ALGORITHMS FOR AUTOMATIC FEEDBACK CONTROL OF AERODYNAMIC FLOWS

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

Karthik Palaniappan
June 2007

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Antony Jameson
(Principal Adviser)

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Juan J. Alonso

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Sanjiva K. Lele

Approved for the University Committee on Graduate Studies.

Abstract

This thesis focuses on deriving algorithmic frameworks for the control of Aerodynamic Phenomena. The application of one such control law to the control of Flutter is discussed in detail. Flutter is an aero-structural instability that arises due to the adverse transfer of energy between the airplane structure and the surrounding fluid.

CFD is now a mature technology and can be used as a design tool in addition to being used as an analysis tool. This is the motivation for much of the research that takes place at the Aerospace Computing Lab at Stanford. Shape optimization involves finding the shape (2-d or 3-d) that optimizes a certain performance index. Clearly, any optimum shape will be optimum only at the design point. It has been found that the aerodynamic performance at neighboring operating points is a lot less optimal than the original shapes. What we need to do is to design and develop a feasible way of controlling the flow at any operating point such that the resulting performance is optimal.

In designing control laws, our philosophy has been to develop an algorithmic framework that enables treating a broad class of control problems rather than design control laws for specific isolated cases. This ensures that once a framework is established, extensions to particular problems can be done with very little effort. The framework we develop is problem independent and controller independent. Moreover, it has been shown that this leads to control laws that are feedback based, hence robust.

Acknowledgements

This thesis was never easy to write, and would have been impossible to finish if not for the people in my life. I take this moment to thank them all.

Firstly, I would like to thank Professor Jameson. He was a great research advisor. He gave me the freedom to work on the subject of my choice. And found the money to support my work. His warmth and genuineness will always be remembered. As also some of the conversations we've had: about research, and about life.

I thank Professors Lele and Alonso for patiently reading my thesis and providing valuable comments. And for all the research related advice, through the years. Professors Levy and MacCormack agreed to serve on my defense committee at extremely short notice. I thank them for finding the time, and for their valuable suggestions at the end of my defense. I would also like to thank Professor Kroo for discussions on aircraft flutter, and Professors West and Lall for advice on the design of nonlinear control systems.

Funding was partly by way of the Semino Fellowship awarded by the School of Engineering and the Department of Aeronautics and Astronautics and partly by way of NASA Langley and Air Force research grants. Their financial support is gratefully acknowledged.

I've had the good fortune of working with some very talented and driven people at the Aerospace Computing Laboratory. Work takes on a completely new dimension when you see motivated people burning the midnight oil all around you. That fact that there is company seems to add some meaning to the struggle. Nawee, Ki Hwan, Sahu and Anup: I will forever remember those midnight coding sessions at Durand 358. Those were good days!

I especially enjoyed the times that I worked with Sahu. It was one of the most inspired phases of my Ph. D. career. I think we bought out the best in each other. I thank him, also, for providing me with most of the infrastructure for a full fledged aero-elastic simulation and the technical expertise to add the missing pieces.

I would especially like to thank Nawee for patiently proof-reading my thesis, and helping me iron out the inconsistencies and Pat for working overtime to bring my code and data files back from the dead when bigarray, the scratch disk that I was working on, crashed just three weeks before my defense.

I would also like to thank my other lab-mates Balaji, Sriram, Seongim, Arathi, Jender, Georg, Andre, Kasidit and Edwin for providing a warm collegial atmosphere.

Lynn Kaiser, Carolyn Edwards, Diane Le, and Ralph Levine have been the go-to people for every administrative question that I've ever had. They've made it possible for me to get through school without ever having to read one of those long boring legal documents drafted by the administrative offices.

And then the gang: Vipin, Vishal, Sahu, Kaustuv, Allex, Shravan, RAP, Dev, Shardul, Srihari and Anshul. When I look back I think of all those lazy Saturday afternoon discussions (about world politics, war strategies, abstract art, literature, the human condition, and mindless local gossip), those late night trips to the city, hot cups of chai, and all those trips around the bay and beyond. Thanks for being around guys!

Finally, and most importantly, I would like to thank my Mother. For teaching me some of the most valuable lessons in life. For having faith in me. For constantly encouraging me to set my goals high. For pushing me that much harder to work towards my goals. For pulling me up all those times that I was down. For helping me realize that the end was in sight, always. For unconditional love, and support. I dedicate this thesis to her.

Contents

List of Figures

Chapter 1

Introduction

An airplane, by its very nature of design, is meant to be a flow control device. This becomes clear when one considers a steady air flow with and without the airplane. The very presence of the airplane alters the flow pattern, significantly so.

Through the ages, the primary goal of the aerodynamicist has been to design airplanes such that they meet certain performance criteria. This could be, for example, the maximum range of the airplane or the drag at cruise conditions. Lower drag immediately translates to lower fuel consumption and hence lower operating costs.

The aerodynamic performance of an airplane is determined by the nature of the surrounding flow field under given flight conditions. The most important motivation for flow control arises thus:

If it becomes possible to control the nature of the surrounding fluid flow, then it is conceivable that both the operating envelope and the aerodynamic performance of an airplane within that envelope can be significantly enhanced.

Exploring techniques for active flow control is the central theme of this thesis.

In this chapter, we trace the development of flow control concepts from single and multi point shape optimization techniques to morphing shapes and virtual aerodynamic shaping. We discuss the benefits and limitations of the current state of the art in flow control. We then highlight the need for an algorithmic approach to Active Flow Control.

1.1 Shape Optimization: A Case Study

The flow field surrounding an airplane is determined by its external shape. Thus, one way of achieving desired performance levels is by suitably designing the external shape of the airplane. This falls under the purview of Aerodynamic Shape Optimization. Aerodynamic Shape Optimization based on Control Theory has been an active research topic for the last twenty years.

The theory of Optimal Control of Systems governed by Elliptic Partial Differential Equations was first developed by Pironneau [64]. This was first applied to Aerodynamic Shape Optimization by Jameson. Jameson developed the *Adjoint* method for Aerodynamic Shape Optimization in a sequence of papers by himself [31, 36, 33] and with Reuther and other co-authors [40, 66, 67, 68, 69]. Giles and Pierce have demonstrated the equivalence of the *Adjoint* equations to Greene's functions [25, 24].

Many other research groups have developed CFD (Computational Fluid Dynamics) codes for aerodynamic optimization as well. Elliot [18, 19] and Anderson [5, 56] have developed codes based on the *Discrete Adjoint* approach that work on unstructured grids. Mohammadi [53, 52] has successfully used Automatic Differentiation software to create an *Adjoint* code from a CFD code.

In this section, we discuss the concept of flow control using shape optimization by studying a simple example problem.

Optimum Profile Shape Design for Supersonic Flow

One of the classic research problems in supersonic flight has been that of finding 2-d and axisymmetric profiles that have minimum pressure drag in supersonic flow. The 2-d sections are used as wing-profile sections, and the axisymmetric profiles are useful in that the distribution of Cross Sectional Area in a real airplane is made to follow the optimum distribution in order to minimize wave drag (The Area Rule). This problem becomes redundant without suitable constraints. We know that the minimum drag shape is a flat plate in 2-d flow and a needle-like profile in axisymmetric flow. But this is not the answer we are looking for. Hence to make the problem more meaningful, we ensure that the enclosed area/volume is constant. We also ensure that the ends are pointed. This is to anchor the shocks firmly to the leading and trailind edges.

Results from Classical Theory

Analytical solutions for the problem being studied have been obtained, assuming a linearized flow model. For the 2-d case the optimum profile is parabolic.

$$y(x) = 3Ax(1-x) \ , \quad \tau = \frac{3A}{2} \ , \tag{1.1}$$

where A is the area enclosed and τ is the thickness-chord ratio. The drag coefficient is given by

$$C_d = \frac{12A^2}{\sqrt{M^2-1}} \ . \tag{1.2}$$

For the axisymmetric case, the profile shapes that solve this problem are the well known Sears – Haack profiles, discovered independently by Sears (1947) and Haack (1947). The derivation of the Sears – Haack profiles is outlined in the book by Ashley and Landahl [6] and also in an article by Carlo Ferrari [21]. The Sears – Haack profile is given by

$$y(x) = \sqrt{\frac{16V}{3\pi^2}} \left[4x(1-x)\right]^{\frac{3}{4}} \ , \tau = \sqrt{\frac{64V}{3\pi^2}} \ , \tag{1.3}$$

where V is the enclosed volume and τ is the fineness ratio. The drag coefficient is given by

$$C_D = 24V \ . \tag{1.4}$$

As can be observed, these profile shapes have some interesting properties. Firstly, they are unique solutions to the optimization problem. Moreover, they are just a function of the enclosed area/volume and not the Mach Number.

Nonlinear Optimization via Control Theory

The aerodynamic shape optimization problem involves minimizing (or maximizing) a given cost function, with parameters that define the shape of the body as the design variables, usually of the form

$$I = \int\limits_{\mathcal{B}_\xi} \mathcal{M}(w,S) \, d\mathcal{B}_\xi \ , \tag{1.5}$$

where w is the vector of flow state variables and S_{ij} are the coefficients of the Jacobian matrix of the transformation from physical space to computational space. $\mathcal{M}(w,S)$ in our

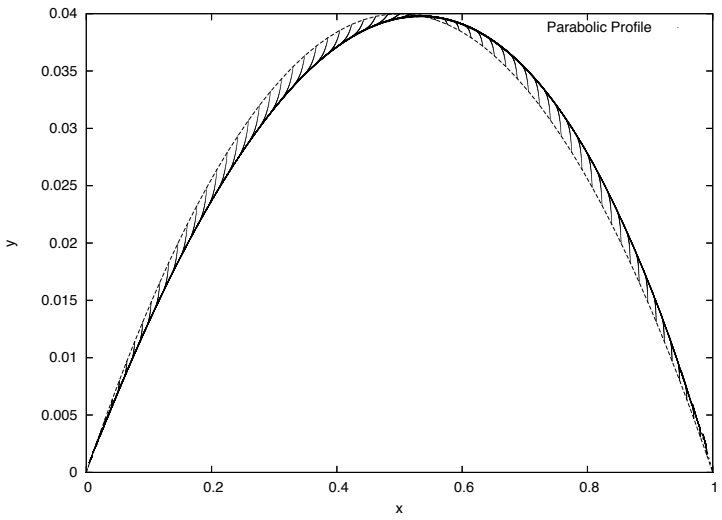

Figure 1.1: Convergence of the optimization algorithm from a parabolic initial profile

case is just C_p, the pressure coefficient. We also have the constraint that the state variables at the computational points have to satisfy the flow equations, irrespective of the shape of the boundary.

Convergence from Different Initial Conditions: The main test of the correctness of the optimization algorithm is to see if it converges to the same optimum profile regardless of what the initial profile is. Figures 1.1 and 1.2 show the optimization history from two different initial profile shapes for 2-d flow. They both enclose the same area. It can be seen that they converge to the same optimum profile. This gives us confidence in the correctness of our optimization setup.

Optimum Profile Shapes: The results of the 2-d optimization can be seen in Figure 1.3 and the results of the axisymmetric optimization can be seen in Figure 1.4. As can be observed, the nonlinear optimum profiles are slightly different from the classical optimum profiles. They have a more rearward point of maximum thickness. The primary difference between a linearized flow model and a nonlinear model is the appearance of shocks at the leading edge in the case of the nonlinear flow model. Reducing the included angle at

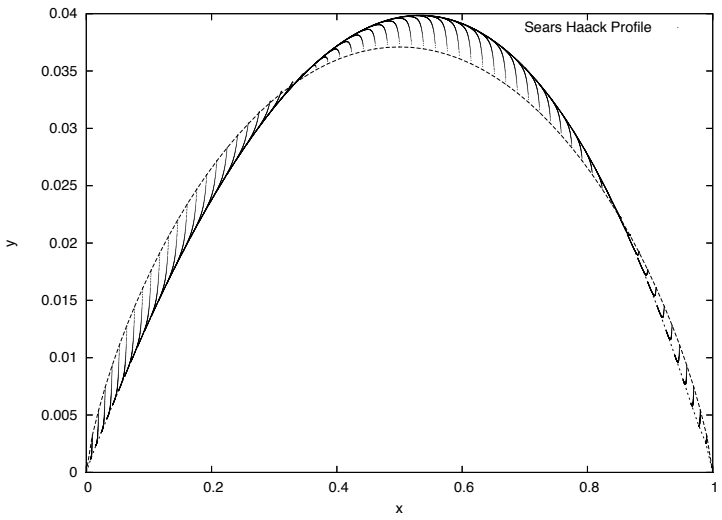

Figure 1.2: Convergence of the optimization algorithm from a Sears – Haack Initial Profile

the leading edge and moving the point of maximum thickness rearward is consistent with reducing the magnitude of the leading edge shock. This results in a lower drag and at the same time brings the flow closer to the linear regime.

Variation with Mach Number: The optimum profile for 2-d flow changes with Mach number. The optimum shape for two Mach numbers is shown in Figure 1.5. It is seen that the point of maximum thickness is more rearward for the higher Mach number. This again is consistent with our earlier argument that the main goal of the nonlinear optimization is to reduce the magnitude of the leading edge shock.

Discussion of Results: The above results highlight the following points:

1. Shape Optimization using Control Theory does, indeed, find the optimum aerodynamic shape for a certain performance criteria. We see this by noting that the results of aerodynamic shape optimization are in the neighborhood of provably optimal solutions based on linear theory.

2. In spite of the fact that we started with different initial profiles, the Shape Optimization Algorithm converged to the same optimal solution. This shows that *Adjoint*

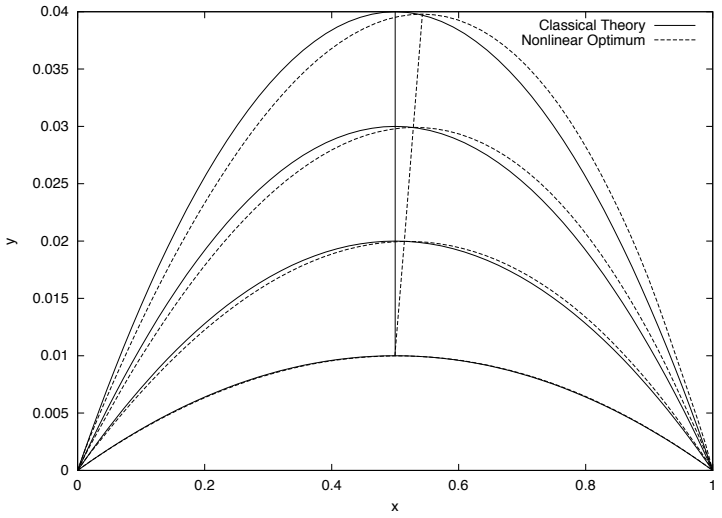

Figure 1.3: Classical and Nonlinear Optimum Profiles for 2-D flow

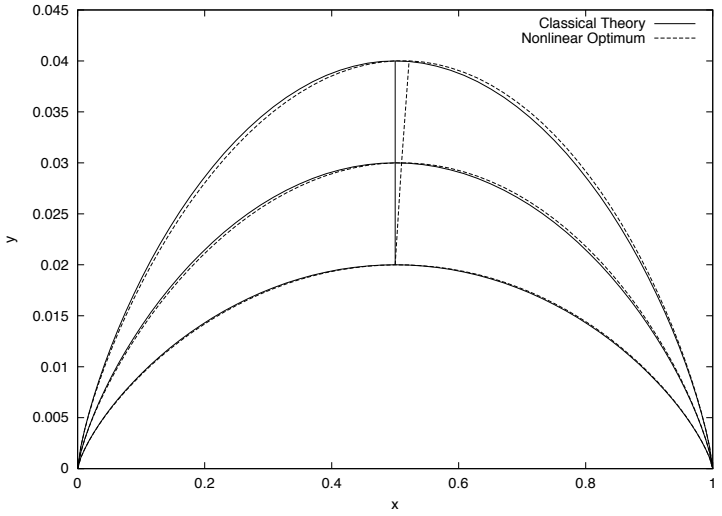

Figure 1.4: Classical and Nonlinear Optimum Profiles for Axisymmetric flow

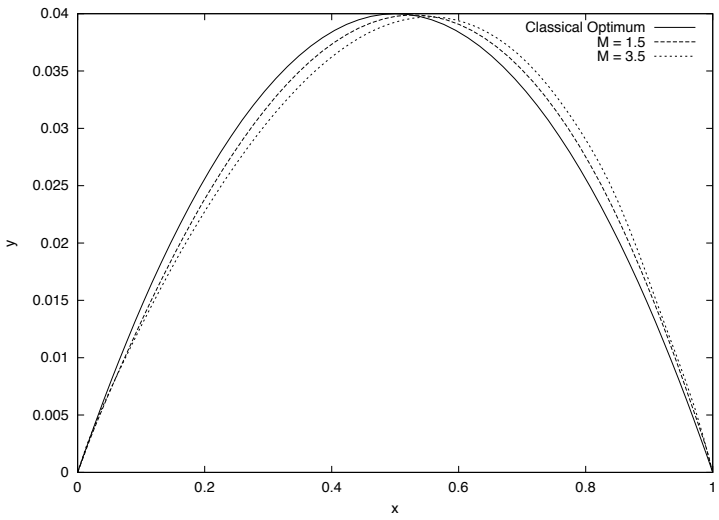

Figure 1.5: Variation of 2-d Optimum Profiles with Mach Number

based Shape Optimization Algorithms do converge to local optima.

3. *The Optimum Shape for each Mach number is different.* This is the most important result as far as this thesis is concerned. Clearly, in order to remain optimal over a range of operating conditions, we need the capability to change the shape to correspond to the optimal shape at that condition.

1.2 Morphing Aircraft

As can be seen from the results of the previous section, the aerodynamic shape that is optimum for a certain performance criterion varies as the operating conditions are changed. While multi-point design is possible, it would involve a sacrifice in performance from the optimum possible at each individual operating point.

What we would prefer, ideally, is to morph the shape throughout the operating envelope of the airplane. The idea of morphing shapes has been implemented in rudimentary fashion through the ages. The deployment of slats and flaps to increase lift during landing and the use of swing wings to achieve optimum performance through a range of Mach numbers are some examples.

Mani, Lagoudas and Rediniotis [51] have done extensive research in the use of active skin actuation for the control of turbulent drag. They hypothesize that turbulent skin friction drag can be minimized by the passage of surface waves of suitable frequencies and fairly small magnitudes. The skin is then controlled actively to affect this reduction. The feasibility of using Shape Memory Alloys and piezo electric actuators is discussed.

Baker and Friswell have studied issues regarding the implementation of shape memory alloy based wing design [7].

Shape Memory Alloys are materials that remember their original shape. They have the tendency to return to this shape after being realigned, when induced by a temperature or a magnetic field. They thus find potential use in any field in which actuators that change their material properties based on external physical conditions are required [71].

While the optimum shape for any operating condition can be calculated, designing a control surface that deforms accordingly is a technological challenge. This involves extensive research to develop new materials that are both compliant enough to change shape, and yet stiff enough to withstand the severe aerodynamic loads through the operating envelope. Developing suitable actuation methodology is also an active area of research.

1.3 Flow Control

Another concept that is fast gaining popularity in Fluid Mechanics circles is that of Active Flow Control. Indeed, it is important to realize that adding or removing fluid at the wing surface is equivalent to effecting a shape modification. Flow control using surface jets should, in principle, have an effect very similar to that of morphing surfaces.

The *controllability* of the *Navier-Stokes* equations has been discussed by Bewley [9, 10]. The optimal flow control problem is formulated as an infinite dimensional optimization problem, where a cost functional is minimized. This cost functional describes the features of the flow being studied. By showing that the cost functional is strictly convex, he establishes the existence and uniqueness of an optimal control.

The capability to directly alter the flow field offers a huge realm of possibilities. Seifert, Theofilis and Joslin [74] categorize the problems that are amenable to using Active Flow Control:

1. Separation (Delay, Reattachment, Stabilization, etc.)
2. Transition (Delay, Promotion)

3. Jet (Spreading, Vectoring, Acoustics)
4. Drag Reduction (Laminar Skin Friction, Turbulent separation control)
5. Thermal Management (Cooling, heating, reduced signature)
6. Guidance, Propulsion and Control (Mild hinge-less maneuvering, gust alleviation)
7. Vortex Dominated Flows
8. Combustion, Turbo machines (Inlets, rotors, stators and diffusors)
9. Cavity (noise, vibration)
10. Optical Distortion

While different types of actuators can be designed for active flow control, Zero Net Mass Flux (ZNMF) Synthetic Jets are gaining popularity as the actuator of choice. A ZNMF synthetic jet is popular for the main reason that it is formed entirely from the working medium of the flow. These eject and remove mass from the flow system through a narrow orifice periodically. This results in altering the momentum field around the orifice without adding or removing mass from the flow. The primary considerations in the design of synthetic jets are the size and positioning of the orifice, and the time frequency of actuation. The design of synthetic jets and the physics of their interaction with a cross-flow are discussed in detail by Glezer and Amitay [26].

Flow control, for aerodynamics, using synthetic jets has been studied experimentally by Amitay [4], Tuck and Soria [78], and Nishizawa et al [1, 57]. Numerical investigations were performed by Nae [55]. It should be noted that in all these experiments, the location and frequencies of the actuators were chosen apriori. The control implemented, therefore, is open loop.

The study of closed loop active flow control techniques is still in its primitive stages. This is because designing a closed loop (*feedback*) control law requires understanding of the system dynamics. In spite of the fact that it is possible to obtain numerical solutions to the *Navier-Stokes* equations, understanding of the behavior of a flow-actuator system is extremely limited.

Feedback laws based on Reduced Order Models have been derived by Samimy *et al.* [73], Kumar and Tewari [44] and Cohen *et al.* [16]. The major drawback of these efforts is

that the actuator dynamics are not modeled as part of the reduced order description of the system.

1.4 Flutter Control

An airplane, by its nature of being, is constructed so that it is as light as possible. One of the most expensive side-effects of light construction is the increased flexibility that comes associated with it.

When a flexible structure interacts with an unsteady flow, there is transfer of energy from the fluid medium to the structure and vice-versa. If this transfer occurs at a frequency close to the flutter frequency, the aero-structural vibrations may rapidly lead to divergence and consequently failure. This phenomenon is called flutter.

In order to maximize the performance of the airplane, it has to be constructed so that it is light. Thus flutter cannot be avoided. One way of pushing the operating envelope of the airplane is to delay the onset of flutter. This is done by structural redistribution. Techniques for flutter control using acoustic waves [49] and micro-trailing edge flaps [11] have been proposed. It should be noted that these techniques are only capable of extending the flight envelope and not eliminating flutter altogether.

1.5 Contributions

All previous attempts at flow control have either involved designing simplistic controls for complex problems or complex *feedback* based controls for simple problems. Problems like separation control, drag reduction and control of the vortex shedding frequency in the flow past a cylinder have all been controlled using open loop controllers.

Closed loop control has been demonstrated only on simplistic models derived from simulation or experiment.

An *Ideal Flow Control Law* should have the following properties:

1. Broadly applicable: we are looking for an algorithmic framework for generating flow control laws for a variety of problems. The development of such a framework would enable easy analysis and design of control laws for a variety of flow control problems.

2. Scientific: the control laws should be based on a realistic model of the fluid system.

3. Robust: should account for variability in measurement, actuation, etc. This would mean that the control u should be *feedback* based

$$u = F(x), \tag{1.6}$$

where x is the current system state.

Our goal, therefore, is to develop *feedback* based control laws that are derived from a realistic representation of the flow. We try to make sure that the framework is as generic as possible, lending easy extension to a variety of situations. We then discuss specific applications of the control law thus derived, including control of Flutter.

The author collaborated with Sahu [72] to develop an integrated hi-fidelity aero-structural simulation based on uflo87 (A 3-d unsteady *Euler* flow code developed by Professor Jameson) and FEAP (Finite Element Analysis Package: A nonlinear structural dynamics solver developed by Professor Taylor), for test purposes.

Inviscid Assumption: The primary emphasis of this thesis is to develop techniques for generating control laws for realistic aerodynamic/aero-structural problems. A control law thus derived is used to demonstrate flutter control.

Our aim is to control flutter at its onset, when the angles of deflection are less than 1°. It can be seen from Figure 2.2, that for the amplitude considered, the unsteady aerodynamics associated with an airfoil pitching in transonic flow is well represented by an inviscid model. The surface jets used for control cause an effective shape change that changes the aerodynamics of the system. This, again, is well represented by an inviscid model.

A *Navier-Stokes* simulation is considerably more involved than an *Euler* simulation. Numerical stability issues have to be addressed in portions of the simulation where the surface jets interact with the surrounding flow. Moreover, the turnaround time taken for *Navier-Stokes* simulations is much higher than that of *Euler* simulations for the problems considered, for very little gain in the representation of the Flow Physics.

Thus an *Euler* model is used in this thesis.

It should, however, be noted that the techniques for developing control laws, as discussed in this thesis, are fairly generic and extensions to the viscous case should be straightforward.

1.6 Thesis Outline

Chapter 2 and Appendix A discuss the flow and structural models used in this thesis and the key computational aspects involved in their simulation.

The Flow Control Algorithm is developed in Chapter 3 and Appendix B. This is then specialized for the case of *Virtual Aerodynamic Shaping*. The *feedback* nature of the control laws thus developed is highlighted.

Chapter 4 discusses the modification of the aero-structural behavior of a system by passive structural optimization.

In Chapter 5, we develop an active control law for flutter, and demonstrate its effectiveness by simulation.

Chapter 2

Flutter: Modeling and Computational Simulation

The primary purpose of this thesis is to demonstrate the feasibility of implementing a *feedback* based control approach for practical aerodynamic problems. The approach developed is first tested on an inverse design problem, and then used to show that flutter can be controlled. Flutter is an aero-elastic phenomenon in which there is adverse transfer of energy from the surrounding flow-field to the structure of the airplane, and vice versa. This leads to rapid divergence of the structure, and consequently failure.

The first step towards controlling flutter, is accurate prediction. To do this, we need to develop a mathematical model that will track the physics accurately and reliably. When forming our mathematical models, we should take into consideration the fact that over the last few decades, people have spent considerable amounts of time developing very sophisticated computational solvers for the separate problems of unsteady aerodynamics [28, 30, 32, 34, 35, 46, 58, 70] and structural dynamics [79] respectively.

The most practical approach to modeling aero-structural interactions, therefore, would involve formulating the problem as an interaction between an aerodynamic component that tracks the unsteady aerodynamics and a structural component that tracks the structural deformations. This will allow us to use previously developed flow and structural solvers with very little modification. Figure 2.1 shows a schematic representation of a coupled aero-structural simulation.

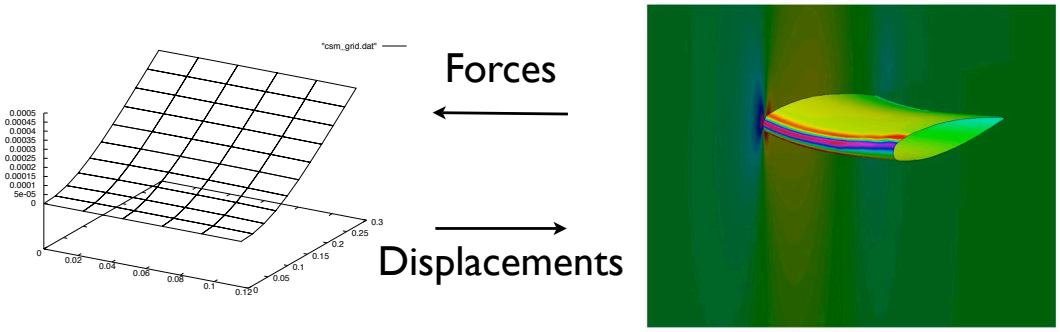

Figure 2.1: Schematic Diagram showing the integration of aerodynamic (CFD, right) and structural (FEM, left) solvers

Objectives: The following sections discuss the flow and structural models used in this thesis, and describe some of the key computational aspects of their simulation, including techniques for aero-structural interfacing and mesh warping.

2.1 Fluid Mechanics

Choosing a fluid model for the flutter control problem involves consideration of the physics of the problem and the question of reasonably fast computation.

An accurate mathematical representation would need to effectively address the issues of unsteadiness and compressibility. Ideally, we would also like to consider viscous effects. But this would require significantly more computational resources in terms of time and computing power. Figure 2.2 presents a comparison of C_l time histories obtained using three different means for the case of a pitching NACA 64A010 airfoil. The simulations were performed by Alonso [2]. The parameters of the experiment were as follows: $M_\infty = 0.796$, $k_\epsilon = 0.202$, $\Delta\alpha = 1.01°$, $Re = 12.56 \times 10^6$ and $T_\infty = 300K$. This experiment clearly shows that for the angle of attack variations and reduced frequencies considered, an inviscid model does not sacrifice much in terms of accuracy.

Our primary emphasis is to control flutter at its onset, before the divergence angles become very high. Moreover, at small angles of attack, almost all the nonlinearity associated with flutter that arises from the unsteady variation of the pressure distribution along the surface of the airfoil. It has been successfully demonstrated over the past few decades by

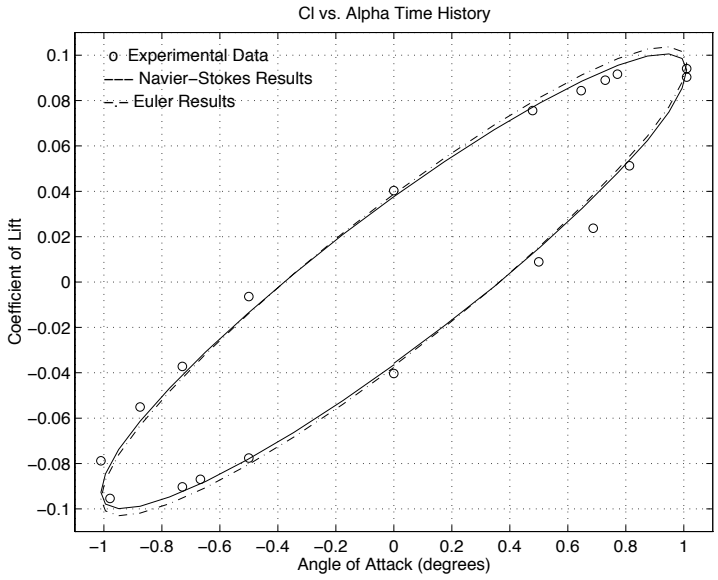

Figure 2.2: Comparison of C_l time histories for the case of a pitching airfoil. From Alonso [2]

Jameson and his associates [30] that the pressure distribution computed using an inviscid model matches very well with experimental results.

