

Aeroelastic Study of a Multi-Hinged Wing

by

Torrey Owen Radcliffe

S. B., Aeronautics and Astronautics
Massachusetts Institute of Technology (1997)

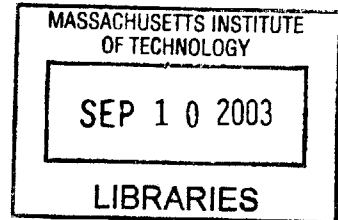
S. M., Aeronautics and Astronautics
Massachusetts Institute of Technology (2002)

Submitted to the Department of Aeronautics and Astronautics in Partial
Fulfillment of the Requirements for the Degree of

DOCTOR OF PHILOSOPHY IN AERONAUTICS AND ASTRONAUTICS
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2003

© Massachusetts Institute of Technology 2003. All rights reserved



Author _____

Department of Aeronautics and Astronautics
October 15, 2002

Certified by _____

Carlos E. S. Cesnik
Visiting Associate Professor of Aeronautics and Astronautics
Chair, Thesis Committee

Certified by _____

Mark Drela
Professor of Aeronautics and Astronautics
Member, Thesis Committee

Certified by _____

John Dugundji
Professor Emeritus of Aeronautics and Astronautics
Member, Thesis Committee

Accepted by _____

Edward M. Greitzer
H. N. Slater Professor of Aeronautics and Astronautics
Chair, Committee on Graduate Students

Aeroelastic Study of a Multi-Hinged Wing

by

Torrey Owen Radcliffe

Submitted to the Department of Aeronautics and Astronautics
on October 15, 2002, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Aeronautics and Astronautics

Abstract

Dynamic aeroelastic response of multi-segmented hinged wings is studied theoretically and experimentally in this thesis. For the theoretical study, a method of modeling the aeroelastic characteristics of multi-hinged wings is proposed. The method employs the Runge-Kutta scheme to solve the governing equations of a flexible multibody dynamic system. The Hénon method is used to switch between bilinear stiffness states of the wing in bending. Experimental wind tunnel tests of one- and five-hinged wings were conducted for better insight into the mechanics of the motion. Correlation between the experimental and theoretical results is presented. The theoretical model is found to capture both the linear and nonlinear aeroelastic behavior of a hinged wing. Adding hinges to a wing is found to significantly alter the speed at which an instability will occur. The stiffness of the hinges is found to play a major role in the determination of flutter speeds with a reduction in hinge stiffness nominally leading to an increase in first bending / first torsion instability speeds. However, for low hinge stiffness, hinged wings were also found to have the possibility of a second bending / first torsion instability at speeds far below the first bending instability. The hinged wing is found to enter into chaotic or limit cycle motion at speeds at, near, or above flutter speeds. The bi-linear nature of a hinge is found to cause a disruption in the coalescence of modes. This limits the energy added to the system while it is in an unstable state. The hinges allow the wing to “fold” under low net loads. The theoretical model can be used for aeroelastic design of future hinged wings for remotely deployable vehicles.

Thesis Supervisor: Carlos E. S. Cesnik

Title: Visiting Associate Professor of Aeronautics and Astronautics

Acknowledgments

There are many people who I wish to thank for their help and support which made this possible. First, I would like to thank my advisor, Professor Carlos Cesnik, who introduced me to the field of aeroelasticity. Without his guidance none of this would have been possible. Despite taking a position at a different university, Professor Cesnik was willing to support all of his old students through the finish of their degrees. On top of that, he is able to spot a sentence with a missing comma from a hundred paces.

Professor John Dugundji was always able to offer a bit of helpful advise on structural dynamics, experimental flutter work and thesis preparation. Professor Dugundji was kind enough to take over for much of the advisor duties that Professor Cesnik could not perform from a distance.

Professor Mark Drela was able to provide invaluable support helping to stir the thesis on the proper course. I would also like to mention Professor Karen Wilcox and Tienie van Schoor who were kind enough to read the thesis before the defense despite its numerous spelling errors.

Professor Eugene Covert, Frank Durgin, Richard Perdichizzi, and Jadon Smith were invaluable in their assistance with the wind tunnel experiments. I would like to thank all of the people who worked in TELAC and put up with me taking up all the computer time to run my inefficient code. A special thanks to Dennis Burianek, who taught me the in's and out's of using \LaTeX which this manuscript was written in.

My room-mates Matthew Congo and Nicole Krumrei have always been there to help me take my mind off of my work and to read through some of the early drafts of this work. I also need to show my appreciation for all the support and encouragement I relieved from Andrea Green over the years, and the years to come.

A special thanks goes out to the folks over at the MIT Edgerton Center for their loan of the high speed video equipment. Finally I need to thank Draper Labs who provided the initial funding for this project and has generously let me borrow one of the wings from their WASP prototype.

Contents

Abstract	i
Acknowledgments	iii
1 Introduction	1
1.1 Motivation	1
1.1.1 Flying Radar Target	1
1.1.2 Wide Area Surveillance Projectile	2
1.2 Previous Work	6
1.3 Present Work	7
2 Experimental Testing of a Five-Hinged Wing	9
2.1 Wing Characterization	9
2.2 Wind Tunnel Setup	11
2.3 Instrumentation	11
2.4 Test Procedure	12
2.4.1 Steady State Performance	14
2.4.2 Response to Disturbance	14
2.4.3 Deployment	14
2.5 Data Postprocessing	15
2.6 Static Testing	16
2.7 Steady State Response	17
2.8 Dynamic Response	19

3	Theoretical Modeling	31
3.1	Multi-Body Dynamics	31
3.2	Elastic Model	38
3.3	Applied Forces	40
3.3.1	Gravitational Force	41
3.3.2	Hinge Torque	42
3.3.3	Aerodynamic Forces	45
3.4	Time Integration Procedure	48
3.5	Solution Procedure	50
4	Model Validation	53
4.1	Model Setup	53
4.2	Validation Results	54
5	Numerical Studies	59
5.1	Linear Model Studies	59
5.1.1	Five-Hinged Wing	59
5.1.2	One-Hinged Wing	62
5.2	Nonlinear Studies	66
5.2.1	Five-Hinged Wing	66
5.2.2	One-Hinged Wing	71
5.2.3	Two-Hinged Wing	74
6	Experimental Testing of Unstable Flight	83
6.1	Hinged Wing Design	83
6.1.1	Wing Layouts	83
6.1.2	Hinge Design	86
6.2	Wing Characterization	87
6.3	Wind Tunnel Setup	87
6.3.1	Instrumentation	87
6.3.2	Test Procedure	88

6.3.3	Data Postprocessing	89
6.4	Wing Bench Test Results	89
6.5	Wind Tunnel Experimental Results	90
6.6	Model Correlation	95
6.6.1	Linear Behavior	95
6.6.2	Nonlinear Behavior	97
7	Concluding Remarks	103
7.1	Overview	103
7.2	Key Contributions	105
7.3	Design Considerations	106
7.4	Future Work	107
7.4.1	Experimental Improvements	107
7.4.2	Model Improvements	107
	Bibliography	109
A	WASP Wing Physical Characteristics	113
A.1	Wing Properties	113
A.2	Airfoil Coordinates	115
B	Foam Wing Characteristics	117
B.1	Wing Planform	117
B.2	Cross-Sectional Properties	118
B.3	Accelerometer Properties	118
C	Experimental Test Matrices	119
C.1	WASP Wing	119
C.2	Foam Wing	120
D	Double Pendulum Solution	123

E	Matlab Scripts	127
E.1	Main.m	128
E.2	Aesfun2.m	132
E.3	Aeinitial.m	135
E.4	Aemrbar.m	139
E.5	Ode45outs.m	141
E.6	Aedivy2.m	147
E.7	Flex.m	154

List of Figures

1-1	FLYRT deployment	2
1-2	Exploded view of WASP and shell	4
1-3	WASP mission	5
1-4	WASP airfoil	5
1-5	WASP wing prototype	6
1-6	Various nonlinear stiffnesses	7
2-1	Schematic of static load testing using Questar microscope	10
2-2	Top view of hinge spring measurement setup	11
2-3	Wing mounted in Wright Brothers Wind Tunnel	12
2-4	High speed camera position for wind tunnel testing	13
2-5	High speed camera image showing targets	13
2-6	Wing folded prior to deployment testing	15
2-7	Deformation of WASP wing due to static tip loads	17
2-8	WASP hinge spring measurements with stiffness curve and stability boundaries	18
2-9	WASP wing lift curve	19
2-10	Video sequence of WASP wing response at 46 m/s and 0° root AOA with 0.006 seconds between images	22
2-11	Bending response of WASP wing at various speeds and 0° root AOA	23
2-12	Bending response of WASP wing at various root AOAs and 46 m/s	24
2-13	Video sequence of WASP wing at 46 m/s and -5° root AOA with 0.02 seconds between images	25

2-14	Bending response of WASP wing at 46 m/s and -5° root AOA (corresponding to case shown in Figure 2-13)	26
2-15	Video sequence of largest amplitude refold motion captured at 46 m/s and -5° root AOA with 0.04 seconds between images	27
2-16	Bending response of WASP wing at 46 m/s and 0° root AOA with different segments deflected (refer to Figure 1-5)	28
2-17	Bending response of WASP wing at 66 m/s and 0° root AOA with different segments deflected (refer to Figure 1-5)	29
2-18	Typical experimental AOA measurements (taken at 46 m/s and 4° root AOA)	29
2-19	Video sequence of wing deployment at 66 m/s and 0° root AOA with 0.02 seconds between images	30
3-1	Diagram of double pendulum	32
3-2	Coordinate systems used in multi-body simulation	33
3-3	Six-degree-of-freedom finite beam element	38
3-4	Mesh of wing cross section using quadrilateral elements for use with VABS	38
3-5	Possible elastic boundary conditions	40
3-6	Loads applied to a flexible segment of a multi-hinge beam structure	41
3-7	Photo of tip-most WASP wing hinge showing typical mechanical connections	42
3-8	Bi-linear nature of hinge stiffness	43
3-9	Possible hinge states	44
4-1	CPU times for one, two and four elements per wing segment	55
4-2	Comparison of model with four, two and one element per segment	56
4-3	Wing tip response at various speeds when the root angle of attack is set to 0°	57
4-4	Wing tip response at various angles of attack when flying at 46 m/s	58
5-1	Eigenvalue diagram of WASP wing ($U_f/\omega_\alpha b = 5.65$, $k = 0.082$)	60
5-2	Eigenvalue diagram of an unhinged wing with WASP properties ($U_f/\omega_\alpha b = 5.12$, $k = 0.112$)	61
5-3	Natural frequencies of a one-hinged wing	63

5-4	Normalized mode shape components of one-hinged wing, solid lines for plunge component, and dashed for pitch angle component.	64
5-5	Characteristics of a one-hinged wing with no second bending flutter instability, hinge stiffness of 50 Nm ($U_f/\omega_\alpha b = 5.09$, $k = 0.103$).	65
5-6	Eigenvalue diagram of a one-hinged wing showing second bending and first torsion instability, hinge stiffness of 25 Nm (1^{st} bending $U_f/\omega_\alpha b = 5.25$, $k = 0.094$, 2^{nd} bending $U_f/\omega_\alpha b = 4.19$, $k = 0.176$).	66
5-7	Eigenvalue diagram of a one-hinged wing with no post second bending stability, hinge stiffness of 10 Nm (1^{st} bending $U_f/\omega_\alpha b = 5.67$, $k = 0.073$, 2^{nd} bending $U_f/\omega_\alpha b = 4.51$, $k = 0.142$).	67
5-8	Map of the one-hinged wing instabilities	68
5-9	Twist response of WASP wing flying at 210 m/s (flutter speed of 214 m/s) with 660 N of static lift	69
5-10	Bending response of WASP wing flying at 216 m/s (flutter speed of 214 m/s) with 26 N of static lift	69
5-11	Zoom into one of the cycles shown in Figure 5-10	70
5-12	Bending phase projection of WASP wing flying at 216 m/s (flutter speed of 214 m/s) with 26 N of static lift	70
5-13	Twist response of WASP wing flying at 216 m/s (flutter speed of 214 m/s) with 26 N of static lift	71
5-14	Diverging bending response of WASP wing flying at 218 m/s (flutter speed of 214 m/s) with 165 N of static lift	72
5-15	Diverging twist response of WASP wing flying at 218 m/s (flutter speed of 214 m/s) with 165 N of static lift	73
5-16	Chaotic bending response of WASP wing flying at 218 m/s (flutter speed of 214 m/s) with 6.5 N of static lift	74
5-17	Characteristics of a one-hinged wing showing second bending and first torsion instability, ‘-’ stiff state of 22.5 Nm, ‘- -’ soft state of 0.0001 Nm	75
5-18	Limit cycle oscillation of one-hinged wing flying at 166 m/s (flutter at 165 m/s) with 53 N static lift	76

5-19	Unstable behavior of one-hinged wing flying at 166 m/s (flutter at 165 m/s) with 309 N static lift	76
5-20	Divergent behavior of one-hinged wing flying at 175 m/s (flutter at 165) with 60 N static lift	77
5-21	Limit cycle oscillation of one-hinged wing flying at 150 m/s (flutter at 165) with 44 N static lift	77
5-22	Limit cycle oscillation of one-hinged wing flying at 135 m/s (flutter at 165 m/s) with 37 N static lift	78
5-23	Bending response one-hinged wing flying at 160 m/s (flutter at 165 m/s) with 60 N static lift and a -4.7° initial hinge angle	78
5-24	Bending response of the one-hinged wing flying at 160 m/s (flutter at 165 m/s) with 60 N static lift and a -27.5° initial hinge angle	79
5-25	Characteristics of a two-hinged wing showing second bending and first torsion instability hinge stiffness of 600 Nm and 26 Nm (1 st bending $U_f/\omega_\alpha b = 5.32$, $k = 0.097$, 2 nd bending $U_f/\omega_\alpha b = 5.32$, $k = 0.161$).	80
5-26	Position and phase of root-most hinge response of two-hinged wing at 140 m/s and 0° AOA ($k_1 = 600$ Nm and $k_2 = 26$ Nm)	80
5-27	Position and phase of tip-most hinge response of two-hinged wing at 140 m/s and 0° AOA ($k_1 = 600$ Nm and $k_2 = 26$ Nm)	81
5-28	Steady-state amplitude of torsional response of two-hinged wing with initial conditions in same state and different state than equilibrium	81
6-1	Unhinged and one-hinged 1-meter span wings (with 30-cm ruler shown for scale)	84
6-2	Diagram of foam wing cross-section	84
6-3	Eigenvalue diagram of one-hinged wing as designed, ‘-’ stiff hinge ($U_f/\omega_\alpha b = 1.86$, $k = 0.436$) ‘- -’ soft hinge ($U_f/\omega_\alpha b = 3.77$, $k = 0.157$).	85
6-4	Photos of foam wing hinge	86
6-5	Placement of accelerometers in foam wing tip	88
6-6	Wing hinge with copper contacts	89
6-7	PSD approximations of plunge data (‘-’ 600 samples, ‘- -’ 750 samples) . . .	91

6-8	First two natural frequencies over range of dynamic pressures for the foam wings ('o' unhinged, '*' hinged)	92
6-9	Mid-chord accelerometer data (solid line) during a hinge-based limit cycle oscillation of hinged wing at 36.5 m/s and -3° root AOA. Hinge state data (dotted line) values of less than 50 indicate a partially closed hinge	93
6-10	Mid-chord accelerometer data (solid line) during a stall based limit cycle oscillation of hinged wing at 38.6 m/s and 0° root AOA. Hinge state data (dotted line) values less than 75 indicate a soft hinge state	94
6-11	Mid-chord accelerometer data (solid line) during a stall-based limit cycle oscillation of unhinged wing at 38.8 m/s and 0° root AOA	95
6-12	Steps followed for tip plunge determination, referring to steps outlined in Section 6.5 for one-hinged wing at 38.6 m/s and 0° AOA	98
6-13	Comparison of first two natural frequencies of the unhinged wing over range of dynamic pressures for model (-) and experimental data (*)	99
6-14	Comparison of first two natural frequencies of the hinged wing over range of dynamic pressures for model (-) and experimental data (*)	99
6-15	Eigenvalue diagram of one-hinged wing as designed, '-' stiff hinge ($U_f/\omega_\alpha b = 3.00$, $k = 0.188$), '- -' soft hinge ($U_f/\omega_\alpha b = 2.56$, $k = 0.264$).	100
6-16	Simulation results for a one-hinged wing flying at 36.5 m/s and a -3° root AOA subjected to a 0.3-second cosine gust starting at 0.2 seconds	101
A-1	WASP prototype wing	113
E-1	Script logic diagram	127

List of Tables

2.1	Angle of attack performance characteristics of the WASP wing	19
6.1	Natural frequencies of stiff state of one-hinged wing as originally designed . .	85
6.2	Foam wing impulse response (in Hz)	90
6.3	Operating conditions where a hinge-based limit cycle was recorded	92
6.4	Effect of model tuning factors on wing frequencies	96
6.5	Effect of model tuning on flutter speed (m/s)	96
A.1	WASP Wing planform dimensions	113
A.2	Cross-sectional properties of WASP wing (referenced to elastic axis)	114
A.3	Cross-sectional properties of one-hinge wing	114
A.4	Experimentally obtained WASP wing hinge properties	114
B.1	Foam wing planform	117
B.2	System parameters of experimental wings	118
B.3	Properties of the Endevco 22 piezoelectric accelerometer	118
C.1	Steady state test matrix	119
C.2	Dynamic test matrix, variable speed, 0° AOA	120
C.3	Dynamic test matrix, variable AOA, speed of 45 m/s	120
C.4	Test points of unhinged foam wing	121
C.5	Test points of one-hinged foam wing	122

Chapter 1

Introduction

1.1 Motivation

The idea of using a hinged wing to fulfill space constraints has been around for a number of years. However, most hinged wings have only one joint and are locked in place during operation. There are a number of roles in which it might be useful to have a wing with multiple hinges, or hinges that deploy in flight, or hinges that do not lock. A hinged wing might be used to ‘morph’ an air vehicle - allowing it to change its wing-spans in mid flight. Another role is remotely deployable vehicles that have extreme packaging constraints, such as the suggested Mars Plane [1] or the Wide Area Surveillance Projectile (WASP) further described below.

1.1.1 Flying Radar Target

A vehicle with a hinged wing that deploys mid-flight has been demonstrated. The Naval Research Laboratory used a folding wing design for the Flying Radar Target (FLYRT) [2, 3]. The FLYRT vehicle was designed to be stored on a naval ship, and to be compatible with the shipboard Mk 36 decoy launching system. FLYRT was to be launched with a rocket motor attached and carry advanced electronic warfare payloads.

The wings of the FLYRT vehicle had one hinge and were latched in place when deployed. The main wing section was aligned with the body and pivoted out for deployment. The

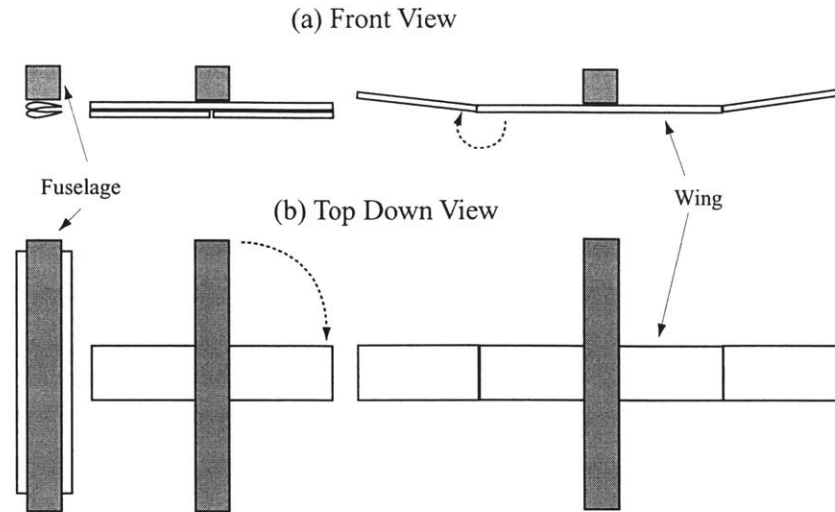


Figure 1-1: FLYRT deployment

outer wing panels were released and would ‘fly’ into place, shown in Figure 1-1. While several methods involving cables, springs and rods were considered, these were deemed awkward and unreliable. It is not known how much, if any, analysis of the aeroelastic effect of the hinges was performed before or after the testing since it was not presented in open literature. However, drop tests of a FLYRT model did prove that gravity and aerodynamic lift alone could deploy those hinged wings. The FLYRT model did have difficulties with nonsynchronous wing deployment causing unrecoverable flight instabilities.

1.1.2 Wide Area Surveillance Projectile

A multi-hinged wing was an integral part in the design of the Wide Area Surveillance Projectile (WASP) project. The following is a basic overview of the WASP project. A more detailed description of the project can be found in [4, 5].

MIT’s Department of Aeronautics and Astronautics and Draper Laboratory began a program in the summer of 1996 utilizing the resources of both to develop a new aerospace system. The objectives of the program were to design a ‘first-of-a-kind’ system and produce and test a prototype within a two-year time-line. This system needed to provide a solution to a national need or problem. It also had to have an element of ‘unobtainium’ or high-risk technology.

The first few months of the program were used to create basic system concepts. These concepts were judged on potential marketability, technological innovation, and feasibility of meeting the two-year deadline. The winning concept was WASP, a concept to ballistically deploy an autonomous sensorcraft to a target area.

WASP was sized to be as small as feasible. While it was thought possible to design a system that could be deployed from a soldier's rifle, this was deemed too difficult to achieve. Thus, WASP was sized to be fired from artillery or naval cannons allowing ship captains and artillery commanders near instantaneous reconnaissance of twenty-mile-away target areas. This would fill a niche between that of long duration type Unmanned Aerial Vehicles (UAVs), like the Predator and Global Hawk, and smaller, troop deployable UAVs that were then under development.

In order to successfully fill its market niche, a set of system level requirements was created to guide the design process. These requirements were determined after consulting with officials from the US Army and Navy.

- Compatible with 5-inch Navy gun
- Survive a 15,000 g acceleration
- Loiter for 15 minutes
- Be autonomous and carry a camera
- Inexpensive and storable for at least three years
- Interact with a ground station and send real-time images and GPS coordinates of targets

To meet the design requirements, a small UAV was chosen to be the deployable portion of the WASP system. This small flyer would be packaged inside of a modified 5-inch naval shell which normally carried a flare, Figure 1-2. This design would help protect the flight vehicle from the launch environment. It also provided a massive projectile (desirable for long range) and a light flyer (desirable for long endurance.)

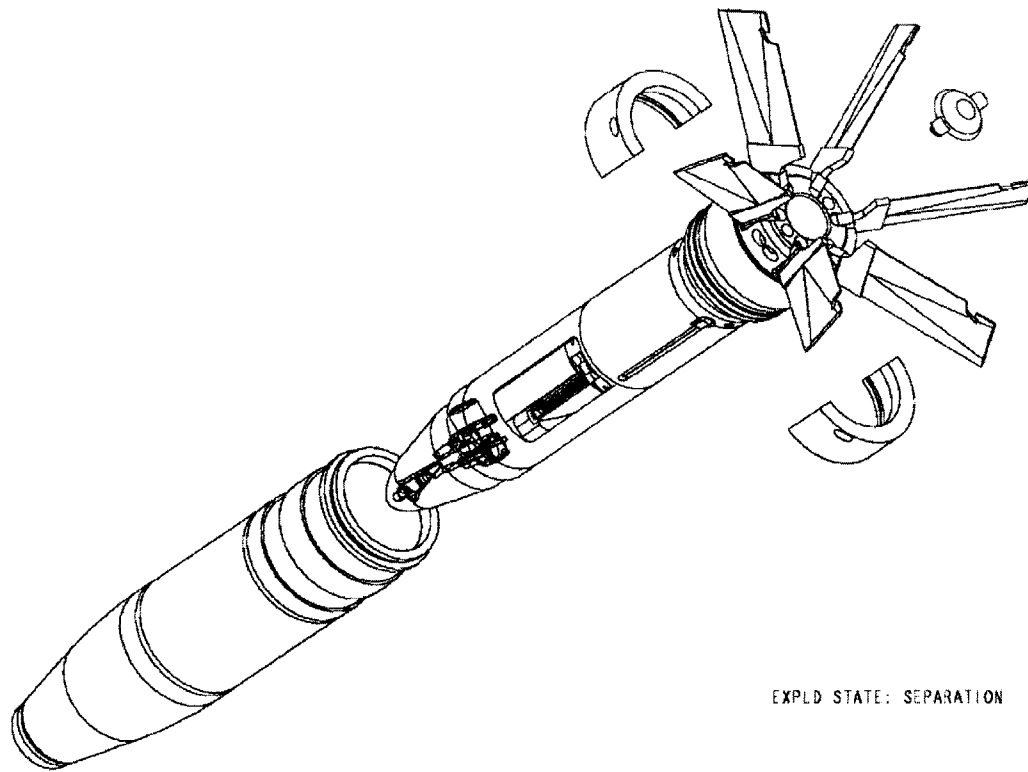


Figure 1-2: Exploded view of WASP and shell

The shell, after leaving the gun, would travel ballistically to the target area. When near the target area, the back end of the shell would separate. A parachute would deploy which would pull the flight vehicle out of the shell and decelerate it to subsonic speeds. The vehicle's engine would start, and the wings and tail would be deployed. Such a scenario is demonstrated in Figure 1-3.

The three major constraints that affected the wing system were the launch loads, size, and loiter time. To achieve a long loiter time, the wings needed to be long and light. A simple unhinged wing large enough to supply the needed lift to the system would not have fit within the allowed space. There was a direct trade off between the space taken up for the wings and the space left for the remaining systems, particularly engine fuel.

Several different concepts were studied for the wings. Telescopic wings were rejected due to the possibility that the sections would jam together under the launch loads. Inflatable wings were deemed too complex to design within the project time-line. Folding wings were chosen due to their simplicity and robustness. Proof of the validity of the concept of a folding wing deployed in flight used on a small UAV came from the Navy's Flying Radar Target

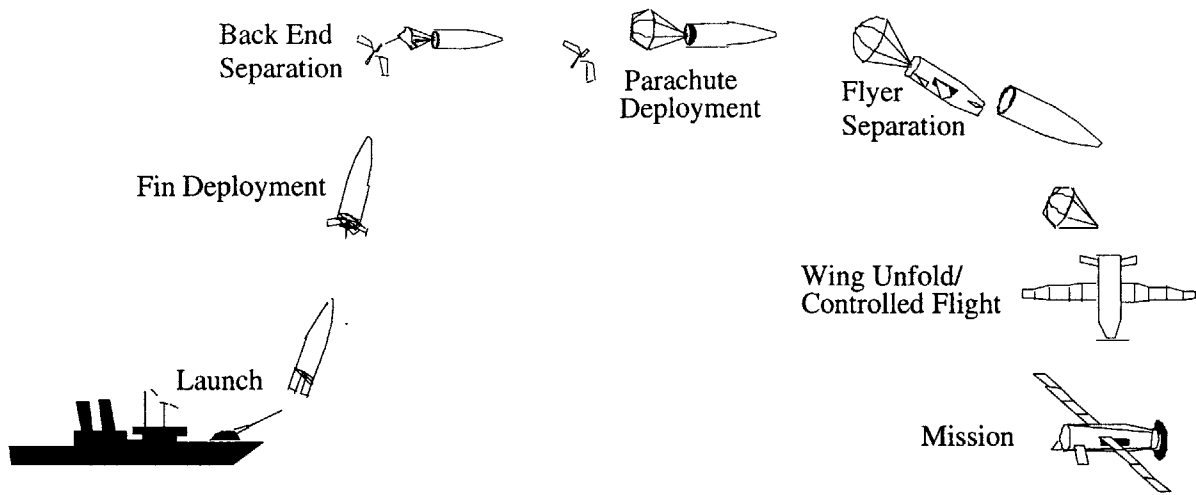


Figure 1-3: WASP mission [5]

discussed in section 1.1.1.

The wing has a constant airfoil shown in Figure 1-4, and the coordinates can be found in Appendix A.2. The thickness of the trailing edge allows the wing to survive the high-g launch loads. The wing is cambered to allow the segments to fold inside of each other. The camber also gives the wing positive lift at low angles of attack which assists in wing deployment and stability.

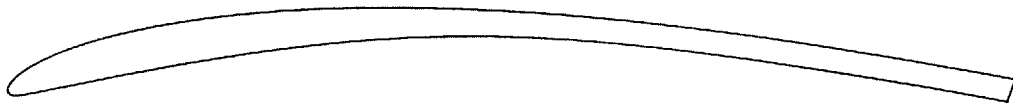


Figure 1-4: WASP airfoil

The wing is divided into six sections separated by five hinges as shown in Figure 1-5. The wing planform can be found in Appendix A.1. The lengths of the sections and the hinge geometries were sized based to meet the launch requirements. Each wing segment would rest on its hinge, which would, in turn, rest on the hinges underneath. Thus each wing section would only have to carry its own weight. Wing machined from aluminum alloy 7075 were tested under launch accelerations [5].

Even though extensive work was performed on the design, characterization and testing to design the WASP high-g hinged wing, no studies were conducted to assess the aeroelastic behavior of the wing.

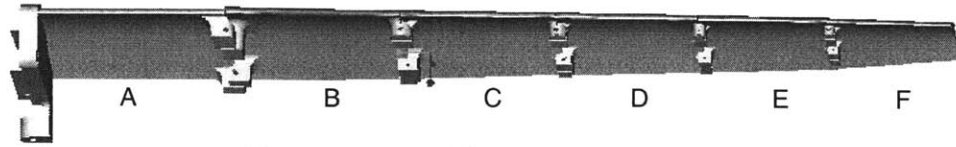


Figure 1-5: WASP wing prototype

1.2 Previous Work

To study the aeroelastic response of a multi-hinged wing, two fundamental problems need to be addressed. The first is multi-body dynamics, and the second is piecewise nonlinear aeroelastics.

Multi-body dynamics models the mechanics of interconnected rigid and elastic bodies. Most multi-body systems are quite complex and non-linear. Several methods are used to solve these systems. One method uses nonlinear finite elements which use direct strain measurements [6]. This technique was specifically tailored toward a system composed of beams connected by hinges. More popular is the use of Hamilton's Principle combined with Lagrangian multipliers to satisfy system constraints [7-9]. This method allows for quick, methodical construction of complex mechanical systems. The dynamics of each of the bodies is determined separately and a series of constraint equations is used to determine the influence of the bodies on each other.

The piecewise nonlinear nature of a multi-hinged wing is similar to that of a control surface with freeplay in that the elastic properties can change dramatically with position. A hinge is either open or closed, and a surface with freeplay is either free or not. The aeroelastic performance of nonlinear structures is an extensive area of research with an abundance of analytical and experimental studies [10-20]. These studies have looked into structures with bilinear stiffness (stiffening and softening), freeplay, and nonlinear stiffness, all of which are illustrated in Figure 1-6. Most of these studies have looked into systems of two or three degrees of freedom with either nonlinear torsional stiffness or flap freeplay as these are the most common occurrences in a typical wing. McIntosh et al. [21] and Hauenstein et al. [22, 23] performed analytical and experimental studies of a system with bilinear and freeplay plunge stiffness respectively. These studies most closely model the structure of a hinged

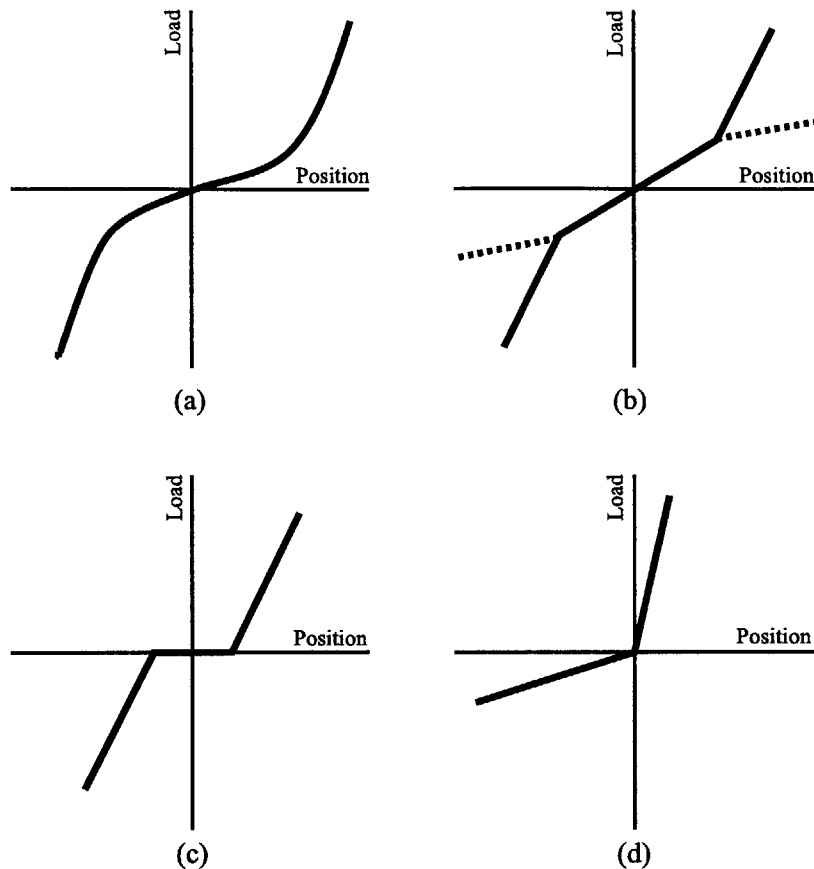


Figure 1-6: Nonlinear stiffnesses (a) cubic, (b) bilinear (‘—’ stiffening, ‘.....’ softening), (c) freeplay, (d) bilinear hinge

wing. A summary of work in nonlinear aeroelasticity was compiled by Lee et al. [24].

A linear aeroelastic system will either have static subsidence or divergence, or a damped or divergent oscillation. However, the previously mentioned studies have shown that a nonlinear aeroelastic system might also go into a limit cycle oscillation and can exhibit chaotic or nonperiodic behavior. While this type of behavior might be expected of a system with a symmetric nonlinear stiffness, the effect of asymmetric nonlinearity has not been studied. In addition, the effect of multiple bilinear pitch degrees of freedom on a three-dimensional body is unknown.

1.3 Present Work

At the end of the WASP project, the structural performance of the wings subjected to launch loads had been well tested. However, the aerodynamic and aeroelastic performance

were unproven. While the FLYRT had demonstrated unassisted deployment, it had only one hinge. The WASP wing has five hinges and they do not have latches. It was unknown whether these hinges would have a large impact on the aeroelastic performance of the wing. As the wing was to deploy at relatively high speeds, it became important to determine the flutter speed of the wing. It was also unknown how the wing would respond to gust loads or rapid maneuvers.

The objective of this thesis is to study the effects of having multiple unlatched hinges on the aeroelastic performance of a wing. To gain a basic understanding of the flight characteristics of hinged wings, the WASP prototype wing was tested in a wind tunnel under various flight conditions. These experiments and their results are described in Chapter 2. The knowledge gained from these flight experiments helped in the development of an analytical model designed to predict the behavior of a multi-hinged wing. The development of the model is detailed in Chapter 3. The model is solved in the time domain to capture any nonperiodic motion. Chapter 4 evaluates the ability of the model to capture the fundamental aspects of the WASP wing in flight. After showing that the analytical model can predict the response of the WASP wing to benign flight conditions, the model was used to explore the full flight envelope of hinged wings. The numerical model predicted that hinged wings exhibit behavior similar to other nonlinear aeroelastic systems such as limit cycle oscillations and chaotic response. The results of these studies are detailed in Chapter 5. To verify accuracy of the model in predicting the nonlinear response of a hinged wing, a new round of experimental tests were performed on custom built wings. Using the theoretical model developed herein, wings were designed to exhibit characteristics unique to a hinged wing. Their design, experimental setup, and results are presented in Chapter 6. A summary and discussion of all the experimental and analytical results is presented in Chapter 7. This chapter theorizes the fundamental differences between a normal and a multi-hinged wing. It also suggests future work that could gain a better understanding of a multi-hinged wing, and tests that could be done to further verify the proposed theories.

Chapter 2

Experimental Testing of a Five-Hinged Wing

A prototype wing from the WASP vehicle was used to gain initial insight into the performance of a more complex multi-hinged wing. Static and dynamic flight tests were done to determine the low speed flight characteristics of that wing. This data was used to assist in the development of the analytical model described in Chapter 3.

2.1 Wing Characterization

Static loading tests were performed on the prototype WASP wing to determine its physical hinge properties.

To determine the hinge properties for an open hinge, the wing was clamped in an inverted orientation. Loads were applied by hanging weights at each of the hinge locations. The vertical displacement of several locations was determined using a Questar microscope. Through the microscope, several targets on the wing were monitored during the loading to determine their vertical displacement (see Figure 2-1).

Most of the springs for the WASP prototype were custom made. To determine the stiffness of the springs, the wing was placed on its side and a load was applied to each hinge and to the wing tip one at a time, as schematically shown in Figure 2-2. The load was applied

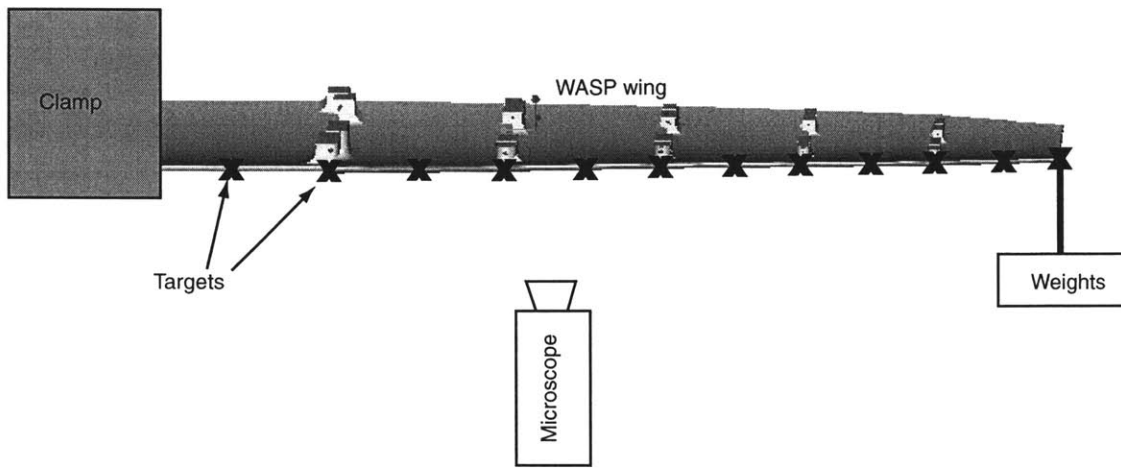


Figure 2-1: Schematic of static load testing using Questar microscope

by using a weight attached to a string that was looped over a pulley. The angles of the string and the segments were measured such that the torque applied could be determined. Only one segment was allowed to deflect at a time by softly clamping the previous segment. A hard clamp resulted in enough wing deformation to impinge the hinge.

Each hinge had a significant amount of static friction, thus two measurements were made for each applied torque. One measurement was taken at the largest angle the spring could maintain for the given torque and the other at the smallest angle. The difference in the angles was the contribution of the static friction. The mean of the two angles is the deflection that would have occurred if there was no static friction.

No experiment was performed to determine the damping or restitution of the hinges. This was partially due to the inability to clamp the individual segments without causing a change in the hinge properties. Other difficulties included the alignments necessary to properly obtain data, and the lack of a quality data acquisition system.

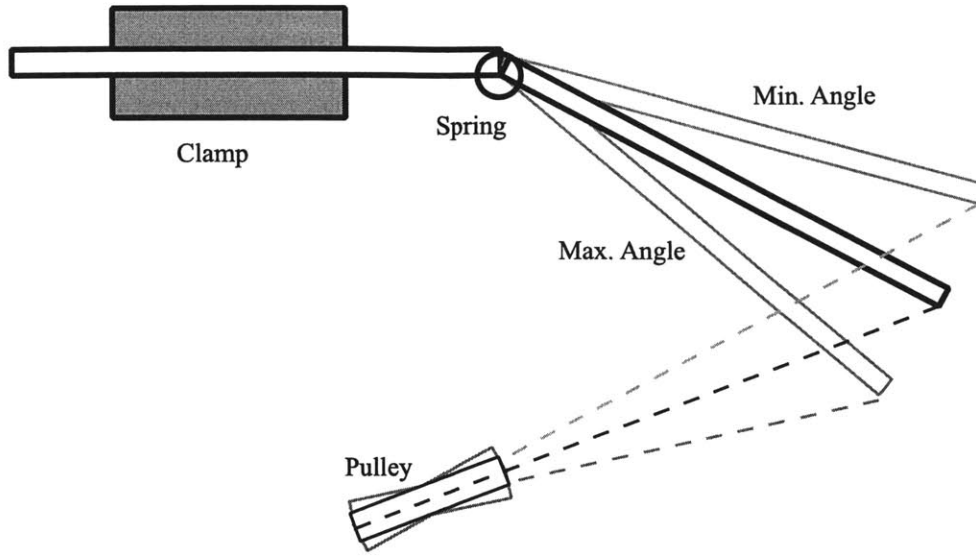


Figure 2-2: Top view of hinge spring measurement setup

2.2 Wind Tunnel Setup

The wing was tested in the Wright Brothers Wind Tunnel at MIT. The low-speed pressurized tunnel has a 3-by-2.3 meter elliptical cross-section and is capable of steady flows up to 90 m/s.

The wing was flown in a horizontal configuration, as shown in Figure 2-3, to orient the gravitational forces correctly. The wing was mounted to a one meter high rigid pedestal. The mount allowed the wing to pivot about its pitch axes, while fixing the yaw and roll axes. The angle of attack (AOA) of the wing was controlled using a pitch arm attached to a screw drive.

2.3 Instrumentation

Static and stagnation air pressure and air temperature were measured using the basic instrumentation of the wind tunnel. The lift forces on the wing were measured using the tunnel's six-axis force balance.

So as not to disturb the airflow around the WASP wing, an Ektapro high-speed video

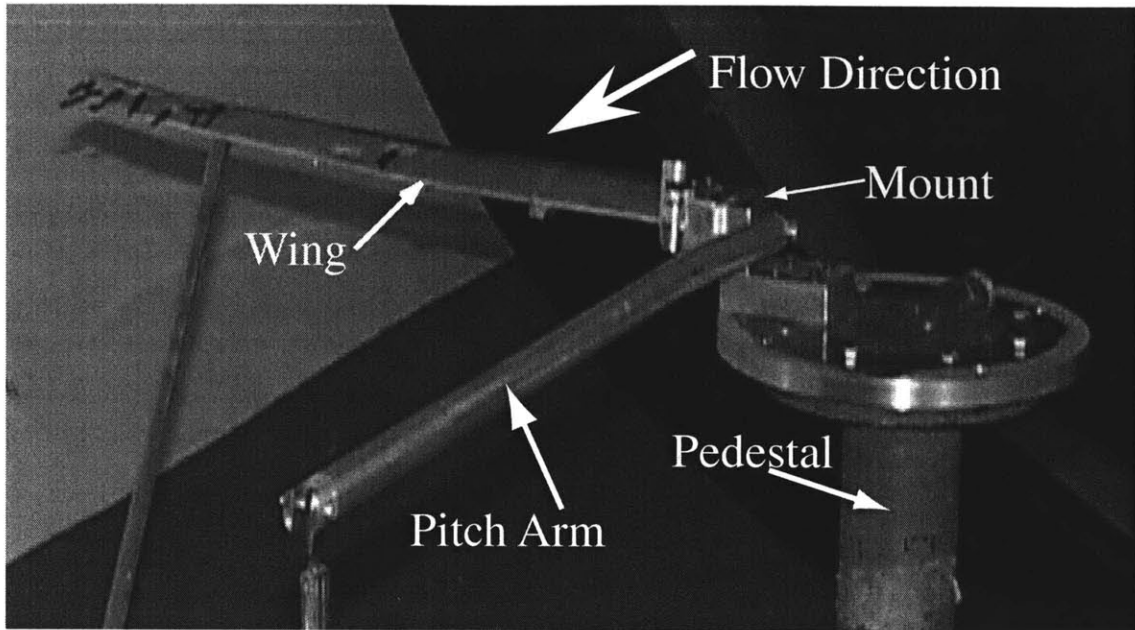


Figure 2-3: Wing mounted in Wright Brothers Wind Tunnel

system was used to record the position of the wing rather than in-situ sensors. The video system recorded the behavior of the wing at 500 frames per second. The video system's camera was placed outside of the wind tunnel, aligned with and just below the wing's span axis as shown in Figure 2-4. Targets were painted on the wing tip at the leading and trailing edges, and on each of the hinges (Figure 2-5). The data from the camera was stored on VHS videotape and then transferred to digital QuickTime 4.0 format using Adobe Premier 5.0.

2.4 Test Procedure

Prior to testing, the tunnel's force balance was calibrated using a spring scale. The wind tunnel was not stopped between individual tests. The testing concentrated on three areas:

1. Steady state performance
2. Response to disturbance (gust)
3. Deployment characteristics

The complete test matrix can be found in Appendix C.1.

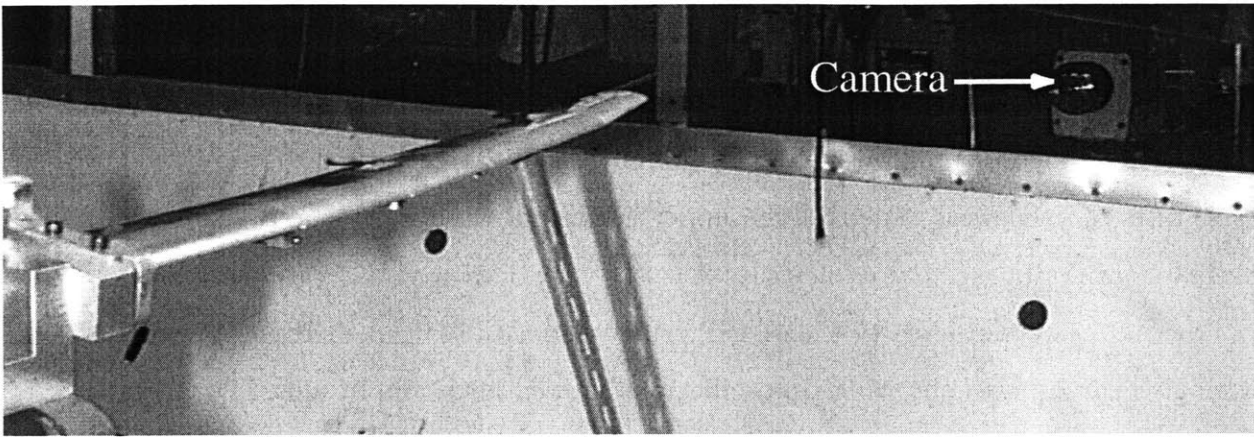


Figure 2-4: High speed camera position for wind tunnel testing

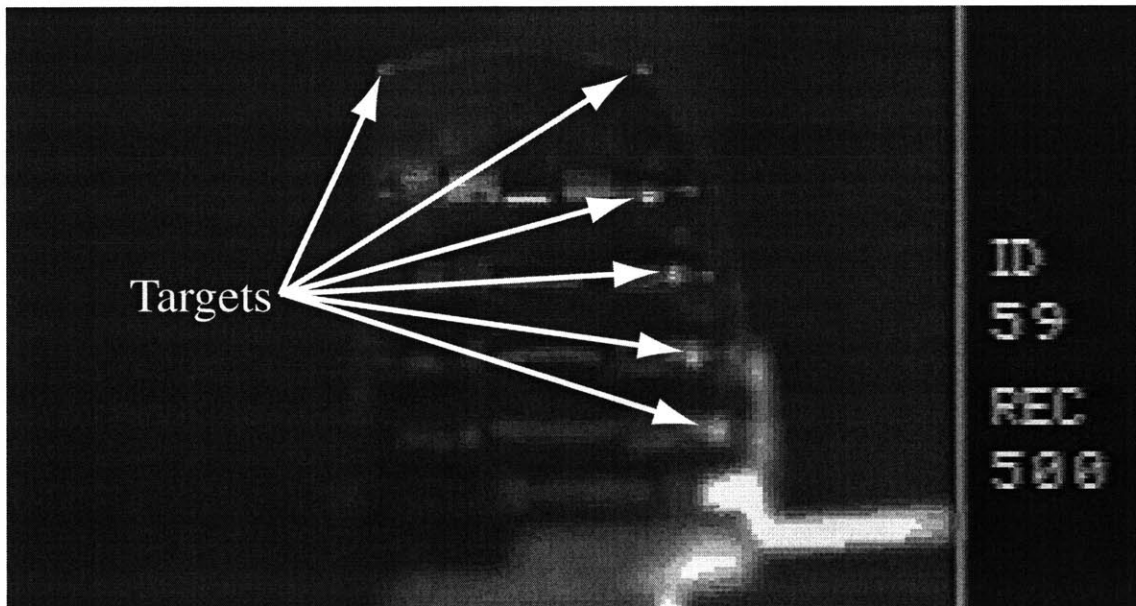


Figure 2-5: High speed camera image showing targets

2.4.1 Steady State Performance

The first set of tests determined the steady state performance of the wing. The test matrix was set up to slowly expand the flight envelope of the wing.

For the WASP wing, high AOA studies were then conducted at the lowest testing speed. The AOA was incremented by single degrees until the lift force started to decrease. For these tests, tufts of yarn were taped to the upper surface of the wing to help visualize the flow reversal near stall.

Low AOAs were studied at the WASP cruising speed (40 m/s). The AOA was decreased by single degrees until the wing tip oscillations showed large amplitudes.

2.4.2 Response to Disturbance

To explore the unsteady performance of the wing, a series of disturbance tests were performed. This was accomplished by placing a rod into the flow and using it to deflect the wing. This method allowed control over which of the hinges were bent by which segment of the wing the rod pushed on. It was decided that the rod should strike the wing swiftly to minimize its impact on the airflow around the wing. This method reduced the amount of control over the initial displacement of the wing, but still regulated which hinge was deflected.

The wing was flown at a constant AOA over a range of speeds. At all of these speeds, the tip-most hinges were disturbed using the rod technique. The disturbance tests were also done at cruising speed of the WASP wing over a range of AOAs. The decay rate of the oscillation of the wing was monitored.

