Aerodynamic Design Optimization of a Supersonic Transport Aircraft Considering Low-Speed Stability

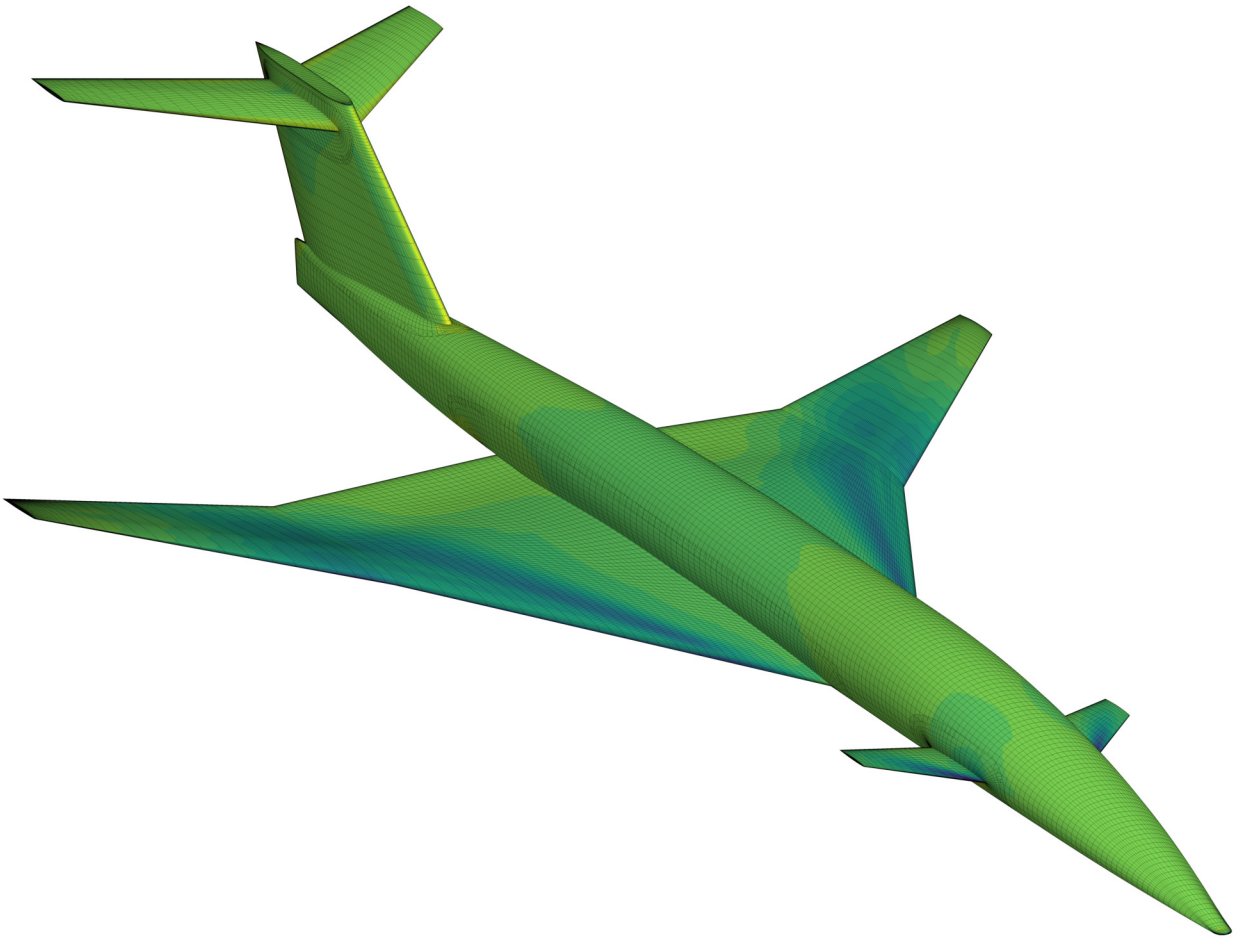
by

Sabet Seraj

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Aerospace Engineering)
in the University of Michigan
2024

Doctoral Committee:

Professor Joaquim R. R. A. Martins, Chair
Professor Carlos E. S. Cesnik
Professor Krzysztof J. Fidkowski
Professor Kevin J. Maki



Sabet Seraj

sseraj@umich.edu

ORCID iD: 0000-0002-7364-0071

Acknowledgments

There are many people I want to thank for helping me during my PhD. First, I thank my advisor, Prof. Joaquim Martins, for being an understanding mentor, running an excellent lab, and providing guidance and support that made this work possible. I thank my dissertation committee, Prof. Carlos Cesnik, Prof. Krzysztof Fidkowski, and Prof. Kevin Maki for their helpful feedback. I also thank Thiago Guimarães for his help with the SCALOS project. I am grateful to have been supported by NASA's Commercial Supersonic Technology project (grant number 80NSSC19K1661) and the Natural Sciences and Engineering Research Council of Canada (funding reference number PGSD3-545678-2020). Computational resources were provided by the NASA High-End Computing Program through the NASA Advanced Supercomputing Division at Ames Research Center.

I have been truly fortunate to be part of the MDO Lab. Working with such a tight-knit and supportive group of people was a highlight of my PhD experience. I want to thank: Anil Yildirim for providing indispensable advice on several aspects of this work, particularly meshing, geometry parameterization, and solvers; Yingqian Liao for many helpful discussions, especially when I was first getting started with MACH; Ella Wu for continuously improving the standards for code in the lab and for the insightful discussions; Marco Mangano and Shugo Kaneko for the camaraderie from when we started our PhDs together and through each step since; Josh Anibal for running a very productive journal club; Alex Kleb for the interesting conversations; Eirikur Jonsson for his long-standing efforts on code maintenance; Don Jones for his encouragement and for sharing his wealth of experience; Ben Brelje, Gustavo Halila, John Jasa, Nick Bons, Ping He, Shamsheer Chauhan, and Sicheng He for welcoming me into the lab and being great role models; Alex Coppeans, Ali Gray, Andrew Lamkin, Bernardo Pacini, David Anderegg, Eytan Adler, Galen Ng, Hannah Hajdik, Hugo Nicolas, Joaquin Exalto, Lucas Bonin, Malhar Prajapati, Marie Vaucher, Safa Bakhshi, Saja Abdul-Kaiyoom, Shubham Deshmukh, and Xiaosong Du for all the fun times.

I thank my friends in the Aerospace Engineering department, especially Caleb Van Beck, Elliot Kimmel, Miles McGruder, Paul Mokotoff, and Ral Bielawski. I also thank Ruthie Freeman for handling all my administrative requests.

I decided to pursue a PhD thanks in part to some exceptional professors I met during my undergraduate studies. I thank Prof. Stephanie Waterman for giving me my first opportunity to work in a research lab. I also thank Prof. Bernardo Galvão-Sousa and Prof. Masayuki Yano for teaching excellent classes and inspiring me to pursue research in numerical methods.

Finally, I thank my parents for being an unwavering source of support. Navigating the ups and downs of this journey was much easier knowing that you were always there for me.

Table of Contents

List of Figures

List of Tables

List of Appendices

Abstract

Designing supersonic transport (SST) aircraft requires accounting for performance and stability at high-speed and low-speed conditions. Previous work demonstrated that there is a trade-off between high-speed performance and low-speed stability. The objective of this dissertation is to use optimization to study design trends for SSTs, with a focus on low-speed stability.

I first assess the accuracy of Reynolds-averaged Navier–Stokes (RANS) and delayed detached eddy simulations (DDES) at predicting the vortex-dominated flows that are relevant for SSTs at subsonic conditions such as takeoff and landing. I compare the predicted aerodynamic coefficients with experimental data for a delta wing, which is a simplified representation of an SST wing. RANS accurately predicts vortex effects in the steady regime but is inaccurate at high angles of attack where the flow is unsteady. DDES is more reliable in the unsteady regime, but the computational cost is at least 100 times that of RANS. This motivates the RANS-based analysis and optimization in the remainder of the dissertation.

Next, I develop methods to improve the speed and accuracy of RANS solvers for the low and high Mach number flow regimes that are relevant for SST design. I propose a method for scaling the artificial dissipation terms in the Jameson–Schmidt–Turkel scheme to improve its accuracy at low Mach numbers while retaining the simplicity of the original scalar dissipation formulation. In addition, I introduce a dissipation-based continuation method for flows with shocks that improves robustness and accelerates convergence without sacrificing accuracy.

The second half of this dissertation focuses on aerodynamic design optimization of a full SST configuration. I perform RANS-based aerodynamic shape optimization to minimize drag at a supersonic cruise condition with and without a static margin constraint at a subsonic takeoff condition. The stable optimized designs use larger leading-edge flap deflections at the subsonic condition and have thicker wings. The increased wing thickness results in a 0.5% increase in supersonic drag for neutral stability and a 0.85% increase in supersonic drag for a 10% static margin. These results demonstrate that aerodynamic shape optimization is a valuable tool for designing SSTs accounting for supersonic performance and subsonic stability.

Finally, I investigate the impact of different trim surface configurations on SST design. I first use RANS-based optimization to compare the trim drag at a supersonic cruise condition for three-surface, canard, and conventional variants of an SST. The three-surface configuration has the lowest trim

drag at the supersonic condition. I then construct a supersonic buildup model to study the effects of variable trim surface sizing. When the trim surface spans are included as design variables, the design for minimum supersonic drag has no horizontal stabilizer and a canard sized at 39% of the wing span. However, the optimized canard configuration is unstable at subsonic conditions. This emphasizes the need to simultaneously consider subsonic stability and supersonic performance for SST design.

Chapter 1

Introduction

1.1 Supersonic transport aircraft

Air travel is the only mode of transportation that can move people long distances in a reasonable amount of time. Long-distance commercial aircraft typically fly at Mach numbers between 0.75 and 0.85 [1, Sec. 9.7]. The maximum speed of commercial aircraft is limited by a rapid drop in lift-to-drag ratio between Mach 0.85 and Mach 1. Lift-to-drag ratio is an important factor in the amount of fuel an aircraft uses during flight. The decrease in lift-to-drag ratio makes it uneconomical to fly at Mach numbers between 0.85 and 1. However, lift-to-drag ratio recovers somewhat at Mach numbers above 1 [1, Sec. 9.7]. This suggests that there may be an opportunity for air travel faster than currently available, albeit at a higher environmental and financial cost.

Supersonic transports (SSTs) are commercial aircraft that fly at Mach numbers greater than 1. To date, only two SSTs have flown commercial flights (Fig. 1.1): the Tupolev Tu-144 and the BAe-Aerospatiale Concorde [2, Ch. 1]. The Tu-144 only flew 55 commercial flights from 1977 to 1978. The Concorde had more success and flew commercial flights from 1976 until it was retired in 2003. By 2003, operating the Concorde had become unprofitable because of a combination of increased maintenance costs, increased fuel prices, and decreased ticket sales [3].

(a) Tupolev Tu-144

(b) BAe-Aerospatiale Concorde

Figure 1.1: The only two SSTs to have flown commercial flights [2, Ch. 1]

The Concorde flew at a cruise Mach number of 2. A recent market study found that designing an SST for lower supersonic speeds would result in more passengers per year and a better overall

business case [4]. Figure 1.2 has two plots from the market study showing contours of market capture (passengers per year) with varying Mach number and range. The best business cases are at the lowest Mach numbers, suggesting that SSTs could be profitable again with a different target market and aircraft design than the Concorde. There have been no commercial supersonic flights since the Concorde was retired. However, several startups have proposed new SST designs in recent years. Boom Supersonic's Overture concept in particular has received interest from airlines, including an order from United Airlines in 2021 [5].

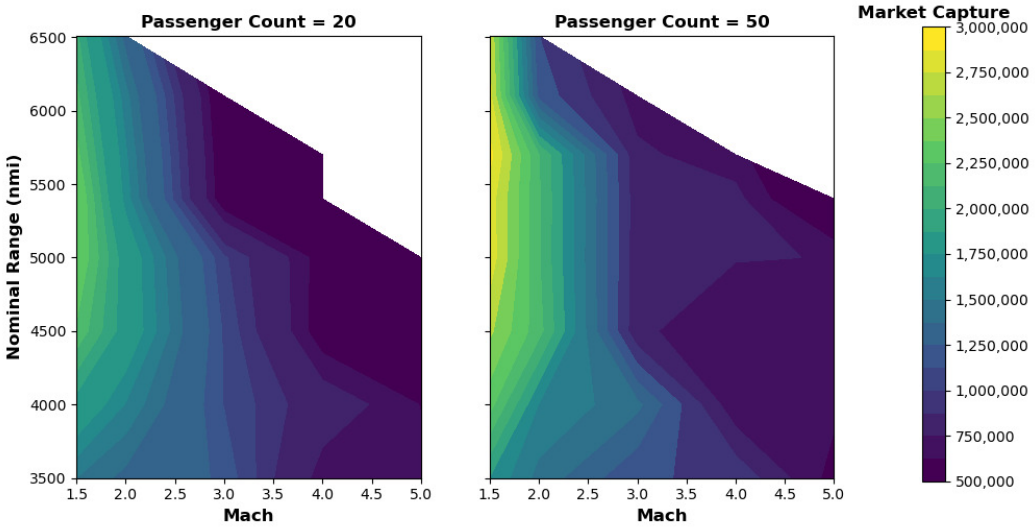


Figure 1.2: Market study showing better business cases at lower Mach numbers [4]

Recent research on SSTs has focused on reducing sonic boom noise, driven primarily by NASA's Commercial Supersonic Technologies project [6]. Reducing boom noise would enable SSTs to fly at supersonic speeds over ground, opening up flight routes and increasing their commercial viability. Advances in low-boom technology include sonic boom prediction [7–11] and optimization [12–14]. One report on low-boom SST design identified low-speed stability as a problem worth further investigation. [15, Sec. 2.3.1].

Some work on low-speed considerations for SST design was conducted during the High Speed Civil Transport (HSCT) era in the 1990s. Nelson [16] studied different SST wing planforms and their impact on supersonic performance and low-speed stability. Benoliel and Mason [17] studied wing planform effects on the subsonic pitch-up behavior of SSTs. Dudley et al. [18] performed aerodynamic and structural optimization of an SST considering low-speed constraints such as preventing engine scrape at landing. Crisafulli et al. [19] followed up on this work by incorporating a subsonic pitch-up model into the optimization framework.

Whereas design considerations for supersonic cruise have continued to be studied after the

HSCT era, there has not been as much work on low-speed design for SSTs. This dissertation is part of the Supersonic Configurations at Low Speeds (SCALOS) project. The SCALOS project focuses on studying the low-speed characteristics of SST configurations [20]. In addition to the work presented in this dissertation, SCALOS efforts include wind tunnel tests [21–24], static and dynamic water tunnel tests [25], numerical validation studies [26], aircraft control studies [27–30], aeroelastic-flight dynamics studies [31, 32], and a comprehensive design space survey [33].

1.2 Computational fluid dynamics for aircraft design

Computational fluid dynamics (CFD) involves using numerical methods to simulate fluid flow. CFD methods have proven to be reliable for predicting aircraft aerodynamics at cruise conditions, but simulating conditions near the edges of the flight envelope remains a challenge [34, Ch. 1]. This disparity was demonstrated in the results from the sixth Drag Prediction Workshop (DPW) [35]. For a wing-body configuration at a transonic cruise condition, the total drag predicted by several different CFD codes had a spread of less than 10 drag counts, with most results falling within the experimental error bounds. Conversely, the spread in the CFD results and error relative to experimental data increased as the angle of attack increased. Capturing the break in the pitching moment curve was particularly challenging. The inaccuracies were a result of separated flow due to transonic buffet at higher angles of attack.

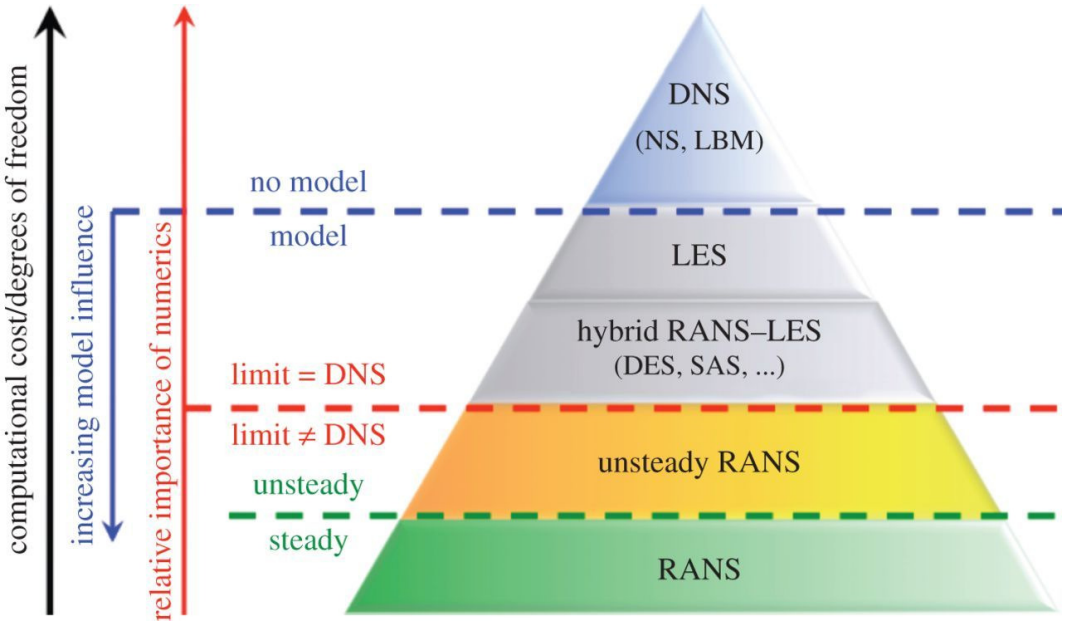


Figure 1.3: Hierarchy of CFD models [36]

Most of the DPW participants used steady Reynolds-averaged Navier–Stokes (RANS) codes.

Figure 1.3 presents a hierarchy of CFD models for turbulent flow simulations. These models can be roughly classified by how much turbulence is resolved compared to how much turbulence is modeled. Resolving turbulence involves using fine meshes and small time steps. RANS models resolve the least turbulence. Unsteady RANS (URANS) methods use the same turbulence models as steady RANS but with an unsteady time-derivative in the governing equations. The physical and mathematical justification for URANS remains in question [37]. Large-eddy simulations (LES) are unsteady methods that model small turbulent structures and resolve large turbulent structures. In between RANS and LES are hybrid RANS/LES methods. These methods typically use RANS models near the wall and LES models away from the wall [38]. Finally, direct numerical simulation (DNS) resolves the entire turbulent spectrum. DNS is currently computationally intractable for aircraft design [34, Ch. 8]. Going back to the transonic buffet case from DPW, one study following the workshop showed that wall-modeled LES captured the shape of the pitching moment curve [39]. RANS methods are currently widely used in the aircraft industry, but LES methods are quickly becoming a promising option for evaluating aircraft at off-design conditions [34, Ch. 1].

Another challenging off-design condition for aircraft is subsonic, high-angle-of-attack flow. These conditions are the focus of the High-Lift Prediction Workshop (HLPW) series. Similar to the transonic buffet case from the sixth DPW, RANS methods did not capture flow separation correctly for the cases from the third HLPW [40]. Two of the most promising methods in the workshop were lattice Boltzmann methods (LBM). Both of these codes, PowerFLOW [41] and XFlow [42], used LBM to perform LES. Similar to the more conventional Navier–Stokes LES methods, LBM can be used such that the larger turbulent features in the flow are resolved and only the subgrid scale is modeled. The LBM codes were among the best at matching experimental pressure distributions and force coefficients.

DPW and HLPW focus on transonic aircraft configurations. The second Cranked-Arrow Wing Aerodynamics Project, International (CAWAPI-2) [43] was a working group investigating problems similar to DPW and HLPW but for supersonic aircraft. Supersonic aircraft often have highly swept wings with sharp leading edges. At high angles of attack, this leads to unsteady, separated flow that is characterized by leading-edge vortices [44]. One of the CAWAPI-2 cases was a subsonic, high-angle-of-attack condition for the F-16XL aircraft. This condition had large vortical flow features, as shown in Fig. 1.4. The conclusions of the CAWAPI-2 results for this case were that steady and unsteady RANS models gave unreliable results, whereas hybrid RANS/LES models matched flight test data more consistently. RANS sometimes gave reasonable results for the pressure distribution in some parts of the flow, but the results were inaccurate in other parts of the flow. The hybrid RANS/LES methods represented the physics more accurately in most cases. This dissertation includes other validation studies comparing different CFD models for supersonic configurations at high angles of attack.

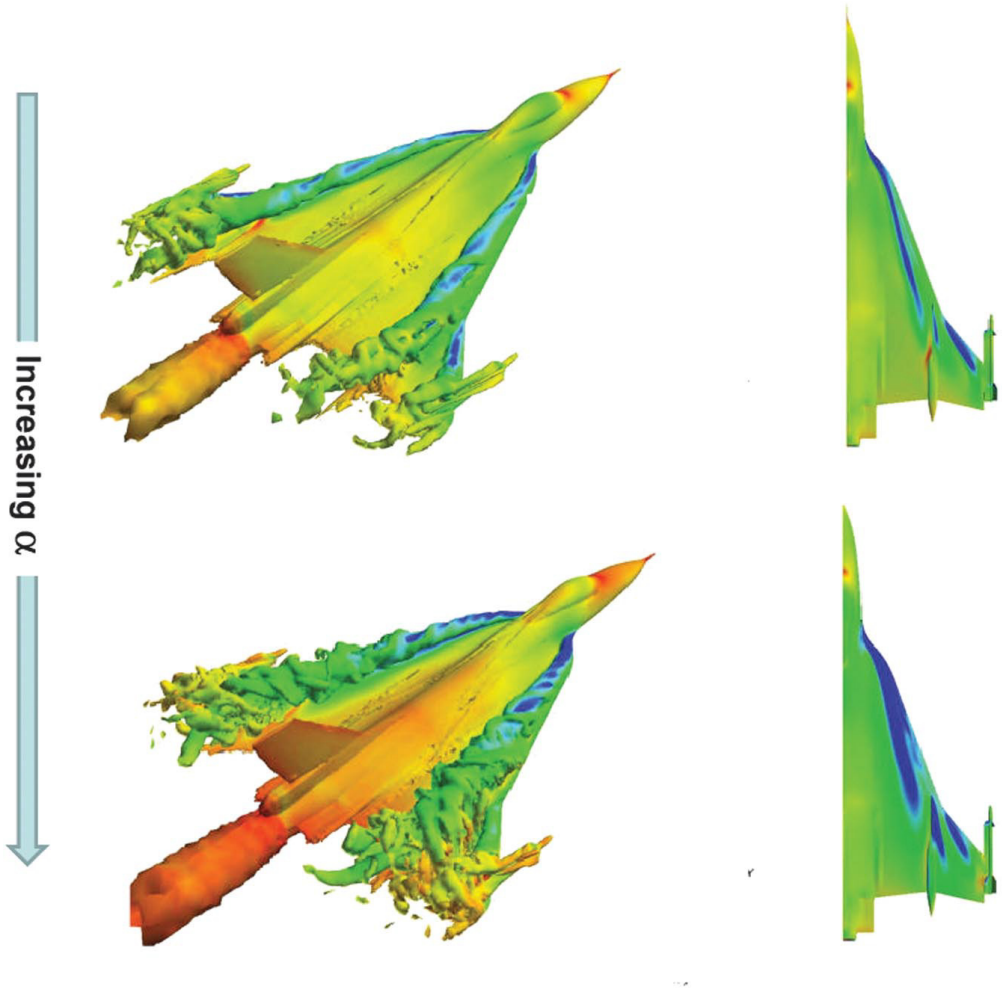


Figure 1.4: F-16XL at subsonic, high-angle-of-attack conditions [43]

1.3 Aerodynamic design optimization

Design optimization is a mathematically precise way of finding the best possible solution to a design problem [45, Ch. 1]. The optimization process involves finding the combination of design variables that minimizes an objective function while satisfying all the constraints. Design optimization can also be used as a tool to study design trends. Running multiple optimizations with different formulations is one way of capturing design trends and trade-offs in a precise manner.

Aerodynamic design optimization typically involves using a numerical model that evaluates aerodynamic functions of interest, such as lift and drag, to optimize a wing or full aircraft configuration. Optimizing the shape of a wing or aircraft is commonly referred to as aerodynamic shape optimization (ASO). ASO requires an aerodynamic model that accounts for shape changes, such as a potential flow panel method, Euler CFD, or RANS CFD. In addition, properly optimizing wing shape requires hundreds of design variables [46]. Gradient-based optimization is the only feasible way of solving such large ASO problems [47]. The adjoint method [48] is a way of computing gradients that scales with the number of outputs of interest instead of the number of design variables. ASO typically has many more design variables than outputs of interest, making the adjoint method an efficient approach.

Supersonic aircraft have been a frequent application for adjoint-based optimization. Reuther et al. [49] used the adjoint method to optimize an SST in an early example of full configuration shape optimization based on the Euler equations. The optimization problem involved minimizing drag at a supersonic cruise condition. Figure 1.5 shows the baseline and optimized designs for the SST configuration considered in that work. Cliff et al. [50] extended this shape optimization work to a multipoint context by considering supersonic and transonic flight conditions. Multipoint optimizations trade performance at individual flight conditions for better aggregate performance across all flight conditions considered in the optimization. This concept is particularly important for optimizing SSTs considering low-speed stability as done in this dissertation. The adjoint method was extended to a coupled aerostructural model by Martins et al. [51] and subsequently used to optimize a supersonic business jet [52]. Figure 1.6 shows the aerodynamic and structural results for the optimized supersonic business jet. The adjoint method has also been used for some of the low-boom shape optimizations mentioned in Sec. 1.1 [13], which involved coupling sonic boom propagation models with CFD. The adjoint method has been generalized to arbitrary coupled systems through the modular analysis and unified derivatives architecture [53], a version of which is implemented in the OpenMDAO framework [54]. Jasa et al. [55] used OpenMDAO to perform aero-thermal-propulsive-mission optimization of a supersonic aircraft, demonstrating the framework's usefulness for coupling several disciplines. The optimizations in this dissertation use either the adjoint-based MACH-Aero framework [56] or OpenMDAO.

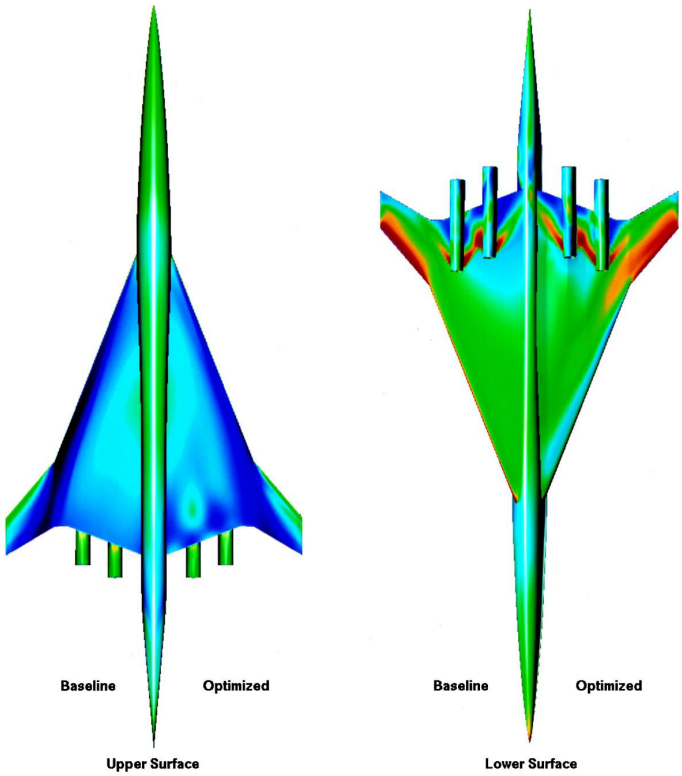


Figure 1.5: Pressure coefficient contours for the baseline and optimized SST configurations from Reuther et al. [49]

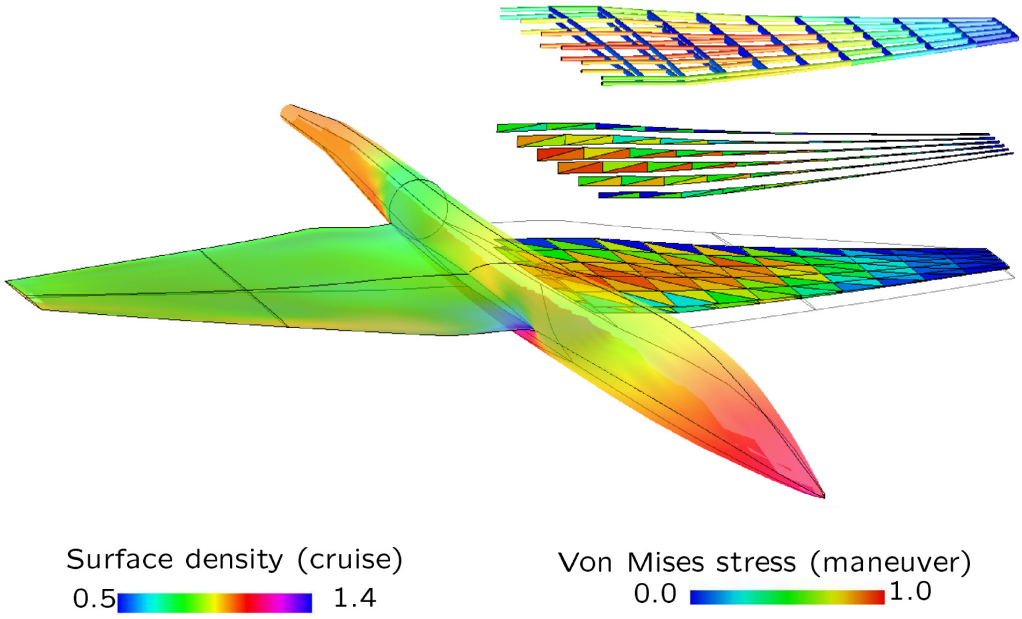


Figure 1.6: Aerostructural results for the optimized supersonic business jet from Martins et al. [52]

1.4 Outline

The objective of this dissertation is to use optimization to study design trends for supersonic transport aircraft, with a focus on low-speed stability. I rely on new and existing CFD and design optimization methods to accomplish this. The questions I aim to answer in this dissertation are:

1. What is the appropriate fidelity for low-speed, high-angle-of-attack CFD of supersonic configurations?
2. How can we develop CFD solvers that are accurate and efficient across a wide range of Mach numbers?
3. What is the effect of wing and fuselage shape on the low-speed stability of SSTs?
4. What is the supersonic drag penalty associated with enforcing subsonic stability for an SST?
5. What is the optimal trim surface layout for an SST?

The dissertation is organized as follows. In Chapter 2, I compare RANS and delayed detached eddy simulations to determine the appropriate fidelity for low-speed, high-angle-of-attack CFD of supersonic configurations. Chapter 3 presents new CFD methods for solving low and high Mach number flows. I develop a new low-speed preconditioner for the Jameson–Schmidt–Turkel scheme and a dissipation-based continuation method for flows with shocks. I then perform RANS-based shape optimizations of a supersonic transport aircraft in Chapter 4. I study designs optimized with and without a subsonic static margin constraint to evaluate the supersonic drag penalty associated with enforcing subsonic stability. In Chapter 5, I optimize the trim surface configuration for the supersonic transport aircraft using a nonlinear buildup model. The buildup model avoids the mesh-related limitations of the RANS-based framework. Finally, I summarize the key results, present my contributions, and provide recommendations for future work in Chapter 6.

Chapter 2

Predicting Low-Speed, High-Angle-of-Attack Characteristics

Understanding the fidelity level required to accurately predict performance is an essential first step to aerodynamic design. This chapter compares different CFD methods for predicting aerodynamic force and moment coefficients for a delta wing, which is a simplified representation of a supersonic transport wing. We focus on low-speed, high-angle-of-attack conditions, which are the most difficult conditions to accurately predict for supersonic aircraft design.

2.1 Introduction

Supersonic transport (SST) aircraft commonly use thin, highly swept wings to reduce supersonic wave drag. This design decision also influences the aircraft's low-speed aerodynamic characteristics, particularly during takeoff and landing when flying at high angles of attack. The flow over highly swept wings at high angles of attack is unsteady, separated, and characterized by leading-edge vortices [44]. Leading-edge vortices can be advantageous in some cases. The Concorde relied on leading-edge vortices instead of high-lift devices to generate lift during subsonic flight [2]. On the other hand, these vortices can contribute to unstable pitch-up behavior at moderate to high angles of attack [16], conditions that fall within the aircraft's flight envelope. Accurately predicting high-angle-of-attack aerodynamics is essential to designing the next generation of SSTs.

Historically, numerical methods for the low-speed analysis of supersonic aircraft have involved vortex lattice methods (VLMs) with empirical corrections for vortex lift, vortex breakdown, or other nonlinearities [17, 57, 58]. Some of the corrections were based on prior theoretical models [59–61], whereas others were developed specifically for the method. The shortcoming of these methods is that their predictive capability is limited. This can be a result of the experimental data used to construct empirical models, the selection of model parameters, or the generality of the model itself. For example, Lan and Hsu [57] constructed a vortex breakdown model using a least-squares fit of delta

wing experimental data. The approximate nature of this fit caused the vortex breakdown behavior of a 70 deg delta wing to be incorrectly modeled as a cross between 60 and 80 deg wings. Their method was also restricted to slender wings because of limitations in the leading edge suction analogy [59] used in the method. The leading-edge suction analogy has since been extended to nonslender delta wings [62]. Another example is the pitch-up estimation model developed by Benoliel and Mason [17]. Whereas the model accurately predicted pitch-up caused by outboard separation, it was less reliable for vortex-dominated pitch-up cases. In addition, the model required the user to specify maximum sectional lift coefficients before the analysis, limiting the model's usefulness for design exploration.

VLM-type methods remain an essential tool for rapid design analysis [27]. However, more recent work has also turned to high fidelity computational fluid dynamics (CFD) to resolve the underlying physics rather than model them. Numerous papers have dealt with CFD for delta wings or similar supersonic wing geometries. We focus on studies that compare unsteady turbulent simulations with experimental data because these are the most relevant for our purposes. The first three papers we mention study a 70 deg delta wing at an angle of attack of 27 deg. This is a popular test case because of the full field experimental data collected by Mitchell et al. [63]. Soemarwoto and Boelens [64] used unsteady Reynolds-averaged Navier–Stokes (unsteady RANS or URANS) to study this case. They found that accuracy was hindered by a lack of mesh density and the inability of URANS to model short time-scale unsteadiness. They suggested that large eddy simulation (LES) or a hybrid RANS/LES method could provide more accurate results. Görtz [65] used a hybrid RANS/LES method called detached eddy simulation (DES) for the same case. They found that sufficiently refined results were accurate but that the unsteady vortex effects were sensitive to the mesh density and the time step. Morton [66] also used DES and found the solution to be mesh sensitive. They showed that adaptive mesh refinement could be used to reduce the computational cost. François et al. [67] considered a 65 deg delta wing at an angle of attack of 8 deg. They found that URANS was accurate near the wing, but hybrid RANS/LES was necessary to predict the downstream evolution of the leading-edge vortex.

There has been some CFD work for realistic supersonic configurations at high angles of attack and flight Reynolds numbers. The F-16XL aircraft was the subject of the Cranked-Arrow Wing Aerodynamics Project, International (CAWAPI) [43]. CAWAPI researchers compared high-fidelity CFD predictions to F-16XL flight test data, including one subsonic case at an angle of attack of 19.8 deg. We highlight two studies on this flight condition. The first by Lofthouse and Cummings [68] showed that delayed detached eddy simulation (DDES) combined with large meshes could accurately match the flight test data. The second study by Tomac et al. [69] compared the results from RANS, URANS, and hybrid RANS/LES models. They found that steady RANS matched the experimental pressure distributions at parts of the wing where the flow was steady. The unsteady models matched

the flight test data more consistently. They noted that RANS was reasonably accurate because the flight condition exhibited only "moderate unsteady aerodynamics". In separate work, Forsythe et al. [70] considered the F-15E aircraft at an angle of attack of 65 deg. The flow regime, in this case, is unlike any of the studies described above. The angle of attack is well past stall, and leading-edge vortex effects are diminished. The authors found that RANS and DES predictions were both within 10% of flight test data, with DES being slightly more accurate. They also reported that the forces were not particularly sensitive to mesh refinement.

All the studies mentioned above deal with a single angle of attack. There are far fewer studies at multiple angles of attack, possibly because of the increased computational cost or a lack of experimental data. One such study by Cummings and Schütte [71] used RANS and hybrid RANS/LES models to simulate the flow over a 65 deg delta wing at angles of attack of 13.3, 18.4, and 23 deg. However, the flow was primarily steady even at the highest angle of attack. As a result, they found that RANS was similarly accurate to the best hybrid RANS/LES model. Another study by Jeans et al. [72] found that DDES provides accurate force and moment values for a generic fighter configuration at angles of attack between 15 and 40 deg. However, they did not show any solutions between 30 and 40 deg where the breaks in the force and moment coefficients occur. It is unclear whether DDES captures this trend correctly.

The key contributions of this chapter are as follows: 1) evaluating the accuracy of RANS and DDES models at predicting the low-speed aerodynamics of a delta wing over multiple angles of attack, including angles up to and past stall; and 2) formulating a metric to quantify how steady the flow is at a given angle of attack. We do not consider URANS because of its shortcomings in resolving unsteady turbulent effects [64, 67], which are crucial at high angles of attack. We are particularly interested in capturing the trends in the force and moment coefficients at high angles of attack because these are important for the low-speed stability of SSTs. This is similar to the approach taken by the High Lift Prediction Workshop [40] for conventional transonic wings.

We choose to work with a delta wing for three reasons. First, delta wings are often the starting point for SST wing designs. Second, the geometry is relatively simple to generate and mesh. Lastly, experimental force and moment coefficients across a range of angles of attack are available for our chosen geometry. We further discuss the geometry and flow conditions in Sec. 2.2. We then present RANS and DDES results in Sec. 2.3 and Sec. 2.4, respectively.

2.2 Delta wing geometry and flow conditions

The geometry and flow conditions are based on the experimental configuration of Jarrah and Ashley [73], who tested delta wings with aspect ratios of 1, 1.5, and 2. We choose to work with the aspect ratio 2 delta wing. An aspect ratio of 2 corresponds to a leading-edge sweep of 63.4 deg. This is

representative of a typical SST wing sweep. For comparison, the Concorde had an average leading edge sweep of 60 deg [2].

Jarrah [74] provides a complete description of the experimental setup. We summarize some important points here. The wind tunnel had a turbulence intensity near 1%, and the test section was divided into channels with plexiglass to minimize interference effects from actuators. The delta wing flow channel was 690 mm wide. The aspect-ratio-2 wing's span was about 44% of the channel width. The dynamic pressure was corrected for both solid blockage and wake blockage. The angle-of-attack measurements were corrected for static and elastic deformations of the sting, yoke, and balance. It is unclear whether the aluminum delta wing model was stiff enough to preclude aeroelastic deflections.

The delta wing (Fig. 2.1) is a flat plate with a sharp leading edge. The trailing edge has finite thickness with sharp edges on the upper and lower surfaces. The leading edge is cut at an angle of 30 deg from the upper surface in the spanwise plane. The root chord length is 306 mm, and the thickness is 6.4 mm. We give the geometry a 0.5 mm radius of curvature at all edges to avoid mesh quality issues at infinitely sharp corners. The radius of curvature is defined normal to each edge.

The Reynolds number based on the root chord length is 590,000. We assume standard sea-level conditions for the rest of the flow properties because they are not specified in the experimental paper. The Mach number under this assumption is 0.083. We compare RANS and DDES simulations against the experimental lift, drag, and pitching moment coefficients at angles of attack from 0 to 40 deg at 5 deg increments. This range of angle of attack includes the point of maximum lift and the break in the lift and moment curves that follow immediately afterward. Table 2.1 lists the reference quantities and flow conditions. The reference chord is only used in nondimensionalizing the pitching moment. The listed reference area is for the entire wing. We use a half-span wing with a symmetry plane at the root for all simulations. All simulations are fully turbulent. At Reynolds numbers on the order of 10^5–10^6, laminar-to-turbulent transition has been shown to affect the secondary separation for delta wings with sharp leading edges [66, 75]. Approaches to modeling transition for delta wings have included prescribing the transition point [66] and using implicit LES [76], but these are beyond the scope of the current work.

The reference point for the pitching moment warrants further discussion because it is most likely misreported in the experimental paper. In the work of Jarrah and Ashley [73], the pitching moment is defined as positive nose-up about a point 77% of the root chord aft of the apex. We instead take the correct definition to be positive nose-down about a point 23% of the root chord in front of the apex. We justify this decision in Appendix A.

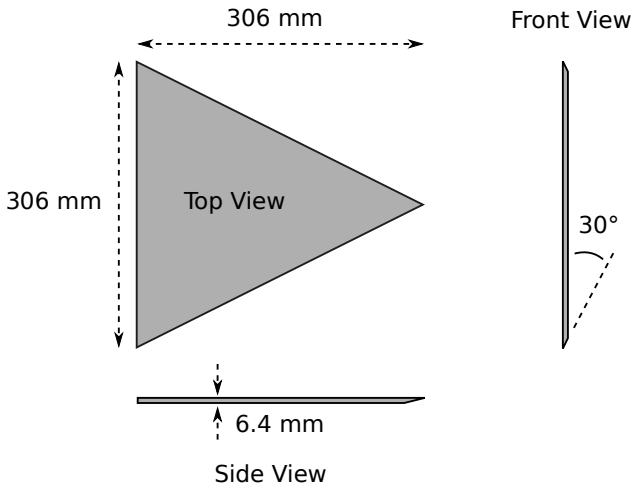


Figure 2.1: Delta wing geometry based on the experimental configuration of Jarrah and Ashley [73]

Table 2.1: Reference quantities and flow conditions

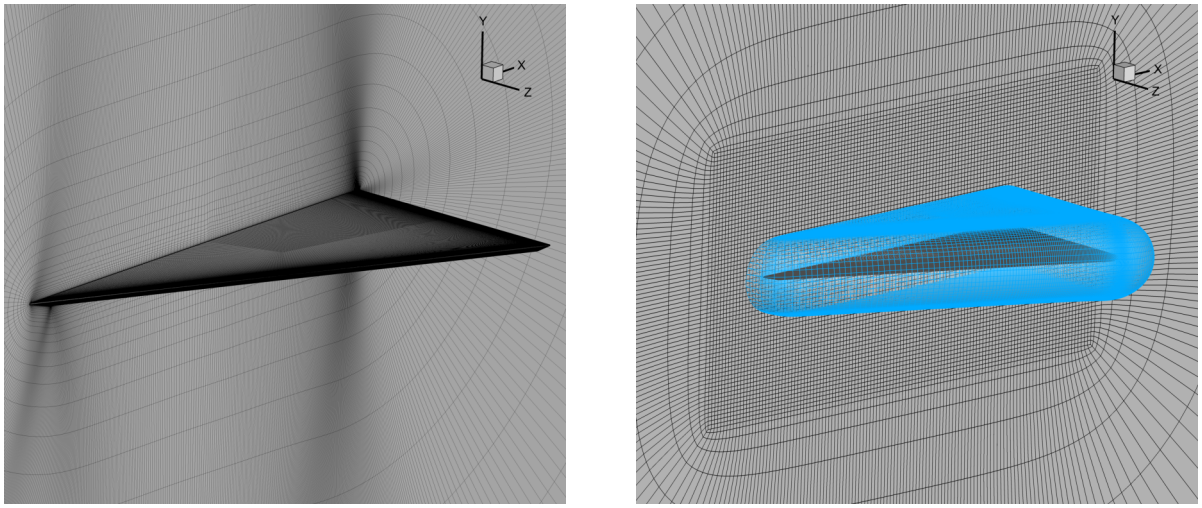
Quantity	Value
Root chord (L_{ref})	0.306 m
Reference chord	0.204 m
Reference area	0.046818 m^2
Freestream velocity (U_∞)	28.16 m/s
Reynolds number	590,000
Mach number	0.083

2.3 RANS simulations using ADflow

ADflow [77] is a second-order finite volume flow solver for multiblock and overset structured meshes. We use ADflow (version 2.2.1) to solve the steady, compressible RANS equations with the Spalart–Allmaras (SA) turbulence model [78]. The SA implementation is similar to the "standard" SA model described in the NASA Turbulence Modeling Resource (https://turbmodels.larc.nasa.gov/spalart.html), except that the production term is computed using strain instead of vorticity. We use the Jameson–Schmidt–Turkel scheme with scalar dissipation [79] for inviscid fluxes and central differencing for viscous fluxes. All solutions are converged to a total residual norm of 10^{-8} relative to the freestream residual. Using the approximate Newton–Krylov startup strategy [80] in ADflow allows us to converge highly separated flows to a steady solution efficiently. Although we expect RANS to be inaccurate at high angles of attack where unsteady effects are important, we use ADflow to determine the regime where RANS results are valid and how large the errors are outside this regime.

2.3.1 Mesh convergence study

We use two different mesh types with ADflow. The first is a multiblock mesh (Fig. 2.2a). We generate the mesh in two steps. First, we create a multiblock surface mesh manually using Ansys ICEM CFD. We then extrude the surface mesh to generate a volume mesh using pyHyp [81], a hyperbolic mesh generation code based on the method proposed by Chan and Steger [82]. The second mesh type is an overset mesh (Fig. 2.2b). We march the same multiblock surface mesh out a small distance (highlighted in blue in the figure) and embed this mesh in a background mesh. The background mesh has a Cartesian refinement region near the wing and is surrounded by a hyperbolic O-mesh. Using an overset mesh allows us to take advantage of the high-quality boundary-layer cells produced by hyperbolic meshing and control the refinement in the separated flow region using Cartesian meshing. For all meshes, the farfield boundary is 20 times the root chord length away from the delta wing surface.