Thus, in the interest of saving computational overhead, almost all the flow situations in this thesis are modeled using the *Euler* equations,

$$\frac{\partial \mathbf{w}}{\partial t} + \frac{\partial \mathbf{f_j}}{\partial x_j} = 0. \tag{2.1}$$

Here,

$$\mathbf{w} = \begin{bmatrix} \rho \\ \rho u_1 \\ \rho u_2 \\ \rho u_3 \\ \rho E \end{bmatrix}, \quad \mathbf{f_j} = \begin{bmatrix} \rho u_j \\ \rho u_1 u_j + P\delta_{1j} \\ \rho u_2 u_j + P\delta_{2j} \\ \rho u_3 u_j + P\delta_{3j} \\ \rho H u_j \end{bmatrix}. \tag{2.2}$$

The *Euler* equations represent the conservation of mass, momentum and energy for fluid flows where viscous effects are not predominant. Here, ρ is the density, u_i are the flow velocities, E is the specific internal energy, H the specific enthalpy, and P is the pressure

at any point in the flow.

The pressure is related to the density and energy by the following state equation

$$P = (\gamma - 1)\,\rho\left(E - \frac{1}{2}u_j^2\right). \tag{2.3}$$

Here γ is the ratio of specific heats. Another equation important for closure is the *Ideal Gas Equation of State*,

$$P = \rho RT. \tag{2.4}$$

The specific internal energy E is related to the Temperature T as follows

$$E = C_v T + \frac{1}{2}u_j^2. \tag{2.5}$$

Here C_v is the specific heat of the fluid at constant volume. The specific enthalpy H is related to the specific internal energy E as follows

$$H = E + \frac{P}{\rho}. \tag{2.6}$$

2.2 Computational Simulation

The *Euler* equations are highly nonlinear partial differential equations. Hence, closed form solutions cannot be obtained in general. We, therefore, resort to computational simulations to obtain the desired solutions.

All calculations in this thesis are performed on structured grids. 2-d calculations are performed on C or O meshes and 3-d calculations are performed on C – H meshes. A typical C mesh used for 2-d calculations is shown in Figure 2.3. The equations are discretized using cell-centered Finite Volume schemes. The spatial derivatives are calculated using central differences, with blended second and fourth order dissipative fluxes. The second order dissipative fluxes serve to damp out unfavorable oscillations that develop in the vicinity of contact discontinuities like shocks. The fourth order dissipative fluxes, when added in a controlled manner serve to increase the order of accuracy in smooth regions of the flow. This scheme belongs to the family of JST schemes for compressible gas dynamics developed by Jameson, Schmidt and Turkel [41]. The *Euler* equations can then be written as the

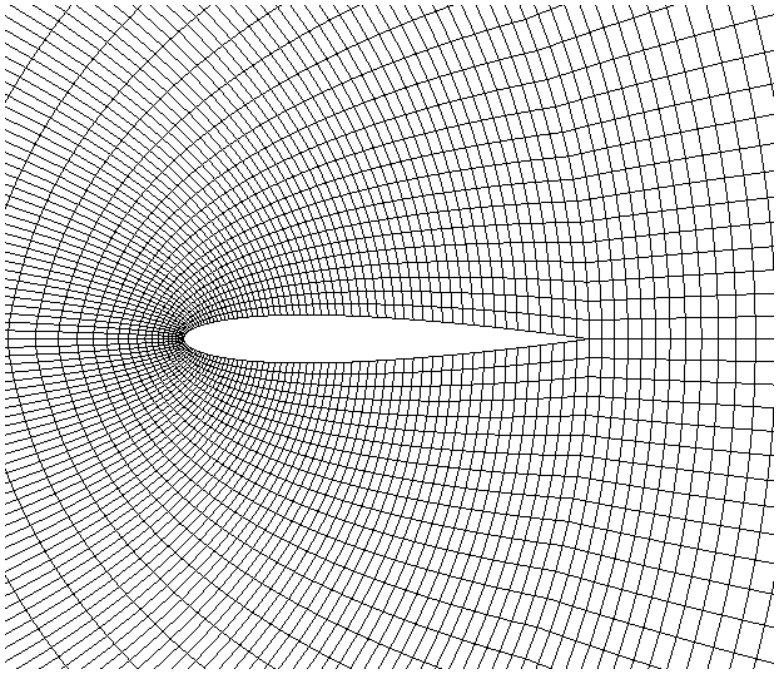

Figure 2.3: A typical C mesh used for a 2-d flow calculation

following set of coupled Ordinary Differential Equations

$$\frac{d\mathbf{w}}{dt} + R(\mathbf{w}) = 0. \tag{2.7}$$

Time integration is done using a *dual* time stepping approach. First, time accurate integration is done using a second order backward difference formula in time. This necessitates the solution of an implicit difference equation of the form

$$\frac{3}{2\Delta t}(\mathbf{w}^{(n+1)}V^{(n+1)}) - \frac{2}{\Delta t}(\mathbf{w}^{(n)}V^{(n)}) + \frac{1}{2\Delta t}(\mathbf{w}^{(n-1)}V^{(n-1)}) + R(\mathbf{w}^{(n+1)}) = 0. \tag{2.8}$$

at each *real* time step. The solution to this equation is obtained as the steady state solution to a problem of the form

$$\frac{\partial \mathbf{w}}{\partial t^*} + R^*(\mathbf{w}) = 0, \tag{2.9}$$

where $R^*(\mathbf{w})$ is a modified residual with source terms. This equation is then solved in *pseudo* time using techniques from the solution of a steady aerodynamic flow.

The convergence to steady state of the *pseudo* time problem is accelerated many fold, using convergence acceleration techniques like multigrid, implicit residual averaging, variable local time stepping and enthalpy damping.

The spatial and time discretization procedures, and the convergence acceleration techniques are discussed in great detail in Appendix A.

2.3 Actuator Modeling

The actuator used in this thesis is a steady jet that is assumed to add a velocity component, that is in a direction normal to the wall, to the fluid near the wall surface. A computational

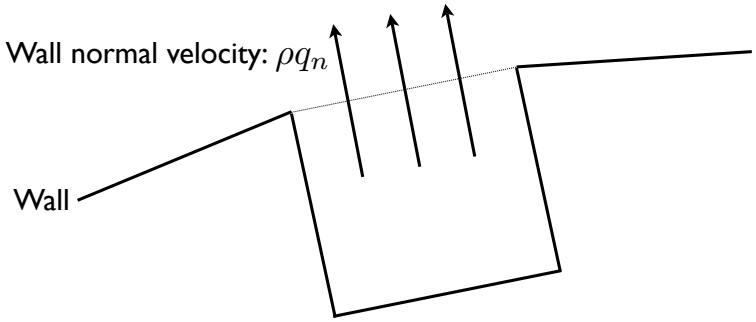

Figure 2.4: Implementation of the Actuator Boundary Condition at the wall

cell near the wall, with the jet coming out is show in Figure 2.4. The flow velocities in the *halo* cell within the wall are set so that the resultant normal velocity computed at the wall is numerically equal to ρq_n. Thus, if $\tan\theta$ is the slope of the tangent at the wall,

$$(\rho_1 v_1 \cos\theta - \rho_1 u_1 \sin\theta) = 2\rho q_n - (\rho_2 v_2 \cos\theta - \rho_2 u_2 \sin\theta)\ . \tag{2.10}$$

Here, the subscript 2 is used to denote the flow values at the cell just outside the wall, and the subscript 1 is used to denote the flow values at the *halo* cell. We also set

$$\rho_1 = \rho_2, \tag{2.11}$$

$$\frac{\partial P}{\partial n}_1 = \frac{\partial P}{\partial n}_2, \tag{2.12}$$

$$\rho_1 E_1 + P_1 = \rho_2 E_2 + P_2. \tag{2.13}$$

The required value of ρq_n is derived from the control algorithm.

2.4 Structural Mechanics

The structural dynamic model is derived from the theory of elasticity, which relates the deformation and internal stresses of the structure to the external loads applied. A Lagrangian frame is used to describe the structure, as contiguous elements of the structure continue to remain contiguous unless structural failure occurs. The state of structure at any point in

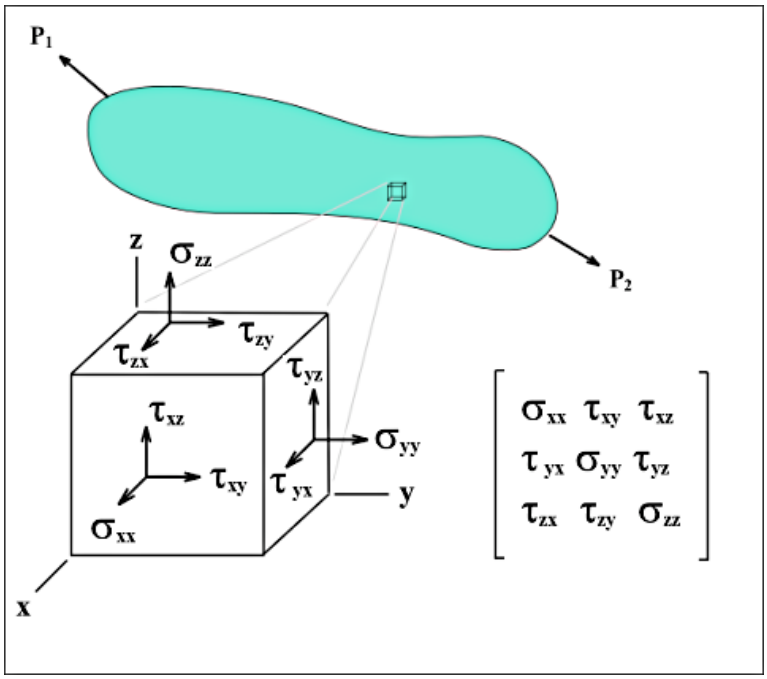

Figure 2.5: Schematic Diagram showing the components of the stress tensor.

time, is represented at each spatial point by 15 state variables. There are three displacements **u**, which represent the deviation of the point from its baseline position. Additionally, the internal state of the structure is represented by the stress and strain tensors, σ_{ij} and ϵ_{ij} respectively, both of which are symmetric and hence have six independent components each.

The normal stress is defined as the force acting in a given direction, per unit area normal to that direction. Shear stress is defined as the force acting tangential to a surface, per unit area. Figure 2.5 shows the components of the stress tensor. Strains are the ratio of the displacement of a point in a particular direction to the original dimension of the object at that point.

There are six equations which relate the strains to the displacements. These are linear for the case of small deformations.

$$\epsilon_{ii} = \frac{\partial u_i}{\partial x_i}, \qquad \epsilon_{ij} = \epsilon_{ji} = \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \ . \tag{2.14}$$

The stresses are related to the strains by six constitutive relationships. If the material is isotropic, these reduce to

$$\sigma_{ii} = \frac{E}{(1+\nu)(1-2\nu)} \left[(1-\nu)\epsilon_{ii} + \nu(\epsilon_{jj} + \epsilon_{kk})\right], \qquad \sigma_{ij} = \sigma_{ji} = \frac{E}{2(1+\nu)}\epsilon_{ij} \ , \tag{2.15}$$

where E is the Young's modulus of the material, and ν is the Poisson's ratio. The external forces can be related to the internal stresses and strains using the Newton's laws.

$$\frac{\partial \sigma_{ij}}{\partial x_j} + \rho \frac{\partial^2 u_i}{\partial t^2} + \kappa \frac{\partial u}{\partial t} = F_i \ . \tag{2.16}$$

Here ρ is the density of the material, κ is the damping factor, and $\mathbf{F}$ is the applied external force. Solving these fifteen equations gives us the stress – strain – displacement distribution throughout the structure.

2.5 The Finite Element Method

The structural dynamics is also simulated computationally. This is done using a Finite Element Method. The finite element method provides one way to discretize the continuous partial differential equations that govern the dynamics of the structure, into a set of coupled discrete ordinary differential equations which can be written in the form

$$[M]\{\ddot{\mathbf{q}}\} + [C]\{\dot{\mathbf{q}}\} + [K]\{\mathbf{q}\} = \{\mathbf{F}\} \ , \tag{2.17}$$

where $[M]$, $[C]$, and $[K]$ are the mass, damping and stiffness matrices respectively. All these matrices are of size $n \times n$, where n is the number of degrees of freedom, as determined by the discretization. $\mathbf{q}$ is a vector of displacements, also of size n. These could be linear displacements or angular rotations, depending on the nature of the discretization. $\mathbf{F}$ is a vector containing applied external forces and moments corresponding to the respective displacements.

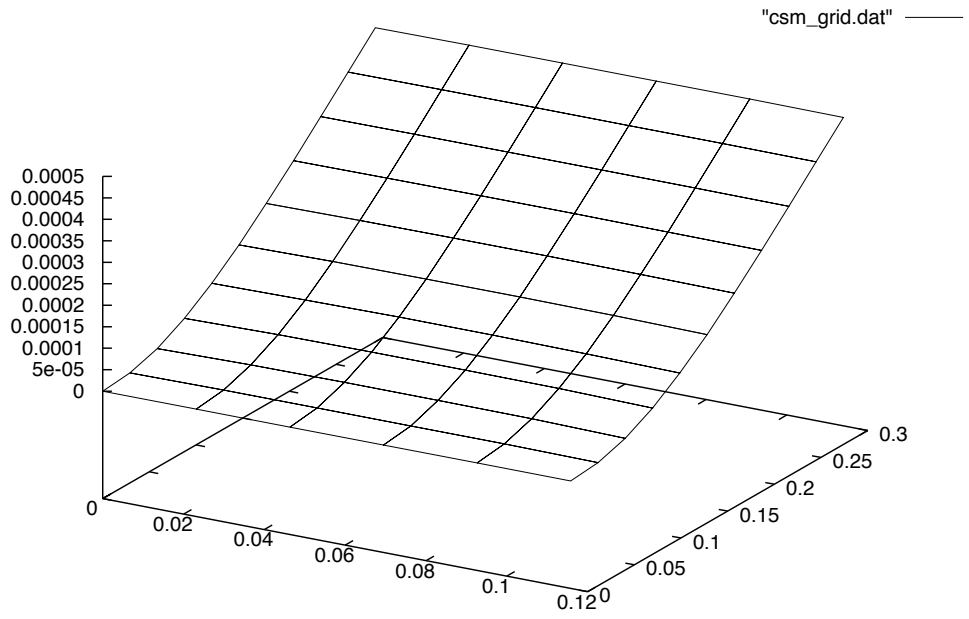

Figure 2.6: Structural Discretization

In this thesis, we study the aero-structural behavior of a wing in Chapter 5. The semi-span of the wing is $0.294m$, and the chord of the wing is $0.1158m$. The wing is represented structurally as an aluminum plate that has the same dimensions as the wing, and is lined along the the symmetry axis. The plate is has the same dimensions as the wing and a thickness of $0.00165m$.

The plate is discretized into 50 plate elements as shown in Figure 2.6. Each plate element has 4 nodes, leading to a total of 66 nodes in the discretization. Each node has 3 translational and 3 rotational degrees of freedom. The structural simulation, therefore, has a total of 396 degrees of freedom.

2.5.1 The Newmark Scheme

Equation (2.17) is solved using the *Newmark* scheme. The *Newmark* scheme is used to solve a second order transient problem of the form

$$R(t) = f(t) - P(q(t), \dot{q}(t), \ddot{q}(t)) = 0 \ , \tag{2.18}$$

where in our case

$$P = [M]\{\ddot{\mathbf{q}}\} + [C]\{\dot{\mathbf{q}}\} + [K]\{\mathbf{q}\} \ . \tag{2.19}$$

The solution is updated as follows

$$q_{n+1} = q_n + \Delta t v_n + \Delta t^2 \left[(0.5 - \beta)a_n + \beta a_{n+1}\right] \ , \tag{2.20}$$

and

$$v_{n+1} = v_n + \Delta t \left[(1 - \gamma)a_n + \gamma a_{n+1} \ ,\right] \tag{2.21}$$

where

$$v_n = \dot{q} \ , \qquad a_n = \ddot{q} \ . \tag{2.22}$$

In the above equations β and γ are parameters controlling the stability and numerical dissipation respectively. Typical values chosen for β and γ are 0.25 and 0.5 respectively. In this thesis, the structural analysis is performed using the *Finite Element Analysis Program* (FEAP), written by Taylor [79] at the University of California, Berkeley.

FEAP is an extremely versatile structural analysis program that allows for the construction of arbitrarily complex finite-element models using a library of 1-, 2- and 3- dimensional

elements for linear and nonlinear deformations. The material under consideration could be isotropic, orthotropic or even plastic. A number of solution procedures are available for linear, nonlinear and time-accurate problems. In addition to this, interfaces are available which enable the use of externally developed solvers. This comes in especially handy in the solution of large, nonlinear structural problems as extremely efficient external solvers that are written especially to solve sparse problems on parallel architectures can be used. This speeds up the solution process significantly. A number of time-accurate integration algorithms are also included with FEAP, which are of particular interest in the solution of aeroelastic problems.

2.6 Aero-Structural Integration

In the simulation of aero-structural systems, the interfacing of the aerodynamic and structural components is very critical. The integration involves transferring the loads from the fluid to the structural solver, and the displacements from the structural to the fluid solver in a consistent and conservative manner. The sequence in which these transfers are performed is of vital importance to the overall stability and efficiency of the procedure. The method used in this thesis is based on the work of Brown [14].

Firstly it is important to recognize that the CFD (Computational Fluid Dynamics) mesh need not be aligned with the CSM (Computational Structural Mechanics) mesh. Thus, it becomes important to first create a one-to-one association between points on the CFD mesh and points on the CSM mesh. This is done as part of a pre-processing step.

2.6.1 Pre-processing

The association is performed by locating the point on the CSM mesh that is closest to each CFD point, as shown in Figure 2.7. The link between these two points is then on assumed to be rigid.

2.6.2 Displacement Transfer

During an aero-structural simulation, the nodal values of the displacements and rotations are first transferred to the associated point on the CSM grid using the shape functions used in the finite element Model. The displacement of the CFD grid point can then be calculated

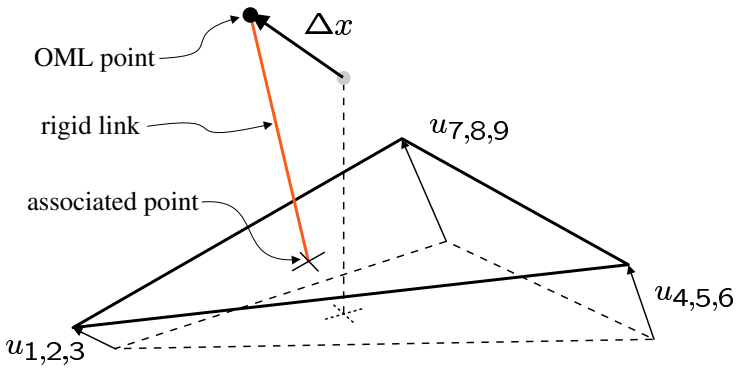

Figure 2.7: Extrapolation of displacements from the CSM mesh to the CFD mesh

as a sum of the displacement of the associated CSM point and a rotation term which is obtained as a cross product of the angle with the length vector of the link.

$$\Delta \mathbf{x} = \Delta \mathbf{x_b} + \Delta \theta_\mathbf{b} \times L \; . \tag{2.23}$$

The displacement and rotation at the associated point on the CSM mesh can be written in terms of the nodal displacements, as discussed earlier. The expression for the displacement of the CFD grid point then becomes

$$\Delta \mathbf{x} = W_t \mathbf{u} + W_r \mathbf{u} \tag{2.24}$$

$$\Delta \mathbf{x} = W \mathbf{u}, \tag{2.25}$$

where W_t and W_r are pre-calculated weighting matrices corresponding to the translational and rotational components of the translation, and $\mathbf{u}$ is the vector of nodal displacements.

2.6.3 Load Transfer

The next critical procedure in the aero-structural integration process is the transfer of loads from the CFD grid to the CSM grid. The pressures at the surface of the CFD mesh are first integrated to obtain the values of the equivalent forces and moments at the CFD nodes.

Usually there are many more CFD points than there are CSM points. Thus, in theory, the load transfer from the CFD mesh to the CSM mesh can be done in an infinite number

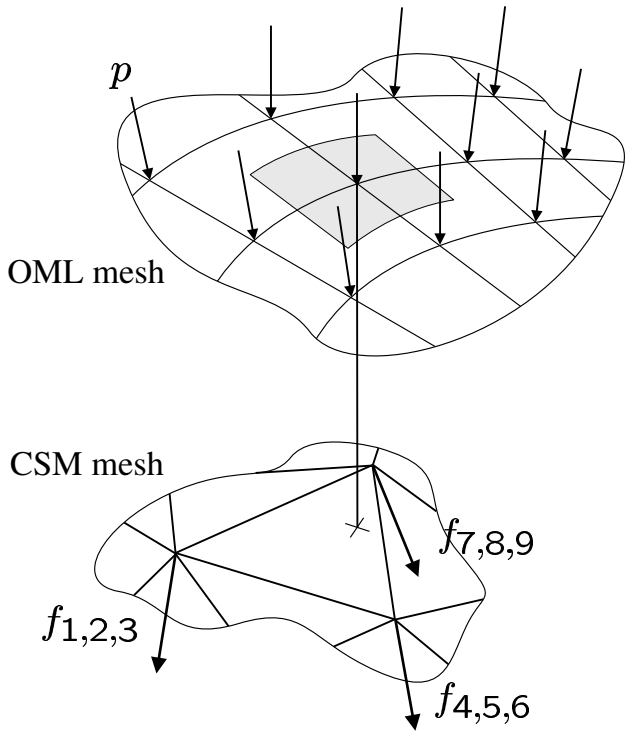

Figure 2.8: Extrapolation of loads from the CFD mesh to the CSM mesh

of ways. All these, however, will not be physically correct. We seek a solution that is, physically speaking, *consistent* and *conservative*. *Consistency* implies that the sum of the forces and moments at the nodes of the CFD mesh be the same as the sum of the forces and moments respectively at the nodes of the CSM mesh. By *conservative* we mean that energy is conserved, and that the transfer doesn't introduce artificial energy sources into the system. The virtual work done by the nodal forces in the CFD mesh $\mathbf{f}_{CFD}$ when the CFD nodes are subjected to a virtual displacement $\delta\mathbf{x}$ is given by

$$\delta w_{CFD} = \mathbf{f}_{CFD}^T \, \delta\mathbf{x}. \tag{2.26}$$

Similarly the virtual work done by the nodal forces at the CSM mesh $\mathbf{f}_{CSM}$ when the CSM mesh is subjected to an equivalent virtual displacement $\delta\mathbf{u}$ is

$$\delta w_{CSM} = \mathbf{f}_{CSM}^T \, \delta\mathbf{u}. \tag{2.27}$$

In order to be *conservative*, the force transfer should occur in a manner that virtual work is conserved, or

$$\mathbf{f}_{CFD}^T \, \delta\mathbf{x} = \mathbf{f}_{CSM}^T \, \delta\mathbf{u}. \tag{2.28}$$

Equation (2.25) relates $\delta\mathbf{x}$ to $\delta\mathbf{u}$. Equation (2.28) can thus be written as

$$\mathbf{f}_{CSM} = W^T \mathbf{f}_{CSM}, \tag{2.29}$$

where W is the same matrix that relates $\delta\mathbf{x}$ to $\delta\mathbf{u}$.

2.6.4 Integration of Aerodynamic and Structural Solvers

The aerodynamic and structural solvers are coupled by exchanging information at regular intervals during the solution process. A schematic diagram outlining the coupling process is presented in Figure 2.9.

Each cycle of the aero-structural iteration starts with the transfer of the current displacement state of the structure from the CSM mesh to the CFD mesh. A CFD calculation is then performed, and new values of the nodal forces and moments are calculated. These loads are then transferred to the CSM mesh. The structural dynamics calculation is then advanced through the same time difference as the CFD mesh.

This cycle is repeated through the time period of interest. Usually, the stability limits impose more stringent limitations on the maximum allowable time step for the fluid solver than the structural solver. This might result in many more CFD calculations being performed in the same time it takes to perform one Finite Element calculation. This technique is referred to as the *sub-cycling* methodology.

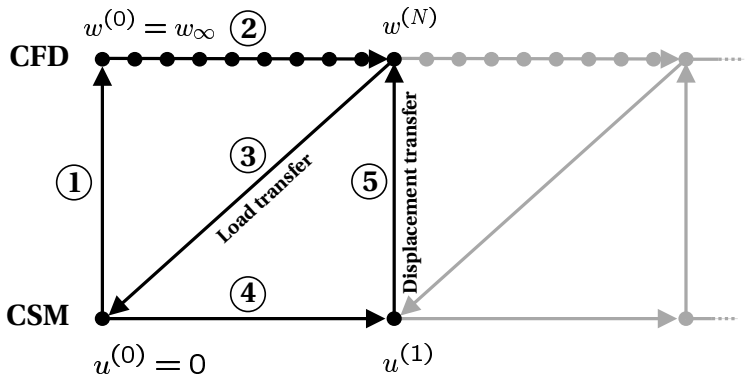

Figure 2.9: Aero-structural Integration

2.7 Mesh Deformation

The displacements that are transferred from the CSM mesh to the CFD mesh, affect only the surface of the CFD mesh. An effective means of propagating these disturbances throughout the grid is required. Mesh deformation is an art in its own right, especially given the complex nature of the meshes used in CFD recently. Some of the problems frequently faced include bad scaling of the resultant new mesh, or worse, inversion of some of the interior cells. These are extremely hard to locate. However, the ill effects of bad scaling can be significantly minimized if sufficient care is taken during the mesh deformation process.

In this thesis, the mesh is assumed to be a solid body satisfying the laws of elasticity as discussed in Section 2.4. A deformation applied to the surface can then be considered as a displacement applied at the surface nodes of the body. The equations of elasticity are then solved, and a deformed grid is obtained. The mesh deformation algorithm used in this thesis was developed by Premasuthan and Jameson [65]. Figure 2.10 shows the deformed mesh, after an airfoil, initially at 0° angle of attack, is subject to a counter-clockwise rotation of 90°. As can be seen, the cells are still reasonably well scaled.

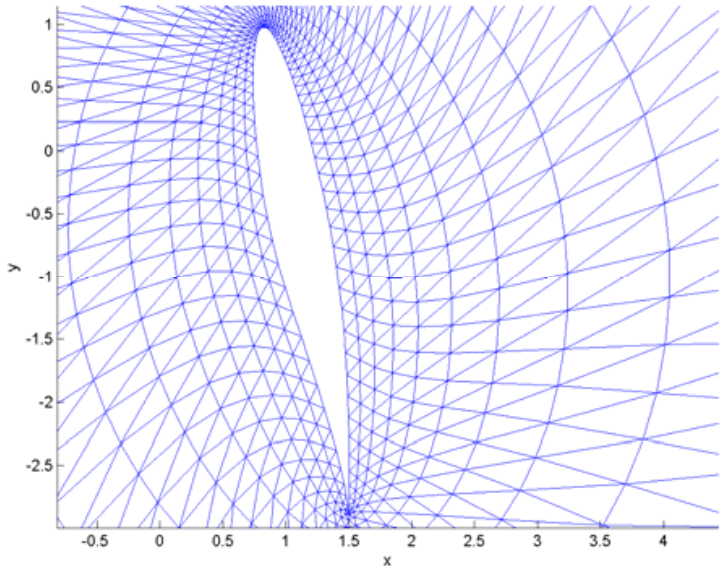

Figure 2.10: Mesh deformation using a material deformation model. From Premasuthan and Jameson [65]

Chapter 3

Flow Control using Adjoint Sensitivities

In this chapter, we lay out the groundwork for the concept of flow control as discussed in this thesis.

Our aim is to control the behavior of an aerodynamic flow that satisfies the *Euler* equations. The controllers we use are steady normal jets at the wall that affect the behavior of the flow by simulating a shape change. We would like to devise a *feedback* based control algorithm [22]. By *feedback* based we mean that the control correction required of the controller at any point in time is a function of the then state of the system. The control law thus derived has the following advantages:

1. Works at any operating condition. Open loop control algorithms are usually effective only in the neighborhood of the design conditions.

2. Higher robustness in the context of uncertainties in state measurements. Since the control correction required is dependent on state at any point in time, the resilience to noise is higher.

3. No control input needed at equilibrium points.

Objectives: The chief objectives of this chapter are as follows:

1. Establish that a *feedback* based control problem can be formulated as an equivalent optimization problem.

2. Formulate the generalized flow control problem as an optimization problem and discuss its solution using *Adjoint* based sensitivities.

3. Highlight the computational advantages offered by the *Adjoint* method over a finite difference method for the calculation of sensitivities.

4. Specialize the control law thus derived for the case of *Virtual Aerodynamic Shaping*, and highlight its *feedback* nature.