2.4.3 Deployment

Full WASP wing deployment was tested at 70 m/s and zero AOA. This speed was set for safety reasons and is below the design deployment speed of the WASP wing. The wing was completely folded and held in place via an elastic band as shown in Figure 2-6. When the wing was being folded for this test, the inboard most hinge spring became disconnected and was not repaired before the test.



Figure 2-6: Wing folded prior to deployment testing

With the wing folded and held in place, the wind tunnel was brought up to speed. After the wind tunnel had been at test speed for over a minute, the rubber band was released and the wing unfolded.

2.5 Data Postprocessing

The pressure and force data were used to find the coefficient of lift at every test point as,

$$C_L = \frac{L}{(P_T - P_s)A} \quad (2.1)$$

where L is the lift force of the wing, A is the wing area (0.0277 m^2 , see Table A.1), and P_T and P_s are the stagnation and static pressures respectively.

Since there were large temperature differences between some of these data points (as much as 25°C), it is believed that this had an effect on the calibration of the force balance. This resulted in large variations in the measured coefficient of lift at various test points with near equal dynamic pressures and AOAs. The force balance had no known correction for temperature. A correction was approximated by finding a correlation between coefficients of

lift and temperature at given dynamic pressures and AOAs.

The QuickTime images made from the high-speed video data were analyzed on a Macintosh computer using the public domain NIH Image program [25]. NIH Image allowed the target image's pixel location to be found in each frame of video. Determining the number of pixels between the wing tip targets gave a scaling of 25 pixels per centimeter. When the wing tip went through rapid motion (on the order of hundreds of centimeters per second) the targets would blur. While some blurring could be handled by finding the center of the blurred image, the location of the targets on some frames could not be determined.

2.6 Static Testing

As described in Section 2.1, tip loads were applied to a clamped wing to determine its open hinge properties. Figure 2-7 shows the wing deflection due to static upward bending loads. The deflection was measured at each of the hinges and at points halfway between the hinges. The deformations are largely dominated by bending about the hinge points. The results from this set of experiments allowed the open hinge stiffness and the angle at which the hinge switches states to be determined.

To characterize non-open hinge properties, torque was applied to each of the WASP wing hinges. Two points were measured for each applied torque. The first is the largest hinge angle that could be maintained under the torque, and the second the smallest. From this information the hinge spring stiffness and static friction could be approximated. Figure 2-8 shows the results of this set of experiments. A best linear fit was found to both the higher and lower displacement data sets. The slopes and intercepts of the two linear fits were averaged together (weighted by the number of data points) to give the hinge stiffness curve. The difference in the intercept between the averaged line and the curve fits gives the static friction of the hinge. The hinge characteristics are summarized in Appendix A, Table A.1.

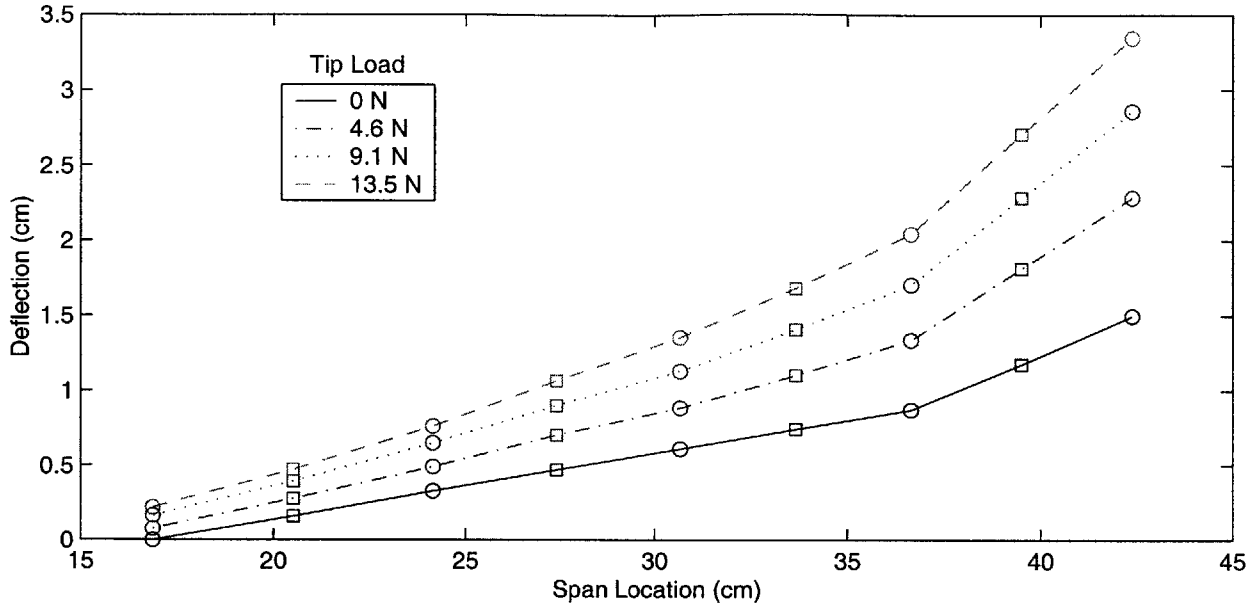


Figure 2-7: WASP wing deformation at hinges (o) and mid segment (□) due to static tip loads (open hinge configuration)

2.7 Steady State Response

During the dynamic phase of testing of the WASP wing, the steady state values of lift were obtained before the wing was perturbed. This data was combined with the data gathered during the steady state phase of testing to determine the flight characteristics of the wing.

Figure 2-9 summarizes the lift results corrected for temperature obtained during both phases of testing. The expected behavior line is an estimate of the experimental lift curve. The experimental data was used to determine the angle of attack required during cruise. This is the angle at which the wing would need to operate such that, at the cruise velocity, they could support the weight of the WASP vehicle. As shown in Table 2.1, the cruise angle of attack was close to the angle predicted [5]. Table 2.1 also shows the angle of attack at which stall was recorded. Stall was determined by a peak in the lift curve as well as by observing flow reversal on the wing.

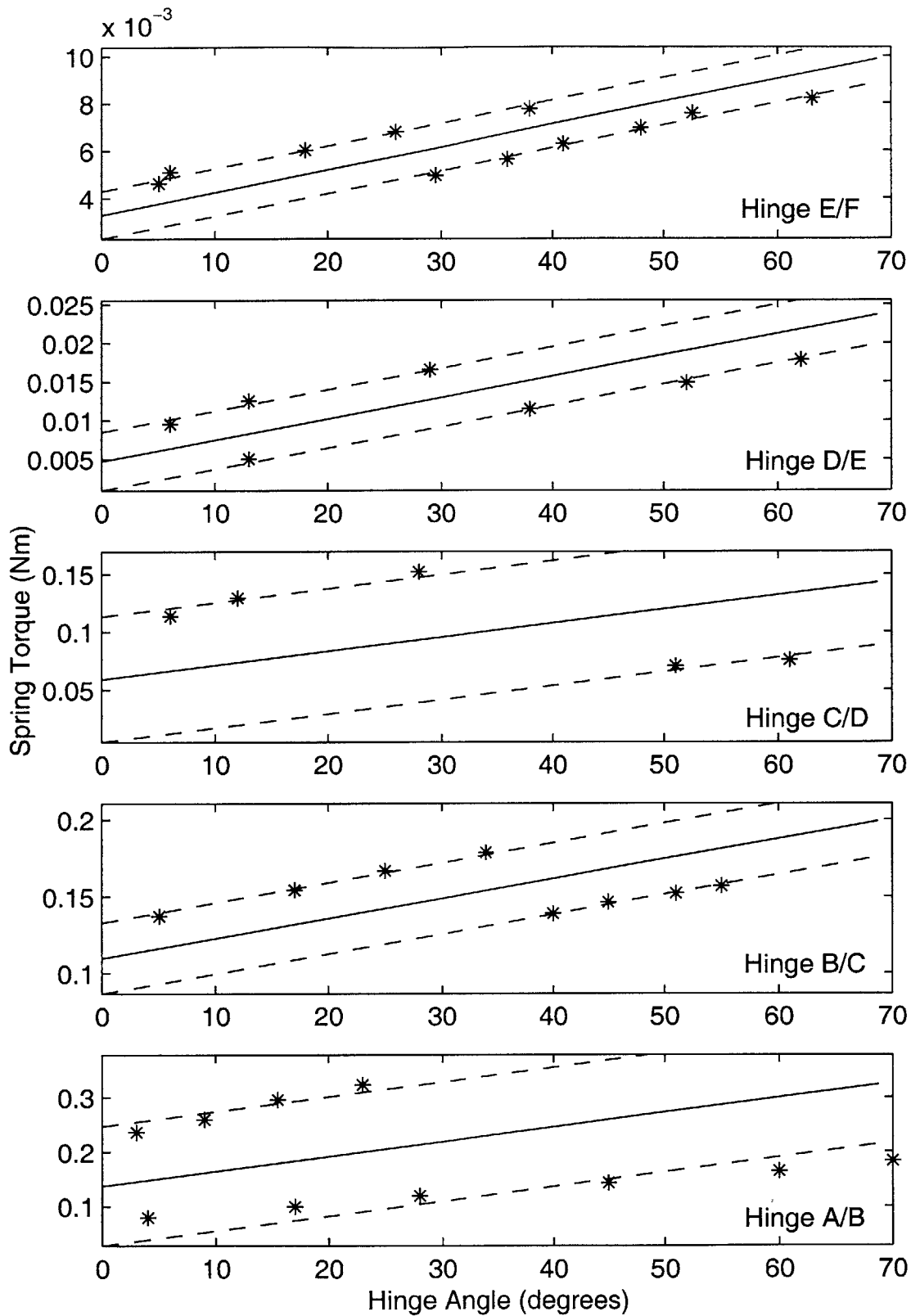


Figure 2-8: WASP hinge spring measurements (*) with stiffness curve (-) and stability boundaries (- -)

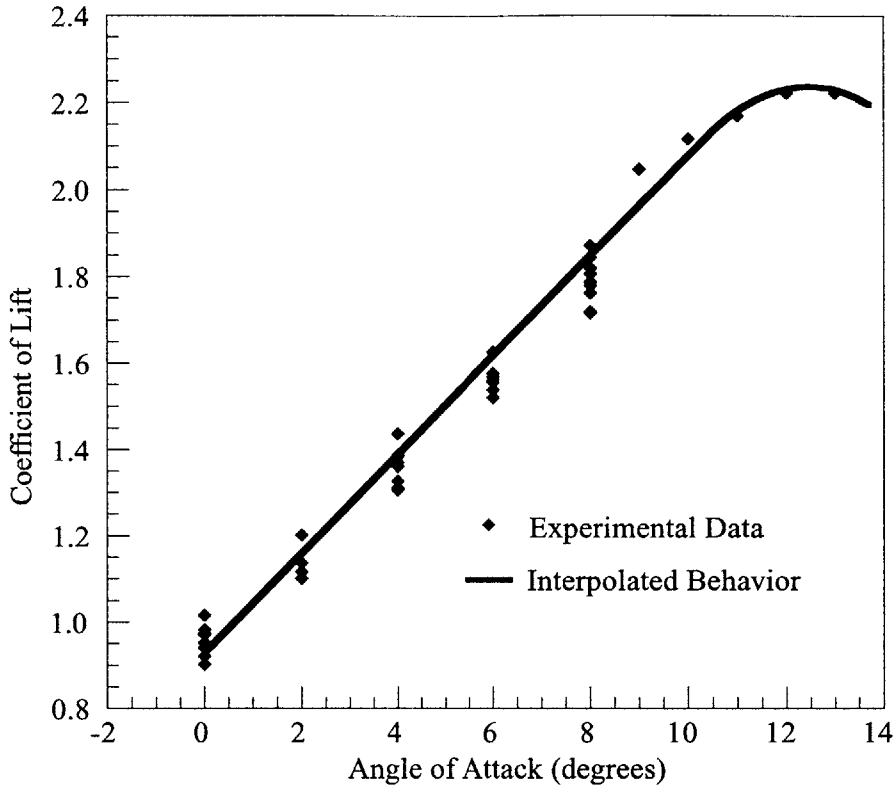


Figure 2-9: WASP wing lift curve

Table 2.1: Angle of attack performance characteristics of the WASP wing

Predicted Cruise AOA	5.5°
Experimental Cruise AOA	6°
Experimental Stall AOA	12°

2.8 Dynamic Response

To study the dynamic response of hinged wings, the wing was flown in a variety of flight conditions and the behavior of the wing was monitored. The wing segments had to be manually disturbed to provide a large enough response for the camera system to measure.

The dynamic testing phase of the WASP wing experiments examined the nonlinear response of hinged wings in flight by perturbing the wing in such a manner that at least one of the hinges would switch from the stiff to the partially closed state.

The high-speed video system was able to capture the motion of the tip of the wing as shown in Figure 2-10. However, some experiments caused the wing tip to move too fast.

This nominally occurred just after the tip was deflected. Moreover, at times (such as the deployment test or a large amplitude oscillation) the tip would move out of the field of view of the camera.

Using the NIH Image program results in accuracy of approximately one pixel. There are 96 pixels from the center of the leading edge target to the center of the trailing edge target, a distance of 3.81 cm. The vertical location of the wing tip was found by averaging the location of the wing tip targets. Therefore, the height of the wing tip was found with an accuracy of ± 0.02 cm, or $\pm 0.05\%$ of the semi-span.

While testing in the wind tunnel, the tip of the wing would oscillate at all flow speeds even when unperturbed. This oscillation was most likely due to a non-uniform wind tunnel flow. The broad band forcing of the non-uniform flow caused the wing to oscillate at its natural frequencies. It is possible that the frequency of the non-uniform flow could have been biased toward the tunnels fan frequency.

Figures 2-11 and 2-12 summarize the vertical locations of the wing tip during the dynamic testing where the wing tip is intentionally perturbed. The origin of the time axis is defined when the tip crosses the steady state value. The vertical tip locations are measured with respect to the steady state tip location for the flow condition. Each test had a different initial condition, thus the amplitude of the responses cannot be directly compared. The oscillations due to the disturbance were not much larger than the natural oscillations due to the tunnel noise for more than three or four cycles. These oscillations would occur at about 25-30 Hz. This gives less than 80 frames of data before the noise of the tunnel overcame the response. That is not enough of a sample to accurately determine the response frequency of the system. However, a few trends can be seen in the experimental data.

The frequency of the response does have a slight, but noticeable, increase as free stream speed is increased. Speed increase also causes a faster decay of the transient response. An increase in the root angles of attack tends to cause an increase in the response amplitude. However, this trend reverses for very low angles of attack, as can be seen in Figure 2-13 (only looking at every 10th frame). Here the wing enters the 'refold' condition. Figure 2-14 shows data generated by analyzing all the frames of data of the sequence shown in Figure 2-13. Figure 2-15 shows the largest oscillation observed at this flight condition. While the tip of

the wing goes outside of the video frame, it is estimated that the tip plunges as much as 30% of the semi-span.

Deflecting more than just the tip segment changed the wing response as can be seen in Figures 2-16 and 2-17. In Figure 2-16 the wing tip motion dips into a smooth curve implying that one of the wing hinges had changed states. Unfortunately, the targets which were used to measure the position of the wing along the span were placed on the under-side of the wing. Thus, the targets were blocked by the tip segment during the time just after the disturbance. By the time the targets were visible, most of their motion had been damped out.

The tip twist angle of the wing oscillated at too high of a frequency and too low of an amplitude to be accurately measured by the high-speed video system. A simple finite element model of a similar hingeless (WASP) wing found the first torsion mode would occur at about 210 Hz. The video system sampled at a rate of 500 frames per second giving less than three samples per expected cycle. The tip twist angle was determined by measuring the difference in height of the targets on the leading edge and trailing edge of the wing tip. This system has an accuracy of approximately $\pm 0.5^\circ$. Figure 2-18 is a plot of a typical tip twist angle as seen by the video system

The wing deployed as expected, even without the inner-most hinge spring. Figure 2-19 shows the video footage of the deployment. It appeared that the wing did straighten out in all but the inner-most hinge first. It then quickly pivoted around the inner-most hinge until it was in a normal flight orientation.

Unfortunately, no instrumentation was available at the time to provide a real time analysis of the lift force of the WASP wing. However, the high-speed video system could measure the time from release to fully open (0.23 seconds), and the additional time until the deployment induced motion became negligible (about 0.3 seconds).



Figure 2-10: Video sequence of WASP wing response at 46 m/s and 0° root AOA with 0.006 seconds between images

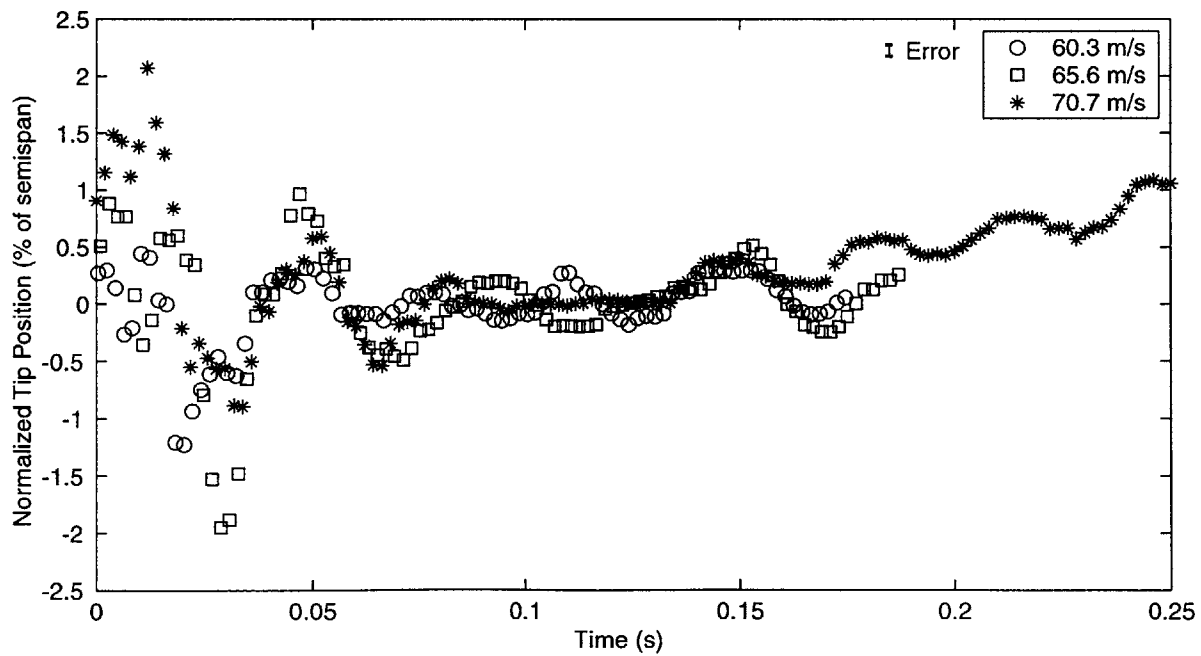
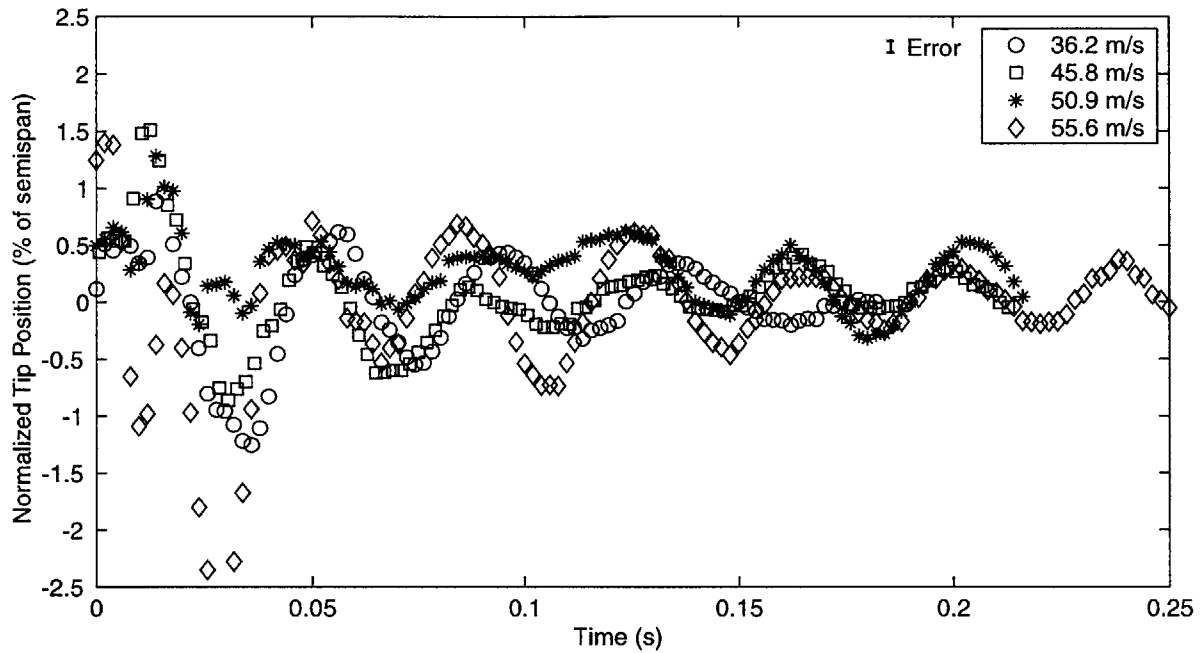


Figure 2-11: Bending response of WASP wing at various speeds and 0° root AOA

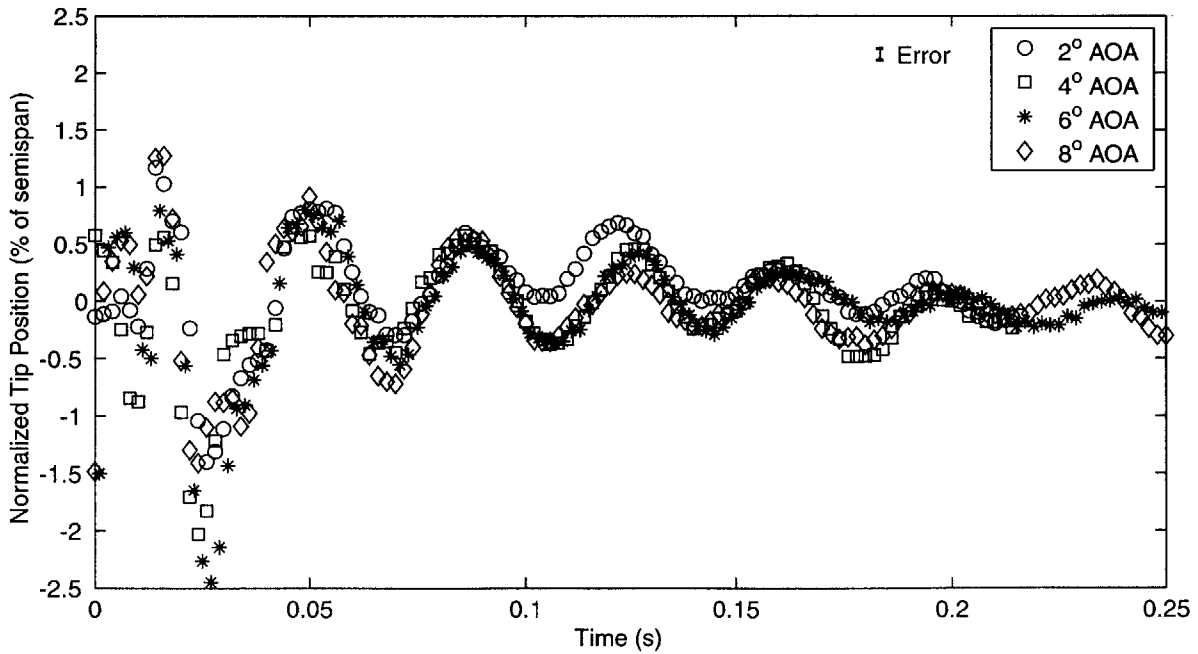
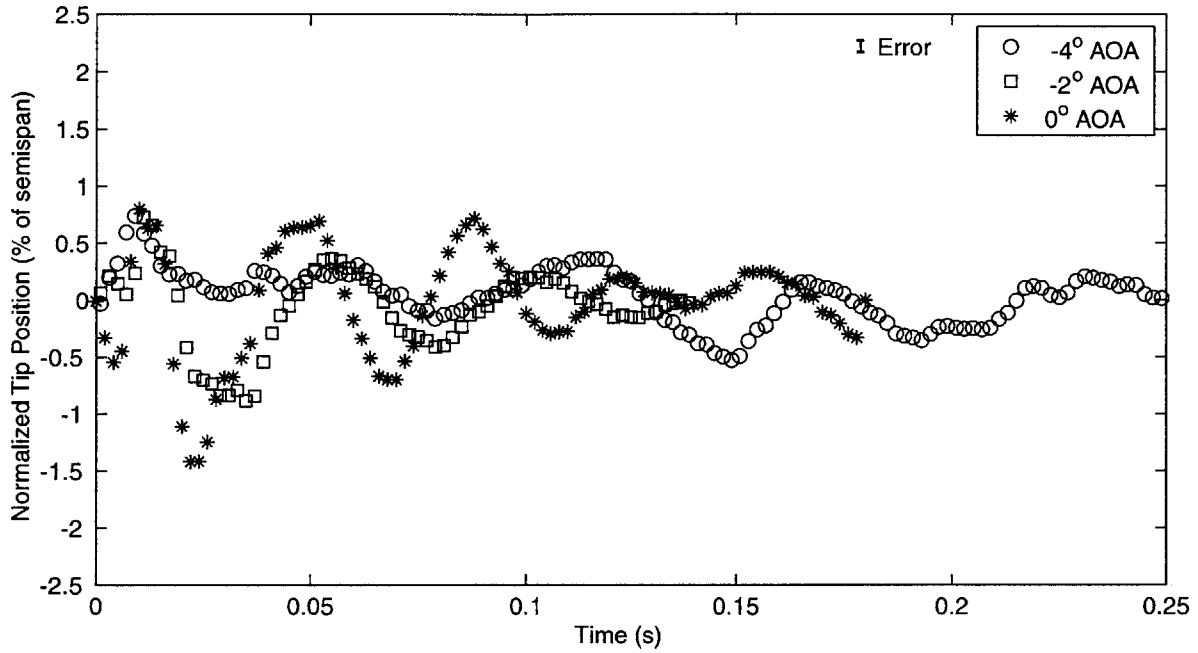


Figure 2-12: Bending response of WASP wing at various root AOAs and 46 m/s

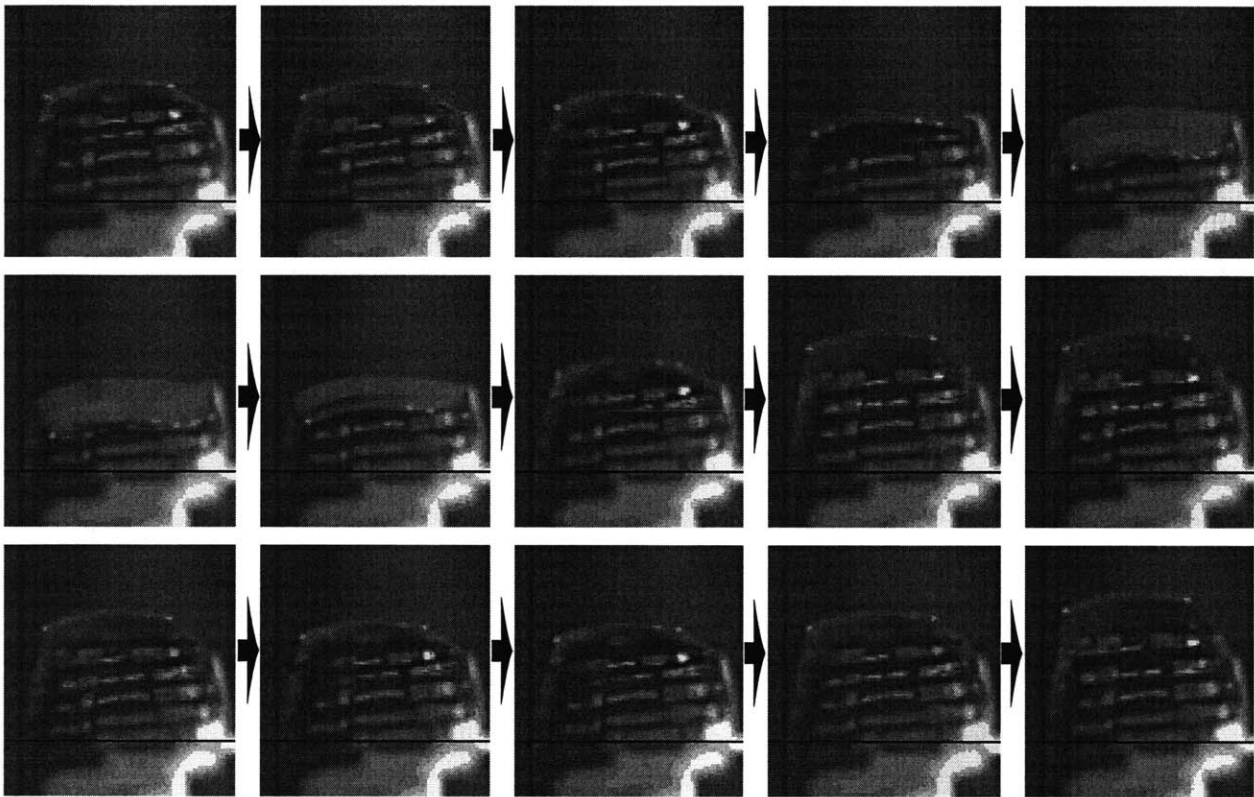


Figure 2-13: Video sequence of WASP wing at 46 m/s and -5° root AOA with 0.02 seconds between images

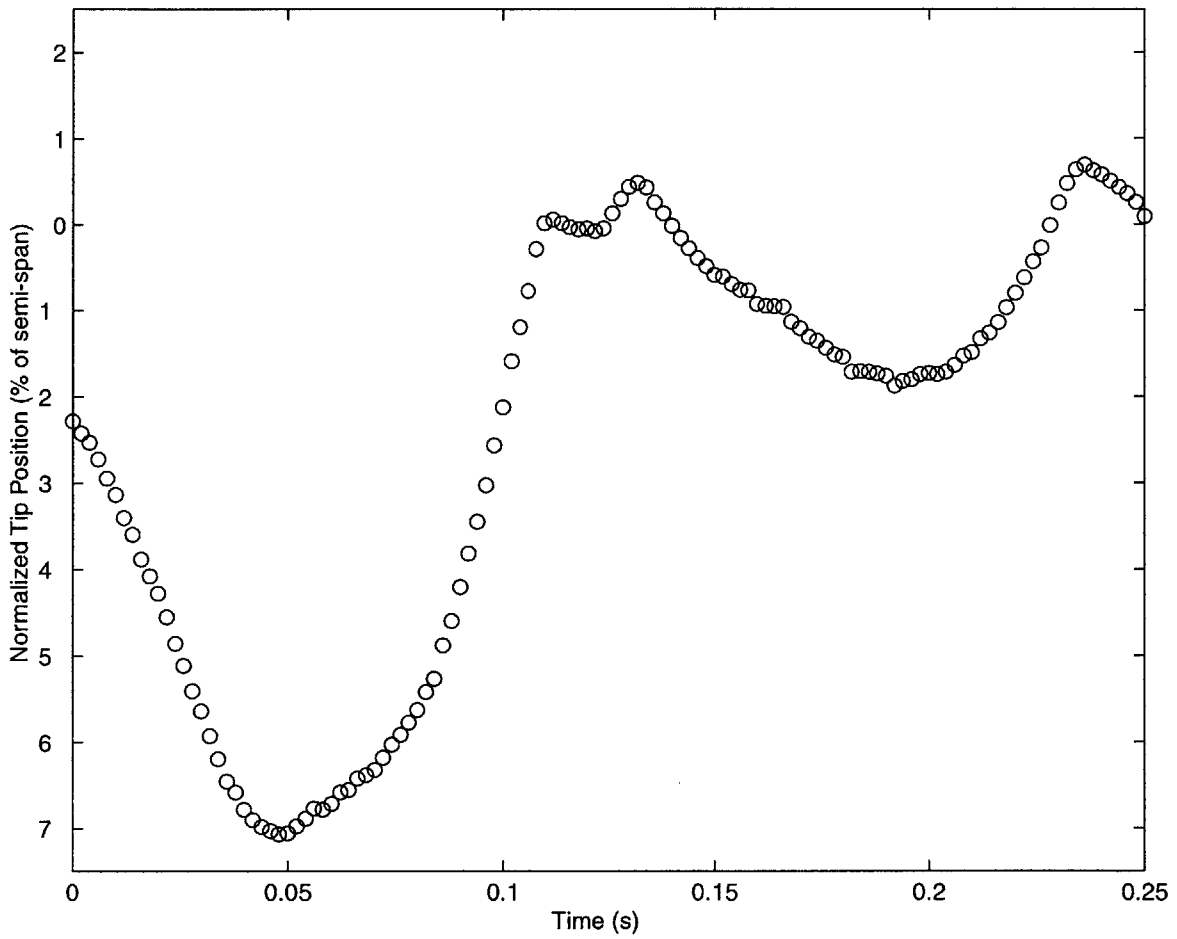


Figure 2-14: Bending response of WASP wing at 46 m/s and -5° root AOA (corresponding to case shown in Figure 2-13)

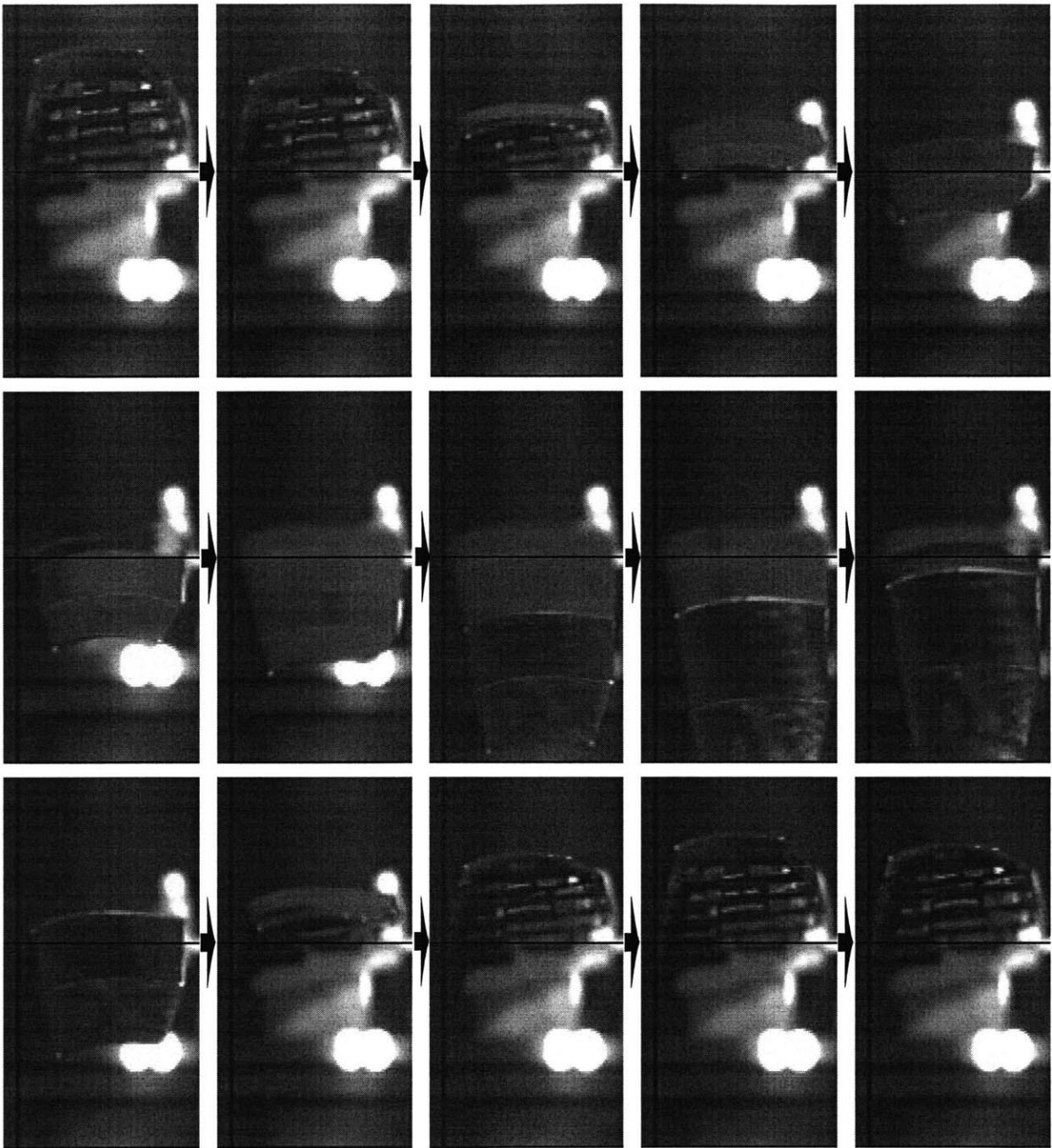


Figure 2-15: Video sequence of largest amplitude refold motion captured at 46 m/s and -5° root AOA with 0.04 seconds between images

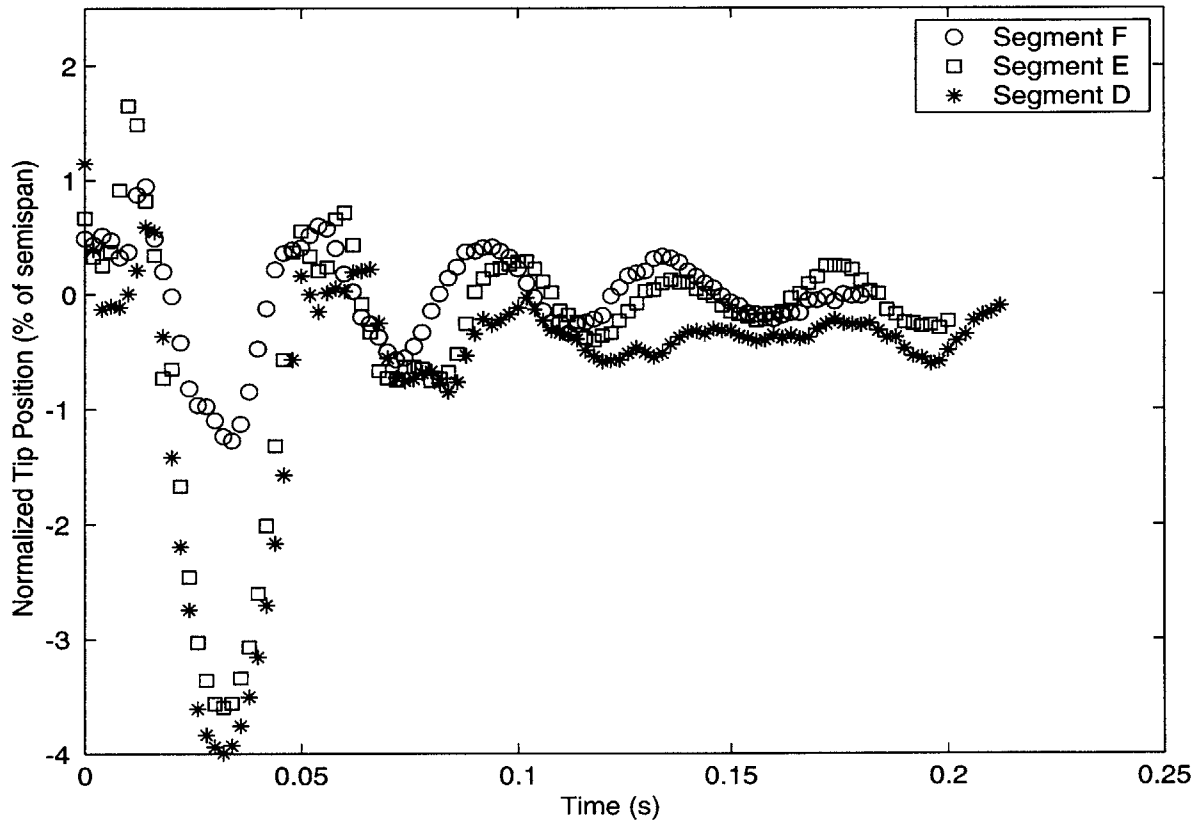


Figure 2-16: Bending response of WASP wing at 46 m/s and 0° root AOA with different segments deflected (refer to Figure 1-5)

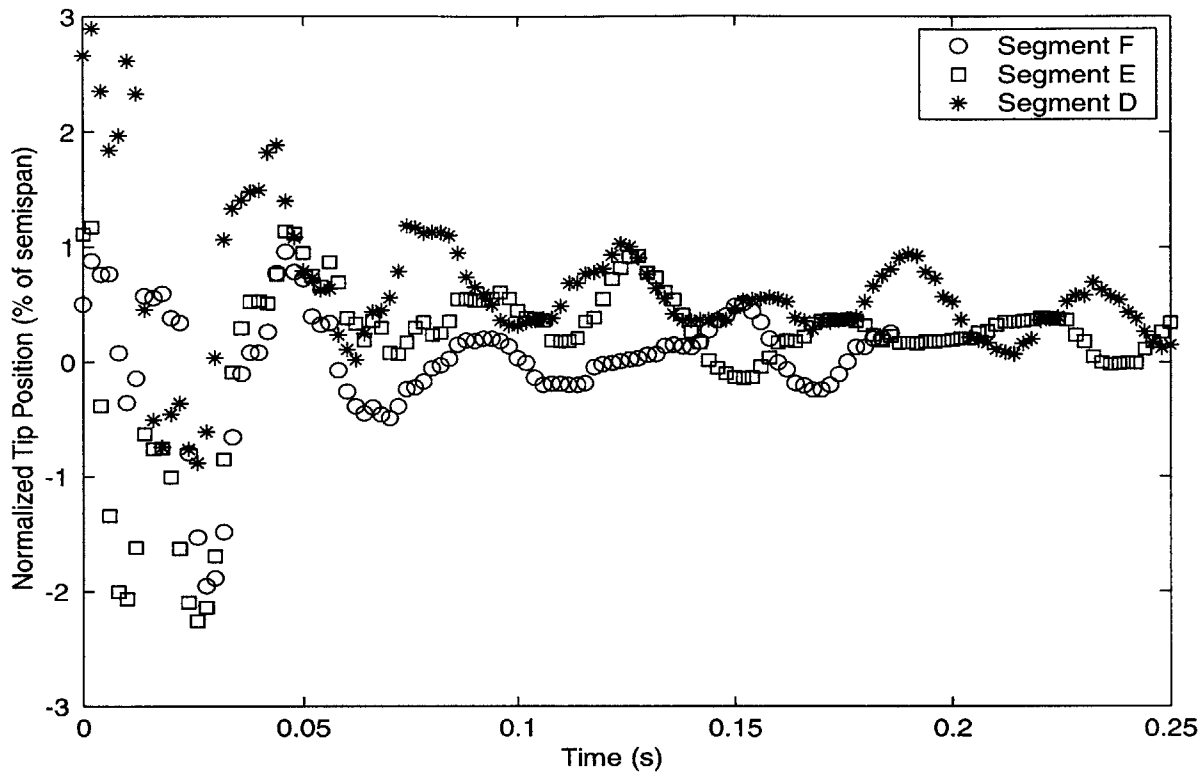


Figure 2-17: Bending response of WASP wing at 66 m/s and 0° root AOA with different segments deflected (refer to Figure 1-5)

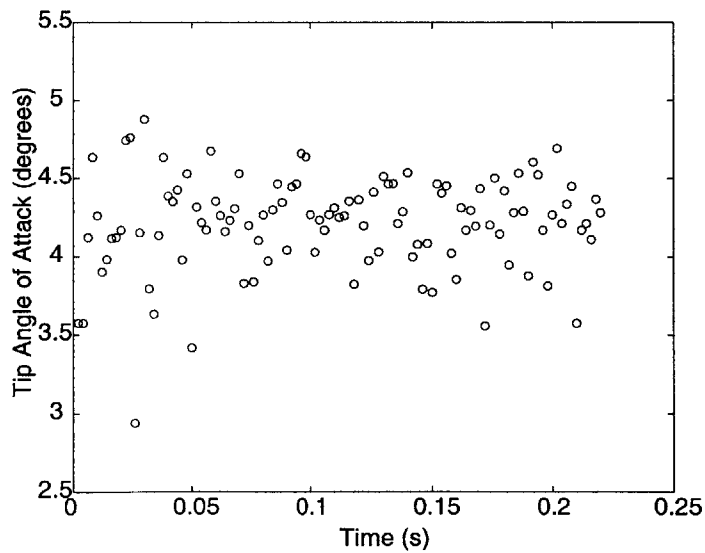


Figure 2-18: Typical experimental AOA measurements (taken at 46 m/s and 4° root AOA)

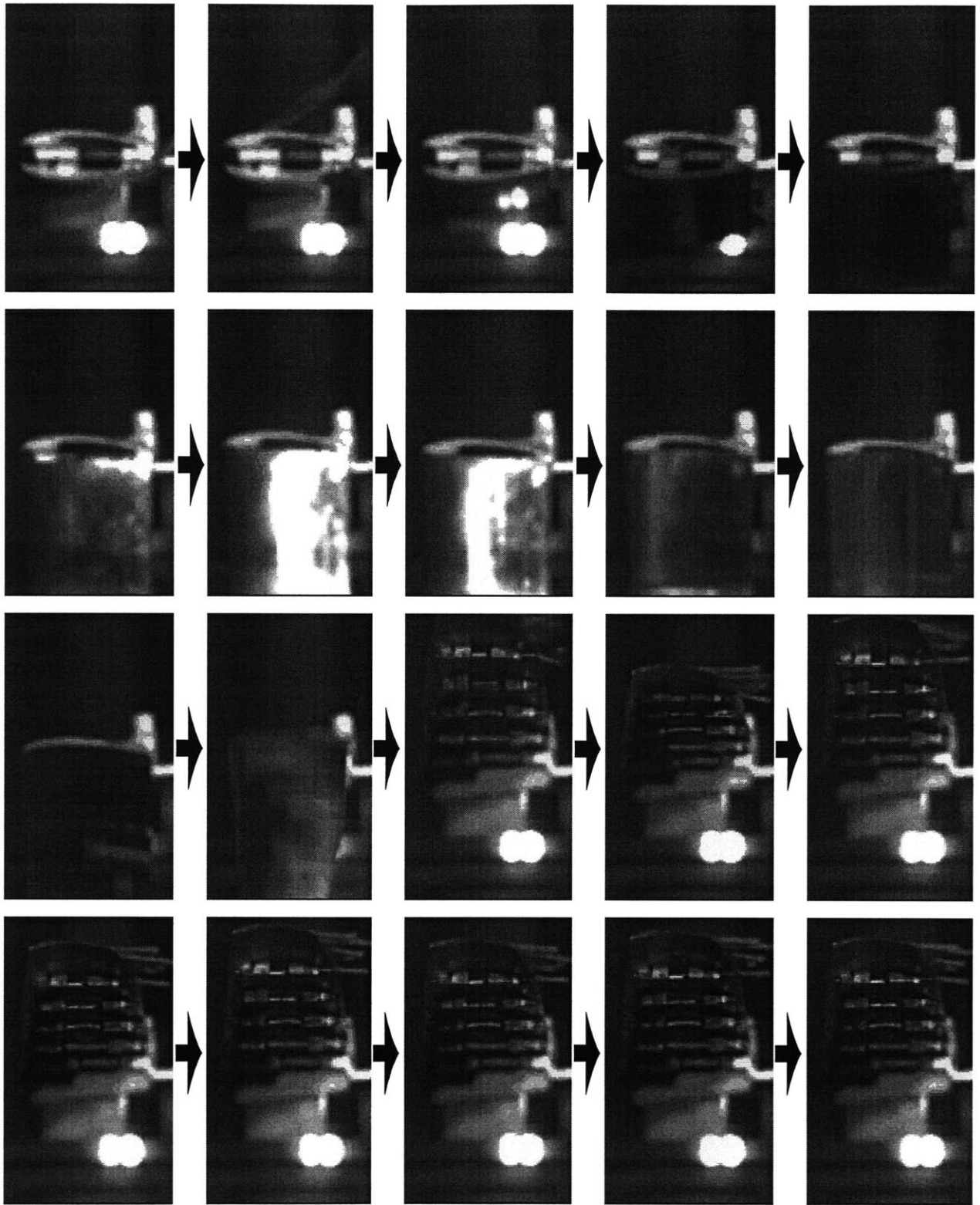


Figure 2-19: Video sequence of wing deployment at 66 m/s and 0° root AOA with 0.02 seconds between images

Chapter 3

Theoretical Modeling

A nonlinear aeroelastic model was created to simulate a multi-hinged wing at different flight conditions. It is able to handle both the large hinge angles and the piecewise stiffness found in a hinged system. The inputs of the model include the wing cross-sectional properties (geometric, structural and aerodynamic), the number of wing segments, the mechanical properties of the hinges, and the flow parameters. Using a direct time marching scheme, the model determines the change in state of the wing over time.

The model was developed by initially simulating a dynamic system with multiple rigid bodies. Next, the forcing terms were added one at a time using models that were independently verified. The aerodynamic model, as well as the integration schemes, were based on methods used by Conner et al. [17].

3.1 Multi-Body Dynamics

The multi-body dynamics are modeled using the methods developed by Shabana [7] which uses Lagrangian multipliers to satisfy system constraints. In order to create a full wing model, a double pendulum problem is used to exemplify its development. The double pendulum is a classic chaotic problem [26] with two rigid beams connected end-to-end by a pin joint. The unconnected end of one of the beams is pinned in space as shown in Figure 3-1. A hinged wing consists of a number of elastic beams connected by pin joints, of which one of the most

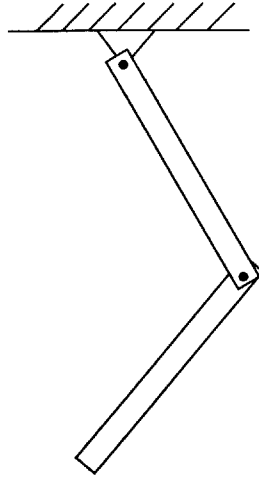


Figure 3-1: Diagram of double pendulum

inboard segment has its unconnected end clamped in space.

