OPTIMIZATION OF DYNAMIC SYSTEMS USING COLLOCATION METHODS

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

By
Michael E. Holden
May 1999

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.

Ilan Kroo
(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.

Holt Ashley

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 Alonso

Approved for the University Committee on Graduate Studies:

Acknowledgments

Above all, I would like to thank my wife, Jen, who provided boundless support in spite of her own free-time constraints.

I would like to thank my reading committee: Professor Kroo, who originally suggested combining collocation with aeroelasticity, and who constantly supported me both intellectually and financially; Professor Ashley, whose works on aeroelasticity are heavily referenced in this thesis, and whose classes taught me the fundamentals of dynamic structural analysis; and Professor Alonso, who graciously allowed me to run the final optimizations in this thesis on his workstation, without which these cases would *still* be running on some crappy two-bit machine.

I must also thank Michael Saunders, whose excellent program MINOS proved itself a true "black box", solving all the optimization problems in this thesis with very little effort on my part.

My fellow students and graduates of the group were an invaluable pool of knowledge and friendship, and I would particularly like to thank Ian, Mark, Valerie, Ben, and Dave, as well as all the older and younger members of the research group.

Writing a book involving many years of work, with the attendant branches and changes of focus, can fill one with both the pride of achievement and the desire to have done a few things differently. I have found the following quote helpful when confronted with this desire:

> To rest below his own aim is incident to everyone whose fancy is active, and whose views are comprehensive; nor is any man satisfied with himself because he has done much, but because he can conceive little.
> *Samuel Johnson* [36]

Abstract

The time-based simulation is an important tool for the engineer. Often a time-domain simulation is the most expedient to construct, the most capable of handling complex modeling issues, or the most understandable with an engineer's physical intuition. Aeroelastic systems, for example, are often most easily solved with a nonlinear time-based approach to allow the use of high fidelity models. Simulations of automatic flight control systems can also be easier to model in the time domain, especially when nonlinearities are present.

Collocation is an optimization method for systems that incorporate a time-domain simulation. Instead of integrating the equations of motion for each design iteration, the optimizer iteratively solves the simulation as it finds the optimal design. This forms a smooth, well-posed, sparse optimization problem, transforming the numerical integration's sequential calculation into a set of constraints that can be evaluated in any order, or even in parallel. The collocation method used in this thesis has been improved from existing techniques in several ways, in particular with a very simple and computationally inexpensive method of applying dynamic constraints, such as damping, that are more traditionally calculated with linear models in the frequency domain.

This thesis applies the collocation method to a range of aircraft design problems, from minimizing the weight of a wing with a flutter constraint, to gain-scheduling the stability augmentation system of a small-scale flight control testbed, to aeroservoelastic design of a large aircraft concept. Collocation methods have not been applied to aeroelastic simulations in the past, although the combination of nonlinear aerodynamic analyses with structural dynamics and stability constraints is well-suited to collocation. The results prove the collocation method's worth as a tool for aircraft design, particularly when applied to the multidisciplinary numerical models used today.

Contents

Acknowledgments **iv**

Abstract **v**

1 Introduction **1**

1.1 Motivation for Research on Collocation 1

1.2 Selected History of Collocation Methods 4

1.3 Suitable Design Problems 5

1.3.1 Aeroelasticity 6

1.3.2 Feedback Control 7

1.3.3 Less-Suitable Problems 7

1.4 Thesis Summary and Contributions 8

2 Formulation and Implementation **11**

2.1 Collocation Constraints 13

2.1.1 Cubic Spline Form 13

2.1.2 Trapezoidal Form 16

2.1.3 Taylor Series Collocation Constraint 17

2.2 Stability Constraint 19

2.3 Implementation Issues 21

2.3.1 Sparse Optimization 22

2.3.2 Sparse Finite Difference Gradients 23

2.3.3 Architecture for Parallel Calculation 24

2.3.4 Efficient Equations of Motion 25

2.4 Typical Section Example 27
2.4.1 Equations of Motion 28
2.4.2 Objective 29
2.4.3 Design Variables 30
2.4.4 Collocation Constraints 30
2.4.5 Objective Gradient 32
2.4.6 Jacobian 32
2.4.7 Stability Constraints 33
2.4.8 Results: Finding Flutter Speed 33
2.5 Conclusions and Summary 34

3 Collocation for Aeroelastic Optimization **36**
3.1 Introduction 36
3.2 Aerodynamic Model 37
3.2.1 Panel Method Framework 38
3.2.2 Additional Stripwise Parameters 39
3.2.3 Linearized Forces 39
3.3 Finite Element Model 40
3.3.1 Motivation for This Model 40
3.3.2 Structural Model 41
3.4 Aeroelastic Equations of Motion 42
3.5 Jet Transport Simulation Results 44
3.5.1 Aerodynamic Model 44
3.5.2 Structural and Mass Parameters 45
3.5.3 Simulation Parameters 47
3.5.4 Linear Stability Results 47
3.5.5 Time Simulation 48
3.6 Optimization Problem 49
3.7 Collocation and Stability Constraints 52
3.8 Structural Scaling with Physical Degrees of Freedom 53
3.9 Nonlinear Dynamics 53

3.10 Modal Degrees of Freedom . . . 58
3.11 Stability and Motion Bounds . . . 58
3.12 Additional Structural Design Variables . . . 62
3.13 Conclusions . . . 65

4 Control Design Using Collocation **67**
4.1 Introduction . . . 67
4.1.1 The Blended-Wing-Body . . . 67
4.1.2 Flight Control Testbed . . . 69
4.2 Application to Blended-Wing-Body Control . . . 72
4.3 Design Problem . . . 73
4.4 Simulation . . . 76
4.4.1 Nonlinearities . . . 76
4.4.2 Initial Conditions and Pilot Inputs . . . 78
4.5 Collocation Method . . . 78
4.6 Solving for Flight Gains . . . 80
4.7 Gains with Neutral Stability . . . 81
4.8 Conclusions . . . 82

5 Aeroservoelastic Optimization **85**
5.1 Introduction . . . 85
5.2 Simulation Model . . . 88
5.2.1 Aerodynamic Model . . . 91
5.2.2 Feedback Control Model . . . 92
5.2.3 Finite Element Model . . . 94
5.2.4 Additional Modifications to Finite-Element Matrices . . . 97
5.2.4.1 Weight Breakdown . . . 98
5.2.4.2 Fuel Weight . . . 99
5.2.4.3 Structural Weight . . . 99
5.2.4.4 Boundary Conditions . . . 100
5.2.4.5 Matching Rigid-Body Inertia . . . 103
5.2.5 Separation of Rigid and Elastic Degrees of Freedom . . . 105

5.2.6 Modal (Eigen) Analysis . . . 108
5.2.7 Rigid-Body Equations of Motion . . . 109
5.2.8 Converting Between Degrees of Freedom . . . 110
5.2.8.1 Dynamic Degrees of Freedom to Physical Nodes . . . 110
5.2.8.2 Inertial Loads . . . 112
5.2.8.3 Aerodynamic-Structural Coupling . . . 112
5.2.9 Initial Structural Sizing . . . 114
5.3 Optimization Problem . . . 115
5.3.1 Design Variables . . . 117
5.3.2 Dynamic Stability and Control Authority Constraints . . . 118
5.3.3 Finite Difference Gradients . . . 119
5.4 Collocation Results . . . 120
5.5 Collocation Simulation Decomposition . . . 124
5.5.1 Coarse-Grained Decomposition Using Collocation . . . 124
5.5.2 Parallel Computations . . . 127
5.6 Conclusions . . . 127

6 Conclusions and Recommendations 129
6.1 Collocation is Suitable for a Variety of Dynamic Simulations . . . 130
6.2 Modifications to the Collocation Method . . . 131
6.2.1 Collocation Constraint Formula . . . 131
6.2.2 Stability Constraint . . . 132
6.3 Aeroelastic Analyses for Collocation . . . 133
6.4 Future Work . . . 135

A "Collocation Method" Anagrams 137

Bibliography 138

List of Tables

2.1 Jacobian Sparsity Pattern 32

3.1 Structural Properties 45
3.2 Mass Properties 46
3.3 Jet Transport Simulation Parameters 47
3.4 Stability Bound Results 59
3.5 Linear Structural Variation Optimization Results 63

5.1 BWB Component Weight Breakdown 99
5.2 Inertial Properties of the Rigid Blended Wing Body 103

List of Figures

1.1 Time-Domain Specifications 2

2.1 Collocation Constraint Curve Fit 14
2.2 Cubic Polynomial Collocation Constraint 14
2.3 Trapezoidal Collocation Constraint 17
2.4 Second Order Taylor Series Collocation Constraint 18
2.5 Error of Trapezoidal and Taylor Series Constraints for Aileron Response 19
2.6 Envelope of Motion for Linear System 20
2.7 Optimization with Integrated Equations of Motion and Collocation . 25
2.8 Typical Section Model 27
2.9 Typical Section Simulation Results 34

3.1 Aerodynamic Model 38
3.2 Typical Section Model 41
3.3 Structural Model 42
3.4 Jet Transport Wing Planform 45
3.5 Location of Wing Structural and Mass Beams 46
3.6 Static Deflection 47
3.7 Real Eigenvalues 48
3.8 Root Loci vs. Speed 49
3.9 First 4 Mode Shapes 50
3.10 Displacements in Time 51
3.11 Wing Node Time History with Active Bound 54
3.12 Nonlinear Stiffness 54

3.13 Wing Tip Torsion Time History 56
3.14 Objective Iteration History 56
3.15 Constraint Iteration History 57
3.16 Motion in Tight Bounds 60
3.17 Motion with Loose Bounds 60
3.18 Sensitivity of Structural Stability to Motion Bounds 61
3.19 Path in Design Space 63
3.20 Objective and Constraint Iteration History 64
3.21 Time History of 2-D Structural Design Problem 64

4.1 Blended-Wing-Body Configuration 68
4.2 Flight Control Development 69
4.3 BWB Flight Control Testbed 70
4.4 Flight Control Testbed Layout 71
4.5 Car-top Test Rig 72
4.6 Car-top Test Data 74
4.7 Sample Time vs Elapsed Time 77
4.8 Optimized Responses, Tight Bounds 80
4.9 Gain Schedule 81
4.10 Optimized Responses, Loose Bounds 82
4.11 Gain Schedule 83

5.1 Blended-Wing-Body Aircraft (Courtesy Mark Page) 86
5.2 Blended-Wing-Body Structural Model 90
5.3 Blended-Wing-Body Aerodynamic Model 92
5.4 Cutoff Function 93
5.5 Wing Structure 95
5.6 Spar, Rib and Skin Construction 95
5.7 Supersonic Transport Wing Structure 97
5.8 Comparison of Stress and Displacement from Two Finite Element Codes 98
5.9 Location of Fuel Tanks 100
5.10 Center Section is Rigidly Attached to Root Rib 101

5.11 Wing Lift Discretization for Structural Loads 113
5.12 Static Sizing Results 116
5.13 Aircraft Pitch Time History 121
5.14 Structural Modes Time History 121
5.15 Open-Loop Pitch Time History 122
5.16 Open-Loop Structural Modes Time History 122
5.17 Decomposition of Equations of Motion 125
5.18 Decomposition Results 126

Chapter 1

Introduction

1.1 Motivation for Research on Collocation

The time-domain simulation is a valuable engineering tool for modeling dynamic systems. While many dynamic systems can be elegantly analyzed by transforming them into the frequency domain, or using other linear techniques, often the "brute force" approach of a time-domain simulation is the most expedient to construct, the most capable of handling complex modeling issues, or the most understandable with an engineer's physical intuition.

For example, consider the time response of a simple second-order dynamic system, shown in figure 1.1. Every elementary control-design textbook has a diagram of the time response of a second-order system showing the design parameters such as rise time, settling time, and overshoot. The design requirements are then mapped into the frequency domain, and relations are made between natural frequencies, damping ratios, and selection of pole and zero locations in the s-plane to meet the required design specifications. Much effort can be spent trying to make a complex system map accurately into the frequency domain. Modeling nonlinearities such as those found with digital computers, aerodynamic forces, and large deflections or rotations may lead an engineer to construct a straightforward time-domain simulation.

While a time simulation can be easier to construct than a frequency-domain analysis, the time-domain results can be difficult to interpret. The settling time is easy

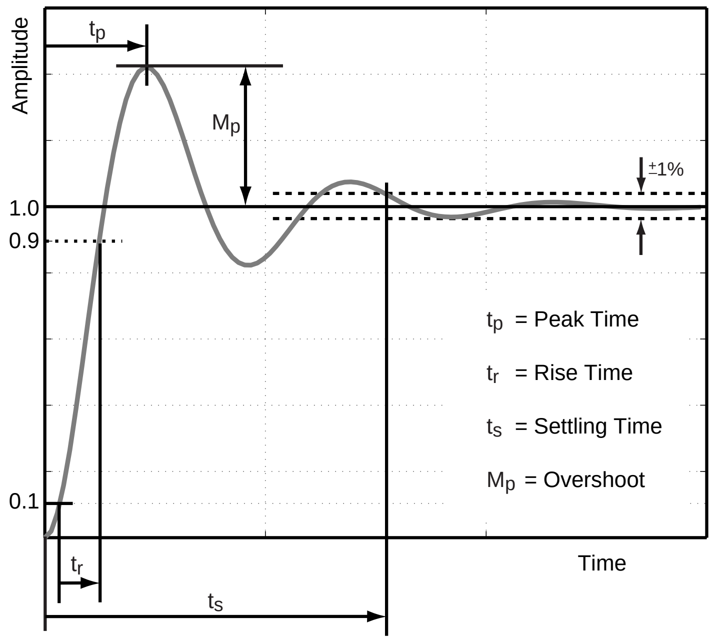

Figure 1.1: Time-Domain Specifications

to visualize, but to calculate it from a time history requires locating the maxima and minima to calculate the amplitude, making the simple relations of the frequency domain look quite attractive. Stability can be seen in the time domain simulation by whether the history is damped or diverges, but is much easier to calculate by looking at a root locus. While an experienced engineer can decide from a time history if the design "looks good", automating the design process with numerical optimization requires that formal measures of stability and damping be found. An optimizer will always understand "real parts of eigenvalues must be less than zero", but formally posing "motion must not diverge" can be a problem. Of course, if the designer wishes to know integrated quantities such as the fuel used during an aircraft's flight, a time domain simulation will yield easy answers.

Optimizing with time-domain simulations has other drawbacks. A time-domain simulation is a sequential calculation that must proceed from the initial conditions to the final time step. Parallelizing this form of analysis to speed up computation time can be difficult. Also, the combination of finite-difference gradients in optimization with a time simulation that uses quadrature with variable length time steps can make the gradients very noisy [24], with disastrous optimization results.

The design engineer is often faced with a choice: construction of a complex simulation in the frequency domain, which will yield straightforward measures of performance; or a straightforward time-domain simulation, whose results may require complex post-processing to evaluate the design. Sometimes with complex nonlinear systems the engineer has no choice but to work in the time domain.

In this thesis, the time-domain route will be explored. A collocation method, used in the past for trajectory optimization [7] [8] [5] [9] [6] [13] [16] [17] [18] [28] [27] [33] [37] [38] , will be extended by applying design constraints that are typically posed in the frequency domain to the optimization of time-domain simulations.

The collocation method is a technique for optimization problems that use time-domain simulations in order to evaluate the design objective and constraints. It transforms the simulation so that it is well-posed for gradient-based optimization and parallel calculation, and adds an efficient way to include design specifications such as damping, overshoot, and settling time without using frequency-domain calculations.

The basic concept in the collocation method is to remove the process of integrating the equations of motion. The optimizer designs the time history as it designs the rest of the system, and is constrained to design a physically realistic time history with collocation constraints. Every state variable in the time history is added to the set of optimization design variables. Collocation constraints, located at time points between those of the design variables, incorporate the equations of motion to keep the simulation physically accurate. The form of the collocation constraint leads to a very sparse constraint Jacobian (the matrix that contains the gradient of the constraints with respect to the design variables), so while the size of the optimization problem using the collocation method is very large, typically with thousands of design variables and constraints, the actual memory requirements are small and solutions can be found efficiently using sparse optimization algorithms. Compared with simply wrapping the optimizer around an integrated-equations-of-motion simulation, the optimization problem is transformed from a small, sequential calculation to a large, sparse calculation that may be parallelized.

1.2 Selected History of Collocation Methods

Collocation was first used to optimize dynamic systems in the late 1970's. Numerical optimization applies a heavy computational load to the computer, and the size of the collocation optimization problem requires a large amount of memory. As computers have become more powerful, the complexity of the problems solved using collocation has increased. Spacecraft and rocket trajectories are the most common subject in the literature of collocation analyses. The equations of motion for rockets and spacecraft are very nonlinear, and the small number of state variables keeps the problem size manageable. Most of the collocation problems to date have been to find the control inputs (such as thruster burn) that achieve a desired trajectory while minimizing some parameter (such as fuel used or time in flight). Much research effort has been spent trying to find the most efficient way to solve this class of problem, from the form of the collocation constraint to the optimization algorithm.

One of the earliest collocation method studies is [28], by Hargraves et al. A large suite of trajectory optimization problems, including the brachistochrone, a subsonic transport aircraft climb problem, supersonic aircraft climb problem, Goddard rocket trajectory, fuel minimization for a subsonic transport, and evasive maneuvers for a fighter, is solved using a modified Newton's method optimizer, enforcing the collocation constraints with penalty functions added to the objective.

A similar set of test problems is solved by a similar set of authors in [27]. The state of the art is advanced by changing the collocation constraint curve-fit to a cubic polynomial instead of Chebychev polynomials, and changing the optimization algorithm to a sequential quadratic programming (SQP) constrained optimizer instead of the Newton method with penalty functions.

Enright and Conway find optimal spacecraft trajectories in [18]. They are able to reduce the size of the optimization problem by using an analytical solution to the equations of motion during the "coast arcs" of the trajectory.

A very thorough look into the mathematics of collocation methods is presented by Betts and Huffman of Boeing in [7], [8], and [33]. Here the sparsity of the Jacobian is first exploited, to reduce storage requirements, speed up the linear algebra involved in the solution, and efficiently calculate finite-difference gradients. Various collocation constraints based on cubic splines, cubic polynomials, and linear fits are compared, showing a trade between computational efficiency and time-step size among collocation constraints with different orders of accuracy.

The trajectory of a commercial airliner is optimized using collocation in [5]. Flight path constraints are imposed of the same form as the stability constraints presented in this thesis. These constraints are not imposed for dynamic stability, but to ensure that the aircraft trajectory is consistent with the phase of the flight, such as constant Mach number or rates of climb.

Braun, in [12], solved the collocation-based optimization of a lunar ascent trajectory problem as a demonstration of the collaborative optimization decomposition technique. The parallel nature of the optimization problem formed with collocation allowed the subproblem size to be varied to examine the efficiency of coarse and fine-grained decomposition.

The cited works show a clear path of the development of the collocation method, in solution methodology, examples of robustness, and suitable problem types. They also indicate avenues that might be explored further, such as extending the family of problems into structural dynamics, frequency-domain style stability constraints, and exploiting the parallel nature of the collocation calculation. Some of these roads will be travelled in this thesis.

1.3 Suitable Design Problems

The collocation method is designed for optimization of a time-domain simulation, with or without the need to apply stability or other frequency-domain constraints. The method can be used to solve the equations of motion for any simulation, but if a single simulation is all that is desired, the collocation method is much less efficient than numerical integration. Its efficiency comes when the simulation is combined with the optimization and solved simultaneously. The optimization problem formed by the collocation method allows constraints on the dynamic motion to be added with negligible computational cost, so systems that have stability constraints on the motion, such as a particular level of damping, are especially suitable for collocation.

1.3.1 Aeroelasticity

The collocation method is applicable to many problems in the field of aeroelasticity. Aeroelastic simulations combine aerodynamic, structural, and dynamic analyses. Most high-fidelity models, especially aerodynamic models, are difficult if not impossible to express in the frequency domain. Experimentally-determined corrections to computational results, such as flow separation and control effectiveness, are common in dynamic simulations, and are usually simpler to model in the time domain. Aircraft structures have long been designed using numerical optimization, and the need to combine optimization with time-based simulation forms the class of problem that the collocation method was designed to solve.

Dynamic constraints, such as flutter, are frequently present in aeroelastic design problems, and easily included with collocation. Structural optimization of an aircraft wing with a flutter constraint is one of the first problems that will be solved with the collocation method in this thesis.

Helicopter blade aeroelasticity [41] is a good candidate for time-domain simulation and optimization [41], with large deflections that require geometric nonlinearities in the structural analyses, and sophisticated unsteady aerodynamic models. Turbine blade aeroelastic simulations also have many of the same qualities, and would be well-matched to collocation.

Another aeroelastic problem that would work well with collocation is aeroelastic design of flapping wings, where the objective would be to minimize drag over a few flapping cycles. A time-domain simulation would be a simple way to analyze the motion; combination with collocation would allow the structure and flapping dynamics to be optimized to maximize lift or thrust during the flight. This same form of analysis could be applied to elastic winged keels on racing sailboats to achieve “negative drag” by favorable dynamic coupling between wave and boat motion.

1.3.2 Feedback Control

Closed loop control is the field for which the collocation method was originally developed. Using the collocation method for optimization of feedback control parameters with respect to a trajectory simulation allows the simulation to be as complex as desired, including nonlinearities such as digital control systems, non-ideal actuator models, or experimentally determined parameters. The stability constraint presented in this thesis broadens the scope of suitable problems to those with dynamic constraints that are usually expressed in the frequency-domain.

1.3.3 Less-Suitable Problems

Collocation is not applicable to all forms of simulation, however. Some dynamic systems, when expressed in collocation form, do not form a sparse optimization problem. The method can still be applied, but the optimization may become prohibitively expensive in terms of computational resources. One aerodynamic problem that seems appropriate for collocation is wake roll-up of the shed vorticity in a panel code. Typically the wake is started as a flat sheet, then moved with the local flow until a zero-force wake is achieved, with every wake panel parallel to the local velocity. As an iterative process, it makes sense to let an optimizer direct the convergence rather than the simple fixed-point iteration schemes usually used (e.g. [47]). The problem with using the collocation method is that the motion of one wake panel is dependent on the location of all the other wake panels, unlike a typical dynamic system where the state derivatives are a function only of the previous state (or perhaps a few states) in the time history. The wake roll-up problem can be posed in collocation form, but as we will see, the collocation constraints will not create a sparse Jacobian, requiring a large amount of calculations and memory for the solution. This added complexity is found only when the state variable derivatives are dependent on the entire time history, rather than just the previous state. Of course, there is no theoretical reason why the dense-Jacobian problem may not be solved; the memory and algebraic requirements are just less manageable.

1.4 Thesis Summary and Contributions

The thesis begins with a thorough explanation of the collocation optimization techniques used in the work that will follow. The general method of transforming a time-based simulation into a collocation optimization problem is shown, and the additions to the set of design variables and constraints in order to solve the equations of motion are listed. The different forms of collocation constraints used in the work of this thesis are given, including a new form based on a Taylor series expansion that is well-suited to equations of motion that are based on Newton's second law. The stability constraint, a method that allows the optimization process to control damping and stability, is introduced. This constraint for dynamic stability is a new technique, and broadens the scope of the collocation method to include a large family of optimization problems where dynamic stability is a design requirement. Implementation issues particular to collocation methods are discussed also, to show how the form of the optimization problem may be exploited to speed up the numerical calculations.

An example collocation problem is given, using a simple two degree of freedom aeroelastic model. The details of implementing the collocation method to solve the equations of motion, and the use of the optimizer to maximize the airspeed subject to a flutter constraint are shown. This example gives a solid foundation upon which to build more complex results.

The first application of the collocation method in this thesis is in Chapter 3, where the structural weight of an aircraft wing is minimized subject to a flutter constraint, and is the first use of collocation with an aeroelastic simulation to date. The collocation algorithm is extended through the use of the Taylor series collocation constraint and the stability constraint. The stability constraint is used to ensure that the motion of the system is damped, while minimizing the wing weight. The effectiveness of the stability constraint in providing damping is shown by comparing the collocation results with solutions from a frequency-domain analysis of the linear aeroelastic model. The frequency-domain results also are used to show that the stability constraint can be posed to find neutrally-stable designs, even with nonlinear systems. Nonlinear equations of motion are solved to show that the problem formulation and the optimizer's solution processes are the same for linear and nonlinear systems. Modal analysis is incorporated into the linear structural model and is shown to increase the optimization speed by reducing the number of degrees of freedom and increasing the time step size.

In addition to the collocation techniques, a new aeroelastic analysis was developed for this work that is a simple, effective method for optimization of moderate to high aspect-ratio wings. The beam-based model developed for Chapter 3 has since been used in research ranging from wind-tunnel aeroelastic models to America's Cup sailboat keel design.

Chapter 4 applies the collocation method to the design of a feedback control system for a remotely-piloted flight vehicle, a 17-foot span dynamically-scaled flight-control testbed for a large aircraft concept. A simple nonlinear time-domain simulation was used, whose parameters could be estimated and validated by comparing the results against experimentally measured data, and the closed-loop damping constraints were posed using the stability constraint on the dynamic motion. The collocation method was not only able to find feedback gains for stable, well-damped flight, it correctly predicted the airspeed at which the original control design was seen to become unstable in experimental testing.

The collocation method is especially suitable for systems that are too complex or nonlinear to simulate by any other means than solving the equations of motion in the time-domain. An example of such a complex system is modeled in Chapter 5: the aeroservoelastic response of a tailless aircraft. Another finite-element analysis tool was created for this work, and it too has been used in other multidisciplinary optimization work [49]. The stability constraints are used to design a well-damped aeroservoelastic response in the design of a feedback control system. The large optimization problem and time-consuming function evaluations created with the high-fidelity analysis illustrate the need for faster methods to solve the collocation problem, such as parallel computation. A decomposition method to reduce the number of optimization design variables is implemented, and the change in solution time is investigated.

The contributions to the state of the art in this thesis lie in extending the applicability and construction of the collocation method, a method that has been applied previously to a limited class of trajectory optimization problems, to aeroelastic and closed-loop control design. The stability constraint is a key modification of the method that allows the implementation of frequency-domain style design specifications. The Taylor series constraint is a new collocation constraint that is especially well-suited to equations of motion based on Newton's second law. Additionally, two new finite-element analyses were created and combined with aerodynamic analyses, to form the aeroelastic simulations that give the results contained herein.

Chapter 2

Formulation and Implementation

The optimization of dynamic simulations using collocation is a relatively straightforward procedure, not much more involved, for instance, than integrating the equations of motion with a numerical quadrature method. As the previous chapter mentioned, the set of design variables is enlarged with the addition of all the state variables in the simulation time history, and collocation constraints are added to enforce the equations of motion. The issues that need further explanation are exactly how collocation constraints are expressed and how different forms of the collocation constraint will affect the solution process, how the stability constraint is imposed to control the dynamic response of the system, and methods for efficiently implementing the collocation problem. Three forms of the collocation constraint will be examined in this chapter: two common forms often seen in collocation literature, and a new form especially created for second-order systems, such as those whose equations of motion are based on Newton's second law. This chapter will introduce a dynamic stability constraint that is a powerful method to shape the dynamic response of a system while adding very little computational expense to the solution process. Practical implementation matters such as efficient optimization algorithms for collocation, gradient calculations for sparse systems, and multi-processor computation will be examined so that the collocation problem is as efficient as possible. A two-degree-of-freedom aeroelastic model will be used to bring the topics of this chapter together in a simple example.

The equations of motion of the system to be simulated can either be written in the most general state variable form [21], equation 2.1, or in a common subset of the state equation based on Newton's second law, equation 2.2.

$$\dot{\vec{x}} = f(\vec{x}) \tag{2.1}$$

$$\ddot{\vec{x}} = f(\vec{x}, \dot{\vec{x}}) \tag{2.2}$$

The time history is the solution of these equations for the state variables ($\vec{x}$ for equation 2.1, $\vec{x}$ and $\dot{\vec{x}}$ for equation 2.2), given a set of state variables at the initial time (the initial conditions).

Any second order system such as that given by equation 2.2 can be transformed into the first order system of equation 2.1 by defining a new state variable, $\vec{\chi}$, that includes both the position and velocity variables:

$$\vec{\chi} = \left\{ \begin{array}{c} \vec{x} \\ \dot{\vec{x}} \end{array} \right\} \tag{2.3}$$

This transformation is often performed on equations of motion so that they are in the most general form to be solved by numerical methods (or, historically, electronic integration using an analog computer). We will see in the next section that although the collocation method can solve problems in either first or second order form, the transformation given in equation 2.3 is not necessary for collocation, and, by hiding the higher derivatives, some information is lost.

All the state variables, either $\vec{\chi}$, or $\vec{x}$ and $\dot{\vec{x}}$, at every time point in the simulation after the initial condition, will be added to the set of design variables the optimizer controls. The optimizer, therefore, designs the response as it designs the rest of the system, with only the collocation constraints to enforce the equations of motion.

2.1 Collocation Constraints

Collocation constraints are used to force the optimizer to choose values for the time-history state variables so that the equations of motion are satisfied. There are two parts to this constraint:

- The equations of motion must be satisfied for every time point in the simulation.

- The derivatives of the state variables must accurately reflect the way the state variables change in time, to form a continuous time history.

Satisfying the equations of motion is obviously required for physically-realistic motion, but it is just as important to also constrain the way the history of the variables changes so that the optimizer will not design systems that are locally in equilibrium but discontinuous in time. This assumes, of course, that the system does not admit discontinuous state variables, but this assumption is also made by the numerical integration routines that collocation replaces.

The general form for a collocation constraint is a curve-fit of the state variables, based on the state variables at one or more design variable time points and the state variable derivatives as calculated from the equations of motion. The curve fits used in the literature have been as varied as Chebyshev polynomials [28], Hermitian splines [8], cubic polynomials [27], Taylor series approximations [7] and forms based on Runge-Kutta integration [8]. Figure 2.1 shows a typical state-variable curve-fit through design variables and a collocation point, used for the collocation constraint. The curve-fit is used to extrapolate the state variables forward and backward in time from design variables to collocation points located between the design variables in time. The collocation constraint is applied at this collocation point to satisfy the equations of motion and maintain continuity of the time history.

2.1.1 Cubic Spline Form

The cubic spline form of the collocation constraint first appears in [27], and assumes that the equations of motion are in state form (equation 2.1). The curve-fit between

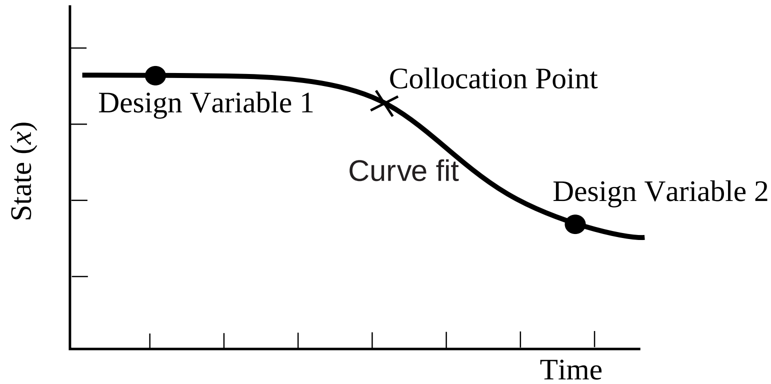

Figure 2.1: Collocation Constraint Curve Fit

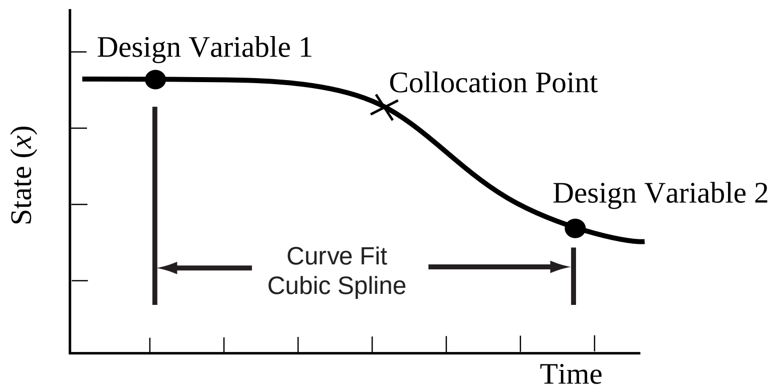

Figure 2.2: Cubic Polynomial Collocation Constraint

design variables is the cubic polynomial given in equation 2.4, where $s = \frac{t-t_1}{t_2-t_1}$.

$$x = c_0 + c_1 s + c_2 s^2 + c_3 s^3 \tag{2.4}$$

The cubic-spline constraint is illustrated in figure 2.2. The four coefficients, $\{c_0, c_1, c_2, c_3\}$, that define the spline between points t_1 and $t_2 \equiv t_1 + dt$, are determined by $\{x_1, \dot{x_1}, x_2, \dot{x_2}\}$, where $\dot{x_1}$ and $\dot{x_2}$ are calculated using the equations of motion.

The equation of the spline is then used to calculate the state at the collocation point, $\vec{x}_{spline}$, and the derivative of the spline equation is calculated at the collocation point to find $\dot{\vec{x}}_{spline}$. The solutions to these calculations are given in equations 2.5 and 2.6, where the spline coefficients have been expressed in terms of the design variables and their derivatives, $\{x_1, \dot{x_1}, x_2, \dot{x_2}\}$, for compactness.

$$x_{spline} = \frac{x_i + x_{i+1}}{2} + \frac{dt}{8}(\dot{x}_i - \dot{x}_{i+1}) \tag{2.5}$$

$$\dot{x}_{spline} = -\frac{3}{2}\frac{(x_i - x_{i+1})}{dt} - \frac{\dot{x}_i + \dot{x}_{i+1}}{4} \tag{2.6}$$

The equations of motion of the system are then used to again find the state derivatives at the collocation point, $f(\vec{x}_{spline})$. The cubic spline collocation constraint specifies that the state derivative calculated by differentiating the spline must equal the state derivative calculated from the equations of motion using the state at the collocation point, as equation 2.7 shows.

$$\vec{c}_i = \dot{\vec{x}}_{spline} - f(\vec{x}_{spline}) = 0 \tag{2.7}$$

The cubic spline curve-fit is by definition continuous and smooth at the collocation point, so the constraint only needs to enforce the equations of motion for a well-behaved physical model that fulfills the two requirements from the beginning of this section.

The advantage of the cubic spline is that its high-order form (cubic polynomial) allows relatively large time steps to be used without inducing errors in the solution to the equations of motion. However, it is not as efficient an algorithm as those presented later because it requires invoking the equations of motion at the collocation point as well as the design variables, which can be costly for complex simulations. It is presented here because it is prominent in the literature and used for the results in some of the preliminary work of this thesis.