5. Discuss 2- and 3- dimensional *Virtual Aerodynamic Shaping* results.

3.1 *Feedback* Based Control

Consider a system similar to the one shown in Figure 3.1. The transfer function of the system is F. Thus, for any control input u, the output of the system is given by

$$y = F(u). \tag{3.1}$$

Now suppose that it is desired to choose a particular value of u, that will produce an

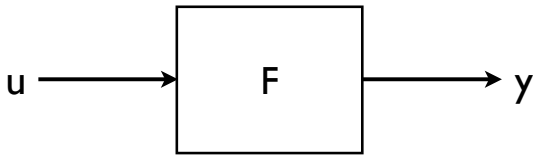

Figure 3.1: Block diagram of an open loop system with Plant Transfer Function F

output y_d. This is a difficult problem and requires detailed knowledge of the plant's transfer function. It is conceivable that a value of u could be chosen by trial and error. This is not a very robust way of choosing u, especially if there is plant and sensor noise.

Thus, we choose instead to implement a closed loop controller as in Figure 3.2. This feeds back the current state of the system y, and the correction $(y - y_d)$ determines the control input required. The objective is to drive the difference $(y - y_d)$ to zero, and this is what the controller is designed to accomplish.

The task of choosing the value of u that drives y to y_d is thus transferred from the designer to the controller. The question then is: how does one design such a controller? We try to achieve this by posing the control problem as an optimization problem. Consider

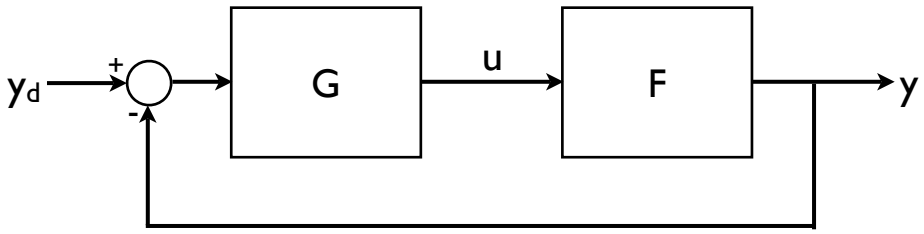

Figure 3.2: Block diagram of a closed loop system with Plant Transfer Function F and Controller Transfer Function G

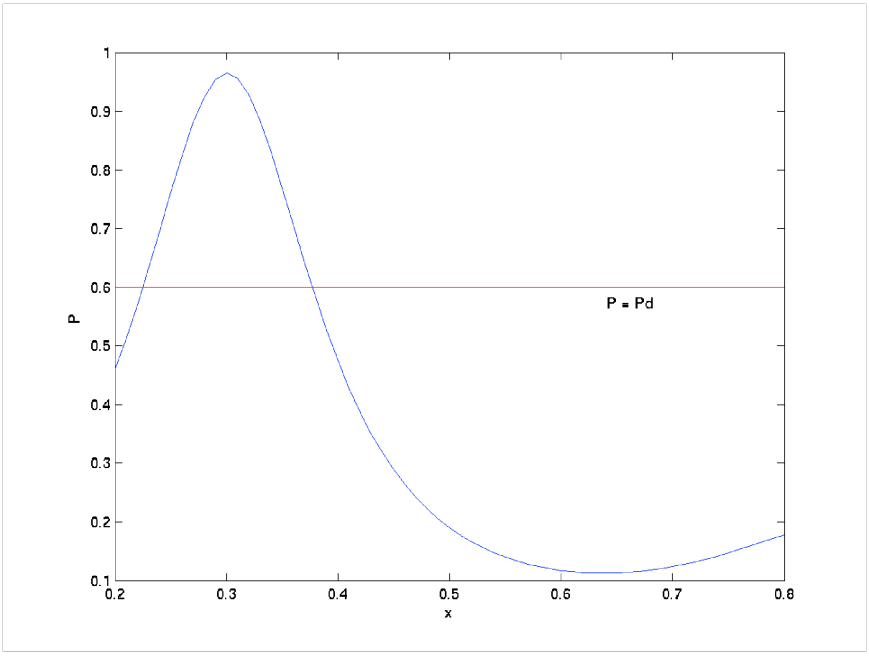

Figure 3.3: Plant Transfer Function F

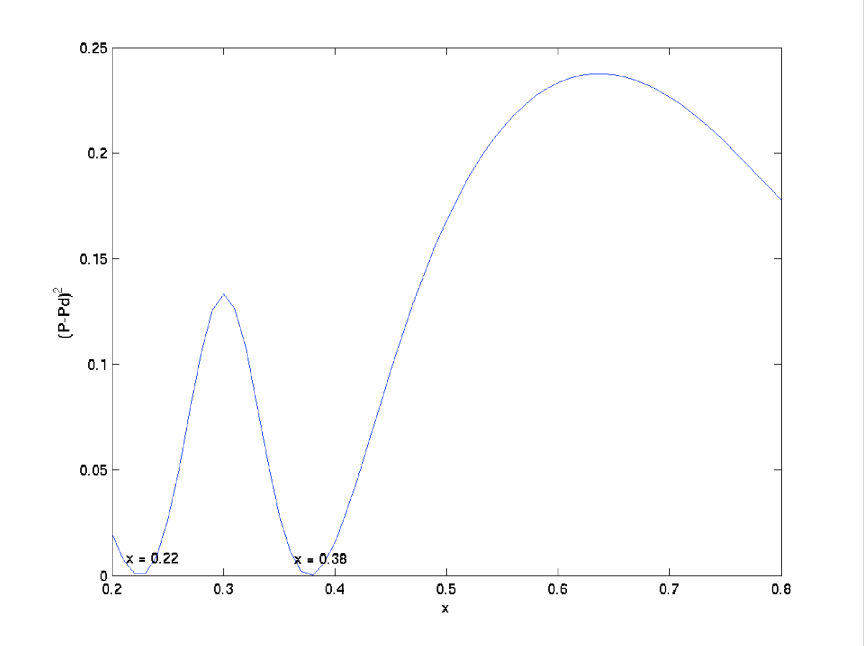

Figure 3.4: Plant Transfer Function F, reformulated as an optimization problem

an optimization problem of the form

$$\begin{aligned} \textbf{Minimize:} \quad & I = (y - y_d)^2 \\ \textbf{Subject to:} \quad & y = F(u) \ . \end{aligned}$$

Clearly, when I is minimum,

$$\frac{\partial I}{\partial u} = 2(y - y_d)\frac{\partial F}{\partial u} = 0. \tag{3.2}$$

If this optimization problem is solved using a *Newton* iteration, the updated value of u, based on the current value of y is given by the *Newton* step

$$u_{n+1} = u_n - \mathcal{H}^{-1}\mathcal{G}(y - y_d), \tag{3.3}$$

where $\mathcal{H}$ is the Hessian of the plant transfer function and $\mathcal{G}$ is the gradient. If the controller transfer function is chosen as above, the system automatically chooses that value of u that drives I to zero.

An example transfer function is shown in Figure 3.3. Here the input variable is x, and

the output is P. Consider now the problem of choosing x such that $P = P_d = 0.6$. An open loop equivalent of this would be to choose that particular value of x that satisfies our requirement. As discussed earlier, this is possible only if we have apriori knowledge of the system dynamics. We will instead re-phrase the control problem as an optimization problem. When the problem is written as an optimization problem, where $(P - P_d)^2$ is required to be minimized, the system automatically settles down at one of the two local minima, both of which are equally satisfactory for our purposes. This can be seen in Figure 3.4.

Thus it is clear that the *feedback* control problem can be recast as an optimization problem.

3.2 Optimal Fluid Flow Control

Following along the lines of the argument made in the previous section, the Flow Control problem can be posed as the following optimization problem:

$$\textbf{Minimize: } I(\mathbf{w}, \mathbf{u}), \tag{3.4}$$

where $\mathbf{w}$ is the vector of state variables and $\mathbf{u}$ is the vector of control variables. I represents the quantity being controlled. It could be a measure of the drag, expressed as a difference between the current surface pressure distribution and an *ideal* surface pressure distribution that has minimum drag. I could also be a measure of the deviation from an equilibrium condition. For example, in the case of flutter control, I is chosen to be a measure of the plunge and pitch of the wing. Minimizing I in this context is equivalent to bringing the system back to equilibrium.

The state vector $\mathbf{w}$ consists of the *Euler* state variables at each Finite Volume in the domain. Thus, if there are a million cells in the domain, the dimension of $\mathbf{w}$ is five million for a 3-d flow. The control vector $\mathbf{u}$ in our case, consists of the surface normal mass fluxes at every cell along the surface of the wing. The dimensionality of the control vector $\mathbf{u}$ is much smaller than that of the state vector $\mathbf{w}$.

The optimal control $\mathbf{u}$ is one that minimizes the cost function (3.4). The first derivative

of the cost function with respect to the control variables $\mathbf{u}$ is

$$\frac{dI}{d\mathbf{u}} = \frac{\partial I}{\partial \mathbf{w}}^T \frac{\partial \mathbf{w}}{\partial \mathbf{u}} + \frac{\partial I}{\partial \mathbf{u}}. \tag{3.5}$$

While the second term is fairly straightforward to evaluate, the first is not. This is because the state $\mathbf{w}$ and the control $\mathbf{u}$ are related by the *Euler* equations which are of the form

$$R(\mathbf{w}, \mathbf{u}) = 0. \tag{3.6}$$

We now present two ways to evaluate the required sensitivity, and highlight the computational advantages presented by using the *Adjoint* method.

3.2.1 Direct Sensitivity Analysis

First we try to estimate the required sensitivity directly. We note that the constraint Equation (3.6) is invariant with respect to $\mathbf{u}$

$$\frac{dR}{d\mathbf{u}} = \frac{\partial R}{\partial \mathbf{w}}^T \frac{\partial \mathbf{w}}{\partial \mathbf{u}} + \frac{\partial R}{\partial \mathbf{u}} = 0. \tag{3.7}$$

Thus

$$\frac{\partial \mathbf{w}}{\partial \mathbf{u}} = -\left[\frac{\partial R}{\partial \mathbf{w}}^T\right]^{-1} \frac{\partial R}{\partial \mathbf{u}} \tag{3.8}$$

The required sensitivity is then

$$G = \frac{dI}{d\mathbf{u}} = \frac{\partial I}{\partial \mathbf{u}} - \frac{\partial I}{\partial \mathbf{w}}^T \left[\frac{\partial R}{\partial \mathbf{w}}^T\right]^{-1} \frac{\partial R}{\partial \mathbf{u}}. \tag{3.9}$$

3.2.2 Adjoint Sensitivity Analysis

In the *Adjoint* framework, we do not evaluate the partial derivative $\frac{\partial \mathbf{w}}{\partial \mathbf{u}}$ explicitly. We first note that Equation (3.7) is identically zero. Therefore, it is permissible to multiply it by a Lagrange multiplier Ψ and add it to Equation (3.5). This gives

$$\frac{dI}{d\mathbf{u}} = \frac{\partial I}{\partial \mathbf{w}}^T \frac{\partial \mathbf{w}}{\partial \mathbf{u}} + \frac{\partial I}{\partial \mathbf{u}} + \Psi^T \left[\frac{\partial R}{\partial \mathbf{w}}^T \frac{\partial \mathbf{w}}{\partial \mathbf{u}} + \frac{\partial R}{\partial \mathbf{u}}\right]. \tag{3.10}$$

Rearranging the terms in Equation (3.10) we get

$$\frac{dI}{d\mathbf{u}} = \left[\frac{\partial I}{\partial \mathbf{w}}^T + \Psi^T \frac{\partial R}{\partial \mathbf{w}}^T\right] \frac{\partial \mathbf{w}}{\partial \mathbf{u}} + \left[\frac{\partial I}{\partial \mathbf{u}} + \Psi^T \frac{\partial R}{\partial \mathbf{u}}\right]. \tag{3.11}$$

Now Ψ is an arbitrary multiplier. We can choose it to make the coefficient of $\frac{\partial \mathbf{w}}{\partial \mathbf{u}}$ zero. This is called the *Adjoint* equation

$$\frac{\partial I}{\partial \mathbf{w}}^T + \Psi^T \frac{\partial R}{\partial \mathbf{w}}^T = 0. \tag{3.12}$$

Thus,

$$\Psi = -\left[\frac{\partial R}{\partial \mathbf{w}}\right]^{-1} \frac{\partial I}{\partial \mathbf{w}}. \tag{3.13}$$

The expression for the gradient of the cost function then becomes,

$$G = \frac{dI}{d\mathbf{u}} = \frac{\partial I}{\partial \mathbf{u}} - \frac{\partial I}{\partial \mathbf{w}}^T \left[\frac{\partial R}{\partial \mathbf{w}}^T\right]^{-1} \frac{\partial R}{\partial \mathbf{u}}. \tag{3.14}$$

3.2.3 Computational Advantages of the Adjoint Method

It can be seen from Equations (3.9) and (3.14) that both methods yield the same value for the gradient. It is instructive to see how the *Adjoint* framework presents a substantial computational advantage in our case.

In the direct method, we have to evaluate the derivative $\frac{\partial \mathbf{w}}{\partial \mathbf{u}}$. Given that the dependence of the state variables $\mathbf{w}$ on the control variables $\mathbf{u}$ cannot be expressed directly, the above derivative can only be evaluated using a Finite Difference technique. If there are m control variables $\mathbf{u}$, the required derivative can be evaluated only if the *Euler* equations are solved $m + 1$ times: once for the baseline case, and m times for a perturbation in the direction of each control variable.

In the *Adjoint* method, we have to evaluate the *Adjoint* variable Ψ. This is readily obtained as a solution to the *Adjoint* equation, whose solution is approximately of the same level of complexity, computationally speaking, as that of the *Euler* equations.

Thus, the computational work required to calculate the gradient is substantially less when the *Adjoint* method is used. Thus, the *Adjoint* method is used to calculate sensitivities throughout this thesis. A detailed derivation of the *Adjoint* equations for the *Euler* equations with blowing/suction at the walls is presented in Appendix B.

3.2.4 Smoothed Gradient

It makes physical sense that the blowing/suction velocities at the wall increase or decrease smoothly, as a function of distance along the surface. However, this is not necessarily reflected when the gradient is evaluated. Numerical experiments conducted by Jameson and Vassberg [42] show that smoothing the gradient, stabilizes the optimization algorithm, and usually leads to faster convergence, even when a Steepest Descent method is used. The smoothed gradient $\bar{G}$ is evaluated implicitly using a Laplacian method as follows

$$\bar{G} - \frac{\partial}{\partial \xi}\epsilon\frac{\partial}{\partial \xi}\bar{G} = G. \tag{3.15}$$

3.2.5 Optimization by the Continuous Descent Method: Convergence

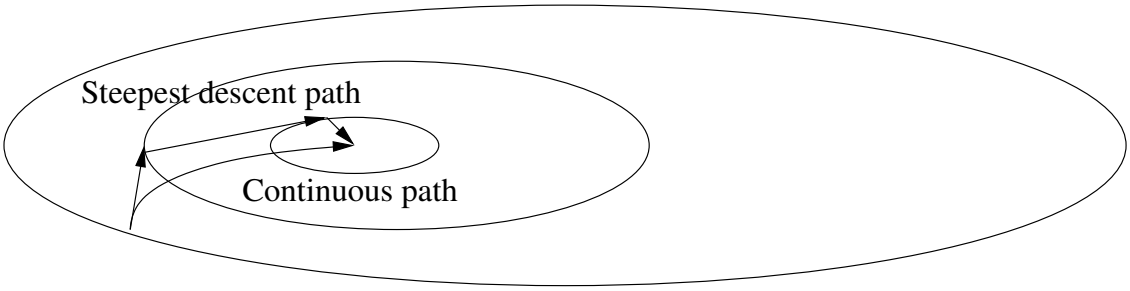

Figure 3.5: Schematic Diagram outlining the continuous descent optimization procedure

Figure 3.5 outlines the optimization algorithm used in this thesis: The Continuous Descent Method. The change in the cost function at any point is given by

$$\delta I = \bar{G}^T \delta u. \tag{3.16}$$

In the continuous descent method, we choose the step δu to be proportional to the gradient

$$\delta u = -\lambda \bar{G}, \tag{3.17}$$

where λ is an arbitrarily small positive value. Obviously, this choice for δu is guaranteed to minimize I

$$\delta I = -\lambda \bar{G}^T \bar{G} \leq 0. \tag{3.18}$$

Thus the continuous descent method is always guaranteed to converge to a local minimum.

3.3 Optimization Algorithm

The optimization algorithm is fairly straightforward and is outlined in Figure 3.6. We need

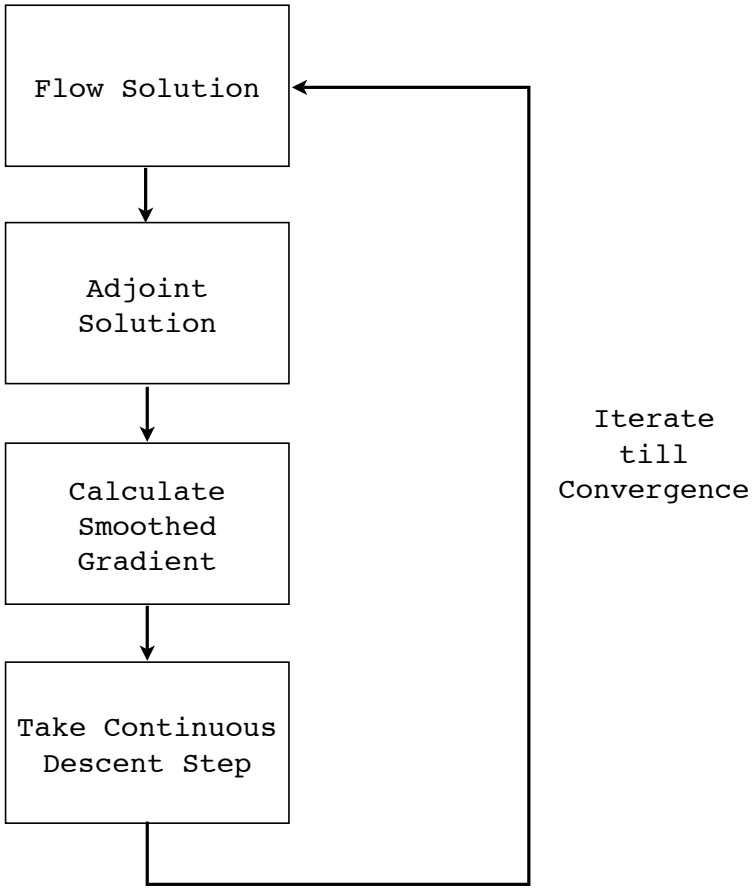

Figure 3.6: Optimization Algorithm

to calculate the flow solution about the nominal point first. We then solve the *Adjoint* equation to obtain the gradient. We then smooth the gradient as shown in Equation (3.15). The control variables are then readjusted according to the continuous descent method.

This procedure is repeated till convergence.

3.4 Virtual Aerodynamic Shaping

We test the flow control algorithms formulated in the previous sections on a *Virtual Aerodynamic Shaping* problem. Here, we take an aerodynamic configuration, and try to make it behave like another aerodynamic configuration under the same free stream conditions by implementing surface flow control.

A basic review of ideal fluid aerodynamics reveals that including a mass source on the surface of an airfoil has the effect of increasing the curvature and including a mass sink has the opposite effect. Given that, for a pre-determined performance measure, every operating condition has an optimum shape, it follows that a combination of a shape that is optimal for one operating condition, combined with suitably placed sources and sinks will result in a system that meets optimum performance criteria for a wide range of operating conditions.

For the purposes of this study, steady jets are used in order to simplify numerical modeling. Mass flows are prescribed at the wall, and the jets are modeled so as to satisfy the normal velocity/flux conditions at the wall. In addition, the nett. mass flow through the wall is assumed to be zero.

$$\int_{\mathcal{B}_\xi} \rho q_n d\mathcal{B}_\xi = 0. \tag{3.19}$$

3.5 Feedback Nature of Adjoint Based Control

The *Feedback* nature of the *Adjoint* based control laws thus derived become clear, when the *Adjoint* boundary conditions are examined.

3.5.1 The Adjoint Boundary Conditions for Virtual Aerodynamic Shaping

We postulate that the behavior of any aerodynamic configuration, at any freestream condition, is defined by its surface pressure distribution $P_d(\xi, \eta)$. Thus, when we talk about *virtually* shaping one configuration to behave like another, what we really mean to do is to places mass sources and sinks at suitable locations on the configuration such that the resulting pressure distribution $P(\xi, \eta)$ is the same as the target pressure distribution.

To achieve this, we try to minimize a cost function of the form

$$I = \frac{1}{2} \int_{\mathcal{B}_\xi} (P - P_d)^2 d\mathcal{B}_\xi, \tag{3.20}$$

where P is the pressure at the surface with no controls, and P_d is the desired target pressure. Observe that this cost function is a special instance of the more generic cost function described in Equation (B.1). We can choose our computational co-ordinates such that $\psi_1 = \psi_3 = 0$, and the normal direction is along ψ_2. When the blowing velocities prescribed at the wall are very small, the *Adjoint* boundary condition (Equation (B.12)) reduces to

$$\frac{\partial \mathcal{M}}{\partial P} = \psi^T \frac{\partial F_2}{\partial P} \text{ , on } \mathcal{B}_\xi \ . \tag{3.21}$$

For *Virtual Aerodynamic Shaping*, this is simply

$$(P - P_d) = \psi_2 S_{21} + \psi_3 S_{22} + \psi_4 S_{23}. \tag{3.22}$$

The corresponding *Adjoint* gradient (following from Equation (B.14)) is then

$$\delta I = -\int_{\mathcal{B}_\xi} \left(\psi_1 + \psi_2 u + \psi_3 v + \psi_4 w + \psi_5 \left(E + \frac{P}{\rho} \right) \right) \delta \rho q_n d\mathcal{B}_\xi \ . \tag{3.23}$$

It can be seen from Equation (3.22) that the *Adjoint* boundary condition is dependent on the difference between the desired state of the system P_d and the current state of the system P. Moreover, Equation (3.23) clearly shows that the *Adjoint* Gradient depends only on the flow variables at the boundaries. Thus, it is clear that the *Adjoint* based control thus derived is *Feedback* based.

It should be noted however, that the *Adjoint* equation is solved using the computed state of the flow field and not the measured state. This bypasses the need for global measurements of the flow-field.

3.6 Results

3.6.1 2-d results

An RAE-82 airfoil was optimized for minimum drag at a Mach number of 0.77. The airfoil was constrained to operate at a C_L of 0.6. The pressure distribution of the optimized section was used as the target distribution for the flow control case, where blowing and suction is used to mimic the shape changes that lead to the desired pressure changes. The flow calculations were done on a 192×32 grid.

The original (solid) and optimized (dotted) airfoil are shown in Figure 3.7. The blowing

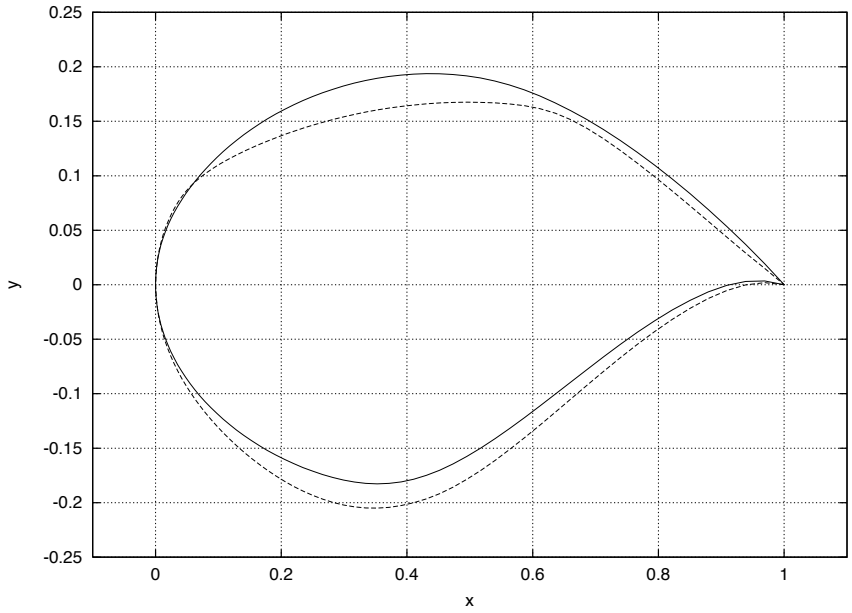

Figure 3.7: RAE-82 optimized for minimum drag at Mach 0.77: Original (solid) and Optimized (dotted)

and suction velocities that produce the same pressure distribution are shown in Figures 3.8 and 3.9. As expected, the controller implements blowing on the lower surface and suction on the upper surface. This corresponds very well with the shape change being represented, where the curvature is increased on the lower surface and decreased on the upper surface.

The Pressure distributions before and after applying flow control are shown in Figures 3.10 and 3.11. It can be seen that the flow control algorithm derived for the *virtual* aerodynamic design case achieves the desired results.

3.6.2 Reduction in the number of actuators: Design Trade-off

The results included in the previous section were for the case where blowing and suction is implemented continuously along the surface of the airfoil. Implementing this is not practical. We therefore try to reduce the number of actuators.

We would prefer to automatically arrive at the optimum number of actuators needed, and their locations. We do this by looking at the *Adjoint* gradient. The *Adjoint* gradient represents the sensitivity of the cost function with respect to the control variables. The

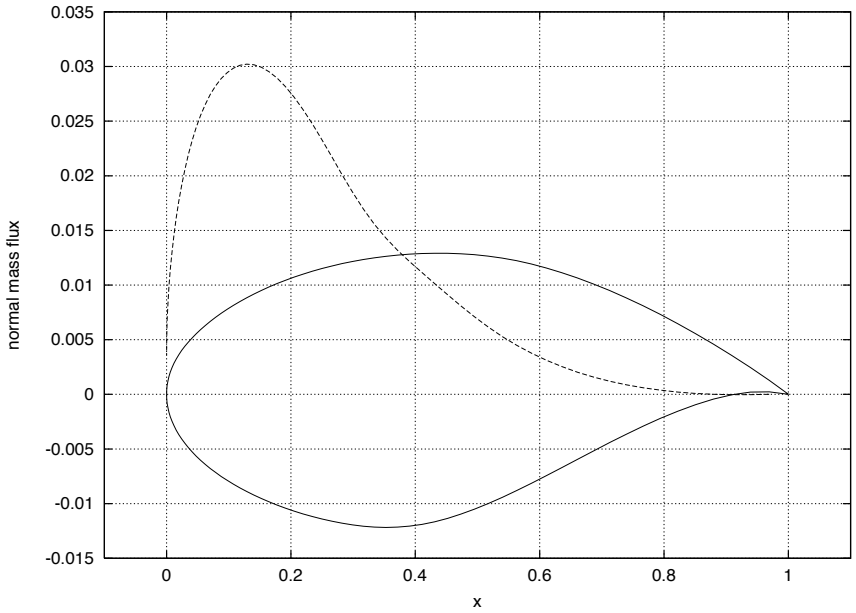

Figure 3.8: RAE-82: Flow control velocities on the lower surface for *Virtual Aerodynamic Shaping*

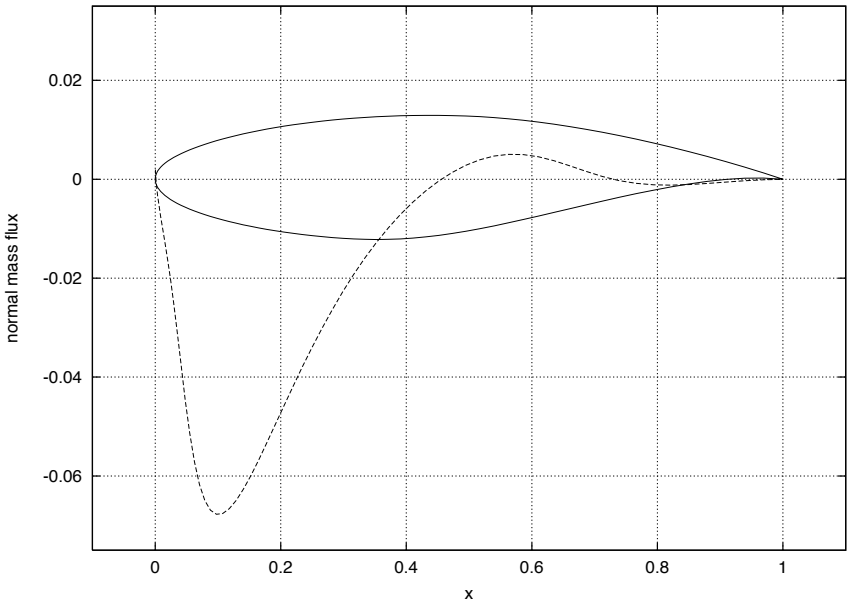

Figure 3.9: RAE-82: Flow control velocities on the upper surface for *Virtual Aerodynamic Shaping*

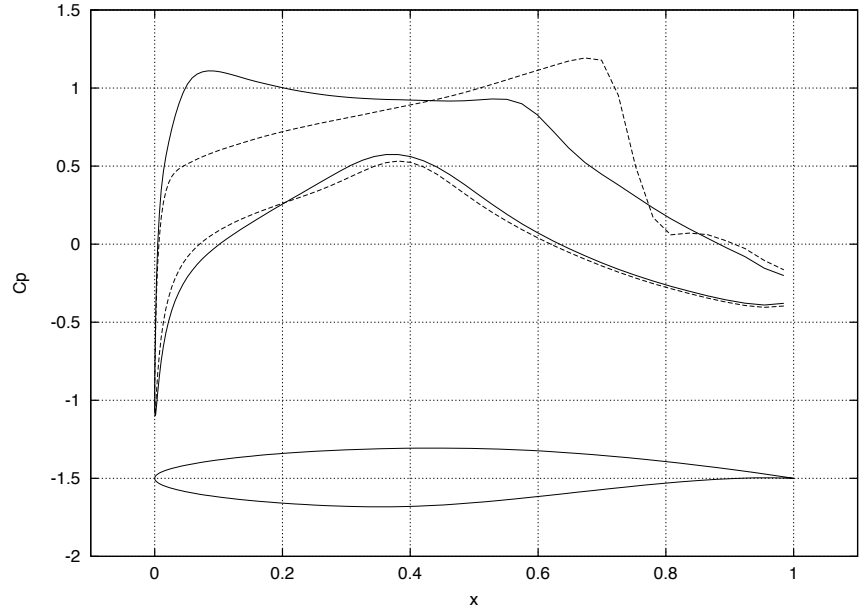

Figure 3.10: Pressure distributions: target (solid) and actual (dotted) before flow control

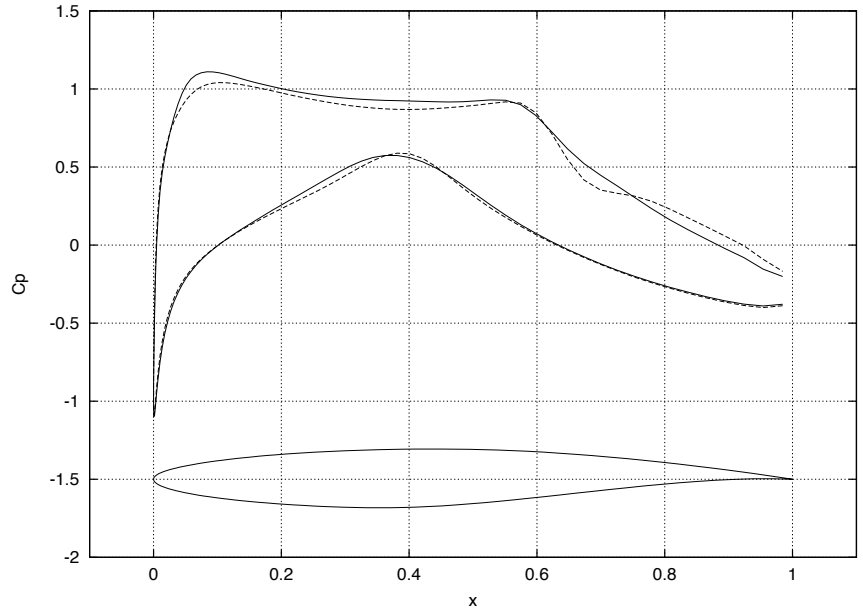

Figure 3.11: Pressure distributions: target (solid) and actual (dotted) after flow control

numerical values of the gradient derived thus indicate which controller locations are most effective and which controller locations are least effective.

