\dot{m}$ is the mass flow rate. The quantities with subscript 0 are the freestream values, and the quantities with subscript 10 are values at the exit of the nozzle or expansion ramp, as shown in Figure 1.1.

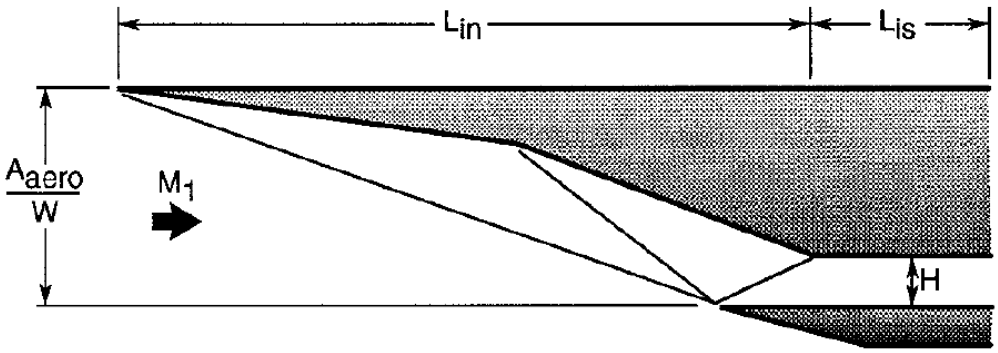


Figure 2.2: Two dimensional inlet (rectangular cross-section).[24]

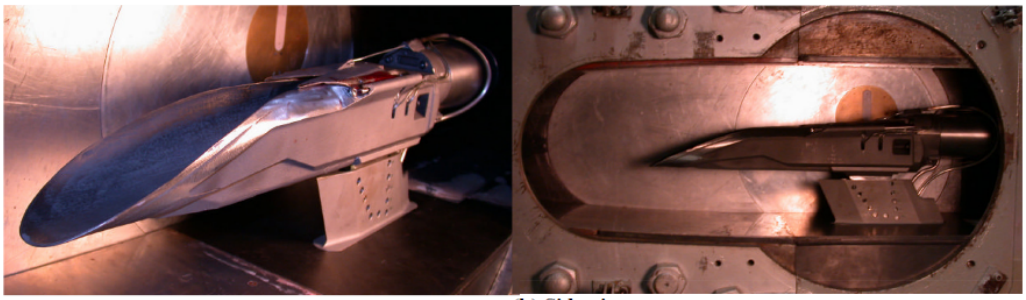


Figure 2.3: "Sugar-Scoop" design in Virginia Tech Wind Tunnel.[79]

$\mathcal{F}_{un}$ is a component of other performance measures, the *specific impulse* and *overall cycle efficiency*:

$$I_{sp} = \frac{\mathcal{F}_{un}}{\dot{m}_f g_0} \tag{2.10}$$

$$\eta_o = \frac{\mathcal{F}_{un} V_0}{\dot{m}_f h_{pr}}. \tag{2.11}$$

I_{sp} compares the stream thrust to the force of gravity acting on the fuel mass flow rate (the *gravimetric fuel rate*). η_o compares the rate of work done by the stream thrust to the energy contained in the fuel.

The fuel fraction f and fuel products enthalpy h_{pr} will be assumed constant, as are the freestream velocity and gravity constant, and so a term $\frac{\mathcal{F}_{un}}{\dot{m}_0}$, the *specific thrust* will be used that is directly

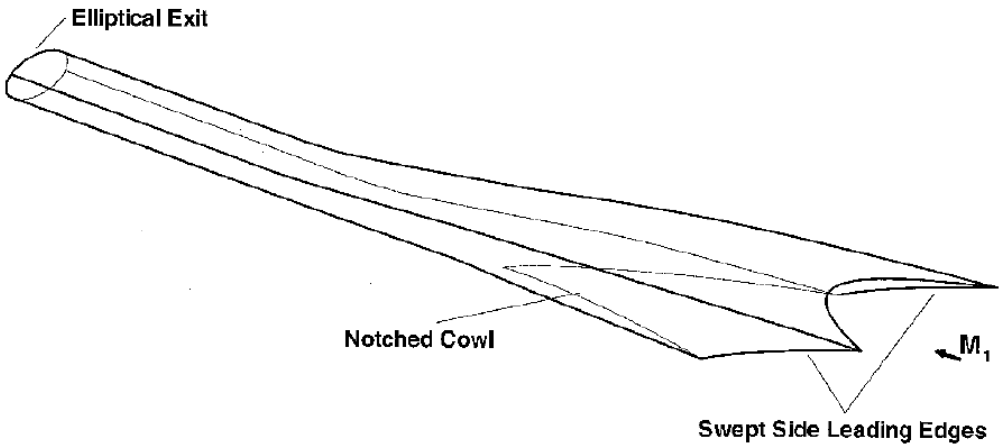


Figure 2.4: REST-class inlet diagram.[24]

proportional to both I_{sp} and η_o. When an estimate of drag is included, the term $\frac{\mathcal{F}_{un}-D_{est}}{\dot{m}_0}$ will be referred to as the *specific installed thrust*.

Other design considerations are the risk of unstart and the heat flux into the surface of the vehicle. Unstart may be caused either by thermally choked flow or by a back pressure rise. To avoid thermally choking, or the *Rayleigh limit*, the total temperature ratio τ_e may be limited. For pressure-based unstart, the *Korkegi limit*, an unsteady simulation would be required. These concepts are discussed in Section 2.1.2.

We can see from the number and interconnected nature of the potential quantities of interest that the choice of an appropriate objective function quickly becomes a complex problem. Many of the design considerations run counter to one another, or require high-fidelity unsteady multi-phase simulation to be evaluated accurately. The utility of the methods presented in this work will be demonstrated using relatively low-fidelity simulation of the performance downstream of the isolator, which determines the quantities that can be included. The tools developed in this work would also be compatible with higher fidelity methods. $\mathcal{F}_{un}$, P_{tr}, $\dot{m}$, inlet surface heat flux, drag, and τ_e will be included. Prior to the developments included in this work, some of these quantities would have required finite difference or automatic differentiation of an additional code. Obtaining the gradient of a combination of the objectives already accessible to the continuous adjoint method (heat flux, $\dot{m}$, and P_{tr}), would have required multiple gradient evaluations.

2.2.3 Gradient-Based Optimization

This section will discuss the mathematical background underpinning automated design optimization. One purpose of this section is to show what the gradient value contributes to the optimization process. In short, it provides both a direction in which to change the design variables, and information necessary to identifying the optimized design. This section will also introduce the concept of Lagrange multipliers, which are used for constrained optimization and lead to the derivation of adjoint equations that can be used to obtain gradient values more efficiently.

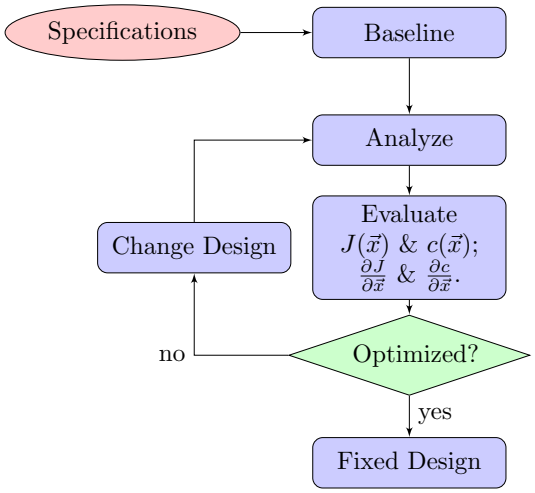


Figure 2.5: Optimization flow chart

During a conventional (non-automated/optimizer-based) design process, an initial design is first chosen that satisfies the requirements of the problem, usually based on previous designs, intuition, and empirical correlations. This baseline design is evaluated using analytical or experimental techniques, and changes are made iteratively until a fixed design is determined that is by some measure a "good" design. The design changes are determined by intuition and experience, and/or by trial and error. Optimization algorithms also start from a baseline design determined by the designer, and changes are made iteratively, however the method by which those changes are made and the decision about whether the design should be fixed are made by an algorithm based on mathematical formulas. This process is shown in Figure 2.5. This section will summarize the mathematical formulations that support optimization algorithms, and motivate the search for efficient gradient evaluations. A discussion of the work available in literature on optimization algorithms can be found in Section 1.3, and the equations presented here follow on Nash and Sofer.[81]

A general constrained optimization problem statement is, or nonlinear program is:

$$\begin{array}{ll} \text{minimize} & f(\vec{x}) \\ \text{with respect to} & \vec{x} \in \mathbb{R}^n \\ \text{subject to} & \hat{c}_j(\vec{x}) = 0, \quad j = 1, ..., \hat{m} \\ & c_k(\vec{x}) \geq 0, \quad k = 1, ..., m \end{array} . \tag{2.12}$$

In this problem statement, f is the objective function, which is minimized by convention. For a quantity that should be maximized, f would be its negative or inverse value. $\vec{x}$ is the vector of n design variables, and c are equality and inequality constraints. When the objective and/or constraints are nonlinear functions of the design variables, this is referred to as a Non-Linear Problem (NLP). When the objective and constraints are linear functions of the design variables, it is referred to as an LP. There has been much work done in the development of algorithms to address both types of problems; this problem addressed in this work is an NLP.

To implement NLP algorithms, we need a mathematical statement of the necessary and sufficient conditions for a local minimum. Taylor's theorem states that for a function f that is n times differentiable, there exists $\theta \in (0, 1)$ such that:

$$f(x+h) = f(x) + hf'(x) + \frac{1}{2}h^2 f''(x) + \ldots + \frac{1}{(n-1)!}h^{n-1}f^{n-1}(x) + \underbrace{\frac{1}{n!}h^n f^n(x+\theta h)}_{O(h^n)} \tag{2.13}$$

At a local minimum the function value will be lower than surrounding points,

$$f(x^*) < f(x^* + \epsilon) = f(x^*) + \epsilon f'(x^*) + \frac{1}{2}\epsilon^2 \frac{\partial^2 f(x^* + \epsilon\theta)}{\partial x^2}, \tag{2.14}$$

where $-\delta \leq \epsilon \leq \delta$ with a small positive δ. For a multi-variate problem,

$$f(\vec{x^*}) < f(\vec{x^*} + \epsilon\vec{p}) = f(\vec{x^*}) + \epsilon\vec{p}^T \nabla f(\vec{x^*}) + \frac{1}{2}\epsilon^2 \vec{p}^T \nabla^2 f(\vec{x^*} + \epsilon\theta\vec{p})\vec{p}. \tag{2.15}$$

For a sufficiently small value of ϵ, $\epsilon f'(x^*) \gg \frac{1}{2}\epsilon^2 f''(x^* + \epsilon\theta)$, which means that in order for $f(x^*)$ to be a local mimimum:

$$\epsilon f'(x^*) \geq 0. \tag{2.16}$$

In order for this to hold for negative as well as positive values of ϵ, $f'(x^*) = 0$, or for a multi-variate problem: $\nabla f(\vec{x^*}) = 0$. This defines 1^{st} order optimality. Since $\epsilon^2 \geq 0$, the 2^{nd} order optimality condition is $f''(x^*) \geq 0$, or for multiple variables: $\vec{p}^T \nabla^2 f(\vec{x^*})\vec{p} \geq 0$ for all $p \in \mathbb{R}^n$, in other words, the Hessian $H = \nabla^2 f$ is positive semi-definite. These are the necessary conditions for optimality, which when violated show that a point is not a minimum. To verify a point as a minimum, sufficient

conditions require that the Hessian is positive-definite.

For a problem with equality constraints, the same conditions must be satisfied; however the search direction $\vec{p}$ is no longer arbitrary, because the solution must lie along a contour defined by the constraints,

$$\begin{aligned} \nabla f^T dx &= 0 \\ \hat{c}_j(x + dx) &= 0 \\ \nabla \hat{c}_j^T dx &= 0 \text{ for } j = 1, \ldots, \hat{m}. \end{aligned} \tag{2.17}$$

This is now a problem with n unknowns, and $(n + \hat{m})$ equations. Lagrange provided a way to address this problem, by subtracting the total derivative of the constraint and introducing Lagrange multipliers (λ_i):

$$\begin{aligned} \nabla f^T dx + \sum_j \lambda_j \nabla \hat{c}_j^T dx = 0 = \sum_i \left(\frac{\partial f}{\partial x_i} dx_i - \sum_j \lambda_j \frac{\partial \hat{c}_j}{\partial x_i} dx_i \right) \\ \sum_i dx_i \left(\frac{\partial f}{\partial x_i} - \sum_j \lambda_j \frac{\partial \hat{c}_j}{\partial x_i} \right) = 0. \end{aligned} \tag{2.18}$$

This creates n equations:

$$\frac{\partial f}{\partial x_i} - \sum_j \lambda_j \frac{\partial \hat{c}_j}{\partial x_i} = 0 \text{ for } i = 1, \ldots, n, \tag{2.19}$$

in $(n + m)$ unknowns, where the Lagrange multipliers λ_j provide the additional m unknowns. The system is closed by satisfying the original constraint equations. Applying this to he optimization problem, a stationary point of the Lagrangian, $\mathcal{L} = f(x) - \lambda^T \hat{c}(x)$, satisfies the constraints and the necessary conditions on the gradient $\nabla f(\vec{x}) = g(\vec{x})$. The necessary conditions of a constrained problem can now be stated in terms of the Lagrangian, and are referred to as the Karush-Kuhn-Tucker (KKT) conditions. Sufficient conditions are defined by the positive-definiteness of a subspace defined by the linearization of the constraints. Table 2.3 summarizes necessary and sufficient conditions for various situations, including the KKT conditions. For nonlinear constraints (of the form $c_k(\vec{k} \geq 0)$, slack variables are introduced that put these constraints in a similar form as the equality constraints. The KKT conditions assume that the constraints are linearly independent, and alternate methods should be used in situations where that assumption is not satisfied, and depending on the problem and methods used it may be more convenient even when the constraints are linearly independent, to use the penalty method, which adds a function dependent on the constraint value to the objective function, resulting in a situation where the optimizer moves towards feasibility as the optimizer progresses.

Table 2.3: Necessary and Sufficient Optimality Conditions

Optimization Problem	Necessary Conditions	Sufficient Conditions
Unconstrained, single variable	$f'(x^*) = 0$ $f''(x^*) \geq 0$	$f'(x^*) = 0$ $f''(x^*) > 0$
Unconstrained, multivariate $g(\vec{x}) = \nabla f(\vec{x})$, $H(\vec{x}) = \nabla^2 f(\vec{x})$	$\|g(\vec{x}^*)\| = 0$ $\vec{p}^T H \vec{p} \geq 0$	$\|g(\vec{x}^*)\| = 0$ $\vec{p}^T H \vec{p} > 0$
Constrained, $\mathcal{L} = f(\vec{x}) - \lambda^T \hat{c}(\vec{x})$	$\frac{\partial \mathcal{L}}{\partial x_i} = \frac{\partial f}{\partial x_i} - \sum_j \frac{\partial \hat{c}_j}{\partial x_i} = 0$ $\frac{\partial \mathcal{L}}{\partial \lambda_j} = \hat{c}_j = 0$ (KKT conditions)	$w^T \nabla^2_{xx} \mathcal{L}(x^*, \hat{\lambda}^*) w > 0$ for all $w \in \mathbb{R}^n$ such that $\nabla \hat{c}_j(x^*)^T w = 0$

There are many optimizer algorithms that draw upon the equations presented in this section, including those mentioned in Section 1.3. In this work, the SNOPT[82] algorithm is used.

2.3 Modeling Fluid Flow & The Adjoint Method

This section will review the conservation equations, put them in the form used for numerical solutions, and introduce the backbone of the adjoint method: linearization of the governing equations and Lagrange multipliers. The history and previous developments in the adjoint method are discussed in Section 1.4.

2.3.1 Conservation Equations

The governing equations of fluid flow are developed from the conservation of mass, momentum, and energy. From these foundational concepts, partial differential equations with boundary conditions are stated in terms of the variables of a fluid flow, and discretized on a grid such that the unknown values of the fluid variables can be found through iterative methods. The boundary conditions of the problem are defined by the physical geometry being modeled, and the flow conditions at a distance away. The unknowns are the conservative variables at points on the computational grid and the thermodynamic state of the fluid. These equations are derived with the assumption that the fluid can be treated as a continuous medium, allowing physical properties of the fluid to be described as scalar or vector fields as a function of time and location. The continuum hypothesis is generally held to be sufficiently satisfied if the *Knudsen number* is much less than 1, $Kn = \lambda/L \ll 1$, where λ is the mean free path of a particle of the gas and L is the length scale of the relevant flow pheomenon. For some hypersonic flows, specifically reentry vehicles, this assumption breaks down. For all cases in this work, $Kn << 1$ and the continuum assumption will be used.

In this section some basic concepts necessary to move forward in the explanation of computational fluid dynamics will be reviewed. Throughout this section, and at other points in this work, Einstein notation will be used; $a_i j b_i = \sum_i a_{ij} b_i$. It may also be helpful to review the *Reynolds transport*

theorem, which states that for any material volume $\Omega(t)$ and differentiable scalar field ϕ we have:

$$\frac{d}{dt}\int_{\Omega(t)} \phi d\Omega = \int_{\Omega(t)} \left(\frac{\partial \phi}{\partial t} + \nabla \cdot (\phi \vec{v}) \right) d\Omega \tag{2.20}$$

For greater detail and the proof of this theorem, the reader should refer to Wesseling[83] and Hirsch,[84] as well as other texts on computational fluid dynamics.

This section will expand the conservation of mass, momentum, and energy in the forms convenient for analysis of fluid flows. The **conservation of mass** requires that the change of mass of an arbitrary volume (Ω) equals the mass production in that volume. Except in multiphase flows, the rate of mass production will be zero;

$$\frac{d}{dt}\int_{\Omega(t)} \rho d\Omega = \int_{\Omega(t)} \frac{d\rho}{dt} d\Omega = \int_{\Omega(t)} \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{v})\, d\Omega = 0\,, \tag{2.21}$$

where the density ρ has replaced ϕ in Reynolds transport theorem. Since this must hold for every Ω,

$$\frac{\partial \rho}{\partial t} + \cdot \nabla \cdot (\rho \vec{v}) = 0\,. \tag{2.22}$$

Equation 2.22 is called the *mass conservation law* or the *continuity equation*.

Newton's law of **conservation of momentum** states that the rate change of momentum of a material volume equals the total force acting on that volume. Splitting the force into body (f^b) and surface (f^s) forces, the i^{th} component of the conservation of momentum is:

$$\frac{d}{dt}\int_{\Omega(t)} \rho v_i d\Omega = \int_{\Omega(t)} f_i^b d\Omega + \int_{S(t)} f_i^s dS\,. \tag{2.23}$$

The surface forces can be stated in terms of a stress tensor and the outward normal direction of the volume surface, $f_i^s = \sigma_{ij} n_j$. The stress tensor can be related to the fluid motion by a *constitutive relation*. For Newtonian fluids,

$$\sigma_{ij} = -P\delta_{ij} + 2\mu \left(\epsilon_{ij} - \frac{1}{3} (\nabla \cdot \vec{v})\, \delta_{ij} \right)\,, \tag{2.24}$$

where the rate of strain tensor is defined as:

$$\epsilon_{ij} = \frac{1}{2}\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) = \frac{1}{2}\left(\nabla \vec{v} + \nabla \vec{v}^T \right)\,. \tag{2.25}$$

For brevity in later equations, the term $\bar{\bar{\tau}}$ will refer to the sum of the rate of strain tensor and the divergence term: $\nabla \vec{v} + \nabla \vec{v}^T - \frac{2}{3}(\nabla \cdot \vec{v})\, \bar{\bar{I}}$. A number of shorthand terms are summarized in Appendix A for easy reference throughout the text. Substituting for f_i^s and expanding the Reynolds

transport theorem on the left hand side of Equation 2.23,

$$\begin{aligned}\int_{\Omega(t)} \left\{ \frac{\partial \rho v_i}{\partial t} + \frac{\partial(\rho v_i v_j)}{\partial x_j} \right\} d\Omega &= \int_{\Omega(t)} \rho f_i^b d\Omega + \int_{S(t)} f_i^s dS \\ &= \int_{\Omega(t)} \left(\rho f_i^b + \frac{\partial \sigma_{ij}}{\partial x_j} \right) d\Omega \\ \frac{\partial \rho v_i}{\partial t} + \frac{\partial(\rho v_i v_j)}{\partial x_j} &= \rho f_i^b + \frac{\partial \sigma_{ij}}{\partial x_j}.\end{aligned} \tag{2.26}$$

The fourth line of Equation 2.26 is found by noting that the equation must hold for every volume Ω. The body force f^b is usually neglected, except cases such as stratified flow under the force of gravity, which is outside the scope of this work.

Combining 2.26 and Equation 2.24 results in the *Navier-Stokes equations*:

$$\frac{\partial \rho v_i}{\partial t} + \frac{\partial(\rho v_i v_j)}{\partial x_j} = -\frac{\partial P}{\partial x_i} + 2\frac{\partial \left\{ \mu(\epsilon_{ij} - \frac{1}{3}(\nabla \cdot \vec{v})\delta_{ij}) \right\}}{\partial x_j} + \rho f_i^b \,. \tag{2.27}$$

In vector form,

$$\frac{\partial \vec{v}}{\partial t} + \nabla \cdot (\rho \vec{v} \otimes \vec{v}) = -\nabla P + \nabla \cdot \bar{\bar{\tau}} \mu + \rho \vec{f^b} \,. \tag{2.28}$$

There are several forms of these equations that are commonly used. Nondimensionalization of these equations produces the dimensionless parameters of the Reynolds number and Mach number.

The **conservation of energy** draws from the first law of thermodynamics, that work done on a closed system plus the heat added is equal to the increase of the kinetic and internal energy ($E = e + \frac{1}{2} v_i v_i$):

$$\frac{d}{dt} \int_{\Omega(t)} \rho E d\Omega = W + Q \,. \tag{2.29}$$

W and Q are respectively the rate of work done on and the rate of heat addition into the fluid volume Ω. The work done on the volume can be expressed in terms of the body and surface forces, and the heat added in terms of the heat flux per unit area, $\sigma_q = k\vec{n} \cdot \nabla T$, where k is the thermal conductivity and T is the temperature. Expanding these terms,

$$\begin{aligned} W &= \int_{\Omega(t)} \frac{\partial v_i \sigma_{ij}}{\partial x_j} + \rho f_i^b \, d\Omega \\ Q &= \int_{\Omega(t)} \rho q \, d\Omega + \int_{S(t)} k\vec{n} \cdot \nabla T ds \\ \int_{\Omega(t)} \frac{\partial \rho E}{\partial t} + \frac{\partial(\rho v_i E)}{\partial x_i} d\Omega &= \int_{\Omega(t)} \left(\frac{\partial v_i \sigma_{ij}}{\partial x_j} + \nabla \cdot (k \nabla T) + \rho v_i f_i^b + \rho q \right) d\Omega \end{aligned} \,. \tag{2.30}$$

Noting that the last line must hold for every Ω,

$$\frac{\partial \rho E}{\partial t} + \frac{\partial (\rho v_i E)}{\partial x_i} = \frac{\partial v_i \sigma_{ij}}{\partial x_j} + \nabla \cdot (k \nabla T) + \rho v_i f_i^b + \rho q \,. \tag{2.31}$$

We have now summarized the conservation of mass (Equation 2.22), momentum (Equation 2.27), and energy (Equation 2.31), which comprise the governing equations of fluid flow.

Closing the Equation Set with Thermodynamic Relations

In addition to the five unknown conservative variables, we must also find two thermodynamic variables to describe the thermodynamic state of the gas. We therefore need two additional equations to close the system. For a perfect gas the system of equations is completed by $P = \rho RT$ and $e = e(T)$. Combining the conservative variables into $U = \left\{\rho, \rho\vec{v}^T, \rho E\right\}$ and defining flux vectors $\vec{F}$, the conservation equations can be summarized:

$$\begin{aligned}
\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{v}) &= 0 \\
\frac{\partial \rho v_i}{\partial t} + \frac{\partial (\rho v_i v_j)}{\partial x_j} &= -\frac{\partial P}{\partial x_i} + \frac{\partial \mu \tau_{ij}}{\partial x_j} + \rho f_i^b \\
\frac{\partial \rho E}{\partial t} + \frac{\partial (\rho v_i E)}{\partial x_i} &= \frac{\partial v_i \sigma_{ij}}{\partial x_j} + \nabla \cdot (k \nabla T) + \rho v_i f_i^b + \rho q \\
P &= \rho RT \\
e &= e(T) \,.
\end{aligned} \tag{2.32}$$

From here on, body forces (f^b) will be neglected.

Particulars for Hypersonic Flow

In hypersonic flow, depending on whether the temperature has exceeded a particular point, additional effects may need to be considered. This especially effects the energy equation, where the energy of chemical reactions may also be needed. The closure of the equations through the equation of state would be modified where real gas effects are expected to be relevant, as the specific heats can no longer be assumed constant. Multi-species flow is sometimes considered. The temperature range experienced in the inlet for this work was low enough to reasonable neglect these effects.

Some of the effect relevant specifically to hypersonic flow are included automatically by the satisfaction of the conservative equations. This includes shock-boundary layer interactions and entropy layers. These effects are expected to be relevant to the design case used in this work, and so the Navier Stokes equations will be used to model the flow.

2.3.2 Fluid Governing Equations in Conservation Form

This section will summarize the translation of the conservation equations into the vector forms used in computational fluid dynamics, referred to as the Conservation Form. Linearization of these equations will also be summarized. Jacobian matrices and flux vectors will be introduced. This material is needed for the derivation of the continuous adjoint problem.

Flux Vectors

Collecting the conservation equations into a vector form will allow them to be expressed in a compressed form, and allow the use of vector calculus solution methods. The first three equations can be abbreviated as follows:

$$\frac{\partial}{\partial t}\begin{Bmatrix}\rho\\ \rho\vec{v}\\ \rho E\end{Bmatrix}+\nabla\cdot\begin{Bmatrix}\rho\vec{v}\\ \rho\vec{v}\otimes\vec{v}\\ \rho\vec{v}E\end{Bmatrix}=\begin{Bmatrix}0\\ -\frac{\partial P}{\partial\vec{x}}+\nabla\cdot[\mu\bar{\bar{\tau}}]^T\\ \frac{\partial v_i\sigma_{ij}}{\partial x_j}+\nabla\cdot(k\nabla T)+\rho q\end{Bmatrix}$$

$$\frac{\partial}{\partial t}\begin{Bmatrix}\rho\\ \rho\vec{v}\\ \rho E\end{Bmatrix}+\nabla\cdot\begin{Bmatrix}\rho\vec{v}\\ \rho\vec{v}\otimes\vec{v}+\bar{\bar{I}}P\\ \rho\vec{v}E+P\vec{v}\end{Bmatrix}=\begin{Bmatrix}0\\ \nabla\cdot\left[\mu(\nabla\vec{v}+\nabla\vec{v}^T-\frac{2}{3}(\nabla\cdot\vec{v})\bar{\bar{I}})\right]^T\\ \nabla\cdot\left(\vec{v}\cdot\mu\left(\nabla\vec{v}+\nabla\vec{v}^T-\frac{2}{3}(\nabla\cdot\vec{v})\bar{\bar{I}}\right)\right)+\nabla\cdot(k\nabla T)+\rho q\end{Bmatrix}. \tag{2.33}$$

Using U to refer to the vector of conservative variables and introducing flux vectors F,

$$\frac{\partial U}{\partial t}+\nabla\cdot\vec{F}^c=Q+\nabla\cdot\mu^1_{tot}\vec{F}^{\nu 1}+\nabla\cdot\mu^2_{tot}\vec{F}^{\nu 2}\,. \tag{2.34}$$

In this equation, $\vec{F}^c$ refers to the vector of convective fluxes. Equation 2.34 is a concise statement of the governing equations of fluid flow, and has introduced notation for the vector of conservative variables,

$$U=\begin{Bmatrix}\rho\\ \rho\vec{v}\\ \rho E\end{Bmatrix}, \tag{2.35}$$

as well as the convective flux vector, viscous flux vector, and source terms:

$$\vec{F}^c=\begin{Bmatrix}\rho\vec{v}\\ \rho\vec{v}\otimes\vec{v}+\bar{\bar{I}}P\\ \rho E\vec{v}+P\vec{v}\end{Bmatrix},\ \vec{F}^{v1}=\begin{Bmatrix}\vec{0}^T\\ \bar{\bar{\tau}}\\ \bar{\bar{\tau}}\cdot\vec{v}\end{Bmatrix},\ \vec{F}^{v2}=\begin{Bmatrix}\vec{0}^T\\ \bar{\bar{0}}\\ c_p\nabla T\end{Bmatrix},\ \mathcal{Q}=\begin{Bmatrix}q_\rho\\ \vec{q}_{\rho\vec{v}}\\ q_{\rho E}\end{Bmatrix}, \tag{2.36}$$

where for generality nonzero source terms have been included, and $q_{\rho E}=\rho q$, the product of heat

addition per mass and density.

Equation 2.34 also transitions from using a generic viscosity μ and thermal conductivity, to using a total viscosity with the first viscous flux μ^1_{tot}, and a thermal conductivity term μ^2_{tot}. These quantities are defined in terms of the laminar and turbulent viscosity:

$$\begin{aligned} \mu^1_{tot} &= \mu_{dyn} + \mu_{tur} \\ \mu^2_{tot} &= \frac{\mu_{dyn}}{Pr_d} + \frac{\mu_{tur}}{Pr_t}\,. \end{aligned} \tag{2.37}$$

The laminar viscosity (μ_{dyn}) is a function of temperature only, specifically Sutherland's law:[85]

$$\mu = \mu_{ref}\left(\frac{T}{T_{ref}}\right)^{3/2}\frac{T_{ref}+S}{T+S}\,, \tag{2.38}$$

where S is Sutherland's constant. The turbulent viscosity is found from a turbulence model. Several options are available, generally grouped into one-equation and two-equation models. The one-equation Spalart-Allmaras[86] and two-equation SST k-omega[87] are implemented in the CFD code used in this work, and the SST model is used for its expected better performance in shock-boundary layer interactions.

Equation 2.34 is now reorganized to define the residual, which will be driven to zero by iterative numerical methods.

$$\mathcal{R}\left(U\right) = \frac{\partial U}{\partial t} + \nabla\cdot\vec{F}^c - \nabla\cdot\left(\mu^1_{tot}\vec{F}^{\nu 1} + \mu^2_{tot}\vec{F}^{\nu 2}\right) - \mathcal{Q} = 0\,. \tag{2.39}$$

The inviscid form of these equations, otherwise known as the Euler equations are:

$$\mathcal{R}\left(U\right) = \frac{\partial U}{\partial t} + \nabla\cdot\vec{F}^c - \mathcal{Q} = 0\,. \tag{2.40}$$

In order to solve this system of equations using PDE methods, we must translate it into a quasi-linear form. This is done with Jacobian matrices, which are expanded in the Appendix, Section A.2.

Implementation

The open-source CFD suite SU2, developed in the Aerospace Design Lab at Stanford University, was used to generate flow solutions and the adjoint solution, and was modified as part of this work. SU2 uses the Finite Volume Method (FVM) to solve partial differential equations on unstructured meshes. Further information is available in works by the SU2 team.[59,63] In the Reynolds-Averaged-Navier-Stokes (RANS) equations, a turbulence model is used to account for the Reynolds stresses. The one-equation Spalart-Allmaras[86] and two-equation SST k-omega[87] turbulence models are available. The continuous adjoint equations are solved in a similar fashion, re-using methods implemented to solve partial differential equations and the information generated by the flow solver. Numerical

methods and other details will be discussed further in Section 4.1

A new boundary condition and other modifications were implemented in this work in order to produce the continuous adjoint solution for generalized outflow-based objective functions. These equations are presented in Section 3.2. The RANS equations with the SST turbulence model were used for viscous simulations. The second order JST scheme was used. Further details of numerical methods are discussed in Section 4.1.2.

2.3.3 The Adjoint Method

Advances in computer speed and efficiency make simulation-based design more feasible, however the CPU time required for high-fidelity optimization with large number of design variables is still significant. Efficient calculation of the gradient of the objective function(s) significantly reduces the computational cost of optimization. The adjoint method provides gradients at a computational cost independent of the step size and of the number of design variables. Adjoints have been applied to scramjet designs in the past, for example Wang et al.[88] used a discrete adjoint for a pressure-based functional on the solid surface of a scramjet inlet to accelerate Monte-Carlo characterization of the probability of unstart.

The continuous and discrete adjoint methods provide surface sensitivities at the approximate cost of a single additional flow solution, as compared to the finite difference method that perturbs each design variable and re-evaluates the objective. The continuous adjoint method solves the discretization of the linearized fluid problem, while the discrete adjoint method solves the linearization of the discretized fluid problem. In other words, the continuous adjoint method solves a problem defined by careful implementation of a PDE defined by taking the linearization of the original continuous fluid problem. The discrete adjoint method solves a problem defined by taking the adjoint of the already-discretized fluid problem. These methods should be equivalent if the discretization of both are perfect and if the numerical tricks such as limiters and other corrections are included.

The discrete adjoint method as implemented in the SU2 framework utilizes automatic differentiation of the initial CFD code to find an exact derivative of the inexact functional. More detail is available in.[64,65] This method has some advantages in obtaining results that are generally much closer to finite difference results, however there are greater memory requirements due to the larger amount of information needed during calculation. The continuous adjoint method, by contrast, is sometimes more removed from finite difference and discrete adjoint result, however it benefits from lower memory usage and, particularly relevant to this work, the boundary conditions are relatively accessible within the implementation.

Some work[47,66] has focused on making the discrete adjoint method more efficient and expanding its capabilities. This work approaches the problem from the other direction - by making the boundary conditions of the continuous adjoint method more general such that the numerical efficiency of the continuous adjoint method can be utilized with a lesser limitation on the objective function. Arian

& Salas[62] have also explored expanding the limits of what objectives can be addressed by the continuous adjoint method, focusing on solid wall boundaries.

Section 3.1 will discuss the adjoint method in the context of fluid flow. Here, this methodology is discussed in general terms applicable to many problems. The adjoint method leverages Lagrange multipliers to solve for the sensitivity of a function with respect to an arbitrary number of variables at a fixed cost. This method relies on linearization, and assumes that the domain is smoothly differentiable. This section will lay out the general methodology of adjoints, so that the reader will be able to see the underlying framework without the distraction of the large and complicated equations required for the adjoint of the governing equations of fluid flow. The derivation here is similar to work presented by Jameson,[51] with modifications to some of the notation for consistency with later equations.

The derivation of an adjoint method starts with defining the objective function J, some functional defined in terms of state variables U and independent design variables x.

$$J(U, x) \tag{2.41}$$

We want to find the variation of J with respect to x, and in order to do so we need to define the relationships between x and U. The state variables and design variables are related to each other by a governing equation, often describing a physical system, stated in the following form as a constraint,

$$\mathcal{R}(U, x) = 0\,. \tag{2.42}$$

The first variation of the functional and governing equations with respect to variations in U and x are as follows:

$$\delta J = \frac{\partial J}{\partial U}\delta U + \frac{\partial J}{\partial x}\delta x, \tag{2.43}$$

$$\delta \mathcal{R} = \frac{\partial \mathcal{R}}{\partial U}\delta U + \frac{\partial \mathcal{R}}{\partial x}\delta x = 0\,, \tag{2.44}$$

where the variation in the governing equations is set to zero to ensure that this constraint is met for every arbitrary variation δx and corresponding δU. This step has linearized the system. Equation 2.44 can be used to find δU for a given δx, and the result substituted into Equation 2.43 in order to find $\frac{\delta J}{\delta x}$, however this process would need to be repeated for every additional x. Therefore, it would be convenient to find a relationship between δJ and δx that does not depend on δU. Because $\delta \mathcal{R}$ can be set to 0,

$$\delta \mathcal{J} = \delta J - \lambda \delta \mathcal{R} = \delta U \left(\frac{\partial J}{\partial U} - \lambda \frac{\partial \mathcal{R}}{\partial U} \right) + \delta x \left(\frac{\partial J}{\partial x} - \lambda \frac{\partial \mathcal{R}}{\partial x} \right), \tag{2.45}$$

where the augmented objective $\mathcal{J}$ is equivalent to J and the adjoint variables λ are arbitrary because $\mathcal{R}$ has been set to zero. Because λ are arbitrary, we can choose them such that this equation is independent of δU, in other words:

$$\frac{\partial J}{\partial U} - \lambda \frac{\partial \mathcal{R}}{\partial U} = 0. \tag{2.46}$$

Equation 2.46 is the adjoint equation, which is used to find the values of λ that make the system independent of the variation in U. At this point we are left with:

$$\frac{\delta \mathcal{J}}{\delta x} = \frac{\delta J}{\delta x} = \left(\frac{\partial J}{\partial x} - \lambda \frac{\partial \mathcal{R}}{\partial x} \right), \tag{2.47}$$

which describes the sensitivity of the functional J with respect to the independent, or design, variables x. The challenge then is to solve the system described by Equation 2.46 for the adjoint variables λ. Methods for doing so for a system of Partial Differential Equations (PDEs) were introduced by Lions[48] and Pironneau[50] for elliptic PDEs, and applied to the hyperbolic PDEs for aerodynamic optimization by Jameson.[51]

2.4 Summary of Challenges

Some of the challenges that are relevant to scramjet design and optimization have been noted in preceding sections, and will be reviewed here in order to provide context for the proceeding sections and developments of this dissertation.

Before embarking on an optimization problem, or even evaluating sensitivities, first the direct or "flow" problem should be solved. A high accuracy is desired, but there are trade-offs between accuracy and computational cost that become more significant in hypersonic flow as various additional phenomena become more significant including non-equilibrium gas effects and shock-boundary layer interactions. For the Mach and Reynolds numbers used in this work, Reynolds-Averaged Navier Stokes is a reasonable choice for the inlet of the scramjet, although some inaccuracy is introduced by neglecting real gas effects. Real gas effects are included in the combustion model, where higher temperatures increase the significance of these effects. Non-equilibrium, ionization, and rarefied gas effects are not expected to be significant for the problem investigated in this work. The phenomenon unstart that is a consideration for hypersonic airbreathing engines is difficult to predict with current techniques, and often requires significantly more computationally expensive techniques including unsteady flow, however some empirical methods exist.

The simulation or prediction of the vehicle performance also encounters the problem of accessing the most relevant quantities of interest. It is common to use simulations that address a single component of the engine, for example the inlet. However, without knowledge of the remainder of the engine flowpath, no matter how high of fidelity is used in the simulation the most relevant

quantities of interest may not be accessible. Specifically, inlet simulations can produce quantities such as the total pressure ratio, static pressure ratio, mass flow rate, and flow distortion measures. Each of these quantities are correlated in different ways with improved engine performance, however the specific balance between them is often not clear, and highly dependent on the specifics of the engine downstream. This motivates the use of multi-fidelity methods that can provide a prediction of the full engine performance without an overly costly increase in computation time. The simulation of the entire flowpath is particularly important for scramjet engines, where phenomena such as unstart and quantities like the maximum pressure in the combustor become important, and are sensitive to the inlet design.

In an optimization process, the result is generally dependent on the initial point. There has been much work that can contribute the the initial point, which is described in Section 2.2.1, however there are often discrepancies between the predicted performance and the performance under higher fidelity (CFD) models, which motivates optimization using those higher fidelity methods in order to achieve the desired performance and satisfy constraints. An additional challenge within the design process is the choice of objective and constraints, as there are often several conflicting quantities of interest, as detailed in Section 2.2.2

After the initial point is created for the optimization process, the next challenge is to produce sufficiently accurate gradients at a reasonable computational cost. The continuous adjoint method provides an attractive option, as its cost is relatively low and constant relative to the number of design variables. However, existing methods for the continuous adjoint method are limited to functionals defined within the CFD volume, to single functionals (requiring additional solution evaluations for additional objectives or constraints), and require additional derivation and code implementation for functionals that had not been previously addressed. Some of these limitations are eased for the discrete adjoint method when automatic differentiation is used, which does not require additional implementation for new objectives or combinations of objectives, but which requires that the code be compatible with the automatic differentiation program. Additional trade-offs between the continuous and discrete adjoint methods are discussed in Section 1.4. The continuous adjoint method is mainly used in this work.

The main challenges addressed by this work are the challenges relevant to the continuous adjoint method and expanding the range and complexity of functionals that it can address. These developments facilitate optimization problems that utilize a compromise between high- and low-fidelity methods through the use of a multi-fidelity flowpath.

Chapter 3

Generalizing the Continuous Adjoint Method

Chapter 3 starts with an overview of the derivation of the continuous adjoint method for the compressible, steady, Navier-Stokes equations in Section 3.1 for a selection of functionals. This leads to Section 3.2 that provides the derivation for generalized functionals, building on the equations developed in Section 3.1 and allowing contrasting between deriving the continuous adjoint method for specific functionals and generalized functionals. The generalized functional facilitates the application of the continuous adjoint method to a broader range of functionals, and Section 3.3 describes how to exploit this methodology through providing the necessary partial derivative terms. Section 3.4 provides details of using superposition to provide gradients of multiple functionals simultaneously. Section 3.1 reviews derivation details that can be found in the literature, and Section 3.2 - Section 3.4 provide the details of the methods developed in this work. Navigation of this chapter may be aided by the flow chart in Figure 3.1.

3.1 The Continuous Adjoint Method for Compressible Viscous Flow

This section will detail the derivation of the continuous adjoint method for compressible fluid flow for a selection of functionals that have been previously addressed in literature. A discussion of available literature is included in Section 1.4. The equations governing fluid flow were reviewed in Section 2.3.2, and the reader may find it helpful to refer to this section. These equations will compose a constraint on the minimization problem that defines the adjoint method. The computational domain is composed of an aerodynamic surface S in a fluid, $\Omega \subset \mathbb{R}^3$. There are a number of different boundary conditions that can be applied. An example of a set of boundary conditions that might

be applied are:

$$\begin{cases} \mathcal{R}(U) = \nabla \cdot \vec{F}^c - \nabla \cdot \left(\mu_{tot}^1 \vec{F}^{v1} + \mu_{tot}^2 \vec{F}^{v2} \right) - \mathcal{Q} = 0 & \text{in } \Omega \\ \vec{v} = 0 & \text{on } S \\ T = T_w & \text{on } S \\ (W)_+ = W_\infty & \text{on } \Gamma_\infty, \Gamma_{in} \\ \vec{v} \cdot \vec{n} = 0 & \text{on } \Gamma_{sym} \\ P = P_e & \text{on } \Gamma_e|_{M<1}\,, \end{cases} \tag{3.1}$$

where the open boundaries are composed of the farfield Γ_∞, inflow Γ_{in}, outflow Γ_e, and symmetry Γ_{sym} components. The open boundaries will sometimes be referred to collectively as Γ. An outflow boundary condition of a constant pressure is applied at outflow boundaries. The characteristic variables W result from the diagonalization of the convective Jacobian matrices, and $(W)_+$ indicates the positive characteristics. Defining a nonlinear program, or minimization problem, which satisfies these governing equations with an objective function defined in terms of the aerodynamic forces and temperature at the surface S,

$$\begin{aligned} &\min_S J(S) = \int_S j(\vec{f}, T, \partial_n T, \vec{n}) ds \\ &\text{subject to: } \mathcal{R}(U) = 0, \end{aligned} \tag{3.2}$$

where $\vec{f}$ is the force on the surface, T is the temperature, and $\vec{n}$ is the outward-pointing unit normal vector to the surface S.A more standard derivation is to address a single functional. Using a Lagrangian formulation,

$$\min_S \mathcal{J}(S) = \int_S j(\vec{f}, T, \partial_n T, \vec{n}) ds - \int_\Omega \Psi^T \mathcal{R}(U) d\Omega, \tag{3.3}$$

where the vector of adjoint variables Ψ are introduced as the Lagrange multipliers, which are arbitrary thanks to the constraint that $\mathcal{R}(U) = 0$. The components of this vector are:

$$\Psi = \begin{Bmatrix} \psi_\rho \\ \psi_{\rho u} \\ \psi_{\rho v} \\ \psi_{\rho w} \\ \psi_{\rho E} \end{Bmatrix} = \begin{Bmatrix} \psi_\rho \\ \vec{\varphi} \\ \psi_{\rho E} \end{Bmatrix}. \tag{3.4}$$

We will now take the first variation of the Lagrangian or augmented functional,

$$\delta \mathcal{J}(S) = \delta J - \int_\Omega \Psi^T \delta \mathcal{R}(U, \nabla U) d\Omega, \tag{3.5}$$

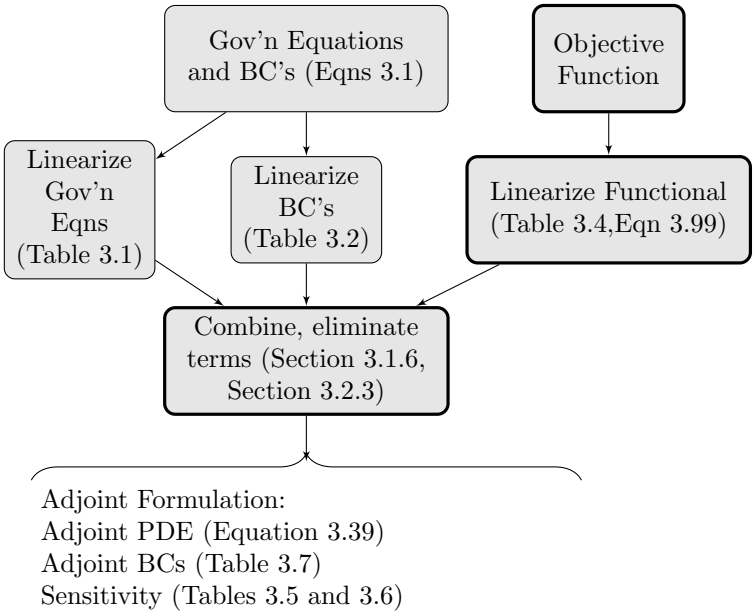


Figure 3.1: Derivation of Adjoint Equations

and solve for the values of Ψ that eliminate dependence on the unknown variations δU (similar to Equation 2.46), in order to find an expression for $\delta\mathcal{J}/\delta S$, the surface sensitivity (similar to Equation 2.47). The surface S is assumed continuously differentiable $\left(C^1\right)$, and the shape perturbation is described by infinitesimal deformations in the normal direction:

$$S' = \{\vec{x} + \delta S(\vec{x})\vec{n}(\vec{x}) : \vec{x} \in S\}. \tag{3.6}$$

In order to solve for the adjoint variables that eliminate dependence on δU, the components of Equation 3.5 will be expanded. The expressions will then be combined to eliminate and simplify terms, resulting in the adjoint PDE and its governing equations. The remaining nonzero quantities comprise the surface sensitivity. Figure 3.1 provides an illustration of the process used to derive the adjoint equations, with reference to the equations and tables that summarize the result of each step. Boxes in this flow chart with bold boundaries indicate the steps that differ between the derivation for a specific functional and the derivation for a generalized functional, which will be presented in Section 3.2.

3.1.1 Surface Variations & Shorthand Terms

Throughout the derivation of the adjoint equations, knowledge of surface variations will be useful. Some useful identities exist that aid in linearizing the equations as well as simplifying the terms on

the boundaries of the adjoint problem:

$$\begin{aligned}
\int_{\delta S}(\cdot)ds &= \int_{S'}(\cdot)ds - \int_{S}(\cdot)ds \\
\int_{\delta S} j ds &= \int_{S}(\partial_n j - 2H_m j)\delta S ds \\
H_m &= (\kappa_1 + \kappa_2)/2 \\
\partial_n(j) &= \vec{n}\cdot\nabla(j) \\
\delta\vec{n} &= -\nabla_S(\delta S),
\end{aligned} \tag{3.7}$$

where H_m is the mean curvature of S and (κ_1, κ_2) are curvatures in two orthogonal directions on the surface. The relationship for $\delta\vec{n}$ holds for small deformations.[89] ∇_S represents the tangential gradient operator on S. Additional shorthand terms will be introduced within the derivation of the adjoint equations, and for reference they are also summarized in Appendix A.1.

A property that will be used repeatedly in these equations is that $\int_S \nabla_S((\cdot)\delta S)ds = 0$ on a closed surface. In order to apply these same equations to domains where the solid surface is not closed (ie, for internal flows where a solid-walled duct is capped by an outflow boundary), we note that $\delta S = 0$ on open boundaries, and require by careful choice of design variables that deformations are chosen such that δS goes smoothly to 0 at the borders between S and Γ.