2.1.2 Trapezoidal Form

The trapezoidal form of the collocation constraint, from [7], is a first-order curve-fit of the state variables, as shown in figure 2.3. The constraint is a simple and intuitive method for equations in state form (like equation 2.1), with only first order time derivatives. Recall that in this form of the equations of motion no distinction is made between position states and velocity states, to create a more general set of dynamic equations. Equation 2.8 gives the curve fit for the trapezoidal constraint, which is a first-order Taylor series expansion of the state variables.

$$x(t) = x_i + (t - t_i)\dot{x}_i \tag{2.8}$$

The curve fit is a line through the state variable with a slope given by the state derivative from the equations of motion. The constraint specifies that the state calculated with an Euler step forward to the collocation point must equal the state calculated by taking an Euler step from the next design variable back to the collocation point. Equation 2.9 gives the trapezoidal collocation constraint.

$$\vec{c}_i = \vec{x}_i + \frac{dt}{2}\dot{\vec{x}}_i - (\vec{x}_{i+1} - \frac{dt}{2}\dot{\vec{x}}_{i+1}) = 0 \tag{2.9}$$

The trapezoidal constraint incorporates the equations of motion into its curve-fit of the state variables so that the system is always in dynamic equilibrium at the design-variable time points. The constraint then only has to ensure that the time history is continuous for an accurate solution to the equations of motion.

The low-order fit of the state variables in the trapezoidal constraint means that smaller time steps must be used to solve the equations of motion accurately, compared to a higher-order method such as the cubic-spline. The simplicity of the constraint does offset this computational penalty somewhat. The constraint is especially simple when the transformation of equation 2.3 is used, because for position state variables

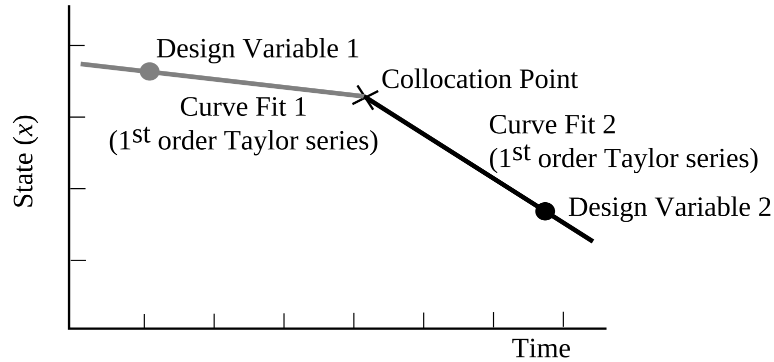

Figure 2.3: Trapezoidal Collocation Constraint

(half of the collocation constraints), both $\vec{x}$ and $\dot{\vec{x}}$ are design variables, so the equations of motion do not need to be solved to calculate the constraint, and even more importantly the gradient of the constraint is a simple analytic function of the design variables. The derivatives of the velocity state variables are functions of the equations of motion, so the gradients of the collocation constraints involving velocity states are therefore functions of the equations of motion as well, so that gradient calculations are not trivial for this half of the design variables.

2.1.3 Taylor Series Collocation Constraint

A general inconsistency of collocation constraints based on the state form of the equations of motion (equation 2.1) is that curve-fits of the same order are used for both the position states and the velocity states, even though the velocity is the derivative of the position and therefore should be modeled with a lower order fit. Converting equations of motion to a set of first-order state equations in time using equation 2.3 is not necessary for collocation, and some information is lost in the transformation. In order to address this inconsistency, the Taylor series collocation constraint, as seen in figure 2.4, was created, for use on equations of motion in the natural form when Newton's second law is applied (equation 2.2). The second-order Taylor series constraint expands the trapezoidal constraint into a higher order form by including one more term in the series expansion, as equation 2.10 shows.

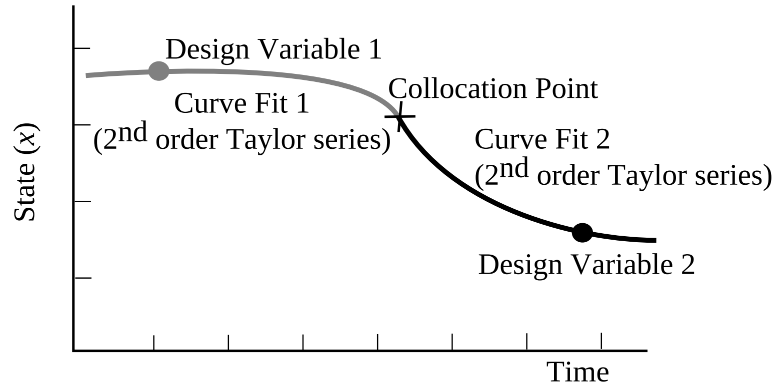

Figure 2.4: Second Order Taylor Series Collocation Constraint

$$x(t) = x_i + (t - t_i)\dot{x}_i + \frac{(t - t_i)^2}{2}\ddot{x}_i \tag{2.10}$$

The Taylor series collocation constraints enforce continuity of the position variables $\vec{x}$, using equation 2.11, and continuity of their derivatives, the velocity variables $\dot{\vec{x}}$, (whose curve-fit is found by differentiating equation 2.10), with equation 2.12.

$$\vec{c}_{(i,1)} = \vec{x}_i + \frac{dt}{2}\dot{\vec{x}}_i + \frac{dt^2}{8}\ddot{\vec{x}}_i - (\vec{x}_{i+1} - \frac{dt}{2}\dot{\vec{x}}_{i+1} + \frac{dt^2}{8}\ddot{\vec{x}}_{i+1}) = 0 \tag{2.11}$$

$$\vec{c}_{(i,2)} = \dot{\vec{x}}_i + \frac{dt}{2}\ddot{\vec{x}}_i - (\dot{\vec{x}}_{i+1} - \frac{dt}{2}\ddot{\vec{x}}_{i+1}) = 0 \tag{2.12}$$

The Taylor series constraint not only makes the curve-fits more consistent, the higher-order fit allows larger time steps to be used in the simulation without losing accuracy. Figure 2.5 shows how solutions to equations of motion for a representative problem converge as the step size decreases. These results are from the simulation of an aileron's response to position commands using a second-order dynamic model, similar to the one described in Chapter 4. Solutions to the equations of motion from integration using a fourth-order Runge-Kutta scheme, and from collocation with the

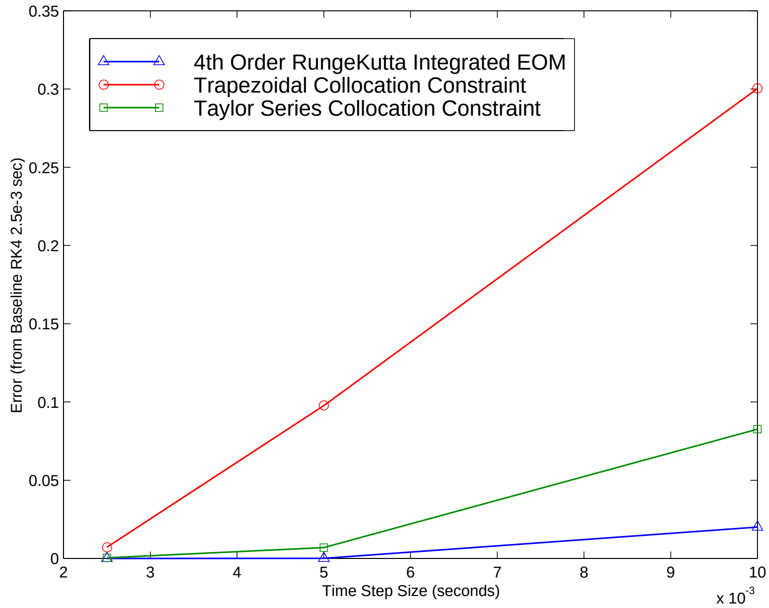

Figure 2.5: Error of Trapezoidal and Taylor Series Constraints for Aileron Response

Taylor series constraint have converged on a solution with a time step of 5×10^{-3} seconds, while the trapezoidal constraint needs a step of 2.5×10^{-3} to match the results. Unlike the cubic spline constraint, which is also higher order, the Taylor series constraint uses the same number of evaluations of the equations of motion as the lower-order trapezoidal constraint. The Taylor series constraint does not have the simple analytic constraint gradient that the trapezoidal method sometimes has, but the benefit of the smaller optimization problem due to large time-steps favors the more accurate Taylor series constraint.

2.2 Stability Constraint

Analyzing a system in the time domain rather than in the frequency domain allows the simulation to include arbitrary levels of nonlinearity, giving the designer more freedom to incorporate accurate models. The frequency domain analysis has a significant advantage, however, in that the stability and damping of linear systems is easy to quantify. Imposing the condition that a design's response must maintain a certain

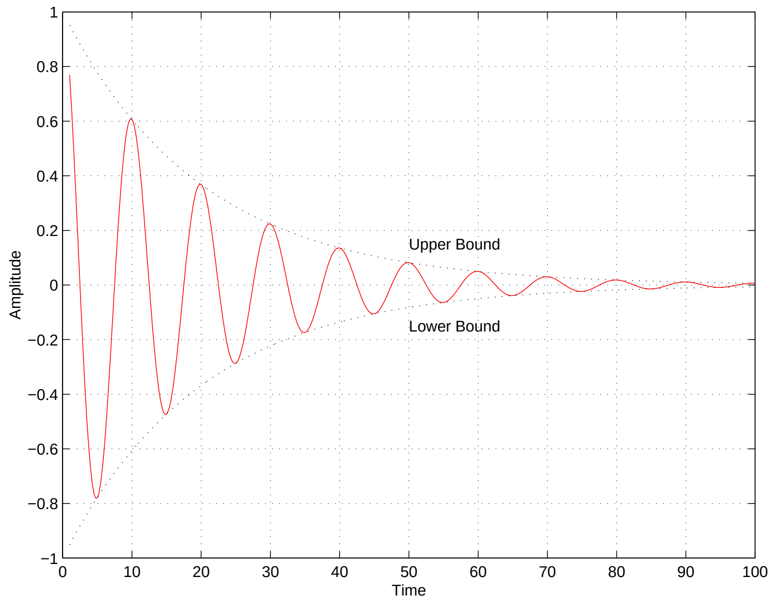

Figure 2.6: Envelope of Motion for Linear System

level of damping, or at least not be divergent, is quite straightforward with a linear simulation, and while it can be possible with a time-based simulation to qualitatively tell whether the motion is damped or stable, quantifying the results is more difficult.

The damped oscillation of a linear system has an exponentially-decaying amplitude proportional to $e^{-\lambda t}$, which defines an "envelope" for the motion. The maxima and minima of each oscillation follow this envelope, as figure 2.6 shows. In the frequency domain, the envelope and the level of damping are specified by a single number, λ. For a time-domain solution, the envelope that determines whether the motion is diverging or converging could be determined by finding the maxima and minima of the motion and using these values to determine an approximate level of damping, but this is an involved procedure, and not nearly as elegant as a frequency-domain solution.

Fortunately for the time domain, the collocation method has an intuitive and computationally simple method to control the state variable amplitudes. Using this technique, the level of damping in the system can be constrained and divergence can be prevented. In the collocation method, every state variable at every time point in the simulation is an optimization design variable, and since in most optimization algorithms upper and lower bounds may be placed on design variables, these upper and lower bounds on the state variables may be used to force the time history of the simulation to lie within any "envelope" desired. It is important to note that the stability constraint does not specify just one bound for each state variable, but bounds on each state variable at every time point in the simulation, so that the bounds may change in time. For example, if the dotted lines in figure 2.6, that indicate the exponential envelope of the linear motion, were used as the upper and lower bounds for this state variable, the collocation optimizer could not converge until the motion had at least the level of damping specified by the bounds. The bounds are specified for each point in the time history, so they can be large initially to allow transient motion and become smaller later to force the motion to be damped. The constraint is a computationally inexpensive addition to the numerical optimization problem because bound constraints do not add new rows to the constraint Jacobian; in fact the addition of bounds may even increase the optimization speed by narrowing the search area [24].

Bounds on the design variables have been used previously in collocation-based optimization of spacecraft [8] and aircraft [5] to ensure that flight path constraints, such as reaching a certain altitude at a certain time, are satisfied. The present work is the first to apply it over the entire simulation for dynamic stability. Chapter 3 uses the collocation stability constraint with a wing structural optimization to prevent flutter, and compares the results to a linear analysis, showing that it is an effective and robust constraint. The stability constraint is used for the work of Chapters 4 and 5 as well, in order to optimize the systems while achieving well-damped motion.

2.3 Implementation Issues

Numerical optimization tends to be a slow, computationally-intensive process. Collocation problems are no exception, so attention must be paid to the efficiency of the solution algorithm. With a bit of planning, the architecture of the collocation optimization problem may be exploited so that solutions can be reached with reasonable computational cost. The sparse nature of the constraint Jacobian is easily exploited with two techniques: with sparse optimization methods, and with sparse finite-difference gradients. The collocation method is highly parallelizable, and multiple processors can be used to speed up computation time, because the collocation time-simulation is not a sequential calculation. The equations of motion calculations used to evaluate the collocation constraints must be as efficient as possible, and although this is highly problem dependent some guidelines for achieving a manageable problem can be stated.

2.3.1 Sparse Optimization

Sparse optimization, as described in [45] and [24], is very important to making the collocation method a practical technique. The collocation constraints, given in section 2.1, create a sparse Jacobian as long as the equations of motion at one time point depend only on the state variables at nearby time points. This is by far the most common case; in fact quite often the equations of motion only depend on one set of state variables, those of the previous time-step. With sparse optimization, only the nonzero elements of the Jacobian matrix are stored in memory, reducing the memory requirement for the optimization problem significantly. The matrix algebra used to solve the optimization problem also exploits the knowledge of the Jacobian sparsity pattern, so that no time is wasted multiplying or adding elements that are zero. With Jacobian matrices for typical collocation problems containing tens of millions of elements, of which over 99% are usually zero, the computational time and storage saved with sparse optimization techniques can be enormous.

The optimizer used to solve the sparse, constrained optimization problems in this thesis is MINOS [45]. It is a well-established and tested code that proved to be an effective workhorse for this research. It has recently been replaced by SNOPT [22], a new, more efficient optimizer that promises even faster collocation results.

2.3.2 Sparse Finite Difference Gradients

The sparsity of the Jacobian can be exploited in another way. Normally, when calculating a gradient matrix such as the Jacobian by finite-differences, each design variable must be perturbed individually to calculate the derivative, according to equation 2.13, which requires one evaluation of the constraints for each design variable if one-sided differences are used.

$$\frac{\partial c_i}{\partial x} = \frac{c_i(x + dx, y) - c_i(x, y)}{dx} \tag{2.13}$$

If more than one variable is perturbed at once, the difference will involve the sum of two partial derivatives, as equation 2.14 shows. This is normally not a useful result.

$$\frac{\partial c_i}{\partial x}dx + \frac{\partial c_i}{\partial y}dy = c_i(x + dx, y + dy) - c_i(x, y) \tag{2.14}$$

If some of these partial derivatives of constraints are known to be zero, more than one design variable can be perturbed at a time and the difference will only involve one partial derivative. For example, if $\frac{\partial c_1}{\partial x} = 0$ and $\frac{\partial c_2}{\partial y} = 0$ then equation 2.14 can be rewritten as:

$$\frac{\partial c_1}{\partial y} = \frac{c_1(x + dx, y + dy) - c_1(x, y)}{dy} \tag{2.15}$$

$$\frac{\partial c_2}{\partial x} = \frac{c_2(x + dx, y + dy) - c_2(x, y)}{dx} \tag{2.16}$$

The nonzero derivatives of both constraints can be calculated by perturbing the variables only once. This idea is used in [15] which presents an algorithm (and source code) to find the best combination of variables to perturb in each constraint evaluation in order to find the sparse Jacobian matrix with the fewest constraint evaluations.

For example, the aeroservoelastic design problem of Chapter 5, with 8989 design variables and 8982 constraints, has only 385,902 nonzero Jacobian elements, meaning only 0.48 % of the Jacobian elements are nonzero. One-sided finite differencing for the Jacobian would take 8990 constraint evaluations for a full Jacobian, but can be accomplished with only 43 evaluations by exploiting the highly sparse Jacobian structure. Obviously, exploiting the sparsity in this manner will result in a much faster-running optimization problem.

2.3.3 Architecture for Parallel Calculation

One way to speed up time-consuming problems such as collocation-based optimization is to spread the calculation load over multiple processors. In order for the parallelization to save time, the processors must be able to work as independently as possible, with a minimum of time spent waiting for communications or data from the other processors.

Simulations in the time domain are by nature sequential calculations. They have a beginning and an end, and integrating the equations of motion in time involves repeating the same calculations for many time points, from the initial conditions to the final time, with the output from one time point becoming the input to the next. While integration of the equations of motion is repetitive, spreading the calculations over many processors will not make it faster because the integration must be performed in temporal order, so that only one processor at a time can be calculating.

Solving the time-simulation with the collocation method, however, transforms the problem to a process that is very suitable for parallel processing. At every design iteration, the entire time history is specified at once. The collocation constraints that force the optimizer to solve the equations of motion can be evaluated in any order or in parallel. Figure 2.7 compares the problem architectures for optimizing with integrated equations of motion and with collocation. Integrating the equations of motion requires that a large, sequential calculation be performed for every design iteration, while the collocation method performs a great many small calculations which may be parallelized. Braun solved a lunar ascent problem with the collocation constraints

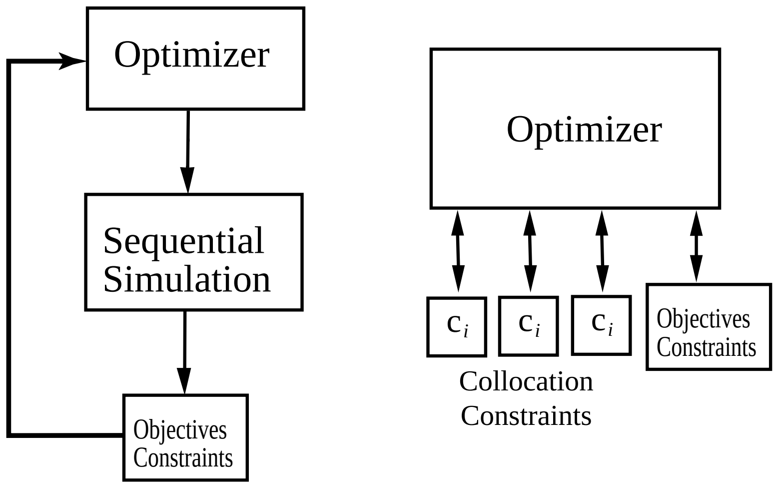

Figure 2.7: Optimization with Integrated Equations of Motion and Collocation

broken into independent subproblems in [12]. With the option of parallel processing, the idea of using complex, nonlinear analyses in collocation-based optimization becomes much more realistic.

2.3.4 Efficient Equations of Motion

When optimizing with the collocation method, the majority of computation time is spent evaluating the equations of motion in order to calculate the collocation constraints and their gradients. The way these equations of motion are posed can influence how easy they are for a collocation-based optimizer to solve, and often it is a simple manner to change the form of the equations of motion to greatly speed up the collocation solution process.

Both the number of design variables and the number of constraints in the collocation optimization problem are proportional to the number of degrees of freedom in the simulation to be solved. Reducing the number of degrees of freedom, when possible, will speed up both evaluating the constraints and the optimizer's search for the next design iteration. Often the number of degrees of freedom are unchangeable aspects of the simulation, set by physically realistic parameters. It is impossible to simulate the complete longitudinal motion of a rigid aircraft without the three longitudinal degrees of freedom, for example. Of course, if only the short-period motion was of interest, the altitude and velocity states could be fixed, saving perhaps a few thousand design variables.

Often the degrees of freedom are created by discretizing a continuous system. The nodes on a finite-element structure or aerodynamic grid are examples of this type of variable. The discretization usually involves a trade between grid coarseness and solution accuracy, so that the designer must be sure the discretization is fine enough to give accurate results no matter what combination of design variables the optimizer chooses to evaluate, but must not be so fine that the extra degrees of freedom will bog down the calculation.

Linear transformations can reduce the number of degrees of freedom in a discretized system. Often there is a way to find a reduced-basis set that can describe the important motion. Modal analyses of a finite-element structure can reduce the number of degrees of freedom to a few important linear combinations of the thousands of physical deflections. This transformation can have a huge impact on the calculation time of the collocation solution. Modal analysis is restricted, of course, to linear systems, a limitation not imposed by the collocation method, but often nonlinear systems have linear pieces that can be treated this way.

The number of design variables and the number of constraints in a collocation method optimization problem are proportional to the number of time-steps in the simulation as well as the number of degrees of freedom. Reducing the number of time-steps will speed up the solution process significantly by making the optimization problem smaller. The number of time steps can be reduced by solving the simulation over a smaller span of time, or by increasing the step size between points. The length of the simulation is very problem dependent, but it is important to make sure it is as short as possible. For example, if the motion is highly damped, there is no need to solve for the unchanging motion once equilibrium has been reached. Conversely, lightly damped systems may need long simulations in order to see if they are converging or diverging. The step size must be small enough that the equations

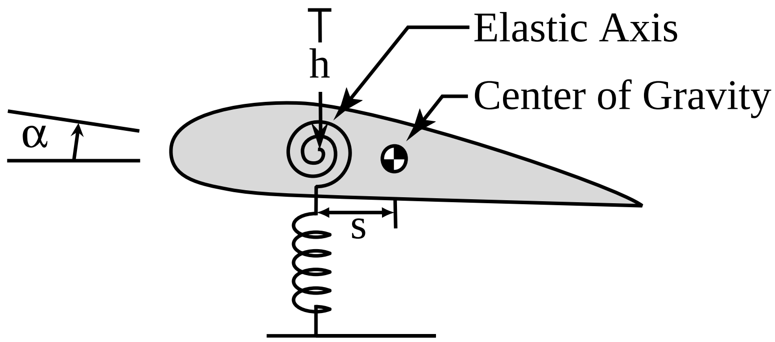

Figure 2.8: Typical Section Model

of motion are solved accurately. Eliminating high-frequency motion, either by eliminating high-frequency degrees of freedom or with modal analysis, is one way that the time-step size can be increased without inducing errors, and may have little impact on the accuracy of the results if the high-frequency motion is highly damped or small in amplitude.

With careful planning and knowledge of the inner workings of the simulation to be used with the collocation method, it is possible to make the solution process efficient enough to solve without resorting to supercomputers, at least for problems such those solved in this thesis. The time spent checking to see if any unnecessary complications have been added before letting the optimizer loose on the problem will always pay off handsomely in reduced computer time.

2.4 Typical Section Example

To tie everything together with an example, a simple aeroelastic system will be presented and optimized with collocation. The typical section, described in [11], is a two dimensional wing section with two degrees of freedom. The goal will be to find the maximum airspeed at which the section can be operated without fluttering.

2.4.1 Equations of Motion

The typical section, shown in figure 2.8, is an airfoil section that is free to move vertically, in the h direction, and rotate, in the α direction. It is attached to a spring that resists motion in both degrees of freedom, with spring constants K_α for rotation, and K_h for vertical deflection. The section has mass m and moment of inertia I_α, and the center of gravity is positioned a distance s from the elastic axis, where the springs are attached.

The equations of motion (from [11]) are given in equations 2.17 and 2.18, where $S_\alpha = ms$, L is the section lift, and M is the section moment about the elastic axis.

$$m\ddot{h} + S_\alpha \ddot{\alpha} + K_h h = -L \tag{2.17}$$

$$S_\alpha \ddot{h} + I_\alpha \ddot{\alpha} + K_\alpha \alpha = M \tag{2.18}$$

For the sake of simplicity in this example, the aerodynamic lift, L, and pitching moment, M, will be calculated using a quasi-steady analysis, so that the aerodynamic forces are only dependent on the current state of the system: h, α, $\dot{h}$, and $\dot{\alpha}$ (and fluid properties). A more accurate aerodynamic model could be used, of course, although the sparsity of the Jacobian can be affected, something that will be discussed in section 2.4.6.

The equations of motion can be transformed into the form of equation 2.19, the form that is most useful to solve them both by integration or with collocation. A subroutine to solve the equations in this form will be the backbone of the collocation simulation.

$$\{\ddot{h}, \ddot{\alpha}\} = f(h, \dot{h}, \alpha, \dot{\alpha}) \tag{2.19}$$

For integration of the equations of motion and for cubic-spline or trapezoidal constraints, the first-order form of the equations of motion, such as equation 2.1, is necessary, using a transformation of the form:

$$\vec{x} = \{h, \dot{h}, \alpha, \dot{\alpha}\} \tag{2.20}$$

While writing the routines to calculate the equations of motion for the simulation, it is usually worthwhile to create a routine that solves them using integration, even if a collocation-based solution is ultimately desired. For a single solution to the equations of motion, it is much faster to use integration instead of collocation, and it is handy to have such routines for debugging the simulation and determining parameters such as the length of the simulation to use, initial conditions, and to check the "envelope" of the motion for the stability constraint. Collocation is best left for problems that use optimization, where the iterations to solve the equations of motion and the iterations to find the optimum design can proceed together.

As with all optimization problems, it pays to be very knowledgeable about the problem to be solved so that the optimizer can be prevented from exploiting weaknesses in the analyses. Integrating the equations of motion can give valuable experience with the simulation. After examining the motion of the typical section model of this example, a 45 second simulation with 150 time points for a time-step size of 0.2956 seconds was chosen as a good combination of simulation length, discretization error, and optimization problem size.

2.4.2 Objective

The typical section's dynamic stability is dependent on the airspeed at which it operates. The objective for this optimization will be to find the maximum airspeed for which the motion of the typical section does not diverge. The airspeed, v, will be a design variable in the optimization problem as well as the objective.

2.4.3 Design Variables

The design variables are the velocity, v, plus the state variables $\{h, \dot{h}, \alpha, \dot{\alpha}\}$ at all time points except the initial conditions, 149 time points total. Thus there are 597 design variables in this problem. The design variables in vector form are:

$$\{v, h_2, \dot{h}_2, \alpha_2, \dot{\alpha}_2, h_3, \dot{h}_3, \alpha_3, \dot{\alpha}_3, \ldots h_{150}, \dot{h}_{150}, \alpha_{150}, \dot{\alpha}_{150}\}$$

where the initial conditions are $\{h_1, \dot{h}_1, \alpha_1, \dot{\alpha}_1\}$.

2.4.4 Collocation Constraints

There are 149 collocation points between the 150 time points of the time history. To ensure that all four states follow the equations of motion at every collocation point requires 596 collocation constraints.

The following pseudo-code calculates the collocation constraints using cubic-spline, trapezoidal, and Taylor series constraints.

$j = 1$

for $i = 1$ to 149 do

$[\ddot{h}_i, \ddot{\alpha}_i] = f(h_i, \alpha_i, \dot{h}_i, \dot{\alpha}_i)$

$[\ddot{h}_{i+1}, \ddot{\alpha}_{i+1}] = f(h_{i+1}, \alpha_{i+1}, \dot{h}_{i+1}, \dot{\alpha}_{i+1})$

Trapezoidal Constraint:

$$c_j = h_i + \frac{dt}{2}\dot{h}_i - \left(h_{i+1} - \frac{dt}{2}\dot{h}_{i+1}\right)$$
$$c_{j+1} = \alpha_i + \frac{dt}{2}\dot{\alpha}_i - \left(\alpha_{i+1} - \frac{dt}{2}\dot{\alpha}_{i+1}\right)$$
$$c_{j+2} = \dot{h}_i + \frac{dt}{2}\ddot{h}_i - \left(\dot{h}_{i+1} - \frac{dt}{2}\ddot{h}_{i+1}\right)$$
$$c_{j+3} = \dot{\alpha}_i + \frac{dt}{2}\ddot{\alpha}_i - \left(\dot{\alpha}_{i+1} - \frac{dt}{2}\ddot{\alpha}_{i+1}\right)$$

Taylor Series Constraint:

$$c_j = h_i + \frac{dt}{2}\dot{h}_i + \frac{dt^2}{8}\ddot{h}_i - \left(h_{i+1} - \frac{dt}{2}\dot{h}_{i+1} + \frac{dt^2}{8}\ddot{h}_{i+1}\right)$$
$$c_{j+1} = \alpha_i + \frac{dt}{2}\dot{\alpha}_i + \frac{dt^2}{8}\ddot{\alpha}_i - \left(\alpha_{i+1} - \frac{dt}{2}\dot{\alpha}_{i+1} + \frac{dt^2}{8}\ddot{\alpha}_{i+1}\right)$$

$$c_{j+2} = \dot{h}_i + \tfrac{dt}{2}\ddot{h}_i - \left(\dot{h}_{i+1} - \tfrac{dt}{2}\ddot{h}_{i+1}\right)$$
$$c_{j+3} = \dot{\alpha}_i + \tfrac{dt}{2}\ddot{\alpha}_i - \left(\dot{\alpha}_{i+1} - \tfrac{dt}{2}\ddot{\alpha}_{i+1}\right)$$

Cubic Spline Constraint:

State at collocation point:

$$h_c = \tfrac{h_i + h_{i+1}}{2} + \tfrac{dt}{8}(\dot{h}_i - \dot{h}_{i+1})$$
$$\alpha_c = \tfrac{\alpha_i + \alpha_{i+1}}{2} + \tfrac{dt}{8}(\dot{\alpha}_i - \dot{\alpha}_{i+1})$$
$$\dot{h}_c = \tfrac{\dot{h}_i + \dot{h}_{i+1}}{2} + \tfrac{dt}{8}(\ddot{h}_i - \ddot{h}_{i+1})$$
$$\dot{\alpha}_c = \tfrac{\dot{\alpha}_i + \dot{\alpha}_{i+1}}{2} + \tfrac{dt}{8}(\ddot{\alpha}_i - \ddot{\alpha}_{i+1})$$

Invoke equations of motion:

$$[\ddot{h}_c, \ddot{\alpha}_c] = f(h_c, \alpha_c, \dot{h}_c, \dot{\alpha}_c)$$

Derivative of splines at collocation point:

$$\dot{h}_{spline} = -\tfrac{3}{2}\tfrac{(h_i - h_{i+1})}{dt} - \tfrac{\dot{h}_i + \dot{h}_{i+1}}{4}$$
$$\dot{\alpha}_{spline} = -\tfrac{3}{2}\tfrac{(\alpha_i - \alpha_{i+1})}{dt} - \tfrac{\dot{\alpha}_i + \dot{\alpha}_{i+1}}{4}$$
$$\ddot{h}_{spline} = -\tfrac{3}{2}\tfrac{(\dot{h}_i - \dot{h}_{i+1})}{dt} - \tfrac{\ddot{h}_i + \ddot{h}_{i+1}}{4}$$
$$\ddot{\alpha}_{spline} = -\tfrac{3}{2}\tfrac{(\dot{\alpha}_i - \dot{\alpha}_{i+1})}{dt} - \tfrac{\ddot{\alpha}_i + \ddot{\alpha}_{i+1}}{4}$$

Constraints:

$$c_j = \dot{h}_c - \dot{h}_{spline}$$
$$c_{j+1} = \dot{\alpha}_c - \dot{\alpha}_{spline}$$
$$c_{j+2} = \ddot{h}_c - \ddot{h}_{spline}$$
$$c_{j+3} = \ddot{\alpha}_c - \ddot{\alpha}_{spline}$$

$j = j + 4$

end do

Comparing the different constraints used, the similarity between the trapezoidal and Taylor series constraints becomes clear, and the extra calculations needed by the cubic spline constraint can be seen.

	v	h_2	α_2	$\dot{h}_2$	$\dot{\alpha}_2$	h_3	α_3	$\dot{h}_3$	$\dot{\alpha}_3$	h_4	α_4	$\dot{h}_4$	$\dot{\alpha}_4$	h_5	α_5	$\dot{h}_5$	$\dot{\alpha}_5$	$\dots$
c_1	$+^*$	$+$	$+^*$	$+$	$+^*$													
c_2	$+^*$	$+^*$	$+$	$+^*$	$+$													
c_3	$+$	$+$	$+$	$+$	$+$													
c_4	$+$	$+$	$+$	$+$	$+$													
c_5	$+^*$	$+$	$+^*$	$+$	$+^*$	$+$	$+^*$	$+$	$+^*$									
c_6	$+^*$	$+^*$	$+$	$+^*$	$+$	$+^*$	$+$	$+^*$	$+$									
c_7	$+$	$+$	$+$	$+$	$+$	$+$	$+$	$+$	$+$									
c_8	$+$	$+$	$+$	$+$	$+$	$+$	$+$	$+$	$+$									
c_9	$+^*$					$+$	$+^*$	$+$	$+^*$	$+$	$+^*$	$+$	$+^*$					
c_{10}	$+^*$					$+^*$	$+$	$+^*$	$+$	$+^*$	$+$	$+^*$	$+$					
c_{11}	$+$					$+$	$+$	$+$	$+$	$+$	$+$	$+$	$+$					
c_{12}	$+$					$+$	$+$	$+$	$+$	$+$	$+$	$+$	$+$					
c_{13}	$+^*$									$+$	$+^*$	$+$	$+^*$	$+$	$+^*$	$+$	$+^*$	
c_{14}	$+^*$									$+^*$	$+$	$+^*$	$+$	$+^*$	$+$	$+^*$	$+$	
c_{15}	$+$									$+$	$+$	$+$	$+$	$+$	$+$	$+$	$+$	
c_{16}	$+$									$+$	$+$	$+$	$+$	$+$	$+$	$+$	$+$	
$\vdots$	$+^*$													$+$	$+^*$	$+$	$+^*$	$+$

(Elements with * are zero for the trapezoidal constraint)

Table 2.1: Jacobian Sparsity Pattern

2.4.5 Objective Gradient

The objective gradient is very simple, since the objective function is a design variable. No finite-differencing is necessary to find it, as the analytic gradient is simply $\{1, 0, 0, 0, \dots\}$.

2.4.6 Jacobian

The Jacobian is the gradient of the constraints with respect to the design variables. Each column corresponds to a design variable, and each row a constraint. For the typical section problem, the Jacobian is an array with 596 rows and 597 columns. The first column is not sparse, because every constraint depends on the airspeed, which is fundamental to the forces in the equations of motion. The rest of the Jacobian is quite sparse, because every constraint depends only on the states at one time point. Table 2.1 shows the sparsity pattern of the Jacobian. Notice the elements marked with an asterisk that are zero in the case of the trapezoidal constraint because the trapezoidal constraint does not invoke the equations of motion for these variables.

The block diagonal form (with blocks the size of the number of state variables) is typical of the sparsity pattern for collocation optimization problems. The pattern shown assumes that the state derivatives from the equations of motion depend only upon the current state of the system. This assumption requires some simplification of the aerodynamic model, because the wake of the airfoil should reflect the time history of the lift. Fully incorporating the unsteady wake for the entire time history would change the form of the Jacobian to a triangular matrix, and could make the optimization problem too large for the memory size and processing speed of typical computers available today. Of course, the primary effect of the wake is made by the near wake [34], and approximations of the far wake can be used to reduce the dependency and make the Jacobian sparse again.