In order to frame this mathematically, we chose to include all locations where the control input required was atleast 70 percent of that where the effort was maximum, and set the blowing/suction velocities at all other locations to zero.

$$\text{if } \rho q_n(\xi) \leq 0.7 \max \rho q_n \ , \quad \rho q_n(\xi) = 0. \tag{3.24}$$

It can be seen from Figures 3.12 and 3.13 that, suction control is applied only between about 5 percent chord and 30 percent chord on the upper surface, and no control is applied otherwise. The magnitude of suction required is about the same as that in the continuous control case (3.9).

Design Trade-off: The results of this experiment can be seen in Figure 3.14. It can be seen that the desired pressure distribution is almost obtained. The match between the desired and actual pressure distributions are quite close especially at the leading edge where the control is applied. In general, we make a compromise when we move from infinite dimensional control to finite dimensional control. The trade-off between the level of control accuracy desired and the number/location/size of the controllers that can be implemented is a design choice. The current section merely presents the algorithm that one would use to study the ramifications of such a tradeoff.

3.6.3 3-d Results

Finally, we check to see if we can achieve similar results in 3 dimensions. The Surface Pressure distribution of an ONERA M6 wing, constrained to operate at a C_L of 0.3 and a Mach number of 0.84 is shown in Figure 3.15. We try to achieve the same surface pressure distribution at the same freestream conditions for another wing that has a NACA 0012 section. The flow calculations are performed on a $192 \times 32 \times 48$ grid. It can be seen from Figure 3.16 that, after 5 control iterations, the surface pressure distribution resembles that of the ONERA M6. The pressure distributions along the front portions of the wing are almost identical. The pressure distributions along the trailing edge exhibit a slight difference. The original pressure distribution on the NACA 0012 wing is shown in the dotted lines.

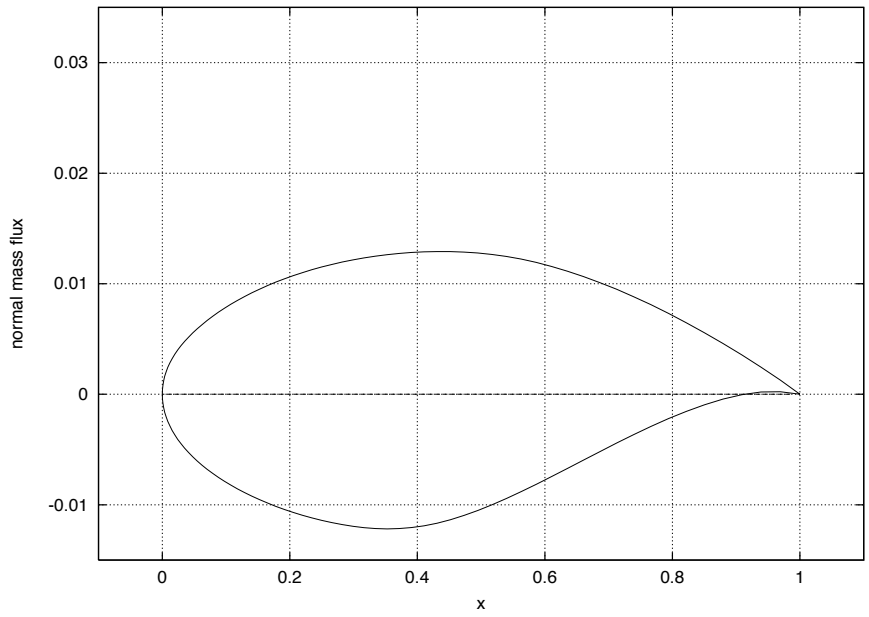

Figure 3.12: RAE-82: Flow control velocities on the lower surface for *Virtual Aerodynamic Shaping* – reduced number of actuators

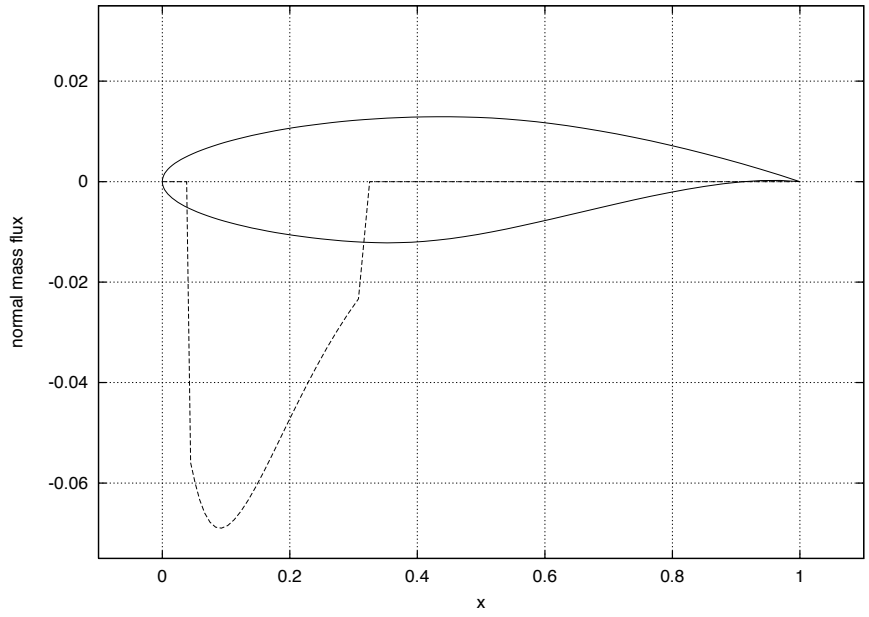

Figure 3.13: RAE-82: Flow control velocities on the upper surface for *Virtual Aerodynamic Shaping* – reduced number of actuators

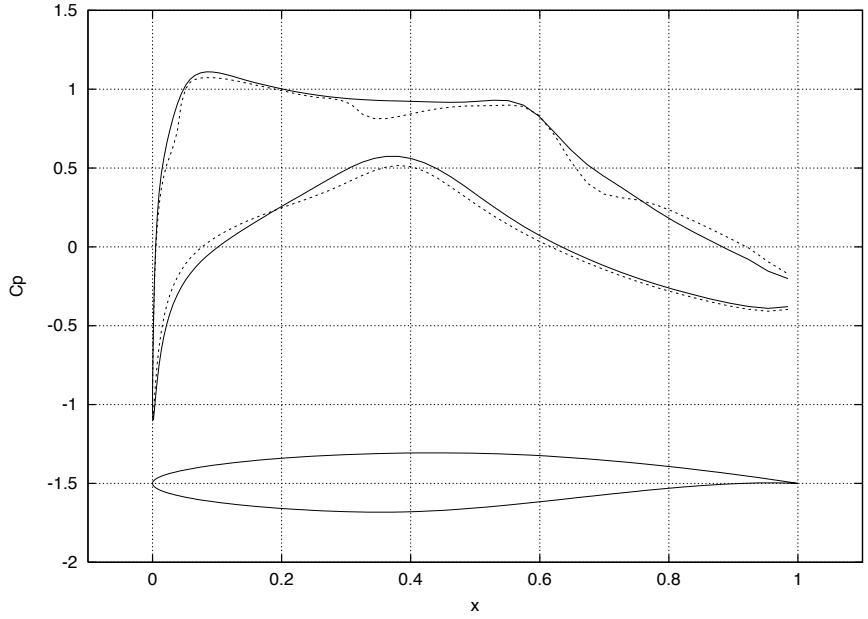

Figure 3.14: Pressure distributions: target (solid) and actual (dotted) after flow control - Reduced number of Actuators

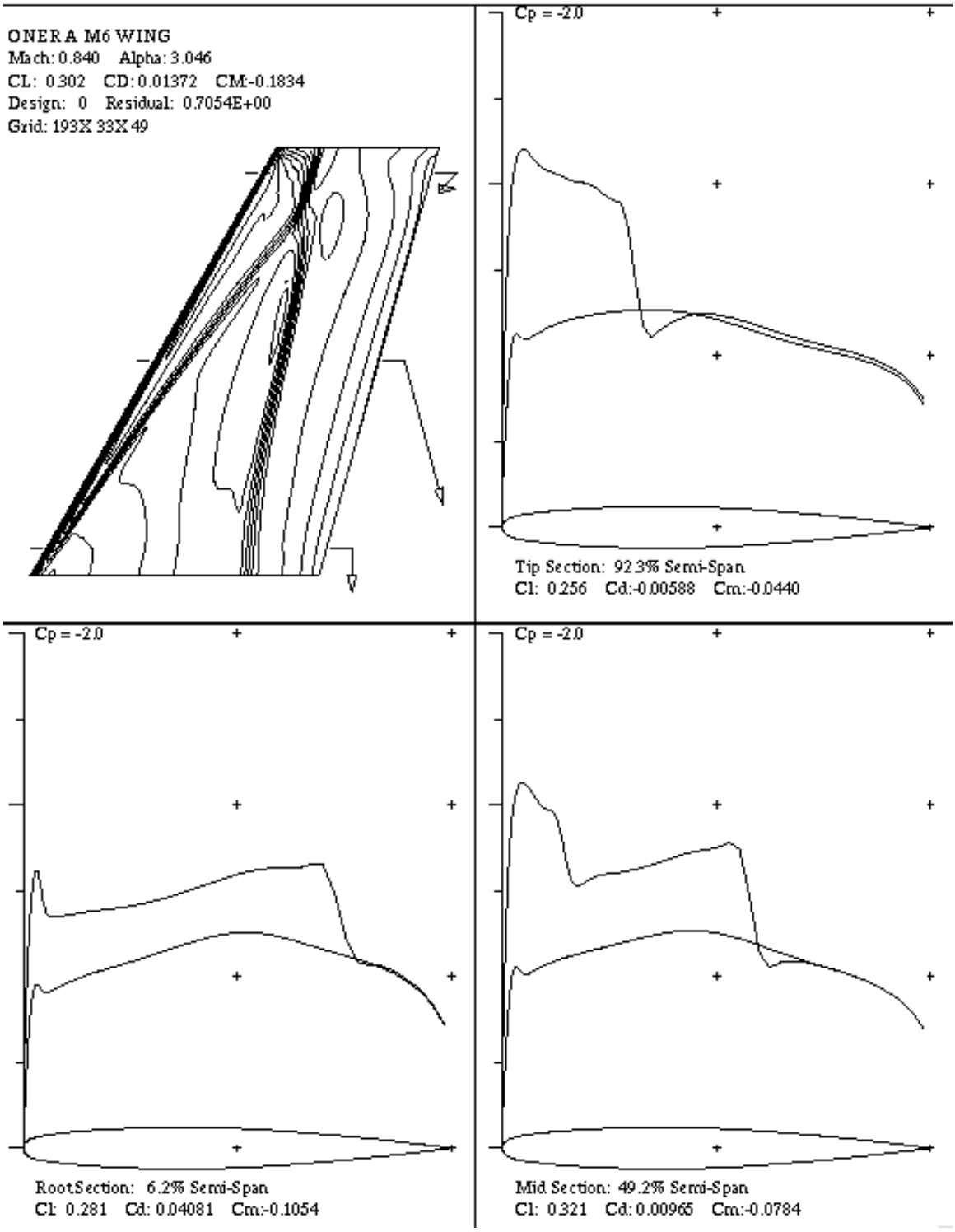

Figure 3.15: Cp distribution over the surface of an Onera M6 wing at $M = 0.84$ and $C_L = 0.3$

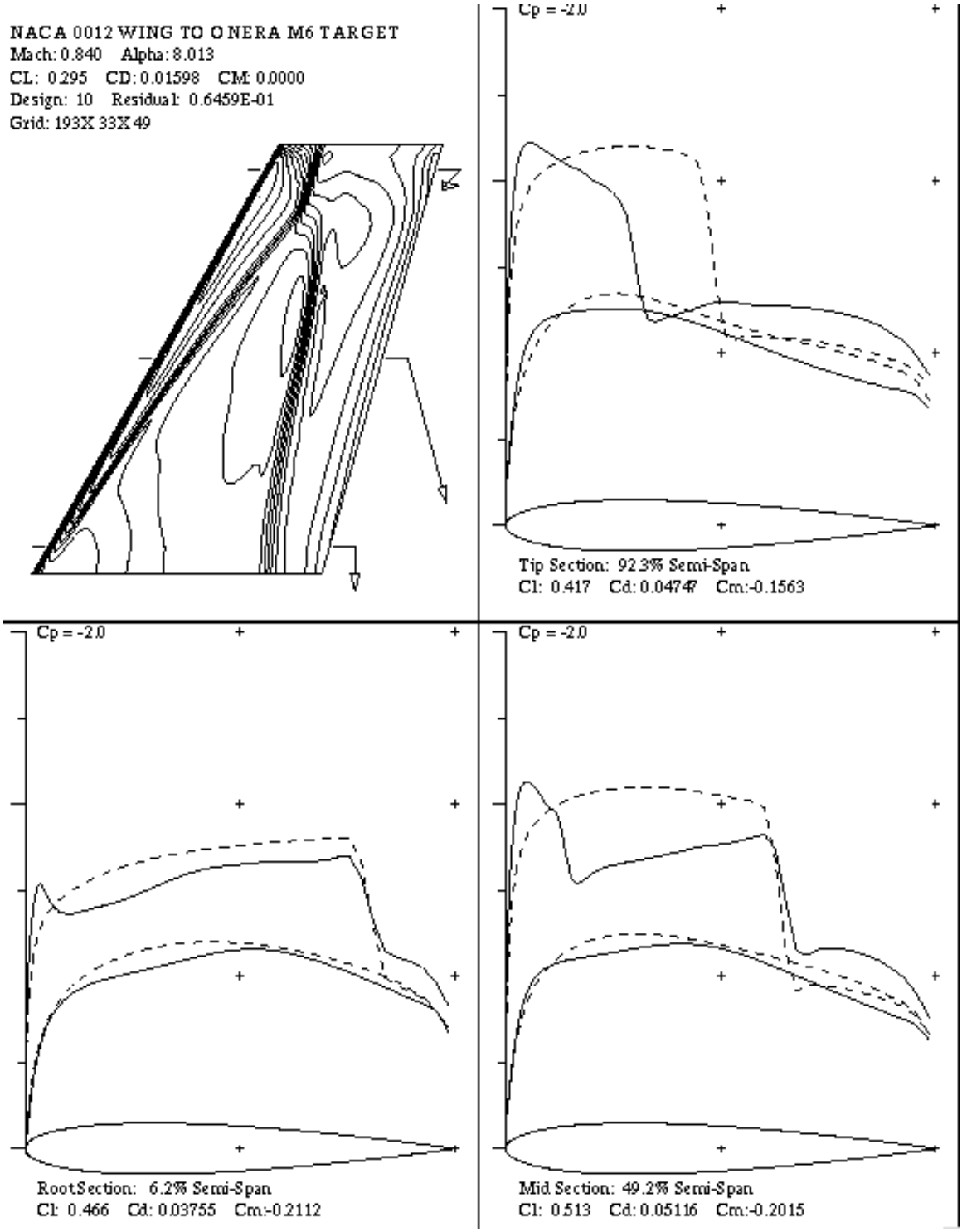

Figure 3.16: *Virtual Aerodynamic Shaping* of a NACA 0012 wing to match the surface pressure distribution of an Onera M6 wing at $M = 0.84$ and $C_L = 0.3$

Chapter 4

Structural Optimization

Flutter is an instability that arises out of adverse transfer of energy from the fluid to the structure and vice versa in an aero-elastic system. While we ultimately seek an active control solution, that addresses this issue, it is instructive to look at passive means first.

Passive flutter control would involve redesigning the structural layout of the airplane wing so that the flutter regime is pushed beyond the envisioned flight regimes. While good designers have reasonable intuition as to how things could be redesigned to achieve this shift, it is helpful to have an algorithmic way of doing this redesign.

We have the mathematical model of a system undergoing flutter. Our first approach therefore should be to see if it is possible to redesign the structure to move the flutter point further beyond the flight regime. We do this, again, by defining a suitable cost function, and finding the optimum structural properties using a continuous descent methods and *Adjoint* sensitivities.

Objectives: In this chapter, we alter the aero-structural behavior of a problem that is much simpler than full wing aero-elastic flutter. Our main objective is to demonstrate the effect of changing the structural properties, in the aero-structural behavior of a system.

We study the supersonic flow past a nonlinearly deforming plate. This results in non-linear Limit Cycle Oscillations under certain freestream conditions. We try to increase the maximum amplitude of the oscillations by altering structural properties. This problem was solved as part of a project with the Wright Patterson Air Force Base at Dayton, Ohio. The ultimate aim of this study was to design wing sections of micro air vehicles with flapping wings. Increasing the amplitude of oscillation was identified as one means of extracting the

maximum possible nonlinear cyclical lift from the wings.

4.1 Optimal Control of Limit Cycle Oscillations in a Nonlinear Panel

One of the most frequently encountered phenomena in aero-structural interactions amongst aerospace systems is the occurence of Limit Cycle Oscillations (LCOs). An LCO occurs when the system being described reaches a periodic steady state where there is a constant exchange of energy between the various degrees of freedom, in this case: the Aero and Structural components. LCOs represent the nonlinear nature of the system being observed. These might prove harmful if the amplitudes of oscillation are fairly high.

Some widely studied LCO problems in aero-elastic systems are that of an airfoil with a cubic nonlinear restoring force and a nonlinear panel interacting with supersonic flow. Beran *et al.* [8] have developed sophisticated computation algorithms for studying LCOs in general and have studied these problems in great detail.

The nonlinear nature of a system, however, is akin to a knife that cuts both ways. Ways have been studied where the nonlinear nature of a system can be exploited to our advantage. For instance, Blackburn *et al.* [13] show that combined translational and rotational motion can be used to propel a bluff body in quiescent fluid. It can be envisioned therefore that the nonlinear nature of aero-elastic systems can be harnessed so that the resulting behavior is beneficial to us. Such studies have a lot of potential in the design of Micro Air Vehicles (MAVs), whose operation is inspired by the flight of insects.

In this chapter, we try to alter the structural properties of a nonlinear panel that undergoes LCOs. Minimization of LCO amplitudes is a problem that is of relevance in the aerospace industry, viz. flutter suppression, etc. The enhancement of the amplitude of oscillation is a problem that is relevant to the design of MAVs and is studied herein. This is important because, at the conditions under which MAVs are operated, there are some very unique nonlinear phenomena that occur that can be exploited to our advantage.

4.2 Mathematical Model

The supersonic flow past a nonlinearly deforming panel is an interesting problem that has been studied in the past by Dowell *et al.* [17], and is especially relevant now in the context

of performance enhancement and control system design for MAVs, that expect to harness inherent nonlinear advantages of the aero-structural interactions. This also serves as a good multi disciplinary model on which our algorithms can be tested.

4.2.1 Structural Model

The nonlinear nature of the system arises because of the large scale deformations of the structure, which are modelled by von Karman's large-deflection plate equations. [23] and [76].

The equations of motion for the structural displacement of the system are as follows:

$$\frac{\partial^2 M_x}{\partial x^2} - F_x \frac{\partial^2 w}{\partial x^2} + \rho h \frac{\partial^2 w}{\partial t^2} = q - F_{AZ}, \tag{4.1}$$

$$\frac{\partial F_x}{\partial x} = 0, \tag{4.2}$$

where

$$\begin{aligned} F_x &= \frac{Eh}{1-\nu^2}\left[\frac{\partial u}{\partial x} + \frac{1}{2}\left(\frac{\partial w}{\partial x}\right)^2\right], \\ M_x &= D\frac{\partial^2 w}{\partial x^2}, \\ D &= \frac{Eh^3}{12(1-\nu^2)}. \end{aligned} \tag{4.3}$$

Here M_x and F_x are the moments and forces respectively in the x direction, E is the Young's modulus, h is the thickness of the structure, ρ is the density and ν is the Poisson's ratio. F_{AZ} is the applied force in the z direction and q is the transverse applied load. q is zero in our case. u and w are the displacements in the x and z directions respectively.

4.2.2 Aerodynamic Model

The aero component comes from the forcing term F_{AZ} in (4.1). This is modeled using Piston theory [12] (page 363) as follows

$$P - P_\infty = \frac{2q_\infty}{\sqrt{M_\infty^2 - 1}}\left[\frac{\partial w}{\partial x} + \left(\frac{M_\infty^2 - 2}{M_\infty^2 - 1}\right)\frac{1}{U_\infty}\frac{\partial w}{\partial t}\right]. \tag{4.4}$$

Here P_∞ is the freestream pressure and P is the pressure on the upper surface of the panel undergoing oscillations. q_∞ is the freestream dynamic pressure and U_∞ is the freestream velocity.

The above partial differential equations are discretized on a grid and values of the displacement w_i and velocity s_i are collected in a state vector $\mathbf{x}$:

$$\mathbf{x} = [s_1, w_1, \ldots, s_N, w_N]. \tag{4.5}$$

Now, Equations (4.1), (4.3) and (4.4) are simplified and non-dimensionalized:

$$\begin{aligned} x &\equiv x/L, \\ \tau &\equiv tU_\infty/L, \\ w &\equiv w/h, \\ \lambda &= 2q_\infty L^3/D\sqrt{M_{\infty^2}-1}, \\ \mu &= \rho L/\rho_m h. \end{aligned} \tag{4.6}$$

Here λ is a parameter that represents the aero-strucutral coupling of the system.

$$\lambda = \frac{24\ q_\infty L^3(1-\nu^2)}{Eh^3\sqrt{M_{\infty^2}-1}}. \tag{4.7}$$

As can be seen from Equation (4.7), λ depends on the freestream dynamic pressure q_∞ and the material Young's Modulus E and the plate thickness h. Changing any one of these directly influences the nonlinear oscillatory behavior of the system. Thus λ is chosen to be the control parameter for the aero-structural optimization.

In the case of the passive performance enhancement method, λ is assumed to take on an average value of $\bar{\lambda}$ throughout the length of the panel, with a perturbation λ'. The simplification involved assumes that the perturbation λ' is relatively small. Having made this assumption, and non-dimensionalizing, Equations (4.1), (4.3) and (4.4) can be written in the form of Equation (4.9) and the sensitivity analysis of Section 4.3 applies directly to the panel problem.

4.2.3 Computational Solution of the Nonlinear Equations

The nonlinear dynamical equations of an autonomous system can be written as

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}, \lambda). \tag{4.8}$$

Here t is the time, $\mathbf{x}(t)$ is an N_f dimensional array of real variables, and $\mathbf{f}$ is an N_f array of nonlinear functions dependent on $\mathbf{x}$ and λ, an N_d array of design parameters. Time is scaled by the LCO period, T, so as to yield a set of evolutionary equations in terms of the scaled system time, s.

$$\frac{d\mathbf{x}}{ds} = T\mathbf{f}(\mathbf{x}, \lambda). \tag{4.9}$$

This equation set is then solved using the cyclic method due to Beran *et. al.* [8]. This is done by expressing the time derivative in (4.9) as a second order central difference of the state vector at that instant in time. M such time instances are chosen to form the collocated vector $\mathbf{X}$, which is a collection of the values of $\mathbf{x}$ at M different time instances.

$$\mathbf{X} = (\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_M}). \tag{4.10}$$

The LCO equations are then

$$\mathbf{G}(\mathbf{X}, T, \lambda) = 0. \tag{4.11}$$

4.3 Adjoint Based Structural Optimization

As discussed earlier, different control objectives are sought for different physical problems. The one potentially most relevant to the design of MAVs is that of LCO amplitude amplification, and will be studied here. A suitable cost function for the case of LCO amplification would then be

$$I = \left(\frac{1}{2}\mathbf{X}^T Q \mathbf{X}\right)^{-1} + \frac{1}{2}\lambda'^T R \lambda', \tag{4.12}$$

where

$$\lambda' = \lambda - \bar{\lambda}, \tag{4.13}$$

where $\bar{\lambda}$ is an average value of the design parameter, and λ' is the deviation about the average. Q and R are suitable symmetric integration matrices. Minimizing the above cost function seeks to maximize the LCO amplitude. The first term in the expression for the cost

function, (4.12) ensures that the LCO amplitude is maximized over the entire time period, and the second term ensures that the values of the design variables don't grow boundlessly.

Taking a variation of the cost function I described in (4.12), we get

$$\delta I = -\frac{\mathbf{X}^T Q \delta \mathbf{X}}{\left(\mathbf{X}^T Q \mathbf{X}\right)^2} + \lambda'^T R \delta \lambda'. \tag{4.14}$$

Variation of the constraint function described by (4.11) yields

$$\delta \mathbf{G} = \frac{\partial \mathbf{G}}{\partial \mathbf{X}} \delta \mathbf{X} + \frac{\partial \mathbf{G}}{\partial T} \delta T + \frac{\partial \mathbf{G}}{\partial \lambda'} \delta \lambda' = 0. \tag{4.15}$$

We now subtract $\mathbf{\Psi}^T \delta \mathbf{G} = 0$ from (4.14), where $\mathbf{\Psi}$ is an arbitrary co-state vector. This co-state vector spans all the LCO states at all instants of time. We then re-arrange to seperate variations of solution variables from the variation of design variables as follows

$$\delta I = -\frac{\mathbf{X}^T Q \delta \mathbf{X}}{\left(\mathbf{X}^T Q \mathbf{X}\right)^2} - \mathbf{\Psi}^T \frac{\partial \mathbf{G}}{\partial \mathbf{X}} \delta \mathbf{X} - \mathbf{\Psi}^T \frac{\partial \mathbf{G}}{\partial T} \delta T + \left(\lambda'^T R - \mathbf{\Psi}^T \frac{\partial \mathbf{G}}{\partial \lambda'}\right) \delta \lambda'. \tag{4.16}$$

The co-state vector $\mathbf{\Psi}$ is then chosen to eliminate the first three terms in the RHS of (4.16) by satisfying

$$-\frac{\mathbf{X}^T Q \delta \mathbf{X}}{\left(\mathbf{X}^T Q \mathbf{X}\right)^2} - \mathbf{\Psi}^T \frac{\partial \mathbf{G}}{\partial \mathbf{X}} \delta \mathbf{X} - \mathbf{\Psi}^T \frac{\partial \mathbf{G}}{\partial T} \delta T = 0. \tag{4.17}$$

Equation (4.17) is called the *Adjoint* equation. With this selection δI is expressed only in terms of variations in the design variables as follows

$$\delta I = \left(\lambda'^T R - \mathbf{\Psi}^T \frac{\partial \mathbf{G}}{\partial \lambda'}\right) \delta \lambda', \tag{4.18}$$

leading to the identification of the gradient $\mathbf{g}$ in the optimization problem:

$$\mathbf{g}^T = \left(\lambda'^T R - \mathbf{\Psi}^T \frac{\partial \mathbf{G}}{\partial \lambda'}\right), \tag{4.19}$$

$$\delta I = \mathbf{g}^T \delta \lambda' \tag{4.20}$$

It should be noted that this is the first time that an *Adjoint* based sensitivity analysis procedure has been developed for a model of an LCO.