3.1.2 Linearizing the Governing Equations

The linearized form of the direct (fluid flow) problem will be used in the derivation of the adjoint equations. This section details the linearization process for the governing PDE as well as several boundary conditions. This section is concluded tables summarizing the resulting linearized form for easy reference in the later derivations. The governing equations of fluid flow were discussed in a previous section. The residual of the steady Reynolds Averaged Navier Stokes (RANS) equations is as follows:

$$\mathcal{R}(U) = \nabla\cdot\vec{F}^c - \nabla\cdot\left(\mu^1_{tot}\vec{F}^{v1} + \mu^2_{tot}\vec{F}^{v2}\right) - \mathcal{Q} = 0 \text{ in } \Omega. \tag{3.8}$$

Linearizing this equation with respect to small deformations δS,

$$\begin{aligned}
\delta\mathcal{R}(U,\nabla U) &= \delta\left(\nabla\cdot\vec{F}^c - \nabla\cdot\mu^k_{tot}\vec{F}^{vk} - \mathcal{Q}\right)\\
&= \nabla\cdot\delta\vec{F}^c - \nabla\cdot\delta(\mu^k_{tot}\vec{F}^{vk}) - \delta\mathcal{Q}\\
&= \nabla\cdot\delta\vec{F}^c - \nabla\cdot\left(\vec{F}^{vk}\delta\mu^k_{tot}\right) - \nabla\cdot\left(\mu^k_{tot}\delta\vec{F}^{vk}\right) - \delta\mathcal{Q}\\
&= \nabla\cdot\left(\frac{\partial\vec{F}^c}{\partial U}\delta U\right) - \nabla\cdot\vec{F}^{vk}\frac{\partial\mu^k_{tot}}{\partial U}\delta U\\
&\quad -\nabla\cdot\mu^k_{tot}\left[\frac{\partial\vec{F}^{vk}}{\partial U}\delta U + \frac{\partial\vec{F}^{vk}}{\partial(\nabla U)}\delta(\nabla U)\right] - \frac{\partial\mathcal{Q}}{\partial U}\delta U\\
&= \nabla\cdot\left(\vec{A}^c - \mu^k_{tot}\vec{A}^{vk}\right)\delta U - \nabla\cdot\vec{F}^{vk}\frac{\partial\mu^k_{tot}}{\partial U}\delta U\\
&\quad - \nabla\cdot\mu^k_{tot}\bar{\bar{D}}^{vk}\delta(\nabla U) - \frac{\partial\mathcal{Q}}{\partial U}\delta U.
\end{aligned} \tag{3.9}$$

Applying the assumption of frozen viscosity ($\delta\mu^k_{tot} = 0$) simplifies the problem significantly, although the assumption of constant viscosity may introduce inaccuracies. In order to fully account for the variation of turbulent velocity, the adjoint of the turbulence model, with additional adjoint variables, would be required. These terms are detailed for the S-A turbulence model by Bueno[90] and Zymaris,[91] however additional derivation would be required to form the adjoint of the SST model, and these effects are outside the scope of this work. Equation 3.9 also introduces the Jacobians $\vec{A}^c = \frac{\partial\vec{F}^c}{\partial U}$, $\vec{A}^{\nu k} = \frac{\partial\vec{F}^{\nu k}}{\partial U}$ and $\bar{\bar{D}}^{\nu k} = \frac{\partial\vec{F}^{\nu k}}{\partial(\nabla U)}$, which are expanded in the Appendix in Equation A.10, A.11, A.12, A.13, and A.16. The final line of Equation 3.9 as well as the corresponding linearization of the Euler Equations are included in Table 3.1.

3.1.3 Expansion of δR

The term including the linearized governing equations, from the right hand side of the Lagrangian defined in Equation 3.5 will now be expanded in order to separate boundary terms and define the volume PDE that will be solved for the adjoint variables Ψ. Using the expression for δR expanded in Equation 3.9, the right hand side of the Lagrangian is now:

$$\begin{aligned}
\delta\mathcal{J} = \delta J &- \int_\Omega \Psi^\mathsf{T}\frac{\partial}{\partial t}(\delta U)\,d\Omega - \int_\Omega \Psi^\mathsf{T}\nabla\cdot\left(\vec{A}^c - \bar{\bar{I}}_5\vec{u}_\Omega - \mu^k_{tot}\vec{A}^{vk}\right)\delta U\,d\Omega\\
&+ \int_\Omega \Psi^\mathsf{T}\nabla\cdot\mu^k_{tot}\bar{\bar{D}}^{vk}\delta(\nabla U)\,d\Omega + \int_\Omega \Psi^\mathsf{T}\frac{\partial\mathcal{Q}}{\partial U}\delta U d\Omega.
\end{aligned} \tag{3.10}$$

Integrating by parts in order to collect terms inside the divergence operator, and then applying the divergence theorem:

$$\begin{aligned}\int_\Omega \nabla\cdot\left(\Psi^\mathsf{T}[\ \ ]\delta U\right)d\Omega &= \int_\Omega \nabla\Psi^\mathsf{T}\cdot[\ \ ]\delta U d\Omega + \int_\Omega \Psi^\mathsf{T}\nabla\cdot[\ \ ]\delta U d\Omega \\ \int_\Omega \Psi^\mathsf{T}\nabla\cdot[\ \ ]\delta U d\Omega &= \int_{\partial\Omega} \Psi^\mathsf{T}[\ \ ]\delta U\cdot\vec{n}ds + \int_\Omega \nabla\Psi^\mathsf{T}\cdot[\ \ ]\delta U d\Omega.\end{aligned} \tag{3.11}$$

Substituting for the placeholder term:

$$\begin{aligned}\int_\Omega \Psi^\mathsf{T}\nabla\cdot(\vec{A}^c - \bar{\bar{I}}_5\vec{u}_\Omega - \mu^k_{tot}\vec{A}^{vk})\delta U d\Omega &= \int_{\partial\Omega}\Psi^\mathsf{T}(\vec{A}^c - \bar{\bar{I}}_5\vec{u}_\Omega - \mu^k_{tot}\vec{A}^{vk})\delta U\cdot\vec{n}ds \\ &\quad + \int_\Omega \nabla\Psi^\mathsf{T}\cdot(\vec{A}^c - \bar{\bar{I}}_5\vec{u}_\Omega - \mu^k_{tot}\vec{A}^{vk})\delta U d\Omega.\end{aligned} \tag{3.12}$$

The viscous stress terms require integrating by parts twice, a technique used by Castro[55] to eliminate the second-order derivatives:

$$\begin{aligned}\int_\Omega \Psi^\mathsf{T}\nabla\cdot\left(\mu^k_{tot}\bar{\bar{D}}^{\nu k}\cdot\delta(\nabla U)\right)d\Omega &= \int_{\partial\Omega}\Psi^\mathsf{T}\left(\Psi^\mathsf{T}\mu^k_{tot}\bar{\bar{D}}^{\nu k}\cdot\delta(\nabla U)\right)\cdot\vec{n}ds \\ &\quad + \int_\Omega \nabla\Psi^\mathsf{T}\cdot\left(\mu^k_{tot}\bar{\bar{D}}^{\nu k}\cdot\delta(\nabla U)\right)d\Omega.\end{aligned} \tag{3.13}$$

Applying integration by parts a second time and using $\nabla(\delta U) = \delta(\nabla U)$ in a continuum:

$$\begin{aligned}\int_\Omega \nabla\cdot(\nabla\Psi^\mathsf{T}\cdot(\cdot)\delta U)d\Omega &= \int_\Omega \nabla\cdot(\nabla\Psi^T\cdot(\cdot)\delta U d\Omega + \int_\Omega(\nabla\Psi^\mathsf{T}\cdot(\cdot))\cdot\nabla(\delta U)d\Omega \\ \int_\Omega\left(\nabla\Psi^\mathsf{T}\cdot\mu^k_{tot}\bar{\bar{D}}^{\nu k}\right)\cdot\nabla(\delta U)d\Omega &= \int_{\partial\Omega}\left(\nabla\Psi^\mathsf{T}\cdot\mu^k_{tot}\bar{\bar{D}}^{\nu k}\delta U\right)\cdot\vec{n}ds \\ &\quad - \int_\Omega\nabla\cdot\left(\nabla\Psi^\mathsf{T}\cdot\mu^k_{tot}\bar{\bar{D}}^{\nu k}\right)\delta U d\Omega.\end{aligned} \tag{3.14}$$

Combining these terms, we can now restate the Lagrangian succinctly, introducing shorthand terms for terms on the surface and collecting the volume terms remaining after the application of the divergence theorem. Shorthand terms B_1, B_2, and B_3 are grouped by convective flux terms that are relevant to both viscous and inviscid problems in B_1, viscous flux terms in B_2, and the terms dependent on ∇U in B_3. B_3 includes both viscous stress terms and heat transfer terms.

$$\begin{aligned}B_1 &= \Psi^\mathsf{T}\left(\vec{A}^c - \bar{\bar{I}}_5\vec{u}_\Omega\right)\delta U\cdot\vec{n}, \\ B_2 &= \Psi^\mathsf{T}\mu^k_{tot}\vec{A}^{vk}\delta U\cdot\vec{n} + \Psi^\mathsf{T}\mu^k_{tot}\bar{\bar{D}}^{vk}\cdot\nabla(\delta U)\cdot\vec{n}, \\ B_3 &= \nabla\Psi^\mathsf{T}\cdot\mu^k_{tot}\bar{\bar{D}}^{vk}\delta U\cdot\vec{n}.\end{aligned} \tag{3.15}$$

Writing the Lagrangian using the shorthand terms introduced in Equation 3.15:

$$\begin{aligned}\delta\mathcal{J} = \delta J &- \int_{\partial\Omega} \left(B_1 - B_2 + B_3\right)\, ds \\ &- \int_{\Omega} \left[-\nabla\Psi^{\mathsf{T}} \cdot \left(\vec{A}^c - \bar{\bar{I}}_5 \vec{u}_\Omega - \mu^k_{tot}\vec{A}^{vk}\right) - \nabla \cdot \left(\nabla\Psi^{\mathsf{T}} \cdot \mu^k_{tot}\bar{\bar{D}}^{vk}\right) - \Psi^{\mathsf{T}}\frac{\partial \mathcal{Q}}{\partial U}\right] \delta U \, d\Omega.\end{aligned} \tag{3.16}$$

δU must now be eliminated from these equations. The bracketed term under the volume integral from Equation 3.16 represents the PDE system known as the adjoint equations. Solving for the values of the adjoint variables Ψ that drive the bracketed term to 0 eliminates dependence on δU from the volume therms. The adjoint variables must also satisfy boundary conditions defined by eliminating dependence on δU from the combination of δJ and the surface terms in Equation 3.16. The remaining terms, dependent only on the flow solution, the values of the adjoint variables, and δS, produce the surface sensitivity. The following sections will expand the boundary integrals, and then combine them with the functional variation to form the boundary conditions on the adjoint equations.

Expanding Adjoint Boundary Integrals

This section will expand the boundary integrals introduced in Equation 3.16, and apply linearized boundary conditions of the direct (flow) problem in order to simplify the integrals prior to combination with expansions of the functional term. The boundaries $\partial\Omega$ can be divided into solid boundaries that lie on the surface of the object in the flow field, farfield boundaries, inflow, outflow, and symmetry boundaries. At the open boundaries, viscous perturbations are generally neglected, making the boundary equations equivalent in the viscous and inviscid forms. This is a reasonable assumption at farfield boundaries where viscous effects are expected to be negligible. Inclusion of viscous terms in outflow boundaries may be reasonably considered for future work. Jacobians, transformation matrix, and shorthand terms can be found in Appendix A. Some of the shorthand terms used include: $a_0 = (\gamma - 1)$, $\phi = (\gamma - 1)\frac{|\vec{v}|^2}{2}$, $v_n = \vec{v} \cdot \vec{n}$ and $v_{nr} = (\vec{v} - \vec{u}_\Omega) \cdot \vec{n}$. The transformation matrix M can be found in the Appendix, Equation A.9. A vector of primitive variable perturbations, $\delta V = \{\delta\rho, \delta\vec{v}, \delta P\}^{\mathsf{T}}$ has been introduced to these equations for easier manipulation.

Expansion & Simplification of B_1 The expression B_1 contains the convective terms as well as terms related to the mesh motion, $\vec{u}_\Omega$. These terms are common to both the viscous (Navier-Stokes, RANS) and inviscid (Euler) equations. Shorthand terms can be found in the appendix,

Equation A.1, along with Jacobians. B_1 will first be expanded using the convective Jacobian:

$$\begin{aligned} B_1 &= \Psi^\mathsf{T}\left(\vec{A}^c - \bar{\bar{I}}\vec{u}_\Omega\right)\delta U\cdot\vec{n} = \Psi^\mathsf{T}\left(\vec{A}^c - \bar{\bar{I}}\vec{u}_\Omega\right)\cdot\vec{n}M\delta V \\ &= \psi^\mathsf{T}\begin{Bmatrix} -\vec{u}_\Omega\cdot\vec{n} & \vec{n}^\mathsf{T} & 0 \\ -(\vec{v}\cdot\vec{n})\vec{v}+\phi\vec{n} & \left[\bar{\bar{I}}(\vec{v}\cdot\vec{n}-\vec{u}_\Omega\cdot\vec{n})+\vec{v}\otimes\vec{n}-a_0\vec{n}\otimes\vec{v}\right] & a_0\vec{n} \\ (\vec{v}\cdot\vec{n})(\phi-H) & -a_0\vec{v}^\mathsf{T}(\vec{v}\cdot\vec{n})+H\vec{n}^\mathsf{T} & \gamma(\vec{v}\cdot\vec{n})-\vec{u}_\Omega\cdot\vec{n} \end{Bmatrix} M\delta V. \\ &= \psi^\mathsf{T}\begin{Bmatrix} v_{nr} & \rho\vec{n}^\mathsf{T} & 0 \\ (v_{nr})\vec{v} & \rho\left[\bar{\bar{I}}v_{nr}+\vec{v}\otimes\vec{n}\right] & \vec{n} \\ \frac{\phi}{a_0}(v_{nr}) & \rho(v_{nr})\vec{v}^\mathsf{T}+\rho H\vec{n}^\mathsf{T} & v_n+\frac{1}{a_0}(v_{nr}) \end{Bmatrix}\delta V, \end{aligned} \tag{3.17}$$

Simplification of B_1 on Solid Walls In either viscous or inviscid flow Equation 3.17 can be simplified by applying the flow tangency boundary condition $\vec{v}\cdot\vec{n}-\vec{u}_\Omega\cdot\vec{n}=0$:

$$\begin{aligned} \Psi^\mathsf{T}\left(\vec{A}^c - \bar{\bar{I}}\vec{u}_\Omega\right)\delta U\cdot\vec{n} &= \psi^\mathsf{T}\begin{Bmatrix} 0 & \rho\vec{n}^\mathsf{T} & 0 \\ \vec{0} & \rho\left[\vec{v}\otimes\vec{n}\right] & \vec{n} \\ 0 & \rho H\vec{n}^\mathsf{T} & v_n \end{Bmatrix}\delta V \\ &= \begin{Bmatrix} 0 \\ (\rho\psi_\rho+\rho\vec{v}\cdot\vec{\varphi}+\rho H\psi_{\rho E})\vec{n} \\ \vec{\varphi}\cdot\vec{n}+\psi_{\rho E}v_n \end{Bmatrix}^\mathsf{T}\begin{Bmatrix} \delta\rho \\ \delta\vec{v} \\ \delta P \end{Bmatrix} \\ &= (\rho\psi_\rho+\rho\vec{v}\cdot\vec{\varphi}+\rho H\psi_{\rho E})(\delta\vec{v}\cdot\vec{n})+(\vec{\varphi}\cdot\vec{n}+\psi_{\rho E}v_n)\delta P. \end{aligned} \tag{3.18}$$

Finally,

$$B_1 = \vartheta(\delta\vec{v}\cdot\vec{n})+(\vec{\varphi}\cdot\vec{n}+\psi_{\rho E}v_n)\delta P\,, \tag{3.19}$$

The shorthand term $\vartheta=(\rho\psi_\rho+\rho\vec{v}\cdot\vec{\varphi}+\rho H\psi_{\rho E})$ has also been introduced here. Note that this result is the same for both viscous and inviscid flows, as well as for symmetry planes where flow tangency is also imposed. In stationary domains, where $\vec{u}_\Omega=\vec{0}$ and the flow tangency boundary condition becomes $v_n=0$, the equation reduces to:

$$B_{1,\vec{u}_\Omega=\vec{0}} = \vartheta(\delta\vec{v}\cdot\vec{n})+(\vec{\varphi}\cdot\vec{n}+\psi_{\rho E})\delta P. \tag{3.20}$$

Simplification of B_1 on Farfield, Inflow, and Outflow Boundaries At open boundaries the characteristics of the equation must be considered. In the boundary conditions of the direct problem, the sign of the eigenvalues of the convective flux Jacobian determines how the solution

state is updated. The linearized boundary condition is expressed as:

$$(\delta W)_+ = 0 \text{ on } \Gamma_\infty, \Gamma_{in}, \ \& \Gamma_e. \tag{3.21}$$

For the terms in B_1, this means that under some conditions components of δV will be set to 0.

Expansion & Simplification of B_2 The boundary integral B_2 is generally only considered on solid surfaces, as the effect of viscosity is neglected at other boundaries. Therefore B_2 will only be considered on a no-slip wall S. In order to simplify this term, we will need the variation of the viscous flux vector, the property that $\delta(\nabla U) = \nabla(\delta U)$ in a continuum, and the application of the no-slip boundary condition, $\vec{v} - \vec{u}_\Omega = \vec{0}$. This variation, under a frozen viscosity assumption ($\delta\mu = 0$), is as follows:

$$\delta(\mu^k_{tot}\vec{F}^{vk}) = \mu^k_{tot}\frac{\partial \vec{F}^{vk}}{\partial U}\delta U + \mu^k_{tot}\frac{\partial \vec{F}^{vk}}{\partial(\nabla U)} \cdot \delta(\nabla U) = \mu^k_{tot}\vec{A}^{vk}\delta U + \mu^k_{tot}\bar{\bar{D}}^{vk} \cdot \delta(\nabla U). \tag{3.22}$$

B_2 can now be simplified:

$$\begin{aligned}
B_2 &= \Psi^\mathsf{T}\mu^k_{tot}\vec{A}^{vk}\delta U \cdot \vec{n} + \Psi^\mathsf{T}\mu^k_{tot}\bar{\bar{D}}^{vk} \cdot \nabla(\delta U) \cdot \vec{n} \\
&= \Psi^\mathsf{T}\left[\mu^k_{tot}\vec{A}^{vk}\delta U + \mu^k_{tot}\bar{\bar{D}}^{vk} \cdot \nabla(\delta U)\right] \cdot \vec{n} \\
&= \Psi^\mathsf{T}\delta(\mu^k_{tot}\vec{F}^{vk}) \cdot \vec{n} \\
&= \Psi^\mathsf{T}\delta(\mu^1_{tot}\vec{F}^{v1}) \cdot \vec{n} + \Psi^\mathsf{T}\delta(\mu^2_{tot}\vec{F}^{v2}) \cdot \vec{n} \\
&= \Psi^\mathsf{T}\delta\begin{Bmatrix} \cdot \\ \bar{\bar{\sigma}} \\ \bar{\bar{\sigma}} \cdot \vec{v} \end{Bmatrix} \cdot \vec{n} + \Psi^\mathsf{T}\delta\begin{Bmatrix} \cdot \\ \cdot \\ \mu^2_{tot}c_p\nabla T \end{Bmatrix} \cdot \vec{n} \\
&= \{\psi_\rho, \vec{\varphi}, \psi_{\rho E}\}\begin{Bmatrix} \cdot \\ \delta\bar{\bar{\sigma}} \cdot \vec{n} \\ \delta(\bar{\bar{\sigma}} \cdot \vec{v}) \cdot \vec{n} \end{Bmatrix} + \{\psi_\rho, \vec{\varphi}, \psi_{\rho E}\}\begin{Bmatrix} \cdot \\ \cdot \\ \mu^2_{tot}c_p\delta(\nabla T) \cdot \vec{n} \end{Bmatrix} \\
&= \vec{\varphi} \cdot \delta\bar{\bar{\sigma}} \cdot \vec{n} + \psi_{\rho E}\delta(\bar{\bar{\sigma}} \cdot \vec{v}) \cdot \vec{n} + \psi_{\rho E}\mu^2_{tot}c_p\delta(\nabla T) \cdot \vec{n},
\end{aligned} \tag{3.23}$$

Resulting in the following expression for B_2:

$$B_2 = \vec{\varphi} \cdot \delta\bar{\bar{\sigma}} \cdot \vec{n} + \psi_{\rho E}\vec{v} \cdot \delta\bar{\bar{\sigma}} \cdot \vec{n} + \psi_{\rho E}\delta\vec{v} \cdot \bar{\bar{\sigma}} \cdot \vec{n} + \psi_{\rho E}\mu^2_{tot}c_p\partial_n(\delta T). \tag{3.24}$$

For flows on stationary domains, $\vec{u}_\Omega = 0$ and $\vec{v} = 0$ on the surface. For stationary domains, the

equation can be further simplified:

$$B_{2,\vec{u}_\Omega=\vec{0}} = \vec{\varphi}\cdot\delta\bar{\bar{\sigma}}\cdot\vec{n} + \psi_{\rho E}\delta\vec{v}\cdot\bar{\bar{\sigma}}\cdot\vec{n} + \psi_{\rho E}\mu_{tot}^2 c_p\partial_n(\delta T). \tag{3.25}$$

The Navier-Stokes or RANS equations generally also add a condition on the temperature, usually either an isothermal, adiabatic, or fixed heat flux condition. Which thermal condition is chosen further simplifies B_2.

Expansion & Simplification of B_3 Similar to terms of B_2, B_3 will only be considered on a no-slip solid wall S. B_3 includes components related to viscous stresses as well as to heat transfer, and these terms will be considered separately here as B_3^1 and B_3^2 respectively:

$$B_3 = B_3^k = B_3^1 + B_3^2. \tag{3.26}$$

Expanding the Viscous Stress Component B_3^1 The Jacobian of the viscous stresses can be expressed compactly, using $\vec{\delta}_i = \{\delta_{i1},\delta_{i2},\delta_{i3}\}^\mathsf{T}$ as:

$$\bar{\bar{D}}_{ij}^{\nu 1} = \begin{bmatrix} 0 & \vec{0}^\mathsf{T} & 0 \\ -v_i\vec{\delta}_j + \frac{2}{3}v_j\vec{\delta}_i - \vec{v}\delta_{ij} & \vec{\delta}_j\otimes\vec{\delta}_i - \frac{2}{3}\vec{\delta}_i\otimes\vec{\delta}_j + \bar{\bar{I}}\delta_{ij} & \vec{0} \\ -\frac{1}{3}v_iv_j - |\vec{v}|^2\delta_{ij} & v_j\vec{\delta}_i^\mathsf{T} - \frac{2}{3}v_i\vec{\delta}_j^\mathsf{T} + \vec{v}^T\delta_{ij} & 0 \end{bmatrix}. \tag{3.27}$$

Expanding

$$\begin{aligned}
\bar{\bar{D}}_{ij}^{\nu 1}\delta U &= \bar{\bar{D}}_{ij}^{\nu 1} M\delta V \\
&= \bar{\bar{D}}_{ij}^{\nu 1}\begin{Bmatrix} \delta\rho \\ \vec{v}\delta\rho + \rho\delta\vec{v} \\ \frac{|\vec{v}|^2}{2}\delta\rho + \rho\vec{v}\cdot\delta\vec{v} + \frac{1}{\gamma-1}\delta P \end{Bmatrix} \\
&= \begin{bmatrix} 0 \\ (-v_i\vec{\delta}_j + \frac{2}{3}v_j\vec{\delta}_i - \vec{v}\delta_{ij})\delta\rho + \left[\vec{\delta}_j\otimes\vec{\delta}_i - \frac{2}{3}\vec{\delta}_i\otimes\vec{\delta}_j + \bar{\bar{I}}\delta_{ij}\right](\vec{v}\delta\rho + \rho\delta\vec{v}) \\ (-\frac{1}{3}v_iv_j - |\vec{v}|^2\delta_{ij})\delta\rho + \left\{v_j\vec{\delta}_i^\mathsf{T} - \frac{2}{3}v_i\vec{\delta}_j^\mathsf{T} + \vec{v}^\mathsf{T}\delta_{ij}\right\}\cdot(\vec{v}\delta\rho + \rho\delta\vec{v}) \end{bmatrix} \\
&= \begin{bmatrix} 0 \\ \left\{-v_i\vec{\delta}_j + \frac{2}{3}v_j\vec{\delta}_i - \vec{v}\delta_{ij}\right\}\delta\rho + \left\{\vec{\delta}_jv_i - \frac{2}{3}\vec{\delta}_iv + \vec{v}\delta_{ij}\right\}\delta\rho + \left\{\vec{\delta}_j\delta v_i - \frac{2}{3}\vec{\delta}_i\delta v_j + \delta_{ij}\delta\vec{v}\right\}\rho \\ -\frac{1}{3}v_iv_j\delta\rho - |\vec{v}|^2\delta_{ij}\delta\rho + \left(v_jv_i - \frac{2}{3}v_iv_j + |\vec{v}|^2\delta_{ij}\right)\delta\rho + \rho\left(v_j\delta v_i - \frac{2}{3}v_i\delta v_j + \vec{v}\cdot\delta\vec{v}\delta_{ij}\right) \end{bmatrix} \\
&= \begin{bmatrix} 0 \\ \left\{\vec{\delta}_j\delta v_i - \frac{2}{3}\vec{\delta}_i\delta v_j + \delta_{ij}\delta\vec{v}\right\}\rho \\ \left(v_jv_i - v_iv_j\right)\delta\rho + \rho\left(v_j\delta v_i - \frac{2}{3}v_i\delta v_j + \vec{v}\cdot\delta\vec{v}\delta_{ij}\right) \end{bmatrix}.
\end{aligned} \tag{3.28}$$

Canceling further,

$$\bar{\bar{D}}_{ij}^{\nu 1}\delta U = \begin{bmatrix} 0 \\ \left\{\vec{\delta}_j \delta v_i - \frac{2}{3}\vec{\delta}_i \delta v_j + \delta_{ij}\delta\vec{v}\right\}\rho \\ \rho\left(v_j\delta v_i - \frac{2}{3}v_i\delta v_j + \vec{v}\cdot\delta\vec{v}\delta_{ij}\right) \end{bmatrix}. \tag{3.29}$$

$$\begin{aligned} B_3^1 &= \nabla\Psi^{\mathsf{T}}\cdot\mu_{tot}^1\bar{\bar{D}}^{v1}\delta U\cdot\vec{n} \\ &= \mu_{tot}^1\nabla\Psi^{\mathsf{T}}\cdot\bar{\bar{D}}^{\nu 1}\delta U\cdot\vec{n} \\ &= \mu_{tot}\nabla\begin{Bmatrix}\psi_\rho \\ \varphi \\ \psi_{\rho E}\end{Bmatrix}^{\mathsf{T}}\cdot\begin{bmatrix} 0 \\ \left\{\vec{\delta}_j\delta v_i - \frac{2}{3}\vec{\delta}_i\delta v_j + \delta_{ij}\delta\vec{v}\right\}\rho \\ \rho\left(v_j\delta v_i - \frac{2}{3}v_i\delta v_j + \vec{v}\cdot\delta\vec{v}\delta_{ij}\right)\end{bmatrix}\cdot\vec{n} \\ &= \rho\mu_{tot}^1\begin{Bmatrix} 0 \\ \nabla\vec{\varphi}\cdot\delta\vec{v} - \delta\vec{v}\cdot\frac{2}{3}\bar{\bar{I}}\nabla\cdot\vec{\varphi} + \nabla\vec{\varphi}^{\mathsf{T}}\delta\vec{v} \\ \nabla\psi_{\rho E}(\vec{v}\cdot\delta\vec{v}) - \frac{2}{3}\bar{\bar{I}}(\nabla\psi_{\rho E}\cdot\vec{v}) + (\delta\vec{v}\cdot\nabla\psi_{\rho E})\vec{v}\end{Bmatrix}\cdot\vec{n}. \end{aligned} \tag{3.30}$$

The result can be rearranged with the introduction of convenient shorthand terms:

$$\begin{aligned} \bar{\bar{\Sigma}}^{\varphi} &= \mu_{tot}^1(\nabla\vec{\varphi} + \nabla\vec{\varphi}^{\mathsf{T}} - \frac{2}{3}\bar{\bar{I}}\nabla\cdot\vec{\varphi}) \\ \bar{\bar{\Sigma}}^{\psi_{\rho E}} &= \mu_{tot}^1(\nabla\psi_{\rho E}\vec{v} + \nabla\psi_{\rho E}\vec{v}^{\mathsf{T}} - \frac{2}{3}\bar{\bar{I}}\nabla\psi_{\rho E}\cdot\vec{v}), \end{aligned} \tag{3.31}$$

which reduces B_3^1 to:

$$\begin{aligned} B_3^1 &= \mu_{tot}^1\rho\left[\vec{n}\cdot(\nabla\vec{\varphi} + \nabla\vec{\varphi}^{\mathsf{T}} - \frac{2}{3}\bar{\bar{I}}\nabla\cdot\vec{\varphi})\cdot\delta\vec{v} + \vec{n}\cdot(\nabla\psi_{\rho E}\vec{v} + \nabla\psi_{\rho E}\vec{v}^{\mathsf{T}} - \frac{2}{3}\bar{\bar{I}}\nabla\psi_{\rho E}\cdot\vec{v})\cdot\delta\vec{v}\right] \\ &= \rho\vec{n}\cdot\bar{\bar{\Sigma}}^{\varphi}\cdot\delta\vec{v} + \vec{n}\cdot\bar{\bar{\Sigma}}^{\psi_{\rho E}}\cdot\delta\vec{v} \\ &= \rho\vec{n}\cdot\left(\bar{\bar{\Sigma}}^{\varphi} + \bar{\bar{\Sigma}}^{\psi_{\rho E}}\right)\cdot\delta\vec{v}. \end{aligned} \tag{3.32}$$

Expanding the Heat Transfer Component B_3^2 The Jacobian of the heat transfer component can be expanded as follows:

$$\bar{\bar{D}}_{ij}^{\nu 2} = \frac{\gamma}{\rho}\begin{pmatrix} 0 & \vec{0}^T & 0 \\ \vec{0} & \bar{\bar{0}} & \vec{0} \\ \frac{\delta_{ij}}{a_0}\left(\phi - \frac{P}{\rho}\right) & -\delta_{ij}\vec{v}^{\mathsf{T}} & \delta_{ij}\end{pmatrix}. \tag{3.33}$$

The term B_3^2 can be expanded in a similar way as B_3^1:

$$\begin{aligned}
B_3^2 &= \nabla\Psi^{\mathsf{T}} \cdot \mu_{tot}^2 \bar{\bar{D}}^{\nu 2} \delta U \cdot \vec{n} \\
&= \mu_{tot}^2 \nabla \begin{Bmatrix} \psi_\rho \\ \varphi \\ \psi_{\rho E} \end{Bmatrix}^{\mathsf{T}} \cdot \delta_{ij} \frac{\gamma}{\rho} \begin{pmatrix} 0 & \vec{0}^T & 0 \\ \vec{0} & \bar{\bar{0}} & \vec{0} \\ \frac{1}{a_0}\left(\phi - \frac{P}{\rho}\right) & -\vec{v}^{\mathsf{T}} & 1 \end{pmatrix} \begin{Bmatrix} \delta\rho \\ \vec{v}\delta\rho + \rho\delta\vec{v} \\ \frac{|\vec{v}|^2}{2}\delta\rho + \rho\vec{v}\cdot\delta\vec{v} + \frac{1}{\gamma-1}\delta P \end{Bmatrix} \cdot \vec{n} \\
&= \mu_{tot}^2 \nabla\left(\psi_{\rho E}\right) \cdot \delta_{ij} \frac{\gamma}{\rho}\left(\frac{\delta\rho}{a_0}\left(\phi - \frac{P}{\rho}\right) + \vec{v}\cdot\left(\vec{v}\delta\rho + \rho\delta\vec{v}\right) + \left(\frac{|\vec{v}|^2}{2}\delta\rho + \rho\vec{v}\cdot\delta\vec{v} + \frac{1}{\gamma-1}\delta P\right)\right) \cdot \vec{n} \\
&= \frac{\gamma\mu_{tot}^2 \nabla\left(\psi_{\rho E}\right)}{\rho}\left(\delta\rho\left(\frac{|\vec{v}|^2}{2} - \frac{P}{\rho(\gamma-1)}\right) - \left(|\vec{v}|^2\delta\rho + \rho\vec{v}\cdot\delta\vec{v}\right) + \left(\frac{|\vec{v}|^2\delta\rho}{2} + \rho\vec{v}\cdot\delta\vec{v} + \frac{\delta P}{\gamma-1}\right)\right) \cdot \vec{n} \\
&= \mu_{tot}^2 \partial_n\left(\psi_{\rho E}\right) \frac{\gamma}{\rho}\left(\left(-\frac{P\delta\rho}{\rho(\gamma-1)}\right) + \left(\frac{\delta P}{\gamma-1}\right)\right).
\end{aligned} \tag{3.34}$$

From the equation of state,

$$\begin{aligned}
\delta T &= \delta\left(\frac{\rho}{PR}\right) = \delta\left(\frac{\gamma}{c_p(\gamma-1)}\frac{P}{\rho}\right) \\
\delta T &= \frac{\gamma}{c_p(\gamma-1)}\frac{1}{\rho}\delta P - \frac{\gamma}{c_p(\gamma-1)}\frac{P}{\rho^2}\delta\rho,
\end{aligned} \tag{3.35}$$

which leads to:

$$B_3^1 = \mu_{tot}^2 \partial_n \psi_{\rho E} c_p \delta T. \tag{3.36}$$

The total contribution from B_3 can now be expressed concisely as:

$$B_3 = B_3^1 + B_3^2 = \vec{n} \cdot \left(\bar{\bar{\Sigma}}^{\varphi} + \bar{\bar{\Sigma}}^{\psi_{\rho E}}\right) \cdot \delta\vec{v} + \mu_{tot}^2 c_p \partial_n(\psi_{\rho E})\delta T \tag{3.37}$$

Combining Terms of δR

Combining the terms from Equation 3.19, 3.24, and 3.37, the variation of the augmented functional is:

$$\begin{aligned}
\delta\mathcal{J} = \delta J \\
&- \int_\Omega \left[-\nabla\Psi^\mathsf{T} \cdot \left(\vec{A}^c - \bar{\bar{I}}_5 \vec{u}_\Omega - \mu^k_{tot} \vec{A}^{vk} \right) - \nabla \cdot \left(\nabla\Psi^\mathsf{T} \cdot \mu^k_{tot} \bar{\bar{D}}^{vk} \right) - \Psi^\mathsf{T} \frac{\partial \mathcal{Q}}{\partial U} \right] \delta U \, d\Omega \\
&- \int_S \vartheta(\delta\vec{v} \cdot \vec{n}) + (\vec{\varphi} \cdot \vec{n} + \psi_{\rho E} v_n) \delta P \, ds \\
&+ \int_S \vec{\varphi} \cdot \delta\bar{\bar{\sigma}} \cdot \vec{n} + \psi_{\rho E} \vec{v} \cdot \delta\bar{\bar{\sigma}} \cdot \vec{n} + \psi_{\rho E} \delta\vec{v} \cdot \bar{\bar{\sigma}} \cdot \vec{n} + \psi_{\rho E} \mu^2_{tot} c_p \partial_n(\delta T) \, ds \\
&- \int_S \vec{n} \cdot \left(\bar{\bar{\Sigma}}^{\varphi} + \bar{\bar{\Sigma}}^{\psi_{\rho E}} \right) \cdot \delta\vec{v} + \mu^2_{tot} c_p \partial_n(\psi_{\rho E}) \delta T \, ds - \int_\Gamma \left(\vec{A}^c - \bar{\bar{I}}_5 \vec{u}_\Omega \right) \cdot \vec{n} M \delta V \, ds.
\end{aligned} \tag{3.38}$$

The volume terms are eliminated by setting

$$\nabla\Psi^\mathsf{T} \cdot \left(\vec{A}^c - \bar{\bar{I}}_5 \vec{u}_\Omega + \mu^k_{tot} \vec{A}^{vk} \right) + \nabla \cdot \left(\nabla\Psi^\mathsf{T} \cdot \mu^k_{tot} \bar{\bar{D}}^{vk} \right) + \Psi^\mathsf{T} \tfrac{\partial \mathcal{Q}}{\partial U} = 0 \quad \text{on}\Omega \quad , \tag{3.39}$$

which is the PDE that defines the adjoint equations. Equation 3.39 is the system of equations that the adjoint variables Ψ will satisfy in the volume of the computational domain, subject to boundary conditions that combine terms from δJ and the surface integrals from Equation 3.39. The following sections will expand the development of the boundary conditions.

3.1.4 Linearizing Boundary Conditions

In order to ensure that the constraint expressed by the Lagrangian ($\mathcal{R}(U) = 0$) is satisfied, the variations δU must be restricted to physically realizable solutions. This is achieved by using the linearized form of the direct problem's boundary conditions to introduce relationships between the flow variations at the boundaries. A selection of boundary conditions will now be linearized: flow tangency, no-slip conditions, adiabatic wall, isothermal wall, and characteristic-based conditions.

Flow Tangency Boundary Condition

In inviscid flow, the boundary condition usually applied at the wall is a flow tangency boundary condition. The same equations are also valid for symmetry planes:

$$(\vec{v} - \vec{u}_\Omega) \cdot \vec{n} = 0 \, \text{on S.} \tag{3.40}$$

Linearizing this boundary condition with respect to a small perturbation of the surface δS leads to:

$$\begin{aligned}(\vec{v}-\vec{u}_\Omega)' &= (\vec{v}-\vec{u}_\Omega)+\delta(\vec{v}-\vec{u}_\Omega)+\partial_n(\vec{v}-\vec{u}_\Omega)\delta S \\ (\vec{n})' &= \vec{n}+\delta\vec{n} \\ (\vec{v}-\vec{u}_\Omega)'\cdot(\vec{n})' &= \{(\vec{v}-\vec{u}_\Omega)+\delta\vec{v}-\delta\vec{u}_\Omega+\partial_n(\vec{v}-\vec{u}_\Omega)\delta S\}\cdot(\vec{n}+\delta\vec{n}) \\ &= (\vec{v}-\vec{u}_\Omega)\cdot\delta\vec{n}+\delta\vec{v}\cdot\vec{n}+\partial_n(\vec{v}-\vec{u}_\Omega)\delta S\cdot\vec{n}.\end{aligned} \tag{3.41}$$

In order to simplify the last line of Equation 3.41 we have used Equation 3.40, neglected products of variations, and applied $\delta\vec{u}_\Omega = \vec{0}$ to apply a constant grid velocity. Rearranging this equation and applying $\delta\vec{n} = -\nabla_S(\delta S)$:

$$\delta\vec{v}\cdot\vec{n} = (\vec{v}-\vec{u}_\Omega)\cdot\nabla_S(\delta S)-\partial_n(\vec{v}-\vec{u}_\Omega)\delta S\cdot\vec{n}. \tag{3.42}$$

In the case of a stationary grid,

$$\delta\vec{v}\cdot\vec{n} = \vec{v}\cdot\nabla_S(\delta S)-\partial_n\vec{v}\delta S\cdot\vec{n}. \tag{3.43}$$

No-Slip Wall Boundary

For a solid wall in viscous flow, a no-slip boundary is usually applied,

$$\vec{v}-\vec{u}_\Omega = \vec{0}. \tag{3.44}$$

In hypersonic flow, a slip wall is sometimes used in regimes where rarefied gas is expected. This is outside the scope of this work. The linearization of the no-slip condition proceeds as follows:

$$\begin{aligned}(\vec{v}-\vec{u}_\Omega)' &= (\vec{v}-\vec{u}_\Omega)+\delta(\vec{v}-\vec{u}_\Omega)+\partial_n(\vec{v}-\vec{u}_\Omega)\delta S \\ (\vec{v}-\vec{u}_\Omega)'-(\vec{v}-\vec{u}_\Omega) &= \delta\vec{v}+\partial_n(\vec{v}-\vec{u}_\Omega)\delta S \\ \delta\vec{v} &= -\partial_n(\vec{v}-\vec{u}_\Omega)\delta S,\end{aligned} \tag{3.45}$$

where the perturbed velocity $(\vec{v}-\vec{u}_\Omega)'$ is set to zero in order to satisfy the original boundary condition, and a constant grid velocity is used: $\delta\vec{u}_\Omega = 0$.

Isothermal Boundary

For an isothermal condition, $T = T_w$, as so the linearization with respect to a small perturbation δS is:

$$T' = T+\delta T+\partial_n(T)\delta S. \tag{3.46}$$

Since $T' = T$ to satisfy the direct problem boundary condition,

$$\delta T = -\partial_n(T)\delta S \text{ on } S. \tag{3.47}$$

The heat flux term from Equation 3.38 is now:

$$\psi_{\rho E}\mu_{tot}^2 c_p \partial_n(\delta T) = -\psi_{\rho E}\mu_{tot}^2 c_p \partial_n^2(T)\delta S, \tag{3.48}$$

which will be used when forming the adjoint boundary conditions after these expressions are combined with the variation of the functional.

Constant Heat Flux or Adiabatic Boundary

For a constant heat flux,

$$\partial_n T = \vec{n}\cdot\nabla T = q_n \text{ on } S, \tag{3.49}$$

where q_n is the known heat flux or 0 for the adiabatic condition. Linearizing this relationship and assuming that the product of variations are negligible,

$$\begin{aligned}(\nabla T)'\cdot(\vec{n})' &= [\nabla T + \delta(\nabla T) + \partial_n(\nabla T)\delta S]\cdot(\vec{n}+\delta\vec{n}) \\ &= \nabla T\cdot\vec{n} + \nabla T\cdot\delta\vec{n} + \delta(\nabla T)\cdot\vec{n} + \partial_n^2(T)\delta S = \frac{q_n}{c_p} = \nabla T\cdot\vec{n}.\end{aligned} \tag{3.50}$$

This expression is equal to q_n because the heat flux must remain constant to satisfy the direct problem boundary condition. Rearranging and using the relationship $\delta\vec{n} = -\nabla_S(\delta S)$ on small deformations, as well as $\delta(\nabla T) = \nabla(\delta T)$:

$$\begin{aligned}\delta(\nabla T)\cdot\vec{n} &= -(\nabla T)\cdot\delta\vec{n} - \partial_n^2(T)\delta S \text{ on } S \\ \partial_n(\delta T) &= \nabla T\cdot\nabla_S(\delta S) - \partial_n^2(T)\delta S \text{ on } S.\end{aligned} \tag{3.51}$$

The heat flux term from Equation 3.38 is now:

$$\psi_{\rho E}\mu_{tot}^2 c_p \partial_n(\delta T) = \psi_{\rho E}\mu_{tot}^2 c_p \left(\nabla T\cdot\nabla_S(\delta S) - \partial_n^2(T)\delta S\right). \tag{3.52}$$

Farfield, Inflow, & Outflow Boundaries

At farfield, inflow, and outflow boundaries the characteristics of the equation must be considered. In the boundary conditions of the direct problem, the sign of the eigenvalues of the convective flux Jacobian determines how the solution state is updated. The linearized boundary condition is

Table 3.1: Linearized Residuals of Fluid Flow Governing Equations

Equation	Residual	Linearized Form
Steady RANS	$\mathcal{R}(U) = \nabla \cdot \vec{F}^c$ $-\nabla \cdot \left(\mu_{tot}^1 \vec{F}^{v1} + \mu_{tot}^2 \vec{F}^{v2}\right) - \mathcal{Q} = 0$	$\delta\mathcal{R}(U, \nabla U) = \nabla \cdot \left(\vec{A}^c - \mu_{tot}^k \vec{A}^{vk}\right) \delta U$ $-\nabla \cdot \vec{F}^{vk} \frac{\partial \mu_{tot}^k}{\partial U} \delta U$ $-\nabla \cdot \mu_{tot}^k \bar{\bar{D}}^{vk} \delta(\nabla U) - \frac{\partial \mathcal{Q}}{\partial U} \delta U.$
Steady Euler	$\mathcal{R}(U) = \nabla \cdot \vec{F}^c - \mathcal{Q} = 0$	$\delta\mathcal{R}(U, \nabla U) = \nabla \cdot \left(\vec{A}^c\right) \delta U - \frac{\partial \mathcal{Q}}{\partial U} \delta U$.

expressed as:

$$\begin{aligned} (W)_+ &= W_\infty \\ (\delta W)_+ &= 0. \end{aligned} \tag{3.53}$$

The way in which this condition is applied depends on the number of characteristics entering or exiting the flow and what physically realizable variations can occur. The characteristic directions describe how information flows through the solution, and they result from the diagonalization of the convective Jacobian. When the flow is locally supersonic in a particular directions, all characteristic speeds have the same sign. This means that at an inflow, all properties of the flow are specified by the inflow boundary conditions, while at the outflow the flow properties are determined by the volume solution. Correspondingly, in the adjoint equation at a supersonic inflow none of the variations are arbitrary because all flow properties have been specified, and at the outflow all variations are arbitrary. When the flow is locally subsonic, one of the characteristic speeds of the flow solution is negative, moving in the opposite direction of the fluid velocity. At an inflow, this means that if $\dot{m}$ is prescribed, P is determined by the volume solution and is not directly controlled by the boundary condition. This is summarized further in Table 3.2.

Summary of Linearized Equations

The linearized form of the direct (fluid flow) problem will be used in the derivation of the adjoint problem. Preceding sections showed the linearization process, and this section tabulates the resulting equations and reviews the assumptions used in their derivation. The linearized form of the residual of the fluid flow governing equations are shown in Table 3.1. The linearization of the RANS equations is shown in Equation 3.9, and the linearization of the Euler equations is arrived at by removing the viscous terms. The linearized form of the boundary conditions of the fluid flow governing equations are shown in Table 3.2. In this table k refers to the number of conservative variables of the problem; for a 2-dimensional problem $k = 4$, and for a 3-dimensional problem $k = 5$. This table includes the conditions relevant to both the RANS equation and the Euler equations. For example, the flow tangency condition is applied both at solid walls in Euler flow and at symmetry planes in either

Table 3.2: Linearized Boundary Conditions for Fluid Flow

Boundary Type	Boundary Condition	Linearized Form
Flow Tangency	$(\vec{v}-\vec{u}_\Omega)\cdot\vec{n}=0$	$\delta\vec{v}\cdot\vec{n}=(\vec{v}-\vec{u}_\Omega)\cdot\nabla_S(\delta S)-\partial_n(\vec{v}-\vec{u}_\Omega)\delta S\cdot\vec{n}$
No-Slip Wall	$\vec{v}-\vec{u}_\Omega=\vec{0}$	$\delta\vec{v}=-\partial_n(\vec{v}-\vec{u}_\Omega)\delta S$
Adiabatic Wall	$\partial_n T=\vec{n}\cdot\nabla T=0$	$\partial_n(\delta T)=\nabla T\cdot\nabla_S(\delta S)-\partial_n^2(T)\delta S$
Constant q_n Wall	$\partial_n T=\vec{n}\cdot\nabla T=\frac{q_n}{c_p}$	$\partial_n(\delta T)=\nabla T\cdot\nabla_S(\delta S)-\partial_n^2(T)\delta S$
Isothermal Wall	$T=T_w$	$\delta T=-\partial_n(T)\delta S$
Farfield, Inflow, Outflow	$(W)_+=W_\infty$	$(\delta W)_+=0.$
Subsonic Outflow: prescribed pressure	1 value prescribed $P=P_e$	$k-1$ variations arbitrary $\delta P=0$
Subsonic Inflow: $\dot{m}$ prescribed	$k-1$ values prescribed $\rho\vec{v}=\rho_0\vec{v}_0$	1 variation arbitrary $\delta\rho=\delta\vec{v}=0$
Supersonic Inflow:	k values prescribed	no variation arbitrary $\delta\rho=\delta\vec{v}=\delta P=0$
Supersonic Outflow:	0 values prescribed	all variations arbitrary

viscous or inviscid flow. The assumptions applied in the linearization of the equations are: negligible products of variations, small δS (allowing the use of $\delta\vec{n}=-\nabla_S(\delta S)$), $\delta(\nabla T)=\nabla(\delta T)$, constant grid velocity $\delta\vec{u}_\Omega=\vec{0}$, and frozen viscosity ($\delta\mu=0$).

3.1.5 Functional Variations

This section will review the variation of a selection of functionals that have been addressed in literature. The derivation of the adjoint equations requires the variation of the chosen functional. This section expands the variations of a selection of functionals. A summary table is provided at the end of this section for reference. Common objective functions for aerospace applications can be expressed in terms of the surface force $\vec{f}$, the temperature T, the normal gradient of $\partial_n T$, and the normal vector $\vec{n}$. It is therefore convenient to express the functional, and its variation, in terms of these values. At outflow boundaries, it is convenient to express the functional directly in terms of the primitive variables V. Including both surface terms and outflow terms:

$$J=\int_S j(\vec{f},T,\partial_n T,\vec{n})ds+\int_{\Gamma_e} j(V)ds. \tag{3.54}$$

Expanding the variation of the functional, keeping the outflow surface undeformed with $\delta\Gamma_e=0$,

and observing that $\delta\vec{n} = -\nabla_S(\delta S)$ holds for small deformations:[89]

$$\begin{aligned}\delta J = &\int_S \left(\frac{\partial j}{\partial \vec{f}} \cdot \partial_n \vec{f} + \frac{\partial j}{\partial T}\partial_n T + \frac{\partial j}{\partial(\partial_n T)}\partial_n^2 T - 2H_m j \right) \delta S ds + \\ &\int_S \left(\frac{\partial j}{\partial \vec{f}} \cdot \delta\vec{f} + \frac{\partial j}{\partial T}\delta T + \frac{\partial j}{\partial(\partial_n T)}\delta(\partial_n T) - \frac{\partial j}{\partial \vec{n}} \cdot \nabla_S(\delta S) \right) ds + \int_{\Gamma_e} \frac{\partial j}{\partial V}\delta V ds.\end{aligned} \tag{3.55}$$

H_m is the mean curvature of S, $(\kappa_1 + \kappa_2)/2$, where κ_i are curvatures in orthogonal directions on the surface. The terms multiplying δS will become part of the surface sensitivity, and the remaining variational terms will be eliminated through the boundary conditions.