2.4.7 Stability Constraints

To keep the design from fluttering, upper and lower bounds must be placed on at least one of the states ($h, \dot{h}, \alpha$, or $\dot{\alpha}$) that will force the motion to be damped. These bounds are dependent on the initial conditions, because the amplitude of the motion depends on the initial energy contained in the system. The best approach for setting the bounds in this type of simulation is to allow them to be large initially so that energy may be exchanged between modes in transient motion, and make the bounds as tight as necessary in the later portions of the time history to enforce damping. A small (but not insignificant) level of damping is placed on the design variables h and α in this problem with a set of bounds that varies linearly from double the initial condition amplitude at the start of the simulation to 90% of the initial condition by the end. Figure 2.9 shows the motion and the bounds.

2.4.8 Results: Finding Flutter Speed

Figure 2.9 shows the motion after optimizing to find the maximum airspeed that produces a response within the bounds on h and α. Note that the critical value of α that touches the bound is not at the maximum amplitude of the oscillation, because it is at the end of the simulation. If the simulation was run a little longer, the velocity would have to be decreased further to keep the motion within the bounds. For tapering bounds such as we have applied in this problem, the maximum final

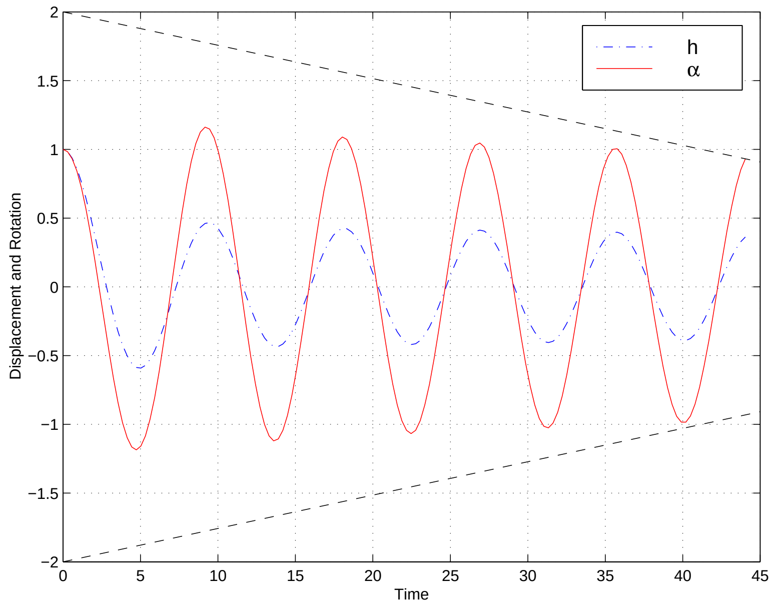

Figure 2.9: Typical Section Simulation Results

amplitude can vary a bit, depending on the phase and frequency of the motion. This variation is small when the slope of the bounds is small, and is easily controlled by the designer.

The final airspeed from the optimization results in stable motion. The damping can be seen in the decreasing oscillation amplitude of h and α in the time history of figure 2.9.

2.5 Conclusions and Summary

This chapter showed the procedures to incorporate a time-based simulation into an optimization problem using collocation. The process involves adding the state variables to the set of design variables, enforcing physically realistic motion with the collocation constraints, and optionally constraining the motion to be damped with a stability constraint.

The collocation constraint can take many forms, and a search of collocation literature will show that the most common are the cubic spline form, with accuracy suitable for large time steps, and the computationally streamlined trapezoidal constraint. These constraints were explained along with the Taylor series constraint for second order equations of motion, a new form that was shown to have computational advantages similar to the trapezoidal constraint and step sizes as large as the cubic spline constraint.

Applying upper and lower bounds to the design variables in the optimization problem can create a time-varying "envelope" that constrains the motion to be damped and prevents the optimizer from designing an unstable system. This constraint is in a form that is intuitive for an engineer to apply and adds almost no computational burden to the optimization problem.

This chapter also discussed aspects of the optimization problem that affect its computational efficiency, including the sparsity of the Jacobian, the parallel nature of the calculations involved with collocation, and simple modifications to the simulation that can drastically improve the computation time of a problem. The collocation method creates a large optimization problem, and use of these techniques is mandatory to keep the solution times reasonable.

With the information presented in this chapter, the reader should be able to set up a problem such as the typical section optimization example given in this chapter. Starting with the equations of motion, the design variables and their gradient were stated, the collocation constraints were formed and their Jacobian illustrated. The stability constraint was applied to keep the motion damped and the airspeed was then maximized subject to the flutter constraint.

The methods of the collocation method should now be firmly established, creating a strong framework to support the results of the following chapters. The applicability of collocation for optimization of time-based aeroelastic and aeroservoelastic systems will now be explored further, using the techniques presented here.

Chapter 3

Collocation for Aeroelastic Optimization

3.1 Introduction

Dynamic aeroelastic analyses are complex calculations that may be performed hundreds of times in the course of a wing design optimization, placing a premium on efficient computational techniques. If the stability of the structure is in question, and linear equations of motion are applicable, frequency domain analyses are often the best approach. For nonlinear systems, however, a time simulation may be the only practical technique. The collocation method is an appealing algorithm for solving this class of problem.

Collocation methods for dynamic simulation have not been applied to aeroelastic simulations in the past, although the combination of aerodynamic analyses, which tend to be nonlinear, with structural dynamics and stability constraints is well-suited to collocation. In this chapter a wing flutter response is simulated using a finite element model. The optimizer minimizes the wing weight while designing a structure that is stable at the specified flight conditions. Stability results from the collocation method are compared with frequency-domain calculations by linearizing the equations of motion of the aeroelastic system. The wing model is the 'typical jet transport' from [11], and although it is hardly typical of today's jet transports it is a nicely documented aeroelastic model with which to start.

Capturing stability information with a time simulation is not as straightforward as with an eigenvalue analysis. Limiting motion to a stable range using upper and lower bounds on the state design variables is investigated in this chapter, and compared with results from eigen-analysis of the linearized system. This stability constraint is physically intuitive, and can be integrated into the collocation optimization problem with very little added computational expense.

A new, simple, finite-element dynamic analysis, suitable for preliminary design and fast enough for optimization, was written for use with the collocation simulation algorithm. This code, written in the MATLAB programming language, is designed to solve a limited class of aeroelastic problems on high aspect-ratio wings. As in most analysis codes, there is a trade between generality and simplicity, and here the trade is made in favor of simplicity and computational speed. The linear equations of motion for the aeroelastic wing can be written:

$$[M]\ddot{\vec{dx}} + [K]\vec{dx} = \vec{F_o} + [F_x]\vec{dx} + [F_{\dot{x}}]\dot{\vec{dx}} \tag{3.1}$$

The finite-element program calculates the matrices in equation 3.1, as well as transformation matrices between the aerodynamic and structural coordinate systems. $\vec{F_o}$, $[F_x]$, and $[F_{\dot{x}}]$ are calculated in the aerodynamic routines, while $[M]$ and $[K]$ come from the structural calculations. Routines for static deflection, eigen stability and modal decomposition, and dynamic simulation have been written.

3.2 Aerodynamic Model

The wing aerodynamics are computed with a quasi-steady vortex-lattice panel code. The wing is paneled with one chordwise panel whose computed forces include a pitching moment correction from two-dimensional unsteady theory. The beam elements and aerodynamic panels are located together so that each horseshoe vortex is associated with a beam element, which greatly simplifies the aero-structural-dynamic

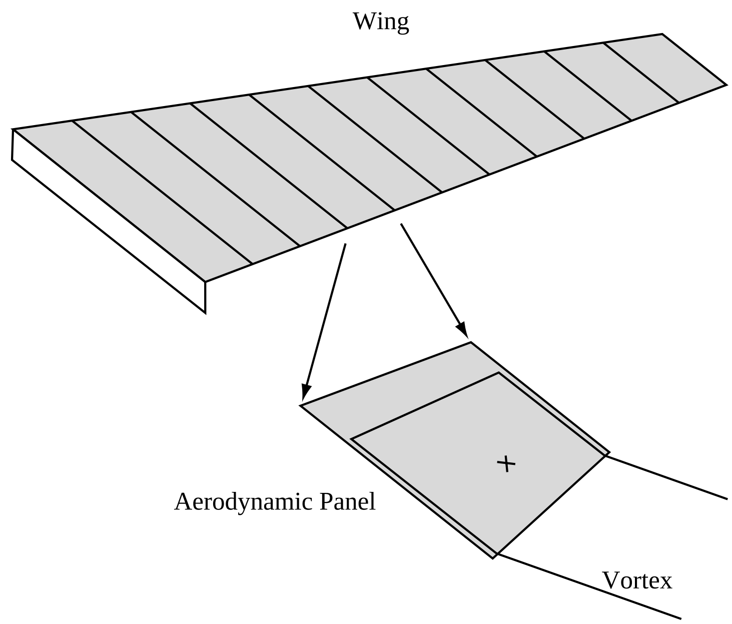

Figure 3.1: Aerodynamic Model

coupling. In order to be able to compare the collocation results to a frequency-domain stability analysis, the aerodynamic forces are linearized in the results of this chapter. This aerodynamic model is able to accurately analyze high-aspect ratio wings where chordwise aerodynamics and chordwise dynamics can be safely neglected.

3.2.1 Panel Method Framework

The aerodynamic analysis is very similar to the program Linair [40] and other analyses such as those of Weissinger [56]. It is a vortex-lattice panel program with one chordwise panel and a streamwise wake, which makes a horseshoe vortex singularity distribution when the Kutta condition is enforced. Figure 3.1 shows the singularity distribution. The horseshoe vortex satisfies Laplace's equation for incompressible fluid flow; the boundary condition of flow tangency at the control point sets the vortex strength. The bound vortex is placed on the 1/4 chord, and the control point is on the 3/4 chord. Forces and moments are calculated from the Kutta-Joukowski law, with an unsteady moment correction that will be discussed later. The Prandtl-Glauert compressibility correction is added to accommodate subsonic Mach numbers.

The aerodynamic shape is defined by the x, y, z coordinates of the leading and trailing edges at the corners of the panels. Each panel has one horseshoe vortex. The coordinate system is the standard aerodynamic (not dynamic) system with $\hat{x}$ in the free-stream direction, $\hat{y}$ out the right wing, and $\hat{z}$ pointing up.

The aerodynamic code uses the velocity of the corner of each panel to calculate unsteady aerodynamic effects. The velocities are incorporated into the boundary condition by adding the velocity of the control point to the free stream flow through the control point; the velocity of the 1/4 chord at the center of the panel is added to the free stream for the force calculations.

3.2.2 Additional Stripwise Parameters

Vortex lattice codes with only one chordwise panel enforce the boundary condition with the bound vortex on the 1/4 chord and the control point on the 3/4 chord. In unsteady motion, the quasi-steady terms in the lift are accounted for correctly, but the change in pitching moment is not. For example, a pitch-up rotation of the panel about the mid-chord line increases the normal flow through the control point, causing a proportional increase in the bound vortex strength. Since the vortex is located at the quarter-chord, its force creates a pitching moment that adds to the pitch-up motion. This force is destabilizing, whereas the actual aerodynamic moment should damp this motion out. An additional two dimensional moment term of $\frac{M_y}{\Delta y} = -\pi U \frac{c^3}{8}\dot{\alpha}$, from quasi-steady two-dimensional theory, is added to the moment about the 1/4 chord to provide the proper pitching moment for this aerodynamic model. This moment correction is found by taking the quasi-steady section terms from equation 5-347 of [11].

3.2.3 Linearized Forces

The gradient terms for the linearized forces, $[F_x]$ and $[F_{\dot{x}}]$ in equation 3.1, are calculated by simply finite differencing results from perturbed inputs, which is much simpler to implement than solving for the closed form of the derivatives. For a linear code, the results of finite differencing are independent of step size and the point about which the differences are taken. The panel code used is very close to linear, so with finite differences about the static equilibrium, very accurate derivatives can be found. Transformation matrices change the gradient with respect to aerodynamic coordinates (leading and trailing edge point displacements) to the gradient with respect to the structural-dynamic coordinates (displacements and rotations of the leading edge, $\vec{dx}$). Small deflection assumptions have already been made in the structural analysis so that the linear transformations from aerodynamics to structures impart no further approximations.

3.3 Finite Element Model

3.3.1 Motivation for This Model

The finite element analysis is designed to use the same geometric description as the aerodynamic code. With one chordwise panel, the geometry in the aerodynamic analysis is defined in a one-dimensional fashion, with spanwise distributions of properties such as chord, twist, and sweep. The finite element model is similarly described by spanwise distributions of structural parameters such as beam areas and moments of inertia. A beam-based finite element model was chosen because of its ability to model the significant deflections of high aspect-ratio wings with a minimum number of degrees of freedom, while remaining consistent with the lack of chordwise specifications in the aerodynamic model.

The typical section model, described in [11] and Chapter 2, also influenced the concept of the basic structural model. The typical section model is a two-dimensional airfoil section attached to a rotary and linear spring. The stability of the typical section is dependent on the difference in location between the elastic axis and the center of gravity (see figure 3.2).

To extrude the cross-section of the typical section to a three-dimensional wing, the wing's structural spar is placed at a spanwise-varying percentage chord. In most finite element beam models, the elastic axis and the center of gravity are located on

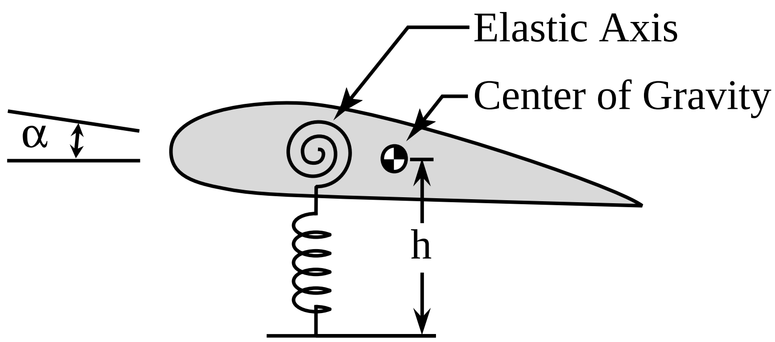

Figure 3.2: Typical Section Model

the same line, unlike the typical section. In the finite element model of this chapter, the static imbalance of the typical section is included by creating a super-element composed of two twelve-degree-of-freedom beams: one beam for the stiffness matrix placed at the elastic axis and one beam for the mass matrix placed at the center of gravity, with a rigid link (e.g. a wing rib) connecting them.

3.3.2 Structural Model

The super-elements, illustrated in figure 3.3, are assembled to build the cantilevered wing structural model used in the aeroelastic simulation. The two distinct beams that make up each element allow the stiffness and mass properties of the wing to be specified completely independently. The 'stiffness beam', located in the wing at a specified fraction of the chord, has no mass, with all mass concentrated on a zero stiffness 'mass beam' that is located independently of the stiffness beam. Beam properties are constant for each aero-structural element so that the overall structural parameters vary piecewise along the wing. The beam element stiffness and mass matrices are composed of the 12 degree of freedom (DOF) elements given in [48]. The beam members are free to bend, twist and shear in any of the 3 dimensions.

The end nodes of the stiffness and mass beams are connected rigidly by virtual ribs, so that one set of 12 displacements and rotations defines the movement of both beams. This allows the two beam system to be used to independently prescribe inertial and stiffness properties with the same number of degrees of freedom as a

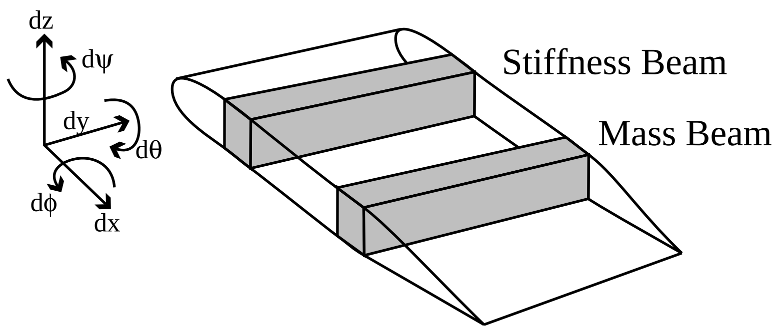

Figure 3.3: Structural Model

single beam model would have. The coordinates which describe the two beams' deflections are located at the leading edge of the aerodynamic panel. Each beam is located a specified percentage of the chord from the leading edge. A simple linear transformation changes the stiffness and mass matrices from [48], which use degrees of freedom at the beam ends, to matrices with degrees of freedom of the leading edge. The small deflection assumption implicit in the finite-element model allows the use of a linear transformation. It is the displacements and rotations of the leading edge nodes that make up the variable $\vec{dx}$ in equation 3.1 for this aeroelastic model.

Figure 3.3 shows the structural and mass beams in an aerodynamic panel, and the dynamic coordinates of one node, $\vec{dx}_i$.

3.4 Aeroelastic Equations of Motion

The structural and aerodynamic models used in this study create a linear set of equations, equation 3.1, repeated here for convenience:

$$[M]\ddot{\vec{dx}} + [K]\vec{dx} = \vec{F_o} + [F_x]\vec{dx} + [F_{\dot{x}}]\dot{\vec{dx}} \tag{3.2}$$

Where $\vec{dx}$ is the set of degrees of freedom of the wing nodes. Equation 3.2 is transformed into equation 3.4 with the definition of the static equilibrium $\vec{dx}_o$ in equation 3.3.

$$\vec{dx}_o = [K - F_x]^{-1}\vec{F}_o \tag{3.3}$$

$$[M]\ddot{\vec{dx}} + [K]\left(\vec{dx} - \vec{dx}_o\right) = [F_x]\left(\vec{dx} - \vec{dx}_o\right) + [F_{\dot{x}}]\dot{\vec{dx}} \tag{3.4}$$

The equation is then transformed into state vector form by defining the state variable vector, $\vec{\chi}$ in equation 3.5. The final, linear equations of motion are given in equation 3.6, where $[D]$ is a matrix composed of the matrices in equation 3.2 and $[I]$, the identity matrix.

$$\vec{\chi} = \begin{Bmatrix} \vec{dx} - \vec{dx}_o \\ \dot{\vec{dx}} \end{Bmatrix} \tag{3.5}$$

$$\dot{\vec{\chi}} = [D]\vec{\chi} \tag{3.6}$$

$$[D] = \begin{bmatrix} 0 & [I] \\ [M]^{-1}[K - F_x] & [M]^{-1}[F_{\dot{x}}] \end{bmatrix} \tag{3.7}$$

Some simulations in this chapter used modal analysis to reduce the degrees of freedom from those of the physical nodes to the first few dynamic modes, η. The eigenvectors, $[V]$, of matrix $[D]$ from equation 3.6 are used to transform the physical degrees of freedom, χ, into modal degrees of freedom, η, using equation 3.8.

$$\vec{\chi} = [V]\vec{\eta} \tag{3.8}$$

Inserting η for χ in equation 3.6 and multiplying through by the transpose of $[V]$ gives the modal equations of motion, equation 3.9.

$$\dot{\vec{\eta}} = [V]^T[D][V]\vec{\eta} \tag{3.9}$$

To use only the first few modal degrees of freedom, only the first few columns of $[V]$ are used in the transformation (the columns associated with the lowest frequency eigenvalues).

Note that although the state equation form is used for the equations of motion, making the equations of motion first-order in time, the Taylor series collocation constraint can be used as long as one remembers which state variables are position variables and which are velocity variables. All the variables needed to calculate the Taylor series constraint are still in the equations of motion, they have merely been renamed.

3.5 Jet Transport Simulation Results

This section shows the results available from the aeroelastic analysis, by analyzing the example structure used throughout [11], called the "typical jet transport". This wing model will be used for the optimizations that follow. Some differences between the reference jet transport model and this one exist, primarily because [11] assumes a 'dumbbell' point mass distribution, while the finite-element code assumes a distributed mass beam.

3.5.1 Aerodynamic Model

Figure 3.4 shows the wing planform for the jet transport, with the bound vortices illustrated by a dashed line. The wing is unswept (on the quarter-chord line) and linearly tapered. A Mach number of zero was used in the aerodynamic analysis to be consistent with the incompressible analysis in [11].

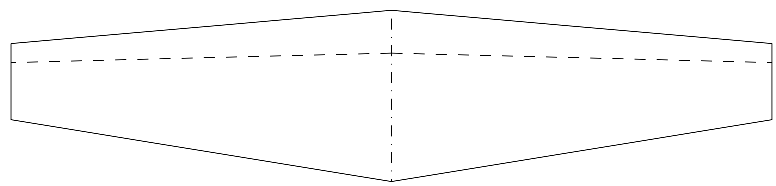

Figure 3.4: Jet Transport Wing Planform

Element	Y_{root} (ft)	Y_{Tip} (ft)	*% Chord*	A (ft^2)	EI_y $(\frac{lb}{ft^2})$	EI_z $(\frac{lb}{ft^2})$	GJ_x $(\frac{lb}{ft^2})$
1	0.00	3.75	35	1.0e6	4.22e+008	4.22e+014	1.87e+008
2	3.75	11.50	35	1.0e6	3.06e+008	3.06e+014	1.88e+008
3	11.50	18.92	35	1.0e6	1.89e+008	1.89e+014	1.86e+008
4	18.92	26.50	35	1.0e6	1.21e+008	1.21e+014	1.57e+008
5	26.50	34.42	35	1.0e6	6.67e+007	6.67e+013	8.47e+007
6	34.42	41.67	35	1.0e6	4.06e+007	4.06e+013	3.60e+007

Table 3.1: Structural Properties

3.5.2 Structural and Mass Parameters

The location of the structural and mass axes for the jet transport are shown in figure 3.5. The aeroelastic elements are numbered on the figure as well. Tables 3.1 and 3.2 list the structural and inertial properties of each element. Note in this table that the x, y, z coordinates are with respect to the beam, where x is along the beam axis and z is upwards. Also the model in [11] does not give information on stiffness in all 6 degrees of freedom, so the cross-sectional area and I_z of the stiffness beams are set to large numbers so the axial deflections and streamwise bending deflections will be small. The inertial properties match the properties of the discrete mass wing model in torsional inertia by distributing the total inertia of each 'dumbbell' set along the mass beam. The inertias in the other degrees of freedom are calculated by equating the inertia of the beam with the inertia the dumbbells would add at the wing nodes. Notice from figure 3.5 and table 3.1 that element 3 includes the mass of the wing-mounted engines. Including such a large mass that is located far from the elastic axis would be difficult without the separate mass and stiffness beams used in the finite-element model.

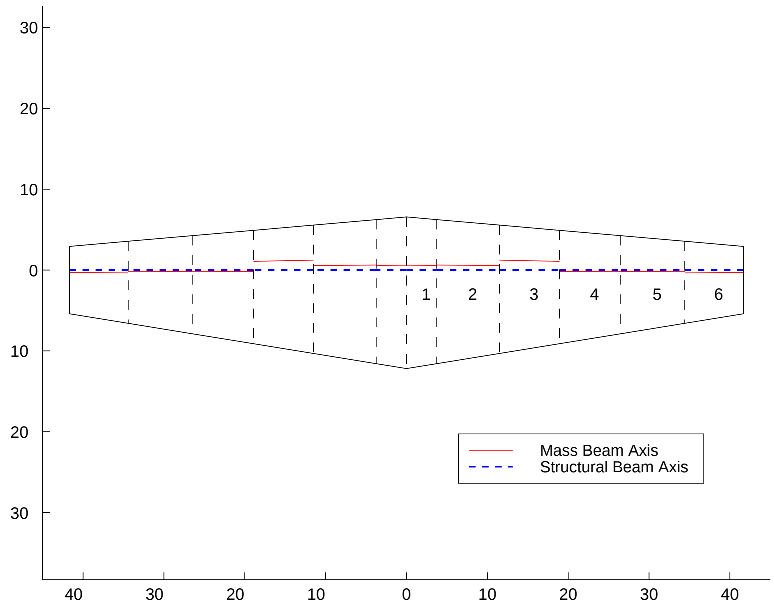

Figure 3.5: Location of Wing Structural and Mass Beams

Element	*% Chord*	$A\ (ft^2)$	$\rho\ (\frac{lb}{ft^3})$	$I_y\ (ft^4)$	$I_z\ (ft^4)$	$J_x\ (ft^4)$
1	32	33.42	0.75	78.33	78.33	335.06
2	32	28.37	0.43	284.01	284.01	284.44
3	27	22.34	0.22	204.85	204.85	12046.12
4	36	17.09	1.01	163.80	163.80	105.12
5	36	12.40	1.08	129.52	129.52	70.15
6	39	8.54	0.34	74.79	74.79	34.87

Table 3.2: Mass Properties

Airspeed:	800 ft/sec
Density:	0.002 slug/ft^3
Mach No:	0
Angle of Attack:	5 Degrees

Table 3.3: Jet Transport Simulation Parameters

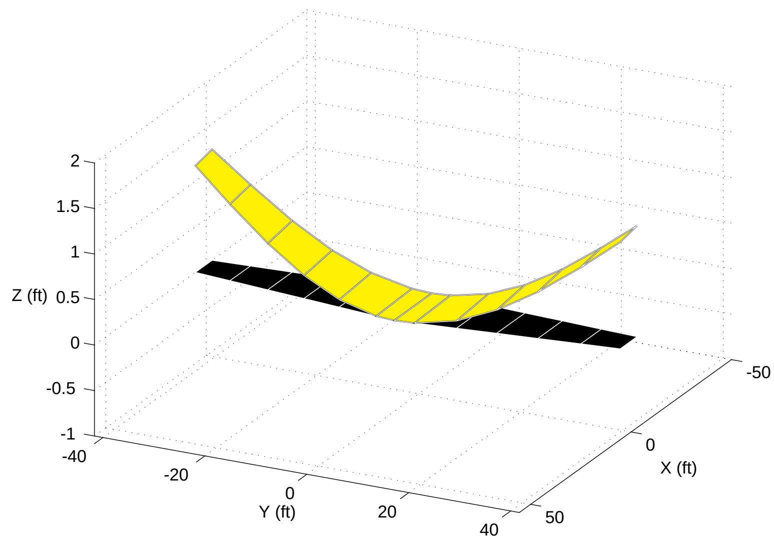

Figure 3.6: Static Deflection

3.5.3 Simulation Parameters

Table 3.3 lists the flight conditions for the simulations in this chapter.

The first example calculation is a static deflection at approximately 200 mi/hr. In figure 3.6 the wing is shown undeflected and deflected with a load (for both wings) of 83,838 lb., the aircraft's weight. Note that the z axis scale is exaggerated in figure 3.6.

3.5.4 Linear Stability Results

Structural dynamic stability can be found using an eigenvalue approach. The sign on the real part of the eigenvalue indicates whether or not the linear equations of motion (equation 3.6) are stable: negative eigenvalues are stable, positive eigenvalues are unstable. This stability information is plotted in figure 3.7 which shows the

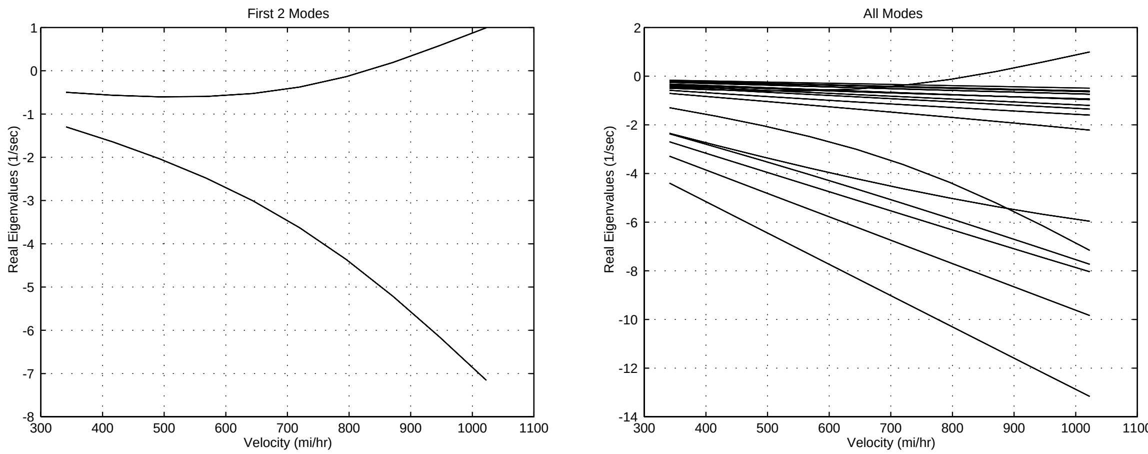

Figure 3.7: Real Eigenvalues

real eigenvalues vs. the freestream velocity at a constant angle-of-attack. The root locus is shown in figure 3.8, where the initial velocity is marked with '$\circ$' and the final velocity '+'. The wing becomes unstable at 830 mi/hr, within 4% of the result in [11]. Reference [11] comments on the supersonic nature of the calculated flutter velocity, which is "far too large to justify the assumption of incompressible flow. It does establish, however, that the jet transport is safe from bending-torsion flutter throughout the operating range of flight speeds."

The four lowest frequency mode shapes (the modes include the linearized aerodynamic forces and so are the damped structural modes) are plotted in figure 3.9. The plots show the z displacement of the leading and trailing edges. If the two lines lie nearly on top of one another, the mode is primarily bending, while separated lines indicate a predominately torsional mode.

3.5.5 Time Simulation

Figure 3.10 shows the time history of the z-displacement of leading and trailing edges from an undeflected initial condition. The motion is damped because the velocity is below the flutter speed.

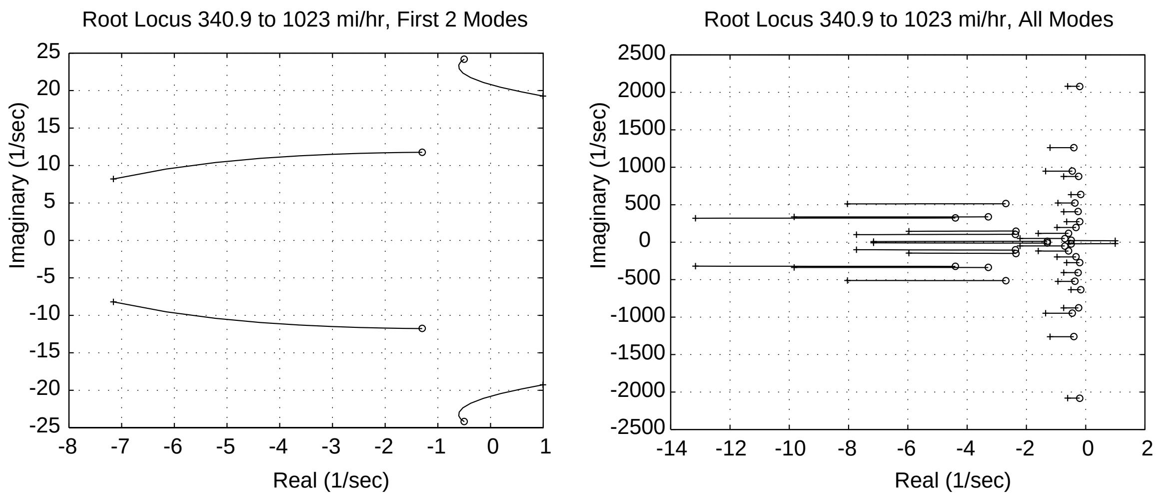

Figure 3.8: Root Loci vs. Speed

3.6 Optimization Problem

As an example of applying the collocation method to aeroelastic optimization, the structural weight of the jet transport wing was minimized at a given flight speed with a flutter divergence constraint. The design is evaluated slightly below the flutter speed. Several optimizations using this model and the collocation method for the dynamic simulation were performed. Initial optimizations had just one structural design variable so that the dividing line between stable and unstable structures could be clearly defined. The dynamic stability of the optimized structural designs using a wide range of stability constraints was confirmed with a frequency-domain analysis. Collocation using the physical degrees of freedom was first used as it is the most direct approach; later modal decomposition was used to reduce the number of degrees of freedom of the system. Nonlinear structural stiffness was included in one case to show that the collocation method is not restricted to linear analyses and to compare the solution processes between the linear and nonlinear simulations. A slightly more complex structural optimization was ultimately performed, and also compared with stability information from the linear frequency-domain analysis.