4.4 Results

The equations described in the previous section were set up computationally and solved using a steepest descent method. The mean value of $\bar{\lambda}$ as specified in Equation (4.13) is set to 4288.75. λ' is initially set to zero, and an optimum design value of λ' is sought. First

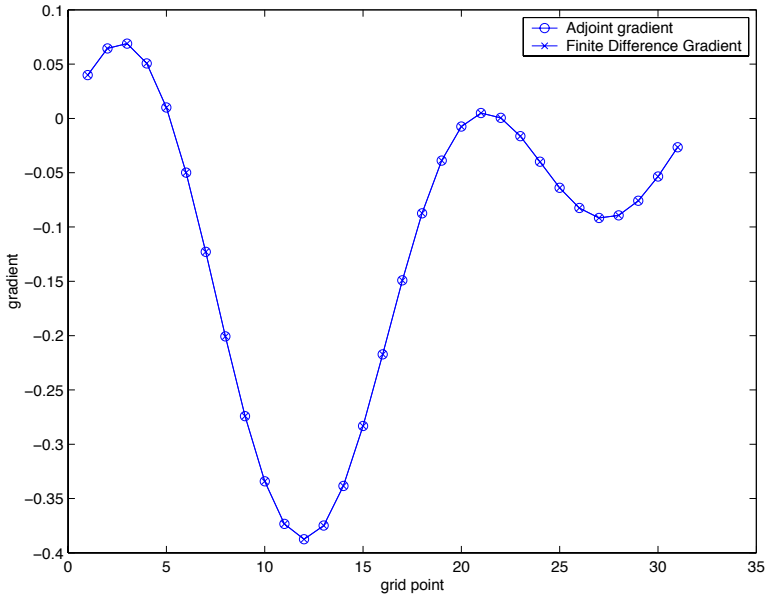

Figure 4.1: Comparison of *Adjoint* and Finite Difference gradients

of all, a comparison was made between the gradients calculated using the *Adjoint* method and the gradients calculated using the Finite Difference method. The results are presented in Figure 4.1. As can be observed from the figure, the two curves are almost identical. The difference in the numerical values of these gradients is shown in Figure 4.2.

The infinity norm of the gradient is plotted in Figure 4.3. As can be seen the changes in the gradient become smaller and smaller with increasing iteration numbers.

The constant property panel ($\lambda' = 0$) and the optimized panel are allowed to undergo LCO and their maximum deflection curves are compared in Figure 4.4. We wanted the amplitude of the oscillations to increase and it can be seen that this is exactly what happens.

Finally, the values of the perturbations in λ are plotted in Figure 4.5. It can be seen that these perturbations are fairly large. Another important observation that can be made is the fact that the optimizer, in order to passivley enhance the oscillation amplitudes, increases the flexibility at the $3/4\ th$ point which is where the amplitude of oscillation is maximum.

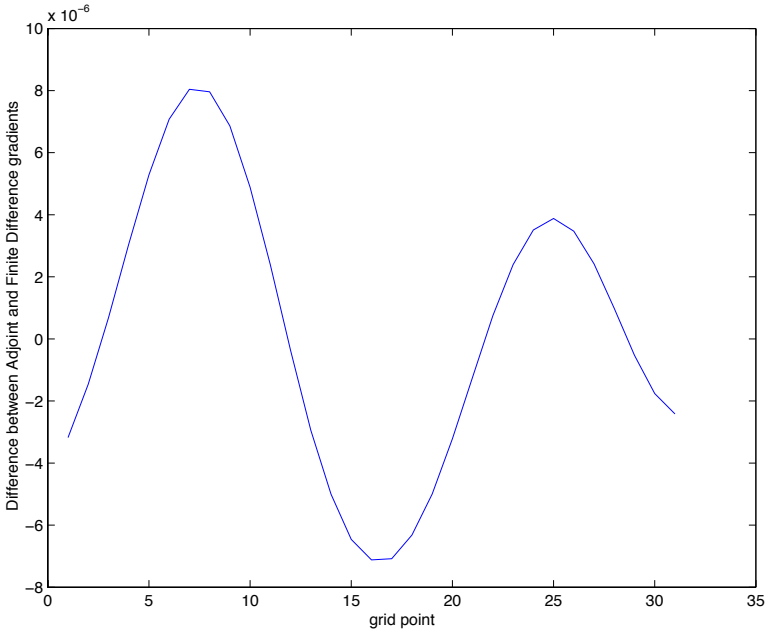

Figure 4.2: Difference between the *Adjoint* and Finite Difference gradients

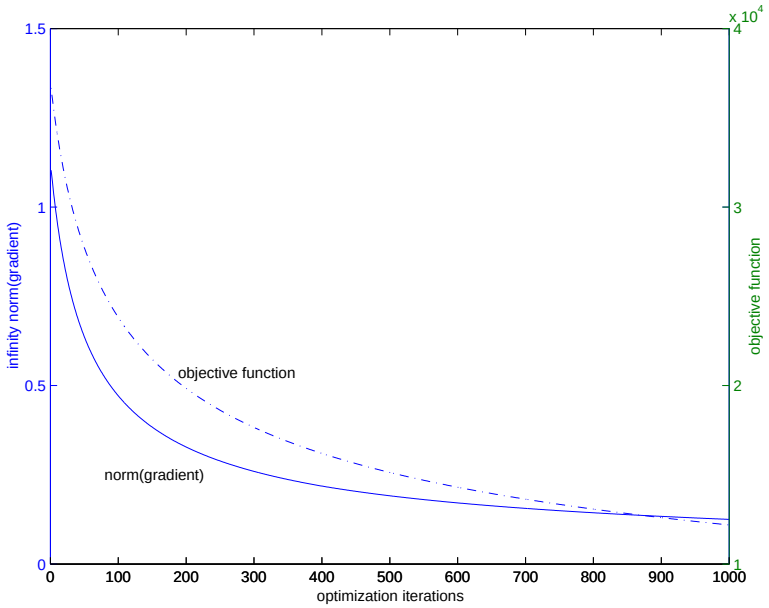

Figure 4.3: Gradient Convergence

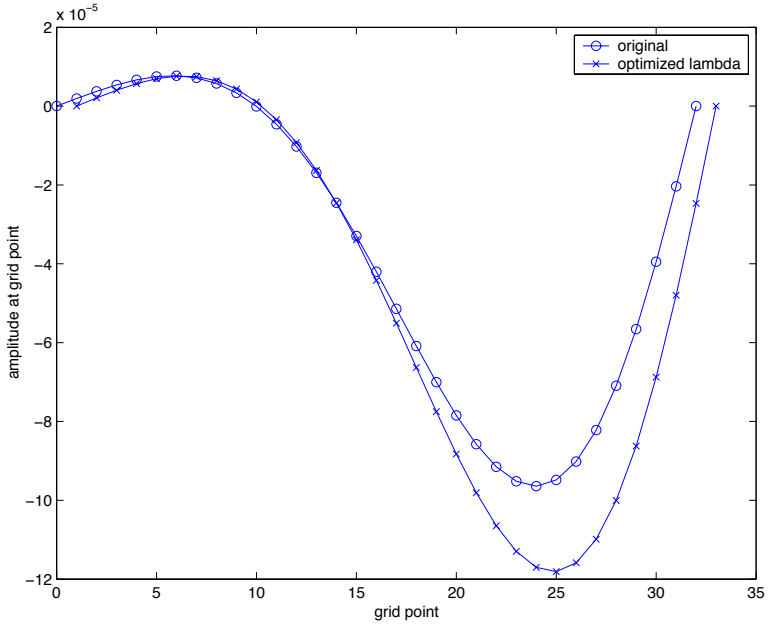

Figure 4.4: Comparison of maximum deflection curves of the base and enhanced panels

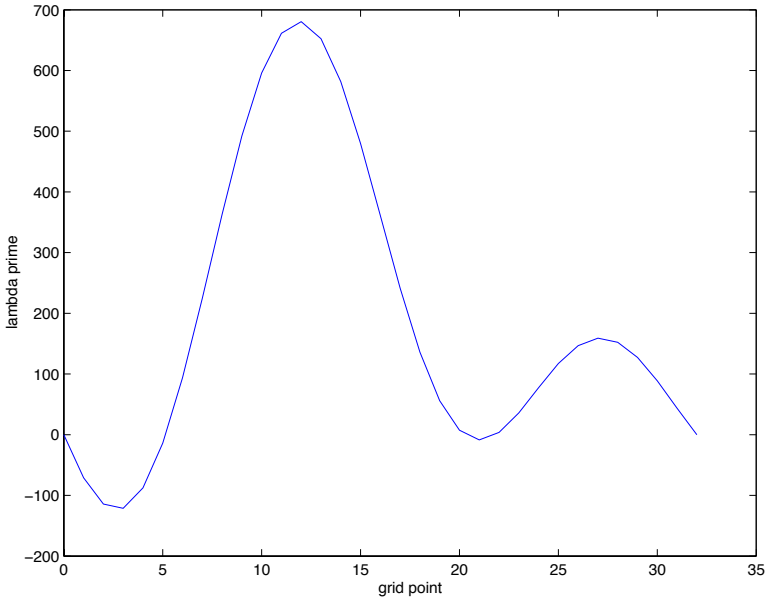

Figure 4.5: Values of λ' vs. x on the optimized Panel

Chapter 5

An Algorithmic Approach to Flutter Control

The structural design of an airplane is guided by static and dynamic factors. The more stringent constraints on the structural design are due to dynamic loads, caused by aero-elastic interactions. One of the most commonly encountered problems in aeroelasticity is flutter [12], a term that is used to recognize the transfer of energy from unsteady aerodynamics associated with the surrounding fluid to the wing structure, resulting in rapidly divergent behavior. If flutter can be controlled at cruise speeds, we can design lighter wings and consequently more efficient airplanes. It is therefore, in the aircraft designers best interest to design innovative ways in which flutter can be controlled without making the resulting structure too heavy.

There are three important choices to make while designing active control strategies for suppresing flutter. The first is the choice of actuator. In this thesis, the actuators we use are jets in the walls through which there is a small mass flow, either by way of blowing or suction. The second is to define a clear control objective. Finally, we need to design a control law that will make suitable state measurements and drive the actuators so that the desired control ob jective is achieved.

Objectives: In this chapter, we derive a control law for controlling flutter, based on the theory of optimal control. We describe the underlying control theory, and then show how to apply the control law thus derived to control flutter. We also discuss aspects of model order reduction and state linearization using *Adjoint* sensitivities. We demonstrate control of a

2-d model, and then show how this technique can be extended to a complete aero-elastic wing configuration.

5.1 Optimal Control of Nonlinear Dynamical Systems

One of the most commonly encountered problem in Control Theory is that of finding the optimum path taken by a system to reach a given state [15]. Consider a nonlinear dynamical system represented by the following set of ordinary differential equations:

$$\dot{\mathbf{x}} = f(\mathbf{x}(t), \mathbf{u}(t)). \tag{5.1}$$

The evolution of the system depends on its state at any given point of time: $\mathbf{x}(t)$, an n-vector, and a user chosen control function: $\mathbf{u}(t)$, an m-vector. The optimal control problem is then one of choosing $\mathbf{u}(t)$ such that a cost function of the form

$$J = \int_0^T L(\mathbf{x}(t), \mathbf{u}(t))\, dt + M(\mathbf{x}(T)), \tag{5.2}$$

is minimized. The cost function can be seen to be dependent on the integral of a function of the state $\mathbf{x}$ and the control $\mathbf{u}$. This is because we eventually want the state to reach the desired control target. Also we want to do this expending as little control effort as possible. In some cases, the control input might take only a few specific forms (e. g. Bang-Bang control). These are cases in which the form of the control input might need to be constrained to follow certain trajectories. This is reflected in the cost function chosen above. Taking the first variation of (5.1), we get

$$\frac{d}{dt}\delta\mathbf{x} = f_{\mathbf{x}}\delta\mathbf{x} + f_{\mathbf{u}}\delta\mathbf{u}. \tag{5.3}$$

Similarly, taking the first variation of the cost function gives

$$\delta J = \int_0^T (L_{\mathbf{x}}\delta\mathbf{x} + L_{\mathbf{u}}\delta\mathbf{u})\ dt + M_{\mathbf{x}}^T\delta\mathbf{x}(\mathbf{T}) \tag{5.4}$$

Multiplying (5.3) by a Lagrange Multiplier $\mathbf{\Psi}$, and adjoining it to (5.4) gives

$$\delta J = \int_0^T (L_{\mathbf{x}}\delta\mathbf{x} + L_{\mathbf{u}}\delta\mathbf{u})\ dt + M_{\mathbf{x}}^T\delta\mathbf{x}(\mathbf{T}) - \int_0^T \mathbf{\Psi}^T\ (\delta\dot{\mathbf{x}} - f_{\mathbf{x}}\delta\mathbf{x} + -f_{\mathbf{u}}\delta\mathbf{u})\ dt. \tag{5.5}$$

Integrating this by parts and collecting terms gives

$$\delta J = \int_0^T \left(L_{\mathbf{x}} + \mathbf{\Psi}^T f_{\mathbf{x}} + \dot{\mathbf{\Psi}}^T \right) \delta \mathbf{x} \, dt + \left(M_{\mathbf{x}}^T - \mathbf{\Psi}(\mathbf{T})^T \right) \delta \mathbf{x}(\mathbf{T}) + \int_0^T \left(L_{\mathbf{u}} + \mathbf{\Psi}^T f_{\mathbf{u}} \right) \delta \mathbf{u} \, dt. \quad (5.6)$$

It can be seen that the variation of the cost funtion depends on the variation in the state, $\delta\mathbf{x}$. This dependance can be eliminated by choosing the Lagrange multipliers $\mathbf{\Psi}$, such that they satisfy the following *Adjoint* Equation:

$$\dot{\mathbf{\Psi}} + f_{\mathbf{x}}^T \mathbf{\Psi} + L_{\mathbf{x}}^T = 0, \quad (5.7)$$

with the following boundary condition

$$\mathbf{\Psi}(\mathbf{T}) = M_{\mathbf{x}}. \quad (5.8)$$

This results in the following expression for δJ,

$$\delta J = \int_0^T \mathbf{g}^T \delta \mathbf{u} \, dt, \quad (5.9)$$

where

$$\mathbf{g}^T = L_{\mathbf{u}} + \mathbf{\Psi}^T f_{\mathbf{u}}. \quad (5.10)$$

5.2 Optimal Control of Linear Dynamical Systems: LQR Control

In the previous section, we designed an *Adjoint* based control algorithm for a nonlinear system. It has to be noted that this is extremely difficult to solve computationally. This is best seen from Equations (5.7) and (5.8). Given that the boundary condition is terminal, the above *Adjoint* equation can only be integrated in reverse time. If $f_{\mathbf{x}}$, $L_{\mathbf{x}}$ and $M_{\mathbf{x}}$ are functions of $\mathbf{x}$, the solution to this equation set is extremely unwieldy, given that the state $\mathbf{x}$ can be obtained only through a forward integration. Moreover, every time the control information is included in the solution of the state trajectories, the state trajectory changes necessitating re-computation of the *Adjoint* equation, and so on till the state-*Adjoint* equation system converges. If the dimensionality of $\mathbf{x}$ and $\mathbf{u}$ is large, as it is for the case of the flutter control problem, this becomes intractable. Therefore, we resort to a

linearization of the system. Consider the following Linear Dynamical System

$$\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}, \tag{5.11}$$

The control $\mathbf{u}$ is chosen so that the following cost function

$$J = \frac{1}{2}\int_0^T \left(\mathbf{x}^T Q\mathbf{x} + \mathbf{u}^T R\mathbf{u}\right)\, dt \tag{5.12}$$

is minimized. Here Q is chosen to be positive semi-definite, and R is chosen to be positive definite. Theoretically, any system is controllable provided an infinite amount of control is applied. This is not possible in practical systems, of course. A suitably chosen R will help circumvent this problem by imposing a cost on the control effort required. The relative magnitudes of the eigenvalues of Q and R determine the relative importance of control accuracy required versus control effort possible.

Based on the above system description, we note that

$$f_{\mathbf{x}} = A, \quad f_{\mathbf{u}} = B, \quad L_{\mathbf{x}} = \mathbf{x}^T Q, \quad L_{\mathbf{u}} = \mathbf{u}^T R. \tag{5.13}$$

The *Adjoint* equation is then given by

$$\dot{\mathbf{\Psi}} + A^T\mathbf{\Psi} + Q\mathbf{x} = 0, \quad \mathbf{\Psi}(\mathbf{T}) = 0. \tag{5.14}$$

The optimality condition is

$$\mathbf{g} = R\mathbf{u} + B^T\mathbf{\Psi} = 0. \tag{5.15}$$

This gives

$$\mathbf{u} = -R^{-1}B^T\mathbf{\Psi}. \tag{5.16}$$

For a linear system,

$$\mathbf{\Psi} = P\mathbf{x}. \tag{5.17}$$

Substituting (5.17) in (5.14), we get

$$P\dot{\mathbf{x}} + \dot{P}\mathbf{x} + A^T P\mathbf{x} + Q\mathbf{x} = 0, \tag{5.18}$$

or

$$PA\mathbf{x} + PB\mathbf{u} + \dot{P}\mathbf{x} + A^T P\mathbf{x} + Q\mathbf{x} = 0. \tag{5.19}$$

Substituting for $\mathbf{u}$,

$$PA\mathbf{x} - PBR^{-1}B^T P\mathbf{x} + \dot{P}\mathbf{x} + A^T P\mathbf{x} + Q\mathbf{x} = 0. \tag{5.20}$$

This gives the *Riccatti* equation:

$$PA - PBR^{-1}B^T P + \dot{P} + A^T P + Q = 0. \tag{5.21}$$

For all times t not close to the horizon T of the problem of interest the above differential equation reduces to the algebraic *Riccati* equation:

$$A^T P_{ss} + P_{ss}A - P_{ss}BR^{-1}B^T P_{ss} + Q = 0. \tag{5.22}$$

This equation can be solved to find the steady state value P_{ss}. The optimal control $\mathbf{u}(\mathbf{t})$ can be represented as a *feedback* control

$$\mathbf{u}(\mathbf{t}) = K_{ss}\mathbf{x}(\mathbf{t}), \tag{5.23}$$

where

$$K_{ss} = -R^{-1}B^T P_{ss}. \tag{5.24}$$

Thus, any linear dynamical system of the form $\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}$ can be controlled by a control law of the form given in Equation (5.23) over an infinite time horizon.

5.3 2-d Flutter Control

5.3.1 Typical Wing Section: Mathematical Model

In the present section we will investigate the aeroelastic behavior and control of a 2-d airfoil whose schematics is shown in Figure 5.1. A 2-d airfoil model can be shown to be a fair representation for flutter prediction as shown by Theodorson and Garrik [77] of a straight wing of a large span by giving it the geometric and inertial properties of the cross-section three quarters of the way from the centerline to the wing tip. The equations of motion of

this simple system can be shown to be as follows.

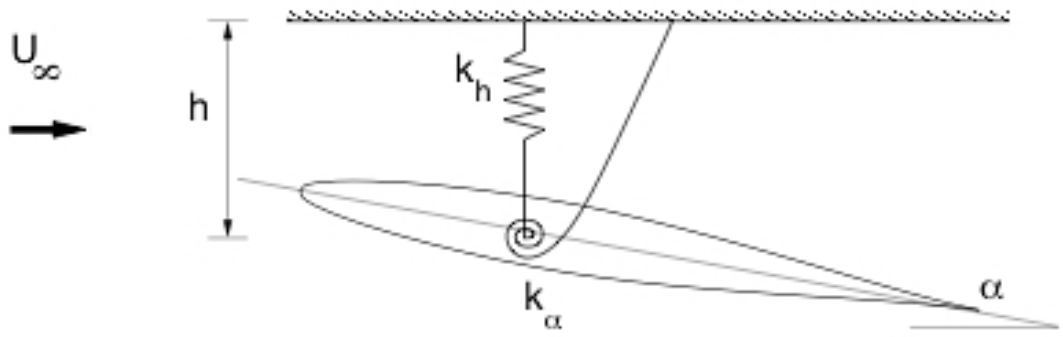

Figure 5.1: Typical Section Wing Model Geometry

$$m\ddot{h} + S_\alpha \ddot{\alpha} + K_h h = -L \tag{5.25}$$

$$S_\alpha \ddot{h} + I_\alpha \ddot{\alpha} + K_\alpha \alpha = M_{ea} \tag{5.26}$$

K_h and K_α are representative of the bending and torsional stiffness of the wing about its elastic axis.

The elastic axis is the locus of points about which, if a force is applied, doesn't result in any rotation about that point. m and I_α are the mass and moment of inertia of the wing section about the elastic axis. S_α is the coupling term which depends on the relative position of the center of gravity and the elastic axis.

We assume that the structural properties are fixed and we have some amount of control of the right hand sides of Equations (5.25) and (5.26) via blowing and suction. The objective is to find a suitable control law which will modify the aerodynamic terms so as to prevent flutter.

5.3.2 Computational simulation

The flow is simulated by solving the unsteady *Euler* equations. The *Euler* equations are solved using a dual time stepping method, using a third order backward difference formula in time, and a symmetric Gauss Seidel scheme for solving the inner iterations. The above mentioned flow simulation code is integrated with a two degree of freedom structural model given by Equations (5.25) and (5.26). The coupled aero-structural system is integrated

using the *Newmark* scheme. The simulation techniques are discussed in detail in Chapter 2 and Appendix B.

5.3.3 System Linearization and Model Order Reduction

In Equations (5.25) and (5.26), the structural parameters are constant. The lift L and the moment M are complex nonlinear functions of the system state $\mathbf{w}$, α, $\dot{\alpha}$, h and $\dot{h}$. Moreover, α, $\dot{\alpha}$, h and $\dot{h}$ are itself functions of the system state $\mathbf{w}$. Here the state $\mathbf{w}$ is the vector consisting of all the *Euler* states at all finite volumes used in the simulation. Thus

$$L = L(\mathbf{w}, \mathbf{u}) \,, \tag{5.27}$$

$$M = M(\mathbf{w}, \mathbf{u}) \,. \tag{5.28}$$

Linearizing about the nominal operating point, we get

$$L = \frac{\partial L}{\partial \mathbf{w}}^T \delta\mathbf{w} + \frac{\partial L}{\partial \mathbf{u}}^T \delta\mathbf{u}, \tag{5.29}$$

$$M = \frac{\partial M}{\partial \mathbf{w}}^T \delta\mathbf{w} + \frac{\partial M}{\partial \mathbf{u}}^T \delta\mathbf{u}. \tag{5.30}$$

It should be noted that for a simulation with one million finite volumes, the dimension of $\mathbf{w}$ is four million for a 2-d simulation and five million for a 3-d simulation. Thus evaluating the above derivatives is a formidable computational challenge. It is also important to recognize that not all the derivatives are significant in the above representation. Consider, for example, a cell in the far-field. The value of the state variables there is not going to change by much, however rapid the oscillations. Therefore, it is of very little use evaluating these derivatives in our linearized model.

Instead, we choose to obtain a suitable reduced order model that captures the essential physics. The most obvious reduction that we can obtain is in terms of α, $\dot{\alpha}$, h and $\dot{h}$. We therefore work with a model of the form:

$$L = L_\alpha \alpha + L_{\dot{\alpha}} \dot{\alpha} + L_h h + L_{\dot{h}} \dot{h} + \frac{\partial L}{\partial \mathbf{u}}^T \mathbf{u}, \tag{5.31}$$

$$M = M_\alpha \alpha + M_{\dot{\alpha}} \dot{\alpha} + M_h h + M_{\dot{h}} \dot{h} + \frac{\partial M}{\partial \mathbf{u}}^T \mathbf{u}. \tag{5.32}$$

Equations (5.31) and (5.32) assume that the nominal values of α, $\dot{\alpha}$, h and $\dot{h}$ and $\mathbf{u}$ are

zero, respectively. Thus for the flutter control problem being studied, the following state vector is used:

$$\mathbf{x} = [\alpha \quad \dot{\alpha} \quad h \quad \dot{h}]^T \tag{5.33}$$

5.3.4 System Identification: Evaluation of Sensitivities

In our aero-strucutral model (5.25) and (5.26), the lift L and the moment M depend on the complete system state $\mathbf{w}$. However, using a full order state model to design a controller is not feasible, given the extremely high dimensionality of the system. We therefore, formulate a reduced order model of the system as shown in Equations (5.31) and (5.32). In order for this model to be complete, we need to evaluate the sensitivities with respect to the reduced order state $\mathbf{x}$ and the control variables $\mathbf{u}$.

Sensitivities with respect to the state variables

The sensitivities of the lift and moment with respect to the state variables are evaluated in two different ways.

Theodorsen theory: First, we use theoretical results from Theodorsen [12]. Theodorsen theory assumes that the airfoil under consideration is thin, and is oscillating in an incompressible flow. Under these considerations

$$\begin{aligned} L_\alpha = \pi\rho v_\infty^2 c \quad &, \quad L_{\dot{\alpha}} = \frac{\pi\rho v_\infty c^2}{4}, \\ L_h = 0 \quad &, \quad L_{\dot{h}} = \pi\rho v_\infty c, \\ M_\alpha = \frac{\pi\rho v_\infty^2 c^2}{4} \quad &, \quad M_{\dot{\alpha}} = 0, \\ M_h = 0 \quad &, \quad M_{\dot{h}} = \frac{\pi\rho v_\infty c^2}{4}, \end{aligned}$$

Here ρ is the freestream density, v_∞ is the freestream velocity and c is the chord of the airfoil.

Least-Squares Method: In the second method, we evaluate the sensitivities, by studying the unforced response of a pitching airfoil, and then estimating the sensitivities by a least-squares technique. The aero-structural response of the system over a period of time is similar to the unforced response reproduced in Figures 5.4, 5.5, 5.6 and 5.7. These simulations

provide numerical values for

$$\begin{aligned} \alpha &= f_1(t) \\ \dot{\alpha} &= f_1'(t) \\ h &= f_2(t) \\ \dot{h} &= f_2'(t) \\ L &= f_3(t) \\ M &= f_4(t) \end{aligned}$$

We now try to fit the data thus obtained to functions of the form

$$\begin{aligned} L &= L_\alpha \alpha + L_{\dot{\alpha}} \dot{\alpha} + L_h h + L_{\dot{h}} \dot{h}, \\ M &= M_\alpha \alpha + M_{\dot{\alpha}} \dot{\alpha} + M_h h + M_{\dot{h}} \dot{h}. \end{aligned}$$

Our goal is to evaluate the sensitivities L_α, $L_{\dot{\alpha}}$, L_h, $L_{\dot{h}}$, M_α, $M_{\dot{\alpha}}$, M_h and $M_{\dot{h}}$. We do this using a least-squares technique.

It can be seen from the simulation results that both techniques work quite well. The system identification by the least-squares technique, works slightly better, in the sense, it achieves faster stabilization. This can be attributed to the fact that this represents the nonlinear system more closely.

Sensitivities with respect to the control variables

We also need to evaluate the sensitivities of L and M with respect to the blowing and suction velocities $\mathbf{u}$, $\frac{\partial L}{\partial \mathbf{u}}$ and $\frac{\partial M}{\partial \mathbf{u}}$ respectively.

We do this are using an *Adjoint* method as outlined in Chapter 3.

5.3.5 Flutter Control: Formulation of the Objective Function

We can define the flutter velocity as that point where we have sustained oscillations of the system. Let us define the state vector $\mathbf{x}$ as follows

$$\mathbf{x} = [\alpha \quad \dot{\alpha} \quad \mathbf{h} \quad \dot{\mathbf{h}}]^{\mathbf{T}} \tag{5.34}$$

The control vector $\mathbf{u}$ is the vector of blowing/suction velocities at the wall. The dynamics of the system is represented by (5.25) and (5.26). For the purposes of designing a controller, we model the lift L and the moment M using a reduced order model as presented in Equations (5.31) and (5.32). The system model used to design a controller is then

$$\begin{aligned} m\ddot{h} + S_\alpha \ddot{\alpha} + K_h h &= -\left(L_\alpha \alpha + L_{\dot{\alpha}} \dot{\alpha} + L_h h + L_{\dot{h}} \dot{h} + \frac{\partial L}{\partial \mathbf{u}}^T \mathbf{u} \right) \\ S_\alpha \ddot{h} + I_\alpha \ddot{\alpha} + K_\alpha \alpha &= \left(M_\alpha \alpha + M_{\dot{\alpha}} \dot{\alpha} + M_h h + M_{\dot{h}} \dot{h} + \frac{\partial M}{\partial \mathbf{u}}^T \mathbf{u} \right). \end{aligned}$$

This can be re-phrased in state space form as follows:

$$M\dot{\mathbf{x}} = \hat{A}\mathbf{x} + \hat{\mathbf{B}}\mathbf{u}. \tag{5.35}$$

Here the matrix $\hat{B}$ represents the sensitivities of the state vectors with respect to the control variables. This can be obtained by solving the *Adjoint* equations. Inverting M, we get a system of the form

$$\dot{\mathbf{x}} = A\mathbf{x} + \mathbf{B}\mathbf{u}. \tag{5.36}$$

It is possible to design a controller for the system (5.36) using LQR techniques as discussed in Section 5.2. The objective of the problem is to control the system given by (5.36), so that the final value of the state vector is given by

$$\mathbf{x_f} = [\alpha_\mathbf{f} \quad \mathbf{0} \quad \mathbf{h_f} \quad \mathbf{0}]^\mathbf{T} \tag{5.37}$$

If this is rephrased as an optimization problem, the objective would be to minimize the following function:

$$J = \frac{1}{2} \int_0^T \left((\mathbf{x} - \mathbf{x_f})^T Q (\mathbf{x} - \mathbf{x_f}) + \mathbf{u}^T R \mathbf{u} \right) \, dt \tag{5.38}$$

where Q is a positive semi-definite weighting matrix and R is a positive definite matrix. In our case,

$$\begin{aligned} Q &= I, \\ R &= \varepsilon I, \end{aligned}$$

where I is the identity matrix, and ε is a small positive constant. R is required to be positive definite, to ensure that the control computed is not of unreasonable magnitudes.

5.3.6 Backsubstitution of the Control Law into the Nonlinear System

A *Feedback* control gain matrix of the form 5.24 is then derived for the flutter control problem. Now, the aero-structural system is simulated with blowing and suction control applied at the actuator locations. The magnitude of control required at each actuator location is given by the control gain matrix K_{ss}

$$\mathbf{u} = K_{ss}\mathbf{x}. \tag{5.39}$$

It can be seen that this control law was successful in stabilizing the system. The results are presented in the next section.