For a system of rigid beams connected by pin joints constrained to move in a plane, only one degree of freedom (DOF) is needed to describe the state of each beam. However, to simplify the process when doing large numbers of connected beams, three DOFs are used, of which only one is independent. A system of constraint equations is used to define the relationships between the dependent and independent DOFs.

Two sets of coordinate systems are used to describe the state of the system (Figure 3-2). The first (X, Y) is a single inertial frame that is common to all bodies. The second set of frames (x_i, y_i) consists of local frames that are each attached to a body and can translate and rotate with respect to the inertial frame. For this discussion i is used to distinguish each body frame and j is used to specify the j th point or element on the i th body. Any arbitrary point in the inertial coordinate system can be described by the position vector, \mathbf{r}_{ij} , which can be determined by:

$$\mathbf{r}_{ij} = \mathbf{R}_i + \mathbf{A}_i \mathbf{u}_{ij} \quad (3.1)$$

where \mathbf{R}_i is the location vector of the origin of the body coordinate frame in the inertial coordinate system. For the double pendulum, the body coordinate systems are placed at the root of the beams to simplify the formulation that will be encountered when elastic beams are considered. \mathbf{u}_{ij} is the location vector of the j th point in the i th body coordinate system. \mathbf{A}_i is the rotation matrix of the i th body frame used to rotate \mathbf{u}_{ij} to the inertial frame, given

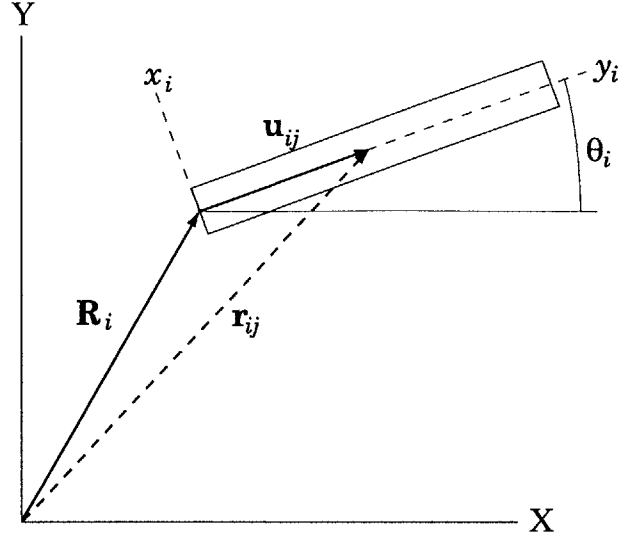


Figure 3-2: Coordinate systems used in multi-body simulation

by:

$$\mathbf{A}_i = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i) \end{bmatrix} \quad (3.2)$$

where θ_i is the rotation angle of the i th body with respect to the inertial frame. Thus, the state of a rigid body can be described using the vector \mathbf{q}_i , i.e.,

$$\mathbf{q}_i = \begin{bmatrix} \mathbf{R}_i \\ \theta_i \end{bmatrix} = \begin{bmatrix} X_i \\ Y_i \\ \theta_i \end{bmatrix} \quad (3.3)$$

The change of \mathbf{r}_{ij} over time can be found by differentiating Equation 3.1, that is

$$\dot{\mathbf{r}}_{ij} = \dot{\mathbf{R}}_i + \frac{\partial \mathbf{A}_i}{\partial \theta_i} \mathbf{u}_{ij} \dot{\theta}_i \quad (3.4a)$$

and

$$\mathbf{A}_{\theta_i} \equiv \frac{\partial \mathbf{A}_i}{\partial \theta_i} = \begin{bmatrix} -\sin(\theta_i) & -\cos(\theta_i) \\ \cos(\theta_i) & -\sin(\theta_i) \end{bmatrix} \quad (3.4b)$$

Using the virtual work, δW , done by all forces, F , acting on the system, a vector of

generalized forces \mathbf{Q} can be found through:

$$\delta W = \mathbf{F}^T \delta \mathbf{r} = \mathbf{Q}^T \delta \mathbf{q} \quad (3.5)$$

where $\delta \mathbf{q}$ is a vector of system virtual displacements. The vector of generalized forces can be used in Lagrange's Equation to determine the equations that describe the motion of body i through

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{\mathbf{q}}_i} \right)^T - \left(\frac{\partial T}{\partial \mathbf{q}_i} \right)^T = \mathbf{Q}_i \quad (3.6)$$

where T is the total kinetic energy of the system and \mathbf{Q}_i is the portion of \mathbf{Q} related to $\delta \mathbf{q}_i$, the virtual displacements of the i th body.

The total kinetic energy of the system is the sum of the kinetic energy of each body, T_i . The kinetic energy of the i th body is given by

$$T_i = \frac{1}{2} \int_{Vol_i} \rho \mathbf{r}_{ij}^T \dot{\mathbf{r}}_{ij} dV \quad (3.7)$$

where ρ is the density of the body and can be a function of location. Using Equation 3.4a in Equation 3.7, a matrix \mathbf{M}_i can be defined from:

$$T_i = \frac{1}{2} \dot{\mathbf{q}}_i^T \mathbf{M}_i \dot{\mathbf{q}}_i \quad (3.8)$$

\mathbf{M}_i can be separated into its subcomponents \mathbf{M}_{RR_i} , $\mathbf{M}_{R\theta_i}$, and $\mathbf{M}_{\theta\theta_i}$. The subindex R refers to a rigid body translation, and the subindex θ refers to rigid body rotation. Thus $\mathbf{M}_{R\theta_i}$ is the subcomponent of \mathbf{M}_i which is the influence on rigid body translations due to rigid body rotation. These subcomponents are found by expanding the above equation giving,

$$\mathbf{M}_i = \int_{Vol} \rho \begin{bmatrix} \mathbf{I} & \mathbf{A}_{\theta_i} \mathbf{u}_{ij} \\ (\mathbf{A}_{\theta_i} \mathbf{u}_{ij})^T & \mathbf{u}_{ij}^T \mathbf{u}_{ij} \end{bmatrix} dV = \begin{bmatrix} \mathbf{M}_{RR_i} & \mathbf{M}_{R\theta_i} \\ \mathbf{M}_{R\theta_i}^T & \mathbf{M}_{\theta\theta_i} \end{bmatrix} \quad (3.9)$$

where \mathbf{I} is the identity matrix.

A double beam pendulum consists of beam-like bodies. The assumptions that the length of the beam, l_i , is much greater than its other dimensions and that $\mathbf{u}_{ij} = [x_i \ 0]^T$ allows

for the mass matrix subcomponents of the double beam pendulum to be determined by:

$$\mathbf{M}_{RR_i} = \int_{Vol} \rho \mathbf{I} dV = \begin{bmatrix} m_i & 0 \\ 0 & m_i \end{bmatrix} \quad (3.10a)$$

$$\mathbf{M}_{R\theta_i} = \int_{Vol} \rho x_i \begin{Bmatrix} -\sin \theta_i \\ \cos \theta_i \end{Bmatrix} dV = \frac{m_i l_i}{2} \begin{Bmatrix} -\sin \theta_i \\ \cos \theta_i \end{Bmatrix} \quad (3.10b)$$

$$\mathbf{M}_{\theta\theta_i} = \int_{Vol} \rho x_i^2 dV = \frac{m_i l_i^2}{3} \quad (3.10c)$$

where m_i is the mass of the i^{th} body. For a body with the coordinate system placed at the center of mass, the $\mathbf{M}_{R\theta_i}$ terms would be zero. For the case of the double pendulum, the local coordinate systems are placed at the root of the beams, which leads to the mass matrix having off-diagonal terms. Note that $\mathbf{M}_{R\theta_i}$ has state variables, thus the model is nonlinear.

The system can be written in the form $a\ddot{x} = b$ by placing Equation 3.8 into Equation 3.6 and using the convenient forcing term \mathbf{Q}_{v_i} , that is

$$\mathbf{M}_i \ddot{\mathbf{q}}_i + \underbrace{\dot{\mathbf{M}}_i \dot{\mathbf{q}}_i - \left[\frac{\partial}{\partial \mathbf{q}_i} \left(\frac{1}{2} \dot{\mathbf{q}}_i^T \mathbf{M}_i \dot{\mathbf{q}}_i \right) \right]^T}_{-\mathbf{Q}_{v_i}} = \mathbf{Q}_i \quad (3.11a)$$

$$\mathbf{M}_i \ddot{\mathbf{q}}_i = \mathbf{Q}_i + \mathbf{Q}_{v_i} \quad (3.11b)$$

\mathbf{Q}_{v_i} is found by solving its two parts separately, and then finding their sum. As shown in Equation 3.12b, due the mass matrix being independent of the translation terms, the derivative of T_i with respect to \mathbf{R}_i is zero. Therefore, the first two terms of the second part are zero.

$$-\dot{\mathbf{M}}_i \dot{\mathbf{q}}_i = \frac{m_i l_i}{2} \begin{bmatrix} 0 & 0 & \dot{\theta}_i \cos \theta_i \\ 0 & 0 & \dot{\theta}_i \sin \theta_i \\ \dot{\theta}_i \cos \theta_i & \dot{\theta}_i \sin \theta_i & 0 \end{bmatrix} \dot{\mathbf{q}}_i = \frac{m_i l_i}{2} \left\{ \begin{array}{l} \dot{\theta}_i^2 \cos \theta_i \\ \dot{\theta}_i^2 \sin \theta_i \\ \left(\dot{X}_i \cos \theta_i + \dot{Y}_i \sin \theta_i \right) \dot{\theta}_i \end{array} \right\} \quad (3.12a)$$

$$\frac{\partial}{\partial \mathbf{R}_i} \left(\frac{1}{2} \dot{\mathbf{q}}_i^T \mathbf{M}_i \dot{\mathbf{q}}_i \right) = \frac{1}{2} \dot{\mathbf{q}}_i^T \left(\frac{\partial \mathbf{M}_i}{\partial \mathbf{R}_i} \right) \dot{\mathbf{q}}_i = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} \quad (3.12b)$$

since

$$\frac{\partial \mathbf{M}_i}{\partial \mathbf{X}_i} = \frac{\partial \mathbf{M}_i}{\partial \mathbf{Y}_i} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\begin{aligned} \left[\frac{\partial}{\partial \theta_i} \left(\frac{1}{2} \dot{\mathbf{q}}_i^T \mathbf{M}_i \dot{\mathbf{q}}_i \right) \right]^T = \\ -\frac{m_i l_i}{4} \dot{\mathbf{q}}_i^T \begin{bmatrix} 0 & 0 & \cos \theta_i \\ 0 & 0 & \sin \theta_i \\ \cos \theta_i & \sin \theta_i & 0 \end{bmatrix} \dot{\mathbf{q}}_i = -\frac{m_i l_i \dot{\theta}_i}{2} (\dot{X}_i \cos \theta_i + \dot{Y}_i \sin \theta_i) \end{aligned} \quad (3.12c)$$

$$\mathbf{Q}_{v,i} = \frac{m_i l_i \dot{\theta}_i^2}{2} \begin{Bmatrix} \cos \theta_i \\ \sin \theta_i \\ 0 \end{Bmatrix} \quad (3.12d)$$

where X_i and Y_i are the components of \mathbf{R}_i .

In order to solve the complete system, constraints must be added to ensure the members remain connected. This is performed by finding the array of holonomic constraint equations such that $\mathbf{C}(\mathbf{q}, t) = 0$ (where the vector \mathbf{q} describes the state of the system and is composed of the body state vectors \mathbf{q}_i , i.e. $\mathbf{q} = [X_1 Y_1 \theta_1 X_2 Y_2 \theta_2 \dots]$). For a double pendulum, this results in four equations: the first two fix the location of the root of the first member, and the last two equations describe the location of the root of the second member in terms of the first member. The resultant \mathbf{C} equation for the double pendulum is:

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \mathbf{q} + \begin{Bmatrix} 0 \\ 0 \\ -l_1 \cos \theta_1 \\ -l_1 \sin \theta_1 \end{Bmatrix} = \mathbf{\Phi} \mathbf{q} + \mathbf{f}_c = 0 \quad (3.13)$$

Therefore \mathbf{C} can be written as a sum of a linear ($\Phi\mathbf{q}$) and a nonlinear (\mathbf{f}_c) part. Virtual displacement is used to find the variation of \mathbf{C} , giving a new equation $\mathbf{C}_q\delta\mathbf{q} = 0$, where \mathbf{C}_q is the system Jacobian constraint matrix, an example of which can be seen in Equations D.3 and D.4 in Appendix D. This is combined with Equation 3.6 to create a new system of equations of motion with a vector of Lagrange multipliers, λ , i.e.,

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{\mathbf{q}}} \right)^T - \left(\frac{\partial T}{\partial \mathbf{q}} \right)^T + \mathbf{C}_q^T \lambda = \mathbf{Q} \quad (3.14)$$

In order to prevent violation of the constraint equations, \mathbf{C} , as the system is solved over time, the constraint equations are rewritten in the form of a critically damped oscillator. The oscillator has a diagonal natural frequency matrix, Ω , based on the natural frequencies of the constraints. Hence,

$$\ddot{\mathbf{C}} + 2\Omega\dot{\mathbf{C}} + \Omega^2\mathbf{C} = \mathbf{0} \quad (3.15a)$$

But from the basic chain rule:

$$\dot{\mathbf{C}} = \mathbf{C}_q\dot{\mathbf{q}} \quad (3.15b)$$

and therefore

$$\ddot{\mathbf{C}} = \mathbf{C}_q\ddot{\mathbf{q}} + \mathbf{Q}_c \quad (3.15c)$$

where the equation has been separated into its linear ($\mathbf{C}_q\ddot{\mathbf{q}}$) and nonlinear (\mathbf{Q}_c) components. Combining Equation 3.15a with Equation 3.15c gives:

$$\mathbf{C}_q\ddot{\mathbf{q}} = -\mathbf{Q}_c - 2\Omega\dot{\mathbf{C}} - \Omega^2\mathbf{C} \quad (3.15d)$$

Placing the final equation into Equation 3.14 yields:

$$\begin{bmatrix} \mathbf{M} & \mathbf{C}_q^T \\ \mathbf{C}_q & 0 \end{bmatrix} \begin{Bmatrix} \ddot{\mathbf{q}} \\ \lambda \end{Bmatrix} = \begin{Bmatrix} \mathbf{Q}_v + \mathbf{Q} \\ -\mathbf{Q}_c - 2\Omega\dot{\mathbf{C}} - \Omega^2\mathbf{C} \end{Bmatrix} \quad (3.16)$$

A full development of the double-pendulum problem can be found in Appendix D.

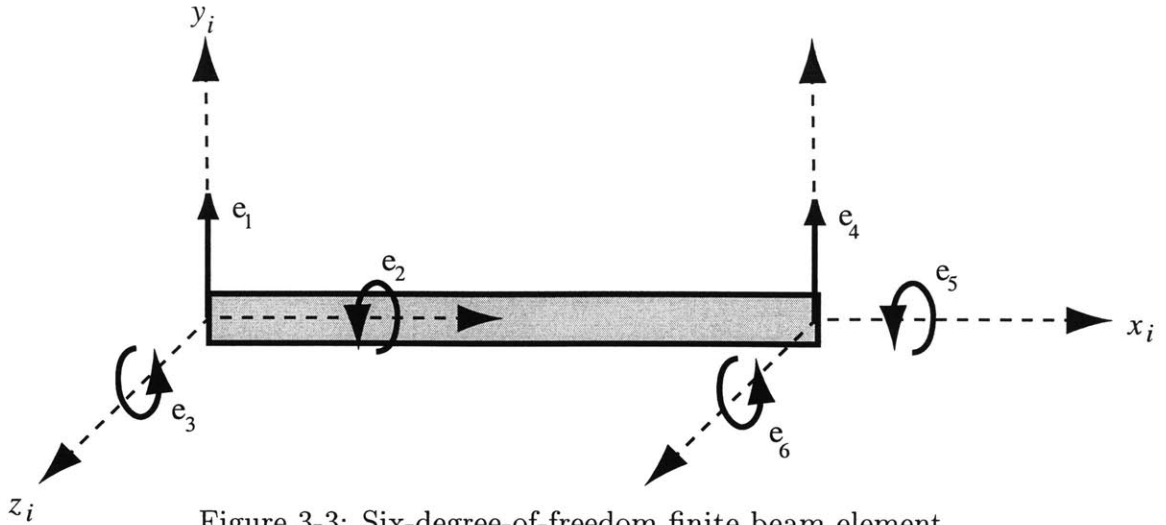


Figure 3-3: Six-degree-of-freedom finite beam element

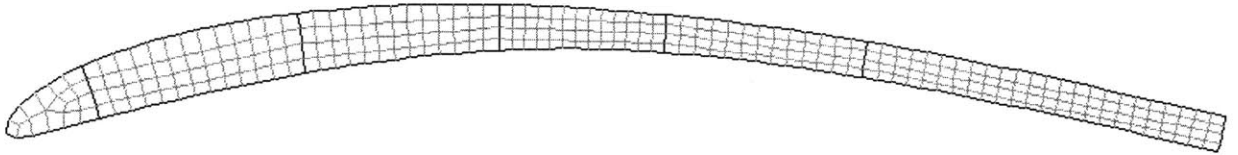


Figure 3-4: Mesh of wing cross section using quadrilateral elements for use with VABS

3.2 Elastic Model

Each wing segment is set up as a linear elastic beam model that is solved using the finite element method (FEM). The segments are divided into an equal number of six-degree-of-freedom beam elements. The element nodes are allowed to translate in the y direction and rotate about the x and z axes as shown in Figure 3-3. The elastic extensional and chordwise motions of the wing are not included in the model. These DOFs are not coupled to those of interest and are too constrained to have significant effect on the aerodynamic forces, and thus they were neglected.

The cross-sectional properties of the beam are solved using the Variational Asymptotic Beam Sectional Analysis code (VABS) [27]. The two dimensional grid shown in Figure 3-4 is used for the VABS input of the WASP wing. VABS outputs a stiffness in all three translational and rotational directions as well as cross-coupling terms.

The elastic axis of the wing is found using the VABS results. Using the elastic axis as the origin of the coordinate system removed all the stiffness coupling terms, but leaves the system with inertial coupling terms that need to be taken into account.

The elastic degrees of freedom are added to the rigid multi-body system developed in Section 3.1 using an element shape function \mathbf{S} and the DOFs shown in Figure 3-3. The displacement of the j th element of the i th body is determined by

$$\mathbf{r}_{ij} = \mathbf{R}_i + \mathbf{A}_i \mathbf{S} \mathbf{e}_{ij} \quad (3.17)$$

Following the standard procedure for the FEM, the elements of each body are assembled together. Each body of the system has the original three rigid DOFs (\mathbf{R}_i and θ_i) and a number of elastic DOFs (which is dependent on the number of elements in the body). The elastic DOFs are measured with respect to the undeformed body in the body's local coordinate system. Using this process, none of the underlying FEM mathematics needs to be altered.

Constraints need to be applied to remove the fundamental rigid body motion present in the finite element. To fix the body relative to its local coordinate system, three elastic DOFs are constrained. This is due to the rigid body motions having already been taken into account, as presented in Section 3.1. The root DOF associated with twist is fixed relative to the previous segment's tip twist. An exception of such is when the previous segment is the most inboard segment, which is fixed relative to the inertial frame. Thus, the DOFs associated with twist are elastically decoupled from the effects of the hinges. This leaves two of the DOFs associated with bending to be fixed. Either a simply supported or a cantilever boundary condition can be used (coordinate systems "a" or "b" in Figure 3-5). The difference between the methods is accounted for when the multi-body system is assembled. Different constraints are required by the methods to insure that the next segment is properly connected. For the multi-hinged wing model with a clamped root, the most inboard segment uses a cantilever constraint, while the other segments are simply supported.

Using Lagrange's Equations again, the kinetic energy of each of the elements can be found and then summed to yield the equations of motion for a flexible multi-body system.

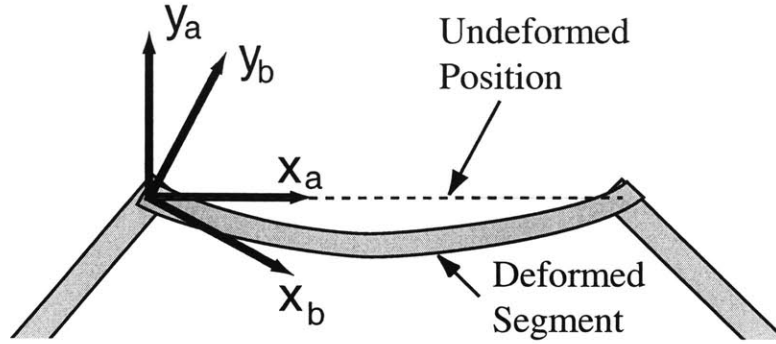


Figure 3-5: Possible elastic boundary conditions

Using the techniques discussed in the previous section, Equation 3.16 can be augmented to include the flexibility effects as:

$$\begin{bmatrix} \mathbf{M}_{rr} & \mathbf{M}_{rf} & \mathbf{C}_{q_r}^T \\ \mathbf{M}_{fr} & \mathbf{M}_{ff} & \mathbf{C}_{q_f}^T \\ \mathbf{C}_{q_r}^T & \mathbf{C}_{q_f}^T & 0 \end{bmatrix} \begin{Bmatrix} \ddot{\mathbf{q}}_r \\ \ddot{\mathbf{q}}_f \\ \lambda \end{Bmatrix} = \begin{Bmatrix} \mathbf{Q}_{v_r} + \mathbf{Q}_r \\ \mathbf{Q}_{v_f} + \mathbf{Q}_f - \mathbf{K}_{ff}q_f \\ -\mathbf{Q}_c - 2\Omega\dot{\mathbf{C}} - \Omega^2\mathbf{C} \end{Bmatrix} \quad (3.18)$$

where the subscript r corresponds to rigid DOF (both translational and rotational) and f corresponds to flexible ones. For example, $\mathbf{q}_{ri} = [X_{1i} Y_{1i} \theta_{1i}]$ and $\mathbf{q}_{fi} = [e_{1i} e_{2i} e_{3i} \dots]$. The forces due to elastic deformation are placed in the right-hand side of the equation as $\mathbf{K}_{ff}q_f$. \mathbf{K}_{ff} is a standard FEM stiffness matrix for the previously mentioned beam elements with applied boundary conditions.

3.3 Applied Forces

The model needs to be excited by forces to have anything other than a trivial solution. For the double pendulum, the only force acting on the system is gravity. For the hinged wing, gravitational, aerodynamic, structural, and spring forces are taken into account. Figure 3-6 shows a representation of the different forces. All of the forces and moments are translated such that they act on the origin of the beam coordinate system. As stated in the previous section, this reference axis is the loci of the shear centers so that the beam is elastically

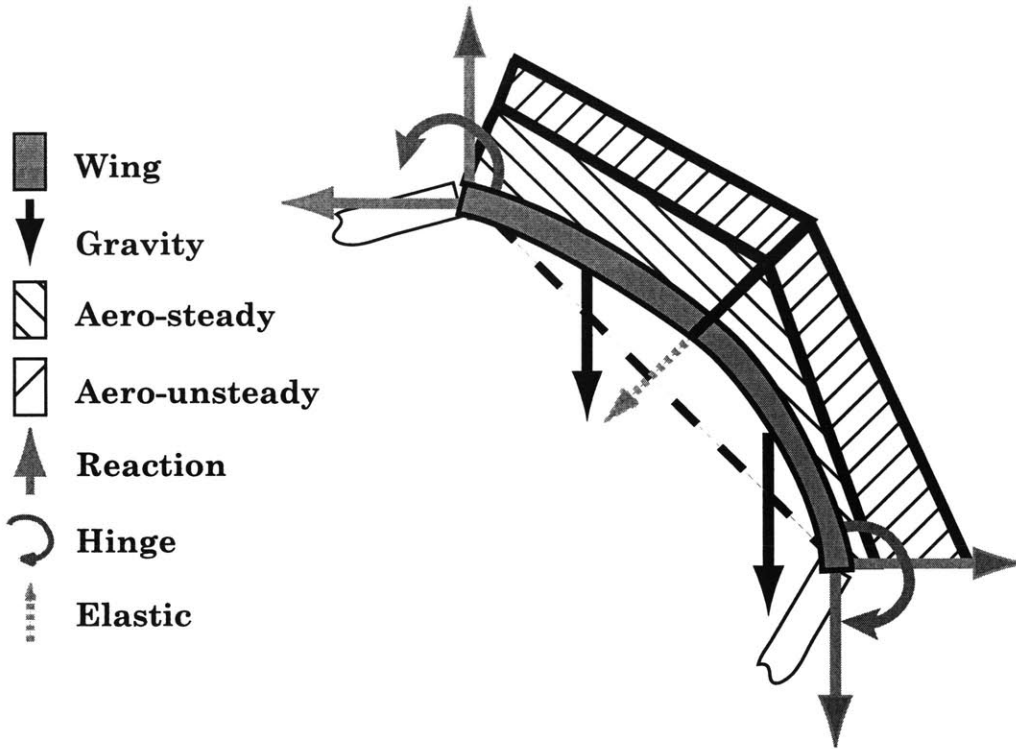


Figure 3-6: Loads applied to a flexible segment of a multi-hinge beam structure

uncoupled.

3.3.1 Gravitational Force

The force of gravity always acts in the same direction relative to the inertial reference frame. If the problem is set up such that the coordinate system is oriented with the Y axis pointing in the opposite direction of the force of gravity, F_g , then the the force on body i acting on its center of gravity is:

$$\mathbf{F}_{g_i} = \begin{Bmatrix} 0 \\ -m_i g \end{Bmatrix} \quad (3.19)$$

where g is the acceleration due to gravity.

A generalized gravitational forcing vector for a rigid body i , \mathbf{Q}_{g_i} , can be found using virtual work, Equation 3.5:

$$\delta W_i = \mathbf{F}_{g_i}^T \delta \mathbf{r}_c = \mathbf{F}_{g_i}^T \left[\mathbf{I} \quad \mathbf{A}_{\theta_i} \begin{Bmatrix} \frac{l_i}{2} \\ 0 \end{Bmatrix} \right] \delta \mathbf{q}_i = \underbrace{\left[0 \quad -m_i g \quad \frac{-m_i g l_i}{2} \cos \theta_i \right]}_{\mathbf{Q}_{g_i}^T} \delta \mathbf{q}_i \quad (3.20)$$

where \mathbf{r}_c is the location of the center of gravity of the body (assumed to be the midpoint of the beam element) in the inertial frame. Using \mathbf{Q}_{g_i} for \mathbf{Q}_i into Equation 3.16 will allow the rigid double pendulum to be solved. To solve for a flexible beam, elastic DOFs are added to the system giving

$$\mathbf{Q}_{g_{\ddot{y}_r}} = \left\{ \begin{array}{c} 0 \\ -m_i g \\ m_i g \left(\sin \theta_i \left[-\frac{1}{2} \quad 0 \quad \frac{-l_{ij}}{12} \quad -\frac{1}{2} \quad 0 \quad \frac{l_{ij}}{12} \right] \mathbf{e}_{ij} - \cos \theta_i \left(\frac{x_{\ddot{y}_1} + x_{\ddot{y}_2}}{2} \right) \right) \end{array} \right\} \quad (3.21)$$

$$\mathbf{Q}_{g_{\ddot{y}_f}} = m_i g \cos \theta_i \left[-\frac{1}{2} \quad 0 \quad \frac{-l_{ij}}{12} \quad -\frac{1}{2} \quad 0 \quad \frac{l_{ij}}{12} \right]^T \quad (3.22)$$

where l_{ij} is the length of the j^{th} element of the i^{th} body and $x_{\ddot{y}_1}$ and $x_{\ddot{y}_2}$ are the locations of the ends of the j^{th} element in the local frame. For the elastic model, the load applied to the rigid body rotational DOF is dependent on the state of the elastic DOFs. The example shown is for the double beam pendulum, in which the elastic axis and the center of gravity of the cross section are aligned, thus there is no resulting applied torsional moment.

3.3.2 Hinge Torque

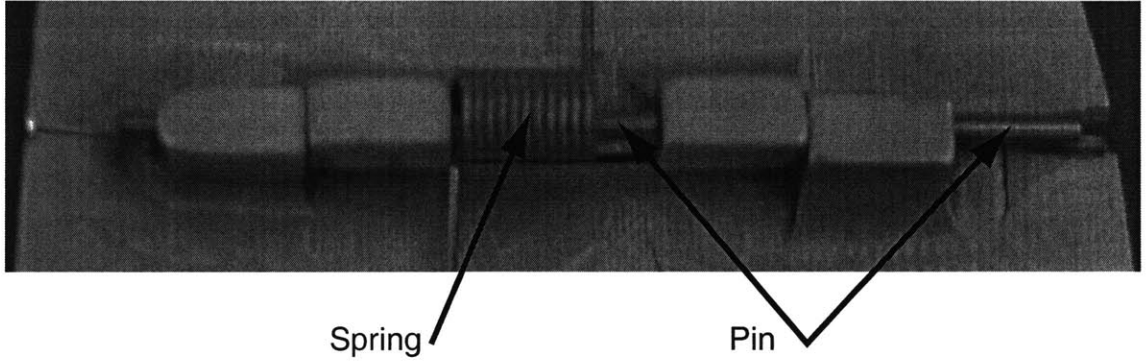


Figure 3-7: Photo of tip-most WASP wing hinge showing typical mechanical connections

The wing segments are connected together by revolute joints, or hinges, as shown in Figure 3-7. Each hinge has six properties that are important to the wing's aeroelastic response: spring stiffness, spring torque at zero angle, open stiffness, damping, mass, and

angle at which the hinge switches states. An open hinge is a fully deployed hinge.

The wing hinges are modeled with a two-state or bilinear piecewise stiffness shown in Figure 3-8. If the hinge is partially closed, the stiffness is due to the attached torsional springs. This will be referred to as the soft state. On the other hand, if the hinge is fully opened, the ends of the segments abut, resulting in a relatively large stiffness. This will be referred to as the stiff state.

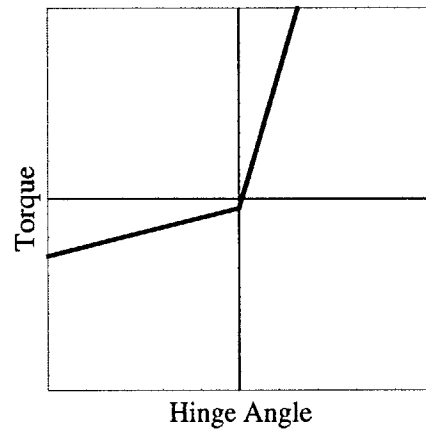


Figure 3-8: Bi-linear nature of hinge stiffness

At first it was thought that, at the hinge point, an open hinge could be modeled as a continuous beam. If that was the case, the moment applied to the beam would have been related to the curvature of the beam at that point. However, the continuous beam assumption led to a model that was stiffer than what was experimentally observed. Therefore, the hinge stiffness was modeled more like a torsional spring; the moment being related to the change in slope at the hinge. In fact, the tip-most hinge pin is observed to deflect if a moment is applied to the hinge. Instead of pivoting about the hinge pin (as is done in the soft state) a single segment revolves around the contact point with the previous segment as shown in Figure 3-9. The flexibility of the hinge is then assumed to be mainly due to the elasticity of the hinge pin. Finding the boundary conditions of the hinge pin may be non-trivial due to the large contribution of any machining tolerances.

A viscous model was used for the hinge damping effects. This leads to a similar beam model to that used by Bowden and Dugundji [28], except that the hinges have a piecewise stiffness.

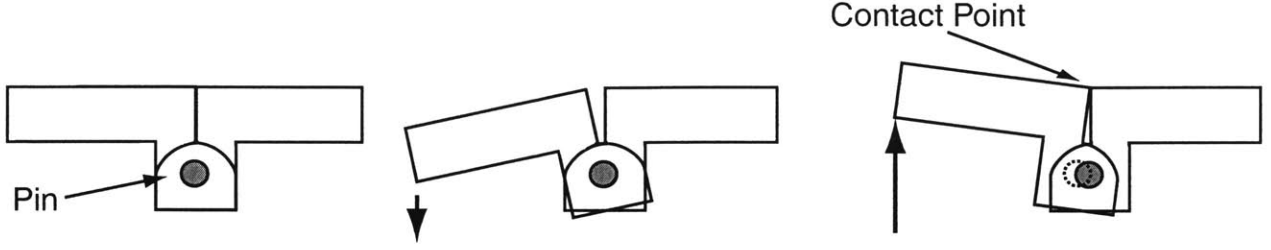


Figure 3-9: Hinge (a) open, (b) partially closed, and (c) open (with deformation)

The torque, τ , at each hinge was modeled by,

$$\tau_i = B_{h_i} \dot{\beta}_i + K_{h_i} (\beta_i - \beta_{0i}) + K_{h0_i} \quad (3.23)$$

where β_i is the relative angle between the root of the i^{th} segment and the tip of the $(i-1)^{th}$ segment. β_i takes into account the possible elastic tip rotation with respect to the body coordinate system. β_{0i} represents the angle at which the state of the hinge changes. B_{h_i} is the viscous damping term. K_{h_i} is the stiffness of the i^{th} segment's hinge, and will be one of two values depending on β_i . K_{h0_i} is the torque of the spring when β_i is β_{0i} . The hinge torques only influence the DOFs associated with the angle of the rigid body and the elastic rotations of the beam tips.

To incorporate the hinge torque into the model, the forcing vector \mathbf{Q}_h is used. For the i^{th} body, \mathbf{Q}_h can be broken down into its rigid body and elastic component and written as

$$\mathbf{Q}_{h_{i_r}} = \begin{Bmatrix} 0 \\ 0 \\ \tau_i - \tau_{i+1} \end{Bmatrix} \quad \mathbf{Q}_{h_{i_f}} = \begin{Bmatrix} \tau_i \\ 0 \\ \vdots \\ 0 \\ -\tau_{i+1} \\ 0 \end{Bmatrix} \quad (3.24)$$

where the first and second to last flexible DOFs of the i^{th} body correspond to the end-most bending rotation DOFs. Again, this is the example case of the double pendulum, where

there is an assumption that the hinge moment acts on the elastic axis of the beams. For an arbitrary case of an off axis moment, additional terms would need to be added to create an equivalent moment about the elastic axis.

3.3.3 Aerodynamic Forces

Both steady and unsteady aerodynamic forces are applied to the wing using strip theory for incompressible, two-dimensional flows. The forces are found at each of the nodes of the FEM model based upon the two-dimensional aerodynamic properties and state of the node. The force is assumed to be linearly distributed across the span of the wing between two nodes.

The steady aerodynamic forces are found for the initial condition by iteratively calculating the local angle of attack and aerodynamic forces at each node point until a stable solution is found. As the wing has a constant cross-section geometry, the aerodynamic coefficients remain constant along the span. The steady coefficient of lift, and the variation of coefficient of lift with respect to the angle of attack are determined from the experimental data or from XFOIL [29]. The steady coefficient of moment, and the variation of coefficient of moment with respect to angle of attack are found using XFOIL. XFOIL is a two-dimensional flow solver that uses the panel method.

The unsteady aerodynamic forces are configured into a state-space formulation similar to that developed by Edwards et al. [30]. The flap DOF in Edwards' model is removed leaving only the pitch and plunge DOFs. Edwards' model is based upon the work of Theodorsen [31], but uses a Jones approximation of Wagner's indicial loading function. Theodorsen's model has been used to accurately predict the pitch response of a wing in limit cycle oscillation, while over predicting the plunge response [20]. The Jones approximation requires another two DOFs to be added to each node of the model, yielding for a 2-D pitch/plunge model

$$\begin{bmatrix} \mathbf{I} & 0 & 0 \\ 0 & \mathbf{M}' & 0 \\ 0 & 0 & \mathbf{I} \end{bmatrix} \begin{Bmatrix} \dot{\mathbf{x}} \\ \ddot{\mathbf{x}} \\ \dot{\mathbf{x}}_p \end{Bmatrix} = \begin{bmatrix} 0 & \mathbf{I} & 0 \\ -\mathbf{K}' & -\mathbf{B}' & \mathbf{D} \\ \mathbf{E}_1 & \mathbf{E}_2 & \mathbf{F}_p \end{bmatrix} \begin{Bmatrix} \mathbf{x} \\ \dot{\mathbf{x}} \\ \mathbf{x}_p \end{Bmatrix} \quad (3.25)$$

where

$$\mathbf{x} = \begin{Bmatrix} h \\ \alpha \end{Bmatrix} \quad (3.26)$$

The vector \mathbf{x} consists of the pitch and plunge variables (α and h), and represents the difference between the height and angle of attack of the node with respect to the steady state values. It is important to note that h is defined in the standard aeroelastic sense, that is positive downward. The matrices \mathbf{M}' , \mathbf{K}' , and \mathbf{B}' are the sum of both the aerodynamic and structural mass, stiffness, and damping matrices respectively. The augmenting vector \mathbf{x}_p and its corresponding matrices \mathbf{E}_1 , \mathbf{E}_2 and \mathbf{F}_p provide the approximation to Wagner's indicial loading function.

The aerodynamic lift is assumed to always act in a perpendicular direction to the surface of the wing. When put into the multi-body model, it is necessary to use the position and velocity of the wing in the inertial frame to determine the aerodynamic forces. Therefore, both the rigid and elastic DOF need to be taken into account. However, since there is no aerodynamic plunge stiffness term in Edwards' model, only the plunge velocity term needs to be determined,

$$\dot{h}_{\dot{y}_1} = \begin{bmatrix} -\sin \theta_i & \cos \theta_i \end{bmatrix} \left(\begin{Bmatrix} -\dot{X}_i \\ -\dot{Y}_i \end{Bmatrix} - x_{\dot{y}_1} \dot{\theta}_i \begin{Bmatrix} -\sin \theta_i \\ \cos \theta_i \end{Bmatrix} \right) - \dot{y}_{\dot{y}_1} \quad (3.27)$$

where $\dot{h}_{\dot{y}_1}$ is the plunge velocity normal to the wing of the first node positive downward, and $\dot{y}_{\dot{y}_1}$ is the elastic DOF of the first mode associated with out-of-plane translation, positive upward (e_1 in Figure 3-3). The angle of attack α of the first node is simply the beam torsional rotation (e_2 in Figure 3-3). Due to assumed linearity of the distributed aerodynamic forces, and only using the states of the element nodes, the DOFs associated with beam bending rotation have no influence. They are also not directly affected by the aerodynamic loads.

Knowing the pitch and plunge of a node, the total two-dimensional aerodynamic loads

(lift, L , and moment, M) at that node can be calculated using Equation 3.25

$$\begin{Bmatrix} L_{\dot{y}_1} \\ M_{\dot{y}_1} \end{Bmatrix} = \begin{Bmatrix} L_{s_{\dot{y}_1}} \\ M_{s_{\dot{y}_1}} \end{Bmatrix} - \mathbf{K}'_{\dot{y}_1} \begin{Bmatrix} 0 \\ \alpha_{\dot{y}_1} - \alpha_{0_{\dot{y}_1}} \end{Bmatrix} - \mathbf{B}'_{\dot{y}_1} \begin{Bmatrix} \dot{h}_{\dot{y}_1} \\ \dot{\alpha}_{\dot{y}_1} \end{Bmatrix} + \mathbf{D}_{\dot{y}_1} \mathbf{x}_{p_{\dot{y}_1}} \quad (3.28)$$

where L_s and M_s are the static aerodynamic loads and α_0 is the steady state angle of attack.

A simplified stall model is used to account for nonlinear lift characteristics at high angles of attack. For apparent angles of attack $\alpha_a = \alpha - \dot{h}/U$ (U is the free stream air velocity) greater than some α_{stall} , or steady state stall angle, the lift terms due to α and \dot{h} are set to the value of lift at α_{stall} . In other words, the lift slope is a piecewise function with a constant slope of 2π for $\alpha_a < \alpha_{stall}$ and zero slope for $\alpha_a > \alpha_{stall}$.

To integrate the aerodynamic loads into the multi-body model, a free forcing vector \mathbf{Q}_a is created. The magnitude of the load is equated to the two-dimensional load found in Equation 3.28 at each node and linearly distributed across the element between the nodes. Therefore, for the j^{th} element of the i^{th} body

$$\mathbf{Q}_{a_{r_{\dot{y}}}} = \begin{Bmatrix} \frac{l_{\dot{y}}(L_{\dot{y}_1} + L_{\dot{y}_2})}{2} \begin{Bmatrix} -\sin \theta_i \\ \cos \theta_i \end{Bmatrix} \\ \frac{1}{l_{\dot{y}}} [(L_{\dot{y}_2} - L_{\dot{y}_1})(x_{\dot{y}_2}^3 - x_{\dot{y}_1}^3)/3 + (L_{\dot{y}_1}x_{\dot{y}_2} - L_{\dot{y}_2}x_{\dot{y}_1})(x_{\dot{y}_2}^2 - x_{\dot{y}_1}^2)/2] \end{Bmatrix} \quad (3.29)$$

$$\mathbf{Q}_{a_{r_{\dot{y}}}} = l_{\dot{y}} \begin{bmatrix} \frac{1}{3} & 0 & \frac{1}{6} & 0 \\ 0 & \frac{1}{3} & 0 & \frac{1}{6} \\ 0 & 0 & 0 & 0 \\ \frac{1}{6} & 0 & \frac{1}{3} & 0 \\ 0 & \frac{1}{6} & 0 & \frac{1}{3} \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} L_{\dot{y}_1} \\ M_{\dot{y}_1} \\ L_{\dot{y}_2} \\ M_{\dot{y}_2} \end{Bmatrix} \quad (3.30)$$

$$\mathbf{Q}_{a_{p\ddot{y}}} = \mathbf{E}_{1\ddot{y}} \begin{Bmatrix} 0 \\ \alpha_{\ddot{y}1} - \alpha_{0\ddot{y}1} \\ 0 \\ \alpha_{\ddot{y}2} - \alpha_{0\ddot{y}2} \end{Bmatrix} - \mathbf{E}_{2\ddot{y}} \begin{Bmatrix} \dot{h}_{\ddot{y}1} \\ \dot{\alpha}_{\ddot{y}1} \\ \dot{h}_{\ddot{y}2} \\ \dot{\alpha}_{\ddot{y}2} \end{Bmatrix} + \mathbf{F}_{p\ddot{y}} \mathbf{x}_{p\ddot{y}} \quad (3.31)$$

Incorporating the applied loads into the elastic multi-body model, Equation 3.18 gives:

$$\begin{bmatrix} \mathbf{M}'_{rr} & \mathbf{M}'_{rf} & \mathbf{C}'_{qr} & 0 \\ \mathbf{M}'_{fr} & \mathbf{M}'_{ff} & \mathbf{C}'_{qf} & 0 \\ \mathbf{C}'_{qr} & \mathbf{C}'_{qf} & 0 & 0 \\ 0 & 0 & 0 & \mathbf{I} \end{bmatrix} \begin{Bmatrix} \ddot{\mathbf{q}}_r \\ \ddot{\mathbf{q}}_f \\ \lambda \\ \dot{\mathbf{x}}_p \end{Bmatrix} = \begin{Bmatrix} \mathbf{Q}_{v_r} + \mathbf{Q}_{g_r} + \mathbf{Q}_{h_r} + \mathbf{Q}_{a_r} \\ \mathbf{Q}_{v_f} + \mathbf{Q}_{g_f} + \mathbf{Q}_{h_f} - \mathbf{K}_{ff}\mathbf{q}_f + \mathbf{Q}_{a_f} \\ -\mathbf{Q}_c - 2\Omega\dot{\mathbf{C}} - \Omega^2\mathbf{C} \\ \mathbf{Q}_{a_p} \end{Bmatrix} \quad (3.32)$$

where r , f , and p refer to rigid body, elastic, and supplemental aerodynamic DOFs, respectively. Notice that the mass matrices are changed to include the aerodynamic inertial terms. This is done by treating the aerodynamic inertial terms as virtual mass and adding them to the true mass matrix. The total number of DOFs in the model is dependent on the number of segments and the number of elements per segment. Each segment has three rigid body DOFs and three flexible DOFs for each element plus an extra flexible DOF for segment root torsion. Additionally, there are two supplemental DOFs for each node, or two times the number of elements plus one per segment. Finally, there are three Lagrange multipliers for each segment plus one for the root hinge angle. Thus, for a typical case of a six-segment wing with two elements per segment, there are 97 DOFs.

3.4 Time Integration Procedure

There has been extensive research to determine an accurate method of integrating non-linear [32] and multi-body [8, 33] systems. While non-linear systems have been solved using the frequency domain [34], the time-domain is used in this case to capture non-periodic behavior. The proposed system is time marched forward using a modified version of MathworksTM ODE45 which is an implementation of the explicit Runge-Kutta (4,5) pair of Dormand and Prince [35]. The method has a variable time-step and error checking. To

ensure that the integration scheme is stable for the non-linearities due to the large angle approximation, the double pendulum problem was solved. The total energy of the system was monitored over time and no significant change was observed.

Conner et al. [16] discuss the importance of finding the exact time for a piecewise linear system changes from one linear region to the other. As suggested, a method developed by Hénon [36] is used to ensure that the change of the system occurs at the beginning of a time step. This method exchanges the piecewise dependent variables (θ_i for the hinged wing) with the independent variable (time). As it is simple to find the required change in the dependent variable to reach the switching point, the value is used for the step size in a single integration step.

This can be demonstrated by considering a simple problem of a vehicle with a positional variable $y(t)$. The vehicle velocity, v , at a given time t would then be $v(t) = \dot{y} = dy/dt$. Assuming a constant velocity, and knowing the state of the system at time t_1 , a normal Euler time integration would be able to find y_2 (y at some later time, t_2), as:

$$y_2 = y_1 + \dot{y}(t_2 - t_1) \quad (3.33)$$

Here y is the dependent variable and t is the independent variable. However, to find what time the vehicle would reach y_2 , the problem becomes:

$$t_2 = t_1 + \frac{(y_2 - y_1)}{\dot{y}} \quad (3.34)$$

Now y is the independent variable and t becomes the dependent variable. Essentially the only difference between the two equations is the division of the system by \dot{y} . This is the fundamental logic behind the Hénon method. A single DOF mass-spring system with mass m , stiffness k and position y is governed by $\ddot{y} = -(k/m)y$. Converting this to Hénon state space form gives:

$$\begin{pmatrix} \frac{dy}{dx} \\ \frac{d^2y}{dt dx} \\ \frac{dt}{dx} \end{pmatrix} = \frac{1}{h} \begin{bmatrix} 0 & 1 & 0 \\ -k/m & 0 & 0 \\ 0 & 0 & 1/t \end{bmatrix} \begin{pmatrix} y \\ \frac{dy}{dt} \\ t \end{pmatrix} \quad (3.35)$$

where x is the independent variable that can be either t for a normal, time based integration step, or y for a state based integration step. h is the Hénon variable and is set to one for a time based integration step. For a state integration step, $h = dy/dt$, and the system is stepped over a Δy . This state integration step allows for the system switching time to be determined in one step with the same accuracy as a normal integration step, as opposed to an iterative method which would require numerous steps to achieve the same accuracy.

3.5 Solution Procedure

The model uses inputs of the wing properties and flow conditions. The model can handle any number of wing segments and any number of elastic beam elements per segment. Additional inputs are: any initial disturbances, the error tolerances, and the time to be integrated.

The following scheme is used to simulate the multi-hinged wing in flight:

1. The non-state dependent information for all of the nodes including inertial, stiffness, and forcing terms is initialized.
2. Using a rigid wing assumption, an approximation of the steady-state hinge angles are calculated from the steady external forcing terms.
3. An iterative algorithm performs a static analysis to determine the steady-state values of the DOFs of the full elastic system taking into account variation of the steady lift values due to alteration of wing twist angles.
4. The initial conditions of the simulation are set using the information determined in step 3, and modified for any user inputs.
5. The initial time step is determined based upon the derivative of the system with respect to time.
6. The system is integrated one step in time.
7. If the error is small relative to the tolerance, the time step is increased

8. Steps 6 and 7 are repeated until:

- (a) Time reaches the maximum to be simulated. If this occurs the simulation stops.
- (b) An integration step has an error above the tolerated value. The system is set to the previous state and the time step is reduced.
- (c) A change occurs in one of the hinge states. The system is returned to the previous time step. The Hénon variable is appropriately changed. The step size necessary to reach the state change is computed. One integration step is executed. The Hénon variable is changed back to $h = 1$. The error is checked.

The model outputs the position and velocity of all the degrees of freedom at each time step. If a constant time step is needed, the model will use the interpolation scheme provided by MathworksTM for the Matlab ordinary differential equations suite to determine the state of the system at the required times while still using its own internal time steps. Appendix E contains the scripts used to implement the model in Matlab 5.3.

A simplified version of the model was created to quickly determine the wing performance for various conditions. It is a linearized model based upon the full model with a two added assumptions. First, the hinges are assumed to have the same stiffness state regardless of the hinge angle (i.e. no bi-linear stiffness). Secondly, the hinge angles are assumed to remain small enough such that the sine and cosine of the hinge angle could be replaced by the angle and unity respectively. With these assumptions the model becomes linear.

A linear model allows the basic response of the wing to small disturbances to be determined very efficiently by observing the eigenvalues of the system. The real part of the eigenvalue of a given mode determines the damping ratio of that mode. The imaginary part of the eigenvalue determines the frequency of the mode.

Chapter 4

Model Validation

In this chapter, the experimental results discussed in Chapter 2 are used to validate the proposed analysis framework. This is then used to numerically simulate potential hinged-wing configurations in flight conditions that could not be realized in the experimental testing.

4.1 Model Setup

The wind tunnel experiments were used to test the validity of the model. To accomplish this, the physical characteristics of the WASP prototype wing were used in the model. The initial state and flight conditions of the model were set to match those recorded for each wind tunnel test.

With the exception of the hinge damping properties, all of the wing characteristics could be determined either from the experimental testing or numerical methods. The hinge damping was assumed to be proportional to the static hinge friction. Therefore, a single tuning factor was used to match the model output to one of the experimental runs, and then held constant for the rest of the simulations.

For each experiment, the steady state condition of the wing was calculated and used for the initial condition of all states except for the tip-most hinge. Each of the experimental tests had variable initial tip hinge angles. This tip hinge angle was set in the model so that the initial position of the wing tip would be the same as that recorded in the experiment.