Variation of the Functionals on S

To simplify the situation, for now only functionals on the solid wall S will be considered. In this case, the variation of the functional is now:

$$\delta J = \int_{\delta S} j(\vec{f}, T, \partial_n T, \vec{n}) ds + \int_S \delta j(\vec{f}, T, \partial_n T, \vec{n}) ds. \tag{3.56}$$

Using the relationships from Equation 3.7, the first term of Equation 3.56 is:

$$\begin{aligned}\int_{\delta S} j(\vec{f}, T, \partial_n T, \vec{n}) ds &= \int_S (\partial_n j - 2H_m j)\delta S ds \\ &= \int_S \left(\frac{\partial j}{\partial \vec{f}} \cdot \partial_n \vec{f} + \frac{\partial j}{\partial T}\partial_n T + \frac{\partial j}{\partial(\partial_n T)}\partial_n^2 T - 2H_m j \right) \delta S ds.\end{aligned} \tag{3.57}$$

The kernal of the second term from Equation 3.56 can be expanded with the chain rule the relation from Equation 3.7 for $\delta\vec{n}$:

$$\delta j(\vec{f}, T, \partial_n T, \vec{n}) = \frac{\partial j}{\partial \vec{f}} \cdot \delta\vec{f} + \frac{\partial j}{\partial T}\delta T + \frac{\partial j}{\partial(\partial_n T)}\delta(\partial_n T) - \frac{\partial j}{\partial \vec{n}} \cdot \nabla_S(\delta S). \tag{3.58}$$

Combining the terms from Equation 3.57 and 3.58:

$$\begin{aligned}\delta J = &\int_S \left(\frac{\partial j}{\partial \vec{f}} \cdot \partial_n \vec{f} + \frac{\partial j}{\partial T}\partial_n T + \frac{\partial j}{\partial(\partial_n T)}\partial_n^2 T - 2H_m j \right) \delta S ds \\ &+ \int_S \left(\frac{\partial j}{\partial \vec{f}} \cdot \delta\vec{f} + \frac{\partial j}{\partial T}\delta T + \frac{\partial j}{\partial(\partial_n T)}\delta(\partial_n T) - \frac{\partial j}{\partial \vec{n}} \cdot \nabla_S(\delta S) \right) ds.\end{aligned} \tag{3.59}$$

The variations $\delta\vec{f}$, δT, $\delta(\partial_n T)$, and $\nabla_S(\delta S)$ will need to be expanded in terms of the direct (flow) solution and the surface variation δS. These variations depend on the constraints of the direct problem, and will vary depending on whether a viscous or inviscid problem is chosen. The partial derivative terms for a selection of commonly used functionals are listed in Table 3.3. The choice of deriving the equations in terms of $\vec{f}$, $\partial_n T$, etc, can be seen from this table to simplify the

Table 3.3: Partial Derivative Terms for Common Surface Functionals

Functional	j	$\frac{\partial j}{\partial \vec{f}}$	$\frac{\partial j}{\partial T}$	$\frac{\partial j}{\partial(\partial_n T)}$	$\frac{\partial j}{\partial \vec{n}}$
C_D	$\frac{1}{C_\infty}\begin{pmatrix}\cos\alpha\cos\beta\\ \sin\alpha\cos\beta\\ \sin\beta\end{pmatrix}\cdot\vec{f}$	$\frac{1}{C_\infty}\begin{pmatrix}\cos\alpha\cos\beta\\ \sin\alpha\cos\beta\\ \sin\beta\end{pmatrix}$	0	0	0
C_L	$\frac{1}{C_\infty}\begin{pmatrix}-\sin\alpha\\ \cos\alpha\\ 0\end{pmatrix}\cdot\vec{f}$	$\frac{1}{C_\infty}\begin{pmatrix}-\sin\alpha\\ \cos\alpha\\ 0\end{pmatrix}$	0	0	0
Heat flux	$c\partial_n T$	0	0	c	$c\nabla T$

process. In this table, $C_\infty = \frac{1}{2}\rho_\infty v_\infty^2 A_{ref}$ where $(\cdot)_\infty$ refers to freestream values and A_{ref} is the reference area used to calculate force coefficients. The angles α and β are the angle of attack and side-slip angles of the flow. Note that the force-based functionals are all stated in the form $j = \vec{d}\cdot\vec{f}$.

The terms of the functional kernal j can be expanded as:

$$\begin{aligned}\vec{f} &= \left(\bar{\bar{I}}P - \bar{\bar{\sigma}}\right)\cdot\vec{n}\\ \partial_n T &= \vec{n}\cdot\nabla T.\end{aligned} \tag{3.60}$$

Correspondingly, their variations are:

$$\begin{aligned}\delta\vec{f} = \delta\left(\bar{\bar{I}}P - \bar{\bar{\sigma}}\right)\cdot\vec{n} \quad &= \left(\bar{\bar{I}}\delta P - \delta\bar{\bar{\sigma}}\right)\cdot\vec{n} - \left(\bar{\bar{I}}P - \bar{\bar{\sigma}}\right)\cdot\nabla_S(\delta S)\\ \delta(\partial_n T) &= \delta(\partial_n T)\\ \delta\vec{n} &= -\nabla_S(\delta S).\end{aligned} \tag{3.61}$$

From Equation 3.58, the terms on the surface of the adjoint problem become:

$$\begin{aligned}\delta J = &\int_S\left(\frac{\partial j}{\partial\vec{f}}\cdot\partial_n\vec{f} + \frac{\partial j}{\partial T}\partial_n T + \frac{\partial j}{\partial(\partial_n T)}\partial_n^2 T - 2H_m j\right)\delta S ds\\ &+\int_S\left(\frac{\partial j}{\partial\vec{f}}\cdot\left(\left(\bar{\bar{I}}\delta P - \delta\bar{\bar{\sigma}}\right)\cdot\vec{n} - \left(\bar{\bar{I}}P - \bar{\bar{\sigma}}\right)\cdot\nabla_S(\delta S)\right)\right)ds\\ &+\int_S\left(\frac{\partial j}{\partial T}\delta T + \frac{\partial j}{\partial(\partial_n T)}\left(\delta(\partial_n T)\right) - \frac{\partial j}{\partial\vec{n}}\cdot\nabla_S(\delta S)\right)ds\\ &=\int_S\left(\frac{\partial j}{\partial\vec{f}}\cdot\left(\partial_n\vec{f} - \left(\bar{\bar{I}}P - \bar{\bar{\sigma}}\right)\cdot\nabla_S\right)\right)\delta S\, ds\\ &+\int_S\left(\frac{\partial j}{\partial T}\partial_n T + \frac{\partial j}{\partial(\partial_n T)}\left(\partial_n^2 T\right) - 2H_m j - \frac{\partial j}{\partial\vec{n}}\cdot\nabla_S\right)\delta S\, ds\\ &+\int_S\left(\frac{\partial j}{\partial\vec{f}}\cdot\left(\bar{\bar{I}}\delta P - \delta\bar{\bar{\sigma}}\right)\cdot\vec{n} + \frac{\partial j}{\partial T}\delta T + \frac{\partial j}{\partial(\partial_n T)}\left(\delta(\partial_n T)\right)\right)ds.\end{aligned} \tag{3.62}$$

This expression will now be simplified for a selection of functionals.

Expansion of Variation for a Projected Force-Based Functionals

Common force-based functionals are based on a projection of the surface forces in a particular direction. For example, lift and drag coefficients are composed of the surface forces projected in the directions perpendicular to and parallel to the direction of fluid flow. Functionals in this form are convenient for the adjoint formulation as they allow elimination of the surface curvature term H_m. Using $\vec{d}$ as the direction in which the force is projected:

$$\begin{aligned} j &= \vec{d} \cdot \vec{f} \\ &\frac{\partial j}{\partial \vec{f}} = \vec{d} \quad \frac{\partial j}{\partial T} = 0 \quad \frac{\partial j}{\partial (\partial_n T)} = 0 \quad \frac{\partial j}{\partial \vec{n}} = \vec{0} \\ \partial_n \vec{f} &= \vec{n} \cdot \nabla \left(\vec{f} \right). \end{aligned} \tag{3.63}$$

the variation of the functional becomes:

$$\begin{aligned} \delta J = &\int_S \left(\vec{d} \cdot \partial_n \vec{f} - 2 H_m \vec{d} \cdot \vec{f} \right) \delta S \, ds \\ &+ \int_S \left(\vec{d} \cdot \delta \vec{f} \right) ds. \end{aligned} \tag{3.64}$$

Expanding the variation of $\delta \vec{f}$,

$$\begin{aligned} \delta J &= \int_S \left(\vec{d} \cdot \partial_n \vec{f} - 2 H_m \vec{d} \cdot \vec{f} \right) \delta S \, ds + \int_S \left(\vec{d} \cdot ((\bar{\bar{I}} \delta P - \delta \bar{\bar{\sigma}}) \cdot \vec{n} - (\bar{\bar{I}} P - \bar{\bar{\sigma}}) \cdot \nabla_S \delta S) \right) ds \\ &= \int_S \left(\vec{d} \cdot \partial_n \vec{f} - 2 H_m \vec{d} \cdot \vec{f} \right) \delta S - \vec{d} \cdot (\bar{\bar{I}} P - \bar{\bar{\sigma}}) \cdot \nabla_S (\delta S) ds \\ &+ \int_S \left(\vec{d} \cdot (\bar{\bar{I}} \delta P - \delta \bar{\bar{\sigma}}) \cdot \vec{n} \right) ds. \end{aligned} \tag{3.65}$$

Using integration by parts to simplify some of the terms:

$$- \int_S \vec{d} \cdot \left(\bar{\bar{I}} P - \bar{\bar{\sigma}} \right) \cdot \nabla_S (\delta S) ds = - \int_S \nabla_S \cdot \left[\vec{d} \cdot (\bar{\bar{I}} P - \bar{\bar{\sigma}}) \delta S \right] ds \quad + \int_S \nabla_S \cdot \left[\vec{d} \cdot (\bar{\bar{I}} P - \bar{\bar{\sigma}}) \right] \delta S ds. \tag{3.66}$$

Recombining:

$$\begin{aligned}\delta J = &\int_S \left(\vec{d}\cdot\left(\bar{\bar{I}}\delta P - \delta\bar{\bar{\sigma}}\right)\cdot\vec{n}\right) ds \\ &+ \int_S \left(\vec{d}\cdot(\partial_n\vec{f}) - 2H_m(\vec{f}\cdot\vec{d}) + \nabla_S\cdot\left[\vec{d}\cdot(\bar{\bar{I}}P)\right]\right)\delta S\, ds \\ &- \int_S \nabla_S\cdot\left[\vec{d}\cdot(\bar{\bar{I}}P - \delta\bar{\bar{\sigma}})\delta S\right] ds \\ &= \int_S \left(\vec{d}\cdot\left(\bar{\bar{I}}\delta P - \delta\bar{\bar{\sigma}}\right)\cdot\vec{n}\right) ds \\ &+ \int_S \left\{\vec{d}\cdot(\partial_n\vec{f}) - 2H_m(\vec{f}\cdot\vec{d}) + \nabla_S\cdot\left[\vec{d}\cdot(\bar{\bar{I}}P - \bar{\bar{\sigma}})\right]\right\}\delta S\, ds\,,\end{aligned} \tag{3.67}$$

where the above equation applies $\int_S \nabla_S(\cdot)\, ds = 0$ on a closed surface.

Further simplification:

$$\begin{aligned}&\vec{d}\cdot(\partial_n\vec{f}) - 2H_m(\vec{f}\cdot\vec{d}) + \nabla_S\cdot\left[\vec{d}\cdot(\bar{\bar{I}}P - \bar{\bar{\sigma}})\right] \\ &\qquad= \nabla_S\cdot\left[\vec{d}\cdot(\bar{\bar{I}}P - \bar{\bar{\sigma}})\right] + \partial_n(\vec{d}\cdot\vec{f}) - \vec{f}\cdot\partial_n\vec{d} - 2H_m(\vec{f}\cdot\vec{d}) \\ &\qquad= \nabla_S\cdot\left[\vec{d}\cdot(\bar{\bar{I}}P - \bar{\bar{\sigma}})\right] + \partial_n(\vec{d}\cdot((\bar{\bar{I}}P - \bar{\bar{\sigma}})\cdot\vec{n})) - ((\bar{\bar{I}}P - \bar{\bar{\sigma}})\cdot\vec{n})\cdot\partial_n\vec{d} \\ &\qquad- 2H_m[\vec{d}\cdot((\bar{\bar{I}}P - \bar{\bar{\sigma}})\cdot\vec{n})] \\ &\qquad= \nabla\cdot\left[\vec{d}\cdot(\bar{\bar{I}}P - \bar{\bar{\sigma}})\right] - ((\bar{\bar{I}}P - \bar{\bar{\sigma}})\cdot\vec{n})\cdot\partial_n\vec{d}.\end{aligned} \tag{3.68}$$

The last line has used $\nabla\cdot\vec{q} = \nabla_S\cdot\vec{q} + \partial_n(\vec{q}\cdot\vec{n}) - 2H_m(\vec{q}\cdot\vec{n})$.

Substituting the result into δJ:

$$\begin{aligned}\delta J &= \int_S \vec{d}\cdot(\bar{\bar{I}}\delta P - \delta\bar{\bar{\sigma}})\cdot\vec{n} ds + \int_S \left\{\nabla\cdot\left[\vec{d}\cdot(\bar{\bar{I}}P - \bar{\bar{\sigma}})\right] - (\bar{\bar{I}}P - \bar{\bar{\sigma}})\cdot\vec{n}\cdot\partial_n\vec{d}\right\}\delta S ds \\ \delta J &= \int_S \vec{d}\cdot(\bar{\bar{I}}\delta P - \delta\bar{\bar{\sigma}})\cdot\vec{n} ds + \int_S \left\{\nabla\vec{d}:(\bar{\bar{I}}P - \bar{\bar{\sigma}}) + \vec{d}\cdot\nabla\cdot(\bar{\bar{I}}P - \bar{\bar{\sigma}}) - (\bar{\bar{I}}P - \bar{\bar{\sigma}})\cdot\vec{n}\cdot\partial_n\vec{d}\right\}\delta S ds \\ \delta J &= \int_S \vec{d}\cdot(\bar{\bar{I}}\delta P - \delta\bar{\bar{\sigma}})\cdot\vec{n} ds + \int_S \left\{\nabla\vec{d}:(\bar{\bar{I}}P - \bar{\bar{\sigma}}) + \vec{d}\cdot(\nabla\cdot(\bar{\bar{I}}P - \bar{\bar{\sigma}})) - (\bar{\bar{I}}P - \bar{\bar{\sigma}})\cdot\vec{n}\cdot\partial_n\vec{d}\right\}\delta S ds.\end{aligned} \tag{3.69}$$

The momentum equation written on the surface is:

$$\nabla\cdot\left(\bar{\bar{I}}P - \bar{\bar{\sigma}}\right) = \vec{q}_{\rho\vec{v}} - \partial_t(\rho\vec{v}). \tag{3.70}$$

Taking the steady form ($\partial_t(\rho\vec{v}) = 0$), the variation of the functional is now:

$$\delta J = \int_S \vec{d}\cdot(\bar{\bar{I}}\delta P - \delta\bar{\bar{\sigma}})\cdot\vec{n} ds + \int_S \left\{\vec{d}\cdot[\vec{q}_{\rho\vec{v}}] + \nabla\vec{d}:(\bar{\bar{I}}P - \bar{\bar{\sigma}}) - (\bar{\bar{I}}P - \bar{\bar{\sigma}})\cdot\vec{n}\cdot\partial_n\vec{d}\right\}\delta S ds. \tag{3.71}$$

The terms in Equation 3.71 dependent on δP and $\delta\bar{\bar{\sigma}}$ in the first integral will be eliminated in later steps of the derivation of the adjoint equations. The terms in the second integral will contribute to the surface sensitivity.

Expansion of Variation for Heat Flux-Based Functionals

For a functional defined as a linear function of the heat flux,

$$j = c\partial_n T, \tag{3.72}$$

with derivatives as noted in Table 3.3, and eliminating zeroed terms from Equation 3.62

$$\begin{aligned}
\delta J &= \int_S \left(c\left(\partial_n^2 T\right) - 2H_m c\partial_n T\right)\delta S\, ds + \int_S c\left(\delta\left(\partial_n T\right)\right) ds \\
&\quad + \int_S -c\nabla T\cdot\nabla_S(\delta S) ds \\
&= \int_S \left(c\left(\partial_n^2 T\right) - 2H_m c\partial_n T\right)\delta S\, ds + \int_S c\left(\delta\left(\partial_n T\right)\right) ds \\
&\quad - \int_S \nabla_S\cdot\left(c\nabla T\delta S\right) - \nabla_S\cdot\left(c\nabla T\right)\delta S\, ds \\
&= \int_S \left(c\left(\partial_n^2 T\right) - 2H_m c\partial_n T + \nabla_S\cdot\left(c\nabla T\right)\right)\delta S\, ds + \int_S c\left(\delta\left(\partial_n T\right)\right) ds,
\end{aligned} \tag{3.73}$$

where integration by parts and the identity $\int_S \nabla_S\cdot(\cdot)ds = 0$ on a close surface have been used to simplify these expressions. Since $\nabla\cdot\vec{a} = \nabla_S\cdot\vec{a} + \partial_n(\vec{a}\cdot\vec{n}) - 2H_m\vec{a}\cdot\vec{n}$, this expression can be further simplified:

$$\delta J = \int_S \left(\nabla\cdot\left(c\nabla T\right)\right)\delta S\, ds + \int_S c\left(\delta\left(\partial_n T\right)\right) ds. \tag{3.74}$$

Expansion of Variation for Total Pressure Objective

For integrated total pressure, evaluated at an outflow, in terms of the primitive variables included in V:

$$\begin{aligned}
J &= \int_\Gamma P_t ds \\
j &= P\left(1 + \frac{(\gamma-1)|\vec{v}|^2\rho}{2\gamma P}\right)^{\frac{\gamma}{\gamma-1}} \\
&= Pg^{\frac{\gamma}{\gamma-1}},
\end{aligned} \tag{3.75}$$

Table 3.4: Functional Variations

Functional	$j =$	Variation $\delta J =$
Projected force	$\vec{d}\cdot\vec{f}$	$\int_S \vec{d}\cdot(\bar{\bar{I}}\delta P - \delta\bar{\bar{\sigma}})\cdot\vec{n}ds$ $+\int_S \left\{\vec{d}\cdot[\vec{q}_{\rho\vec{v}}] + \nabla\vec{d}:(\bar{\bar{I}}P-\bar{\bar{\sigma}}) - (\bar{\bar{I}}P-\bar{\bar{\sigma}})\cdot\vec{n}\cdot\partial_n\vec{d}\right\}\delta S ds$
Heat flux	$c\partial_n T$	$\int_S (\nabla\cdot(c\nabla T))\,\delta S\,ds + \int_S c\,(\delta\,(\partial_n T))\,ds$
Total Pressure	$Pg^{\frac{\gamma}{\gamma-1}}$ $g = 1+\frac{(\gamma-1)\vec{v}\vert^2\rho}{2\gamma P}$	$\int_S (g)^{\frac{1}{\gamma-1}}\left(\frac{\vec{v}\vert^2}{2}\delta\rho + \rho\vec{v}\cdot\delta\vec{v} + \left(g-\frac{\vec{v}\vert^2\rho}{2P}\right)\delta P\right)ds$

where a shorthand $g = \left(1+\frac{(\gamma-1)\vec{v}\vert^2\rho}{2\gamma P}\right)$ is introduced as shorthand. Expanding $\frac{\partial j}{\partial V}\delta V$,

$$\begin{aligned}\delta j &= P\frac{\gamma}{\gamma-1}g^{\frac{1}{\gamma-1}}\left(\frac{\gamma-1}{\gamma P2}\vec{v}\vert^2\right)\delta\rho + P\frac{\gamma}{\gamma-1}g^{\frac{1}{\gamma-1}}\left(\frac{\rho(\gamma-1)}{P\gamma}\vec{v}\right)\cdot\delta\vec{v}\\ &+ g^{\frac{1}{\gamma-1}}\left(g - P\frac{\gamma}{\gamma-1}\frac{\gamma-1}{2}\frac{\vec{v}\vert^2\rho}{P^2\gamma}\right)\delta P\\ &= g^{\frac{1}{\gamma-1}}\left(\frac{\vec{v}\vert^2}{2}\delta\rho + \rho\vec{v}\cdot\delta\vec{v} + \left(g-\frac{\vec{v}\vert^2\rho}{2P}\right)\delta P\right).\end{aligned} \tag{3.76}$$

This is similar to the terms found by Papadimitriou[56] minimizing total pressure loss for turbomachinery applications.

Summary of Functional Variations

Section 3.1.5 has detailed the expansion of a number of a selection of functional variations, including force-based, heat-flux based, and outflow-based functionals. Table 3.4 summarizes the resulting equations, which will be used later in the adjoint equation derivations. Many further functionals are available in the literature; the discussion is restricted to a small number to serve as an example and provide background for the generalized form that follows in Section 3.2.

3.1.6 Forming the Adjoint Boundary Conditions & Surface Sensitivity

In order to form the adjoint equations, we need to expand the variation of the Lagrangian, apply known relationships for δU from the linearized boundary conditions of the direct problem, and eliminate the remaining dependence on δU from the equations to form the system of adjoint equations. Components of the variation of the Lagrangian have been expanded in Section 3.2.1 and Section 3.1.2. Each of these sections concludes with summary tables that will be referenced during the remainder of the derivation.

Taking the combination of the terms from δJ shown in Table 3.4, the relevant linearized boundary condition from Table 3.2, and the adjoint boundary terms from Equation 3.38, produces the boundary conditions that Ψ should satisfy to eliminate dependence on δU. This section will start

with the solid wall boundary conditions for a force-based functional under adiabatic, isothermal, and inviscid conditions. This is followed by the boundary conditions for a heat flux functional with an isothermal wall, the boundary conditions for outflows and inflows, and finally a summary of the terms found in this section. This is intended as a review of the derivation of the established adjoint equations, and lays the groundwork for the generalized form and multi-objective form included in a later section.

Adjoint Boundary Conditions for a Force-Based Functional and an Adiabatic Boundary

At an adiabatic boundary, the no-slip condition and a zero heat-flux constraint are applied in the direct problem, and these terms must be linearized and factored into the adjoint equations. From Table 3.2, the linearized no-slip condition is:

$$\delta\vec{v} = -\partial_n(\vec{v} - \vec{u}_\Omega)\delta S \text{ on } S. \tag{3.77}$$

The linearized adiabatic wall condition is:

$$\partial_n(\delta T) = \nabla T \cdot \nabla_S(\delta S) - \partial_n^2(T)\delta S \text{ on} S. \tag{3.78}$$

Factoring these into Equation 3.38, and assuming that the volume terms have been eliminated:

$$\begin{aligned}
\delta\mathcal{J} = \delta J &+ \int_\Omega (\ldots) d\Omega \\
&- \int_S \vartheta(-\partial_n(\vec{v} - \vec{u}_\Omega)\delta S \cdot \vec{n}) + (\vec{\varphi} + \psi_{\rho E}\vec{v})(\bar{\bar{I}}\delta P - \delta\bar{\bar{\sigma}}) \cdot \vec{n}\, ds \\
&+ \int_S \psi_{\rho E}(-\partial_n(\vec{v} - \vec{u}_\Omega)\delta S) \cdot \bar{\bar{\sigma}} \cdot \vec{n} + \psi_{\rho E}\mu_{tot}^2 c_p(\nabla T \cdot \nabla_S(\delta S) - \partial_n^2(T)\delta S)\, ds \\
&- \int_S \vec{n} \cdot \left(\bar{\bar{\Sigma}}^{\varphi} + \bar{\bar{\Sigma}}^{\psi_{\rho E}}\right) \cdot (-\partial_n(\vec{v} - \vec{u}_\Omega)\delta S) + \mu_{tot}^2 c_p \partial_n(\psi_{\rho E})\delta T\, ds.
\end{aligned} \tag{3.79}$$

Substituting in the variation of a projected-force functional,

$$\begin{aligned}
\delta\mathcal{J} = &\int_S \vec{d} \cdot (\bar{\bar{I}}\delta P - \delta\bar{\bar{\sigma}}) \cdot \vec{n} ds + \int_S \left\{\vec{d} \cdot [\vec{q}_{\rho\vec{v}}] + \nabla\vec{d} : (\bar{\bar{I}}P - \bar{\bar{\sigma}}) - (\bar{\bar{I}}P - \bar{\bar{\sigma}}) \cdot \vec{n} \cdot \partial_n\vec{d}\right\} \delta S ds \\
&- \int_S \vartheta(-\partial_n(\vec{v} - \vec{u}_\Omega)\delta S \cdot \vec{n}) + (\vec{\varphi} + \psi_{\rho E}\vec{v})(\bar{\bar{I}}\delta P - \delta\bar{\bar{\sigma}}) \cdot \vec{n}\, ds \\
&+ \int_S \psi_{\rho E}(-\partial_n(\vec{v} - \vec{u}_\Omega)\delta S) \cdot \bar{\bar{\sigma}} \cdot \vec{n} + \psi_{\rho E}\mu_{tot}^2 c_p(\nabla T \cdot \nabla_S(\delta S) - \partial_n^2(T)\delta S)\, ds \\
&- \int_S \vec{n} \cdot \left(\bar{\bar{\Sigma}}^{\varphi} + \bar{\bar{\Sigma}}^{\psi_{\rho E}}\right) \cdot (-\partial_n(\vec{v} - \vec{u}_\Omega)\delta S) + \mu_{tot}^2 c_p \partial_n(\psi_{\rho E})\delta T\, ds.
\end{aligned} \tag{3.80}$$

Dependence on δP and $\delta\bar{\bar{\sigma}}$ by setting $\vec{\varphi} = \vec{d} - \psi_{\rho E}\vec{v}$. The dependence on δT is eliminated by setting $\partial_n(\psi_{\rho E}) = 0$.

Summarizing the equations for the force-based functional with an adiabatic wall boundary,

$$\begin{cases} \nabla\Psi^{\mathsf{T}}\cdot\left(\vec{A}^{c}-\bar{\bar{I}}_5\vec{u}_\Omega+\mu_{tot}^{k}\vec{A}^{vk}\right)+\nabla\cdot\left(\nabla\Psi^{\mathsf{T}}\cdot\mu_{tot}^{k}\bar{\bar{D}}^{vk}\right)+\Psi^{\mathsf{T}}\frac{\partial\mathcal{Q}}{\partial U}=0 & \text{on } \Omega \\ \vec{\varphi}=\vec{d}-\psi_{\rho E}\vec{v} & \\ \partial_n(\psi_{\rho E})=0 & \text{on } S \\ \psi_{\rho E,M<1}=-\vec{\varphi}\cdot\vec{n}\frac{(\gamma-1)}{\gamma\vec{v}\cdot\vec{n}-\vec{u}_\Omega\cdot\vec{n}} & \text{on } \Gamma_{in} \\ \begin{Bmatrix}\psi_\rho\\ \vec{\varphi}\end{Bmatrix}=\psi_{\rho E}\begin{Bmatrix}\frac{2c^2+\vec{v}^2(\gamma-1)}{2(\gamma-1)}\\ -\vec{n}\frac{c^2}{v_n(\gamma-1)}-\vec{v}\end{Bmatrix} & \text{on } \Gamma_{out}. \\ \psi_{\rho E,M_e>1}=0 & \end{cases} \tag{3.81}$$

The term involving ∇_S and the second derivative of temperature from Equation 3.80 should now be simplified in order to avoid the necessity of evaluating the second derivative. Via a sequence of manipulations including application of the adiabatic boundary condition, the identity $\int_S \nabla_S\cdot(\cdot)ds = 0$ on a closed surface, and integration by parts, which can be found expanded in 92,

$$\begin{aligned}&\psi_{\rho E}\mu_{tot}^2c_p[\nabla T\cdot\nabla_S(\delta S)-\partial_n^2(T)\delta S]\\ &\quad=-\mu_{tot}^2c_p\nabla_S T\cdot\nabla_S(\psi_{\rho E})\delta S-\psi_{\rho E}\nabla\cdot(\mu_{tot}^2c_p\nabla T)\delta S.\end{aligned} \tag{3.82}$$

The energy equation on the surface under no-slip and adiabatic conditions can be expressed as:

$$\nabla\cdot(\mu_{tot}^2c_p\nabla T)=P(\nabla\cdot\vec{v}-\bar{\bar{\sigma}}:\nabla\vec{v}+\partial_t(\rho E)+(\vec{q}_{\rho\vec{v}}-\partial_t(\rho\vec{v}))\cdot\vec{v}-q_{\rho E}. \tag{3.83}$$

Using the steady form, and substituting the previous expressions into Equation 3.80,

$$\begin{aligned}\delta\mathcal{J}=&\int_S\left\{\vec{n}\cdot\left(\vartheta(\partial_n(\vec{v}-\vec{u}_\Omega))+\psi_{\rho E}\left(-\partial_n(\vec{v}-\vec{u}_\Omega)\cdot\bar{\bar{\sigma}}\right)+\left(\bar{\bar{\Sigma}}^{\varphi}+\bar{\bar{\Sigma}}^{\psi_{\rho E}}\right)\cdot(\partial_n(\vec{v}-\vec{u}_\Omega))\right)\right\}\delta S\,ds\\ &+\int_S\left\{\vec{d}\cdot[\vec{q}_{\rho\vec{v}}]+\nabla\vec{d}:(\bar{\bar{I}}P-\bar{\bar{\sigma}})-(\bar{\bar{I}}P-\bar{\bar{\sigma}})\cdot\vec{n}\cdot\partial_n\vec{d}\right\}\delta S\,ds\\ &+\int_S\left\{\mu_{tot}^2c_p\psi_{\rho E}\left(\nabla T\cdot\nabla_S(\delta S)-\partial_n^2(T)\delta S\right)-\psi_{\rho E}(P(\nabla\cdot\vec{v})-\bar{\bar{\sigma}}:\nabla\vec{v}+(\vec{q}_{\rho\vec{v}}\cdot\vec{v}-q_{\rho E})\right\}\delta S\,ds\,,\end{aligned} \tag{3.84}$$

The surface sensitivity for a projected-force functional with an adiabatic wall is:

$$\begin{aligned}\frac{\partial\mathcal{J}}{\partial S}\approx\frac{\delta\mathcal{J}}{\delta S}=&\left\{\vec{n}\cdot\left(\vartheta(\partial_n(\vec{v}-\vec{u}_\Omega))+\psi_{\rho E}\left(-\partial_n(\vec{v}-\vec{u}_\Omega)\cdot\bar{\bar{\sigma}}\right)+\left(\bar{\bar{\Sigma}}^{\varphi}+\bar{\bar{\Sigma}}^{\psi_{\rho E}}\right)\cdot(\partial_n(\vec{v}-\vec{u}_\Omega))\right)\right\}\\ &+\left\{\vec{d}\cdot[\vec{q}_{\rho\vec{v}}]+\nabla\vec{d}:(\bar{\bar{I}}P-\bar{\bar{\sigma}})-(\bar{\bar{I}}P-\bar{\bar{\sigma}})\cdot\vec{n}\cdot\partial_n\vec{d}\right\}\\ &+\left\{-\mu_{tot}^2c_p\nabla_S(\psi_{\rho E})\cdot\nabla_S(T)-\psi_{\rho E}(P(\nabla\cdot\vec{v})-\bar{\bar{\sigma}}:\nabla\vec{v}+(\vec{q}_{\rho\vec{v}}\cdot\vec{v}-q_{\rho E})\right\}.\end{aligned} \tag{3.85}$$

Adjoint Boundary Conditions for a Force-Based Functional and an Isothermal Boundary

At an isothermal boundary, a similar process is used, this time substituting the linearized isothermal wall condition, $\delta T = -\partial_n(T)\delta S$:

$$\begin{aligned}\delta\mathcal{J} = &\int_S \vec{d}\cdot(\bar{\bar{I}}\delta P - \delta\bar{\bar{\sigma}})\cdot\vec{n}ds + \int_S \left\{\vec{d}\cdot[\vec{q}_{\rho\vec{v}}] + \nabla\vec{d}:(\bar{\bar{I}}P-\bar{\bar{\sigma}}) - (\bar{\bar{I}}P-\bar{\bar{\sigma}})\cdot\vec{n}\cdot\partial_n\vec{d}\right\}\delta S ds\\ &-\int_S \vartheta(-\partial_n(\vec{v}-\vec{u}_\Omega)\delta S\cdot\vec{n}) + (\vec{\varphi}+\psi_{\rho E}\vec{v})(\bar{\bar{I}}\delta P - \delta\bar{\bar{\sigma}})\cdot\vec{n}\,ds\\ &+\int_S \psi_{\rho E}(-\partial_n(\vec{v}-\vec{u}_\Omega)\delta S)\cdot\bar{\bar{\sigma}}\cdot\vec{n} + \psi_{\rho E}\mu_{tot}^2 c_p\partial_n(\delta T)\,ds\\ &-\int_S \vec{n}\cdot\left(\bar{\bar{\Sigma}}^{\varphi}+\bar{\bar{\Sigma}}^{\psi_{\rho E}}\right)\cdot(-\partial_n(\vec{v}-\vec{u}_\Omega)\delta S) + \mu_{tot}^2 c_p\partial_n(\psi_{\rho E})(-\partial_n(T)\delta S)\,ds.\end{aligned} \tag{3.86}$$

Once again the dependence on δP and $\delta\bar{\bar{\sigma}}$ can be eliminated by setting $\vec{\varphi} = \vec{d} - \psi_{\rho E}\vec{v}$. In order to eliminate dependence on $\partial_n(\delta T)$, $\psi_{\rho E} = 0$. Summarizing the equations for a force-based functional with an isothermal wall:

$$\begin{cases}\nabla\Psi^{\mathsf{T}}\cdot\left(\vec{A}^c - \bar{\bar{I}}_5\vec{u}_\Omega + \mu_{tot}^k\vec{A}^{vk}\right) + \nabla\cdot\left(\nabla\Psi^{\mathsf{T}}\cdot\mu_{tot}^k\bar{\bar{D}}^{vk}\right) + \Psi^{\mathsf{T}}\frac{\partial\mathcal{Q}}{\partial U} = 0 & \text{on } \Omega\\ \vec{\varphi} = \vec{d} - \psi_{\rho E}\vec{v} & \text{on } S\\ \psi_{\rho E} = 0 & \\ \psi_{\rho E, M<1} = -\vec{\varphi}\cdot\vec{n}\frac{(\gamma-1)}{\gamma\vec{v}\cdot\vec{n}-\vec{u}_\Omega\cdot\vec{n}} & \text{on } \Gamma_{in}\\ \begin{Bmatrix}\psi_\rho\\ \vec{\varphi}\end{Bmatrix} = \psi_{\rho E}\begin{Bmatrix}\frac{2c^2+\vec{v}^2(\gamma-1)}{2(\gamma-1)}\\ -\vec{n}\frac{c^2}{v_n(\gamma-1)} - \vec{v}\end{Bmatrix} & \text{on } \Gamma_{out}.\\ \psi_{\rho E, M_e>1} = 0 & \end{cases} \tag{3.87}$$

Using a similar procedure as for the adiabatic wall,

$$\begin{aligned}\frac{\partial\mathcal{J}}{\partial S} \approx \frac{\delta\mathcal{J}}{\delta S} = &\{\vec{n}\cdot(\vartheta(\partial_n(\vec{v}-\vec{u}_\Omega)) + \psi_{\rho E}(-\partial_n(\vec{v}-\vec{u}_\Omega)\cdot\bar{\bar{\sigma}}))\}\\ &+\left\{\vec{n}\cdot\left(\left(\bar{\bar{\Sigma}}^{\varphi}+\bar{\bar{\Sigma}}^{\psi_{\rho E}}\right)\cdot(\partial_n(\vec{v}-\vec{u}_\Omega))\right)\right\}\\ &+\left\{\vec{d}\cdot[\vec{q}_{\rho\vec{v}}] + \nabla\vec{d}:(\bar{\bar{I}}P-\bar{\bar{\sigma}}) - (\bar{\bar{I}}P-\bar{\bar{\sigma}})\cdot\vec{n}\cdot\partial_n\vec{d}\right\}\\ &+\left\{-\mu_{tot}^2 c_p\partial_n(\psi_{\rho E})\partial_n(T)\right\}.\end{aligned} \tag{3.88}$$

3.1.7 Summary of the Continuous Adjoint Method Derivation

The derivation of the adjoint equations has, so far, followed a similar method as the derivations available in literature, where a defined functional is required prior to determining the adjoint boundary

conditions and the surface sensitivity. This methodology will be slightly re-organized when deriving the generalized form in following sections. To summarize the equations found in this section, Table 3.4 shows the variations for a selection of functionals that exist in the literature, Table 3.2 shows the linearized direct problem boundary conditions, and Table 3.1 summarizes the linearized volume integral from the direct problem. The information from these tables is combined to provide the governing equations of the adjoint problem in Equation 3.39. The boundary conditions applied to this PDE under a selection of functionals and boundary conditions are shown in 3.81 and 3.87. The remaining nonzero terms are collected into the surface sensitivty, again for a selection of circumstances, in 3.85 and 3.88.

The purpose of this section has been to illustrate the established process of deriving the adjoint equations, such that the reader can better understand the different tactics undertaken in the proceeding sections. Some of the difficulties encountered when deriving the adjoint equations include a certain amount of repeated work done when deriving the adjoint formulation for a new functional. As will be shown in following sections, the functional contributions to the surface sensitivity and to the boundary conditions can be generalized, lessening the work required for additional functionals and allowing the adjoint method to address functionals that previously would have been inaccessible.

3.2 Generalized Functionals

Section 3.1 detailed the derivation of the adjoint equations for a selection of specific functionals. This section will demonstrate the derivation of the adjoint equations for arbitrary functionals, generating a generalized form that depends on the partial derivatives of the functional. This methodology broadens the number of functionals that can be addressed and opens the door to addressing functionals that depend on external models rather than being limited to functionals that depend only on quantities within the flow solution. The derivation for generalized functionals follows a similar technique, with the derivation diverging at the expansion of the functional variations. The reader is referred to Table 3.2 and 3.1 for the terms derived in Section 3.1 that will be needed in this section. Section 3.1.5 detailed the functional variations for a selection of functionals that have been previously addressed in literature. This section will start from the expansion of arbitrary functional variations expressed in terms of convenient terms.

In outline, the derivation starts with the application of the divergence theorem and collection of terms on the boundaries (Section 3.1.3), expansion and simplification of boundary terms (Section 3.1.3), elimination of remaining arbitrary δU from the equations (Section 3.2.3), and finally collection of the terms dependent on δS to form the surface sensitivity (Section 3.2.4). The equations derived are summarized in Section 3.2.5.

3.2.1 Functional Variations

The derivation of the adjoint equations requires the variation of the chosen functional. This section expands the variations of a selection of functionals. A summary table is provided at the end of this section for reference. For functions defined on solid surfaces, it is convenient to express the functions in terms of the surface force $\vec{f}$, the temperature T, the normal gradient of $\partial_n T$, and the normal vector $\vec{n}$. At outflow boundaries, it is convenient to express the objective directly in terms of the primitive variables V. Equation 3.55 is restated here for convenient reference, and its components will now be expanded. The resulting variations are summarized in Equation 3.99

$$\begin{aligned} J &= \int_S j(\vec{f}, T, \partial_n T, \vec{n}) ds + \int_{\Gamma_e} j(V) ds \\ \delta J &= \int_S \left(\frac{\partial j}{\partial \vec{f}} \cdot \partial_n \vec{f} + \frac{\partial j}{\partial T} \partial_n T + \frac{\partial j}{\partial (\partial_n T)} \partial_n^2 T - 2 H_m j \right) \delta S ds + \\ &\quad \int_S \left(\frac{\partial j}{\partial \vec{f}} \cdot \delta \vec{f} + \frac{\partial j}{\partial T} \delta T + \frac{\partial j}{\partial (\partial_n T)} \delta(\partial_n T) - \frac{\partial j}{\partial \vec{n}} \cdot \nabla_S (\delta S) \right) ds + \int_{\Gamma_e} \frac{\partial j}{\partial V} \delta V ds. \end{aligned} \tag{3.89}$$

Expansion of Variations for a Generalized Force-Based Functional

A slight modification can be made to the force-based variation to generalize the equation to functionals that cannot be expressed as $j = \vec{d} \cdot \vec{f}$. In this case, the variation of the functional is:

$$\delta J = \int_S \left(\frac{\partial j}{\partial \vec{f}} \cdot \partial_n \vec{f} - 2 H_m j \right) \delta S ds + \int_S \left(\frac{\partial j}{\partial \vec{f}} \cdot \delta \vec{f} \right) ds, \tag{3.90}$$

where the remainder of the partial derivatives are set to zero. The expansion proceeds similarly as before, up until the point where the terms involving H_m were eliminated. Instead of being eliminated, the expressions are simplified as follows:

$$\begin{aligned} \delta J &= \int_S \left(\frac{\partial j}{\partial \vec{f}} \cdot \left(\bar{\bar{I}} \delta P - \delta \bar{\bar{\sigma}} \right) \cdot \vec{n} \right) ds \\ &\quad + \int_S \left\{ \frac{\partial j}{\partial \vec{f}} \cdot (\partial_n \vec{f}) - 2 H_m (j) + \nabla_S \cdot \left[\frac{\partial j}{\partial \vec{f}} \cdot (\bar{\bar{I}} P - \bar{\bar{\sigma}}) \right] \right\} \delta S \, ds. \end{aligned} \tag{3.91}$$

Performing a simplification similar to Equation 3.68:

$$\begin{aligned} &\tfrac{\partial j}{\partial \vec{f}} \cdot (\partial_n \vec{f}) - 2 H_m (j) + \nabla_S \cdot \left[\tfrac{\partial j}{\partial \vec{f}} \cdot (\bar{\bar{I}} P - \bar{\bar{\sigma}}) \right] \\ &= \tfrac{\partial j}{\partial \vec{f}} \cdot (\partial_n \vec{f}) - 2 H_m (j) + \nabla_S \cdot \left[\tfrac{\partial j}{\partial \vec{f}} \cdot (\bar{\bar{I}} P - \bar{\bar{\sigma}}) \right] + 2 H_m \left(\tfrac{\partial j}{\partial \vec{f}} - \tfrac{\partial j}{\partial \vec{f}} \right) \cdot \vec{f} \\ &= \nabla \cdot \left[\tfrac{\partial j}{\partial \vec{f}} \cdot (\bar{\bar{I}} P - \bar{\bar{\sigma}}) \right] - ((\bar{\bar{I}} P - \bar{\bar{\sigma}}) \cdot \vec{n}) \cdot \partial_n \tfrac{\partial j}{\partial \vec{f}} + 2 H_m \left(\tfrac{\partial j}{\partial \vec{f}} \cdot \vec{f} - j \right). \end{aligned} \tag{3.92}$$

Where $\vec{f} = (\bar{\bar{I}} P - \bar{\bar{\sigma}}) \cdot \vec{n}$ has been substituted and the expansion for $\nabla \cdot [\vec{q} \cdot \vec{n}]$ has been used to simplify at the last line. The additional term $2 H_m \left(\frac{\partial j}{\partial \vec{f}} \cdot \vec{f} - j \right)$ contributes only to the surface sensitivity,

and is eliminated when $j = \frac{\partial j}{\partial \vec{f}} \cdot \vec{f}$. Otherwise the expansion of this variation can proceed exactly as for the projected-force objective, leading to:

$$\begin{aligned}\delta J = &\int_S \frac{\partial j}{\partial \vec{f}} \cdot (\bar{\bar{I}}\delta P - \delta\bar{\bar{\sigma}}) \cdot \vec{n} ds \\ &+ \int_S \left\{ \frac{\partial j}{\partial \vec{f}} \cdot [\vec{q}_{\rho\vec{v}} - \partial_t(\rho\vec{v})] + \nabla \frac{\partial j}{\partial \vec{f}} : (\bar{\bar{I}}P - \bar{\bar{\sigma}}) - (\bar{\bar{I}}P - \bar{\bar{\sigma}}) \cdot \vec{n} \cdot \partial_n \frac{\partial j}{\partial \vec{f}} \right\} \delta S ds \\ &+ \int_S \left\{ 2H_m (\vec{f} \cdot \frac{\partial j}{\partial \vec{f}} - j) \right\} \delta S ds.\end{aligned} \tag{3.93}$$

Expansion of Variation for General Heat Flux-Based Functionals

For a more general heat-flux based functional,

$$j = j(\partial_n T), \tag{3.94}$$

with derivatives noted in Table 3.4, we once again substitute terms into Equation 3.62:

$$\begin{aligned}\delta J &= \int_S \left(\frac{\partial j}{\partial(\partial_n T)} \left(\partial_n^2 T\right) - 2H_m j \right) \delta S \, ds + \int_S \frac{\partial j}{\partial(\partial_n T)} \left(\delta\left(\partial_n T\right)\right) ds \\ &+ \int_S -\frac{\partial j}{\partial(\partial_n T)} \nabla T \cdot \nabla_S(\delta S) ds \\ &= \int_S \left(\frac{\partial j}{\partial(\partial_n T)} \left(\partial_n^2 T\right) - 2H_m j \right) \delta S \, ds + \int_S \frac{\partial j}{\partial(\partial_n T)} \left(\delta\left(\partial_n T\right)\right) ds \\ &- \int_S \nabla_S \cdot \left(\frac{\partial j}{\partial(\partial_n T)} \nabla T \delta S \right) - \nabla_S \cdot (c\nabla T) \, \delta S \, ds \\ &= \int_S \left(\frac{\partial j}{\partial(\partial_n T)} \left(\partial_n^2 T\right) - 2H_m j + \nabla_S \cdot \left(\frac{\partial j}{\partial(\partial_n T)} \nabla T \right) \right) \delta S \, ds + \int_S \frac{\partial j}{\partial(\partial_n T)} \left(\delta\left(\partial_n T\right)\right) ds \\ &= \int_S \left(\frac{\partial j}{\partial(\partial_n T)} \left(\partial_n^2 T\right) - 2H_m \frac{\partial j}{\partial(\partial_n T)} \partial_n T + \nabla_S \cdot \left(\frac{\partial j}{\partial(\partial_n T)} \nabla T \right) \right) \delta S \, ds \\ &+ \int_S 2H_m \left(\frac{\partial j}{\partial(\partial_n T)} \partial_n T - j \right) \delta S ds + \int_S \frac{\partial j}{\partial(\partial_n T)} \left(\delta\left(\partial_n T\right)\right) ds.\end{aligned} \tag{3.95}$$

Applying the same identities as with previous expansions leads to a simplified form:

$$\delta J = \int_S \left(\nabla \cdot \left(\frac{\partial j}{\partial(\partial_n T)} \nabla T \right) + 2H_m \left(\frac{\partial j}{\partial(\partial_n T)} \partial_n T - j \right) \right) \delta S \, ds + \int_S \frac{\partial j}{\partial(\partial_n T)} \left(\delta\left(\partial_n T\right)\right) ds. \tag{3.96}$$

Expansion of Variation for a General Temperature-Based Functional

For a general temperature-based functional where the only non-zero partial derivative is $\frac{\partial j}{\partial T}$,

$$\delta J = \int_S \left(\frac{\partial j}{\partial T} \partial_n T - 2H_m \right) \delta S ds + \int_S \left(\frac{\partial j}{\partial T} \delta T \right) ds. \tag{3.97}$$

Expansion of Variation for a General Outflow-Based Functional

For a generalized outflow-based functional,

$$\begin{aligned} J &= \int_\Gamma j(V) ds \\ \delta J &= \int_\Gamma \frac{\partial j}{\partial V} \delta V ds + \int_{\delta\Gamma} j(V) ds \\ &= \int_\Gamma \frac{\partial j}{\partial V} \delta V ds, \end{aligned} \tag{3.98}$$

where in the last line the variation of the outflow surface shape is neglected. This applies an assumption that the outflow boundary will remain undeformed, which is a reasonable assumption for many applications such as an engine flowpath where the combustor shape may already be frozen.

3.2.2 Summary of Functional Variations

Section 3.2.1 has detailed the expansion of a number of functional variations, including force-based, heat-flux based, and outflow-based functionals. Combining terms together,

$$\begin{aligned} \delta J = &\int_{\Gamma_e} \left(\frac{\partial j}{\partial \rho} \delta\rho + \frac{\partial j}{\partial \vec{v}} \cdot \delta\vec{v} + \frac{\partial j}{\partial P} \delta P \right) ds \\ &+ \int_S \left(\frac{\partial j}{\partial \vec{f}} \cdot (\bar{\bar{I}} \delta P - \delta\bar{\bar{\sigma}}) \cdot \vec{n} + \frac{\partial j}{\partial (\partial_n T)} \left(\delta \left(\partial_n T \right) \right) + \frac{\partial j}{\partial T} \delta T \right) ds \\ &+ \int_S \left\{ \frac{\partial j}{\partial \vec{f}} \cdot [\vec{q}_{\rho\vec{v}}] + \nabla \frac{\partial j}{\partial \vec{f}} : (\bar{\bar{I}} P - \bar{\bar{\sigma}}) - (\bar{\bar{I}} P - \bar{\bar{\sigma}}) \cdot \vec{n} \cdot \partial_n \frac{\partial j}{\partial \vec{f}} \right\} \delta S ds \\ &+ \int_S \left\{ \nabla \cdot \left(\frac{\partial j}{\partial (\partial_n T)} \nabla T \right) + \frac{\partial j}{\partial T} \partial_n T \right\} \delta S \, ds + \int_S \left\{ 2H_m \left(\vec{f} \cdot \frac{\partial j}{\partial \vec{f}} + \frac{\partial j}{\partial (\partial_n T)} \partial_n T - j \right) \right\} \delta S ds. \end{aligned} \tag{3.99}$$

H_m is eliminated when the functional is such that $\vec{f} \cdot \frac{\partial j}{\partial \vec{f}} + \frac{\partial j}{\partial (\partial_n T)} \partial_n T - j = 0$, which is true for projected forces and linear functions of heat flux. Adjoint functionals are usually chosen such that H_m is eliminated due to the difficulties that arise when it is necessary to calculate the mean curvature.

3.2.3 Forming Generalized Adjoint Boundary Conditions & Surface Sensitivity

This section follows a similar pattern as Section 3.1.3, with the significant difference that in this section, a generalized form of the functional is used and terms of the surface sensitivity are separated into components arising from the boundary conditions and functionals respectively.

Taking the combination of the terms from δJ shown in Equation 3.99, the relevant linearized boundary condition from Table 3.2, and the adjoint boundary terms from Equation 3.38, produces the boundary conditions that Ψ should satisfy to eliminate dependence on δU. This section will start with the boundary conditions and sensitivity for an adiabatic boundary, followed by an isothermal boundary, an inviscid wall, and finally outflow boundaries. The terms derived in this section are summarized in Table 3.5 and 3.6.

Adjoint Boundary Conditions for An Adiabatic Boundary

Equation 3.80 incorporated the linearized adiabatic and no-slip boundary conditions into Equation 3.38. Combining this with the surface terms from the expression for δJ in Equation 3.99, in order to eliminate dependence on δP and $\delta\bar{\bar{\sigma}}$:

$$\vec{\varphi} = \frac{\partial j}{\partial \vec{f}} - \psi_{\rho E}\vec{v}. \tag{3.100}$$

The remaining terms depend on δS and δT, since terms of $\delta(\partial_n T)$ have been eliminated by the application of the boundary condition. The terms of δT can be eliminated by canceling the contribution from δJ with the contribution from δR:

$$\partial_n \psi_{\rho E} = \frac{1}{c_p \mu_{tot}^2} \frac{\partial j}{\partial T}. \tag{3.101}$$

All the variations of δU have now been eliminated from the adiabatic boundary, and the remaining nonzero terms contribute to the surface sensitivity. It is assumed that the functional should have no terms of heat flux at an adiabatic boundary.