5.3.7 Results

The following experiments were conducted on a symmetric NACA 0012 airfoil at a freestream Mach number of 0.3. A 160×32 grid was used for the CFD simulation.

The structural properties were chosen as follows: $I_\alpha = 60$, $M = 60$, $K_h = 60$, $K_\alpha = 60$, and $S_\alpha = 30$. Our nominal rest point is $\alpha = 0^\circ$ and $h = 0$.

Adjoint Gradients

As discussed in the last section, the *Adjoint* method is used to find the gradients of lift and moment with respect to the control variables, namely the blowing and suction velocities on the surface. It should be noted that this is done using a steady flow assumption about the nominal rest point of the system. We used a symmetric NACA 0012 section. So for our case, this nominal rest point was at $\alpha = 0$, and $h = 0$. These gradients are shown in Figures 5.2 and 5.3.

Application of Feedback Control to the Nonlinear Flutter Problem

The uncontrolled and controlled aero-structural simulations are represented in Figures 5.4, 5.5, 5.6, and 5.7. It should be noted that even though the *feedback* law is derived from a linearized model of the system, the control is applied to a complete nonlinear model.

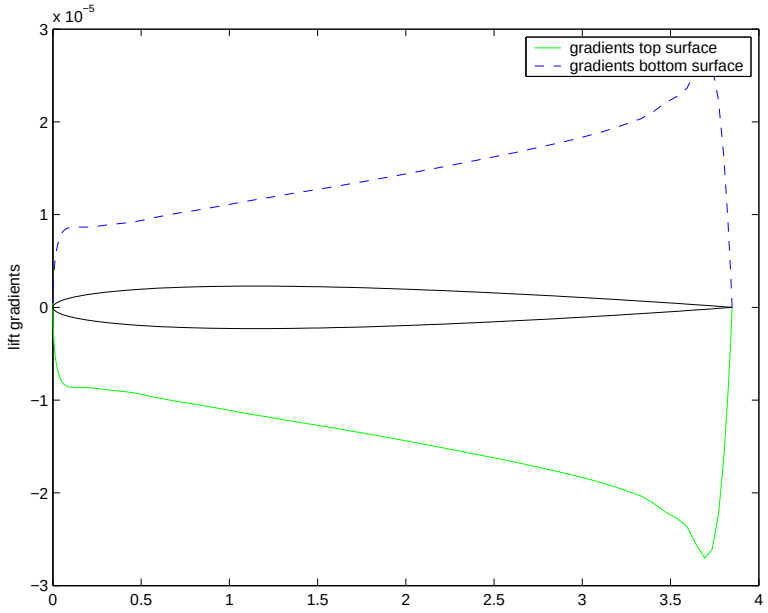

Figure 5.2: Gradient of lift with respect to control mass fluxes

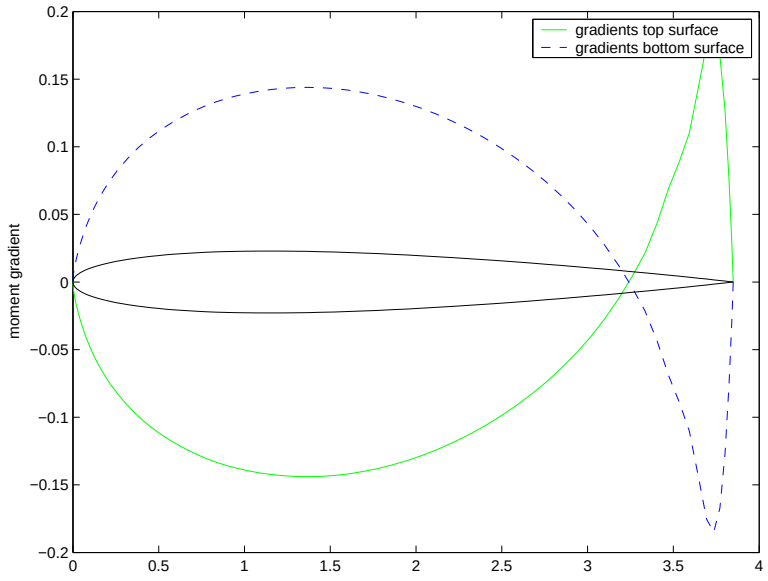

Figure 5.3: Gradient of moment with respect to control mass fluxes

Two different methods are used to find the aerodynamic derivatives. It can be seen that the least-squares method does a better job than the Theodorsen method for flutter control. This is obvious because this represents the nonlinear system more accurately. The

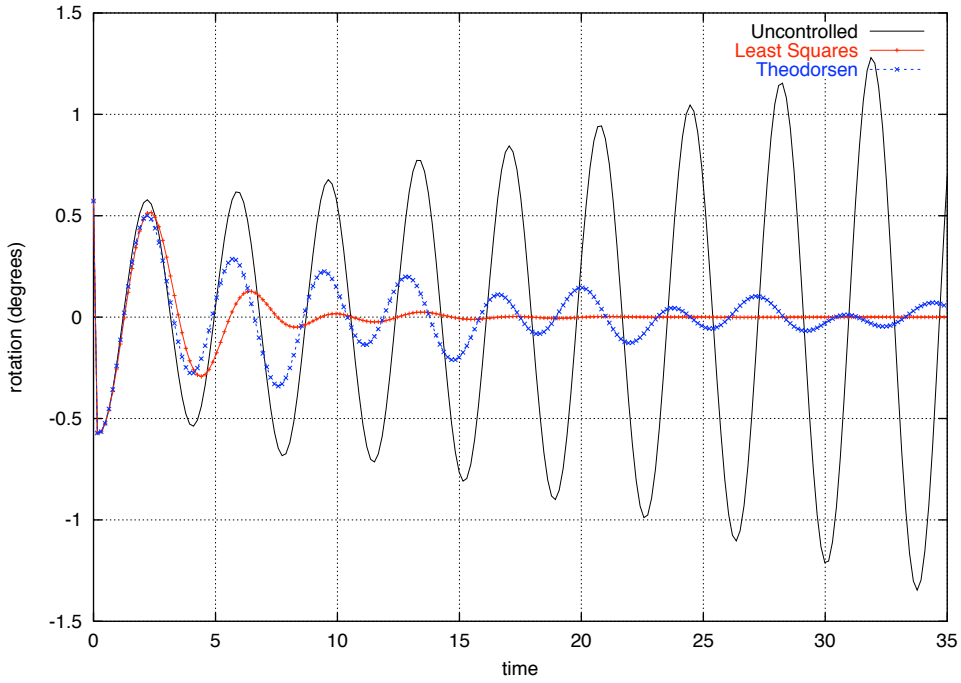

Figure 5.4: Variation of angle of attack (degrees) with time: controlled and uncontrolled cases

corresponding blowing/suction velocities are shown in Figure 5.8. It should be noted that the freestream value of ρq_n in our simulation was 1. So the values of blowing and suction controls required is quite small. Moreover, we need zero control input at the equilibrium point, which is what we desire.

Time step refinement studies

To ensure that the flutter control simulations are correct, the time step for the nonlinear aero-structural solver is made smaller and smaller and the controlled behaviour is observed. It can be seen that the pattern of variation of the angle of attack with time is fairly well predicted by the solver. (See Figure 5.9).

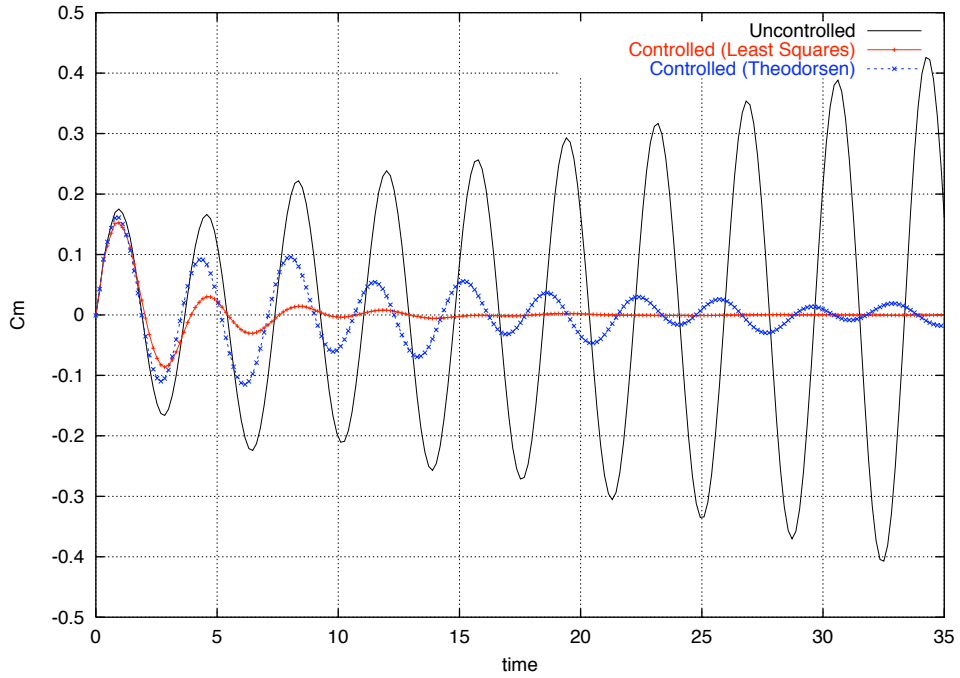

Figure 5.5: Variation of C_m with time: controlled and uncontrolled cases

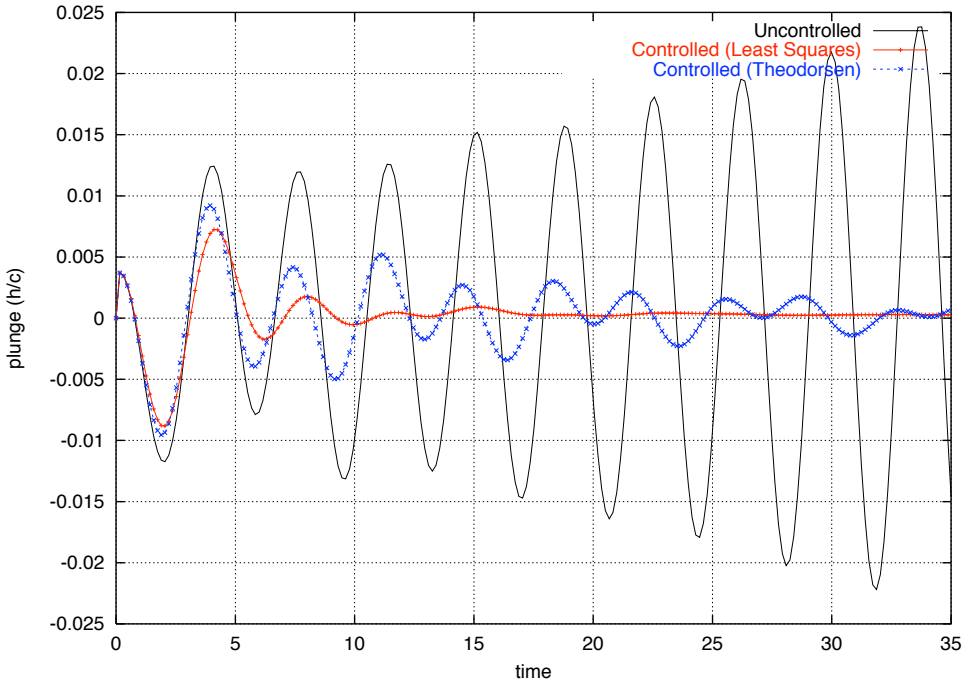

Figure 5.6: Variation of plunge h/c with time: controlled and uncontrolled cases

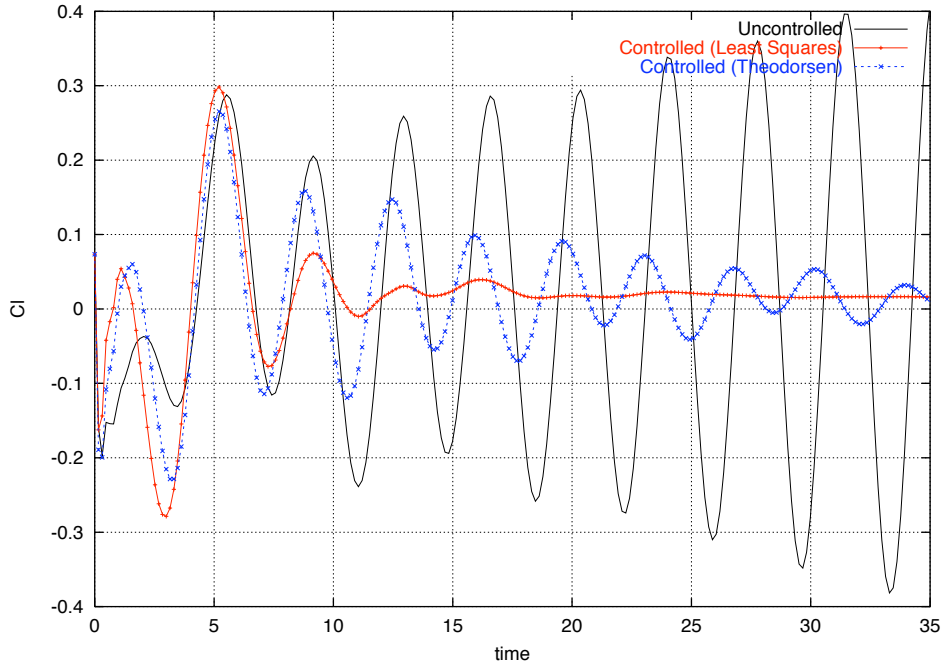

Figure 5.7: Variation of C_l with time: controlled and uncontrolled cases

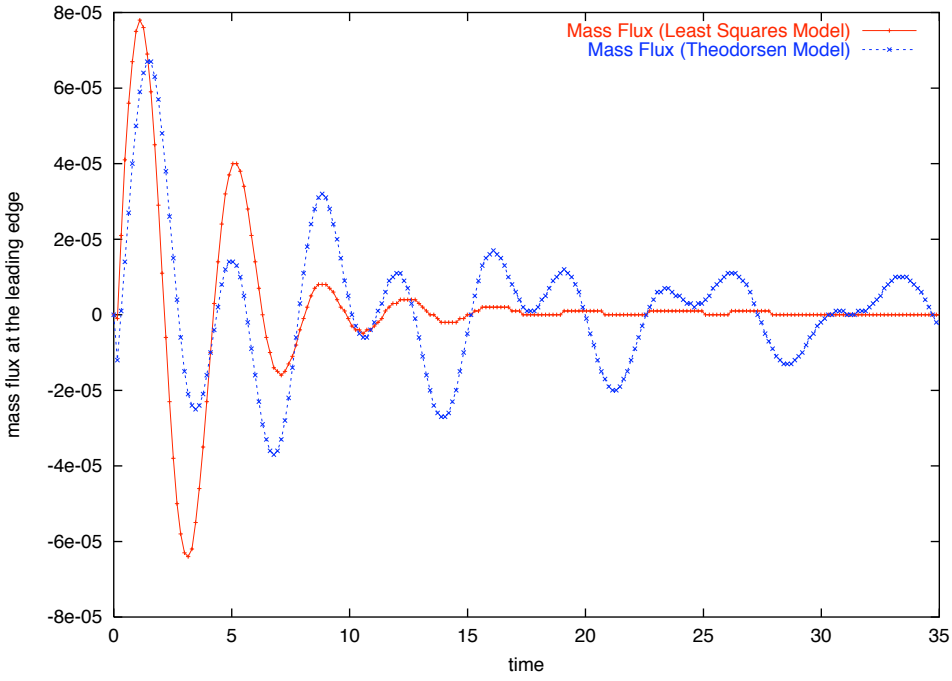

Figure 5.8: Blowing/Suction mass fluxes at the Leading Edge

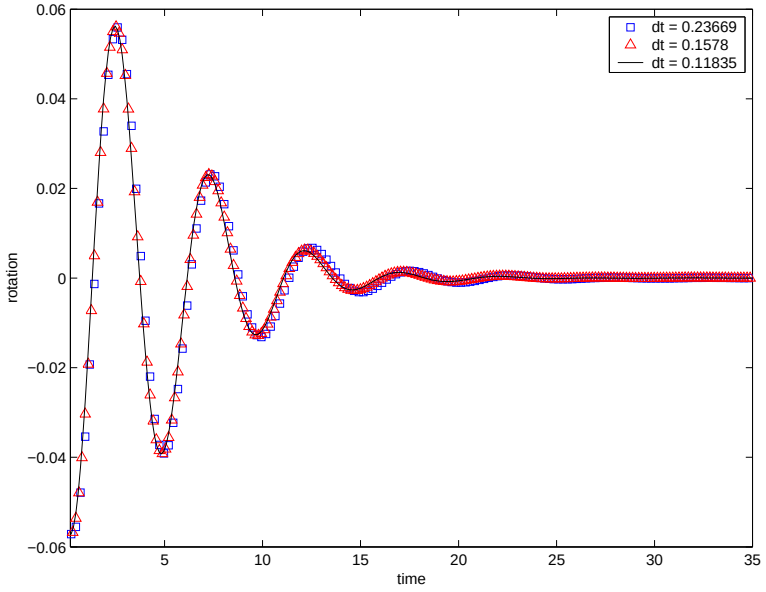

Figure 5.9: Time step refinement studies for the variation of angle of attack with time

5.4 Reduction in the number of Actuators

Our next step is to specialize the control law thus derived to work when the number of actuators is finite. It was found that flutter could be controlled with as few as four actuators: one each in the leading and trailing edges and one each in the middle of the upper and lower surfaces. The fact that there are only four actuation points is represented by zeroing out the gradient shown in Figures 5.2 and 5.2 everywhere except at these four locations. (Every location is represented by a small cluster of CFD cells to prevent numerical instability and damping of the actuation values.)

The entire procedure outlined in the previous section is then repeated to derive the *feedback* gain matrix K_{ss}. It can be seen from Figures 5.10, 5.11, 5.12 and 5.13 that the matrix has non-zero values only at the desired locations of the controllers. Consequently, actuation is performed only at these sites. This is equivalent to controlling the problem with a finite number of actuators.

It can be seen from Figure 5.14 that flutter is controlled successfully even with a finite number of actuators. This is an important result, as it implies that this system can be implemented on a practical aerodynamic configuration.

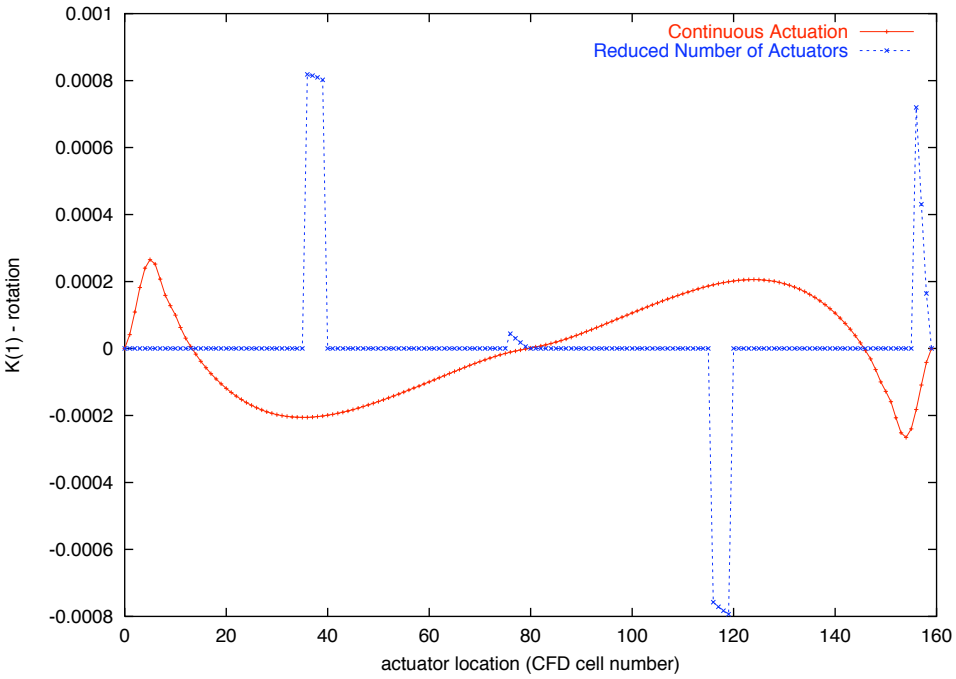

Figure 5.10: Coefficient of rotation angle vs. actuator number in the feedback gain matrix

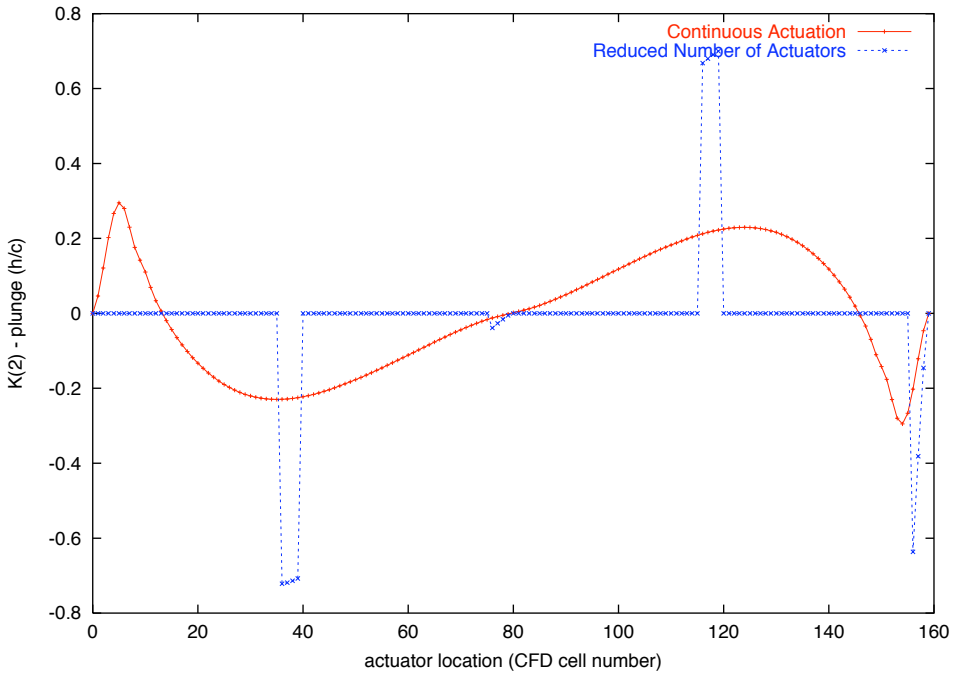

Figure 5.11: Coefficient of plunge vs. actuator number in the feedback gain matrix

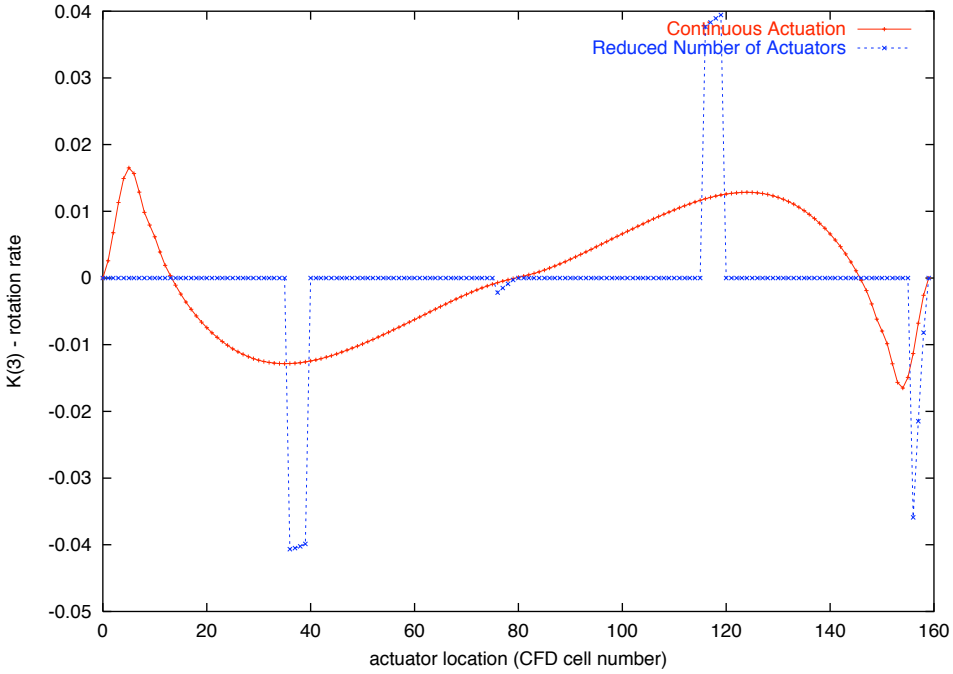

Figure 5.12: Coefficient of rotation angle rate vs. actuator number in the feedback gain matrix

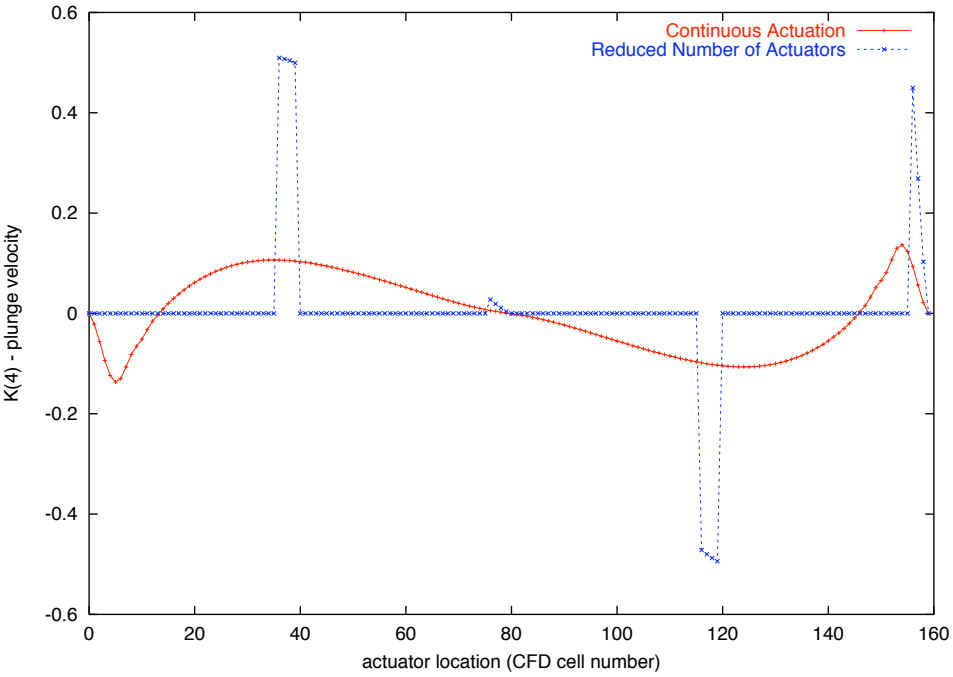

Figure 5.13: Coefficient of plunge velocity vs. actuator number in the feedback gain matrix

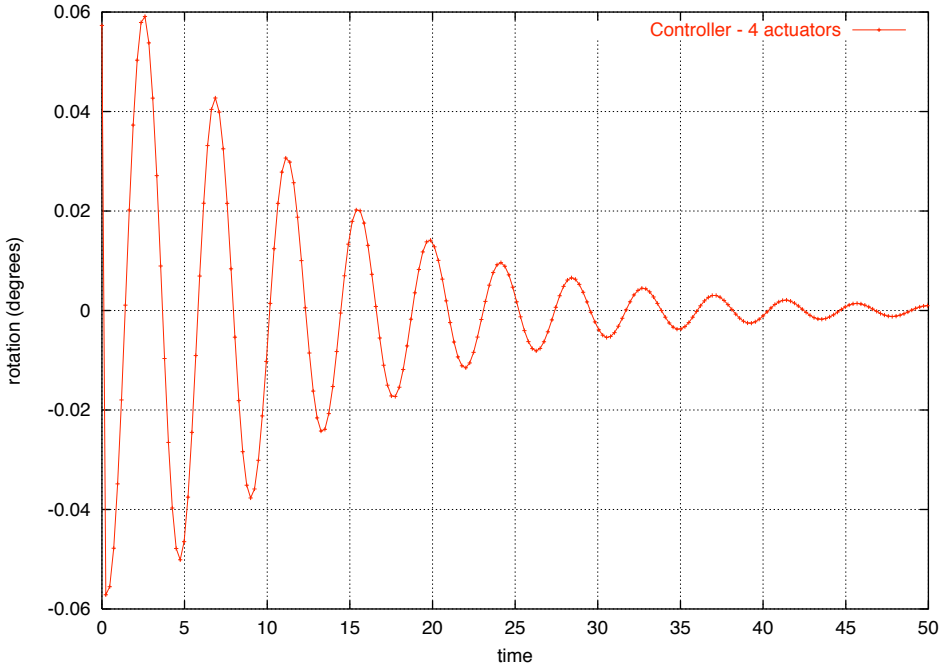

Figure 5.14: Variation of angle of attack (degrees) with time: with 4 actuators

5.5 3-d Results

We then try to control the flutter of a realistic airplane wing. The wing is unswept and the cross-section is that of a 6 percent thick airfoil obtained by scaling down a NACA 0012 airfoil. The semi-span of the wing is 11.5 inches, and the chord is 4.56 inches. This corresponds to an aspect ratio of about 5.

Structurally, the wing is modeled as a plate of thickness 0.065 inches that is placed along the centerline of the wing-section. The density of the material of the wing is 0.003468 slug/sq. inch. The Young's modulus is 9.848×10^6 slug/sq. inch and the torsional rigidity is 3.639×10^6 slug/sq. inch.