The initial collocation optimizations used the most simple structural optimization: scaling the structure up or down from the base configuration. The spar-scale design

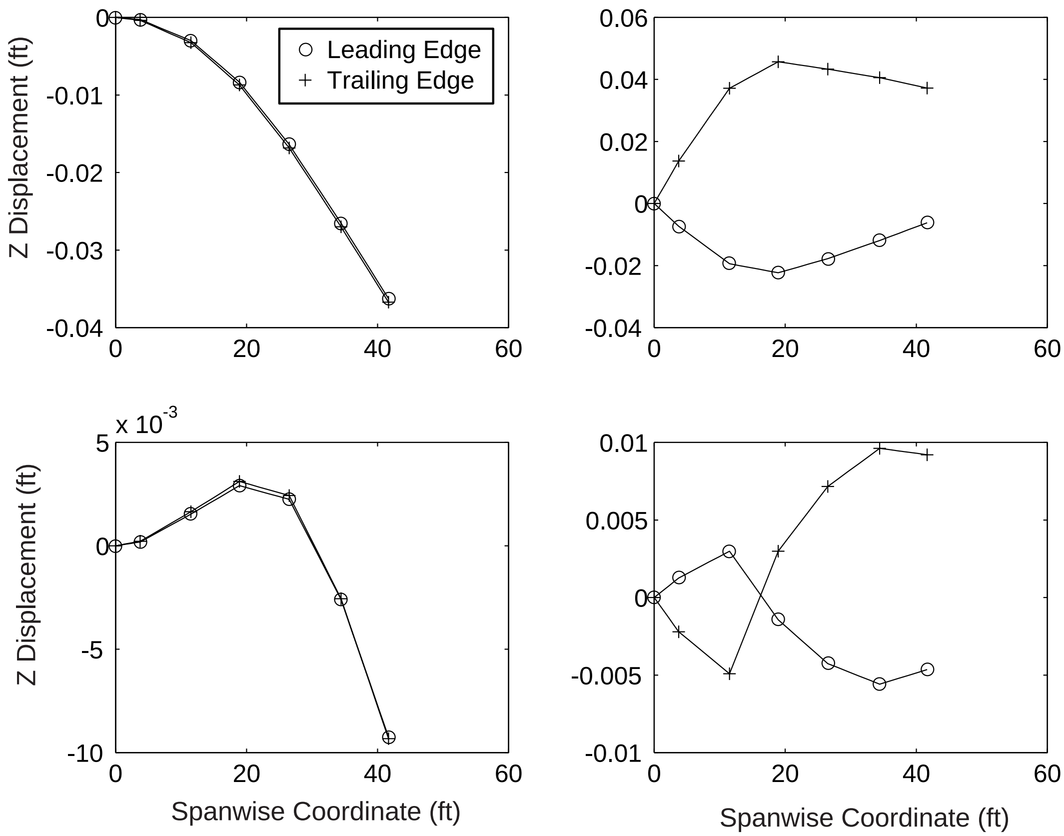

Figure 3.9: First 4 Mode Shapes

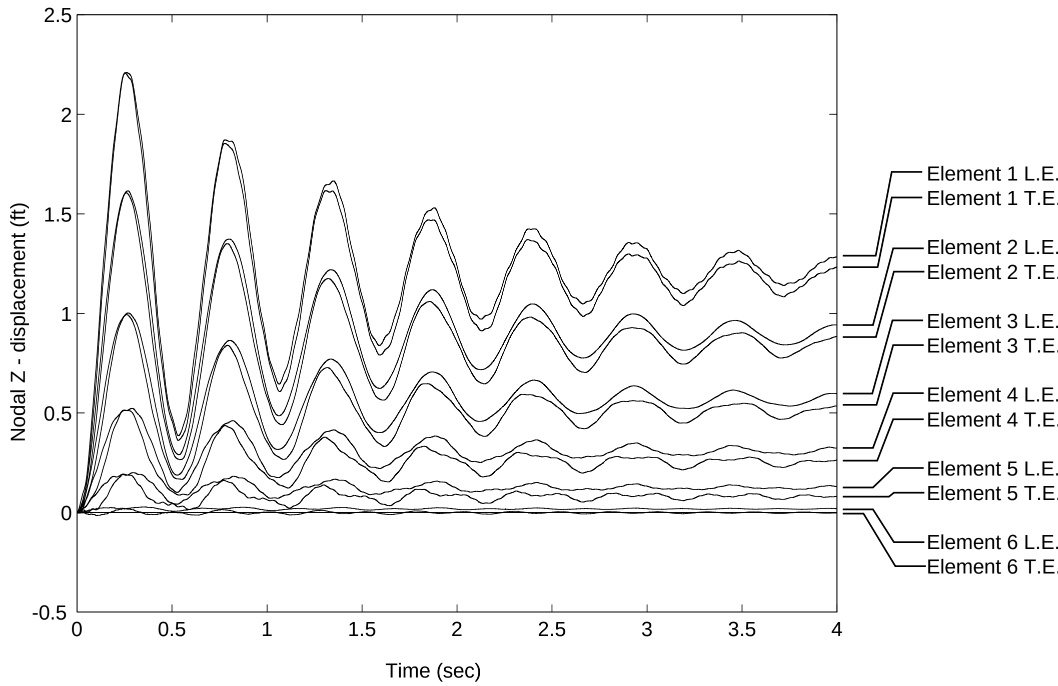

Figure 3.10: Displacements in Time

variable is proportional to a beam cross-sectional length; the weight, inertias and stiffness are all functions of the spar scale. Equations 3.10 through 3.12 show how the properties of both the mass and stiffness beam elements change as a function of the scale parameter, w, and the original element cross-sectional properties of area, A_o, moment of inertia, I_o, and polar moment of inertia, J_o.

$$\text{Area} = A_o w^2 \tag{3.10}$$

$$\text{Moment of Inertia} = I_o w^4 \tag{3.11}$$

$$\text{Polar Moment} = J_o w^4 \tag{3.12}$$

The optimization problem with a single structural design variable is stated formally in equation 3.13.

$$\begin{aligned} \min_{\vec{x}_i, w} \quad & w \\ \text{subject to} \quad & \vec{c}_i = 0 \\ & \vec{l}_i \le \vec{x}_i \le \vec{u}_i \end{aligned} \tag{3.13}$$

Here w is the beam scale factor, $\vec{x}_i$ is the state variable vector for all time points i after the initial condition, and $\vec{c}_i$ is the set of collocation constraints for each state at the collocation point between time points i and $i+1$. The lower and upper bounds for $\vec{x}_i$ are $\vec{l}_i$ and $\vec{u}_i$, and form the stability constraint. The state variables in the earliest optimization results were the physical degrees of freedom of the nodes, while later modal degrees of freedom were used to allow more time steps and a longer simulation for a given number of design variables.

3.7 Collocation and Stability Constraints

The collocation approach adds all the state variables $\vec{x}_i$ for each time point i to the set of optimization design variables. The equations of motion are enforced at collocation points located midway in time between the design variables. Equation 3.14 is a set of collocation constraints for the states at the i^{th} collocation point. This form of collocation constraint is the trapezoidal constraint from [6] and Chapter 2.

$$\vec{c}_i = \vec{x}_i + \frac{dt}{2}\dot{\vec{x}}_i - (\vec{x}_{i+1} - \frac{dt}{2}\dot{\vec{x}}_{i+1}) = 0 \tag{3.14}$$

The second-order Taylor constraint was developed during this work because it seemed inconsistent to not use the higher order derivatives when available. This form of the constraint, equation 3.15, was used to obtain later results of this chapter.

$$\vec{c}_i = \vec{x}_i + \frac{dt}{2}\dot{\vec{x}}_i + \frac{dt^2}{8}\ddot{\vec{x}}_i - (\vec{x}_{i+1} - \frac{dt}{2}\dot{\vec{x}}_{i+1} + \frac{dt^2}{8}\ddot{\vec{x}}_{i+1}) = 0 \tag{3.15}$$

The flutter constraint is applied by limiting the motion of the structure to fall within upper and lower bounds: $\vec{l}_i \leq \vec{x}_i \leq \vec{u}_i$. The state variables at every point in time are optimization design variables, whose values can be limited without adding significant computational operations and without changing the size of the Jacobian. Limiting the motion in this manner might seem to require more physical insight than an eigenvalue stability analysis, because during a preliminary design phase reasonable bounds may not be known to great accuracy, but further investigation will show that detailed knowledge is not necessary. This is because when a structure such as this wing model, or for that matter most dynamic systems, becomes unstable, the motion diverges at an exponential rate so that any reasonable bound will be quickly exceeded.

3.8 Structural Scaling with Physical Degrees of Freedom

The optimization problem in equation 3.13 was solved using the physical wing nodes as the state variables in the equations of motion and stability constraints to force the motion to be damped in time. Figure 3.11 shows the vertical motion of a wing node with an active stability constraint. Some damping has been forced in the motion by changing the stability bounds in time.

The optimization problem for the 36 degree of freedom aeroelastic system was solved using 100 time steps with 3565 design variables and 3564 constraints. The final wing scale factor, w, was 0.6730. The wing flutters at a structural scale factor of 0.6356 or less (from an eigenvalue calculation), proving that the design is in fact safely damped.

3.9 Nonlinear Dynamics

The collocation technique can be used to solve nonlinear dynamic equations as easily as linear equations, because it does not include frequency domain simulations or any

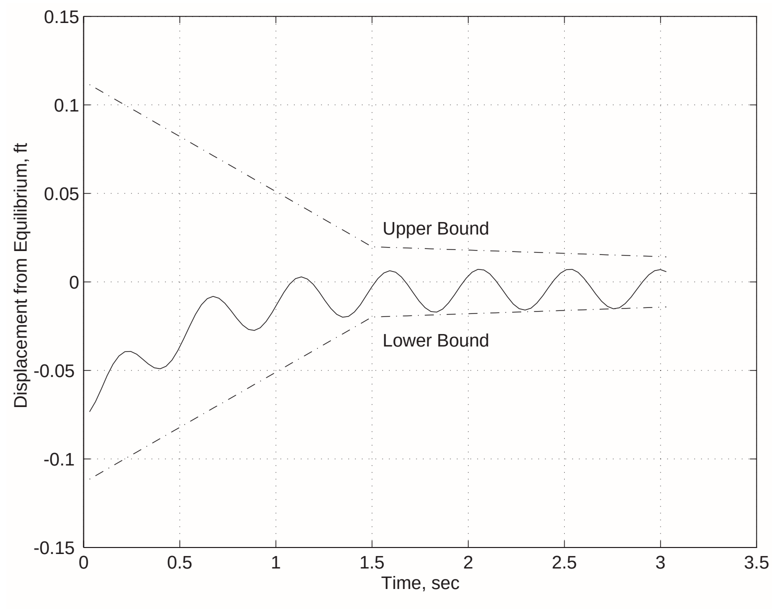

Figure 3.11: Wing Node Time History with Active Bound

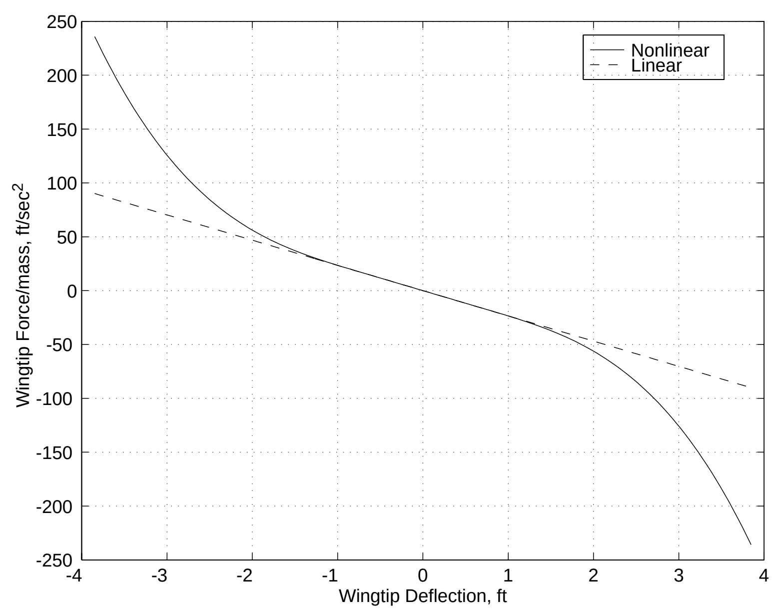

Figure 3.12: Nonlinear Stiffness

other form of linearization. Equation 3.6 gives the wing model linear equations of motion, used in the previous section to assess the stability of the optimizer's solutions. For systems where a linear model can be used, there are many other optimization techniques that can be used that are often more computationally efficient than collocation [14]. In order to compare the solution process with linear and nonlinear models, the linear aeroelastic equations of motion were made nonlinear.

The nonlinear equations of motion, where x_j is the j^{th} element of $\vec{x}$, and $[D]_j$ is the j^{th} row of $[D]$ from equation 3.6, are:

$$\dot{x}_j = [D]_j\vec{x} + \begin{cases} a_j(x_j - x_{nl_j})^3 & \text{if } x_j > x_{nl_j} \\ 0 & \text{if } -x_{nl_j} \leq x_j \leq x_{nl_j} \\ a_j(x_j + x_{nl_j})^3 & \text{if } x_j < -x_{nl_j} \end{cases} \tag{3.16}$$

The new equations are linear near the equilibrium point, but past the deflection limit $\vec{x}_{nl}$ the stiffness becomes cubic, as equation 3.16 shows. The state vector $\vec{x}$ in general includes both displacement and velocity variables; the variables a_j in equation 3.16 are proportional to the static deflection where they multiply the displacement state variables and zero for the velocity state variables, so that the stiffness is nonlinear while the damping remains linear. Figure 3.12 shows the nonlinear nature of the force vs. displacement curve for a typical degree of freedom.

The linear and nonlinear structures were optimized to minimize weight with the same initial conditions and bound constraints. The linear and nonlinear time histories of the degree of freedom whose bound constraint is active (rotation of the wing tip node) are plotted in figure 3.13. The linear case was solved with 10 major iterations, while the nonlinear case required 14 major iterations. The objective at each iteration is plotted in figure 3.14, and the norm of the constraints at each iteration is shown in figure 3.15. The problems are similar enough that the initial iterations are almost identical for both cases, differing significantly only in the last iterations where the nonlinear case takes a few more steps to solve its more difficult dynamic equations and move to the better optimum.

The nonlinear case was able to take advantage of the different dynamics and

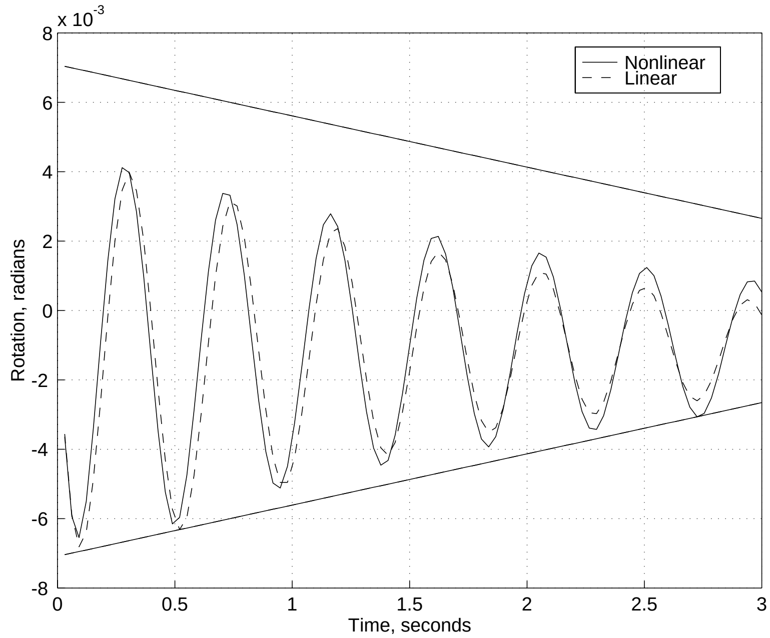

Figure 3.13: Wing Tip Torsion Time History

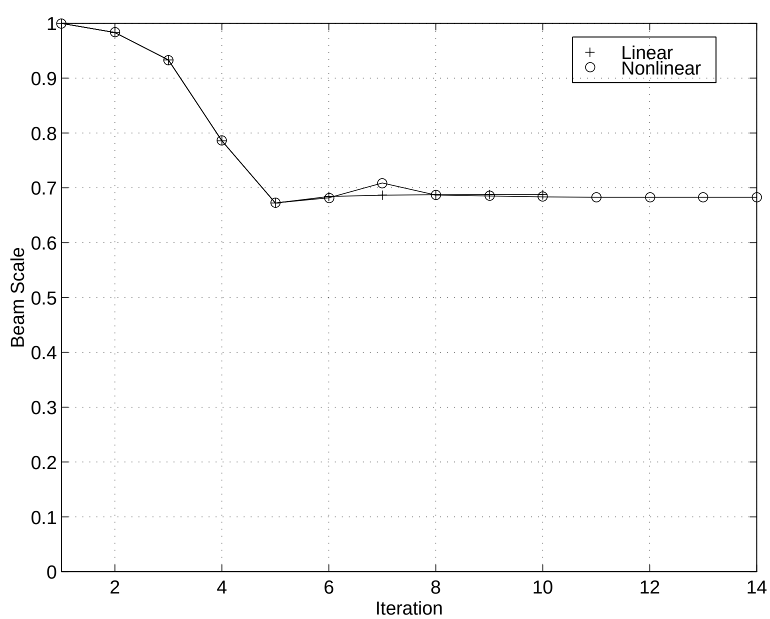

Figure 3.14: Objective Iteration History

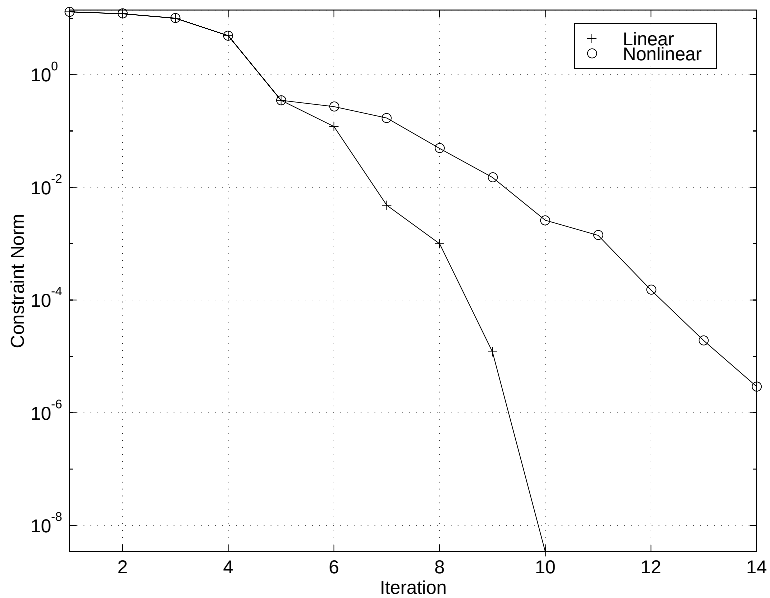

Figure 3.15: Constraint Iteration History

use a smaller, lighter spar. The nonlinear spar can be lighter because for a given spar size it is stiffer in the extremes of motion than the linear structure. The final nonlinear structural spar scale is 0.6827, compared to the linear value of 0.6876. The linear structure has larger initial deflections than the nonlinear model, because the nonlinear model becomes stiffer at large deflections such as those found in the initial oscillations. The linear model's spar is limited in size because it reaches the motion constraints in its first few oscillations. The nonlinear model is stiffer for these initial large deflections, so the spar can be smaller, and its size becomes critical when the last few oscillations reach their limits.

Beyond changing the equations of motion into the nonlinear form, no changes to the collocation code were made to go from the linear equations to the nonlinear. While a linear model was used in this chapter so that the results could be compared to frequency-domain analyses, the real advantage of the collocation method is its ability to solve nonlinear problems. The dynamic simulation is posed in exactly the same form whether linear or nonlinear, and the optimizer solves the problem in roughly the same manner and time.

3.10 Modal Degrees of Freedom

Using the physical degrees of freedom for each node, the simple wing model has 36 degrees of freedom. Because each degree of freedom is a design variable, reducing the number of degrees of freedom directly reduces the size of the optimization problem. With the linear dynamic simulation, modal analysis is a good way to reduce the degrees of freedom of the system while capturing the important dynamics. Using the first few low-frequency modes of the system not only shrinks the optimization problem by reducing the number of degrees of freedom, a further decrease in size may be found if larger time steps can be used in the time simulation because the simulation does not have to resolve the high-frequency motion. The first four modes were used in the simulations in this section, using the modal analysis presented in section 3.4.

The numerical calculation of eigenvalues and eigenvectors is, in general, an iterative process [39]. Therefore, recalculation of the eigenvalues and eigenvectors while calculating finite-difference gradients can give invalid results if the iterations are not converged quite tightly. For this reason, and to speed up calculations during the optimizations, an assumed-mode method was used, where the eigenvectors used to transform the equations of motion were calculated for the initial structure and then reused for the structural designs of later iterations. For the simple structural scaling used it is reasonable to assume the mode shapes do not change very much.

3.11 Stability and Motion Bounds

The stability constraint, in the results of the aeroelastic optimizations presented thus far, has successfully prevented the optimizer from reducing the weight of the wing so much that it becomes unstable and flutters. Using bounds on state variables to provide damping and prevent divergence is not as abstract a process as a frequency-domain calculation, because the damping level is specified as limits on the physical

Oscillation Amplitude (see figure 3.16)	Beam Scale, 3 Sec Sim	Beam Scale 9 Sec Sim	Beam Scale Neutrally Stable
0.5	0.6810	0.6803	0.6356
1	0.6651	0.6652	0.6356
2	0.6507	0.6515	0.6356
3	0.6447	0.6450	0.6356
5	0.6375	0.6380	0.6356
10	0.6303	0.6348	0.6356

Table 3.4: Stability Bound Results

deflections of the system, instead of the single eigenvalue that gives the damping in the linear system. The bounds used may depend on somewhat arbitrary parameters such as the initial conditions, and because in preliminary design stages maximum allowable deflections may not be accurately known, a series of optimizations were performed in order to quantify how accurately the stability bounds must be prescribed in order to prevent the optimizer from designing unstable systems.

In order to investigate the reliability and utility of using motion bounds to predict structural instability, a number of optimizations were performed with different bounds and simulations of different lengths of time. With the structure initially in an unloaded configuration, it was allowed to oscillate about equilibrium as the optimizer simultaneously sized the wing spar and solved the equations. The simulations were run for two lengths of time: a three second simulation with 793 design variables and 792 constraints, and a nine second simulation with 2393 design variables and 2392 constraints. The eigenvalues of the dynamic equations were also calculated using a frequency-domain analysis to find the beam size that made the structure unstable.

Starting with bounds on the wing's motion that produced a well damped structure, the amplitude of the bounds was increased by a uniform factor to increase the allowable amplitude of the oscillations, and the structure was re-optimized. Table 3.4 lists the results. Figure 3.16 shows the highly damped motion of the initial stability bound for the larger optimization problem, while figure 3.17 shows the loose bounds for the small and large optimization problems.

Figure 3.18 shows the size of the optimal wing for each set of bounds, with the

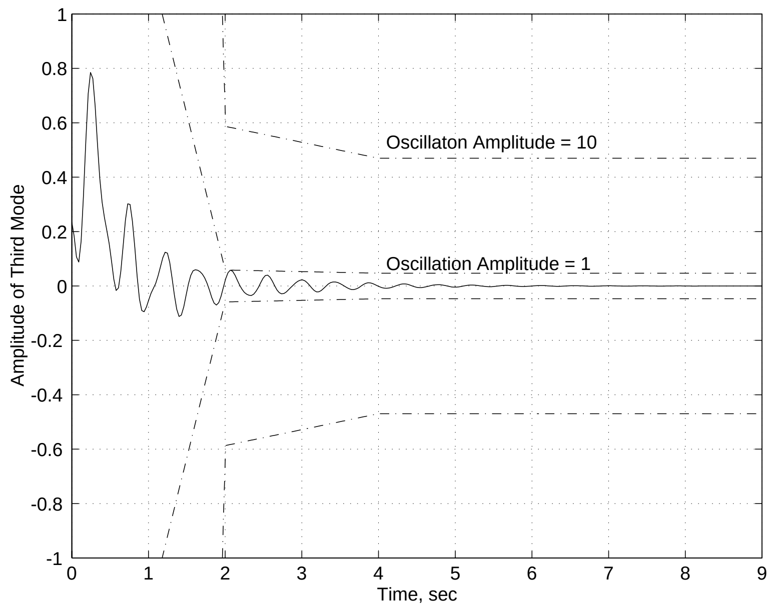

Figure 3.16: Motion in Tight Bounds

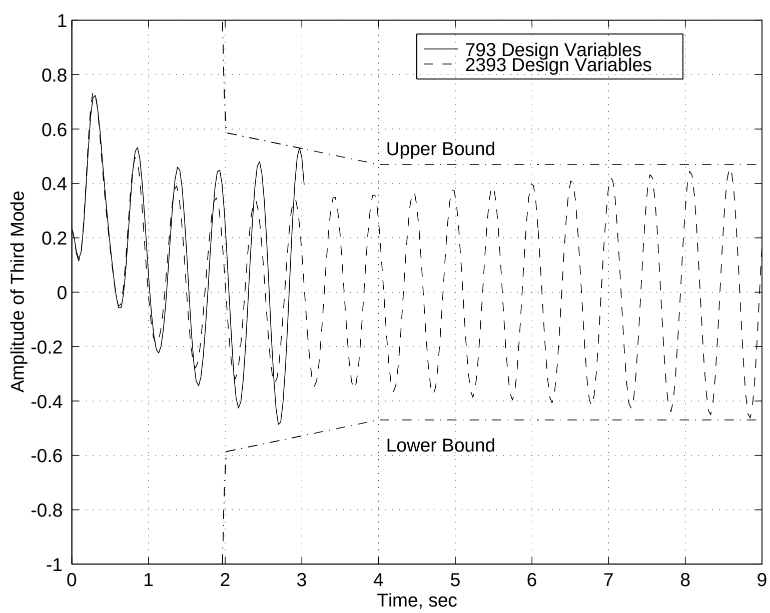

Figure 3.17: Motion with Loose Bounds

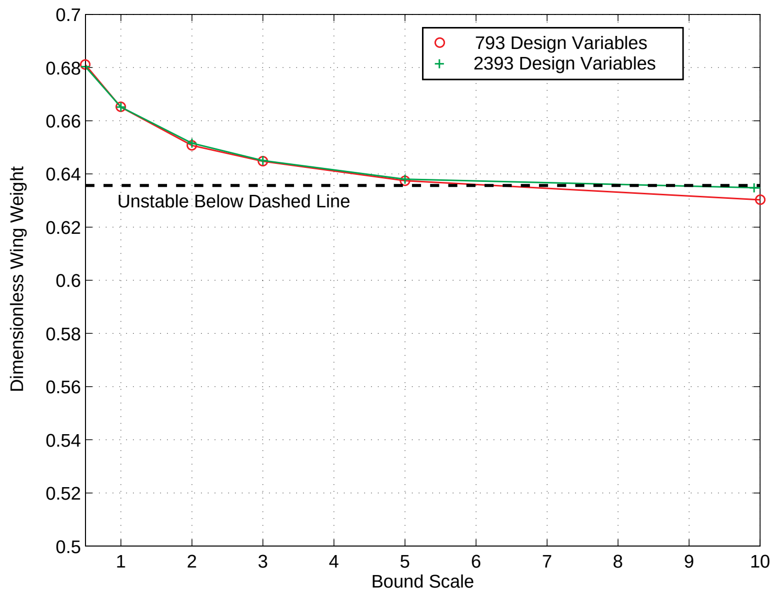

Figure 3.18: Sensitivity of Structural Stability to Motion Bounds

neutrally stable size as a dashed line. The longer simulation is able to prevent the optimizer from making the structure unstable even with very large stability bounds, because it can capture the slowly diverging motion that appears stable to the shorter time simulation. The trade between accuracy and problem size must be made carefully if a structure that is very close to neutrally stable is desired; of course, to design the typically more useful well-damped system there is much less sensitivity to the length of the simulation. Figure 3.18 shows that the designs of both the large and small optimization problems match quite closely when well-damped bounds (bound scale around 1) are applied.

The results in figure 3.18 show that if the structure is prohibited from oscillating ‘too much’ using the simple bound constraints, the optimizer is obliged to design stable structures. Even if the bounds are increased by an *order of magnitude*, the optimizer will converge on a structure that is, in the worst case (largest bounds and shortest simulation), scaled to 99.15 percent of the stable value. The stability bound prohibits even optimizers, which are notorious for exploiting weaknesses in analyses, from tweaking a structure past its stability limit. These results indicate that if the maximum bounds on displacement or velocity are only roughly estimated, an acceptable design will still be found.

Of course, accurate knowledge of the dynamic bounds can only make the design better, by allowing the optimizer to push the physically realistic limits of the design and freeing it from the requirement of designing overly conservative structures. If possible, the equations of motion should be derived in terms of variables whose acceptable ranges are known; when the physical finite element nodes are used as the degrees of freedom the designer should be able to estimate bounds quite easily, while with modal decomposition some inspection of the mode shape can indicate the critical displacement. An ideal, although perhaps less practical, approach for collocation stability constraints would be to write the equations of motion in terms of variables that directly correspond to some failure criteria, such as stresses in the wing spar. Whatever the failure criteria, the designer can rest assured that if the stability bounds are even somewhat accurate, divergence can be prevented.

3.12 Additional Structural Design Variables

A more realistic wing design problem used a linear distribution of the wing beam scaling parameter, instead of the single value used in the previous section. The design variables were the structural scale at the wing root and tip, w_r and w_t. The stiffness and mass beam parameters were still scaled using equations 3.10 through 3.12, but w was a function of span varying linearly from w_r at the wing root to w_t at the wing tip. The objective was to minimize the weight, $(w_r + w_t)$. Equation 3.17 states the formal optimization problem. The Taylor series collocation constraint, equations 2.11 and 2.12 were used to enforce the equations of motion.

$$\begin{aligned} \min_{\vec{x}_i, w_r, w_t} \quad & (w_r + w_t) \\ \text{subject to} \quad & \vec{c}_i = 0 \\ & \vec{l}_i \le \vec{x}_i \le \vec{u}_i \end{aligned} \tag{3.17}$$

The initial design was chosen to be both unstable and a poor distribution of

Case	w_r	w_t	Objective
Initial Design	0.6000	0.9500	1.55
Final Design	0.6665	0.6456	1.312

Table 3.5: Linear Structural Variation Optimization Results

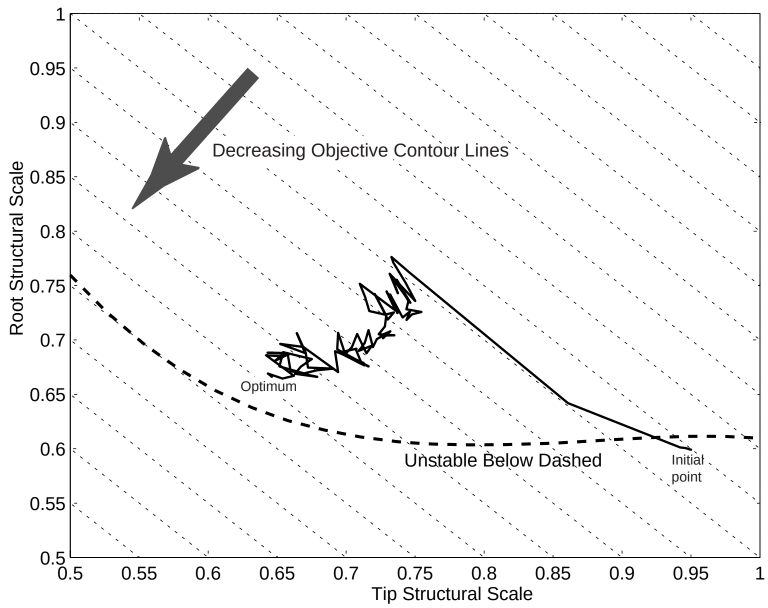

Figure 3.19: Path in Design Space

stiffness. Table 3.5 compares the initial and final designs. Figure 3.19 shows the iteration path of the optimizer in the design space. The optimizer quickly moves into the feasible range, then consistently reduces the objective. Eigenanalysis of the linear equations of motion gives the neutrally stable flutter design, indicated by the dashed line in figure 3.19, which shows that the final design is well inside the stability limit. Figure 3.20 plots the objective and norm of the constraints for each major iteration. The stability bounds were set rather tight for a conservative, well damped design, as seen by the motion in figure 3.21 and the distance from the final design to the neutrally stable line in the structural design space plot of figure 3.19.

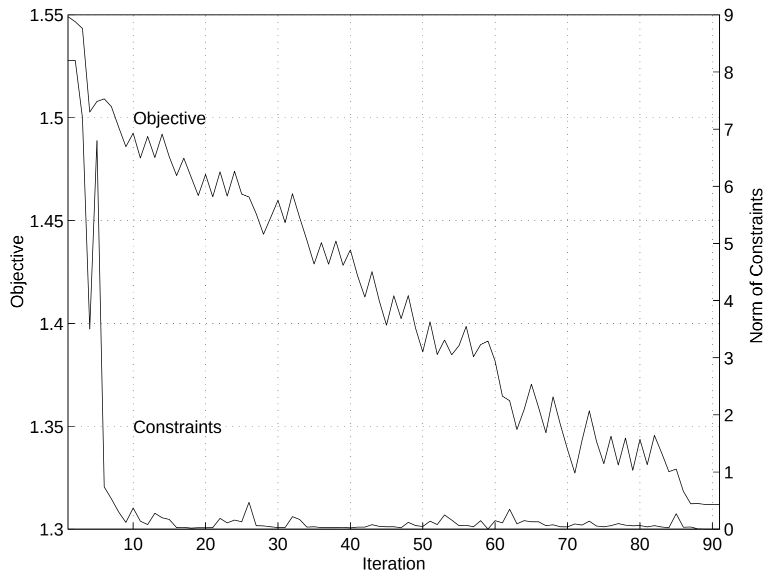

Figure 3.20: Objective and Constraint Iteration History

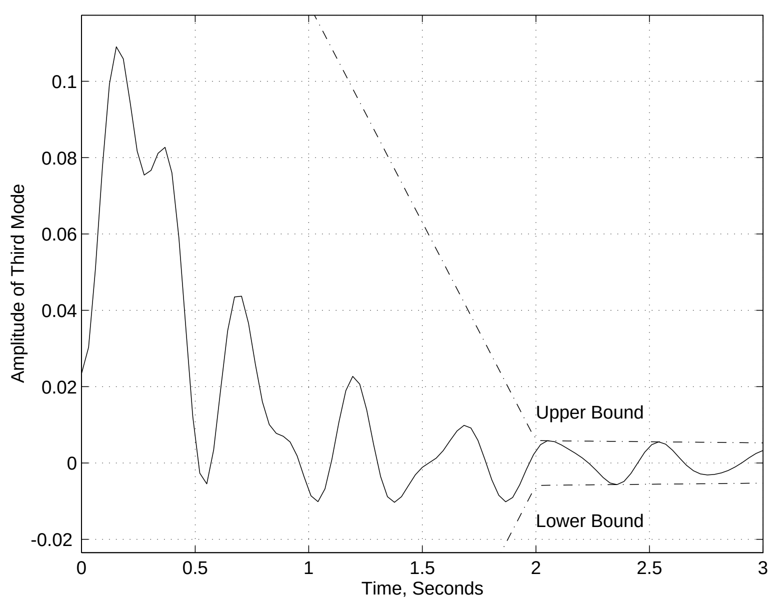

Figure 3.21: Time History of 2-D Structural Design Problem

3.13 Conclusions

This chapter has shown that collocation is an effective tool for aeroelastic optimization. The simple aeroelastic design problems demonstrate techniques that can be applied to more complex optimizations and higher-fidelity models without modification. Both linear and nonlinear aeroelastic time-domain simulations were solved, the use of modal analysis was introduced, and the effectiveness of the stability constraint was evaluated for both well-damped and neutrally stable designs.

Using the collocation method on the jet transport finite-element model, the optimizer was able to find the minimum weight structure and the time history of the motion, while the collocation constraints always forced the optimizer to find solutions to the equations of motion consistent with the physical model. The method is as easily applied to nonlinear problems as it is to linear simulations, and changing the structural behavior of the finite-element model to include nonlinear stiffness was accomplished with no changes to the problem formulation and very similar optimizer performance.

The number of design variables and constraints in the collocation optimization problem is directly proportional to both the number of degrees of freedom in the time-simulation and the number of time steps that cover the duration of the simulation. A finite-element representation of a structure generally will include many nodes to minimize errors due to the discretization into elements, creating a large number of states in the collocation design variable set. The high-frequency motion of the structure, which is often of little consequence to the overall motion (because the corresponding high velocities are well-damped), will require that a small time-step be used for accurate solution to the equations, increasing the size of the optimization problem. Modal analysis of the linear finite-element structure, and simulation using just the low-frequency modes, can reduce both of these contributions to the size of the optimization problem by reducing the number of state variables while at the same time increasing the lowest natural frequencies involved in the simulation to allow a larger time-step.