Adjoint Boundary Conditions for An Isothermal Boundary

At an isothermal boundary, a fixed temperature has been applied. Factoring the linearized isothermal boundary condition from Table 3.2, 3.38 becomes:

$$\begin{aligned}\delta\mathcal{J} = \delta J + &\int_\Omega (\ldots) d\Omega \\ &- \int_S \vartheta(-\partial_n(\vec{v} - \vec{u}_\Omega)\delta S \cdot \vec{n}) + (\vec{\varphi} + \psi_{\rho E}\vec{v})(\bar{\bar{I}}\delta P - \delta\bar{\bar{\sigma}}) \cdot \vec{n}\, ds \\ &+ \int_S \psi_{\rho E}(-\partial_n(\vec{v} - \vec{u}_\Omega)\delta S) \cdot \bar{\bar{\sigma}} \cdot \vec{n} + \psi_{\rho E}\mu_{tot}^2 c_p \partial_n(\delta T)\, ds \\ &- \int_S \vec{n} \cdot \left(\bar{\bar{\Sigma}}^{\varphi} + \bar{\bar{\Sigma}}^{\psi_{\rho E}}\right) \cdot (-\partial_n(\vec{v} - \vec{u}_\Omega)\delta S) + \mu_{tot}^2 c_p \partial_n(\psi_{\rho E})(-\partial_n(T)\delta S)\, ds.\end{aligned} \tag{3.102}$$

Dependence on the force terms can be eliminated once again by Equation 3.100. This time, δT has been eliminated by application of the boundary conditions, and we eliminate the remaining $\partial_n(\delta T)$ by:

$$\psi_{\rho E} = -\frac{\partial j}{\partial(\partial_n T)} \frac{1}{c_p \mu_{tot}^2}. \tag{3.103}$$

Adjoint Boundary Conditions for Flow Tangency (Inviscid Wall) Boundaries

For the inviscid case, terms of μ disappear and the remaining variations depend only on δP. As shown by Arian & Salas,[62] functions of density can also be used, when the relationships between P and ρ are included. For the inviscid case, dependence on δP is eliminated by:

$$\vec{n} \cdot \vec{\varphi} = \frac{\partial j}{\partial \vec{f}} - \psi_{\rho E} v_n, \tag{3.104}$$

where v_n is only nonzero for a moving mesh because at the wall $v_{nr} = \vec{n} \cdot (\vec{v} - \vec{u}_\Omega) = 0$.

Adjoint Boundary Conditions at the Outflow

At an outflow boundary with viscous effects neglected, and an assumption of zero deformation of the outflow plane,

$$\frac{\partial j}{\partial V}\delta V - \Psi^T \vec{A}^c \cdot \vec{n} M \delta V = 0, \tag{3.105}$$

where the objective is defined in terms of the primitive variables V at the outflow,

$$J = \int_{\Gamma_e} j(V) ds. \tag{3.106}$$

When the flow is subsonic, a single characteristic is entering the flow solution. This means that

wherever the flow is subsonic, the terms of δV are not independent, and one of adjoint variables will be determined from the volume adjoint solution. The proceeding sections will detail the boundary conditions for a pressure-based boundary on a static mesh ($\vec{u}_\Omega = 0$), on a moving mesh, and a characteristic-value based boundary on a static mesh. For a supersonic boundary, the conditions are equivalent because all characteristics exit the flow solution, or enter the adjoint solution.

Generalized Outflow Boundary Conditions with Fixed Back Pressure

For a fixed back pressure and subsonic flow, $\delta P = 0$. This means that for subsonic flow, we will have one more variable than equations and one of the adjoint variables will depend on the volume solution. For supersonic flow δP is arbitrary and we will have the same number of equations as variables, meaning that all adjoint values will be specified at the boundary for supersonic flow. Expanding Equation 3.105,

$$\begin{Bmatrix} \frac{\partial j}{\partial \rho} \\ \frac{\partial j}{\partial \vec{v}} \\ \frac{\partial j}{\partial P} \end{Bmatrix}^T \begin{Bmatrix} \delta\rho \\ \delta\vec{v} \\ \delta P \end{Bmatrix} - \begin{Bmatrix} \psi_\rho v_n + \vec{v}\cdot\vec{\varphi}v_n + \psi_{\rho E} v_n \left(\frac{\vec{v}^2}{2}\right) \\ \rho(\vec{v}\cdot\vec{\varphi})\vec{n} + \rho v_n\vec{\varphi} + \rho\psi_\rho\vec{n} + \psi_{\rho E}\left(\rho v_n\vec{v} + \rho(\frac{c^2}{\gamma-1} + \frac{\vec{v}^2}{2})\vec{n}\right) \\ \vec{\varphi}\cdot\vec{n} + \psi_{\rho E}(v_n\frac{\gamma}{\gamma-1}) \end{Bmatrix}^T \begin{Bmatrix} \delta\rho \\ \delta\vec{v} \\ \delta P \end{Bmatrix}. \quad (3.107)$$

Eliminating the dependence on the variations $\delta\rho$ and $\delta\vec{v}$ leads to:

$$\begin{aligned} \frac{\partial j}{\partial \rho} - \left(\psi_\rho v_n + \vec{v}\cdot\vec{\varphi}v_n + \psi_{\rho E} v_n\left(\frac{\vec{v}^2}{2}\right)\right) &= 0 \\ \frac{\partial j}{\partial \vec{v}} - \left(\vec{n}\left(\rho(\vec{v}\cdot\vec{\varphi}) + \rho\psi_\rho + \psi_{\rho E}\rho\left(\frac{c^2}{\gamma-1} + \frac{\vec{v}^2}{2}\right)\right) + \vec{\varphi}\left(\rho v_n\right) + \vec{v}\left(\psi_{\rho E}\rho v_n\right)\right) &= \vec{0}. \end{aligned} \quad (3.108)$$

Solving the first line of Equation 3.108 for ψ_ρ:

$$\psi_\rho = \left(\frac{\partial j}{\partial \rho} - \vec{v}\cdot\vec{\varphi}v_n - \psi_{\rho E} v_n\left(\frac{\vec{v}^2}{2}\right)\right)\frac{1}{v_n}. \quad (3.109)$$

Plugging this quantity into the second line of Equation 3.108,

$$\begin{aligned} \vec{0} &= \frac{\partial j}{\partial \vec{v}} - \vec{n}\left(\rho(\vec{v}\cdot\vec{\varphi}) + \rho\frac{\left(\frac{\partial j}{\partial \rho} - \vec{v}\cdot\vec{\varphi}v_n - \psi_{\rho E} v_n\left(\frac{\vec{v}^2}{2}\right)\right)}{v_n} + \psi_{\rho E}\rho\left(\frac{c^2}{\gamma-1} + \frac{\vec{v}^2}{2}\right)\right) \\ &\quad - \vec{\varphi}\left(\rho v_n\right) - \vec{v}\left(\psi_{\rho E}\rho v_n\right) \\ \vec{0} &= \frac{\partial j}{\partial \vec{v}} - \left(\vec{n}\left(\frac{\rho}{v_n}\frac{\partial j}{\partial \rho} + \psi_{\rho E}\rho\left(\frac{c^2}{\gamma-1}\right)\right) + \vec{\varphi}\left(\rho v_n\right) + \vec{v}\left(\psi_{\rho E}\rho v_n\right)\right) \\ \vec{\varphi}(\rho v_n) &= \frac{\partial j}{\partial \vec{v}} - \left(\vec{n}\left(\frac{\rho}{v_n}\frac{\partial j}{\partial \rho} + \psi_{\rho E}\rho\left(\frac{c^2}{\gamma-1}\right)\right) + \vec{v}\left(\psi_{\rho E}\rho v_n\right)\right) \\ \vec{\varphi} &= \frac{\partial j}{\partial \vec{v}}\frac{1}{\rho v_n} - \vec{n}\frac{\partial j}{\partial \rho}\frac{1}{v_n^2} + \psi_{\rho E}\left(-\vec{n}\frac{c^2}{v_n(\gamma-1)} - \vec{v}\right). \end{aligned} \quad (3.110)$$

Plugging this result for $\vec{\varphi}$ into Equation 3.109:

$$\begin{aligned}\psi_\rho &= \left(\frac{\partial j}{\partial \rho} - \vec{v}\cdot\left(\frac{\partial j}{\partial \vec{v}}\frac{1}{\rho v_n} - \vec{n}\frac{\partial j}{\partial \rho}\frac{1}{v_n^2} + \psi_{\rho E}\left(-\vec{n}\frac{c^2}{v_n(\gamma-1)} - \vec{v}\right)\right)v_n - \psi_{\rho E}v_n\left(\frac{\vec{v}^2}{2}\right)\right)\frac{1}{v_n} \\ \psi_\rho &= -\left(\frac{\partial j}{\partial \vec{v}}\cdot\vec{v}\frac{1}{\rho v_n}\right) + \left(\frac{\partial j}{\partial \rho}\frac{2}{v_n}\right) + \psi_{\rho E}\left(\frac{2c^2+\vec{v}^2(\gamma-1)}{2(\gamma-1)}\right).\end{aligned} \tag{3.111}$$

The boundary condition at the outlet in terms of the energy adjoint variable reduces to:

$$\left\{\begin{array}{c}\psi_\rho \\ \vec{\varphi}\end{array}\right\} = \psi_{\rho E}\left\{\begin{array}{c}\frac{2c^2+\vec{v}^2(\gamma-1)}{2(\gamma-1)} \\ -\vec{n}\frac{c^2}{v_n(\gamma-1)} - \vec{v}\end{array}\right\} + \left\{\begin{array}{c} -\left(\frac{\partial j}{\partial \vec{v}}\cdot\vec{v}\frac{1}{\rho v_n}\right) + \left(\frac{\partial j}{\partial \rho}\frac{2}{v_n}\right) \\ \left(\frac{\partial j}{\partial \vec{v}}\frac{1}{\rho v_n} - \vec{n}\frac{\partial j}{\partial \rho}\frac{1}{v_n^2}\right)\end{array}\right\}. \tag{3.112}$$

The remaining adjoint variable $\psi_{\rho E}$ is interpolated from the volume solution, often by taking the value at the nearest node in the volume (0^{th} order interpolation). For supersonic flow, δP is unknown, introducing an additional equation that can be solved to find the value of $\psi_{\rho E}|_{M>1}$:

$$\psi_{\rho E, M_e>1} = \frac{\gamma-1}{v_n^2 - c^2}\left(\frac{\partial j}{\partial \rho}\frac{1}{v_n} + \frac{\partial j}{\partial P}v_n - \frac{\partial j}{\partial \vec{v}}\cdot\vec{n}\frac{1}{\rho}\right). \tag{3.113}$$

Generalized Outflow-Based Functional With A Moving Mesh

For a moving mesh with nonzero $\vec{u}_\Omega$, the equation that must be satisfied is:

$$\left\{\begin{array}{c}\frac{\partial j}{\partial \rho} \\ \frac{\partial j}{\partial \vec{v}} \\ \frac{\partial j}{\partial P}\end{array}\right\}^T \left\{\begin{array}{c}\delta\rho \\ \delta\vec{v} \\ \delta P\end{array}\right\} - \psi^{\mathsf{T}}\left\{\begin{array}{ccc} v_{nr} & \rho\vec{n}^{\mathsf{T}} & 0 \\ (v_{nr})\vec{v} & \rho\left[\bar{\bar{I}}v_{nr} + \vec{v}\otimes\vec{n}\right] & \vec{n} \\ \frac{\phi}{a_0}(v_{nr}) & \rho(v_{nr})\vec{v}^{\mathsf{T}} + \rho H\vec{n}^{\mathsf{T}} & v_n + \frac{1}{a_0}(v_{nr}) \end{array}\right\}\delta V = 0. \tag{3.114}$$

Following the same procedure as used for a stationary mesh,

$$\begin{aligned}\psi_\rho &= \frac{1}{v_{nr}}\left(\frac{\partial j}{\partial \rho} - v_{nr}\vec{v}\cdot\vec{\varphi} - \psi_{\rho E}v_{nr}\frac{\phi}{a_0}\right) \\ &= \frac{1}{v_{nr}}\frac{\partial j}{\partial \rho} - \vec{v}\cdot\vec{\varphi} - \psi_{\rho E}\frac{|\vec{v}|^2}{2},\end{aligned} \tag{3.115}$$

leading to:

$$\begin{aligned}&\delta\vec{v}\cdot\left(\frac{\partial j}{\partial \vec{v}} - \rho\vec{n}\psi_\rho - \rho v_{nr}\vec{\varphi} - \rho\vec{n}(\vec{\varphi}\cdot\vec{v}) - (\rho v_{nr}\vec{v} + \rho H\vec{n})\psi_{\rho E}\right) \\ &= \delta\vec{v}\cdot\left(\frac{\partial j}{\partial \vec{v}} - \rho\vec{n}\left(\frac{1}{v_{nr}}\frac{\partial j}{\partial \rho} - \vec{v}\cdot\vec{\varphi} - \psi_{\rho E}\frac{|\vec{v}|^2}{2}\right) - \rho v_{nr}\vec{\varphi} - \rho\vec{n}(\vec{\varphi}\cdot\vec{v}) - (\rho v_{nr}\vec{v} + \rho H\vec{n})\psi_{\rho E}\right) \\ &= \delta\vec{v}\cdot\left(\frac{\partial j}{\partial \vec{v}} - \rho\vec{n}\left(\frac{1}{v_{nr}}\frac{\partial j}{\partial \rho} - \psi_{\rho E}\frac{|\vec{v}|^2}{2}\right) - \rho v_{nr}\vec{\varphi} - (\rho v_{nr}\vec{v} + \rho H\vec{n})\psi_{\rho E}\right).\end{aligned} \tag{3.116}$$

In order to eliminate dependence on $\delta\vec{v}$,

$$\begin{aligned}\vec{\varphi} &= \frac{1}{\rho v_{nr}}\left(\frac{\partial j}{\partial \vec{v}} - \frac{\rho\vec{n}}{v_{nr}}\frac{\partial j}{\partial \rho} + \psi_{\rho E}\left(\frac{|\vec{v}|^2}{2}\rho\vec{n} - (\rho v_{nr}\vec{v} + \rho H\vec{n})\right)\right)\\ &= \frac{1}{\rho v_{nr}}\frac{\partial j}{\partial \vec{v}} - \frac{\vec{n}}{v_{nr}^2}\frac{\partial j}{\partial \rho} + \psi_{\rho E}\left(\frac{|\vec{v}|^2}{2}\frac{\vec{n}}{v_{nr}} - \vec{v} - H\frac{\vec{n}}{v_{nr}}\right)\\ &= \frac{1}{\rho v_{nr}}\frac{\partial j}{\partial \vec{v}} - \frac{\vec{n}}{v_{nr}^2}\frac{\partial j}{\partial \rho} + \psi_{\rho E}\left(\frac{-c^2}{(\gamma-1)v_{nr}}\vec{n} - \vec{v}\right).\end{aligned} \tag{3.117}$$

Expanding the expression for ψ_ρ:

$$\begin{aligned}\psi_\rho &= \frac{1}{v_{nr}}\frac{\partial j}{\partial \rho} - \vec{v}\cdot\left(\frac{1}{\rho v_{nr}}\frac{\partial j}{\partial \vec{v}} - \frac{\vec{n}}{v_{nr}^2}\frac{\partial j}{\partial \rho}\right) + \psi_{\rho E}\left(-\frac{|\vec{v}|^2}{2} + \frac{c^2 v_n}{(\gamma-1)v_{nr}} + |\vec{v}|^2\right)\\ &= \frac{v_{nr}+v_n}{v_{nr}^2}\frac{\partial j}{\partial \rho} - \frac{1}{\rho v_{nr}}\vec{v}\cdot\frac{\partial j}{\partial \vec{v}} + \psi_{\rho E}\left(\frac{|\vec{v}|^2}{2} + \frac{c^2 v_n}{(\gamma-1)v_{nr}}\right).\end{aligned} \tag{3.118}$$

Summarizing,

$$\begin{Bmatrix}\psi_\rho\\ \vec{\varphi}\end{Bmatrix} = \psi_{\rho E}\begin{Bmatrix}\frac{2c^2 v_n + \vec{v}^2 v_{nr}(\gamma-1)}{2(\gamma-1)v_{nr}}\\ -\vec{n}\frac{c^2}{v_{nr}(\gamma-1)} - \vec{v}\end{Bmatrix} + \begin{Bmatrix}-\left(\frac{\partial j}{\partial \vec{v}}\cdot\vec{v}\frac{1}{\rho v_{nr}}\right) + \left(\frac{\partial j}{\partial \rho}\frac{v_{nr}+v_n}{v_{nr}^2}\right)\\ \left(\frac{\partial j}{\partial \vec{v}}\frac{1}{\rho v_{nr}} - \vec{n}\frac{\partial j}{\partial \rho}\frac{1}{v_{nr}^2}\right)\end{Bmatrix}. \tag{3.119}$$

The final energy term, found when $M_e > 1$ making δP arbitrary, is:

$$\begin{aligned}0 &= \frac{\partial j}{\partial P} - \vec{n}\cdot\vec{\varphi} - \psi_{\rho E}(v_n + \frac{1}{a_0}v_{nr})\\ &= \frac{\partial j}{\partial P} - \vec{n}\cdot\left(\frac{1}{\rho v_{nr}}\frac{\partial j}{\partial \vec{v}} - \frac{\vec{n}}{v_{nr}^2}\frac{\partial j}{\partial \rho} + \psi_{\rho E}\left(\frac{-c^2}{(\gamma-1)v_{nr}}\vec{n} - \vec{v}\right)\right) - \psi_{\rho E}(v_n + \frac{1}{a_0}v_{nr})\\ &= \frac{\partial j}{\partial P} - \frac{\vec{n}}{\rho v_{nr}}\cdot\frac{\partial j}{\partial \vec{v}} + \frac{1}{v_{nr}^2}\frac{\partial j}{\partial \rho} + \psi_{\rho E}\left(-v_n - \frac{v_{nr}}{\gamma-1} + \frac{c^2}{(\gamma-1)v_{nr}} + v_n\right)\\ &= \frac{\partial j}{\partial P} - \frac{\vec{n}}{\rho v_{nr}}\cdot\frac{\partial j}{\partial \vec{v}} + \frac{1}{v_{nr}^2}\frac{\partial j}{\partial \rho} + \psi_{\rho E}\left(\frac{c^2 - v_{nr}^2}{(\gamma-1)v_{nr}}\right).\end{aligned} \tag{3.120}$$

Solving for the energy term,

$$\psi_{\rho E, M_e>1} = \frac{\gamma-1}{v_{nr}^2 - c^2}\left(\frac{\partial j}{\partial \rho}\frac{1}{v_{nr}} + \frac{\partial j}{\partial P}v_{nr} - \frac{\partial j}{\partial \vec{v}}\cdot\vec{n}\frac{1}{\rho}\right). \tag{3.121}$$

When the mesh is stationary $v_n = v_{nr}$, and this reduces to Equation 3.112 through 3.113.

Generalized Outflow-Based Boundary Conditions with Fixed Characteristic Variable

Although the fixed pressure outflow boundary is used in this work, and is the boundary condition currently available in SU2, the alternative characteristic-based condition can be derived in the same way and will be presented here. The significance of the characteristic variables is that they result

from the diagonalization of the Jacobians. This allows us to derive the adjoint boundary conditions consistent with Riemann conditions in the direct flow problem. Because the supersonic case results in all characteristics exiting in the direct problem, and all entering in the adjoint problem, the resulting values of the adjoint variables will be the same. The difference is only relevant for the subsonic case and inside the boundary layer of viscous problems. Transforming Equation 3.105 to be in terms of characteristic variables W:

$$\begin{aligned}
0 = \Psi^T \vec{A}^c \cdot \vec{n} M L \delta W - \frac{\partial j}{\partial U}^T M L \delta W = \Psi^T \vec{A}^c \cdot \vec{n} M L \delta W - \frac{\partial j}{\partial V}^T L \delta W \\
= \delta w_1 \left(-\frac{\partial j}{\partial \rho} + \frac{1}{2} v_n \left(2\psi_\rho + \vec{v}^2 \psi_{\rho e} + 2\vec{v} \cdot \vec{\varphi} \right) \right) \\
+ \delta w_2 \left(\frac{\partial j}{\partial \vec{v}} \cdot \vec{n} + v_n \rho (\vec{\varphi} \times \vec{n} + \psi_{\rho e} \vec{v} \times \vec{n}) \right) \\
+ \delta w_3 \left(\frac{-2}{4c} \left(c \frac{\partial j}{\partial \vec{v}} \cdot \vec{n} + c^2 \frac{\partial j}{\partial P} \rho + \frac{\partial j}{\partial \rho} \rho \right) \right. \\
\left. + \frac{\rho}{4c} \left(\psi_{\rho e} \left(\frac{2c^3 \gamma}{\gamma - 1} + 2c^2 (\frac{v_n \gamma}{\gamma - 1}) + 2c v_n^2 \right) + (v_n + c) \left(2c(\varphi_n) + 2\psi_\rho + \vec{v}^2 \psi_{\rho e} + 2\vec{v} \cdot \vec{\varphi} \right) \right) \right) \\
+ \delta w_4 \left(\frac{-2}{4c} \left(-c (\frac{\partial j}{\partial \vec{v}} \cdot \vec{n}) + c^2 \frac{\partial j}{\partial P} \rho + \frac{\partial j}{\partial \rho} \rho \right) \right. \\
\left. + \frac{\rho}{4c} \left(\psi_{\rho e} \left(\frac{-2c^3 \gamma}{\gamma - 1} + \frac{2c^2 v_n \gamma}{\gamma - 1} - 2c v_n^2 \right) + (v_n - c) \left(2\psi_\rho - 2c\varphi_n + \vec{v}^2 \psi_{\rho e} + 2\vec{v} \cdot \vec{\varphi} \right) \right) \right).
\end{aligned} \tag{3.122}$$

For the subsonic case, $\delta w_4 = 0$ because that characteristic variable is associated with the negative characteristic speed $u - c$ from the direct problem, and so it is positive and exiting the volume in the adjoint problem. The remaining characteristics are entering the volume in the adjoint problem, and so their values are arbitrary. Setting the coefficients for the variation of the first three characteristics to 0 and solving for three of the adjoint variables:

$$\begin{Bmatrix} \psi_\rho \\ \vec{\varphi} \end{Bmatrix} = \psi_{\rho e} \begin{Bmatrix} \frac{c v_n}{\gamma - 1} + \frac{\vec{v}^2}{2} \\ -\vec{v} - \vec{n} \frac{c}{\gamma - 1} \end{Bmatrix} + \begin{Bmatrix} \frac{1}{c + v_n} \left(\frac{\partial j}{\partial \rho} \frac{c + 2v_n}{v_n} - c v_n \frac{\partial j}{\partial P} - \frac{\frac{\partial j}{\partial \vec{v}} \cdot \vec{v}}{\rho} - \frac{c(\vec{v} \times \vec{n}) \cdot (\frac{\partial j}{\partial \vec{v}} \times \vec{n})}{\rho v_n} \right) \\ \frac{\vec{n}}{c + v_n} \left(\frac{\partial j}{\partial P} c - \frac{\partial j}{\partial \rho} \frac{1}{v_n} - \frac{\partial j}{\partial \vec{v}} \cdot \vec{n} \frac{c}{v_n \rho} \right) + \frac{\partial j}{\partial \vec{v}} \frac{1}{\rho v_n} \end{Bmatrix}. \tag{3.123}$$

Applying the expansion of the dot product of the cross products with the normal vector, expanded in Equation A.6,

$$\begin{Bmatrix} \psi_\rho \\ \vec{\varphi} \end{Bmatrix} = \psi_{\rho e} \begin{Bmatrix} \frac{c v_n}{\gamma - 1} + \frac{\vec{v}^2}{2} \\ -\vec{v} - \vec{n} \frac{c}{\gamma - 1} \end{Bmatrix} + \begin{Bmatrix} \frac{1}{c + v_n} \left(\frac{\partial j}{\partial \rho} \frac{c + 2v_n}{v_n} - c v_n \frac{\partial j}{\partial P} - \frac{\frac{\partial j}{\partial \vec{v}} \cdot \vec{v}}{\rho} - \frac{c\left(\left(\vec{v} \cdot \frac{\partial j}{\partial \vec{v}} \right) - v_n \left(\vec{n} \cdot \frac{\partial j}{\partial \vec{v}} \right) \right)}{\rho v_n} \right) \\ \frac{\vec{n}}{c + v_n} \left(\frac{\partial j}{\partial P} c - \frac{\partial j}{\partial \rho} \frac{1}{v_n} - \frac{\partial j}{\partial \vec{v}} \cdot \vec{n} \frac{c}{v_n \rho} \right) + \frac{\partial j}{\partial \vec{v}} \frac{1}{\rho v_n} \end{Bmatrix}. \tag{3.124}$$

$$\begin{Bmatrix}\psi_\rho\\ \vec{\varphi}\end{Bmatrix} = \psi_{\rho e}\begin{Bmatrix}\frac{cv_n}{\gamma-1}+\frac{\vec{v}^2}{2}\\ -\vec{v}-\vec{n}\frac{c}{\gamma-1}\end{Bmatrix} + \begin{Bmatrix}\frac{\partial j}{\partial \vec{v}}\cdot\vec{v}\left(\frac{-1}{\rho}\right)+\frac{1}{c+v_n}\left(\frac{\partial j}{\partial \rho}\frac{c+2v_n}{v_n}-cv_n\frac{\partial j}{\partial P}+\frac{c\left(\vec{n}\cdot\frac{\partial j}{\partial \vec{v}}\right)}{\rho}\right)\\ \frac{\vec{n}}{c+v_n}\left(\frac{\partial j}{\partial P}c-\frac{\partial j}{\partial \rho}\frac{1}{v_n}-\frac{\partial j}{\partial \vec{v}}\cdot\vec{n}\frac{c}{v_n\rho}\right)+\frac{\partial j}{\partial \vec{v}}\frac{1}{\rho v_n}\end{Bmatrix}. \quad (3.125)$$

For the supersonic case, all of the characteristics of the adjoint problem are entering the volume, and so the value of its variation δw_4 is now arbitrary. This means that in order to solve for the adjoint variables the coefficient on this term must be set to zero, which provides an equation we can now solve for the remaining adjoint variable, $\psi_{\rho e}$:

$$\psi_{\rho e, M>1} = \frac{\gamma-1}{v_n^2-c^2}\left(\frac{\partial j}{\partial \rho}\frac{1}{v_n}+\frac{\partial j}{\partial P}v_n-\frac{\partial j}{\partial \vec{v}}\cdot\vec{n}\frac{1}{\rho}\right). \quad (3.126)$$

Note that the result for the supersonic case is identical regardless of the direct problem boundary condition because for supersonic flow all characteristics for the adjoint problem are entering the volume. The supersonic solution for either boundary condition reduces to:

$$\begin{Bmatrix}\psi_\rho\\ \vec{\varphi}\\ \psi_{\rho e}\end{Bmatrix}_{M>1} = \begin{Bmatrix}\left(\frac{\partial j}{\partial \rho}\right)\frac{2v_n^2+\vec{v}^2(\gamma-1)}{v_n(v_n^2-c^2)} & +\left(\frac{\partial j}{\partial \vec{v}}\cdot\vec{n}\right)\frac{-(2c^2-\vec{v}^2(\gamma-1))}{(v_n^2-c^2)\rho} & +\left(\frac{\partial j}{\partial \vec{v}}\cdot\vec{v}\right)\frac{-1}{\rho v_n} & +\left(\frac{\partial j}{\partial P}\right)\frac{-v_n(2c^2+\vec{v}^2(\gamma-1))}{2(c^2-v_n^2)}\\ \left(\frac{\partial j}{\partial \rho}\right)\frac{-\vec{n}-\vec{v}(\gamma-1)}{v_n^2-c^2} & +\left(\frac{\partial j}{\partial \vec{v}}\cdot\vec{n}\right)\frac{\vec{n}c^2+\vec{v}v_n(\gamma-1)}{v_n(v_n^2-c^2)\rho} & +\left(\frac{\partial j}{\partial \vec{v}}\right)\frac{1}{\rho v_n} & +\left(\frac{\partial j}{\partial P}\right)\frac{\vec{n}c^2+\vec{v}v_n(\gamma-1)}{c^2-v_n^2}\\ \left(\frac{\partial j}{\partial \rho}\right)\frac{\gamma-1}{v_n(v_n^2-c^2)} & +\left(\frac{\partial j}{\partial \vec{v}}\cdot\vec{n}\right)\frac{1-\gamma}{\rho(v_n^2-c^2)} & & +\left(\frac{\partial j}{\partial P}\right)\frac{v_n(\gamma-1)}{v_n^2-c^2}\end{Bmatrix}. \quad (3.127)$$

Equation 3.112- 3.127 constitute the generalized outflow boundary conditions for the continuous adjoint method. As these equations utilize $\partial j/\partial V$, the formulation is independent the choice of j, and the gradient information can be provided numerically by an external script, or replaced by analytical expressions for a specific objective. For example, in this work the gradients are provided by an external script that models the flow downstream of the isolator.

3.2.4 Surface Sensitivity

All arbitrary variations of δU have now been eliminated from the equations - by identifying terms that are not arbitrary due to the conditions imposed in the direct problem, and by setting values of Ψ such that the remaining terms are eliminated through the solution of the PDE and boundary conditions derived in previous sections. The only terms remaining are terms depended on δS, and this section will expand and simplify those terms to form the surface sensitivity, $\frac{\partial J}{\partial S}$.

Surface Sensitivity Resulting from Linearized Direct Problem Boundary Conditions

Note that the surface terms of the adjoint formulation, before expansion of δJ, shown in Equation 3.38, do not explicitly contain terms of δS. During the derivation of the boundary conditions in Section 3.2.3, the application of the linearized boundary conditions introduce terms of δS. The reader is directed to Table 3.2 to review the linearized boundary conditions. Since terms from the

Table 3.5: Sensitivity Contributions from Direct Problem Boundary Conditions

Boundary Type	Contribution to Surface Sensitivity
Flow Tangency	$\{\vartheta\nabla\cdot(\vec{v}-\vec{u}_\Omega)+\nabla(\vartheta)\cdot(\vec{v}-\vec{u}_\Omega)\}$
No-Slip	$\left\{\vec{n}\cdot\left(\vartheta(\partial_n(\vec{v}-\vec{u}_\Omega))+\psi_{\rho E}\left(-\partial_n(\vec{v}-\vec{u}_\Omega)\cdot\bar{\bar{\sigma}}\right)+\left(\bar{\bar{\Sigma}}^{\varphi}+\bar{\bar{\Sigma}}^{\psi_{\rho E}}\right)\cdot(\partial_n(\vec{v}-\vec{u}_\Omega))\right)\right\}$
Adiabatic Wall	$\left\{-\mu_{tot}^2 c_p\nabla_S(\psi_{\rho E})\cdot\nabla_S(T)-\psi_{\rho E}(P(\nabla\cdot\vec{v})-\bar{\bar{\sigma}}:\nabla\vec{v}+(\vec{q}_{\rho\vec{v}}\cdot\vec{v}-q_{\rho E})\right\}$
Isothermal Wall	$\left\{-\mu_{tot}^2 c_p\partial_n(\psi_{\rho E})\partial_n(T)\right\}$

Table 3.6: Sensitivity Contributions from Functional Variations

Functional	Contribution to Surface Sensitivity
Generalized force-based	$\left\{\frac{\partial j}{\partial\vec{f}}\cdot[\vec{q}_{\rho\vec{v}}-\partial_t(\rho\vec{v})]+\nabla\frac{\partial j}{\partial\vec{f}}:(\bar{\bar{I}}P-\bar{\bar{\sigma}})-(\bar{\bar{I}}P-\bar{\bar{\sigma}})\cdot\vec{n}\cdot\partial_n\frac{\partial j}{\partial\vec{f}}\right\}$ $+\left\{2H_m(\vec{f}\cdot\frac{\partial j}{\partial\vec{f}}-j(\vec{f}))\right\}$
Generalized heat flux (isothermal)	$\left\{\nabla\cdot\left(\frac{\partial j}{\partial(\partial_n T)}\nabla T\right)+2H_m\left(\frac{\partial j}{\partial(\partial_n T)}\partial_n T-j(\partial_n T)\right)\right\}$
Generalized temperature (adiabatic)	$\left\{\frac{\partial j}{\partial T}\partial_n T-2H_m j(T)\right\}$

functional have been left as arbitrary expressions, we can easily separate the contribution to the surface sensitivity from the boundary conditions and functionals respectively. Setting $j=0$ and applying the adjoint boundary conditions for each solid wall boundary type results in the expressions shown in Table 3.5.

Contributions to Surface Sensitivity From Functionals

Surface-based objectives also contribute terms to the surface sensitivity. The details can be found in the expansion of the various functionals, in Table 3.4, and generalized functionals in Equation 3.99. The contribution to the surface sensitivity from functionals are shown in Table 3.6. Note that the heat-flux and temperature-based functionals are only provided for isothermal and adiabatic conditions respectively.

3.2.5 Summary of Adjoint Equations

The boundary conditions and expressions for surface sensitivity derived in previous sections will be summarized here for easy reference and comparison. Table 3.7 shows the equations for the generalized functional, under the limitations that the functional should have $\frac{\partial j}{\partial(\partial_n(T))}$ for an adiabatic wall. Note that at inlets and outlets the local Mach number determines how many of the adjoint variables are specified by the boundary conditions. This is due to the characteristic directions of the solution. To construct the adjoint equations for some particular objective, one takes the volume

Table 3.7: Adjoint Boundary Conditions for Generalized Functionals

Boundary Condition	Adjoint Boundary Condition (Stationary Domain)
Flow tangency	$\vec{\varphi}\cdot\vec{n} = \frac{\partial j}{\partial \vec{f}}\cdot\vec{n} - \psi_{\rho E}\vec{v}\cdot\vec{n}$ $\left(\vec{\varphi}\cdot\vec{n} = \frac{\partial j}{\partial \vec{f}}\cdot\vec{n}\right)$
Adiabatic wall	$\vec{\varphi} = \frac{\partial j}{\partial \vec{f}} - \psi_{\rho E}\vec{v}$ $\partial_n(\psi_{\rho E}) = \frac{1}{c_p\mu_{tot}^2}\frac{\partial j}{\partial T}$ $\left(\begin{array}{c}\vec{\varphi} = \frac{\partial j}{\partial \vec{f}} \\ \partial_n(\psi_{\rho E}) = \frac{1}{c_p\mu_{tot}^2}\frac{\partial j}{\partial T}\end{array}\right)$
Isothermal wall	$\vec{\varphi} = \frac{\partial j}{\partial \vec{f}} - \psi_{\rho E}\vec{v}$ $\psi_{\rho E} = -\frac{\partial j}{\partial(\partial_n T)}\frac{1}{c_p\mu_{tot}^2}$ $\left(\begin{array}{c}\vec{\varphi} = \frac{\partial j}{\partial \vec{f}} \\ \psi_{\rho E} = -\frac{\partial j}{\partial(\partial_n T)}\frac{1}{c_p\mu_{tot}^2}\end{array}\right)$
Pressure-based Outflow (inviscid & viscous neglecting $\delta\mu$)	$\left\{\begin{array}{c}\psi_\rho \\ \vec{\varphi}\end{array}\right\} = \psi_{\rho E}\left\{\begin{array}{c}\frac{2c^2 v_n + \vec{v}^2 v_{nr}(\gamma-1)}{2(\gamma-1)v_{nr}} \\ -\vec{n}\frac{c^2}{v_{nr}(\gamma-1)} - \vec{v}\end{array}\right\} + \left\{\begin{array}{c}-\left(\frac{\partial j}{\partial \vec{v}}\cdot\vec{v}\frac{1}{\rho v_{nr}}\right) + \left(\frac{\partial j}{\partial \rho}\frac{v_{nr}+v_n}{v_{nr}^2}\right) \\ \left(\frac{\partial j}{\partial \vec{v}}\frac{1}{\rho v_{nr}} - \vec{n}\frac{\partial j}{\partial \rho}\frac{1}{v_{nr}^2}\right)\end{array}\right\}$ $\psi_{\rho E, M_e>1} = \frac{\gamma-1}{v_{nr}^2 - c^2}\left(\frac{\partial j}{\partial \rho}\frac{1}{v_{nr}} + \frac{\partial j}{\partial P}v_{nr} - \frac{\partial j}{\partial \vec{v}}\cdot\vec{n}\frac{1}{\rho}\right)$ $\left(\begin{array}{c}\left\{\begin{array}{c}\psi_\rho \\ \vec{\varphi}\end{array}\right\} = \psi_{\rho E}\left\{\begin{array}{c}\frac{2c^2 + \vec{v}^2(\gamma-1)}{2(\gamma-1)} \\ -\vec{n}\frac{c^2}{v_n(\gamma-1)} - \vec{v}\end{array}\right\} + \left\{\begin{array}{c}-\left(\frac{\partial j}{\partial \vec{v}}\cdot\vec{v}\frac{1}{\rho v_n}\right) + \left(\frac{\partial j}{\partial \rho}\frac{2}{v_n}\right) \\ \left(\frac{\partial j}{\partial \vec{v}}\frac{1}{\rho v_n} - \vec{n}\frac{\partial j}{\partial \rho}\frac{1}{v_n^2}\right)\end{array}\right\} \\ \psi_{\rho E, M_e>1} = \frac{\gamma-1}{v_n^2 - c^2}\left(\frac{\partial j}{\partial \rho}\frac{1}{v_n} + \frac{\partial j}{\partial P}v_n - \frac{\partial j}{\partial \vec{v}}\cdot\vec{n}\frac{1}{\rho}\right)\end{array}\right)$
Inlet	$\psi_{\rho E, M<1} = -\vec{\varphi}\cdot\vec{n}\frac{(\gamma-1)}{\gamma\vec{v}\cdot\vec{n} - \vec{u}_\Omega\cdot\vec{n}}$ $\left(\psi_{\rho E, M<1} = -\vec{\varphi}\cdot\vec{n}\frac{(\gamma-1)}{\gamma\vec{v}\cdot\vec{n}}\right)$

term shown in Equation 3.39, find the partial derivatives of the functional, and inserts those partial derivatives into the the appropriate boundary conditions from Table 3.7. After solving the PDE defined by these equations, one finds the surface sensitivity by summing the appropriate terms from Table 3.5 and Table 3.6, which may also depend on the partial derivatives. At this point, we have the adjoint formulation for any number of functionals - assuming that the partial derivative values can be provided. Following sections will expand on how those terms can be provided for outflow boundaries.

Implementation of Boundary Conditions

The outflow boundary conditions are imposed as weak boundary conditions. Rather than directly setting the values of the variables (as in a strong boundary condition), corresponding flux is imposed. The values will satisfy the boundary conditions at convergence.

3.3 Exploiting the Generalized Adjoint Method to Address Previously-Inaccessible Functionals

With the introduction of the generalized form, stated in terms of partial derivatives of the functional, it is now possible to find the adjoint solution for arbitrary functions defined outside the computational volume. This arbitrary function is not limited to being an analytical function, or to being a particular level of fidelity. As long as the required partial derivative quantities can be produced, the adjoint solution can be found - although it is possible to result in boundary values that will produce a numerically challenging solution that will not converge easily. This means that the range of functions that can be addressed by the continuous adjoint method now includes empirical functions, "black box" functions for which an explicit derivative is not available, and any other model where the appropriate partial derivatives can be produced.

In contrast, established adjoint methods depend on every model used to find the functional being contained in the adjoint formulation - whether using continuous or discrete methodologies, as discussed in Section 2.3.3.

The generalized form also facilitates an understanding of what functionals can be addressed by the adjoint method. Only those functionals with defined partial derivatives that can be used to cancel terms within the adjoint formulation can be addressed. Previously, this limitation also implicitly limited the admissible functions to those that are defined as simple, integral, functions on the boundaries of the computational volume. In other words, limited to functions where the partial derivatives can be expressed as a function of the flow state on the relevant boundary.

This generalized form has many potential applications. In Chapter 5 and Chapter 6, this methodology is applied to a multi-fidelity flowpath where the surface sensitivity on a scramjet inlet is desired for functions defined based on a low fidelity model of the combustor and nozzle of the engine. In this case, the partial derivatives are based on perturbing one-dimensional inputs to the low-fidelity model combined with the Jacobian that transforms between one-dimensionalized quantities and the local values on the outflow of the CFD volume.

This methodology could also be applied to experimental results—if it is known experimentally how quickly some QOI changes with respect to the outflow quantities, this information can be introduced to the adjoint solution to find which portions of the upstream geometry may effect the QOI. It could also be applied to situations where the external model is a higher-fidelity simulation. For example, many highly complex simulations of combustors exist, some of which include the adjoint formulation. Either by using the adjoint result of the higher-fidelity simulation or by perturbing the inputs if the adjoint method is not available, this methodology could be used to find the surface sensitivity on the lower-fidelity component, thereby reducing the computational cost necessary to analyze the entire system.

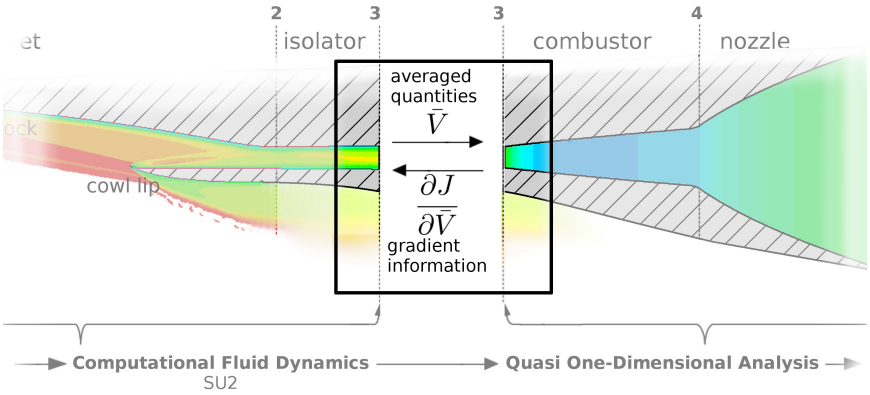


Figure 3.2: Information transfer occurring at station 3; see also Figure 1.1.

In this work, as a one-dimensional model is used to provide the partial derivative terms, the one-dimensionalization of the flow quantities must be taken into account. There are several established options for one-dimensionalization, which come with various advantages and disadvantages. For any averaging technique, some manipulation of the equations is required in order to transfer the appropriate partial derivative terms to the solution. The interface where the averaged quantities and partial derivative terms are transferred is illustrated in Figure 3.2, focusing on one portion of the flowpath shown in Figure 1.1.

A selection of one-dimensionalization techniques are included in this section. The mass flux averaged and area-averaged techniques have been implemented in SU2 within the generalized boundary condition. Although only one-dimensionalization is addressed in this section, it should be noted that in order to find the adjoint of a function of any intermediate quantity (such as distortion, or other methods of one-dimensionalization), the only step needed to derive the appropriate boundary conditions is the derivation of the Jacobian between the partial derivatives with respect to primitive variables and the partial derivatives with respect to the intermediate functions. In the illustrative cases included, those intermediate functions are averaged flow quantities, but any number of other values could be used.

3.3.1 One-Dimensionalization

When the objective is expressed as a function of averaged outflow values $\bar{V}$,

$$J = J(\bar{V}). \tag{3.128}$$

The variation of the objective becomes:

$$\begin{aligned}\delta J &= \frac{\partial J}{\partial \bar{V}} \int_\Gamma \frac{\partial \bar{V}}{\partial V} \delta V ds = \int_\Gamma \frac{\partial J}{\partial \bar{V}} \frac{\partial \bar{V}}{\partial V} \delta V ds \\ &= \int_\Gamma \frac{\partial j}{\partial V} \delta V ds,\end{aligned} \tag{3.129}$$

where the term $\frac{\partial J}{\partial \bar{V}}$ is constant over Γ, and the Jacobian $\frac{\partial \bar{V}}{\partial V}$ may vary over the boundary depending on the method of one-dimensionalization chosen. The last line of Equation 3.129 puts these terms into the same form used in Equation 3.98. In other words,

$$\frac{\partial j}{\partial V} = \frac{\partial J}{\partial \bar{V}} \frac{\partial \bar{V}}{\partial V}. \tag{3.130}$$

The equations for a generalized outflow functional can then be used directly. For each one - dimensionalization method, we need only to produce the Jacobian $\frac{\partial \bar{V}}{\partial V}$.

Area Averaging

Area averaging offers a simple and straight-forward option. For area-averaged quantities,

$$\begin{aligned}\bar{V} &= \left\{ \begin{array}{c} \int \rho ds \\ \int v_n ds \\ \int P ds \end{array} \right\} \frac{1}{\mathcal{A}_e} \\ \frac{\partial \bar{V}}{\partial V} &= \frac{1}{\mathcal{A}_e} \left\{ \begin{array}{ccc} 1, & 0, & 0 \\ \vec{0}, & \vec{n}, & \vec{0} \\ 0, & 0, & 1 \end{array} \right\}.\end{aligned} \tag{3.131}$$

Mass Flux Averaging

Mass flux averaging one-dimensionalizes the flow by using a mass flux weighted average of each of the flow quantities of interest. This method conserves mass flux.

$$\begin{aligned}\bar{V} &= \frac{\int \rho v_n V ds}{\int \rho v_n ds} = \frac{\int \rho v_n V ds}{\dot{m}} \\ \bar{\rho} &= \frac{\int \rho^2 v_n}{\dot{m}} \\ \bar{v} &= \sqrt{\frac{\int \rho v_n |\vec{v}|^2}{\dot{m}}} \\ \bar{P} &= \frac{\int \rho v_n P}{\dot{m}}.\end{aligned} \tag{3.132}$$

$$\frac{\partial \bar{V}}{\partial V} = \begin{Bmatrix} \frac{\partial \bar{\rho}}{\partial \rho} & \frac{\partial \bar{v}}{\partial \rho} & \frac{\partial \bar{P}}{\partial \rho} \\ \frac{\partial \bar{\rho}}{\partial \vec{v}} & \frac{\partial \bar{v}}{\partial \vec{v}} & \frac{\partial \bar{P}}{\partial \vec{v}} \\ \frac{\partial \bar{\rho}}{\partial P} & \frac{\partial \bar{v}}{\partial P} & \frac{\partial \bar{P}}{\partial P} \end{Bmatrix} = \begin{Bmatrix} \frac{2\rho v_n}{\dot{m}} - \frac{\bar{\rho}}{\dot{m}} v_n, & \frac{1}{2\bar{v}} \left(\frac{v_n |\vec{v}|^2}{\dot{m}} - \frac{\bar{v}^2}{\dot{m}} v_n \right), & \frac{v_n P}{\dot{m}} - \frac{\bar{P}}{\dot{m}} v_n \\ \frac{\rho^2 \vec{n}}{\dot{m}} - \frac{\bar{\rho}}{\dot{m}} \rho \vec{n}, & \frac{1}{2\bar{v}} \left(\frac{\rho v_n \vec{v} + \rho |\vec{v}|^2 \vec{n}}{\dot{m}} - \frac{\bar{v}^2}{\dot{m}} \rho \vec{n} \right), & \frac{\rho P \vec{n}}{\dot{m}} - \frac{\bar{P}}{\dot{m}} \rho \vec{n} \\ 0 & \vec{0} & \frac{\rho v_n}{\dot{m}} \end{Bmatrix}$$
$$= \begin{Bmatrix} \frac{v_n}{\dot{m}} (2\rho - \bar{\rho}), & \frac{v_n}{2\bar{v}\dot{m}} \left(|\vec{v}|^2 - \bar{v}^2 \right), & \frac{v_n}{\dot{m}} \left(P - \bar{P} \right) \\ \frac{\rho \vec{n}}{\dot{m}} (\rho - \bar{\rho}), & \frac{\rho}{2\bar{v}\dot{m}} \left(v_n \vec{v} + |\vec{v}|^2 \vec{n} - \bar{v}^2 \vec{n} \right), & \frac{\rho \vec{n}}{\dot{m}} \left(P - \bar{P} \right) \\ 0, & \vec{0}, & \frac{\rho v_n}{\dot{m}} \end{Bmatrix}. \tag{3.133}$$

Langley Distortion Method

The Langley distortion method[93–95] provides an additional technique for one-dimensionalization that has benefits in conserving convective flux quantities and leading to quantifications of distortion. This methodology is based on Conserved Mass/Momentum/Energy (CMME) methods. This is a class of averaging methodologies that have the property that the flux of mass, momentum, and energy computed with the averaged quantities is equivalent to the integral flux values. The Langley distortion method can be summarized as consisting of area averaged pressure, mass flux averaged total enthalpy, and mass flux averaged mean kinetic energy. The averaged density and velocity can be found as functions of these quantities using the equation of state.

$$\begin{aligned} \bar{P} &= \frac{\int P ds}{\mathcal{A}} \\ \bar{h} &= \frac{\int \rho v_n h_t ds}{\dot{m}} = \frac{\int \rho v_n \left(\frac{P}{\rho} \frac{\gamma}{\gamma - 1} + \frac{1}{2} |\vec{v}|^2 \right) ds}{\dot{m}} \\ \overline{(\rho |\vec{v}|^2)} &= \frac{\int \rho^2 v_n \vec{v}^2 ds}{\dot{m}} \\ \bar{\rho} &= \frac{\bar{P} \frac{\gamma}{\gamma - 1} + \frac{1}{2} (\rho |\vec{v}|^2)}{\bar{h}} \\ \bar{v} &= \sqrt{\frac{\overline{(\rho |\vec{v}|^2)}}{\bar{\rho}}}. \end{aligned} \tag{3.134}$$

This method conserves the fluxes of mass, kinetic energy, and enthalpy. The Jacobian for this technique can be found similar to the other one-dimensionalization techniques, although its terms will be more complex. This method of one-dimensionalization is included as it was used for reference values in some initial optimization cases.