The wing was operated under a freestream Mach number of 0.79 and a freestream dynamic pressure of 5241 Pascal.

The aero-structural integration was performed as discussed in Chapter 2. The structure is modeled using 50 plate elements. The aerodynamic simulation is done on a $96 \times 32 \times 48$ grid. It can be seen from Figure 5.15 that the uncontrolled system diverges fairly rapidly. In the time frame considered, the plunge diverges from a negligible amount to 10 percent

chord very quickly.

Our task, now, is to design a controller using the techniques described in the previous sections. It has been shown that the flutter of a wing can be studied by studying the dynamics of a section three quarters of the distance from the wing center-line to the tip. We identify the structural properties of the section located at this point, and model it using the typical section wing model, discussed previously. Following the techniques in the previous section, we quickly derive the *feedback* gain matrix K_{ss} for this section.

We make the assumption that this matrix is valid throughout the wing. This is a valid assumption, as the control is implemented in a *feedback* fashion. The tip is expected to go through the maximum deflection, and therefore will be subject to the maximum amount of control. (Since the control is proportional in nature). The root does not move at all, and thus there is no control applied at the root. The results of this simulation are shown in Figure 5.15. It can be seen that the control law thus derived is successful in controlling flutter. The mass fluxes at an actuator location at the tip, along the trailing edge are shown in Figure 5.16. Again, it can be seen that the mass fluxes required for control, when compared to the freestream mass flux of $\rho q_n = 1$ are very small.

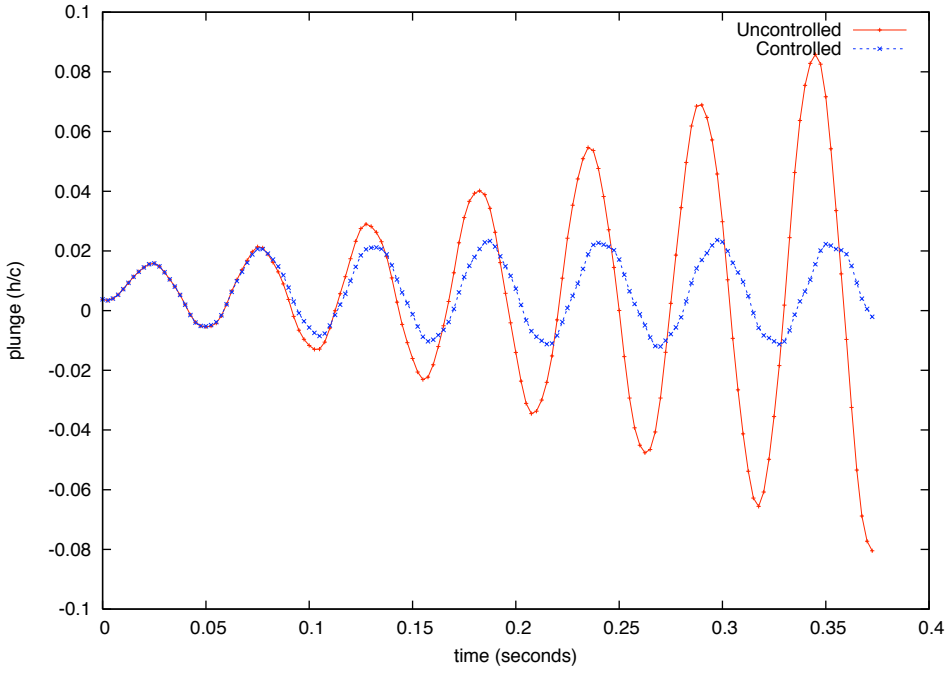

Figure 5.15: Variation of plunge h/c with time: controlled and uncontrolled cases

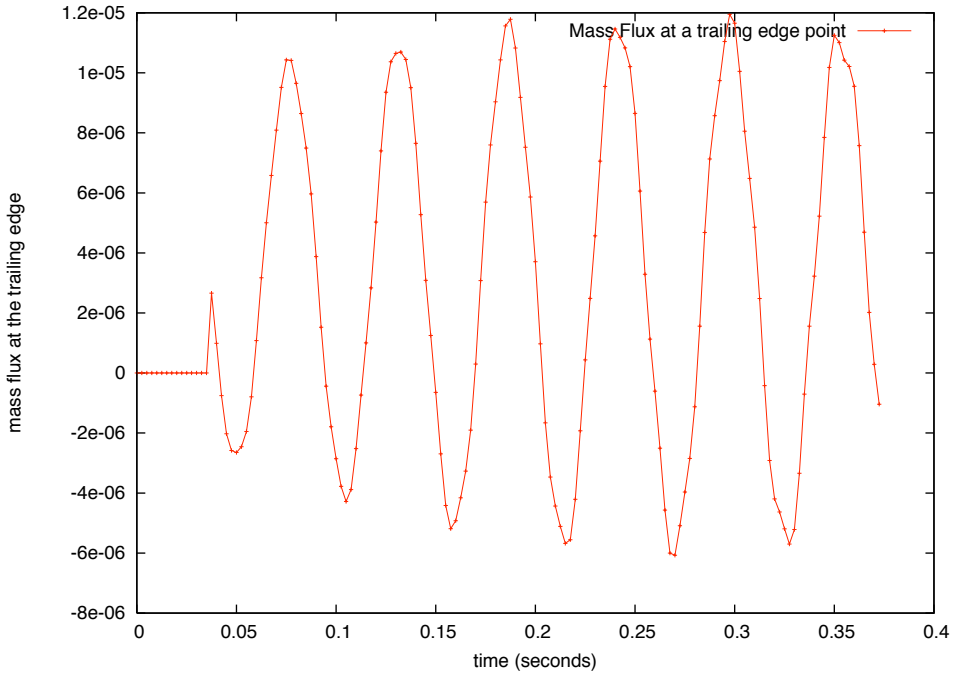

Figure 5.16: Blowing/Suction mass fluxes at a trailing edge point

Figures 5.17 and 5.18 show results from the complete 3-d simulation at identical times for the uncontrolled and controlled cases respectively. It can be seen, especially from the last pictures in both sequences that the deflections in the controlled case are smaller than those in the uncontrolled case. In fact, in the controlled case the wing settles into a limit cycle oscillation of small magnitude as can be seen from Figure 5.15. This is in spite of the fact that an approximate structural model was used in the calculation of the control law.

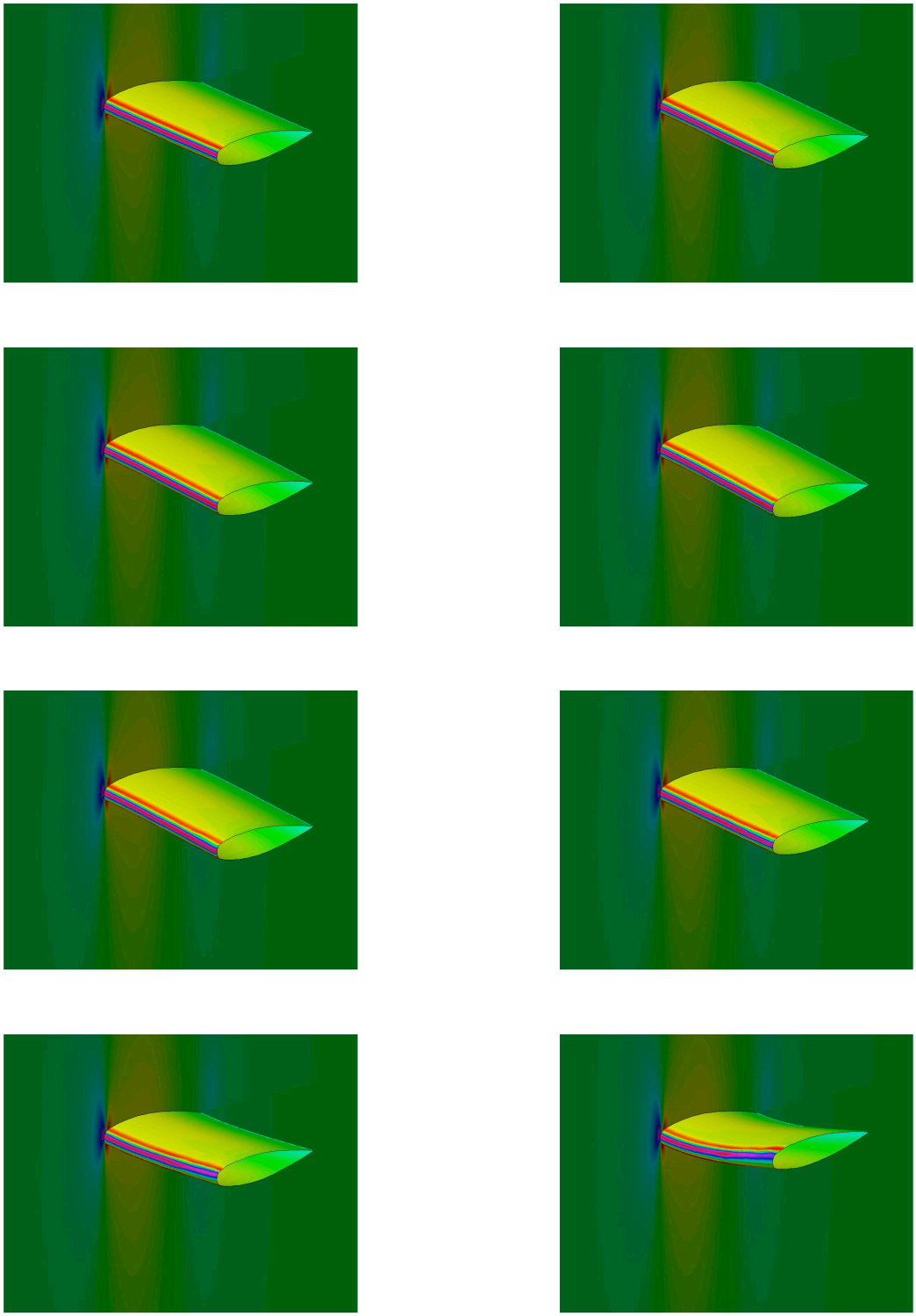

Figure 5.17: Uncontrolled simulation (Plunge variation at the tip shown in Figure 5.15)

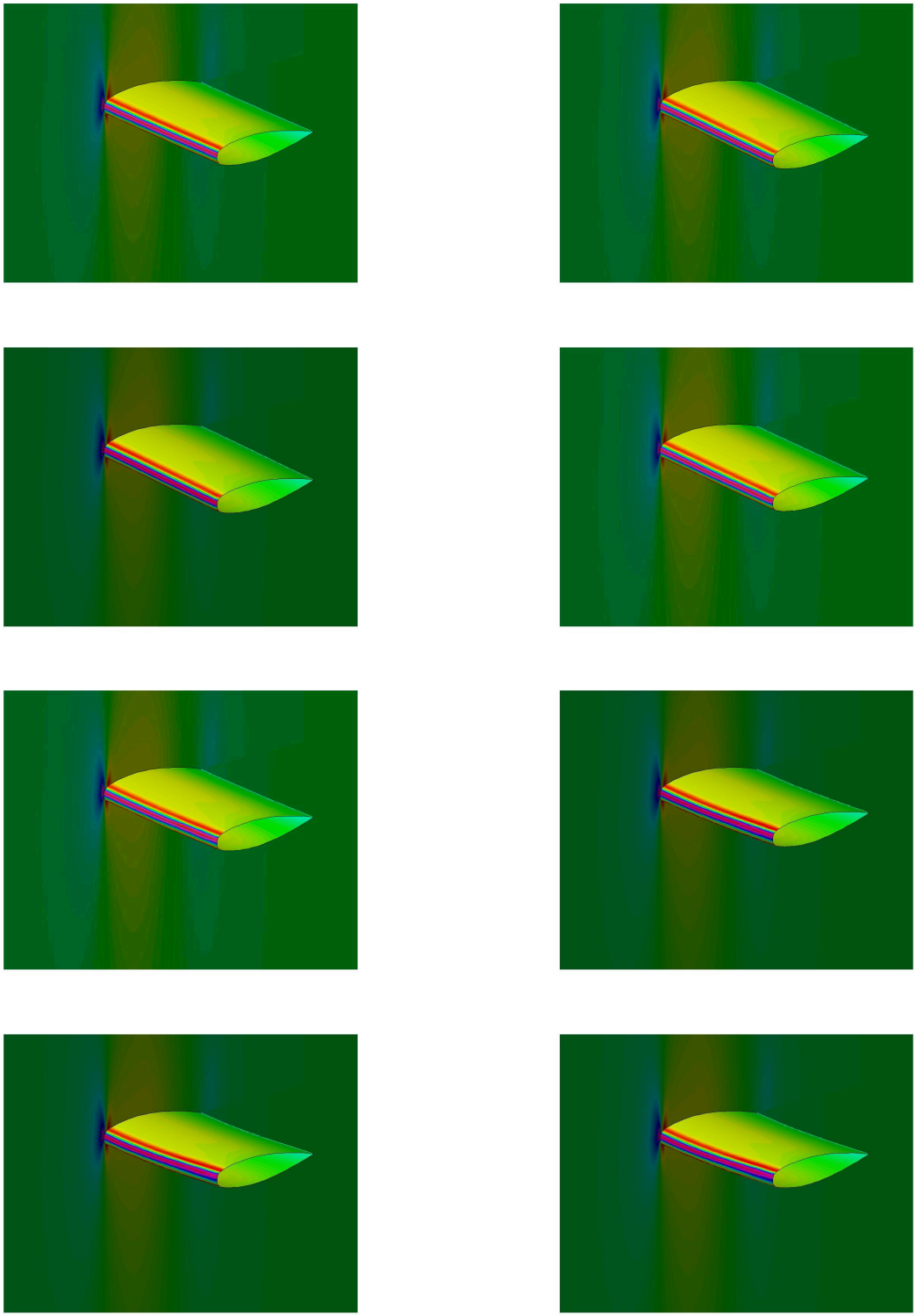

Figure 5.18: Controlled simulation (Plunge variation at the tip shown in Figure 5.15)

Chapter 6

Conclusions and Future Work

The primary objective of this thesis was to develop a computational framework for designing control laws for flow control problems. A strong emphasis was placed on deriving a generic control law. This affords extensions to multiple specific cases with very little effort.

The control law thus derived was specialized for the case of *Virtual Aerodynamic Shaping*. Here the surface controls on the surface of an airfoil/wing were adjusted so that the surface pressure distribution resembled that of another airfoil/wing. In doing so, we obtained the following result for the *Adjoint* gradient

$$\delta I = -\int_{\mathcal{B}_\xi} \left(\psi_1 + \psi_2 u + \psi_3 v + \psi_4 w + \psi_5 \left(E + \frac{P}{\rho} \right) \right) \delta \rho q_n d\mathcal{B}_\xi \ . \tag{6.1}$$

The *Adjoint* boundary condition is given by

$$(P - P_d) = \psi_2 S_{21} + \psi_3 S_{22} + \psi_4 S_{23}. \tag{6.2}$$

Here P is the current surface pressure, and P_d is the desired (target) surface pressure. The variables ψ represent the *Adjoint* co-state variables. As can be seen from Equation 6.2, the *Adjoint* boundary condition is directly proportional to the difference between the current and the desired pressures, and becomes zero when the pressures become identical. This implies that the control law thus derived is *feedback* in nature.

Having established the framework for developing control laws, we try to develop techniques for flutter control. Flutter is inherently an aero-structural instability. This can be controlled by structural and aerodynamic damping.

We first demonstrated the fact that the aero-structural behavior of a system can be affected passively by structural optimization. We studied the case of enhancement of LCOs in a panel flutter problem. This is because of its immediate relevance to the design of micro air vehicles with flapping wings. However, the techniques developed are equally valid for LCO amplitude reduction.

Finally, we developed a *feedback* algorithm for the control of flutter. We demonstrated the effectiveness of this control law in 2-d and 3-d simulations. We also explored possibilities for the reduction in the number of actuators.

All the simulations in this thesis were done using an inviscid model. The next obvious step is to implement the control laws derived in this thesis in a fully viscous simulation. This will allow us to test the validity of our current simulations. It will also allow us to study the class of control problems where viscosity plays an important role: separation, transition, buffeting, etc.

Finally, in order for a *feedback* control system to be practical, it should be capable of generating control input in real time. To do this, it becomes necessary to work with reduced order models. The main problem here is that almost all reduced order models don't model the actuator dynamics sufficiently well. If this problem can be solved, online control based on a reduced order model will be a realistic possibility.

Part I

Appendix

Appendix A

Computational Algorithms

Most of the flow situations discussed in this thesis can be adequately represented by an inviscid flow model. The *Euler* equations (2.1), derived in the previous section are reproduced below for reference

$$\frac{\partial \mathbf{w}}{\partial t} + \frac{\partial \mathbf{f_j}}{\partial x_j} = 0. \tag{A.1}$$

These are highly unsteady, nonlinear equations, and in most cases a closed form solution cannot be obtained. It, therefore, becomes necessary to resort to a computational simulation to study the features and the dynamics of the flow. Depending on the flow conditions, we might expect the presence of shocks and other such contact discontinuities.

A.1 Numerical Solution of Partial Differential Equations

The numerical integration of flows with singularities is tricky and special algorithms have to be used. Another important factor that needs to be considered in the large scale simulations of unsteady flows is speed of solution. Special techniques are used for solution convergence acceleration. The different stages involved in a numerical solution of the unsteady *Euler* equations are shown in Figure A.1. These algorithms are discussed in this Appendix Section.

A.1.1 Convergence

Theorem (Lax): *If a numerical scheme is consistent and stable, then it is convergent.*

This was proved by Lax in [45]. **Consistency** refers to the fact that the leading order

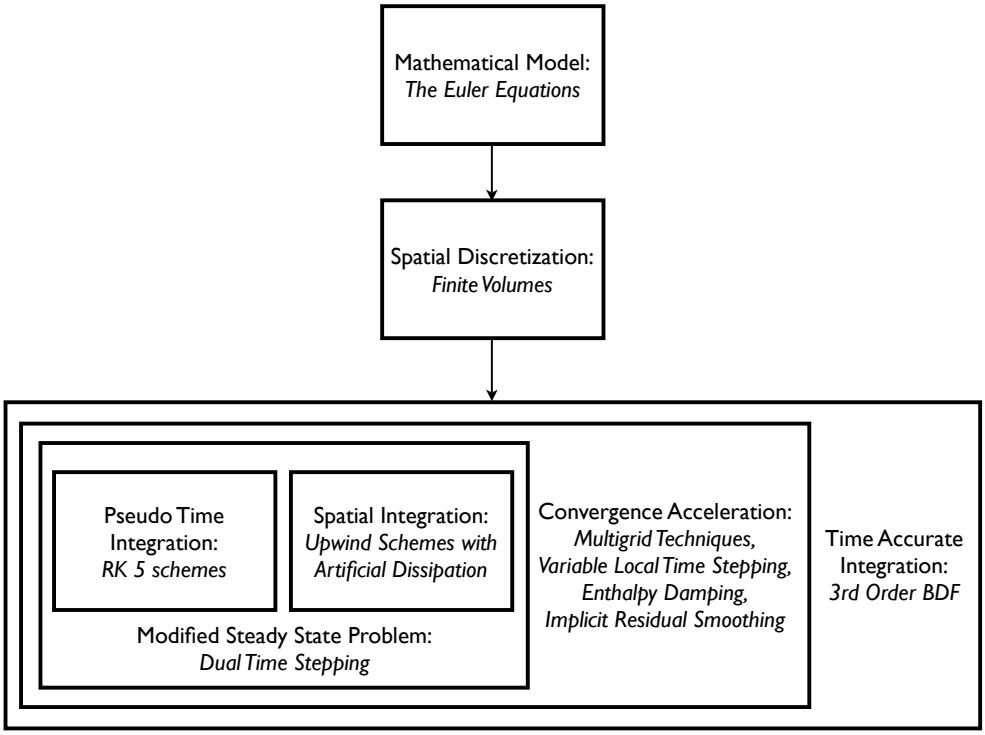

Figure A.1: Numerical Simulation of the Euler Equations

terms of a Taylor Series discretization of the difference scheme reduce to the differential equation whose solution is required. **Stability** of the scheme refers to the property that any identifiable error term in the difference equation does not grow without bounds. For more detail, the reader is referred to the textbook by Isaacson and Keller [29].

A.2 Algorithms for the Solution of Steady Flows with Singularities: Numerical Discretization

The solution of transonic flow past airplanes has been the main driver for the development of algorithms for CFD over the last twenty years. This is because of the presence of discontinuities in the solutions (like shocks). Unless adequate care is taken in formulating a numerical scheme, the numerical solution develops oscillations that could lead to instability. Numerical schemes developed for the solution of transonic flow depend on Total Variation Diminishing (TVD) or the Local Extremum Diminishing (LED) principles. Algorithms based on variants of these principles were developed by Godunov [27], Harten [28], Van Leer [46], Osher and Chakravarthy [58] and Roe [70]. In this thesis, we use the JST and CUSP schemes developed by Jameson [34, 35].

A.2.1 Stability of a Numerical Scheme: The LED principle

Consider a one dimensional conservation law of the form

$$\frac{\partial v}{\partial t} + \frac{\partial}{\partial x} f(v) = 0. \tag{A.2}$$

This could be written in semi-discrete form as

$$\frac{\partial v_j}{\partial t} = \sum_k c_{jk} v_k, \tag{A.3}$$

where v_k are the points around v_j which contribute to the difference scheme. v_k can be expanded in a Taylor Series about v_j. If this is done for all v_j, then on comparison with Equation (A.3), it can be seen that

$$\sum_k c_{jk} = 0. \tag{A.4}$$

It should be noted that this is true only if there is no source term. Now, without loss of generality, Equation (A.3) can be rewritten as

$$\frac{\partial v_j}{\partial t} = \sum_k c_{jk}(v_k - v_j). \tag{A.5}$$

Equation (A.5) presents a very important result. If the co-efficients c_{jk} were non-negative, then it can be seen that if v_j is a minimum, then it always remains a minimum, and if v_j is a maximum then it always remains a maximum. Moreover, if $c_{jk} = 0$ if j and k are not nearest neighbors, then a local minimum can never increase and a local maximum can never decrease. This property is known as the *Local Extremum Diminishing* property, and any scheme satisfying the LED property will be bounded and stable.

A.2.2 Finite Volume Discretization of the Governing Equations

We use a cell-averaged finite volume scheme for all the solutions in this thesis. The domain of interest is divided into a large number of finite-volumes. The *Euler* equations can be integrated over any one of these finite volumes as follows

$$\frac{\partial}{\partial t}\int \mathbf{w}\, dV + \int \nabla.\mathbf{f}\, dV = 0. \tag{A.6}$$

Using the *Gauss Divergence Theorem*, the above equation can be rewritten as

$$\frac{\partial \mathbf{w}V}{\partial t} + \int \mathbf{f}.\mathbf{n}\, dS = 0, \tag{A.7}$$

where $\mathbf{w}$ now represents the cell-average value. If the cells are polyhedral in shape, (A.7) can be approximated by

$$\frac{\partial \mathbf{w}V}{\partial t} + \sum_{i=1}^{n} \mathbf{f_i}.\mathbf{S_i} = 0, \tag{A.8}$$

where $\mathbf{S_i}$ is the area and $\mathbf{f_i}$ is the average flux through the the i^{th} face. The areas $\mathbf{S_i}$ and volumes V are functions of the spatial discretization chosen. It is common to use both structured and unstructured grids for CFD purposes. All calculations in this thesis were performed on structured grids. The state vector $\mathbf{w}$ is a cell average value, and consequently is assumed constant within the cell. The fluxes at the surface should be evaluated with adequate care to prevent undesirable oscillations, which might lead to instability. This will be discussed in Section A.2.3.

A.2.3 Upwind Schemes with Artificial Dissipation

Consider a finite volume discretization of Equation (A.2), written in the j^{th} cell:

$$\Delta x \frac{\partial v}{\partial t} + h_{j+\frac{1}{2}} - h_{j-\frac{1}{2}} = 0. \tag{A.9}$$

Here $h_{j+\frac{1}{2}}$ is the flux between cells j and $j+1$. A simple estimate for the flux $h_{j+\frac{1}{2}}$ is

$$h_{j+\frac{1}{2}} = \frac{1}{2}\left(f_j + f_{j+1}\right) - \alpha_{j+\frac{1}{2}}\left(v_{j+1} - v_j\right) \tag{A.10}$$

The second term is a dissipative term, which stabilizes the numerical scheme. We rewrite equation (A.10) as

$$\begin{aligned} h_{j+\frac{1}{2}} &= f_j + \frac{1}{2}\left(f_{j+1} - f_j\right) - \alpha_{j+\frac{1}{2}}\left(v_{j+1} - v_j\right) \\ &= f_j + \left(\frac{1}{2}a_{j+\frac{1}{2}} - \alpha_{j+\frac{1}{2}}\right)\left(v_{j+1} - v_j\right), \end{aligned} \tag{A.11}$$

where $a_{j+\frac{1}{2}}$ is a numerical estimate of the wave speed. Similarly,

$$h_{j-\frac{1}{2}} = f_j + \left(\frac{1}{2}a_{j-\frac{1}{2}} + \alpha_{j-\frac{1}{2}}\right)\left(v_{j-1} - v_j\right), \tag{A.12}$$

Now,

$$h_{j+\frac{1}{2}} - h_{j-\frac{1}{2}} = \left(\frac{1}{2}a_{j+\frac{1}{2}} - \alpha_{j+\frac{1}{2}}\right)\left(v_{j+1} - v_j\right) - \left(\frac{1}{2}a_{j-\frac{1}{2}} + \alpha_{j-\frac{1}{2}}\right)\left(v_{j-1} - v_j\right) \tag{A.13}$$

The positivity condition (A.5), that ensures the stability of the above scheme prescribes that

$$\alpha_{j+\frac{1}{2}} > \frac{1}{2}\left|a_{j+\frac{1}{2}}\right|, \tag{A.14}$$

for stability. The diffusive flux is then given by

$$d_{j+\frac{1}{2}} = \frac{1}{2}\left|a_{j+\frac{1}{2}}\right|\Delta v_{j+\frac{1}{2}}, \tag{A.15}$$

where

$$\Delta v_{j+\frac{1}{2}} = v_{j+1} - v_j. \tag{A.16}$$

The disadvantage of using a dissipative flux as in Equation is that it is extremely dissipative, and so reduces the order of accuracy also in regions where stability issues are not expected.

A.2.4 The JST scheme

The JST scheme [41], overcomes the difficulty encountered above, by adding anti-diffusive terms in smooth regions. The JST flux is of the form

$$d_{j+\frac{1}{2}} = \epsilon^{(2)}\Delta v_{j+\frac{1}{2}} - \epsilon^{(4)}\left(\Delta v_{j+\frac{3}{2}} - 2\Delta v_{j+\frac{1}{2}} + \Delta v_{j-\frac{1}{2}}\right), \tag{A.17}$$

where the second term approximates a fourth order derivative. The co-efficients $\epsilon^{(2)}$ and $\epsilon^{(4)}$ are chosen such that the diffusion levels in smooth regions is minimal, and the combination of diffusive terms ensures that positivity is satisfied in the regions where extrema are present.