The collocation stability bound, a simple upper and lower bound on the time history design variables, was used for the first time with collocation to effectively control the damping of the aeroelastic structure. When tightly constrained, the optimizer designed a wing structure whose motion was well-damped; with looser constraints the optimizer was able to design a lighter, less-stable wing. By varying the magnitude of the oscillations by over an order of magnitude it was shown that the designer does not need to have detailed knowledge of the bounds to use for each degree of freedom in the preliminary design stages because the exponential divergence of an unstable structure will quickly exceed any reasonable bound. The length of the simulation in time was shown to influence the design of neutrally stable structures, because the convergence or divergence of the motion will be slow; on the other hand when designing more practical highly-damped structures only a few oscillations are necessary and a short simulation is fine.

In combination with the stability constraint, the collocation method is quite capable of solving dynamic problems that have historically been solved in the frequency domain. Because the collocation approach uses a time-domain simulation, it requires no linearizations or other simplifications to solve the equations of motion. The stability constraint worked so well to include a specified level of damping in the structural dynamics problems of this chapter that it is natural to consider collocation for feedback control design, which will be investigated next.

Chapter 4

Control Design Using Collocation

4.1 Introduction

Feedback control system design encompasses a family of problems naturally suited to the collocation approach. Closed-loop control system designs are often simulated in the time domain, especially systems not easily linearized. The collocation method's stability constraint can be used to force the optimizer to design stable control parameters for the system as it solves the equations of motion of the simulation.

As an example of control system design, the collocation method for optimizing time simulations will be applied to a time-domain simulation of a closed-loop system in order to find a roll-damping gain schedule for a dynamically-scaled flight testbed of the blended-wing-body transport aircraft.

4.1.1 The Blended-Wing-Body

The blended-wing-body (BWB) is a novel tailless aircraft concept originally developed at McDonnell-Douglas and currently studied by Boeing, several universities, and NASA. Figure 4.1 shows a display model of the blended-wing-body. Typical of modern transport designs, the aircraft will include a stability augmentation computer, to allow the aircraft to increase aerodynamic and load-carrying efficiency by reducing the configuration's open-loop stability.

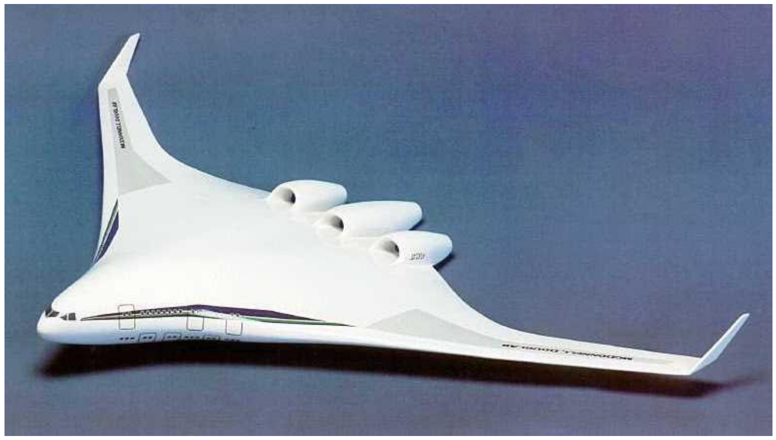

Figure 4.1: Blended-Wing-Body Configuration

The blended-wing-body is a concept for a large transport aircraft, with initial designs sized to carry 855 passengers with a maximum gross takeoff weight of approximately 900,000 pounds. The aircraft's unconventional tailless configuration and larger size than any existing aircraft provide many opportunities for new research. Some of the areas of research [3] [4] include improving propulsive efficiency through boundary layer ingestion using the thick boundary layer at the trailing edge of the aircraft center-section; structural design including the complexities of such a large aircraft and non-cylindrical pressurized cabins; multidisciplinary design to find beneficial trades between competing analyses such as aerodynamic efficiency and structural weight; and flight dynamics, to ensure that the unconventional configuration is stable and controllable throughout its flight envelope.

The stability augmentation system (SAS) for the blended-wing-body must solve flight control problems not usually encountered with normally sized and conventionally configured aircraft. There are aerodynamic drag benefits to reducing the pitch stability of the aircraft by changing the spanwise lift distribution, which require a SAS to keep the flight dynamics acceptably stable. The large root chord and (relatively) small tip chord lengths cause the section lift coefficients at the tip to be much higher than at the center-section, and could contribute to a pitch-up at stall if the aft-located tip sections lose lift before the root sections. The stability augmentation will have to be especially vigilant in preventing stall as a pitch-up at stall inhibits

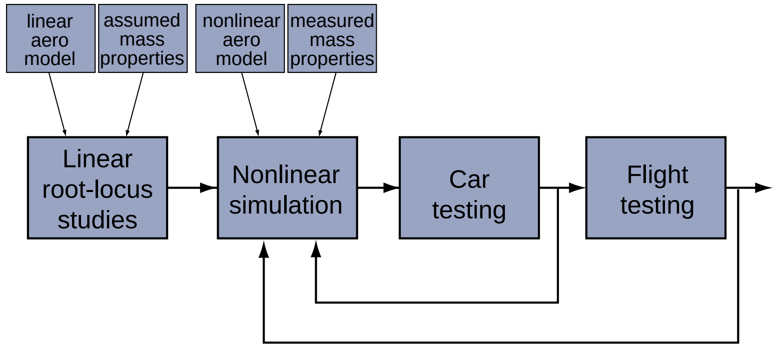

Figure 4.2: Flight Control Development

natural recovery. The flight controls will be actuated by deflecting the trailing-edge surfaces together to provide pitch, roll and yaw control, and the combinations of control surface deflections required to achieve the necessary control power in all axes at the same time must be found. The aerodynamic ground effect is a function of the distance from the ground relative to the size of the wing [47], causing the large BWB to have proportionally large ground-effect changes to its lift and drag, which must be included in the stability-augmentation model for takeoff and landing control.

4.1.2 Flight Control Testbed

The importance of the control system design to blended-wing-body performance has led researchers at Stanford, with guidance from Boeing, to build a small-scale flight control testbed with computer-augmented stability, the BWB-17 [54]. As part of its design and test program, time-based simulations were used in conjunction with captive car-top "flights" to test control algorithms and identify system parameters. Figure 4.2 shows the control design process, including linear and nonlinear simulations, car-top testing, and flight testing. The testbed aircraft was designed to have negligible elastic deflections, to focus the design effort on the rigid-body control design.

The flight-control testbed is a dynamically scaled 17 foot span model of the blended-wing-body, designed at Stanford to investigate the active control system required to fly blended-wing aircraft. Figure 4.3 is a picture of the testbed in flight.

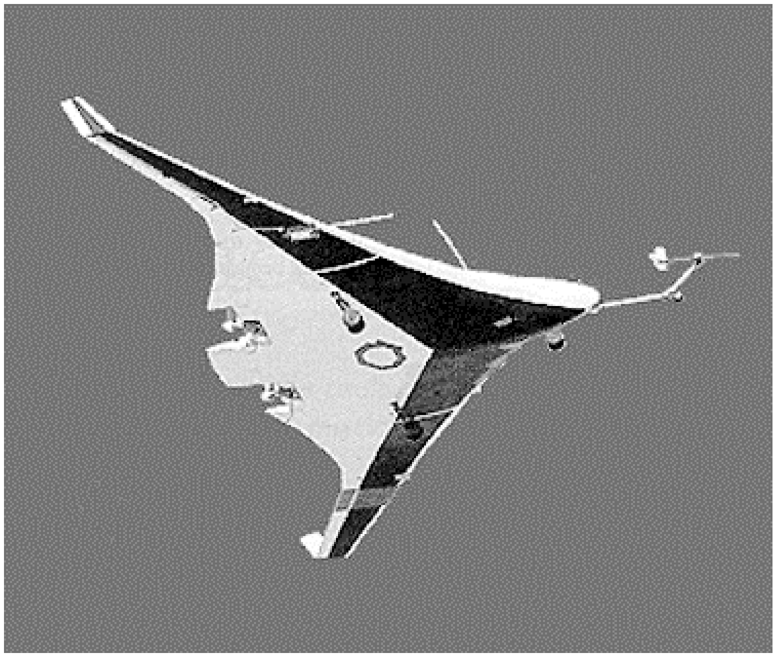

Figure 4.3: BWB Flight Control Testbed

The aircraft has multiple sensors to monitor the state of the aircraft and an onboard computer that commands the control surfaces using partial state feedback. The computer combines the flight control commands from a pilot on the ground with sensor data and actuates the controls on the testbed so that it responds to pilot commands like a naturally stable aircraft.

The BWB-17's flight computer senses airspeed, angle of attack, sideslip angle, roll rate, pitch rate and yaw rate. The computer takes sensor readings twenty times a second and uses this information to control the aircraft's eighteen servos. The fifteen trailing edge surfaces consist of winglet rudders, inboard simply hinged surfaces, and outboard split flaps which can move together as ailerons or opposite as drag rudders or drag brakes. Figure 4.4 shows the layout of the flight control testbed.

As part of the control system design process, the testbed was mounted on top of a car by a gimballed mount located at the aircraft's center of gravity. This test rig, shown in figure 4.5, was used by the design team to test the aircraft at flight conditions and verify the control algorithm without risking a crash. The BWB-17

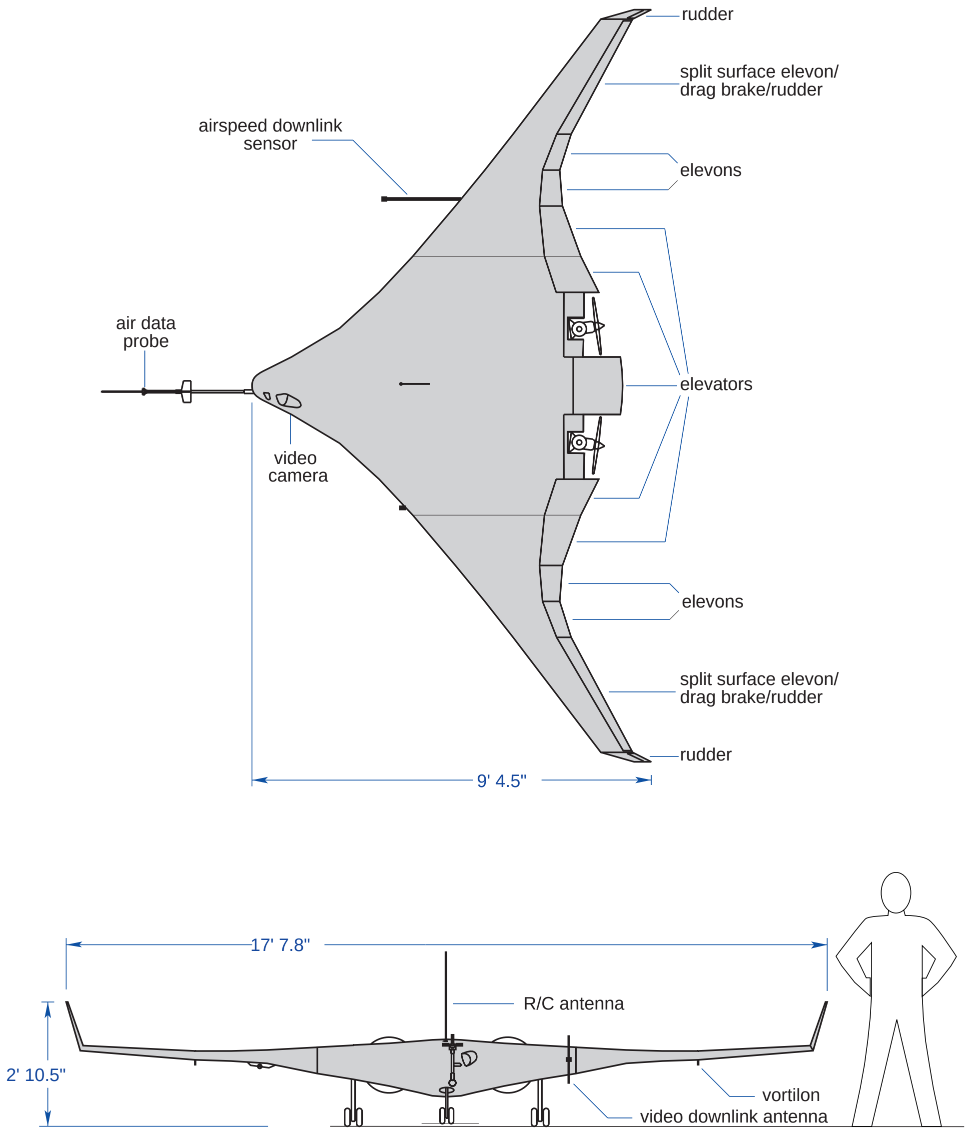

Figure 4.4: Flight Control Testbed Layout

Figure 4.5: Car-top Test Rig

was tested on the NASA-Ames runway for several weeks before the initial flights in the Mojave desert in July of 1997. The car-top experiments were extremely valuable for two reasons: to test the control laws without the risk of crashing the aircraft, and also to determine the physical properties used in the numerical simulations for control system design. The simulations were largely time-based, in order to include accurate nonlinear models of the actuators and sensors, and so that car-top test data could easily be used as inputs to validate new control designs and parameters. Any problems uncovered during the car-top testing were repeated in the numerical simulations, which were then used to verify solutions before modifying the flight hardware and software. The combination of simulation and captive flight-testing was invaluable for designing a flight-ready vehicle with low development costs.

4.2 Application to Blended-Wing-Body Control

The testbed project's use of time-domain simulations and experimental verification provided an opportunity to use collocation-based design for the selection of flight control gains for the BWB flight control testbed. In the course of the initial design and testing of the testbed, an instability in the roll response made gain scheduling of the roll damping necessary. The availability of experimental stability information and a flight-tested gain schedule invited a comparison between the design achieved through manual design iteration and that calculated with the collocation method.

The collocation results were derived after the testbed had been flown successfully with a gain-schedule derived by manually modifying the design and testing it with the simulation until it was satisfactory. The form of the gain schedule (the functional relationship between gain and airspeed) was simple to estimate, which made an iterative manual design process feasible. The collocation method was tasked with solving the problem without applying the knowledge of the gain schedule function, in order to simulate many real-world problems where the control system is too complex to design by hand. The comparison between the collocation method's solution and the experienced designer's flight-tested answers will show how well collocation works on this type of problem.

4.3 Design Problem

A stability problem with the testbed's initial control system was discovered during car-top testing at high speeds (up to 60 mph). The aircraft oscillated in roll as the speed increased, and while the motion never resulted in structural failure, the oscillations were clearly unacceptable for controlled flight. Figure 4.6 shows a plot of roll rate and airspeed for one such oscillation. The oscillation begins as the airspeed rises above 50 mph and is damped as the aircraft slows down. An accurate time-domain simulation was created and showed the same instability when the car-top test data was used as inputs, as the bottom plot in figure 4.6 shows. The physical properties of the simulation were determined by analyzing the results of car-top test runs and other experiments on the aircraft.

The instability was due to the feedback control of the ailerons in response to roll-rate, which was implemented to add roll damping. High effective roll-damping gain at high speeds, in combination with lags in the sensor data and finite servo response, was able to drive the closed-loop motion unstable. Because the rolling moment produced by a given aileron deflection increases with the square of the airspeed, the effective aileron feedback gain also increases. The higher gain raises the frequency of the roll response of the aircraft to inputs or disturbances. The response becomes out of phase

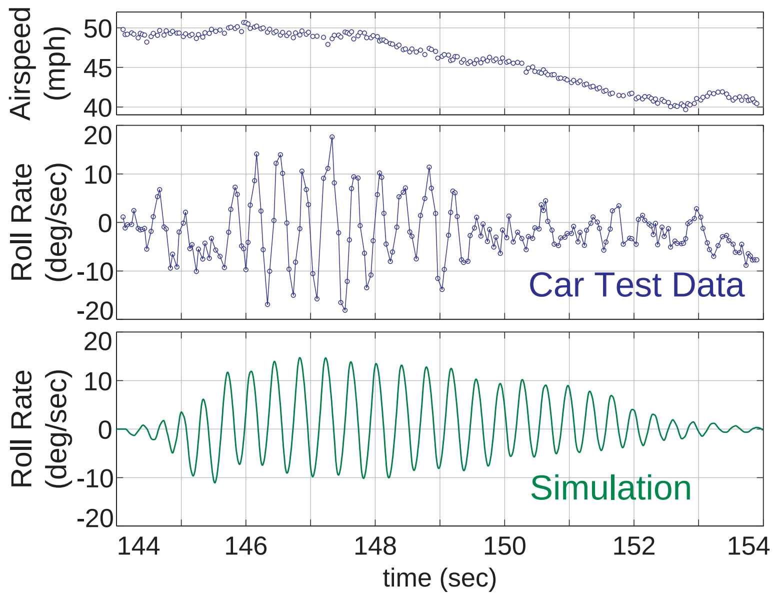

Figure 4.6: Car-top Test Data

and unstable when this high frequency motion is controlled with lagged sensor data and slow actuators. The aircraft's motion is in the form of a limit cycle because the servos have a nonlinear physical slew rate limit, where linear servos would give divergent motion.

A simple cure for the oscillations is to reduce the aileron feedback gain as the airspeed increases and the ailerons become more responsive, which is called "gain scheduling". The aileron effectiveness is proportional to the square of the velocity, so reducing the feedback gain so that it is inversely proportional to the square of the velocity should keep the roll-damping response about the same at all flight speeds. This relationship was used to design the gain schedule by hand, but not included in the collocation problem to see if the optimizer would find the same relationship between feedback gain and speed. The collocation problem sized the gains based strictly on the level of damping in the stability constraint.

The baseline gain used in the original flight control code, whose response is shown in figure 4.6, was determined during car-top testing using the test-pilot's criteria for a good-handling aircraft with respect to roll-damping and roll-rate. The pilot's control requirements place an upper-bound on the roll-damping gain at low speeds to give good roll-response flight qualities. At high speeds the roll-damping gain must be reduced to keep the system stable, but must still be large enough to give some damping.

The design problem is to maximize the roll-damping gain at a given velocity with a constraint enforcing stable motion. The upper bound on the gain at low speeds is found not from stability but roll performance; by requiring a certain steady-state roll rate from a step input in aileron, the aircraft's roll response will not be overly damped. At higher speeds the stability constraint sets the upper bound on the gain.

The design problem is complicated by the nonlinear nature of the testbed's flight controls. Nonlinearities in the data acquisition and the servo response add complexity to classical control design methods. A time-based simulation is the most straightforward analysis, and is easily combined with the time-based test data, which suggests the collocation method as a technique to optimize the gains. The collocation method used here does not use the knowledge of the functional form of the gain schedule, and arrives at a very close approximation with just the stability constraint.

A second set of optimizations were performed to try to predict the onset of instability with the original feedback gain, by applying the stability constraint with loose bounds. A set of roll-damping gains was found that gives nearly neutrally stable motion, and the nonlinear-simulation collocation results were compared with the flight test data, to see how well collocation can predict the point at which a system becomes neutrally stable, without any linearizations in the simulation.

The objective of maximizing the gain is not the most obvious choice, because the response will become unstable if the gain is too large. When optimizing to find the neutrally stable design, it makes sense to encourage the optimizer to make the system unstable, as that will force the neutrally-stable constraints to become active. When applying the collocation method to find the flight gains, it was known that the test pilot favored a roll feedback gain that was high enough to cause the aircraft to become unstable at high speeds. The gain had to be reduced for stability, but the maximum gain was desired to approach the control system that the test pilot picked for its handling qualities. The well-damped stability constraint was used to ensure that the gain was never in the range to cause any instability.

4.4 Simulation

The original simulation, written by Dr. Benjamin Tigner [54] [4] for the gain-schedule he designed by hand, is a straightforward Euler integration of the equations of motion describing the roll response of the testbed aircraft. It is a two degree of freedom simulation of the roll angle (ϕ) and the aileron deflection angle (δa). This is reduced to a system with three state variables by leaving out the roll angle, as only the roll rate, p, is found in the equations of motion. The state variables are therefore $\{p, \delta a, \dot{\delta a}\}$. The major components of the design problem and simulation are the nonlinearities, in both data acquisition and servo response, the initial conditions, the stability criteria, and the optimization problem including the collocation and stability constraints.

4.4.1 Nonlinearities

The data acquisition system is the source of the most significant nonlinearities and time lags. There is a finite lag in getting the data from the sensor to the computer, and a discrete lag because the sensor is read at intervals whose frequency is related to the speed of the computer. Figure 4.7 shows the relation of the sample time (vertical axis) to elapsed time (horizontal axis). If data could be taken instantaneously the sample time would be the same as the elapsed time, shown by the solid line in the figure. A simple lag would offset the line downward: this lag in the aircraft's sensor data is shown in the figure. The stair-step shape of the sampled time (dashed) is a result of the discrete sampling of the computer. A reading is made and held until the next program cycle. These two lags are the main reason the blended-wing model becomes unstable if the magnitude of the gain is too high.

The servo response is another nonlinear component in the simulation. For the aileron deflection equations of motion, a second order system was used with experimentally determined natural frequency ω and damping ζ in response to a commanded aileron input δa_c:

$$\ddot{\delta a} = -\omega^2 \left(\delta a - \delta a_c\right) - 2\zeta\omega\dot{\delta a} \tag{4.1}$$

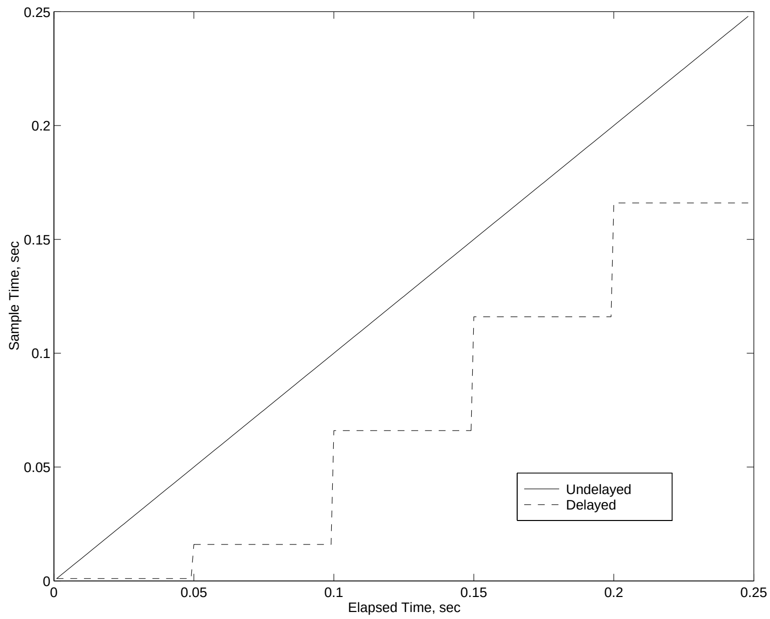

Figure 4.7: Sample Time vs Elapsed Time

When the equations of motion are integrated (not posed as collocation constraints), a nonlinear maximum slew rate is imposed by comparing the updated time step's $\dot{\delta a}$ to the maximum slew rate, and if it exceeds the rate it is reduced to the maximum. In the collocation method the equations of motion are not explicitly integrated– the optimizer can set $\dot{\delta a}$ at will subject to the collocation constraints, so the slew rate limit has to be expressed in the acceleration ($\ddot{\delta a}$). In the collocation method's equations of motion if an integration step performed with the calculated acceleration ($\ddot{\delta a}$) causes the servo to exceed the maximum rate, the acceleration is reduced so that an integration step will bring the new $\dot{\delta a}$ to the maximum slew rate. In both the straightforward integration calculations and the collocation calculations this form of nonlinearity is dependent on the time step size, so some care must be taken when solving the equations of motion. An alternate method to enforce this type of limit will be presented in Chapter 5.

Another approach to the maximum slew rate with collocation would be to use limits on the state variables in the collocation problem just like the stability constraints. This would force the optimizer to change the design to keep the servos from exceeding their maximum slew rate. The problem with this approach is that the slew rate constraint is not something the designers wish to impose, it is something the physical world imposes on the designers. Forcing the optimizer to change control gains so that the linear servo model is accurate does not improve the control system; in fact it may be necessary to have the gains such that they cause nonlinear motion. The optimizer needs to know how to predict the nonlinear motion so that it can work around the physical limitations of the actuators, instead of being constrained to work within the linearized system.

4.4.2 Initial Conditions and Pilot Inputs

The commanded aileron and the initial conditions determine the shape of the response. A step aileron input causes the plane to reach a steady state roll rate. Early collocation efforts used a step input in aileron to test the response, but this approach was discarded because the steady state value of roll rate, p, is dependent on the gain. Defining the collocation stability constraint around a varying steady state value is difficult, because the collocation stability constraint would need to vary with design parameters. Final results were obtained with an initially perturbed roll rate and commanded aileron of zero, so that the equilibrium roll rate was always zero.

4.5 Collocation Method

The collocation method used for the simulation is similar to the simulations in the last chapter. The optimizer is free to choose values for the complete time history of p, δa, and $\dot{\delta a}$ with the exception of the initial conditions. The first order equations of motion for p and $\dot{\delta a}$ are enforced with the trapezoidal form of the collocation constraint, while the second order equations of motion for δa are enforced with the second order Taylor series form of the collocation constraint.

Stability constraints, in the form of upper and lower bounds on the time history design variables, are imposed on the roll-rate, p. The bounds were either very tight bounds which forced the amplitude of the oscillations to decay quite rapidly, for flight-control gain scheduling, or very loose bounds intended to find the flight speed where the original design would become unstable. The damping of the flight-control bounds was chosen to match roll-response levels that the pilot favored from car-top testing at low speeds.

The formal optimization problem can be stated:

$$
\begin{array}{ll}
\max\limits_{\vec{p},\vec{\delta a},\dot{\vec{\delta a}},K} & K \\
\\
\text{subject to} & p_i + \frac{dt}{2}\dot{p}_i = p_{i+1} - \frac{dt}{2}\dot{p}_{i+1} \\
& \delta a_i + \frac{dt}{2}\dot{\delta a}_i + \frac{dt^2}{8}\ddot{\delta a}_i \\
& = \delta a_{i+1} - \frac{dt}{2}\dot{\delta a}_{i+1} + \frac{dt^2}{8}\ddot{\delta a}_{i+1} \\
& \delta a_i + \frac{dt}{2}\dot{\delta a}_i = \delta a_{i+1} - \frac{dt}{2}\dot{\delta a}_{i+1} \\
& l_{i+1} \le p_{i+1} \le u_{i+1} \\
& K \le K_{rollresponse} \\
& (i = 1 \ldots nt - 1)
\end{array}
\tag{4.2}
$$

Where K is the feedback gain of p to δa_c, u and l are the upper and lower bounds on p, and $K_{rollresponse}$ is the gain that gives a steady-state roll rate of 30 degrees/sec per degree of aileron deflection at 30 mph. Note that the stability bounds, l and u are not placed on the initial conditions because these state variables are not part of the set of design variables.

The sparse optimizer MINOS [45] solved the optimization problems. For the most part, 1000 time points were used giving a problem with 2998 design variables and 2997 constraints. The constraint Jacobian was quite sparse with only 15,920 nonzero elements, meaning the matrix was 99.82 % zeros. It is this sparsity which allows such a large optimization problem to be solved on ordinary workstations instead of needing the memory of a supercomputer.

Roll-damping feedback gains were found over a range of velocities for two sets of stability constraints. One set of constraints gives well-damped response that is

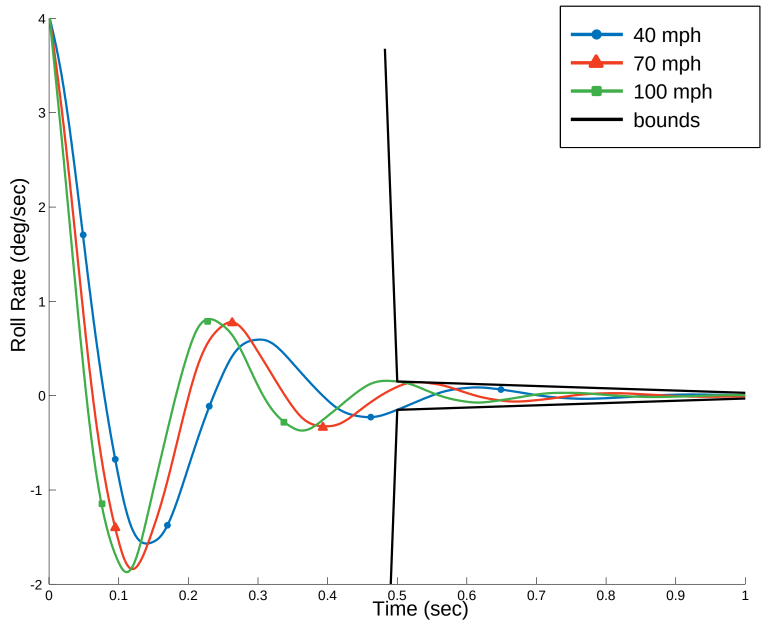

Figure 4.8: Optimized Responses, Tight Bounds

suitable for flight control, the other constraints are set loose to find the neutrally stable gain for each flight speed.

4.6 Solving for Flight Gains

The first stability constraint is very tight, forcing the highly stable, well damped response required for flight. Figure 4.8 shows the time history of the motion with optimal gains at the stable limits for several velocities. It is apparent that the stability constraint is active for the velocities shown because the oscillations touch the bounds at least once for all the cases.

Plots of the gain scheduling determined with the collocation method are illustrative of the power of the technique for control design. The conservative, highly damped gains shown in figure 4.9 are very close to the gain scheduling used in flight of the aircraft which were determined through manual design iteration with knowledge of the relationship between gain and airspeed. The optimizer was able to arrive at a flight-worthy gain schedule merely by being constrained to give a well-damped

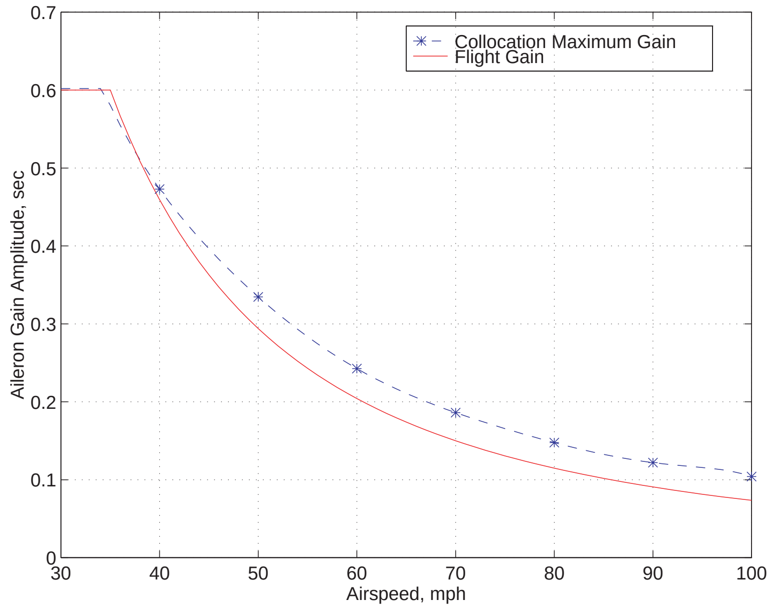

Figure 4.9: Gain Schedule

response. At slow speeds, the roll-damping gain is not set by stability constraints but by the flight-response roll rate requirements, indicated by the straight segment of the plot.

4.7 Gains with Neutral Stability

The second stability constraint's amplitude was set to be the magnitude of the initial disturbance. For a perfectly undamped linear system, the amplitude of the oscillations would stay constant for all time unless energy was exchanged internally between degrees of freedom. For this reason a gain which causes the motion to reach this second form of stability bound should be near the neutral stability gain of the system. The flight speed at which the collocation method predicts the onset of instability can be compared with the observed speeds from car-top testing.

Figure 4.10 shows the time history of the motion with optimal gains for several velocities using the neutral stability bound constraints. Note that while a linear system would oscillate with constant amplitude forever, the nonlinear action of the

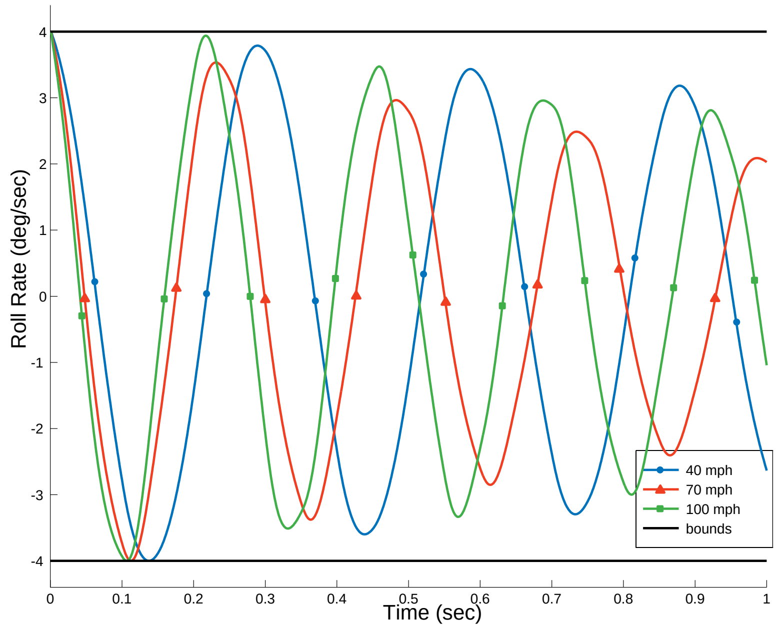

Figure 4.10: Optimized Responses, Loose Bounds

servos causes this motion to be slightly damped.

The gains near the stability limit show the maximum magnitude of gain allowable without becoming unstable. If the straight line of the unscheduled gain is extrapolated as in figure 4.11, it intersects the stability limit at about 50 mph, which is the airspeed where the roll oscillations were first noticed (see figure 4.6). Thus by designing neutrally-stable systems, the collocation method can be used with the stability constraint in a "root locus mode" to find the limit of stability in a nonlinear time-domain control system simulation.