(a) Multiblock mesh (M1)

(b) Overset mesh (C1)

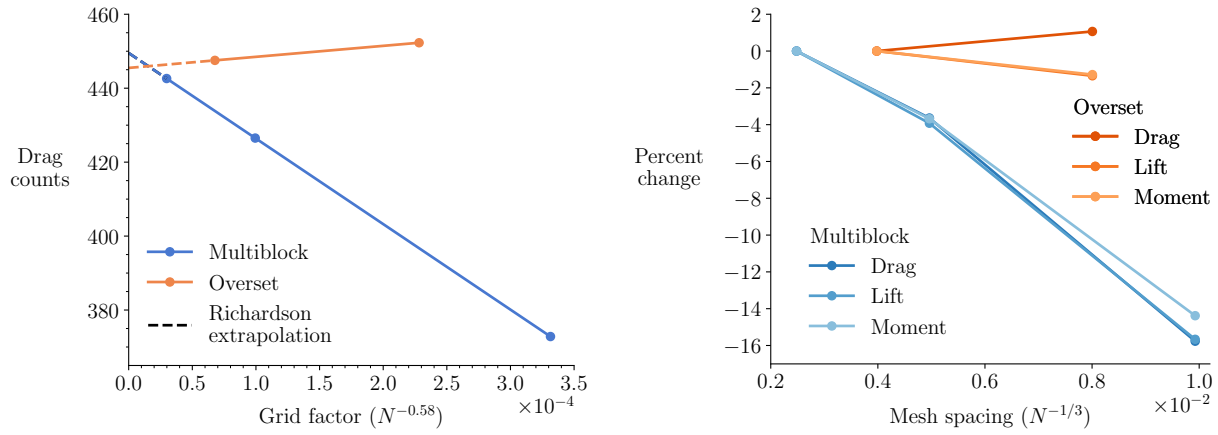
Figure 2.2: RANS meshes

We generate two mesh families by uniformly coarsening the finest mesh, twice for the multiblock mesh and once for the overset mesh. Table 2.2 lists the characteristics for each mesh. The number of cells for the overset meshes is the number of "compute cells" after implicit hole cutting [83]. We use these meshes to run a mesh convergence study. We choose an angle of attack of 5 deg for the study because we expect RANS to resolve the physics at this condition and consequently converge to a realistic solution as the mesh is refined. Figure 2.3 shows the results of the study. We also report the maximum y^+ from the mesh convergence solutions in Table 2.2. The extrapolated drag values (Fig. 2.3a) differ by 4 counts, suggesting that the multiblock and overset meshes will converge to similar solutions as the mesh is refined. The order of accuracy used in the extrapolation is 1.74,

which is computed from the three multiblock solutions. We use the multiblock order of accuracy to extrapolate the overset mesh solutions because we cannot compute order of accuracy from only two solutions. The overset meshes provide closer to mesh-converged results with fewer cells than the multiblock meshes. This is because the overset mesh has more effective local refinement, particularly in the off-wall direction. The C1 coefficients differ from the C0 coefficients by less than 2%, and the M1 coefficients differ from the M0 coefficients by less than 4% (Fig. 2.3b). Based on this, we use the C1 and M1 meshes for the angle-of-attack sweep in Sec. 2.3.2. Although the M1 mesh provides results that are further from mesh-converged than the C1 mesh, we use it to compare the differences between the multiblock and overset topologies across different angles of attack.

Table 2.2: RANS mesh characteristics

Mesh type	Label	Cells (N)	$y^+_{\max}$
Multiblock	M0	65,458,176	0.62
	M1	8,182,272	1.29
	M2	1,022,784	2.56
Overset	C0	15,867,929	1.22
	C1	1,948,651	1.88



(a) Extrapolated drag

(b) Coefficient convergence

Figure 2.3: RANS mesh convergence at $\alpha = 5°$

2.3.2 Angle-of-attack sweep

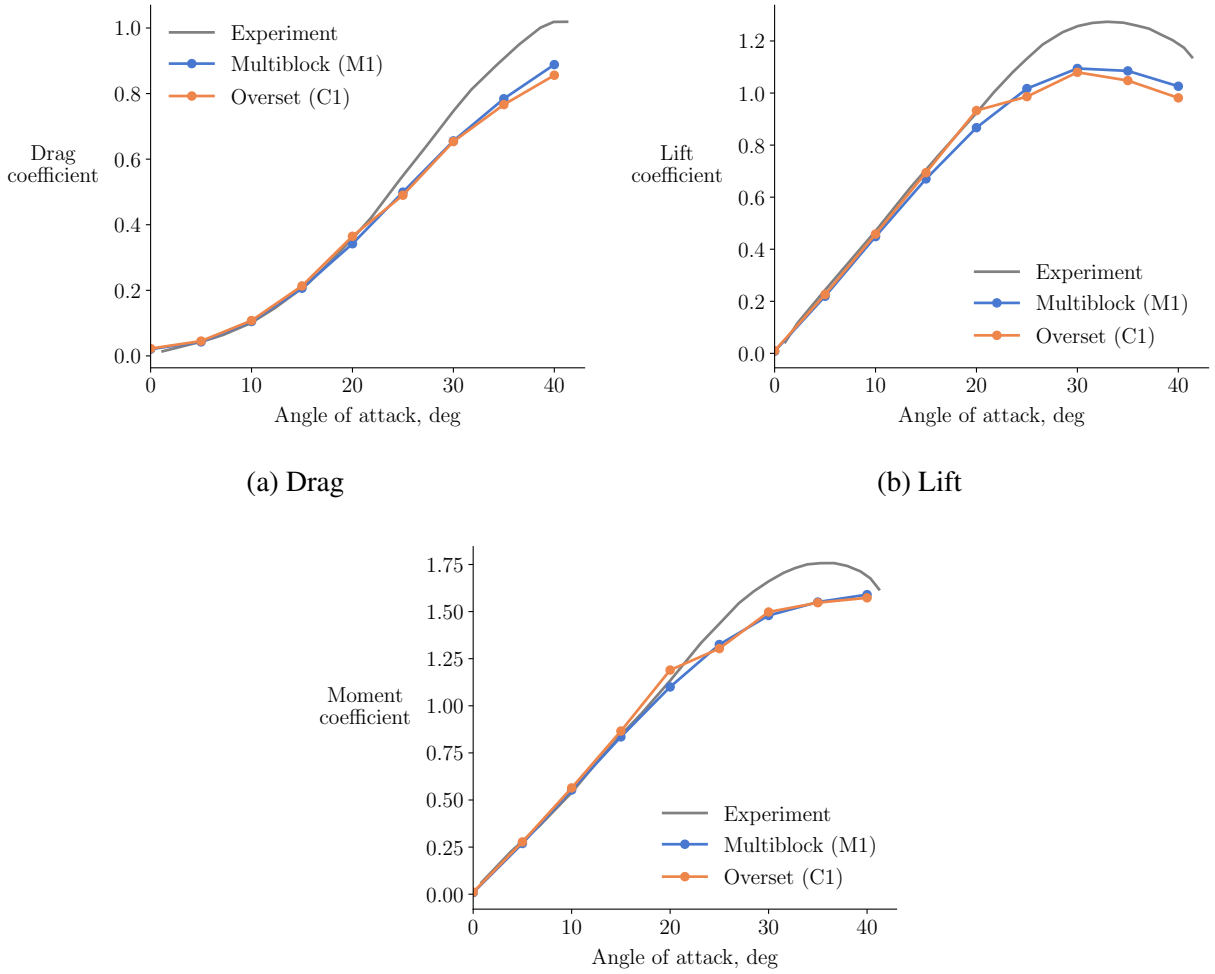
Figure 2.4 shows RANS results for angles of attack from 0 to 40 deg using the C1 and M1 meshes. The solution for each angle of attack starts from freestream conditions. The aerodynamic coefficients match the experimental data well up to 20 deg. We show in Sec. 2.4.3 that this corresponds to the steady regime. The main advantage of using the overset mesh is better lift prediction at 15 and 20 deg. However, the pitching moment at 20 deg is less accurate than the multiblock solution. The accuracy of RANS drops off sharply once the flow is unsteady. The lift and drag are underpredicted by 10–20% because RANS does not resolve the forces generated by unsteady vortex effects. In addition, the break in the pitching moment is not captured.

To visualize the RANS results in the steady regime, we plot the upper surface pressure coefficient and streamlines in Fig. 2.5. The leading-edge vortex strength increases with the angle of attack. In addition, the flow moves increasingly in the spanwise direction as the angle of attack increases. This corresponds to the primary reattachment line moving closer to the wing root, indicating that the leading-edge vortex is becoming larger. RANS can resolve these large-scale vortices if the flow is steady. This was also discussed by Cummings and Schütte [71].

2.4 DDES using pimpleFoam

For DDES, we use pimpleFoam, a time-accurate, incompressible finite volume flow solver that is part of the OpenFOAM toolbox [84]. The version of OpenFOAM we use is OpenFOAM 7, released by the OpenFOAM Foundation. An incompressible solver is a reasonable choice because we expect the low freestream Mach number to result in minimal compressibility effects. We use the SA-DDES formulation [85], a hybrid RANS/LES model that uses the Spalart–Allmaras turbulence model near the wall and a subgrid scale model away from the wall. We use the second-order backward difference in time, the Beam–Warming scheme [86] for inviscid fluxes, and central differencing for viscous fluxes. The resulting discretization is second-order accurate in time and space.

The PIMPLE algorithm used by pimpleFoam is an extension of the Pressure-Implicit with Splitting of Operators (PISO) algorithm [87]. The most widely used version of the PISO algorithm solves the discretized equations at each time step by performing a velocity predictor step followed by two pressure corrector steps. The PIMPLE algorithm loops over this entire iteration process multiple times at each time step. This enables the use of larger time steps. Each iteration of the outermost loop is termed an outer corrector iteration. Unless otherwise stated, we use one outer corrector iteration, which is equivalent to the PISO algorithm. However, using pimpleFoam lets us test for outer loop convergence, which we discuss in Sec. 2.4.1.

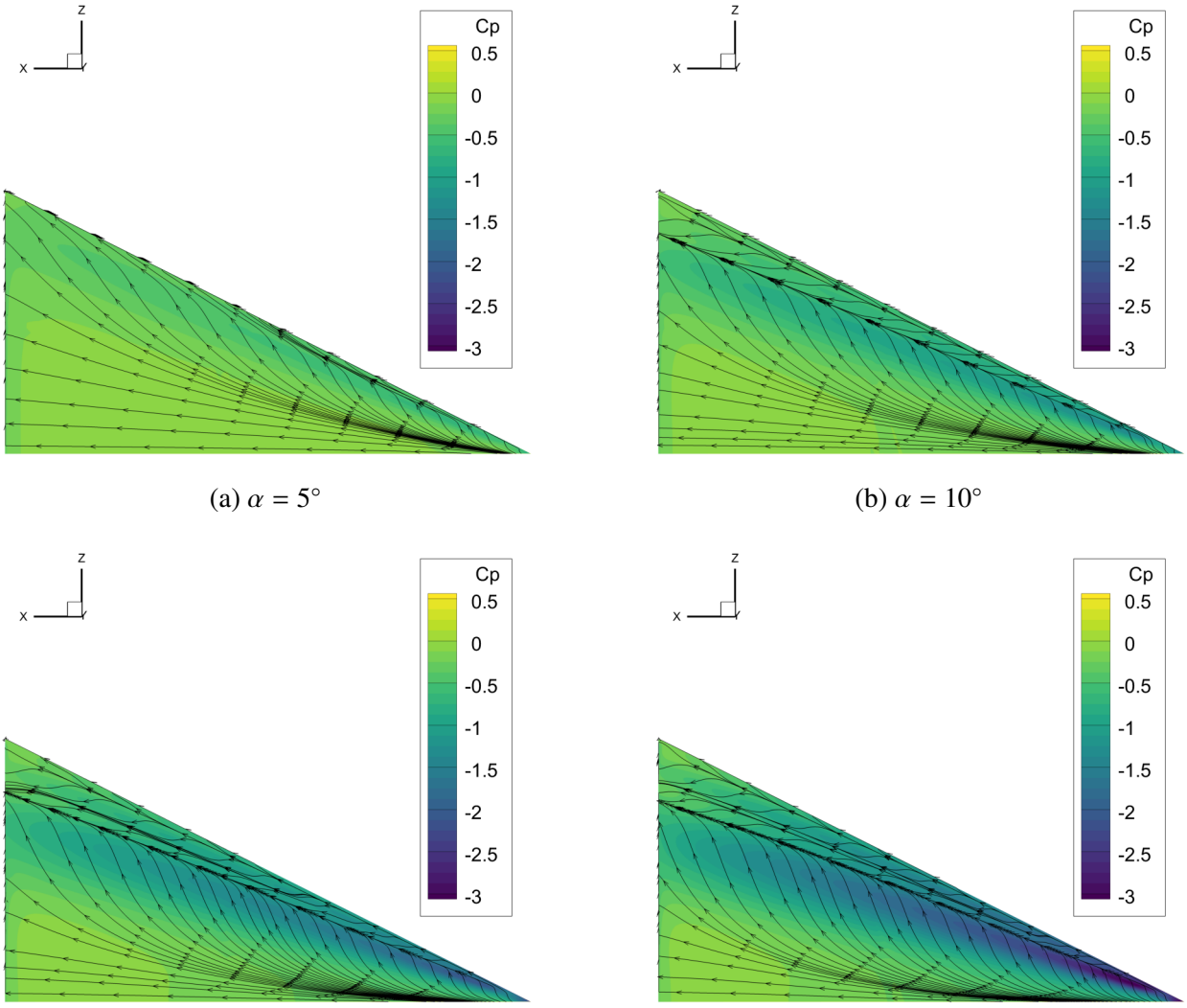


(a) Drag

(b) Lift

(c) Pitching moment

Figure 2.4: Comparison of RANS force and moment coefficients with experimental data from Jarrah and Ashley [73]



(a) $\alpha = 5°$

(b) $\alpha = 10°$

(c) $\alpha = 15°$

(d) $\alpha = 20°$

Figure 2.5: RANS (C1) upper surface pressure coefficient contours and streamlines

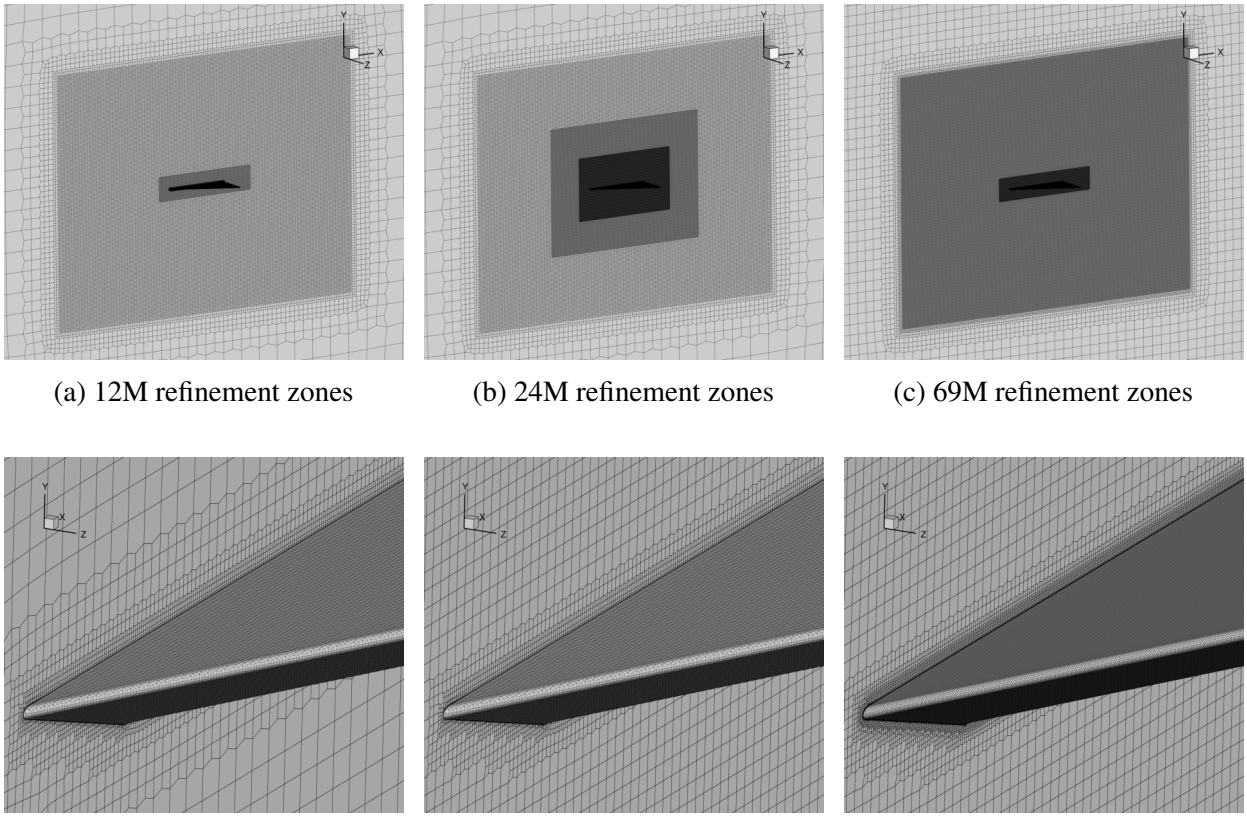
2.4.1 Convergence studies

Assessing what constitutes a converged solution is more involved for DDES than for RANS. For DDES, we must consider both spatial and temporal discretization errors. We also look at pimpleFoam's iterative convergence. We use an angle of attack of 25 deg for all convergence studies in this section for two reasons. First, we expect DDES to be accurate and convergent at this condition. Second, based on the flow regimes described by Hummel [88] for a 65 deg delta wing, we expect an angle of attack of 25 deg to be high enough to see unsteadiness caused by vortex breakdown. This will provide convergence information that is applicable to unsteady cases.

Figure 2.6 shows the unstructured meshes that we use with pimpleFoam. We use the cfMesh library to generate Cartesian meshes with nested refinement zones. The meshes also have off-wall layers extruded from the wing surface to resolve the boundary layer. Table 2.3 lists the cell counts for each mesh, and the maximum y^+ values from the mesh convergence study discussed later in this section. The 69M mesh is a nearly uniform refinement of the 12M mesh, including surface refinement. However, the number of off-wall layers is the same for both meshes, which is why the 69M mesh has only 5.6 times the number of cells rather than 8 times. The 24M mesh adds local refinement to the 12M mesh in the separated flow region near the wing as a compromise between the 12M and 69M meshes. The farfield boundary for all meshes is 20 times the root chord length away from the delta wing surface.

We start by running a time-step refinement study with the coarsest (12M) mesh. Ashton et al. [89] suggest that choosing the time step such that the Courant number is less than one in the LES regions provides suitable accuracy for DDES. This equates to a time step of 1.4×10^{-4} s for the 12M mesh. Based on this rule of thumb, we conduct a time step refinement study with time steps ranging from 2×10^{-4} s to 0.25×10^{-4} s. The force and moment coefficients change less than 1% when refining the time step from 0.5×10^{-4} s to 0.25×10^{-4} s (Fig. 2.7a). Based on this, we use a time step of 0.5×10^{-4} s for all subsequent results. Running a mesh convergence study shows that the coefficients change less than 1% from the 24M mesh to the 69M mesh (Fig. 2.7b). This suggests that the 24M mesh should be sufficiently fine for angles of attack up to at least 25 deg.

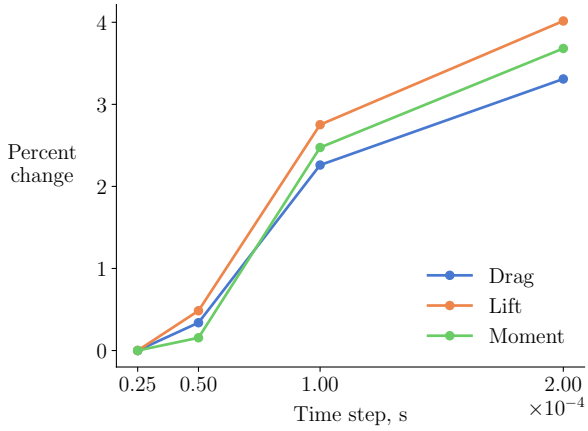
Lastly, we check for convergence with the number of outer corrector iterations. Because we are not enforcing a residual tolerance at each time step, it is crucial to verify that the solution is independent of the iteration settings. The computational cost scales linearly with the number of outer corrector iterations. As a result, we use the 12M mesh to reduce the computational cost of this study. The number of outer corrector iterations has a minimal effect on the coefficients (Fig. 2.7c). This justifies using only one iteration. We expect similar levels of convergence for the more refined meshes because the outer loop convergence is mainly dependent on the Courant number. With a time step of 0.5×10^{-4} s, the Courant number in the LES region is less than one for all meshes.



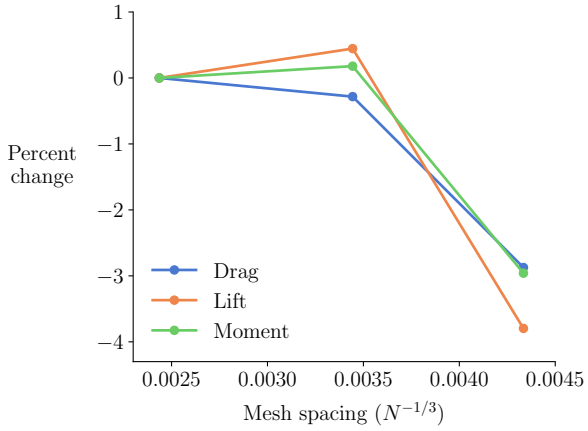
(a) 12M refinement zones (b) 24M refinement zones (c) 69M refinement zones

(d) 12M close-up at apex (e) 24M close-up at apex (f) 69M close-up at apex

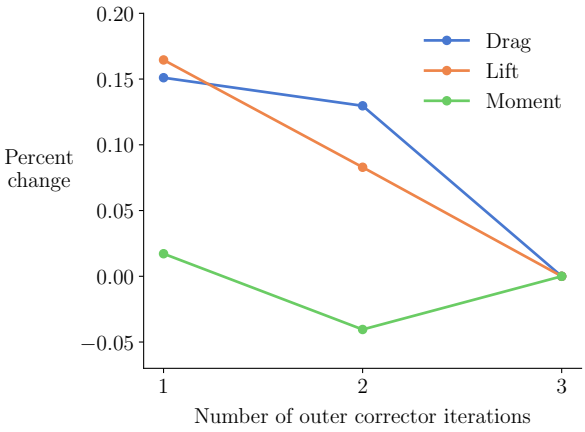
Figure 2.6: DDES meshes



(a) Time step refinement with 12M mesh



(b) Mesh refinement with $\Delta t = 0.5 \times 10^{-4}$ s



(c) Outer loop convergence for 12M mesh, $\Delta t = 0.5 \times 10^{-4}$ s

Figure 2.7: DDES convergence with discretization and solver parameters at $\alpha = 25°$

Table 2.3: DDES mesh characteristics

Label	Cells (N)	$y^+_{\max}$
69M	69,167,178	0.49
24M	24,481,833	1.24
12M	12,281,039	1.24

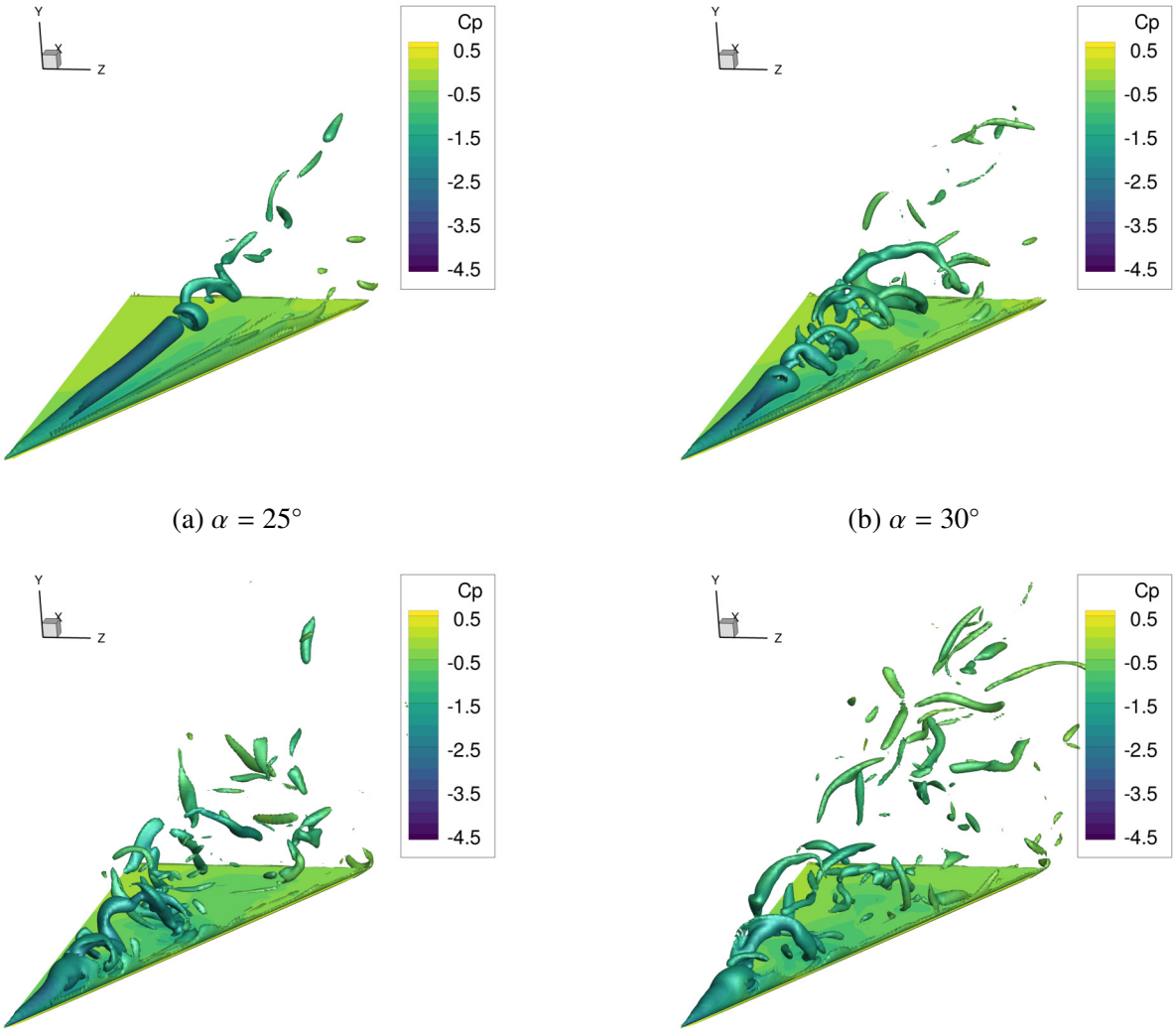
2.4.2 Angle-of-attack sweep

We run DDES with the 24M mesh for angles of attack from 0 to 40 deg. From 25 to 40 deg, we also run DDES with the 69M mesh because these are the most challenging flow conditions to resolve accurately. The aerodynamic characteristics near the point of maximum lift are dominated by unsteady vortex effects, such as vortex breakdown [44]. Figure 2.8 shows how vortex breakdown causes the vortex structure to become progressively less coherent as the angle of attack increases and the vortex breakdown location moves closer to the apex.

To run the angle-of-attack sweep, we initialize the flow from freestream conditions and run each case for 2.0 seconds in simulation time. We then compute the coefficients as the time average from 0.5 to 2.0 seconds to avoid including the transients at the start of the simulation. Figure 2.9 shows the time histories and cumulative mean of the moment coefficient for a few representative cases. We find that with this averaging window, the cumulative mean flattens out by the end of the simulation for all but one case. For the 40 deg case with the 69M mesh, the cumulative mean has a downward trend when the simulation ends. It is possible that running this case for longer would result in a closer match with the experimental data. However, we did not run this any further to be consistent with the other results. Increasing the simulation time is equivalent in some sense to obtaining a more refined solution.

The predicted coefficient values match the experimental data well up to 25 deg (Fig. 2.10). There are some discrepancies at higher angles of attack. The lift and drag at 30 deg are underpredicted by 4–6%. Refining the mesh from 24M to 69M does not improve the accuracy at this condition. This suggests that the DDES model does not fully resolve the unsteady vortex effects near the point of maximum vortex strength. The 40 deg condition is the most sensitive to mesh refinement. The 69M mesh correctly predicts a break in the pitching moment between 35 and 40 deg, whereas the 24M mesh does not. The suction predicted by the 69M mesh at 40 deg is lower and more localized near the wing apex compared to the 24M mesh (Fig. 2.11). This results in lower lift and drag, less pitch-down moment, and a better match with the experimental data. However, DDES does not fully capture the shape of the experimental pitching moment curve from 30 to 40 deg, even with the 69M mesh.

Figure 2.10 also shows the unsteady variation of the coefficients, where the variation is computed



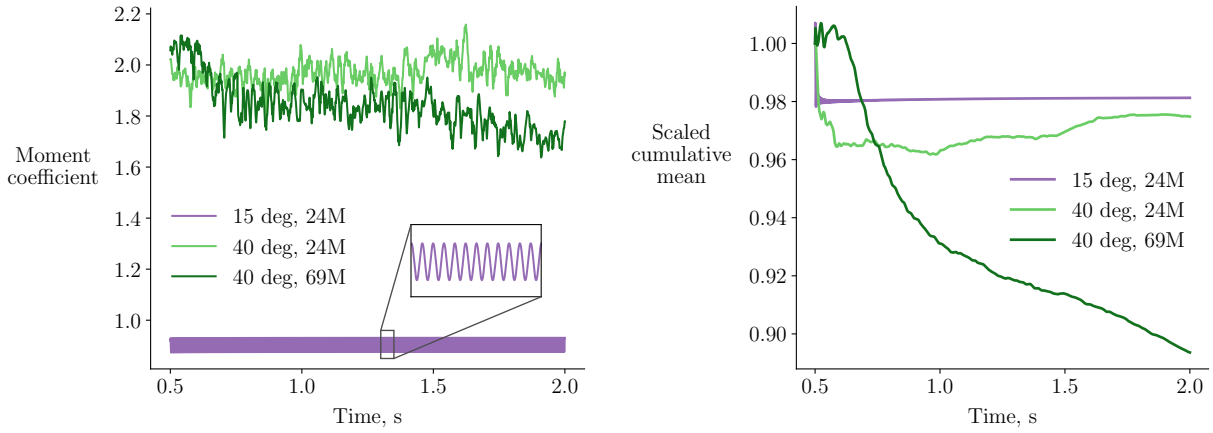
(a) $\alpha = 25°$

(b) $\alpha = 30°$

(c) $\alpha = 35°$

(d) $\alpha = 40°$

Figure 2.8: DDES (24M) Q-criterion isosurfaces for $Q = 10^7$ at $t = 1.2\,\mathrm{s}$



(a) Time histories

(b) Time-averaging

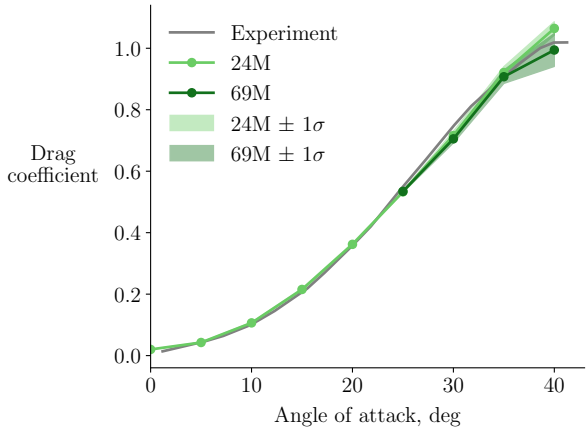
Figure 2.9: DDES time histories are time-averaged to obtain coefficient values

as one standard deviation of the unsteady time history. At 25 deg and lower angles of attack, the unsteady variation is small. This coincides with the regime where DDES is most accurate. At higher angles of attack, the variation is around 2–3% of the mean value for almost all cases. The one exception is the 40 deg case with the 69M mesh, which has a variation of 5–6%. This is another indication that a time-averaging window that is longer or shifted forward in time would give more accurate results for this case.

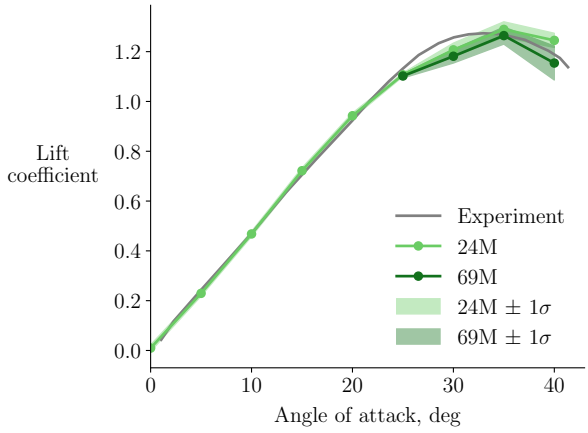
More generally, the predicted coefficients could be improved by using high-order schemes. High-order wall-modeled large-eddy simulations (WMLES) have been used to accurately capture pitch break for the NASA Common Research Model at transonic conditions [39]. However, WMLES is more costly than DDES, and the cost of DDES may already be prohibitive for iterative design. Using 320 Skylake cores on NASA's Electra supercomputer, the 40 deg DDES case takes approximately 24 and 81 hours to run with the 24M and 69M meshes, respectively. This is approximately 100–1000 times more expensive than RANS in terms of total CPU time. This comparison is meant to be an order of magnitude estimate, and we acknowledge that there are many differences between the RANS and DDES solvers that contribute to the cost disparity.

2.4.3 Steadiness metric

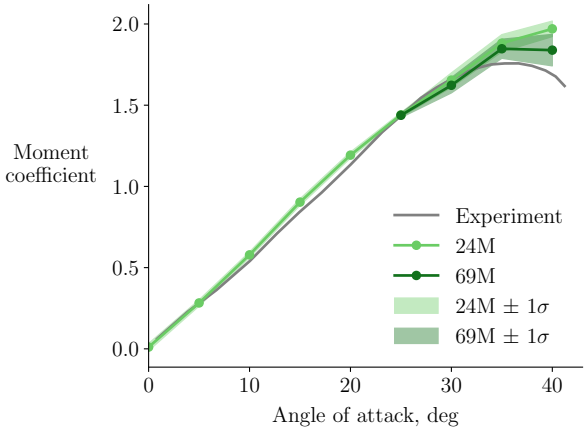
In Fig. 2.9b, we observe that the cumulative mean for the 15 deg case flattens out quickly. This is because the flow is steady. We now further analyze the time histories to formalize the difference between steady and unsteady conditions. The time histories for steady and unsteady conditions have distinct characteristics (Fig. 2.9a). Steady time histories are nearly periodic and oscillate about the



(a) Drag

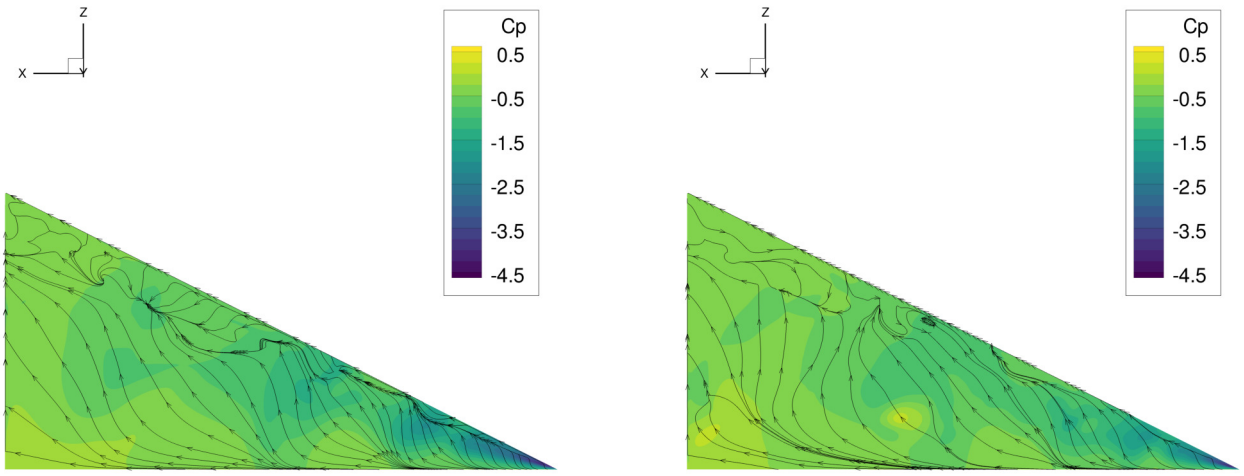


(b) Lift



(c) Pitching moment

Figure 2.10: Comparison of DDES force and moment coefficients with experimental data from Jarrah and Ashley [73]



(a) 24M

(b) 69M

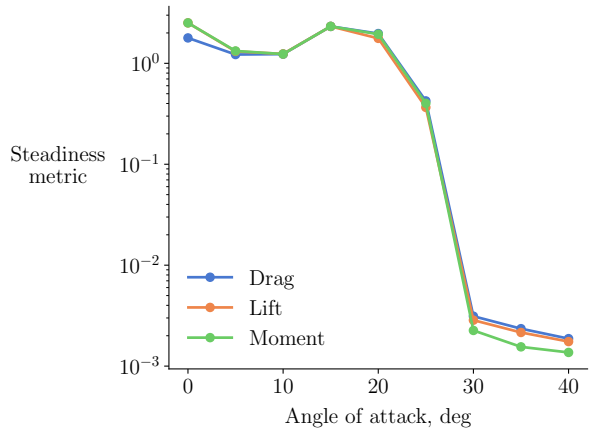
Figure 2.11: DDES upper surface pressure coefficient contours and streamlines for $\alpha = 40°$ at $t = 1.2\,\mathrm{s}$

steady-state solution at a high frequency. Conversely, unsteady time histories are not periodic and have lower frequency content. Based on these observations, we can formulate a steadiness metric as a product of the periodicity and the dominant frequency of the time series. Using Fisher's g statistic [90, 91] as an estimate for the periodicity and nondimensionalizing the frequency, we define the steadiness metric (SM) as

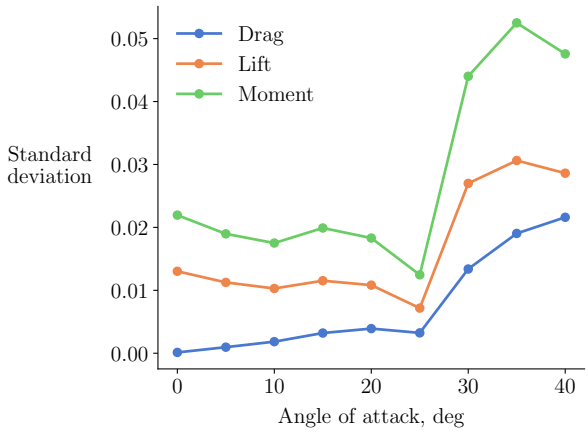
$$\mathrm{SM} = \frac{\max_i P_i}{\sum_i P_i} \frac{f_{P_{\max}} L_{\mathrm{ref}}}{U_\infty}, \tag{2.1}$$

where P_i represents the periodogram of the time series and $f_{P_{\max}}$ is the frequency associated with the peak of the periodogram. We plot the steadiness metric for the 24M DDES results in Fig. 2.12a. The steadiness metric is high from 0 to 20 deg, indicating that the flow is steady. This corresponds precisely to the regime where RANS is accurate. The steadiness metric drops off above 20 deg, indicating unsteady flow. The flow regimes identified using the steadiness metric are consistent with the regimes described by Hummel [88] for a 65 deg delta wing. This compares favorably to using the standard deviation as a measure of unsteadiness (Fig. 2.12b), which would incorrectly group the 25 deg condition with the steady angles of attack.

The proposed steadiness metric is computed from DDES time histories, which means that an unsteady analysis must be run to determine whether a case is steady or unsteady. However, when running multiple cases, the steadiness metric can help automate which cases to run with a steady method instead of an unsteady method. For example, consider a DDES angle of attack sweep starting from the highest angle. The steadiness metric is computed after each DDES run is completed.



(a) Steadiness metric suggests steady flow up to $\alpha = 20°$



(b) Standard deviation misclassifies $\alpha = 25°$ as steady

Figure 2.12: Quantifying unsteadiness at different angles of attack (24M)

Once a prescribed threshold for the steadiness metric is crossed, the analysis can switch to a steady method for the lower angles of attack, decreasing the computational cost of the sweep. The results in Fig. 2.12a suggest a threshold value of 1, but this may not be applicable in general. For a different geometry or solver, a threshold can be identified by one unsteady analysis on a case that is known to be steady.

2.5 Conclusions

This chapter considers the flow over a delta wing with an aspect ratio of 2 at low-speed, high-angle-of-attack conditions. The accuracy of RANS and DDES at predicting force and moment coefficients for angles of attack from 0 to 40 deg is evaluated. The main contribution of this chapter is looking at the predicted trends in the coefficients for multiple angles of attack around the stall angle. These trends have not been well explored in prior high-fidelity work. A steadiness metric based on DDES frequency information is also proposed to distinguish between steady and unsteady flow conditions.

RANS provides accurate results for angles of attack up to 20 deg. The steadiness metric is used to show that this corresponds to the steady flow regime. RANS is inaccurate at higher angles of attack where the flow is unsteady. The lift and drag in this regime are underpredicted, and the pitching moment trend is incorrect.

DDES is more accurate than RANS at higher angles of attack. The errors in the lift and drag are within 6% for DDES compared to 20% for RANS. The DDES results are most sensitive to mesh refinement at the highest angles of attack. At 40 deg, the 69M mesh provides a much

closer match with the experimental data than the 24M mesh, especially for the pitching moment. However, the break in the pitching moment is not fully captured even with the 69M mesh. For the solvers considered in this chapter, the computational cost of DDES is 100–1000 times that of RANS. The ideal solver is accurate and fast enough to be used in an iterative design or optimization procedure. RANS satisfies these requirements up to moderate angles of attack, but predicting high-angle-of-attack characteristics at a computational cost suitable for iterative design remains a challenge.

Chapter 3

Strategies for Solving Low and High Mach Number Flows

Designing a supersonic transport aircraft requires evaluating performance at high Mach numbers in addition to the low-speed flows discussed in Chapter 2. CFD solvers that are designed for conventional aircraft applications can run into accuracy and efficiency issues at low and high Mach numbers. This chapter presents methods to address these issues.

3.1 Introduction

The flight envelope of a supersonic transport aircraft can span Mach numbers from 0.25 to 2.0 [15]. Designing more conventional aircraft also requires evaluating performance over a wide range of Mach numbers. A conventional transonic aircraft will fly through Mach numbers from 0.2 to 0.85 in a typical mission [92]. In addition, certain applications produce flow fields with a range of Mach numbers at a single operating condition. Helicopter rotors experience nearly incompressible flow at blade roots and compressible flow at the tips [93]. Nacelles in crosswind can experience supersonic flow around the inlet lips despite low subsonic flow outside the nacelle [94]. The Mach number range in applications such as these motivates the need for CFD solvers that are accurate and efficient across a wide range of Mach numbers.

In theory, compressible flow solvers represent the actual physics across all Mach numbers, making them a natural choice for aerodynamic design. In practice, these solvers commonly suffer from reduced accuracy and speed at low Mach numbers. Modifying compressible flow solvers to resolve these issues is usually called low-speed preconditioning. Low-speed preconditioning has been studied in depth by several authors for both explicit and implicit time-stepping schemes. Turkel [95] introduced a class of preconditioning matrices for low-speed flows based on analysis of the Euler equations. This approach modifies the time-stepping terms to equalize the advective and acoustic wave speeds, accelerating convergence. The same preconditioners can improve accuracy at

low Mach numbers if the artificial dissipation in the scheme is based on preconditioned fluxes [96]. This approach has also been applied to viscous flows [94, 97, 98]. van Leer et al. [99] proposed a preconditioner for all Mach numbers that is also based on the idea of minimizing the spread in wave speeds. They interpreted this approach as using different time steps along different characteristic directions, which led to the name "characteristic time-stepping". The authors demonstrated the preconditioner's effectiveness on the Euler equations at Mach numbers from 0.01 to 1.8 and on viscous flow at Mach 7.95. Lee [100] showed that preconditioning approaches based on the Euler equations work well for viscous flows in most practical cases.

Some work has also been done on low-speed preconditioning for Newton-based solvers. Newton-based solvers are fully implicit and offer more flexibility in addressing the accuracy and speed issues independently without introducing unwelcome stability restrictions. To speed up convergence with Newton–Krylov solvers, Knoll et al. [101] considered multiplicative Schwarz preconditioning, and Weston et al. [102] used approximate block factorizations. Both of these papers take an algebraic approach to preconditioning. In contrast, Mary et al. [103] applied characteristic-based preconditioning to an approximate Newton solver. This work was limited to a maximum Reynolds number of 800 and primarily considered accuracy. These limitations suggest an unexplored potential for improved performance with Newton-based solvers using characteristic preconditioning methods.

In addition, high-speed flows have not been explored as extensively as low-speed cases. Kaushik et al. [104] and Olawsky et al. [105] found that Newton-based solvers can run into difficulties converging supersonic flows with second-order schemes compared to more dissipative first-order schemes. They resolved this by starting the solution with a first-order scheme before switching to a second-order scheme closer to convergence. This approach lacks reliability because it is not always clear when to switch to a second-order scheme for a given case. A related approach for improving convergence for flows with shocks is to increase the artificial dissipation in the discretization. For upwind schemes, this can be achieved by limiting the minimum eigenvalue of the flux Jacobian, commonly referred to as an entropy fix. For example, changing the entropy fix to be more dissipative is an option in the widely used FUN3D code [106]. The amount of dissipation in upwind schemes can also be increased by using a more dissipative flux limiter [107]. The equivalent for central schemes is increasing the artificial dissipation constants, as done by Öhrman [108] and Aprovitola et al. [109], for example. Increasing dissipation improves robustness in exchange for reduced accuracy. We desire an automated approach that increases robustness without sacrificing accuracy.

In this chapter, we introduce three methods to improve the efficiency and accuracy of Newton-based compressible flow solvers across low and high Mach number regimes. In Sec. 3.3, we describe a simple method for improving low-speed accuracy by scaling the artificial dissipation of a scalar dissipation scheme. This method preserves the simplicity of the original scheme and is easy to implement. We then formulate a characteristic time-stepping approach for an approximate Newton

solver in Sec. 3.4. Characteristic time-stepping improves nonlinear convergence rates by reducing the stiffness of the linear system at each nonlinear iteration. Lastly, we present a dissipation-based continuation method in Sec. 3.5 that addresses nonlinear convergence difficulties that arise from high Mach number flows. This method leverages the idea that more dissipative methods are easier to converge by starting with high artificial dissipation and smoothly reducing the dissipation to the desired level as the solution converges.

3.2 Baseline RANS solver

The CFD solver we use is ADflow [77], a second-order finite volume code for multiblock and overset structured meshes. We solve the steady, compressible Reynolds-averaged Navier–Stokes (RANS) with the Spalart–Allmaras (SA) turbulence model [78]. The flow is assumed to be fully turbulent.

The Navier–Stokes equations can be written as

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_V Q\mathrm{d}V + \oint_{\partial V} \vec{F}_c \cdot \vec{n}\mathrm{d}A - \oint_{\partial V} \vec{F}_d \cdot \vec{n}\mathrm{d}A = 0, \tag{3.1}$$

where $Q = [\rho, \rho u, \rho v, \rho w, \rho E]^T$ represents the conservative variables, $\vec{F}_c$ is the convective flux, and $\vec{F}_d$ is the diffusive flux. This system is augmented by the SA equation, which adds the turbulent contribution to the diffusive flux. Discretizing in space with a cell-centered finite volume method, the equation for a cell i is

$$\frac{\mathrm{d}Q_i}{\mathrm{d}t}V_i + \sum_f \hat{F}_c \Delta A_f - \sum_f \hat{F}_d \Delta A_f = 0, \tag{3.2}$$

where the discretized fluxes, $\hat{F}_c$ and $\hat{F}_d$, are summed over the faces of the cell. We use the Jameson–Schmidt–Turkel (JST) scheme [79, 110] to discretize the convective flux and a central scheme with Green–Gauss gradients for the diffusive flux.