Unfortunately, the positions of the other hinges, and the velocity of the hinges could not be accurately deduced from the experimental data. So the initial angle was measured when the hinge was near zero rotational speed. In some cases, due to other hinges being deflected in the experiment, the wing tip position could not be matched in the model by only deflecting the tip most hinge. For these cases, the tip hinge angle was set to -90° and the next hinge was deflected.

The model was implemented using the MatlabTM software package. The calculations were performed on DellTM computers with PentiumTM III Processors of various speeds.

For validation runs, the model used two structural elements for each wing segment, for a total of 12 beam finite elements. When the system was marched forward in time, a requirement for a relative and an absolute accuracy of 10^{-5} was enforced at each time step. To confirm the accuracy of such a coarse model, one of the runs was performed with four elements per wing segment and an accuracy of 10^{-6} . Figure 4-2 compares the results of running at a different number of elements per segment. The low order model captures the low frequency motion well, but is less accurate for higher frequencies. However, the time required to simulate runs with the more refined models proved to be prohibitive. The single element model performs the simulation in 8.43×10^3 seconds on a 700 MHz processor, while the two element model takes 2.47×10^4 seconds and the four element model takes 9.87×10^4 seconds (see Figure 4-1).

4.2 Validation Results

Figures 4-3 and 4-4 show the numerical results overlaid with the experimental ones. In these figures, the zero time is set when the experimental data first crosses the steady state value. Both the experimental data and the analytical models originally have tip positions significantly lower than the -2% of semispan shown. This is not shown so that the details of what happens after the steady state values is first reached and can better be seen. The numerical model captures well the trends of the transient response of the experimental results. Both theoretical and experimental results show an increase in frequency and damping with speed and an increase in amplitude with the root angle of attack. However, the numer-

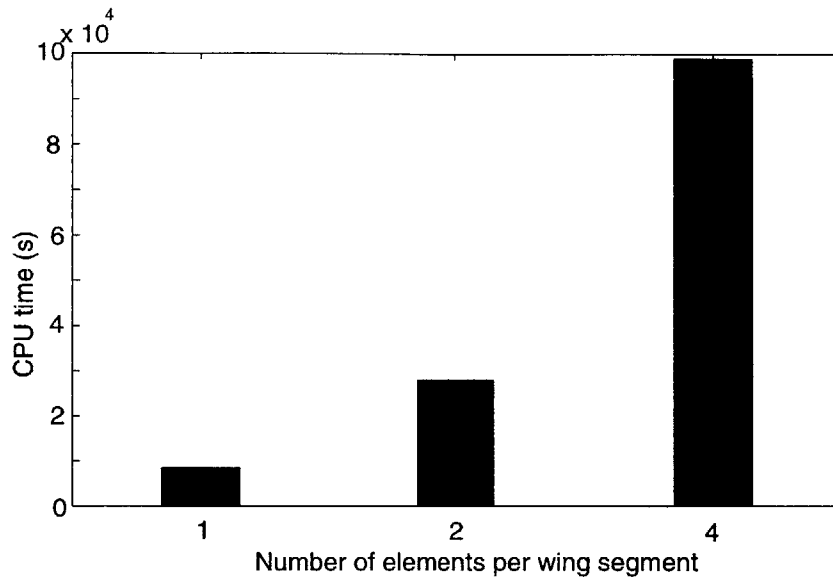


Figure 4-1: CPU times for one, two and four elements per wing segment

ical model does not capture any of the steady state response, nor can it capture the refold phenomenon. If the angle of attack of the model is reduced to -8.5° , it will refold, yet it never enters into the chaotic oscillations seen experimentally. As the root angle of attack increases, the model is better able to capture the wing response. This is due to an increase in the ratio of steady lift to wind-tunnel turbulence effect.

When the wing is near the refold state, minor changes to the load on the wing can cause large changes to the wings equilibrium position. The analytical model is unable to reproduce this effect because it does not account for the turbulence in the wind tunnel flow. Near the refold point, the ratio of noise caused by turbulent tunnel flow to the loads that are captured by the model increases, and the response to the noise dominates the system.

These results validate the accuracy of the analytical model in capturing the transient response of a multi-hinged wing at low speeds. The motion captured by the model includes the response of a hinged wing to a change from one hinge state to the other. The WASP wing experiments provided data over a wide range of speeds and angles of attack, but did not approach the speeds at which aeroelastic instabilities might occur.

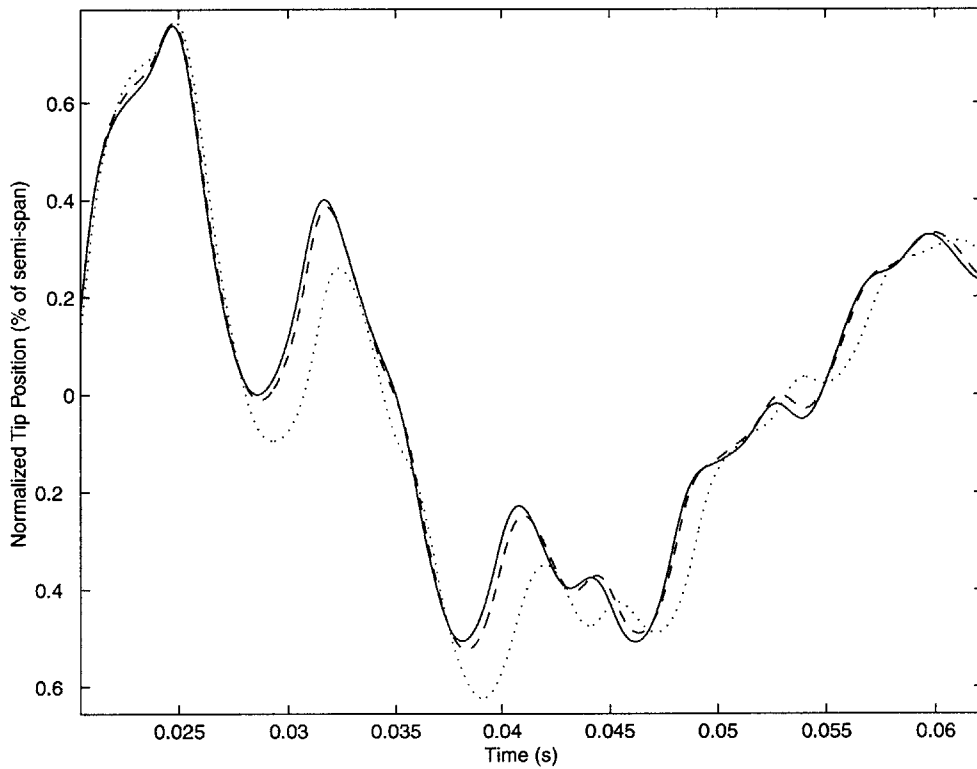


Figure 4-2: Comparison of model with ‘—’ four elements per segment, ‘- - -’ two elements per segment, and ‘.....’ one element per segment

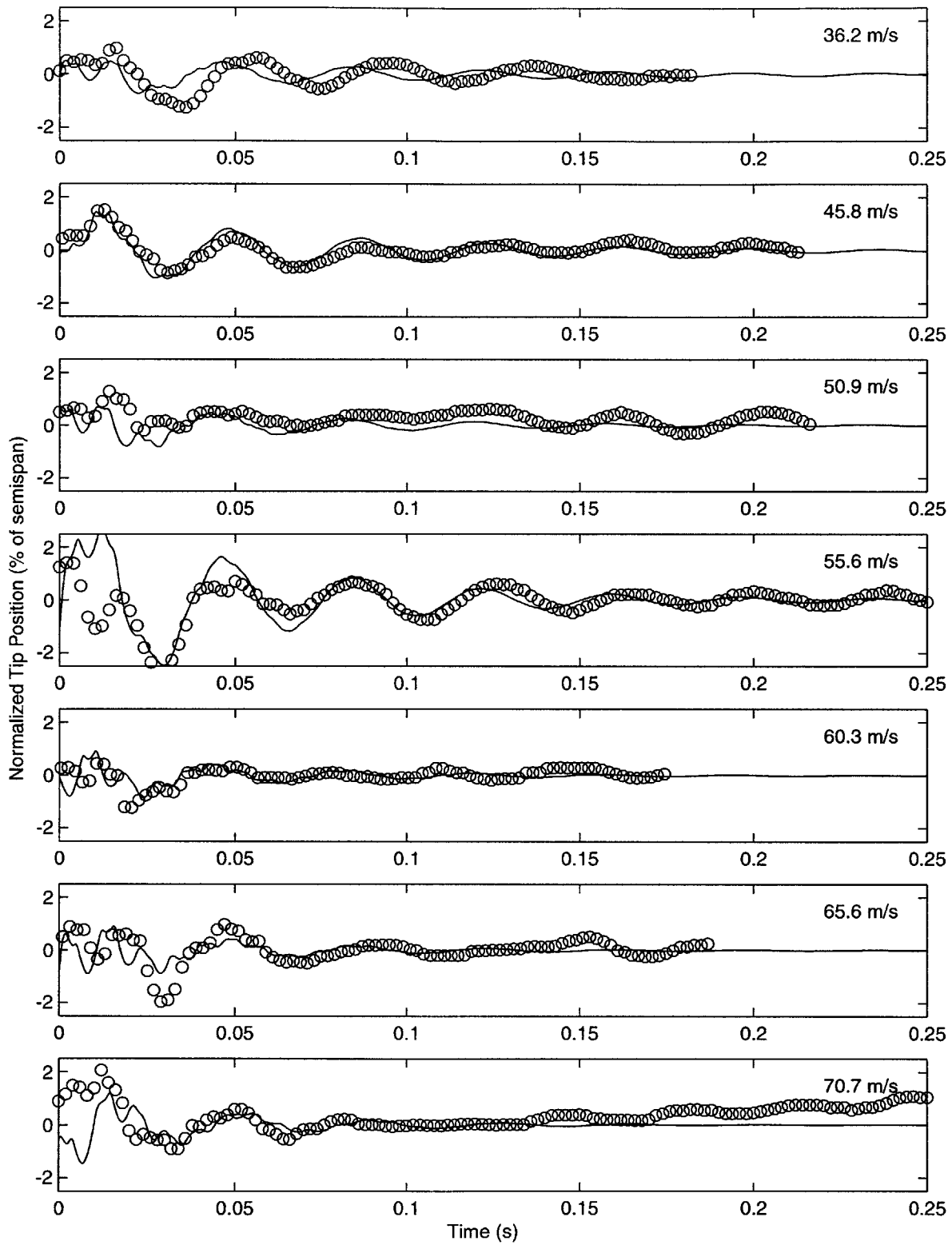


Figure 4-3: Wing tip response at various speeds when the root angle of attack is set to 0° ('—' model, 'o' experimental data)

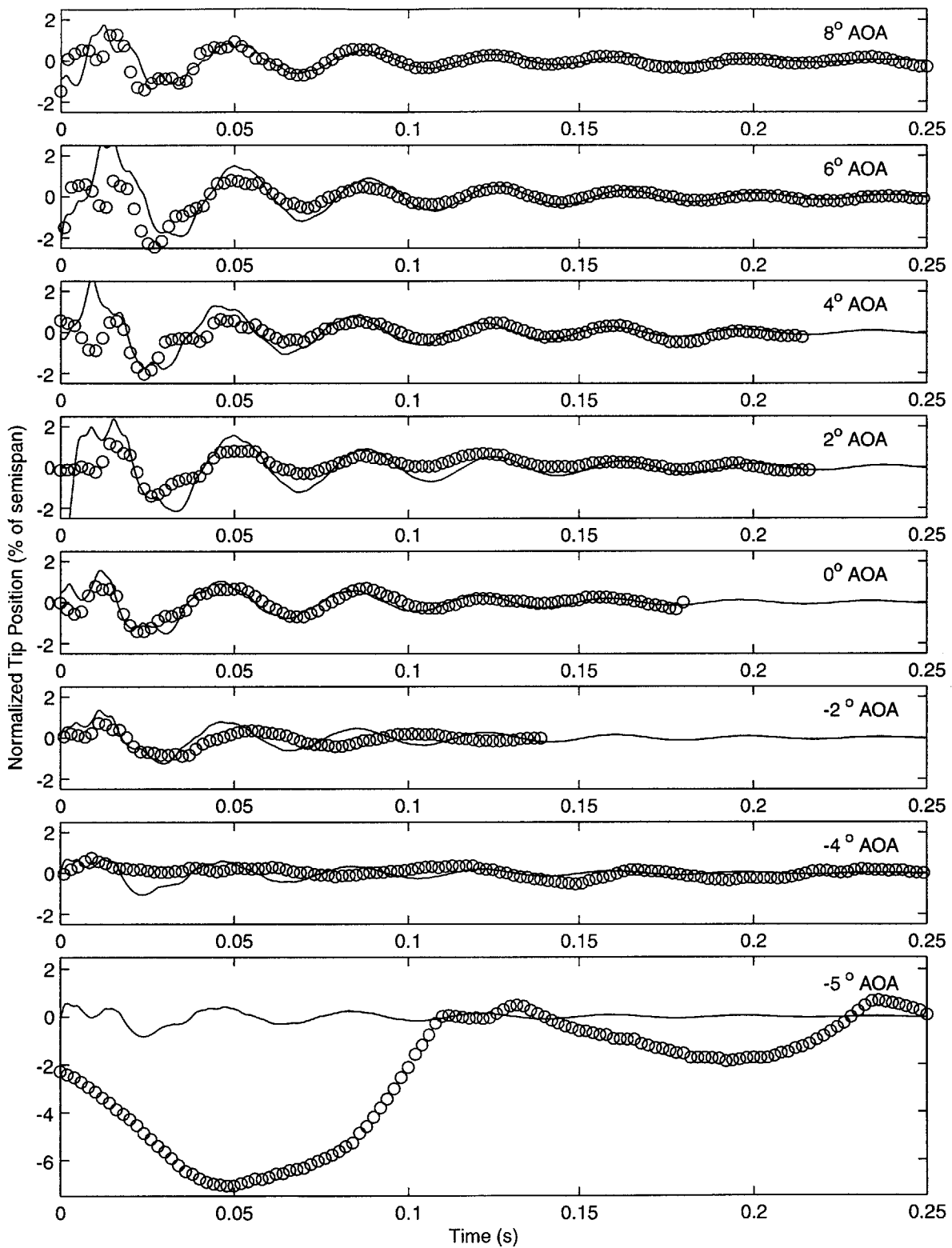


Figure 4-4: Wing tip response at various angles of attack when flying at 46 m/s ('—' model, 'o' experimental data)

Chapter 5

Numerical Studies

In this chapter, the analysis framework developed in Chapter 3 is used to numerically simulate potential hinged-wing configurations at different flight conditions. Using the analytical model allows for parametric studies of hinged wing characteristic to be performed, and phenomena unique to hinged wings to be examined.

5.1 Linear Model Studies

The linear model is used to characterize the aeroelastic performance of multi-hinged wings. This linear version of the model is able to quickly determine the effect of changing physical wing characteristics on the linear response of the system.

5.1.1 Five-Hinged Wing

The eigenvalues of the linearized model are used to determine the characteristics of the WASP wing over a range of flight velocities. The linear model is accurate for only small disturbances, but it can be used to observe some basic trends. This characterization was performed assuming that all the hinges are in the stiff state, as would be the case for a normal flight condition. There are no static aerodynamic or gravitational loads in the linearized model.

Figure 5-1 is a eigenvalue diagram of the first three modes of the WASP wing showing

the variation of system frequency and damping over a range of flight speeds. The frequency and damping values are determined using the eigenvalues of the linear model. The first torsional frequency (1T) decreases as the flight velocity increases. However, the second bending frequency (2B) is not significantly affected by velocity changes. The first bending frequency (1B) does not change much in the lower velocities, but does increase at the higher velocities. There is some negative damping at zero velocity due to hinge friction. The magnitude of the damping of all modes increases with velocity. These trends continue until the first bending and torsional modes begin to interact.

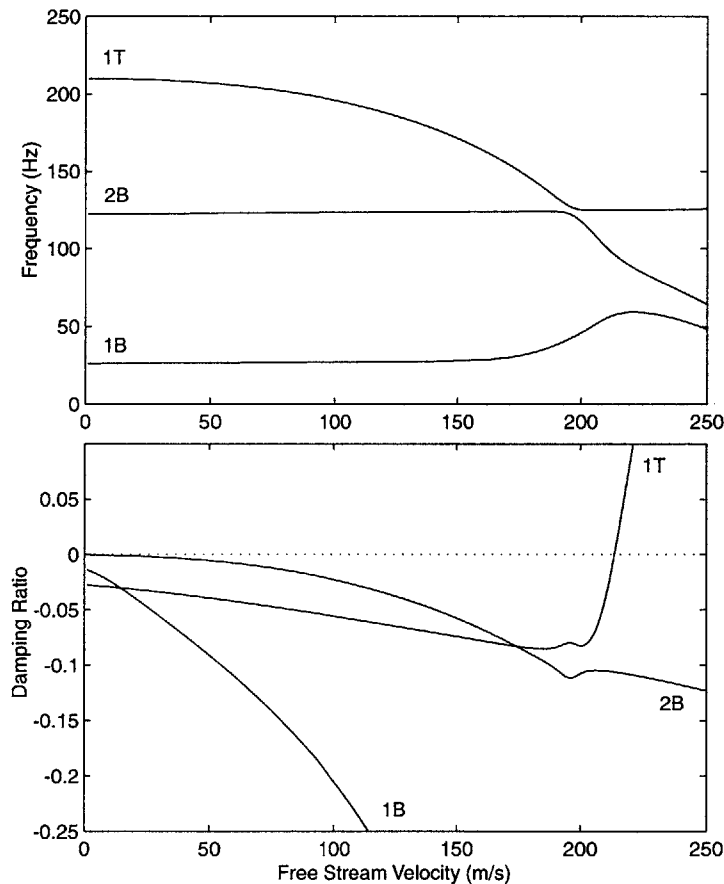


Figure 5-1: Eigenvalue diagram of WASP wing ($U_f/\omega_\alpha b = 5.65$, $k = 0.082$)

Under classical aeroelastic theory [37], a wing flutter occurs when two wing modes coalesce. In this state, the energy of the two modes ‘feed’ each other and the system becomes unstable. Figure 5-1 shows that, for the WASP wing, flutter occurs at 214 m/s, corresponding to a nondimensional value of $U_f/\omega_\alpha b = 5.65$ and a reduced frequency, k , of 0.0819. This

is the speed at which the real part of an eigenvalue becomes positive. A positive real part of an eigenvalue implies that the system has negative damping. While the aerodynamic model is not quantitatively valid at these speeds due to the incompressible flow assumption of the model, it is used to observe trends.

The WASP wing has a higher flutter speed than that of an unhinged wing with the same planform and cross-sectional properties. This can be seen in Figure 5-2, a eigenvalue diagram of the WASP wing with fixed hinges. The unhinged wing response was obtained using the same model, but setting the hinge stiffness six orders of magnitude higher than those found for the WASP wing. The flutter speed of an unhinged WASP-like wing is 194 m/s.

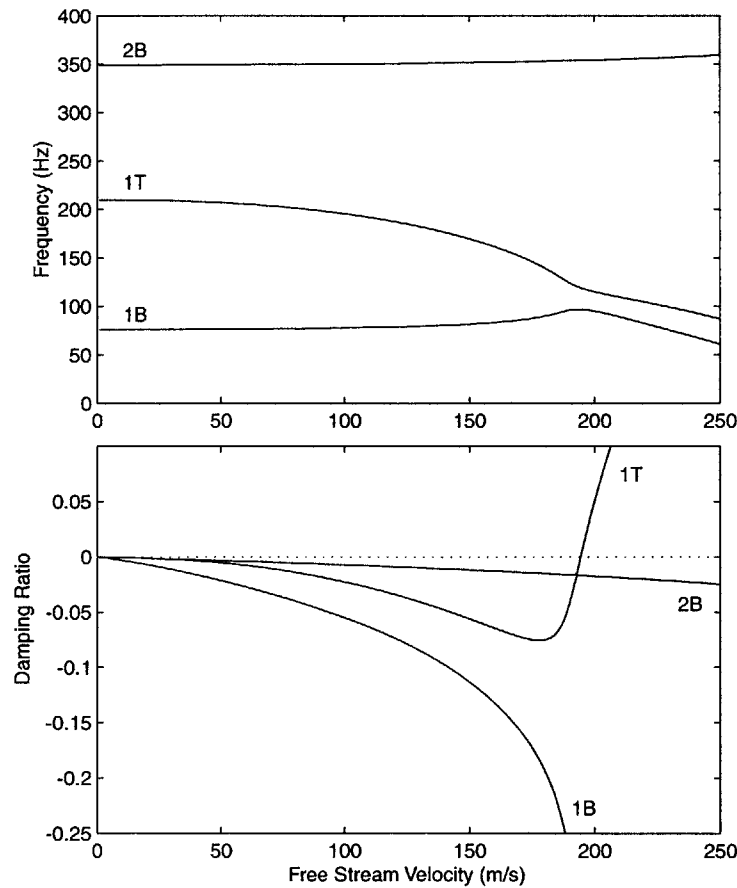


Figure 5-2: Eigenvalue diagram of an unhinged wing with WASP properties ($U_f/\omega_\alpha b = 5.12$, $k = 0.112$)

The model assumes that hinges have a negligible effect on the torsional stiffness of the wing. Reducing the bending frequencies without a similar reduction in torsional frequencies increases the amount of change needed before the frequencies can coalesce as can be seen

comparing Figure 5-1 with Figure 5-2. Therefore, this reduction of bending stiffness is the cause of the increase in flutter speed of the hinged WASP wing compared to the unhinged WASP wing. Furthermore, any hinged wing will tend to enter into the first bending / first torsion instability at a higher speed than the equivalent unhinged wing. Exceptions will occur in systems with a large amount of bend-twist coupling, where the addition of hinges also reduces the torsional frequencies.

5.1.2 One-Hinged Wing

To further understand the effects of adding a hinge to a wing, the characteristics of a one-hinged wing were studied. This wing has the same cross sectional properties of the five-hinged WASP wing. However, this wing has a constant cross section with the same chord as the mid-span of the WASP wing and a single hinge located at the 75% span and a total span of 0.406 meters. The cross sectional properties of this wing can be found in Table A.1 in Appendix A.

The structural natural frequencies of a hinged wing system are highly dependent on the stiffness of the hinges. Figure 5-3 shows the effect of modifying hinge stiffness on the one-hinged wing. Here the first bending (1B), second bending (2B), and first torsion (1T) frequencies are shown as a function of the hinge stiffness. If hinge stiffness value is large in comparison to the stiffness of the wing segments, then the behavior approaches that of an unhinged wing. As the hinge stiffness approaches zero, the system resembles a rigid pendulum. Since the model does not take into account any torsional effect of the hinges themselves, the first torsion frequency stays almost unaltered. However, it is affected when the second bending mode is near the same frequency. Due to the bend-twist coupling of the WASP wing, those two modes become indistinguishable (except in their phasing) at a hinge stiffness of about 67 Nm. This can be seen in Figure 5-4, which shows the shapes of the first three modes at various hinge stiffnesses. The amplitude of the modes are normalized with respect to the largest amplitude (pitch or plunge) and the pitch mode is multiplied by chord to allow for comparison.

Figure 5-1 shows that for the WASP wing, the second bending mode and first torsional

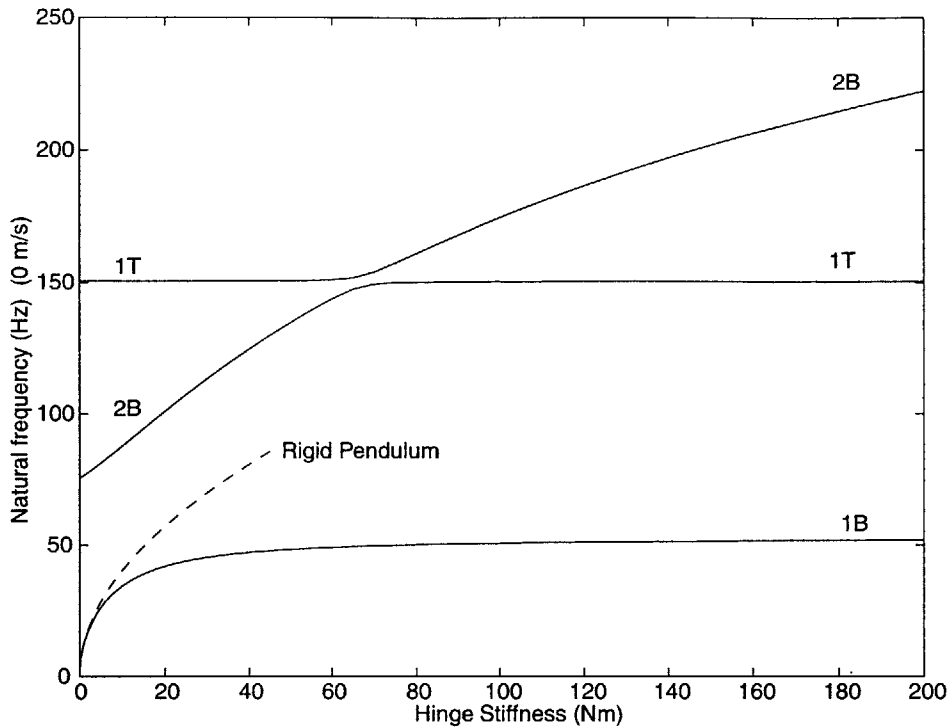


Figure 5-3: Natural frequencies of a one-hinged wing

mode cross each other. For a one-hinged wing with a large enough hinge stiffness, a similar condition occurs (Figure 5-5). However, for the one-hinged wing within the same linear model, a region of instability can be found in which the second bending mode interacts with the first torsional mode. This instability will occur at a lower speed than the traditional flutter caused by the first bending mode interacting with the first torsional mode. Figure 5-6 shows the eigenvalue diagram of such an occurrence. This figure illustrates that for speeds significantly below the onset of first bending/first torsion flutter, the system presents a region of instability.

Instead of a large increase in the real part of the eigenvalue of one of the modes (as in the case of a first bending/first torsion flutter) the second bending/first torsion coalescence causes a “hump” in the eigenvalue of that mode on the eigenvalue plot. If the hump has a high enough peak, then the real part of an eigenvalue is positive and the system becomes unstable. The width and height of this hump are dependent on the wing structural frequencies and hinge damping. A decrease in either the hinge damping or the second bending frequency causes an increase in the hump height. There can be a region of stability between the second

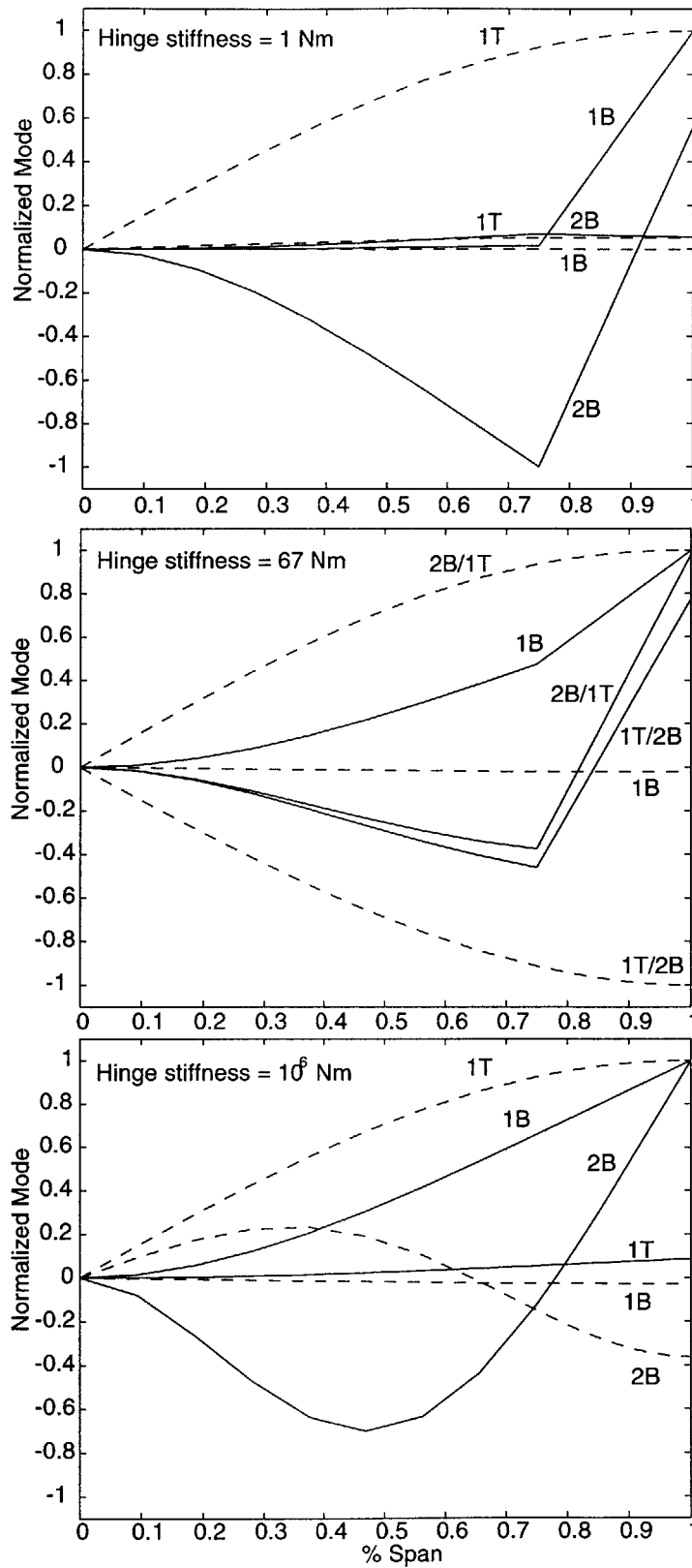


Figure 5-4: Normalized mode shape components of one-hinged wing, solid lines for plunge component, and dashed for pitch angle component.

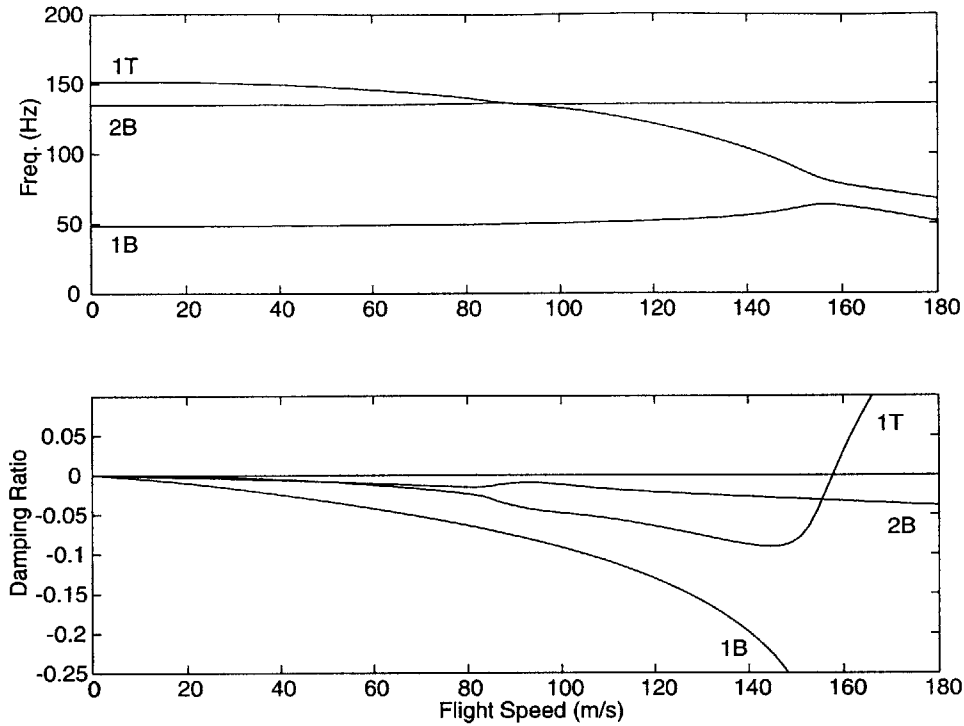


Figure 5-5: Characteristics of a one-hinged wing with no second bending flutter instability, hinge stiffness of 50 Nm ($U_f/\omega_\alpha b = 5.09$, $k = 0.103$).

bending and first bending instabilities (as in the case of Figure 5-6) or the two can blend into each other (as seen in Figure 5-7).

Figure 5-8 shows a map the instabilities of the one-hinged wing. As the hinge stiffness is reduced, the ratio of the second bending and first torsion frequencies is lowered. This map shows the second bending/first torsion (2B/1T) instability as a pocket. The size, shape, and location of this pocket is highly dependent on a number of wing parameters such as hinge location, unhinged bending to torsional stiffness ratio and others. Sometimes the pocket resides wholly inside the first bending/first torsion (1B/1T) instability. It can also be observed that the 1B/1T instability has an increasing flutter speed as the hinge stiffness is reduced. This is the same effect that was observed earlier, that hinged wing will always have a 1B/1T flutter speed higher than that of a similar unhinged wing due to the reduction in the first bending frequency.

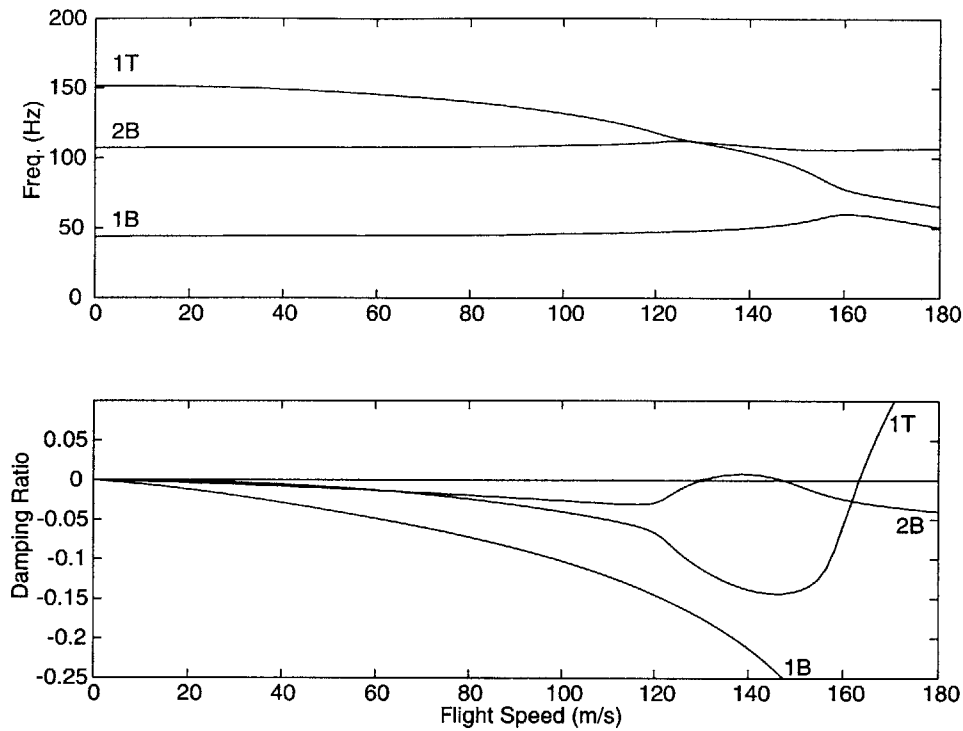


Figure 5-6: Eigenvalue diagram of a one-hinged wing showing second bending and first torsion instability, hinge stiffness of 25 Nm (1st bending $U_f/\omega_\alpha b = 5.25$, $k = 0.094$, 2nd bending $U_f/\omega_\alpha b = 4.19$, $k = 0.176$).

5.2 Nonlinear Studies

Using the full nonlinear model, pre-flutter and post-flutter flight regimes were studied. Below all the flutter speeds, a wing behaves as predicted by the linear model. All disturbances tested are damped out with a decreasing damping ratio as the speed is increased toward flutter speed. This is illustrated in Figure 5-9, which is a tip position trace at 210 m/s using the full WASP model.

5.2.1 Five-Hinged Wing

For a limited range beyond the flutter speed, the WASP wing enters a limit cycle oscillation. The range of speeds at which the wing can maintain a limit cycle oscillation is dependent on the initial energy of the system. The lower the amount of initial energy in the system, the higher the speeds at which the wing will develop a limit cycle oscillation. For these numerical tests the initial energy was adjusted by modifying the amount of static lift applied to the

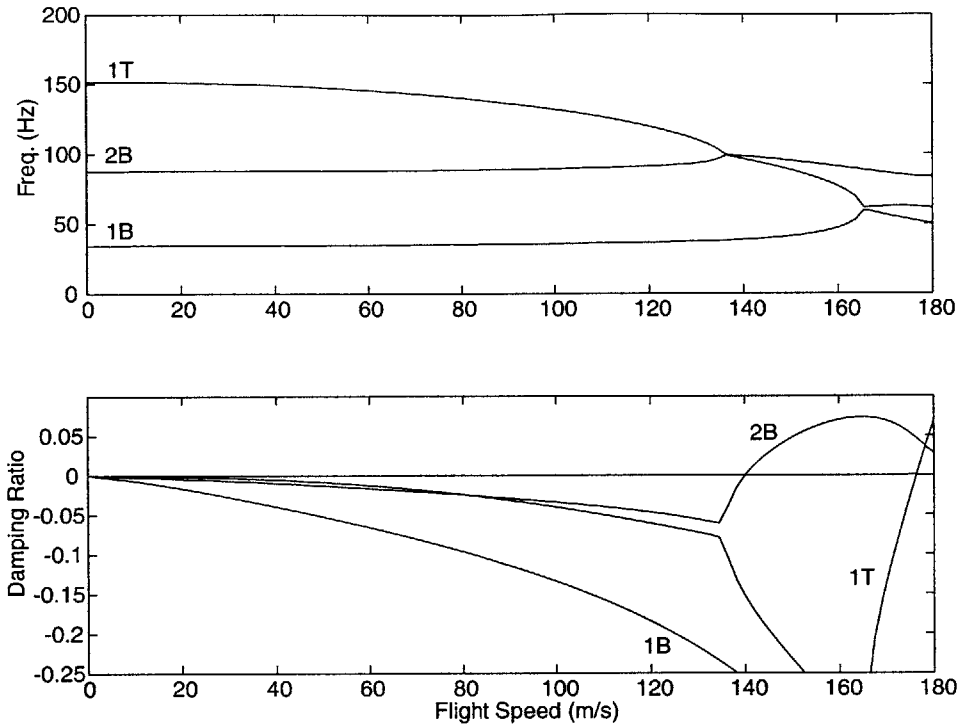


Figure 5-7: Eigenvalue diagram of a one-hinged wing with no post second bending stability, hinge stiffness of 10 Nm (1st bending $U_f/\omega_\alpha b = 5.67$, $k = 0.073$, 2nd bending $U_f/\omega_\alpha b = 4.51$, $k = 0.142$).

wing. The more static lift, the more elastic energy stored in the wing structure.

Figures 5-10 through 5-13 show the WASP wing in the typical limit cycle oscillation. Initially, all hinges are in the stiff state and the wing behaves in a linear manner by oscillating with an exponentially growing amplitude. However, when the amplitude of the pitch oscillation grows to a point where the wing hinges change state, the oscillation becomes highly damped. When the hinge switches from the stiff to the soft state, its frequency of motion is greatly reduced. The wing slowly returns to the initial condition, and the cycle repeats itself. In this oscillation, the tip of the wing is twisted as much as 35° according to the model. Recall that experimental tests show that the wing will stall at approximately 12° angle of attack. Since these runs were conducted without stall effects, these solutions are not actually achievable in the given conditions. However, due to the primarily linear nature of the model, these trends scale with the problem and will be found in a similar wing which does not have large enough oscillation to encounter stall.

The bending stiffness of a hinged wing is dependent on the state of hinges. Thus, in

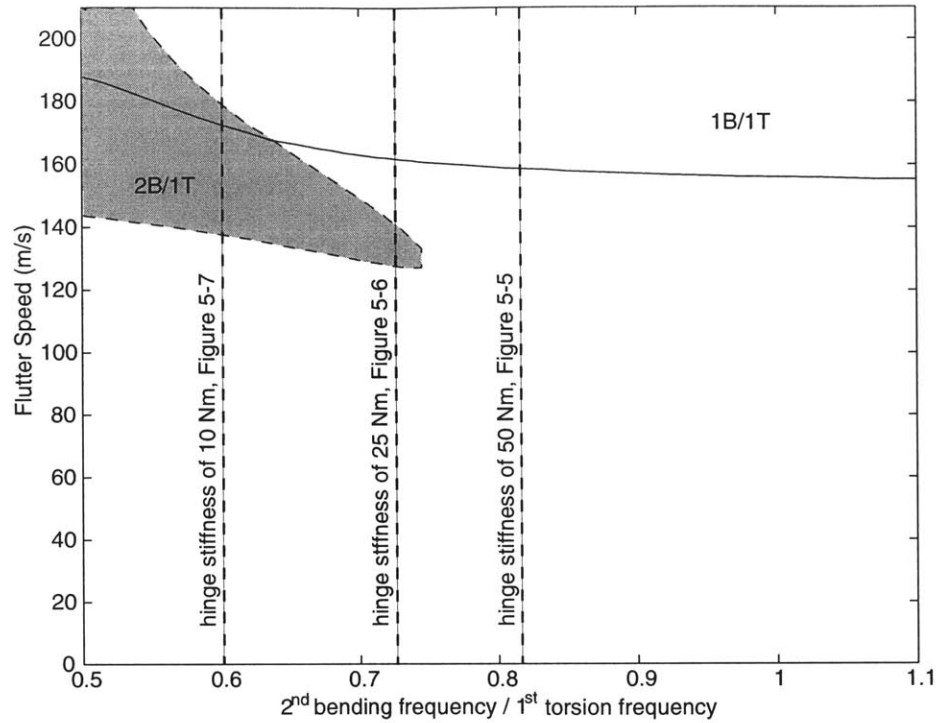


Figure 5-8: Map of the one-hinged wing instabilities

post-flutter flight the wing behaves as predicted by the linear model until the hinges begin to change state. A change in hinge state causes the large change in the bending frequency of the wing. This change in bending frequency interferes with the exchange of energy between the bending and torsional modes and can halt the amplitude growth as shown in Figures 5-10 through 5-13.

At a high enough speed the analytical model predicts a divergent motion as seen in Figures 5-14 and 5-15. As the model does not capture the effect of a hinge rotation enough for a wing segment to close on and impact the neighboring segment, any motion which induces a hinge angle of over 180° is not valid. So, while the response shown in Figures 5-14 and 5-15 is growing beyond bounds, it is interesting to see how, in this case, the model has an oscillatory divergent plunge motion, but a static divergent pitch. At the speed shown, the response of the system is dependent on the initial condition of the system. The initial conditions determine the amount of energy in the system at the start of the simulation. At lower initial energy levels, the system might enter the limit cycle oscillation state similar to that shown earlier, or a chaotic motion such as that shown in Figure 5-16 for higher levels

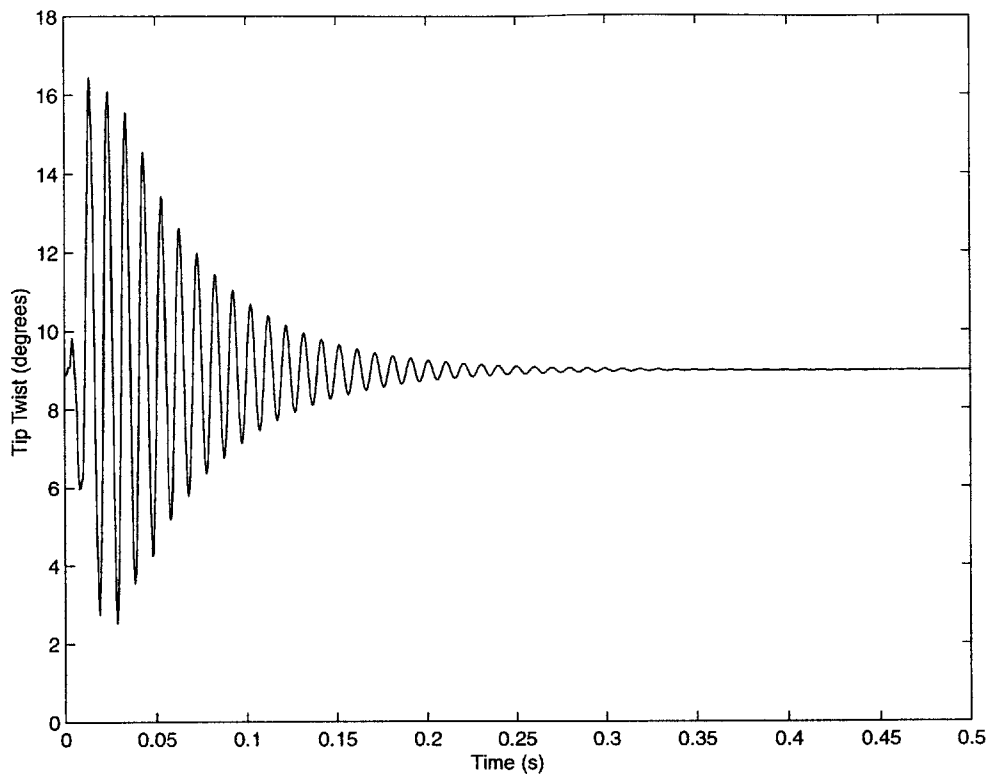


Figure 5-9: Twist response of WASP wing flying at 210 m/s (flutter speed of 214 m/s) with 660 N of static lift

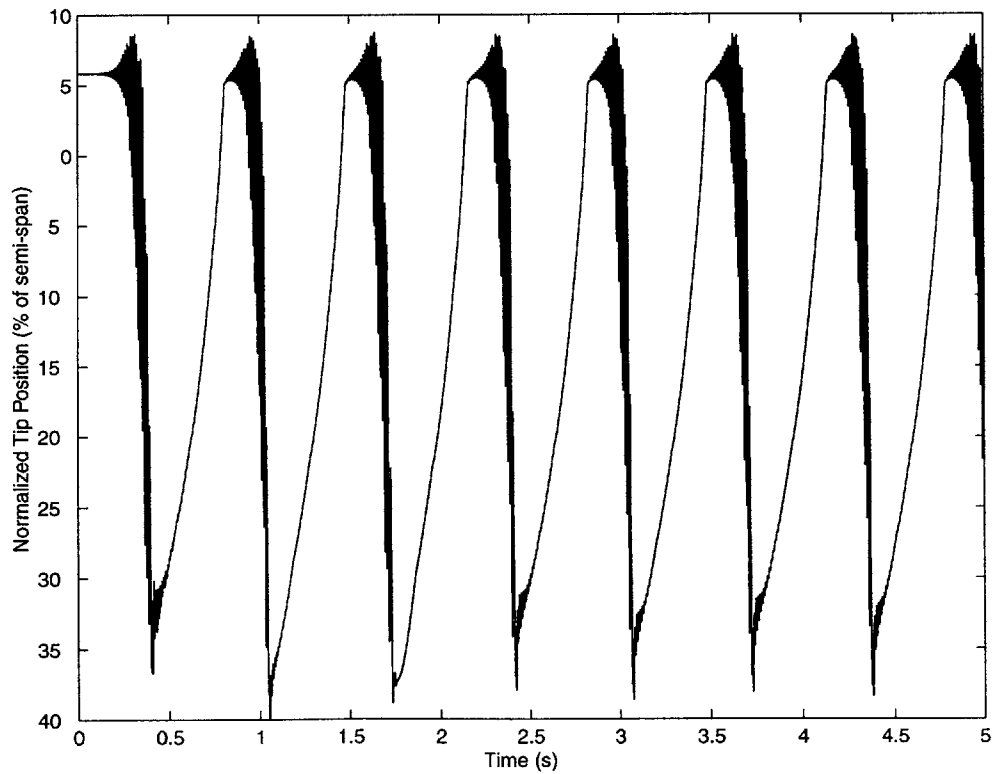


Figure 5-10: Bending response of WASP wing flying at 216 m/s (flutter speed of 214 m/s) with 26 N of static lift

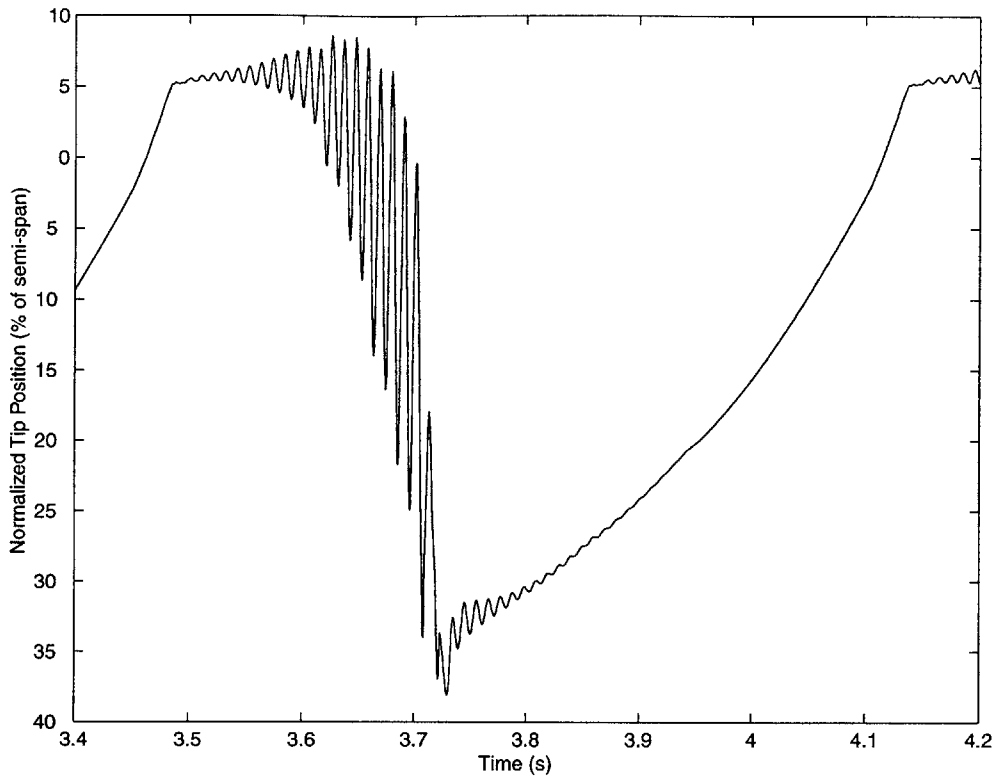


Figure 5-11: Zoom into one of the cycles shown in Figure 5-10

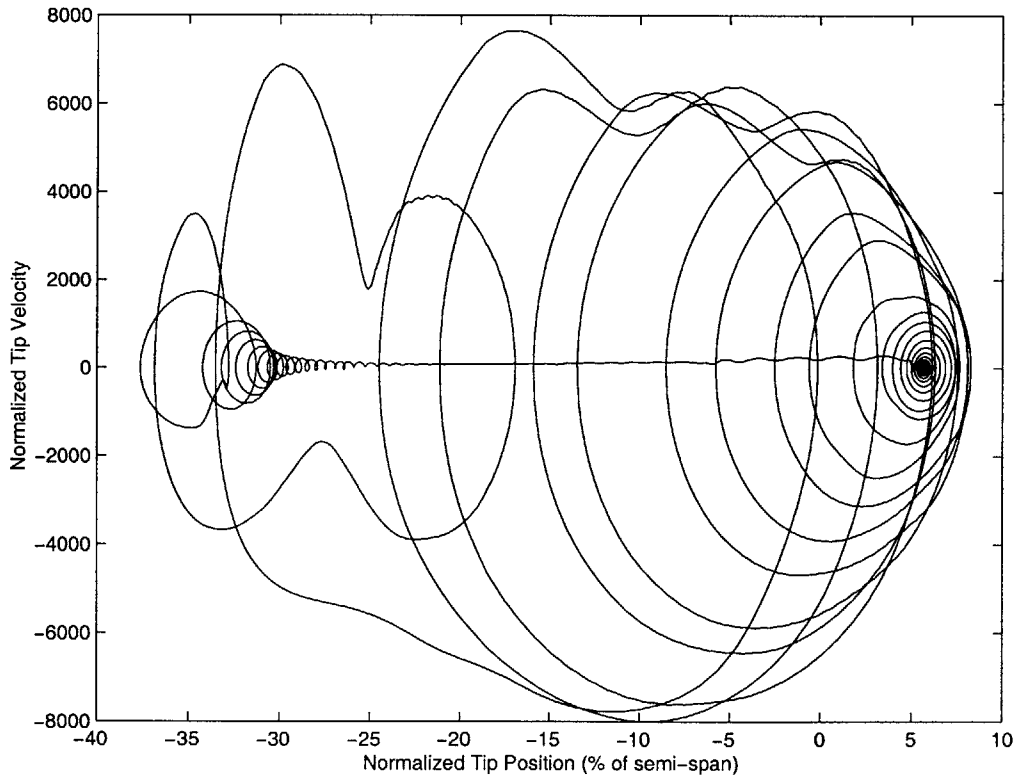


Figure 5-12: Bending phase projection of WASP wing flying at 216 m/s (flutter speed of 214 m/s) with 26 N of static lift

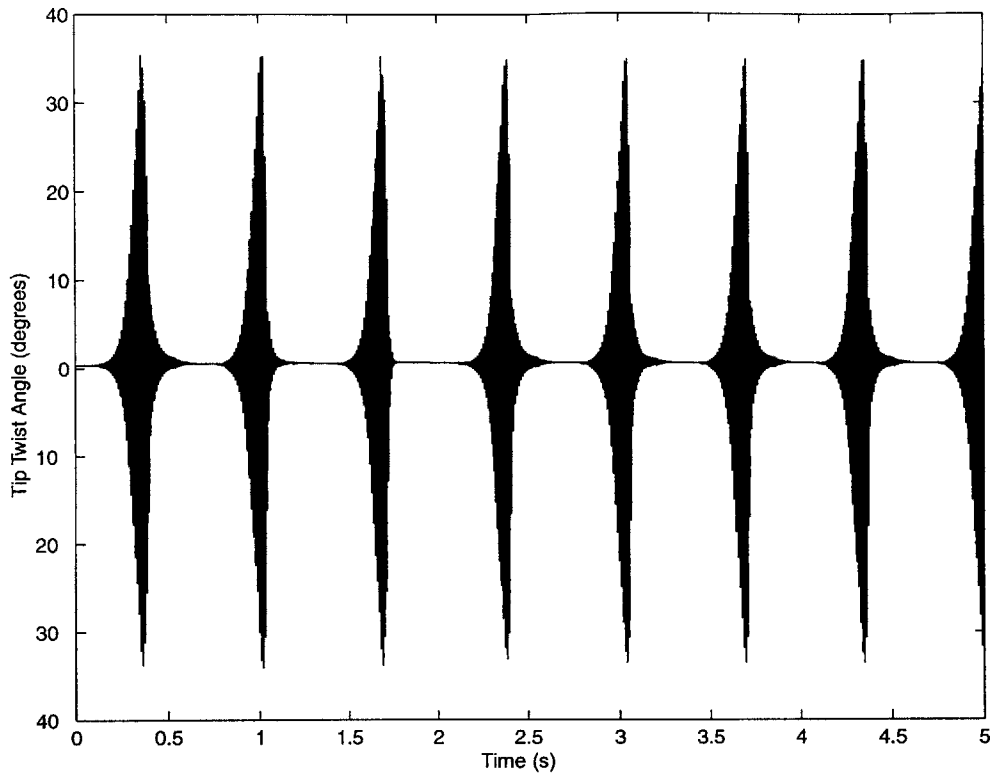


Figure 5-13: Twist response of WASP wing flying at 216 m/s (flutter speed of 214 m/s) with 26 N of static lift

of initial energy. What triggers one response over the other is not known.