4.8 Conclusions

Because of its ability to treat nonlinear time-domain design problems with criteria similar to a linear frequency-domain analysis, the collocation method is an effective tool for closed-loop control design. The techniques used here on the gain-schedule design of the flight control testbed's roll-damping feedback could just as easily be applied to even more complex and nonlinear time-based simulations. The collocation

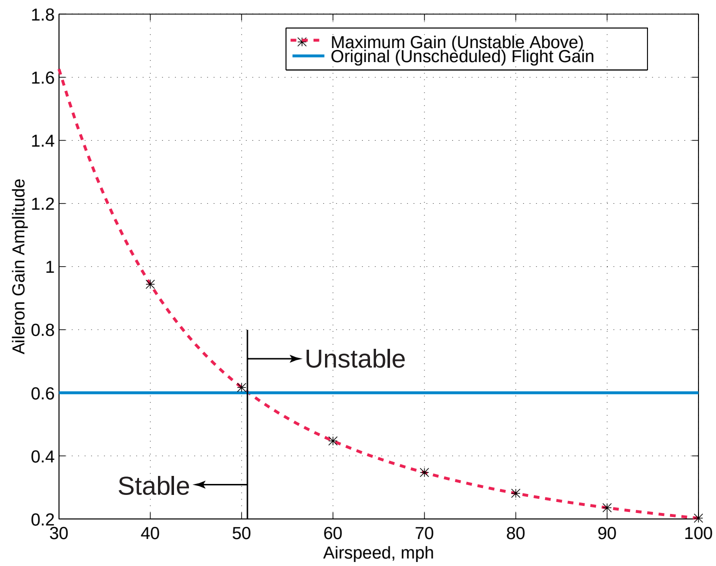

Figure 4.11: Gain Schedule

method was able to match a gain schedule designed by hand and was also successfully used to find the airspeed at which the original control design became unstable.

The simple stability constraint applied by limiting the range of the state variables with side constraints on the design variables worked quite well to force the optimizer to design a control system with the specified level of damping. The resulting gain schedule very closely matched the gain schedule that was designed by hand and successfully flight-tested in the aircraft. The ability of the collocation method to find the gain schedule without knowing the form of the answer shows that it would be useful for more complex problems where the form of the control law is not known to the designers, especially problems with more control system design variables that would make a design by manual iteration impractical.

The stability constraint can also be applied to encourage the optimizer to design neutrally stable systems, in order to find the limit of stability for nonlinear time-based systems. The original unscheduled gains were predicted by the collocation method to become unstable at almost exactly the same airspeed that the instability was observed in the car-top testing. Solving the time-domain equations of motion using the collocation method allows the designer to not only design practical, well-damped systems, but also to find where any nonlinear design will become unstable. This gives an equivalent to the root locus design technique that is not only applicable to nonlinear systems, but able to find the limit of stability with respect to more than one parameter at once.

With the successful solution of both aeroelastic and feedback control design problems using the collocation method, it is appropriate to investigate the ability of the technique to solve a simulation that incorporates both disciplines into one model. Such a design task will be formulated in the next chapter.

Chapter 5

Aeroservoelastic Optimization

5.1 Introduction

The appeal of the collocation method lies in its ability to solve nonlinear time-domain optimization problems. For simulations that can be linearized in a simple manner, there are often methods that will solve the same optimization problems more easily than collocation. Complex, nonlinear problems work well within the collocation framework because of its time-domain nature and parallelizable architecture. Examples of the types of problems that can be solved with collocation are finite-element models with a large number of nodes, nonlinear aerodynamics, and dynamic motion with large deflections.

To show that the collocation method is suitable for solving such complex problems, a simulation that can be used for aeroservoelastic design of a large-scale tailless transport has been created and combined with collocation to optimize the performance of the aircraft while designing a stable feedback control system.

The blended-wing-body (BWB) aircraft will once again be the subject of the analyses. This very large tailless aircraft was introduced in Chapter 4, and is depicted in figure 5.1. This chapter will be concerned with a simulation of the full-scale aircraft's flight dynamics, including a stability augmentation system and aeroelasticity.

The BWB is an interesting subject because it is quite different from existing aircraft. New design rules and design tools must be created for this aircraft, because it is

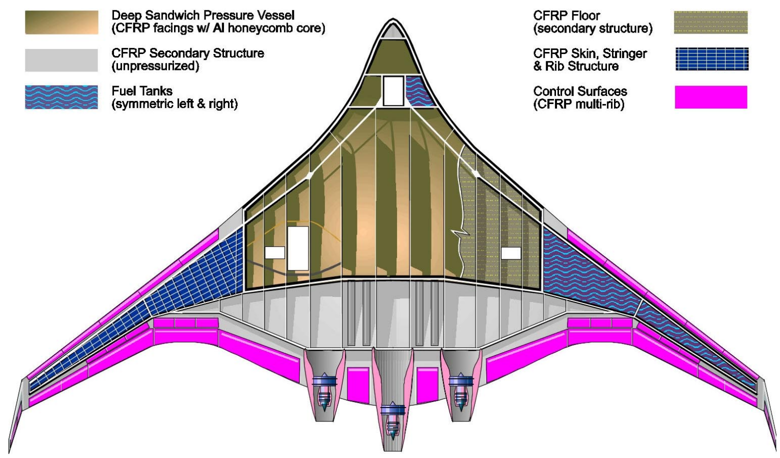

Figure 5.1: Blended-Wing-Body Aircraft (Courtesy Mark Page)

bigger than any built before, with a propulsion system unlike any other, a pressurized structure and cabin layout never before used, and a more heavily burdened stability augmentation system than on more traditional aircraft [4]. The blending of the wing and body also causes the analysis disciplines to be more tightly coupled than on conventional aircraft: for instance the pressure distribution over the top of the wing is both affected by and affects the performance of boundary-layer ingesting engines; the payload and passenger layout changes the wing thickness and spar locations, which modifies the aerodynamic performance as well as the structural properties of the aircraft; and the flight control system and rigid-body motion of the aircraft can couple with the elastic dynamics due to the all-wing configuration. Since the collocation method has been successfully used for the design of both aeroelastic and closed-loop control systems, it seems appropriate to use collocation design techniques for a design problem focusing on the closed-loop control of the blended-wing-body, using an analysis that simulates both the elastic dynamics and unrestrained-vehicle motion.

Tailless aircraft often have stronger coupling between the elastic and rigid-body motion than conventional aircraft. The frequency of the short-period rigid-body motion increases because the pitch inertia of the fuselage and damping from the tail are removed, and the elastic frequencies of the wing decrease because the payload distributed in the wing increases its inertia. As the frequencies of the elastic and rigid-body motion become closer, the dynamic coupling increases. Reference [51] gives an example of a tailless aircraft where the handling qualities and stability were reduced through just such a coupling of rigid and elastic dynamics.

The blended-wing-body, like most modern transport designs, will have a flight computer providing closed-loop control for stability augmentation. This allows the designers to trade inherent stability for increased aerodynamic performance and more flexible payload placement, by relying on the artificial stability augmentation. Because of the tight coupling between rigid-body and elastic motion of the BWB, the flight-control system must be considered not only in the design of the guidance and control of the rigid aircraft, but in the structural design as well. The elastic dynamics of the blended-wing-body in flight cannot be simulated without considering the stability augmentation system and the rigid-body motion at the same time.

One approach to the design of the blended-wing-body is to divide the work into separate disciplines, such as aerodynamics, guidance and control, and structures. Each design team uses the appropriate disciplinary tools to analyze and optimize the design while considering the other disciplines' designs to be fixed. This is the approach taken by most large corporations and indeed is used in Boeing's design of the BWB: for example [4] contains reports by each disciplinary design team on the results of their current design efforts.

Instead of dividing the design of the aircraft into disciplinary parts, one multidisciplinary simulation could be created that incorporates each disciplinary analysis, so that a global design can be evaluated and the overall performance of the aircraft can be measured. This form of simulation requires that the disciplinary teams collaborate to model every system correctly. Boeing is using this technique in combination with separate disciplinary analyses in the design of the BWB, using multidisciplinary design and optimization to make trades between such basic sizing variables as the wing's planform and thickness distribution, fuel weight, structural parameters and payload location, in order to minimize the BWB's takeoff weight [4], while using the disciplinary design teams for the high-fidelity detail design.

As computer design tools improve, it becomes possible to incorporate higher fidelity models such as those usually seen in detailed design into general multidisciplinary simulations [35] [41], by using ever-faster computers to solve intricate design problems quickly, and through more efficient means of coupling separate disciplinary analyses into one grand simulation [50] [53]. The collocation method fits into this design approach quite nicely, as it combines optimization and simulation in a well-posed problem architecture, and allows the analyses to be combined in the time domain for convenience. To illustrate the use of the collocation method in a complex multidisciplinary simulation, a time-domain analysis of the blended-wing-body has been created that can simulate structural deflections and dynamics, structural stresses, flight dynamics including control-surface deflections, and stability-augmentation closed-loop control.

The simulation, written in FORTRAN, is based on a finite-element analysis of the blended-wing-body that creates a model with linear stiffness and mass matrices. The finite-element model is modified to include the dynamic properties of the entire aircraft, and the elastic and rigid-body dynamics are decoupled using an orthogonal transformation to allow the nonlinear flight-dynamic equations of motion to be used with the linear structural model. The aerodynamic forces on the aircraft, including control surface deflections, are found using a panel method analysis similar to that of Chapter 3, and a closed-loop control system is included in the equations of motion so that flight-control laws may be included. The simulation is able to calculate the time histories of the flight path of the BWB, its elastic deflections, control surface deflections, structural stresses, and aerodynamic forces.

5.2 Simulation Model

A finite-element analysis will be used as the basis of the simulation, since a linear structural model is perfectly capable of finding the deflections and dynamics of the BWB. The model is more detailed than the beam-based model of Chapter 3, with elements that more accurately represent the spars, ribs, and skins that make up the structure of the BWB wing. The elements of a spar, rib, and skin model can more precisely adjust the structural properties of the wing in the chordwise as well as spanwise directions. Additionally, the structural stresses can be calculated and used as design constraints with the more detailed model, while the beam model is limited to accurate deflections only.

The elastic model of the blended-wing-body in this study will be limited to the wing box of the unpressurized outboard section of the wing. This decision can be justified for aeroservoelastic design because the center-section must bear the pressurization loads of the non-cylindrical cabin and payload compartments, which, in combination with the large structural depth, makes it so stiff that the structural deflections due to flight loads are very small. The elastic deflections that contribute to the flight dynamics are largely restricted to the outboard panels. The very detailed BWB finite-element model of [4] provides support for this approximation: for example, in a 2.5 g pullup maneuver, the center-section front spar deflects vertically only 4.55 inches, while the wing tip deflects 9.74 feet. The center-section is defined here to mean any portion of the aircraft inboard of 56.75 feet. Simplifying the structural representation by eliminating the center section makes the finite-element model simpler, with fewer nodes and elements, but makes simulating the dynamics of the complete aircraft much more difficult, because the inertial properties of the entire BWB must be included in the model, even for the center section which has no nodes convenient for adding non-structural mass. Figure 5.2 shows the finite-element model in the overall layout of the blended-wing-body; the inertial properties of the center-section are indicated by the rectangular cube at the center and will be discussed in section 5.2.4.

The aerodynamic analysis is similar to that of Chapter 3: a quasi-steady vortex-lattice code. The paneling arrangement uses two chordwise panels so that moveable control surfaces can be included in the simulation. Using two chordwise panels also eliminates the need for the pitching-moment correction of Chapter 3. The aerodynamic-structural coupling is more complex than for the jet transport model used previously, because the structural nodes and aerodynamic panels are spaced independently of one another. The method of moving the aerodynamic loads to the structural nodes and the structural displacements to the aerodynamic panels will be

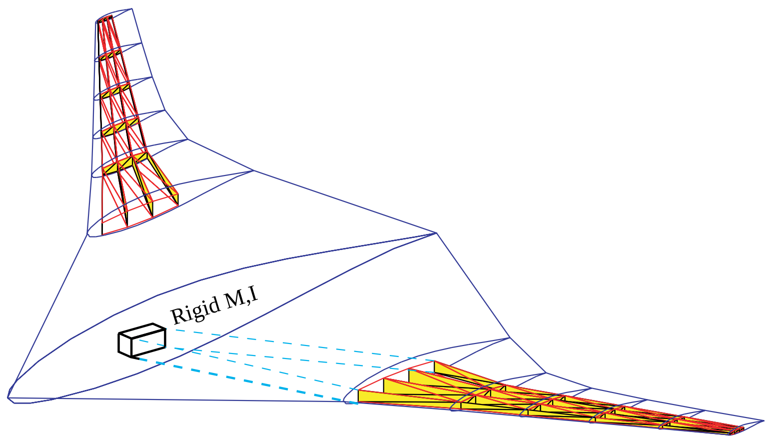

Figure 5.2: Blended-Wing-Body Structural Model

explained in section 5.2.8.3.

If the simulation is to be useful for designing a stability augmentation system, the aircraft model must be able to move freely in space, unlike the clamped wing root boundary condition imposed on the jet transport flutter model in Chapter 3. In the current simulation, the unrestrained degrees of freedom for the longitudinal motion of the aircraft will be included in the equations of motion so that pitch stability can be investigated in combination with the elastic motion. While the lateral motion of the aircraft may need stability augmentation, as the flight-control-testbed has shown, and the elastic dynamics may be as strongly coupled to the lateral rigid body motion, this aspect of the design will be saved for future work. The restriction to longitudinal motion greatly simplifies the simulation, and can be easily removed to change this model to a full six degree of freedom simulation.

In comparison with the aeroelastic analysis of Chapter 3, the changes to the finite-element model and the addition of the unrestrained vehicle degrees of freedom and feedback control increase the complexity of the analysis much more than initial expectations. The large number of degrees of freedom involved in the high-fidelity structural model, the increased complexity of the aerodynamic-structural coupling, and the addition of control-surface deflections to the aerodynamic model and dynamic degrees of freedom increase the size of the collocation optimization problem and the optimization run time well beyond anything seen in the previous chapters.

Using the complex, multidisciplinary analysis for collocation based design shows the fundamental strength of the approach, while bringing to light some areas where gains in efficiency might be sought.

5.2.1 Aerodynamic Model

The aerodynamic model is a quasi-steady vortex-lattice program, very similar to that used in Chapter 3. Each panel contains a horseshoe vortex whose legs trail in the freestream direction. The vortex-lattice method is a very fast routine for calculating the aerodynamic loads, although it is not a particularly high-fidelity method.

Aerodynamic forces due to the dynamic motion are included by changing the panel tangential flow boundary conditions to include the velocity induced by the panel motion. The force calculations using the Kutta-Joukowski law also include the velocity from the panel's motion in the freestream. No wake shedding is performed; the wake panel is extended infinitely in the freestream direction with a constant circulation, which is why the analysis is called quasi-steady.

Two chordwise panels are used in the BWB aerodynamic model, so that one panel may represent the trailing-edge control surfaces. The angle of attack of each trailing-edge panel is changed to match the elevon deflection given by the control law. Using two chordwise panels also eliminates the pitch-damping problem seen with the jet transport aerodynamic analysis of Chapter 3, so that the correction used by the analysis in Chapter 3 is not necessary. Forty spanwise panels per half-span were used to give ample fidelity to the lift distribution. The paneling arrangement is shown in the left half of figure 5.3, with the control points where the boundary conditions are enforced marked with a '+'; the elevons are outlined on the right half of the figure.

The 40 trailing-edge panels are linked together to form 3 separate control surfaces whose deflections are controlled by the stability augmentation system. The percentage-chord of each control surface was chosen to match the control-surface sizes given in [4].

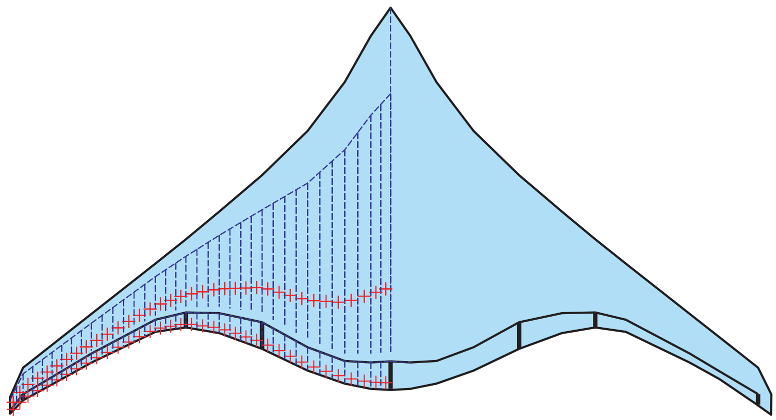

Figure 5.3: Blended-Wing-Body Aerodynamic Model

5.2.2 Feedback Control Model

The three control surfaces are controlled by the feedback control laws, which use the current state of the system to calculate a commanded position for each elevon. Any of the states can easily be used as control inputs, including rigid-body and elastic motion.

Two types of control-surface deflection models can be used in the simulation: an idealized actuator and a physical actuator. The idealized actuator instantaneously moves the elevon to the commanded deflection, with an upper and lower bound on the deflection. This model is simple and places limits on the control authority through the deflection limits. The deflection limit must be smooth for the gradient-based optimization to efficiently solve the dynamic equations, so the deflection limit has to be more sophisticated than a simple logical test such as:

```
if deflection > max_deflection
then deflection = max_deflection
```

A smooth cutoff function based on the arctan function is used, given in equation 5.1.

$$x_{cutoff} = \frac{2x_{max}}{\pi} \arctan\left(2\frac{x}{x_{max}}\right) \tag{5.1}$$

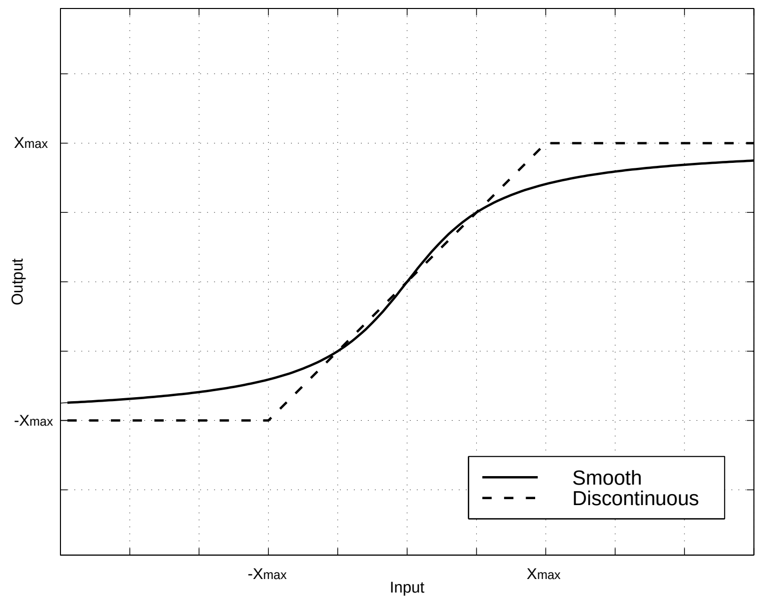

Figure 5.4: Cutoff Function

Figure 5.4 shows the cutoff function. The simple logical test is the dashed line, the cutoff function is shown with the solid line.

The second control surface dynamic model incorporates a physical model of the elevon actuator, using a method identical to the BWB-17 aileron model of Chapter 4. Each elevon has a natural frequency and damping associated with it, as shown in equation 5.2.

$$\ddot{\delta a} = -\omega^2 \left(\delta a - \delta a_c\right) - 2\zeta\omega\dot{\delta a} \tag{5.2}$$

This equation is used to solve for the elevon degrees of freedom in exactly the same way as the flight-control-testbed's equations of motion, including the rate and deflection limits. The physical model includes the lags that real control systems have, making the simulation more accurate and imposing control-effectiveness constraints on the design. Using the physical actuator model adds several additional degrees of freedom to the simulation, making the collocation optimization problem even larger, and the high-frequency elevon dynamics may require that smaller time steps be taken in the simulation, which further increases the size of the optimization problem. These additional complexities must be weighed against the need for the additional fidelity in the control system model.

5.2.3 Finite Element Model

The geometry of the finite-element model is defined by a grid of nodes that make up the wing ribs and spars. The wing structure is defined in a general way that allows the whole structure to be defined with a moderate number of parameters, so that each node and element do not need to be individually placed. These parameters were varied to initially size the structure to match deflections predicted by a high-fidelity aeroelastic model of the blended-wing-body, and could also be used for structural optimization.

The outer-wing box is composed of an inboard section and an outboard section. The spars and ribs are evenly distributed within each section, with the front and rear spars of the box following the lines specified in [4]. The rib outlines are parabolic, so that the vertical location of the nodes can be specified with the thickness of the wing box at three chordwise positions. Figure 5.5 shows the geometry of the BWB wing structure used in this simulation.

After the geometric skeleton has been defined by the nodes, the structural members of the wing are added by connecting the nodes with elements that make up the wing spars, wing ribs, and wing skins. The wing spars are composed of spar-caps and shear-webs. The spar-caps are made from truss elements that run along the upper and lower edges of the spar, while the shear-webs are modeled by 9 degree-of-freedom triangular plate elements connecting the spar-caps. The wing ribs are constructed using the same truss spar-cap and plate shear-web concept as the wing spars. The construction of the spars and ribs is shown in figure 5.6. The wing box is covered on the top and bottom surfaces by the wing skin, which is modeled by triangular plate elements. Figure 5.6 shows how the plate elements are arranged into the skins of the wing structural box.

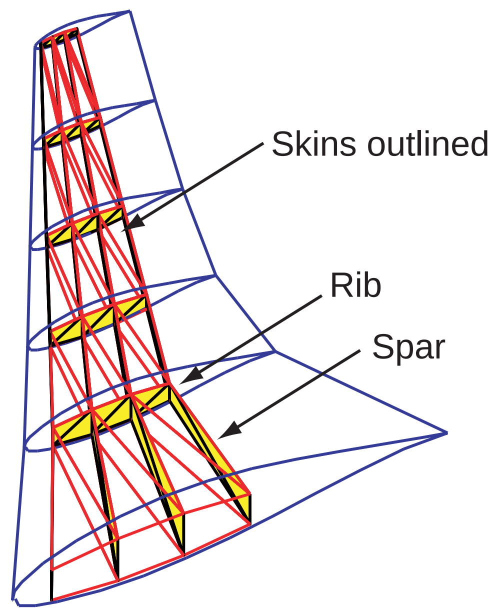

Figure 5.5: Wing Structure

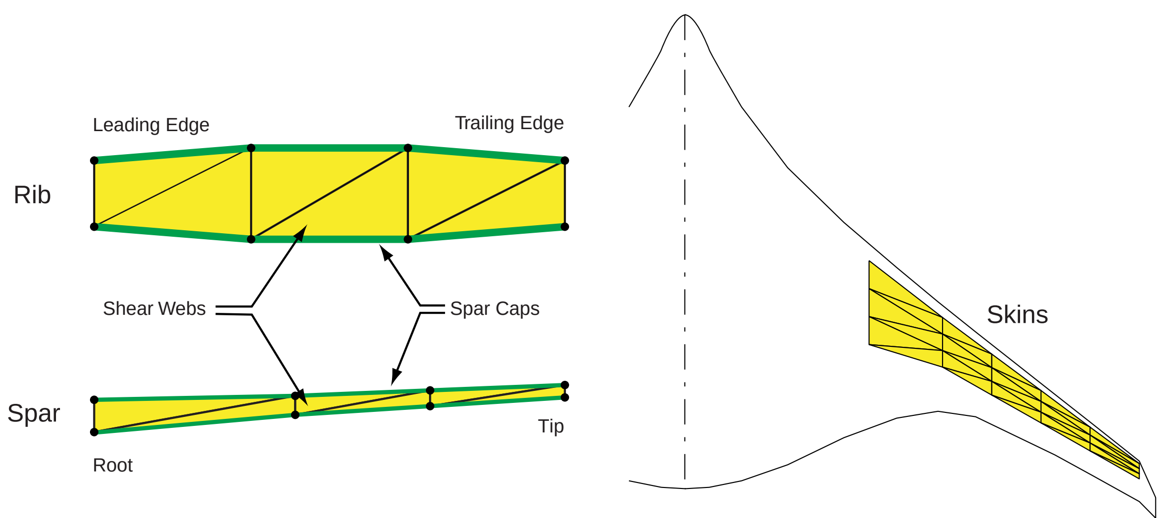

Figure 5.6: Spar, Rib and Skin Construction

Like the geometric placement of the nodes, the structural properties of the elements are defined by specifying the parameters at key locations, with smooth variation between. The plate and truss elements that make up the ribs have a constant thickness and area, respectively, that is applied to the entire wing. As chordwise bending is not considered as important to the behavior of the BWB, fewer design variables were used to describe the rib elements. The thicknesses of the top and bottom skin elements vary linearly from values defined at the wing root and wing tip, with constant thickness in the chordwise direction. The spar elements are defined by thicknesses at the leading and trailing edge at three spanwise locations, which are linearly interpolated to find the leading and trailing edge properties for a given spanwise location, and then interpolated again chordwise onto each individual spar element.

The structure for this model is fabricated from aluminum, unlike the composite model used in [4]. This is a significant simplification that allows an isotropic finite-element model to be used. The density of the structure in both this model and that of [4] is adjusted to account for structural material not explicitly modeled, a procedure (explained in section 5.2.4) that removes the weight difference between the two material types. The homogeneous model may be insufficient for a detailed structural design with anisotropic stress constraints, but for this example it allows the use of existing finite-element sub-analyses with results that closely match those of the composite model in [4].

The finite-element analysis used in this chapter was created specifically for this blended-wing-body simulation, and was compared to a code developed by NASA for use in multidisciplinary analysis and optimization [46] for validation. The original research plan was to use the NASA code, FESMDO, for the blended-wing-body analysis of this chapter, but its incomplete dynamic analysis capability, and program structure designed for solving problems with huge numbers of nodes, whose matrices are too large to be held in memory, led to the development of the new code used here: FESMEH. The codes are quite different, with FESMEH using simpler elements and able to do dynamic analysis, with the assumption that the memory is sufficient to avoid scratch file slowdowns. The similarity in names is due to the lack of originality of the author, who was preoccupied with replacing the NASA code, and not due to

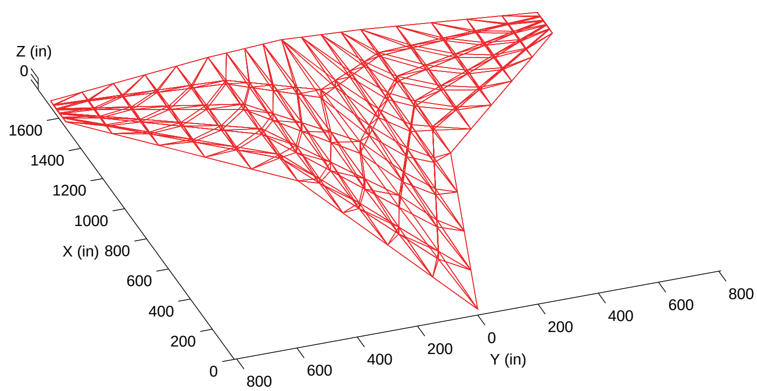

Figure 5.7: Supersonic Transport Wing Structure

any actual re-use of code.

A supersonic transport (SST) wing, shown in figure 5.7, was modeled in both systems to verify that the new code gives valid answers. The model, from [42], uses the same rib, spar, and skin model that is used for the BWB to model the SST wing structure. The deflections and principal stresses for a static load case were calculated with each analysis, and are shown plotted against one another in figure 5.8. The reference code results are plotted on the horizontal axis and those from the code used in this chapter are plotted on the vertical axis. The locus of points lies almost exactly on a line with slope of 1, showing that the two codes provide essentially the same answers. The NASA code's results for this supersonic wing model were validated against NASTRAN computations for [42], so the new code's results should match NASTRAN as well.

5.2.4 Additional Modifications to Finite-Element Matrices

The dynamic model of the blended-wing-body must simulate the motion of more than just the wing structural elements: the inertia of the non-structural parts of the wing must also be included in the mass matrix, and for the unrestrained boundary

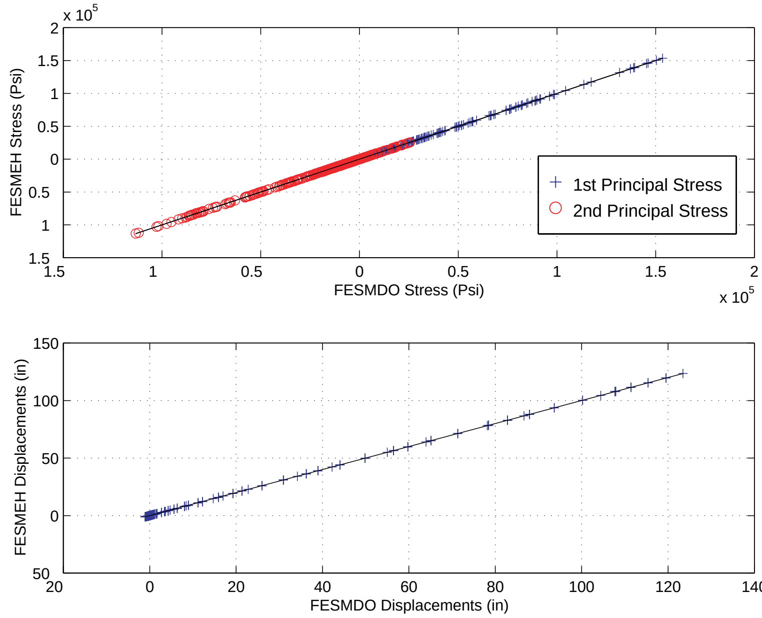

Figure 5.8: Comparison of Stress and Displacement from Two Finite Element Codes

conditions to be realistically applied, the inertia of the entire aircraft, including the center section, must be modeled. The non-structural wing weight, fuel weight, and the inertial properties of the entire BWB center section will be added into the basic mass matrix that the FEM code produces for the structural material. References [3] and [4] give detailed descriptions of the mass properties of the BWB; these parameters guided the changes to the model.

5.2.4.1 Weight Breakdown

The weight estimate of the blended-wing-body in [4] shows that most of the components are contained in the inboard section of the wing– the part that is not modeled by the finite-element structure. Table 5.1 lists the components from the weight breakdown and their spanwise locations. The finite-element model begins at a spanwise station of 56.75 feet. The only components other than the structural material that are included in the part of the aircraft covered by the finite-element model are the fuel tank and the structural "non-optimal factor". The remaining mass and moments of inertia of the center-section will be concentrated at a center section node that will

Item	*Spanwise Location (ft)*
Systems, furnishings, and operational items	0 to 56.75
Center Engine	0
Outboard Engines	25
Main Gear	35
Nose Gear	0
Payload	0 to 56.75
Fuel	56.75 to 93
Center Section	0 to 57
Outboard Wing	57 to 140

Table 5.1: BWB Component Weight Breakdown

be added into the finite-element representation later.

5.2.4.2 Fuel Weight

The fuel weight is included in the finite element model by increasing the density of the lower skin panels in the location of the fuel tank so that they weigh as much as the skins plus the fuel weight. The fuel tank has a quadrilateral planform, and any lower-skin panel whose centroid lies within the box has its density increased so that its weight includes that of the fraction of the fuel tank it covers. Figure 5.9 shows the location of the fuel tanks in the wing. The fuel tanks' sizes and locations are taken from the structural model of [4], and differ slightly from those shown in figure 5.1, which is from an earlier report.

5.2.4.3 Structural Weight

The wing-box structure used in the finite element model is a small fraction of the structural weight of the wing. Leading and trailing edges, structural joints, doublers, fittings and fasteners are not included in the model but make up a significant amount of the wing weight. In order to include this general increase in weight, the density of the structural material was increased by a multiplier (called the "non-optimum factor" in [4]). As in [4], the factor is calculated from the difference between the weight of the modeled structure and the projected structural weight from [3], and is

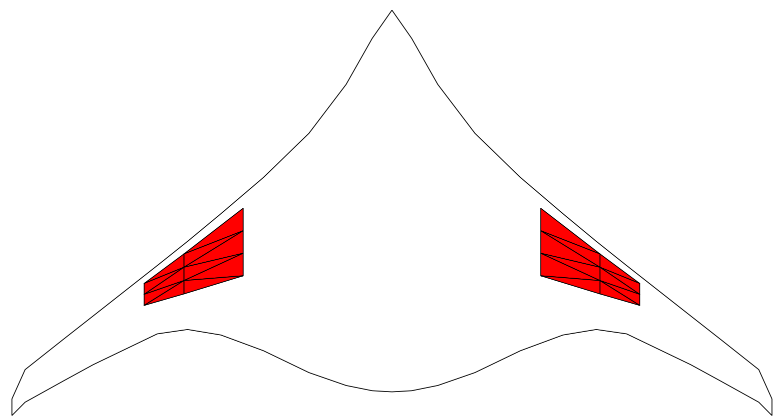

Figure 5.9: Location of Fuel Tanks

evenly applied to the density of all structural elements.

5.2.4.4 Boundary Conditions

The boundary conditions used with the finite element matrices are rather complex. The structure must be free to move rigidly in the longitudinal degrees of freedom, as well as elastically, while being constrained from lateral rigid-body motion. The transformation from the original nodal degrees of freedom of the finite element code to a set of degrees of freedom that allow the properties stated above is described in this section.

Beginning with the results from the finite-element program, we have a set of nodal degrees of freedom, which shall be called $\vec{dx}_0$, and equations of motion in the form given by equation 5.3.

$$[M]\ddot{\vec{dx}}_0 + [K]\vec{dx}_0 = \vec{F} \tag{5.3}$$

The nodal deflections are denoted dx rather than x to emphasize the fact that the finite element model assumes these deflections to be small. While the rigid-body displacements may be large, the structural deflections must remain small or the linear finite element model will not be valid. The mass matrix, $[M]$, includes the fuel and

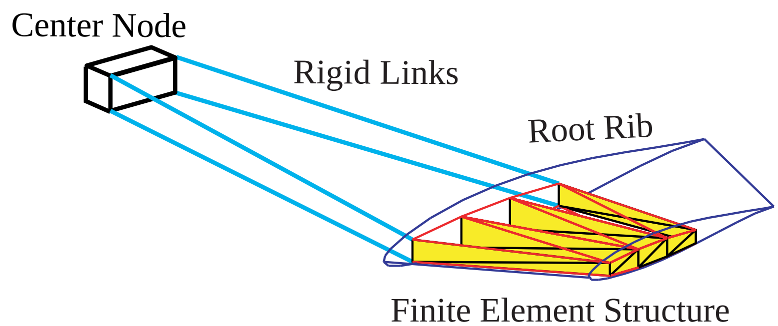

Figure 5.10: Center Section is Rigidly Attached to Root Rib

non-optimal masses as well as the original structure, but does not yet include the inertia of the center-section.

With the three degree of freedom nodes used in the finite element analysis, the set of nodal degrees of freedom, $\vec{dx}_0$, is a vector containing the x, y, and z displacements of each of the n nodes. Equation 5.4 lists the degrees of freedom, $\vec{dx}_0$, used by the finite-element code in creating the stiffness and mass matrices.

$$\vec{dx}_0 = \{dx_1, dy_1, dz_1, dx_2, dy_2, dz_2, dx_3, dy_3, dz_3, \ldots dx_n, dy_n, dz_n\} \tag{5.4}$$

The first modification is to apply the boundary condition that the root rib is rigidly linked to the center section of the aircraft, by adding a six degree of freedom (6-DOF) node located at the center of gravity of the rigid structure and linking it to the wing root. This arrangement is shown in figure 5.10. If the root rib is rigid, the displacements of each of the rib nodes (nodes 1 through n_r) can be described by the displacements and rotations of the 6-DOF center section node, $\{X_c, Y_c, Z_c,$ $\Phi_c, \Theta_c, \Psi_c\}$.