3.2.1 Artificial dissipation formulation

JST is a central scheme that combines second and fourth difference artificial dissipation terms for stability. We now describe the artificial dissipation formulation in the I mesh direction. The formulation for the J and K directions follow similarly. The second-difference dissipation coefficient at a face between cell i and $i+1$ is given by

$$\epsilon^{(2)}_{i+\frac{1}{2}} = \kappa_2 \Upsilon_{i+\frac{1}{2}} \hat{\Lambda}^I_{i+\frac{1}{2}}, \tag{3.3}$$

and the fourth-difference coefficient is given by

$$\epsilon^{(4)}_{i+\frac{1}{2}} = \max\left(0, \kappa_4 \hat{\Lambda}^I_{i+\frac{1}{2}} - c_4 \epsilon^{(2)}_{i+\frac{1}{2}}\right). \tag{3.4}$$

By default, we use $\kappa_2 = 0.25$, $\kappa_4 = 0.0156$, $c_4 = 1$. $\Upsilon_{i+1/2}$ is the shock sensor at the face, which ensures that second-difference dissipation is only applied near shocks and fourth-difference dissipation is only applied away from shocks. The shock sensor at the face is computed as the maximum of the neighboring cell sensor values:

$$\Upsilon_{i+\frac{1}{2}} = \max\left(\Upsilon_i, \Upsilon_{i+1}\right). \tag{3.5}$$

For viscous flows, the shock sensor value in each cell is based on the local change in entropy, s:

$$\Upsilon_i = \frac{|s_{i+1} - 2s + s_{i-1}|}{|s_{i+1} + 2s + s_{i-1}|}. \tag{3.6}$$

$\hat{\Lambda}^I_{i+1/2}$ is the spectral radius of the flux Jacobian in the I direction at the face and is computed as the average of the neighboring cell radii:

$$\hat{\Lambda}^I_{i+\frac{1}{2}} = \frac{1}{2}\left(\hat{\Lambda}^I_i + \hat{\Lambda}^I_{i+1}\right). \tag{3.7}$$

To compute the spectral radius in the I direction, we first compute the isotropic spectral radius, Λ, in each direction. The isotropic spectral radius in the I direction is

$$\Lambda^I = \vec{U} \cdot \vec{n}^I + c|\vec{n}^I|, \tag{3.8}$$

where $\vec{U} = [u, v, w]^T$ is the local velocity vector, c is the local speed of sound, and $\vec{n}^I$ is the cell's average face area normal in the I direction. Similarly, for the J and K directions, we have

$$\Lambda^J = \vec{U} \cdot \vec{n}^J + c|\vec{n}^J|, \tag{3.9}$$

$$\Lambda^K = \vec{U} \cdot \vec{n}^K + c|\vec{n}^K|. \tag{3.10}$$

We then compute a scaled spectral radius that is appropriate for high aspect ratio cells as suggested by Martinelli [111]:

$$\hat{\Lambda}^I = \Lambda^I\left(1 + \left(\frac{\Lambda^J}{\Lambda^I}\right)^{0.67} + \left(\frac{\Lambda^K}{\Lambda^I}\right)^{0.67}\right). \tag{3.11}$$

The artificial dissipation terms are proportional to the spectral radius, so appropriately scaling the spectral radius is critical for solver convergence and solution accuracy.

3.2.2 Solver algorithm

To converge the semi-discrete form (Eq. 3.2) to a steady solution, we define the residual in each cell, R_i, as the time-derivative term and drive it to zero:

$$R_i = \frac{\mathrm{d}Q_i}{\mathrm{d}t} V_i = \sum_f \hat{F}_d \Delta A_f - \sum_f \hat{F}_c \Delta A_f = 0. \tag{3.12}$$

The baseline nonlinear solver algorithm we use is the approximate Newton–Krylov (ANK) method with pseudo-transient continuation implemented in ADflow [80]. The solver is primarily designed for robustness and speed at transonic flow conditions. The solution update at each iteration of the ANK solver, $\Delta Q^{(n)}$, is obtained by solving the linear system

$$\left[\left(T^{-1} \right)^{(n)} + \left(\frac{\partial R}{\partial Q} \right)^{(n)} \right] \Delta Q^{(n)} = -R(Q^{(n)}), \tag{3.13}$$

where R is the residual vector, $Q^{(n)}$ is the state vector, and T is the time-step matrix. T is a diagonal matrix, where the diagonal terms for each cell are the time step, Δt_i, that results in the desired global CFL number. The global CFL number is initially small and is increased as the solution converges. The SA turbulence equation is solved separately after each ANK update for the flow equations. A complete description of the ANK algorithm is given by Yildirim et al. [80].

3.2.3 Accuracy and cost comparisons

We compare the methods presented in this paper to the baseline solver in terms of accuracy and cost. Unless stated otherwise, all solutions are converged to a total residual of 10^{-10} relative to the freestream residual. To provide a hardware-independent measure of cost, we report total CPU times in TauBench work units [112]. We run all computations on NASA's Aitken supercomputer using Cascade Lake nodes, where one work unit equals 3.9078 processor seconds. We determine this by running TauBench ten times with the command `mpiexec -np 1 ./TauBench -n 250000 -s 10` and computing the average run time.

3.3 Scaling artificial dissipation to improve accuracy at low Mach numbers

Artificial dissipation is essential for nonlinear convergence but can compromise accuracy. The JST scalar dissipation scheme is generally more dissipative than second-order upwind schemes [113]. Central schemes that are less dissipative than scalar dissipation include matrix dissipation [113] and

the convective-upstream-split-pressure scheme [114]. However, these central and upwind schemes are primarily designed for transonic flows and run into accuracy issues at low Mach numbers because of improperly scaled artificial dissipation [96]. One common approach to scale the dissipation is to multiply the flux Jacobian by a local preconditioning matrix. For JST, this modifies the spectral radius and, consequently, the dissipation. Using a local preconditioning matrix can also accelerate convergence, which we discuss in Sec. 3.4. To improve low-speed accuracy, we opt to modify the spectral radius directly. The advantage of this approach over prior ones is that it requires minimal changes to the JST scalar dissipation formulation.

In the low Mach number limit, the advective contribution to the spectral radius (Eq. 3.8) is $O(1)$, whereas the acoustic contribution is $O(1/M)$ [96, 115]. Rieper [116] showed that artificial dissipation must be independent of the Mach number for accuracy in the incompressible limit. To scale the artificial dissipation for low Mach numbers, we introduce the acoustic scaling factor ζ in the isotropic spectral radius:

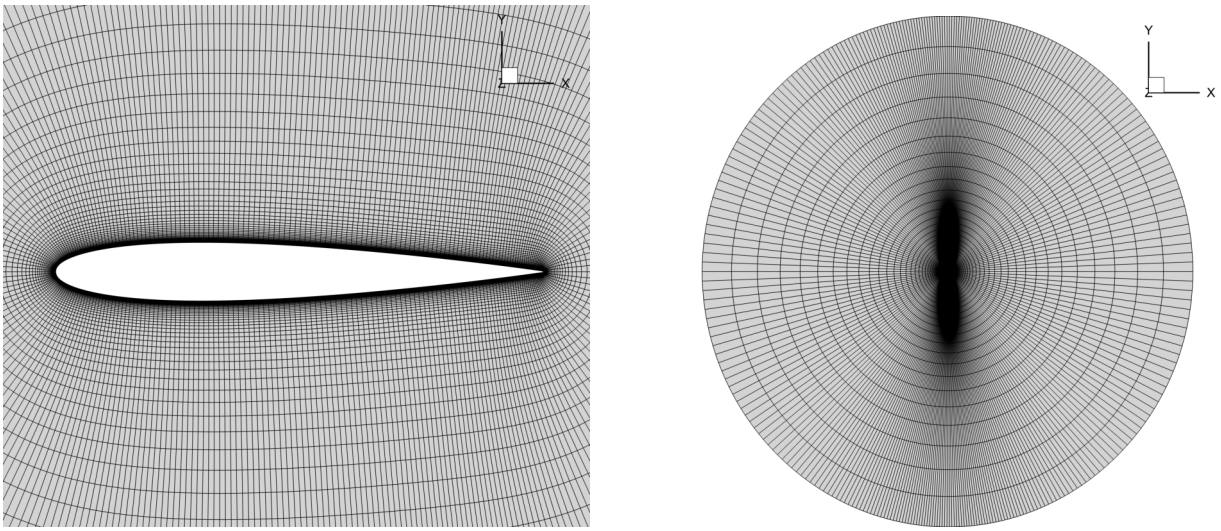
$$\Lambda^I = \vec{U} \cdot \vec{n}^I + \zeta c |\vec{n}^I|. \tag{3.14}$$

Using this formulation, we can select values for ζ that reduce or eliminate the dependence of artificial dissipation on the Mach number. Similar dissipation scaling approaches have been proposed in the context of upwind schemes [117] and kinetic energy preserving schemes [118].

To study the effect of ζ on low-speed accuracy, we consider a NACA 0012 airfoil at a Reynolds number of 10^6, 3° angle of attack, and Mach numbers from 0.01 to 0.4. We use an O-mesh with 296 cells around the airfoil and 128 cells in the offwall direction, for a total of 37888 cells (Fig. 3.1). The initial offwall spacing is 10^{-6} chord lengths, and the farfield is 100 chord lengths away from the airfoil, which is consistent with the Drag Prediction Workshop guidelines [119].

To evaluate accuracy at different values of ζ, we compare the RANS results with predictions using XFOIL [120], a 2D panel method coupled with an integral boundary layer model. We use 300 panels for the inviscid discretization in the XFOIL analyses. Increasing the number of panels to 360 gives the same coefficient values to 4 significant digits. We force transition at the 0.3% chord location to match the fully turbulent RANS simulations. This is the smallest positive value we could use for the trip location without encountering numerical difficulties.

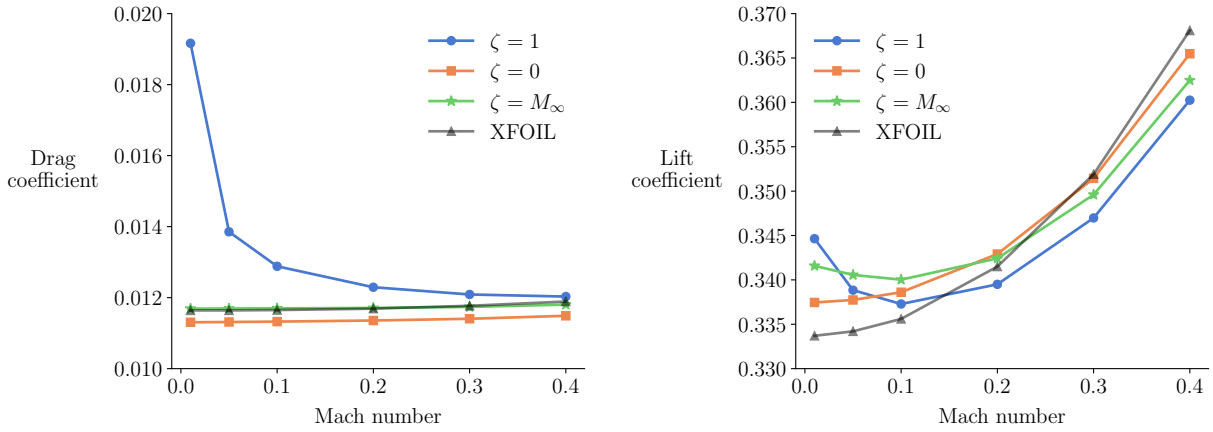
Fig. 3.2a compares the drag for the baseline ($\zeta = 1$), no acoustic contribution ($\zeta = 0$), and scaling the acoustic contribution by the freestream Mach number ($\zeta = M_\infty$). With no scaling, drag diverges as the Mach number is reduced. The drag overprediction at low Mach numbers for the baseline is primarily caused by a nonphysical suction peak on the trailing edge surface (Figs. 3.3 and 3.4). The baseline also has small inaccuracies in pressure at the leading edge. These nonphysical pressure fluctuations are a direct result of improperly scaled dissipation, as shown by Guillard and Viozat [121]. Chen et al. [122] also found similar trailing edge pressure spikes for low-speed inviscid flow. Scaling the acoustic contribution removes the trailing edge artifact and results in more accurate drag



(a) Nearfield (b) Farfield

Figure 3.1: NACA 0012 O-mesh with 296×128 cells

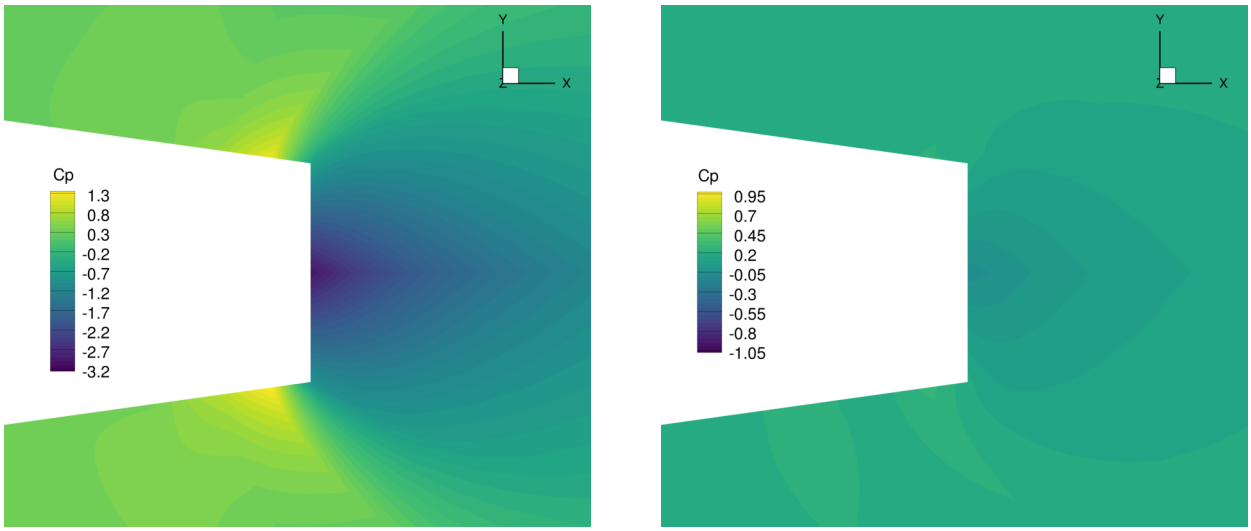
values. In addition, the change in drag flattens out as the Mach number is reduced. Scaling by the freestream Mach number yields the most accurate drag predictions compared to XFOIL.



(a) (b)

Figure 3.2: Scaling artificial dissipation improves accuracy

The baseline errors in the lift at low Mach numbers are not as significant as for drag (Fig. 3.2b), but the lift still diverges as the Mach number approaches zero. Reducing artificial dissipation avoids divergence. The value of ζ is not critical for lift prediction because the differences are primarily caused by discretization errors. We run a mesh refinement study at Mach 0.1 to demonstrate this effect. In addition to the 296×128 mesh, we use meshes of size 1184×512, 592×256, and

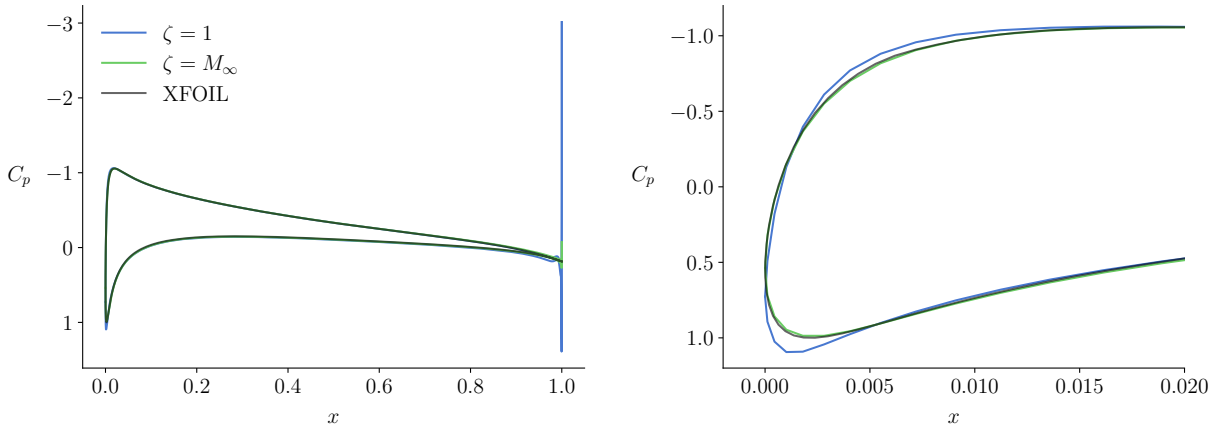


(a) $\zeta = 1$ (b) $\zeta = 10^{-2}$

Figure 3.3: Reducing artificial dissipation removes the spurious pressure spikes at the trailing edge (Mach 0.01)

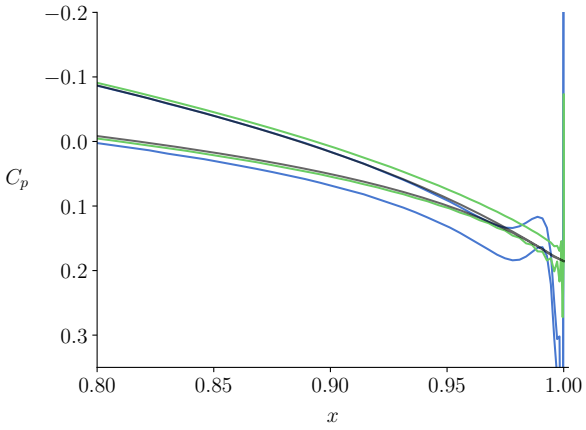
148×64 for the study. Figures 3.5 and 3.6 show the drag and lift for these four meshes, as well as the Richardson extrapolation [123]. The Richardson extrapolation estimates the values we would obtain on an infinitely fine mesh. The order of accuracy used for the extrapolation, p, is estimated from the three finest mesh solutions for each combination of output and ζ. For cases where the estimated order of accuracy is less than one, we use first-order extrapolation [124]. The extrapolated drag values depend strongly on ζ. The extrapolated drag for $\zeta = 1$ is 6.9% higher than for $\zeta = 0$ and 4.7% higher than for $\zeta = M_\infty$. This means that adequately scaling the dissipation is essential for drag prediction, even for fine meshes. On the other hand, the extrapolated lift values are within 1.0% of each other. The differences in the lift with ζ are significant only for the 148×64 mesh. However, such a coarse mesh is outside the asymptotic range [123] in terms of mesh size and should not be used in general. The mesh refinement study also shows that drag for $\zeta < 1$ is less sensitive to mesh size than baseline. As a result, scaling the artificial dissipation can indirectly speed up analyses by allowing for coarser meshes at the same accuracy as the baseline.

Based on these results, we recommend using $\zeta = M_\infty$ for cases where a freestream Mach number exists. For other cases, ζ should equal the dominant or average Mach number. An alternative approach would be to use local values for ζ instead of using the same value in all cells. For example, we could use the local Mach number instead of the freestream Mach number in each cell. For the NACA 0012 case, this results in similar accuracy and convergence to a global value of $\zeta = 0$. This is because the local Mach number near the airfoil is close to zero, and the cells closest to the airfoil dictate the overall behavior of the solver. However, local values may be more applicable for cases



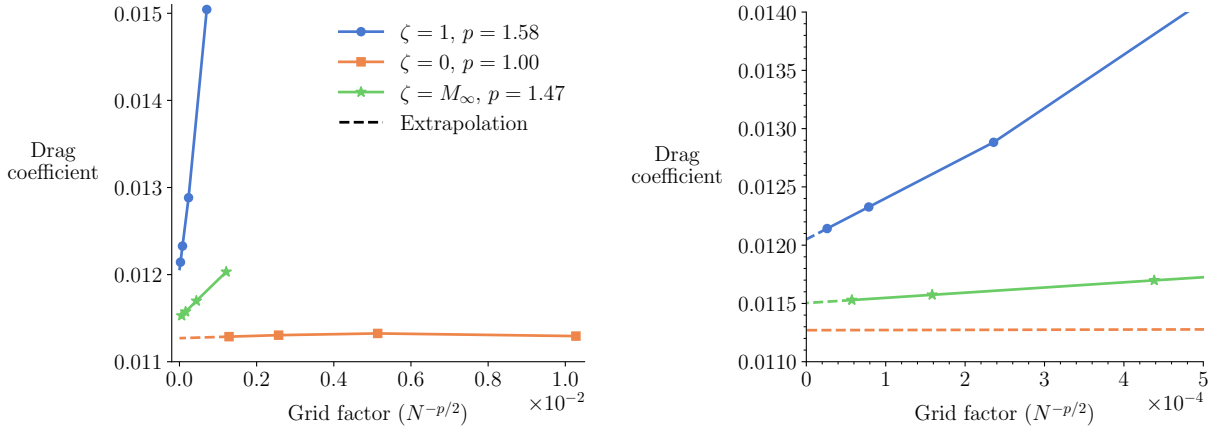
(a) Full airfoil

(b) Zoom-in on leading edge



(c) Zoom-in on trailing edge

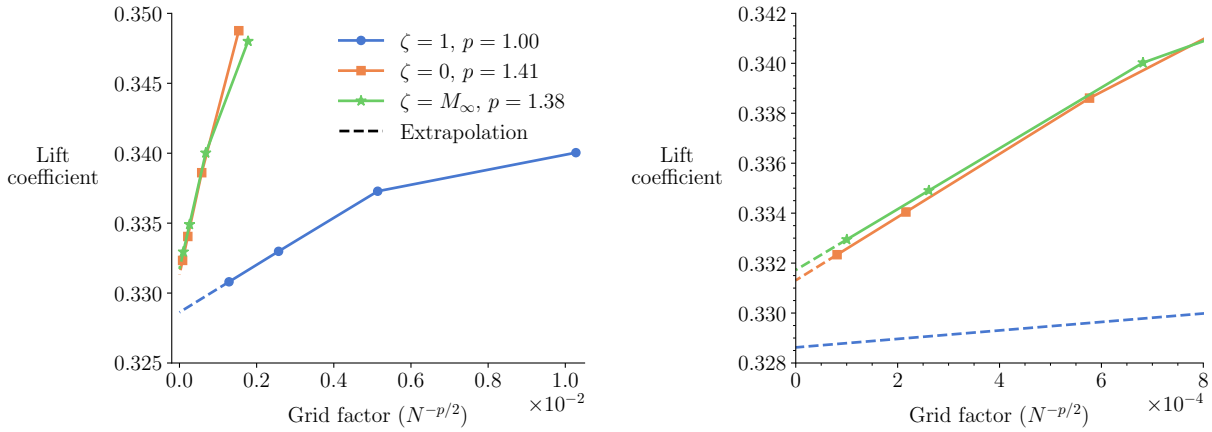
Figure 3.4: Reducing artificial dissipation improves pressure coefficient accuracy (Mach 0.01)



(a) Full convergence study

(b) Zoom-in on extrapolation

Figure 3.5: Mesh convergence of drag at Mach 0.1



(a) Full convergence study

(b) Zoom-in on extrapolation

Figure 3.6: Mesh convergence of lift at Mach 0.1

where a freestream Mach number is not defined. The proposed scaling approach with global or local ζ is easy to implement and improves the accuracy of compressible solvers at low Mach numbers.

3.4 Accelerating convergence at low Mach numbers with characteristic time-stepping

In addition to lower accuracy, compressible flow solvers face slower convergence at low Mach numbers. For the NACA 0012 case from Sec. 3.3 with the 296×128 mesh and $\zeta = 1$, the solution for Mach 0.01 is three times slower than Mach 0.4 (Fig. 3.7). The slower convergence results from the difference in magnitude between the advective and acoustic wave speeds. This results in a more restrictive upper bound on the time step for explicit solvers. For Newton-based solvers, this is reflected in the stiffness of the linear system that is solved at each nonlinear iteration.

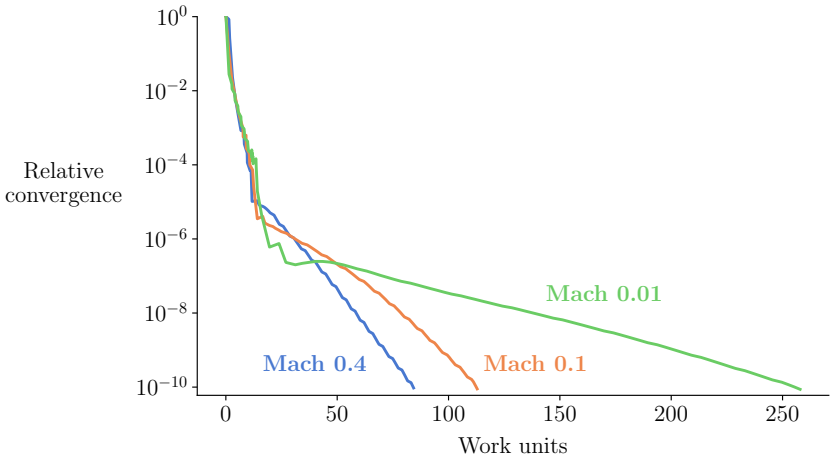


Figure 3.7: Cost increases as Mach number decreases (NACA 0012)

The baseline ANK solver defaults are tuned for compressible flow. However, we use settings that are beneficial for low Mach number flows for all cases with freestream Mach numbers of 0.4 or lower. The main changes are using more linear solver iterations and a stronger preconditioner for the linear system. Solving the linear system more tightly for higher Mach numbers and flows with shocks is not as beneficial for improving nonlinear convergence.

We use a local preconditioning or characteristic time-stepping (CTS) approach to accelerate convergence at low Mach numbers further. This approach involves modifying the time-stepping terms to equalize the advective and acoustic wave speeds. We refer to this approach as characteristic time-stepping to emphasize that we are not modifying the artificial dissipation using the same preconditioner. To apply CTS to the ANK solver, we multiply the time-step matrix by the

preconditioning matrix P. The linear system with the ANK step (Eq. 3.13) requires the inverse of the time-step matrix, so it is more convenient to work directly with the inverse of the preconditioner, P^{-1}:

$$\left[\left(P^{-1}T^{-1}\right)^{(n)} + \left(\frac{\partial R}{\partial Q}\right)^{(n)}\right] \Delta Q^{(n)} = -R(Q^{(n)}). \tag{3.15}$$

We make one change to adapt the preconditioner for use with pseudo-transient continuation. When the CFL number is low, the linear system is not stiff, and the baseline solver regularly outperforms CTS. A similar lack of stiffness was noted by Turkel and Vatsa [125] for small physical time steps in a dual time-stepping context. To take advantage of the baseline solver in this situation, we use a blended preconditioner based on the CFL number:

$$P^{-1}_{\text{blend}} = \left(\frac{\text{CFL}}{\text{CFL}_{\max}}\right) P^{-1} + \left(1 - \frac{\text{CFL}}{\text{CFL}_{\max}}\right) I. \tag{3.16}$$

With this blending, the solver initially acts like the baseline and reaches the fully preconditioned method when the CFL number reaches its maximum value.

The inverse preconditioner, P^{-1}, is a block diagonal matrix. Each diagonal block corresponds to one cell, and we denote the preconditioner for cell i as P_i^{-1}. Several different preconditioning matrices can be used in this formulation. We consider the Turkel preconditioner [95], denoted here as P_{T}, and the van Leer–Lee–Roe (VLR) preconditioner [99], denoted as P_{VLR}.

3.4.1 Turkel preconditioner

It is convenient to write the Turkel preconditioner in terms of the Euler symmetrizing variables,

$$\tilde{Q} = [p/(\rho c), u, v, w, p - c^2\rho]^T. \tag{3.17}$$

With these variables, the preconditioner for cell i is

$$\tilde{P}_{\mathrm{T},i}^{-1} = \begin{bmatrix} \frac{1}{M_\mathrm{T}^2} & 0 & 0 & 0 & 0 \\ \frac{\alpha u}{c M_\mathrm{T}^2} & 1 & 0 & 0 & 0 \\ \frac{\alpha v}{c M_\mathrm{T}^2} & 0 & 1 & 0 & 0 \\ \frac{\alpha w}{c M_\mathrm{T}^2} & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{3.18}$$

where we use M_T instead of the usual symbol β to avoid confusion with the β introduced for the VLR preconditioner in Sec. 3.4.2. As in prior work, we define M_T as a truncated form of the local Mach number:

$$M_\mathrm{T}^2 = \min\left(1, \max(M^2, \phi M_\infty^2)\right). \tag{3.19}$$

The lower bound on M_T^2 prevents the matrix from becoming ill-conditioned. Using $\phi > 0$ bounds M_T^2 away from zero in cells where the local Mach number is nearly zero. The ill-conditioning is mainly an issue at stagnation points. We find that $\phi = 10^{-4}$ provides the best performance while avoiding convergence issues. Larger values of ϕ do not show speedup compared to the baseline, whereas smaller values fail to resolve the stagnation point issue.

The upper bound on M_T^2, combined with defining α as

$$\alpha = 1 - \left(M_\mathrm{T}^2\right)^{10}, \tag{3.20}$$

eliminates preconditioning for locally supersonic flow. For $M \geq 1$, $M_\mathrm{T}^2 = 1$ and $\alpha = 0$, which is equivalent to no preconditioning. $(M_\mathrm{T}^2)^{10}$ quickly approaches 0 for subsonic Mach numbers. Consequently, $\alpha \approx 1$ for Mach numbers less than 0.8. This results in better conditioning than $\alpha = 0$. Prior time-stepping approaches typically used the preconditioner with $\alpha = 0$ for its improved robustness despite not providing the optimal conditioning in the low Mach number limit. The difference in robustness has been attributed to $\alpha = 1$ having lower artificial dissipation than $\alpha = 0$ [96]. However, we do not use the same scaling for the dissipation as for the time step, so we avoid this robustness issue.

Finally, we transform the matrix to conservatives variables through

$$P_{\mathrm{T},i}^{-1} = S\tilde{P}_{\mathrm{T},i}^{-1}S^{-1}, \tag{3.21}$$

where S is the transformation matrix from symmetrizing to conservative variables:

$$S = \begin{bmatrix} \frac{\rho}{c} & 0 & 0 & 0 & -\frac{1}{c^2} \\ \frac{\rho u}{c} & \rho & 0 & 0 & -\frac{u}{c^2} \\ \frac{\rho v}{c} & 0 & \rho & 0 & -\frac{v}{c^2} \\ \frac{\rho w}{c} & 0 & 0 & \rho & -\frac{w}{c^2} \\ c\rho\left(\frac{M^2}{2} + \frac{1}{\gamma - 1}\right) & \rho u & \rho v & \rho w & -\frac{M^2}{2} \end{bmatrix}. \tag{3.22}$$

3.4.2 van Leer-Lee-Roe (VLR) preconditioner

The VLR preconditioner is usually expressed in terms of the Euler symmetrizing variables and in a flow-aligned coordinate frame:

$$\tilde{\bar{P}}_{\mathrm{VLR},i}^{-1} = \begin{bmatrix} \frac{\beta^2 + \tau}{M_{\mathrm{VLR}}^2 \tau} & \frac{1}{M} & 0 & 0 & 0 \\ \frac{1}{M} & 1 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{\tau} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\tau} & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{3.23}$$

where

$$\beta = \begin{cases} \sqrt{1 - M_{\text{VLR}}^2}, M < 1 \\ \sqrt{M_{\text{VLR}}^2 - 1}, M \geq 1 \end{cases}, \qquad \tau = \begin{cases} \sqrt{1 - M_{\text{VLR}}^2}, M < 1 \\ \sqrt{1 - 1/M_{\text{VLR}}^2} + \epsilon, M \geq 1 \end{cases}. \tag{3.24}$$

We define M_{VLR}^2 like M_{T}^2 for the Turkel preconditioner but without any restrictions for supersonic flow:

$$M_{\text{VLR}}^2 = \max(M^2, \phi M_\infty^2). \tag{3.25}$$

We use $\phi = 10^{-4}$, the same value we use with the Turkel preconditioner. We also set $\epsilon = 10^{-4}$ when defining τ (Eq. 3.24) to avoid dividing by zero for sonic flow. We transform the preconditioner first to Cartesian coordinates with the rotation matrix Z and then to conservatives variables with the state transformation matrix S (Eq. 3.22):

$$P_{\text{VLR},i}^{-1} = SZ\tilde{\tilde{P}}_{\text{VLR},i}^{-1} Z^T S^{-1}. \tag{3.26}$$

The rotation matrix [126] is

$$Z = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \cos\alpha\cos\theta & -\sin\theta & -\sin\alpha\cos\theta & 0 \\ 0 & \cos\alpha\sin\theta & \cos\theta & -\sin\alpha\sin\theta & 0 \\ 0 & \sin\alpha & 0 & \cos\alpha & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{3.27}$$

where

$$\sin\theta = \frac{v}{\sqrt{u^2 + v^2}}, \qquad \sin\alpha = \frac{w}{\sqrt{u^2 + v^2 + w^2}},$$

$$\cos\theta = \frac{u}{\sqrt{u^2 + v^2}}, \qquad \cos\alpha = \frac{\sqrt{u^2 + v^2}}{\sqrt{u^2 + v^2 + w^2}}.$$

3.4.3 The importance of dissipation scaling

When introducing the VLR preconditioner, van Leer et al. [99] used an upwind scheme and explicit time-stepping. They found that applying preconditioning to only the time-stepping terms resulted in severe restrictions on the time step. To avoid these restrictions, it was necessary to use a preconditioned form of the dissipation matrix in the Roe flux. However, they were unclear on whether this necessity extends to central schemes such as scalar dissipation. We show that dissipation

scaling is also necessary for scalar dissipation schemes and Newton-based solvers.

We first apply CTS to the NACA 0012 case at Mach 0.01. For this section, we only show results with the Turkel preconditioner. However, we see similar trends with VLR. We find that the performance strongly depends on the choice of ζ (Fig. 3.8). Using $\zeta = M_\infty$ results in faster convergence than the baseline, whereas CTS is slower than the baseline for $\zeta = 1$ and $\zeta = 0$. For $\zeta = 1$, CTS takes more than twice as long as the baseline to converge and exhibits noisy convergence below a relative residual of 10^{-7}. Convergence with CTS is noisier than the baseline, even for cases where CTS is faster. This is related to the preconditioner becoming ill-conditioned at stagnation points. The noise is worse for lower freestream Mach numbers (Fig. 3.9). In addition, using larger values of ϕ reduces the noise but results in slower convergence. The noise is most significant for $\zeta = 1$ because of the nonphysical stagnation regions shown in Fig. 3.3a. We conclude that applying local preconditioning to the time step is problematic when the flow solution is locally nonphysical. The slowdown with CTS is less severe for $\zeta = 0$ than $\zeta = 1$ but still indicates that the solver is not working as intended. As mentioned before, removing too much dissipation can compromise solver robustness. In this case, the solver still converges but performs poorly.

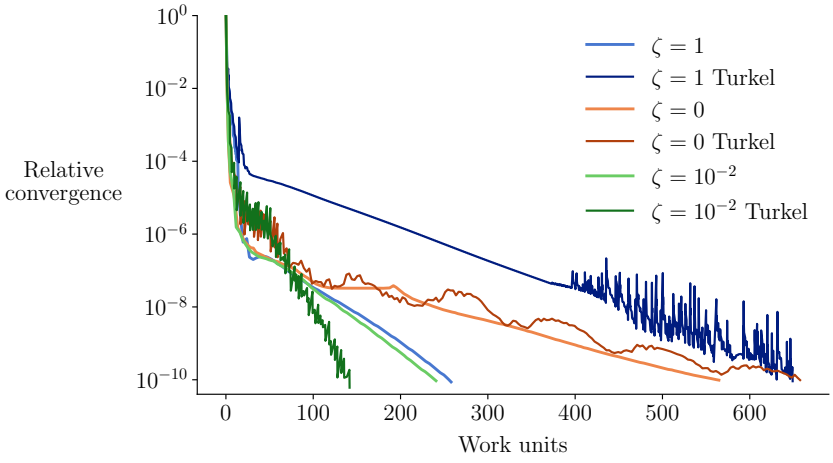


Figure 3.8: CTS reduces cost for $\zeta = 10^{-2}$ (NACA 0012 at Mach 0.01)

We can look at linear convergence to see how CTS achieves speedup for $\zeta = M_\infty$. The linear system for each ANK step is solved using the generalized minimum residual (GMRES) method [127]. The target linear residual is 0.05. We also set the maximum number of GMRES iterations to 100. If the iteration limit is reached before the linear residual target is met, the solver will continue with the partially converged step. We use the linear residual as an indicator of the linear system stiffness. For $\zeta = M_\infty$, CTS reduces the linear residuals such that they are all less than 0.1 (Fig. 3.10a). This results in faster nonlinear convergence. However, the same is not true for $\zeta = 0$ (Fig. 3.10b). Despite reducing the linear residuals, nonlinear convergence is not improved with CTS. As a result, we use

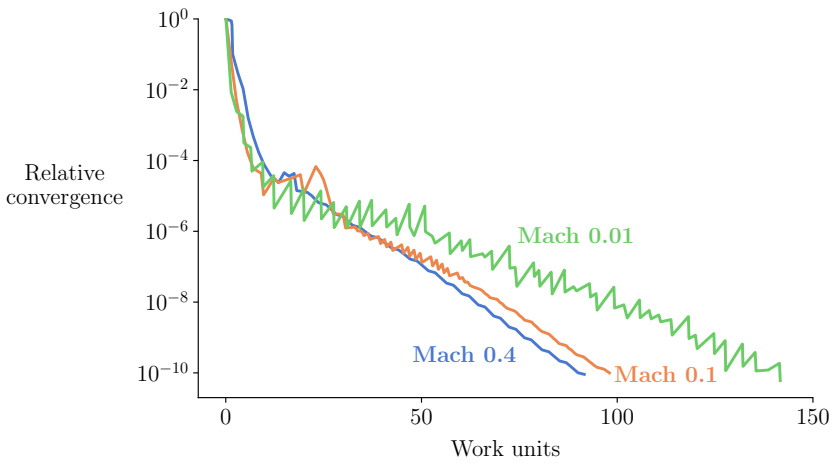
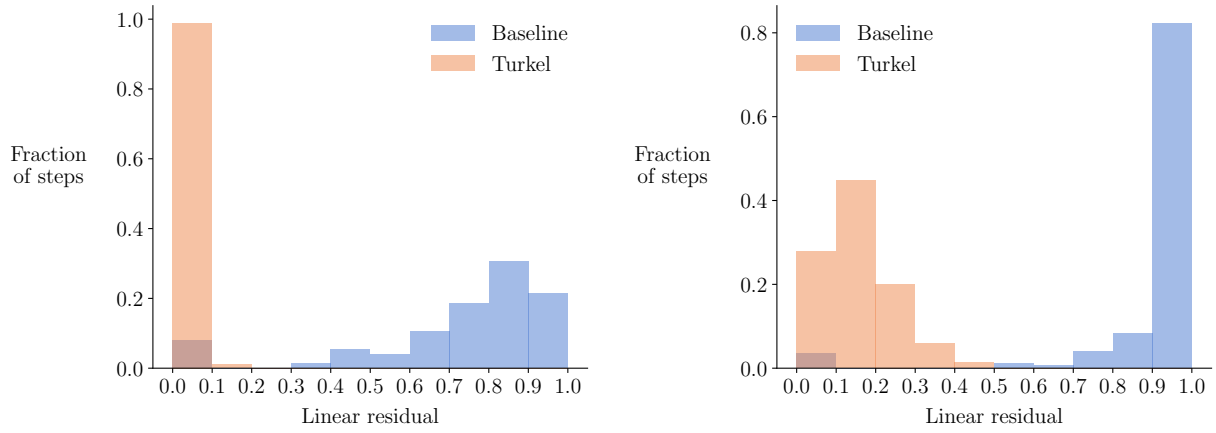


Figure 3.9: CTS convergence is noisier at lower Mach numbers (NACA 0012, $\zeta = M_\infty$, Turkel)

$\zeta = M_\infty$ for all baseline and CTS cases with freestream Mach numbers of 0.4 or lower in Sec. 3.4.4 and Sec. 3.4.5.



(a) $\zeta = 10^{-2}$ (b) $\zeta = 0$

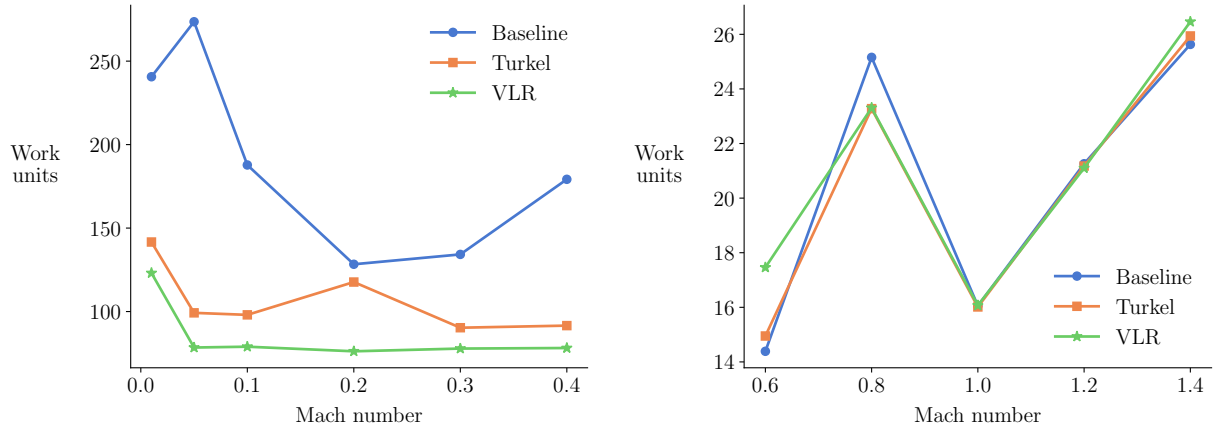
Figure 3.10: CTS improves linear convergence, but this does not guarantee improved nonlinear convergence (NACA 0012 at Mach 0.01)

3.4.4 NACA 0012 Mach number sweep

We now compare Turkel and VLR preconditioning with the baseline for a range of Mach numbers. Both preconditioners converge faster than the baseline for all Mach numbers we tried between 0.01 to 0.4 (Fig. 3.11a). The most significant speedup results are for Mach 0.1 and lower, where Turkel

is 41%-64% faster than baseline. VLR is consistently faster than Turkel and achieves 49%-71% speedup compared to the baseline in this regime. There is also a considerable speedup for Mach 0.4, but we do not expect to see the same speedup if the baseline solver is tuned for compressible flow.

We also compare CTS and baseline for Mach numbers between 0.6 to 1.4. We use $\zeta = 1$ for these cases because we are no longer in the low Mach number regime. We also tune the solver for compressible flow. These higher Mach number cases converge faster if we switch to a Newton–Krylov (NK) solver after converging to a relative residual of 10^{-5} with ANK. The NK solver has no time-step matrix and solves the flow and turbulence equations simultaneously in the same Newton system. The lack of the time-step matrix means that CTS has no effect. However, this switch ensures a fair cost comparison in which the baseline solver is not restricted to less effective methods. The performance of Turkel and VLR is similar to the baseline at higher Mach numbers (Fig. 3.11b). This is the expected behavior for the Turkel preconditioner because we turn off preconditioning for locally supersonic flow. VLR is designed to be an all-speed preconditioner but performs similarly to baseline for higher Mach numbers. This is because the difference in wave speeds and the resulting linear system stiffness is not an issue for Mach numbers around 1. We could try Mach numbers much higher than 1, but this runs into other challenges. We discuss these challenges and propose a solution in Sec. 3.5.



(a) Low Mach

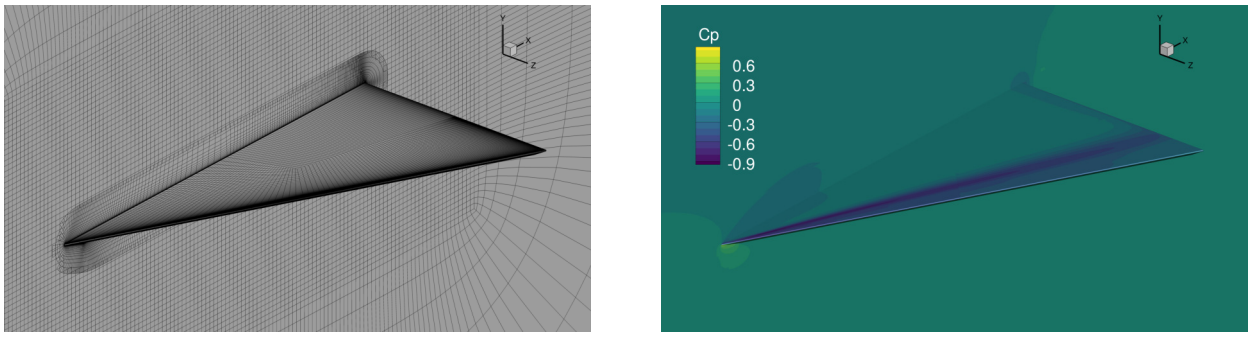
(b) High Mach

Figure 3.11: CTS reduces cost for low Mach numbers (NACA 0012)

3.4.5 Delta wing

There are also low Mach number cases for which CTS does not result in any speedup. We demonstrate this on a delta wing with an aspect ratio of two at a Mach number of 0.083, Reynolds number

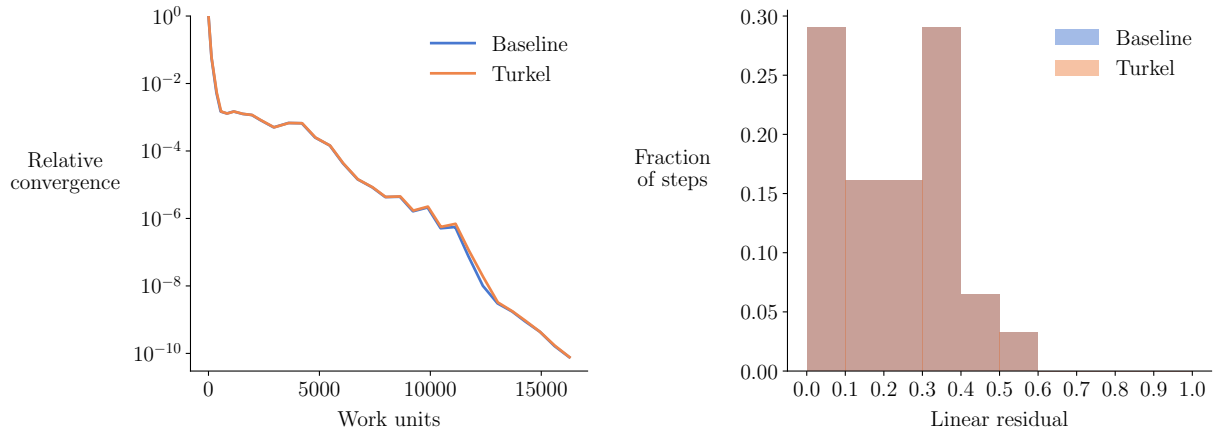
of 5.9×10^5, and angle of attack of 5° (Fig. 3.12). The geometry and flow conditions are taken from experimental work by Jarrah and Ashley [73]. We use an overset mesh with 1.9 million cells (Fig. 3.12a). More details on the mesh are given in Sec. 2.3.1. For this case, stagnation point flow is restricted to a small region near the wing apex (Fig. 3.12b). Despite the low freestream Mach number, the flow field does not have enough cells with nearly zero local Mach number to make the global linear system stiff. As a result, CTS produces nearly identical convergence behavior as the baseline (Fig. 3.13). This also explains why the linear residuals are lower than the NACA 0012 baseline case. We show only Turkel results for clarity, but VLR also gives nearly identical results.