3.4 Combinations of Functionals

The adjoint equations form a linear PDE system, and as such the principle of superposition can be applied. In other words, the linear combination of adjoint solutions for multiple objectives is equivalent to the adjoint solution for a linear combination of those objectives. In the surface formulation of the adjoint method for objectives defined on the boundaries this translates to a superposition of the boundary conditions.

In other words,

$$\mathcal{F}(x_1 + x_2) = \mathcal{F}(x_1) + \mathcal{F}(x_2), \tag{3.135}$$

where $\mathcal{F}$ represents the adjoint PDE, and x_1 and x_2 represent the two sets of boundary conditions associated with two objectives. This can be accomplished by summing the contribution to boundary condition values that are dependent on the functional terms.

Referring to Table 3.7, where the adjoint boundary conditions are expanded in terms of partial derivatives, this can be seen easily since, for some objective $J = w_1 J_1 + w_2 J_2$:

$$\frac{\partial(w_1 j_1(\vec{f}) + w_2 j_2(\vec{f})}{\partial \vec{f}} = w_1 \frac{\partial j_1}{\partial \vec{f}} + w_2 \frac{\partial j_2}{\partial \vec{f}}. \tag{3.136}$$

In order to implement this methodology within SU2, modifications were made to the code structure to (1) allow lists of objectives where previously a single quantity was allowed, (2) progressively sum terms in the formation of the adjoint boundary conditions, and (3) store a combined objective in the output of the direct problem. Although this methodology, and its implementation are relatively simple, the result is useful to many problems. In the literature, multiple objectives have sometimes been treated either with the discrete adjoint method[96] or by deriving the boundary conditions for a single combination of objectives.[97]

The difference in this implementation lies in its generality, being applicable to any combination of objectives where each individual objective can be addressed by the adjoint method, and in its flexibility. This method is flexible in that it can address sums of more than two objectives, objectives defined on multiple surfaces, and more complex functions of these objectives. All of this is done while producing gradients or surface sensitivities at the approximate cost of finding the sensitivity of a single function. This is achieved mainly because, from the perspective of the numerical solution of the adjoint PDE, a combination of objectives has been converted into a single functional.

More complex functions can be addressed by exploiting the chain rule. For example, if a quadratic penalty function is introduced in order to address a constraint:

$$\begin{aligned} &\min_x J = J_1 + P(J_2) \\ &\min_x J = J_1 + (\max(J_2 - J_{2,lim}, 0.0)^2), \end{aligned} \tag{3.137}$$

the gradient used during the optimization can be found by:

$$\delta J = \delta J_1 + (2\max(J_2 - J_{2,lim}, 0.0))\,\delta J_2, \tag{3.138}$$

where the weighting value w_2 in Equation 3.137 is now replaced by the partial derivative $w_2 = \frac{\partial P}{\partial J_2}$. This means that the implementation of the multi-objective adjoint formulation does not need to be changed in order to address more complex function - one only needs to modify the application of the weighting values. This is implemented through the python scripts associated with SU2.

Chapter 4

Methodology Details

This chapter discusses the implementation of the methodology used in this work. Section 4.1 discusses Computational Fluid Dynamics methodology including a review of numerical methods based on existing literature in Section 4.1.2, along with the implementation of methods developed in this work including generalized outflow-based functionals in Section 4.1.3 and Section 4.1.4. Models of combustion and expansion implemented for this work and based on literature are discussed in Section 4.2. Design variables used for optimization problems are discussed in Section 4.3.

4.1 Computational Fluid Dynamics Implementation

The open-source CFD suite SU2, developed in the Aerospace Design Lab at Stanford University, was used to generate flow solutions and the adjoint solution. SU2 uses the Finite Volume Method (FVM) to solve partial differential equations on unstructured meshes. Further information is available in.[59,63] In the Reynolds-Averaged-Navier-Stokes (RANS) equations, a turbulence model is used to account for the Reynolds stresses. The one-equation Spalart-Allmaras[86] and two-equation SST k-omega[87] turbulence models are available. The continuous adjoint equations are solved in a similar fashion, re-using methods implemented to solve partial differential equations and the information generated by the flow solver.

A new boundary condition and other modifications were implemented in this work in order to produce the continuous adjoint solution for generalized outflow-based objective functions. The boundary conditions are discussed at length in Chapter 3, and further implementation details are discussed in later sections. Here CFD is described in a general manner, and some details of the implementation of CFD in SU2. Section 4.1.1 discusses the volume discretization used by SU2, followed by Section 4.1.2 that discusses a selection of numerical methods that can be applied to this type of problem. Section 4.1.3 and Section 4.1.4 discuss implementation specific to the developments in this work within the SU2 framework. A number of options were tested during the simulations

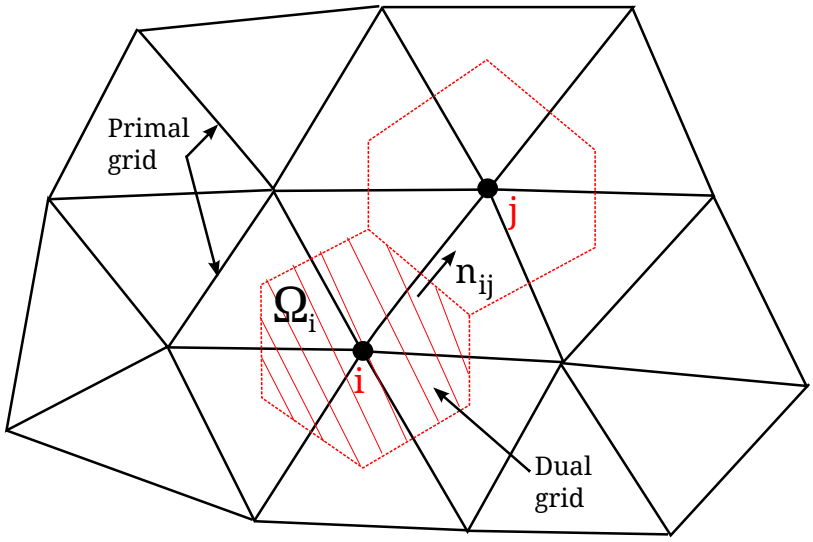


Figure 4.1: Dual grid control volume, image credit: Sanchez.[98]

completed in this work. The JST 2^{nd} order method, discussed in Section 4.1.2, was found through trial and error to operate the best from the perspective of reliably converging numerical solutions without a need to modify options such as limiter settings between designs in the optimization process. The SST turbulence model was used, with settings drawn from previous simulations of similar geometries.

4.1.1 Discretization

SU2 uses a vertex-based finite volume method (FVM) discretization with a median-dual scheme. In this scheme, the control volumes are constructed by connecting the midpoints and centroids of all of the edges and faces of the cells of the initial, or primal, grid that share a specific node.[63] A median-dual control volume is illustrated in Figure 4.1. This is an edge-based structure, where the fluxes are computed over the edges defined in the primal grid.

To transform the governing equations onto the discretized domain, the governing PDE is integrated over a control volume, and the divergence theorem is applied in order o arrive at a semi-discretized form. For the RANS equations previously given in Equation 2.34,

$$\int_{\Omega_i} \frac{\partial U}{\partial t} d\Omega + \sum_{j \in N(i)} (\tilde{F}^c_{ij} + \tilde{F}^{\nu k}_{ij}) \Delta S_{ij} - Q|\Omega_i| = \int_{\Omega_i} \frac{\partial U}{\partial t} d\Omega + R_i(U) = 0, \tag{4.1}$$

where $N(i)$ is the set of neighboring nodes connected to node i, $|\Omega_i|$ is the volume of the dual control volume, ΔS_{ij} is the area of face associated with the edge than runs between node i and node

j, and $R_i(U)$ is the numerical residual at node i. The approximate flux terms $\tilde{F}_{ij}$ are numerical approximations projected in the normal direction along the edge ij, and evaluated at the midpoint of the edge. Q is a source term, and U is the vector of state variables. The approximation of the convective and viscous fluxes is determined by the numerical scheme chosen. Some of these schemes will be discussed in Section 4.1.2. The solution procedure loops over the edges in the mesh, calculating the numerical fluxes, and then numerically integrates the quantities to produce the residual $R_i(U)$ at each node point.

Gradients of the flow variables are required at the cell faces in order to compute the terms needed for the viscous fluxes, and for second order approximations. These are computed using either Green-Gauss or weighted least-squares. In SU2, several numerical schemes have been implemented,[63] including Jameson-Schmidt-Turkel (JST),[99] Roe,[100] and Lax-Friedrich.[101] Section 4.1.2 will discuss a selection of numerical methods.

4.1.2 Numerical Methods

This section will contain an overview and discussion of numerical schemes used to solve the governing equations of fluid flow on a discretized domain. This is included to provide the necessary background for the CFD methods used in this work. The reader is referred to the literature for further detail on the development of numerical methods,[84,99–102] and on their implementation in SU2.[59,60,63,103] The governing equations of fluid flow were introduced in Section 2.3.2.

There are many methods available in the discretization of the PDE governing fluid flow. Flux-splitting,[100,102,104] limiters,[105] local time-stepping, and multi-grid[106–108] methods were developed to address issues of stability, shock-capturing, and convergence behavior of the numerical PDE solutions. Some of these methods will be discussed here, and more thorough information is available in the sources cited.

The notation used in describing numerical schemes employs subscripts that indicate the space discretization, and superscripts that indicate the solution time. Each numerical scheme is generally providing a solution for the state variables at the next time step, U_i^{n+1}, in terms of the solution at the adjacent points and the properties of the discretization: $U_i^n, U_{i+1}^n, U_{i-1}^n, f_i^n, f_{i+1}^n, f_{i-1}^n, \Delta t,$ and Δx. The subscript i indicates the current spatial grid point, n indicates the current time, U is the vector of state variables, f is the flux vector at the indicated point, Δx is the local spatial grid spacing, and Δt is the local time step.

Courant Friedrichs-Lewy[109] number, $CFL = \left\| \frac{a\Delta t}{\Delta x} \right\|$, is a parameter that is used to control the speed with which a numerical solution converges, with limitations related to the stability of the numerical method chosen. In the definition of the CFL number, a is the local convection speed, which is also the spectral radius of the convective Jacobian. This term arises from stability analysis of the numerical methods, and becomes a convenient parameter to modify when attempting to control the stability and convergence speed of the numerical solutions.

Popular and widely studies schemes of second-order accuracy with a combined space-time discretization are the Lax-Wendroff family of centered schemes.[84] These schemes produce oscillations around discontinuities, which motivate the use of limiters. The introduction of the Lax-Wendroff method in 1960[101] enabled many future schemes and was an important step in being able to solve both the Euler and Navier-Stokes equations.[84] Prior to discussing Lax-Wendroff second-order schemes, the first-order Lax-Friedrich scheme will be presented.

Local Time-Stepping

For steady solutions, where time accuracy is not required, it is possible to accelerate convergence by using local time steps. The benefit of local time-stepping is that areas of the domain that have a smooth, stable solution can progress quickly by locally increasing the time step, while areas of the domain that require a smaller time step remain stable. The time step varies as function of the local propagation speed $u + c$.

Lax-Friedrichs Scheme

The Lax-Friedrichs scheme is a first-order scheme that stabilizes the explicit central scheme resulting from taking the central difference of the first derivative of the flux term. The explicit central scheme is unstable, and the one-dimensional Lax-Friedrichs scheme stabilizes it by replacing U_i^n with the average of the two adjacent points:

$$U_i^{n+1} = \frac{U_{i+1}^n + U_{i-1}^n}{2} - \frac{\tau}{2}(f_{i+1}^n - f_{i-1}^n), \tag{4.2}$$

where $\tau = \frac{\Delta t}{\Delta x}$. This scheme effectively stabilizes the procedure by adding a term proportional to the second derivative, providing dissipation. The Lax-Friedrichs scheme is conditionally stable, with:

$$\sigma = \frac{\Delta t}{\Delta x}|u + c| \leq 1, \tag{4.3}$$

where σ is the CFL number, and $u + c$ is the spectral radius, or maximum characteristic value, of the Jacobian A^c.

Lax-Wendroff Family of Schemes

Implicit methods have unconditional linear stability, allowing faster solution times as the CFL number can now be increased without limit. Nonlinear stability, and accuracy must still be considered, and in practice leads to a limit on the CFL number. The family of Lax-Wendroff implicit schemes can be stated as:

$$[1 + \alpha\sigma\bar{\delta} + \frac{1}{2}(\beta\sigma^2 + \gamma)\delta^2]\Delta U_i^n = -\sigma\bar{\delta}U_i^n + \frac{1}{2}(1 - 2\alpha)\sigma^2\delta^2 U_i^n, \tag{4.4}$$

where α, β, and γ are the coefficients that are varied to arrive at the various schemes. This form of the Lax-Wendroff family of schemes can be found in Hirsch[84] Chapter 17.4.2. The CFL number is indicated by σ, and δ is a difference operator, where δ^2 indicates the central second difference, $\bar{\delta}$ is the average between a forward and backwards difference, and δ is a central difference. The implicit methods combine time and space discretizations, by contrast to the explicit schemes that introduce dissipation in order to stabilize the schemes. Lerat[110] is credited with developing implicit schemes of the Lax-Wendroff type. The family of schemes described with Equation 4.4 can be discussed in terms of the values of the real coefficients α, β, and γ, as well as associated limitations on the CFL number σ based on stability analysis.

MacCormack

The MacCormack predictor-corrector scheme is a popular member of the Lax-Wendroff family of schemes, which utilizes operator splitting. Operator splitting speeds up matrix solutions and transforms complex problems into a sequence of simpler problems. This methodology also results in greater accuracy. MacCormack and Paullay,[102] used a finite difference operator with second order accuracy for inviscid flows with shock waves. Their methodology uses a multi-step process, referred to as a "predictor-corrector scheme". They also introduce the use of a local CFL number for greater efficiency when a local region requires a stricter CFL condition, and so a smaller step size. The scheme combines first-order forward difference and a first-order backward difference, each of which are individually unstable for some circumstances, but are stable in combination. The details of this method can be found in Section 17.2.2 of Hirsch.[84]

JST

The JST method, developed by Jameson, Schmidt, and Turkel,[99] is a finite volume, central scheme with independent time integration that uses multi-stage Runge-Kutta time stepping. This scheme was developed for inviscid, transonic flow, and also operates well at other conditions. The JST scheme introduces dissipative terms, and offers greater stability without the need for splitting. Dissipative terms are constructed from a blend of second and fourth order differences that can be tuned with coefficient values, and depend on the local gradient of pressure. The one-dimensional version of this method can be shown to be stable for CFL numbers up to 2.[99] Further discussion of this scheme, and the higher-order dissipation terms can be found in Section 17.3.3 of Hirsch.[84]

Boundary Conditions

The boundary conditions are divided into physical boundary conditions, which arise from the physical definition of the problem, and numerical conditions, which apply to the remaining variables that have not been specified by the physical conditions and must be determined from the numerical solution. The boundary conditions should be applied in a way that is compatible with the accuracy

and stability of the numerical scheme applied to the volume of the domain. The number of physical variables specified at the boundary depends on the propagation of information from the boundary towards the interior of the solution. The numerical boundary conditions must be compatible with the specified physical variables as well as with the discretized equations. Specific equations that are applied at different types of boundaries (solid walls, outflow, and inflow boundaries) were discussed in Section 3.1.4.

Boundary conditions determined by the direction(s) of propagation within the solution are referred to as *characteristic boundary conditions*. The *characteristic variables* or *Riemann invariants* are found from the diagonalization of the convective Jacobian, and represent the quantities that are invariant along the characteristic directions of the solution. The number of characteristic directions pointing inwards, flowing into the volume of the solution, determines the number of physical boundary conditions that will be used. The application of compatibility relationships can also be used. Extrapolation can be used to form the numerical boundary conditions.

4.1.3 Implementation of Generalized Outflow-Based Functionals

Derivations of generalized functionals have been provided in Section 3.2. For the purposes of this work, the characteristic-based with fixed pressure outflow boundary, assuming a stationary domain ($\vec{u}_\Omega$), was implemented. Implementation of further generalized functionals are left for future work. This section summarizes the modifications made to SU2 that were necessary to implement the adjoint boundary conditions for a generalized functional based on one-dimensionalized outflow properties.

The methodology used was based on inputting a vector of coefficients to the adjoint solver. These coefficients are the partial derivatives with respect to averaged outflow quantities. Area averaging and mass-flux averaging Jacobians were implemented within the outflow boundary condition. Output of one-dimensionalized outflow quantities was also added to the operation of SU2.

Modifications were also made to the python scripts used with SU2 such that these coefficients could be automatically computed with a python script during optimization routines. The script must have a default name ("downstream_function.py"), and include functions "downstream_function(config, state)" and "downstream_gradient(config, state)" that respectively return a scalar value of the functional and a vector of the gradients with respect to the one-dimensionalized primitive variables. The variables "config" and "state" are used to store information from the flow solution and configuration file, and are generated by the python scripts that have been developed with SU2. One-dimensionalized quantities are stored in the "state" variable.

4.1.4 Implementation of Multi-Objective Functionals

Several modifications were required to implement multiple objectives. These included modifications to the data structures within SU2 to allow listing of multiple objective functions, as well as loops over this list where previously a single value had been used. Modifications were also required within

the expressions of boundary conditions and sensitivity computation. A scaling factor is applied to force-based functionals to convert between the dimensional projected force and non-dimensional force coefficients. This was previously applied after solving for the adjoint variables using boundary conditions based on the dimensional quantities. For the multi-objective formulation to work properly all scaling factors must be applied prior to calculating the adjoint variables.

4.2 Models of Combustion & Expansion

This section will describe the models used to simulate the flowpath downstream of the isolator. So far, the objective function has been treated entirely as a black box. In theory that black box could contain any level of fidelity, as long as the appropriate gradient terms required by Equation 3.123 and 3.126 are accessible. In this work, a relatively low fidelity model has been used to compute the installed thrust, while still incorporating information about maximum combustor pressure and a relationship between the capture area and the external drag of the vehicle. This choice of model allows the application of the methodology developed in Section 3.2 to objectives that are relevant to the design problem of a hypersonic engine and that were not previously accessible to the continuous adjoint method, while keeping the computational cost lower. No modification to the methodology described in Chapter 3 would be required to apply these methods with more advanced models - for example models including finite rate or non-equilibrium chemistry, flamelet models, more detailed geometry of the combustion and expansion components, and other details and phenomena that are outside the scope of this work.

Figure 4.2 illustrates the flow of information during the optimization loop. The design variables $\vec{x}$ are input into both the direct CFD solver and into the external function, which requires the flow solution to provide the gradients needed for the adjoint formulation and the evaluation of the objective function defined in the external function. Because this external function is implemented as a Python script, it makes it possible to obtain the adjoint formulation for new objective functions without recompiling the SU2 C++ code. The gradients required for the boundary condition are computed via finite difference within the python script, and passed into the configuration information for the adjoint simulation.

Area averages or mass-flux weighted averages over the outflow of the isolator were used to link the CFD model to the one-dimensional or control volume models at station 3 shown in Figure 1.1 and Figure 3.2. An estimate of the affect on drag of changes in geometry is included, in order to ensure that we consider only design changes where the increased engine efficiency outweighs increases in external drag. This is particularly important for the design problem addressed, since it is expected that reductions in capture area might increase efficiency but also increase the wave drag. Increasing the ratio between the nozzle exit area and the capture area increases drag as this leads to a greater external surface area and structural weight. The capture area may change as the inlet geometry

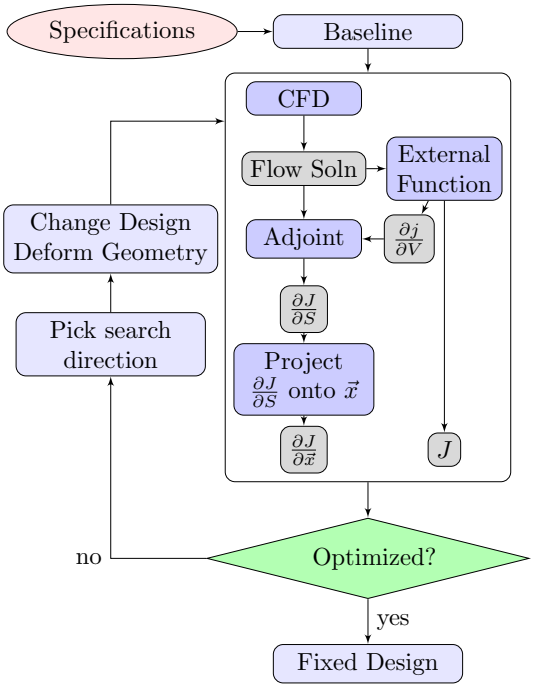


Figure 4.2: Flow chart showing relation of external script and CFD.

is modified, which affects the external drag. Estimates for drag are calculated from equations for conceptual design,[111] and include lift-induced drag based on estimated additional weight of the structure due to the engine length and areas, volume wave drag assuming a the area of a truncated cone based on entrance and nozzle areas, and friction drag. These estimates neglect contributions from the larger vehicle geometry and drag sources such as control surface gaps and other features that are outside the scope of this work. The estimated drag described here will be referred to as D_{est}. More detailed evaluation of the drag is important for design, but outside the scope of this work.

At station 3 of the flowpath through the engine, the air enters a combustion chamber. For this work we will assume that the fuel and air are sufficiently mixed, and use a 1^{st} order model for the combustion and expansion processes. One method of combustion analysis is to treat the flow as a one-dimensional channel with heat addition. This follows analysis from Shapiro,[112] and has been used in previous work by Kline[32] and by Smart[21,22] as well as Heiser & Pratt.[80] This method uses the conservation of mass, momentum, and energy along with the definition of Mach number and the equation of state for the gas. Shapiro's analysis includes the derivatives of the ratio of specific heats γ and the gas constant R, however these will be neglected here.

Following the analysis of Shapiro,[112] these equations are derived starting with the equation of

state,

$$\begin{aligned} p &= \rho R T \\ \log p &= \log \rho + \log R + \log T \\ \frac{dp}{p} &= \frac{d\rho}{\rho} + \frac{dT}{T}. \end{aligned} \tag{4.5}$$

From the definition of Mach number,

$$\frac{dM^2}{M^2} = \frac{dv^2}{v^2} - \frac{dT}{T} \tag{4.6}$$

Mass, momentum, and energy conservation give us:

$$\begin{aligned} \frac{dv}{v} &= -\frac{d\rho}{\rho} - \frac{dA}{A} \\ \frac{dp}{p} + \frac{\gamma M^2}{2}\frac{4C_f dx}{D} + \frac{\gamma M^2}{2}\frac{dv^2}{v^2} &= 0 \\ \frac{dT}{T} + \frac{\gamma - 1}{2}M^2\frac{dv^2}{v^2} &= \left(1 + \frac{\gamma - 1}{2}M^2\right)\frac{dT_t}{T_t} \end{aligned} \tag{4.7}$$

Skin friction has been taken into account in the momentum equation with $\tau_w = C_f \rho v^2 \frac{1}{2}$ and wetted area $dA_w = 4Adx/D$, where D is the hydraulic diameter. Although in a real combustor additional terms would be required to account for mass addition, for simplicity we will assume that the mass is added immediately at the beginning of the combustor. The resulting ordinary differential equation governing the change in Mach number with known area change and total temperature distribution is:

$$\frac{dM}{dx} = M\left(\frac{1 + \frac{\gamma_b - 1}{2}M^2}{1 - M^2}\right)\left\{-\left(\frac{1}{A}\frac{dA}{dx}\right) + \frac{1 + \gamma_b M^2}{2}\left(\frac{1}{T_t}\frac{dT_t}{dx}\right)\right\} \tag{4.8}$$

The area change and the total temperature are now needed in order to find the Mach number distribution. Note that a singularity exists at $M = 1.0$, which in the implementation used has been circumvented by taking a more approximate relation when the Mach number is within a tolerance. The area change is known from the geometry of the duct, which is specified. The temperature change is determined by a mixing profile and heat transfer with the wall of the channel dQ. The skin friction coefficient C_f is assumed to be 0.002. With h_{pr} as the enthalpy of the combustion products, f as the fuel fraction, and η_b as the combustion efficiency:

$$dT_t = dH_t/c_p = \frac{1}{c_p}\left(h_{pr} f \eta_b d\tau - dQ\right) \tag{4.9}$$

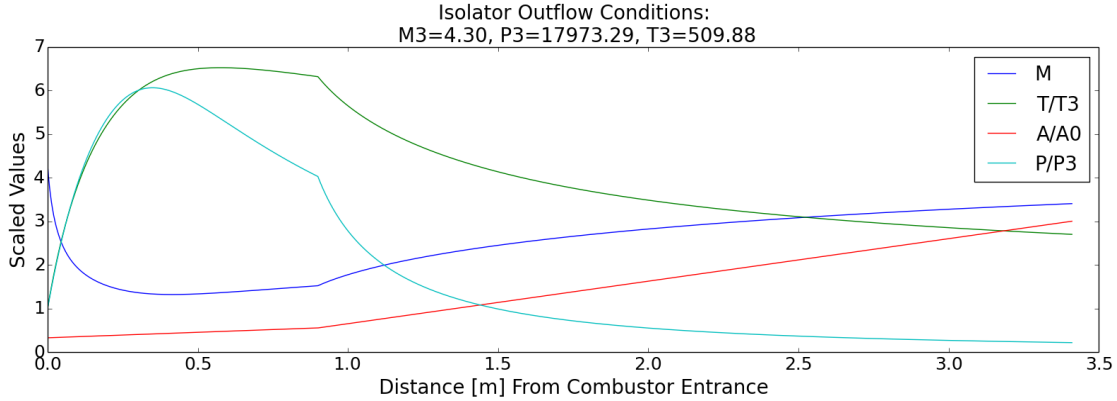


Figure 4.3: Example output from combustion model & expansion model. Conditions match initial geometry and Mach 7 flow.

Where the function τ is:

$$\tau(x) = \left\{ \frac{\theta\chi}{1+(\theta-1)\chi} \right\} \qquad \chi = \frac{x - x_3}{x_4 - x_3} \tag{4.10}$$

This profile is defined by an assumed shape parameter θ, and assumed burner efficiency, η_b. Following the values used in Heiser & Pratt,[80] $\eta_b = 0.8$ and θ=5.The ratio of specific heats, γ_b, is determined from the temperature and tabulated values calculated using the method of element potentials[113] with 9 species. The nozzle of the vehicle is modeled using the same method, with $d\tau = 0$. The outputs of this model provide the quantities necessary to calculate the stream thrust as well as the pressure, Mach, and temperature profiles along the combustor and nozzle.

4.2.1 Gas Properties

As gas temperatures increase, the assumption of constant specific heats breaks down. Chemical reactions may occur, and additional changes to the thermodynamic properties of the gas result from the addition of fuel and the combustion process. In order to account for these phenomena, the method of element potentials was used to construct thermochemical tables in the same format as JANAF tables, for a 6-species mixture of Nitrogen and Oxygen and for a 9-species mixture of Nitrogen, Oxygen, and Hydrogen. These tables were calculated at a reference pressure of 1e5 Pa. By calculating the tables beforehand, the computational cost of updating the specfic heat and gas constant of the fluid is reduced. The assumption applied by this method is that the gas has reached chemical equilibrium at every point.

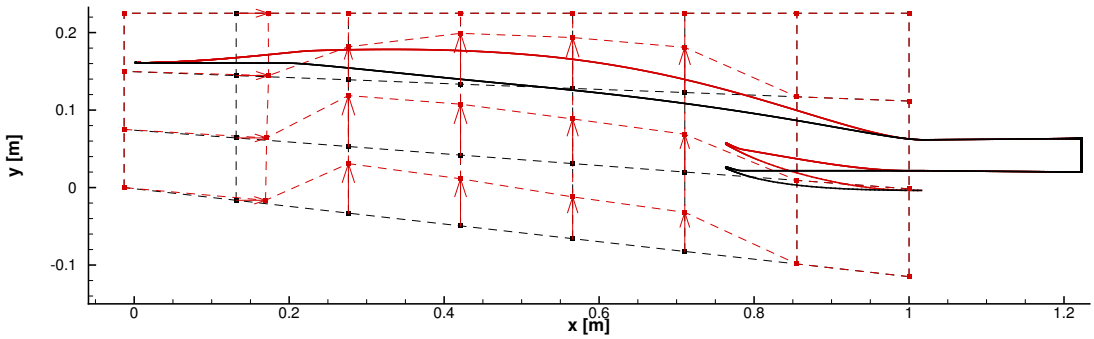


Figure 4.4: Two-dimensional inlet deformed by FFD technique. Initial geometry in black, deformed geometry in red.

4.3 Design Variables

A Free-Form Deformation[114] (FFD) box technique is used to achieve smooth shape deformations in both the verification cases in Chapter 5 and the optimization problems presented in Chapter 6. The FFD strategy has become a popular geometry parameterization technique[115,116] due to the ability to define a wide range of smooth deformations with simple parameters. An initial box surrounding the object to be redesigned (in this work, the compression ramp and cowl lip of a hypersonic inlet) is parameterized as a Bézier solid, parameterized by Bernstein polynomials B^i:

$$X(u,v,w) = \sum_{i,j,k=0}^{l,m,n} P_{i,j,k} B_j^l(u) B_j^m(v) B_k^n(w), \tag{4.11}$$

where l, m, and n are the orders of the Bernstein polynomials, with one polynomial needed for each of the three dimensions. The surface of the deformed object is transformed into the parametric coordinates u, v, $w \in [0, 1]$. Control points are defined on the surface of the box in terms of the indices i, j, and k and the direction of deformation. When the control points are modified the box enclosing the geometry is deformed, all the points inside the box inherit a smooth deformation defined by the mapping in Equation 4.11. An example of FFD control point deformation appears in Figure 4.4.

The FFD strategy provides the displacement of the boundaries of the computational volume, and the remaining volume vertices are deformed using a classical sprint method. A stiffness matrix is defined based on a choice of constant stiffness, stiffness inversely proportional to wall distance in order to maintain boundary layer integrity in the mesh, or stiffness inversely proportional to the cell volume in order to preserve areas of higher mesh refinement. Mesh deformations are then computed by iteratively solving for the deformations that produce force equilibrium.

4.4 Third-Party Software

In addition to the simulation suite SU2,[63] which was developed in the Aerospace Design Lab at Stanford University and on which the author is a developer, several additional software packages were used in the course of this research. Of particular note is the optimization algorithm SNOPT developed by Gill, Murray, and Saunders.[82] This routine is accessed through the interface PyOpt.[117] Both in the application of optimization routines and in physical models used for combustion and expansion, Python[118] was used. The generation of computational grids was accomplished using the meshing software Pointwise, and visualization of results including plots of surface sensitivities, gradient comparisons, and flow fields was accomplished using Tecplot.

Chapter 5

Verification

5.1 Verification Test Case

The verification geometry is a 7.5° ramp 40 mm downstream of the inflow of the domain. The flow conditions are Mach 6.0, 470° K total temperature, and 1×10^7 per meter Reynolds number.

For this problem we assume a symmetry plane on the upper boundary, and the objectives will be defined on the right-hand plane of the volume shown in Figure 5.1. This case matches geometry and flow conditions presented in literature.[119] The FFD box used is also shown in Figure 5.1. The

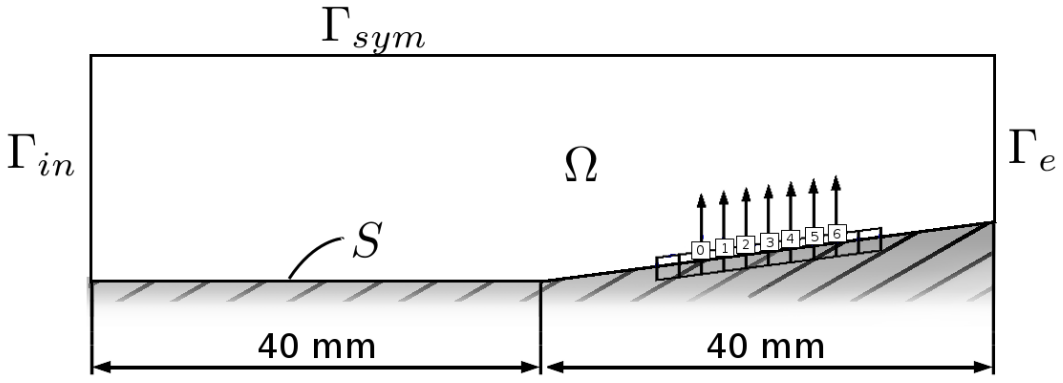


Figure 5.1: Geometry used for verification. Design variables are numbered at the associated FFD control point.

design variables are the 7 control points labelled 0-6. A sample deformation, with exaggerated scale for visualization purposes, is shown in Figure 5.2.

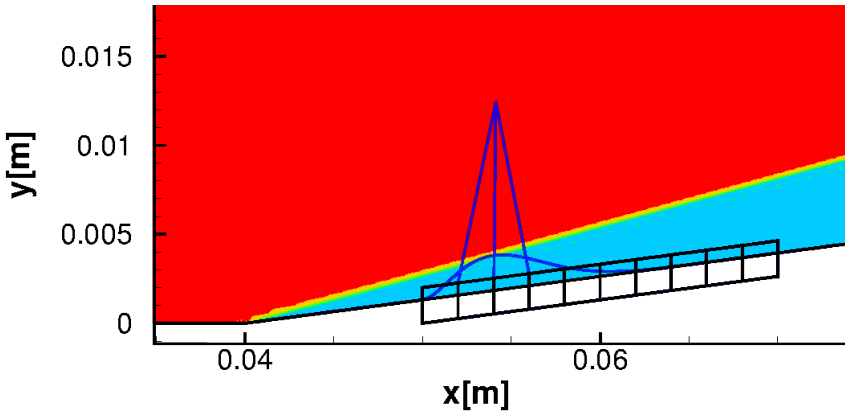


Figure 5.2: Exagerated deformation of the 0^{th} variable, with flow solution.

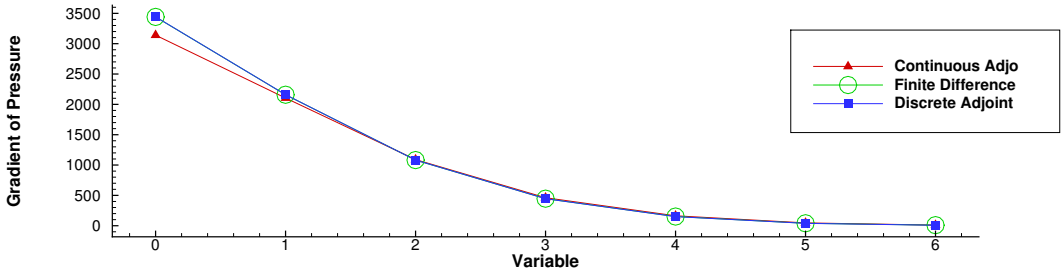


Figure 5.3: Gradient of area averaged static pressure.

5.2 Verification of Continuous and Discrete Adjoint Gradients for Static Pressure

Figure 5.3 and Table 5.1 compare the gradient calculated for area-averaged static pressure using finite difference and adjoint methods. For this function,

$$\begin{aligned} J &= \frac{\int_{\Gamma_e} P ds}{\int_{\Gamma_e} ds} \\ \frac{\partial j}{\partial V} &= \left\{ 0, \vec{0}, \frac{1}{\mathcal{A}_e} \right\}. \end{aligned} \tag{5.1}$$

From Figure 5.3 the gradients match well over all the design variables, with the largest absolute error appearing in the 0^{th} variable, closest to the sharp leading edge of the ramp. Discrepancies between the continuous adjoint and finite difference are discussed in Section 5.6.

Table 5.1: Gradient results for area average static pressure.

Var.	Finite Difference ($\Delta x = 1 \times 10^{-5}$)	Continuous Adjoint	% Error relative to Finite Difference	Discrete Adjoint	% Error relative to Finite Difference
0	3.44×10^3	3.14×10^3	-8.75 %	3.44×10^3	0.04 %
1	2.16×10^3	2.10×10^3	-2.49 %	2.16×10^3	0.04 %
2	1.08×10^3	1.09×10^3	0.77 %	1.08×10^3	0.03 %
3	4.46×10^2	4.62×10^2	3.55 %	4.45×10^2	-0.14 %
4	1.51×10^2	1.64×10^2	8.49 %	1.50×10^2	-0.52 %
5	4.06×10^1	4.69×10^1	15.38 %	3.93×10^1	-3.34 %
6	9.12×10^0	1.07×10^1	17.58 %	7.80×10^0	-14.43 %

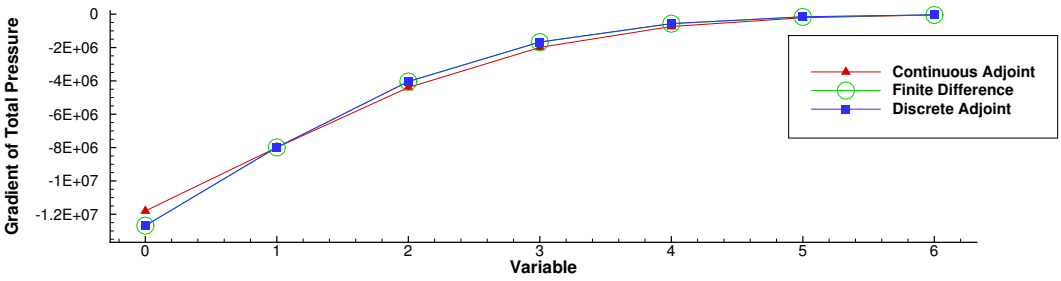


Figure 5.4: Gradient of area-averaged total pressure at the outflow of the computational volume.

5.3 Verification of Continuous and Discrete Adjoint Gradients for Total Pressure

With area averaged total pressure as the objective function,

$$J = \frac{\int_{\Gamma_e} P\left(1+\frac{\gamma-1}{2}M^2\right)^{\frac{\gamma}{\gamma-1}} ds}{\int_{\Gamma_e} ds} = \frac{\int_{\Gamma_e} P\left(1+\frac{\gamma-1}{2}\frac{\vec{v}^2\rho}{\gamma P}\right)^{\frac{\gamma}{\gamma-1}} ds}{\int_{\Gamma_e} ds}$$
$$\frac{\partial j}{\partial V} = \left\{ \frac{\gamma-1}{2}\frac{\vec{v}^2}{\gamma P}, \frac{(\gamma-1)\rho}{\gamma P}\vec{v}, -\frac{\gamma-1}{2}\frac{\vec{v}^2\rho}{\gamma P^2} + \left(1+\frac{\gamma-1}{2}M^2\right)^{\frac{\gamma-1}{\gamma-1}} \right\} \frac{d}{\mathcal{A}_e} \quad , \tag{5.2}$$

where as a shorthand $d = P\left(1+\frac{\gamma-1}{2}M^2\right)^{\frac{1}{\gamma-1}}$. Figure 5.4 and Table 5.2 compare the gradient calculated for area-averaged total pressure. Once again the largest absolute error is in the 0^{th} variable close to the ramp leading edge. This function, which depends on density and velocity in addition to pressure, seems to have slightly higher errors than the gradient of static pressure alone.

Table 5.2: Gradient results for area-average total pressure

Var.	Finite Difference ($\Delta x = 1 \times 10^{-8}$)	Continuous Adjoint	% Error relative to Finite Difference	Discrete Adjoint	% Error relative to Finite Difference
0	-1.27 $\times 10^7$	-1.17 $\times 10^7$	-7.90 %	-1.27 $\times 10^7$	-0.01 %
1	-7.99 $\times 10^6$	-7.91 $\times 10^6$	-1.00 %	-7.99 $\times 10^6$	-0.02 %
2	-4.04 $\times 10^6$	-4.35 $\times 10^6$	7.73 %	-4.04 $\times 10^6$	-0.08 %
3	-1.67 $\times 10^6$	-1.97 $\times 10^6$	17.85 %	-1.67 $\times 10^6$	-0.33 %
4	-5.71 $\times 10^5$	-7.30 $\times 10^5$	27.94 %	-5.62 $\times 10^5$	-1.51 %
5	-1.63 $\times 10^5$	-2.11 $\times 10^5$	28.91 %	-1.53 $\times 10^5$	-6.37 %
6	-4.69 $\times 10^4$	-4.45 $\times 10^4$	-5.08 %	-3.82 $\times 10^4$	-18.54 %

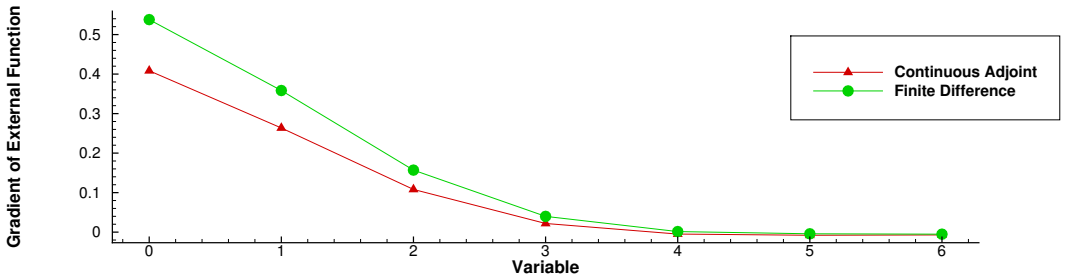


Figure 5.5: Gradient of specific installed thrust computed in an external script.

5.4 Verification of Continuous Adjoint Gradients for a Generalized Objective: Area-Average Based

Figure 5.5 and Table 5.3 compare the gradient calculated for an external function, the specific installed thrust as described in Section 4.2,

$$J = \frac{\mathcal{F}_{un} - D_{est}}{\dot{m}} \tag{5.3}$$

Both direct and adjoint problems were converged to 9 orders of magnitude reduction in the density residual. The largest absolute error in the gradient occurs towards the leading edge of the ramp angle, close to a sharp edge in the solid wall geometry. A further discussion of sources of error in the continuous adjoint is included in Section 5.6. Despite the discrepancies between the continuous adjoint and finite difference results, these results indicate that the continuous adjoint will provide gradients with the correct sign, and proportional magnitude, to appropriately guide a gradient-based optimization process.

Table 5.3: Gradient results for specific installed thrust using the generalized adjoint method.

Var.	Finite Difference ($\Delta x = 1 \times 10^{-5}$)	Continuous Adjoint	% Error relative to Finite Difference
0	5.38 $\times 10^{-1}$	4.08 $\times 10^{-1}$	-24.07 %
1	3.58 $\times 10^{-1}$	2.64 $\times 10^{-1}$	-26.44 %
2	1.57 $\times 10^{-1}$	1.08 $\times 10^{-1}$	-31.13 %
3	3.99 $\times 10^{-2}$	2.19 $\times 10^{-2}$	-45.03 %
4	1.41 $\times 10^{-3}$	-4.87 $\times 10^{-3}$	-445.56 %
5	-4.24 $\times 10^{-3}$	-7.89 $\times 10^{-3}$	86.02 %
6	-5.17 $\times 10^{-3}$	-7.02 $\times 10^{-3}$	35.68 %

Table 5.4: Gradient results for the sum of total pressure on the outflow of the CFD volume and drag coefficient on the surface of a ramp.

Var.	$\frac{\partial C_D}{\partial x_i}$	$\frac{\partial \bar{P}_t}{\partial x_i}$	$\frac{\partial\left(C_D \times 10^5 + \bar{P}_t \times 10^{-5}\right)}{\partial x_i}$ (simultaneous)	$\frac{\partial C_D \times 10^5}{\partial x_i} + \frac{\bar{P}_t \times 10^{-5}}{\partial x_i}$ (separate)
0	-1.51140586 $\times 10^{-5}$	-1.48721592 $\times 10^{7}$	-1.50232998 $\times 10^{2}$	-1.50232998 $\times 10^{2}$
1	-4.18860614 $\times 10^{-6}$	-9.36202300 $\times 10^{6}$	-9.40390906 $\times 10^{1}$	-9.40390906 $\times 10^{1}$
2	-1.37971476 $\times 10^{-6}$	-4.39568841 $\times 10^{6}$	-4.40948556 $\times 10^{1}$	-4.40948556 $\times 10^{1}$
3	-4.61739113 $\times 10^{-7}$	-1.58315833 $\times 10^{6}$	-1.58777572 $\times 10^{1}$	-1.58777572 $\times 10^{1}$
4	9.50142780 $\times 10^{-8}$	-4.31543419 $\times 10^{5}$	-4.30593276 $\times 10^{0}$	-4.30593276 $\times 10^{0}$
5	5.42320259 $\times 10^{-7}$	-8.02000234 $\times 10^{4}$	-7.47768208 $\times 10^{-1}$	-7.47768208 $\times 10^{-1}$
6	1.14402343 $\times 10^{-6}$	-2.82232556 $\times 10^{3}$	8.6179087**9** $\times 10^{-2}$	8.6179087**4** $\times 10^{-2}$

5.5 Verification of Multiple Objective Gradients

In order to test the implementation of multiple objective boundary conditions, which superimpose the weighted boundary conditions for the individual objectives in order to obtain the combined sensitivity at the same cost as a single objective, we observe the gradients produced by the combined method, and when the gradients are computed separately. The two objectives are the total pressure averaged over the outflow boundary, and the drag coefficient on the solid ramp surface. Since the gradients are of significantly different magnitude, weights of 1×10^{-5} and 1×10^{5} are applied to the total pressure and drag coefficient respectively. The results are tabulated in Table 5.4, and illustrate that the gradient calculated simultaneously by superimposing boundary conditions and the sum of the gradients computed separately are identical, out to the 9^{th} digit that differs only on the 6^{th} variable. The combined adjoint method is of about the same computational cost as a single-objective evaluation.

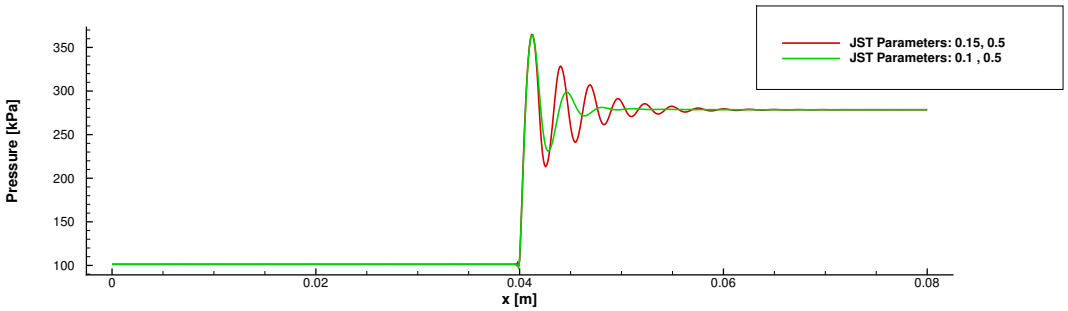


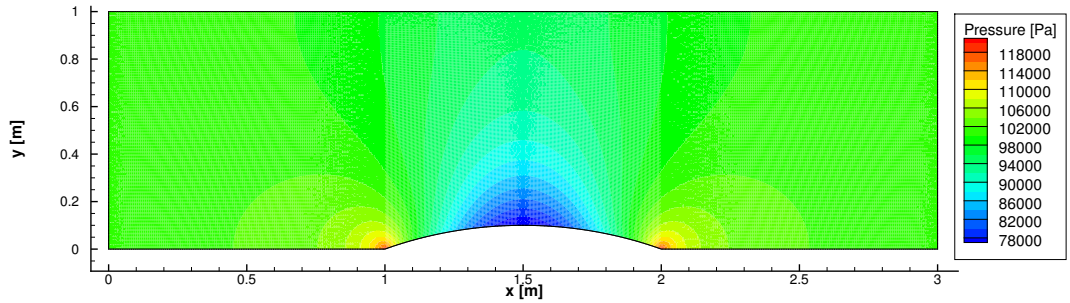
Figure 5.6: Pressure distributions for a 7.5° ramp in Mach 6.0 flow, with various values for the JST 1^{st}, 2^{nd}, and 4^{th} order artificial dissipation coefficients.

5.6 Discussion of Discrepancies

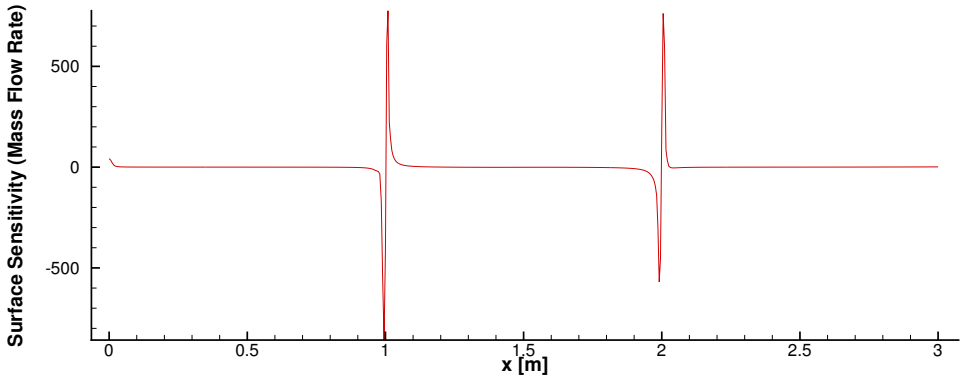
All numerical simulations have some level of error—discretization, cancellation, and truncation are common sources of error, and are often dealt with by increasing the resolution of the mesh, decreasing step size in the finite difference method, or introducing higher-order methods. This section will discuss the sources of error that may have led to the discrepancies in gradient evaluations seen in this chapter.

In supersonic flow, the presence of shocks presents problems for the continuous adjoint method, since these discontinuous changes in flow quantities result in the solution no longer being continuously differentiable. For the direct flow problem, accurately capturing the shock and limiting numerical oscillations of the solution in the neighborhood of the shock are achieved through the use of limiters that apply first-order numerical methods in the neighborhood of the shock, or in the case of the JST method introduce artificial dissipation near large gradients of pressure. To illustrate the effect of, and need for, these methods, the pressure distribution over the ramp case described in Section 5.1 is shown for the 2^{nd} order JST method in Figure 5.6. In this plot, two pressure contours are shown with different values used for the artificial dissipation coefficients. Although the shocks are numerically continuous, avoiding the issues of a discontinuous solution, they may be the source of some error in the adjoint solution, due to either a lack of convergence of the residuals near the shock in the direct solution and/or due to reduced solution accuracy from increasing the level of artificial dissipation.