5.2.2 One-Hinged Wing

To examine the aeroelastic effect of a hinge without the complexity of the five-hinged wing, the behavior of the one-hinged wing introduced in Section 5.1.2 was examined using the non-linear model. For all cases the hinge stiffness is 22.5 Nm in the stiff state and 0.0001 Nm in the soft hinge state. This soft state stiffness approximates the behavior of a hinge with no spring, and allows for a more numerically stable simulation than a stiffness of zero. The linear behavior of this wing in each state is presented in Figure 5-17. In its stiff state, the wing has a 2B/1T hump instability from 132 m/s to 155 m/s and a 1B/1T instability which starts at 165 m/s. For the soft state, this wing would be statically divergent at all speeds due to the springless hinge. However, it can also be seen that the wing would also have a 2B/1T instability which starts at 145 m/s.

This one-hinged wing can be flown in three different flight regimes. In one flight regime

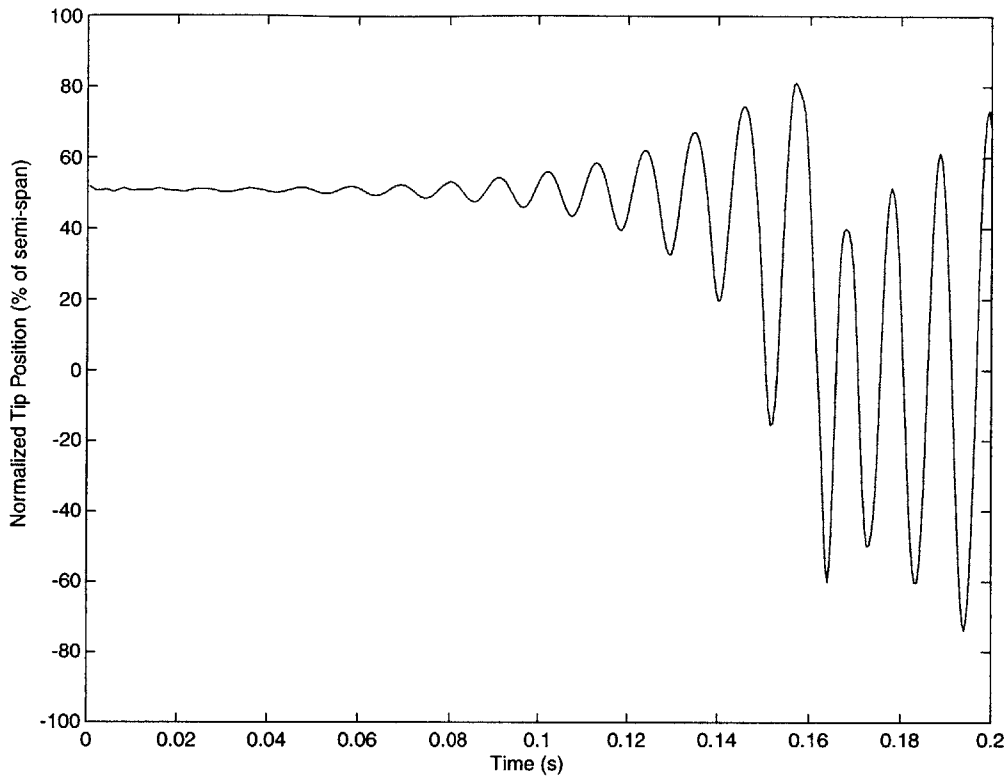


Figure 5-14: Diverging bending response of WASP wing flying at 218 m/s (flutter speed of 214 m/s) with 165 N of static lift

the wing is stable in both states, stiff and soft. In this case any perturbation of the wing will be damped out. Another flight regime occurs when the wing is unstable in both states. Here the wing will always enter an unstable flight condition. Figure 5-18 shows the one-hinged wing entering a limit cycle oscillation at 166 m/s. At this speed the wing has a 1B/1T instability in the stiff state and a 2B/1T instability in the soft state. Like the five-hinged wing described in Section 5.2.1, switching of the hinge states disrupts mode coalescence and prevents a divergent oscillation. Additionally, this response can be either periodic, as shown in Figure 5-18, or non-periodic, as shown in Figure 5-19. The only difference between these simulations is the amount of static energy applied to the system. For speeds well above the onset of instability, the wing will go into a divergent oscillation as shown in Figure 5-20.

Figure 5-21 shows a simulation of flight at 150 m/s. Here the wing has a 2B/1T instability for both hinge states. The wing enters a limited oscillation, similar to the post-flutter behavior of the five-hinged wing, with a large amount of rigid body rotation about the hinge. Thus, while the wing has the same type of instability in both hinge states, the change

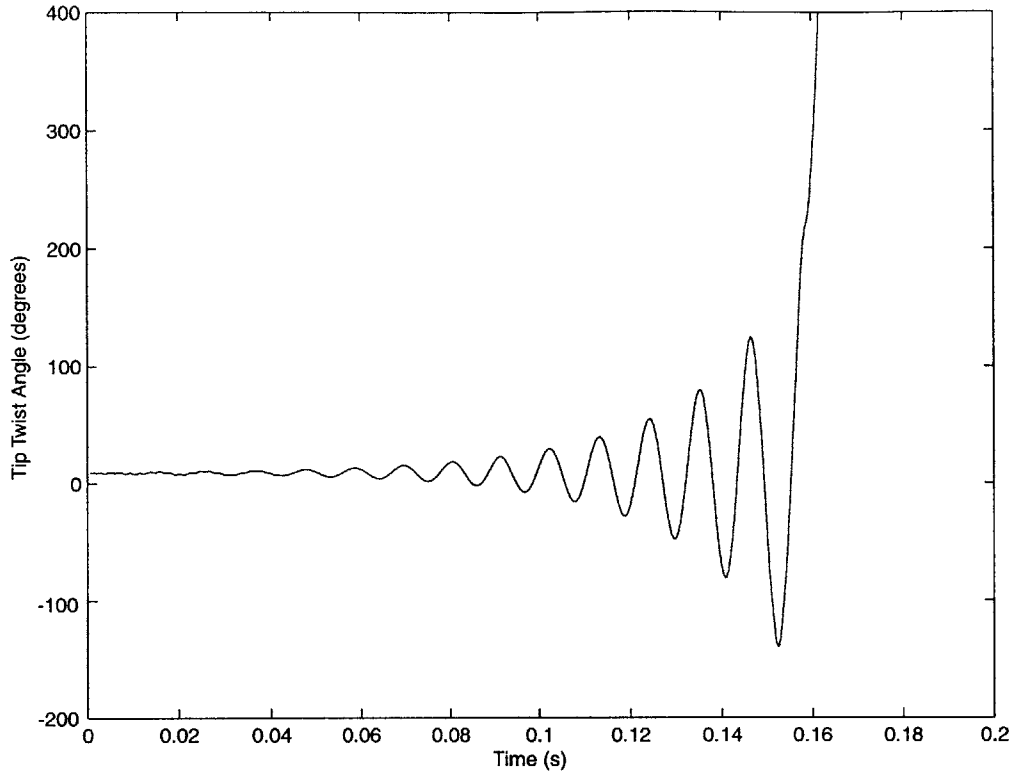


Figure 5-15: Diverging twist response of WASP wing flying at 218 m/s (flutter speed of 214 m/s) with 165 N of static lift

in the bending frequency between the states is enough to stunt the oscillation growth.

The third flight regime exists when the wing is stable in one of the hinge states and unstable in the other. Figure 5-22 shows the tip plunge of the one-hinged wing flying at 135 m/s. In this case the wing has 2B/1T instability in the stiff state, but is stable in the soft state. The wing enters a limit cycle oscillation similar to the one shown in Figure 5-18. The wing starts in an unstable state due to the static lift on the system. When the wing motion grows large enough for the wing hinge to switch to a stable state, the oscillation becomes limited.

For the case where the stiff hinge state is stable and the soft state is not, the wing will nominally exhibit a stable behavior as shown in Figure 5-23. In this case the steady state lift forces return the system to the stable hinge state and all motion is damped out. However, if there is a large enough disturbance, the wing may exhibit a divergent behavior as shown in Figure 5-24. The only difference between Figures 5-23 and 5-24 is between the initial hinge angles, which are -4.7° for the former and -27.5° for the latter. The larger hinge angle

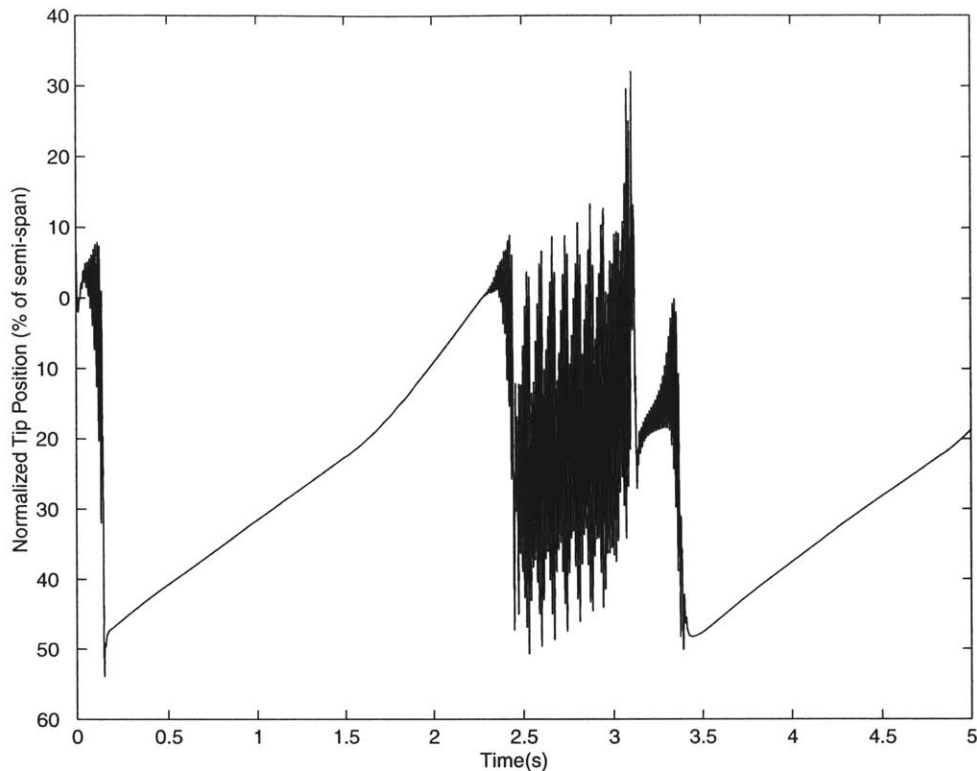


Figure 5-16: Chaotic bending response of WASP wing flying at 218 m/s (flutter speed of 214 m/s) with 6.5 N of static lift

allows the second case a longer time to build up system energy before the hinge switches to the stable state. In this case the extra energy is enough to cause the system to become unstable.

5.2.3 Two-Hinged Wing

A two-hinged wing model was used to further explore the second bending/first torsion hump instability. Like the one-hinged wing, a wing with similar characteristics as the WASP was used. In this case, the same planform as the WASP was used with all but the second and fourth hinges were removed. Figure 5-25 shows the eigenvalue diagram for such a wing with an inner hinge stiffness of 600 Nm and an outer hinge stiffness of 26 Nm.

A nonlinear analysis of the two-hinged model shows that in this region of second bending instability, the system will nominally enter into a limit cycle oscillation, an example of which can be seen in Figs. 5-26 and 5-27. These figures illustrate the dynamics of the first and second hinge of the two-hinged wing. Both hinges change stiffness states at 0° (shown in

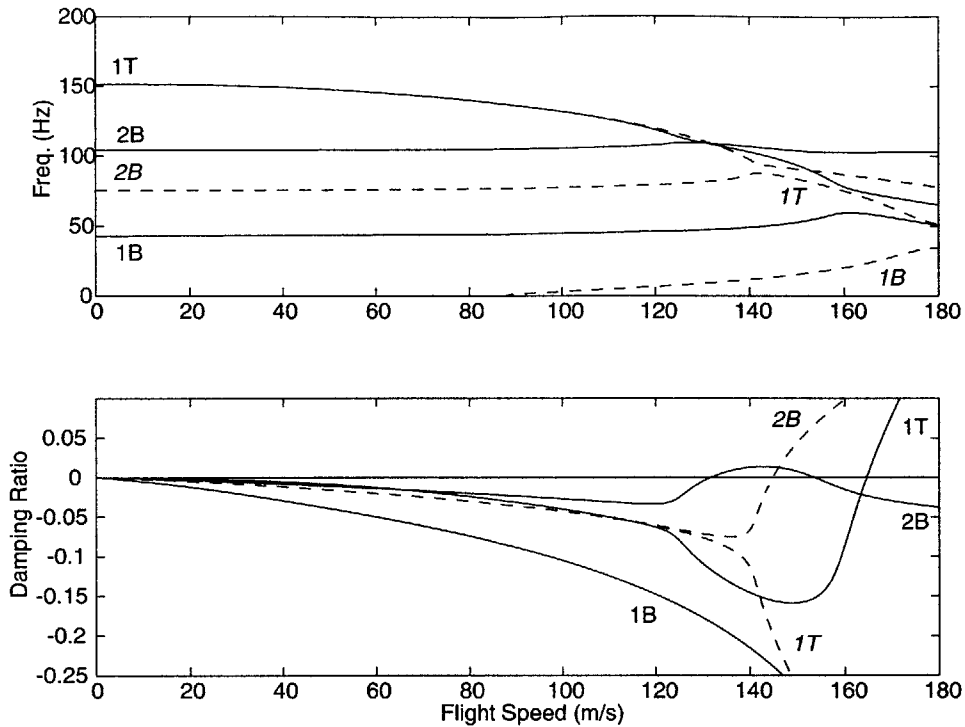


Figure 5-17: Characteristics of a one-hinged wing showing second bending and first torsion instability, ‘—’ stiff state of 22.5 Nm, ‘- -’ soft state of 0.0001 Nm

the figures as a dashed line). The frequency of this response will decrease with an increase in flight speed. The limit cycle amplitude of the oscillation scales with the distance of the hinge equilibrium to the state switching point. In addition, this amplitude is also dependent on the flight speed, but is independent from the initial condition.

For flight speeds between the second and first bending induced instabilities, the response is dependent on the initial conditions of the system. If the system begins with the hinges in the same state that they would be in equilibrium, then the system will have a decaying oscillation as predicted by the linear model. However, if one of the hinges is not in the same state as in equilibrium, then the system goes into a steady state oscillation. Figure 5-28 shows the steady state amplitudes of the system for both types of initial conditions. The figure shows the steady state amplitude of zero between the linear instability region if the hinges start in the same state as equilibrium, showing a decaying oscillation. Unlike the one-hinged wing, the two-hinged wing has divergent 2B/1T oscillations regardless of the initial conditions. This is due to the second hinge further limiting the system instabilities.

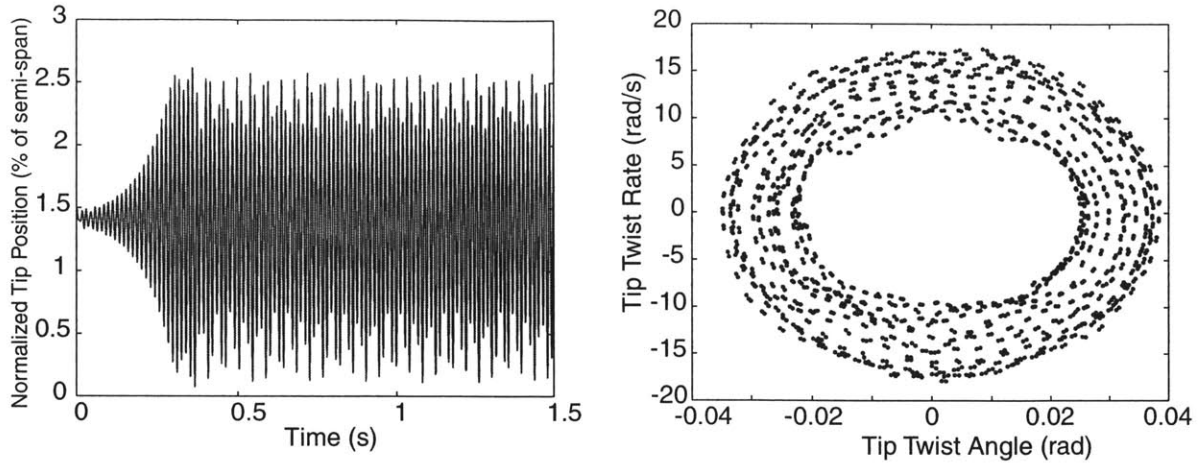


Figure 5-18: Limit cycle oscillation of one-hinged wing flying at 166 m/s (flutter at 165 m/s) with 53 N static lift

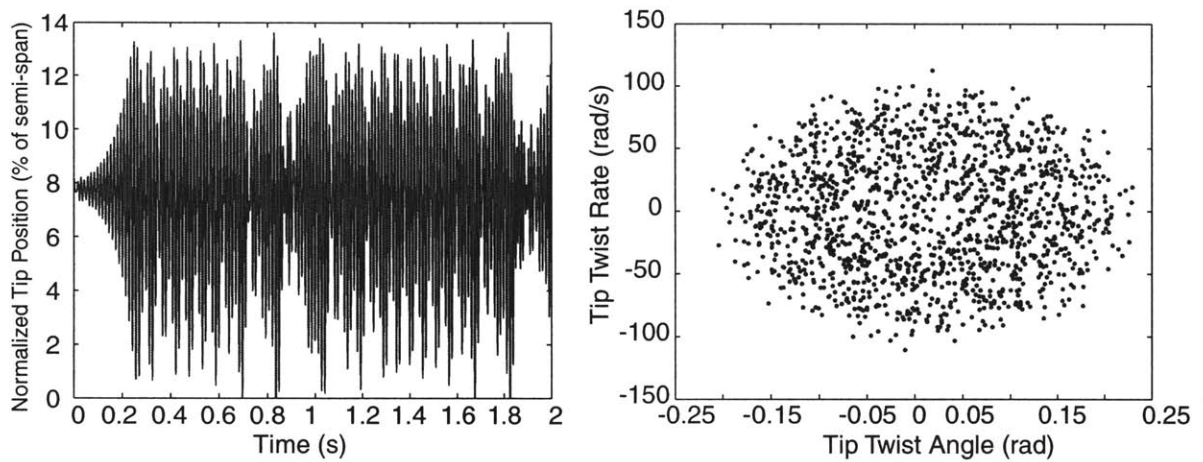


Figure 5-19: Unstable behavior of one-hinged wing flying at 166 m/s (flutter at 165 m/s) with 309 N static lift

For a two-hinged wing that does not have a second bending instability in the linear model, it is still possible for the system to enter into a limit cycle oscillation at speeds below the linear first bending instability. The initial condition of the system is the determining factor of whether the system will enter into a limit cycle oscillation.

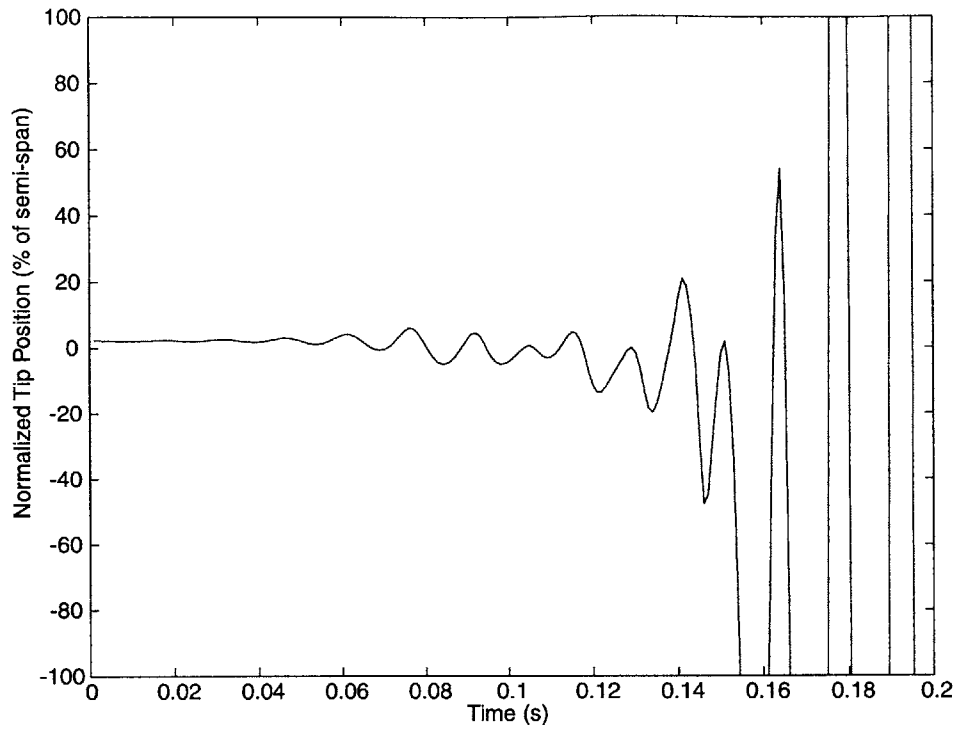


Figure 5-20: Divergent behavior of one-hinged wing flying at 175 m/s (flutter at 165) with 60 N static lift

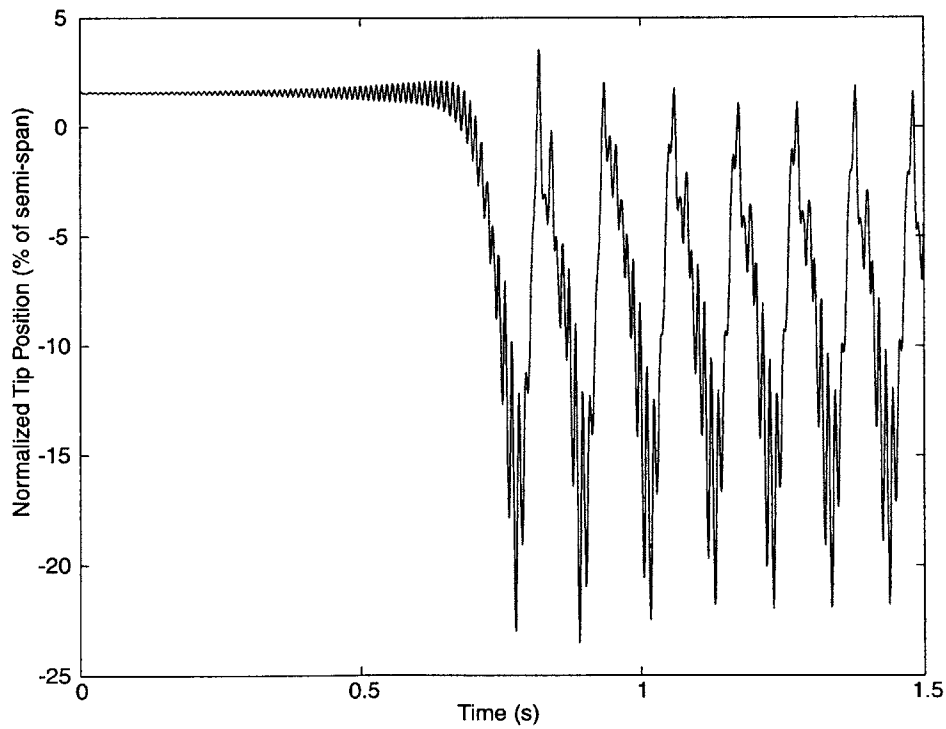


Figure 5-21: Limit cycle oscillation of one-hinged wing flying at 150 m/s (flutter at 165) with 44 N static lift

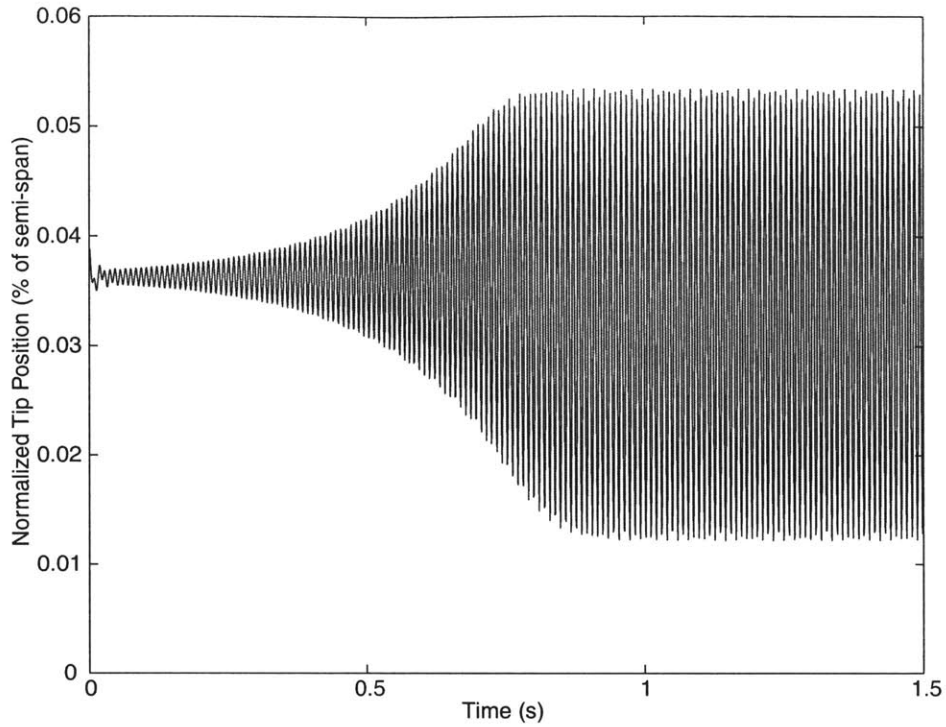


Figure 5-22: Limit cycle oscillation of one-hinged wing flying at 135 m/s (flutter at 165 m/s) with 37 N static lift

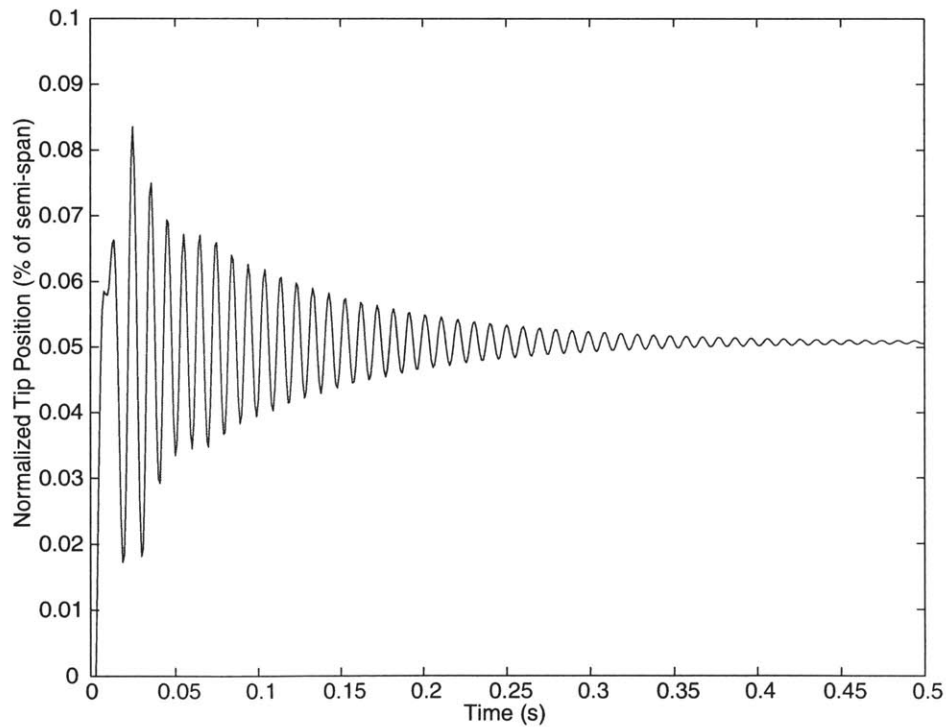


Figure 5-23: Bending response one-hinged wing flying at 160 m/s (flutter at 165 m/s) with 60 N static lift and a -4.7° initial hinge angle

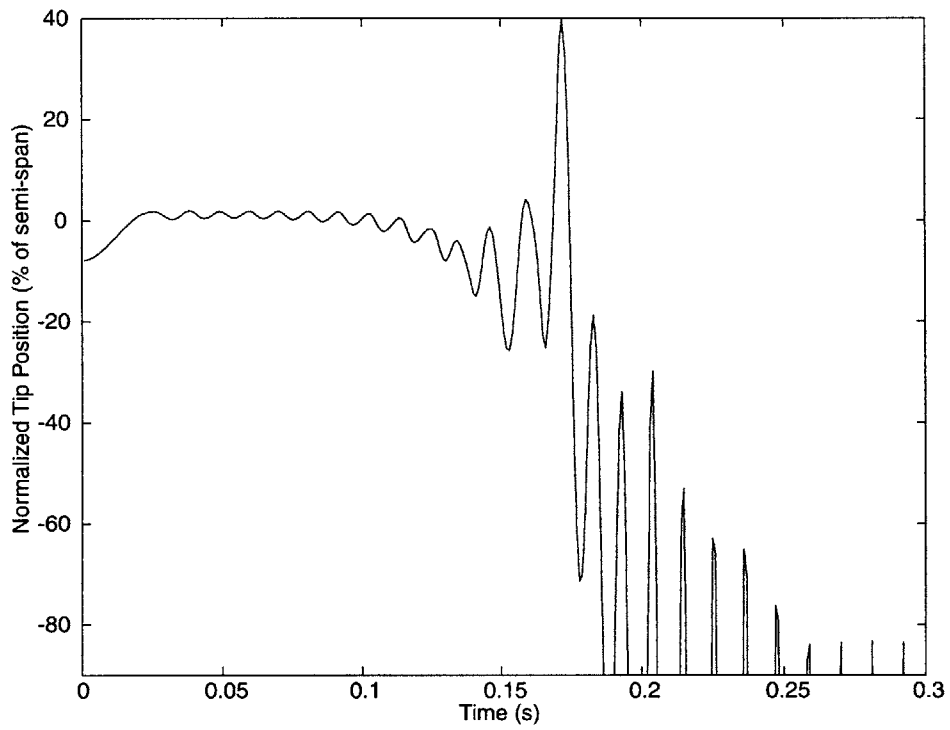


Figure 5-24: Bending response of the one-hinged wing flying at 160 m/s (flutter at 165 m/s) with 60 N static lift and a -27.5° initial hinge angle

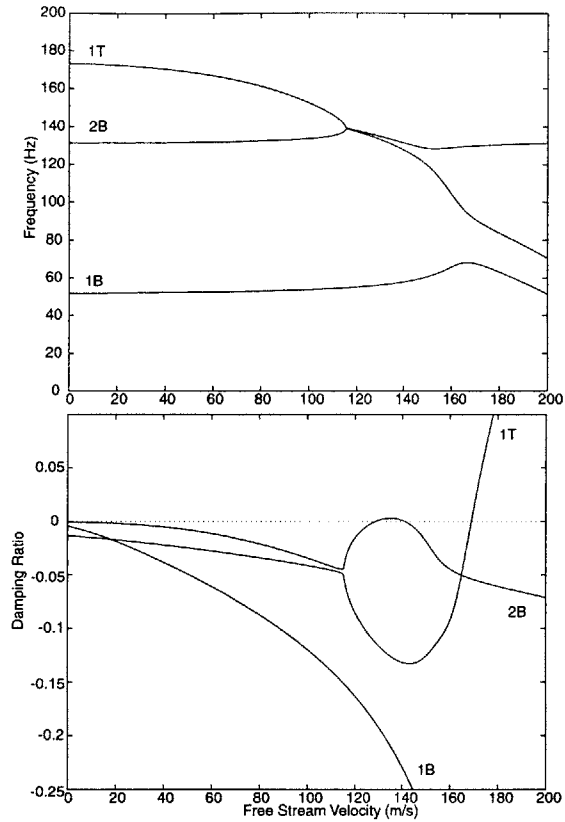


Figure 5-25: Characteristics of a two-hinged wing showing second bending and first torsion instability hinge stiffness of 600 Nm and 26 Nm (1st bending $U_f/\omega_\alpha b = 5.32$, $k = 0.097$, 2nd bending $U_f/\omega_\alpha b = 5.32$, $k = 0.161$).

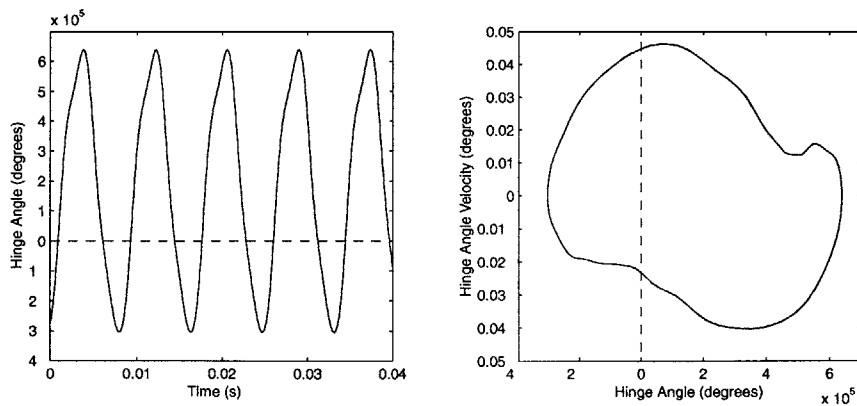


Figure 5-26: Position and phase of root-most hinge response of two-hinged wing at 140 m/s and 0° AOA ($k_1 = 600$ Nm and $k_2 = 26$ Nm)

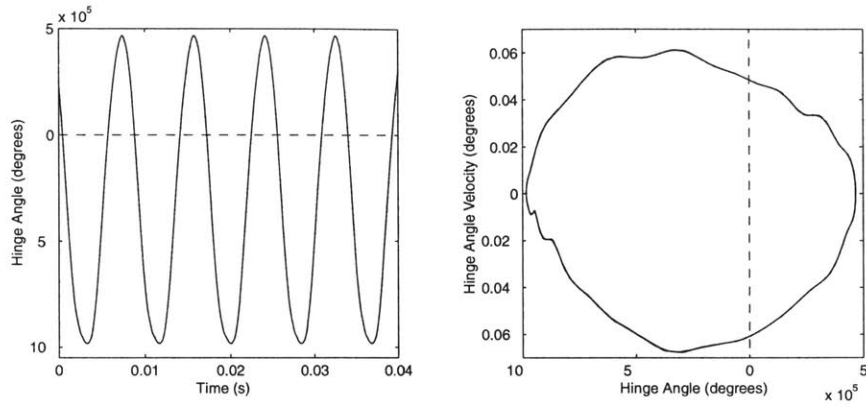


Figure 5-27: Position and phase of tip-most hinge response of two-hinged wing at 140 m/s and 0° AOA ($k_1 = 600$ Nm and $k_2 = 26$ Nm)

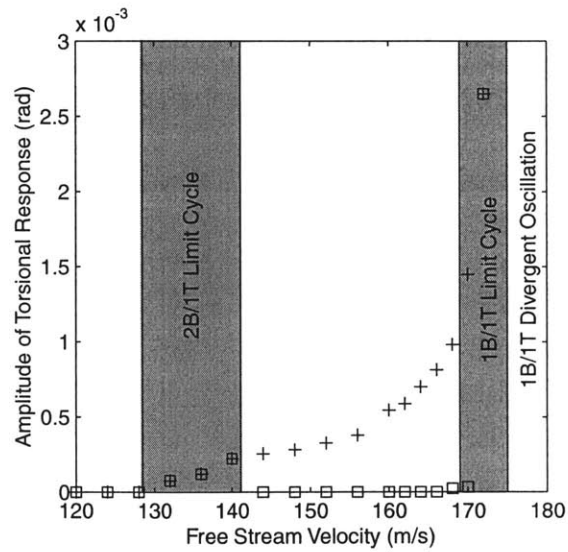


Figure 5-28: Steady-state amplitude of torsional response of two-hinged wing with initial conditions in same state (\square) and different state ($+$) than equilibrium ($k_1 = 600$ Nm and $k_2 = 26$ Nm)

Chapter 6

Experimental Testing of Unstable Flight

To explore regions of unstable flight and further validate the analytical model developed in Chapter 3, a series of experiments were conducted on a simple one-hinged wing. This wing was designed to flutter at low flight speeds so to study the non-linear behavior of a hinged wing.

6.1 Hinged Wing Design

A set of wings was designed and built to examine the behavior of hinged wings at and around the flight speeds which would cause an unhinged wing to flutter. Both an unhinged and a single hinged wing were built, as shown in Figure 6-1. The unhinged wing was created with the same planform and cross-section as the one-hinged wing to provide a baseline. The physical characteristics of the foam wings can be found in Appendix B.

6.1.1 Wing Layouts

To create a wings that would enter into flight conditions favorable for limit cycle oscillations at low flight speeds required tailoring geometric, elastic, and inertial properties.

Each of the foam wings has the same constant cross-section. To simplify the analysis and

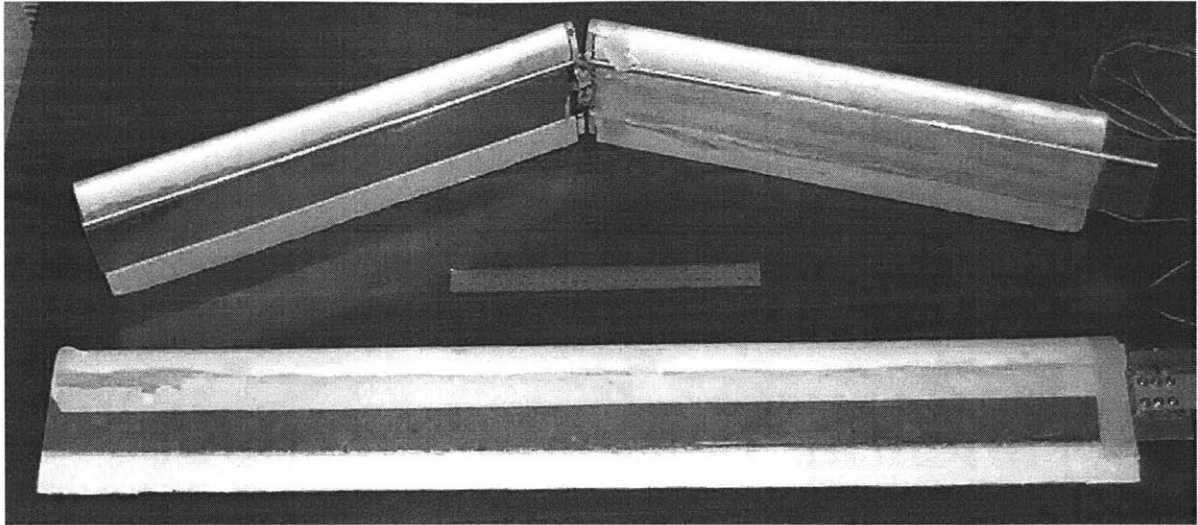


Figure 6-1: UnHINGED and one-HINGED 1-meter span wings (with 30-cm ruler shown for scale)

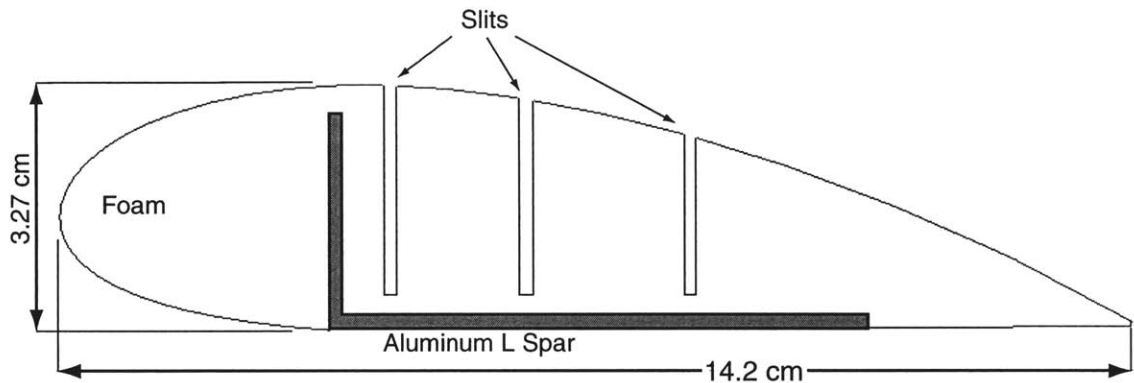


Figure 6-2: Diagram of foam wing cross-section

construction of the wings, each is built with a single ‘L’ shaped aluminum spar covered by foam with the shape of a NACA 4523 airfoil. Initial analyses of the wing were performed assuming that the aluminum spar is the only significant structure in the wing. To reduce the effect of the foam on the stiffness of the wing, spanwise cuts were made in the foam. These cuts were made to open up the closed cell cross-section that is created with the addition of the foam. The spar was secured into the foam using two-part E-Z POXY from Composite Design. The cross-section of the wing is shown in Figure 6-2.

There were two desired features placed on the one-hinged wing design. The first was that the wing would have a lower flutter speed with the hinge in a stiff configuration than the hinge in a soft configuration. As shown in Section 5.2.2, this results in a wing that enters into a LCO once the lower flutter speed is reached. The second design requirement

was that the wing would not stall during the LCO. The amplitude of this LCO would be dependent on the root angle of attack of the wing. The chosen angle of attack was one that would be high enough for the wing to not re-fold at a speed of 10 m/s below the flutter speed. Figure 6-3 shows the eigenvalue plot of the designed wing. This wing design has a low enough torsional stiffness relative to its bending stiffness that there is no second bending / first torsion interaction and any hinge stiffness. In this design, the wing has a 790-Nm hinge stiffness in the stiff state and a 0.0001-Nm hinge stiffness in the soft state. The natural frequencies for the stiff state are given in Table 6.1.

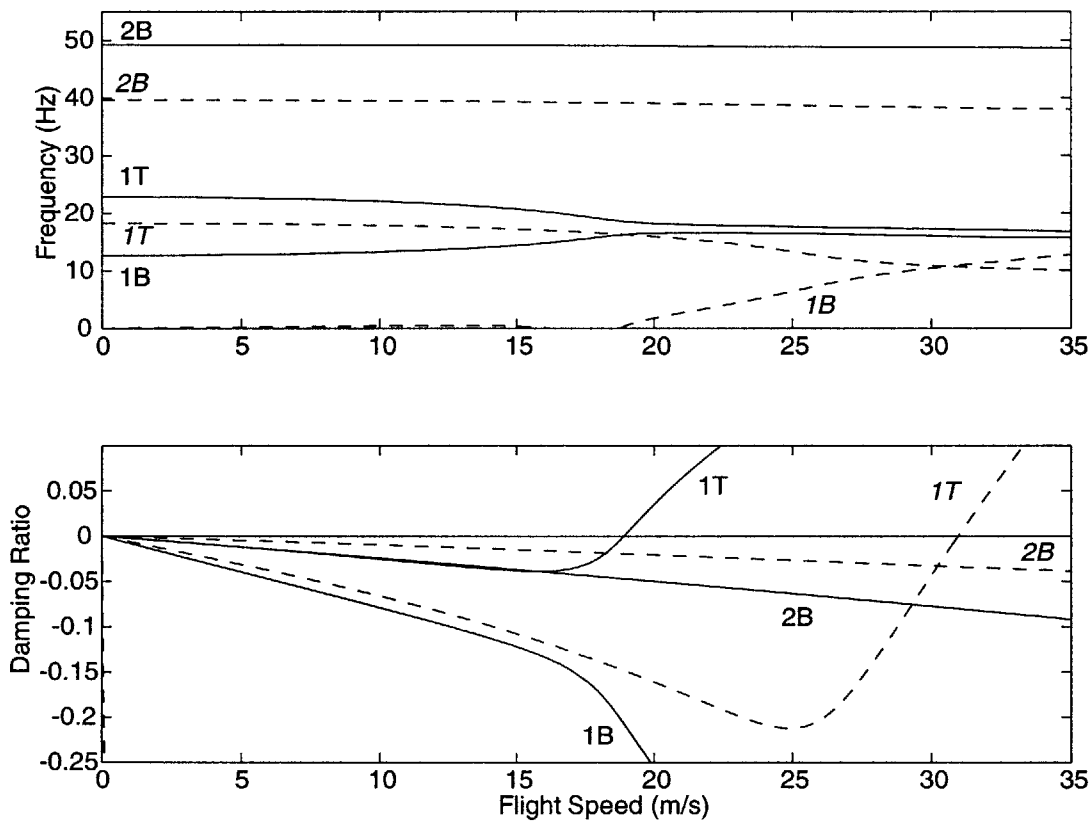


Figure 6-3: Eigenvalue diagram of one-hinged wing as designed, ‘—’ stiff hinge ($U_f/\omega_\alpha b = 1.86$, $k = 0.436$) ‘- -’ soft hinge ($U_f/\omega_\alpha b = 3.77$, $k = 0.157$).