(a) Overset mesh with 1.9 million cells

(b) Pressure contours at Mach 0.083, $\alpha = 5°$

Figure 3.12: Delta wing geometry from Jarrah and Ashley [73]



(a)

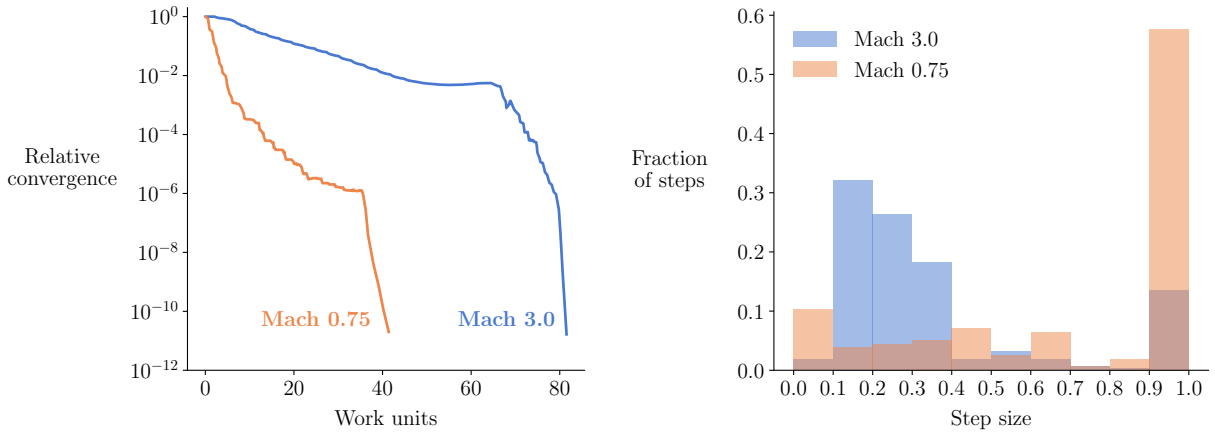
(b)

Figure 3.13: CTS is nearly identical to baseline for the delta wing

3.5 Dissipation-based continuation for flows with shocks

Flows with shocks pose different challenges for solvers than the low Mach number flows we have discussed. Shocks can move within the domain as the solution converges, causing significant changes in cells' states. To improve robustness, the ANK solver employs a physicality check that limits the update step size such that the density and energy do not change by more than 20% in any cell. A critical outcome of this approach is that the density and energy cannot become negative. Despite this added robustness, ANK can still perform poorly or fail to converge when the step sizes become small.

We demonstrate the effect of step size on convergence using transonic and supersonic flow over a NACA 0012 airfoil. At Mach 3.0, the step sizes are severely limited compared to Mach 0.75, which slows down convergence (Fig. 3.14). For the NACA 0012 cases in this section, we use the 296×128 mesh from Sec. 3.3. The Reynolds number is 10^7, and the angle of attack is 3° unless stated otherwise. As with the high Mach number cases in Sec. 3.4.4, we use the NK solver during the final stages of convergence for all cases in this section. We switch to the NK solver after converging to a relative residual of 10^{-8} with ANK. We could have converged these cases using only ANK. However, switching to NK results in faster convergence and represents typical settings for the baseline solver.



(a) (b)

Figure 3.14: Limited step sizes slow down convergence for supersonic flows (NACA 0012)

One approach to accelerating convergence is to loosen the physicality check to accept larger changes in the states. However, this reduces the robustness of the solver and can often lead to stalled convergence. Another approach is to increase the second-difference artificial dissipation constant, κ_2. This improves convergence by reducing the spatial gradient in the states but has the disadvantage

of reducing solution accuracy. To accelerate convergence while preserving accuracy, we propose a dissipation-based continuation (DBC) method where κ_2 starts high and is continuously reduced as the solution converges.

DBC was previously explored by Hicken et al. [128, 129]. The DBC approach presented here differs from this prior work in that we do not use it as a general globalization strategy. We retain the shock sensor (Eq. 3.6) for the second-difference dissipation, so DBC is only applied near shocks. In addition, we use DBC in combination with pseudo-transient continuation for globalization. A related but less automated approach is to start the solution with a first-order scheme and switch to a second-order scheme at some point during the solution [104, 105].

3.5.1 DBC formulation

We write the DBC formulation as

$$\kappa_2^{(n)} = \kappa_2 + f_c^{(n)} \kappa_2^c, \tag{3.28}$$

where $\kappa_2^{(n)}$ is the dissipation constant at nonlinear iteration n, κ_2 is the desired final dissipation constant, κ_2^c is the initial value of the additional dissipation, and $f_c^{(n)}$ is the continuation parameter at iteration n. We require that the continuation parameter starts at 1 and approaches 0 as the solution converges. We choose to compute the continuation parameter using a generalized sigmoid function:

$$f_c^{(n)} = \frac{1}{1 + e^{-\sigma\left(\log_{10}\left(\eta_{\text{rel}}^{(n)}\right)+\lambda\right)}}, \tag{3.29}$$

where σ is the sharpness parameter, λ is the midpoint parameter, and $\eta_{\text{rel}}^{(n)}$ is the relative convergence at iteration n. The parameters σ and λ determine the shape of the sigmoid. Larger values of σ will result in a steeper drop in dissipation. We call λ the midpoint parameter because the additional dissipation is equal to $0.5\kappa_2^c$ at a relative residual of $10^{-\lambda}$. We use $\sigma = 3$ and $\lambda = 3$, which results in the continuation function shown in Fig. 3.15. When the solution is converged, the additional dissipation is small enough to recover the solution of the original problem. The shape parameters generally do not need to be tuned for each case. However, the appropriate value for κ_2^c depends on the Mach number. We find that $\kappa_2^c = 0.2M_\infty$ works well, and we use this value for all cases presented here.

3.5.2 NACA 0012

We first compare the baseline solver and DBC for the NACA 0012 case at Mach numbers from 0.75 to 3.0 at a 3° angle of attack (Fig. 3.16a). DBC is between 3% and 13% faster than baseline for Mach numbers of 2.0 and above. Similarly, DBC is between 4% and 8% faster than baseline at

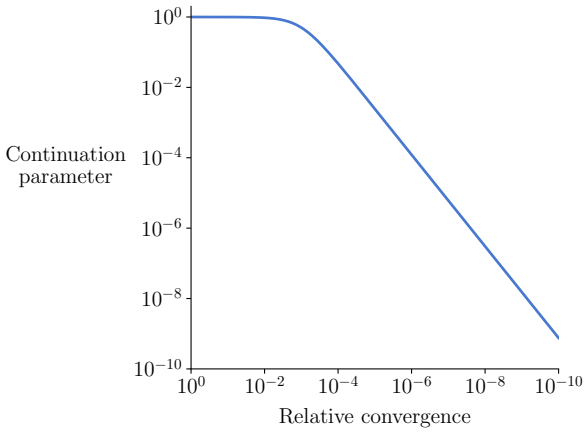
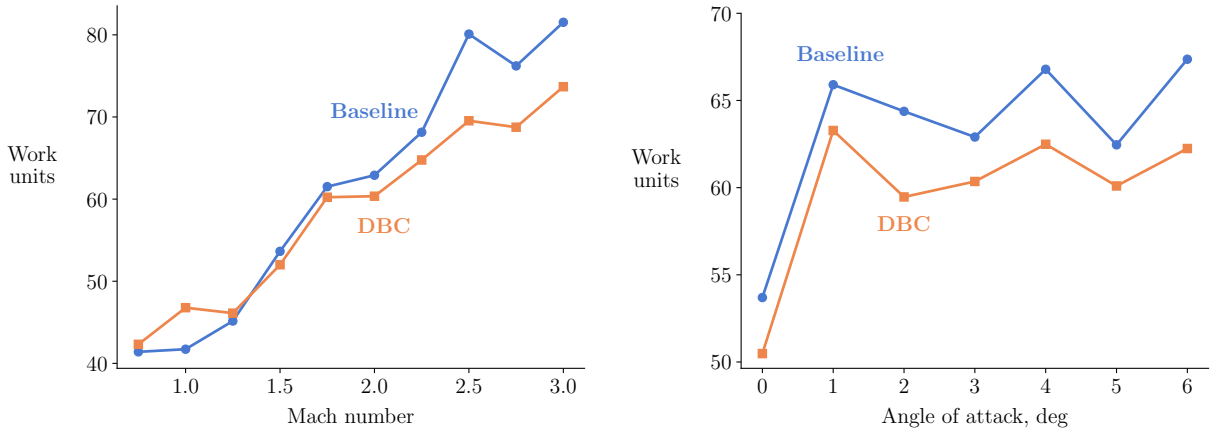


Figure 3.15: The additional dissipation decreases smoothly as the flow converges

Mach 2.0 and angles of attack from 0° to 6° (Fig. 3.16b). DBC increases the number of near-unit steps, which improves convergence (Fig. 3.17). The converged lift and drag values with DBC match the baseline values to 8 significant digits.



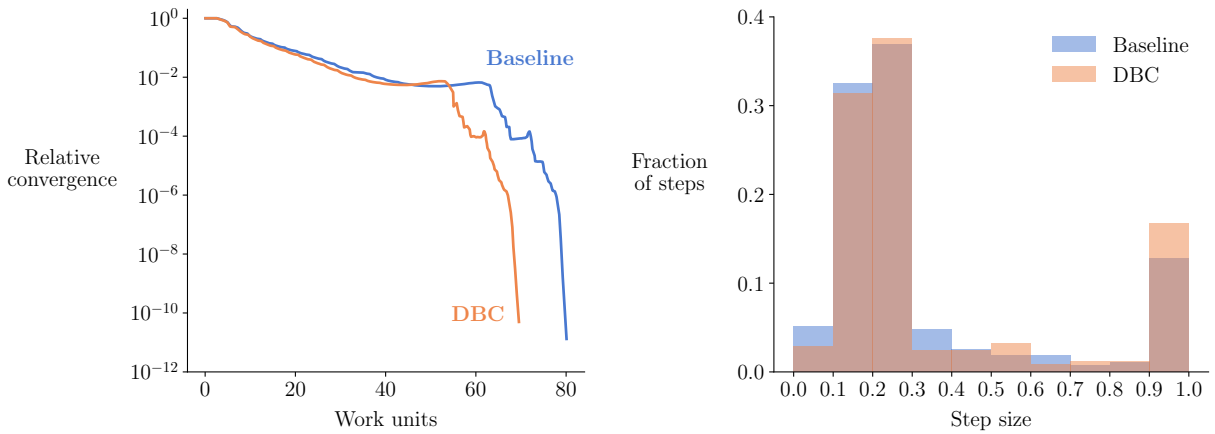
(a) Varying Mach at $\alpha = 3°$

(b) Varying α at Mach 2

Figure 3.16: DBC offers slight speedup for the NACA 0012 case

3.5.3 Supersonic transport configuration

The advantages of DBC are more pronounced when applied to a more complex case. We now consider a supersonic transport (SST) configuration with a cranked-arrow wing, T-tail, and canard (Fig. 3.18). We consider flow at Mach numbers of 1.8 and 0.95, representing realistic supersonic and



(a) (b)

Figure 3.17: DBC improves convergence by increasing step sizes (NACA 0012 at Mach 2.5)

transonic cruise conditions for the next generation of SSTs [15]. Both cases have a 4° angle of attack. DBC converges both cases, whereas the baseline solver does not converge either (Fig. 3.19). We also compare DBC to the baseline solver with κ_2 increased from the default value of 0.25 to the initial DBC value of $0.25 + 0.2M_\infty$. For Mach 1.8, increasing κ_2 converges faster than DBC. However, this solution is also less accurate because of the increased dissipation near shocks. For Mach 0.95, the solver still stalls with the higher κ_2, albeit after about three more orders of convergence.

In some cases, robustness is more valuable than faster convergence. The SST configuration analyzed here is the same as the one we use in Chapter 4 for aerodynamic shape optimization studies. Reliably converging the flow at different design points is essential for a well-behaved optimization [56]. DBC offers more robustness than the baseline solver for complex geometries at high Mach numbers, making it more suitable for many-query scenarios such as parameter sweeps or design optimization.

3.6 Conclusions

Using compressible flow solvers for low Mach number flows typically results in poor accuracy and speed. In addition, high Mach number flows can be challenging to converge reliably without increasing the amount of dissipation in the spatial discretization. We present three contributions to address these challenges.

First, we propose a simple modification to the Jameson–Schmidt–Turkel scheme to make the artificial dissipation appropriate for low Mach number flows. We show that scaling down the acoustic contribution of the spectral radius improves accuracy and reduces sensitivity to mesh size. We

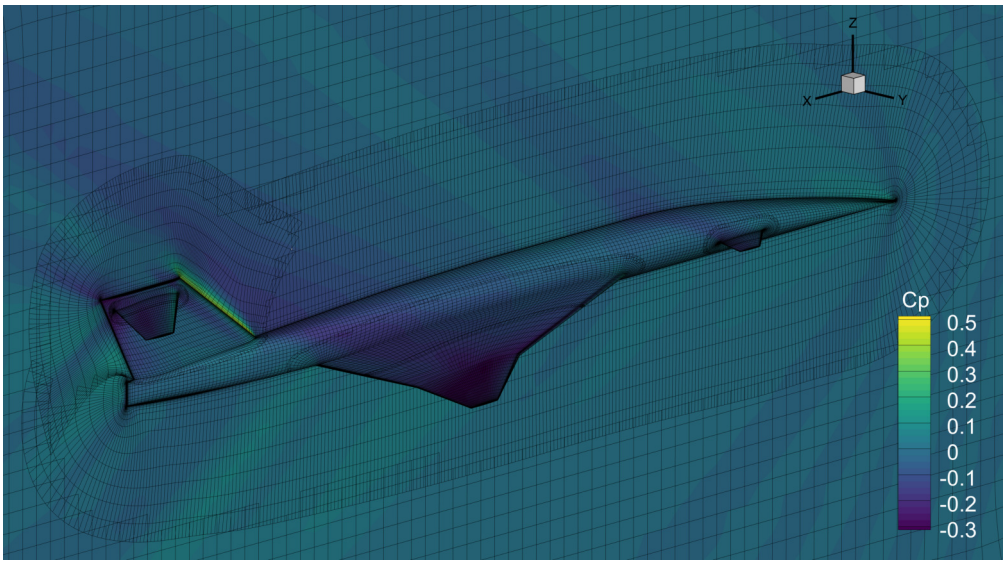
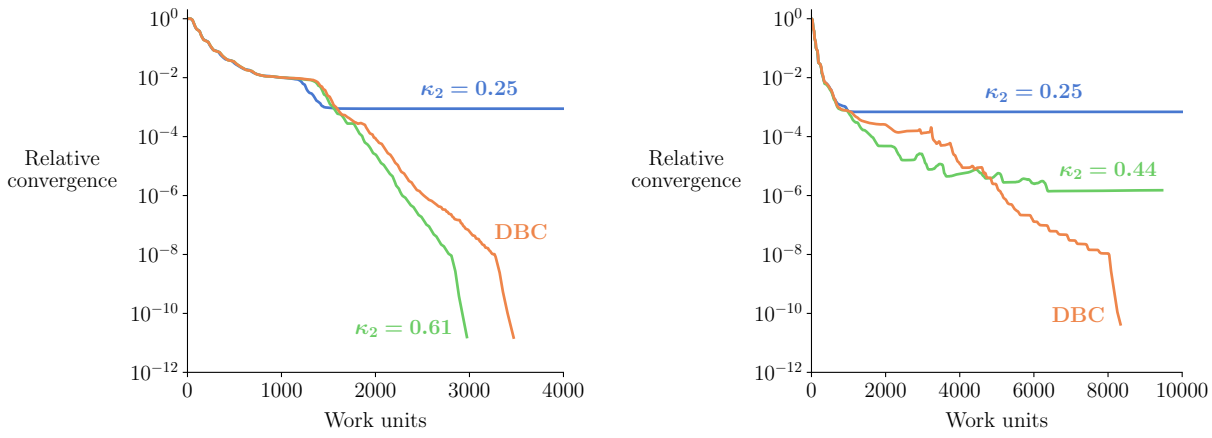


Figure 3.18: SST overset mesh with pressure contours at Mach 1.8



(a) Mach 1.8

(b) Mach 0.95

Figure 3.19: DBC converges the SST cases without sacrificing accuracy

recommend using a scaling factor equal to the freestream Mach number. Second, we demonstrate the effectiveness of a characteristic time-stepping method for approximate Newton–Krylov solvers. This approach reduces the stiffness of the linear system at low Mach numbers by modifying the time-step matrix. We show that the artificial dissipation must be adequately scaled to achieve speedup with characteristic time-stepping. In addition, cases with small regions of stagnation point flow do not benefit from characteristic time-stepping. Third, we present a dissipation-based continuation method for flows with shocks. The continuation approach introduces additional dissipation near shocks during the initial stages of the solution and smoothly reduces the dissipation as the solution converges. This approach is faster and more robust than the baseline solver, particularly for more complex geometries. These contributions enable more accurate and efficient flow solvers across low and high Mach numbers, making CFD more dependable for applications involving a wide range of flow conditions, such as aircraft design.

Chapter 4

Aerodynamic Shape Optimization with a Subsonic Static Margin Constraint

Chapters 2 and 3 dealt with CFD methods relevant for supersonic transport (SST) design. The rest of this dissertation applies some of the contributions from prior chapters to the aerodynamic design optimization of a full SST configuration. We start with aerodynamic shape optimization of an SST where we minimize supersonic drag while maintaining subsonic stability.

4.1 Introduction

SSTs must be designed with high-speed and low-speed flight regimes in mind. Efficiency and stability across different flight conditions are often competing objectives. For example, double-delta or cranked-arrow wings are known for providing a balance between supersonic and transonic performance. However, the same wings often exhibit inadequate pitch stability at low-speed, high-angle-of-attack conditions [16]. Designing for subsonic stability is complicated by the leading-edge vortices that are characteristic of flow over highly swept wings at high angles of attack [44].

The three main contributors to the pitch-up of cranked-arrow wings are a strong inboard leading-edge vortex, vortex breakdown, and outboard flow separation [17]. A strong inboard leading-edge vortex causes an increase in lift on the forward part of the wing as the angle of attack increases. Vortex breakdown causes a loss of lift on the aft part of the wing as the angle of attack increases. Outboard flow separation also results in a loss of lift on the aft part of the wing. This is the only pitch-up mechanism that is not caused by leading-edge vortices. All three of these effects result in increased nose-up moment. The dominant pitch-up mechanism is dependent on the wing planform [17]. Airfoil shape also has a strong influence on the leading-edge vortices. Kulfan [60, 61] observed that greater wing thickness delays the growth of leading-edge vortices and increases pitch stability. Similarly, Nelson [16] found that blunt leading edges and increased leading-edge thickness delay subsonic pitch-up at the cost of increased supersonic wave drag. This represents a fundamental trade-off in SST design.

Computational fluid dynamics (CFD) combined with numerical optimization presents the opportunity to resolve flow features across different flight conditions and capture the trade-offs between competing design objectives. Several authors have demonstrated the value of applying multipoint optimization to supersonic aircraft. Cliff et al. [50] minimized a weighted sum of the thrust required for an SST design at supersonic and transonic conditions using the Euler equations as the aerodynamic model. Carrier [130] also used the Euler equations to minimize supersonic and transonic drag for an SST with the drag at the two conditions being equally weighted in the objective function. Sasaki et al. [131] used the thin layer Navier–Stokes equations to optimize supersonic wings for an objective that included supersonic and transonic drag. More recent work has turned to the Reynolds-averaged Navier–Stokes (RANS) equations instead of the Euler or thin layer Navier–Stokes equations. Bons et al. [132] performed aerostructural optimization of a supersonic business jet where the objective was to maximize a weighted sum of the supersonic and transonic range. Mangano and Martins [133] optimized the shape of a trapezoidal wing considering drag at subsonic, transonic, and supersonic Mach numbers. Despite considering multipoint performance trade-offs, none of these papers considered stability in the optimization problem.

CFD-based optimizations with stability considerations have focused mainly on subsonic aircraft. The most common approach is to apply a static margin constraint at a cruise condition. This approach has been used in the optimization of conventional transport aircraft [134], flying wings [135], and blended or hybrid wing-body aircraft [136–138]. Lee et al. [139] optimized a hybrid-wing-body aircraft with static margin constraints at a Mach 0.78 cruise condition and a Mach 0.2 low-speed condition after finding that constraining only the cruise static margin resulted in unacceptable low-speed stability. Mader and Martins [140] developed a method to compute static, dynamic, and transient stability derivatives using the time spectral method. They used this method to study the effect of stability constraints on the optimal flying wing design at Mach numbers of 0.5, 0.7, and 0.85 [135]. Lee and Kim [141] used optimization to improve the pitch stability of a lambda wing at low-speed, high-angle-of-attack conditions. Similar to supersonic wing designs, the flow over the lambda wing at these conditions is characterized by leading-edge vortices. By maximizing the drop in pitching moment between 0 and 13 deg angles of attack at Mach 0.2, they showed that the optimized planform delays pitch-up by moving the leading-edge vortices further outboard.

A few papers have included low-speed constraints in the optimization of supersonic configurations, albeit with lower-fidelity methods. Dudley et al. [18] performed aerodynamic and structural optimization to minimize the takeoff gross weight of an SST subject to practical design constraints. Some of these were low-speed constraints such as limiting the angle of attack at landing and preventing engine scrape at landing. In a follow-up paper, Crisafulli et al. [19] augmented the vortex-lattice method used for stability derivative computation in their framework with a pitch-up estimation method [17] to capture nonlinear pitch-up characteristics in the model. However, subsonic

pitch stability was not included as a constraint in either of these two papers. The papers also primarily looked at planform effects and lacked the fidelity and scalability for detailed airfoil shape parameterization. More recently, Li and Geiselhart [142, 143] optimized an SST using a sequence of optimization problems to enforce low-boom constraints and mission constraints, including static margin requirements at landing and takeoff (LTO). However, the LTO analysis was done using a linear aerodynamic code. In addition, they did not explicitly consider subsonic pitch-up or the trade-off between supersonic performance and subsonic stability.

The goal of this work is to use RANS-based optimization to study the effect of aerodynamic shape on the subsonic pitch stability of an SST and to quantify the supersonic drag penalty associated with enforcing a subsonic stability constraint. We first describe the aircraft geometry and evaluate the accuracy of RANS at low-speed, high-angle-of-attack conditions in Sec. 4.2. We then present the optimization formulation and optimized aircraft designs in Sec. 4.3.

4.2 Validation at subsonic conditions

4.2.1 Aircraft geometry

The aircraft we are interested in is a variant of the UW-S-20A model developed by Nelson et al. [20] that excludes the nacelles. The aircraft has a cranked-arrow wing, a T-tail, and a canard (Fig. 4.1). This configuration was designed as a baseline model to study the low-speed characteristics of SSTs. The flight scale dimensions for the wing are listed in Table 4.1. This is the scale we use for the optimizations in Sec. 4.3. The wind tunnel model is scaled down by a factor of 22. This is the scale we use for the validation study in Sec. 4.2.3. The reference point for the pitching moment is at 25% of the mean aerodynamic chord.

Table 4.1: Flight scale wing dimensions

Quantity	Value
Reference area	373.03 m^2
Mean aerodynamic chord	14.565 m
Half-span	14.760 m

4.2.2 CFD solver and meshes

The CFD solver we use is ADflow [77], a finite volume code for multiblock and overset structured meshes. We use ADflow to solve the compressible RANS equations with the Spalart–Allmaras (SA)

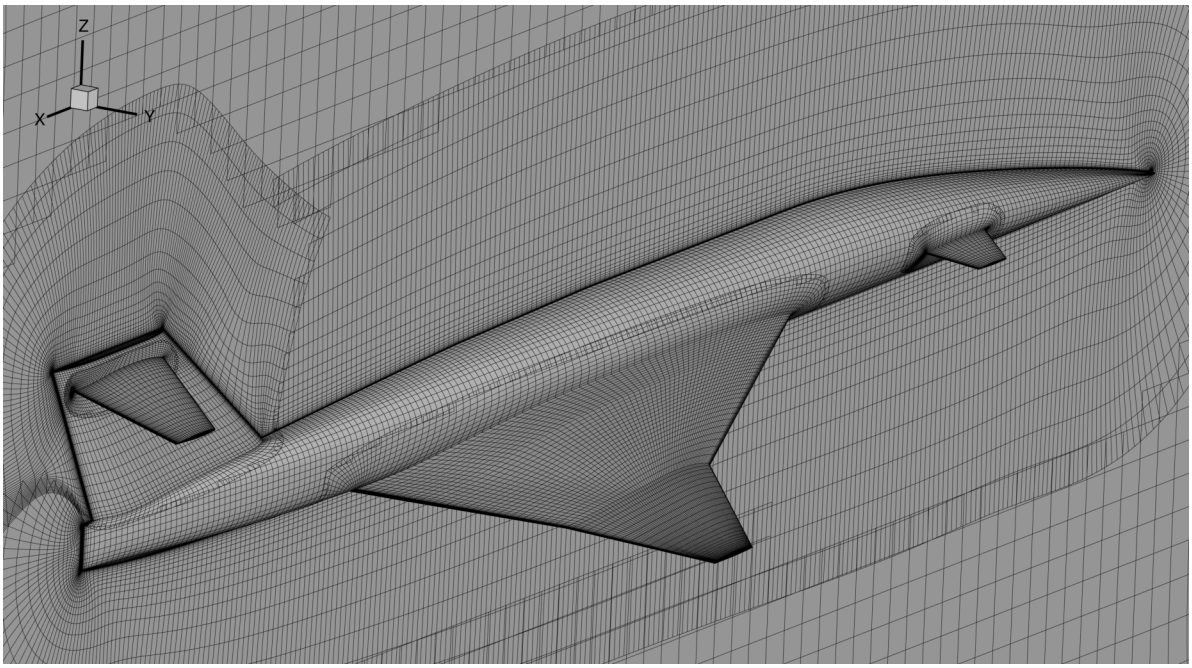
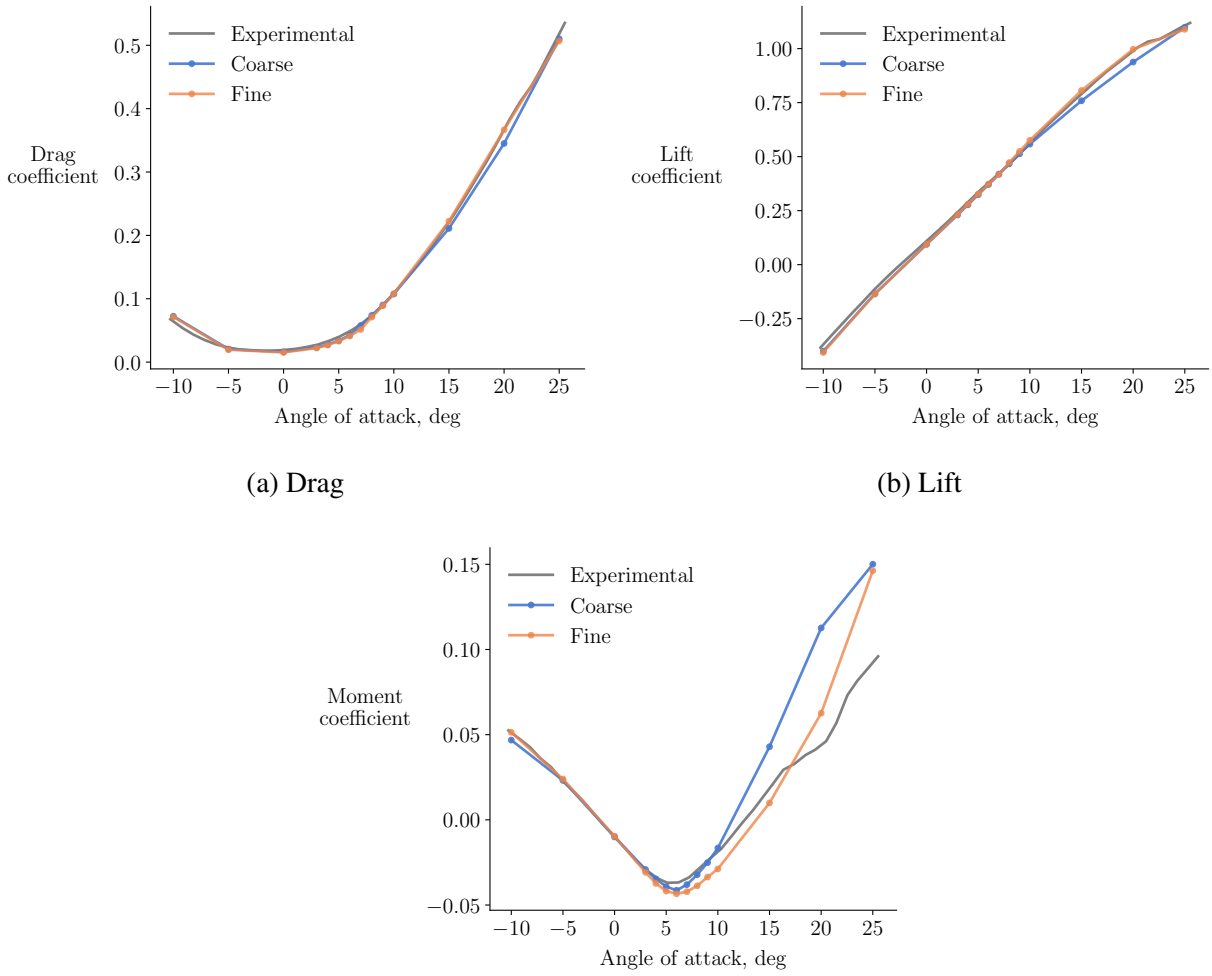


Figure 4.1: Coarse overset mesh of the supersonic transport configuration considered in this work

turbulence model [78]. We generate an overset mesh for the aircraft by first creating multiblock component and collar surface meshes [144]. We then extrude the surface meshes to generate volume meshes using pyHyp [81], a hyperbolic mesh generation code based on the work of Chan and Steger [82]. Finally, we combine the component meshes with a background mesh to form the complete overset mesh. The background mesh consists of a Cartesian mesh in the overlapping region near the aircraft, surrounded by a hyperbolic O-mesh. The farfield boundary of the O-mesh extends to 50 times the mean aerodynamic chord away from the Cartesian region. We use a zipper mesh approach [145] to account for overlapping regions on the surface mesh when computing integrated quantities such as drag. The total number of cells and the number of compute cells after implicit hole cutting [83] for the two overset meshes used in the validation study are listed in Table 4.2. The coarse overset mesh, shown in Fig. 4.1, is a uniformly coarsened version of the fine mesh. We also generate a version of the coarse mesh with added refinement on the wing to resolve flap hinges and interfaces. More details on this mesh are presented in Sec. 4.3.3.

Table 4.2: CFD mesh characteristics

Mesh	Total cells	Compute cells
Coarse	2,497,986	2,105,258
Coarse + flaps	4,505,598	4,024,248
Fine	19,983,888	17,014,902



(a) Drag

(b) Lift

(c) Pitching moment

Figure 4.2: Comparison of ADflow force and moment coefficients with experimental data

4.2.3 Comparison between RANS and experimental data

We use wind tunnel data from Ting et al. [21] to check whether our aerodynamic model is suitable for the low-speed, high-angle-of-attack flows relevant for the subsonic pitch-up of the SST we are interested in. Figure 4.2 shows a comparison between RANS results and the wind tunnel data for angles of attack from −10 deg to 25 deg. We use a Reynolds number based on the mean aerodynamic chord of 2.2 million and a Mach number of 0.15 to match the wind tunnel conditions. To simplify the overset meshing, we exclude a small wing-fuselage fairing that is present in the wind tunnel model.

All RANS solutions are converged to a total residual of 10^{-8} or tighter relative to the freestream

residual. The fine mesh provides an excellent match for the lift and drag across the entire angle of attack range. The coarse mesh is nearly as accurate from -10 to 10 deg but is slightly less accurate at higher angles of attack. The discrepancies between the experiment and CFD are most pronounced for the pitching moment. However, we are mostly interested in capturing the shape of the pitching moment curve, especially around the pitch-up angle, because this determines the pitch stability of the aircraft. The coarse and fine meshes both overpredict the pitch-up angle by about one degree. The fine mesh predicts the shape of the pitching moment more accurately up to 15 deg, whereas the coarse mesh predicts a sharper increase in moment after pitch-up onset. Overall, the coarse mesh provides reasonable accuracy at about 2% of the computational cost of the fine mesh. The results for the coarse mesh with refinement at the flaps are nearly identical to the results for the coarse mesh without the additional refinement, so they are omitted from the plots for clarity. We use the coarse mesh with refinement at the flaps for all optimizations.

The accuracy of steady RANS for this case is a result of the pitch-up mechanism. The pitch-up mechanism for this configuration is not vortex breakdown or outboard flow separation because there is no loss of lift associated with the pitch-up. The lift is still increasing linearly at the angle of attack where the pitching moment curve changes slope. This suggests that the dominant pitch-up mechanism is inboard vortex strengthening, which is primarily a steady effect unlike vortex breakdown or outboard flow separation. We showed in Ch. 2 that RANS is accurate for steady vortex flows.

4.3 Aerodynamic shape optimization

We use the MACH-Aero framework to perform aerodynamic shape optimization. MACH-Aero has been used extensively for aerodynamic shape optimization of wings and full aircraft configurations [56]. We briefly describe each component of the framework here. As described in Sec. 4.2, we use ADflow as the CFD solver. We converge the RANS equations to a total residual of 10^{-10} relative to freestream using the approximate Newton–Krylov solver [80] in ADflow. We also use the dissipation-based continuation method described in Sec. 3.5 to improve solver robustness for the supersonic flight condition. The discrete adjoint implementation in ADflow allows for efficient computation of gradients with respect to many design variables [146]. The adjoint solution is challenging to converge tightly for the mesh and flow conditions considered in this work. We run the adjoint solver until the linear residual convergence flattens out. This typically occurs after converging to a residual between 10^{-4} to 10^{-6} relative to the initial linear residual. In Sec. 4.3.5, we show that this results in accurate gradients despite the relatively loose convergence.

We parameterize geometry changes using free-form deformation (FFD) [147], implemented in pyGeo [148]. We use a component-based approach to deform the surface mesh, which we present

in Sec. 4.3.2. The changes to the surface mesh are propagated through the volume mesh using an inverse-distance mesh deformation algorithm [149], implemented in IDWarp [81]. Both pyGeo and IDWarp are differentiated so that they can be used in gradient-based optimization.

For the optimization, the analysis and gradient computation are wrapped with pyOptSparse [150], which provides a Python interface to different optimizers. The optimizer we use is SNOPT [151], a sequential quadratic programming algorithm designed for large-scale problems.

4.3.1 Optimization problems

We are interested in evaluating the supersonic drag penalty associated with enforcing a subsonic stability constraint. To formulate this optimization problem, we first define a supersonic cruise condition and a subsonic takeoff condition (Table 4.3). These flight conditions are based on flight envelopes and weight estimates from previous SST studies [15, 152]. The target lift coefficients, C_L^*, are computed from estimates of the takeoff weight for the subsonic condition and the midcruise weight for the supersonic condition.

Table 4.3: Flight conditions

	Mach number	Altitude, m	Altitude, ft	Reynolds number	C_L^*	$\alpha_{\min}$	$\alpha_{\max}$
Supersonic	1.8	16,764	55,000	80.4×10^6	0.1665	$-3°$	$3°$
Subsonic	0.3	0	0	101.8×10^6	0.6933	$-5°$	$20°$

We formulate supersonic drag minimization problems without a subsonic stability constraint (Table 4.4) and with a subsonic stability constraint (Table 4.5). For the stability-constrained optimization, we run cases with target static margin values of 0%, 5%, and 10%. Each flight condition has separate angle of attack, tail rotation, and canard rotation variables to trim the aircraft. For the moment computation, we assume that the center of gravity (CG) is fixed at 25% of the mean aerodynamic chord. The angle of attack at the supersonic condition is limited to ±3 deg based on cabin deck angle requirements at cruise [153]. We also constrain the total angle for the trim surfaces to ±20 deg. This avoids separated flow conditions where RANS could be inaccurate and unrealistic designs where the trim surfaces are stalled. The subsonic condition has leading-edge and trailing-edge flap deflection variables because we expect future SSTs to deploy flaps at high-lift conditions. The fuselage design variables and the wing design variables other than the flap deflections are shared across the supersonic and subsonic conditions. The optimization without the subsonic stability constraint excludes the subsonic design variables and constraints. We further discuss the geometric design variables and constraints in Sections 4.3.2 and 4.3.3 and the stability constraint in Sec. 4.3.4.

Table 4.4: Supersonic drag minimization

		Quantity	Lower	Upper	Scaling
minimize	C_D	1			100
with respect to	Angle of attack	1	$\alpha_{\min}$	$\alpha_{\max}$	0.1
	Tail rotation	1	$-9°$	$9°$	0.1
	Canard rotation	1	$-6°$	$14°$	0.1
	Wing twist deformation	7	$-10°$	$5°$	0.1
	Wing sectional shape	84	−2 m	2 m	1
	Wing leading-edge thickness	7	0 m	2 m	1
	Wing vertical displacement	1	0 m	0.1 m	1
	Fuselage shape	78	-0.5 m	0.5 m	1
	Total number of design variables	**180**			
subject to	C_L / C_L^*	1	1	1	1
	C_M	1	0	0	1
	Wing thickness / Baseline wing thickness	90	1		1
	Wing volume / Baseline wing volume	1	1		1
	Fuselage thickness / Baseline fuselage thickness	72	0.8	1.2	1
	Fuselage volume / Baseline fuselage volume	1	1		1
	Tail rotation + Angle of attack	1	$-20°$	$20°$	–
	Canard rotation + Angle of attack	1	$-20°$	$20°$	–
	Total number of constraints	**168**			

4.3.2 Component-based geometry parameterization

To parameterize the geometry, we split the aircraft into five components: the fuselage, wing, canard, horizontal stabilizer, and vertical stabilizer. We define FFD volumes for each component, shown in Fig. 4.3a. The black spheres are FFD control points, and the blue spheres are reference axis control points. The reference axes are used to define design variables that act on groups of FFD control points. The horizontal stabilizer and canard reference axes are used to define the tail and canard rotation variables, respectively. Both rotation variables rotate the entire component about the mid-chord. The vertical stabilizer FFD volume is stationary but is needed to define and preserve the shape of the component.

The wing reference axis is used to define the twist variables, each of which rotates one spanwise section of the wing about the trailing edge, and the vertical displacement variable, which moves the entire wing in the z-direction. The optimized wing twist is equal to the twist deformation plus the existing 1.55 deg wing twist on the baseline design. The wing shape variables are defined by the movement of individual FFD control points in the z-direction. At the leading edge, we couple the upper and lower surface control points to move in opposite directions with equal magnitude. This allows the leading-edge thickness to change while avoiding shear deformations where the upper and

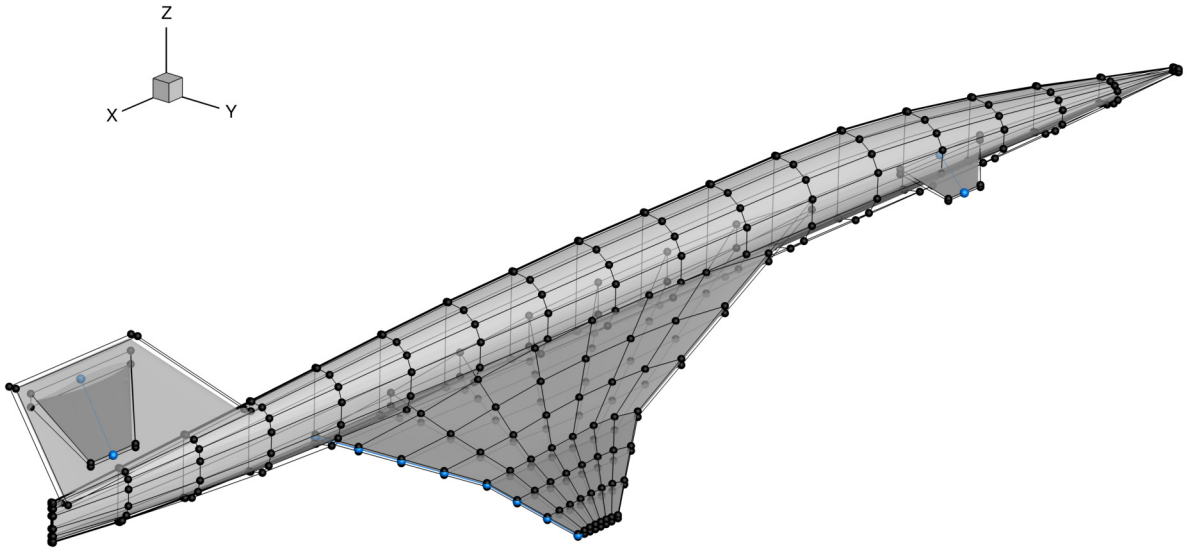
Table 4.5: Supersonic drag minimization with subsonic static margin constraint

		Quantity	Lower	Upper	Scaling
minimize	$C_{D,\,\text{supersonic}}$	1			100
with respect to	Angle of attack	2	$\alpha_{\min}$	$\alpha_{\max}$	0.1
	Tail rotation	2	$-9°$	$9°$	0.1
	Canard rotation	2	$-6°$	$14°$	0.1
	Wing twist deformation	7	$-10°$	$5°$	0.1
	Wing sectional shape	84	−2 m	2 m	1
	Wing leading-edge thickness	7	0 m	2 m	1
	Wing vertical displacement	1	0 m	0.1 m	1
	Fuselage shape	78	-0.5 m	0.5 m	1
	Inboard leading-edge flap deflection	2	$0°$	$15°$	0.1
	Outboard leading-edge flap deflection	2	$0°$	$30°$	0.1
	Inboard trailing-edge flap deflection	2	$0°$	$10°$	0.1
	Outboard trailing-edge flap deflection	2	$0°$	$20°$	0.1
	Total number of design variables	**191**			
subject to	C_L / C_L^*	2	1	1	1
	C_M	2	0	0	1
	$K_{n,\,\text{subsonic}} - K^*_{n,\,\text{subsonic}}$	1	0		1
	Wing thickness / Baseline wing thickness	90	1		1
	Wing volume / Baseline wing volume	1	1		1
	Fuselage thickness / Baseline fuselage thickness	72	0.8	1.2	1
	Fuselage volume / Baseline fuselage volume	1	1		1
	Tail rotation + Angle of attack	2	$-20°$	$20°$	–
	Canard rotation + Angle of attack	2	$-20°$	$20°$	–
	Total number of constraints	**173**			

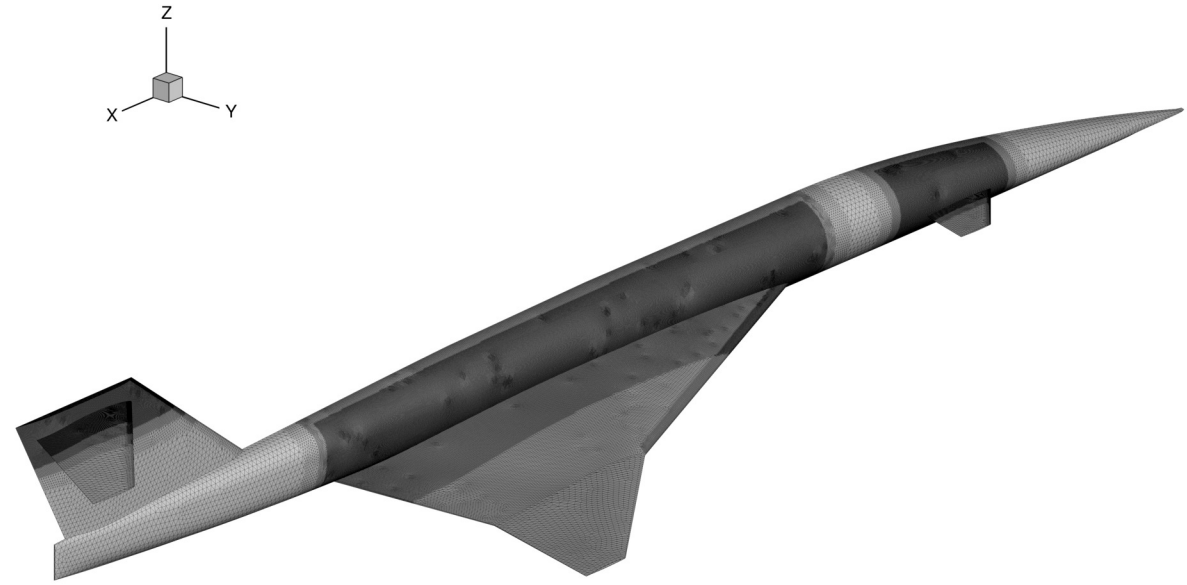
lower surface points move in the same direction to twist the wing. There are no shape variables at the trailing edge or the wing root, which is located inside the fuselage. There is also no twist variable at the wing root. We use wing thickness and volume constraints to ensure that the optimized wing is not unrealistically thin. The wing thickness constraints are applied on a grid of 9 points in the spanwise direction by 10 points in the chordwise direction.

The fuselage shape variables are deformations in the y-direction and are defined on the fuselage FFD control points starting from one section aft of the nose to the wing trailing edge. We apply fuselage thickness constraints on a grid of 12 points in the streamwise direction by 6 points in the vertical direction. These constraints prevent the optimizer from creating unrealistically large fuselage shape changes. We also approximate internal packaging constraints with a fuselage volume constraint that prevents the optimizer from reducing the total fuselage volume.

This component-based approach allows for geometry changes at component intersections. Away from intersections, the changes to the CFD surface mesh are defined solely by FFD. Near component



(a) Top-level FFD control points (black) and reference axes (blue)



(b) Triangulated surface meshes

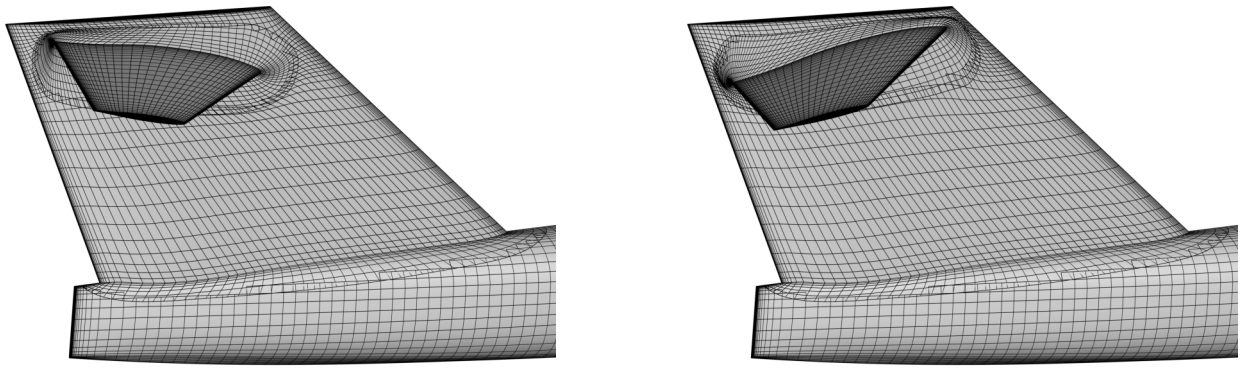
Figure 4.3: Component-based geometry parameterization

intersections, we use the surface mesh deformation method developed by Yildirim et al. [154] to maintain a valid CFD mesh. This method relies on triangulated surface meshes (Fig. 4.3b) and the pySurf module of Secco et al. [155] to compute intersections between components, perform projections, and remesh curves. Table 4.6 lists the number of nodes and triangles in each triangulated surface we use. The triangulated surfaces are refined in regions of high curvature and near component intersections.