In addition to flow discontinuities, geometry discontinuities should also be considered. The continuous adjoint method is derived with the assumption that the boundaries are continuously differentiable, in other words class C^1. However, this is often not the case, and sharp edges may exist on the solid walls and at in internal flow problems at the junctions between inflow or outflow boundaries and solid walls. Since these points are often coincident with shocks, as is the case with the ramp case, the effect of sharp edges is illustrated in Figure 5.7(a) and Figure 5.7(b) for a subsonic bump-in-channel example in Mach 0.5 flow. The surface sensitivity of mass flow rate at the outflow



(a) Pressure contours for Mach 0.5 bump-in-channel.



(b) Sensivity of mass flow rate for Mach 0.5 bump-in-channel.

of the volume can be seen in Figure 5.7(b) to diverge close to the sharp edges of the geometry that exist at the edges of the bump. At these points the accuracy of the continuous adjoint degrades due to cancellation error introduced near values approaching infinity. In practice, neglecting sensitivity values near sharp edges in the geometry is an effective method of reducing these errors, and this is the method used in SU2, with an optional parameter that controls the distance from a sharp edge where the sensitivity will be neglected.

The effect of neglecting these surface sensitivities in the neighborhood of a shock is seen in Figure 5.7, where sensitivities near the sharp ramp leading edge have been neglected. Although the inaccuracies introduced by values approaching infinity have been removed from the surface sensitivity plot, information about the numerical solution near the shock has been lost. In practice, coefficients controlling sharp edge removal are tuned for the specific case. In the cases in this chapter, the sharp edge removal coefficient setting ranged between 0.1 and 0.35.

The error in the continuous adjoint may also be attributed to neglecting the effects of volume mesh deformation in the continuous problem. Kavvadias, Papoutsis-Kiachagias and Gianakoglou[120] including terms of the volume mesh deformation into the surface formulation of the continuous

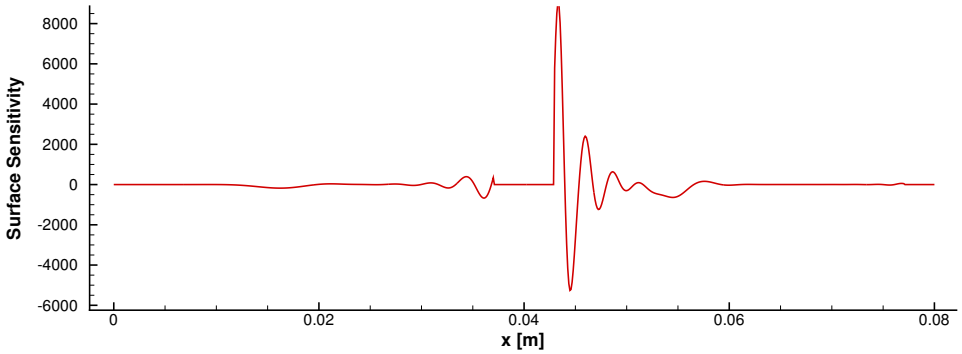


Figure 5.7: Surface sensitivity of specific installed thrust, with sharp edge removal coefficient of 0.1 for a 7.5° ramp in Mach 6 flow.

adjoint has been successful in achieving results closer to the finite difference, and further discuss the sources of inaccuracy for the surface formulation of the continuous adjoint.

Chapter 6

Results

6.1 Design Problem Description

Section 2.2.2 discussed potential quantities of interest to a scramjet design problem. Some quantities of interest, such as the total pressure ratio over the inlet, were accessible using methods established prior to this work. Others, such as the full engine thrust or the maximum pressure in the combustor, were either not accessible or would have required significant increases in computational cost. A selection of quantities will be investigated in this section: total pressure ratio over the inlet P_{tr}, maximum pressure in the combustor/nozzle system $P_{\max}$, total temperature ratio over the engine flowpath $\tau_e = \frac{\tau_{10}}{\tau_0}$, and specific installed thrust $\frac{F_{un}-D_{est}}{\dot{m}}$. The installed specific thrust incorporates the uninstalled stream thrust F_{un}, an estimate of external drag that depends on the capture area and geometry parameters of the combustor and nozzle, and the mass flow rate $\dot{m}$ through the engine. Models used to compute the thrust, estimated external drag, and other quantities downstream of the hypersonic inlet are described in Section 4.2. The integrated heat flux, IHF, will also be considered. With the exception of P_{tr} and IHF, the quantities of interest investigated in this section could not previously be addressed by the continuous adjoint method. By coupling lower-fidelity methods to the higher-fidelity CFD and continuous adjoint methods, the sensitivity of the remaining quantities of interest can be addressed.

In order to start from reasonable performance, and to demonstrate the ability of methods in this work to improve on already highly-designed inlet geometries, a REST-class inlet geometry will be used. Scramjet design techniques are introduced in Section 1.2 and Section 2.2.1.

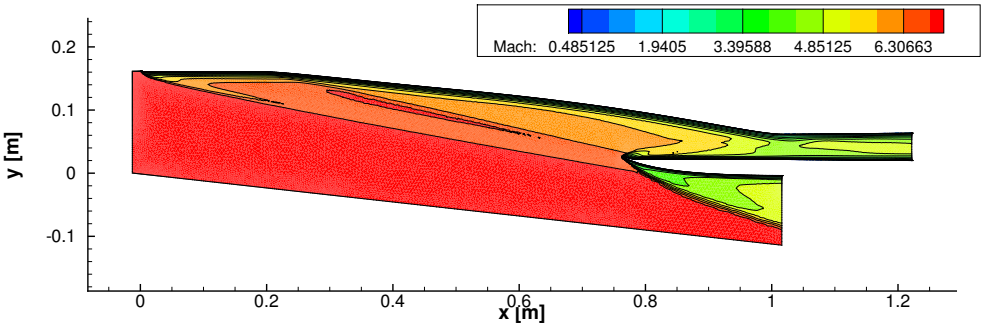


Figure 6.1: REST inlet cut along symmetry plane.

6.2 Two-Dimensional Design Problem

The initial geometry, shown in Figure 6.1 is a Rectangular-to-Elliptical-Shape-Transition (REST) scramjet inlet that was designed in the Hypersonic Airbreathing Propulsion Branch at NASA Langley. The associated study is detailed by Ferlemann & Gollan.[25,36] This is "case 17" from a parametric study[25] that examined the performance of inlets designed using the REST method with varying parameters such as lip angle the cross-sectional shapes of the geometry. The REST geometry will be used as a starting point on top of which we will apply high-fidelity gradient-based optimization. A two-dimensional version of this geometry generated by taking a symmetry-plane slice will also be used. REST-class inlets are described further in Section 2.2.1. The design conditions for this inlet were freestream conditions of $M_0 = 7$, $q_0 = 82.833$ kPa (1730 psf) with a desired compression ratio of 37.55. These conditions correspond to ≈ 30 km altitude. The Reynolds number is $\approx 5 \times 10^6$ m.

The meshes were generated using the Pointwise mesh generation software. These meshes are unstructured, using "t-rex" meshing to create anisotropic cells near the wall. The two-dimensional viscous mesh has approximately $400,000$ vertices, mesh spacing at the wall of 1×10^{-6} m with a growth rate of 1.05 in order to obtain a y^+ value of less than 1 along the wall surfaces, as shown in Figure 6.2. The mesh size of approximately $400,000$ vertices was chosen following a mesh refinement study using structured meshes.

6.2.1 Initial Design Point Performance

Flow Description

In order to confirm that SU2 is able to appropriately simulate this inlet, the results can be compared to an alternate simulation tool. Figure 6.3 compares surface heat flux and pressure at the surface of the two-dimensional symmetry plane of the REST-class inlet, using two different CFD software. Both cases used the same structured mesh of approximately 1.45 million points and elements, with

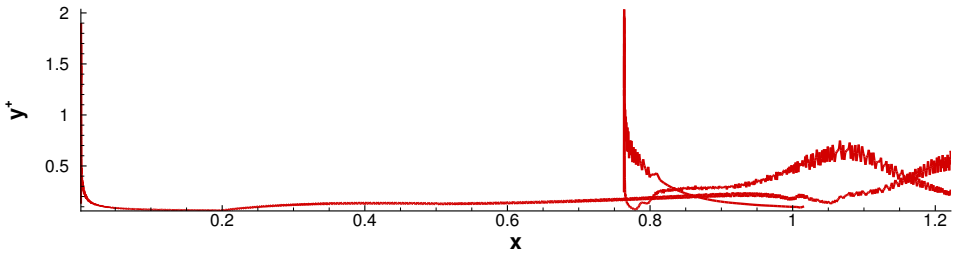


Figure 6.2: y^+ value at the solid walls

an HLLC solver and the SST turbulence model. The mesh spacing normal to the wall is 5×10^{-7} m with a stretching ratio of 1.05, the ratio that defines how quickly the mesh spacing grows going outwards from the wall. Surface values were indistinguishable using a mesh spacing of 5×10^{-6} m. VULCAN[121] is a CFD software developed in the Hypersonic Airbreathing Propulsion Branch at NASA Langley, and is particularly designed for hypersonic flow simulation. Based on these results, a wall spacing of 5×10^{-6} meters was used in later results. An unstructured mesh with similar boundary layer refinement has fewer nodes.

Check of Assumptions

This section will review the modeling assumptions and provide a rough check of their appropriateness for this case. This includes assumption of ideal gas in the inlet but not the combustor, equilibrium flow, etc. An ideal gas assumption is applied for the flow through the inlet. This assumption breaks down at around $800\,°K$. Figure 6.4 illustrates the points in the flow where the temperature exceeds $800\,°K$, where this assumption would lead to potential inaccuracies. In this figure, it is clear that the boundary layer exceeds this limit over large portions of the surface. Since the boundary layer affects the shock structures, and has a large impact on the heat flux into the surface, this would need to be addressed for better accuracy. A potential solution would be to introduce a wall function with real gas effects to model the boundary layer. Use of wall functions would also decrease the mesh size required. This is outside the scope of this work.

Another assumption applied is that gas dissociation is negligible. This is checked by plotting the points where temperature exceeds $2000\,°K$, a rule of thumb limit where Nitrogen is expected to dissociate. This is shown in Figure 6.5, where only a small area at the stagnation point of the cowl lip experiences temperatures above this limit. The assumption of no gas dissociation is therefore reasonable under these circumstances.

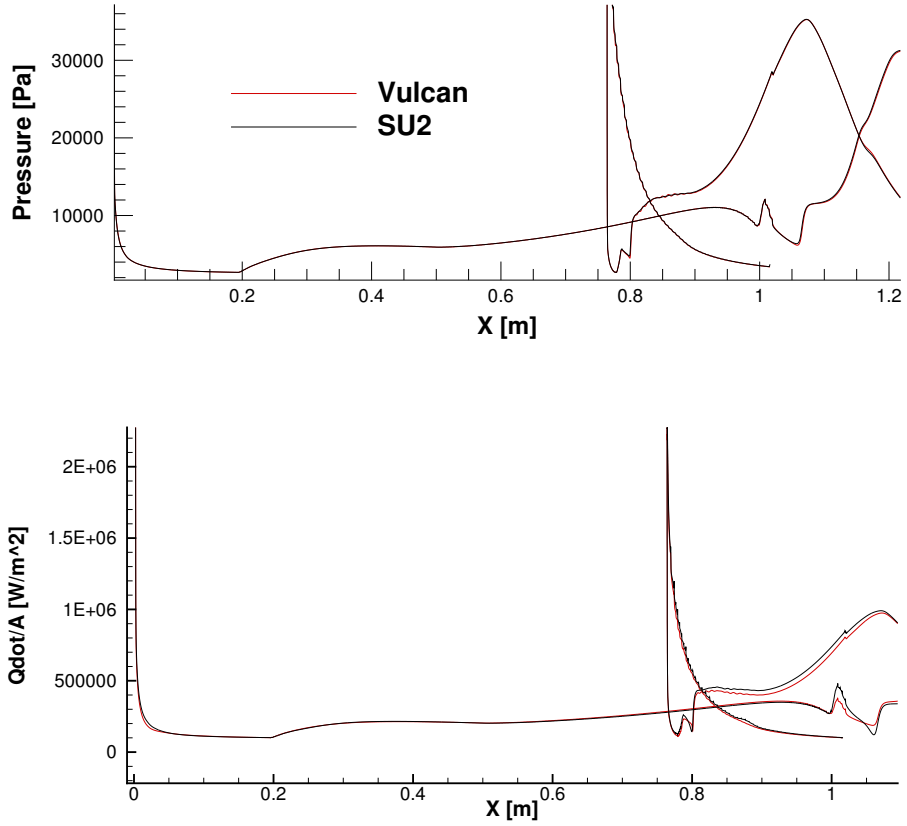


Figure 6.3: Comparison between VULCAN and SU2 results.

Combustor & Nozzle Geometry

The initial geometry is expected to have already high performance. In order to choose appropriate combustor and nozzle dimensions, a small optimization study with $J = -\frac{F_{un} - D_{est}}{\dot{m}}$ was performed using an initial flow solution. Four design parameters, the combustor length, nozzle length, combustor exit area, and nozzle exit area were varied to increase the specific installed thrust. The resulting geometric parameters are a combustor length of 0.9 m, a nozzle length of 2.51 m, combustor exit area $\mathcal{A}_4 = 0.073$ m, and nozzle exit area of $\mathcal{A}_{10} = 0.394$ m. These values result in area ratios of $\frac{\mathcal{A}_0}{\mathcal{A}_{10}} = 3.06$ and $\frac{\mathcal{A}_4}{\mathcal{A}_3} = 1.79$.

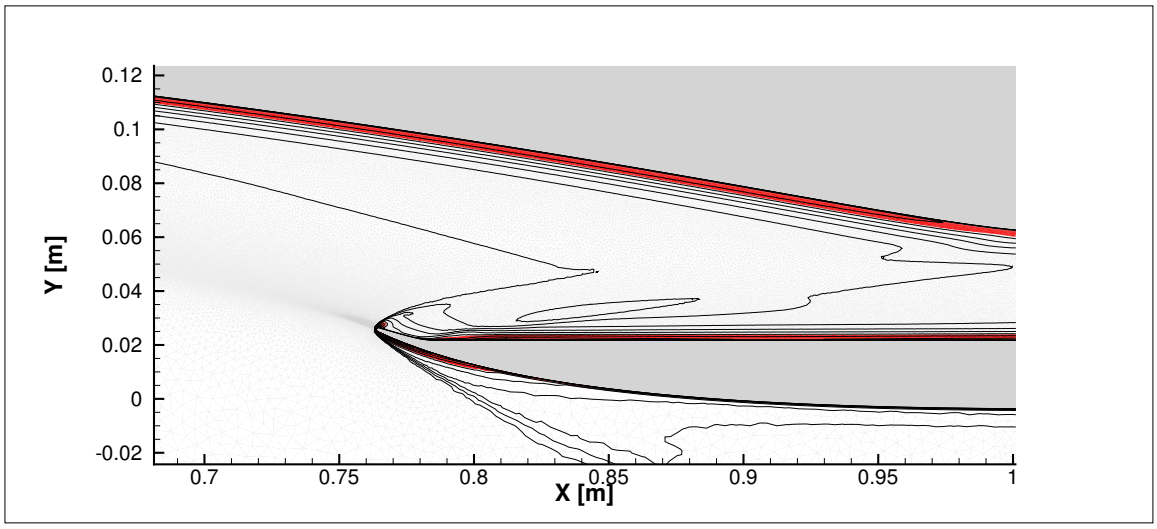


Figure 6.4: Contours of temperature with $T > 800\,°K$ highlighted.

6.2.2 Sensitivity Comparison

Prior to observing the optimization results, it is informative to compare the surface sensitivity between the different objectives. Figure 6.6 compares the surface sensitivity for each of the objectives. Our definition of surface sensitivity is the partial derivative of a given quantity of interest with respect to an infinitesimal change in the surface shape in the normal direction, at any point on the surface. Over much of the ramp, the sensitivities are similar, however differences appear near the nose of the compression ramp, the nose of the cowl, and in the isolator in the neighborhood of shock reflections. The total pressure and thrust objectives have similar trends throughout, with some differences in relative magnitude: thrust seems to have a greater sensitivity to the nose geometry than to the cowl geometry, while the total pressure has approximately the same sensitivity to the shape of these two locations. Although sensitivity signs match at the nose, when considering the penalty on total temperature the sign changes within the isolator and at the cowl nose, indicating a location where changes to the total temperature would overpower an increase to thrust. When the objective with or without a penalty disagree on the sign of the sensitivity this means that a design change that increases the thrust will bring the design closer to thermal choking, or to exceeding the structural limits of the combustor. Areas where the sensitivities share the same sign indicate areas where a design change can be made that increases the thrust without increasing the risk of thermal unstart. Modifying the weighting value placed on the penalty function may alter this relationship. The relative signs of the sensitivity in the isolator indicate that a complex relationship exists between the different objective functions at shock reflection points, which are aligned with the peaks in sensitivity in this area. Based on these sensitivities, we might expect that the total pressure and

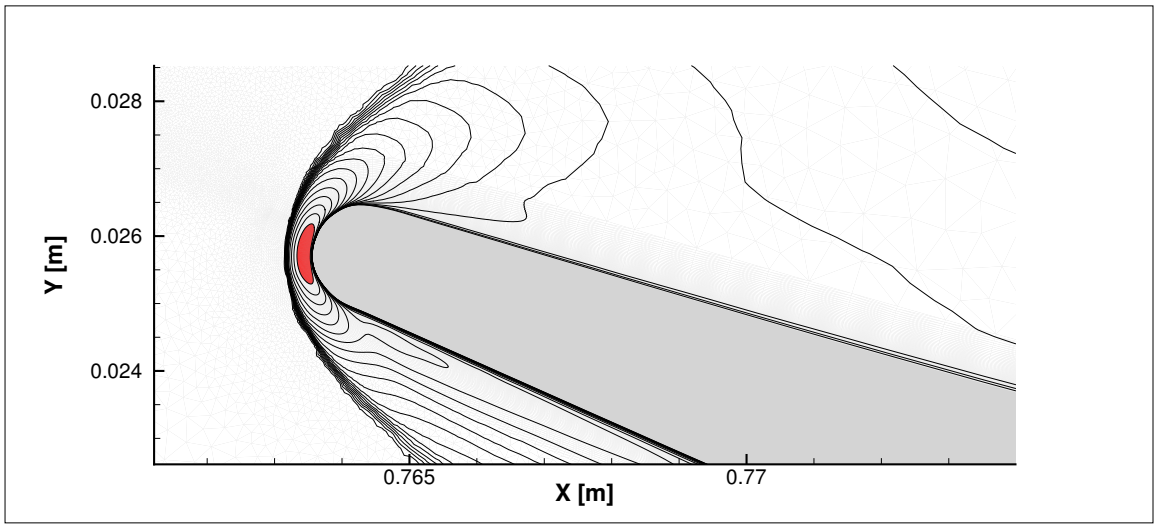


Figure 6.5: Contours of temperature with $T > 800°K$ highlighted.

installed thrust will be loosely correlated, and that there will be an inverse relationship with $P_{\max}$. This plot also shows that while an increase in τ_e should increase the thrust, for some geometry variations the thrust can be increased without increasing τ_e significantly.

When comparing the $-P_{tr}$ and $P_{\max}$-penalized sensitivity contours, it is intuitively expected that these quantities should have opposing signs, since an increase of total pressure should be associated with an increase of pressure in the combustor. This is the case over the majority of the geometry, with exceptions at the cowl and some areas on the interior lower wall downstream of the cowl lip, up until the shock reflection point occurring partway down the isolator (at $\approx x = 1.1$).

In a practical setting, this sensitivity information could be used both to make improvements to the design (as demonstrated in the optimization studies included in Section 6.2.3 - Section 6.3), and to augment designer intuition about which areas of the geometry are most critical and about what relationships exist between the different quantities of interest. In this example, an engineer might want to increase the thrust but knows that they are close to a limit τ_e—using this sensitivity information, they would first make modifications to the nose of the inlet knowing that modifications to this region are less likely to detrimentally affect τ_e. These sensitivities can also be used to choose an appropriate model—if there were no regions where the P_t and the specific installed thrust were perfectly aligned, a designer might choose not to include the more complicated model required to evaluate the thrust. In this case, where differences are seen between total pressure and specific installed thrust, it would be expected to see a difference in the optimization results, which will be seen in Section 6.2.3. An engineer attempting to explain discrepancies between CFD and wind tunnel results might first investigate the regions near the nose, the cowl lip, and the shock reflection points in the isolator since these high-sensitivity areas are locations where small errors in the physical

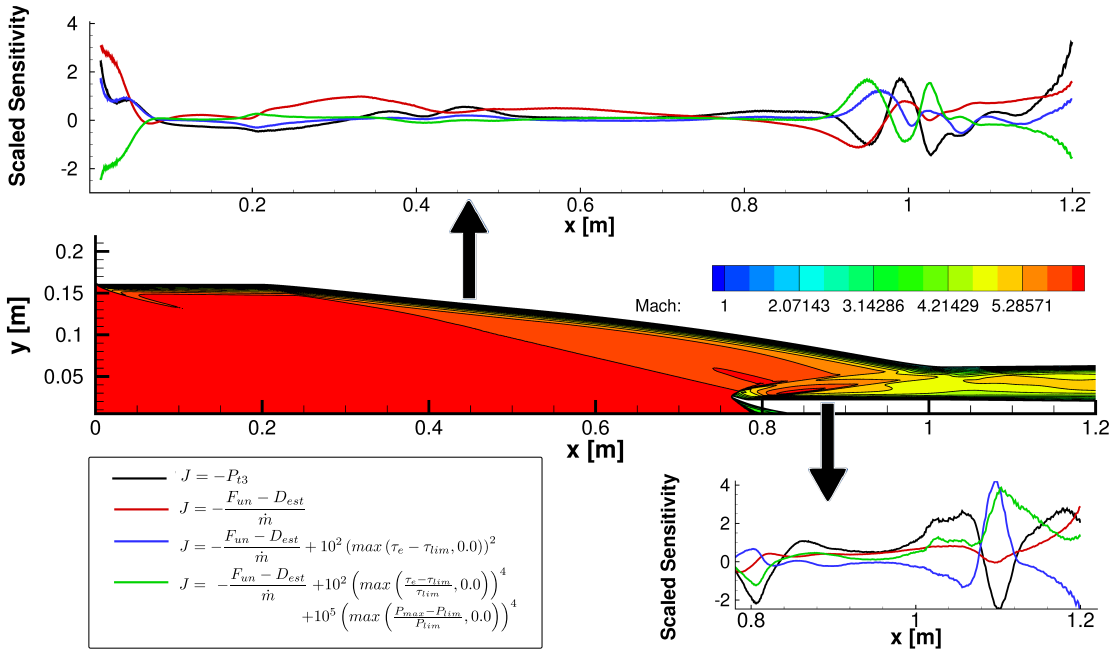


Figure 6.6: Surface sensitivity on the ramp and cowl surfaces. Sensitivities are scaled by the integrated norm of surface sensitivity to aid comparison, and smoothed by a moving average.

geometry would have a large effect on performance. During CFD simulations, high-sensitivity regions can also be used to inform mesh refinement, increasing the accuracy of the simulation while limiting the increase in computational cost.

6.2.3 Single-objective Design Results

In this section, the results of maximizing total pressure recovery P_{tr} will be compared to using the generalized outflow boundary conditions presented in Section 3.2.1 to maximize the specific installed thrust, which depends on an external model of the combustor and nozzle. A penalty on the total temperature ratio τ_e is then added, followed by a penalty on the maximum pressure in the combustor $P_{\max}$. These results were evaluated at Mach 7, dynamic pressure of 82.83 kPa.

Each of these functions were used with the pyOpt SNOPT optimization routine,[82,117,118] using the methodology discussed in this work with the continuous adjoint method to produce gradients. Relative changes in performance metrics are shown in Table 6.1. All optimization problems are

Table 6.1: Relative Performance of Optimization Results

Objective Function	$\frac{\mathcal{F}_{un}-D_{est}}{\dot{m}}$	P_{t3}	$\dot{m}$	τ_e	$P_{\max}$
$J = -P_{t3}$	-0.57%	+1.86%	+0.29%	-0.08%	+1.32%
$J = -\frac{F_{un}-D_{est}}{\dot{m}}$	+17.08%	+6.54%	+1.09%	+0.07%	+0.06%
$J = -\frac{F_{un}-D_{est}}{\dot{m}} +$ $1 \times 10^2(\max(\tau_e - \tau_{\lim}, 0))^2$	+24.47%	-2.54%	+2.22%	-0.20%	+1.72%
$J = -\frac{F_{un}-D_{est}}{\dot{m}} +$ $1 \times 10^2 \left(\max\left(\frac{\tau_e-\tau_{\lim}}{\tau_{\lim}}, 0\right)\right)^4 +$ $1 \times 10^5 \left(\max\left(\frac{P_{\max}-P_{\lim}}{P_{\lim}}, 0\right)\right)^4$	-10.00%	+0.45%	-0.92%	+0.10%	-0.94%

unconstrained with the exception of bounds on the design variables:

$$\begin{array}{ll} \text{minimize} & J(\vec{x}) \\ \text{with respect to} & \vec{x} \in \mathbb{R}^n \\ \text{subject to} & -0.02 \leq x_i \leq 0.02, \quad i = 1, ..., n \end{array} \quad . \tag{6.1}$$

In some cases a constraint is introduced as a penalty function, which is included in the objective function J, as listed in Table 6.1.

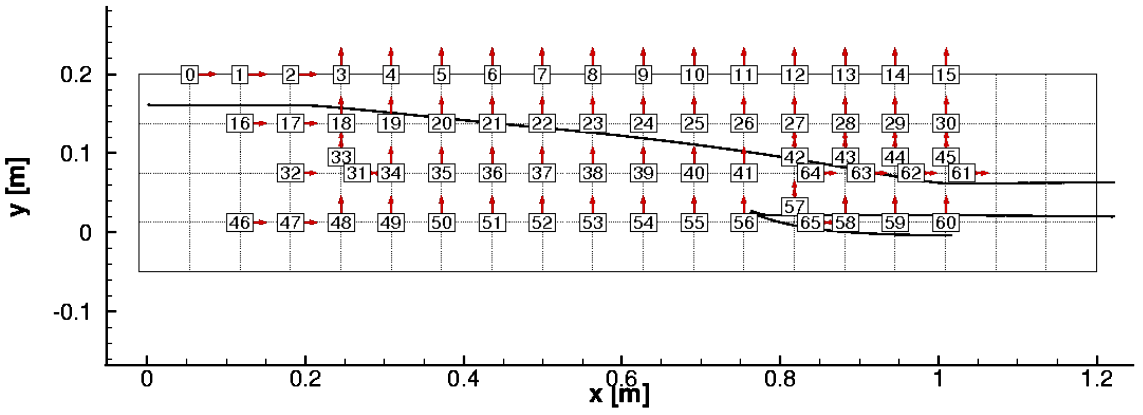
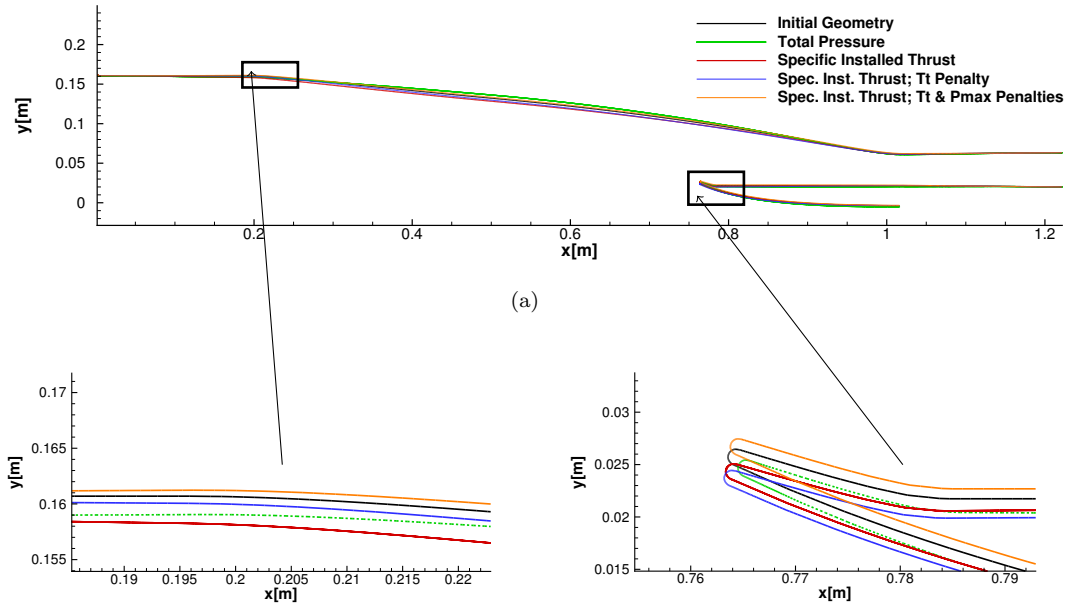


Figure 6.7: Design variable numbers for two-dimensional cases.

The design variables are Free-Form Deformation box points, illustrated in Figure 6.7. Comparing the performance metric results, it becomes clear that although total pressure increase may be correlated with increased thrust in some cases and in the surface sensitivity at the initial point, it is not a guarantee that an increase in P_{tr} will always lead to an increase in specific installed thrust. This is due to the fact that the thrust is related not just to the inlet efficiency (indicated by a high

P_{t3}/P_{t0}), but also to $\dot{m}$ and combustor efficiency. $P_{\max}$ always increased when not controlled by a penalty function, which would increase the required structural weight of the combustor. The most complex case, imposing penalties on both the total temperature ratio and maximum pressure, and choosing limits such that these constraints would be active at the initial point leads to a design that has lower thrust, but succeeds in lowering $P_{\max}$ and τ_e. The installed thrust is still positive in this case.

Figure 6.8 compares the geometry changes corresponding to the results in Table 6.1. We can see that there are subtle differences in the geometry depending on which objective is chosen. The maximum difference, taken as $max(\sqrt{\Delta x^2 + \Delta y^2})$ over the entire surface geometry, is $\mathcal{O}\left(1 \times 10^{-3}\right)$ m for each of the optimization cases, which is larger than manufacturing tolerances, although still presenting potential difficulties for manufacturing these engines as the resulting change in performance is $\mathcal{O}(10)\%$. As an example of manufacturing tolerances, some machining services[122] offer tolerances of ten thousandths of an inch, or 2×10^{-5} m, and tighter tolerances are possible.



(b) Detail of ramp deformation.

(c) Detail of cowl deformation.

Figure 6.8: Optimizer results comparing varying objective functions.

These results reflect the need for high-fidelity tools that can discern between these slightly different geometries as performance changes on the order of 10% as seen in Table 6.1 may not be shown by lower fidelity methods, especially due to the influence of viscous effects. A lower fidelity mesh, or the use of an inviscid model, would likely either not reveal the performance difference of such small changes, or predict them inaccurately. In order to further investigate the performance differences

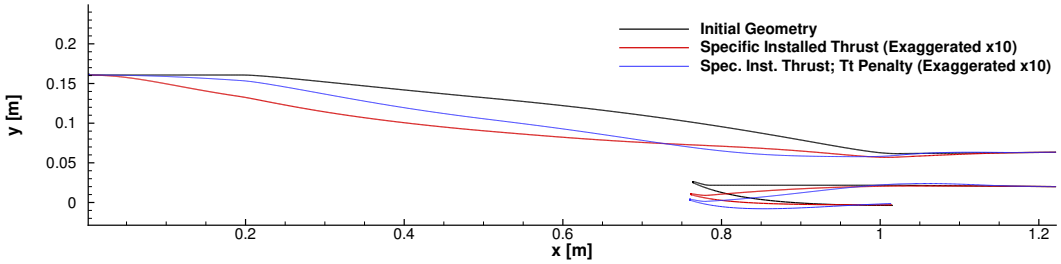


Figure 6.9: Exaggerated deformations associated with the specific installed thrust and τ_e penalty cases of Table 6.1.

Table 6.2: Relative Performance of Optimization Results - Additional Iterations

Objective Function	$\frac{\mathcal{F}_{un}-D_{est}}{\dot{m}}$	P_{t3}	$\dot{m}$	τ_e	$P_{\max}$
$J = -\frac{F_{un}-D_{est}}{\dot{m}}$	26.73%	-2.52%	2.15%	-0.18%	2.83%
$J = -\frac{F_{un}-D_{est}}{\dot{m}} + 1 \times 10^2(\max(\tau_e - \tau_{\lim}, 0))^2$	27.77%	-5.79%	2.64%	-0.30%	3.11%

of these geometries, an evaluation of an empirical correlation for unstart developed by Sullins[72] is included in Section 6.2.3.

Constraining the total temperature ratio (τ_e) with a penalty function and weighting factor of 10 as shown in Table 6.1 initially led to an increase in thrust greater than that when optimizing for the thrust alone, however on closer inspection this was due to the un-penalized case terminating prematurely and the penalty having only a small effect on the result. The associated geometry changes relative to the initial geometry are exaggerated in Figure 6.9, which shows that the resulting geometries from the 2^{nd} and 3^{rd} rows of Table 6.1 are similar to each other, with slight differences in the change to the ramp shape and cowl location. Additional design iterations for each of these two cases were run, starting from the final geometry associated with the results of Table 6.1, and using an updated version of the simulation code that is discussed in Appendix B.1. It can be seen in Table 6.2 and Figure 6.10 that these two cases approach each other after additional design iterations, which indicates that the presence of the penalty has not had a large effect on the outcome. A larger weight on the τ_e penalty would likely result in a lower thrust as well as lower τ_e, similar to what is seen when a heavily weighted penalty is applied to the value of P_{max} in the last row of Table 6.1.

Empirically-Based Unstart Prediction

It is informative to observe whether the resulting inlet designs are relatively more or less likely to experience unstart. Although detailed prediction of unstart is outside the scope of this work, we can use empirical correlations to observe the relative capability of each of the inlets designed in this

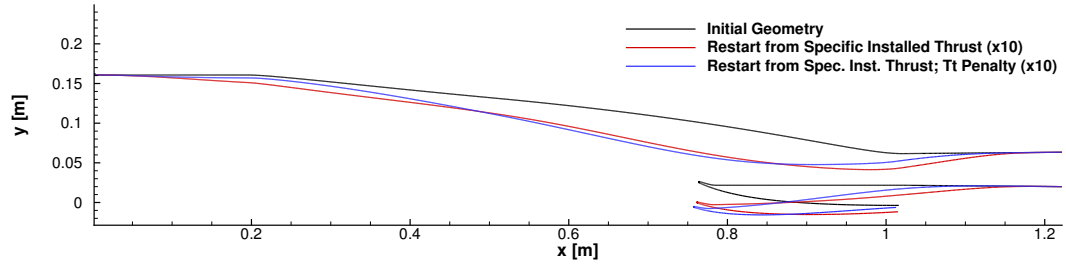


Figure 6.10: Exaggerated deformations associated with additional design iterations for the two cases shown inTable 6.2.

Table 6.3: Estimate of maximum backpressure that could be contained by the isolator.

Optimization Case	P_2 [Pa]	M_2	Re_θ [m]	P_3 estimate
Initial Geometry	1.8327×10^4	4.1672	4.3208×10^4	1.2660×10^5
P_{t3}	1.8210×10^4	4.2654	4.5976×10^4	1.2785×10^5
$\frac{D_{est}-F_{un}}{\dot{m}}$	1.8496×10^4	4.1197	4.2245×10^4	1.2716×10^5
$\frac{D_{est}-F_{un}}{\dot{m}}$ + penalty on τ_e	1.9236×10^4	4.0782	4.4838×10^4	1.3044×10^5
$\frac{D_{est}-F_{un}}{\dot{m}}$ + penalty on τ_e & $P_{\max}$	1.8004×10^4	4.2015	4.2588×10^4	1.2525×10^5

section. Sullins[72] provides a correlation for the pressure rise through rectangular ducts, which is discussed further in Section 2.1.2. Equation 2.4 predicts the pressure rise that would occur over a back-pressured isolator of a given length. The quantities in this correlation were not used directly in the objective functions, and the inlet was assumed to be un-back-pressured. The resulting P_3, shown in Table 6.3, represents the maximum back pressure that one would expect the isolator to contain.

To check the back pressure that might occur, the combustor-nozzle model from Section 4.2 was run in reverse, using the freestream pressure and the exit Mach number achieved from the forward flow on the intial geometry as the initial conditions. The pressure at station 3 predicted through this method on the initial design is 2.7885×10^4 Pa, an order of magnitude less than pressure rise that could be contained by the isolator. In other words, it is reasonable to assume that the bulk flow would be supersonic throughout.

A lower P_3 estimate corresponds to reduced ability to prevent unstart, and by this measure only the case with a heavy penalty on the maximum pressure in the combustor has resulted in a larger risk of unstart. The thermal penalty would be expected to reduce the risk of unstart, and while this represents a different mechanism for unstart, the associated case has the lowest τ_e and also resulted in the largest P_3 estimate. In other words, the τ_e-penalized case has the least chance of exceeding both the Rayleigh and Korkegi limits. Further observing the changes in τ_e in Table 6.1

and comparing to the resulting P_3 in Table 6.3 reveals that the largest increase in τ_e (greatest risk of thermal unstart) is also associated with the smallest P_3 estimate. Both of the remaining two cases raise the P_3 estimate relative to the initial point, to a greater extent for the case that reduces τ_e. Further study would be required to expand on a correlation between these two different mechanisms of unstart, which are described in more detail in Section 2.1.2. Explicitly including this estimate of the inlet's ability to reject pressure disturbances would be reasonable to include in future studies, although further development would be required to include values from the volume solution rather than the outflow.

6.2.4 Multi-objective Design Results: Low Fidelity Mesh

This optimization study will exercise the new implemented capability of addressing multiple objectives. This could be accomplished by evaluating the gradients separately, however due to the linear nature of the adjoint formulation, it is possible to find the gradient of a linear combination of objectives with a single adjoint evaluation, by superimposing the relevant boundary conditions. The flow conditions for this study are slightly different than the other cases. Following an investigation aimed at improving the performance of REST-class inlets through manual design changes by Gollan,[36] which utilize an array of engines installed on a planar vehicle, the flow conditions used are associated with the conditions downstream of a shock on a forebody angled at 6° with free-stream Mach of 8, and dynamic pressure of 48 kPa. The conditions downstream of this shock are used as the inflow conditions to the CFD volume.

The objective function is:

$$J = w_1 \int_S -k\nabla T \cdot \vec{n} ds + w_2 \frac{D_{est} - F_{un}}{\dot{m}}, \tag{6.2}$$

where k is the thermal conductivity at $\vec{n}$ is the normal vector on the surface. The weighting values, w_1 and w_2, are varied to control whether the design favors heat flux over thrust or vice versa. The heat flux is integrated over the solid walls included in the CFD volume, including the compression ramp, inner and outer surfaces of the cowl, and the interior surface of the isolator.

Figure 6.11 illustrates the results of varying the objective weights, showing the integrated heat flux and specific installed thrust values as well as the resulting geometries at a selection of points. The results are also summarized in Table 6.4. When $w_2 = 0$, a single-objective problem to reduce heat flux only, it is possible to significantly reduce the heat flux, however the installed thrust is significantly reduced as well. Comparing the results for $w_1 = 0$ and a small ratio of w_2/w_1, a relatively small change in the heat flux results in a relatively large change in installed thrust.

It is expected that as the surface heat flux decreases, the boundary layers should thicken and overall reduce the thrust produced by the engine. These results are consistent with that concept- it is possible to reduce the heat flux, and for some cases the thrust is still greater than drag, however

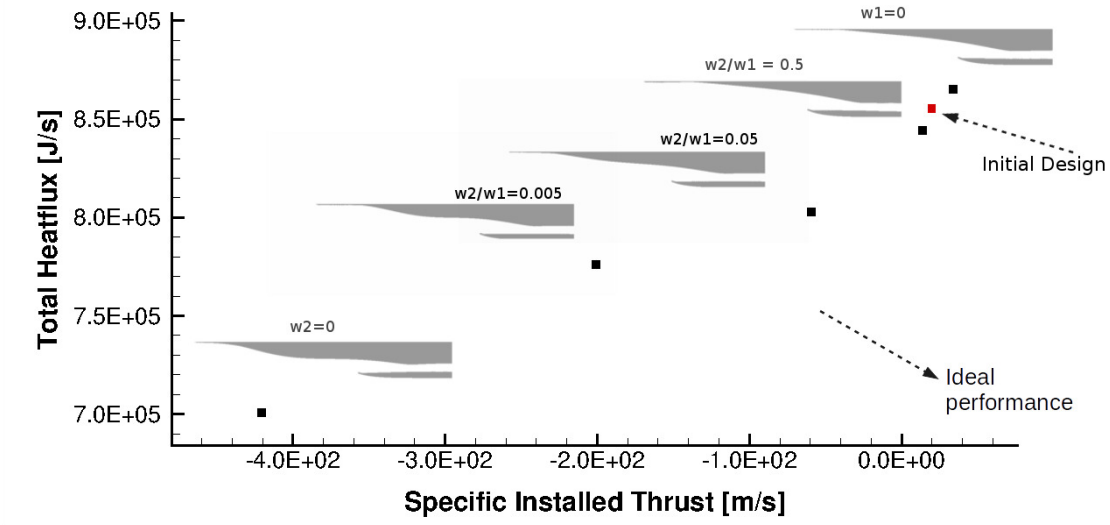


Figure 6.11: Results of varying w_1 and w_2 in $J = w_1 \int_S -k\nabla T \cdot \vec{n} ds + w_2 \frac{D_{est} - F_{un}}{\dot{m}}$.

Table 6.4: Summary of Multi-Objective Results With Lower-Fidelity Mesh

w_1/w_2	$\int_S -k\nabla T \cdot \vec{n} ds \ [J/s]$	$\frac{F_{un} - D_{est}}{\dot{m}} \ [m/s]$
0 / 1	865088.1104629348	33.9911764662
1 / 0.5	844393.606078	13.6214411258
1 / 0.05	802825.778907	-59.5060821488
1 / 0.005	776151.528706	-200.799249389
1 / 0	700704.1	-420.506072727

as the heat flux is reduced further the thrust produced by the engine is no longer able to overcome the drag. The engine performance also suffers from decreasing the capture area. Qualitatively, designs with less heat flux have a more blunted nose, forming an expansion region and causing the leading shock to miss the cowl lip by a wide margin. These results trace a Pareto frontier, illustrating trade-offs between two objectives. More detail on Pareto frontiers and their application in multi-disciplinary and multi-objective design application is provided by Huang.[123]

6.2.5 Multi-objective Design Results: High Fidelity Mesh

The low-fidelity case in Section 6.2.4 provides interesting trends in geometry changes, demonstrates the multi-objective capabilities and a trade-off between heat flux and thrust. The low-fidelity results demonstrate that this method of gradient evaluation is able to be used successfully with a multi-objective optimization problem, and that there is a trade-off between the heat flux and the thrust of

Table 6.5: Summary of Multi-Objective Results on Higher-Fidelity Mesh $J = w_1 \int_S -k\nabla T \cdot \vec{n} ds + w_2 \frac{D_{est} - F_{un}}{\dot{m}}$

w_1/w_2	$\int_S -k\nabla T \cdot \vec{n} ds \ [J/s]$	$\frac{F_{un} - D_{est}}{\dot{m}} \ [m/s]$
0.5/0.5	1046154.38	31.342070619
0.95/0.05	1010273.52	22.020443739
0.995/0.005	983619.63	27.5286437628

the vehicle. However, at the lower-fidelity level of simulation the heat flux is not captured accurately. The higher fidelity mesh will now be used to check that similar trends result at higher accuracy, although a smaller number of cases will be used. The same objective function and flow conditions are used. Figure 6.12 illustrates the results of varying the objective weights, showing the percent

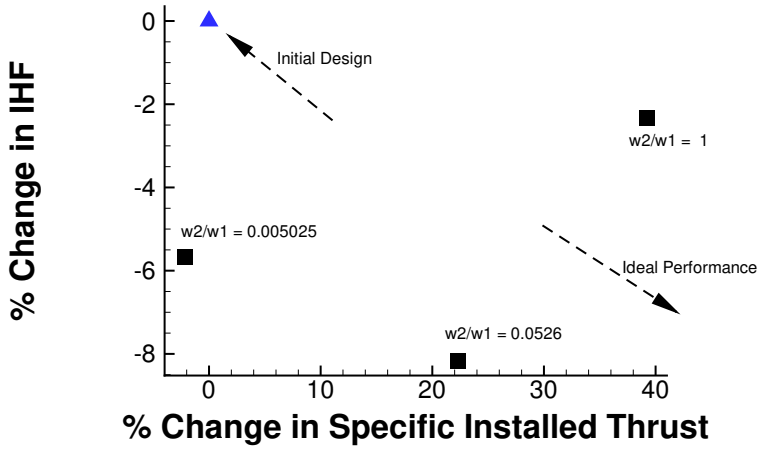


Figure 6.12: Results of varying w_1 and w_2 in $J = w_1 \int_S -k\nabla T \cdot \vec{n} ds + w_2 \frac{D_{est} - F_{un}}{\dot{m}}$.

change in integrated heat flux and specific installed thrust values. Resulting geometries at a selection of points are shown in Figure 6.13. The results are also summarized in Table 6.5. As mentioned in Section 6.2.4, it is expected that as the surface heat flux decreases the thrust should decrease as well. These results are generally consistent with that observation, showing a trade-off where more heavily weighting the thrust results in designs where the thrust is higher but the heat flux has also undesirably increased. It should also be noted that, possibly due to choosing an off-design flow condition, it is possible to simultaneously improve the heat flux and the thrust relative to the initial point. This was not shown by the lower fidelity results in Section 6.2.4, where the heat flux was under-predicted.

At higher fidelity, smaller deformations have resulted as compared to the lower fidelity results. This is partially due to grid deformation limitations; taking a smaller step size results in more easily

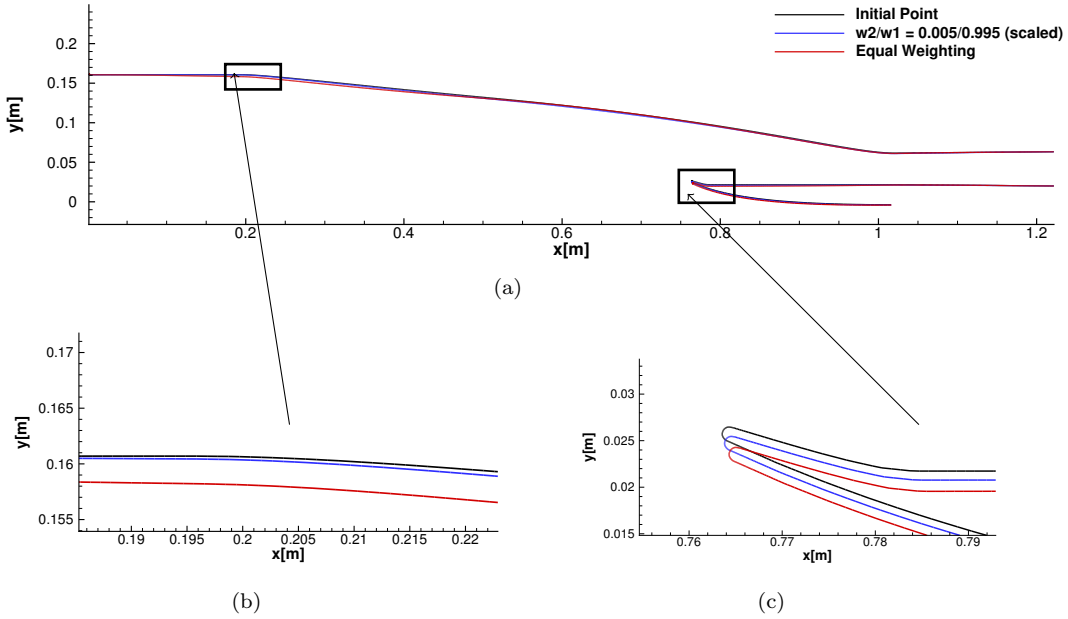


Figure 6.13: Geometry changes with different objective weights in Equation 6.2

deforming the larger mesh, and also results in the optimizer settling on a solution closer to the initial point. In addition, the finer mesh, with higher accuracy, more closely matches the high performance of the initial geometry. Despite the small deformations, the trend is similar—the ramp of the inlet deforms down making the ramp more convex as the weight on the heat flux is increased, and a trade-off between heat flux and thrust can be seen. The cowl is deflected downwards to increase $\dot{m}$ and bring the cowl lip closer to the new location of the leading shock.

6.2.6 Two-Dimensional Optimization With Heat Flux Penalty

Although modifying weights to produce the Pareto front shown in Section 6.2.5 illustrates interesting trends, a more practical case from a design perspective is to constrain some quantities while minimizing another. In the methodology implemented in this work, using a penalty on a functional rather than a weighted sum of functionals is in fact very similar. By using a penalty function, the constrained problem can proceed similarly to a weighted sum, with the computation of the gradients being done efficiently as described in Section 3.4.

The optimization problem used in this two-dimensional case, starts from the same geometry as used in previous cases. The model for the specific installed thrust, τ_e, and $P_{\max}$ is also the same, however the input values are now the mass flux averaged quantities rather than area averaged

Table 6.6: Penalty settings for two-dimensional optimization problem.