Table 6.1: Natural frequencies of stiff state of one-hinged wing as originally designed

1 st Mode (bending)	2 nd Mode (torsion)	3 rd Mode (bending)
12.6 Hz	22.8 Hz	49.2 Hz

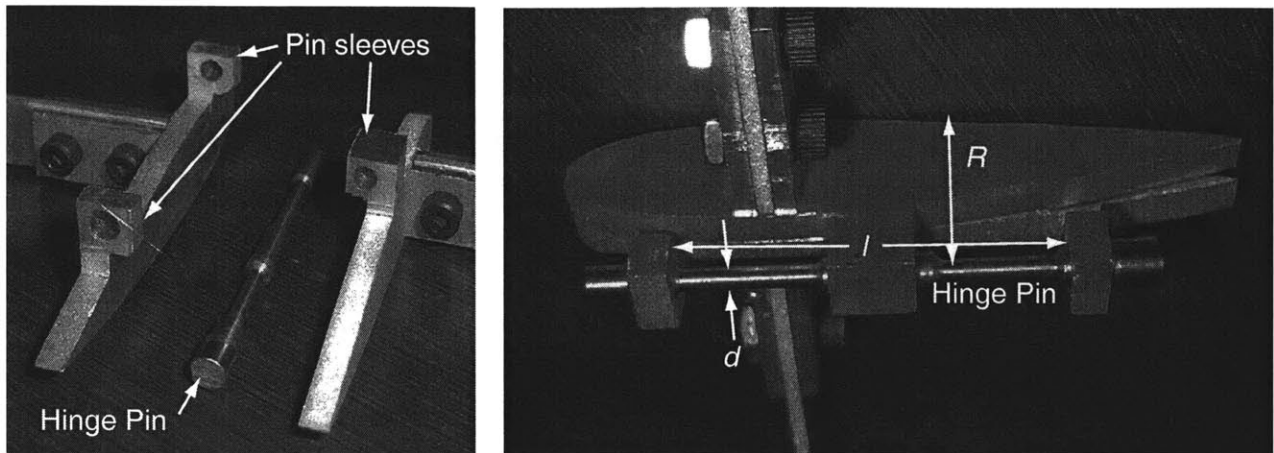


Figure 6-4: Photos of foam wing hinge

6.1.2 Hinge Design

The hinges for the foam wings were designed to simplify analysis and allow for modification of the hinge stiffnesses. The foam wing hinge has two inboard and one outboard pin sleeves attached to thick aluminum airfoil sections (Figure 6-4). This causes the hinge pin to act like a beam in a three point bending fixture. The deflection of the hinge pin was then estimated using Euler beam analysis. Using the assumption that the hinge pin has clamped end conditions leads to a hinge torsional stiffness, k_θ where

$$k_\theta = \frac{3\pi E d^4 R^2}{4l^3} \quad (6.1)$$

where E is the pin material Young's modulus, d is the diameter, l is the length between the inboard sleeves and R is the vertical distance from the pin center to the top of the airfoil.

The foam wings were designed so to modify the torsional stiffness of the hinges by using quickly interchangeable hinge pins. Three sets of stainless steel hinge pins were produced with 1.9, 2.5, and 3.2 millimeters (0.075, 0.100 and 0.125 inch) diameters.

6.2 Wing Characterization

A series of vibration tests were performed to determine the foam wing characteristics. The roots of the wings were clamped and impulse loads were applied to each wing. The wing tip motion was recorded and analyzed to determine its frequency response. To ensure that the hinge stiffness remained linear, the tests were performed with the wing inverted (in the horizontal position) so that the hinge remained in the stiff position.

Tests were performed on just the aluminum spar connected by hinges and on the complete wings (as used in the wind tunnel). This allows the effect of the foam to be isolated. The data obtained was analyzed to find the natural frequencies of the wings. Comparing the hinged and the unhinged wings allows the effect of the hinges on the frequency response to be found experimentally. Additional frequency response tests were performed in the wind tunnel to measure any softness in the wing mount.

6.3 Wind Tunnel Setup

The wings were tested in the Wright Brothers Wind Tunnel at MIT. The configuration was similar to that used during the WASP testing. However, to save time, the standard wind tunnel pedestal was used. Additionally, a right wing was flown instead of a left wing. This was so that the low speed video system of the tunnel could better monitor the behavior of the wing.

6.3.1 Instrumentation

Static and stagnation air pressure and air temperature were measured using the basic instrumentation of the wind tunnel. The lift forces on the wings were measured using a (112-N) load cell.

Both of the foam wings had two Endevco model 22 accelerometers embedded at the wing tip (see Table B.3 for properties); one located near the leading edge, and one near the mid-chord as shown in Figure 6-5. This placement allowed for both plunge and pitch measurements at the wing tip. The mid-chord placement was chosen to prevent damage to

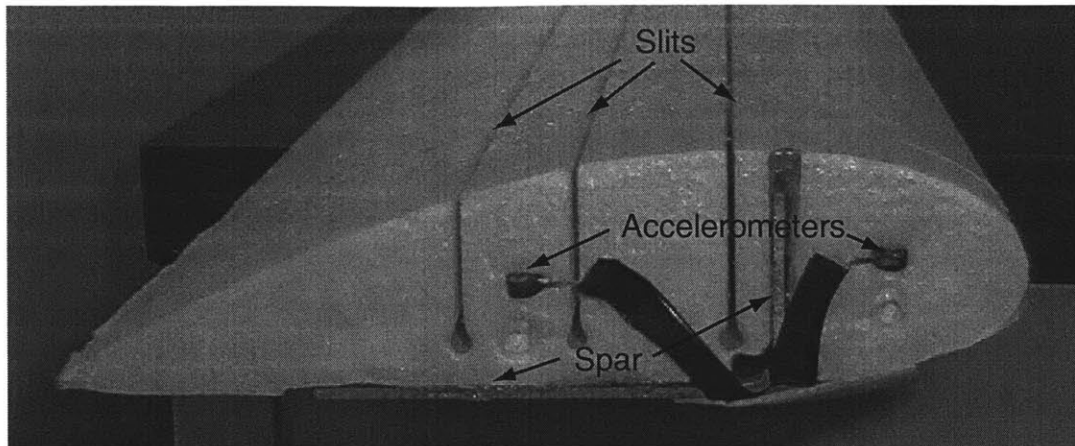


Figure 6-5: Placement of accelerometers in foam wing tip

the accelerometer if the trailing edge foam segment became separated from the wing during testing.

The hinged foam wing had a switch installed at the hinge to monitor the hinge state. This switch was made by placing copper tape on both side of the hinge (as shown in Figure 6-6.) When the hinge was in a fully opened position (or stiff state), the copper tape from both side would come into contact, closing the switch. One end of the switch was wired to a voltage source and the other was monitored with respect to a ground reference.

During testing, the data from the accelerometers and hinge switch were recorded at a rate of 1 kHz in ten second buffers. The airflow data was collected over the same ten seconds and then averaged before being recorded.

6.3.2 Test Procedure

The foam wings were flown at increasing speeds until the wing displayed unsteady flight behavior. The tests were performed at various AOAs, but for each test the root AOA was held constant as the air speed was increased. For the first flight of each wing, a larger number of data points were taken as the wing was brought to unsteady flight speeds. This allowed the variation in the natural frequencies of the system to be monitored. Once the speed at which the wing went unstable was established, the wings were brought up to speed quickly.

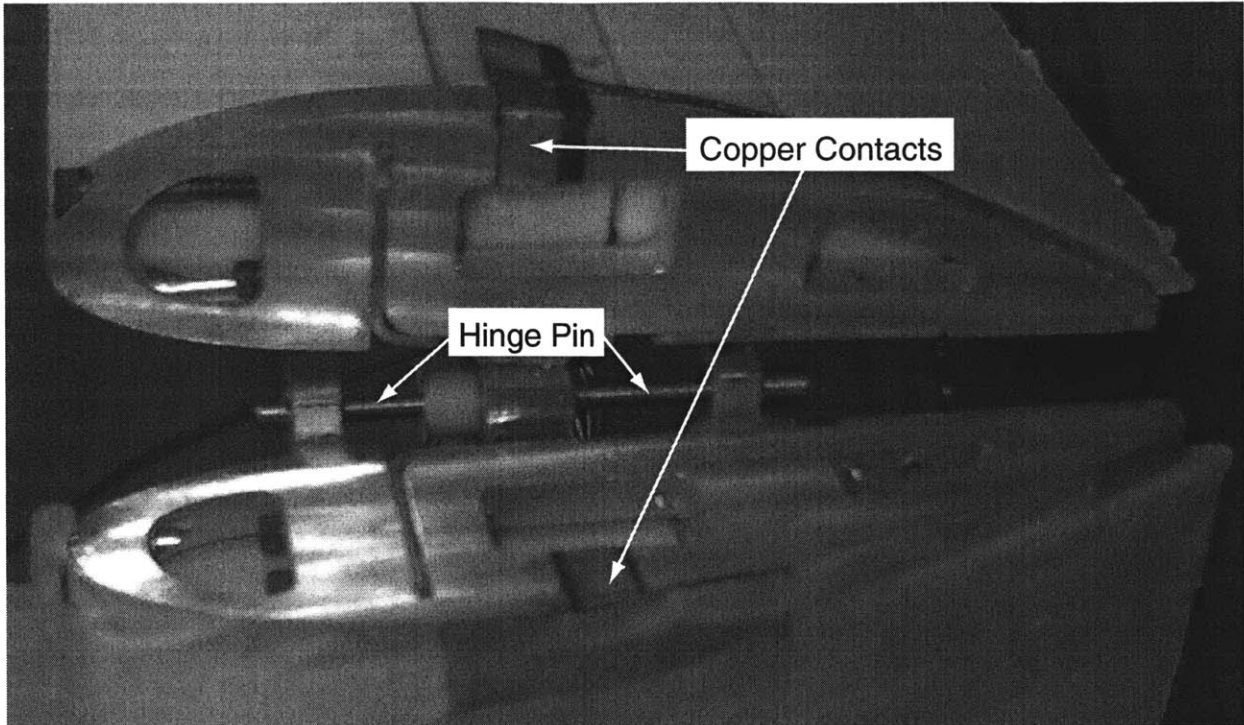


Figure 6-6: Wing hinge with copper contacts

The different test points can be found in Appendix C.2.

6.3.3 Data Postprocessing

The data from the accelerometers on the foam wings was converted into positional data by twice integrating the signal over time. A high-pass, fourth order, Butterworth filter was used to remove the low frequency drift of the accelerometers. The accelerometers have a calibration factor of 0.38 pC/g.

6.4 Wing Bench Test Results

To characterize the foam wings, the natural response frequencies were found through a series of impulse response tests. The tests were performed on both the hinged and the unhinged wings. Additionally, the tests were performed before and after the foam was attached to the spar. Impulse tests were also performed on the wings after they had been mounted in the wind tunnel, just prior to flight testing.

To determine the natural frequencies, the wing was impacted with impulse loads. The wing response was recorded using the two tip mounted accelerometers. With one accelerometer placed near the leading edge and one near the trailing edge of the wing, both the plunge and pitch behavior could be determined. The accelerometer data was recorded and analyzed at 1kHz in ten-second buffers. Table 6.2 summarizes the data collected from the impulse testing. In all cases, the first, second and third modes corresponded to first bending, first torsion and second bending, respectively. These tests show that the foam has a significant effect on the frequency response of the wing, and that the wing mounted in the wind tunnel does not have a perfectly clamped boundary condition, particularly along the bending direction.

Table 6.2: Foam wing impulse response (in Hz)

	1 st Mode (bending)	2 nd Mode (torsion)	3 rd Mode (bending)
Unhinged spar	14.3	32.4	58.8
Unhinged wing	12.6	34.2	61.0
Unhinged wing in tunnel	8.5	34.0	52.5
Hinged spar	11.9	23.4	48.2
Hinged wing	11.2	30.0	49.8
Hinged wing in tunnel	7.5	29.0	36

6.5 Wind Tunnel Experimental Results

To study the dynamic response of hinged wings, the wings were flown in a variety of flight conditions and their behavior was monitored. In the case of the foam wings, the turbulence of the tunnel was enough disturbance to obtain accelerometer data.

The two goals of the foam wing flight tests were to measure the frequency response of the wings over a wide range of speeds and to capture limit cycle behavior of a hinged wing.

At each flight condition, ten seconds of 1kHz data was recorded from both accelerometers to determine the system frequencies. This data was then analyzed using the Welch's method of Power Spectral Density (PSD) approximation [38]. To find the plunge and pitch modes, both the sum and the difference of the accelerometer data were used, respectively. In the

Welch's method, hanning windows are used to divided the data into segments. The PSD of each section is averaged with the rest to give an overall approximation of the PSD. When the accelerometer data was analyzed, many different size hanning windows were used. This is due to the fact that the size of the window used can affect the approximated PSD. This is demonstrated in Figure 6-7, which shows a Welch PSD approximation of accelerometer data. The solid line in the figure shows a PSD for a window size of 600 samples and results in a first mode of 9.9 Hz. The dashed line corresponds to the same data but with a window size of 750 samples. Here the first mode is at 9.6 Hz. The frequency response peaks for each window size were recorded and the mean value and standard deviation of those peaks were found. The frequency response for both the hinged and unhinged foam wings is shown in Figure 6-8. In this figure, the error bars represent the standard deviation in measurement when using varying window sizes.

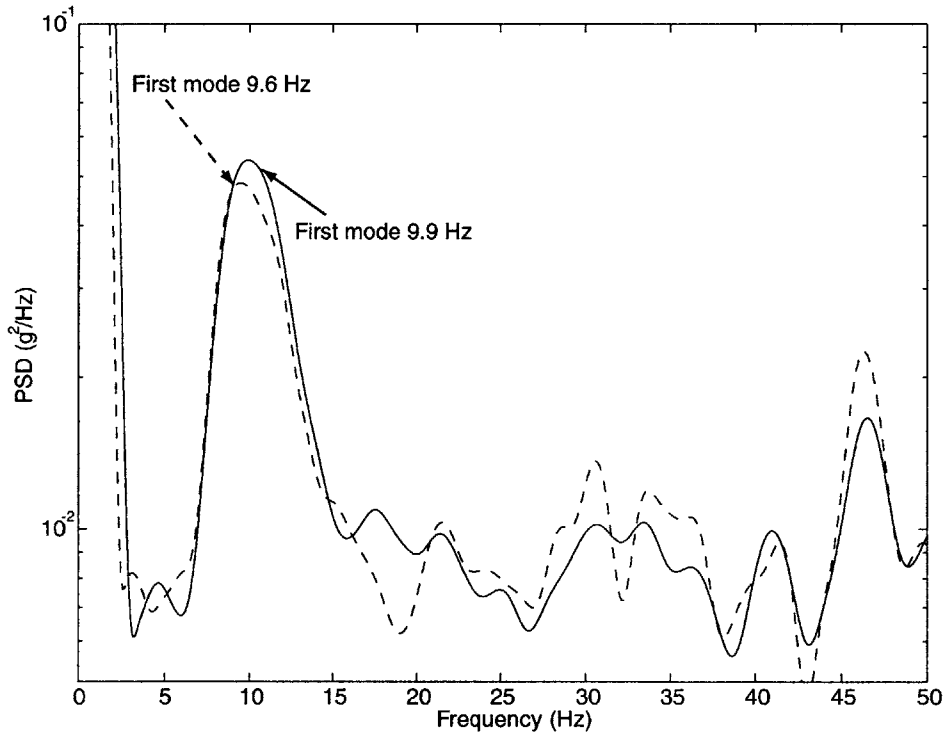


Figure 6-7: PSD approximations of plunge data (‘—’ 600 samples, ‘- -’ 750 samples)

During the flight testing, the hinged wing entered into a limit cycle oscillation (LCO) numerous times. All of these flight conditions were captured using the low speed video system. However, due to the hazards of flying a wing with a large amplitude motion, only

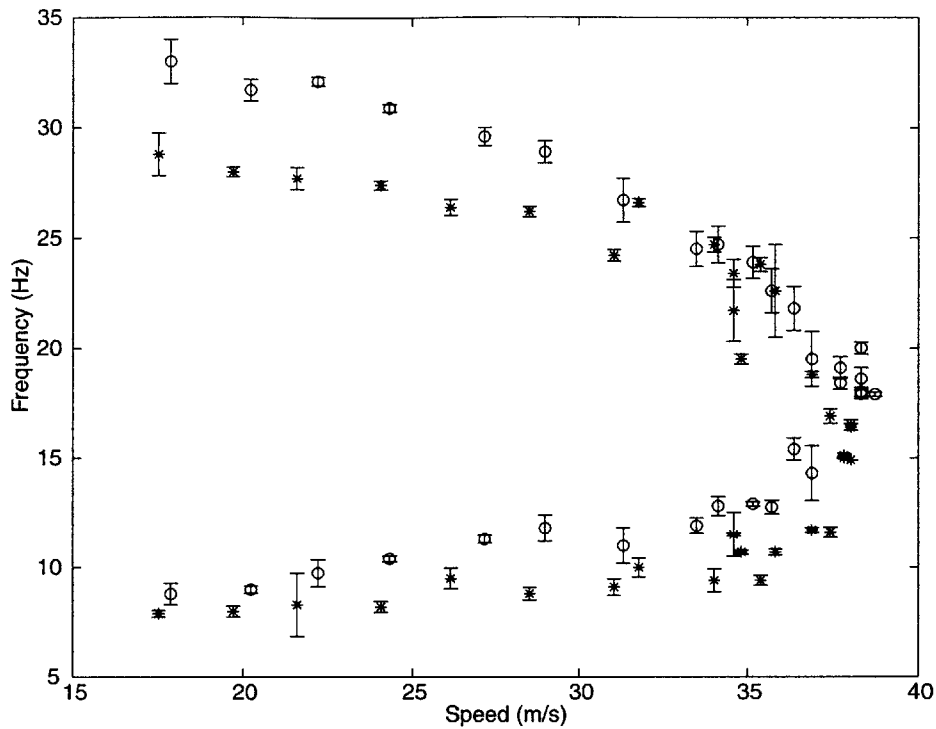


Figure 6-8: First two natural frequencies over range of dynamic pressures for the foam wings ('o' unhinged, '*' hinged)

limited amounts of accelerometer data was recorded. There were two different types of LCO behavior that were observed. One was a hinged-based LCO and the other was a stall-based LCO.

The hinged-based LCO tended to be transient in nature. This behavior occurred when a gust would cause the outer-wing segment to deflect downward, opening the hinge. At this point the wing motion amplitude would grow dramatically in just a few cycles, as seen in Figure 6-9. However, when the hinge started to change states, from soft to stiff, the motion would damp out quickly. Table 6.3 shows at what operating conditions these bursts of wing motion were recorded.

Table 6.3: Operating conditions where a hinge-based limit cycle was recorded

Flight Speed (m/s)	AOA (degrees)	Notes
37.6	-3	
37.9	-2.5	two recorded
37.1	-3	
36.5	-2	two overlapped

Stall based LCO behavior was only encountered twice during testing. Once with the hinged wing and once with the unhinged wing. In both cases the wings were being flown at a zero degree AOA. The wing entered into a stable LCO (shown in Figures 6-10 and 6-11) and remained there until the flight speed was reduced. The amplitude of the hinged wing (0.043 meters at half chord) was greater than that of the unhinged wing (0.030 meters at half chord) due to the decreased bending stiffness.

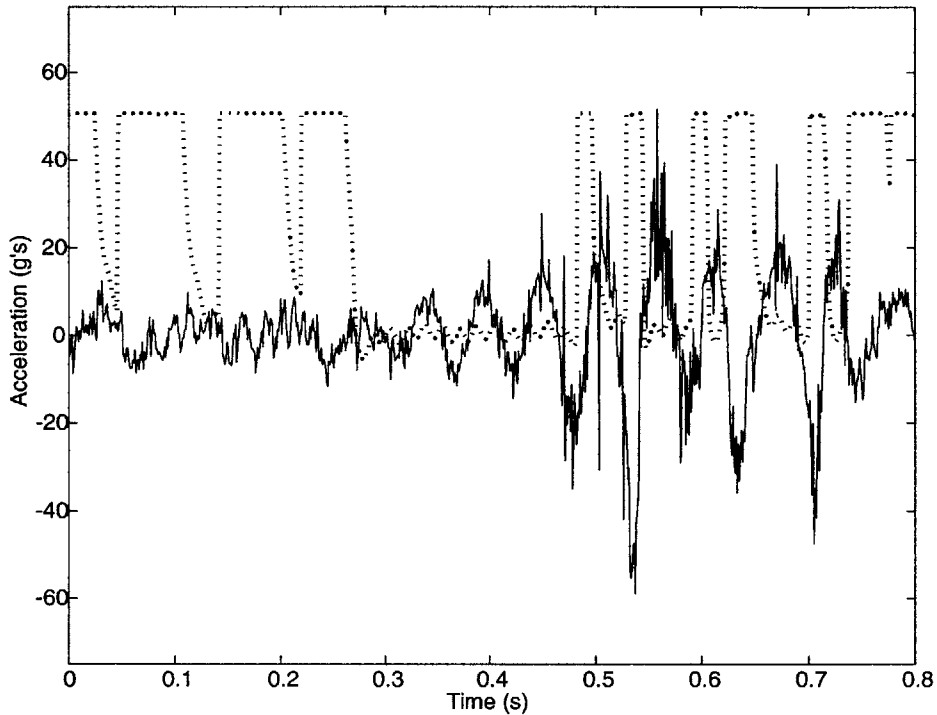


Figure 6-9: Mid-chord accelerometer data (solid line) during a hinge-based limit cycle oscillation of hinged wing at 36.5 m/s and -3° root AOA. Hinge state data (dotted line) values of less than 50 indicate a partially closed hinge

To find positional data of the wing tip the following steps are used (and the results of which are illustrated in Figure 6-12).

1. Multiply raw data by 9.8 to change from g 's to m/s^2 .
2. Remove the constant and linear error by finding a linear curve fit to the ten seconds of data.
3. Integrate using a trapezoidal scheme to find velocity.

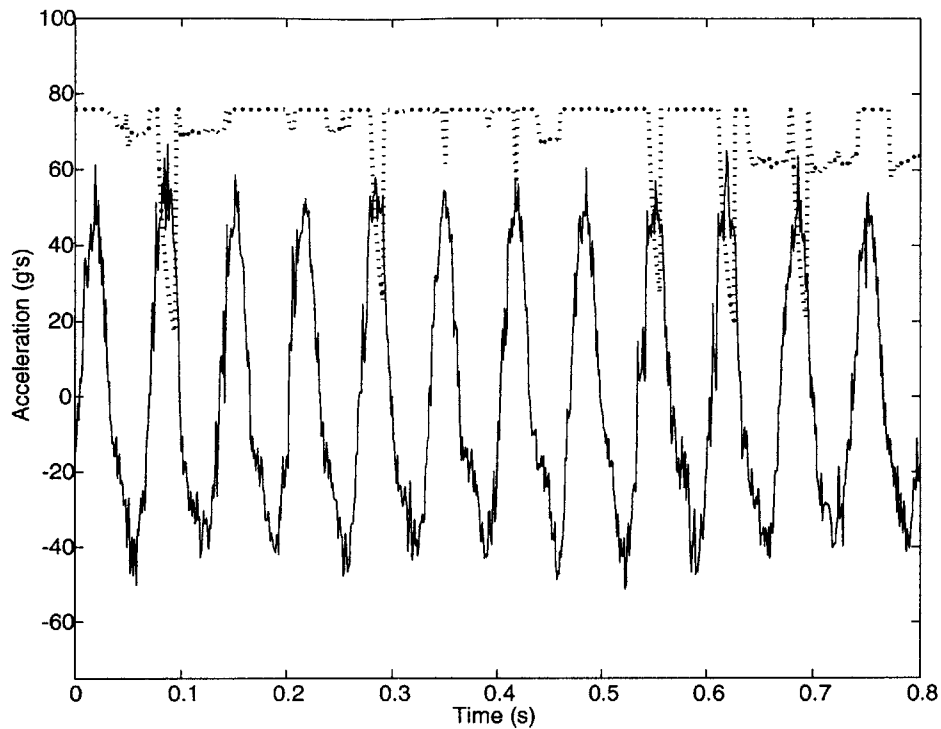


Figure 6-10: Mid-chord accelerometer data (solid line) during a stall based limit cycle oscillation of hinged wing at 38.6 m/s and 0° root AOA. Hinge state data (dotted line) values less than 75 indicate a soft hinge state

4. Perform the same constant and linear error removal to the velocity.
5. Integrate again to obtain position.
6. Run the position results through a high-pass filter to reduce the low frequency error.
7. Find the plunge and pitch values by weighted average and difference of the positional data, respectively.

The method can use less than the whole ten seconds of accelerometer data for greater accuracy. The filter used in step (6) was a fourth-order Butterworth high pass filter. The pass band of the filter was changed each time to insure that it was slightly lower than the first mode of the data. The main limitation of this method is that the low frequency, rigid body motion of the wing tip is lost. Unfortunately, no way was found to keep this information while reducing the low frequency drift of the accelerometers.

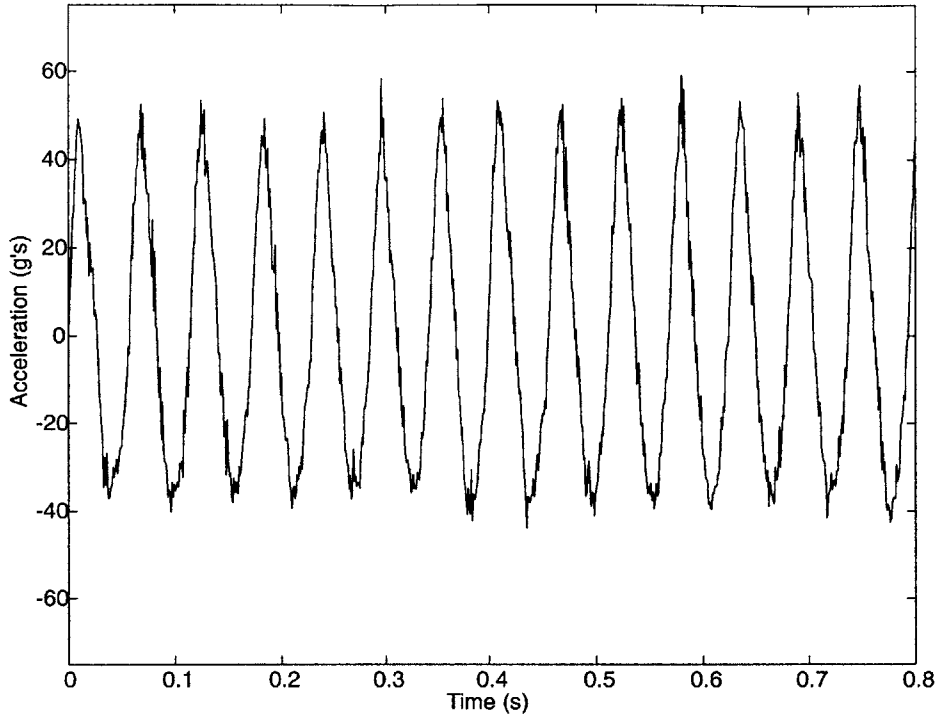


Figure 6-11: Mid-chord accelerometer data (solid line) during a stall-based limit cycle oscillation of unhinged wing at 38.8 m/s and 0° root AOA

6.6 Model Correlation

6.6.1 Linear Behavior

Based upon the bench test data presented in Section 6.4, it is apparent that the manufactured wing would not behave as designed. To recreate the experimental results, elastic tuning factors had to be added to account for the effect of the foam and the soft wind tunnel mount. Table 6.4 shows the effect of the tuning factors on the natural frequencies of the model. The first tuning factor was the hinge stiffness. Using Equation 6.1, the hinge stiffness for the 2.9-mm diameter pin is 790 Nm. However, a value of 350 Nm was used to reproduce the bench test results. The difference is attributed to tolerances in the hinge design. The second tuning factor added the effect of the foam on the torsional stiffness. To match the bench test data, the torsional stiffness of the wing was multiplied by a factor of 2.9. Finally, to account for an imperfect clamping in the wind tunnel, a linear hinge was placed at the root of the model. This hinge had a stiffness of 800 Nm. A similar procedure was followed for the

unhinged wing where the torsional tuning factor was 2.8 and the same root hinge stiffness of 800 Nm was used. The reason for the difference in the torsional factors is attributed to slight differences in the manufacturing of the foam.

Table 6.4: Effect of model tuning factors on wing frequencies

	Experiment			Model		
	1 st Mode	2 nd Mode	3 rd Mode	1 st Mode	2 nd Mode	3 rd Mode
Unhinged spar	14.3	32.4	58.8	14.2	32.7	59.0
Hinged spar	11.9	23.4	48.2	11.9	23.4	48.1
Hinged wing	11.2	30.0	49.8	11.3	30.1	49.5
Hinged mounted	7.5	29.0	36	7.5	29.6	35.4

Once the model was tuned using the bench test data, it was able to reproduce the wing frequencies recorded during the experiments. Figure 6-13 and 6-14 show how the model results compare with the experimental data. Only the frequency response is shown. In these experiments, the wings were excited by the flow turbulence in the wind tunnel. This did not allow for any transients to be observed, thus the damping values of the modes could not be extracted from the data.

The model also predicts that the increased torsional stiffness from the foam and the soft root cause significant change to the flutter behavior of the wing. Table 6.5 shows the change in flutter speed of the hinge wing model caused by the off-design wing. In this table, the values for the experimental flutter speed are based upon the minimum observed flutter speed for each condition. Due to the increased torsional stiffness and the decreased bending stiffness between the original wing design and the tuned model, the tuned model predicts a second bending / first torsion instability at low hinge stiffnesses. Figure 6-15 is an eigenvalue plot of the tuned model.

Table 6.5: Effect of model tuning on flutter speed (m/s)

	Unhinged	Stiff	Soft
Original design	21.6	18.8	30.1
Tuned model	39.2	39.9	36
Experimental	38.8	38.6	36.5

6.6.2 Nonlinear Behavior

The linear model predicts that the wing that was flown in the wind tunnel flutters first with the wing in the soft state. Thus the wing flown did not meet the design characteristics established earlier in this chapter. However, this does explain the hinge-based LCO observed in Figure 6-9. The wing was flying at a speed above the soft state flutter, but below the stiff state flutter. As long as the hinge remained in the stiff state, the wing had a benign behavior. However, if a gust caused the hinge to change states for a certain minimum length of time, the wing would respond with an exponential growth in amplitude. This growth would be limited when the hinge started to switch states. When the gust is over, the hinge returns to the stiff state and the wing behavior becomes stable again.

Although the model was unable to reproduce the exact response observed in the experiment because of the difficulty of determining the initial conditions and gust properties, it was able to reproduce a similar burst response to a gust. Figure 6-16 show the tip plunge acceleration and the tip twist response to a 0.3-second duration cosine gust which starts at 0.2-seconds. In this simulation, there is an exponential growth in motion while the hinge is in the soft state. When the steady state forces return the hinge to the stiff state, the motion is damped out in a similar manner that which was observed experimentally.

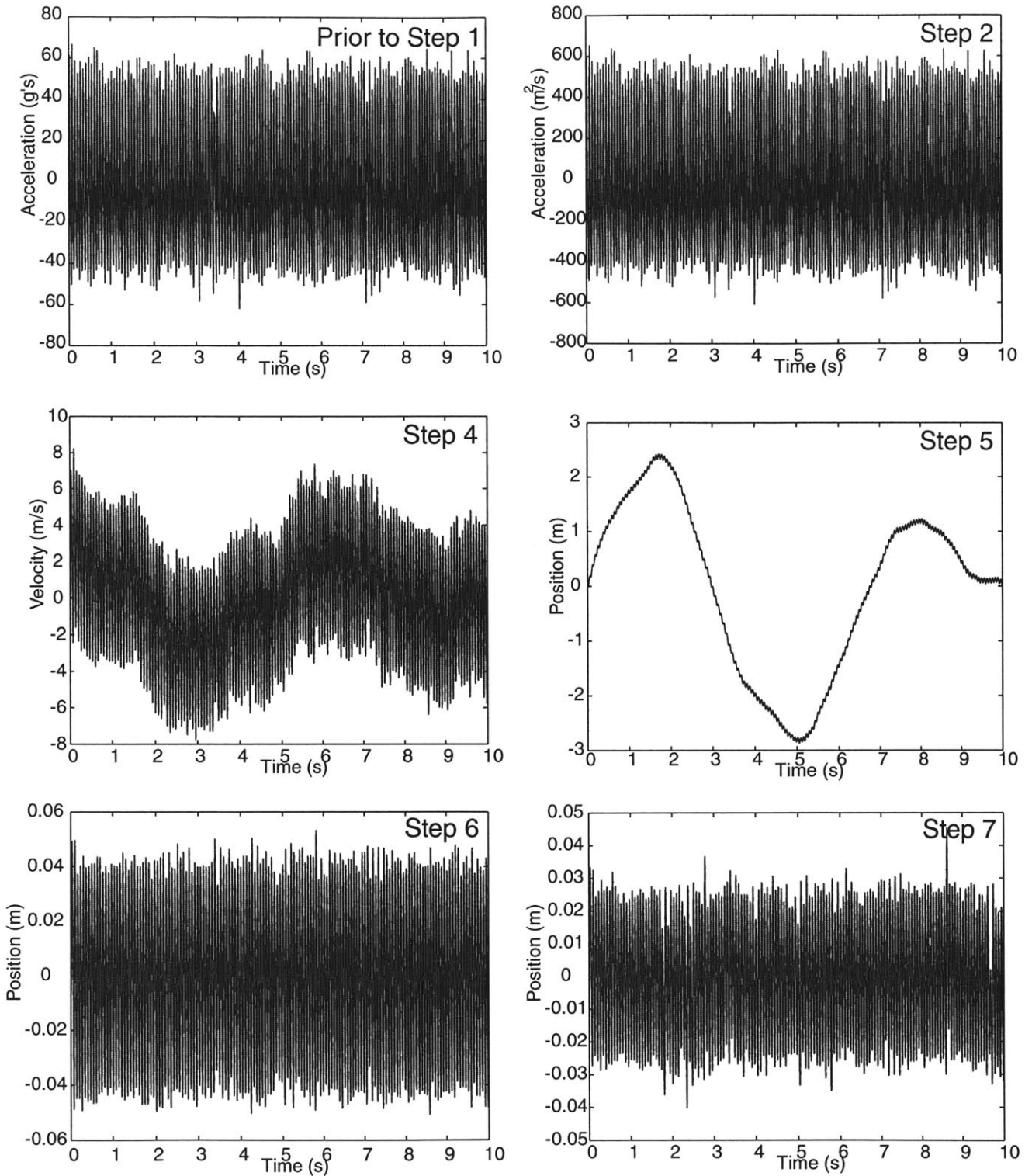


Figure 6-12: Steps followed for tip plunge determination, referring to steps outlined in Section 6.5 for one-hinged wing at 38.6 m/s and 0° AOA

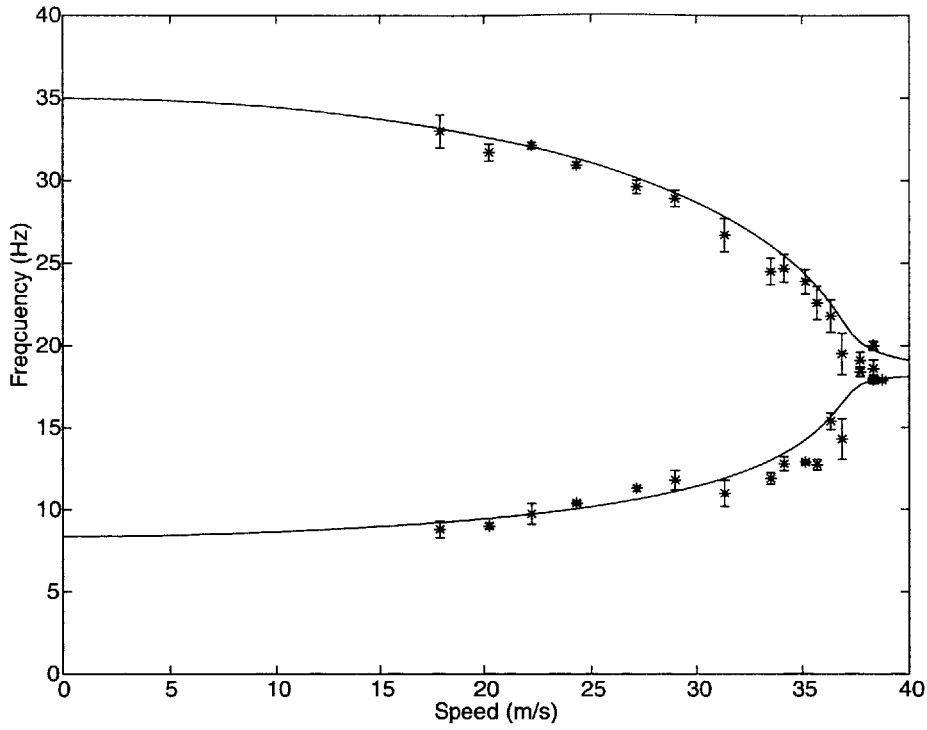


Figure 6-13: Comparison of first two natural frequencies of the unhinged wing over range of dynamic pressures for model (-) and experimental data (*)

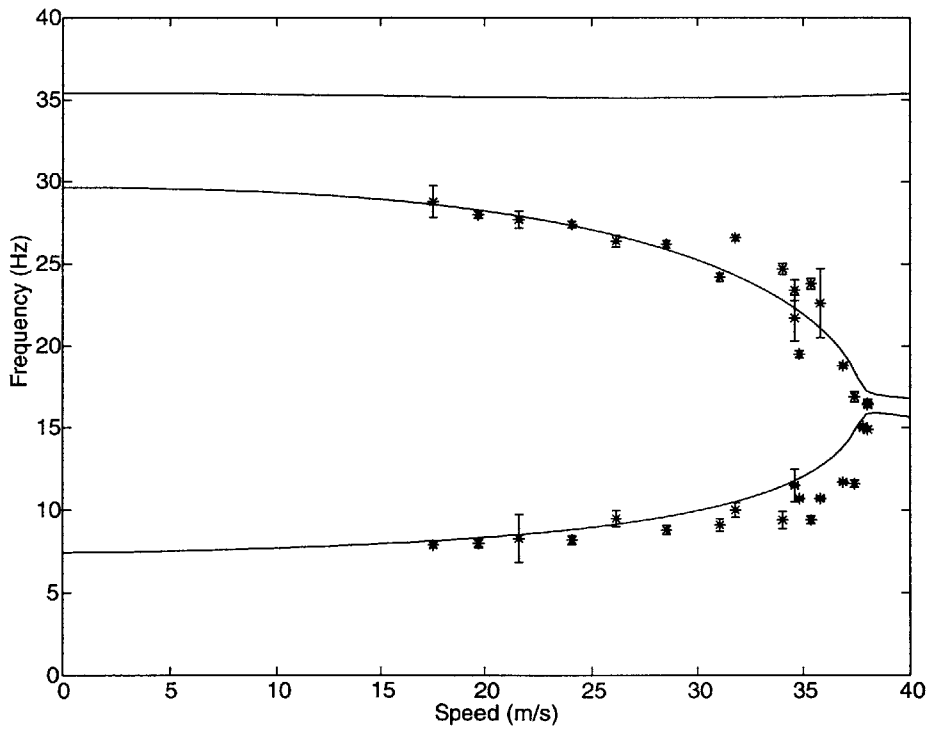


Figure 6-14: Comparison of first two natural frequencies of the hinged wing over range of dynamic pressures for model (-) and experimental data (*)

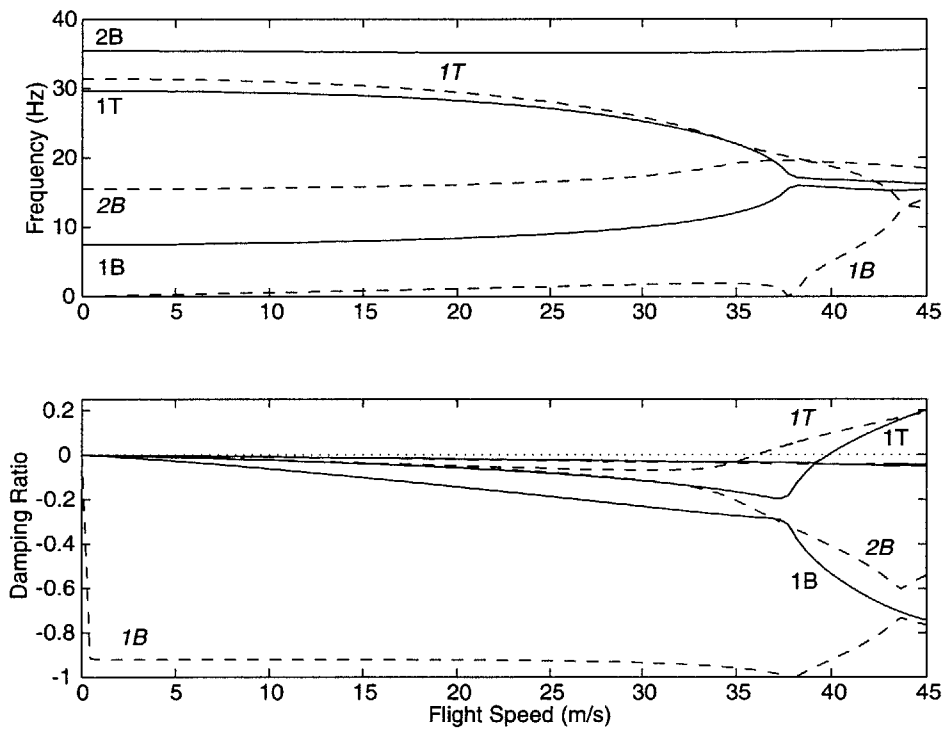


Figure 6-15: Eigenvalue diagram of one-hinged wing as designed, ‘—’ stiff hinge ($U_f/\omega_\alpha b = 3.00$, $k = 0.188$), ‘- -’ soft hinge ($U_f/\omega_\alpha b = 2.56$, $k = 0.264$).

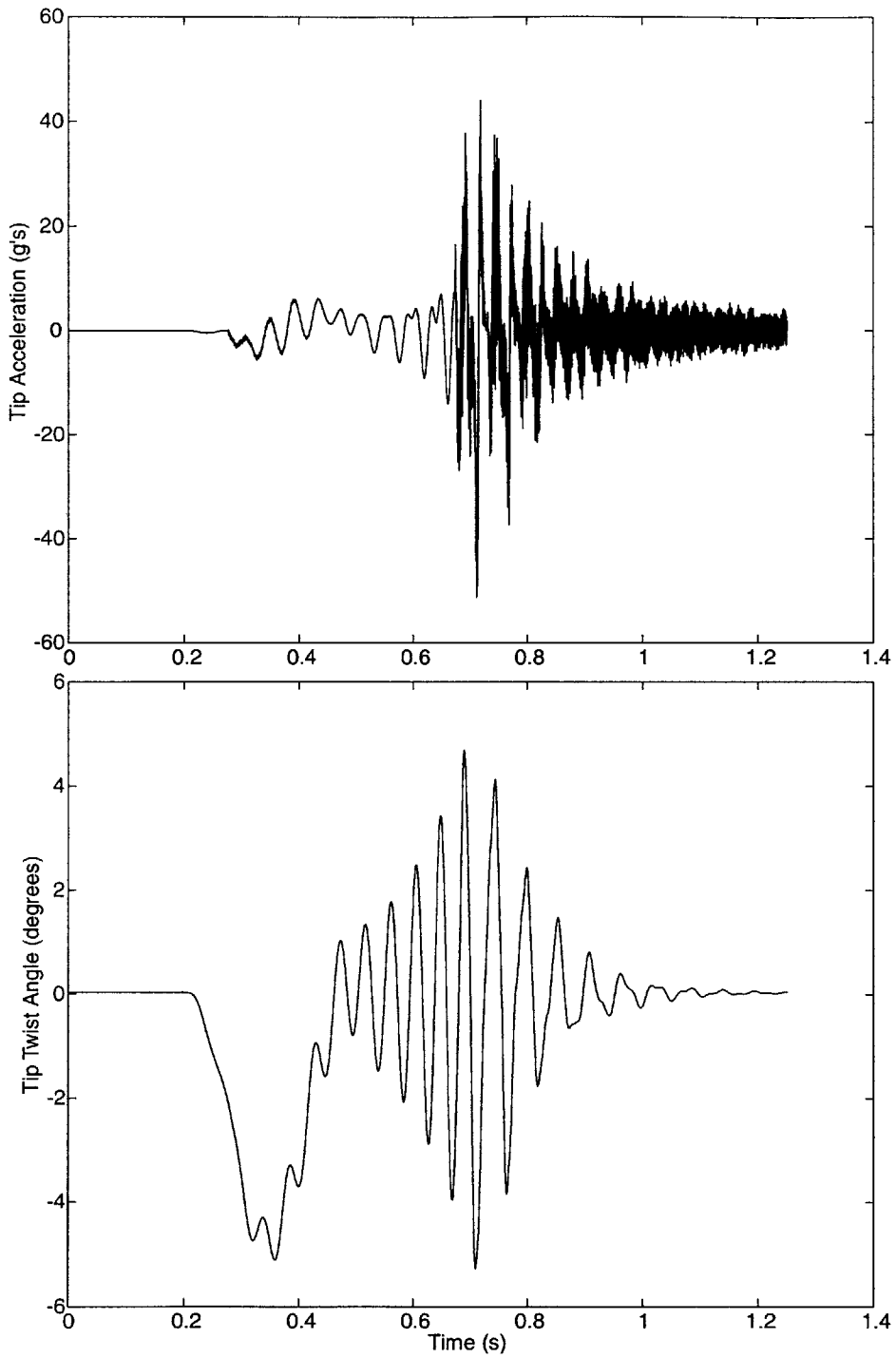


Figure 6-16: Simulation results for a one-hinged wing flying at 36.5 m/s and a -3° root AOA subjected to a 0.3-second cosine gust starting at 0.2 seconds

Chapter 7

Concluding Remarks

This chapter summarizes the theoretical and experimental work presented in this thesis and its key contributions. Suggestions are also made for further work which could improve the understanding of multi-hinged wings.

7.1 Overview

Motivated by the need to understand the aeroelastic behavior of the multi-hinged wing designed for use with the high-g WASP vehicle, analytical and experimental studies were conducted in this thesis. The analytical model captures the stiffness nonlinearities introduced by adding hinges to a wing. It uses linear strip theory to model the unsteady aerodynamic forces on the wing, and the response of the wing is determined using an elastic multi-body dynamics model.

To gain initial insight into the performance of a multi-hinged wing, a series of wind tunnel tests were performed on the prototype five-hinged WASP wing. These tests involved disturbing the wing under various flight conditions and recording the response. The data collected from these tests was used to validate the analytical model and to observe phenomenon that the analytical model is not able to capture.

After showing that it was capable of reproducing much of the same five-hinged wing characteristics observed in the experimental testing, the analytical model was used to study

flight regimes of a five-hinged wing that could not be obtained in the physical testing. Additionally, various wing configurations, most notably the simple one-hinged wing, was used to study the effect of adding unlatched hinges to a wing on more generic terms.

The results from the analytical studies showed that adding hinges to a wing causes a significant change in its aeroelastic behavior. The presence of hinges can cause the bending frequencies of the wing to be significantly lower without significantly changing the torsional frequencies. This nominally increases the speed at which the first bending / first torsion (1B/1T) instability will occur. However, it might also induce a second bending / first torsion (2B/1T) instability which (if present) occurs at speeds lower than the 1B/1T instability of a normal wing.

Additionally, the nonlinearity of the hinges can cause the wing to enter into a limit cycle or non-periodic motion at speeds near the 1B/1T instability boundary. At speeds above the 2B/1T instability boundary, but below the 1B/1T instability boundary, the model will either enter a limit cycle, decaying, or divergent oscillation depending on the initial conditions of the wing.

Finally, a hinged wing has a tendency to refold when subject to negative or near zero net loads. The analytical model predicts a static refold at -8.5° root AOA for the WASP wing. However, experimental results show this refolding more as an oscillation which begins around -4.5° AOA and increases in amplitude and mean tip displacement as the AOA is decreased. The inability of the model to capture the experimental results is due to the forced excitation from the wind tunnel turbulent flow not being modeled.

Another set of wind tunnel experiments was devised to verify the results of the analytical studies. These experiments involved flying both a hinged and an unhinged wing in flow conditions which caused the wings to exhibit unsteady behavior. Data of both a hinge-limited and a stall-limited wing flutter were obtained. The analytical model was able to approximate the flutter speeds within 3.1% of the experimentally verified ones (see Table 6.5).

7.2 Key Contributions

This thesis presented a novel study on the effect of hinges on the aeroelastic performance of a wing. Hinged wings were built and tested in a wind tunnel. The experimental results of those tests were used to develop and verify a simulation capable of investigating hinged wing behavior. This simulation was used to produce a linear and non-linear framework for the design of future hinged wings. It was uncovered that:

- The non-symmetric, bi-linear nature of a hinge causes a disruption in the coalescence of modes in a flutter state which stops the exponential growth that would occur on an equivalent linear system. This can lead to either periodic or non-periodic behavior in post-flutter flight.
- Hinge stiffness plays a major role in determining the flutter speeds of a hinged wing through a modification of the bending frequencies. A hinged wing cannot be simply modeled as a continuous beam. Hinges always cause a significant change in the behavior of a wing, either by locally reducing the stiffness or the structural inertia of the wing. Varying the hinge stiffness can cause both an increase or a decrease in the flutter speed and can change the flutter mode. Any hinged wing that flutters at a lower speed than a similar unhinged wing will eventually go into a hinge limited cycle whose amplitude is determined by the initial energy of the system (unless other non-linear effects such as stall are first encountered).
- A designer can determine the impact of adding hinges to a wing using a linear model with a classic unsteady model. The flutter speeds for each hinge in each state need to be found and the lowest speed would set the flight envelope.
- A hinged wing that is not flown near a flutter speed or a low angle of attack will have the same performance as a similar unhinged wing except for whatever drag and weight penalties the hinges might cause.