Table 4.6: Triangulated surface characteristics

Component	Nodes	Triangles
Fuselage	242,436	482,773
Wing	96,162	191,239
Canard	58,348	116,443
Horizontal stabilizer	35,720	70,978
Vertical stabilizer	31,472	62,234

We demonstrate the effectiveness of the component-based parameterization using the tail rotation as an example. Figure 4.4 shows the T-tail surface mesh at the lower and upper bounds of the tail rotation variable. We obtain a valid mesh for rotation angles from -9 deg to $+9$ deg. In addition, the vertical stabilizer's shape is preserved even after applying large rotations to the horizontal stabilizer.



(a) **$-9°$** tail rotation

(b) **$+9°$** deg tail rotation

Figure 4.4: Lower and upper bounds for the tail rotation

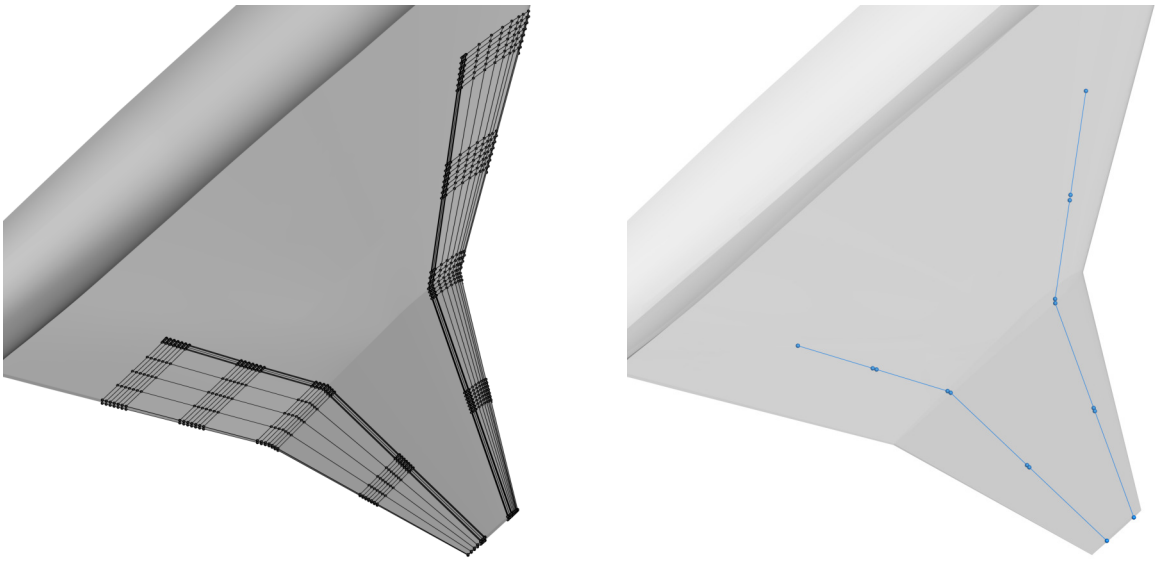
4.3.3 Wing flap deflections

The subsonic design variables include leading-edge and trailing-edge flap deflections. We use plain flaps, which are typically chosen for SST designs to avoid the supersonic drag penalty associated with flap track fairings [156]. The flap layout we use is loosely based on the layouts presented by Gaffuri and Brezillon [156] and Nicholls [157]. There are 4 leading-edge flaps and 4 trailing-edge flaps, with 2 each on the inboard and outboard parts of the wing. The leading-edge flaps cover 15% of the chord length. The outboard trailing-edge flaps cover 30% of the chord length. The length of the inboard trailing-edge flaps is nearly constant, which corresponds to 30% chord at the wing break and 15% chord at the most inboard section. The inboard flaps cover 13.2% of the wing span each. The outboard flaps cover 23.5% of the wing span each.

We approximate the deflected flaps as a continuous surface, similar to prior work on shape optimization with flap deflections [133, 136, 158]. However, the methodology presented here has a few improvements over prior work, particularly in ensuring that the CFD mesh resolves the deflected geometry. We use a hierarchical FFD approach [148] to define the flap rotations. The top-level wing FFD volume has two FFD volumes embedded inside it: one for the leading-edge flaps and one for the trailing-edge flaps (Fig. 4.5a). The wing FFD volume defines shape changes and the flap FFD volumes apply flap rotations on top of the wing shape changes. Reference axes define the hinge lines (Fig. 4.5b) and the rotations applied to the FFD control points. Positive rotations correspond to downward flap deflections.

Each flap has an associated reference axis control point on each spanwise end of the flap. At the spanwise interface between flaps, the deflection is interpolated between the two adjacent flap deflections. There are three spanwise layers of FFD control points on either side of an interface. We use these 'buffer layers' to smoothly transition between flap deflections. The sharpness of the transition can be controlled by changing the buffer layer spacing. Similarly, there are three buffer layers on either side of the hinge line. For a flap deflection of δ, most of the FFD control points associated with that flap rotate by δ. The exceptions to this are the FFD control points on the hinge line, which rotate by $\delta/2$, and the buffer layers towards the interior of the wing, which do not rotate. This leads to a smooth transition from the deflected to undeflected shape across the hinge line. In addition, the wing thickness at the hinge line is contracted by a factor of $1/\cos\delta$ to eliminate bumps at the hinge line.

The parameterization described above allows for precise deflections to be applied to the FFD volumes. To ensure that the CFD mesh retains the same precision, we refine the mesh at the hinge lines and spanwise interfaces (Fig. 4.6c). We demonstrate the flap parameterization using leading-edge flap deflections of 0, 15, 0, 30 deg and trailing-edge flap deflections of 10, 0, 20, 0 deg, where the flaps are ordered from inboard to outboard. This demonstrates large flap deflections and large changes between adjacent flaps. The wing mesh after applying these deflections (Fig. 4.6d) has



(a) FFD control points (b) Hinge line reference axes

Figure 4.5: FFD parameterization for the wing flaps

enough cells at the hinge lines and interfaces to resolve the curvature in the deflected FFD volumes (Fig. 4.6b).

4.3.4 Subsonic stability constraint

The stability constraint enforces a target static margin at the subsonic condition. We derive the equation for the static margin by starting from the definition of the moment coefficient about the CG ($C_{M_{\text{CG}}}$) in terms of the moment coefficient about the neutral point ($C_{M_{\text{NP}}}$):

$$C_{M_{\text{CG}}} = C_{M_{\text{NP}}} + C_L \cos\alpha (h_{\text{CG}} - h_{\text{NP}}) + C_D \sin\alpha (h_{\text{CG}} - h_{\text{NP}}), \tag{4.1}$$

where h_{CG} and h_{NP} are the longitudinal locations of the CG and the neutral point, respectively, normalized by the mean aerodynamic chord. Substituting the definition of the normal force coefficient

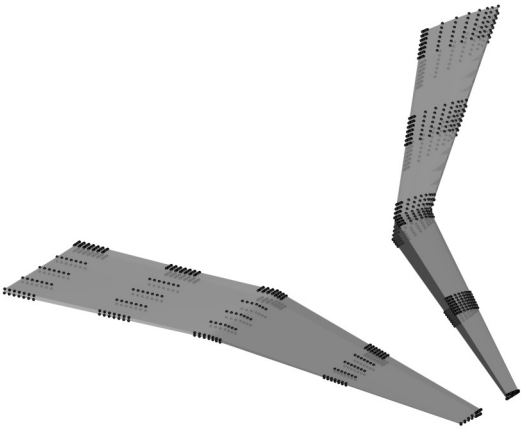
$$C_N = C_L \cos\alpha + C_D \sin\alpha, \tag{4.2}$$

we have

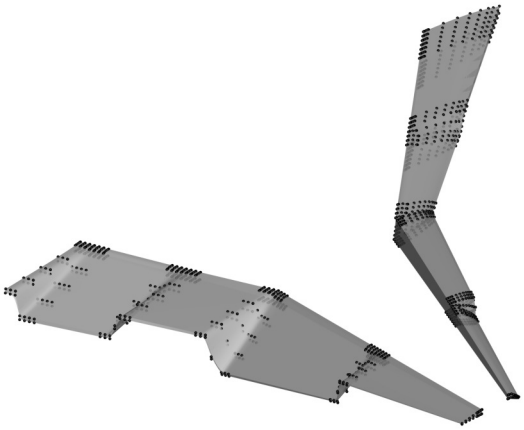
$$C_{M_{\text{CG}}} = C_{M_{\text{NP}}} + C_N (h_{\text{CG}} - h_{\text{NP}}). \tag{4.3}$$

Taking the derivative with respect to α gives

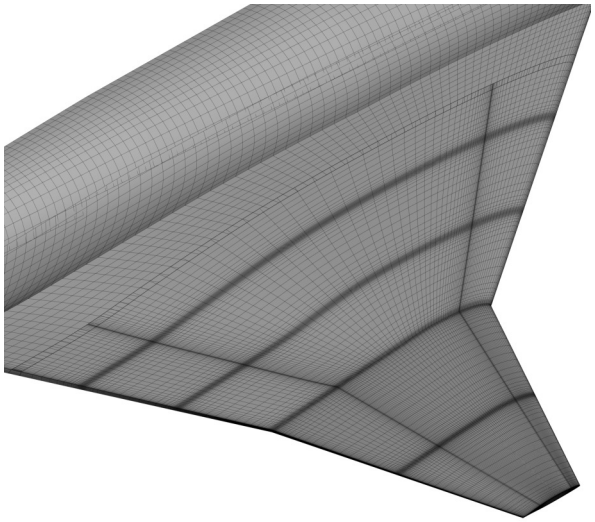
$$C_{M_\alpha} = C_{N_\alpha} (h_{\text{CG}} - h_{\text{NP}}). \tag{4.4}$$



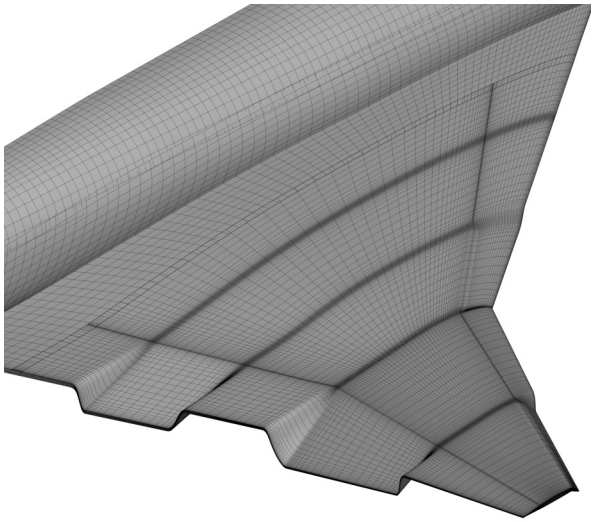
(a) Undeflected FFD control points and FFD volumes



(b) Deflected FFD control points and FFD volumes



(c) Undeflected surface mesh



(d) Deflected surface mesh

Figure 4.6: The deflected wing mesh has enough resolution to match the deflected FFD volumes

Substituting the definition of the static margin

$$K_n = h_{\mathrm{NP}} - h_{\mathrm{CG}}, \tag{4.5}$$

we can express the static margin in terms of aerodynamic force and moment derivatives:

$$K_n = -\frac{C_{M_\alpha}}{C_{N_\alpha}}. \tag{4.6}$$

In previous work on CFD-based aircraft optimization [134–138], the static margin has been computed as

$$K_n = -\frac{C_{M_\alpha}}{C_{L_\alpha}} \tag{4.7}$$

This approach assumes that the angle of attack is small and that the drag is much smaller than the lift, both of which are true for a typical cruise condition. At off-design conditions, such as a supersonic aircraft in subsonic flight, these assumptions may be false. Replacing the lift with the normal force includes the effect of drag at high angles of attack.

For the static margin constraint, we need to compute both the stability derivatives and their gradients. Similar to prior work, we use a finite-difference approach to compute the stability derivatives. For example, we compute the pitching moment derivative as

$$C_{M_\alpha} = \frac{C_M|_{\alpha+\Delta\alpha} - C_M|_\alpha}{\Delta\alpha}, \tag{4.8}$$

where $\Delta\alpha = 0.1^\circ$ is a small perturbation in the angle of attack. This requires two flow solutions. By computing this slope as a finite difference, we can leverage the adjoint implementation in ADflow to compute the gradient of C_{M_α} with respect to the design variables. More explicitly, the gradient is computed as

$$\frac{\mathrm{d}C_{M_\alpha}}{\mathrm{d}x} = \frac{\left.\frac{\mathrm{d}C_M}{\mathrm{d}x}\right|_{\alpha+\Delta\alpha} - \left.\frac{\mathrm{d}C_M}{\mathrm{d}x}\right|_\alpha}{\Delta\alpha}, \tag{4.9}$$

where x is the vector of design variables and $\mathrm{d}C_M/\mathrm{d}x$ is computed using the adjoint method. We apply the same procedure to the lift and drag to compute the normal force coefficient derivative and its gradient.

4.3.5 Gradient verification

Before running optimizations, we check that the adjoint gradients are accurate by comparing them to gradients computed using the complex-step method [159]. The complex-step method provides

exact gradients given a sufficiently small imaginary step size and tightly converged nonlinear solutions. We use an imaginary step size of 10^{-200} and converge the complex-valued flow solution to a total residual of 2×10^{-16} relative to the freestream residual. We check the gradient of the supersonic drag coefficient with respect to angle of attack, wing root twist, and wing tip twist variables (Table 4.7). These three design variables test the gradient accuracy without geometry changes, with geometry changes near a component intersection, and with geometry changes away from component intersections, respectively. The adjoint gradients match the complex-step gradients to 6 significant digits, which is consistent with the expected accuracy when using overset meshes with ADflow [146].

Table 4.7: The adjoint gradients match the complex-step gradients to 6 significant digits

	Angle of attack	Root twist	Tip twist
Complex step	$5.254738495556 \times 10^{-3}$	$7.103406534136 \times 10^{-4}$	$1.139548637609 \times 10^{-4}$
Adjoint	$5.254733424776 \times 10^{-3}$	$7.103408921134 \times 10^{-4}$	$1.139549476348 \times 10^{-4}$
Relative error	9.65×10^{-7}	3.36×10^{-7}	7.36×10^{-7}

4.3.6 Optimized designs

We run optimizations without the subsonic static margin constraint (K_n unconstrained) and with different target subsonic static margin values (0% K_n, 5% K_n, 10% K_n). We also run a pre-optimization to trim the baseline design. The pre-optimization involves minimizing supersonic drag with only the angle of attack, tail rotation, and canard rotation as design variables. This provides a fair baseline supersonic drag value to compare the optimized results against. We run similar trim optimizations on the baseline and K_n unconstrained designs at the subsonic condition to evaluate their subsonic static margins. The subsonic trim optimizations include the flap deflections as design variables. Table 4.8 shows the supersonic drag, subsonic static margin, and trim variables for the baseline and optimized designs. The table also shows the maximum constraint violation and optimality convergence for each optimization. All optimizations are converged well enough that we can be confident in the trends presented by the optimized designs.

The baseline design is unstable at the subsonic condition, which is expected because of the early pitch-up tendencies shown in Fig. 4.2c. The K_n unconstrained optimization decreases the drag from the baseline by 85.4 counts or 24.9% but is unstable with a more negative subsonic static margin than the baseline. This demonstrates the need for a low-speed stability constraint when performing supersonic shape optimization. The stability-constrained designs are all stable at the subsonic condition, showing that it is possible to use aerodynamic shape optimization to enforce

Table 4.8: Optimization results and convergence

	Baseline	K_n unconstrained	0% K_n	5% K_n	10% K_n
Constraint violation		1.7×10^{-7}	3.4×10^{-7}	1.7×10^{-6}	2.9×10^{-6}
Initial optimality		4.8×10^{-2}	4.7×10^{-2}	4.7×10^{-2}	4.7×10^{-2}
Final optimality		3.5×10^{-5}	9.9×10^{-5}	2.5×10^{-4}	2.5×10^{-4}
Supersonic					
Drag counts	343.66	258.26	259.55	260.25	260.45
Angle of attack	3.00°	3.00°	3.00°	3.00°	3.00°
Tail rotation	−2.60°	−2.92°	−0.17°	−0.21°	−0.22°
Canard rotation	13.5°	3.99°	4.03°	4.06°	4.07°
Subsonic					
Static margin	−10.9%	−18.9%	0.00%	5.00%	10.0%
Angle of attack	10.5°	11.0°	11.5°	12.1°	12.2°
Tail rotation	−4.97°	−6.13°	−2.83°	−1.54°	−1.31°
Canard rotation	8.91°	9.01°	8.46°	7.86°	7.77°
Inboard LE flaps	[15.0°, 15.0°]	[15.0°, 0.00°]	[5.08°, 15.0°]	[4.75°, 15.0°]	[4.68°, 15.0°]
Outboard LE flaps	[30.0°, 30.0°]	[0.00°, 5.16°]	[30.0°, 10.8°]	[26.6°, 22.2°]	[26.5°, 24.9°]
Inboard TE flaps	[10.0°, 10.0°]	[10.0°, 10.0°]	[10.0°, 10.0°]	[10.0°, 10.0°]	[10.0°, 10.0°]
Outboard TE flaps	[20.0°, 17.6°]	[20.0°, 0.00°]	[20.0°, 20.0°]	[20.0°, 3.07°]	[20.0°, 0.00°]

subsonic stability for supersonic aircraft design. Relative to the K_n unconstrained design, there is a supersonic drag penalty of 0.50% for a 0% static margin, 0.77% for a 5% static margin, and 0.85% for a 10% static margin (Fig. 4.7). The supersonic static margin is not considered in the optimizations because this would increase the computational cost and because we expect the aircraft to remain stable at the supersonic cruise condition. Computing the supersonic static margin for the baseline and optimized designs justifies this decision because the supersonic static margin is between 20% and 24% for all designs.

We now compare the baseline, K_n unconstrained, and 5% K_n designs. Figures 4.8 and 4.9 compare the wing cross-sections and pressure distributions. Figure 4.10 shows spanwise distributions and Fig. 4.11 shows streamwise distributions and cross-sections. The K_n unconstrained design primarily uses twist and fuselage shape to minimize the supersonic drag. The combination of decreasing the canard rotation, increasing wing twist at the root, and decreasing twist outboard of 30% of the wing span results in smoother spanwise and streamwise lift distributions, decreasing wave drag [16]. The fuselage shape also contributes to the drag reduction by increasing fuselage area near the wing leading and trailing edges and decreasing fuselage area in the middle of the wing and canard. Comparing the K_n unconstrained and 5% K_n designs reveals the design changes required to make the aircraft stable at subsonic conditions. One main difference is that the 5% K_n design uses larger leading-edge flap deflections. Leading-edge flap deflections reduce the local

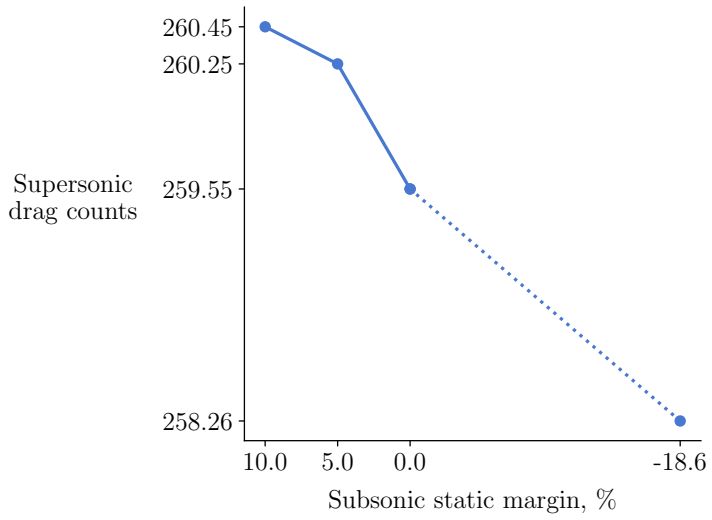


Figure 4.7: Pareto front of supersonic drag and subsonic static margin

angle of attack at the leading edge, which reduces leading-edge vortex strength [17, 160] and leads to a more stable aircraft. However, the leading-edge flap deflections alone are not enough to achieve the target static margin. This is demonstrated by the 5% K_n design having higher supersonic drag than the K_n unconstrained design. The increase in supersonic drag comes from increased wing thickness. The increase in thickness can be seen in the spanwise thickness-to-chord distribution. However, this only shows the changes in maximum thickness. Looking at the wing cross-sections shows that the 5% K_n design also has rounder leading edges. The increase in leading-edge radius is particularly noticeable at the 30% and 40% spanwise sections. At the subsonic condition, the larger leading-edge radius and wing thickness weaken the inboard leading-edge vortex and improve pitch stability [16, 60, 61]. The difference in vortex strength is shown by the upper surface pressure coefficient contours (Fig. 4.12b) and isosurfaces of the Lambda2 vortex criterion [161] (Fig. 4.13). The 5% K_n design has weaker leading-edge vortices on the inboard wing and near the midspan. The 5% K_n design makes up for the loss of lift by increasing lift further outboard.

The trailing-edge flaps are not as important as the leading-edge flaps for stability. The inboard trailing-edge deflections are at their upper bound for the stable and unstable designs. This increases the wing camber and allows the aircraft to trim at a lower angle of attack. The outboard trailing-edge flap deflections have large variations between designs, but this does have much of an effect on the flow. This is because the flow on the outboard trailing-edge region at the subsonic condition is almost entirely in the spanwise direction (Fig. 4.14), reducing the effectiveness of the outboard trailing-edge flaps at generating lift.

Next, we compare the 0% K_n, 5% K_n, and 10% K_n designs. These designs are very similar,

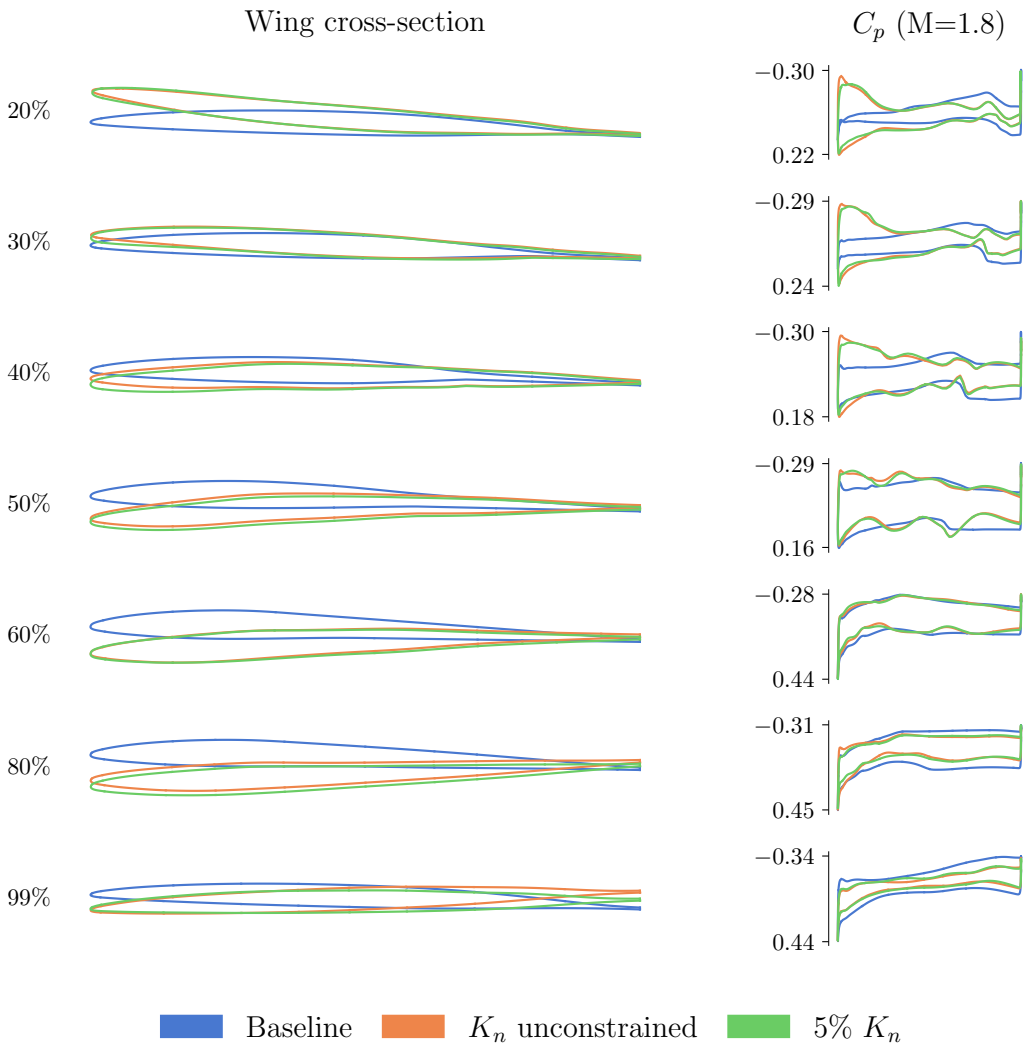


Figure 4.8: Supersonic wing cross-section and pressure distributions for baseline and optimized designs

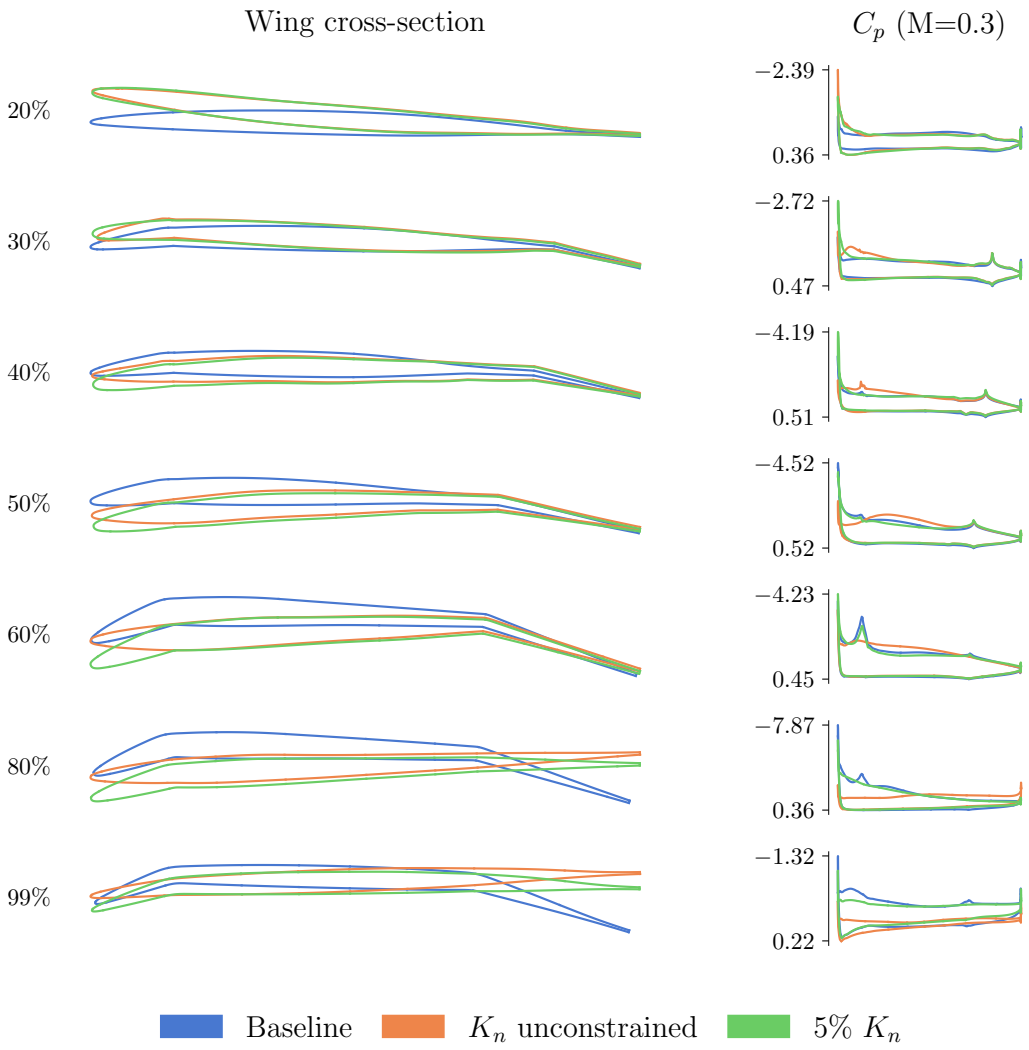


Figure 4.9: Subsonic wing cross-section and pressure distributions for baseline and optimized designs

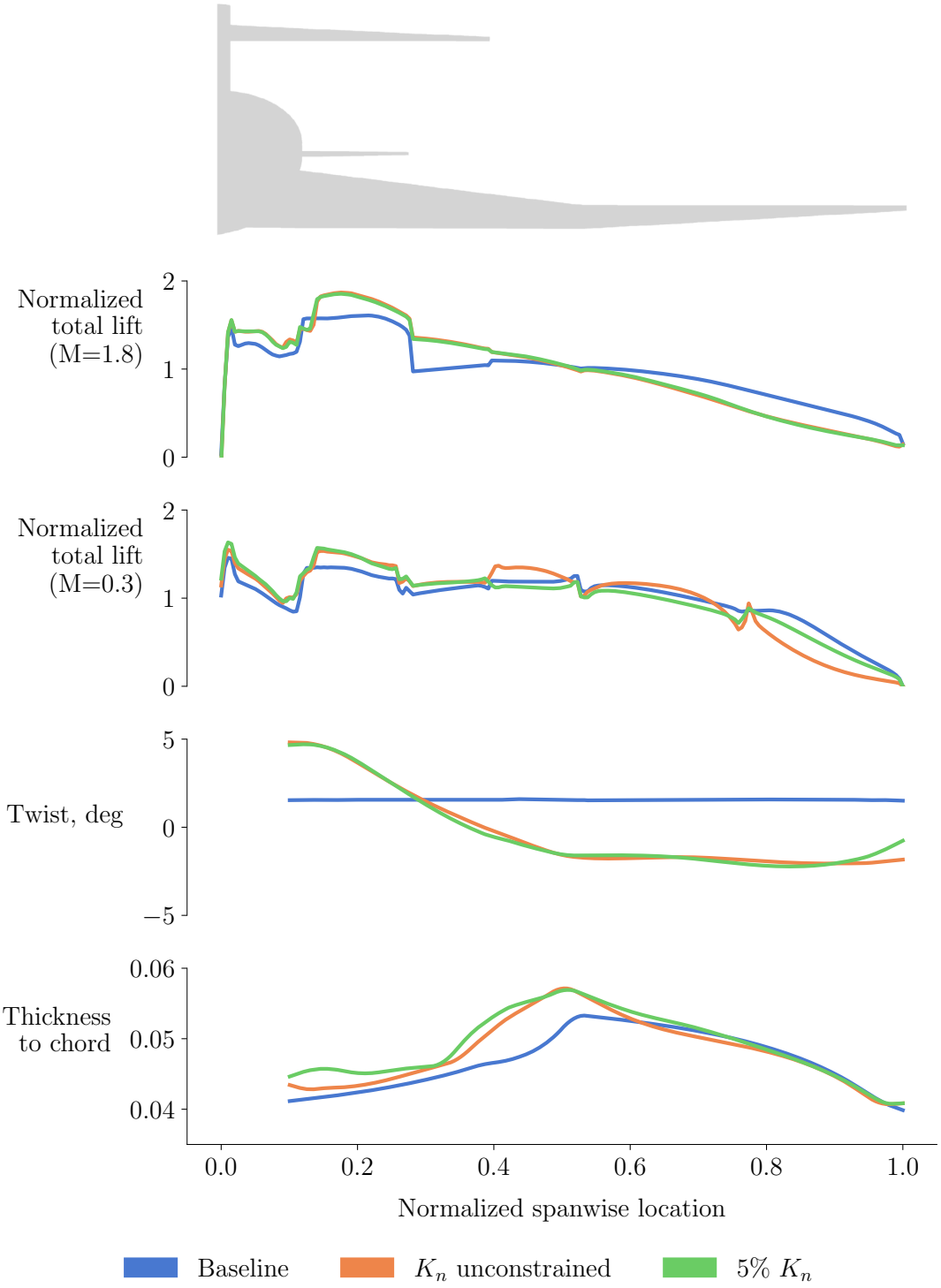


Figure 4.10: Spanwise distributions of total lift, wing twist, wing thickness for baseline and optimized designs

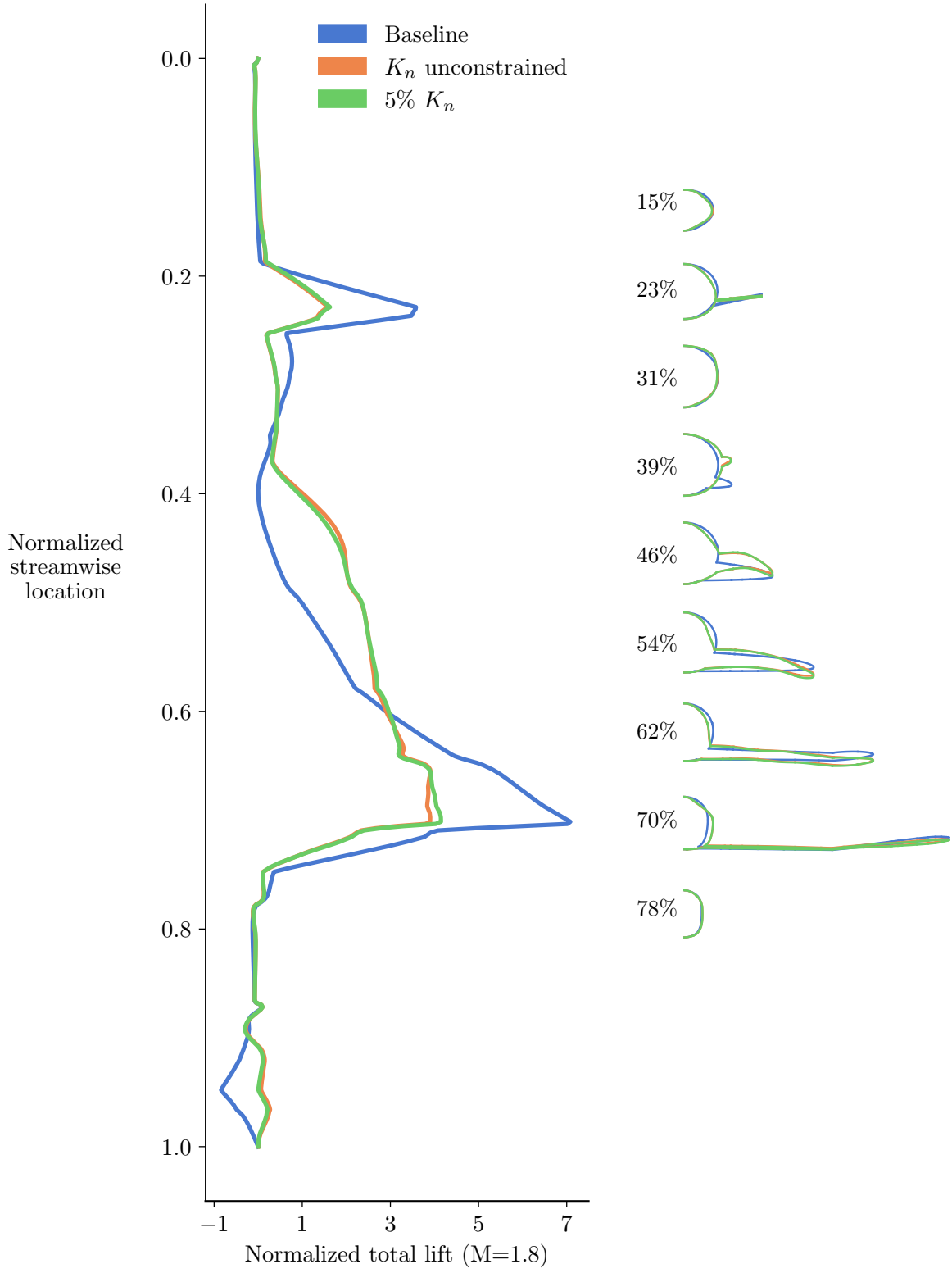
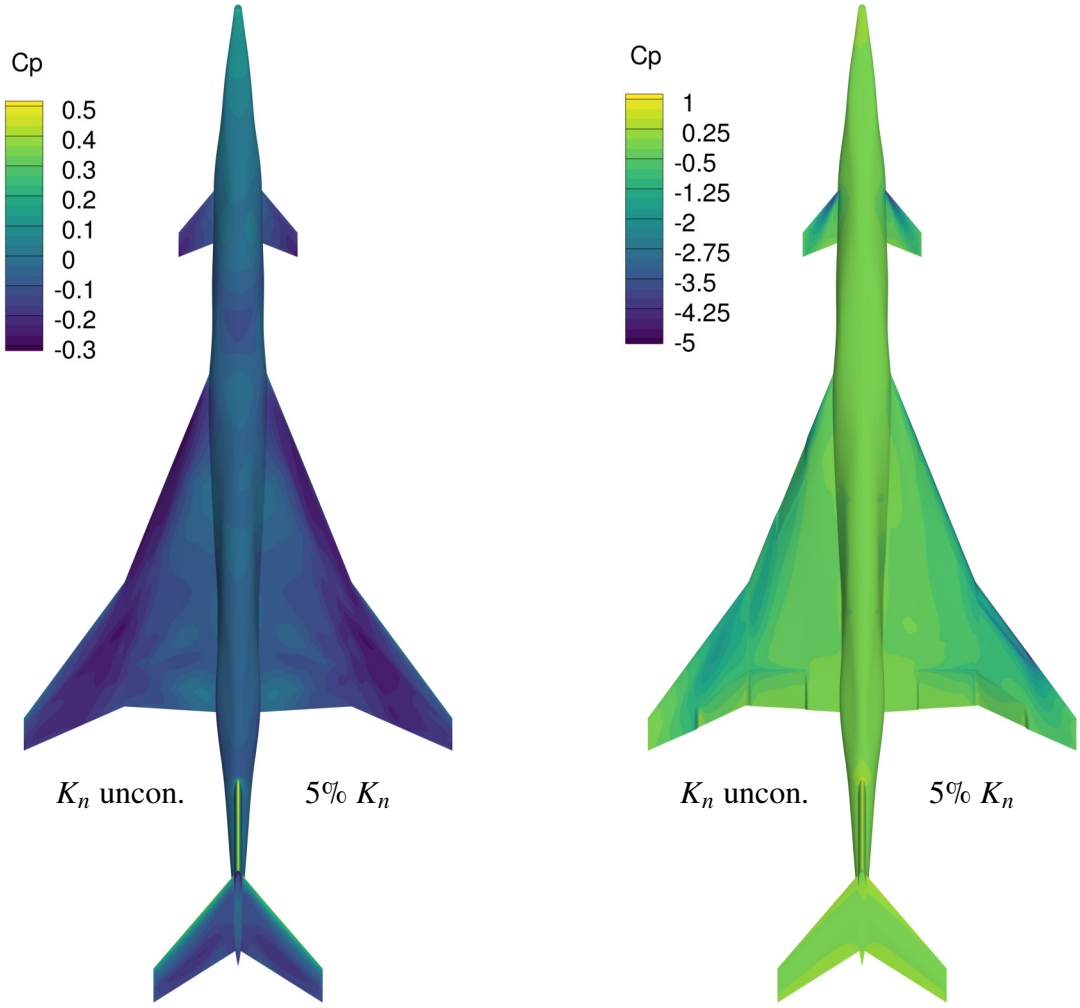


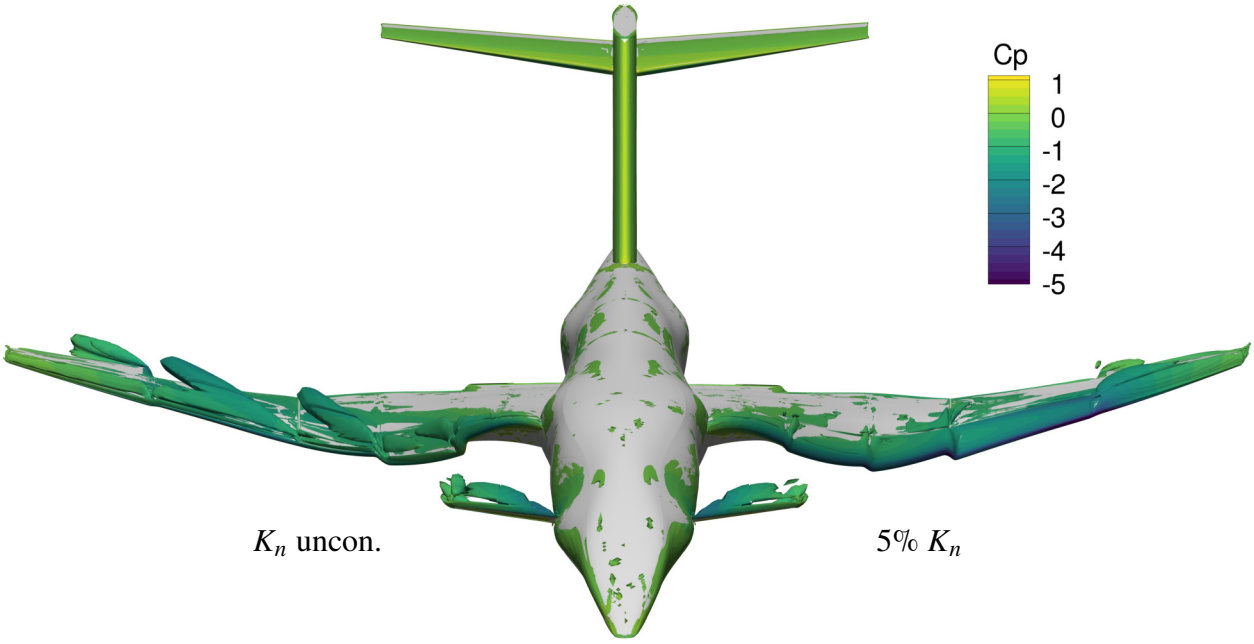
Figure 4.11: Supersonic streamwise lift distribution and cross-sections for baseline and optimized designs



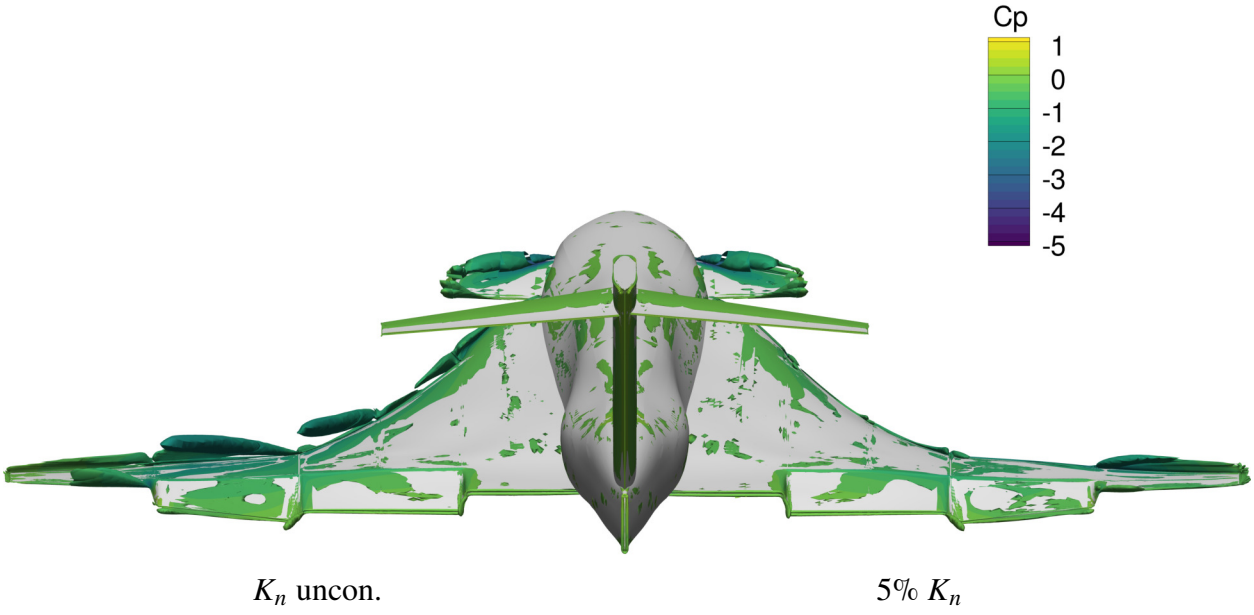
(a) Supersonic (M=1.8)

(b) Subsonic (M=0.3)

Figure 4.12: Upper surface pressure contours for the K_n unconstrained and 5% K_n designs



(a) Head-on view



(b) Tail view

Figure 4.13: Subsonic isosurfaces of $\lambda_2 = -0.2$ for the K_n unconstrained and 5% K_n designs

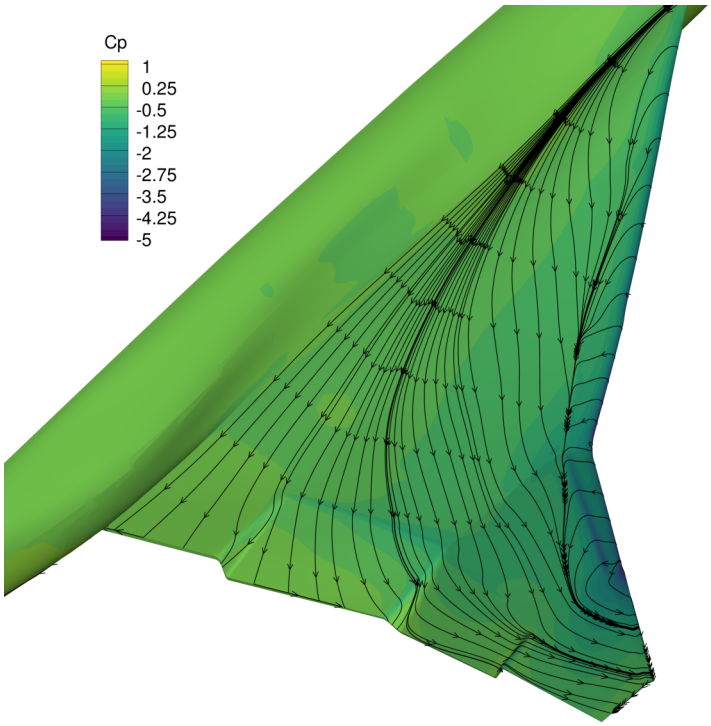


Figure 4.14: Subsonic streamlines on the 5% K_n wing

especially at the supersonic condition (Figs. 4.15, 4.16, 4.17, 4.18). However, we can identify consistent trends in the designs as the target static margin increases, which builds confidence in the optimization formulation and optimization convergence. Increasing the target static margin increases wing thickness and supersonic drag. The main difference at the subsonic condition is the outboard leading-edge flap deflections. The 10% K_n design uses a larger deflection on the most outboard leading-edge flap than the 0% K_n design, which results in a weaker outboard leading-edge vortex (Figs. 4.19b and 4.20). Both of these trends are consistent with the trends identified by comparing the K_n unconstrained and 5% K_n designs.