Quantity	Upper Limit	Weighting Value	Initial Point
IHF	6.16×10^5	1×10^{-4}	Active: $\approx 0.5\%$ above limit
$P_{\max}$	1.2×10^5	1×10^{-4}	Inactive
τ_e	2.055	1×10^4	Inactive

quantities. To state the optimizer goals:

$$\begin{aligned} \min_x J(\vec{x}) &= -\frac{\mathcal{F}_{un} - D_{est}}{\dot{m}} \\ \text{subject to: } & \vec{x}_l \leq \vec{x} \leq \vec{x}_u, \\ & IHF \leq IHF_{\lim}, \\ & P_{\max} \leq P_{\lim}, \\ & \tau_e \leq \tau_{\lim}. \end{aligned} \tag{6.3}$$

$IHF = \int k \nabla T ds$ is the integrated heat flux (per meter) over the wetted surfaces included in the CFD volume. The limits in the integrated heat flux, the maximum pressure in the combustor ($P_{\max}$), and the engine total temperature ratio (τ_e) are applied as quadratic penalty functions, resulting in a new objective function J' and NLP:

$$\begin{aligned} \min_x J'(\vec{x}) = &-\frac{\mathcal{F}_{un} - D_{est}}{\dot{m}} + 10^{-4} \left(\max(IHF - IHF_{\lim}, 0.0)\right)^2 \\ &+ 10^{-4} \left(\max(P_{\max} - P_{\lim}, 0.0)\right)^2 \\ &+ 10^{4} \left(\max(\tau_e - \tau_{\lim}, 0.0)\right)^2 \\ \text{subject to: } & \vec{x}_l \leq \vec{x} \leq \vec{x}_u. \end{aligned} \tag{6.4}$$

Constraint limits and weighting values are taken from the initial point design, with each of the limits chosen to be close to the initial value and the weights chosen such that a 10% deviation from the limit would result in a value of the penalty function of $\approx \mathcal{O}(10)$. This results in the values shown in Table 6.6.

The design variables in this case are FFD box control points, as in Section 6.2.5 and Section 6.2.3. However, the box has been modified with fewer degrees of freedom and larger coverage of the geometry. These design variables are shown in Figure 6.14.

The SNOPT[82] algorithm using PyOpt[117] was used. After running the optimization with an initial, small, major step size limit of 1×10^{-4} for 7 major steps, moderate improvement was seen. As the optimization process was proceeding well at this point, the case was restarted from the final design for an additional 5 major iterations, with a larger major step size limit of 1×10^{-3}. The results shown concatenate the results, and the values in Table 6.7 show the final results. Figure 6.15 illustrates the

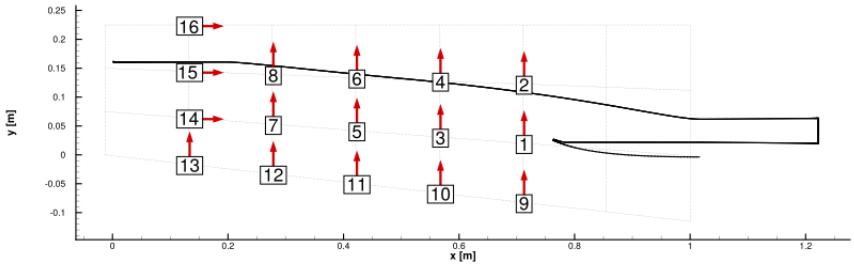


Figure 6.14: Design Variables for 2D Optimization with Heat Flux Constraint.

Table 6.7: Change in Performance Metrics for 2D Optimization with Heat Flux Penalty

J'	$\frac{\mathcal{F}_{un}-D_{est}}{\dot{m}}$	P_{t3}	$\dot{m}$	$IHF - IHF_{\text{lim}}$	τ_e	P_{max}
-148.94%	$+0.178\%$	-0.718%	$+0.534\%$	-85.68%	-0.054%	$+0.186\%$

percent change in the augmented objective function J'. Figure 6.16 illustrates the percent changes in the components of this function, the specific installed thrust (J) and the difference of the integrated heat flux and its limit ($IHF - IHF_{\text{lim}}$). The integrated heat flux in the final design is within 0.07% of the desired limit, while the specific installed thrust has slightly improved. Figure 6.17 illustrates the change in geometry associated with this optimization case. As the geometry changes are small, Figure 6.18 shows the same geometry change, with the difference exaggerated by a factor of 25. In this exaggerated plot, it can be seen more easily that, similar to the low-fidelity multi-objective case in Section 6.2.5, the ramp bulges outwards in order to reduce the integrated heat flux.

6.3 Three-Dimensional Optimization With Heat Flux Penalty

This section will discuss an optimization problem similar to Section 6.2.6, with a three-dimensional inlet. The optimization problem used in the three-dimensional case is:

$$\begin{aligned} \min_x J(\vec{x}) &= -\frac{\mathcal{F}_{un} - D_{est}}{\dot{m}} \\ \text{subject to: } & \vec{x}_l \leq \vec{x} \leq \vec{x}_u, \\ & IHF \leq IHF_{\text{lim}}, \\ & P_{\text{max}} \leq P_{\text{lim}}, \\ & \tau_e \leq \tau_{\text{lim}}. \end{aligned} \tag{6.5}$$

$IHF = \int k \nabla T ds$ is the integrated heat flux over the wetted surfaces included in the CFD volume. The limits in the integrated heat flux, the maximum pressure in the combustor (P_{max}), and the engine total temperature ratio (τ_e) are applied as quadratic penalty functions, resulting in a new

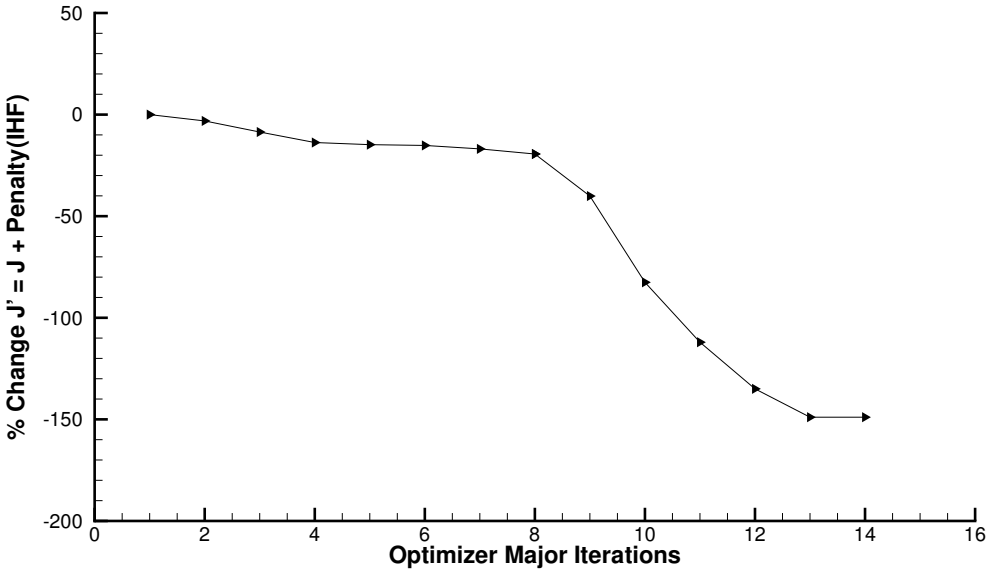


Figure 6.15: Optimization of 2D Geometry For Equation 6.4

Table 6.8: Penalty settings for three-dimensional optimization problem.

Quantity	Upper Limit	Weighting Value	Initial Point
IHF	1.35×10^5	1×10^{-5}	Active: $\approx 0.5\%$ above limit
$P_{\max}$	1.45×10^5	1×10^{-5}	Inactive
τ_e	2.055	1×10^5	Inactive

.

objective function:

$$\begin{aligned} \min_x J'(\vec{x}) = &-\frac{\mathcal{F}_{un} - D_{est}}{\dot{m}} + 10^{-5} \left(\max(IHF - IHF_{\lim}, 0.0)\right)^2 \\ &+ 10^{-5} \left(\max(P_{\max} - P_{\lim}, 0.0)\right)^2 \\ &+ 10^{5} \left(\max(\tau_e - \tau_{\lim}, 0.0)\right)^2 \\ \text{subject to: } &\vec{x}_l \leq \vec{x} \leq \vec{x}_u. \end{aligned} \tag{6.6}$$

Constraint limits and weighting values are taken from the initial point design, with each of the limits chosen to be close to the initial value and the weights chosen such that a 10% deviation from the limit would result in a value of the penalty function of $\approx \mathcal{O}(1000)$. This results in the values shown in Table 6.8.

The design variables in the three-dimensional case are once again defined by Free-Form Deformation box points, which are discussed further in Section 4.3. The control points used in this case

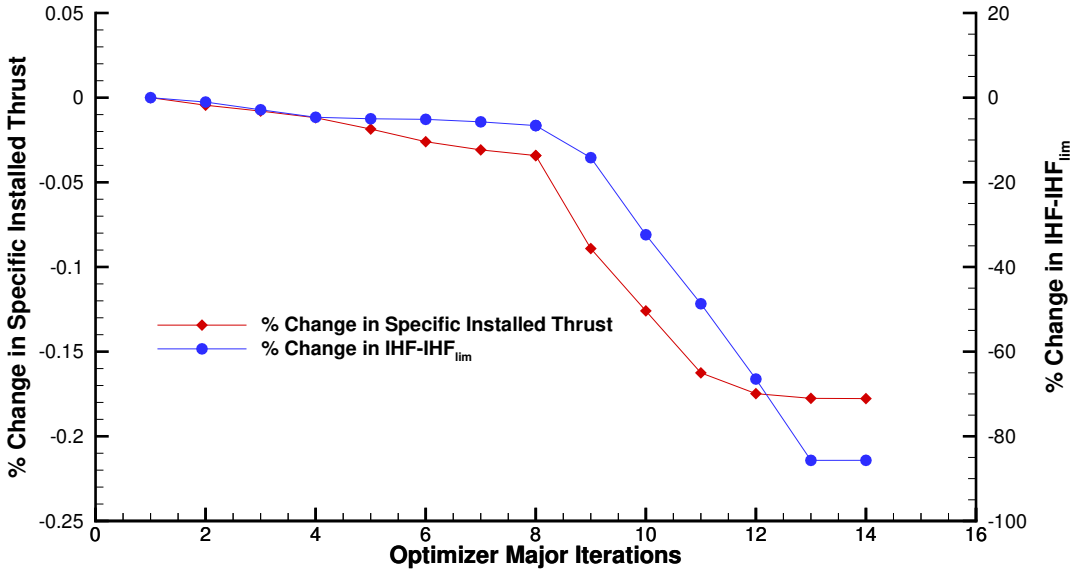
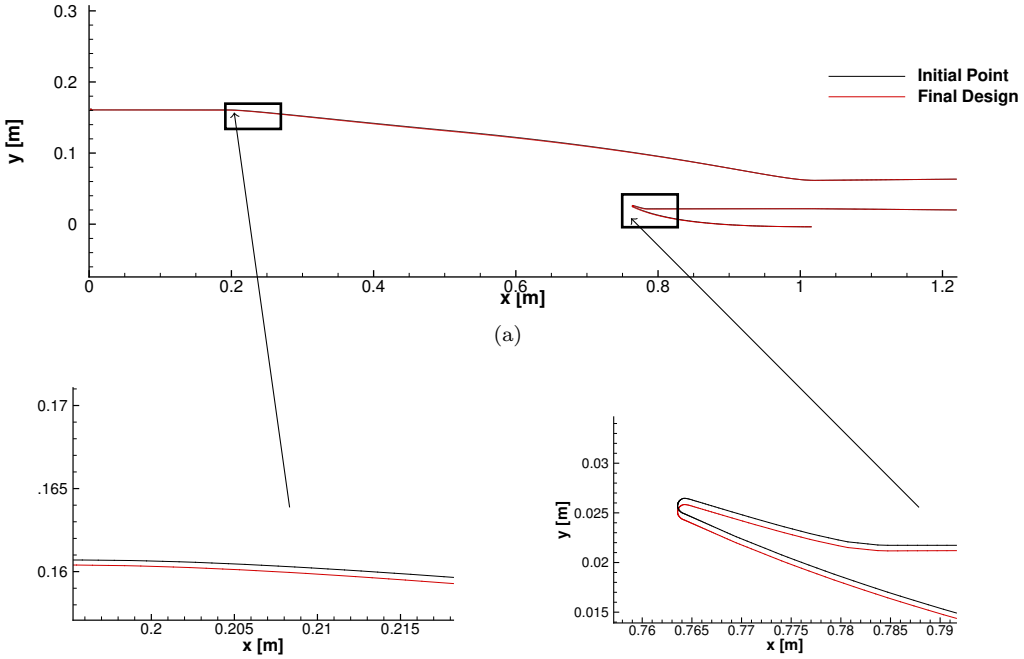


Figure 6.16: Optimization of 2D Geometry For Equation 6.4

are shown in Figure 6.19.

Similar to the two-dimensional case, an initial step is performed to optimize the geometry parameters in the combustor and nozzle, which are described simply by lengths and entrance/exit areas. The areas follow a linear distribution between the end points. The resulting parameters are $0.476\,m$ long combustor with exit area $0.0204\,m^2$, and a $3.876\,m$ long nozzle with $0.1128\,m^2$ exit area. This is an extremely long nozzle; an internal expansion nozzle has been used in this model to be consistent with the one-dimensional channel flow assumptions, however better performance may be possible with an external expansion ramp.

The model of the combustor-nozzle component was updated to correct for an over-expanded nozzle by calculating the properties after an adiabatic compression process; this was necessary as otherwise the optimization of the nozzle and combustor geometry in an initial step resulted in an over-expanded nozzle that would be less efficient in reality. The baseline mass of the vehicle and the material weight used in estimating the added mass from the engine shell (used in estimating drag by accounting for lift-induced drag) were reduced due to the initial design failing to produce thrust greater than drag, even after a separate optimization of the combustor and nozzle dimensions. The updated baseline mass is 500 kg, and the material weight is 144 $\frac{kg}{m^3}$, matching the density of LI-900 TPS tiles, which were the lower-density tiles used over a large portion of the Space Shuttle surface area.[124] This is an optimistic estimate of the weight, and although this was necessary to arrive at an initial point with positive thrust for the initial inlet geometry, lower-drag and higher-thrust designs



(a)

(b) Detail of ramp deformation.

(c) Detail of cowl deformation.

Figure 6.17: Geometry changes associated with results of Section 6.2.6.

may be possible, for example through using a different area distribution or a more sophisticated outer mold line, or an expansion ramp that would allow a shorter length. Such improvements are outside the scope of this work, however it should be noted that these, and other improvements to the external model could be added without changes to the adjoint formulation. Since during the main shape optimization process only the inlet shape is changed, the effects on the optimization are limited to the effect of a changing capture area, which changes the external drag estimation slightly.

Applying the selected constraints via a quadratic penalty functions allows the use of the multi-objective adjoint method with the weighting value w_i replaced by the product of the partial derivative of the constraint and the weighting value, as described in Section 3.4. The major iteration history of this optimization, using the SNOPT[82] algorithm and PyOpt,[117] is shown in Figure 6.20. This plot shows the percent change in selected quantities relative to the initial point value, and shows that the optimizer has succeeded in reducing the sum of the installed thrust and the penalty function on the integrated heat flux. This quantity has been reduced by 17.9%. The installed thrust function has improved by 4.10%, which by itself is a significant change in this quantity of interest. The integrated heat flux has approached the constraint value by 33.9% relative to the initial different between the heat flux and its constraint. Investigating the quantities not included in the optimization

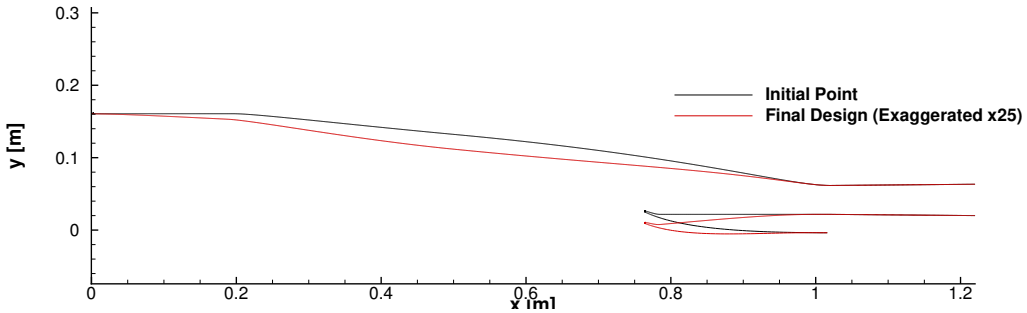


Figure 6.18: Exaggerated deformation associated with 2D optimization with a penalty on heat flux.

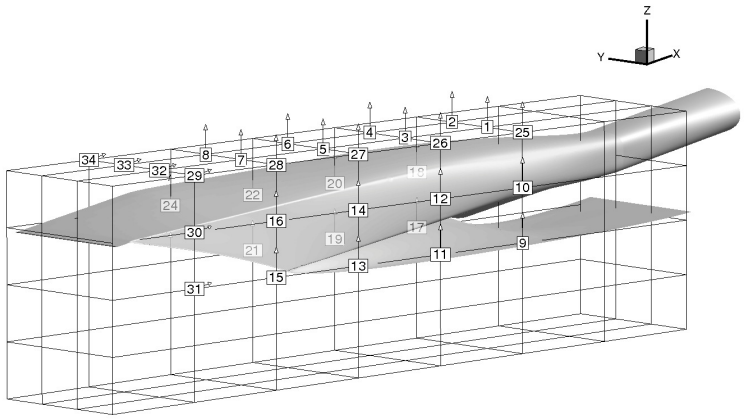


Figure 6.19: Design Variables for 3D Optimization

process is also interesting. The total pressure, a quantity that is traditionally maximized to increase inlet performance, has decreased by a small percentage. Although this metric of inlet performance decreases, the effect is outweighed by the increased thrust achieved. The uninstalled thrust, F_{un} increased slightly, by 0.1%, and similarly small improvements in F_{un} were seen in the two-dimensional optimization problems with similar objective functions. The increase in uninstalled thrust is possible in part due to the small size of the thrust margin $F_{un} - D_{est}$ relative to the magnitude of the uninstalled thrust.

The black line in Figure 6.20 indicates that the optimizer has not yet reached a limit—the value of the augmented objective is still decreasing. The limiting factor in this optimization problem was

Table 6.9: Change in Performance Metrics During 3D Optimization

J'	$\frac{\mathcal{F}_{un}-D_{est}}{\dot{m}}$	P_{t3}	$\dot{m}$	$IHF - IHF_{\text{lim}}$	τ_e	$P_{\max}$
-17.9%	$+4.10\%$	-0.194%	$+0.206\%$	-33.9%	$-2.1 \times 10^{-3}\%$	$+0.108\%$

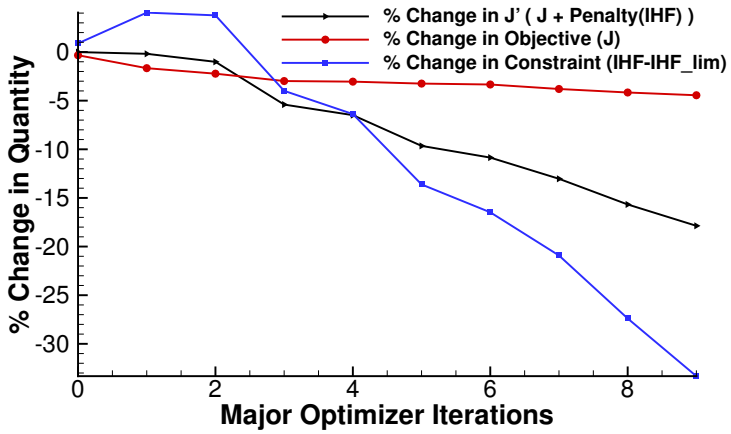


Figure 6.20: Optimization of 3D Geometry For Equation 6.6

the grid deformation, and it is likely that further improvement is possible. Issues with grid deformation resulted in a need to take relatively small steps and that resulted in a failed simulation that terminated the optimization process. Some literature suggests alternatives to the spring-analogy method used for grid deformation, for example radial basis functions.[125] Due to the need for boundary layer refinement, an alternative would be to use wall functions to model the boundary layer behavior close to the wall, which lessens the requirement for $y^+ \approx 1$, and would likely result in a more easily-deformed mesh. Improvements to mesh deformation are outside the scope of this work, and despite these issues the optmization process has still shown significant improvement in the objective and in the constraint value. In addition, these results identify design changes that could be considered via other methods of scramjet design, such as parametric design techniques that utilize automated meshing.[36]

Figure 6.21 illustrates the deformation of the geometry. Slices in the x-direction with exaggerated deformations are included to aid visualization of the geometry changes. A contour plot of the deformations shows the largest deformations are near the cowl lip and in the corners of the cross-section partway down the external compression ramp, where the geometry has a rectangular cross-section with rounded corners. The change the the cowl lip position should have a strong effect on the mass flow rate, and the changes to the corners of the cross section shape should influence the cross-flow vortices that develop in these corners. The geometry change along the centerline of the three-dimensional geometry is qualitatively similar to the geometry change seen for the similar, two-dimensional, case discussed in Section 6.2.6.

The difference between the initial and optimized geometry is on the order of 0.05 mm at the cowl lip and corners of the inlet ramp. These geometry changes appear to introduce a rotation to the cross-sectional geometry, suggesting that parametric studies similar to the studies that produced the initial

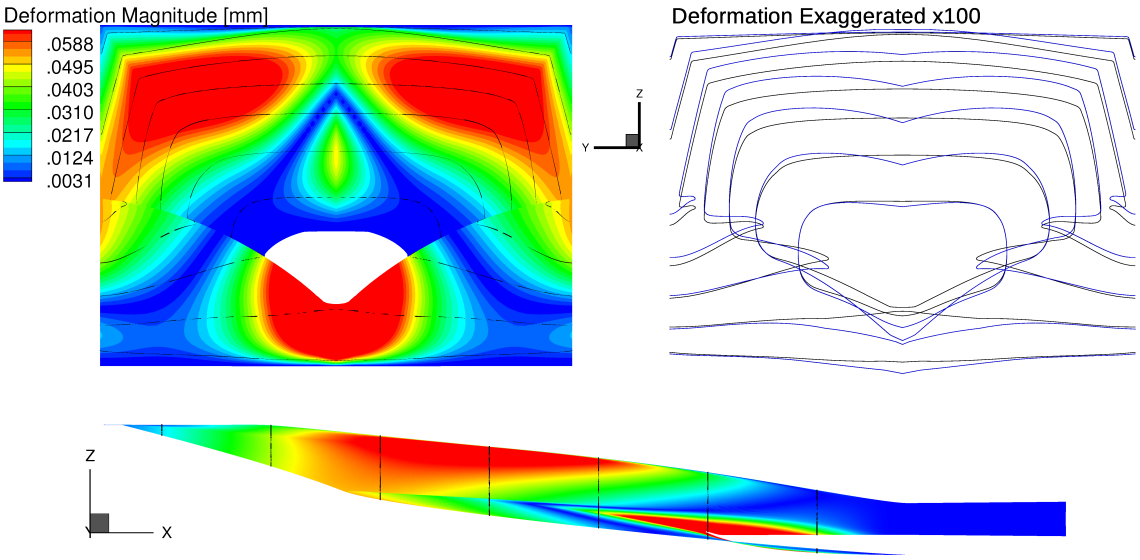


Figure 6.21: Visualization of 3D Optimization Result. Cross-section profile deformation (shown in blue) is exaggerated by a factor of 100 for ease of visualization.

geometry might include rotation of the cross section as a new parameter. The small magnitude of the geometry changes are nevertheless larger than aerospace manufacturing tolerances, although a tight tolerance does increase manufacturing costs and leads to greater concern for in-flight deformation. The geometry changes at the corners of the inlet ramp may be related to the thick boundary layers that form in these corners.

The three-dimensional case shown in this section has successfully improved performance of a complex inlet using RANS simulation of the flow through the inlet, and a quasi one-dimensional channel flow model of the combustor and nozzle. Using the adjoint of a generalized outflow-based functional developed in Section 3.2 in combination with the multi-objective adjoint developed in Section 3.1.6, the gradient evaluations required only a single adjoint evaluation, with a computational cost approximately equal to a single CFD evaluation, to address the penalized objective function shown in Equation 6.6.

Chapter 7

Conclusions & Closing Remarks

7.1 Summary & Contributions

The continuous adjoint method for a generalized outflow-based functional as well as for combinations of multiple functionals was developed in this dissertation. Generalized functionals on solid surfaces were also derived, but not implemented. These developments allow the use of the continuous adjoint method with functionals defined in arbitrary external models, requiring only that the external model be able to provide the relevant partial derivative values, and allow the use of the continuous adjoint method with a wide range of combinations of already-implemented functionals including penalty functions of those functionals. In other words, it is now possible to find the surface sensitivity with respect to a much larger range and complexity of functions, including but not limited to functions defined in a multi-fidelity flowpath model.

These methods were used in optimization problems of a particular hypersonic inlet geometry. The results of evaluating the surface sensitivity for several functionals, and the results of the optimization problems, demonstrated relationships between different performance metrics. A three-dimensional optimization problem involving complex geometry, use of the generalized outflow-based functional, and multiple functionals with a penalty function demonstrated the capabilities of this methodology to address complex design problems as well as the sensitivity and potential design improvements of this particular inlet.

Prior to this work, the only way to find the adjoint solution of a full flowpath was to automatically differentiate a multi-fidelity model, or simulate the entire flowpath with a single model using the highest fidelity desired. The first of these options allows the use of cheaper low-fidelity tools, but requires more memory in the adjoint formulation and does not allow modifications to the component models, or the use of models that are not compatible with automatic differentiation, for example, legacy codes or third party software where the source code is not available. The second option of simulating the full flowpath at high fidelity carries a much higher computational cost, to the

point where it may not be feasible to run the simulation, much less an iterative design process. By contrast, the methods developed in this work allow finding the sensitivities of the outputs of a multi-fidelity model with the only limitations on the external model being that it must be able to provide the relevant partial derivative terms—something that can be accomplished easily through perturbing averaged flow quantity inputs to the external model, regardless of whether the source code is available for that model.

7.2 Summary of Results

Section 6.2.1 examined assumptions employed in this work, by plotting the areas of the fluid flow around the inlet geometry that exceeded temperature limits. Neglecting gas dissociation effects is reasonable, while inclusion of real gas effects may be considered in improving the accuracy of the flow within the boundary layers. However, since a turbulence model was used in this work, the use of LES or DES would be likely to have a larger effect on the accurate prediction of the boundary layer as compared to real gas effects. Section 6.2.2 presented a comparison between surface sensitivity results using the generalized adjoint functional for several different quantities of interest. High sensitivity points occurred at the nose, cowl lip, and at shock reflection points inside the isolator. Contrasting the sensitivities of different quantities showed that while quantities that are expected to act in opposition to one another (i.e., specific installed thrust and τ_e) do have opposing sensitivities in some areas of the geometry, the effect is not uniform over the entire geometry, and in fact in some areas it may be possible to improve both quantities simultaneously. This was confirmed by optimization results examined in Section 6.2.3, where both τ_e and $\frac{\mathcal{F}_{un} - D_{est}}{\dot{m}}$ were improved simultaneously. The results in Section 6.2.3 also showed that an increase in P_{t3} does not always imply an increase in thrust and overall cycle efficiency. Section 6.2.3 demonstrated the use of the generalized outflow functional implementation to address optimization problems constrained through a penalty function, and included an examination of unstart capability through an empirical function used with post-processing of the flow solution on the resulting geometries.

A multi-objective design problem was addressed in Section 6.2.4 and Section 6.2.5, combining the generalized outflow functional with integrated heat flux over the solid walls of the inlet geometry. The implementation of a multi-objective adjoint described in Section 3.4 was utilized in these results. A trade-off between surface heat flux and thrust can be seen, and in some cases it was possible to improve both the heat flux and thrust relative to the initial point. In order to reduce surface heat flux, the ramp geometry was bulged outwards and the cowl deflected downwards to increase $\dot{m}$ and bring the cowl lip closer to the new location of the leading shock.

Combining the implementation of the multi-objective adjoint with the partial derivative of a penalty function facilitated further optimization problems to increase specific installed thrust and

penalize integrated heat flux. A two-dimensional case in Section 6.2.6 met the constraint on the integrated heat flux within 0.07%, and improved the specific installed thrust by 0.178%. Repeating this same optimization problem with updated constraint values for a more complex and computationally expensive three-dimensional geometry was able to increase the thrust by 4.1% while reducing surface heat flux, with small geometry changes on the order of 0.05 mm. These geometry changes appear to introduce a rotation to the cross-sectional geometry, suggesting that parametric studies similar to the studies that produced the initial geometry might include rotation of the cross section as a new parameter.

7.3 Future Work

The developments and simulations included in this work reveal some topics that deserve further investigation. In deriving the equations for generalized functionals, it can be seen in Section 3.2.1 that for surface-based functionals, only certain functionals allow cancellation of the surface curvature term H_m (See Equation 3.99). Calculating this term introduces algorithmic difficulties and computational cost, and so those functionals that require it are generally avoided for the continuous adjoint. Efficient computation of the surface curvature term would be a useful contribution, which may facilitate a wider range of surface-based functionals. In addition, implementation of generalized surface functionals may be useful to some applications.

An assumption made in this work was that real gas effects could be neglected on the inlet, and that RANS would be sufficient to model turbulence. LES, although impractical for optimization at the current time, would likely provide more accurate results where the necessary computational resources and time are available, without requiring modifications to the adjoint problem boundary conditions developed in this work. Section 6.2.1 showed that temperatures rise high enough in the boundary layer that inclusion of real gas effects could be considered, although this effect is expected to be less significant than the effect of using a turbulence model. In addition, small grid cell sizes introduced difficulties in mesh deformation. These issues may be addressed by further developments in wall functions that use a model of boundary layer behavior near the wall to reduce the number of boundary cells needed. Inclusion of real gas effects through the entire volume and the associated computational cost, as developed by Vitale,[29] is not necessary as the bulk of the flow is below the temperature limit. Wall functions considering some hypersonic effects such as developed by Lofthouse,[126] on the other hand, do not include real gas effects. Some combination of the two methodologies may be useful for situations such as the hypersonic inlet, where real gas effects are only needed in the boundary layer region, and where reduced mesh refinement would lead to benefits in mesh deformation.

Further work within the generalized outflow-based functional formulation introduced in this dissertation should include the derivation and implementation of the Jacobians necessary to translate

between the partial derivatives required by the adjoint formulation and functions defined in terms of measures of distortion. Measures of flow distortion are of great interest to propulsion design, and finding the adjoint formulation in terms of such quantities would require only the derivation and implementation of the necessary Jacobian. The concept of passing partial derivative information between different physical models in order to compute high-fidelity sensitivities has many applications. For example, it may be possible to implement the generalized surface-based functionals in a way that would allow design of thermal protection systems (TPS), where the external gradient information could come from a model of heat flux through the TPS material. These methods are also applicable to fluid-structure interactions, as well as other multidisciplinary simulations that require information transfer between models. The superposition of multiple objectives is likely applicable to a wide range of models and design problems.

7.4 Concluding Thoughts

The adjoint formulation for generalized functionals has many applications, from easing the process of deriving and implementing novel functionals, to allowing the evaluation of sensitivities for external functionals. Previously, some quantities of interest to hypersonic propulsion could not be addressed directly by the adjoint of the CFD simulation of an inlet geometry. This meant that optimization studies on inlet geometries were limited to addressing functions merely correlated with increased propulsion performance, such as total pressure ratio. By contrast, the methodology introduced in this work allows the direct optimization for outflow-based quantities of interest including installed thrust and the maximum pressure in the combustor. The multi-objective formulation additionally allows optimization for more complex problems that incorporate multiple quantities of interest and penalty functions to address constraints of the problem.

Much contemporary work in adjoints has focused on the discrete adjoint, and specifically the use of automatic differentiation has become popular in recent years as this capability became more generally available. Based on experience in using both discrete and continuous adjoint methods and on working to develop and implement a generalized outflow functional, the author believes the best option to be a hybrid of approaches. Analytically derived expressions from the continuous adjoint require less memory and provide information about the relationships between the adjoint variable values with the flow variable values prior to computations. The discrete adjoint can automatically account for numerical methods and discretization effects that otherwise become unwieldy in the continuous form. Moving forward from the developments in this work, it may be possible to apply similar techniques in introducing externally-provided partial derivative terms into the discrete adjoint.

Scramjet design has progressed a long way over the past several decades, and yet there is still further to go before these engines will be in production for airbreathing access-to-space and hypersonic

cruise applications. As seen in the results of this work, even advanced designs like the REST-class inlet are sensitive to small changes in geometry, especially when considering the difference of thrust and drag rather than uninstalled thrust on its own. Achieving good performance of these engines in flow fields including shock-boundary layer interactions requires advancements in the techniques of computing sensitivities, especially when "good performance" includes multiple physical models and many design constraints. This dissertation has made progress towards computing the sensitivities for relevant quantities, while further developments in simulation techniques, mesh deformation, uncertainty quantification, and addressing measures of flow distortion will be useful to continued advancements in scramjet design.

Appendix A

Shorthand Terms, Jacobians, & Expansions

A.1 Abbreviations and Shorthand Terms

This section summarizes a number of shorthand terms, abbreviations, and Jacobians for easy reference. The following shorthand terms are defined to abbreviate fluid governing equations:

$$\begin{aligned} a_0 &= (\gamma - 1) \\ \phi &= (\gamma - 1)\frac{\vec{v}^2}{2} \\ H &= E + \frac{P}{\rho} = \left(\frac{\vec{v}^2}{2} + \frac{c^2}{\gamma - 1}\right) \\ E &= \frac{\vec{v}^2}{2} + \frac{c^2}{\gamma(\gamma - 1)}. \end{aligned} \tag{A.1}$$

In these equations γ is the ratio of specific heats, H is the total enthalpy per unit mass, P is the fluid static pressure, ρ is the fluid density, c is the local speed of sound, E is the total energy per unit mass, and $\vec{v}$ is the vector of flow velocity.

The Kronecker delta, δ_{ij} will also be used, and $\vec{\delta}_i$ is a vector of Kronecker deltas,

$$\delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}, \quad \vec{\delta}_i = \begin{Bmatrix} \delta_{i1} \\ \delta_{i2} \\ \delta_{i3} \end{Bmatrix}. \tag{A.2}$$

The following shorthand matrices, listed in vector notation and index notation, are used in

abbreviating viscous terms of the flux Jacobians,

$$\begin{aligned}
\bar{\bar{\tau}} &= \nabla\vec{v}^T + \left[\nabla\vec{v}^T\right]^T - \frac{2}{3}\bar{\bar{I}}\left(\nabla\cdot\vec{v}\right), & \tau_{ij} &= \partial_i v_j + \partial_j v_i - \frac{2}{3}\delta_{ij}\partial_k v_k \\
\bar{\bar{\eta}} &= \nabla\left(\frac{\vec{v}^T}{\rho}\right) + \left[\nabla\left(\frac{\vec{v}^T}{\rho}\right)\right]^T - \frac{2}{3}\bar{\bar{I}}\left(\nabla\cdot\frac{\vec{v}}{\rho}\right), & \eta_{ij} &= \partial_i(v_j/\rho) + \partial_j(v_i/\rho) - \frac{2}{3}\delta_{ij}\partial_k(v_k/\rho) \\
\bar{\bar{\pi}} &= \bar{\bar{\eta}} - \frac{1}{\rho}\bar{\bar{\tau}}, & \pi_{ij} &= v_j\partial_i\left(\frac{1}{\rho}\right) + v_i\partial_j\left(\frac{1}{\rho}\right) - \frac{2}{3}\delta_{ij}\left(v_k\partial_k\left(\frac{1}{\rho}\right)\right) \\
\bar{\bar{\xi}} &= \nabla\left(\frac{1}{\rho}\right)\otimes\vec{v}^T - \frac{2}{3}\vec{v}\otimes\nabla\left(\frac{1}{\rho}\right)^T, & \xi_{ij} &= v_j\partial_i\left(\frac{1}{\rho}\right) - \frac{2}{3}v_i\partial_j\left(\frac{1}{\rho}\right).
\end{aligned} \tag{A.3}$$

A number of identities related to surface variations are used during the derivation of the continuous adjoint in Section 3.1,

$$\begin{aligned}
\int_{\delta S}(\cdot)ds &= \int_{S'}(\cdot)ds - \int_S(\cdot)ds \\
\int_{\delta S} j ds &= \int_S(\partial_n j - 2H_m j)\delta S ds \\
H_m &= (\kappa_1+\kappa_2)/2 \\
\partial_n(j) &= \vec{n}\cdot\nabla(j) \\
\delta\vec{n} &= -\nabla_S(\delta S),
\end{aligned} \tag{A.4}$$

where H_m is the mean curvature of S and (κ_1, κ_2) are curvatures in two orthogonal directions on the surface. The relationship for $\delta\vec{n}$ holds for small deformations.[89] The term ∇_S represents the tangential gradient operator on S.

Within the derivation of the adjoint equation boundary terms, in Section 3.1, several shorthand terms are introduced. These are repeated here for convenient reference:

$$\begin{aligned}
\vartheta &= (\rho\psi_\rho + \rho\vec{v}\cdot\vec{\varphi} + \rho H\psi_{\rho E}) \\
\bar{\bar{\Sigma}}^{\varphi} &= \mu^1_{tot}(\nabla\vec{\varphi} + \nabla\vec{\varphi}^{\mathsf{T}} - \frac{2}{3}\bar{\bar{I}}\nabla\cdot\vec{\varphi}) \\
\bar{\bar{\Sigma}}^{\psi_{\rho E}} &= \mu^1_{tot}(\nabla\psi_{\rho E}\vec{v} + \nabla\psi_{\rho E}\vec{v}^{\mathsf{T}} - \frac{2}{3}\bar{\bar{I}}\nabla\psi_{\rho E}\cdot\vec{v}).
\end{aligned} \tag{A.5}$$

Within the derivation of the adjoint of a generalized outflow-based functional using characteristic variables in Section 3.2.3, we need the expansion of the dot product of two cross products with the

normal unit vector:

$$\begin{aligned}(\vec{a}\times\vec{n})\cdot\left(\vec{b}\times\vec{n}\right) &= \epsilon_{ijk}(a_i n_j)\epsilon_{lmk}(b_l n_m)\\ &= (\delta_{il}\delta_{jm}-\delta_{im}\delta_{jl})(a_i n_j b_l n_m)\\ &= (a_i b_i)(n_j n_j)-(a_m n_m)(b_l n_l)\\ &= \left(\vec{a}\cdot\vec{b}\right)(\vec{n}\cdot\vec{n})-(\vec{a}\cdot\vec{n})\left(\vec{b}\cdot\vec{n}\right)\end{aligned} \tag{A.6}$$

Where the permutation symbol ϵ_{ijk} is defined as:

$$\epsilon_{ijk}=\begin{cases}1 & \text{for even permutations (123,231,312)}\\ -1 & \text{for odd permutations}\\ 0 & \text{where two or more indices are equal}\end{cases} \tag{A.7}$$

A.2 Jacobians & Transformation Matrices

This section expands various forms of the Jacobians and transformation matrices used in the RANS equations and adjoint equations. The Jacobians express a matrix of the partial derivatives of a set of functions with respect to a set of variables.

A transformation matrix is needed between two sets of variables, referred to as the vector of conservative variables U and the vector of primitive variables, V,

$$U=\begin{Bmatrix}\rho\\ \rho\vec{v}\\ \rho E\end{Bmatrix},\quad V=\begin{Bmatrix}\rho\\ \vec{v}\\ P\end{Bmatrix}, \tag{A.8}$$

where ρ is the fluid density, $\vec{v}$ is the vector of flow velocity, p is the fluid static pressure, and E is the total energy per unit mass. A transformation matrix, M, is needed to convert between the conservative and primitive variables. The transformation matrix between conservative and primitive variables, M, and its inverse are shown here in vector form:

$$\begin{aligned}M=\frac{\partial U}{\partial V}&=\begin{bmatrix}1 & \vec{0}^T & 0\\ \vec{v} & \bar{\bar{I}}\rho & 0\\ \frac{|\vec{v}|^2}{2} & \rho\vec{v}^T & \frac{1}{\gamma-1}\end{bmatrix}\\ M^{-1}=\frac{\partial V}{\partial U}&=\begin{bmatrix}1 & \vec{0}^T & 0\\ -\vec{v}/\rho & \bar{\bar{I}}\frac{1}{\rho} & 0\\ \frac{|\vec{v}|^2}{2}(\gamma-1) & -\vec{v}^T(\gamma-1) & (\gamma-1)\end{bmatrix}.\end{aligned} \tag{A.9}$$

Jacobians

The convective flux Jacobian is:

$$\vec{A}_i^c = \frac{\partial F_i^c}{\partial U} = \begin{bmatrix} 0 & \vec{\delta}_i^T & 0 \\ -\vec{v}v_i + \vec{\delta}_i\phi & \bar{\bar{I}}v_i + \vec{v}\otimes\vec{\delta}_i - a_0\vec{\delta}_i\otimes\vec{v} & \vec{\delta}_i a_0 \\ v_i\left(\phi - H\right) & \vec{\delta}_i^T(H) - a_0\vec{v}^T v_i & \gamma v_i \end{bmatrix}. \tag{A.10}$$

The viscous stress Jacobian terms are:

$$A_i^{\nu 1} = \frac{\partial F_i^{\nu 1}}{\partial U} = \begin{bmatrix} 0 & \vec{0}^T & 0 \\ -\bar{\bar{\eta}}\cdot\vec{\delta}_i & \left[\bar{\bar{I}}(\nabla\frac{1}{\rho}\cdot\vec{\delta}_i + \nabla\frac{1}{\rho}\otimes\vec{\delta}_i - \frac{2}{3}\vec{\delta}_i\otimes\nabla\frac{1}{\rho}\right] & \vec{0} \\ (\bar{\bar{\pi}}\vec{v})\cdot\vec{\delta}_i & \left\{[\bar{\bar{\zeta}} + \frac{1}{\rho}\bar{\bar{\tau}}]\vec{\delta}_i + (\vec{v}\cdot\nabla\frac{1}{\rho})\vec{\delta}_i\right\}^T & 0 \end{bmatrix}, \tag{A.11}$$

where shorthand terms $\bar{\bar{\eta}}$, $\bar{\bar{\pi}}$ and $\bar{\bar{\zeta}}$ in the viscous Jacobian are included in Equation Set A.3. The viscous Jacobian of the heat transfer terms is:

$$A_i^{\nu 2} = \frac{\partial F_i^{\nu 2}}{\partial U} = \gamma \begin{pmatrix} 0 & \vec{0}^T & 0 \\ \vec{0} & \bar{\bar{0}} & \vec{0} \\ \frac{1}{a_0}\frac{\partial}{\partial x_i}\left(\frac{\phi}{\rho} - \frac{p}{\rho^2}\right) & -\frac{\partial}{\partial x_i}\left(\frac{\vec{v}}{\rho}\right) & \frac{\partial}{\partial x_i}\left(\frac{1}{\rho}\right) \end{pmatrix}, \tag{A.12}$$

The Jacobian of the viscous stresses with respect to flow gradients, used in the viscous adjoint equations, can be expressed compactly using $\vec{\delta}_i = \{\delta_{i1}, \delta_{i2}, \delta_{i3}\}^\intercal$ as:

$$\bar{\bar{D}}_{ij}^{\nu 1} = \begin{bmatrix} 0 & \vec{0}^\intercal & 0 \\ -v_i\vec{\delta}_j + \frac{2}{3}v_j\vec{\delta}_i - \vec{v}\delta_{ij} & \vec{\delta}_j\otimes\vec{\delta}_i - \frac{2}{3}\vec{\delta}_i\otimes\vec{\delta}_j + \bar{\bar{I}}\delta_{ij} & \vec{0} \\ -\frac{1}{3}v_i v_j - |\vec{v}|^2\delta_{ij} & v_j\vec{\delta}_i^\intercal - \frac{2}{3}v_i\vec{\delta}_j^\intercal + \vec{v}^T\delta_{ij} & 0 \end{bmatrix}. \tag{A.13}$$

It is sometimes useful to expand the components of the viscous stress Jacobian:

$$
\begin{aligned}
D_{ii}^{\nu 1} &= \frac{\partial F_i^{\nu 1}}{\partial(\partial U/\partial x_i)} = \frac{1}{\rho}\begin{pmatrix}
\cdot & \cdot & \cdot & \cdot & \cdot \\
-\left(1+\frac{1}{3}\delta_{i1}\right)v_1 & \left(1+\frac{1}{3}\delta_{i1}\right) & \cdot & \cdot & \cdot \\
-\left(1+\frac{1}{3}\delta_{i2}\right)v_2 & \cdot & \left(1+\frac{1}{3}\delta_{i2}\right) & \cdot & \cdot \\
-\left(1+\frac{1}{3}\delta_{i3}\right)v_3 & \cdot & \cdot & \left(1+\frac{1}{3}\delta_{i3}\right) & \cdot \\
-|\vec{v}|^2-\frac{1}{3}v_i^2 & \left(1+\frac{1}{3}\delta_{i1}\right)v_1 & \left(1+\frac{1}{3}\delta_{i2}\right)v_2 & \left(1+\frac{1}{3}\delta_{i3}\right)v_3 & \cdot
\end{pmatrix}, \\
&= \frac{1}{\rho}\begin{pmatrix}
0 & \vec{0}^T & 0 \\
-\left(\bar{\bar{I}}+\frac{1}{3}\bar{\bar{I}}\vec{\delta}_i\right)\vec{v} & \left(\bar{\bar{I}}+\frac{1}{3}\bar{\bar{I}}\vec{\delta}_i\right) & \vec{0} \\
-|\vec{v}|^2-\frac{1}{3}v_i^2 & \left[\left(\bar{\bar{I}}+\frac{1}{3}\bar{\bar{I}}\vec{\delta}_i\right)\vec{v}\right]^T & 0
\end{pmatrix}, \\
D_{ij}^{\nu 1} &= \frac{\partial F_i^{\nu 1}}{\partial(\partial U/\partial x_j)} \\
&= \frac{1}{\rho}\begin{pmatrix}
\cdot & \cdot & \cdot & \cdot & \cdot \\
-v_i\delta_{j1}+\frac{2}{3}v_j\delta_{i1} & \delta_{j1}\delta_{i1}-\frac{2}{3}\delta_{i1}\delta_{j1} & \delta_{j1}\delta_{i2}-\frac{2}{3}\delta_{i1}\delta_{j2} & \delta_{j1}\delta_{i3}-\frac{2}{3}\delta_{i1}\delta_{j3} & \cdot \\
-v_i\delta_{j2}+\frac{2}{3}v_j\delta_{i2} & \delta_{j2}\delta_{i1}-\frac{2}{3}\delta_{i2}\delta_{j1} & \delta_{j2}\delta_{i2}-\frac{2}{3}\delta_{i2}\delta_{j2} & \delta_{j2}\delta_{i3}-\frac{2}{3}\delta_{i2}\delta_{j3} & \cdot \\
-v_i\delta_{j3}+\frac{2}{3}v_j\delta_{i3} & \delta_{j3}\delta_{i1}-\frac{2}{3}\delta_{i3}\delta_{j1} & \delta_{j3}\delta_{i2}-\frac{2}{3}\delta_{i3}\delta_{j2} & \delta_{j3}\delta_{i3}-\frac{2}{3}\delta_{i3}\delta_{j3} & \cdot \\
-\frac{1}{3}v_iv_j & v_j\delta_{i1}-\frac{2}{3}v_i\delta_{j1} & v_j\delta_{i2}-\frac{2}{3}v_i\delta_{j2} & v_j\delta_{i3}-\frac{2}{3}v_i\delta_{j3} & \cdot
\end{pmatrix} \quad (i\neq j), \\
&= \frac{1}{\rho}\begin{pmatrix}
0 & \vec{0}^T & 0 \\
-v_i\vec{\delta}_j+\frac{2}{3}v_j\vec{\delta}_i & \left[\vec{\delta}_j\otimes\vec{\delta}_i-\frac{2}{3}\vec{\delta}_i\otimes\vec{\delta}_j\right] & \vec{0} \\
-\frac{1}{3}v_iv_j & v_j\vec{\delta}_i^T-\frac{2}{3}v_i\vec{\delta}_j^T & 0
\end{pmatrix} \quad (i\neq j),
\end{aligned}
\tag{A.14}
$$

The Jacobian of heat transfer components with respect to flow gradients can be expressed as:

$$
\begin{aligned}
D_{ii}^{\nu 2} &= \frac{\partial F_i^{\nu 2}}{\partial(\partial U/\partial x_i)} = \frac{\gamma}{\rho}\begin{pmatrix}
\cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot \\
\frac{1}{a_0}\left(\phi-\frac{p}{\rho}\right) & -v_1 & -v_2 & -v_3 & 1
\end{pmatrix}, \\
&= \frac{\gamma}{\rho}\begin{pmatrix}
0 & \vec{0}^T & 0 \\
\vec{0} & \bar{\bar{0}} & \vec{0} \\
\frac{1}{a_0}\left(\phi-\frac{p}{\rho}\right) & -\vec{v}^T & 1
\end{pmatrix}, \\
D_{ij}^{\nu 2} &= \frac{\partial F_i^{\nu 2}}{\partial(\partial U/\partial x_j)} = \mathbf{0}_{5\times 5} \quad (i\neq j).
\end{aligned}
\tag{A.15}
$$

Compactly,

$$\bar{\bar{D}}_{ij}^{\nu 2} = \delta_{ij}\frac{\gamma}{\rho}\begin{pmatrix} 0 & \vec{0}^T & 0 \\ \vec{0} & \bar{\bar{0}} & \vec{0} \\ \frac{1}{a_0}\left(\phi - \frac{P}{\rho}\right) & -\vec{v}^{\intercal} & 1 \end{pmatrix}. \tag{A.16}$$

Diagonalization

Diagonalization of the Euler Jacobian is used to find characteristics of the problem. In two dimensions,

$$\begin{aligned} \Delta &= L^{-1}M^{-1}\left(\vec{A}\cdot\vec{n}\right)ML \\ L = \frac{\partial W}{\partial V} &= \begin{bmatrix} 1 & 0 & \frac{\rho}{2c} & \frac{\rho}{2c} \\ 0 & n_y & \frac{n_x}{2} & \frac{-n_x}{2} \\ 0 & -n_x & \frac{n_y}{2} & \frac{-n_y}{2} \\ 0 & 0 & \frac{\rho}{2c} & \frac{\rho}{2c} \end{bmatrix} \\ \Delta &= \begin{bmatrix} \vec{v}\cdot\vec{n} & 0 & 0 & 0 \\ 0 & \vec{v}\cdot\vec{n} & 0 & 0 \\ 0 & 0 & (\vec{v}\cdot\vec{n}+c) & 0 \\ 0 & 0 & 0 & (\vec{v}\cdot\vec{n}-c) \end{bmatrix} \\ W &= L^{-1}M^{-1}U. \end{aligned} \tag{A.17}$$

W is the vector of characteristic variables, which are constant along characteristics of these equations, and result from the diagonalization of the convective Jacobian that also provides the characteristic speeds in the resulting eigenvalues shown in the matrix Δ.