7.3 Design Considerations

The stiffness properties of the hinges themselves are very important in the design of a hinged wing. The natural frequencies of the wing can be tuned to alter its flutter speed. However, careful attention must be paid to possible interaction between second bending and first torsion modes. Lowering the hinge stiffness increases the speed at which the 1B/1T instability occurs, but also increases the chance of a low speed limit cycle oscillation. While there is no significant interaction observed in this study involving the second torsion or third bending mode, they should still be monitored.

If hinge springs¹ are used, they increase the angle of attack range at which the wing can operate without refolding. Hinge springs increase the amount of negative external loading the wing can see before it refolds. However, care must be used when adding springs to a wing that may encounter flight speeds near flutter. Springs increase the bending frequencies for wings with hinges in the soft state. This increases the chance the wing will not go into a limit cycle oscillation at speeds higher than flutter, but will instead have a divergent oscillation.

At speeds below flutter, the springs can increase the amount of energy needed for the hinges to switch states. The amplitude of the oscillations seen in 2B/1T instability is directly related to the energy needed to switch states. Thus, if the hinges do switch states, it is advantageous to have springs that do not impart a large torque at zero angle as this increases the energy of the hinge. However, the hinges need to switch states for limit cycle oscillation to occur between the 2B/1T instability and 1B/1T instability. Although it might be thought to be advantageous to have large torque at zero angle such that the hinges do not switch states, most cases would involve the wing first entering this state after seeing the 2B/1T instability region. Thus, the hinges would already be switching states.

A low-speed aeroelastically conservative hinged wing can be designed by using a linear analytical model with classical unsteady aerodynamic theory. Such an analysis could find the linear flutter points of all combinations of hinge states. As long as the lowest flutter speed is not reached, the designed wing could be flown without encountering either a limit cycle oscillation or a divergent flutter.

¹For design purposes, hinge latches can be modeled as very stiff hinge springs.

7.4 Future Work

In this thesis, an analytical model was created and validated using experimental data. However, both the model and the data could be improved for further studies of a multi-hinged folding wing. Additionally, further studies could be performed on the effects of the various parameters such as location and number of hinges. These parametric studies could be used to determine effects of manufacturing variability and/or wear of hinges during operation.

7.4.1 Experimental Improvements

Further studies could be conducted using a one-hinged wing. Testing of a wing that would enter into a steady, hinge limited oscillation would allow for more accurate measurements of the limit cycle behavior. Comparisons of predicted and measured frequency and amplitude of the oscillations could be made. Additionally, a test could be developed that would show a steady limit cycle developing out of finite wing perturbation. This might involve using large hinge springs or multiple hinges.

Using a tunnel with a higher flow quality would allow for the wing refold to be better investigated. In addition, by reducing the random loads on the wing, the point at which the wing refolds could be more accurately determined. Using a more precise force balance would allow for more insight on what net loads cause a refold. This could also be facilitated by using a wing with symmetric airfoil.

Instrumenting the wing with improved on-board sensors would allow improved positional measurements to be made.

7.4.2 Model Improvements

The analytical model could be improved in several ways. Initially, the addition of a broadband noise generator in the aerodynamic loading would allow for the refolding conditions to be better determined. Secondly, including means by which the various segments could interact with each other mechanically and aerodynamically could allow for precise deployment predictions. Thirdly, adding more nonlinear effects (such as elastically stiffening springs,

hinges, and segments or spring free play) would allow for accuracy to be retained in large amplitude motions. Fourth, improved methods for prediction of hinge properties need to be found. Fifth, the computational efficiency of the model could be improved to allow for less expensive computations. Sixth, the model could be modified to allow for an arbitrary reference axis, instead of using the elastic axis.

Finally, if a proposed hinged wing vehicle would either deploy or cruise at higher speeds, compressibility effects must be added to the unsteady aeroelastic model. Additionally, a more sophisticated unsteady aerodynamic model with improved accuracy for large apparent angles of attack would improve the model performance in simulating limit cycle oscillations.

Bibliography

- [1] Ledé, J.-C., “http://www.hiflight.com/gallery_jason.html.” Aurora Flight Science, Inc., 2002.
- [2] Bovais, C. S., M. E. Haupt, and P. L. Toot, “Flight Testing with Sub-Scale Remotely Piloted Drop Models.” AIAA Paper 92-4079-CP, 1992.
- [3] Bovais, C. S. and P. L. Toot, “Flight Testing the Flying Radar Target (FLYRT).” AIAA Paper 94-2144-CP, 1994.
- [4] Bernstein, J. I., *System Design for a Rapid Response Autonomous Aerial Surveillance Vehicle*, Master’s thesis, Massachusetts Institute of Technology, 1997.
- [5] Casiez, T. D., *Compact High-g, Efficient Folding Wing for a Cannon-Launched Reconnaissance Vehicle*, Master’s thesis, Massachusetts Institute of Technology, 1998.
- [6] Mitsugi, J., “Direct Strain Measure for Large Displacement Analyses on Hinge Connected Beam Structures,” *Computers and Structures*, Vol. 64, No. 1-4, 1997, pp. 509–517.
- [7] Shabana, A. A., *Dynamics of Multibody Systems: Second Edition*, Cambridge, 1998.
- [8] Bauchau, O. A., “Computational Schemes for Flexible Nonlinear Multi-Body Systems,” *Multibody Systems Dynamics*, Vol. 2, 1998, pp. 169–225.
- [9] Schiehlen, W. O., ed., *Multibody Systems Handbook*, Springer Verlag, Berlin, 1990.
- [10] Breitbach, E., “Effects of Structural Non-Linearities on Aircraft Vibration and Flutter.” AGARD Report 665, 1977.

- [11] Tang, D. M. and E. H. Dowell, "Flutter and Stall of a Helicopter Blade with Structural Nonlinearity," *Journal of Aircraft*, Vol. 29, No. 5, Sept.-Oct. 1992, pp. 953–960.
- [12] Price, S. J., B. H. K. Lee, and H. Alighanbari, "An Analysis of the Post-Instability Behavior of a Two-Dimensional Airfoil with a Structural Nonlinearity." AIAA paper 93-1474-CP, 1993.
- [13] Price, S. J., H. Alighanbari, and B. H. K. Lee, "The Aeroelastic Response of a Two-Dimensional Airfoil with Bilinear and Cubic Structural Nonlinearities." AIAA paper 94-1546-CP, 1994.
- [14] Lee, I. and S.-H. Kim, "Aeroelastic Analysis of a Flexible Control Surface with Structural Nonlinearity," *Journal of Aircraft*, Vol. 32, No. 4, 1995, pp. 868–874.
- [15] Kim, S.-H. and I. Lee, "Aeroelastic Analysis of a Flexible Airfoil with Freeplay Nonlinearity," *Journal of Sound and Vibration*, Vol. 193, No. 4, 1996, pp. 823–846.
- [16] Conner, M. D., L. N. Virgin, and E. H. Dowell, "Accurate Numerical Integration of State-Space Models for Aeroelastic Systems with Free Play," *AIAA Journal*, Vol. 43, No. 10, 1996, pp. 2202–2205.
- [17] Conner, M. D., D. M. Tang, E. H. Dowell, and L. Virgin, "Nonlinear Behavior of a Typical Airfoil Section with Control Surface Freeplay: a Numerical and Experimental Study," *Journal of Fluids and Structures*, Vol. 11, 1997, pp. 89–109.
- [18] O’Neil, T. and T. W. Strganac, "Aeroelastic Response of a Rigid Wing Supported by Nonlinear Springs," *Journal of Aircraft*, Vol. 35, No. 4, 1998, pp. 616–622.
- [19] Preidkman, S. and D. Mook, "Time-Domain Simulations of Linear and Nonlinear Aeroelastic Behavior," *Journal of Vibration and Control*, Vol. 6, 2000, pp. 1135–1175.
- [20] Sheta, E., V. Harrand, D. Thompson, and T. W. Strganac, "Computational and Experimental Investigation of Limit Cycle Oscillations of Nonlinear Aeroelastic Systems," *Journal of Aircraft*, Vol. 39, No. 1, January 2002, pp. 133–141.

- [21] McIntosh, Jr., S. C., R. E. Reed, Jr., and W. P. Rodden, "Experimental and Theoretical Study of Nonlinear Flutter," *Journal of Aircraft*, Vol. 18, No. 12, Dec. 1981, pp. 1057–1063.
- [22] Hauenstein, A. J., R. M. Laurenson, W. Eversman, G. Galecki, I. K. Qumei, and A. K. Amos, "Chaotic Response of Aerosurfaces with Structural Nonlinearities." AIAA Paper 90-1034-CP, 1990.
- [23] Hauenstein, A. J., J. A. Zara, W. Eversman, and I. K. Qumei, "Chaotic and Nonlinear Dynamic Response of Aerosurfaces with Structural Nonlinearities." AIAA Paper 92-2547-CP, 1992.
- [24] Lee, B. H. K., S. J. Price, and Y. S. Wong, "Nonlinear Aeroelastic Analysis of Airfoils: Bifurcation and Chaos," *Progress in Aerospace Sciences*, Vol. 35, No. 3, 1999, pp. 205–334.
- [25] "NIH Image." <http://rsb.info.nih.gov/nih-image/>. U.S. National Institutes of Health, 2002.
- [26] Shinbrot, T., C. Grebogi, J. Wisdom, and J. A. Yorke, "Chaos in a Double Pendulum," *American Journal of Physics*, Vol. 60, No. 6, June 1992, pp. 491–498.
- [27] Cesnik, C. E. S. and D. H. Hodges, "VABS: A New Concept for Composite Rotor Blade Cross-Sectional Modeling," *Journal of the American Helicopter Society*, Vol. 42, No. 1, 1997, pp. 27–38.
- [28] Bowden, M. and J. Dugundji, "Joint Damping and Nonlinearity in Dynamics of Space Structures," *AIAA Journal*, Vol. 28, No. 4, 1989, pp. 740–749.
- [29] Drela, M., "XFOIL: An Analysis and Design System for Low Reynolds Number Airfoils," in *Low Reynolds Number Aerodynamics* (Mueller, T. J., ed.), No. 54, Springer-Verlag, June 1989. Lecture Notes in Engineering.
- [30] Edwards, J. W., H. Ashley, and J. V. Breakwell, "Unsteady Aerodynamic Modeling for Arbitrary Motions," *AIAA Journal*, Vol. 17, No. 4, 1979, pp. 365–374.

- [31] Theodorsen, T., "General Theory of Aerodynamic Instability and the Mechanism of Flutter." NACA Report 496, 1935.
- [32] Cardona, A. and M. Geradin, "Time Integration of the Equations of Motion in Mechanism Analysis," *Computers and Structures*, Vol. 33, No. 3, 1989, pp. 801–820.
- [33] Baumgarte, J., "Stabilization of Constraints and Integrals of Motion in Dynamic Systems," *Computer Methods in Applied Mechanics and Engineering*, 1972, pp. 1–16.
- [34] Dunn, P. and J. Dugundji, "Nonlinear Stall Flutter and Divergence Analysis of Cantilevered Graphite / Epoxy Wings," *AIAA Journal*, Vol. 30, No. 1, January 1992, pp. 153–162.
- [35] Dormand, J. R. and P. J. Prince, "A Family of Embedded Runge-Kutta Formulae," *J. Comp. Appl. Math.*, Vol. 6, No. 19-26, 1980.
- [36] Hénon, M., "On the Numerical Computation of Poincaré Maps," *Physica D*, Vol. 5, 1982, pp. 412–414.
- [37] Bisplinghoff, R. L., H. Ashley, and R. L. Halfman, *Aeroelasticity*, Addison-Wesley, 1955.
- [38] Welch, P. D., "The Use of Fast Fourier Transform for the Estimation of Power Spectra: A Method Based on Time Averaging Over Short, Modified Periodograms," *IEEE Trans. Audio Electroacoust.*, Vol. AU-1, June 1967, pp. 70–73.

Appendix A

WASP Wing Physical Characteristics

A.1 Wing Properties

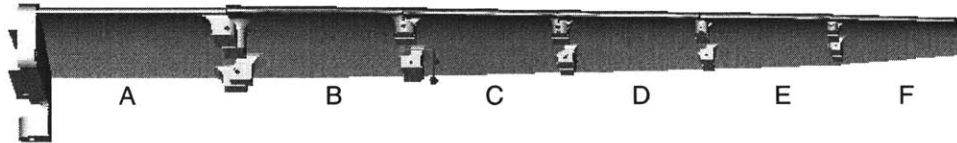


Figure A-1: WASP prototype wing

Table A.1: WASP Wing planform dimensions

Wing Segment	Root Chord (cm)	Tip Chord (cm)	Span (cm)
A	7.62	7.62	8.64
B	7.62	7.19	8.26
C	7.19	6.53	7.24
D	6.53	5.92	6.53
E	5.92	5.31	6.00
F	5.31	3.86	5.72
Total Span (cm)			42.36
Total Wing Area (cm ²)			277.0

Table A.2: Cross-sectional properties of WASP wing (referenced to elastic axis)

Property	root	75% span
m	1.203 kg/m	0.697 kg/m
I_α	3.05×10^{-4} kg m	1.04×10^{-4} kg m
r_α	0.418	0.418
S_α	1.80×10^{-3} kg	7.95×10^{-4} kg
x_α	0.0393	0.0393
μ	215.4	215.4
b	0.038 m	0.029 m
a	0.0225	0.0225
EI	232 N m ²	77.8 N m ²
GJ	18.0 N m ²	6.03 N m ²

Table A.3: Cross-sectional properties of one-hinge wing

m	0.883 kg/m
I_α	1.64×10^{-4} kg m
r_α	0.418
S_α	1.13×10^{-3} kg
x_α	0.0393
μ	215.4
b	0.0326 m
a	0.0225
EI	232 N m ²
GJ	18.0 N m ²

Table A.4: Experimentally obtained WASP wing hinge properties

Hinge	Spring Stiffness (Nm/rad)	Torque @ Switching Angle (Nm)	Static Friction (Nm)	Closed Stiffness (Nm/rad)	Closure Angle [†] (degrees)
A-B	155×10^{-3}	138×10^{-3}	109×10^{-3}	194	-0.1
B-C	74×10^{-3}	106×10^{-3}	23×10^{-3}	300	2.6
C-D	69×10^{-3}	59×10^{-3}	54×10^{-3}	200	-0.15
D-E	16×10^{-3}	5×10^{-3}	4×10^{-3}	130	0.5
E-F	5×10^{-3}	3×10^{-3}	1×10^{-3}	20	4.5

[†]Measured with respect to angle at which the two segments would be parallel, positive in the direction of upward bending.

A.2 Airfoil Coordinates

x/c	y/c	x/c	y/c
0.000000	0.000000	0.000000	0.000000
0.000230	0.002388	0.000228	-0.002388
0.001245	0.005756	0.001797	-0.005515
0.003109	0.009479	0.004939	-0.007261
0.005955	0.013686	0.010610	-0.007266
0.010076	0.018540	0.017697	-0.005980
0.015904	0.024201	0.025064	-0.004242
0.023890	0.030697	0.033766	-0.002010
0.034213	0.037803	0.044624	0.000836
0.046605	0.045086	0.057778	0.004278
0.060561	0.052159	0.072485	0.008092
0.075638	0.058806	0.088276	0.012154
0.091535	0.064948	0.104769	0.016360
0.108054	0.070566	0.121666	0.020610
0.125063	0.075670	0.138752	0.024820
0.142465	0.080282	0.155946	0.028936
0.160190	0.084425	0.173232	0.032922
0.178185	0.088125	0.190611	0.036751
0.196407	0.091407	0.208092	0.040402
0.214818	0.094296	0.225681	0.043871
0.233377	0.096809	0.243379	0.047172
0.252058	0.098962	0.261209	0.050284
0.270847	0.100770	0.279201	0.053204
0.289731	0.102249	0.297356	0.055921
0.308693	0.103414	0.315658	0.058416

x/c	y/c	x/c	y/c
0.327725	0.104279	0.334093	0.060671
0.346814	0.104854	0.352644	0.062671
0.365955	0.105155	0.371300	0.064397
0.385138	0.105190	0.390042	0.065833
0.404356	0.104970	0.408856	0.066964
0.423605	0.104505	0.427728	0.067772
0.442879	0.103805	0.446646	0.068243
0.462172	0.102878	0.465595	0.068361
0.481483	0.101733	0.484565	0.068116
0.500806	0.100377	0.503545	0.067515
0.520137	0.098818	0.522527	0.066580
0.539476	0.097064	0.541507	0.065333
0.558819	0.095121	0.560485	0.063793
0.578164	0.092995	0.579455	0.061984
0.597509	0.090693	0.598416	0.059927
0.616854	0.088222	0.617364	0.057642
0.636197	0.085586	0.636296	0.055153
0.655535	0.082792	0.655207	0.052476
0.674869	0.079846	0.674097	0.049625
0.694197	0.076754	0.692973	0.046607
0.713515	0.073520	0.711835	0.043437
0.732824	0.070148	0.730691	0.040121
0.752124	0.066642	0.749553	0.036670
0.771417	0.063007	0.768433	0.033094
0.790707	0.059248	0.787340	0.029399
0.810000	0.055369	0.806289	0.025596
0.829303	0.051379	0.825288	0.021692
0.848627	0.047285	0.844251	0.017717
0.867983	0.043098	0.863370	0.013640
0.887378	0.038837	0.882527	0.009488
0.906782	0.034521	0.901640	0.005275
0.926175	0.030154	0.920644	0.001020
0.945494	0.025765	0.939472	-0.003263
0.964322	0.021457	0.957803	-0.007498
0.981260	0.017559	0.974420	-0.011390
0.994192	0.014565	0.987333	-0.014451
1.002940	0.012528	0.996067	-0.016540

Appendix B

Foam Wing Characteristics

B.1 Wing Planform

Table B.1: Foam wing planform

Semi-Span	1.0 m
Chord	14.2 cm
Aspect Ratio	7

B.2 Cross-Sectional Properties

Table B.2: System parameters of experimental wings

Mass hinge	0.085 kg
S_α hinge	9.6×10^{-4} kg m
I_α hinge	1.6×10^{-5} kg m ²
m	0.4676 kg/m
I_α	6.14×10^{-4} kg m
r_α	0.5074
S_α	1.21×10^{-2} kg
x_α	0.3630
μ	23.8
b	0.0714 m
a	-0.50
EI	416 N m ²
GJ	3.46 N m ²

B.3 Accelerometer Properties

Table B.3: Properties of the Endevco 22 piezoelectric accelerometer

Mass	1.4×10^{-4} kg
Charge sensitivity	0.38 pC/g
Resonance frequency	54 kHz
Amplitude Response ($\pm 5\%$)	1 to 10,000 Hz
Amplitude Response (± 1 dB)	0.5 to 10,000 Hz
Transverse Sensitivity	$\leq 5\%$
Amplitude linearity to 500 g	1%

Appendix C

Experimental Test Matrices

C.1 WASP Wing

Table C.1: Steady state test matrix

	Speed (m/s)								
AOA	31.3	35.8	40.2	44.7	46.9	49.2	51.4	53.6	55.9
0°	x	x	x	x [†]	x	x	x	x	x
2°						x	x	x	x
4°	x	x	x	x	x	x	x	x	x
6°	x	x	x	x	x	x	x	x	x
8°	x [‡]	x	x	x	x	x	x	x	x

[†]Additional tests performed at -5° , -4° and -2° .

[‡]Additional tests performed at 9° , 10° , 11° , 12° and 13° .

Table C.2: Dynamic test matrix, variable speed, 0° AOA

	Speed (m/s)							
†	35.8	44.7	49.2	53.6	58.1	62.6	67.0	71.5
1	x	x	x	x	x	x	x [†]	x [§]
2	x	x	x	x	x	x	x	
3	x	x	x	x	x	x	x	

† Number of wing segments deflected.

‡ Normal deflection test and full deployment test.

§ Additional test done at 2° AOA.

Table C.3: Dynamic test matrix, variable AOA, speed of 45 m/s

	AOA (degrees)					
†	-4	-2	2	4	6	8
1	x	x	x	x	x	x
3		x				x

† Number of wing segments deflected.

C.2 Foam Wing

Due to the nature of the tests on the foam wings, no preset test matrix was defined. The following tables show the flight conditions at which data was recorded.

Table C.4: Test points of unhinged foam wing

Speed (m/s)	AOA (degrees)	Speed (m/s)	AOA (degrees)
17.9	-3	33.4	0
20.2	-3	34.5	0
22.2	-3	35.3	0
24.3	-3	35.6	0
27.2	-3	36.3	0
29.0	-3	37.0	0
31.3	-3	37.6	0
33.5	-3	38.3	0
34.1	-3	38.8 †	0
35.2	-3		
35.7	-3		
36.4	-3		
36.9	-3		
37.7	-3		
38.3	-3		

† Instability recorded.

Table C.5: Test points of one-hinged foam wing

Speed (m/s)	AOA (degrees)	Speed (m/s)	AOA (degrees)
17.5	-3.0	33.6	-2.5
19.7	-3.0	34.4	-2.5
21.6	-3.0	35.2	-2.5
24.1	-3.0	35.9	-2.5
26.2	-3.0	36.5	-2.5
28.5	-3.0	36.6	-2.5
31.0	-3.0	36.9	-2.5
31.8	-3.0	37.4	-2.5
33.7	-3.0	37.9 †	-2.5
34.0	-3.0	38.4	-2.5
34.6	-3.0	33.7	-2.0
35.0	-3.0	35.0	-2.0
35.4	-3.0	35.3	-2.0
35.5	-3.0	35.6	-2.0
35.8	-3.0	36.0	-2.0
36.4	-3.0	36.5 †	-2.0
36.5	-3.0	34.8	0.0
36.9	-3.0	37.4	0.0
37.0	-3.0	37.8	0.0
37.1 †	-3.0	38.0	0.0
37.6 †	-3.0	38.6 †	0.0
38.0	-3.0		
38.9	-3.0		

† Instability recorded.

Appendix D

Double Pendulum Solution

This appendix uses the multi-body dynamics presented in Chapter 3 to solve the double pendulum problem. In that chapter, the system of equations for a multi-body system takes onto the form of Equation 3.16

$$\begin{bmatrix} \mathbf{M} & \mathbf{C}_q^T \\ \mathbf{C}_q & 0 \end{bmatrix} \begin{Bmatrix} \ddot{\mathbf{q}} \\ \lambda \end{Bmatrix} = \begin{Bmatrix} \mathbf{Q}_v + \mathbf{Q} \\ -\mathbf{Q}_c - 2\boldsymbol{\Omega}\dot{\mathbf{C}} - \boldsymbol{\Omega}^2\mathbf{C} \end{Bmatrix} \quad (\text{D.1})$$

where the mass matrix \mathbf{M} for the double pendulum was found in Equation 3.10 to be:

$$\mathbf{M} = \begin{bmatrix} m_1 & 0 & -\frac{m_1 l_1 \sin \theta_1}{2} & 0 & 0 & 0 \\ 0 & m_1 & \frac{m_1 l_1 \cos \theta_1}{2} & 0 & 0 & 0 \\ -\frac{m_1 l_1 \sin \theta_1}{2} & \frac{m_1 l_1 \cos \theta_1}{2} & \frac{m_1 l_1^2}{3} & 0 & 0 & 0 \\ 0 & 0 & 0 & m_2 & 0 & -\frac{m_2 l_2 \sin \theta_2}{2} \\ 0 & 0 & 0 & 0 & m_2 & \frac{m_2 l_2 \cos \theta_2}{2} \\ 0 & 0 & 0 & -\frac{m_2 l_2 \sin \theta_2}{2} & \frac{m_2 l_2 \cos \theta_2}{2} & \frac{m_2 l_2^2}{3} \end{bmatrix} \quad (\text{D.2})$$

\mathbf{C}_q is found by using virtual displacement to find the variation of \mathbf{C} given in Equation

3.13 as

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \mathbf{q} + \begin{Bmatrix} 0 \\ 0 \\ -l_1 \cos \theta_1 \\ -l_1 \sin \theta_1 \end{Bmatrix} \quad (\text{D.3})$$

thus

$$\mathbf{C}_{\mathbf{q}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & l_1 \sin \theta_1 & 1 & 0 & 0 \\ 0 & 0 & -l_1 \cos \theta_1 & 0 & 1 & 0 \end{bmatrix} \quad (\text{D.4})$$

For the right side of Equation D.1, \mathbf{Q}_v was shown in Equation 3.12d to be

$$\mathbf{Q}_{v_i} = \begin{Bmatrix} \frac{m_1 l_1 \cos \theta_1 \dot{\theta}_1^2}{2} \\ \frac{m_1 l_1 \sin \theta_1 \dot{\theta}_1^2}{2} \\ 0 \\ \frac{m_2 l_2 \cos \theta_2 \dot{\theta}_2^2}{2} \\ \frac{m_2 l_2 \sin \theta_2 \dot{\theta}_2^2}{2} \\ 0 \end{Bmatrix} \quad (\text{D.5})$$

For the case of the double pendulum with gravity acting in the negative Y direction, Equation 3.20 shows that \mathbf{Q} will become

$$\mathbf{Q} = \begin{Bmatrix} 0 \\ -m_1 g \\ \frac{-m_1 g l_1}{2} \cos \theta_1 \\ 0 \\ -m_2 g \\ \frac{-m_2 g l_2}{2} \cos \theta_2 \end{Bmatrix} \quad (\text{D.6})$$

Q_c is determined by using the procedure of Equation 3.15, yielding

$$Q_c = \begin{Bmatrix} 0 \\ 0 \\ l_1 \cos \theta_1 \dot{\theta}_1^2 \\ l_1 \sin \theta_1 \dot{\theta}_1^2 \end{Bmatrix} \quad (D.7)$$

Finally the matrix Ω is found using the natural frequency of each pendulum leg, that is:

$$\Omega = \begin{bmatrix} \sqrt{\frac{g}{l_1}} & 0 & 0 & 0 \\ 0 & \sqrt{\frac{g}{l_1}} & 0 & 0 \\ 0 & 0 & \sqrt{\frac{g}{l_2}} & 0 \\ 0 & 0 & 0 & \sqrt{\frac{g}{l_2}} \end{bmatrix} \quad (D.8)$$

Gathering all the terms and using Equation D.1 yields

$$\begin{bmatrix} m_1 & 0 & -\frac{m_1 l_1 S(\theta_1)}{2} & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ & m_1 & \frac{m_1 l_1 C(\theta_1)}{2} & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ & & \frac{m_1 l_1^2}{3} & 0 & 0 & 0 & 0 & 0 & l_1 S(\theta_1) & -l_1 C(\theta_1) \\ & & & m_2 & 0 & -\frac{m_2 l_2 S(\theta_2)}{2} & 0 & 0 & 1 & 0 \\ & & & & m_2 & \frac{m_2 l_2 C(\theta_2)}{2} & 0 & 0 & 0 & 1 \\ & & & & & \frac{m_2 l_2^2}{3} & 0 & 0 & 0 & 0 \\ & & & & & & 0 & 0 & 0 & 0 \\ & \text{sym.} & & & & & & 0 & 0 & 0 \\ & & & & & & & & 0 & 0 \\ & & & & & & & & & 0 \end{bmatrix} \begin{Bmatrix} \ddot{X}_1 \\ \ddot{Y}_1 \\ \ddot{\theta}_1 \\ \ddot{X}_2 \\ \ddot{Y}_2 \\ \ddot{\theta}_2 \\ \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{Bmatrix} =$$

$$\left. \begin{aligned}
& \frac{m_1 l_1 C(\theta_1) \dot{\theta}_1^2}{2} \\
& \frac{m_1 l_1 S(\theta_1) \dot{\theta}_1^2}{2} - m_1 g \\
& -\frac{m_1 g l_1}{2} C(\theta_1) \\
& \frac{m_2 l_2 C(\theta_2) \dot{\theta}_2^2}{2} \\
& \frac{m_2 l_2 S(\theta_2) \dot{\theta}_2^2}{2} - m_2 g \\
& -\frac{m_2 g l_2}{2} C(\theta_2) \\
& -\frac{g X_1}{l_1} - 2 \dot{X}_1 \sqrt{\frac{g}{l_1}} \\
& -\frac{g Y_1}{l_1} - 2 \dot{Y}_1 \sqrt{\frac{g}{l_1}} \\
& -l_1 C(\theta_1) \dot{\theta}^2 - \frac{g}{l_2} (-l_1 C(\theta_1) + X_2) - 2(l_1 S(\theta_1) \dot{\theta}_1 + \dot{X}_2) \sqrt{\frac{g}{l_2}} \\
& -l_1 S(\theta_1) \dot{\theta}^2 - \frac{g}{l_2} (-l_1 S(\theta_1) + Y_2) - 2(l_1 C(\theta_1) \dot{\theta}_1 + \dot{Y}_2) \sqrt{\frac{g}{l_2}}
\end{aligned} \right\} \quad (\text{D.9})$$

where $S(\theta) \equiv \sin \theta$ and $C(\theta) \equiv \cos \theta$.

Appendix E

Matlab Scripts

The Matlab scripts and functions created to implement the analytical model are presented in this appendix. Figure E-1 illustrates how the various functions work together for the nonlinear analysis.

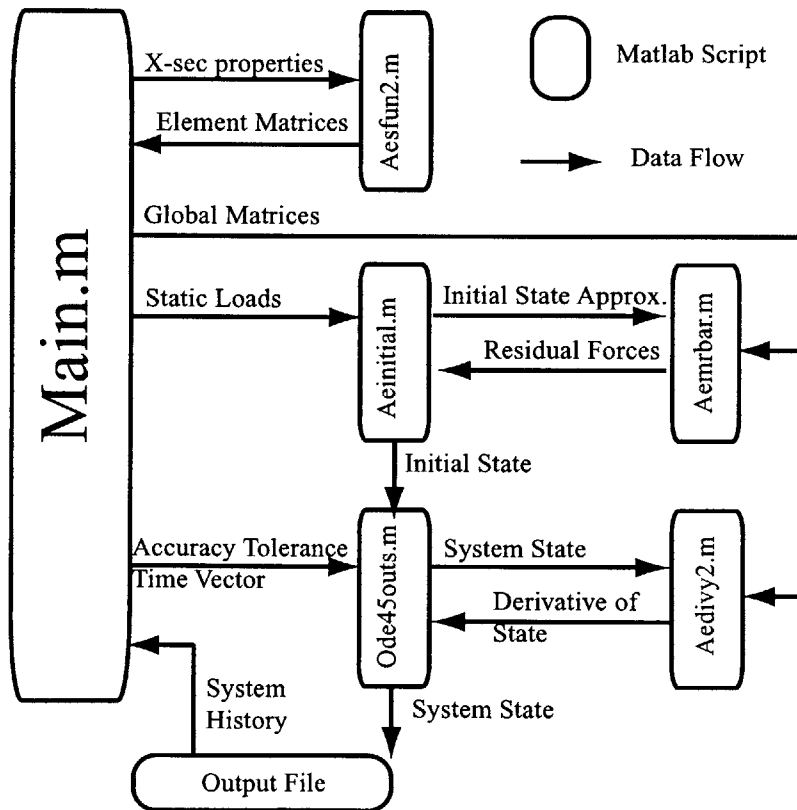


Figure E-1: Script logic diagram

E.1 Main.m

```

% Aeroelastic code for WASP-type wing.
% Written by Torrey Radcliffe, last worked on: Feb. 2001
% The mechanics are based on the book:
% Shabana, Ahmed A., "Dynamics of Multibody Systems: 2nd Edition,"
% Cambridge Press 1998
% The aerodynamic forces are based on the paper:
% Edwards, Ashley and Breakwell, "Unsteady Aerodynamic Modeling
% for Arbitrary Motions" AIAA Journal April 1979
clear
clear global
tic
global DAMP BETA MHINGE MAERR MFF KFF MAE KAE BAE DAE E1 ...
    E2 Fp I1 IBAR0 SBAR MFF1 SBAR1 STIL1 NBODY NELEM M L ...
    C N Q HINGE Ndoff Ndofb Y0 LIFT RLIFT FNAM
% NELEM is the number of FEM element per wing section
% NBODY is the number of wing sections, starting at the root
NELEM = 4;
NBODY = 6;
% Ndoff is the number of flexible DOF for each body (both dependent and
% independent). There are Ndofb degrees of freedom for each body,
% 3 for each node and three to describe the rigid body motion of each body.
% However two of the flexible DOF for each body are always constrained.
Ndoff = (NELEM+1)*3;
Ndofb = Ndoff+1;
N = (Ndofb)*NBODY;
Np = NBODY*(NELEM+1)*2;
% C is the wing length properties, one row for each section:
% initial chord, final chord, span data given in inches and
% converted to metric
C = [3.00    3.00    3.40;
    3.00    2.83    3.25;
    2.83    2.57    2.85;
    2.57    2.33    2.57;
    2.33    2.09    2.36;
    2.09    1.52    2.25].*2.54/100;
% Hinge Stiffness Properties in metric units
HINGE = [155.0210e-3  74.4193e-3  69.5623e-3  15.7922e-3  5.4885e-3;
    137.4096e-3  109.7764e-3  59.1013e-3  4.7010e-3  3.2915e-3;
    109.3506e-3  23.2463e-3  54.1398e-3  3.7830e-3  1.0043e-3;
    194 300 200 130 20];
% Hinge Mass
MH = 2*[0 7.646e-3 6.023e-3 3.643e-3 2.159e-3 1.308e-3];

```

```

MHINGE = MH(1:NBODY);
% Hinge offset angles
BETA = [-.1 2.6 -.15 .5 4.5]/180*pi;
%Hinge Damping constant
DAMP = .4;
% the following are x-section constants normalized by chord
ar = 3.744e-2;
rho = .1*0.4536/.0254^3;
E = 1.03e7*6894;
ei = 997.29*6894;
gj = 77.23*6894;
Is = 3.114e-3*rho;
In = 9.66929e-5*rho;
Qn = -1.6947e-3*rho;
Qs = -1.5983e-3*rho;
a = 2.2480e-002;
% Aerodynamic properties
Cl0 = .9771;
Cla = 6.3742;
Cm0 = 0.2519;
Cma = 3.025;
rhoas = 1.225;
Ts = 288.15;
% Time vector
t=0:5e-5:.35;
% creating output matrices
h = zeros(12,length(t)-1);
aa = h;
% begining of main loop, currently setup to run through all
% 13 experimental test cases
for ru = 1:13
% Names the output file
fclose('all');
FNAME = 'datarun.bin';
! del datarun.bin
% Conditions of each experimental test
T = [74.8 80.5 86.4 93.6 98.7 104.3 112.5 121.6 121.5 121.6 121.7 122]*.555+255.3;
rhoa = rhoas*Ts./T;
DP = [5.83 9.24 11.29 13.3 15.5 18.17 20.8 9.56 9.57 9.5 9.58 9.56]*.133.3;
alphr = [0 0 0 0 0 0 0 2 4 6 8];
u = (2.*DP./rhoa).^5;
dt = [.002 .002 .002 .002 .002 .002 .002 .024 .032 .018 .014 .006];
hm = [5.6422 5.75 5.90 6.02 6.1 6.2 6.3 5.95 6.05 6.15 6.25 6.29];
h1 = [5.1178 5.769 5.4685 6.227 4.6898 4.9554 4.8416 3.0468 2.7734 ...

```

```

    3.1796 3.5703 4.3515];
h2 = [5.3516 5.8245 5.777 6.2533 5.0103 5.30468 5.1265 5.9072 6.027 ...
    5.0156 4.5659 4.4921];
h3 = [.6 .3 .3 .6 .5 .5 .5 .5 .5 .5 .5]*pi/4+pi/4;
h4 = h3(ru);
hmean = hm(ru)*2.54/100;
h0 = h1(ru)*2.54/100;
h0dot = (h2(ru)-h1(ru))/dt(ru)*2.54/100;
U = u(ru);
alpharoot = alphi(ru)/180*pi;
rhoa = rhoa(ru);
% Initalizing all the property matrices
M = zeros(NELEM,NBODY);
L = M;
MFF = zeros(Ndoff*NBODY,Ndoff);
KFF = zeros(Ndoff*NBODY,Ndoff);
Q = zeros(NELEM*NBODY*6,4);
MAE = zeros(NELEM*NBODY*4,4);
KAE = zeros(NELEM*NBODY*4,4);
BAE = zeros(NELEM*NBODY*4,4);
DAE = zeros(NELEM*NBODY*4,4);
E1 = zeros((NELEM+1)*NBODY*2,2);
E2 = zeros((NELEM+1)*NBODY*2,2);
Fp = zeros((NELEM+1)*NBODY*2,2);
LIFT = zeros((NELEM+1)*NBODY*2,1);
MAERR = zeros(NBODY,NELEM);
trash = zeros(6);
Mhinge = zeros(6);
% this next section sets up the structural elements. They can't be finalized
% as they are dependent on the state of the system
for i = 1:NBODY
    Mhinge(1,1) = MHINGE(i);
    for k = 1:NELEM
        % first the geomtric and mass properties of each element
        % c is mean chord length of element i,k
        c = C(i,1)+(2*k-1)/(2*NELEM)*(C(i,2)-C(i,1));
        l = C(i,3)/NELEM; L(k,i) = l;
        m = ar*rho*c^2*l; M(k,i) = m;
        b1 = (C(i,1)+(k-1)/NELEM*(C(i,2)-C(i,1)))/2;
        b2 = (C(i,1)+k/NELEM*(C(i,2)-C(i,1)))/2;
        m1 = (b1*2)^2*ar*rho; m2 = (b2*2)^2*ar*rho;
        x0 = sum(L(1:k,i))-l;
        eik = ei*c^4; gjk = gj*c^4; Isk = Is*c^4/l^2; Ink = In*c^4/l^2;
        Qnk=Qn*c^3/l; Qsk = Qs*c^3/l;

```

```

% This function (aesfun2) fills all the property matrices
% The matrices are filled element by element, and those
% that are constant are put in their final states here
[q, Mff, Kff, Maerr, Mae, Bae, Kae, Dae, e1, e2, fp, ...
    I1((((i-1)*NELEM+k)*2-1):((i-1)*NELEM+k)*2,:), ...
    IBAR0((((i-1)*NELEM+k)*6-5):((i-1)*NELEM+k)*6,:), ...
    SBAR((((i-1)*NELEM+k)*2-1):((i-1)*NELEM+k)*2,:), ...
    MFF1((((i-1)*NELEM+k)*6-5):((i-1)*NELEM+k)*6,:), ...
    SBAR1((((i-1)*NELEM+k)*2-1):((i-1)*NELEM+k)*2,:), ...
    STIL1((((i-1)*NELEM+k)*2-1):((i-1)*NELEM+k)*2,:)] = ...      140
aesfun2(m,m1,m2,l,a,b1,b2,rhoa,U,x0,eik,gjk,Ink,Isk,Qnk, ...
    Qsk,k,MHINGE(i));
B = zeros(Ndoff,6);
B((k-1)*3+1:(k-1)*3+6,:) = eye(6);
if k == 1
    MFF((i-1)*Ndoff+1:i*Ndoff,:) = ...
        MFF((i-1)*Ndoff+1:i*Ndoff,:)+B*(Mff+Mhinge)*B';
else
    MFF((i-1)*Ndoff+1:i*Ndoff,:) = ...
        MFF((i-1)*Ndoff+1:i*Ndoff,:)+B*Mff*B';      150
end
KFF((i-1)*Ndoff+1:i*Ndoff,:) = KFF((i-1)*Ndoff+1:i*Ndoff,:)+B*Kff*B';
Q((i-1)*NELEM*6+(k-1)*6+1:(i-1)*NELEM*6+(k-1)*6+6,:) = q;
MAE((i-1)*NELEM*4+(k-1)*4+1:(i-1)*NELEM*4+(k-1)*4+4,:) = Mae;
KAE((i-1)*NELEM*4+(k-1)*4+1:(i-1)*NELEM*4+(k-1)*4+4,:) = Kae;
BAE((i-1)*NELEM*4+(k-1)*4+1:(i-1)*NELEM*4+(k-1)*4+4,:) = Bae;
DAE((i-1)*NELEM*4+(k-1)*4+1:(i-1)*NELEM*4+(k-1)*4+4,:) = Dae;
E1((i-1)*(NELEM+1)*2+(k-1)*2+1:(i-1)*(NELEM+1)*2+(k-1)*2+2,:) = e1;
E2((i-1)*(NELEM+1)*2+(k-1)*2+1:(i-1)*(NELEM+1)*2+(k-1)*2+2,:) = e2;
Fp((i-1)*(NELEM+1)*2+(k-1)*2+1:(i-1)*(NELEM+1)*2+(k-1)*2+2,:) = fp; 160
LIFT((i-1)*(NELEM+1)*2+(k-1)*2+1:(i-1)*(NELEM+1)*2+(k-1)*2+2,:)...
    = .5*rhoa*U^2.*[2*b1*(Cl0+Cla*alpharoot); ...
        (2*b1)^2*(Cm0+Cma*alpharoot)];
MAERRs(i,k) = Maerr;
end
[q, Mff, Kff, Maerr, Mae, Bae, Kae, Dae, e1, e2, fp, ...
    I1((((i-1)*NELEM+k)*2-1):((i-1)*NELEM+k)*2,:), ...
    IBAR0((((i-1)*NELEM+k)*6-5):((i-1)*NELEM+k)*6,:), ...
    SBAR((((i-1)*NELEM+k)*2-1):((i-1)*NELEM+k)*2,:), ...
    trash, SBAR1((((i-1)*NELEM+k)*2-1):((i-1)*NELEM+k)*2,:), ...      170
    STIL1((((i-1)*NELEM+k)*2-1):((i-1)*NELEM+k)*2,:)] = ...
aesfun2(m,m2,m2,l,a,b2,b2,rhoa,U,x0,eik,gjk,Ink,Isk,Qnk,Qsk,k,MHINGE(i));
E1((i-1)*(NELEM+1)*2+(NELEM)*2+1: ...
    (i-1)*(NELEM+1)*2+(NELEM)*2+2,:) = e1;

```

```

E2((i-1)*(NELEM+1)*2+(NELEM)*2+1:..
    (i-1)*(NELEM+1)*2+(NELEM)*2+2,:) = e2;
Fp((i-1)*(NELEM+1)*2+(NELEM)*2+1:..
    (i-1)*(NELEM+1)*2+(NELEM)*2+2,:) = fp;
LIFT((i-1)*(NELEM+1)*2+(NELEM)*2+1:..
    (i-1)*(NELEM+1)*2+(NELEM)*2+2,:) = ...
    .5*rho*U^2.*[2*b2*(Cl0+Cl*alpharoot)]; (2*b2)^2*(Cm0+Cma*alpharoot)];
end
ltip = C(NBODY,3)
% y is the state matrix, where each colum is the state at a given time.
% the first N (# number of DOF) is the positional value of y and the
% second N is related to the change in y over time. The final Np states
% are used in the unsteady aerodynamic equations.
%
% Within the first N values, the first three are the rigid DOF for the
% first body, the next NELEM*3+1 values are the DOF of the first body.
% the next three are the rigid DOF of the second body, etc.
% the second N values follow a similar pattern.
%
% First the steady state values of y must be found using aeinitial
Y0 = zeros(2*N+Np,1);
Y0(1:N) = aeinitial;
y0 = Y0;
% This is where the system is distrubed from the steady state
y0((NBODY-1)*Ndofb+3) = Y0((NBODY-1)*Ndofb+3)-h4;
% ode45outs is the numerical integration scheme
% aedivy2 is the file which finds the temporal
% differentiation of the system of a given state
[T,Y]=ode45outs('aedivy2',t,y0,1e-6,1e-6);
toc
% here the output file is read
% this method is used to reduce the memory needed to
% run the simulation
fid = fopen(FNAME,'r');
ym = fread(fid,[2*N+Np+1,inf],'single');
h(ru,:) = (ym(38,:)+C(6,3)*sin(ym(39,:)))*100/2.54;;
aa(ru,:) = ym(N,:);
tt((p-1)*3+r,:) = tm(1,:);
save runsm4
end

```

E.2 Aesfun2.m

```
function [Q, Mff, Kff, Maerr, Mae, Bae, Kae, Dae, E1, E2, ...
```

```

Fp, I1, Ibar0 ,Sbar, Mff1, Sbar1, Still] = ...
aesfun(m,m1,m2,l,a,b1,b2,rho,U,x0,ei,gj,In,Is,Qn,Qs,k,mhinge)
%create Mass and Stiffness matrices for an element;
S11 = [6*Is*l/5      0 Is*l^2/10  -6*Is*l/5      0 Is*l^2/10;
        0            0 0          0              0 0;
        Is*l^2/10    0 2*Is*l^3/15 -l^2*Is/10    0 -l^3*Is/30;
        -6*Is*l/5    0 -l^2*Is/10 6*Is*l/5      0 -Is*l^2/10;
        0            0 0          0              0 0;
        Is*l^2/10    0 -l^3*Is/30 -Is*l^2/10    0 2*Is*l^3/15]
S22 = [ 13*m/35      -7*l^2*Qs/20  11*m*l/210 ...
        9*m/70       -3*l^2*Qs/20  -13*m*l/420;
        -7*l^2*Qs/20 l^3*In/3    -l^3*Qs/20 ...
        -3*l^2*Qs/20 l^3*In/6    l^3*Qs/30;
        11*m*l/210  -l^3*Qs/20    m*l^2/105 ...
        13*m*l/420  -l^3*Qs/30    -m*l^2/140;
        9*m/70     -3*l^2*Qs/20  13*m*l/420 ...
        13*m/35    -7*l^2*Qs/20  -11*m*l/210;
        -3*l^2*Qs/20 l^3*In/6    -l^3*Qs/30 ...
        -7*l^2*Qs/20 l^3*In/3    l^3*Qs/20;
        -13*m*l/420 l^3*Qs/30    -m*l^2/140 ...
        -11*m*l/210 l^3*Qs/20    m*l^2/105]
S33 = [
        0 0      0 0 0      0;
        0 l^3*Is/3 0 0 l^3*Is/6 0;
        0 0      0 0 0      0;
        0 0      0 0 0      0;
        0 l^3*Is/6 0 0 l^3*Is/3 0;
        0 0      0 0 0      0];
if k == 1
Sbar = [l*Qn  m+mhinge/2;
        0    -l^2/2*Qs;
        0    m*l/12;
        -Qn*l  m/2;
        0    -l^2/2*Qs;
        0    -m*l/12]';
else
Sbar = [l*Qn  m/2;
        0    -l^2/2*Qs;
        0    m*l/12;
        -Qn*l  m/2;
        0    -l^2/2*Qs;
        0    -m*l/12]';
end
Mff = S11+S22+S33;

```

```

B = [1 0 0 0 0 0; 0 1 0 0 0 0; 0 0 0 1 0 0; 0 0 0 0 1 0];
Q = l*[1/3 0 1/6 0;
      0 1/3 0 1/6;
      0 0 0 0;
      1/6 0 1/3 0;
      0 1/6 0 1/3;
      0 0 0 0];
R1 = [-2 ; 2*(a+.5)*b1];
R2 = [-2 ; 2*(a+.5)*b2];
S1 = [0 1];
S21 = [1/b2 (.5-a)];
S22 = [1/b2 (.5-a)];
Mae1 = (pi*rho*b1^3)*[-1/b1 a; a -(1/8+a^2)*b1];
Mae2 = (pi*rho*b2^3)*[-1/b2 a; a -(1/8+a^2)*b2];
Bae1 = (pi*rho*b1^3)*U/b1*[0 -1; 0 -(1/2-a)*b1]+.5*R1*S21];
Bae2 = (pi*rho*b2^3)*U/b2*[0 -1; 0 -(1/2-a)*b2]+.5*R2*S22];
Kae1 = (pi*rho*b1^3)*(U/b1)^2*(.5*R1*S1);
Kae2 = (pi*rho*b2^3)*(U/b2)^2*(.5*R2*S1);
Dae1 = (pi*rho*b1^3)*(U/b1)^2*R1*[.006825*U/b1 0.10805];
Dae2 = (pi*rho*b2^3)*(U/b2)^2*R2*[.006825*U/b2 0.10805];
Mae = [Mae1 zeros(2); zeros(2) Mae2];
Bae = [Bae1 zeros(2); zeros(2) Bae2];
Kae = [Kae1 zeros(2); zeros(2) Kae2];
Dae = [Dae1 zeros(2); zeros(2) Dae2];
Maerr = l*(Mae1(1,1)+Mae2(1,1))/2;
E1 = U/b1*[ 0 0; S1];
E2 = [0 0; S21];
Fp = [ 0 1; -0.01365*(U/b1)^2 -.3455*U/b1];
Mff = Mff-Q*Mae*B;
Kff = [12*ei/l^3 0 6*ei/l^2 -12*ei/l^3 0 6*ei/l^2;
      0 gj/l 0 0 -gj/l 0;
      6*ei/l^2 0 4*ei/l -6*ei/l^2 0 2*ei/l;
      -12*ei/l^3 0 -6*ei/l^2 12*ei/l^3 0 -6*ei/l^2;
      0 -gj/l 0 0 gj/l 0;
      6*ei/l^2 0 2*ei/l -6*ei/l^2 0 4*ei/l];
I1 = m*l*[(x0/l+1)^2/2-(x0/l)^2/2;0];
Ibar0 = m*l*[(x0/l+1)^2/2-(x0/l+1)^3/3-(x0/l)^2/2+(x0/l)^3/3; ...
      0; 0; (x0/l+1)^3/3-(x0/l)^3/3; 0; 0]
Mff1 = m.*[
      1/3 0 0 1/6 0 0
      0 13/35 11*l/210 0 9/70 -13*l/420
      0 11*l/210 l^2/105 0 13*l/420 -l^2/140
      1/6 0 0 1/3 0 0
      0 9/70 13*l/420 0 13/35 -11*l/210

```

```

    0    -13*l/420    -l^2/140    0    -11*l/210    l^2/105];
Sbar1 = m/12.*[6 0 0 6 0 0; 0 6 1 0 6 -l]
Still = [
    7*m/20    -l^2/3*Qs    m*l/20    3*m/20    -l^2/6*Qs    -m*l/30;
    3*m/20    -l^2/6*Qs    m*l/30    7*m/20    -l^2/3*Qs    -m*l/20];
if k ==1
    Mff1(1,1) = Mff1(1,1)+mhinge;
    Mff1(2,2) = Mff1(2,2)+mhinge;
    Sbar1(1) = Sbar1(1)+mhinge;
end
end