One limitation of these optimization results is the design variable bounds. Some of the active design variable bounds, such as the upper bound on supersonic angle of attack and lower bound on wing leading-edge thickness, are intentional constraints. However, some variable bounds, namely the upper and lower bounds on the fuselage shape variables and the upper bounds on the flap deflections, are set based on the maximum deformation that produces a valid mesh. Table 4.8 shows that between 4 and 7 of 8 flap deflections hit their upper bound, depending on the case. In addition, between 22 and 24 out of 78 fuselage shape variables are at a bound. Even the remaining design variables, which are not at a bound, are affected by the active bounds because the overall design space is restricted. However, we use the same bounds for all the optimizations, so we expect the design trends to be consistent with optimizations that are not restricted by the bounds. Moreover,

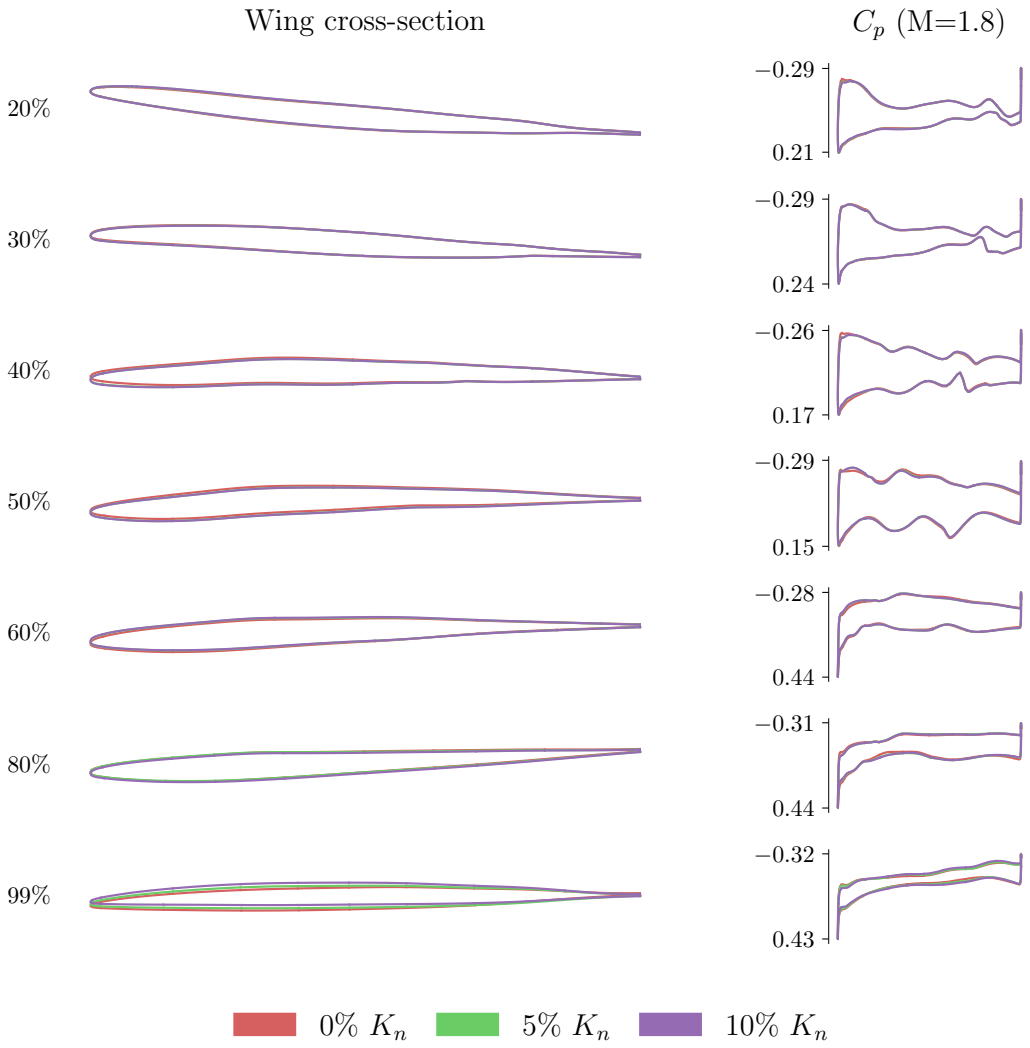


Figure 4.15: Supersonic wing cross-section and pressure distributions for stability-constrained designs

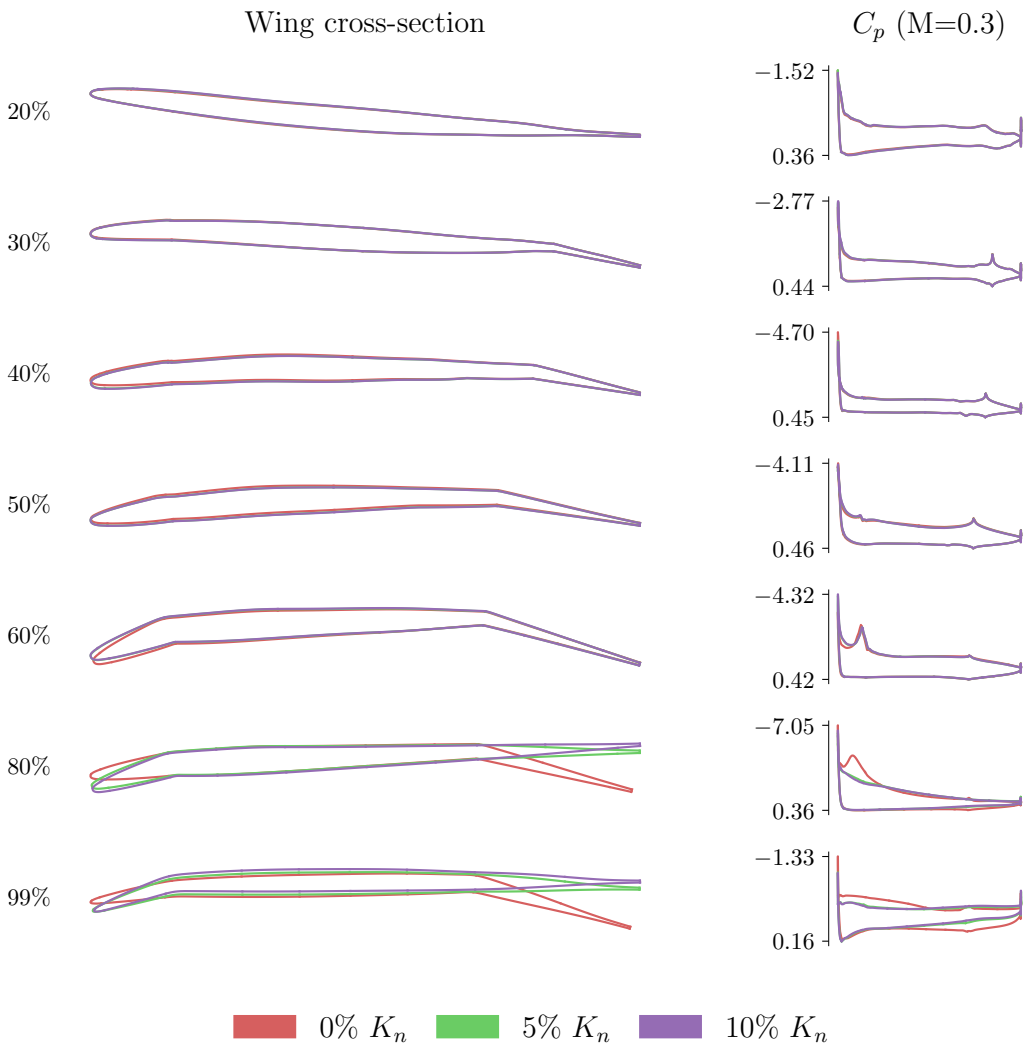


Figure 4.16: Subsonic wing cross-section and pressure distributions for stability-constrained designs

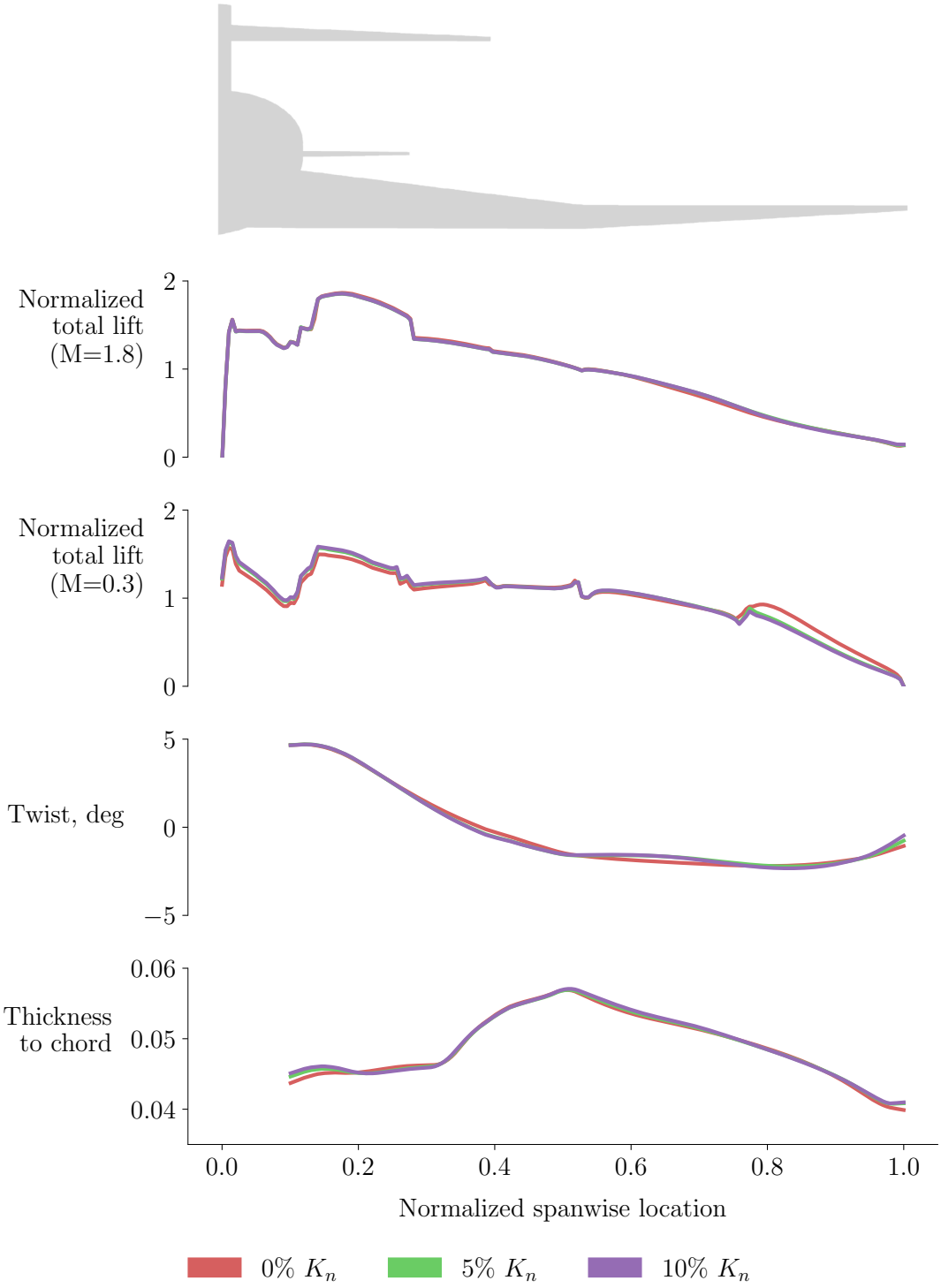


Figure 4.17: Spanwise distributions of total lift, wing twist, wing thickness for stability-constrained designs

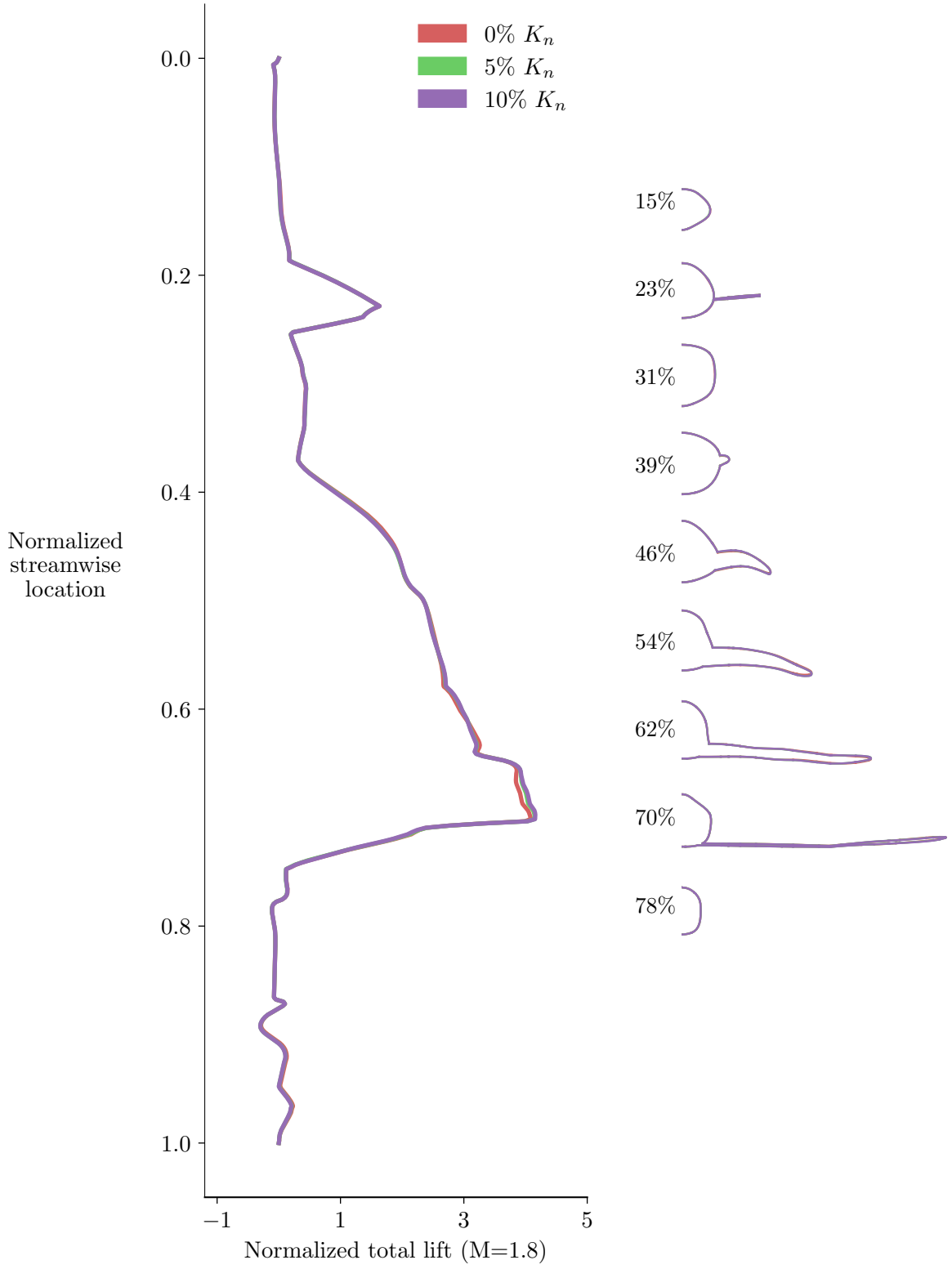
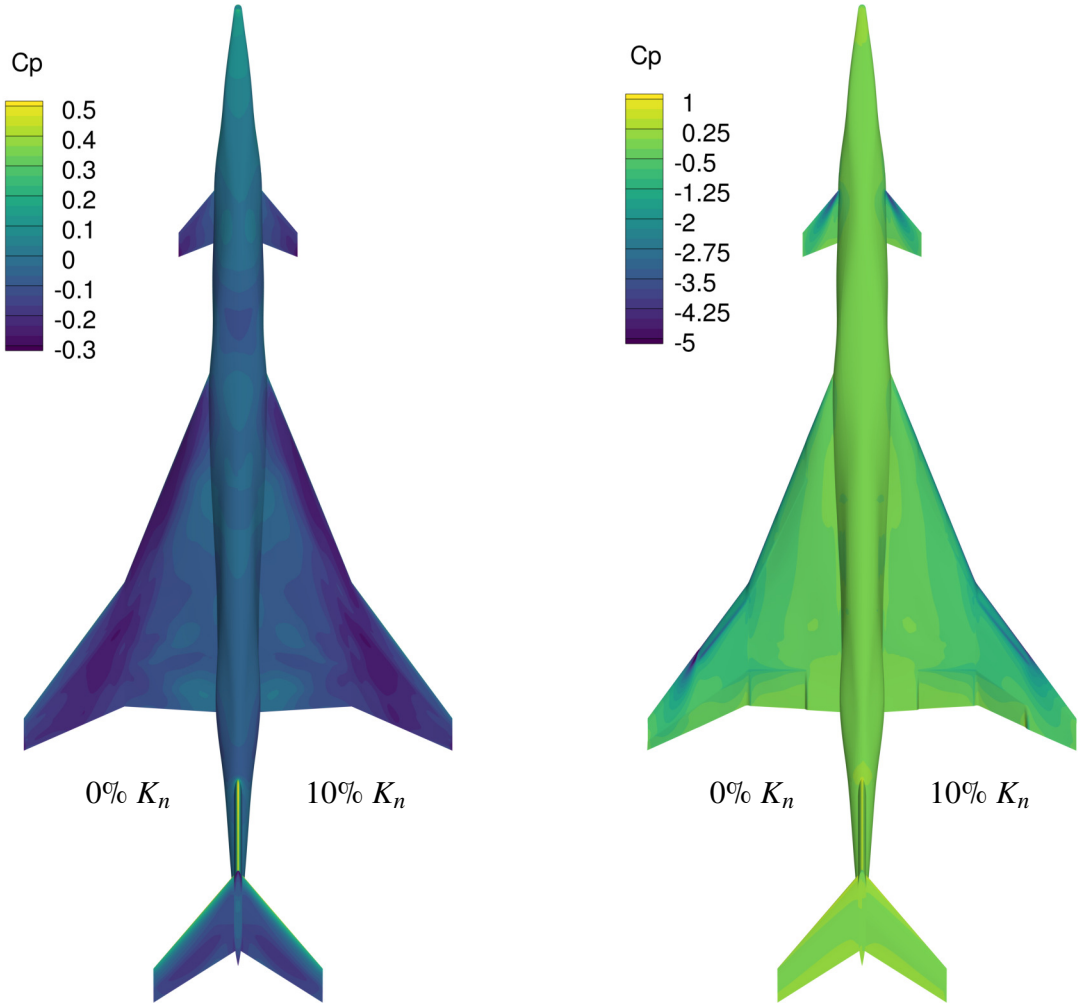


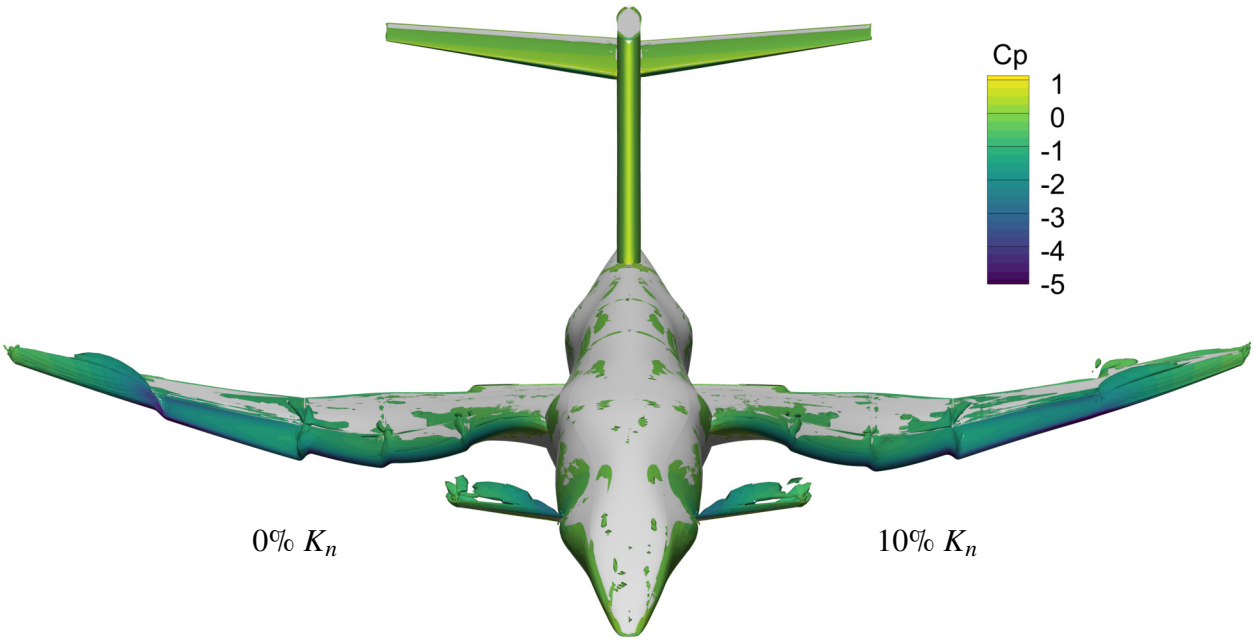
Figure 4.18: Supersonic streamwise lift distribution and cross-sections for stability-constrained designs



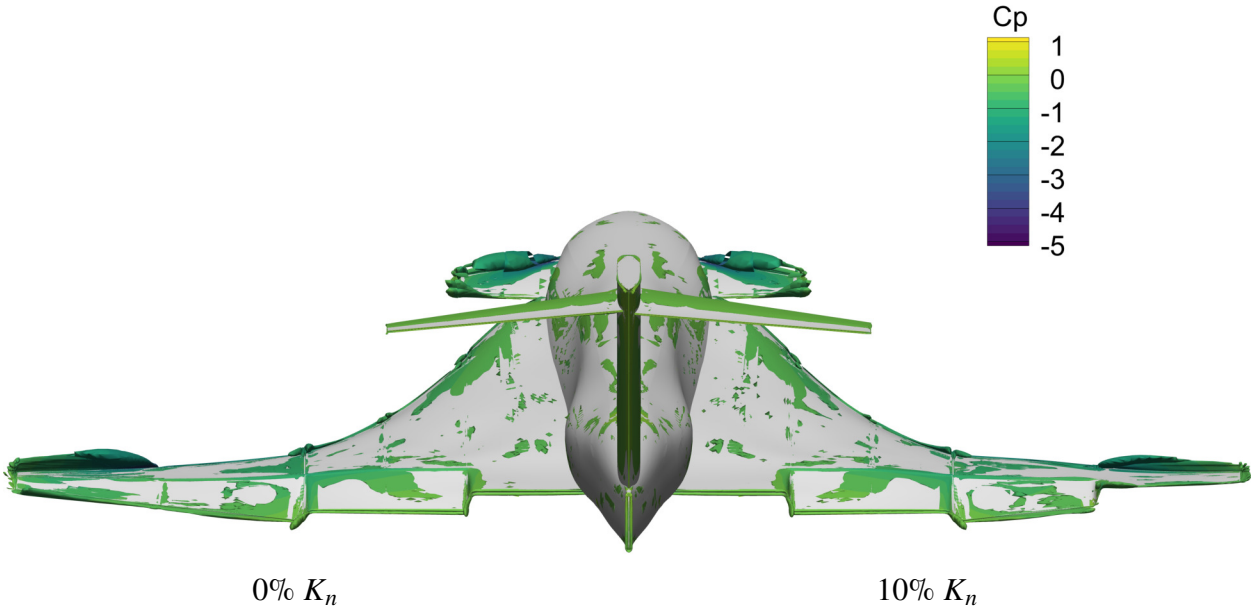
(a) Supersonic (M=1.8)

(b) Subsonic (M=0.3)

Figure 4.19: Upper surface pressure contours for the 0% K_n and 10% K_n designs



(a) Head-on view



(b) Tail view

Figure 4.20: Subsonic isosurfaces of $\lambda_2 = -0.2$ for the 0% K_n and 10% K_n designs

the 24.9% reduction in supersonic drag from the baseline to the K_n unconstrained design shows that the design space is large enough to make significant design changes.

4.4 Conclusions

In this work, we use RANS-based aerodynamic shape optimization to study the low-speed stability of a supersonic transport configuration. We show that RANS captures the lift, drag, and pitching moment trends of the supersonic configuration at low-speed, high-angle-of-attack conditions by comparing against experimental data. To parameterize the complete aircraft geometry, we use a component-based approach that works well with intersections. We minimize supersonic drag with and without a static margin constraint at a subsonic takeoff condition. The static margin computation differs from previous work on stability-constrained optimization by accounting for the effect of drag at high angles of attack. The baseline design and design optimized without the static margin constraint are both unstable at subsonic speeds. The stability-constrained optimizations have target static margins of 0%, 5%, and 10%. Pitch stability is achieved by deflecting the leading-edge flaps and increasing the wing thickness. The increase in wing thickness results in a supersonic drag penalty of 0.50% for a 0% static margin, 0.77% for a 5% static margin, and 0.85% for a 10% static margin. These results demonstrate that aerodynamic shape optimization is a valuable tool for designing SSTs accounting for supersonic performance and subsonic stability.

Chapter 5

Trim Surface Optimization

In Chapter 4, we optimized the wing and fuselage shape for a supersonic transport (SST) with a constant three-surface planform geometry. In this chapter, we consider configuration-level design changes. In particular, we use optimization to investigate different trim surface configurations for SST design and study the resulting trade-off between performance and stability.

5.1 Introduction

Three-surface aircraft have a theoretical advantage over canard and conventional two-surface configurations because three lifting surfaces allow the aircraft to achieve minimum induced drag for any center of gravity location [162]. However, practical considerations, such as viscous drag and stability, have limited the use of three-surface configurations for subsonic aircraft. Kroo [163] used a linear vortex-based method and an analytic viscous drag model to determine that three-surface designs have slightly lower drag than canard designs and slightly higher drag than conventional configurations. Similarly, Rokhsaz and Selberg [164] used a vortex lattice method to study trim for a general aviation aircraft and found that a three-surface configuration achieved a trimmed lift-to-drag ratio that is higher than a canard configuration but lower than a conventional configuration. Accounting for viscous drag reduced the performance difference between the three-surface and conventional configurations. As a result, they concluded that factors other than drag, such as structures or stability, must be considered to determine the best configuration. Other studies found that three-surface configurations can be advantageous in certain cases. Owens and Perkins [165] conducted wind tunnel experiments to compare three-surface and conventional configurations for a business aircraft with a forward-swept wing. They found that the three-surface configuration had better longitudinal control characteristics than the conventional configuration. Strohmeyer et al. [166] compared three-surface and conventional configurations for a transonic transport aircraft using a linearized potential flow solver with transonic corrections. They used optimization to design the canard for the three-surface configuration and found that the three-surface design had lower trim

drag than the conventional configuration but also had a smaller static margin.

Different trim configurations have also been studied for supersonic aircraft. Lacey [167] conducted transonic and supersonic wind tunnel tests on a fighter-type aircraft with different trim surface geometries. They found that the three-surface configuration had the lowest drag at high angles of attack but that the conventional configuration had lower drag at lower angles. In addition, the drag advantage of the three-surface configuration at high angles of attack decreased with increasing Mach number. Agnew and Hess Jr. [168] also ran wind tunnel tests on a three-surface and conventional fighter aircraft and found that the three-surface design had lower trim drag at Mach 0.9 and lower static margins at subsonic and supersonic conditions. The reduced static margin contributed to improved maneuverability. Covell [169] used supersonic wind tunnel tests to compare conventional, canard, and tailless fighter configurations. The tailless configurations used elevons for trim. They found that the tailless configurations had the lowest trim drag at low lift coefficients because of their low zero-lift drag. At higher lift coefficients, the conventional configurations had the lowest trim drag, followed by the canard configurations. In addition, they found that a linear aerodynamic model was not accurate enough to reliably compare canard, conventional, and tailless configurations at supersonic speeds. Finally, the Concorde was a tailless SST configuration that moved fuel to shift the center of gravity for trim [3]. This reduced trim drag in exchange for the added weight of the fuel transfer system.

There are two main contributions of this chapter. The first is studying different trim configurations for an SST rather than a fighter-type aircraft. The second is using nonlinear aerodynamic models for trim analysis and optimization. We determine the minimum trim drag for three-surface, canard, and conventional variants of an SST configuration with fixed trim surface sizing using the Reynolds-averaged Navier–Stokes (RANS) equations in Sec. 5.2. We then formulate a supersonic buildup model to optimize the aircraft with trim surface sizing variables in Sec. 5.3. The design space for this optimization includes the option to remove either trim surface and consequently achieve the best trade-off between parasite and induced drag. We also analyze the optimized trim surface configuration at a subsonic takeoff condition using RANS and compare its subsonic stability to the baseline planform in Sec. 5.4.

5.2 RANS-based trim optimization

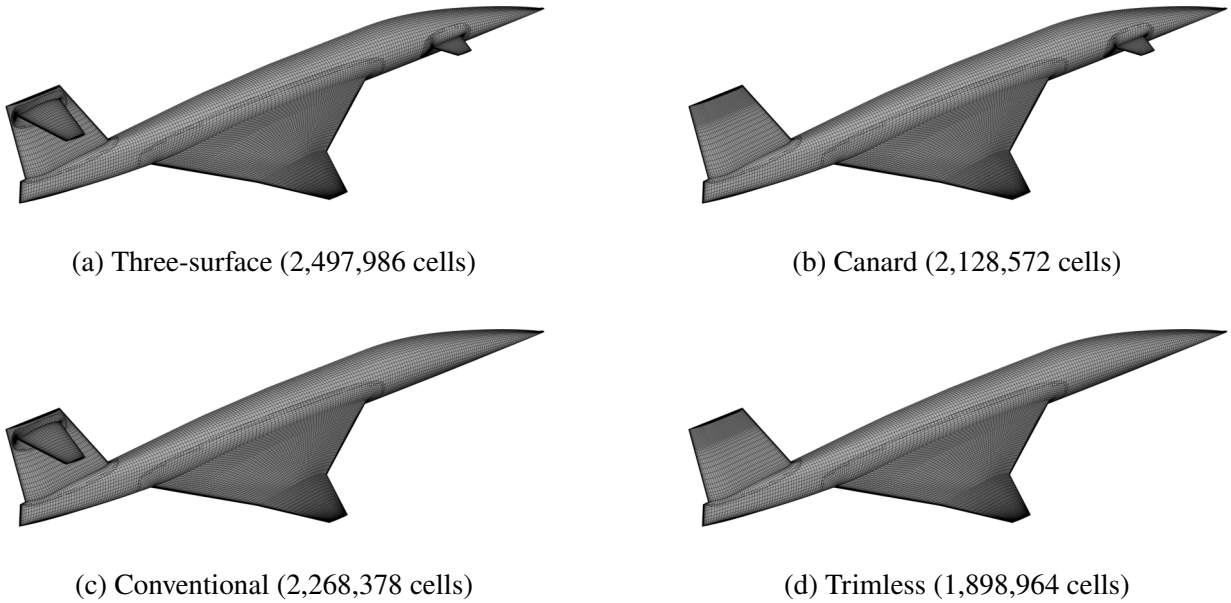
5.2.1 Aircraft geometry and flight condition

We first use RANS-based optimization to study trim for a constant geometry. The aircraft configuration we study is a variant of the UW-S-20A model developed by Nelson et al. [20] that excludes the nacelles (Fig. 5.1a). This aircraft is an SST with a cranked-arrow wing, a T-tail, and

a canard. This is the same configuration as described in Sec. 4.2.1. We use overset structured meshes to model the geometry in our RANS solver. One advantage of using overset meshes is that we can easily remove mesh blocks from the three-surface configuration to create meshes for canard, conventional, and trimless versions of the aircraft (Fig. 5.1). Some important dimensions are listed in Table 5.1. The supersonic cruise condition, based on flight envelopes from previous SST studies [15, 152], is chosen to be at a Mach number of 1.8 and an altitude of 55,000 feet or 16,764 meters. The target lift coefficient, C_L^*, based on the baseline midcruise weight is 0.1665. The center of gravity is assumed to be at 25% of the mean aerodynamic chord for the baseline configuration. For configurations deviating from the three-surface baseline configuration, we recompute the target lift and center of gravity assuming a mass per planform area of 27 kg/m^2 [170, Table 15.2] for the trim surfaces.

Table 5.1: Aircraft dimensions

Quantity	Value
Wing reference area (S_{ref})	373.03 m^2
Wing mean aerodynamic chord ($\bar{c}$)	14.565 m
Wing half-span	14.760 m
Canard half-span	2.327 m
Tail half-span	5.543 m



(a) Three-surface (2,497,986 cells)

(b) Canard (2,128,572 cells)

(c) Conventional (2,268,378 cells)

(d) Trimless (1,898,964 cells)

Figure 5.1: Overset configuration meshes

5.2.2 RANS-based optimization setup

We use the MACH-Aero framework [56] for the RANS-based optimizations. We use ADflow [77] to solve the compressible RANS equations with the Spalart–Allmaras (SA) turbulence model [78] on overset structured meshes. We converge the RANS equations to a total residual of 10^{-10} relative to freestream using the approximate Newton–Krylov solver [80] in ADflow. We also use the dissipation-based continuation method described in Sec. 3.5 to improve solver robustness. The surface mesh that represents the geometry is modified using a combination of inverse-distance surface mesh deformation near component intersections [154] and free-form deformation [147, 148] away from the intersections. The surface mesh changes are propagated to the volume mesh using an inverse-distance deformation method [81, 149]. To run the optimizations, we use the SNOPT optimizer [151] through the interface provided by pyOptSparse [150]. The gradients are computed using an adjoint approach [146].

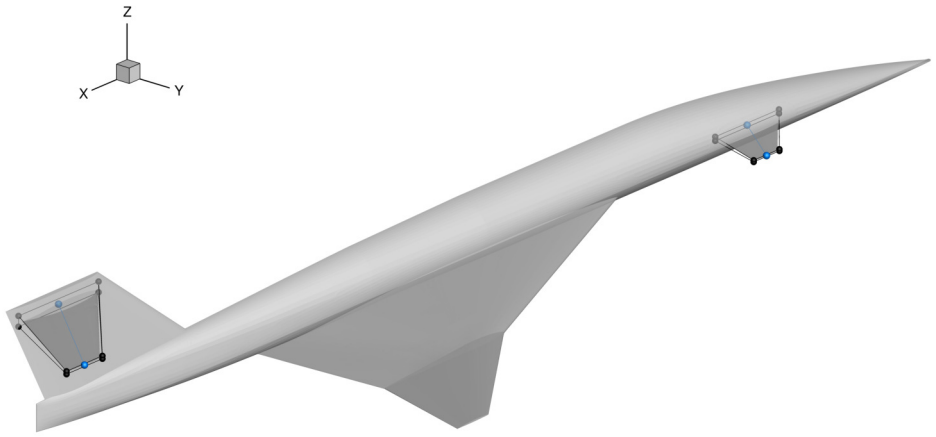


Figure 5.2: The trim surfaces can rotate about their 50% chord axis

The optimization formulation for the RANS-based trim-constrained drag minimization is shown in Table 5.2. The canard and tail rotation design variables are defined about the 50% chord axis of the canard and horizontal stabilizer, respectively. Figure 5.2 shows the FFD control points (black) and the reference axis control points (blue) that are used to define the rotations. The trim surface rotation bounds are based on the maximum rotation that produces a valid mesh. More details on the geometry parameterization for this SST configuration are in Sec. 4.3.2.

Table 5.2: RANS-based trim drag minimization problem

		Lower	Upper	Scaling
minimize	C_D			100
with respect to	Angle of attack	$-6°$	$6°$	0.1
	Canard rotation	$-5°$	$14°$	0.1
	Tail rotation	$-9°$	$9°$	0.1
subject to	C_L / C_L^*	1	1	1
	C_M	0	0	1

5.2.3 RANS-based optimization results

The optimizations for the three-surface, canard, and conventional configurations are shown in Table 5.3. The constraint violation for the converged optimizations is 6.3×10^{-7} or lower, and the optimality is 2.4×10^{-6} or lower. The conventional configuration has 8% higher drag than the three-surface configuration. The canard configuration optimization is infeasible because of the upper bound on the canard rotation. It is not possible to trim the aircraft without a larger canard rotation. This motivates the development of the buildup model presented in Sec. 5.3.1.

Table 5.3: Trim optimization results

	Three-surface	Canard	Conventional
Supersonic RANS			
Drag counts	332.0	–	358.7
Angle of attack	$3.20°$	–	$3.58°$
Canard rotation	$7.21°$	–	–
Tail rotation	$-4.72°$	–	$-8.72°$
Supersonic buildup			
Drag counts	330.5	360.5	365.3
Angle of attack	$3.28°$	$3.04°$	$3.59°$
Canard rotation	$8.27°$	$23.0°$	–
Tail rotation	$-4.88°$	–	$-8.26°$

5.3 Buildup-based trim optimization

5.3.1 Supersonic buildup model

One disadvantage of optimizations that rely on mesh deformation is that the design space is limited to geometries that produce a valid mesh. To avoid this limitation, we construct a buildup model that uses a combination of RANS data and analytic functions. We start by running RANS on the trimless configuration at angles of attack from -6 to 6 deg. The RANS solutions are converged to a total residual of 10^{-8} or tighter relative to the freestream residual. We then construct a quartic least-squares fit on the lift, drag, and moment curves. The trim surfaces and the interactions between components are modeled analytically based on the methods presented by Raymer [170] and USAF Datcom [171]. However, we use RANS simulations to inform some aspects of the analytic equations. We generate meshes for five trapezoidal wings with varying thickness, aspect ratio, taper ratio, and leading-edge sweep (Table 5.4). Each wing has a constant symmetric NACA 4-digit airfoil cross-section. All five NACA wing meshes have 468,992 cells (Fig. 5.3). We simulate all five wings at angles of attack from 0 to 50 deg. These solutions are converged to a total residual of 10^{-6} or tighter relative to the freestream residual. This gives us data to ensure that the analytic equations correctly model the nonlinear aerodynamics and that the model is applicable for trim surfaces with different planforms and thicknesses.

Table 5.4: NACA wing geometries

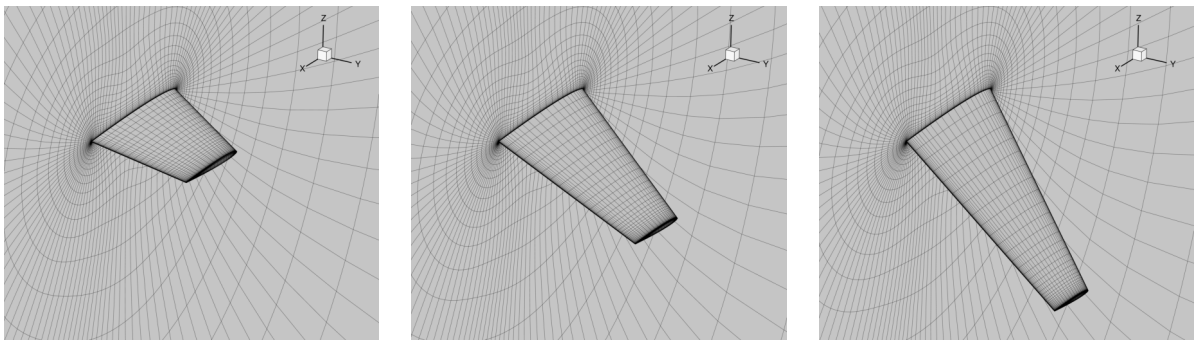
Wing	NACA airfoil	Aspect ratio	Taper ratio	Leading-edge sweep
A-06	0006	1	0.6	40°
B-03	0003	2	0.5	45°
B-06	0006	2	0.5	45°
B-09	0009	2	0.5	45°
C-06	0006	3	0.4	50°

5.3.1.1 Trim surface lift

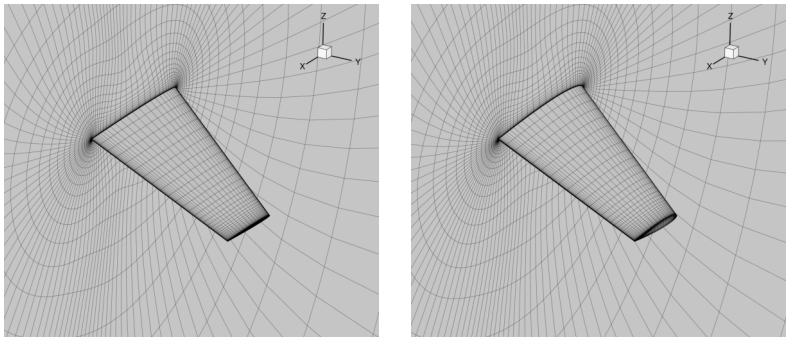
We model the trim surface lift as a cubic function:

$$C_L = C_{L_0} + C_{L_\alpha}\alpha - \frac{C_{L_\alpha}}{3\alpha_{L_{\max}}^2}\alpha^3, \tag{5.1}$$

where C_{L_α} is the lift curve slope at zero angle of attack, and the coefficient for the cubic term is set such that the lift is maximized at $\alpha_{L_{\max}}$. We set $\alpha_{L_{\max}}$ to 40 deg based on the RANS simulations. The



(a) A-06 (b) B-06 (c) C-06



(d) B-03 (e) B-09

Figure 5.3: NACA wing meshes

trim surfaces are approximately symmetric, so we assume $C_{L_0} = 0$. The lift curve slope is computed based on data from Datcom [171, Fig. 4.1.3.2-56]. We use this data to construct a surrogate model that computes the lift curve slope as a function of aspect ratio, leading-edge sweep, taper ratio, and Mach number. Using a surrogate model allows the lift curve slope to be computed in a manner suitable for gradient-based optimization. We use the regularized minimal-energy tensor-product B-splines [172] surrogate model as implemented in the Surrogate Modeling Toolbox [173].

We find that the Datcom data does not accurately represent the lift curve slope for low aspect-ratio wings. To correct this, we use a transformed aspect ratio, $\hat{A}$, in place of the actual aspect ratio, A, as the input to the surrogate model. $\hat{A}$ is computed as

$$\hat{A} = \left(\bar{A}^{n-1} A\right)^{\frac{1}{n}}, \tag{5.2}$$

where $\bar{A}$ is a constant that determines the aspect ratio that is invariant under the transformation and n is a constant that determines the steepness of the transformation. We use $\bar{A} = 6$ and $n = 1.7$. The transformation function represented by these constants is shown in Fig. 5.4. The Datcom lift curve slope also does not account for thickness effects. We apply a correction based on the thickness-to-chord ratio, t/c, to decrease the lift curve slope for thick wings:

$$C_{L_\alpha} = \frac{C_{L_\alpha,\text{surrogate}}}{\left(1 + \frac{t}{c}\right)^{1.5}}. \tag{5.3}$$

The lift curve slopes computed with the aspect ratio and thickness corrections match RANS well for all five NACA wings (Table 5.5).

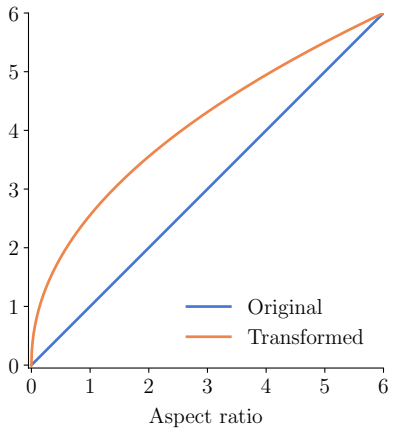


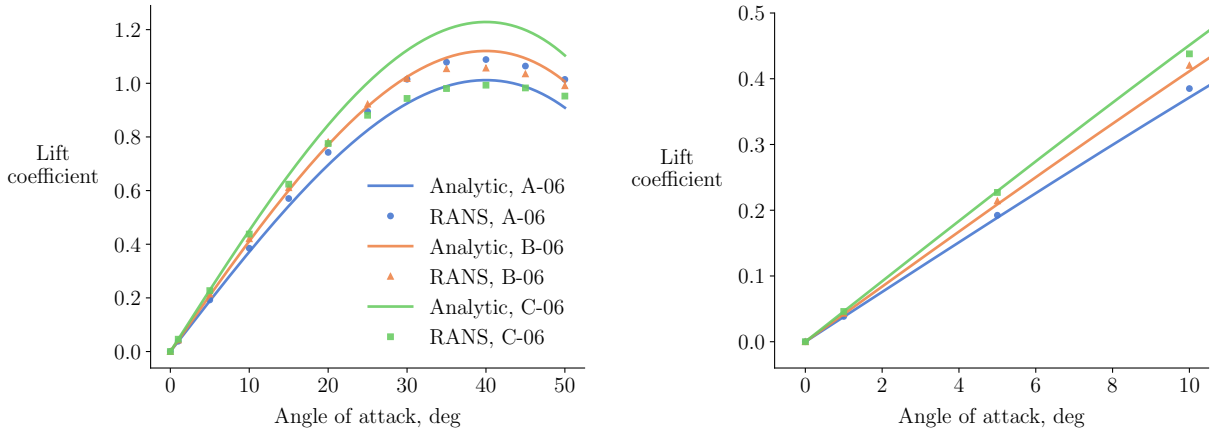
Figure 5.4: Aspect ratio transformation function

The nonlinear lift model is approximately linear for low angles of attack and agrees well with

Table 5.5: The corrected lift curve slope matches RANS well

Wing	Uncorrected	RANS	Corrected
A-06	1.631	2.202	2.174
B-03	2.365	2.637	2.513
B-06	2.365	2.469	2.407
B-09	2.365	2.313	2.309
C-06	2.666	2.635	2.640

the RANS simulations at low angles (Figs. 5.5b and 5.6b). At higher angles of attack, the nonlinear lift model follows a similar shape as RANS but deviates from the RANS values (Figs. 5.5a and 5.6a). This is particularly true for wing planforms A and C. However, the main reason for using a nonlinear lift model is to discourage the optimizer from increasing the angle of attack to unreasonably high values. The cubic lift model captures the lift trends at high angles of attack well enough to accomplish this goal. In addition, we do not expect RANS to represent the true physics at the highest angles of attack, so matching RANS in this regime would not make the model more realistic.