Expressing the root rib degrees of freedom in terms of the center section node creates a new set of the structural degrees of freedom, $\vec{dx}_1$, given in equation 5.5. These new degrees of freedom constrain the root rib to move as a rigid plate and add a node to the model upon which the center-section inertias can be placed.

$$\vec{dx}_1 = \{X_c, Y_c, Z_c, \Phi_c, \Theta_c, \Psi_c, dx_{nr+1}, dy_{nr+1}, dz_{nr+1} \ldots dx_n, dy_n, dz_n\} \tag{5.5}$$

The small-deflection assumption already implicit in the finite-element analysis allows a linear transformation to be used between the center node deflections and those of the rib nodes. Equation 5.6 shows the linear transformation between $\vec{dx}_0$ and $\vec{dx}_1$ due to rigidly connecting the root rib to the center section.

$$\vec{dx}_0 = [TR]\vec{dx}_1 \tag{5.6}$$

Because we are concerned only with the longitudinal degrees of freedom for the aircraft, we can constrain the lateral degrees of freedom of the center section node, (Y_c, Φ_c, and Ψ_c) to be zero, partitioning the degrees of freedom into a still-smaller set, $\vec{dx}_{1c}$, given by equation 5.7.

$$\vec{dx}_{1c} = \{X_c, Z_c, \Theta_c, dx_{nr+1}, dy_{nr+1}, dz_{nr+1}, \ldots dx_n, dy_n, dz_n\} \tag{5.7}$$

By removing the columns for the lateral rigid degrees of freedom from $[TR]$, the cropped matrix $[TR_c]$ is created that transforms the new degrees of freedom to the original nodal DOFs:

$$\vec{dx}_0 = [TR_c]\,\vec{dx}_{1c} \tag{5.8}$$

The equations of motion expressed in terms of the new degrees of freedom are found by substituting equation 5.8 into equation 5.3 and multiplying through by $[TR_c]^T$:

Mass (M)	823 000 lb
Roll Radius of Gyration (K_{roll})	45.3 ft
Pitch Radius of Gyration (K_{pitch})	27.2 ft
Yaw Radius of Gyration (K_{yaw})	50.7 ft

Table 5.2: Inertial Properties of the Rigid Blended Wing Body

$$[TR_c]^T[M][TR_c]\ddot{\vec{dx}}_{1c} + [TR_c]^T[K][TR_c]\vec{dx}_{1c} = [TR_c]^T\vec{F} \tag{5.9}$$

The multiplication of $[TR_c]^T$ in every term of equation 5.9 maintains the symmetry of the stiffness and mass matrices. Redefining the matrices for brevity, the equations of motion can be rewritten as:

$$[M_c]\ddot{\vec{dx}}_{1c} + [K_c]\vec{dx}_{1c} = \vec{F}_c \tag{5.10}$$

These new degrees of freedom reflect the rigid rib at the wing root that attaches the wing to the rest of the aircraft, as well as the constraint that there be only longitudinal rigid-body motion. With the equations of motion using the new degrees of freedom, the inertial properties of the center-section can easily be added to the finite-element mass matrix.

5.2.4.5 Matching Rigid-Body Inertia

When moved as a rigid-body, either with a displacement or a rotation, the dynamic model must have the same inertia as the entire rigid aircraft. The rigid aircraft inertias from [3] are summarized in table 5.2. The rigid inertia of the outer wing (including structural, fuel and non-structural mass) is calculated from the mass matrix, $[M_c]$, and subtracted from the total aircraft inertia to find the center section inertia. This remaining inertia is added to the center node that is rigidly connected to the root rib of the wing structure.

The inertial properties of the finite element structure must be calculated so that they can be subtracted from the inertia of the rigid aircraft. The kinetic energy from rigidly moving the outer wing, represented by $[M_c]$, at a unit velocity, v, is equated to the kinetic energy of the outer wing represented as a rigid system (notice that v cancels out):

$$v\{\vec{dx}_{rb1}\}^T[M_c]v\{\vec{dx}_{rb1}\} = m_{struct}v^2 \tag{5.11}$$

$$v\{\vec{dx}_{rb2}\}^T[M_c]v\{\vec{dx}_{rb2}\} = m_{struct}v^2 \tag{5.12}$$

$$v\{\vec{dx}_{rb3}\}^T[M_c]v\{\vec{dx}_{rb3}\} = I_{YY_{struct}}v^2 \tag{5.13}$$

The rigid-body motion vectors for the $\vec{dx}_{1c}$ coordinate system are given in equations 5.14, 5.15 and 5.16. Recall that the first three degrees of freedom are the center node longitudinal DOFs, $\{X_c, Z_c, \Theta\}$, followed by $\{dx, dy, dz\}$ of the outboard rib nodes.

$$\vec{dx}_{rb1} = \{1, 0, 0, 1, 0, 0, 1, 0, 0, \ldots\} \tag{5.14}$$

$$\vec{dx}_{rb2} = \{0, 1, 0, 0, 0, 1, 0, 0, 1, \ldots\} \tag{5.15}$$

$$\vec{dx}_{rb3} = \{0, 0, 1, 0, (z_i - z_c), -(x_i - x_c), \ldots\} \tag{5.16}$$

In equation 5.16, every outboard node i is located at (x_i, y_i, z_i) and the center section node is located at (x_c, y_c, z_c), and is the point about which $\vec{dx}_{rb3}$ rotates.

Obviously m_{struct} only needs to be found once, so only one of equations 5.11 and

5.12 need to be solved. The additional inertial properties of the center section to are then found by subtracting the outer-wing inertial properties from those of the entire aircraft:

$$m_{center} = m - m_{struct}$$

$$I_{YY_{center}} = I_{YY} - I_{YY_{struct}}$$

These terms are added to the diagonal terms of the mass matrix corresponding to X_c, Z_c, and Θ_c, the center node degrees of freedom. Terms for lateral rigid-body motion could be found in exactly the same way if a full 6-DOF rigid-body simulation is desired instead of the longitudinal simulation presented here.

With these modifications, the mass and stiffness matrices can be used to directly solve for the motion of the complete elastic aircraft, including the longitudinal unrestrained vehicle rigid-body modes. The next step is to decouple these rigid-body modes from the elastic motion.

5.2.5 Separation of Rigid and Elastic Degrees of Freedom

Decoupling the rigid-body modes from the elastic motion is useful for several reasons. First, by decoupling the rigid modes the complete nonlinear longitudinal equations of motion in the traditional body-fixed coordinate system can be used to solve for the rigid motion, removing the linearizations (small angle assumptions) of the rigid motion performed in the finite-element analysis. This allows the use of traditional aerodynamic derivatives if necessary, and likewise any other flight simulation parameters may be calculated with existing tools since the body-fixed coordinate system is used. Also, if the modes have been uncoupled, the rigid motion may be easily found without solving for the elastic motion by setting the elastic degrees of freedom to zero for all time, which is useful for debugging purposes and in cases where the elastic motion is not important.

The method for decoupling the rigid-body modes is described in [43] and [11]. The idea is to find elastic modes that are orthogonal to the rigid body modes. A rigid mode, $\{\vec{dx}_{rb}\}$, is a combination of displacements that does not change the elastic potential energy of the structure no matter what the amplitude:

$$\{\vec{dx}_{rb}\}^T[K]\{\vec{dx}_{rb}\} = 0$$

The rigid modes are characterized by free-vibration eigenvalues that are zero. The simplest set of rigid modes are the modes corresponding to rigid translation and rotation of the structure in the coordinate axes. These rigid modes will be used with the definition of orthogonality to find the elastic modes of the system, which are modes that contribute no net change in the total inertia of the system. That is, no matter how large or fast the oscillations of the elastic modes, they are incapable of creating inertial forces or moments about the center of gravity. The inertial decoupling allows the elastic motion to be solved for relatively independently of the rigid-body motion. Not completely independently however, because the inertially-decoupled modes are recoupled through the aerodynamic forces, as well as through inertial loads that can be applied to the elastic structure by the rigid-body motion. Still, finding elastic modes that do not inertially affect the center of gravity motion greatly simplifies the time-domain simulation.

The rigid-body modes for this longitudinal simulation are those corresponding to rigid longitudinal (x) and vertical (z) translation, and rotation about the pitch axis (Θ). The rigid-body modes for the $\vec{dx}_{1c}$ coordinate system of equation 5.7 were given by equations 5.14, 5.15, and 5.16, where they were used to find the structural rigid-body inertial properties.

For an elastic mode $\vec{dx}_{1c}$ to be orthogonal to a rigid mode $\vec{dx}_{rb}$, the following equation must be true:

$$\{\vec{dx}_{rb}\}^T[M_c]\{\vec{dx}_{1c}\} = 0$$

Application of this equation to the three rigid-body modes ($\vec{dx}_{rb1}$, $\vec{dx}_{rb2}$, and $\vec{dx}_{rb3}$ of equations 5.14 through 5.16) gives three linear equations involving all the degrees of freedom. The three equations can be used to solve for any three degrees of freedom (elements of $\vec{dx}_{1c}$) in terms of the others. For example, the three degrees of freedom of the center section node (X_c, Z_c, Θ_c) can be expressed in terms of the other degrees of freedom $(dx_{n_r+1}, dy_{n_r+1}, dz_{n_r+1}, \ldots dx_n, dy_n, dz_n)$, which will be called $\vec{dx}_e$, as in equation 5.17.

$$\left\{ \begin{array}{c} X_c \\ Z_c \\ \Theta_c \end{array} \right\} = [N2R]\{\vec{dx}_e\} \tag{5.17}$$

The remaining degrees of freedom $\vec{dx}_e$ are called the elastic degrees of freedom because they are decoupled from the rigid-body motion. With any combination of displacements of the elastic nodes, equation 5.17 will give the motion of the other node so that the center of gravity of the system remains in one place and no net angular momentum is produced.

Equation 5.17 can be re-written easily to create a transformation matrix that reduces the degrees of freedom allowing rigid-body motion, $\vec{dx}_{1c}$, to the elastic degrees of freedom, $\vec{dx}_e$:

$$\vec{dx}_{1c} = [TX]\vec{dx}_e$$

The upper rows of the transform $[TX]$ are the matrix $[N2R]$, while the lower rows are simply the identity matrix. With this transformation, the equations of motion may be written in the purely elastic form of equation 5.18:

$$
\begin{aligned}
[TX]^T[TR_c]^T[M][TR_c][TX]\ddot{\vec{dx}}_e \\
+ [TX]^T[TR_c]^T[K][TR_c][TX]\vec{dx}_e \\
= [TX]^T[TR_c]^T\vec{F}
\end{aligned} \quad (5.18)
$$

Or, by defining some new matrices to shorten the equations:

$$
[M_e]\ddot{\vec{dx}}_e + [K_e]\vec{dx}_e = \vec{F}_e \quad (5.19)
$$

These new equations can be solved to find the elastic motion of the structure, and the motion of the center of gravity can be found using traditional rigid-body simulation techniques.

5.2.6 Modal (Eigen) Analysis

The final transformation is to use standard eigenanalysis methods, as in Chapter 3, to transform from the physical elastic degrees of freedom to modal elastic degrees of freedom, so that only the first few low-frequency modes can be used for the simulation. The eigenvector matrix $[V]$ transforms from the modal degrees of freedom, $\vec{\eta}$, to the physical degrees of freedom, $\vec{dx}_e$ as shown in equation 5.20.

$$
\vec{dx}_e = [V]\vec{\eta} \quad (5.20)
$$

The modal equations of motion are given in equation 5.21.

$$
[V]^T[M_e][V]\ddot{\vec{\eta}} + [V]^T[K_e][V]\vec{\eta} = [V]^T\vec{F}_e \quad (5.21)
$$

If only the first m modes are used, $[V]$ has m columns so that equation 5.21 gives a dynamic system with m elastic degrees of freedom.

5.2.7 Rigid-Body Equations of Motion

To find the rigid-body motion, the standard aircraft body-fixed equations of motion are used, equation 5.22, from [2].

$$F_x - mg\sin(\Theta) = m[\dot{U}_r + Q_r W_r] \tag{5.22a}$$

$$F_z + mg\cos(\Theta) = m[\dot{W}_r - Q_r U_r] \tag{5.22b}$$

$$M_p = I_{YY}\dot{Q}_r \tag{5.22c}$$

$$\dot{\Theta} = Q_r \tag{5.22d}$$

$$\dot{X}_i = U_r\cos(\Theta) + W_r\sin(\Theta) \tag{5.22e}$$

$$\dot{Z}_i = -U_r\sin(\Theta) + W_r\cos(\Theta) \tag{5.22f}$$

F_x and F_z are the forces on the aircraft, M_p is the pitching moment, U_r, W_r and Q_r are the velocities in the body-fixed coordinate system, and X_i, Z_i and Θ give the position in an inertial coordinate system. The use of the body-fixed coordinate system makes the aerodynamic calculations much simpler and is the standard technique [2] [19].

The final rigid and elastic equations of motion, equations 5.22 and 5.21, are expressed in terms of the modal degrees of freedom $\vec{\eta}$ and $\dot{\vec{\eta}}$, and the rigid-body state variables X_i, Z_i, Θ, U_r, W_r and Q_r that describe the motion of the BWB's center of gravity. These degrees of freedom are the most concise for finding the motion of the structure, but at times they must be presented in other forms. They need to be transformed to the other geometric descriptions in the simulation, to show the structural deflections at the finite element nodes, and to solve for the aerodynamic loads.

5.2.8 Converting Between Degrees of Freedom

There are several systems of geometrical descriptions used within the simulation. There are the degrees of freedom of the equations of motion (modal amplitudes and rigid-body states), the finite-element degrees of freedom (displacements of the structural nodes and the center node), and the aerodynamic description (positions and velocities of the panels). Conversions between the different systems for both position and velocity must be made at every time point in the simulation, and are a significant fraction of the total calculation time when optimizing with the simulation.

5.2.8.1 Dynamic Degrees of Freedom to Physical Nodes

The aerodynamic force calculations require that the equations of motion degrees of freedom be translated into the motion of the panels, in a body-fixed coordinate system. The aerodynamic calculations treat the aircraft as if it were fixed in a moving fluid, and following this convention the freestream velocity in the aerodynamic code is attributed to the rigid-body velocities U_r and W_r. The motion of the aerodynamic panels in the moving fluid is found by adding the elastic dynamic motion to that caused by the rigid-body rotation, Q_r. This motion enters both the boundary conditions and force calculations of the quasi-steady panel code.

Equations 5.23, 5.24, and 5.25 summarize the way the dynamic degrees of freedom are expressed for the inputs to the panel code, and show how the decoupled rigid and elastic equations become coupled again through the force calculations.

The freestream velocity and angle of attack are due to the rigid-body velocities:

$$U_\infty = \sqrt{U_r^2 + W_r^2} \tag{5.23}$$

$$\alpha = \arctan \frac{W_r}{U_r} \tag{5.24}$$

The elastic motion and rigid-body rotation rate are used to find the dynamic motion of the panels in the aerodynamic analyses:

$$\dot{\vec{dx}}_{1c} = Q_r\{\vec{dx}_{rb3}\} + [TX][V]\dot{\eta} \tag{5.25}$$

To find the actual positions of the finite element nodes, a little linear algebra must be performed. In the case of linear displacements, the motion from the rigid and elastic modes can be summed together:

$$\vec{dx}_{1c} = X_r\{\vec{dx}_{rb1}\} + Z_r\{\vec{dx}_{rb2}\} + \Theta\{\vec{dx}_{rb3}\} + [TX][V]\{\vec{\eta}\} \tag{5.26}$$

However, since the nonlinear flight-dynamic equations have been used, the small rotation of Θ is no longer a safe assumption, and a real rotation matrix $[R(\Theta)]$ [25] must be used to find the position of the nodes in space:

$$\vec{dx}_{1c} = X_r\{\vec{dx}_{rb1}\} + Z_r\{\vec{dx}_{rb2}\} + [R(\Theta)]\,\{[TX][V]\{\vec{\eta}\}\} \tag{5.27}$$

Of course, once the degrees of freedom $\vec{dx}_{1c}$ have been found, it is a simple matter to convert them to the actual nodes of the finite element structure (including the root rib nodes) using equation 5.7.

5.2.8.2 Inertial Loads

Another coupling between rigid and elastic dynamics that must be included explicitly is the inertial load (from D'Alembert forces) that is induced on the structure by rigid-body accelerations. For instance, accelerating an aircraft rapidly upwards should make the wing bend downwards from inertial forces, but these forces have been excluded from the elastic equations of motion. All that the decoupling of the rigid and elastic motion accomplishes is to provide a set of elastic deflections that does not move the center of gravity or produce any net change in the angular momentum about the center of gravity, so that the elastic deflections can never inertially affect the rigid-body motion. The converse is not true: the rigid-body accelerations can affect the elastic deformations, so the D'Alembert forces due to linear acceleration, centrifugal and coriolis forces must be added to the elastic system if they are not negligible.

5.2.8.3 Aerodynamic-Structural Coupling

Applying the aerodynamic forces to the structural nodes and incorporating the structural deflections and nodal velocities to the deflection and motion of the aerodynamic panels can be a complete topic of research in and of itself [20] [26] [52] [49]. A simple method is used here, so that it may be computationally efficient and easy to implement. The simplicity comes primarily from the assumption that the chordwise bending of the wing is small compared to the spanwise bending, which is valid for the high aspect-ratio outer wing panel sections. This assumption allows the use of one-dimensional curve fits when translating the aerodynamic forces to the structural nodes, and allows neglecting chordwise camber changes when moving the structural deflections and velocities to the aerodynamic panels.

The aerodynamic forces are moved from the aerodynamic panels to the structural nodes using a one-dimensional curve-fit of the spanwise lift and moment distributions. The contributions of the two chordwise panels are summed together to define the lift and pitching-moment distributions. These curves are interpolated to find values for the lift and pitching moment carried by every rib in the structure. The linear

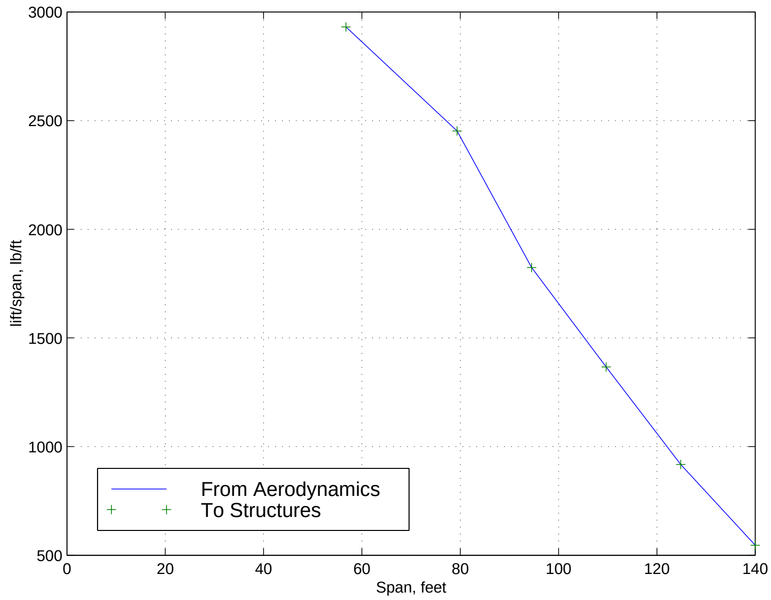

Figure 5.11: Wing Lift Discretization for Structural Loads

interpolation gives good resolution of the lift and moment distribution, as figure 5.11 shows for an example load case. The chordwise distribution of forces is then found by assuming the nodal forces vary linearly from leading to trailing edge, and adjusting the slope and offset of the line describing the chordwise force distribution so that the lift and moment are matched. The linear distribution is a fairly unrealistic assumption for a subsonic airfoil pressure distribution, and using two pressure 'mode shapes' was considered, where the amplitudes of the two modes would be selected to match the section lift and moment, and the mode shapes could be chosen to mimic the actual pressure distribution over an airfoil. The very small chordwise bending deflections seen with the linear distribution favored the simplicity of the linear distribution over any added accuracy, however.

The structural deflections and nodal velocities must be translated to the twist and motion of the aerodynamic panels. Again, the assumption of negligible chordwise bending simplifies this process. The positions of the nodes of the leading and trailing edge spars are used to define the deflected aerodynamic shape, neglecting any camber changes due to chordwise bending. The deflections of the leading and trailing edges of the spars are used to find the displacement and velocity of each wing section. These section properties are then interpolated spanwise from the structural nodes to find the properties at the edges of the aerodynamic panels. This method is efficient and in keeping with the assumptions made in the aerodynamic force distributions, because errors made in the chordwise bending due to the linear force distribution are not propagated into the aerodynamic deflections.

5.2.9 Initial Structural Sizing

The elements of the structural model must be sized to match the blended-wing-body's properties as closely as possible. Reference [4] includes a highly detailed NASTRAN model of the BWB, including composite elements and cabin pressurization. Static deflection cases were calculated using the high-fidelity model for several flight conditions. As an illustration of the level of detail of the model (and the reason a simpler model was constructed for this work), solutions for static deflection using a Cray supercomputer took several hours to calculate. No dynamic analysis was presented in [4], although the mass of the structure was modeled to obtain inertial loads in the static deflections. Results of two static load cases were used to size the structural elements of the new finite-element BWB model used in the collocation work.

An optimizer (NPSOL [23]) was used to find the skin and shear-web thicknesses and the spar-cap areas that gave deflections closest to those given in [4] in a least-squared sense. Elevons were deflected so that the aircraft was in trimmed static equilibrium. Static deflections were compared for two flight conditions: a 2.5 g pullup and -1 g push-over. Inertial loads for these flight conditions allowed the mass distribution of the structure to contribute to the deflections.

Figure 5.12 shows the results of the sizing optimization. The leading and trailing edge deflections are plotted in the figure, with good agreement between Boeing's model and the new collocation model. Sizing the structure statically does not, of course, guarantee that the dynamics of the new structural model will match the dynamics of the NASTRAN model. Matching mode shapes and frequencies would be the preferred method of sizing the dynamic model; however this data was not available at the time the sizing was done. Based on the inertial loading in the static cases, the new structural model should exhibit dynamic behavior similar to the BWB's, whose structural design is of course not finalized with the results of [4], and the model will certainly be accurate enough to draw conclusions about using collocation with this class of simulation.

5.3 Optimization Problem

The simulation of the blended-wing-body finds the time history of the longitudinal flight dynamics of the aircraft and the elastic dynamics of the outer wing structure. The simulation can also calculate the aerodynamic lift and drag, structural weight, and structural stresses at every point in the simulation, and can analyze a vast variety of control-system designs, center of gravity locations, fuel weight and payloads. The complete multi-disciplinary analysis and optimization of the blended-wing-body aircraft is a huge task, suitable for a complete thesis (such as [55]). For example, the MDO studies in [4], which do not include any dynamic simulations of the aircraft or its structure, analyzed the BWB at 20 design conditions, trimmed to 18 flight conditions and subject to 705 constraints. The object of the work in this chapter is to show that a simulation capable of being used by such an MDO study can also be used effectively within a collocation method framework, and to answer this question simpler optimization problems may be posed.

To see how well the multidisciplinary blended-wing-body simulation works within the collocation framework, a stability and control design problem was created. The goal is to move the center of gravity as far aft as possible while maintaining stable pitch dynamics, with the elastic motion included in the simulation. The location of the center of gravity (c.g.) of the BWB affects the longitudinal stability of the aircraft, making it less stable as the c.g. moves aft. Achieving stable flight with an aft c.g. location can increase the aerodynamic performance of the aircraft and allow more freedom in the distribution of fuel, payload, and passengers. To solve the collocation problem, the optimizer controls the center of gravity location, the feedback gains for the stability-augmentation system, steady-state control inputs for trim, and

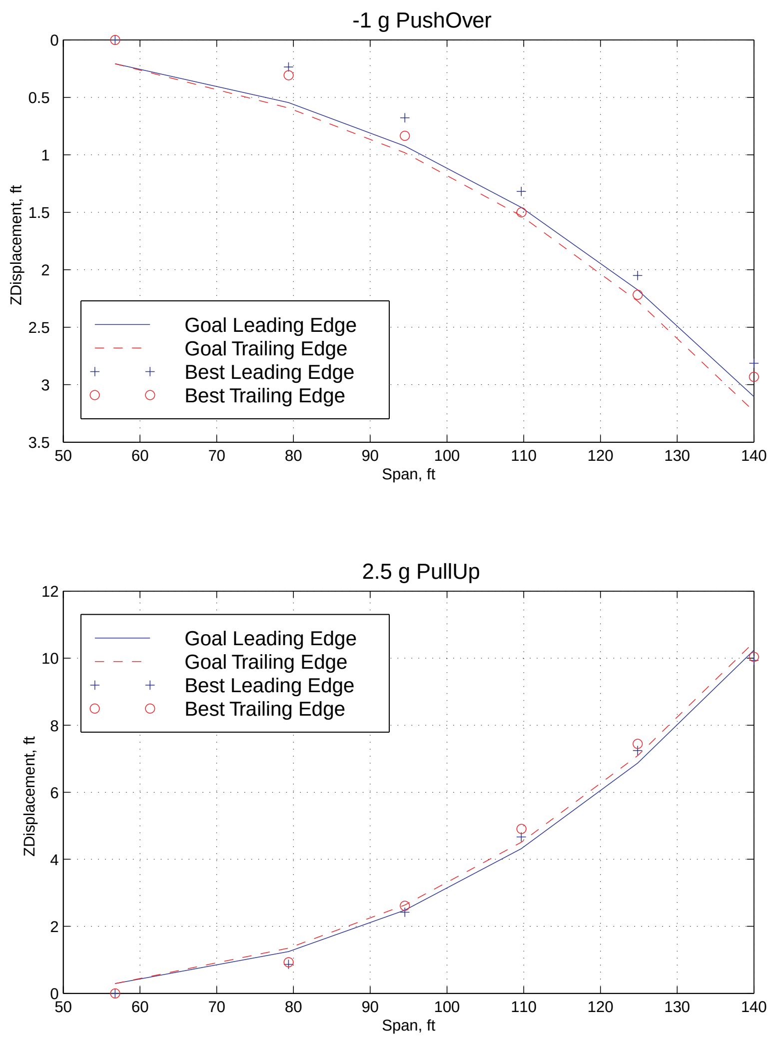

Figure 5.12: Static Sizing Results

the state variable time history. The optimizer's ability to shift the center of gravity aft is constrained by the available control authority and by stability constraints that require well-damped pitch oscillations.

The simulation of the cruise condition with a damping constraint is but one aspect of the flight performance envelope in which the stability augmentation system will operate. It is usual to adapt optimization problems to include constraints that become apparent after initial runs (e.g. [55]), so we are most concerned in this section with finding a reasonable solution to the problem posed, knowing that if solutions can be found, the correct physical problem can be developed if new constraints are necessary.

5.3.1 Design Variables

The optimizer sizes the state variables in the time history to solve the equations of motion and at the same time must move the center of gravity, size the feedback gains, and trim the plane using the control-surfaces and thrust. The feedback gains control each of the three elevons equally, although each elevon has a different trim value from which its motion is offset by the controller. Aside from the state variable time history, the design variables are:

x_{cg}	Center of gravity position
K_Θ	Θ to flap feedback gain
K_Q	Q_r to elevon feedback gain
δ_1	Inboard elevon trim angle
δ_2	Middle elevon trim angle
δ_3	Outboard elevon trim angle
T	Thrust

The states that the optimizer may vary to solve the equations of motion are:

X_i	C.G. Longitudinal position in inertial coordinates
Z_i	C.G. Vertical position in inertial coordinates
Θ	Rotation angle
U_r	Horizontal rigid-body velocity in local coordinates
W_r	Vertical rigid-body velocity in local coordinates
Q_r	Rigid rotation rate
$\eta_1 \ldots \eta_n$	First n elastic mode amplitudes
$\dot{\eta}_1 \ldots \dot{\eta}_n$	First n elastic mode velocities

The results that follow used six structural modes, for a total of 18 state variables. The sharp-eyed reader has no doubt already noticed that the control-surface states are not included in the list of state variables; the simpler actuator model that instantly moves as commanded until it reaches its stop was used. This choice greatly reduces the size of the optimization problem, by eliminating three elevon positions and three elevon velocities from the states (2994 design variables), but even more importantly it allows the time-step to increase from the tiny step needed to accurately simulate effective control-surfaces. Even with this simplification, the baseline five-second simulation requires 500 time points (including the initial conditions) to properly resolve the high-frequency elastic motion. Thus the problem is solved with 8989 design variables (7 control variables, 499×18 state variables), and 8912 constraints.

5.3.2 Dynamic Stability and Control Authority Constraints

The optimizer's objective is to move the center of gravity as far aft as possible. It is limited by a stability constraint placed on the rotation rate Q_r that forces the pitch oscillations to be damped to a small value within 2.5 seconds. The constraint forces the optimizer to design a control system that is both stable enough to provide the damping, and also trimmed so that the steady-state pitch rate is close to zero.

The stability constraint requires that the pitch rate be within ± 0.15 degrees per second after 2.5 seconds of simulation. The initial conditions start the aircraft with no elastic deflection and zero angle of attack, so the motion is that of the aircraft coming to 1-g equilibrium from approximately zero-g flight.

The optimizer may trim the aircraft using the elevons and varying the thrust so that the final pitch rate lies within the stability bounds. The trimming degrees of freedom are necessary along with the feedback-control gains so that the optimizer has the power to change the design to meet its constraints as the center of gravity moves.

The control authority of the aircraft is limited by the size of the control-surfaces and their maximum deflection. The maximum deflection for all surfaces is ± 40 degrees. As we will see, these limits give the controller plenty of authority to make the open-loop aircraft quite unstable and still control the cruise-condition simulation.

5.3.3 Finite Difference Gradients

To follow the idea of testing the applicability of collocation to complex simulations, a finite-difference approach that is independent of the simulation has been implemented. Previous chapters used finite-differences only for a few design variables whose relationship to the equations of motion were difficult to analyze, and analytic gradients for nearly all the constraints. As the analyses become more complex, the equations of motion cannot be solved to find analytic gradients without using specialized techniques [10], and moreover the sparsity of the Jacobian can be exploited for very efficient finite-difference gradients.

As Chapter 2 has explained, the sparsity in the Jacobian allows multiple design variables to be perturbed simultaneously when calculating the Jacobian elements by finite-differences. The method of [15] was implemented in the collocation simulation to calculate the most efficient scheme of perturbing multiple variables given the sparsity pattern of the Jacobian matrix. The savings in calculations using this method is immense compared to blindly using finite-differences; for example the 385,902 nonzero Jacobian elements for the 8989 design variable, 8982 constraint BWB simulation were calculated with one-sided differences using only 43 constraint evaluations.

The objective gradient was easily found analytically, as the objective, x_{cg}, is a design variable. The gradient of a design variable with respect to all the design variables is clearly 1 for the objective design variable and 0 for the other design variables. In general, of course, the objective function is not a design variable and the gradient must be found either analytically or numerically, as the form of the objective function suggests.

The finite-difference step sizes, and scalings for the design-variables and constraints were estimated using standard optimization techniques, a good review of which are found in [1], [24], and [55].

5.4 Collocation Results

The results of the control-sizing problem show that collocation can solve problems with the level of complexity of the blended-wing-body simulation. The center of gravity was moved aft until the closed-loop control could no longer keep the motion inside the stability constraints. Figure 5.13 shows the pitch rate Q_r and flight path angle Θ for the final simulation. The elastic motion is plotted in figure 5.14, and its influence on the dynamics can be seen in the high-frequency pitch rate oscillations. The stability constraint on Q_r is active, as we can see in the figure where Q_r just touches the bounds shown by dashed lines. The optimization process was quite a job for the Silicon Graphics Octane workstation used to find the solution, taking just under three days to complete, with 26 major iterations.

The feedback control does a good job of stabilizing the motion, as we can see from running the simulation with the final aft c.g. position but without the control system. Figure 5.15 shows the pitch rate and flight path angle for the open-loop simulation. The plane pitches up rapidly past 90 degrees in the first 5 seconds of flight, and is clearly unstable. The large pitch changes load the structure very highly, as the modal amplitudes in figure 5.16 show. The elastic deflections are almost 10 times greater without the feedback control.