A.2.5 Integration in pseudo-time: Runge-Kutta Methods

Having successfully represented the spatial derivatives, Equation (A.8) can now be written as the following system of ODEs.

$$\frac{\partial \mathbf{w}}{\partial t} + R(\mathbf{w}) = 0. \tag{A.18}$$

Our task is now to integrate Equation (A.18) to steady state. Here again we have a choice between several different methods. We choose an explicit method based on a Runge-Kutta scheme [32]. When a Fourier mode we^{ipt} is inserted into the one-dimensional form of the model problem above (A.18), the resulting Fourier symbol has an imaginary part proportional to the wave speed and a negative real part proportional to the diffusion. Thus the integration scheme chosen should have a stability region that covers substantial portions along the real and imaginary axes of the stability diagram. To achieve this, it is convenient to split the residual into a convective part and a dissipative part

$$R(\mathbf{w}) = Q(\mathbf{w}) + D(\mathbf{w}), \tag{A.19}$$

where $Q(\mathbf{w})$ is the convective part and $D\mathbf{w})$ is the dissipative part. The Runge-Kutta method used can then be written as a multistage method of the form:

$$\mathbf{w}^{(n+1,0)} \quad = \quad \mathbf{w}^{(n)},$$

$$\begin{aligned} & \cdots \\ \mathbf{w}^{(n+1,k)} &= \mathbf{w}^{(n)} - \alpha_k \Delta t \left[\mathbf{Q}^{(k-1)} + \mathbf{D}^{(k-1)} \right], \\ & \cdots \\ \mathbf{w}^{(n+1)} &= \mathbf{w}^{(n+1,m)}, \end{aligned} \tag{A.20}$$

where $\alpha_m = 1$, and

$$\begin{aligned} \mathbf{Q}^{(k)} &= \mathbf{Q}\left[\mathbf{w}^{(n+1,k)}\right], \\ \mathbf{D}^{(k)} &= \beta^{(k)} \mathbf{D}\left[\mathbf{w}^{(n+1,k)}\right] + \left(1 - \beta^{(k)}\right) \mathbf{D}^{(k-1)}. \end{aligned} \tag{A.21}$$

The coefficients α_k and β_k are chosen so as to maximize the stability region of the integration scheme. In this thesis, a five stage scheme with three evaluations of the dissipative terms is adopted:

$$\begin{array}{lllllll} \alpha_1 &=& 1/4 & \qquad \beta_1 &=& 1 \\ \alpha_2 &=& 1/6 & \qquad \beta_2 &=& 0 \\ \alpha_3 &=& 3/8 & \qquad \beta_3 &=& .56 \\ \alpha_4 &=& 1/2 & \qquad \beta_4 &=& 0 \\ \alpha_5 &=& 1 & \qquad \beta_5 &=& .44 \end{array}$$

A.3 Acceleration of Convergence to Steady State

A.3.1 Multigrid Acceleration

The use of multigrid techniques for convergence acceleration was first demonstrated by Federenko. [20]. There is now a huge body of knowledge devoted to the study of multigrid techniques. Their successful application to the solution of elliptical partial differential equations has been proven and well documented. Here, the multigrid operator is treated as a smoothing operator on each grid. This theory, however, has not been proved for hyperbolic systems. Still, it has been experimentally verified that multigrid techniques work quite well for hyperbolic problems. This is because, on coarse grids bigger time steps can be used which help to expel disturbances away rapidly. The multigrid scheme used in this thesis was developed for the *Euler* and *Navier-Stokes* equations by Jameson *et al.* [37]. This method uses a sequence of coarser meshes that are generated by eliminating alternate points along each coordinate direction. The grid level is indicated by the subscript k, with higher values

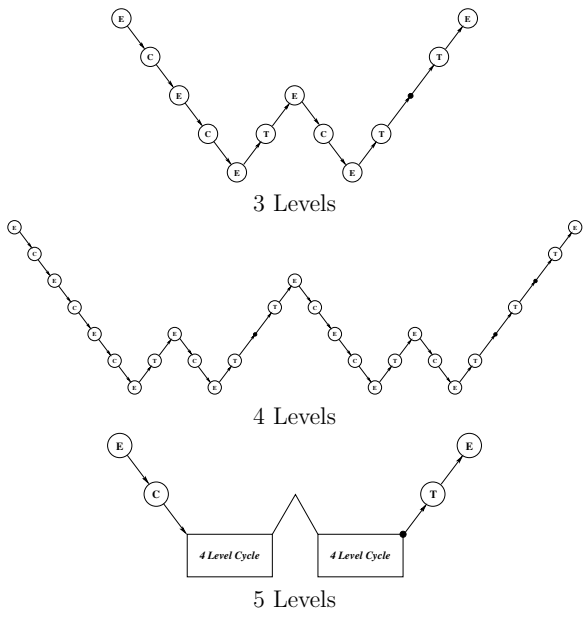

Figure A.2: Multigrid algorithm for flow calculations; E: evaluate the change in flow for one time step; C: Collect the solution and; T: Transfer the data without updating the solution

of k representing successfully coarser meshes. To make mathematical representation of the multigrid scheme easier, it is useful to define the collection and interpolation operators. When moving from a fine grid to a coarser grid, the solution vector on grid k is represented as

$$\mathbf{w}_k^{(0)} = T_{k,k-1}\mathbf{w}_{k-1} \tag{A.22}$$

Conversely, when moving from a coarser grid to a finer grid, the accumulated correction on grid k is transferred back using the interpolation operator $I_{k-1,k}$. At grid level k, a residual forcing function is defined such that the multistage scheme on grid k is driven by the residuals calculated on grid $k-1$. This is of the form

$$P_k = Q_{k,k-1}R_{k-1}(\mathbf{w}_{k-1}) - R_k\left[\mathbf{w}_k^{(0)}\right], \tag{A.23}$$

where $Q_{k,k-1}$ is another transfer operator. In the multistage Runge-Kutta scheme described in the previous section, $R_k(\mathbf{w}_k)$ is then replaced by $R_k(\mathbf{w}_k) + P_k$.

$$\mathbf{w}_k^{(0)} \quad = \quad T_{k,k-1}\mathbf{w}_{k-1},$$

$$
\begin{aligned}
&\cdots \\
\mathbf{w}_k^{(q)} &= \mathbf{w}_k^{(0)} - \alpha_q \Delta t_q \left[R_k^{(q)} + P_k \right], \\
&\cdots \\
\mathbf{w}_{k+1}^{(0)} &= T_{k+1,k} \mathbf{w}_k^{(m)},
\end{aligned} \tag{A.24}
$$

and so on till the coarsest mesh is reached. At that point, the corrections are transferred back to the finer meshes using the respective interpolation operators. It has been shown that the above scheme works well when used in a W-cycle as shown in Figure A.2. Finally, it is important to understand the extent of acceleration obtained by using the multigrid method. In the case of a three dimensional grid, the number of cells in each coarser grid is reduced by a factor of eight. Assuming the work done per time step on the finest grid is 1, the total work done is then of the order of

$$1 + 2/8 + 4/64 + \ldots < 4/3. \tag{A.25}$$

Thus it can be seen that the large improvement in convergence acceleration that is obtained as a result of using a multigrid algorithm has a computational cost that is only slightly more than a calculation on the finest grid.

A.3.2 Variable local time stepping

The most intuitive approach to the solution of a collection of ordinary differential equations obtained as a result of discretization of a partial differential equation, is to advance the solution in each cell by the same time step. This ensures that the solutions at all domain points are at the same time level at all times. The time step chosen in such a scheme would be the largest time step that will not cause instability in any one of the cells. This translates to advancing the solution over the entire domain by a time step that might be significantly lower than that permitted by the *Courant-Friedrichs-Lewy* condition at any cell volume.

If, only the steady state solution is desired, it might not be very important to capture the transients accurately. Also, it becomes unnecessary to propagate the solution forward in time such that all the volumes are always at the same time level.

In calculations performed in this thesis a variable local time step is used in each cell, such that the solution in each cell can be advanced at the fastest rate possible for that cell without being constrained by limits imposed at other parts of the domain.

A.3.3 Implicit Residual Smoothing

The use of an explicit scheme to integrate the *Euler* equations, imposes severe restrictions on the maximum allowable time step during the integration process. This indirectly affects the speed of convergence. It has been shown by Jameson and Baker [38] that using a smoothed residual instead of the calculated residual can help increase the maximum allowable CFL number. This is somewhat similar to the Pade class of schemes, and achieves the objective of increasing the maximum allowable CFL number by increasing the stencil to include all the cells in the domain. The residual smoothing operation is of the form

$$(1 - \epsilon_i \delta_{xx})\,(1 - \epsilon_j \delta_{yy})\,(1 - \epsilon_k \delta_{zz})\,\bar{R}_{i,j,k} = R_{i,j,k}, \tag{A.26}$$

where $\bar{R}_{i,j,k}$ is the smoothed residual, obtained by using a tri-diagonal solver in each coordinate direction. The values of the smoothing coefficients ϵ_i, ϵ_j and ϵ_k can be chosen to control the level of smoothing.

A.4 Dual Time Stepping

The dual time stepping algorithm for solving an unsteady flow physics problem was first discussed by Jameson [32]. Equation (A.8) can be written down as follows:

$$\frac{\partial \mathbf{w} V}{\partial t} + R(\mathbf{w}) = 0. \tag{A.27}$$

When (A.27) is discretized in time, we get the difference equation given by

$$D_t(\mathbf{w}^{(n+1)} V^{(n+1)}) + R(\mathbf{w}^{(\mathbf{n+1})}) = 0, \tag{A.28}$$

where D_t is a k^{th} order accurate backward difference operator. Calculations performed in this thesis were done using a third order backward difference formula. Consider the following second order backward difference formula

$$\frac{3}{2\Delta t}(\mathbf{w}^{(n+1)} V^{(n+1)}) - \frac{2}{\Delta t}(\mathbf{w}^{(n)} V^{(n)}) + \frac{1}{2\Delta t}(\mathbf{w}^{(n-1)} V^{(n-1)}) + R(\mathbf{w}^{(\mathbf{n+1})}) = 0. \tag{A.29}$$

The solution to (A.29) is obtained as the steady state solution to a problem of the form

$$\frac{\partial \mathbf{w}}{\partial t^*} + R^*(\mathbf{w}) = 0, \tag{A.30}$$

where $R^*(\mathbf{w})$ is a modified residual with source terms.

$$R^*(\mathbf{w}) = \frac{3}{2\Delta t}(\mathbf{w}^{(n+1)}V^{(n+1)}) - \frac{2}{\Delta t}(\mathbf{w}^{(n)}V^{(n)}) + \frac{1}{2\Delta t}(\mathbf{w}^{(n-1)}V^{(n-1)}) + R(\mathbf{w}^{(\mathbf{n+1})}). \tag{A.31}$$

Equation (A.30) can then be solved using a multigrid based algorithm for the solution of steady flows as discussed in the previous sections.

Appendix B

The Adjoint Equations for Fluid Flow Control

The theory of Optimal Control [15] involves casting the control problem as an optimization problem, and then solving it using methods from the Calculus of Variations. Pironneau [64] used the theory of Optimal Control to solve Fluid Control problems.

Techniques for Aerodynamic Shape Optimization based on Optimal Control Techniques have been well developed by Jameson and his colleagues [33, 39, 54, 47, 75] over the last few years. The concept of flow control discussed in this thesis follows closely along these lines.

We assume that we are trying to minimize a cost function of the form

$$I = \int_{\mathcal{B}_\xi} \mathcal{M}(\mathbf{w}, \rho q_n)\, d\mathcal{B}_\xi \,, \tag{B.1}$$

where ρq_n is the normal mass flux at the surface. Our task is to find the value of the normal mass flux at the surface that minimizes the above cost function (B.1), such that $\mathbf{w}$ and ρq_n satisfy the steady *Euler* equations (2.1).

The steady state *Euler* equations can be expressed in conservative form as follows

$$\int_{\mathcal{B}} n_i \phi^{\mathbf{T}} \mathbf{f_i}(\mathbf{w})\, d\mathcal{B} = \int_{\mathcal{D}} \frac{\partial \phi^{\mathbf{T}}}{\partial \mathbf{x_i}} \mathbf{f_i}(\mathbf{w})\, d\mathcal{D} \,, \tag{B.2}$$

where ϕ is any test function. If a transformation is made from physical space x_i to computational space ξ_i , defined by the mapping functions

$$K_{ij} = \left[\frac{\partial x_i}{\partial \xi_j}\right] \ , \quad J = \det(K) \ , \quad K^{-1}_{ij} = \left[\frac{\partial \xi_i}{\partial x_j}\right] \ , \tag{B.3}$$

and

$$S = JK^{-1} \ , \tag{B.4}$$

the Euler Equations (B.2) become

$$\int_{\mathcal{B}_\xi} n_i \phi^{\mathbf{T}} \mathbf{S_{ij}} \mathbf{f_j}(\mathbf{w}) \, d\mathcal{B}_\xi = \int_{\mathcal{D}_\xi} \frac{\partial \phi^{\mathbf{T}}}{\partial \xi_{\mathbf{i}}} \mathbf{S_{ij}} \mathbf{f_j}(\mathbf{w}) \, d\mathcal{D}_\xi \ . \tag{B.5}$$

We can choose our computational coordinate system so that $\xi_1 = \xi_3 = 0$ at the physical boundary. The boundary conditions for the case where we have blowing or suction at the boundary can then be prescribed in terms of the normal mass flux as follows

$$F_2 = \begin{bmatrix} \rho q_n \\ \rho q_n u + S_{21} P \\ \rho q_n v + S_{22} P \\ \rho q_n w + S_{23} P \\ \rho q_n H \end{bmatrix} \ , \tag{B.6}$$

where ρq_n is initially set to zero in the design problem. Equation (B.5) is true for any test function ϕ. We choose it to be the *Adjoint* function ψ. We can then add the constraint given by the *Euler* Equations to (B.1) to form the augmented cost function given by

$$\begin{aligned} I &= \int_{\mathcal{B}_\xi} \mathcal{M}(\mathbf{w}, \rho q_n) \, d\mathcal{B}_\xi \\ &- \int_{\mathcal{B}_\xi} n_i \psi^{\mathbf{T}} \mathbf{S_{ij}} \mathbf{f_j}(\mathbf{w}, \rho \mathbf{q_n}) \, d\mathcal{B}_\xi \\ &+ \int_{\mathcal{D}_\xi} \frac{\partial \psi^{\mathbf{T}}}{\partial \xi_{\mathbf{i}}} \mathbf{S_{ij}} \mathbf{f_j}(\mathbf{w}, \rho \mathbf{q_n}) \, d\mathcal{D}_\xi \ . \end{aligned} \tag{B.7}$$

Taking the first variation of the augmented cost function (B.7) we have

$$\begin{aligned}
\delta I &= \int_{\mathcal{B}_\xi} \left(\frac{\partial \mathcal{M}}{\partial \mathbf{w}} \delta \mathbf{w} + \frac{\partial \mathcal{M}}{\partial \rho q_n} \delta(\rho q_n) \right) d\mathcal{B}_\xi \\
&- \int_{\mathcal{B}_\xi} n_i \psi^{\mathbf{T}} \mathbf{S_{ij}} \left(\frac{\partial \mathbf{f_j}}{\partial \mathbf{w}} \delta \mathbf{w} + \frac{\partial \mathbf{f_j}}{\partial \rho \mathbf{q_n}} \delta(\rho \mathbf{q_n}) \right) d\mathcal{B}_\xi \\
&+ \int_{\mathcal{D}_\xi} \frac{\partial \psi^{\mathbf{T}}}{\partial \xi_\mathbf{i}} \mathbf{S_{ij}} \left(\frac{\partial \mathbf{f_j}}{\partial \mathbf{w}} \delta \mathbf{w} + \frac{\partial \mathbf{f_j}}{\partial \rho \mathbf{q_n}} \delta(\rho \mathbf{q_n}) \right) d\mathcal{D}_\xi \ .
\end{aligned} \tag{B.8}$$

Equation (B.6) gives the flux at the surface.

$$F_2 = \begin{bmatrix} \rho q_n \\ \rho q_n u + S_{21} P \\ \rho q_n v + S_{22} P \\ \rho q_n w + S_{23} P \\ \rho q_n H \end{bmatrix} , \tag{B.9}$$

Therefore,

$$\begin{aligned}
\delta F_2 &= \begin{bmatrix} 1 \\ u \\ v \\ w \\ E + \frac{P}{\rho} \end{bmatrix} \delta(\rho q_n) \\
&+ q_n \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ -u & 1 & 0 & 0 & 0 \\ -v & 0 & 1 & 0 & 0 \\ -w & 0 & 0 & 1 & 0 \\ (\gamma-1)\left(u^2+v^2+w^2\right) - \gamma E & -(\gamma-1)u & -(\gamma-1)v & -(\gamma-1)w & \gamma \end{bmatrix} \delta(w) \\
&+ (\gamma - 1) \begin{bmatrix} 0 \\ S_{21} \\ S_{22} \\ S_{23} \\ 0 \end{bmatrix} \begin{bmatrix} \frac{1}{2}\left(u^2+v^2+w^2\right) & -u & -v & -w & 1 \end{bmatrix} \delta(w)
\end{aligned} \tag{B.10}$$

So far, the choice of ψ has been arbitrary. We choose ψ to satisfy the *Adjoint* equations

$$\mathbf{S_{ij}} \frac{\partial \mathbf{f_j}}{\partial \mathbf{w}}^{\mathbf{T}} \frac{\partial \psi}{\partial \mathbf{w}} = 0 \text{ , on } \mathcal{D}_\xi \text{ ,} \tag{B.11}$$

and

$$\frac{\partial \mathcal{M}}{\partial \mathbf{w}} = \psi^{\mathbf{T}} \frac{\partial \mathbf{F_2}}{\partial \mathbf{w}} \text{ , on } \mathcal{B}_\xi \text{ .} \tag{B.12}$$

We also observe that

$$\frac{\partial f_j}{\partial \rho q_n} = 0 \text{ , on } \mathcal{D}_\xi \text{ .} \tag{B.13}$$

The expression for the *Adjoint* gradient then becomes

$$\begin{aligned} \delta I &= \int_{\mathcal{B}_\xi} \left(\frac{\partial \mathcal{M}}{\partial \rho q_n} \delta(\rho q_n) \right) \, d\mathcal{B}_\xi \\ &- \int_{\mathcal{B}_\xi} \left(\psi_1 + \psi_2 u + \psi_3 v + \psi_4 w + \psi_5 \left(E + \frac{P}{\rho} \right) \right) \delta \rho q_n \, d\mathcal{B}_\xi \\ &= \int_{\mathcal{B}_\xi} \mathcal{G}^T \delta \rho q_n \, d\mathcal{B}_\xi \end{aligned} \tag{B.14}$$

The gradient is then modified to account for the fact that the net mass flow through the boundaries is zero.

From Equations (B.12) and (B.14) it is clear that the *Adjoint* gradient depends only on the flow and *Adjoint* Variables at the Boundaries. Therefore this is clearly a case of *feedback* control, where the *feedback* is the values of the state variables at the boundary.

Bibliography

[1] H. Abe, T. Segawa, Y. Kikushima, H. Yoshida, A. Nishizawa, and S. Takagi. Towards smart control of separation around a wing, part 2. Technical report, Japan Aerospace Exploration Agency, 2003.

[2] J. J. Alonso. *Parallel Computation of unsteady aeroelastic flows using an implicit multigrid - driven algorithm.* PhD thesis, Princeton University, June 1997.

[3] J. J. Alonso and A. Jameson. Fully implicit time marching aeroelastic solutions. AIAA Paper 1994-0056, 32nd AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, January 1994.

[4] M. Amitay, M. Horvath, M. Michaux, and A. Glezer. Virtual aerodynamic shape modification at low angles of attack using synthetic jet actuators. AIAA Paper 2001-2975, 31st AIAA Fluid Dynamics Conference and Exhibit , Anaheim, CA, 2001.

[5] W. K. Anderson and D. L. Bonhaus. Aerodynamic design optimization on unstructured grids for turbulent flows. *AIAA Journal*, 37:185–191, 1999.

[6] H. Ashley and M. Landahl. *Aerodynamics of Wings and Bodies.* Dover Publications Inc., New York, 1985.

[7] D. Baker and M. I. Friswell. Design of a compliant aerofoil using topology optimisation. International Conference on Smart Materials and Structures, Toronto, CA, October 2006.

[8] P. S. Beran and D. J. Lucia. A reduced order cyclic method for computation of limit cycles. *Nonlinear Dynamics*, 39:143–158, 2005.

[9] T. Bewley. Flow control: new challenges for a new renaissance. *Progress in Aerospace Sciences*, 2001.

[10] T. Bewley, R. Temam, and M. Ziane. Existence and uniqueness of optimal control to the Navier-Stokes equations. *Mathematical Problems in Mechanics*, 2000.

[11] S. Bieniawski and I. Kroo. Development and testing of an Experimental Aeroelastic model with micro-trailing edge effectors. AIAA Paper 2003-220, 41st Aerospace Sciences Meeting and Exhibit, Reno, Nevada, 2003.

[12] R. L. Bisplinghoff, H. Ashley, and R. L. Halfman. *Aeroelasticity*. Dover Publications, 1996.

[13] H. M. Blackburn, J. R. Elston, and J. Sheridan. Bluff-body propulsion produced by combined rotary and translational oscillation. *Physics of Fluids*, January 1999.

[14] S. A. Brown. Displacement extrapolation for CFD + CSM aeroelastic analysis. AIAA Paper 1997-1090, 38th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference and Exhibit, Kissimmee, Florida, April 1997.

[15] A. E. Bryson and H. Yu-Chi. *Applied Optimal Control: Optimization, Estimation and Control*. Taylor and Francis In., 1988.

[16] K. Cohen, S. Siegel, and T. McLaughlin. Control issues in reduced order feedback flow control. AIAA Paper 2004-575, 42nd AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, January 2004.

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

[18] J. Elliot. *Aerodynamic design using unstructured meshes*. PhD thesis, MIT, 1998.

[19] J. Elliot and J. Peraire. Practical 3-d aerodynamic design and optimization using unstructured meshes. *AIAA Journal*, 35:1479–1485, 1997.

[20] R. P. Federenko. The speed of convergence of one iterative process. *USSR Comp. Math. and Math. Phsics*, 4:227–235, 1964.

[21] C. Ferrari. *Theory of Optimum Aerodynamic Shapes*. Academic Press Inc., New York, 1965.

[22] G. F. Franklin, D. Powell, and A. Emami-Naeini. *Feedback Control of Dynamic Systems*. Pearson Education, 2002.

[23] Y. C. Fung. *Foundations of Solid Mechanics*. Prentice Hall, 1965.

[24] M. B. Giles and N. A. Pierce. An Introduction to the Adjoint Approach to Design. *Flow, Turbulence and Combustion*, 65:393–415, 2000.

[25] M. B. Giles and N. A. Pierce. Analytic Adjoint Solutions for the quasi-one-dimensional Euler Equations. *Journal of Fluid Mechanics*, 426:327–345, 2001.

[26] A. Glezer and M. Amitay. *Annual Review of Fluid Mechanics*, chapter Synthetic Jets. 2002.

[27] S. K. Godunov. A difference method for the numerical calculation of discontinous solutions of hydrodynamic equations. *Mat. Sbornik*, 47:271 – 306, 1959. Translated as JPRS 7225 by U. S. Department of Commerce, 1960.

[28] A. Harten. High resolution schemes for hyperbolic conservation laws. *Journal of Computational Physics*, 49:357–393, 1983.

[29] E. Isaacson and H. B. Keller. *Analysis of Numerical Methods*. Dover Publications Inc., New York, 1994.

[30] A. Jameson. Calculation of inviscid transonic flow over a complete aircraft. AIAA Paper 1986-0103, 24th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, January 1986.

[31] A. Jameson. Optimum aerodynamic design using CFD and Control Theory. AIAA Paper 1995-1729, 12th AIAA Computational Fluid Dynamics Conference, San Diego, California, June 1995.

[32] A. Jameson. Time dependant calculations using multigrid, with applications to unsteady flows past airfoils and wings. AIAA Paper 1991-1596, 10th AIAA Computational Fluid Dynamics Conference, Honolulu, HI, June 1991.

[33] A. Jameson. Aerodynamic Design via Control Theory. *Journal of Scientific Computing*, pages 233–260, 1988.

[34] A. Jameson. 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*, 4:171–218, 1995.

[35] A. Jameson. Analysis and Design of Numerical Schemes for Gas Dynamics 2: Artificial Diffusion and Discrete Shock Structure. *International Journal of Computational Fluid Dynamics*, 5:1–38, 1995.

[36] A. Jameson. *Computational Fluid Dynamics Review*, chapter Optimum aerodynamic design using control theory. John Wiley and Sons, New York, 1995.

[37] A. Jameson. Multigrid Algorithms for Compressible Flow Calculations. Technical report, Princeton University Report MAE 1743, October 1985. In Second European Conference on Multigrid Methods.

[38] A. Jameson and T. Baker. Solution of the Euler Equations for Complex configurations. AIAA Paper 1983-1929, 6th AIAA Computational Fluid Dynamics Conference, Danvers, MA, July 1983.

[39] A. Jameson and S. Kim. Reduction of the Adjoint Gradient Formula in the Continuous Limit. AIAA Paper 2003-0040, 41st AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, January 2003.

[40] A. Jameson, N. A. Pierce, and L. Martinelli. Optimum aerodynamic design using the Navier-Stokes equations. AIAA Paper 1997-0101, 35th Aerospace Sciences Meeting and Exhibit, Reno, NV, January 1997.

[41] A. Jameson, W. Schmidt, and E. Turkel. Numerical Solution of the Euler Equations by Finite Volume Methods using Runge-Kutta Time-Stepping Schemes. AIAA Paper 1981-1259, 14th AIAA Fluid and Plasma Dynamics Conference, Palo Alto, June 1981.

[42] A. Jameson and J. C. Vassberg. Studies of alternative numerical optimization methods applied to the Brachistochrone problem. *Computational Fluid Dynamics Journal*, 9:281–296, 2000.

[43] A. Kuethe and C. Chow. *Foundations of Aerodynamics*. John Wiley and Sons, Inc., 1998.

[44] G. V. R. Kumar and A. Tewari. Active closed loop control of supersonic flow with transverse injection. AIAA Paper 2004-2699 2nd AIAA Flow Control Conference, Portland, Oregon, June 2004.

[45] P. Lax. Numerical Solution of Partial Differential Equations. *American Math Monthly*, 72(2):74–84, 1965.

[46] B. V. Leer. Towards the ultimate conservative difference scheme. *Journal of Computational Physics*, 14:361–370, 1974.

[47] K. Leoviriyakit. *Wing Planform Optimization via an Adjoint Method*. PhD thesis, Stanford University, 2005.

[48] H. W. Liepmann and A. Roshko. *Elements of Gas Dynamics*. Dover Publications, Reprint 2002.

[49] P.-J. Lu and S.-K. Chen. Acoustic Flutter Suppression of airfoil in Viscous Transonic flow. The 6th National Computational Fluid Dynamics Conference, Taiwan, August 1999.

[50] C. R. MacCluer. *Calculus of Variations*. Prentice Hall, 2005.

[51] R. Mani, D. C. Lagoudas, and O. K. Rediniotis. Active skin for turbulent drag reduction. *Smart Materials and Structures*, 2004.

[52] B. Mohammadi. Optimal Shape design, reverse mode of automatic differentiation and turbulence. AIAA Paper 1997-0099, 35th Aerospace Sciences Meeting and Exhibit, Reno, Nevada, January 1997.

[53] B. Mohammadi, editor. *Practical Applications to Fluid Flows of automatic differentiation for design problems*, 1997. VKI Lecture Series on Inverse Design.

[54] S. Nadarajah. *The Discrete Adjoint Approach to Aerodynamic Shape Optimization*. PhD thesis, Stanford University, 2003.

[55] C. Nae. Unsteady Flow Control using Synthetic Jet Actuators. AIAA Paper 2000-2403, Fluids 2000 Conference and Exhibit, Denver, Colarado, June 19-22, 2000.

[56] E. Nielsen and W. K. Anderson. Aerodynamic design optimization on unstructured meshes using the Navier-Stokes equations. AIAA Paper 1998-4809, 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, St. Louis, Missouri, September 1998.

[57] A. Nishizawa, S. Takagi, H. Abe, T. Segawa, and H. Yoshida. Towards smart control of separation around a wing, part 1. Technical report, Japan Aerospace Exploration Agency, 2003.

[58] S. Osher and S. Chakravarthy. High resolution schemes and the entropy condition. *SIAM Journal of Numerical Analysis*, 21:955–984, 1984.

[59] K. Palaniappan, P. S. Beran, and A. Jameson. Optimal Control of LCOs in Aero-Structural Systems. AIAA Paper 2006-1621, 2nd AIAA Multidisciplinary Design Optimization Specialist Conference, May 1-4, 2006, Newport, Rhode Island.

[60] K. Palaniappan and A. Jameson. An analysis of bodies having minimum pressure drag in supersonic flow: Exploring the nonlinear domain. International Conference on Computational Fluid Dynamics (ICCFD3), Toronto, Canada, July 12-16 2004.

[61] K. Palaniappan and A. Jameson. Bodies having minimum pressure drag in supersonic flow: Investigating nonlinear effects. AIAA Paper 2004-5383, 22nd AIAA Applied Aerodynamics Conference and Exhibit, Rhode Island, August 16-19, 2004.

[62] K. Palaniappan and A. Jameson. Feedback control of aerodynamic flows. AIAA Paper 2006-843, 44th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, 2006.

[63] K. Palaniappan, P. Sahu, J. J. Alonso, and A. Jameson. Active flutter control using an adjoint method. AIAA Paper 2006-844, 44th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, 2006.

[64] O. Pironneau. *Optimal Shape Design for Elliptic Systems*. Springer-Verlag, 1984.

[65] S. Premasuthan and A. Jameson. Mesh deformation using a material model. Technical report, Stanford University, 2005.

[66] J. Reuther and A. Jameson. Control based airfoil desing using the Euler equations. AIAA Paper 1994-4272, 5th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Panama City Beach, Florida, September 1994.

[67] J. Reuther, A. Jameson, J. J. Alonso, M. J. Remlinger, and D. Saunders. Constrained multipoint aerodynamic shape optimization using an adjoint formulation and parallel computers, Part 1. *Journal of Aircraft*, 36(1):51–60, 1999.

[68] J. Reuther, A. Jameson, J. J. Alonso, M. J. Remlinger, and D. Saunders. Constrained multipoint aerodynamic shape optimization using an adjoint formulation and parallel computers, Part 2. *Journal of Aircraft*, 36(1):61–74, 1999.

[69] J. Reuther, A. Jameson, J. Farmer, L. Martinelli, and D. Saunders. Aerodynamic shape optimization of complex aircraft configurations via an adjoint formulation. AIAA Paper 1996-0094, 34th Aerospace Sciences Meeting and Exhibit, Reno, Nevada, January 1996.

[70] P. L. Roe. Approximate Riemann solvers, parameter vectors, and difference schemes. *Journal of Computational Physics*, 1981.

[71] C. Rogers. Intelligent materials. *Scientific American*, 1995.

[72] P. Sahu. Simulation and control of flutter. Master's thesis, Stanford University, 2006.

[73] M. Samimy, M. Debiasi, E. Carabello, J. Malone, J. Little, H. Ozbay, M. O. Efe, P. Yan, X. Yuan, J. DeBonis, J. H. Myatt, and R. C. Camphouse. Exploring strategies for closed loop cavity flow control. AIAA Paper 2004-576 42nd AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, January 2004.

[74] A. Seifert, V. Theofilis, and R. D. Joslin. Issues in active flow control: Theory, simulation and experiment. AIAA Paper 2002-3277, 1st AIAA Flow Control Conference, Missouri, 2002.

[75] S. Shankaran. *Numerical Analysis and Design of Upwind Sails*. PhD thesis, Stanford University, 2003.

[76] E. Swim. Unpublished notes on Von Karman's Large Deflection Plate Equations. 2005.

[77] T. Theodorsen and I. E. Garrick. Mechanism of flutter, a theoretical and experimental investigation of the flutter problem. Technical report, N.A.C.A. Report 685, 1940.

[78] A. Tuck and J. Soria. Active Flow Control of a NACA 0015 Airfoil using a ZNMF Jet. 15th AIAA Australian Fluid Mechanics Conference, December 2004.

[79] O. C. Zienkiewicz and R. L. Taylor. *The Finite Element Method*, volume 1. Butterworth-Heinemann, 2000.