(a) Full angle sweep

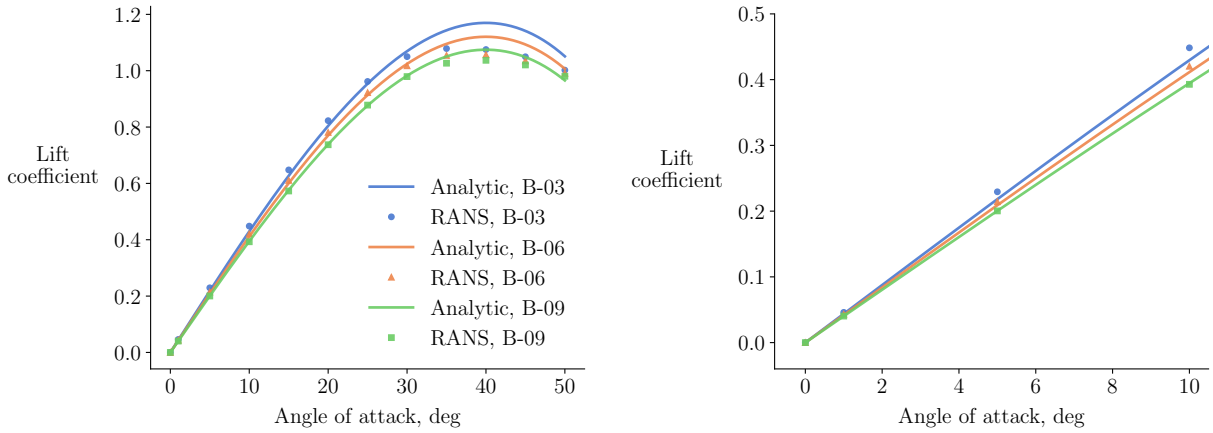
(b) Zoom-in on low angles

Figure 5.5: Analytic lift compared to RANS for different planforms

5.3.1.2 Trim surface drag

The trim surface drag consists of skin friction drag, wave drag, and induced drag. We compute the turbulent skin friction as [170, Eq. 12.27]

$$C_f = \frac{0.455}{(\log_{10} R)^{2.58}(1 + 0.144M^2)^{0.65}}, \tag{5.4}$$



(a) Full angle sweep

(b) Zoom-in on low angles

Figure 5.6: Analytic lift compared to RANS for different thicknesses

where R is the Reynolds number based on the component's mean aerodynamic chord and M is the Mach number. The wave drag is [171, Eq. 4.1.5.1-n]

$$C_{D_{\text{wave}}} = E_{\text{WD}} C_{D_{\text{LE}}} + \frac{16}{3\sqrt{M^2 - 1}} \left(\frac{t}{c}\right)^2, \tag{5.5}$$

where $C_{D_{\text{LE}}}$ is a drag correction term for rounded leading edges and E_{WD} is a wave drag correction factor that is not included in Datcom. We include a wave drag correction factor of $E_{\text{WD}} = 2.5$ to approximate the zero-lift drag from RANS. $C_{D_{\text{LE}}}$ is computed as [171, p. 4.1.5.1-30]

$$\frac{1.28 M^3 \cos^6 \Lambda_{\text{LE}}}{1 + M^3 \cos^3 \Lambda_{\text{LE}}} \frac{A_{\text{LE}}}{S_{\text{comp}}}, \tag{5.6}$$

where Λ_{LE} is the leading-edge sweep, A_{LE} is the frontal area of the leading edge, and S_{comp} is the component's planform area. The zero-lift drag is the sum of the skin friction drag and wave drag:

$$C_{D_0} = C_f \frac{S_{\text{wet}}}{S_{\text{comp}}} + C_{D_{\text{wave}}}, \tag{5.7}$$

where the skin friction drag scales with the wetted area S_{wet}. We compute the induced drag with the leading-edge suction method [170, Eq. 12.56]:

$$C_{D_i} = \frac{C_L^2}{C_{L_\alpha}}. \tag{5.8}$$

The total drag is

$$C_D = C_{D_0} + C_{D_i}. \tag{5.9}$$

The drag buildup described above works reasonably well at low angles of attack for different planforms (Fig. 5.7b) and thicknesses (Fig. 5.8b). However, it does not model the increase in drag that occurs once the lifting surface begins to stall at higher angles of attack. To account for stall effects, we use the model developed by Tangler and Ostowari [174]. This model was originally created for wind turbine design, but it has also been applied to subsonic aircraft design [175]. We show that the model is also useful for supersonic aircraft. The poststall drag is given by

$$C_D = B_1 \sin\alpha + B_2 \cos\alpha, \tag{5.10}$$

where

$$B_1 = C_{D_{\max}} \tag{5.11}$$

and

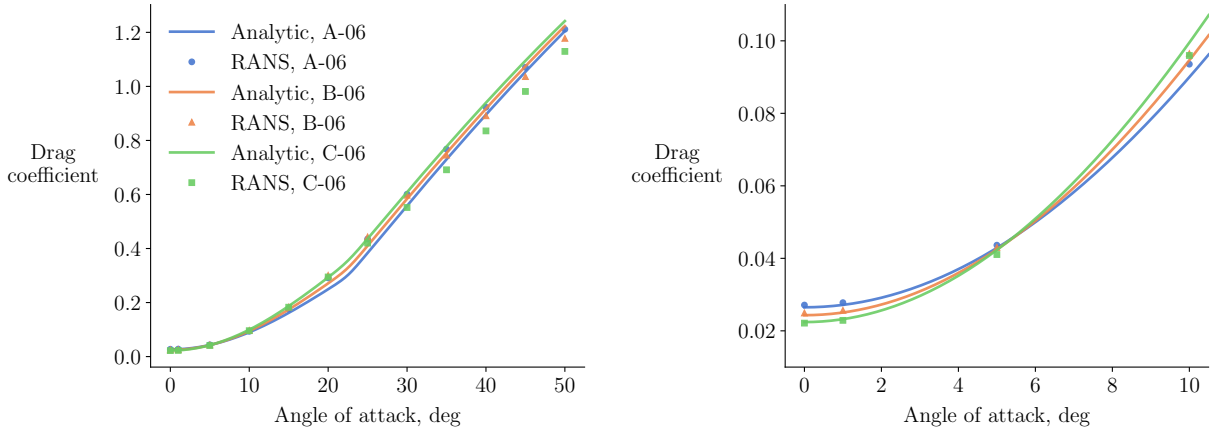
$$B_2 = C_{D_{\text{stall}}} - B_1 \frac{\sin\alpha_{\text{stall}}}{\cos\alpha_{\text{stall}}}. \tag{5.12}$$

This model requires setting values for $C_{D_{\max}}$ and α_{stall}, the stall angle of attack. $C_{D_{\text{stall}}}$ is the drag at the stall angle of attack computed using the prestall model. We find that using $C_{D_{\max}} = 2$ and a stall angle of attack of 22.5 deg results in good agreement between the drag buildup and RANS at high angles of attack (Figs. 5.7a and 5.8a). The poststall drag is only weakly dependent on the planform and thickness, so no additional parameters are included in the poststall model. Similar to the lift model, we are only interested in approximately capturing the trends at high angles of attack instead of precisely matching the RANS data. We use the Kreisselmeier–Steinhauser function [176] to smoothly transition from the prestall model to the poststall model. This approach was also used by Chauhan and Martins [175].

5.3.1.3 Downwash

Accounting for downwash is critical for accurate buildup results. To simplify the downwash computation, we assume that the trimless part of the aircraft is entirely downstream of the canard and entirely upstream of the horizontal stabilizer. We account for the canard downwash on the trimless configuration and the trimless downwash on the horizontal stabilizer. The downwash derivative for an upstream component is computed as [170, Eq. 16.21b]

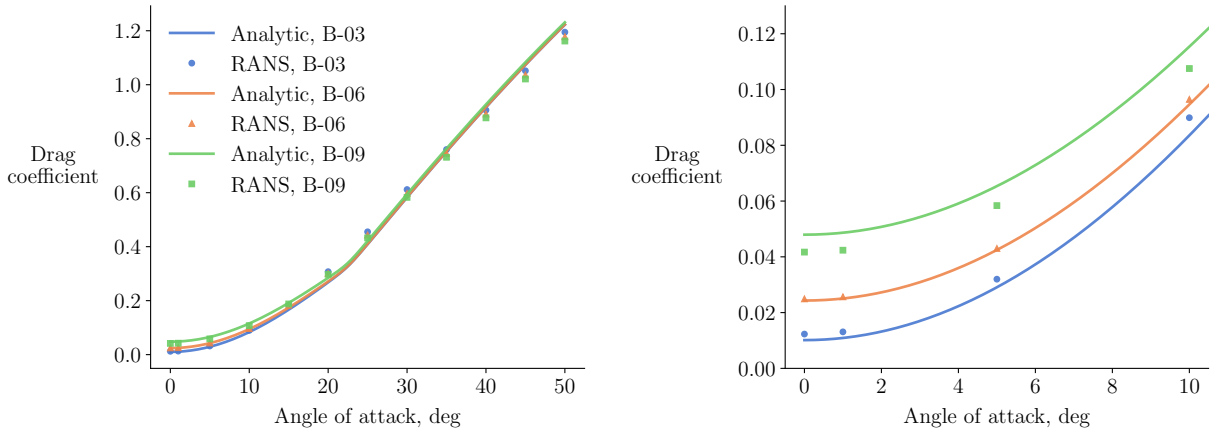
$$\left(\frac{\partial\epsilon}{\partial\alpha}\right)_{\text{upstream}} = \left(\frac{1.62 C_{L_\alpha}}{\pi A}\right)_{\text{upstream}}. \tag{5.13}$$



(a) Full angle sweep

(b) Zoom-in on low angles

Figure 5.7: Analytic drag compared to RANS for different planforms



(a) Full angle sweep

(b) Zoom-in on low angles

Figure 5.8: Analytic drag compared to RANS for different thicknesses

A component's total angle of attack is then computed as [170, Eq. 16.24]

$$\alpha_{\text{comp}} = \left(\alpha + i_{\text{upstream}}\right)\left(1 - \left(\frac{\partial \epsilon}{\partial \alpha}\right)_{\text{upstream}}\right) + i_{\text{comp}} - i_{\text{upstream}}, \tag{5.14}$$

where i_{comp} is the component's incidence angle and i_{upstream} is the upstream component's incidence angle. In addition, we assume that the trimless downwash on the horizontal stabilizer is zero at negative angles of attack. This is a reasonable assumption for a T-tail configuration and improves the match between the buildup results and RANS. We multiply the downwash derivative for the trimless configuration by a sigmoid function,

$$\frac{1}{1 + e^{-\alpha}}, \tag{5.15}$$

to smoothly transition from zero downwash at negative angles of attack to the downwash derivative in Eq. 5.13 at positive angles of attack.

5.3.1.4 Total forces and pitching moment

Finally, we combine the forces and moments from the trim surfaces and the trimless configuration. The lift is computed by adding all the component lift forces normalized using the wing reference area:

$$C_L = C_{L_{\text{trimless}}} + \sum_{\text{comp}} C_{L_{\text{comp}}} \frac{S_{\text{comp}}}{S_{\text{ref}}}. \tag{5.16}$$

The drag is computed in the same manner. The total moment about the center of gravity is

$$C_M = C_{M_{\text{trimless}}} + \sum_{\text{comp}} \left(C_{N_{\text{comp}}} \frac{S_{\text{comp}}}{S_{\text{ref}}} \frac{\Delta x_{\text{comp}}}{\bar{c}} + C_{A_{\text{comp}}} \frac{S_{\text{comp}}}{S_{\text{ref}}} \frac{\Delta z_{\text{comp}}}{\bar{c}} \right), \tag{5.17}$$

where the normal force coefficient is

$$C_{N_{\text{comp}}} = C_{L_{\text{comp}}} \cos\alpha + C_{D_{\text{comp}}} \sin\alpha, \tag{5.18}$$

the axial force coefficient is

$$C_{A_{\text{comp}}} = -C_{L_{\text{comp}}} \sin\alpha + C_{D_{\text{comp}}} \cos\alpha, \tag{5.19}$$

Δx_{comp} is the x-component of the vector from the center of gravity to the component's aerodynamic center, Δz_{comp} is the z-component of the vector from the center of gravity to the component's aerodynamic center. We neglect the moments on the trim surfaces because these are much smaller

than the trim surface forces multiplied by the moment arms.

5.3.1.5 Model verification

We verify the buildup model by comparing the buildup results to RANS. We first compare the buildup to RANS on the three-surface configuration at angles of attack from -6 to 6 deg (Fig. 5.9). The buildup drag matches RANS well but slightly underpredicts drag at higher angles of attack. The buildup lift matches RANS nearly exactly for the entire angle of attack range. The buildup moment is similar to RANS at positive angles of attack but does not have the correct slope at negative angles of attack. The trim condition for the aircraft occurs at a low positive angle of attack, so we expect the buildup to perform well despite this discrepancy.

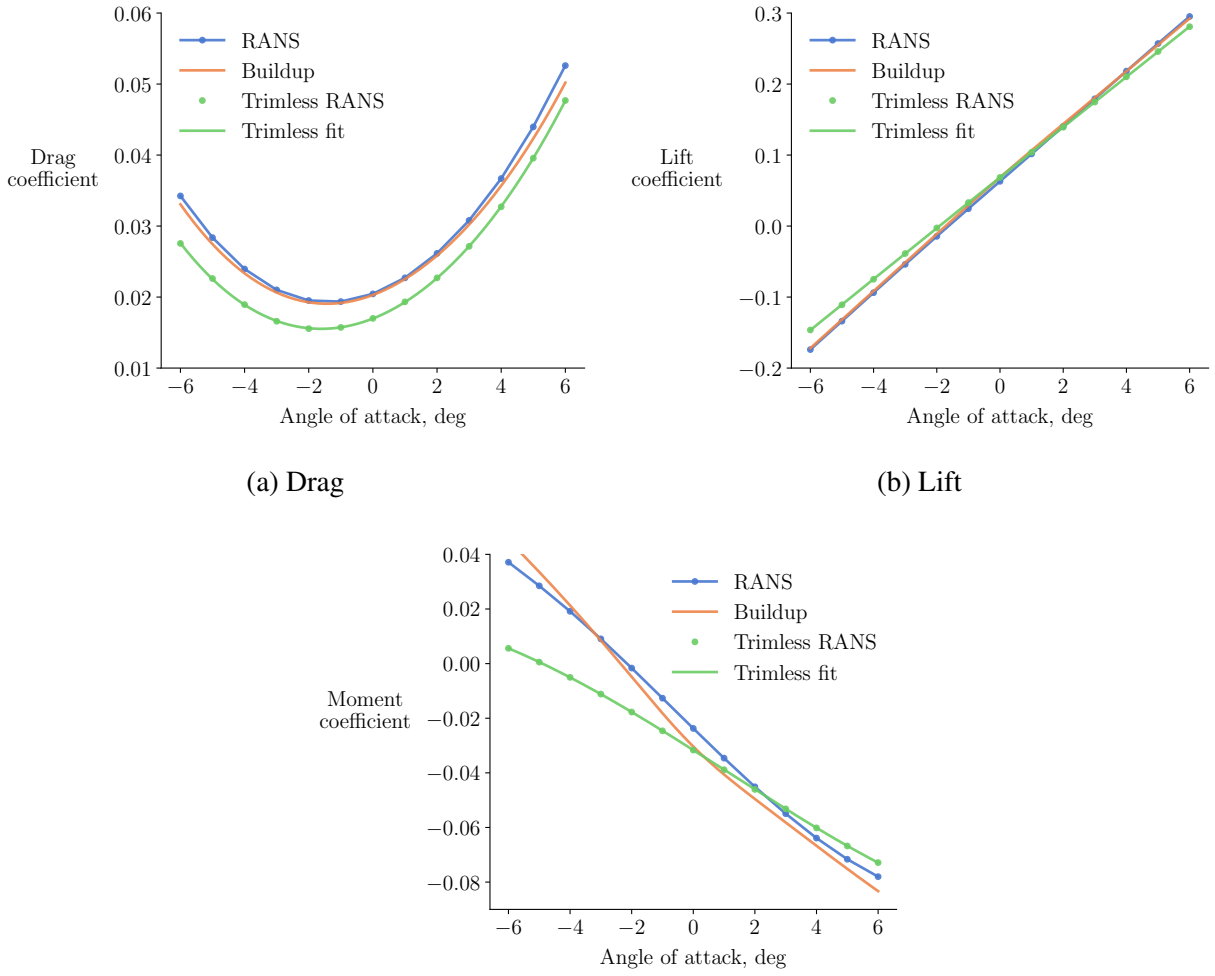
Next, we look at the incremental changes in the aerodynamic coefficients for canard rotations (Fig. 5.10) and tail rotations (Fig. 5.11) at zero angle of attack. The buildup does not precisely match the RANS increments, but the trends are correct. In particular, the buildup has a negative lift increment for a positive canard rotation (Fig. 5.10b), which is the result of including the canard downwash effect on the trimless configuration. The tail rotation increments match RANS more closely than the canard rotation increments. This suggests that the canard's interaction with the fuselage and wing is more complex than interactions involving the tail.

5.3.2 Buildup-based optimization setup

We wrap the buildup model in OpenMDAO [54] to run optimizations. As with the RANS-based optimizations, we use SNOPT through pyOptSparse as the optimizer. The gradients of the lift curve slope surrogate model are computed analytically. All other gradients are computed using the complex-step method [159]. We consider optimizations with fixed trim surface sizing (Table 5.6) and variable trim surface sizing (Table 5.7). Unlike the RANS optimizations, the rotation variables are not bounded by mesh deformation limits. Instead, we bound them to the range where we expect the buildup model to be reasonably accurate. The lower bound for the trim surface span variables is zero. This means that the optimizer has the option of removing either trim surface. Aside from speed, this is the main advantage of using the buildup model in an optimization instead of RANS. Removing a trim surface or doubling its span would not be possible with a mesh deformation approach. For the optimization with span variables, we constrain the static margin to be at least 5% to avoid unrealistically sized trim surfaces. The static margin is computed as in Sec. 4.3.4:

$$K_n = -\frac{C_{M_\alpha}}{C_{N_\alpha}}, \tag{5.20}$$

where the derivatives are computed using the finite-difference method with a step size of 0.01 deg.

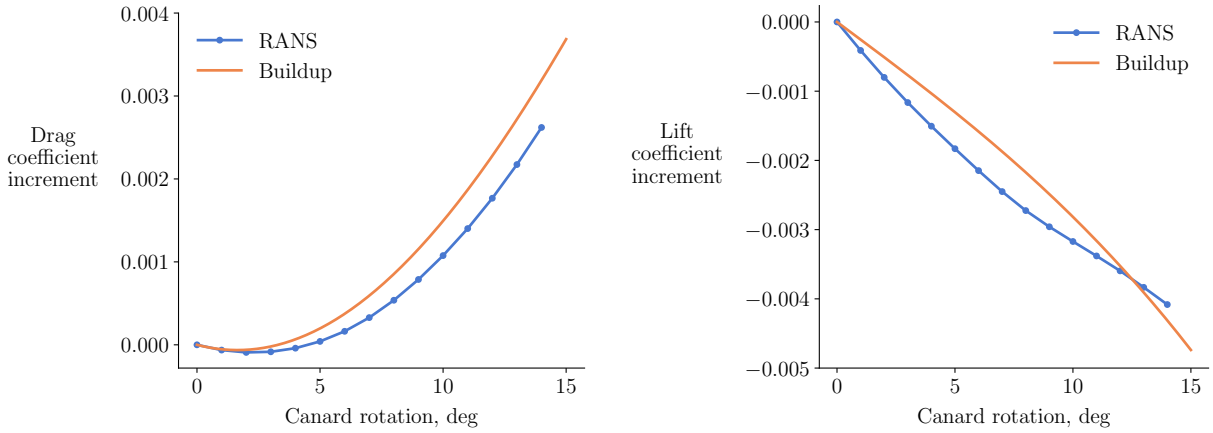


(a) Drag

(b) Lift

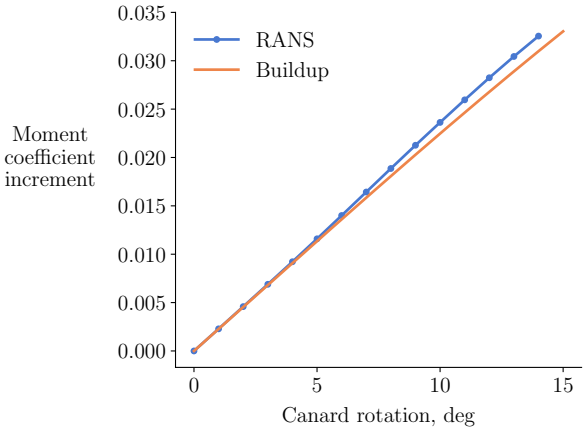
(c) Pitching moment

Figure 5.9: Buildup verification for different angles of attack



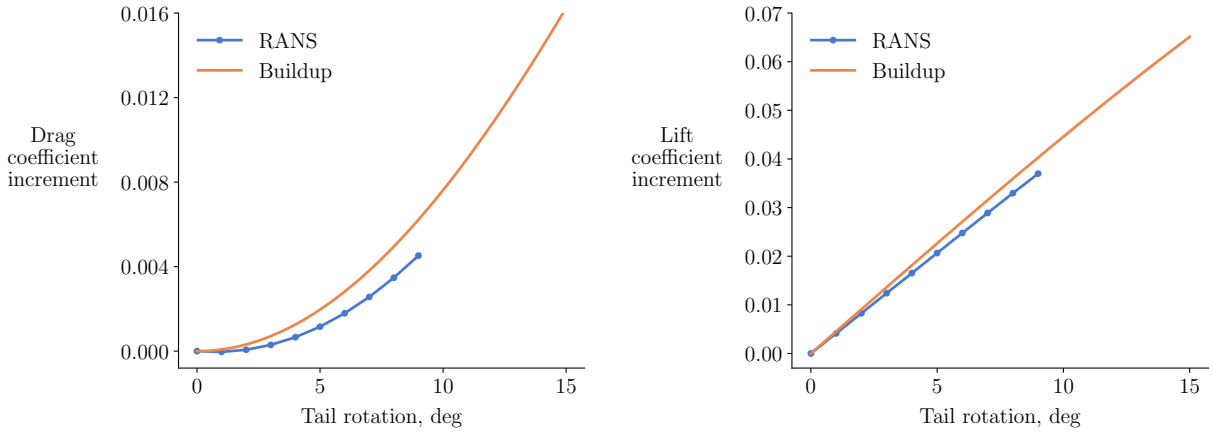
(a) Drag

(b) Lift



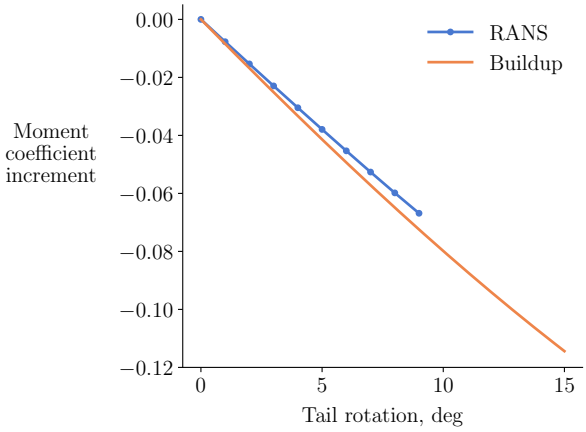
(c) Pitching moment

Figure 5.10: Buildup verification for canard rotation increments



(a) Drag

(b) Lift



(c) Pitching moment

Figure 5.11: Buildup verification for tail rotation increments

Table 5.6: Buildup-based optimization with fixed trim surface sizing

		Lower	Upper	Scaling
minimize	C_D			100
with respect to	Angle of attack	$-6°$	$6°$	0.1
	Canard rotation	$-25°$	$25°$	0.1
	Tail rotation	$-25°$	$25°$	0.1
subject to	C_L / C_L^*	1	1	1
	C_M	0	0	1

Table 5.7: Buildup-based optimization with trim surface sizing variables

		Lower	Upper	Scaling
minimize	C_D			100
with respect to	Angle of attack	$-6°$	$6°$	0.1
	Canard rotation	$-25°$	$25°$	0.1
	Canard half-span	0 m	10 m	0.01
	Tail rotation	$-25°$	$25°$	0.1
	Tail half-span	0 m	10 m	0.01
subject to	C_L / C_L^*	1	1	1
	C_M	0	0	1
	K_n	0.05	0	1

5.3.3 Buildup-based optimization results

We first use the buildup model to run the same trim-constrained drag minimizations as we did for RANS. These results are presented in Table 5.3 for ease of comparison with the RANS results. The trimmed buildup drag for the three-surface and conventional configurations match RANS to within 2%. The optimized angle of attack and tail rotation match to within 6%. The optimized canard rotation has the largest difference at around 15%. These optimization results provide further verification of the buildup model. The canard configuration optimization is feasible without the mesh-induced upper bound on the canard rotation. However, the design is not realistic because the canard has to enter the poststall regime to satisfy the trim constraints.

The speed of the buildup model allows us to explore the design space multimodality of the three-surface trim optimization. We run optimizations from 100 random starting points. We consider the buildup-based optimizations converged when they reach feasibility and optimality values of 10^{-8} or lower. These tolerances are much easier to achieve with an analytic model than the RANS

equations. All 100 optimizations converge to the same design. Figure 5.12 shows the initial and optimized design variables for the first 20 optimizations. This strongly suggests that the optimization problem is unimodal. The unimodality of the three-surface optimization is an important result for aerodynamic shape optimization studies. For shape optimizations, we typically compare the trimmed optimized result with the trimmed baseline configuration such as in Sec. 4.3.6. If the trim optimization is unimodal, we can start from any untrimmed state to get the lowest trim drag for the baseline design. This allows for a fair comparison with the trimmed shape optimized design.

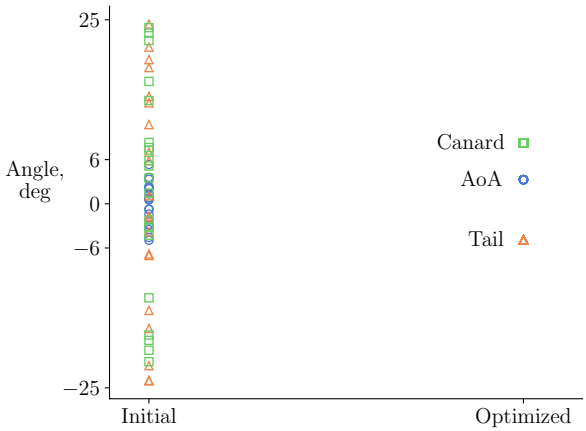
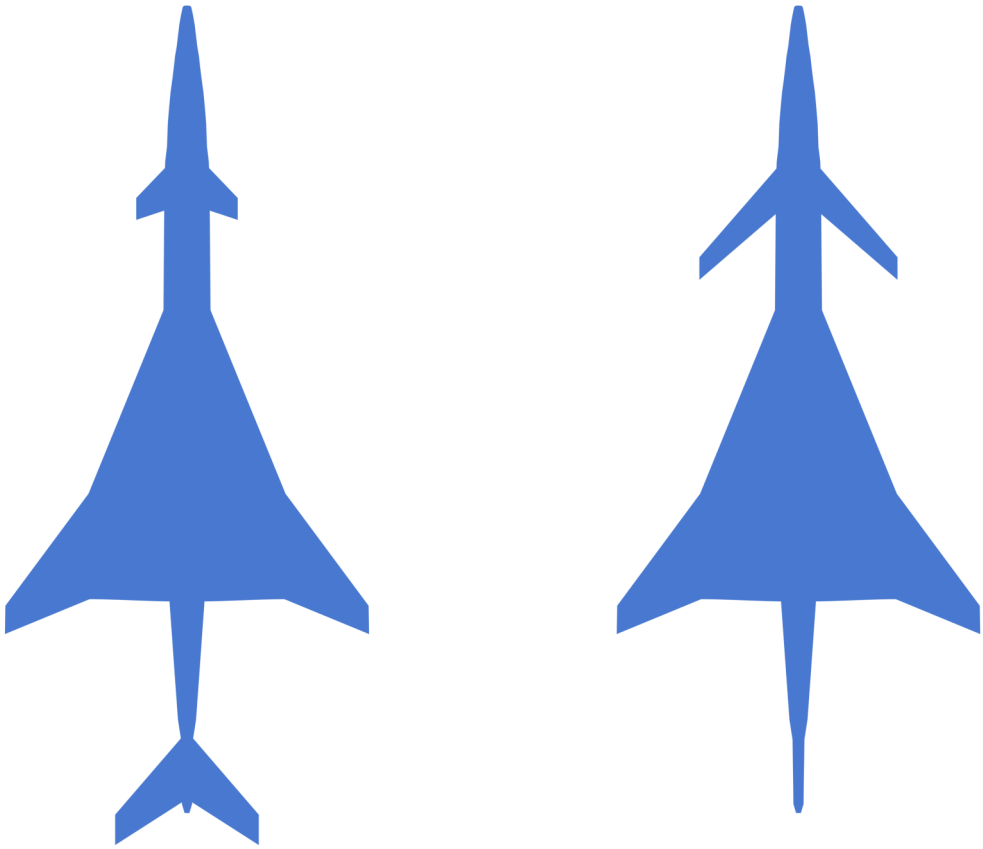


Figure 5.12: The three-surface trim optimization is unimodal

Next, we run the trim optimization with span variables. Similar to the previous trim optimization, we run 100 optimizations from different starting points. Unlike the optimization with fixed sizing, this optimization is multimodal. Table 5.8 shows the best optimized design. The optimized planform has no horizontal stabilizer and a canard that is more than twice the span of the baseline (Fig. 5.13). The canard has a lower lift penalty for a positive moment increment than the tail. Increasing the canard's size allows the aircraft to trim at a lower angle of attack, which reduces drag by 15.5% compared to the baseline design. One concern with canard designs is that they are more likely to be statically unstable in pitch [2, Sec. 8.5]. This is reflected in the optimization because the static margin constraint is active. The optimized design has a static margin equal to the constraint value of 5%, whereas the baseline design has a static margin of 23.1%.

5.4 Subsonic considerations

The planform optimization in Sec. 5.3.3 enforced a supersonic static margin constraint, which ensures that the optimized design is longitudinally stable at the cruise condition. Supersonic transport aircraft must also be stable at subsonic conditions such as takeoff and landing. However, these



(a) Baseline (b) Optimized

Figure 5.13: Planforms for the trim surface sizing optimization

Table 5.8: Trim surface sizing optimization results

	Baseline	Optimized
Drag counts	330.5	279.6
Static margin	23.1%	5.00%
Angle of attack	3.28°	2.28°
Canard rotation	8.27°	6.67°
Canard half-span	2.327 m	5.711 m
Tail rotation	−4.88°	–
Tail half-span	5.543 m	0.000 m

conditions involve high-angle-of-attack, vortex-dominated flow [44], which is not well-suited for a buildup procedure like the one we formulated for supersonic flow.

In Sec. 4.2.3, we showed that RANS captures the subsonic characteristics of the baseline configuration up to moderate angles of attack. Therefore, we use RANS to analyze the baseline and optimized planforms at a subsonic takeoff condition. The subsonic condition is at a Mach number of 0.3 and at sea-level altitude. The overset mesh for the optimized planform is shown in Fig. 5.14. We analyze both planforms at angles of attack from -10 to 15 deg with zero trim surface rotations.

We plot the moment coefficient against the angle of attack for the baseline and optimized designs in Fig. 5.15. The moment coefficient is computed about the center of gravity of each configuration accounting for the shift caused by resizing the trim surfaces. Longitudinal stability requires the pitching moment to decrease with angle of attack, which is indicated by a negative slope on this plot. The baseline design is stable at low angles of attack but becomes unstable at higher angles. This is typical for cranked-arrow wings at subsonic conditions [16]. The optimized planform is unstable across the entire angle of attack range. Removing the horizontal stabilizer and increasing the canard's span has a destabilizing effect even at the supersonic cruise condition. The forward shift of the neutral point is larger at the subsonic condition, resulting in an unstable aircraft.

5.5 Conclusions

In this chapter, we use nonlinear aerodynamic models and numerical optimization to determine the minimum trim drag configuration for a supersonic transport aircraft. We determine the minimum trim drag for three-surface, canard, and conventional configurations with fixed trim surface sizing using RANS-based optimizations. The three-surface configuration has the lowest trim drag at a supersonic cruise condition.

To overcome the mesh-related limitations of the RANS-based optimizations, we formulate a supersonic buildup model. The model is accurate enough to approximate the RANS-based trim

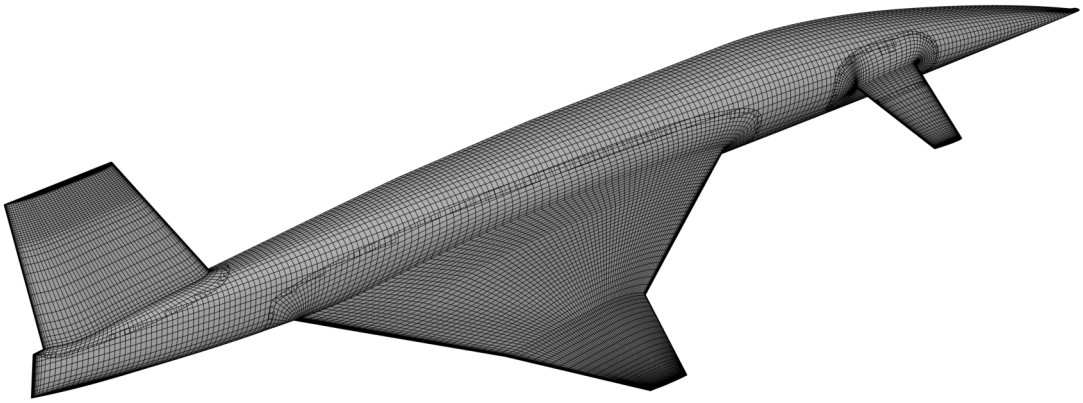


Figure 5.14: Overset mesh for the optimized planform (2,413,908 cells)

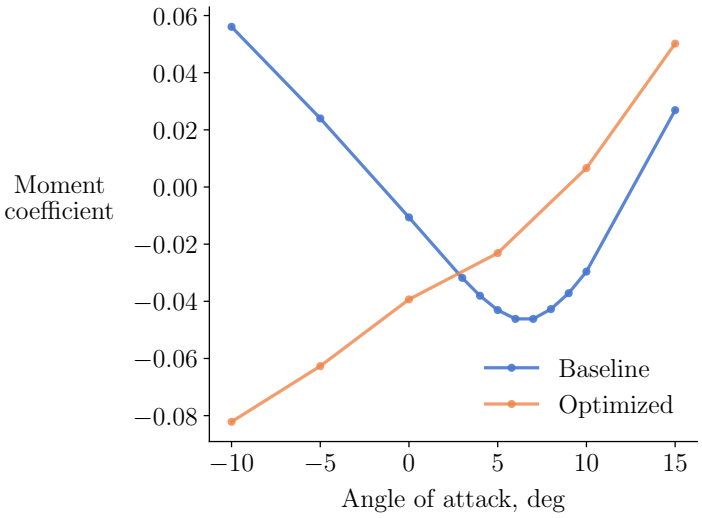


Figure 5.15: The optimized planform is unstable at subsonic conditions

optimization results. We use the supersonic buildup model to optimize the aircraft with variable trim surface sizing. The design space for this optimization includes the option to remove either trim surface, which would not be possible with an optimization involving mesh deformation. When considering trim surface sizing, the design for minimum supersonic drag has no horizontal stabilizer and a canard sized at 39% of the wing span. The optimized planform has 15.5% lower supersonic drag than the baseline planform.

We then analyze the baseline and optimized planforms at a subsonic takeoff condition using RANS. Whereas the baseline design becomes unstable at high angles of attack, the optimized design is unstable even at low angles of attack. This highlights the trade-off between supersonic performance and subsonic stability and the need to consider both simultaneously when designing supersonic transport aircraft.

Chapter 6

Conclusion

In this dissertation, I used CFD and optimization to study SST design with a focus on low-speed stability. The first half of this dissertation dealt with CFD methods for high-angle-of-attack conditions, low Mach number flows, and high Mach number flows. The second half of this dissertation focused on aerodynamic design optimization of a full SST configuration, including wing and fuselage shape optimization and trim surface sizing optimization. In the remainder of this chapter, I summarize the key results from the dissertation, present my contributions, and provide recommendations for future work.

6.1 Key results

Chapter 2 considered the flow over a delta wing with an aspect ratio of 2 at low-speed, high-angle-of-attack conditions. I evaluated the accuracy of RANS and DDES at predicting force and moment coefficients for angles of attack from 0 to 40 deg. I also proposed a steadiness metric based on DDES frequency information to distinguish between steady and unsteady flow conditions. RANS provided accurate results for angles of attack up to 20 deg. The steadiness metric showed that this corresponds to the steady flow regime. RANS was inaccurate at higher angles of attack where the flow is unsteady. The lift and drag in this regime were underpredicted, and the pitching moment trend was incorrect. DDES was more accurate than RANS at higher angles of attack. The errors in the lift and drag were within 6% for DDES compared to 20% for RANS. The DDES results were most sensitive to mesh refinement at the highest angles of attack. The break in the pitching moment was not fully captured even with a 69-million-cell mesh. The computational cost of DDES was 100–1000 times that of RANS.

Chapter 3 presented three contributions to improve the speed and accuracy of compressible flow solvers for low and high Mach number flows. First, I proposed a simple modification to the Jameson–Schmidt–Turkel scheme to make the artificial dissipation appropriate for low Mach number flows. I showed that scaling down the acoustic contribution of the spectral radius by the freestream

Mach number improves accuracy and reduces sensitivity to mesh size. Second, I demonstrated the effectiveness of a characteristic time-stepping method for approximate Newton–Krylov solvers. This approach reduces the stiffness of the linear system at low Mach numbers by modifying the time-step matrix. I showed that the artificial dissipation must be adequately scaled to achieve speedup with characteristic time-stepping. In addition, cases with small regions of stagnation point flow do not benefit from characteristic time-stepping. Third, I presented a dissipation-based continuation method for flows with shocks. The continuation approach introduces additional dissipation near shocks during the initial stages of the solution and smoothly reduces the dissipation as the solution converges. This approach was faster and more robust than the baseline solver, particularly for more complex geometries.

In Chapter 4, I used RANS-based aerodynamic shape optimization to study the low-speed stability of an SST. I showed that RANS captures the lift, drag, and pitching moment trends of the supersonic configuration at low-speed, high-angle-of-attack conditions by comparing against experimental data. The shape optimizations involved minimizing supersonic drag with and without a static margin constraint at a subsonic takeoff condition. The baseline design and design optimized without the static margin constraint were both unstable at subsonic speeds. The stability-constrained optimizations had target static margins of 0%, 5%, and 10%. Pitch stability was achieved by deflecting the leading-edge flaps and increasing the wing thickness. The increase in wing thickness resulted in a supersonic drag penalty of 0.50% for a 0% static margin, 0.77% for a 5% static margin, and 0.85% for a 10% static margin.

In Chapter 5, I used nonlinear aerodynamic models and optimization to determine the minimum trim drag configuration for an SST. I determined the minimum trim drag for three-surface, canard, and conventional variants of an SST using RANS-based optimizations. The three-surface configuration had the lowest trim drag at a supersonic cruise condition. To overcome mesh-related limitations of the RANS-based optimizations, I formulated a supersonic buildup model. The model was accurate enough to approximate the RANS-based trim optimization results. I then used the supersonic buildup model to optimize the SST with variable trim surface sizing. The design space for this optimization included the option to remove either trim surface, which would not be possible with an optimization involving mesh deformation. When considering trim surface sizing, the design for minimum supersonic drag had no horizontal stabilizer and a canard sized at 39% of the wing span. The optimized planform had 15.5% lower supersonic drag than the baseline planform. I then analyzed the baseline and optimized planforms at a subsonic takeoff condition using RANS. Whereas the baseline design was unstable at high angles of attack, the optimized design was unstable even at low angles of attack.

6.2 Contributions

The main contributions of this dissertation are as follows:

1. *I determined the appropriate fidelity for low-speed, high-angle-of-attack CFD of supersonic configurations.*
   Prior studies on CFD for supersonic configurations at low speeds focused on one or a few angles of attack. My contribution was to study the accuracy of CFD models at predicting aerodynamic coefficients over a wide range of angles of attack, including angles up to and past stall. I paid particular attention to studying the breaks in the force and moment coefficients, which are important for stability.

2. *I developed a new low-speed preconditioner for the Jameson–Schmidt–Turkel scheme.*
   There are numerous low Mach number schemes in the literature. The method I developed is a single parameter modification to the widely-used Jameson–Schmidt–Turkel scheme. This makes it easy to implement and potentially widely adopted. The method has already been used in our research group for hydrofoil optimization and paraglider optimization.

3. *I developed a dissipation-based continuation method for flows with shocks.*
   Common approaches for solving challenging flows at high Mach numbers include increasing the numerical dissipation in the solution or starting the solution with a first-order scheme and switching to a second-order scheme at some point during the solution. The method I developed is an automated approach that increases robustness without sacrificing accuracy. Using this method improved the robustness and accuracy of all the RANS-based optimizations in this dissertation.

4. *I performed the first RANS-based supersonic aircraft optimization with variable wing-fuselage intersection.*
   The close coupling between the fuselage and wing aerodynamics on supersonic aircraft makes the wing-fuselage intersection an important aspect for SST design. Prior supersonic shape optimization work considering the wing-fuselage intersection was limited to the Euler equations [177]. Using the RANS equations is particularly important for the subsonic takeoff condition, which involves separated flow. RANS also allows for more accurate supersonic drag prediction.

5. *I quantified the supersonic drag penalty associated with enforcing subsonic stability for an SST using shape optimization.*
   No prior study has done a direct comparison of optimized SST designs with and without a subsonic stability constraint. By using optimization in this manner, I quantified how much

supersonic drag increases as the subsonic static margin is increased. In addition, I identified the design trends that lead to the trade-off between supersonic drag and subsonic stability.

6. *I developed a nonlinear supersonic buildup model to optimize the trim surface sizing for an SST.*
   The buildup model uses a novel approach of combining CFD data and analytic equations. This approach could also be applied to subsonic aircraft. Performing trim surface optimization on an SST is also a novel contribution. Prior studies on evaluating different trim surface configurations for supersonic aircraft have mainly been limited to wind tunnel testing a few different configurations. Using optimization, I surveyed a much larger design space.

6.3 Recommendations for future work

During the course of my research, I identified some limitations of current methods. I addressed some of these limitations in the contributions listed above. The other limitations that I did not address present opportunities for future research. In addition, there are some potential extensions of the research presented in this dissertation. These limitations and extensions are listed below:

1. *More robust geometry changes for RANS-based optimizations*
   Mesh deformation is a major limiting factor for RANS-based shape optimizations, including the results presented in Sec. 4.3. Mesh deformation can impose artificial bounds on design variables and prevent wider exploration of the design space. In addition, mesh deformation approaches do not work with topological changes to the geometry. Embedded boundary methods are one option that can enable more flexible geometry changes [178]. This would allow, for example, including trim surface sizing variables in the RANS-based optimizations in Sec. 4.3 to capture configuration-level trade-offs considering subsonic stability.

2. *Include structural considerations for the fuselage and wing*
   SST designs often have slender fuselages and thin wings, so aeroelastic effects can be important at supersonic and subsonic conditions [32, 179]. Including structural considerations in the optimization in Sec. 4.3 would result in more realistic optimized designs. Modeling the wing structure would also enable adding wing planform variables such as chord, span, and sweep to the optimization [132]. However, changing the wing planform could also change the subsonic pitch-up mechanism [17] and require capturing unsteady flow effects to accurately determine subsonic stability characteristics.

3. *Include sonic boom loudness constraints*
   SSTs have so far been limited to flying at supersonic speeds over water because of noise

concerns when flying over ground [3]. Flying at supersonic speeds over ground would open up flight routes for SSTs and increase their commercial viability. As a result, designing low-boom supersonic aircraft is an active area of research [11]. Including sonic boom loudness constraints in the shape optimization in Sec. 4.3 would enable the study of potential trade-offs between supersonic drag, sonic boom noise, and subsonic stability.

4. *Low-speed preconditioning for the RANS adjoint*
   Sec. 3.4 presents methods to speed up Newton-based flow solvers at low Mach numbers by modifying the linear system being solved at each nonlinear iteration. This process reduces the linear system stiffness but does not affect the flow solution as long as the nonlinear residual is converged tightly. When solving the adjoint equation, we cannot modify the linear system in the same manner without changing the adjoint solution. The stiffness of the residual Jacobian at low Mach numbers results in the adjoint convergence being much slower than at compressible Mach numbers. Developing methods for preconditioning the adjoint solution for low Mach number flows could significantly speed up optimizations at these flow conditions.

Appendix A

Delta Wing Pitching Moment Reference Point

The choice of pitching moment reference point in Chapter 2 requires justification because it is different from what is reported in the experimental paper by Jarrah and Ashley [73]. Another paper by Jarrah [180] presents results using the same delta wing model and experimental setup. Both papers show steady coefficient values for angles of attack from 0 to 90 deg (Fig. 4 in Jarrah and Ashley [73] and Fig. 10 in Jarrah [180]). The Reynolds numbers for the two angle-of-attack sweeps were 5.9×10^5 and 8.5×10^5, respectively, but this difference had only a small effect on the coefficients [180]. The lift and drag between the two papers are similar. However, the moment coefficient values differ substantially. Both papers report the moment reference point as the 77% root chord location. We can conclude that there is likely an error in one of the papers. In Fig. A.1, we plot the moment data from both experimental papers and the C1 results from Fig. 2.4c with the reported and shifted reference points. Defining the moment as positive nose-up about 77% root chord matches the data from Jarrah [180]. Defining the moment as positive nose-down about −23% root chord instead matches the data from Jarrah and Ashley [73]. We prefer to use the data from Jarrah and Ashley [73] to avoid ambiguities from the overlapping lines in Jarrah [180].

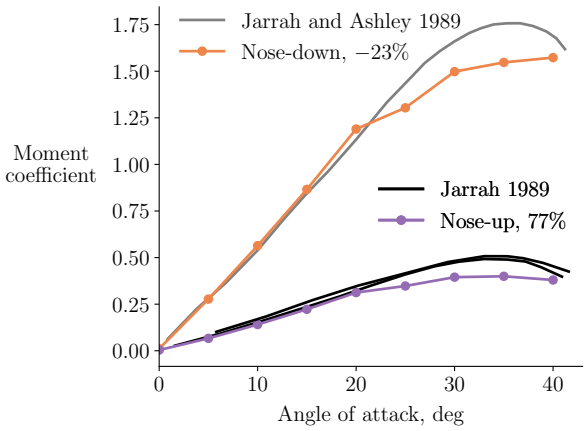


Figure A.1: Comparison of pitching moment definitions

Appendix B

pimpleFoam Instabilities

The results in Sec. 2.4 use the pimpleFoam solver in OpenFOAM with a fixed pressure boundary condition at the farfield. We initially used a zero gradient boundary condition at the farfield and found that this results in unstable solutions as the time step is decreased. This is shown by the nonphysical jump in the drag coefficient in Fig. B.1. However, the coefficients converge using a fixed pressure boundary condition. The instabilities are a result of how the pressure Poisson equation is solved. The Poisson equation is ill-posed if all boundaries have Neumann boundary conditions [181]. If a solution exists, the solution plus a constant is also a solution. In such a case, OpenFOAM uses a reference pressure approach, which avoids nonunique solutions by fixing the pressure at one cell in the domain. When the time step is large, this works well, and the coefficients are identical to the fixed pressure case. However, the reference pressure implementation in OpenFOAM makes the pressure equation ill-conditioned as the time step is refined. This ill-conditioning is documented here as a possible pitfall when using incompressible solvers in OpenFOAM.

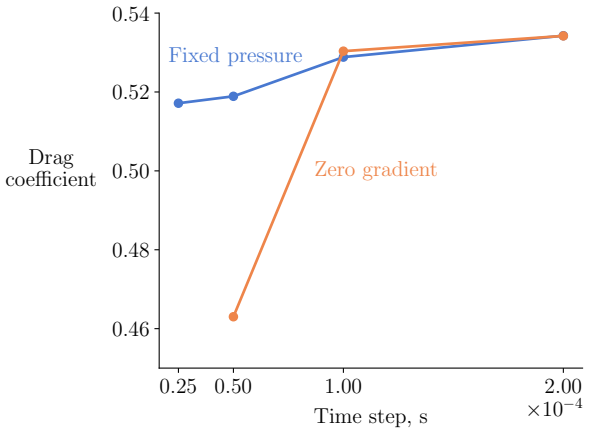


Figure B.1: Zero gradient pressure boundary conditions cause instabilities for small time steps

Bibliography

[1] Torenbeek, E., and Wittenberg, H., *Flight Physics*, Springer, 2009.

[2] Torenbeek, E., *Essentials of Supersonic Commercial Aircraft Conceptual Design*, John Wiley & Sons Ltd., 2020. doi:10.1002/9781119667063.

[3] Candel, S., "Concorde and the Future of Supersonic Transport," *Journal of Propulsion and Power*, Vol. 20, No. 1, 2004, pp. 59–68. doi:10.2514/1.9180.