The optimizer is able to move the center of gravity 42 feet while preventing divergence from the initial conditions specified. The final location is farther aft than any proposed center of gravity ranges for the blended-wing-body, and is almost assuredly not the optimum center of gravity location for the final design. The fact that the optimizer is able to move the center of gravity so far leads to the conclusion that there is room in the design for relaxed stability, and that the simulation should include effects

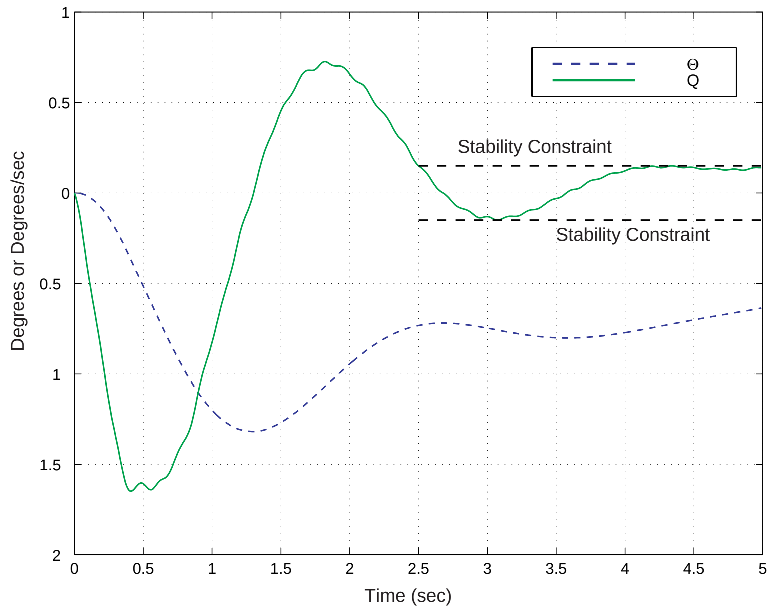

Figure 5.13: Aircraft Pitch Time History

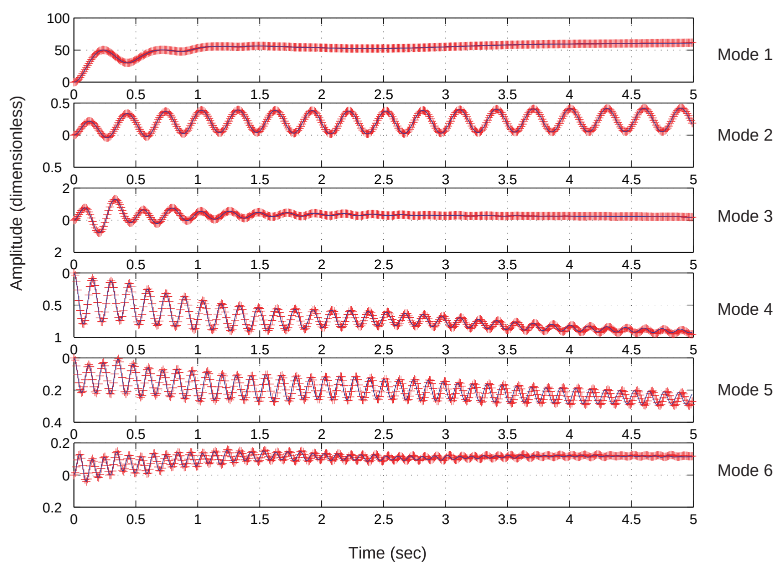

Figure 5.14: Structural Modes Time History

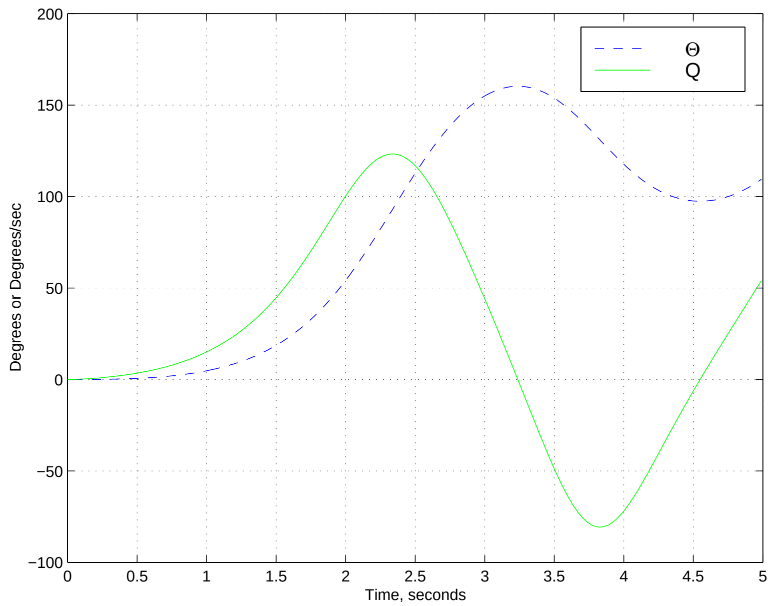

Figure 5.15: Open-Loop Pitch Time History

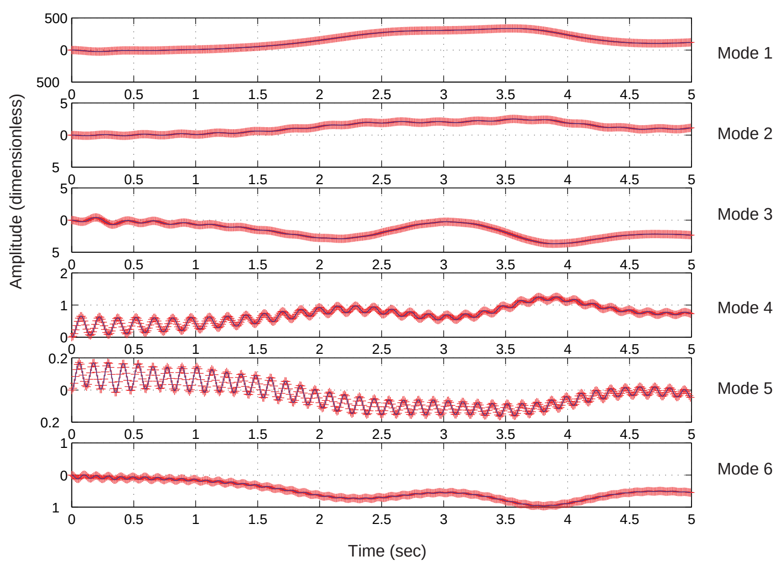

Figure 5.16: Open-Loop Structural Modes Time History

such as finite-bandwidth control actuators and nonlinear flap aerodynamics. Also, a 5 second simulation at one flight condition does not include all the possible cases that influence the design of the stability augmentation system. Nonetheless, the results show that the collocation method was quite capable of solving the problem as it was posed, and designing a stable, trimmed system.

Close inspection of figure 5.13 reveals that the stability constraint is active in three places. The motion is touching both the upper and lower bounds in the last part of the simulation, and the first oscillation touches the bound at the instant the bound is implemented. The stability constraint affects the control design in two ways: towards the end of the simulation the amplitude of the oscillation is constrained, and the time at which the bound is activated constrains the frequency of the initial oscillation. The constraint has been used in previous chapters as an amplitude constraint, making sure that the oscillations stay within the upper and lower bounds as they do in the figure. The constraint on the frequency was not expected, and is due to the response making one large oscillation after being released from the initial conditions, and the relationship of the pitch oscillation frequency to the location of the center of gravity. As the center of gravity moves aft, the short-period frequency decreases, and unless it is modified by the closed-loop control the initial oscillation will violate the bound constraint by coming at it from above. The results show that the optimizer has moved the center of gravity so far aft that the closed-loop response just satisfies the stability constraint in both amplitude and frequency.

Figure 5.14 shows that the second elastic mode of the wing is not well-damped in the final design. While the structural deflections due to this oscillation are small, it is clear that the combination of the aft-c.g. and high-gain feedback has coupled with the structural dynamics and caused a flutter-like motion. This illustrates the need for the complete simulation of the aeroelastic dynamics and control-system together, so that structural dynamics such as this oscillation can be eliminated. No stability constraints were placed on the structural dynamics in this optimization, because no structural design variables were used, and flutter was not expected, but these results show that stability constraints on the structural dynamics should be included in future optimizations to keep the system well-damped.

This optimization problem was quite time-consuming for the computer to solve. The increase in complexity and fidelity of the simulation model is the primary cause of the increase in CPU time over the collocation problems solved in previous chapters, since the number of major iterations remains about the same. The increase in the size of the optimization problems from previous work also contributed to the longer run-times. While faster computers than the SGI workstation that was used can be found, the usefulness of collocation optimization is reduced if supercomputers must be used, and methods to speed up the collocation implementation should be sought so that more complex simulations may be used.

5.5 Collocation Simulation Decomposition

The easiest and most reliable way to decrease the time spent solving collocation problems is simply to wait. During the course of this research, over a span of about four years, the computers available to the author to solve the optimization problems have increased in speed by about a factor of 5. Of course this method of increasing speed has no effect on the calculation times with a given computer, but does indicate that in the future the problems that today's designers abandon as too large or complicated will be solved without trouble.

There are other means to re-formulate the optimization problem that may speed it up. Parallel processing can be used [9] [12] to simultaneously evaluate the constraints and their gradients, since the collocation method transforms the simulation from a sequential calculation into a parallel set of calculations. A method for single-processor computers, that reduces the size of the optimization problem but increases the calculations required in the constraint evaluations, was tested on the BWB aeroservoelastic simulation.

5.5.1 Coarse-Grained Decomposition Using Collocation

The large number of state variables and the small time step required in the blended-wing-body collocation simulation creates a large optimization problem, with almost

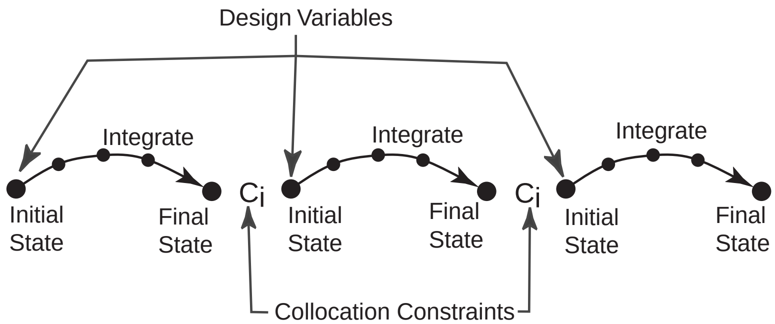

Figure 5.17: Decomposition of Equations of Motion

9000 design variables and constraints for a five second simulation. Since the number of calculations performed by the optimizer with each iteration is related to the number of design variables and constraints, reducing the number of design variables should speed up optimization, all other things being equal. This is why the time step used should be as large as possible while retaining accuracy.

One method to reduce the number of optimization variables for a given simulation is to apply the collocation constraints to every few points in the time simulation, with the intermediate points found by numerical integration of the equations of motion. The simulation can be broken into small blocks, the initial conditions of which are collocation design variables, from which a few steps of the equations of motion are integrated. The final state from integrating the equations of motion on one block is required to match the initial conditions of the next block through the collocation constraints. Figure 5.17 shows this method of solving the equations of motion. The sequential simulation is still broken down into parallel calculations, but there are fewer of them and each one is a sequential calculation. The optimization problem size (in terms of the number of design variables and constraints) is reduced by a factor equal to the number of integrated time steps used in the collocation constraints.

This decomposition method reduces the size of the optimization problem, but at the same time increases the computational cost of evaluating the constraints by adding the integrated segments of the time history. Any time savings will depend on the computational cost of the optimizer versus the computational cost of the constraint evaluations. Figure 5.18 shows the run-times and problem size of several

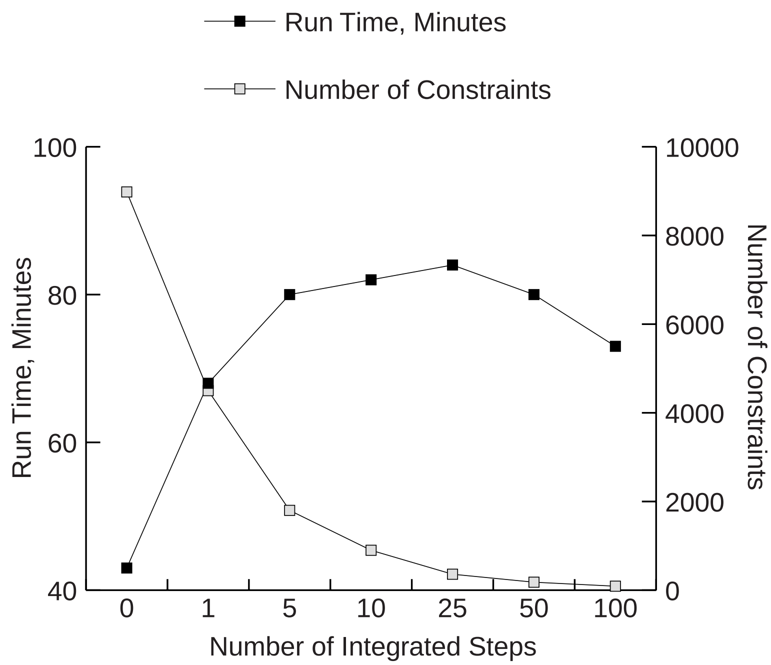

Figure 5.18: Decomposition Results

BWB simulations using this decomposition method. The optimization problem was simplified by fixing the center of gravity location and control design variables to make the problem one of constraint satisfaction only, so that the computer solution would take hours instead of days.

The run times in figure 5.18 show that the smaller optimization problems take longer to solve than the problems with fewer integrated steps. With no integration at all (0 integrated steps) the run time is the shortest by a wide margin. The run times continue to increase as the optimization gets smaller, until it gets very small, at which point the times decrease but are still greater than the original collocation method. With the time-consuming calculations of the BWB simulation, there is no benefit to simplifying the optimization if the constraint evaluations become more difficult in the process.

5.5.2 Parallel Computations

Solving the collocation problem using parallel processors should speed up the solution process immensely. Since most of the optimization time is spent evaluating the constraints and their gradients, and these calculations are independent of one another and therefore easily parallelized, a parallel collocation implementation should see real time savings. Parallel calculations are slowed by communications between processors, such as the optimizer processor sending the current design variables to the constraint-evaluating processors. For this reason the decomposition method using integrated steps between collocation constraints could potentially yield time savings on a parallel machine by reducing the number of design variables passed, which in turn reduces the communication overhead. The integrated states are unknown to the optimizer, and calculated internally in the constraint routine. It is possible that the time saving due to a smaller optimization problem, combined with time savings due to a lower level of communication between processors, could outweigh the extra computation time due to integrating equations in the constraint evaluations. This is a good topic for further research.

5.6 Conclusions

Combining the complex finite-element simulation, that includes unrestrained degrees of freedom, closed-loop stability augmentation, and aerodynamic calculations with control-surface deflections in the collocation method framework showed that the method can solve problems with time-consuming dynamic equation calculations in a reasonable time. The more complex the problem, of course, the longer the optimization time, but the method is well within the computational speed range needed for typical multidisciplinary analyses and optimizations on aircraft such as the blended-wing-body.

The attempt at speeding up the collocation calculation time by reducing the size of the optimization problem did not pay off as hoped, because in the process the constraint calculations were made more difficult. The use of parallel computer processors does hold promise for greatly reducing the optimization time for complex simulations such as that of this chapter, and the decomposition scheme may yield benefits when combined with this approach.

The simulation of the blended-wing-body showed that it is important to consider the often-separated disciplines of aeroelasticity and rigid-body control. The dynamics of the closed-loop control and elastic structure influenced each other to a large degree. Design problems with structural variables as well as the control-system variables used in this chapter would see an even stronger coupling of the disciplinary analyses in the optimization process. Because simulations in the time-domain are easier to combine into large multidisciplinary analyses, and easier to modify to include nonlinearities required for high-fidelity models, methods for optimizing time-domain simulations, such as collocation, are valuable tools for navigating today's design spaces.

Chapter 6

Conclusions and Recommendations

In this thesis, a collocation method that was initially developed for spacecraft trajectory optimization was applied to a family of new aircraft design problems, with aeroelastic and closed-loop-control design and simulation. Collocation was shown to be an effective method of optimizing aircraft with time-domain, nonlinear aeroelastic simulations, and to apply damping constraints similar to those calculated with a frequency-domain analysis. The method was shown to be equally effective for the design of closed-loop control systems. The collocation method's domain was extended by using two new aeroelastic analyses: a beam-based linear finite-element analysis for high-aspect ratio wings, and a spar, rib, and skin model with the ability to simulate the actual structure used in an aircraft wing. Both of these structural analyses were coupled with quasi-steady vortex-lattice aerodynamic analyses, to form the complete aeroelastic or aeroservoelastic simulations.

The collocation algorithm was improved with a new form of collocation constraint and the addition of a stability constraint. The new collocation constraint enforces the equations of motion using the second-order state derivatives (the accelerations), which are typically available in time-domain simulations but often hidden to create first-order equations. The stability constraint that was developed in this thesis is a powerful addition to the collocation method because it allows the designer to specify the shape of the dynamic response. The large, computationally intensive optimization problem of the last chapter showed that efforts to speed up the collocation are worthwhile. A decomposition method was developed that decreased the optimization problem size but increased the solution time by making the constraints more computationally intensive. Parallel processors could be used to speed up the collocation process, as it is well-posed for parallelization, and the decomposition method may yet be of use by reducing the communication load between processors.

6.1 Collocation is Suitable for a Variety of Dynamic Simulations

This thesis extended the collocation method from its spacecraft simulation roots to a new set of aircraft optimization problems that include damping constraints on the motion. Structural dynamics and collocation had never been used together prior to this work, and the method was shown to be very compatible with time-domain aeroelastic simulations, from the typical-section simulation of Chapter 2 to the elastic dynamics of the blended-wing-body, including unrestrained-vehicle boundary conditions and closed-loop control, studied in Chapter 5.

Structural optimization is a problem well-suited for collocation, as the jet transport optimizations of Chapter 3 showed. The wing weight was minimized subject to a flutter constraint, and the collocation method solved linear and nonlinear simulations with the same problem formulation and very similar performance. The stability constraint was able to force stable systems to be designed with a wide range of amplitudes, showing that its application does not require detailed knowledge of the system.

The damping constraint used to control flutter in Chapter 3 was shown to be equally effective for closed-loop control design, by designing a gain-schedule for the BWB flight control testbed. The optimizer, without knowledge of the functional relationship between gain and airspeeed, designed a control law that obeyed this relationship strictly based on the level of damping prescribed by the stability constraint, indicating that collocation would be quite useful in cases too complex to design by hand and where the feedback system relationships are not clear. The flight conditions for the aircraft to become unstable were also predicted using neutrally-stable stability constraints, and these results were compared with data from flight tests.

The successful solution of aeroelastic and closed-loop control problems with collocation led to the development of an aeroservoelastic analysis tool that combined elastic dynamics with rigid-body motion and closed-loop control, to show that the collocation method is capable of optimizing the very large optimization problem that such a time-consuming nonlinear simulation creates. The blended-wing-body collocation simulation could be used to optimize the design with respect to many different design goals and flight conditions.

6.2 Modifications to the Collocation Method

As part of the process of solving the optimization problems just described, the collocation method was improved by the addition of a new form of collocation constraint that improves the accuracy of the solutions to the equations of motion, and a stability constraint that allows simulations with constraints on the dynamic response to be tackled with collocation.

6.2.1 Collocation Constraint Formula

The Taylor series collocation constraint uses the second derivative of state variables, which is found in equations of motion that are based on Newton's second law, $F = m\ddot{x}$. While many numerical methods transform the equations of motion into a set of first-order differential equations through the definition of a state variable, hiding the acceleration derivatives in order to be able to solve the equations, the Taylor series constraint takes advantage of the acceleration derivatives, using them to create a higher-order curve-fit for the collocation constraint. This constraint form allows larger time-steps to be used than a first-order constraint such as the trapezoidal constraint, and has the physically realistic property that the derivative of the position curve fit gives the velocity curve fit everywhere in the simulation.

6.2.2 Stability Constraint

The stability constraint uses upper and lower bound side constraints on the design variables to force the dynamic motion to fit within an "envelope" specified by the designer. The envelope may be shaped to force the motion to be highly damped, or may allow the motion to be neutrally stable. This constraint opens up a whole new class of optimization problems to be solved with collocation and nonlinear time-domain simulations. The stability constraint is very computationally efficient, because side constraints are treated differently in optimization than explicit constraints, making their computational cost practically free.

When compared with a linear stability analysis, the wing structural designs using a collocation stability constraint to control flutter were found to be as well-damped as the constraint demanded. Additionally, when used to find the neutrally stable design, collocation again matched the results from frequency-domain calculations, with the advantage that collocation can be used for nonlinear time-domain simulations. The length of the collocation simulation was shown to have an effect on its ability to predict neutrally stable systems, since slowly diverging motion may take some time to become apparent in a simulation. The well-damped design problems, however, were much less sensitive to the simulation length, and in general the method proved to be quite robust in preventing divergence with a wide range of stability constraints and simulation lengths.

The stability constraint was also compared against experiment, using a nonlinear simulation of the flight-control-testbed's feedback control system, and the results of both flight tests and captive car-top testing. When applied to give a well-damped response, the stability constraint automatically guided the optimizer to match a gain-schedule that was tediously designed by hand and successfully tested in flight. The stability constraint was also used to find the neutrally stable set of gains for the flight-control system, which matched the experimentally-determined stability limit quite closely.

Because many interesting dynamic design problems require both nonlinear simulations and well-damped motion, the collocation stability constraint makes collocation methods applicable to a much broader variety of problems. Instead of linearizing and using frequency-domain analyses, the complete nonlinear simulation can be used with numerical optimization to find the best design. The neutrally-stable design can also be found to place limits on design parameters much like a root-locus plot for linear systems.

6.3 Aeroelastic Analyses for Collocation

The lack of suitable design tools for aeroelastic analysis and optimization led to the creation of two new analysis packages to support the investigations into the collocation method. Two different finite element codes, with the ability to analyze dynamic as well as static systems were created, and quasi-steady aerodynamic analyses were also developed from existing static analyses to meet the needs of the aeroelastic simulations.

The first finite-element model is based on a beam representation of the aircraft wing, and was used for flutter analysis of the jet transport wing from [11]. The typical-section aeroelastic model's dynamics are strongly influenced by the separation of the elastic axis and center of gravity, and this effect is included in the three-dimensional finite-element model by using two separate beam elements in the wing, one for stiffness and one for mass, and linking them together to a common structural node. The aerodynamic model is tightly coupled to the structural representation, with a fixed panel spacing based on the structural elements that allows fast transformation of the aerodynamic loads to the structure, and the structural deflections to the aerodynamic panels. The analysis is very well-suited to aircraft structures whose parameters can be specified with spanwise distributions of properties, yielding a complete set of dynamic equations with a very reasonable computational cost.

The second finite-element model was created to allow a more realistic structural representation of an aircraft wing, and was used for the aeroservoelastic simulation of the blended-wing-body. The beam model was replaced by the spars, ribs, and skins that make up an aircraft wing's structural box. This structural representation allows the element stresses to be calculated for structural design, and gives more freedom to change the structural parameters throughout the wing. The higher-fidelity simulation comes with a price of increased computational time to evaluate a design, as well as a more complicated aerodynamic-structural coupling method.

The blended-wing-body simulation needed to include the unrestrained, or rigid-body, longitudinal degrees of freedom, since coupling between rigid-body and elastic modes is known to be significant on some tailless aircraft [51]. Including the rigid-body dynamics of the complete aircraft required significant modifications to the finite-element matrices to add the inertial properties of the complete aircraft, and to decouple the rigid-body and elastic dynamics so that the nonlinear rigid-body equations of motion could be used.

The aeroservoelastic design of the blended-wing-body requires a closed-loop control system in the simulation. Aerodynamic control surfaces were modeled using a second chordwise panel in the vortex lattice code, and moved by control laws programmed into the equations of motion.

Modal analysis, using eigenvectors to transform the linear structural matrices into a system where the significant dynamics are captured by a few modal degrees of freedom, was shown to have a large impact on the efficiency of collocation in solving structural-dynamic optimization problems. Using the first few modes instead of the hundreds of physical degrees of freedom kept the number of state variables, and hence the optimization problem, relatively small. Additionally, neglecting the high-frequency dynamics allows the time-step in the simulation to be increased, further reducing the size of the optimization problem.

Along with the aerodynamic analysis tools, two aeroelastic aircraft models were created for the optimization problems. The jet transport wing model was created for the beam-based finite-element code, and the blended-wing-body model was created for the aeroservoelastic analysis. The properties of the jet transport wing were sized to match those given in [11], while the structural and inertial properties of the BWB were found in [3] and [4].

The analyses and models created for the work of this thesis are well-suited for collocation-based optimization because they are computationally efficient, and create a set of equations of motion that form a sparse optimization problem. Sparsity in the constraint Jacobian keeps optimization problems with thousands of design variables and constraints from overwhelming the optimizer with algebraic computations. The sparsity also allows very efficient finite-difference methods to be used to calculate the constraint Jacobian.

6.4 Future Work

The capabilities of the blended-wing-body simulation for multidisciplinary optimization have only been used to a small degree in this thesis. There are many other studies and optimizations that could be performed using these tools. For example, structural optimization using a flutter constraint like that of Chapter 3 could reduce structural weight. Synthesis of a well-founded structural and stability-augmentation system design problem with the appropriate flight conditions, dynamic constraints, and design objective is a logical extension of this work.

The fact that several days were required to find a solution to the blended-wing-body control design problem of Chapter 5 indicates that future work should investigate methods to increase the speed of collocation. Adding structural design variables along with stress and flutter constraints to the BWB control-system optimization problem would be a reasonable next case for the BWB simulation, but would be difficult to implement at the solution pace now seen. Parallel processing would seem to be a natural method to reduce optimization time, since the collocation architecture is well-suited for parallelization, as [9] and [12] have shown. Methods for breaking up the problem into parallel blocks should be investigated to see if decomposition as in Chapter 5 or methods such as those in [1] or [53] will reduce the optimization time.

The new optimizer SNOPT [22] could reduce optimization time by a great deal over the now-outdated MINOS. Its ability to satisfy tougher constraints and its need for fewer function evaluations should reduce the computation time by a large amount for the types of problems solved in this thesis. It is recommended that future work use this optimizer instead of MINOS.

There are other aeroelastic problems that would be interesting to pose using collocation. Flapping flight is seeing renewed interest in the micro-air-vehicle arena [44], and is a good example of highly elastic structures and complex, unsteady aerodynamics. The collocation method would be a good tool, for example, to optimize the flexible wing structure of an ornithopter to maximize thrust using accurate nonlinear aerodynamic models, and allowing a structural model with geometric nonlinearities. Structural design and actuation constraints could be included in the simulation so that the optimal design has a buildable structure and motion that is physically realizable with the chosen powerplant. Helicopter blade aeroelasticity and turbine blade aeroelasticity are two analysis disciplines that have many nonlinear components and could be solved with collocation. The winged keels on sailboats, particularly those of the America's cup class, also exhibit large structural deflections that may require nonlinear structural analyses, and operate with very complicated unsteady flows. Time-based simulation of an America's cup sailboat keel using a collocation method could provide the performance advantage constantly sought by their designers.

The time-domain simulation is an important tool for the engineer. Expanding the application of collocation methods from spacecraft trajectory optimization to aeroelasticity and closed-loop control is merely one avenue of the time-domain simulation. Other fields, both within aircraft design and outside it, use nonlinear time-domain simulations and optimization, and may benefit from collocation design techniques.

Appendix A

"Collocation Method" Anagrams

comical holden toot
calico holden motto
technical doom tool (or loot)
methodical not cool
economic total hold
alcoholic mood tent
chilled tomato loon
locomotion latched
clothed atomic loon
comical olden tooth

Courtesy of [32].

Bibliography

[1] Stephen Altus. *Multidisciplinary Aircraft Design Optimization Using Single-Level Decomposition.* Ph.D. thesis, Stanford University, 1997.

[2] Holt Ashley. *Engineering Analysis of Flight Vehicles.* Dover, New York, 1974.

[3] Blended-wing-body configuration control document (ccd2). Report CRAD-9405-TR-3780, Boeing, October 1997.

[4] Blended-wing-body configuration control document (ccd3). Report CRAD-9405-TR-3780, Boeing, October 1997.

[5] J. T. Betts and E. J. Cramer. Application of direct transcription to commercial aircraft trajectory optimization. *Journal of Guidance, Control, and Dynamics*, 18(1), Jan–Feb 1995.

[6] J. T. Betts and W. P. Huffman. Application of sparse nonlinear programming to trajectory optimization. *Journal of Guidance, Control, and Dynamics*, 15(1), Jan–Feb 1992.

[7] John T. Betts and William P. Huffman. Path constrained trajectory optimization using sparse sequential quadratic programming. In *Proceedings of the AIAA Guidance, Navigation and Control Conference*, number AIAA-91-2739. AIAA, August 1991.

[8] John T. Betts and William P. Huffman. Application of sparse nonlinear programming to trajectory optimization. *Journal of Guidance, Control, and Dynamics*, 15(1), January–February 1992.

[9] J.T. Betts and W.P. Huffman. Trajectory optimization on a parallel processor. *Journal of Guidance, Control, and Dynamics*, 14(2), March-April 1991.

[10] Christian Bischof, Peyvand Khademi, Andrew Mauer, and Alan Carle. ADIFOR 2.0: Automatic Differentiation of FORTRAN 77 Programs. *IEEE Computational Science and Engineering*, 3(3), Fall 1996.

[11] R. L. Bisplinghoff, H. Ashley, and R. L. Halfman. *Aeroelasticity*. Addison-Wesley Publishing Company, 1955.

[12] Robert Braun. *Collaborative Optimization: An Architecture for Large–Scale Distributed Design*. Ph.D. thesis, Stanford University, 1996.

[13] R. G. Brusch. A nonlinear programming approach to space shuttle trajectory optimization. *Journal of Optimization Theory and Applications*, 13:94–118, 1974.

[14] Arthur E. Bryson, Jr. *Dynamic Optimization*. Addison-Wesley, 1999.

[15] Thomas F. Coleman, Burton S. Garbow, and Jorge J. More. Software for estimating sparse jacobian matrices. *ACM Transactions on Mathematical Software*, 10(3):329–345, September 1984.

[16] E. D. Dickmanns and H. Well. Aproximate solution of optimal control problems using third-order hermite polynomial functions. In *Proceedings of the 6th Technical Conference on Optimization Techniques*, 1975.

[17] P. J. Enright and B. A. Conway. Discrete approximations to optimal trajectories using direct transcription and nonlinear programming. In *Proceeding of the AIAA/AAS Astrodynamics Conference*, 1990. Paper 90-2963.

[18] Paul J. Enright and Bruce A. Conway. Optimal finite-thrust spacecraft trajectories using collocation and nonlinear programming. *Journal of Guidance, Control, and Dynamics*, 14(5), September–October 1991.

[19] Bernard Etkin. *Dynamics of Flight: Stability and Control*. Wiley, 3rd edition, 1996.

[20] C. Farhat, M. Lesoinne, and P. LeTallec. Load and motion transfer algorithms for fluid/structure interaction problems with non-matching discrete interfaces: Momentum and energy conservation, optimal discretization and application to aeroelasticity. *Computer Methods in Applied Mechanics and Engineering*, 157(1-2), Apr 1998.

[21] G. Franklin, J. D. Powell, and A. Emami-Naeini. *Feedback Control of Dynamic Systems*. Addison-Wesley, 1994.

[22] P. E. Gill, W. Murray, and M. A. Saunders. *User's Guide for SNOPT 5.3: A FORTRAN Package for Large-Scale Nonlinear Programming*. Department of Operations Research Technical Report SOL 98–1, Stanford University, 1998.

[23] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright. *User's Guide for NPSOL 5.0: A FORTRAN Package for Nonlinear Programming*. Department of Operations Research Report SOL 94-?, Stanford University, 1994.

[24] P. E. Gill, W. Murray, and M. Wright. *Practical Optimization*. Academic Press, 1981.

[25] Donald T. Greenwood. *Principles of Dynamics*. Prentice-Hall, Englewood Cliffs, N.J., 2nd edition, 1988.

[26] Guru P. Guruswamy. Integrated approach for active coupling of structures and fluids. *AIAA Journal*, 27(6), Jun 1989.

[27] C. R. Hargraves and S. W. Paris. Direct trajectory optimization using nonlinear programming and collocation. *Journal of Guidance, Control, and Dynamics*, 10(4), July–August 1987.

[28] Charles Hargraves, Forrester Johnson, Stephen Paris, and Ian Rettie. Numerical computation of optimial atmospheric trajectories. *Journal of Guidance, Control, and Dynamics*, 4(4), July–August 1981.

[29] M. Holden and I. Kroo. A collocation method for aeroelastic optimization. In *Proceedings of the Sixth AIAA/USAF/NASA/ISSMO Multidisciplinary Analysis and Optimization Symposium*, 1996. AIAA-96-3980-CP.

[30] M. Holden and I. Kroo. Using collocation methods for aeroelastic optimization. In *3rd Annual Computational Aerosciences Workshop*. NASA Ames HPCCP program, August 1996.

[31] M. Holden and I. Kroo. Aeroelastic design and control for blended-wing-body configurations using a collocation method. In *Proceedings of the Seventh AIAA/USAF/NASA/ISSMO Multidisciplinary Analysis and Optimization Symposium*, 1998.

[32] `http://anagram.avatartech.com/anagram.html`

.

[33] William P. Huffman. Direct transcription and the optimal control problem: Putting the pieces together. Number AIAA-92-4527-CP. AIAA, 1992.

[34] Jack Morito Ii. Synthesis of aerodynamic transfer functions for elastic flight vehicles. *Journal of Aircraft*, 15(11):777–785, Nov 1978.

[35] Antony Jameson. Re-engineering the design process through computation. *Journal of Aircraft*, 36(1), January-February 1999.

[36] Samuel Johnson. *A Dictionary of the English Language*. 1st edition, 1774. From the Preface.

[37] D. Kraft. On converting optimal control problems into nonlinear programming problems. *Computational Mathematical Programming*, F15, 1985. NATO ASI Series, Springer-Verlag, Berlin.

[38] D. Kraft. Nonlinear system analysis by direct collocation. In *Lecture Notes in Control and Information Sciences, Optimal Control*, 1987.

[39] Erwin Kreyszig. *Advanced Engineering Mathematics*. Wiley, 6th edition, 1988.

[40] Ilan Kroo. *Aerodynamics, Aeroelasticity, and Stability of Hang Gliders*. Ph.D. thesis, Stanford University, 1983.

[41] Eli Livne. Integrated aeroservoelastic optimization: Status and direction. *Journal of Aircraft*, 36(1), January-February 1999.

[42] Valerie Manning. *Large Scale Design of Supersonic Aircraft Via Collaborative Optimization*. Ph.D. thesis, Stanford University, 1999.

[43] Leonard Meirovitch. *Elements of Vibration Analysis*. McGraw-Hill, 2nd edition, 1986.

[44] Robert C. Michelson. Update on flapping wing micro air vehicle research: Ongoing work to develop a flapping wing, crawling entomopter. In *Proceedings of the 13th Bristol International RPV/UAV Systems Conference*, 1998.

[45] B. A. Murtagh and M. A. Saunders. *MINOS 5.4 User's Guide*. Department of Operations Research Report SOL 83–20R, Stanford University, 1983.

[46] Duc T. Nguyen. Finite element software for multidisciplinary design optimization. Final Report NAS1-19858, NASA, 1995.

[47] A. Plotkin and J. Katz. *Low Speed Aerodynamics: From Wing Theory to Panel Methods*. McGraw-Hill, 1991.

[48] J. S. Przemieniecki. *Theory of Matrix Structural Analysis*. Dover, New York, 1968.

[49] James J. Reuther, Juan J. Alonso, Joaquim R. R. A. Martins, and Stephen C. Smith. A coupled adjoint-based method for aero-structural optimization of complete aircraft configurations. In *Proceedings of the 37th AIAA Aerospace Sciences Meeting*, 1999.

[50] James L. Rogers. Tools and techniques for decomposing and managing complex design projects. *Journal of Aircraft*, 36(1), January-February 1999.

[51] J. Schweiger, O. Sensburg, and H. Burns. Problems and structural design of a tailless cfc sailplane. In *Proceedings of the Second International Symposium on Aeroelasticity in Structural Dynamics*, 1985.

[52] Marilyn J. Smith, Carlos E.S. Cesnik, Dewey H. Hodges, and Kenneth J. Moran. Evaluation of Computational Algorithms to Interface Between CFD and CSD Methodologies. In *Proceeding of the 1996 37th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 1996.

[53] Ian Sobieski. *Multidisciplinary Design Using Collaborative Optimization*. Ph.D. thesis, Stanford University, 1998.

[54] B. Tigner, M. Meyer, M. Holden, B. Rawdon, M. Page, W. Watson, and I. Kroo. Test techniques for small-scale research aircraft. In *16th Applied Aerodynamics Conference*, number AIAA-98-2726. AIAA, June 1998.

[55] Sean Wakayama. *Lifting Surface Design Using Multidisciplinary Optimization*. Ph.D. thesis, Stanford University, 1994.

[56] J Weissinger. *The Lift Distribution of Swept-Back Wings*. NACA TM 1120, 1947.