Projections & Transformations of Jacobians

The Jacobians projected in the direction of the normal vector $\vec{n}$ are sometimes useful:

$$\vec{A}^c\cdot\vec{n} = \begin{bmatrix} 0 & \vec{n}^T & 0 \\ -\vec{v}(\vec{v}\cdot\vec{n}) + \phi\vec{n} & \left[\bar{\bar{I}}(\vec{v}\cdot\vec{n}) + \vec{v}\otimes\vec{n}^T - a_0\vec{n}\otimes\vec{v}^T\right] & a_0\vec{n} \\ (\vec{v}\cdot\vec{n})(\phi - H) & H\vec{n}^T - a_0(\vec{v}\cdot\vec{n})\vec{v}^T & \gamma(\vec{v}\cdot\vec{n}) \end{bmatrix}, \tag{A.18}$$

$$\vec{A}^{\nu 1}\cdot\vec{n} = \begin{bmatrix} 0 & \vec{0}^T & 0 \\ -\bar{\bar{\eta}}\cdot\vec{n} & \left[\bar{\bar{I}}(\nabla\frac{1}{\rho}\cdot\vec{n}) + \nabla\frac{1}{\rho}\otimes\vec{n}^T - \frac{2}{3}\vec{n}\otimes\nabla\frac{1}{\rho}^T\right] & \vec{0} \\ (\bar{\bar{\pi}}\vec{v})\cdot\vec{n} & \left\{[\bar{\bar{\xi}}^T + \frac{1}{\rho}\bar{\bar{\tau}}^T]\vec{n} + (\vec{v}\cdot\nabla\frac{1}{\rho})\vec{n}\right\}^T & 0 \end{bmatrix} = \begin{bmatrix} 0 & \vec{0}^T & 0 \\ -\bar{\bar{\eta}}\cdot\vec{n} & \vec{C}^{\nu 1} & \vec{0} \\ (\bar{\bar{\pi}}\vec{v})\cdot\vec{n} & \vec{C}^{\nu 2\,T} & 0 \end{bmatrix}, \tag{A.19}$$

$$\vec{A}^{\nu 2}\cdot\vec{n}=\gamma\begin{bmatrix} 0 & \vec{0}^T & 0 \\ \vec{0} & \bar{\bar{0}} & \vec{0} \\ \frac{1}{a_0}\vec{n}\cdot\nabla\left(\frac{\phi}{\rho}-\frac{P}{\rho^2}\right) & -(\vec{n}\cdot\nabla)\frac{\vec{v}^T}{\rho} & \vec{n}\cdot\nabla\frac{1}{\rho} \end{bmatrix}, \tag{A.20}$$

where new shorthand terms are:

$$\begin{aligned} C_{ij}^{\nu 1} &= \delta_{ij}\left(n_k\partial_k\left(\frac{1}{\rho}\right)\right)+n_j\partial_i\left(\frac{1}{\rho}\right)-\frac{2}{3}n_i\partial_j\left(\frac{1}{\rho}\right) \\ C_i^{\nu 2} &= \xi_{ji}n_j+\frac{1}{\rho}\tau_{ji}n_j+v_i\partial_j\left(\frac{1}{\rho}\right)n_j\,. \end{aligned} \tag{A.21}$$

In order to find the boundary conditions of the direct and adjoint problems, we need to find the convective and viscous Jacobians projected in the normal direction $\vec{n}$ in terms of the primitive variables V. In order to express these terms, the transformation matrix M shown in Equation Set A.9 is used,

$$\left(\vec{A}^c\cdot\vec{n}\right)M=\begin{bmatrix} \vec{v}\cdot\vec{n} & \rho\vec{n}^T & 0 \\ \vec{v}(\vec{v}\cdot\vec{n}) & \left[\rho(\vec{v}\cdot\vec{n})\bar{\bar{I}}+\rho\vec{v}\otimes\vec{n}^T\right] & \vec{n} \\ v_n\frac{\vec{v}^2}{2} & \rho(H\vec{n}^T+(\vec{v}\cdot\vec{n})\vec{v}^T) & \frac{\gamma\vec{v}\cdot\vec{n}}{\gamma-1} \end{bmatrix}, \tag{A.22}$$

$$\left(\vec{A}^{\nu 2}\cdot\vec{n}\right)M=\gamma\begin{bmatrix} 0 & \vec{0}^T & 0 \\ \vec{0} & \bar{\bar{0}} & \vec{0} \\ -\vec{n}\cdot\nabla\left(\frac{c^2}{\rho(\gamma-1)}\right)+\vec{v}^2\vec{n}\cdot\nabla\left(\frac{1}{\rho}\right) & \rho\vec{v}^T\left(\vec{n}\cdot\nabla\frac{1}{\rho}\right)-(\vec{n}\cdot\nabla)\vec{v}^T & \frac{\vec{n}\cdot\nabla\frac{1}{\rho}}{\gamma-1} \end{bmatrix}. \tag{A.23}$$

More compactly:

$$\left(\vec{A}^{\nu 1}\cdot\vec{n}\right)M=\begin{bmatrix} 0 & \vec{0}^T & 0 \\ \left(\bar{\bar{C}}^{\nu 1}\vec{v}-\bar{\bar{\eta}}\cdot\vec{n}\right) & \rho\bar{\bar{C}}^{\nu 1} & \vec{0} \\ (\bar{\bar{\pi}}\vec{v})\cdot\vec{n}+\vec{C}^{\nu 2}\cdot\vec{v} & \rho\vec{C}^{\nu 2T} & 0 \end{bmatrix}, \tag{A.24}$$

$$\left(\vec{A}^{\nu 2}\cdot\vec{n}\right)M=\gamma\begin{bmatrix} 0 & \vec{0}^T & 0 \\ \vec{0} & \bar{\bar{0}} & \vec{0} \\ C^{\nu 3} & \vec{C}^{\nu 4T} & C^{\nu 5} \end{bmatrix}. \tag{A.25}$$

A.3 Generalized Adjoint Formulation Quick-Reference

The volume integral that defines the adjoint PDE, and was previously developed in literature is:

$$\left[-\nabla\Psi^{\mathsf{T}}\cdot\left(\vec{A}^{c}-\bar{\bar{I}}_5\vec{u}_\Omega-\mu_{tot}^{k}\vec{A}^{vk}\right)-\nabla\cdot\left(\nabla\Psi^{\mathsf{T}}\cdot\mu_{tot}^{k}\bar{\bar{D}}^{vk}\right)-\Psi^{\mathsf{T}}\frac{\partial\mathcal{Q}}{\partial U}\right]=0\quad\text{on }\Omega. \tag{A.26}$$

Table A.1 and Table A.2 follow existing literature, although the terms of sensitivity are not usually separated as shown in Table A.2. Table A.3 summarizes the adjoint boundary conditions for generalized functionals, which were derived as part of this dissertation. The generalized functional variation, dependent only on partial derivative terms, and which was derived as part of this dissertation is:

$$\begin{aligned}\delta J=&\int_{\Gamma_e}\left(\frac{\partial j}{\partial\rho}\delta\rho+\frac{\partial j}{\partial\vec{v}}\cdot\delta\vec{v}+\frac{\partial j}{\partial P}\delta P\right)ds\\&+\int_S\left(\frac{\partial j}{\partial\vec{f}}\cdot(\bar{\bar{I}}\delta p-\delta\bar{\bar{\sigma}})\cdot\vec{n}+\frac{\partial j}{\partial(\partial_n T)}\left(\delta\left(\partial_n T\right)\right)+\frac{\partial j}{\partial T}\delta T\right)ds\\&+\int_S\left\{\frac{\partial j}{\partial\vec{f}}\cdot[\vec{q}_{\rho\vec{v}}]+\nabla\frac{\partial j}{\partial\vec{f}}:(\bar{\bar{I}}p-\bar{\bar{\sigma}})-(\bar{\bar{I}}p-\bar{\bar{\sigma}})\cdot\vec{n}\cdot\partial_n\frac{\partial j}{\partial\vec{f}}\right\}\delta S ds\\&+\int_S\left\{\nabla\cdot\left(\frac{\partial j}{\partial(\partial_n T)}\nabla T\right)+\frac{\partial j}{\partial T}\partial_n T\right\}\delta S\,ds+\int_S\left\{2H_m\left(\vec{f}\cdot\frac{\partial j}{\partial\vec{f}}+\frac{\partial j}{\partial(\partial_n T)}\partial_n T-j\right)\right\}\delta S ds\,.\end{aligned} \tag{A.27}$$

Table A.1: Linearized Boundary Conditions for Fluid Flow

Boundary Type	Boundary Condition	Linearized Form
Flow Tangency	$(\vec{v}-\vec{u}_\Omega)\cdot\vec{n}=0$	$\delta\vec{v}\cdot\vec{n}=(\vec{v}-\vec{u}_\Omega)\cdot\nabla_S(\delta S)-\partial_n(\vec{v}-\vec{u}_\Omega)\delta S\cdot\vec{n}$
No-Slip Wall	$\vec{v}-\vec{u}_\Omega=\vec{0}$	$\delta\vec{v}=-\partial_n(\vec{v}-\vec{u}_\Omega)\delta S$
Adiabatic Wall	$\partial_n T=\vec{n}\cdot\nabla T=0$	$\partial_n(\delta T)=\nabla T\cdot\nabla_S(\delta S)-\partial_n^2(T)\delta S$
Constant q_n Wall	$\partial_n T=\vec{n}\cdot\nabla T=\frac{q_n}{c_p}$	$\partial_n(\delta T)=\nabla T\cdot\nabla_S(\delta S)-\partial_n^2(T)\delta S$
Isothermal Wall	$T=T_w$	$\delta T=-\partial_n(T)\delta S$
Farfield, Inflow, Outflow	$(W)_+=W_\infty$	$(\delta W)_+=0.$
Subsonic Outflow: prescribed pressure	1 value prescribed $P=P_e$	$k-1$ variations arbitrary $\delta P=0$
Subsonic Inflow: $\dot{m}$ prescribed	$k-1$ values prescribed $\rho\vec{v}=\rho_0\vec{v}_0$	1 variation arbitrary $\delta\rho=\delta\vec{v}=0$
Supersonic Inflow:	k values prescribed	no variation arbitrary $\delta\rho=\delta\vec{v}=\delta P=0$
Supersonic Outflow:	0 values prescribed	all variations arbitrary

Table A.2: Sensitivity Contributions from Direct Problem Boundary Conditions

Boundary Type	Contribution to Surface Sensitivity
Flow Tangency	$\{\vartheta\nabla\cdot(\vec{v}-\vec{u}_\Omega)+\nabla(\vartheta)\cdot(\vec{v}-\vec{u}_\Omega)\}$
No-Slip	$\left\{\vec{n}\cdot\left(\vartheta(\partial_n(\vec{v}-\vec{u}_\Omega))+\psi_{\rho E}\left(-\partial_n(\vec{v}-\vec{u}_\Omega)\cdot\bar{\bar{\sigma}}\right)+\left(\bar{\bar{\Sigma}}^{\varphi}+\bar{\bar{\Sigma}}^{\psi_{\rho E}}\right)\cdot(\partial_n(\vec{v}-\vec{u}_\Omega))\right)\right\}$
Adiabatic Wall	$\left\{-\mu_{tot}^2 c_p\nabla_S(\psi_{\rho E})\cdot\nabla_S(T)-\psi_{\rho E}(P(\nabla\cdot\vec{v})-\bar{\bar{\sigma}}:\nabla\vec{v}+(\vec{q}_{\rho\vec{v}}\cdot\vec{v}-q_{\rho E})\right\}$
Isothermal Wall	$\left\{-\mu_{tot}^2 c_p\partial_n(\psi_{\rho E})\partial_n(T)\right\}$

Table A.3: Adjoint Boundary Conditions for Generalized Functionals

Boundary Condition	Adjoint Boundary Condition (Stationary Domain)
Flow tangency	$\vec{\varphi}\cdot\vec{n}=\frac{\partial j}{\partial\vec{f}}\cdot\vec{n}-\psi_{\rho E}\vec{v}\cdot\vec{n}$ $\left(\vec{\varphi}\cdot\vec{n}=\frac{\partial j}{\partial\vec{f}}\cdot\vec{n}\right)$
Adiabatic wall	$\vec{\varphi}=\frac{\partial j}{\partial\vec{f}}-\psi_{\rho E}\vec{v}$ $\partial_n(\psi_{\rho E})=\frac{1}{c_p\mu_{tot}^2}\frac{\partial j}{\partial T}$ $\left(\begin{array}{c}\vec{\varphi}=\frac{\partial j}{\partial\vec{f}}\\ \partial_n(\psi_{\rho E})=\frac{1}{c_p\mu_{tot}^2}\frac{\partial j}{\partial T}\end{array}\right)$
Isothermal wall	$\vec{\varphi}=\frac{\partial j}{\partial\vec{f}}-\psi_{\rho E}\vec{v}$ $\psi_{\rho E}=-\frac{\partial j}{\partial(\partial_n T)}\frac{1}{c_p\mu_{tot}^2}$ $\left(\begin{array}{c}\vec{\varphi}=\frac{\partial j}{\partial\vec{f}}\\ \psi_{\rho E}=-\frac{\partial j}{\partial(\partial_n T)}\frac{1}{c_p\mu_{tot}^2}\end{array}\right)$
Pressure-based Outflow (inviscid & viscous neglecting $\delta\mu$)	$\left\{\begin{array}{c}\psi_\rho\\ \vec{\varphi}\end{array}\right\}=\psi_{\rho E}\left\{\begin{array}{c}\frac{2c^2v_n+\vec{v}^2v_{nr}(\gamma-1)}{2(\gamma-1)v_{nr}}\\ -\vec{n}\frac{c^2}{v_{nr}(\gamma-1)}-\vec{v}\end{array}\right\}+\left\{\begin{array}{c}-\left(\frac{\partial j}{\partial\vec{v}}\cdot\vec{v}\frac{1}{\rho v_{nr}}\right)+\left(\frac{\partial j}{\partial\rho}\frac{v_{nr}+v_n}{v_{nr}^2}\right)\\ \left(\frac{\partial j}{\partial\vec{v}}\frac{1}{\rho v_{nr}}-\vec{n}\frac{\partial j}{\partial\rho}\frac{1}{v_{nr}^2}\right)\end{array}\right\}$ $\psi_{\rho E,M_e>1}=\frac{\gamma-1}{v_{nr}^2-c^2}\left(\frac{\partial j}{\partial\rho}\frac{1}{v_{nr}}+\frac{\partial j}{\partial P}v_{nr}-\frac{\partial j}{\partial\vec{v}}\cdot\vec{n}\frac{1}{\rho}\right)$ $\left(\begin{array}{c}\left\{\begin{array}{c}\psi_\rho\\ \vec{\varphi}\end{array}\right\}=\psi_{\rho E}\left\{\begin{array}{c}\frac{2c^2+\vec{v}^2(\gamma-1)}{2(\gamma-1)}\\ -\vec{n}\frac{c^2}{v_n(\gamma-1)}-\vec{v}\end{array}\right\}+\left\{\begin{array}{c}-\left(\frac{\partial j}{\partial\vec{v}}\cdot\vec{v}\frac{1}{\rho v_n}\right)+\left(\frac{\partial j}{\partial\rho}\frac{2}{v_n}\right)\\ \left(\frac{\partial j}{\partial\vec{v}}\frac{1}{\rho v_n}-\vec{n}\frac{\partial j}{\partial\rho}\frac{1}{v_n^2}\right)\end{array}\right\}\\ \psi_{\rho E,M_e>1}=\frac{\gamma-1}{v_n^2-c^2}\left(\frac{\partial j}{\partial\rho}\frac{1}{v_n}+\frac{\partial j}{\partial P}v_n-\frac{\partial j}{\partial\vec{v}}\cdot\vec{n}\frac{1}{\rho}\right)\end{array}\right)$
Inlet	$\psi_{\rho E,M<1}=-\vec{\varphi}\cdot\vec{n}\frac{(\gamma-1)}{\gamma\vec{v}\cdot\vec{n}-\vec{u}_\Omega\cdot\vec{n}}$ $\left(\psi_{\rho E,M<1}=-\vec{\varphi}\cdot\vec{n}\frac{(\gamma-1)}{\gamma\vec{v}\cdot\vec{n}}\right)$

Table A.4: Sensitivity Contributions from Functional Variations

Functional	Contribution to Surface Sensitivity
Generalized force-based	$\left\{\frac{\partial j}{\partial\vec{f}}\cdot[\vec{q}_{\rho\vec{v}}-\partial_t(\rho\vec{v})]+\nabla\frac{\partial j}{\partial\vec{f}}:(\bar{\bar{I}}p-\bar{\bar{\sigma}})-(\bar{\bar{I}}p-\bar{\bar{\sigma}})\cdot\vec{n}\cdot\partial_n\frac{\partial j}{\partial\vec{f}}\right\}$ $+\left\{2H_m(\vec{f}\cdot\frac{\partial j}{\partial\vec{f}}-j)\right\}$
Generalized heat flux (isothermal)	$\left\{\nabla\cdot\left(\frac{\partial j}{\partial(\partial_n T)}\nabla T\right)+2H_m\left(\frac{\partial j}{\partial(\partial_n T)}\partial_n T-j\right)\right\}$
Generalized temperature (adiabatic)	$\frac{\partial j}{\partial T}\partial_n T-2H_m j$

Appendix B

Code Reference

B.1 Versions of SU2

At the time of this dissertation, SU2 uses Github for version control. The code and version history can be accessed at `www.github.com/su2code/SU2`, along with installation instructions and tutorials in the wiki included on this site.

The modifications made to this code for the work in this dissertation can be accessed in a forked repository of this code, under the branch name "version_Dissertation". This code can be viewed at: `https://github.com/hlkline/SU2/tree/version_Dissertation`. This is the code version used for the results produced in Section 6.2.6 and Section 6.3, and includes all of the developments noted as having been implemented in this dissertation, including: a generalized outflow functional for the continuous adjoint in terms of either area-averaged or mass-flux averaged quantities, combinations of objectives in the continuous and discrete adjoint, a wrapper for accessing SNOPT utilities through the SU2 shape_optimization.py script, and the ability to address quadratic penalty functions. Earlier versions of this code were used to produce the results in prior sections, with some differences such as what quantities are output. This earlier version, which lacks the full functionality of "version_Dissertation", can be found in the code branch "version_Disseration_partial". This branch was used for results in Section 6.2.3 through Section 6.2.5.

Some of the functionality described in this work has been added to the develop branch of SU2, and may be available in future releases, however no guarantee can be made that this functionality will be maintained.

B.2 Use of Generalized Outflow Function

This section describes the use of the generalized outflow functional and the interface with external functions. In order to evaluate an external function based on averaged outflow quantities, first the

SU2 configuration file must be set up to output the averaged outflow quantities. This is accomplished with the following settings, in addition to the normal SU2 settings required to run the flow and adjoint simulations:

```
%
% Outlet boundary marker(s) (NONE = no marker)
% Format: ( outlet marker, back pressure (static), ... )
MARKER_OUTLET = (outflow_marker_name, 101325.0)
% Boundary marker(s) where one-dimensionalized outputs will
% be evaluated.
MARKER_OUT_1D = (outflow_marker_name)
% Objective function in optimization problem
OBJECTIVE_FUNCTION = OUTFLOW_GENERALIZED
% Kind of one-dimensionalization used
KIND_ONE_DIMENSIONALIZATION = AREA
```

The "outflow_marker_name" should be replaced with the name of the relevant marker in the mesh file, and a reasonable back pressure value should be chosen. For supersonic flows, choose a low pressure. Area-averaging is used by default.

An external file, named "downstream_function.py" must be defined in the workspace. This file name is hard-coded, and modifications to the SU2 python scripts and functions will be required if a different file name is desired. This external file must include the function definitions for "downstream_function(config,state)" and "downstream_gradient(config,state,step=default)". These functions are called during pre- and post- processing steps for running the continuous adjoint problem and evaluating the function value. An example of such a function is included below.

```
#!/usr/bin/env python
## \file downstream_function.py

# imports
sys.path.append(os.environ['SU2_RUN'])
import SU2

def downstream_function(config, state ):
  d_in = [0.0]*nvar
  return objective(config, state,d_in)

def downstream_gradient(config,state, step=1e-3):
  # this function provides 5 gradients whether 2D or 3D
  # [ dj / drho , dj/dv1, dj/dv2, dj/dv3 or 0, dj/dP]
```

```
  nvar = 5 # density, 3 directions of velocity, pressure
  d_in = [0.0]*nvar
  dJ = [0.0]*nvar
  J0 = objective(config,state,d_in)
  # Loop over variables
  for i in range(nvar):
    d_in = [0.0]*nvar
    d_in[i]=step
    dJ[i] = (objective(config,state,d_in)-J0)/step

  gradient = tuple(dJ)
  return gradient

def objective(config,state,d_in):
  # Reference values from last iteration of history
  rho = state['HISTORY']['DIRECT']['AVG_OUTLET_DENSITY'][-1];
  V = state['HISTORY']['DIRECT']['AVG_OUTLET_VELOCITY'][-1];
  P3 = float(state['HISTORY']['DIRECT']['AVG_OUTLET_PRESSURE'][-1])
  # Mass flow rate is also often useful
  mdot = float(state['HISTORY']['DIRECT']['MASS_FLOW_RATE'][-1])
  rho+=d_in[0]
  V  +=d_in[1] #For +x outflow normal direction
  P  +=d_in[4]
  # A dummy function
  obj = f(P,rho,V)
  return obj
```

This example demonstrates using a finite difference to evaluate the gradients of the "downstream" function. The accuracy of these gradients is important to the adjoint solution, and the step size can be varied proportional to the scale of the input quantities of density, velocity, and pressure as these quantities may vary widely in their magnitude, and so not produce the same level of accuracy for the same step size. The gradient values could also be provided analytically, or through other methods.

Once the configuration, mesh, and downstream_function.py files have been set up appropriately in the working directory, either the finite difference or continuous adjoint scripts can be used. Further information about these scripts is provided with the SU2 tutorials.

```
$ finite_difference.py -f config.cfg
$ continuous_adjoint.py -f config.cfg
```

Note that the actual "downstream" function value will not be included in the normal SU2 output files, with the exception of the optimization history file. Printed output or redirection to files can be included in the downstream_function.py routines.

B.3 Multi-Objective Gradients

In order to evaluate a weighted sum of gradients, the following settings should be included in the SU2 configuration file:

```
% Objective function in optimization problem
OBJECTIVE_FUNCTION = OUTFLOW_GENERALIZED, DRAG
% Coefficients for weighted sum of objectives
OBJECTIVE_WEIGHT = 1.0, -2.0
% Marker(s) of the surface where the functional (Cd, Cl, etc.)
% will be evaluated
MARKER_MONITORING = (outflow_marker_name, wall_marker_name)
```

The settings listed in B.2 should also be included if the OUTFLOW_GENERALIZED objective is used. The sample as shown would evaluate the gradient of $1.0 \times f(\bar{P}, \bar{\rho}, \bar{v}) - 2.0 \times C_D$.

B.4 Multi-Objective and Penalized Optimization

This section describes the configuration settings and other steps needed to run a multi-objective optimization problem or a constrained optimization problem using a penalty function. The steps to set up the downstream function file and configuration file described in B.2 should be completed first, and the user should check that the gradients are successfully evaluated before proceeding. The settings listed in B.2 should also be included if the OUTFLOW_GENERALIZED objective is used, and for any optimization problem the design variables and other optimizer parameters should be set.

For a weighted sum of objectives, a list of objective components are separated by semicolons.

```
% Marker(s) of the surface where the functional (Cd, Cl, etc.)
% will be evaluated
MARKER_MONITORING = (wall_marker_name, outflow_marker_name)
% Optimization objective function with scaling factor
% ex= Objective * Scale
OPT_OBJECTIVE=LIFT*-1.0E4;OUTFLOW_GENERALIZED*0.01
%
% Optimization constraint functions with scaling factors,
```

```
% separated by semicolons
% ex= (Objective = Value ) * Scale, use '>','<','='
OPT_CONSTRAINT= NONE
%
% Maximum number of iterations
OPT_ITERATIONS= 100
%
...
```

To apply a quadratic penalty, the objective component is listed in the format of a constraint.

```
% Marker(s) of the surface where the functional (Cd, Cl, etc.)
% will be evaluated
MARKER_MONITORING = (wall_marker_name, outflow_marker_name)
% Optimization objective function with scaling factor
% ex= Objective * Scale
OPT_OBJECTIVE=LIFT*-1.0E4;(OUTFLOW_GENERALIZED>5.0)*0.01
% When using multiple objectives, choose whether to evaluate
% the gradients sequentially or as a single, combined gradient.
OPT_COMBINE_OBJECTIVE = YES
```

During the optimization process when OPT_COMBINE_OBJECTIVE is set to "YES", the gradients will be evaluated as a weighted sum, with the weight option shown in B.3 automatically set to the partial derivative of the penalty function with respect to the functional. If this option is not provided or set to "NO", the gradients will be evaluated one by one, which may be preferable if a component of the objective function is also used in OPT_CONSTRAINT (for example, if 2 functionals are combined in the objective and the same 2 are also in constraints, combining the gradient will require 3 adjoint evaluations while separate evaluation will only require 2). In order to use some penalty function other than the quadratic penalty, the file SU2_PY/SU2/eval/design.py in the SU2 source code directory should be modified, and the code recompiled.

Bibliography

[1] J. Towns, T. Cockerill, M. Dahan, I. Foster, K. Gaither, A. Grimshaw, V. Hazlewood, S. Lathrop, D. Lifka, G.D. Peterson, et al. XSEDE: Accelerating scientific discovery. *Computing in Science & Engineering*, 16(5):62–74, 2014.

[2] L. Marshall, C. Bahm, G. Corpening, and R. Sherrill. Overview with results and lessons learned of the X-43A Mach 10 flight. *AIAA/CIRA 13th International Space Planes and Hypersonics Systems and Technologies Conference*, pages 1–23, May 2005.

[3] J.M. Hank, J.S. Murphy, and R.C. Mutzman. The X-51A scramjet engine flight demonstration program. *15th AIAA International Space Planes and Hypersonic Systems and Technologies Conference*, pages 1–13, May 2008.

[4] K.R. Jackson, M.R. Gruber, and S. Buccellato. HIFiRE flight 2 overview and status update 2011. In *17th AIAA International Space Planes and Hypersonic Systems and Technologies Conference*, 2011.

[5] J. Slotnick, A. Khodadoust, J. Alonso, D. Darmofal, W. Gropp, E. Lurie, and D. Mavriplis. CFD vision 2030 study: A path to revolutionary computational aerosciences, 2014.

[6] Thomas J.R. Hughes, Luca Mazzei, and Kenneth E. Jansen. Large eddy simulation and the variational multiscale method. *Computing and Visualization in Science*, 3(1):47–59, 2000.

[7] Philippe R Spalart. Detached-eddy simulation. *Annual review of fluid mechanics*, 41:181–202, 2009.

[8] Parviz Moin and Krishnan Mahesh. Direct numerical simulation: a tool in turbulence research. *Annual review of fluid mechanics*, 30(1):539–578, 1998.

[9] Michael Keidar, Minkwan Kim, and Iain Boyd. Electromagnetic reduction of plasma density during atmospheric reentry and hypersonic flights. *Journal of Spacecraft and Rockets*, 45(3):445–453, 2008.

[10] Khaled Abdol-Hamid, Farhad Ghaffari, and Edward Parlette. Overview of Ares-I CFD ascent aerodynamic data development and analysis based on USM3D. In *49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, page 15, 2011.

[11] Edward T Curran. Scramjet engines: the first forty years. *Journal of Propulsion and Power*, 17(6):1138–1148, 2001.

[12] R.L. Chase and M.H. Tang. A history of the NASP program from the formation of the joint program office to the termination of the HySTP scramjet performance demonstration program. In *International Aerospace Planes and Hypersonics Technologies*, page 6031, 1995.

[13] P. A. Czysz, C. Bruno, and K. Kato. Interactions between propulsion systems and the configuration concepts defines the design space. In *10th International Space Planes and Hypersonic Systems and Technologies Conference*, 2001.

[14] L. McKinney, D.J. Farrell, T. Bogar, and J. Stemler. Investigations of TSTO propulsion system options. In *14th AIAA/AHI Space Planes and Hypersonic Systems and Technologies Conference*, 2006.

[15] J.L. Hunt. Airbreathing/rocket single-stage-to-orbit design matrix. In *6th International Aerospace Planes and Hypersonics Technologies*, Chattanooga, TN, 1995.

[16] J.L. Hunt and E.A. Eiswirth. NASA's dual-fuel airbreathing hypersonic vehicle study. In *Space Plane and Hypersonic Systems and Technology Conference*, 1996.

[17] J.R. Olds, L. Ledsinger, J. Bradford, A. Charania, D. McCormick, D.R. Komar, et al. Stargazer: A TSTO bantam-x vehicle concept utilizing rocket-based combined cycle propulsion. In *9th International Space Planes and Hypersonic Systems and Technologies Conference*, 1999.

[18] C. J. Trefny and J. M. Roche. Performance validation approach for the GTX air-breathing launch vehicle. In *Combustion, Airbreathing Propulsion, Propulsion Systems Hazards and Modelling and Simulation Subcommittee Joint Meeting*, April 2002.

[19] L.F. Scuderi, G.F. Orton, and J.L. Hunt. Mach 10 cruise/space access vehicle study. In *8th AIAA International Space Planes and Hypersonic Systems and Technologies Conference*, 1998.

[20] M.K. Smart, N.E. Hass, and A. Paull. Flight data analysis of the hyshot 2 scramjet flight experiment. *AIAA journal*, 44(10):2366–2375, 2006.

[21] M.K. Smart. Scramjets. *The Aeronautical Journal*, 111(1124):605–619, 2007.

[22] M.K.. Smart. How much compression should a scramjet inlet do? *AIAA Journal*, 50(3):610–619, March 2012.

[23] J.C. Turner and M.K.. Smart. Application of inlet injection to a three-dimensional scramjet at Mach 8. *AIAA journal*, 48(4):829–838, 2010.

[24] M.K.. Smart. Design of three-dimensional hypersonic inlets with rectangular-to-elliptical shape transition. *Journal of Propulsion and Power*, 15(3):408–416, May 1999.

[25] P.G. Ferlemann and R.J. Gollan. Parametric geometry, structured grid generation, and initial design study for rest-class hypersonic inlets. In *JANNAF 43rd Combustion, 31st Airbreathing Propulsion, 25th Propulsion Systems Hazards Joint Subcommittee Meeting*, La Jolla, CA, December 2009.

[26] P.L. Moses, V. L. Rausch, L. T. Nguyen, and J.R. Hill. NASA hypersonic flight demonstratorsoverview, status, and future plans. *Acta Astronautica*, 55(3):619–630, 2004.

[27] S. Walker, F. Rodgers, A. Paull, and D. M. Van Wie. Hycause flight test program. In *15th AIAA International Space Planes and Hypersonic Systems and Technologies Conference*, 2008.

[28] J.J. Spravka and T.R. Jorris. Current hypersonic and space vehicle flight test instrumentation challenges. In *AIAA Flight Testing Conference*, page 3224, 2015.

[29] S. Vitale, M. Pini, P. Colonna, G. Gori, A. A. Guardone, T.D. Economon, F. Palacios, and J.J. Alonso. Extension of the SU2 open source CFD code to the simulation of turbulent flows of fluids modelled with complex thermophysical laws. In *22nd AIAA Computational Fluid Dynamics Conference*, 2015.

[30] H.L. Kline, T.D. Economon, and J.J. Alonso. Multi-objective optimization of a hypersonic inlet using generalized outflow boundary conditions in the continuous adjoint method. In *54th AIAA Aerospace Sciences Meeting*, 2016.

[31] H.L. Kline, F. Palacios, T.D. Economon, and J.J. Alonso. Adjoint-based optimization of a hypersonic inlet. In *22nd AIAA Computational Fluid Dynamics Conference*, Dallas, TX, June 2015.

[32] H.L. Kline, F. Palacios, and J.J. Alonso. Sensitivity of the performance of a 3-dimensional hypersonic inlet to shape deformations. In *19th AIAA International Space Planes and Hypersonic Systems and Technologies Conference*, Atlanta, GA, June 2014.

[33] R.H. Ricketts, T.E. Noll, W. Whitlow Jr, and L.J. Huttsell. An overview of aeroelasticity studies for the National Aero-Space Plane. In *AIAA/ASME/ASCE/AHS/ASC 34th Structures, Structural Dynamics, and Materials Conference*, volume 1, pages 152–162, 1993.

[34] A. Kopp and N. Garbers. Investigation of structure, thermal protection system, and passenger stage integration for the hypersonic transport system spaceliner. In *19th AIAA International Space Planes and Hypersonic Systems and Technologies Conference*, page 2531, 2014.

[35] K.R. Jackson, M.R. Gruber, and T.F. Barhorst. The HIFiRE flight 2 experiment: an overview and status update. In *45th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, Denver, Colorado, August 2009.

[36] Rowan Gollan and Paul Ferlemann. Investigation of REST-class hypersonic inlet designs. *17th AIAA International Space Planes and Hypersonic Systems and Technologies Conference*, April 2011.

[37] A.R. Conn, K. Scheinberg, and L.N. Vicente. *Introduction to Derivative-Free Optimization*, volume 8. Siam, 2009.

[38] J.E. Dennis and V. Torczon. Derivative-free pattern search methods for multidisciplinary design problems. In *The Fifth AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, pages 922–932, 1994.

[39] R.B. Wilson. *A Simplicial Algorithm for Concave Programming*. PhD thesis, Harvard University, 1963.

[40] M.J.D. Powell. A fast algorithm for nonlinearly constrained optimization calculations. In *Numerical analysis*, pages 144–157. Springer, 1978.

[41] M.J.D. Powell. Algorithms for nonlinear constraints that use lagrangian functions. *Mathematical programming*, 14(1):224–248, 1978.

[42] S. Han. A globally convergent method for nonlinear programming. *Journal of Optimization Theory and Applications*, 22(3):297–309, 1977.

[43] P.T. Boggs and J.W. Tolle. Sequential quadratic programming. *Acta numerica*, 4:1–51, 1995.

[44] T.W. Lukaczyk, F. Palacios, and J.J. Alonso. Response surface methodologies for low-boom supersonic aircraft design using equivalent area distributions. In *12th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference and 14th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, pages 1–25, 2012.

[45] A.D. Wendorff, J.J. Alonso, and S.R. Bieniawski. A multi-fidelity approach to quantification of uncertainty in stability and control databases for use in stochastic aircraft simulations. In *16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, page 3439, 2015.

[46] S.K. Nadarajah and A. Jameson. A comparison of the continuous and discrete adjoint approach to automatic aerodynamic optimization. In *38th Aerospace Sciences Meeting and Exhibit*, 2000.

[47] C. A. Mader, J.R. R.A. Martins, J.J. Alonso, and E. van der Weide. ADjoint: An approach for the rapid development of discrete adjoint solvers. *AIAA Journal*, 46(4):863–873, April 2008.

[48] J.L. Lions. *Optimal Control of Systems Governed by Partial Differential Equations*. Springer-Verlag, New York, 1971.

[49] O. Pironneau. On optimum design in fluid mechanics. *Journal of Fluid Mechanics*, 64:97–110, 1974.

[50] O. Pironneau. *Optimal Shape Design for Elliptic Systems*. Springer-Verlag, New York, 1984.

[51] A. Jameson. Aerodynamic design via control theory. *Journal of Scientific Computing*, 3(3):233–260, 1988.

[52] J. Reuther, A. Jameson, J. Farmer, L. Martinelli, and D. Saunders. Aerodynamic shape optimization of complex aircraft configurations via an adjoint formulation. In *34th Aerospace Sciences Meeting and Exhibit*, 1996.

[53] A. Jameson and L. Martinelli. Aerodynamic shape optimization techniques based on control theory. In *Computational Mathematics Driven by Industrial Problems*, pages 151–221. Springer, 2000.

[54] M. B. Giles and N. A. Pierce. Adjoint equations in CFD : Duality , boundary conditions and solution behaviour. *AIAA*, 1997.

[55] C. Castro, C. Lozano, F. Palacios, and E. Zuazua. Systematic continuous adjoint approach to viscous aerodynamic design on unstructured grids. *AIAA Journal*, 45(9):2125–2139, September 2007.

[56] D.I. Papadimitriou and K.C. Giannakoglou. Total pressure loss minimization in turbomachinery cascades using a new continuous adjoint formulation. *Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy*, 221(6):865–872, 2007.

[57] C. Othmer. A continuous adjoint formulation for the computation of topological and surface sensitivities of ducted flows. *International Journal for Numerical Methods in Fluids*, 58(8):861–877, 2008.

[58] M. Hayashi, M. Ceze, and E. Volpe. Characteristics-based boundary bonditions for the Euler adjoint problem. *International Journal for Numerical Methods in Fluids*, 71(10):1297–1321, 2012.

[59] F. Palacios, M.R. Colonno, A.C. Aranake, A. Campos, S.R. Copeland, T.D. Economon, A.K. Lonkar, T.W. Lukaczyk, T.W.R Taylor, and J.J. Alonso. Stanford University Unstructured (SU2): An open-source integrated computational environment for multi-physics simulation and design. In *51st AIAA Aerospace Sciences Meeting and Exhibit.*, Grapevine, TX, January 2013.

[60] T.D. Economon, F. Palacios, and J.J. Alonso. Unsteady continuous adjoint approach for aerodynamic design on dynamic meshes. *AIAA Journal*, 53(9):2437–2453, 2015.

[61] S.R. Copeland, F. Palacios, and J.J.. Alonso. Adjoint-based aerothermodynamic shape design of hypersonic vehicles in non-equilibrium flows. In *52nd Aerospace Sciences Meeting*, 2014.

[62] E. Arian and M. D. Salas. Admitting the inadmissible: Adjoint formulation for incomplete cost functionals in aerodynamic optimization. *AIAA Journal*, 37(1):37–44, 1999.

[63] T.D. Economon, F. Palacios, S.R. Copeland, T. W. Lukaczyk, and J.J. Alonso. SU2 : An open-source suite for multiphysics simulation and design. *AIAA Journal*, 54(3), 2016.

[64] N. R. Gauger, A. Walther, C. Moldenhauer, and M. Widhalm. Automatic differentiation of an entire design chain for aerodynamic shape optimization. In *New Results in Numerical and Experimental Fluid Mechanics VI*, pages 454–461. Springer, 2007.

[65] B.Y. Zhou, T.A. Albring, N.R. Gauger, T.D. Economon, F. Palacios, and J.J. Alonso. A discrete adjoint framework for unsteady aerodynamic and aeroacoustic optimization. In *The 16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 2015.

[66] T. Albring, M. Sagebaum, and N. R. Gauger. Development of a consistent discrete adjoint solver in an evolving aerodynamic design framework. In *16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 2015.

[67] S.S. Altus, I.M. Kroo, and P.J. Gage. A genetic algorithm for scheduling and decomposition of multidisciplinary design problems. *Journal of mechanical design*, 118(4):486–489, 1996.

[68] R.W. Cubbison, D.F. Johnson, and E.T. Meleason. Performance characteristics from Mach 2.58 to 1.98 of an axisymmetric mixed-compression inlet system with 60-percent internal contraction. Technical Report NASA-TM-X-1739, NASA, 1969.

[69] G.L. Cole, M.J. Crosby, and G.H. Neiner. An automatic restart control system for an axisymmetric mixed-compression inlet. Technical Report NASA-TN-D-5590, NASA, Cleveland, Ohio, 1969.

[70] P.J. Waltrup and F.S. Billig. Structure of shock waves in cylindrical ducts. *AIAA journal*, 11(10):1404–1408, 1973.

[71] F.S. Billig, P.J. Waltrup, and R.D. Stockbridge. Integral-rocket dual-combustion ramjets: a new propulsion concept. *Journal of Spacecraft and Rockets*, 17(5):416–424, 1980.

[72] G.A. Sullins and G.H. McLafferty. Experimental results of shock trains in rectangular ducts. In *4th Symposium on Multidisciplinary Analysis and Optimization*, Cleveland, OH, 1992.

[73] R.H. Korkegi. Comparison of shock-induced two-and three-dimensional incipient turbulent seperation. *AIAA Journal*, 13(4):534–535, 1975.

[74] A. Kantrowitz and C. D. Donaldson. Preliminary investigation of supersonic diffusers. NACA Wartime Report L-713, NACA, National Advisory Committee for Aeronautics. Langley Aeronautical Lab.; Langley Field, VA, United States, May 1945.

[75] I. Jang, J.W. Nichols, and P. Moin. Bifurcation analysis of scramjet unstart. In *Center for Turbulence Research Annual Research Briefs*, pages 153–160. Citeseer, 2012.

[76] F. S. Billig, R.A. Baurle, C.J. Tam, and S. F. Wornom. Design and analysis of streamline traced hypersonic inlets. In *9th International Space Planes and Hypersonic Systems and Technologies Conference*, 1999.

[77] A. Busemann. Die achsensymmetrische kegelige überschallströmung. *Luftfahrtforschung*, 19(4):137–144, 1942.

[78] C.J. Tarn and R.A. Baurle. Inviscid CFD analysis of streamline traced hypersonic inlets at off-design conditions. In *39th Aerospace Sciences Meeting and Exhibit, Reno, NV*, 2001.

[79] L. Jacobsen, C. Tam, R. Behdadnia, and F. Billig. Starting and operation of a streamline-traced busemann inlet at Mach 4. In *42nd AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, page 4508, 2006.

[80] W.H. Heiser and D.T. Pratt. *Hypersonic Airbreathing Propulsion.* AIAA Education Series, AIAA, Washington, D.C., 1994.

[81] S.G. Nash and A. Sofer. *Linear and Nonlinear Programming.* McGraw-Hill Inc., New York, 1996.

[82] P.E. Gill, W. Murray, and M.A. Saunders. Users guide for SNOPT version 7: Software for large-scale nonlinear programming. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.217.5929, 2008.

[83] P. Wesseling. *Principles of Computational Fluid Dynamics.* Springer, 2001.

[84] C. Hirsch. *Numerical Computation of Internal and External Flows*, volume 2. John Wiley & Sons, 1990.

[85] W. Sutherland. The viscosity of gases and molecular force. *Philosophical Magazine*, 338(S. 5):507–531, 1893.

[86] P. Spalart and S. Allmaras. A one-equation turbulence model for aerodynamic flows. In *30th aerospace sciences meeting and exhibit*, 1992.

[87] F.R. Menter. Zonal two equation $k - \omega$, turbulence models for aerodynamic flows. In *23rd fluid dynamics, plasmadynamics, and lasers conference*, 1993.

[88] Q. Wang, K. Duraisamy, J.J. Alonso, and G. Iaccarino. Risk assessment of scramjet unstart using adjoint-based sampling methods. *AIAA journal*, 50(3):581–592, 2012.

[89] J. Sokolowski and J.P. Zolesio. *Introduction to Shape Optimization*. Springer Verlag, New York, 1991.

[90] A. Bueno-Orovio, C. Castro, F. Palacios, and E. Zuazua. Continuous adjoint approach for the Spalart–Allmaras model in aerodynamic optimization. *AIAA Journal*, 50(3), 2012.

[91] A.S. Zymaris, D.I. Papadimitriou, K.C. Giannakoglou, and Othmer C. Continuous adjoint approach to the spallart-allmaras turbulence model for incompressible flows. *Computers and Fluids*, 38:1528–1538, 2009.

[92] T.D. Economon. *Optimal Shape Design Using an Unsteady Continuous Adjoint Approach*. PhD thesis, Department of Aeronautics and Astronautics, Stanford University, 2014.

[93] A.H. Auslender. An application of distortion analysis to scramjet-combustor performance assessment. scramjet engine performance analysis, evaluation, and optimization. In *JANNAF Propulsion and Joint Subcommittee Meeting Scramjet Performance Workshop*, Albuquerque, New Mexico, 1996.

[94] R.A. Baurle and R.L. Gaffney. The art of extracting one-dimensional flow properties from multi-dimensional data sets. *AIAA*, pages 1–19, January 2007.

[95] R.A. Baurle and R.L. Gaffney. Extraction of one-dimensional flow properties from multidimensional data sets. *Journal of Propulsion and Power*, 24(4):704–714, 2008.

[96] M.C. Duta and M.D. Duta. Multi-objective turbomachinery optimization using a gradient-enhanced multi-layer perceptron. *International Journal for Numerical Methods in Fluids*, 61(6):591–605, 2009.

[97] C.H. Hinterberger and M. Olesen. Industrial application of continuous adjoint flow solvers for the optimization of automotive exhaust systems. *CFD & Optimization, Antalya, Turkey*, 2011.

[98] R. Sanchez, R. Palacios, T.D. Economon, H.L. Kline, J.J. Alonso, and F. Palacios. Towards a fluid-structure interaction solver for problems with large deformations within the open-source SU2 suite. In *57th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2016.

[99] A. Jameson, W. Schmidt, and E. Turkel. Numerical solution of the Euler equations by finite volume methods using Runge-Kutta time stepping schemes. In *14th fluid and plasma dynamics conference*, 1981.

[100] P.L. Roe. Approximate Riemann solvers, parameter vectors, and difference schemes. *Journal of Computational Physics*, 43:357–372, 1981.

[101] P. Lax and B. Wendroff. Systems of conservation laws. *Communications on Pure and Applied mathematics*, 13(2):217–237, 1960.

[102] R. MacCormack and A. Paullay. Computational efficiency achieved by time splitting of finite difference operators. In *10th Aerospace Sciences Meeting*, page 154, 1972.

[103] T.D. Economon, F. Palacios, J.J. Alonso, G. Bansal, D. Mudigere, A. Deshpande, A. Heinecke, and M. Smelyanskiy. Towards high-performance optimizations of the unstructured open-source SU2 suite. In *AIAA Infotech@ Aerospace*, 2015.

[104] J.L. Steger and R.F. Warming. Flux vector splitting of the inviscid gasdynamic equations with application to finite-difference methods. *Journal of computational physics*, 40(2):263–293, 1981.

[105] V. Venkatakrishnan. On the accuracy of limiters and convergence to steady state solutions. In *31st Aerospace Sciences Meeting*, 1993.

[106] A. Jameson, L. Martinelli, and F. Grasso. A multigrid method for the Navier-Stokes equations. In *24th Aerospace Sciences Meeting*, 1986.

[107] D.J. Mavriplis. Accurate multigrid solution of the Euler equations on unstructured and adaptive meshes. *AIAA Journal*, 28(2):213–221, February 1990.

[108] N.A. Pierce and M.B. Giles. Preconditioned multigrid methods for compressible flow calculations on stretched meshes. *J. Comput. Phys.*, 136:425–445, 1997.

[109] R. Courant, K. Friedrichs, and H. Lewy. uber die partiellen differenzengleichungen der mathematischen physik. *Mathematische Annalen*, 100(1):32–74, December 1928.

[110] A. Lerat and J. Sides. Implicit transonic calculations without artificial viscosity or upwinding. In *Numerical Simulation of Compressible Euler Flows*, pages 227–250. Springer, 1989.

[111] D.P. Raymer. *Aircraft Design: A Conceptual Approach and Rds-student, Software for Aircraft Design, Sizing, and Performance Set (AIAA Education)*. American Institute of Aeronautics & Astronautics, 2006.

[112] A.H. Shapiro. *The Dynamics and Thermodynamics of Compressible Fluid Flow*. John Wiley & Sons, New York, 1953.

[113] W. B. White, S. M. Johnson, and G. B. Dantzig. Chemical equilibrium in complex mixtures. *The Journal of Chemical Physics*, 28(5):751–755, 1958.

[114] J. A. Samareh. Aerodynamic shape optimization based on free-form deformation. In *10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 2004.

[115] G. Kenway, G. Kennedy, and J.R.R.A. Martins. A CAD-free approach to high-fidelity aerostructural optimization. In *13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference*. American Institute of Aeronautics and Astronautics, 2010.

[116] G. Anderson, M. Aftosmis, and M. Nemec. Parametric deformation of discrete geometry for aerodynamic shape design. In *50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*. American Institute of Aeronautics and Astronautics, 2012.

[117] R.E. Perez, P.W. Jansen, and J.R.R.A. Martins. pyOpt: A Python-based object-oriented framework for nonlinear constrained optimization. *Structures and Multidisciplinary Optimization*, 45(1):101–118, 2012.

[118] Python Software Foundation. Python 2.7.6 Release. `https://www.python.org/download/releases/2.7.6/`. Accessed: May 2015.

[119] G. Simeonides, W. Haase, and M. Manna. Experimental, analytical, and computational methods applied to hypersonic compression ramp flows. *AIAA journal*, 32(2):301–310, 1994.

[120] IS Kavvadias, EM Papoutsis-Kiachagias, and KC Giannakoglou. On the proper treatment of grid sensitivities in continuous adjoint methods for shape optimization. *Journal of Computational Physics*, 301:1–18, 2015.

[121] R.A. Baurle. VULCAN Home Page. `http://vulcan-cfd.larc.nasa.gov`. Accessed: July 2013.

[122] CNC precision machining. `http://www.stellarprecision.com/machining-capabilities/cnc-precision-machining.html`. Accessed: April 2017.

[123] C-H. Huang, J. Galuski, and C.L. Bloebaum. Multi-objective pareto concurrent subspace optimization for multidisciplinary design. *AIAA journal*, 45(8):1894–1906, 2007.

[124] R.L. Dotts, D.M. Curry, and D.J. Tillian. Orbiter thermal protection system. In *Space Shuttle Technical Conference, Part 2*, pages 1062–1081, 1985.

[125] A. De Boer, M.S. Van der Schoot, and H. Bijl. Mesh deformation based on radial basis function interpolation. *Computers & structures*, 85(11):784–795, 2007.

[126] A.J. Lofthouse, L. C. Scalabrin, and I. D. Boyd. Velocity slip and temperature jump in hypersonic aerothermodynamics. *Journal of Thermophysics and Heat Transfer*, 22(1):38–49, 2008.