[4] Magill, H. R., Bradford, J. E., Patel, A. N., and Boysen, A. A., "Life Cycle Cost Modeling of High-speed Commercial Aircraft - Final Report," Tech. Rep. NASA/CR—-20230012245, NASA, Langley Research Center, Hampton, VA, 2023.

[5] O'Neill, S., "Supersonic Travel Seeks an Encore," *Engineering*, Vol. 10, 2022, pp. 1–3. doi:10.1016/j.eng.2022.01.004.

[6] Tegler, J., "Supersonic Thump," *Aerospace America*, 2019, pp. 36–44.

[7] Rallabhandi, S. K., "Advanced Sonic Boom Prediction Using the Augmented Burgers Equation," *Journal of Aircraft*, Vol. 48, No. 4, 2011, pp. 1245–1253. doi:10.2514/1.C031248.

[8] Housman, J. A., Kenway, G. K., Jensen, J. C., and Kiris, C. C., "Efficient Near-Field to Mid-Field Sonic Boom Propagation using a High-Order Space Marching Method," *AIAA Aviation Forum*, 2019. doi:10.2514/6.2019-3487.

[9] Yamashita, R., Makino, Y., and Roe, P. L., "Fast Full-Field Simulation of Sonic Boom Using a Space Marching Method," *AIAA Journal*, Vol. 60, No. 7, 2022, pp. 4103–4112. doi:10.2514/1.J061363.

[10] Rallabhandi, S. K., and Loubeau, A., "Summary of Propagation Cases of the Third AIAA Sonic Boom Prediction Workshop," *Journal of Aircraft*, Vol. 59, No. 3, 2022, pp. 578–594. doi:10.2514/1.C036327.

[11] Park, M. A., and Carter, M. B., "Low-Boom Demonstrator Near-Field Summary for the Third AIAA Sonic Boom Prediction Workshop," *Journal of Aircraft*, Vol. 59, No. 3, 2022, pp. 563–577. doi:10.2514/1.C036323.

[12] Makino, Y., and Kroo, I., "Robust Objective Functions for Sonic-Boom Minimization," *Journal of Aircraft*, Vol. 43, No. 5, 2006, pp. 1301–1306.

[13] Rallabhandi, S. K., "Sonic-Boom Mitigation Through Aircraft Design and Adjoint Methodology," *Journal of Aircraft*, Vol. 51, No. 2, 2014, pp. 502–510. doi:10.2514/1.C032189.

[14] Sun, Y., and Smith, H., "Low-boom low-drag optimization in a multidisciplinary design analysis optimization environment," *Aerospace Science and Technology*, Vol. 94, 2019, p. 105387. doi:10.1016/j.ast.2019.105387.

[15] Morgenstern, J., Buonanno, M., Yao, J., Murugappan, M., Paliath, U., Cheung, L., Malcevic, I., Ramakrishnan, K., Pastouchenko, N., Wood, T., Martens, S., Viars, P., Tersmette, T., Lee, J., Simmons, R., Plybon, D., Alonso, J., Palacios, F., Lukaczyk, T., and Carrier, G., "Advanced Concept Studies for Supersonic Commercial Transports Entering Service in the 2018-2020 Period Phase 2," Tech. Rep. NASA/CR—2015-218719, NASA, Glenn Research Center, Cleveland, OH, 2015.

[16] Nelson, C. P., "Effects of Wing Planform on HSCT Off-design Aerodynamics," *Proceedings of the 10th Applied Aerodynamics Conference*, 1992. doi:10.2514/6.1992-2629.

[17] Benoliel, A. M., and Mason, W. H., "Pitch-up characteristics for HSCT class planforms: Survey and estimation," *12th Applied Aerodynamics Conference*, 1994. doi:10.2514/6.1994-1819.

[18] Dudley, J., Huang, X., MacMillin, P., Grossman, B., Haftka, R., and Mason, W., "Multidisciplinary Optimization of the High-speed Civil Transport," *33rd Aerospace Sciences Meeting & Exhibit*, 1995. doi:10.2514/6.1995-124.

[19] Crisafulli, P., Kaufman, M., Giunta, A., Mason, W., Grossman, B., Watson, L., and Haftka, R., "Response surface approximations for pitching moment including pitch-up in the MDO design of an HSCT," *6th Symposium on Multidisciplinary Analysis and Optimization*, 1996. doi:10.2514/6.1996-4136.

[20] Nelson, C. P., Ting, K.-Y., Mavriplis, N., Soltani, R., and Livne, E., "Supersonic Configurations at Low Speeds (SCALOS): Project Background and Progress at University of Washington," *AIAA SciTech Forum*, 2022. doi:10.2514/6.2022-1803.

[21] Ting, K.-Y., Mavriplis, N., Soltani, R., Nelson, C., and Livne, E., "Supersonic Configurations at Low Speeds (SCALOS): Model Geometry and Aerodynamic Results," *AIAA SciTech Forum*, 2022. doi:10.2514/6.2022-1800.

[22] Ting, K.-Y., Mavriplis, N., Soltani, R. M., Nelson, C. P., and Livne, E., "Supersonic Configurations at Low Speeds (SCALOS): The Aerodynamic Effects of Control Surfaces," *AIAA SciTech Forum*, 2023. doi:10.2514/6.2023-0229.

[23] Ting, K.-Y., Mavriplis, N., Soltani, R. M., Nelson, C. P., and Livne, E., "Supersonic Configurations at Low Speeds (SCALOS) Longitudinal Aerodynamics: Configuration Variations and Control Surfaces Effects," *AIAA SciTech Forum*, 2023. doi:10.2514/6.2023-0230.

[24] Mavriplis, N., Ting, K.-Y., Soltani, R. M., Nelson, C. P., and Livne, E., "Supersonic Configurations at Low Speeds (SCALOS): CFD Aided Wind Tunnel Data Corrections," *AIAA SciTech Forum*, 2023. doi:10.2514/6.2023-0231.

[25] Magee, T., Hayes, P., Khodadoust, A., and Dorgan, A., "Dynamic Stability Characteristics for Commercial Supersonic Configurations at Low-Speed Flight Conditions," *AIAA SciTech Forum*, 2022. doi:10.2514/6.2022-1802.

[26] Mavriplis, N., Ting, K.-Y., Moustafa, A., Hill, C., Soltani, R., Nelson, C., and Livne, E., "Supersonic Configurations at Low Speeds (SCALOS): Test / Simulation Correlation Studies," *AIAA SciTech Forum*, 2022. doi:10.2514/6.2022-1801.

[27] Guimarães, T. A. M., Cesnik, C. E. S., and Kolmanovsky, I., "Low Speed Aerodynamic Modeling for Control-related Considerations in Supersonic Aircraft Design," *AIAA Aviation Forum*, 2021. doi:10.2514/6.2021-2531.

[28] Bertolin, R. M., Barbosa, G. C., Cunis, T., Kolmanovsky, I., and Cesnik, C. E. S., "Gust Rejection of a Supersonic Aircraft During Final Approach," *AIAA SciTech Forum*, 2022. doi:10.2514/6.2022-2174.

[29] Cunis, T., Kolmanovsky, I., and Cesnik, C. E. S., "Integrating Nonlinear Controllability into a Multidisciplinary Design Process," *Journal of Guidance, Control, and Dynamics*, Vol. 46, No. 6, 2023, pp. 1026–1037. doi:10.2514/1.G007067.

[30] Guimarães, T. A. M., Cesnik, C. E. S., and Kolmanovsky, I., "Unsteady Vortex Lattice Linearization and Sensitivity Analyses for Control Models in Supersonic Aircraft Design," *AIAA SciTech Forum*, 2023. doi:10.2514/6.2023-0416.

[31] Guimarães, T. A. M., Cesnik, C. E. S., and Kolmanovsky, I., "An Integrated Low-Speed Aeroelastic-Flight Dynamics Framework for Modeling Supersonic Aircraft," *AIAA SciTech Forum*, 2022. doi:10.2514/6.2022-2175.

[32] Guimarães, T. A. M., Cesnik, C. E. S., and Kolmanovsky, I., "Flexibility Assessment of the Aeroelastic-flight-dynamic Behavior for Supersonic Aircraft," *AIAA SciTech Forum*, 2023. doi:10.2514/6.2023-0417.

[33] Nelson, C. P., Ting, K.-Y., Ignacio, J., Mavriplis, N., Soltani, R. M., and Livne, E., "Supersonic Configurations at Low Speeds (SCALOS): Configuration Comparison of SCALOS to the Existing Designs," *AIAA SciTech Forum*, 2023. doi:10.2514/6.2023-0228.

[34] Jameson, A., *Computational Aerodynamics*, Cambridge University Press, Cambridge, UK, 2022. doi:10.1017/9781108943345.

[35] Tinoco, E. N., Brodersen, O. P., Keye, S., Laflin, K. R., Feltrop, E., Vassberg, J. C., Mani, M., Rider, B., Wahls, R. A., Morrison, J. H., Hue, D., Roy, C. J., Mavriplis, D. J., and Murayama, M., "Summary Data from the Sixth AIAA CFD Drag Prediction Workshop: CRM Cases," *Journal of Aircraft*, Vol. 55, No. 4, 2018, pp. 1352–1379. doi:10.2514/1.C034409.

[36] Deck, S., Gand, F., Brunet, V., and Khelil, S. B., "High-fidelity simulations of unsteady civil aircraft aerodynamics: stakes and perspectives. Application of zonal detached eddy simulation," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, Vol. 372, No. 2022, 2014, p. 20130325. doi:10.1098/rsta.2013.0325.

[37] Israel, D. M., "The myth of URANS," *Journal of Turbulence*, Vol. 24, No. 8, 2023, pp. 367–392. doi:10.1080/14685248.2023.2225140.

[38] Spalart, P. R., "Detached-Eddy Simulation," *Annual Review of Fluid Mechanics*, Vol. 41, No. 1, 2009, pp. 181–202. doi:10.1146/annurev.fluid.010908.165130.

[39] Ghate, A. S., Kenway, G. K., Stich, G.-D., Browne, O. M. F., Housman, J. A., and Kiris, C. C., "Transonic Lift and Drag Predictions using Wall Modelled Large Eddy Simulations," *AIAA Scitech 2021 Forum*, 2021. doi:10.2514/6.2021-1439.

[40] Rumsey, C. L., Slotnick, J. P., and Sclafani, A. J., "Overview and Summary of the Third AIAA High Lift Prediction Workshop," *Journal of Aircraft*, Vol. 56, No. 2, 2019, pp. 621–644. doi:10.2514/1.C034940.

[41] König, B., Fares, E., Murayama, M., and Ito, Y., "PowerFLOW Simulations for the Third AIAA High-Lift Prediction Workshop," *AIAA Aerospace Sciences Meeting*, 2018. doi:10.2514/6.2018-1255.

[42] Trapani, G., Brionnaud, R., and Holman, D., "XFlow Contribution to the Third High-Lift Prediction Workshop," *Applied Aerodynamics Conference*, 2018. doi:10.2514/6.2018-2847.

[43] Rizzi, A., and Luckring, J. M., "What Was Learned in Predicting Slender Airframe Aerodynamics with the F-16XL Aircraft," *Journal of Aircraft*, Vol. 54, No. 2, 2017, pp. 444–455. doi:10.2514/1.C033569.

[44] Luckring, J. M., "The discovery and prediction of vortex flow aerodynamics," *The Aeronautical Journal*, Vol. 123, No. 1264, 2019, pp. 729–804. doi:10.1017/aer.2019.43.

[45] Martins, J. R. R. A., and Ning, A., *Engineering Design Optimization*, Cambridge University Press, Cambridge, UK, 2022. doi:10.1017/9781108980647, URL https://mdobook.github.io.

[46] Lyu, Z., Kenway, G. K. W., and Martins, J. R. R. A., "Aerodynamic Shape Optimization Investigations of the Common Research Model Wing Benchmark," *AIAA Journal*, Vol. 53, No. 4, 2015, pp. 968–985. doi:10.2514/1.J053318.

[47] Yu, Y., Lyu, Z., Xu, Z., and Martins, J. R. R. A., "On the Influence of Optimization Algorithm and Starting Design on Wing Aerodynamic Shape Optimization," *Aerospace Science and Technology*, Vol. 75, 2018, pp. 183–199. doi:10.1016/j.ast.2018.01.016.

[48] Jameson, A., "Aerodynamic Design via Control Theory," *Journal of Scientific Computing*, Vol. 3, No. 3, 1988, pp. 233–260. doi:10.1007/BF01061285.

[49] Reuther, J., Alonso, J. J., Rimlinger, M. J., and Jameson, A., "Aerodynamic shape optimization of supersonic aircraft configurations via an adjoint formulation on distributed memory parallel computers," *Computers & Fluids*, Vol. 28, No. 4, 1999, pp. 675–700. doi:10.1016/S0045-7930(98)00050-4.

[50] Cliff, S. E., Reuther, J. J., Saunders, D. A., and Hicks, R. M., "Single-Point and Multipoint Aerodynamic Shape Optimization of High-Speed Civil Transport," *Journal of Aircraft*, Vol. 38, No. 6, 2001, pp. 997–1005.

[51] Martins, J. R. R. A., Alonso, J. J., and Reuther, J. J., "A Coupled-Adjoint Sensitivity Analysis Method for High-Fidelity Aero-Structural Design," *Optimization and Engineering*, Vol. 6, No. 1, 2005, pp. 33–62. doi:10.1023/B:OPTE.0000048536.47956.62.

[52] Martins, J. R. R. A., Alonso, J. J., and Reuther, J. J., "High-Fidelity Aerostructural Design Optimization of a Supersonic Business Jet," *Journal of Aircraft*, Vol. 41, No. 3, 2004, pp. 523–530. doi:10.2514/1.11478.

[53] Hwang, J. T., and Martins, J. R. R. A., "A computational architecture for coupling heterogeneous numerical models and computing coupled derivatives," *ACM Transactions on Mathematical Software*, Vol. 44, No. 4, 2018, p. Article 37. doi:10.1145/3182393.

[54] Gray, J. S., Hwang, J. T., Martins, J. R. R. A., Moore, K. T., and Naylor, B. A., "OpenMDAO: An open-source framework for multidisciplinary design, analysis, and optimization," *Structural and Multidisciplinary Optimization*, Vol. 59, No. 4, 2019, pp. 1075–1104. doi:10.1007/s00158-019-02211-z.

[55] Jasa, J., Brelje, B., Gray, J., Mader, C. A., and Martins, J. R. R. A., "Large-Scale Path-Dependent Optimization of Supersonic Aircraft," *Aerospace*, Vol. 7, No. 152, 2020. doi:10.3390/aerospace7100152.

[56] Martins, J. R. R. A., "Aerodynamic Design Optimization: Challenges and Perspectives," *Computers & Fluids*, Vol. 239, 2022, p. 105391. doi:10.1016/j.compfluid.2022.105391.

[57] Lan, C. E., and Hsu, C. H., "Effects of vortex breakdown on longitudinal and lateral-directional aerodynamics of slender wings by the suction analogy," *9th Atmospheric Flight Mechanics Conference*, 1982. doi:10.2514/6.1982-1385.

[58] Carlson, H. W., and Walkley, K. B., "A Computer Program for Wing Subsonic Aerodynamic Performance Estimates Including Attainable Thrust and Vortex Lift Effects," Tech. Rep. NASA CR 3515, NASA, Langley Research Center, Hampton, VA, 1982.

[59] Polhamus, E. C., "Predictions of Vortex-Lift Characteristics by a Leading-Edge Suction Analogy," *Journal of Aircraft*, Vol. 8, No. 4, 1971, pp. 193–199. doi:10.2514/3.44254.

[60] Kulfan, R. M., "Wing Airfoil Shape Effects on the Development of Leading Edge Vortices," *5th Atmospheric Flight Mechanics Conference for Future Space Systems*, 1979. doi:10.2514/6.1979-1675.

[61] Kulfan, R. M., "Wing Geometry Effects on Leading Edge Vortices," *Aircraft Systems and Technology Meeting*, 1979. doi:10.2514/6.1979-1872.

[62] Traub, L. W., "Extending the Leading-Edge Suction Analogy to Nonslender Delta Wings," *Journal of Aircraft*, Vol. 55, No. 5, 2018, pp. 2174–2177. doi:10.2514/1.C034939.

[63] Mitchell, A. M., Barberis, D., Molton, P., and Délery, J., "Oscillation of Vortex Breakdown Location and Blowing Control of Time-Averaged Location," *AIAA Journal*, Vol. 38, No. 5, 2000, pp. 793–803. doi:10.2514/2.1059.

[64] Soemarwoto, B. I., and Boelens, O. J., "Simulation of vortical flow over a slender delta wing experiencing vortex breakdown," Tech. Rep. NLR-TP-2003-396, NLR, Netherlands, 2003.

[65] Görtz, S., "Detached-Eddy Simulations of a Full-Span Delta Wing at High Incidence," *21st Applied Aerodynamics Conference*, 2003. doi:10.2514/6.2003-4216.

[66] Morton, S., "Detached-Eddy Simulations of Vortex Breakdown over a 70-Degree Delta Wing," *Journal of Aircraft*, Vol. 46, No. 3, 2009, pp. 746–755. doi:10.2514/1.4659.

[67] François, D. G., Probst, S., Knopp, T., Grabe, C., Landa, T., and Radespiel, R., "Numerical Simulation of the Streamwise Transport of a Delta Wing Leading-Edge Vortex," *Journal of Aircraft*, Vol. 58, No. 6, 2021, pp. 1281–1293. doi:10.2514/1.C036273.

[68] Lofthouse, A. J., and Cummings, R. M., "Numerical Simulations of the F-16XL at Flight-Test Conditions Using Delayed Detached-Eddy Simulation," *Journal of Aircraft*, Vol. 54, No. 6, 2017, pp. 2077–2099. doi:10.2514/1.C034045.

[69] Tomac, M., Jirasek, A., and Rizzi, A., "Hybrid Reynolds-Averaged Navier–Stokes/Large-Eddy Simulations of F-16XL in Low-Speed High-Alpha Flight," *Journal of Aircraft*, Vol. 54, No. 6, 2017, pp. 2070–2076. doi:10.2514/1.C034283.

[70] Forsythe, J. R., Squires, K. D., Wurtzler, K. E., and Spalart, P. R., "Detached-Eddy Simulation of the F-15E at High Alpha," *Journal of Aircraft*, Vol. 41, No. 2, 2004, pp. 193–200. doi:10.2514/1.2111.

[71] Cummings, R. M., and Schütte, A., "Detached-Eddy Simulation of the vortical flow field about the VFE-2 delta wing," *Aerospace Science and Technology*, Vol. 24, No. 1, 2013, pp. 66–76. doi:10.1016/j.ast.2012.02.007.

[72] Jeans, T. L., McDaniel, D. R., Cummings, R. M., and Mason, W. H., "Aerodynamic Analysis of a Generic Fighter Using Delayed Detached-Eddy Simulation," *Journal of Aircraft*, Vol. 46, No. 4, 2009, pp. 1326–1339. doi:10.2514/1.40955.

[73] Jarrah, M. A., and Ashley, H., "Impact of flow unsteadiness on maneuvers and loads of agile aircraft," *30th Structures, Structural Dynamics and Materials Conference*, 1989. doi:10.2514/6.1989-1282.

[74] Jarrah, M.-A. M., "Unsteady aerodynamics of delta wings performing maneuvers to high angle of attack," Ph.D. thesis, Stanford University, 1989.

[75] Hummel, D., "Effects of Boundary Layer Formation on the Vortical Flow above Slender Delta Wings," *RTO AVT Symposium on Enhancement of NATO Military Flight Vehicle Performance by Management of Interacting Boundary Layer Transition and Separation*, 2004. doi:10.14339/RTO-MP-AVT-111-30.

[76] Gordnier, R. E., Visbal, M. R., Gursul, I., and Wang, Z., "Computational and Experimental Investigation of a Nonslender Delta Wing," *AIAA Journal*, Vol. 47, No. 8, 2009, pp. 1811–1825. doi:10.2514/1.37848.

[77] Mader, C. A., Kenway, G. K. W., Yildirim, A., and Martins, J. R. R. A., "ADflow: An open-source computational fluid dynamics solver for aerodynamic and multidisciplinary optimization," *Journal of Aerospace Information Systems*, Vol. 17, No. 9, 2020, pp. 508–527. doi:10.2514/1.I010796.

[78] Spalart, P., and Allmaras, S., "A One-Equation Turbulence Model for Aerodynamic Flows," *La Recherche Aerospatiale*, Vol. 1, 1994, pp. 5–21.

[79] Jameson, A., Schmidt, W., and Turkel, E., "Numerical Solution of the Euler Equations by Finite Volume Methods Using Runge–Kutta Time Stepping Schemes," *14th Fluid and Plasma Dynamics Conference*, 1981. doi:10.2514/6.1981-1259.

[80] Yildirim, A., Kenway, G. K. W., Mader, C. A., and Martins, J. R. R. A., "A Jacobian-free approximate Newton–Krylov startup strategy for RANS simulations," *Journal of Computational Physics*, Vol. 397, 2019, p. 108741. doi:10.1016/j.jcp.2019.06.018.

[81] Secco, N., Kenway, G. K. W., He, P., Mader, C. A., and Martins, J. R. R. A., "Efficient Mesh Generation and Deformation for Aerodynamic Shape Optimization," *AIAA Journal*, Vol. 59, No. 4, 2021, pp. 1151–1168. doi:10.2514/1.J059491.

[82] Chan, W. M., and Steger, J. L., "Enhancements of a three-dimensional hyperbolic grid generation scheme," *Applied Mathematics and Computation*, Vol. 51, No. 2–3, 1992, pp. 181–205. doi:10.1016/0096-3003(92)90073-A.

[83] Kenway, G. K. W., Secco, N., Martins, J. R. R. A., Mishra, A., and Duraisamy, K., "An Efficient Parallel Overset Method for Aerodynamic Shape Optimization," *Proceedings of the 58th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, AIAA SciTech Forum*, Grapevine, TX, 2017. doi:10.2514/6.2017-0357.

[84] Weller, H. G., Tabor, G., Jasak, H., and Fureby, C., "A tensorial approach to computational continuum mechanics using object-oriented techniques," *Computers in Physics*, Vol. 12, No. 6, 1998, pp. 620–631. doi:10.1063/1.168744.

[85] Spalart, P. R., Deck, S., Shur, M. L., Squires, K. D., Strelets, M. K., and Travin, A., "A New Version of Detached-eddy Simulation, Resistant to Ambiguous Grid Densities," *Theoretical and Computational Fluid Dynamics*, Vol. 20, No. 3, 2006, pp. 181–195. doi:10.1007/s00162-006-0015-0.

[86] Warming, R. F., and Beam, R. M., "Upwind second-order difference schemes and applications in aerodynamic flows," *AIAA Journal*, Vol. 14, No. 9, 1976, pp. 1241–1249. doi:10.2514/3.61457.

[87] Issa, R. I., "Solution of the implicitly discretised fluid flow equations by operator-splitting," *Journal of Computational Physics*, Vol. 62, No. 1, 1986, pp. 40–65. doi:10.1016/0021-9991(86)90099-9.

[88] Hummel, D. J., "The International Vortex Flow Experiment 2 (VFE-2): Background, objectives and organization," *Aerospace Science and Technology*, Vol. 24, No. 1, 2013, pp. 1–9. doi:10.1016/j.ast.2012.08.008.

[89] Ashton, N., West, A., Lardeau, S., and Revell, A., "Assessment of RANS and DES methods for realistic automotive models," *Computers & Fluids*, Vol. 128, 2016, pp. 1–15. doi:10.1016/j.compfluid.2016.01.008.

[90] Fisher, R. A., "Tests of significance in harmonic analysis," *Proceedings of the Royal Society of London A*, Vol. 125, No. 796, 1929, pp. 54–59. doi:10.1098/rspa.1929.0151.

[91] Quinn, B. G., "Fisher's g Revisited," *International Statistical Review*, Vol. 89, No. 2, 2021, pp. 402–419. doi:10.1111/insr.12437.

[92] Bons, N. P., and Martins, J. R. R. A., "Aerostructural Design Exploration of a Wing in Transonic Flow," *Aerospace*, Vol. 7, No. 8, 2020, p. 118. doi:10.3390/aerospace7080118.

[93] Gleize, V., and Costes, M., "Low-Mach-Number Preconditioning Applied to Turbulent Helicopter Fuselage Flowfield Computation," *AIAA Journal*, Vol. 41, No. 4, 2003, pp. 653–662. doi:10.2514/2.1995.

[94] Colin, Y., Deniau, H., and Boussuge, J.-F., "A robust low speed preconditioning formulation for viscous flow computations," *Computers & Fluids*, Vol. 47, No. 1, 2011, pp. 1–15. doi:10.1016/j.compfluid.2011.01.015.

[95] Turkel, E., "Preconditioned methods for solving the incompressible and low speed compressible equations," *Journal of Computational Physics*, Vol. 72, No. 2, 1987, pp. 277–298. doi:10.1016/0021-9991(87)90084-2.

[96] Turkel, E., "Preconditioning techniques in computational fluid dynamics," *Annual Review of Fluid Mechanics*, Vol. 31, No. 1, 1999, pp. 385–416. doi:10.1146/annurev.fluid.31.1.385.

[97] Weiss, J. M., and Smith, W. A., "Preconditioning applied to variable and constant density flows," *AIAA Journal*, Vol. 33, No. 11, 1995, pp. 2050–2057. doi:10.2514/3.12946.

[98] Nemec, M., and Zingg, D. W., "Aerodynamic Computations Using the Convective-Upstream Split-Pressure Scheme with Local Preconditioning," *AIAA Journal*, Vol. 38, No. 3, 2000, pp. 402–410. doi:10.2514/2.998.

[99] van Leer, B., Lee, W.-T., and Roe, P., "Characteristic time-stepping or local preconditioning of the Euler equations," *10th Computational Fluid Dynamics Conference*, 1991. doi:10.2514/6.1991-1552.

[100] Lee, D., "The Design of Local Navier–Stokes Preconditioning for Compressible Flow," *Journal of Computational Physics*, Vol. 144, No. 2, 1998, pp. 460–483. doi:10.1006/jcph.1998.5994.

[101] Knoll, D. A., McHugh, P. R., and Keyes, D. E., "Newton–Krylov methods for low-Mach-number compressible combustion," *AIAA Journal*, Vol. 34, No. 5, 1996, pp. 961–967. doi:10.2514/3.13174.

[102] Weston, B., Nourgaliev, R., Delplanque, J.-P., and Barker, A. T., "Preconditioning a Newton–Krylov solver for all-speed melt pool flow physics," *Journal of Computational Physics*, Vol. 397, 2019, p. 108847. doi:10.1016/j.jcp.2019.07.045.

[103] Mary, I., Sagaut, P., and Deville, M., "An algorithm for low Mach number unsteady flows," *Computers & Fluids*, Vol. 29, No. 2, 2000, pp. 119–147. doi:10.1016/S0045-7930(99)00007-9.

[104] Kaushik, D. K., Keyes, D. E., and Smith, B. F., "Newton–Krylov–Schwarz Methods for Aerodynamics Problems: Compressible and Incompressible Flows on Unstructured Grids," *11th International Conference on Domain Decomposition Methods*, 1998.

[105] Olawsky, F., Infed, F., and Auweter-Kurtz, M., "Preconditioned Newton Method for Computing Supersonic and Hypersonic Nonequilibrium Flows," *Journal of Spacecraft and Rockets*, Vol. 41, No. 6, 2004, pp. 907–914. doi:10.2514/1.4010.

[106] Biedron, R. T., Carlson, J.-R., Derlaga, J. M., Gnoffo, P. A., Hammond, D. P., Jacobson, K. E., Jones, W. T., Kleb, B., Lee-Rausch, E. M., Nielsen, E. J., Park, M. A., Rumsey, C. L., Thomas, J. L., Thompson, K. B., Walden, A. C., Wang, L., and Wood, W. A., "FUN3D Manual: 13.7," Tech. Rep. NASA/TM–20205010139, NASA Langley Research Center, Hampton, Virginia, November 2020.

[107] Berger, M., Aftosmis, M. J., and Murman, S. M., "Analysis of Slope Limiters on Irregular Grids," *43rd AIAA Aerospace Sciences Meeting and Exhibit*, 2005. doi:10.2514/6.2005-490.

[108] Öhrman, J., "Evaluation of a CFD method for estimating aerodynamic loads on external stores on JAS 39 Gripen," Ph.D. thesis, Umeå University, 2011.

[109] Aprovitola, A., Dyblenko, O., Pezzella, G., and Viviani, A., "Aerodynamic Analysis of a Supersonic Transport Aircraft at Low and High Speed Flow Conditions," *Aerospace*, Vol. 9, No. 8, 2022, p. 411. doi:10.3390/aerospace9080411.

[110] Jameson, A., "Origins and Further Development of the Jameson–Schmidt–Turkel Scheme," *AIAA Journal*, Vol. 55, No. 5, 2017, pp. 1487–1510. doi:10.2514/1.J055493.

[111] Martinelli, L., "Calculations of Viscous Flows with a Multigrid Method," Ph.D. thesis, Princeton University, 1987.

[112] Wang, Z., Fidkowski, K., Abgrall, R., Bassi, F., Caraeni, D., Cary, A., Deconinck, H., Hartmann, R., Hillewaert, K., Huynh, H., Kroll, N., May, G., Persson, P.-O., van Leer, B., and Visbal, M., "High-order CFD methods: current status and perspective," *International Journal for Numerical Methods in Fluids*, Vol. 72, No. 8, 2013, pp. 811–845. doi:10.1002/fld.3767.

[113] Swanson, R. C., and Turkel, E., "On central-difference and upwind schemes," *Journal of Computational Physics*, Vol. 101, No. 2, 1992, pp. 292–306. doi:10.1016/0021-9991(92)90007-L.

[114] Jameson, A., "Analysis and Design of Numerical Schemes for Gas Dynamics 1—Artificial Diffusion, Upwind Biasing, Limiters and Their Effect on Accuracy and Multigrid Convergence," *International Journal of Computational Fluid Dynamics*, Vol. 4, No. 3–4, 1995, pp. 171–218. doi:10.1080/10618569508904524.

[115] Gustafsson, B., "Unsymmetric hyperbolic systems and the Euler equations at low Mach numbers," *Journal of Scientific Computing*, Vol. 2, No. 2, 1987, pp. 123–136. doi:10.1007/BF01061482.

[116] Rieper, F., "A low-Mach number fix for Roe's approximate Riemann solver," *Journal of Computational Physics*, Vol. 230, No. 13, 2011, pp. 5263–5287. doi:10.1016/j.jcp.2011.03.025.

[117] Liou, M.-S., "A sequel to AUSM, Part II: AUSM+-up for all speeds," *Journal of Computational Physics*, Vol. 214, No. 1, 2006, pp. 137–170. doi:10.1016/j.jcp.2005.09.020.

[118] Berberich, J. P., and Klingenberg, C., "Entropy Stable Numerical Fluxes for Compressible Euler Equations Which Are Suitable for All Mach Numbers," *Recent Advances in Numerical Methods for Hyperbolic PDE Systems*, 2021. doi:10.1007/978-3-030-72850-2_8.

[119] Vassberg, J. C., Tinoco, E. N., Mani, M., Brodersen, O. P., Eisfeld, B., Wahls, R. A., Morrison, J. H., Zickuhr, T., Laflin, K. R., and Mavriplis, D. J., "Abridged Summary of the Third AIAA Computational Fluid Dynamics Drag Prediction Workshop," *Journal of Aircraft*, Vol. 45, No. 3, 2008, pp. 781–798. doi:10.2514/1.30572.

[120] Drela, M., "XFOIL: An Analysis and Design System for Low Reynolds Number Airfoils," *Low Reynolds Number Aerodynamics*, edited by T. J. Mueller, Springer Berlin Heidelberg, Berlin, Heidelberg, 1989, pp. 1–12. doi:10.1007/978-3-642-84010-4_1.

[121] Guillard, H., and Viozat, C., "On the behaviour of upwind schemes in the low Mach number limit," *Computers & Fluids*, Vol. 28, No. 1, 1999, pp. 63–86. doi:10.1016/j.camwa.2018.02.028.

[122] Chen, S., Yan, C., and Xiang, X., "Effective low-Mach number improvement for upwind schemes," *Computers & Mathematics with Applications*, Vol. 75, No. 10, 2018, pp. 3737–3755. doi:10.1016/j.camwa.2018.02.028.

[123] Roache, P. J., "Perspective: A Method for Uniform Reporting of Grid Refinement Studies," *Journal of Fluids Engineering*, Vol. 116, No. 3, 1994, pp. 405–413. doi:10.1115/1.2910291.

[124] Brown, D. A., and Zingg, D. W., "Performance of a Newton–Krylov–Schur Algorithm for Solving Steady Turbulent Flows," *AIAA Journal*, Vol. 54, No. 9, 2016, pp. 2645–2658. doi:10.2514/1.J054513.

[125] Turkel, E., and Vatsa, V. N., "Local preconditioners for steady and unsteady flow applications," *ESAIM: Mathematical Modelling and Numerical Analysis*, Vol. 39, No. 3, 2005, pp. 515–535. doi:10.1051/m2an:2005021.

[126] Moragues Ginard, M., Vázquez, M., and Houzeaux, G., "Local preconditioning and variational multiscale stabilization for Euler compressible steady flow," *Computer Methods in Applied Mechanics and Engineering*, Vol. 305, 2016, pp. 468–500. doi:10.1016/j.cma.2016.02.027.

[127] Saad, Y., and Schultz, M. H., "GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems," *SIAM Journal on Scientific and Statistical Computing*, Vol. 7, No. 3, 1986, pp. 856–869. doi:10.1137/0907058.

[128] Hicken, J., and Zingg, D., "Globalization Strategies for Inexact-Newton Solvers," *19th AIAA Computational Fluid Dynamics*, American Institute of Aeronautics and Astronautics, 2009. doi:10.2514/6.2009-4139.

[129] Hicken, J., Buckley, H., Osusky, M., and Zingg, D., "Dissipation-based continuation: a globalization for inexact-Newton solvers," *20th AIAA Computational Fluid Dynamics Conference*, American Institute of Aeronautics and Astronautics, 2011. doi:10.2514/6.2011-3237.

[130] Carrier, G., "Single and Multi-Point Aerodynamic Optimizations of a Supersonic Transport Aircraft Wing Using Optimization Strategies Involving Adjoint Method and Genetic Algorithm," *Proceedings of ERCOFTAC Workshop*, 2006.

[131] Sasaki, D., Obayashi, S., and Nakahashi, K., "Navier–Stokes Optimization of Supersonic Wings with Four Objectives Using Evolutionary Algorithm," *Journal of Aircraft*, Vol. 39, No. 4, 2002, pp. 621–629. doi:10.2514/2.2974.

[132] Bons, N. P., Martins, J. R. R. A., Mader, C. A., McMullen, M., and Suen, M., "High-fidelity Aerostructural Optimization Studies of the Aerion AS2 Supersonic Business Jet," *Proceedings of the AIAA Aviation Forum*, 2020. doi:10.2514/6.2020-3182.

[133] Mangano, M., and Martins, J. R. R. A., "Multipoint Aerodynamic Shape Optimization for Subsonic and Supersonic Regimes," *Journal of Aircraft*, Vol. 58, No. 3, 2021, pp. 650–662. doi:10.2514/1.C036216.

[134] Kenway, G. K. W., and Martins, J. R. R. A., "Multipoint High-Fidelity Aerostructural Optimization of a Transport Aircraft Configuration," *Journal of Aircraft*, Vol. 51, No. 1, 2014, pp. 144–160. doi:10.2514/1.C032150.

[135] Mader, C. A., and Martins, J. R. R. A., "Stability-Constrained Aerodynamic Shape Optimization of Flying Wings," *Journal of Aircraft*, Vol. 50, No. 5, 2013, pp. 1431–1449. doi:10.2514/1.C031956.

[136] Lyu, Z., and Martins, J. R. R. A., "Aerodynamic Design Optimization Studies of a Blended-Wing-Body Aircraft," *Journal of Aircraft*, Vol. 51, No. 5, 2014, pp. 1604–1617. doi:10.2514/1.C032491.

[137] Reist, T. A., and Zingg, D. W., "High-Fidelity Aerodynamic Shape Optimization of a Lifting-Fuselage Concept for Regional Aircraft," *Journal of Aircraft*, Vol. 54, No. 3, 2017, pp. 1085–1097. doi:10.2514/1.C033798.

[138] Liou, M. F., Kim, H., Lee, B., and Liou, M. S., "Aerodynamic design of integrated propulsion–airframe confguration of a hybrid wing body aircraft," *Shock Waves*, Vol. 29, 2019, pp. 1043—-1064. doi:10.1007/s00193-019-00933-z.

[139] Lee, A. G., Reist, T. A., and Zingg, D. W., "Further Exploration of Regional-Class Hybrid Wing-Body Aircraft Through Multifidelity Optimization," *AIAA SciTech Forum*, 2021. doi:10.2514/6.2021-0014.

[140] Mader, C. A., and Martins, J. R. R. A., "Computing Stability Derivatives and their Gradients for Aerodynamic Shape Optimization," *AIAA Journal*, Vol. 52, No. 11, 2014, pp. 2533–2546. doi:10.2514/1.J052922.

[141] Lee, S., and Kim, C., "Design Optimization of Lambda-Wing Planform and Vortex Generators for Longitudinal Instability Alleviation," *Journal of Aircraft*, 2024. doi:10.2514/1.C037495.

[142] Li, W., and Geiselhart, K., "Multidisciplinary Design Optimization of Low-Boom Supersonic Aircraft with Mission Constraints," *AIAA Journal*, Vol. 59, No. 1, 2021, pp. 165–179. doi:10.2514/1.J059237.

[143] Li, W., and Geiselhart, K., "Integration of Low-Fidelity MDO and CFD-Based Redesign of Low-Boom Supersonic Transports," *AIAA Journal*, Vol. 59, No. 10, 2021, pp. 3923–3936. doi:10.2514/1.J060368.

[144] Chan, W., Gomez, R., Rogers, S., and Buning, P., "Best Practices in Overset Grid Generation," *32nd AIAA Fluid Dynamics Conference and Exhibit*, American Institute of Aeronautics and Astronautics, 2002. doi:10.2514/6.2002-3191.

[145] Chan, W. M., "Enhancements to the Hybrid Mesh Approach to Surface Loads Integration on Overset Structured Grids," *19th AIAA Computational Fluid Dynamics*, 2009. doi:10.2514/6.2009-3990.

[146] Kenway, G. K. W., Mader, C. A., He, P., and Martins, J. R. R. A., "Effective Adjoint Approaches for Computational Fluid Dynamics," *Progress in Aerospace Sciences*, Vol. 110, 2019, p. 100542. doi:10.1016/j.paerosci.2019.05.002.

[147] Sederberg, T. W., and Parry, S. R., "Free-form Deformation of Solid Geometric Models," *SIGGRAPH Comput. Graph.*, Vol. 20, No. 4, 1986, pp. 151–160. doi:10.1145/15886.15903.

[148] Hajdik, H. M., Yildirim, A., Wu, N., Brelje, B. J., Seraj, S., Mangano, M., Anibal, J. L., Jonsson, E., Adler, E. J., Mader, C. A., Kenway, G. K. W., and Martins, J. R. R. A., "pyGeo: A geometry package for multidisciplinary design optimization," *Journal of Open Source Software*, Vol. 8, No. 87, 2023, p. 5319. doi:10.21105/joss.05319.

[149] Luke, E., Collins, E., and Blades, E., "A Fast Mesh Deformation Method Using Explicit Interpolation," *Journal of Computational Physics*, Vol. 231, No. 2, 2012, pp. 586–601. doi:10.1016/j.jcp.2011.09.021.

[150] Wu, N., Kenway, G., Mader, C. A., Jasa, J., and Martins, J. R. R. A., "pyOptSparse: A Python framework for large-scale constrained nonlinear optimization of sparse systems," *Journal of Open Source Software*, Vol. 5, No. 54, 2020, p. 2564. doi:10.21105/joss.02564.

[151] Gill, P. E., Murray, W., and Saunders, M. A., "SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization," *SIAM Review*, Vol. 47, No. 1, 2005, pp. 99–131. doi:10.1137/S0036144504446096.

[152] Welge, H. R., Bonet, J., Magee, T., Chen, D., Hollowell, S., Kutzmann, A., Mortlock, A., Stengle, J., Nelson, C., Adamson, E., Baughcum, S., Britt, R. T., Miller, G., and Tai, J., "N+2 Supersonic Concept Development and Systems Integration," Tech. Rep. NASA/CR-2010-216842, NASA, Langley Research Center, Hampton, VA, 2010.

[153] Liebeck, R. H., "Design of the Blended Wing Body Subsonic Transport," *Journal of Aircraft*, Vol. 41, No. 1, 2004, pp. 10–25. doi:10.2514/1.9084.

[154] Yildirim, A., Mader, C. A., and Martins, J. R. R. A., "A Surface Mesh Deformation Method Near Component Intersections for High-Fidelity Design Optimization," *Engineering with Computers*, 2021. doi:10.1007/s00366-020-01247-w.

[155] Secco, N. R., Jasa, J. P., Kenway, G. K. W., and Martins, J. R. R. A., "Component-based Geometry Manipulation for Aerodynamic Shape Optimization with Overset Meshes," *AIAA Journal*, Vol. 56, No. 9, 2018, pp. 3667–3679. doi:10.2514/1.J056550.

[156] Gaffuri, M., and Brezillon, J., "High lift devices design of a supersonic transport aircraft based on 3D computational fluid dynamics," *European Congress on Computational Methods in Applied Sciences and Engineering*, 2012.

[157] Nicholls, K. P., "Flap systems on supersonic transport aircraft," *20th Congress of the International Council of the Aeronautical Sciences*, 1996.

[158] Reist, T. A., Koo, D., and Zingg, D. W., "Aircraft Cruise Drag Reduction Through Variable Camber Using Existing Control Surfaces," *Journal of Aircraft*, Vol. 59, No. 6, 2022, pp. 1406–1415. doi:10.2514/1.C036754.

[159] Martins, J. R. R. A., Sturdza, P., and Alonso, J. J., "The Complex-Step Derivative Approximation," *ACM Transactions on Mathematical Software*, Vol. 29, No. 3, 2003, pp. 245–262. doi:10.1145/838250.838251.

[160] Lei, Z., "Flow Simulation of a Supersonic Transport Configuration at Low-Speed and High-Lift Conditions," *Journal of Aircraft*, Vol. 45, No. 5, 2008, pp. 1514–1521. doi:10.2514/1.33856.

[161] Jeong, J., and Hussain, F., "On the identification of a vortex," *Journal of Fluid Mechanics*, Vol. 285, 1995, pp. 69–94. doi:10.1017/S0022112095000462.

[162] Kendall, E. R., "The theoretical minimum induced drag of three-surface airplanes in trim," *Journal of Aircraft*, Vol. 22, No. 10, 1985, pp. 847–854. doi:10.2514/3.45214.

[163] Kroo, I., "A General Approach to Multiple Lifting Surface Design and Analysis," *Aircraft Design Systems and Operations Meeting*, 1984. doi:10.2514/6.1984-2507.

[164] Rokhsaz, K., and Selberg, B. P., "Three-Surface Aircraft – Optimum vs Typical," *Journal of Aircraft*, Vol. 26, No. 8, 1989, pp. 699–704. doi:10.2514/3.45827.

[165] Owens, D. B., and Perkins, J. N., "Stability and Control of a Three-Surface, Forward-Swept Wing Configuration," *Journal of Aircraft*, Vol. 33, No. 6, 1996, pp. 1206–1209. doi:10.2514/3.47079.

[166] Strohmeyer, D., Seubert, R., Heinze, W., Österheld, C., and Fornasie, L., "Three Surface Aircraft - A Concept for Future Transport Aircraft," *38th AIAA Aerospace Sciences Meeting and Exhibit*, 2000. doi:10.2514/6.2000-566.

[167] Lacey, D. W., "Aerodynamic Characteristics of the Close-Coupled Canard as Applied to Low-to-Moderate Swept Wings. Volume 3. Transonic-Supersonic Speed Regime," Tech. rep., Naval Ship Research and Development Center, Bethesda, Maryland, USA, 1979.

[168] Agnew, J. W., and Hess Jr., J. R., "Benefits of Aerodynamic Interaction to the Three-Surface Configuration," *Journal of Aircraft*, Vol. 17, No. 11, 1980, pp. 823–827. doi:10.2514/3.57971.

[169] Covell, P. F., "Supersonic Aerodynamic Characteristics of Canard, Tailless, and Aft-Tail Configurations for Two Wing Planforms," Tech. Rep. NASA-TP-2434, NASA Langley Research Center, Hampton, Virginia, USA, 1985.

[170] Raymer, D. P., *Aircraft Design: A Conceptual Approach*, 5th ed., AIAA, Reston, VA, 2012.

[171] Finck, R. D., "USAF Stability and Control DATCOM," Tech. rep., Air Force Flight Dynamics Laboratory, Wright–Patterson Air Force Base, Ohio, USA, 1978.

[172] Hwang, J. T., and Martins, J. R. R. A., "A fast-prediction surrogate model for large datasets," *Aerospace Science and Technology*, Vol. 75, 2018, pp. 74–87. doi:10.1016/j.ast.2017.12.030.

[173] Bouhlel, M. A., Hwang, J. T., Bartoli, N., Lafage, R., Morlier, J., and Martins, J. R. R. A., "A Python surrogate modeling framework with derivatives," *Advances in Engineering Software*, Vol. 135, 2019, p. 102662. doi:10.1016/j.advengsoft.2019.03.005.

[174] Tangler, J. L., and Ostowari, C., "Horizontal axis wind turbine post stall airfoil characteristics synthesization," Conference paper presented at the DOE/NASA Wind Turbine Technology Workshop, May 1984. In Collected Papers on Wind Turbine Technology, NASA-CR-195432, May 1995.

[175] Chauhan, S. S., and Martins, J. R. R. A., "Tilt-wing eVTOL takeoff trajectory optimization," *Journal of Aircraft*, Vol. 57, No. 1, 2020, pp. 93–112. doi:10.2514/1.C035476.

[176] Kreisselmeier, G., and Steinhauser, R., "Systematic Control Design by Optimizing a Vector Performance Index," *International Federation of Active Controls Symposium on Computer-Aided Design of Control Systems, Zurich, Switzerland*, 1979. doi:10.1016/S1474-6670(17)65584-8.

[177] Nemec, M., and Aftosmis, M. J., "Parallel Adjoint Framework for Aerodynamic Shape Optimization of Component-Based Geometry," *49th AIAA Aerospace Sciences Meeting*, 2011. doi:10.2514/6.2011-1249.

[178] Ho, J., and Farhat, C., "Aerodynamic optimization with large shape and topology changes using a differentiable embedded boundary method," *Journal of Computational Physics*, Vol. 488, 2023, p. 112191. doi:10.1016/j.jcp.2023.112191.

[179] Bhatia, K. G., and Wertheimer, J., "Aeroelastic Challenges for a High Speed Civil Transport," *34th Structures, Structural Dynamics and Materials Conference*, 1993. doi:10.2514/6.1993-1478.

[180] Jarrah, M.-A. M., "Low speed wind tunnel investigation of the flow about delta wing, oscillating in pitch to very high angle of attack," *27th Aerospace Sciences Meeting*, 1989. doi:10.2514/6.1989-295.

[181] LeVeque, R. J., *Finite Difference Methods for Ordinary and Partial Differential Equations*, Society for Industrial and Applied Mathematics, 2007. doi:10.1137/1.9780898717839.