```

E.3 Aeinital.m

```

function [y0] = aeinitial()
% this function finds the steady state of the system
% with known parameters under given static loads g and LIFT
global BETA MHINGE MRR MFF KFF I1 IBAR0 SBAR MFF1 SBAR1 ...
    STIL1 NBODY NELEM M L C N HINGE Ndoft Ndoftb LIF
% gravitational acceleration
g = 9.8
nbody = NBODY;
% first the hinge angles are approximated
% this is an iterative process which itself
% is first intialized
H = zeros(NBODY-1);
G = zeros(NBODY-1,1);
if NELEM==1
    l = L; m=M+MHINGE;
else
    l = sum(L); m = sum(M)+MHINGE;
end
lift = zeros(NBODY-1,1);
for i = 1:NBODY-1
    lift(i) = LIFT((i)*(NELEM+1)*2+1)*l(i+1);
end
for i = 1:NBODY-1
    if i == NBODY-1
        G(i,1) = l(i+1)*(lift(i)-g*m(i+1))/2+HINGE(2,i);
    else
        G(i,1) = l(i+1)*((lift(i)/2-g*m(i+1)/2)+(sum(lift(i+1:NBODY-1)) ...
            - g*sum(m(i+2:NBODY))))+HINGE(2,i)-HINGE(2,i+1);
    end
end
end

```



```

for i = 1:NBODY-1
  if G(i)>0
    if i==1
      H(i,i) = HINGE(4,i);
    else
      H(i,i) =HINGE(4,i)+H(i,i);
      H(i-1,i) = -HINGE(4,i)+H(i-1,i);
      H(i,i-1) = -HINGE(4,i)+H(i,i-1);
      H(i-1,i-1) = HINGE(4,i)+H(i-1,i-1);
    end
    if i == NBODY-1
      G(i,1) = l(i+1)*(lift(i)-g*m(i+1))/2+HINGE(2,i);
    else
      G(i,1) = l(i+1)*((lift(i)-g*m(i+1))/2+(sum(lift(i+1:NBODY-1)) ...
        + g*sum(m(i+2:NBODY))))+HINGE(2,i)-HINGE(2,i+1);
    end
  else
    if i==1
      H(i,i) = HINGE(1,i);
    else
      H(i,i) =HINGE(1,i)+H(i,i);
      H(i-1,i) = -HINGE(1,i)+H(i-1,i);
      H(i,i-1) = -HINGE(1,i)+H(i,i-1);
      H(i-1,i-1) = HINGE(1,i)+H(i-1,i-1);
    end
    if i == NBODY-1
      G(i,1) = l(i+1)*(lift(i)-g*m(i+1)/2)+HINGE(2,i);
    else
      G(i,1) = l(i+1)*(lift(i)/2+sum(lift(i+1:NBODY-1)) ...
        - g*(m(i+1)/2+sum(m(i+2:NBODY))))+HINGE(2,i)-HINGE(2,i+1);
    end
  end
end
y1 = H\G;
y1 = [0;y1];
y = 0;
k = 1;
% this is the iteration cycle
while k<100
  y2(:,k) = y1;
  y = mean(y2,2);
  H = zeros(NBODY-1);
  for i = 1:NBODY-2
    if G(i)-H(i,i+1)*(y(i+2)-y(i+1))>0

```

40

50

60

70

```

    if i==1
        H(i,i) = HINGE(4,i);
    else
        H(i,i) =HINGE(4,i)+H(i,i);
        H(i-1,i) = -HINGE(4,i)+H(i-1,i);
        H(i,i-1) = -HINGE(4,i)+H(i,i-1);
        H(i-1,i-1) = HINGE(4,i)+H(i-1,i-1);
    end
    G(i,1) = l(i+1)*(lift(i)/2+sum(lift(i+1:NBODY-1)) ...
    - g*(m(i+1)/2+sum(m(i+2:NBODY))))+HINGE(2,i)-HINGE(2,i+1);
else
    if i==1
        H(i,i) = HINGE(1,i);
    else
        H(i,i) =HINGE(1,i)+H(i,i);
        H(i-1,i) = -HINGE(1,i)+H(i-1,i);
        H(i,i-1) = -HINGE(1,i)+H(i,i-1);
        H(i-1,i-1) = HINGE(1,i)+H(i-1,i-1);
    end
    G(i,1) = l(i+1)*(lift(i)/2+sum(lift(i+1:NBODY-1)) ...
    - cos(y(i+1))*g*(m(i+1)/2+sum(m(i+2:NBODY)))) ...
    + HINGE(2,i)-HINGE(2,i+1);
end
end
i = NBODY-1;
if G(i)>0
    H(i,i) =HINGE(4,i)+H(i,i);
    H(i-1,i) = -HINGE(4,i)+H(i-1,i);
    H(i,i-1) = -HINGE(4,i)+H(i,i-1);
    H(i-1,i-1) = HINGE(4,i)+H(i-1,i-1);
    G(i,1) = l(i+1)*(lift(i)-g*m(i+1))/2-HINGE(2,i);
else
    H(i,i) =HINGE(1,i)+H(i,i);
    H(i-1,i) = -HINGE(1,i)+H(i-1,i);
    H(i,i-1) = -HINGE(1,i)+H(i,i-1);
    H(i-1,i-1) = HINGE(1,i)+H(i-1,i-1);
    G(i,1) = l(i+1)*(lift(i)-cos(y(i+1))*g*m(i+1)/2)+HINGE(2,i);
end
end
y1 = H\G;
y1 = [0;y1];
k = k+1;
end
% theta is a vector of hinge angles
theta = y+[0;cumsum(BETA)'];

```

```

% the rest of the system's states are added
% with zero elastic deformation assumed
y0 = zeros(N,1);
for i = 1:NBODY-1
    y0(i*Ndofb+1)=C(i,3)*cos(y0((i-1)*Ndofb+3))+y0((i-1)*Ndofb+1);
    y0(i*Ndofb+2)=C(i,3)*sin(y0((i-1)*Ndofb+3))+y0((i-1)*Ndofb+2);
    y0(i*Ndofb+3)=theta(i+1);
end
% Rbar is a vector of residual forces based upon the state of
% the system found by aemrbar, for a system in equilibrium Rbar will be zero
Rbar = aemrbar(y0);
nind = length(Rbar);
b = 1;
% qi is a vector of independent DOFs (qd would be the dependent)
qi = zeros(nind,1);
% an iterative cycle to determine the intial state
% of all DOFs by reducing Rbar
while b>1e-2
    k = 0;
    for i =1:nind
        y1 = y0;
        if i <=Ndofb-4
            y1(i+4) = y0(i+4)+1e-7;
            qi(i) = y0(i+4);
        elseif rem(i,Ndofb-3)==0
            k = k+1;
            y1(k*Ndofb+3) = y0(k*Ndofb+3)+1e-7;
            qi(i) = y0(k*Ndofb+3);
        else
            y1(k*Ndofb+4+rem(i,Ndofb-3)) = ...
                y0(k*Ndofb+4+rem(i,Ndofb-3))+1e-7;
            qi(i) = y0(k*Ndofb+4+(rem(i,Ndofb-3)));
        end
        dRbar(:,i) = (aemrbar(y1)-Rbar)./1e-7;
    end
    dq1 = dRbar\(-Rbar);
    qi = qi+dqi;
    % findy is a function which determines the entire state vector
    % based on the state of qi
    y0 = findy(qi);
    Rbar = aemrbar(y0);
    b = sum(abs(Rbar));
end

```

E.4 Aemrbar.m

```

function [Rbar] = aemrbar(y)
% this function finds Rbar, the residual forces on the system.
% for a detailed explanation please see aediv2
global BETA MHINGE Q MRR MFF KFF I1 IBAR0 SBAR MFF1 SBAR1 ...
      STIL1 NBODY NELEM M L N HINGE Ndoft Ndofb LIFT
g = 9.8
nbody = NBODY;
for i=1:NBODY
    theta(i) = y((i-1)*Ndofb+3);
end
beta = zeros(nbody,1); torque = zeros(nbody+1,1);
for k = 1:nbody-1
    beta(k+1) = y((k-1)*Ndofb+3)+y(k*Ndofb)-y(k*Ndofb+3)-y(k*Ndofb+5);
    if beta(k+1) < -BETA(k)
        torque(k+1) = HINGE(4,k)*(beta(k+1)+BETA(k))+HINGE(2,k);
    else
        torque(k+1) = HINGE(1,k)*(beta(k+1)+BETA(k))+HINGE(2,k);
    end
end
for i = 1:NBODY
    A = [cos(theta(i)) -sin(theta(i)); sin(theta(i)) cos(theta(i))];
    Qsi = zeros(Ndoft+3,1);
    Qai = zeros(Ndoft+3,1);
    if i == 1
        B2 = zeros(Ndofb,Ndoft+3);
        B2(1:3,1:3) = eye(3);
        B2(4,5) = 1;
        B2(5:Ndoft,7:Ndoft+3) = eye(Ndoft-3);
    else
        B2 = zeros(Ndoft,Ndoft+3);
        B2(1:3,1:3) = eye(3);
        B2(4:Ndoft-2,5:Ndoft) = eye(Ndoft-4);
        B2(Ndoft-1:Ndoft,Ndoft+2:Ndoft+3) = eye(2);
    end
for k = 1:NELEM
    m = M(k,i); l = L(k,i); lb = sum(L(1:k,i));
    liftk = LIFT((i-1)*(NELEM+1)*2+(k-1)*2+1:(i-1)* ...
        (NELEM+1)*2+(k-1)*2+4,1);
    P1 = liftk(1); M1 = liftk(2); P2 = liftk(3); M2 = liftk(4);
    q = Q((i-1)*NELEM*6+(k-1)*6+1:(i-1)*NELEM*6+(k-1)*6+6,:);
    Qar = l*(P1+P2)/2*[-sin(theta(i)); cos(theta(i))];
    Qat = 1/l*((P2-P1)*(lb^3-(lb-1)^3)/3+ ...

```

```

        (P1*lb-P2*(lb-l))*(lb^2-(lb-l)^2)/2);
    Qaf = q*[P1; M1; P2; M2];
    Qsr = [0; -m*g];
    Qsf = m*g*cos(theta(i)).*[-1/2; 0 ; -l*(1/20+1/30); -1/2; 0; l*(1/20+1/30)];
    Qst = -g*m*(lb-l/2)*cos(theta(i));
    if k == 1
        Qsr(2) = Qsr(2)-g*MHINGE(i);
        Qsf(1) = Qsf(1)-MHINGE(i)*g*cos(theta(i));
    end
    B = zeros(Ndoff+3,9);
    B(1:3,1:3) = eye(3);
    B(k*3+1:k*3+6,4:9) = eye(6);
    Qak = B*[Qar; Qat; Qaf];
    Qai = Qai+Qak;
    Qsk = B*[Qsr; Qst; Qsf];
    Qsi = Qsi+Qsk;
end
Ki = zeros(Ndoff+3);
Kffi = KFF((i-1)*Ndoff+1:i*Ndoff,:);
Ki(4:Ndoff+3,4:Ndoff+3) = Kffi;
KT(Ndofb*(i-1)+1:Ndofb*i,Ndofb*(i-1)+1:Ndofb*i) = B2*Ki*B2';
QaT(Ndofb*(i-1)+1:Ndofb*i,1) = B2*Qai;
QsT(Ndofb*(i-1)+1:Ndofb*i,1) = B2*Qsi;
%add hinge force
QsT(Ndofb*(i-1)+3,1) = QsT(Ndofb*(i-1)+3,1)+torque(i)-torque(i+1);
QsT(Ndofb*(i-1)+5,1) = QsT(Ndofb*(i-1)+5,1)+torque(i);
QsT(Ndofb*i,1) = QsT(Ndofb*i,1)-torque(i+1);
end
%constraint equations
Bi = zeros(N,N-4-3*(NBODY-1));
Bd = zeros(N,4+3*(NBODY-1));
Cq = zeros(3*nbody+1,N);
Qc = zeros(3*nbody+1,1);
fc = Qc;
W = zeros(3*nbody+1);
Cq(1,1) = 1; Cq(2,2) = 1; Cq(3,3) = 1; Cq(4,4) = 1;
Bi(5:Ndofb,1:Ndofb-4) = eye(Ndofb-4);
Bd(1:4,1:4) = eye(4);
Lb = sum(L(:,1));
Cq(5,Ndofb+1) = 1;
Cq(6,Ndofb-2) = -1; Cq(6,Ndofb+2) = 1;
Cq(7,Ndofb-1) = -1; Cq(7,Ndofb+4) = 1;
Bi(Ndofb+3,Ndofb-3) = 1;
Bi(Ndofb+5:2*Ndofb,Ndofb-2:2*Ndofb-7) = eye(Ndofb-4);

```

```

Bd(Ndofb+1:Ndofb+2,5:6) = eye(2); Bd(Ndofb+4,7) = 1;
phi = Cq;
for i = 3:nbody
    Lb = sum(L(:,i-1));
    Cq(3*(i-1)+1+1,(i-2)*Ndofb+1) = -1; Cq(3*(i-1)+1+1,(i-2)*Ndofb+3) = ...
        Lb*sin(theta(i-1)); Cq(3*(i-1)+1+1,(i-1)*Ndofb+1) = 1;
    Cq(3*(i-1)+1+2,(i-2)*Ndofb+2) = -1; Cq(3*(i-1)+1+2,(i-2)*Ndofb+3) = ...
        -Lb*cos(theta(i-1)); Cq(3*(i-1)+1+2,(i-1)*Ndofb+2) = 1;
    Cq(3*(i-1)+1+3,(i-1)*Ndofb-1) = -1; Cq(3*(i-1)+1+3,(i-1)*Ndofb+4) = 1;
    Bi((i-1)*Ndofb+3,(i-1)*(Ndofb-3)-1+1)=1; Bi((i-1)*Ndofb+5:i*Ndofb,(i-1) ...
        *(Ndofb-3)-1+2:i*(Ndofb-3)-1) = eye(Ndofb-4);
    Bd((i-1)*Ndofb+1:(i-1)*Ndofb+2,(i-1)*3+1+1:(i-1)*3+1+2) = eye(2); ...
        Bd((i-1)*Ndofb+4,i*3+1)=1;
    phi(3*(i-1)+1+1,(i-2)*Ndofb+1) = -1; phi(3*(i-1)+1+2,(i-1)*Ndofb+1) = 1;
    phi(3*(i-1)+1+2,(i-2)*Ndofb+2) = -1; phi(3*(i-1)+1+2,(i-1)*Ndofb+2) = 1;
    phi(3*(i-1)+1+3,(i-1)*Ndofb-1) = -1; phi(3*(i-1)+1+3,(i-1)*Ndofb+4) = 1;
    fc(3*(i-1)+1+1) = -Lb*cos(theta(i-1));
    fc(3*(i-1)+1+2) = -Lb*sin(theta(i-1));
end
Cqd = Cq*Bd;
Cqi = Cq*Bi;
B = [Bi Bd];
Cdi = -inv(Cqd)*Cqi;
Rbar = -(B'*(QsT+QaT-KT*y))'*[eye(N-4-3*(NBODY-1));Cdi];
Rbar = Rbar';

```

90

100

110

E.5 Ode45outs.m

```

function [tout,yout] = ode45outs(odefile,tspan,y0,atol,rtol)
%MODIFIED FOR THE PURPOSES OF TORREY RADCLIFFE
% ODE45 is an implementation of the explicit Runge-Kutta (4,5) pair of
% Dormand and Prince called variously RK5(4)7FM, DOPRI5, DP(4,5) and DP54.
% It uses a "free" interpolant of order 4 communicated privately by
% Dormand and Prince. Local extrapolation is done.
% Details are to be found in The MATLAB ODE Suite, L. F. Shampine and
% M. W. Reichelt, SIAM Journal on Scientific Computing, 18-1, 1997.
% Mark W. Reichelt and Lawrence F. Shampine, 6-14-94
% Copyright (c) 1984-98 by The MathWorks, Inc.
% $Revision: 5.56 $ $Date: 1998/05/14 11:28:39 $
global TRUE FALSE NBODY NELEM N Ndofb FNAME BETA
TRUE = 1;
FALSE = ~TRUE;
true = 1;

```

10

```

false = ~true;
fid = fopen(FNAME,'a');
nsteps = 0; % stats
nfailed = 0; % stats
nfevals = 0; % stats
npds = 0; % stats
ndecomps = 0; % stats
nsolves = 0; % stats
t0 = 1e-14;
next = 2;
hstate = zeros(NBODY-1,1);
t = t0;
y = y0(:);
neq = length(y);
tfinal = max(tspan);
tmax = tfinal;
ntspan= length(tspan);
% waitbar2 allows the progress of the integration
% to be monitored on the screen
wait = waitbar2(0,'progress');
% Get options, and set defaults.
threshold = atol / rtol;
% By default, hmax is 1/10 of the interval.
hmax = min(abs(tfinal-t),0.1*(tfinal-t));
nout = 1;
tout(nout) = t;
yout(nout,:) = y.';
next =2;
% Initialize method parameters.
pow = 1/5;
A = [1/5; 3/10; 4/5; 8/9; 1; 1];
B = [
    1/5      3/40   44/45  19372/6561   9017/3168   35/384
    0       9/40   -56/15  -25360/2187  -355/33     0
    0       0     32/9   64448/6561  46732/5247  500/1113
    0       0     0     -212/729   49/176     125/192
    0       0     0     0         -5103/18656 -2187/6784
    0       0     0     0         0          11/84
    0       0     0     0         0          0
];
E = [71/57600; 0; -71/16695; 71/1920; -17253/339200; 22/525; -1/40];
f = zeros(neq,7);
dt = zeros(1,7);
[f0,dt0] = feval(odefile,t,y,hstate);

```

```

nfevals = nfevals + 1;           % stats           60
[m,n] = size(f0);
hmin = 16*eps*abs(t);
% Compute an initial step size h using y'(t).
absh = min(hmax, abs(tmax*atol - t));
rh = norm(f0 ./ max(abs(y),threshold),inf) / (0.8 * rtol^pow);
if absh * rh > 1
    absh = 1 / rh;
end
absh = max(absh, hmin);
f(:,1) = f0;                     70
dt(1) = dt0;
% THE MAIN LOOP
done = false;
while ~done
    % By default, hmin is a small number such that t+hmin is only slightly
    % different than t. It might be 0 if t is 0.
    hmin = 16*eps*abs(t);
    absh = min(hmax, max(hmin, absh)); % couldn't limit absh until new hmin
    h = absh;
    % Stretch the step if within 10% of tfinal-t.           80
    if 1.1*absh >= abs(tfinal - t)
        h = tfinal - t;
        absh = abs(h);
        done = true;
    end
    % LOOP FOR ADVANCING ONE STEP.
    nofailed = true;           % no failed attempts
    while true
        hA = h * A;
        hB = h * B;           90
        [f(:,2),dt(2)] = feval(odefile,t,y+f*hB(:,1),hstate);
        [f(:,3),dt(3)] = feval(odefile,t,y+f*hB(:,2),hstate);
        [f(:,4),dt(4)] = feval(odefile,t,y+f*hB(:,3),hstate);
        [f(:,5),dt(5)] = feval(odefile,t,y+f*hB(:,4),hstate);
        [f(:,6),dt(6)] = feval(odefile,t,y+f*hB(:,5),hstate);
        tnew = t + hA(6);
        ynew = y + f*hB(:,6);
        if sum(imag(ynew))~=0
            t
        end           100
        [f(:,7),dt(7)] = feval(odefile,tnew,ynew,hstate);
        nfevals = nfevals + 6;           % stats
        htrue = find(hstate);

```



```

if length(htrue) ~= FALSE
    %this checks to see if the solution is in the "Henon" mode
    % this has to be checked before the normal error checking
    % as this mode will take negative steps
    %estimate the error
err = abs(ynew(htrue(1)*Ndofb+3)+ynew(htrue(1)*Ndofb+5)- ...
    ynew((htrue(1)-1)*Ndofb+3)-ynew(htrue(1)*Ndofb)+BETA(htrue(1))); 110
    % see below
if err > rtol
    nfailed = nfailed+1;
    if absh <= hmin
        msg = sprintf(['Failure at t=%e. Unable to meet integration ' ...
            'tolerances without reducing the step size below ' ...
            'the smallest value allowed (%e) at time t.\n'],t,hmin);
        warning(msg);
    end
    if nfailed
        nfailed = false;
        absh = max(hmin, abs(htemp) * max(0.1, 0.8*(rtol/err)^pow));
    else
        absh = max(hmin, 0.5 * abs(htemp));
    end
    h = absh;
    hstate(htrue(1)) = 0;
    done = false;
else
    hstate(htrue(1)) = 0;
    tnew = t+dt*hB(:,6);
    [f0,dt0] = feval(odefile,tnew,ynew,hstate);
    h = tnew-t; absh = abs(h);
    f = ftemp;
    f(:,7) = f0;
    dt(1) = dt0;
    break;    %sucsessfull Henon step
end
else
    % Estimate the error.
err = absh * norm((f * E) ./ max(max(abs(y),abs(ynew)),threshold),inf);
    %check for change in hinge state
for i = 1:NBODY-1
    if abs(y(i*Ndofb+3)+y(i*Ndofb+5)-y((i-1)*Ndofb+3)- ...
        y(i*Ndofb)+BETA(i))>1e-15
        hstate(i) = sign(ynew(i*Ndofb+3)+ynew(i*Ndofb+5)-...
            ynew((i-1)*Ndofb+3)-ynew(i*Ndofb)+BETA(i))- ...

```

```

        sign(y(i*Ndofb+3)+y(i*Ndofb+5)-y((i-1)*Ndofb+3)- ...
        y(i*Ndofb)+BETA(i));
end
end
htrue = find(hstate);
waitbar2(tnew/tmax,wait);
% Accept the solution only if the weighted error is no more than the
% tolerance rtol. Estimate an h that will yield an error of rtol on
% the next step or the next try at taking this step, as the case may be,
% and use 0.8 of this value to avoid failures.
if err > rtol                                % Failed step
    nfailed = nfailed + 1;                    % stats
    if absh <= hmin                            160
        msg = sprintf(['Failure at t=%e. Unable to meet integration ' ...
            'tolerances without reducing the step size below ' ...
            'the smallest value allowed (%e) at time t.\n'],t,hmin);
        warning(msg);
    end
    if nfailed
        nfailed = false;
        absh = max(hmin, absh * max(0.1, 0.8*(rtol/err)^pow));
    else
        absh = max(hmin, 0.5 * absh);          170
    end
    end
    h = absh;
    hstate = zeros(NBODY-1,1);
    done = false;
elseif length(htrue) == 1                    %change in state
    htemp = h; ftemp = f;
    hmin = 16*eps*abs(y(htrue(1)*Ndofb+3)+y(htrue(1)*Ndofb+5)- ...
        y((htrue(1)-1)*Ndofb+3)-y(htrue(1)*Ndofb)+BETA(htrue(1)));
    h = sign(y(htrue(1)*Ndofb+3+N)+y(htrue(1)*Ndofb+5+N)- ...
        y((htrue(1)-1)*Ndofb+3+N)-y(htrue(1)*Ndofb+N))* ...
        abs(y(htrue(1)*Ndofb+3)+y(htrue(1)*Ndofb+5) ...
        -y((htrue(1)-1)*Ndofb+3)-y(htrue(1)*Ndofb)+BETA(htrue(1)));
    if h == 0
        h = 1e-14;
    end
    absh = min(hmax, max(hmin, abs(h)));
    [f0,dt0] = feval(odefile,t,y,hstate);
    f(:,1) = f0;
    dt(1) = dt0;
    if tnew <= tmax                            190
        done = false;

```

```

    else
        done = true;
    end
elseif length(htrue) > 1 %if two hinges change state, decrease h
    nfailed = nfailed + 1; % stats
    if absh <= hmin
        msg = sprintf(['Failure at t=%e. Unable to meet integration ' ...
            'tolerances without reducing the step size below ' ...
            'the smallest value allowed (%e) at time t.\n'],t,hmin); 200
        warning(msg);
    end
    if nfailed
        nfailed = false;
        absh = max(hmin, absh * max(0.1, 0.8*(rtol/err)^pow));
    else
        absh = max(hmin, 0.5 * absh);
    end
    h = absh
    hstate = zeros(NBODY-1,1); 210
    done = false;
else % Successful step
    break;
end
end
end
nsteps = nsteps + 1; % stats
if nargout > 0
    oldnout = nout;
    % output only at tspan points 220
    while next <= nspan
        if (tnew - tspan(next)) < 0
            break;
        elseif tnew == tspan(next)
            nout = nout + 1;
            fwrite(fid,tnew,'single');
            fwrite(fid,ynew,'single');
            tout = tnew;
            yout = ynew.';
            next = next + 1; 230
            break;
        end
        nout = nout + 1; % tout and yout are already allocated
        tout = tspan(next);
        yout = ntrp45(tspan(next),t,y,[],[],h,f).';
    end
end

```

```

        fwrite(fid,tout,'single');
        fwrite(fid,yout,'single');
        next = next + 1;
    end
end
% If there were no failures compute a new h.
if nofailed
    % Note that absh may shrink by 0.8, and that err may be 0.
    temp = 1.25*(err/rtol)^pow;
    if temp > 0.2
        absh = absh / temp;
    else
        absh = 5.0*absh;
    end
end
% Advance the integration one step.
t = tnew;
y = ynew;
f(:,1) = f(:,7);           % Already evaluated odefile(tnew,ynew)
dt(1) = dt(7);
end
if nargout > 0
    tout = tout;
    yout = yout;
end
fclose(fid);

```

240

250

260

E.6 Aedivy2.m

```

function [ydot,dt] = aedivy(t,y,hstate,dt)
% this gives the differential of the states y at time t
% hstate determines if henon calcs need to be done as
% one of the hinges is in a transitional state
global DAMP BETA MHINGE MAERR MFF KFF MAE KAE BAE DAE ...
    E1 E2 Fp I1 IBAR0 SBAR MFF1 SBAR1 STIL1 NBODY NELEM ...
    M L N Q HINGE Ndoff Ndofb Y0 LIF
ydot = zeros(length(y),1);
g = 9.8
dy = y(1:N)-Y0(1:N);
nbody = NBODY;
htrue = find(hstate);
% this checks to see if the system is in a normal integration step
% or a Henon step

```

10

```

if length(htrue) ~=0
    hvelocity = 1/(y(htrue(1)*Ndofb+3+N)+y(htrue(1)*Ndofb+5+N) ...
        - y((htrue(1)-1)*Ndofb+3+N)-y(htrue(1)*Ndofb+N));
    if hvelocity == Inf
        hvelocity = 1;
    end
else
    hvelocity = 1;
end
% beta is a vector containing the angles of the hinges where beta(n)
% is the angle of the N-1 body with the nth body, eg. beta = pi is the
% state of a folded wing, and beta(1). The kstate for each
% hinge: 1 = soft, 3 = hard (open all the way) is found for debugging
beta = zeros(nbody,1); betadot = beta; kstate = beta; torque = ...
    zeros(nbody+1,1);
for k = 1:nbody-1
    beta(k+1) = y((k-1)*Ndofb+3)+y(k*Ndofb)-y(k*Ndofb+3)-y(k*Ndofb+5);
    betadot(k+1) = y((k-1)*Ndofb+3+N)+y(k*Ndofb+N)-y(k*Ndofb+3+N) ...
        - y(k*Ndofb+5+N);
    if beta(k+1) < -BETA(k)
        kstate(k+1) = 3;
        torque(k+1) = HINGE(4,k)*(beta(k+1)+BETA(k))+HINGE(2,k) ...
            + DAMP*HINGE(3,k)*betadot(k+1);
    else
        kstate(k+1) = 1;
        torque(k+1) = HINGE(1,k)*(beta(k+1)+BETA(k))+HINGE(2,k) ...
            + DAMP*HINGE(3,k)*betadot(k+1);
    end
end
MT = zeros(N);
KT = zeros(N);
QsT = zeros(N,1);
QvT = zeros(N,1);
h1 = 0; h2 = 0;
% This loop determines the mass and forcing matrices for the system
% Here i is the current body and j is the current element being calculated
for i = 1:nbody
    % theta is the rigid body rotation angle
    theta(i) = y((i-1)*Ndofb+3);
    thetadot(i) = y((i-1+nbody)*Ndofb+3);
    A = [cos(theta(i)) -sin(theta(i)); sin(theta(i)) cos(theta(i))];
    At = [-sin(theta(i)) -cos(theta(i)); cos(theta(i)) -sin(theta(i))];
    sint = sin(theta(i));
    cost = cos(theta(i));

```

```

% Ef and Oef are the elastic states
Ef = zeros(Ndoff,1); Oef = zeros(Ndoff,1);
Efdot = zeros(Ndoff,1);
Hdot = zeros(NELEM+1,1);
% recall that the rootmost hinges has a clamped free condition
% while the rest are pinned pinned
if i == 1
    Ef(2) = y((i-1)*Ndofb+4);
    Ef(4:Ndoff) = y((i-1)*Ndofb+5:i*Ndofb) ;
    Oef(2) = Y0((i-1)*Ndofb+4);
    Oef(4:Ndoff) = Y0((i-1)*Ndofb+5:i*Ndofb) ;
    EFdot(2) = y((i-1+nbody)*Ndofb+4);
    Efdot(4:Ndoff) = y((i-1+nbody)*Ndofb+5:(i+nbody)*Ndofb);
    B2 = zeros(Ndofb,Ndoff+3);
    B2(1:3,1:3) = eye(3);
    B2(4,5) = 1;
    B2(5:Ndofb,7:Ndoff+3) = eye(Ndoff-3);
else
    Ef(2:Ndoff-3) = y((i-1)*Ndofb+4:i*Ndofb-2);
    Ef(Ndoff-1:Ndoff) = y(i*Ndofb-1:i*Ndofb);
    Oef(2:Ndoff-3) = Y0((i-1)*Ndofb+4:i*Ndofb-2);
    Oef(Ndoff-1:Ndoff) = Y0(i*Ndofb-1:i*Ndofb);
    Efdot(2:Ndoff-3) = y((i-1+nbody)*Ndofb+4:(i+nbody)*Ndofb-2);
    Efdot(Ndoff-1:Ndoff) = y((i+nbody)*Ndofb-1:(i+nbody)*Ndofb);
    B2 = zeros(Ndofb,Ndoff+3);
    B2(1:3,1:3) = eye(3);
    B2(4:Ndofb-2,5:Ndoff) = eye(Ndoff-4);
    B2(Ndofb-1:Ndofb,Ndoff+2:Ndoff+3) = eye(2);
end
Mrr_i = zeros(2);
Mtti = 0;
Mrti = zeros(2,1);
Mtri = zeros(1,2);
Mrfi = zeros(2,Ndoff);
Mfri = zeros(Ndoff,2);
Mtfi = zeros(1,Ndoff);
Mfti = zeros(Ndoff,1);
Qai = zeros(Ndoff+3,1);
Qsi = zeros(Ndoff+3,1);
Qvi = zeros(Ndoff+3,1);
for k = 1:NELEM
    m = M(k,i); l = L(k,i);
    e0 = zeros(6,1);
    ef = Ef(3*k-2:3*k+3);

```

```

oef = Oef(3*k-2:3*k+3);
e = e0+ef;
efdot = zeros(6,1);
efdot = Efdot(3*k-2:3*k+3);
xp = zeros(4,1);
xp = y(2*N+(i-1)*(NELEM+1)*2+(k-1)*2+1: ...
      2*N+(i-1)*(NELEM+1)*2+(k-1)*2+4);
e01 = zeros(6,1);
e01(4) = sum(L(1:k,i));
e01(1) = e01(4)-1;
ef1 = zeros(6,1);
efdot1 = zeros(6,1);
ef1(2:3) = Ef(3*k-2:2:3*k); ef1(5:6) = Ef(3*k+1:2:3*k+3);
efdot1(2:3) = Efdot(3*k-2:2:3*k); efdot1(5:6) = Efdot(3*k+1:2:3*k+3);
e1 = e01+ef1;
q = Q((i-1)*NELEM*6+(k-1)*6+1:(i-1)*NELEM*6+(k-1)*6+6,:);
%state dependent mass elements with aerodynamic mass included
mae = MAE((i-1)*NELEM*4+(k-1)*4+1:(i-1)*NELEM*4+(k-1)*4+4,:);
if k == 1
    mrr = [m+MHINGE(i) 0; 0 m+MHINGE(i)]-1/2*(mae(1,1)+mae(3,3)) ...
          * [-sint; cost]*[-sint cost];
else
    mrr = [m 0; 0 m]-1/2*(mae(1,1)+mae(3,3))*[-sint; cost]*[sint cost];
end
mtt = e1'*MFF1((((i-1)*NELEM+k)*6-5):((i-1)*NELEM+k)*6, :) * e1 ...
      - 1/l*((mae(3,3)*e01(4)-mae(1,1)*e01(1))*(e01(4)^3-e01(1)^3)/3 ...
      + (mae(1,1)*e01(1)*e01(4)-mae(3,3)*e01(1)*e01(4)) ...
      * (e01(4)^2-e01(1)^2)/2);
mrt = At*SBAR1((((i-1)*NELEM+k)*2-1):((i-1)*NELEM+k)*2, :) * e1;
mrf = A*SBAR((((i-1)*NELEM+k)*2-1):((i-1)*NELEM+k)*2, :) ;
mtf = [e1(1) e1(4)]*STIL1((((i-1)*NELEM+k)*2-1):((i-1)*NELEM+k)*2, :) ;
mtr = mrt'-1/l*((mae(3,3)-mae(1,1))*(e01(4)^3-e01(1)^3)/3+(mae(1,1) ...
      * e01(4)-mae(3,3)*e01(1))*(e01(4)^2-e01(1)^2)/2)*[-sint; cost]';
mrt = mrt'-1/2*(e01(1)*mae(1,1)+e01(4)*mae(3,3))*[-sint; cost];
mfr = mrf'-q*[mae(1,1); mae(2,1); mae(3,3); mae(4,3)]*[-sint cost];
mrf = mrf'-1/2*[-sint; cost]*[mae(1,1) mae(1,2) 0 mae(3,3) mae(3,4) 0];
mft = mtf'-q*[e01(1)*mae(1,1); e01(1)*mae(2,1); e01(4)*mae(3,3); ...
      e01(4)*mae(4,3)];
mtf = mtf'-1/l*[((e01(4)^3-e01(1)^3)/3+(e01(4)^2-e01(1)^2)/2*e01(4)) ...
      *mae(1,1); ((e01(4)^3-e01(1)^3)/3+(e01(4)^2-e01(1)^2)/2*e01(4)) ...
      *mae(1,2); 0; ...
      ((e01(4)^3-e01(1)^3)/3+(e01(4)^2-e01(1)^2)/2*e01(1))*mae(3,3); ...
      ((e01(4)^3-e01(1)^3)/3+(e01(4)^2-e01(1)^2)/2*e01(4))*mae(3,4); 0]';
% Gravitational loads

```

```

Qsr = [0; -m*g];
Qsf = m*g*cos(theta(i)).*[-1/2; 0 ; -l*(1/20+1/30); -1/2; 0; ...
    l*(1/20+1/30)];
Qst = m*g*(sin(theta(i)).*[-1/2; 0 ; -l*(1/20+1/30); -1/2; 0; ...
    l*(1/20+1/30)]' *ef-(e01(1)+(e01(4)-e01(1))/2)*cost);
if k ==1
    Qsr(2) = Qsr(2)-g*MHINGE(i);
    Qsf(1) = Qsf(1)-MHINGE(i)*g*cos(theta(i));
end
%external aero forces
% First the absolute plunge motion must be determined
Hd1 = A(:,2)' * ([y(N+(i-1)*Ndofb+1);y(N+(i-1)*Ndofb+2)])+ ...
    At*[e01(1);0]*thetadot(i)+efdot(1);
Hd2 = A(:,2)' * ([y(N+(i-1)*Ndofb+1);y(N+(i-1)*Ndofb+2)])+ ...
    At*[e01(4);0]*thetadot(i)+efdot(4);
% the both steady and unsteady aero loads at each node are calculated
liftk = LIFT((i-1)*(NELEM+1)*2+(k-1)*2+1: ...
    (i-1)*(NELEM+1)*2+(k-1)*2+4) ...
    +KAE((i-1)*NELEM*4+(k-1)*4+1:(i-1)*NELEM*4+(k-1)*4+4,:) ...
    *[0;ef(2)-oef(2);0;ef(5)-oef(5)]+...
    BAE((i-1)*NELEM*4+(k-1)*4+1:(i-1)*NELEM*4+(k-1)*4+4,:)...
    *[Hd1;efdot(2);Hd2;efdot(5)]+...
    DAE((i-1)*NELEM*4+(k-1)*4+1:(i-1)*NELEM*4+(k-1)*4+4,:)*xp;
P1 = liftk(1); M1 = liftk(2); P2 = liftk(3); M2 = liftk(4);
% The aero forcing terms are found
Qar = l*(P1+P2)/2*[-sint; cost];
Qat = 1/l*((P2-P1)*(e01(4)^3-e01(1)^3)/3+(P1*e01(4)-P2*e01(1)) ...
    *(e01(4)^2-e01(1)^2)/2);
Qaf = q*[P1; M1; P2; M2];
%can solve for derivative of aerodynamic states
ydot(2*N+(i-1)*(NELEM+1)*2+(k-1)*2+1: ...
    2*N+(i-1)*(NELEM+1)*2+(k-1)*2+2)...
    =hvelocity*(E1((i-1)*(NELEM+1)*2+(k-1)*2+1:(i-1)*(NELEM+1)*2+ ...
    (k-1)*2+2,:)*[0;ef(2)-oef(2)]+ ...
    E2((i-1)*(NELEM+1)*2+(k-1)*2+1:(i-1)*(NELEM+1)*2+(k-1)*2+2,:) ...
    *[Hd1;efdot(2)]+Fp((i-1)*(NELEM+1)*2+(k-1)*2+1:(i-1)*(NELEM+1)*2 ...
    +(k-1)*2+2,:)*xp(1:2,1));
%quadratic velocity vector
Qvr = thetadot(i)^2*A*(SBAR((((i-1)*NELEM+k)*2-1): ...
    ((i-1)*NELEM+k)*2,:)*ef+I1((((i-1)*NELEM+k)*2-1): ...
    ((i-1)*NELEM+k)*2,:))-2*thetadot(i) ...
    *At*SBAR((((i-1)*NELEM+k)*2-1):((i-1)*NELEM+k)*2,:)*efdot;
Qvt = -2*thetadot(i)*efdot'*(MFF((((i-1)*(NELEM+1)+k-1)*3+1): ...
    ((i-1)*(NELEM+1)+k)*3+3),(k-1)*3+1:k*3+3)*ef);

```



```

Qvf = thetadot(i)^2*(MFF((((i-1)*(NELEM+1)+k-1)*3+1):...
    ((i-1)*(NELEM+1)+k)*3+3),(k-1)*3+1:k*3+3)*ef+ ...
    IBAR0((((i-1)*NELEM+k)*6-5):((i-1)*NELEM+k)*6,:));
B = zeros(Ndoff+3,9);
B(1:3,1:3) = eye(3);
B(k*3+1:k*3+6,4:9) = eye(6);
B1 = zeros(6,Ndoff);
B1(:,(k-1)*3+1:(k-1)*3+6) = eye(6);
Qsk = B*[Qsr; Qst; Qsf];
Qak = B*[Qar; Qat; Qaf];
Qvk = B*[Qvr; Qvt; Qvf];
% the element mass and forces are added to the body matrices
Mrri = Mrri+mrr;
Mtti = Mtti+mtt;
Mrti = Mrti+mrt;
Mtri = Mtri+mtr;
Mrfi = Mrfi+mrf*B1;
Mfri = Mfri+B1'*mfr;
Mtfi = Mtfi+mtf*B1;
Mfti = Mfti+B1'*mft;
Qsi = Qsi+Qsk;
Qai = Qai+Qak;
Qvi = Qvi+Qvk;
end
Mi = zeros(Ndoff+3);
Ki = zeros(Ndoff+3);
Mffi = MFF((i-1)*Ndoff+1:i*Ndoff,:);
Kffi = KFF((i-1)*Ndoff+1:i*Ndoff,:);
Mi = [Mrri Mrti Mrfi; Mtri Mtti Mtfi; Mfri Mfti Mffi];
Ki(4:Ndoff+3,4:Ndoff+3) = Kffi;
% The body matrices are incorporated into the global
KT(Ndofb*(i-1)+1:Ndofb*i,Ndofb*(i-1)+1:Ndofb*i) = B2*Ki*B2';
MT(Ndofb*(i-1)+1:Ndofb*i,Ndofb*(i-1)+1:Ndofb*i) = B2*Mi*B2';
QsT(Ndofb*(i-1)+1:Ndofb*i,1) = B2*Qsi;
QaT(Ndofb*(i-1)+1:Ndofb*i,1) = B2*Qai;
QvT(Ndofb*(i-1)+1:Ndofb*i,1) = B2*Qvi;
%add hinge force
%QsT(Ndofb*(i-1)+3,1) = QsT(Ndofb*(i-1)+3,1)+torque(i);
QsT(Ndofb*(i-1)+3,1) = QsT(Ndofb*(i-1)+3,1)+torque(i)-torque(i+1);
QsT(Ndofb*(i-1)+5,1) = QsT(Ndofb*(i-1)+5,1)+torque(i);
QsT(Ndofb*i,1) = QsT(Ndofb*i,1)-torque(i+1);
%find aero states for final node
ydot(2*N+(i-1)*(NELEM+1)*2+NELEM*2+1: ...
    2*N+(i-1)*(NELEM+1)*2+NELEM*2+2)...

```

```

    =hvelocity*(E1((i-1)*(NELEM+1)*2+NELEM*2+1: ...
    (i-1)*(NELEM+1)*2+NELEM*2+2,:)*[0;ef(5)-oef(5)]+ ...
    E2((i-1)*(NELEM+1)*2+NELEM*2+1:(i-1)*(NELEM+1)*2+NELEM*2+2,:)* ...
    [Hd2;efdot(5)]+Fp((i-1)*(NELEM+1)*2+NELEM*2+1: ...
    (i-1)*(NELEM+1)*2+NELEM*2+2,:)*xp(3:4,1));
end
% the total forcing vector is found
QT = QsT+QaT+QvT-KT*y(1:N);
%constraint equations
Cq = zeros(3*nbody+1,N);
Qc = zeros(3*nbody+1,1);
fc = Qc;
W = zeros(3*nbody+1);
Cq(1,1) = 1; Cq(2,2) = 1; Cq(3,3) = 1; Cq(4,4) = 1;
Lb = sum(L(:,1));
Cq(5,Ndofb+1) = 1;
Cq(6,Ndofb-2) = -1; Cq(6,Ndofb+2) = 1;
Cq(7,Ndofb-1) = -1; Cq(7,Ndofb+4) = 1;
phi = Cq;
fc(5) = -Lb;
W(1,1) = (g/Lb)^.5; W(2,2) = (g/Lb)^.5; W(3,3)=(g/Lb)^.5; W(4,4) = ...
(MT(Ndofb-1,Ndofb-1)/KT(Ndofb-1,Ndofb-1))^ .5;
for i = 3:nbody
    Lb = sum(L(:,i-1));
    Cq(3*(i-1)+1+1,(i-2)*Ndofb+1) = -1; Cq(3*(i-1)+1+1,(i-2)*Ndofb+3) = ...
    Lb*sin(theta(i-1)); Cq(3*(i-1)+1+1,(i-1)*Ndofb+1) = 1;
    Cq(3*(i-1)+1+2,(i-2)*Ndofb+2) = -1; Cq(3*(i-1)+1+2,(i-2)*Ndofb+3) = ...
    -Lb*cos(theta(i-1)); Cq(3*(i-1)+1+2,(i-1)*Ndofb+2) = 1;
    Cq(3*(i-1)+1+3,(i-1)*Ndofb-1) = -1; Cq(3*(i-1)+1+3,(i-1)*Ndofb+4) = 1;
    Qc(3*(i-1)+1+1)=Lb*cos(theta(i-1))*thetadot(i-1)^2;
    Qc(3*(i-1)+1+2)=Lb*sin(theta(i-1))*thetadot(i-1)^2;
    phi(3*(i-1)+1+1,(i-2)*Ndofb+1) = -1; phi(3*(i-1)+1+1,(i-1)*Ndofb+1) = 1;
    phi(3*(i-1)+1+2,(i-2)*Ndofb+2) = -1; phi(3*(i-1)+1+2,(i-1)*Ndofb+2) = 1;
    phi(3*(i-1)+1+3,(i-1)*Ndofb-1) = -1; phi(3*(i-1)+1+3,(i-1)*Ndofb+4) = 1;
    fc(3*(i-1)+1+1) = -Lb*cos(theta(i-1));
    fc(3*(i-1)+1+2) = -Lb*sin(theta(i-1));
    W(3*(i-2)+1+1,3*(i-2)+1+1) = (g/Lb)^.5;
    W(3*(i-2)+1+2,3*(i-2)+1+2) = (g/Lb)^.5;
    W(3*(i-2)+1+3,3*(i-2)+1+3) = (MT(Ndofb*(i-2)-1, ...
    Ndofb*(i-2)-1)/KT(Ndofb*(i-2)-1,Ndofb*(i-2)-1))^ .5;
end
W(3*(nbody-1)+1+1,3*(nbody-1)+1+1) = sum(L(:,nbody));
W(3*(nbody-1)+1+2,3*(nbody-1)+1+2) = sum(L(:,nbody));
W(3*(nbody-1)+1+3,3*(nbody-1)+1+3) = (MT(Ndofb*(nbody-1)-1, ...

```

```

    Ndofb*(nbody-1)-1)/KT(Ndofb*(nbody-1)-1,Ndofb*(nbody-1)-1))^ .5;
C = phi*y(1:N)+fc;
Cdot = Cq*y(N+1:2*N);
W = 1*W.^ .5;
% the system matrices are formed and inverted
b = [QT; -Qc-2.*W*Cdot-W.^2*C];
a = [MT Cq'; Cq zeros(3*nbody+1)];
x = a\b;
lam= x(N+1:N+3*nbody+1)
Qdotdot=x(1:N);
temp = hvelocity.*[zeros(N) eye(N) zeros(N,1); zeros(N,2*N+1); ...
    zeros(1,2*N) 1/t]*[y(1:2*N);t]+hvelocity.*[zeros(N,1); Qdotdot;0];
ydot(1:2*N) = temp(1:2*N);
dt = temp(2*N+1);

```

E.7 Flex.m

```

%Linear code
%Initally very similar to main.m
clear
clear global
tic
global DAMP BETA MHINGE MAERR MFF KFF MAE KAE BAE DAE E1 ...
    E2 Fp I1 IBAR0 SBAR MFF1 SBAR1 STIL1 NBODY NELEM M L ...
    C N Q HINGE Ndoff Ndofb Y0 LIFT FNAME P
% NELEM is the number of FEM element per wing section
% NBODY is the number of wing sections, starting at the root
NELEM = 1;
NBODY = 6;
% Ndoff is the number of flexible DOF for each body (both dependent and
% independent). There are Ndofb degrees of freedom for each body,
% 3 for each node and three to discribe the rigid body motion of each body.
% however two of the flexible DOF for each body are always
Ndoff = (NELEM+1)*3;
Ndofb = Ndoff+1;
N = (Ndofb)*NBODY;
Np = NBODY*(NELEM+1)*2;
% C is the wing length properties, one row for each section:
% intial chord, final chord, span data given in inches and converted
C = [3.00    3.00    3.40;
    3.00 2.83    3.25;
    2.83 2.57    2.85;
    2.57 2.33    2.57;

```

```

    2.33 2.09 2.36;
    2.09 1.52 2.25].*2.54/100;
% the following are x-section constants normalized by chord
ar = 3.744e-2;
rho = .1*0.4536/.0254^3;
E = 1.03e7*6894;
ei = 997.29*6894;
gj = 77.23*6894;
Is = 3.114e-3*rho;
In = 9.66929e-5*rho;
Qn = -1.6947e-3*rho;
Qs = -1.5983e-3*rho;
%Qn = 0;
%Qs = 0;
a = 2.2480e-002;
%Cl0 = .9771;
%Cla = 6.1742;
%Cm0 = 0.2519;
%Cma = 3.025;
Cl0 = 0; Cla = 0; Cm0 = 0; Cma = 0;
rhoas = 0;
Ts = 288.15;
DAMP = .4;
S = 2;
ru = 1;
alpharoot = 0;
rhoa = 1.225;
%This for loop sets which speeds the code will look at
for sti = 1:250
U = sti;
NELEM = 2;
NBODY = 6;
Ndoff = (NELEM+1)*3;
Ndofb = Ndoff+1;
N = (Ndofb)*NBODY;
Np = NBODY*(NELEM+1)*2
HINGE = [155.0210e-3 74.4193e-3 69.5623e-3 15.7922e-3 5.4885e-3;
137.4096e-3 109.7764e-3 59.1013e-3 4.7010e-3 3.2915e-3;
109.3506e-3 23.2463e-3 54.1398e-3 3.7830e-3 1.0043e-3;
1*ei*C(2:6,1)' .^4];
HINGE(4,:) = [194 300 200 130 20];
MH = 2*[0 7.646e-3 6.023e-3 3.643e-3 2.159e-3 1.308e-3];
MHINGE = MH(1:NBODY);
BETA = zeros(1,NBODY-1);

```

```

M = zeros(NELEM,NBODY);
L = M;
MFF = zeros(Ndoff*NBODY,Ndoff);
KFF = zeros(Ndoff*NBODY,Ndoff);
Q = zeros(NELEM*NBODY*6,4);
MAE = zeros(NELEM*NBODY*4,4);
KAE = zeros(NELEM*NBODY*4,4);
BAE = zeros(NELEM*NBODY*4,4);
DAE = zeros(NELEM*NBODY*4,4);
E1 = zeros((NELEM+1)*NBODY*2,2);
E2 = zeros((NELEM+1)*NBODY*2,2);
Fp = zeros((NELEM+1)*NBODY*2,2);
LIFT = zeros((NELEM+1)*NBODY*2,1);
MAERRs = zeros(NBODY,NELEM);
trash = zeros(6);
Mhinge = zeros(6);
% this next section sets up the sturcutral elements. They can't be finalized
% as they are dependent on the state of the system
for i = 1:NBODY
    Mhinge(1,1) = MHINGE(i);
    for k = 1:NELEM
        % c is mean chord length of element i,k
        c = C(i,1)+(2*k-1)/(2*NELEM)*(C(i,2)-C(i,1));
        l = C(i,3)/NELEM; L(k,i) = l;
        m = ar*rho*c^2*l; M(k,i) = m;
        b1 = (C(i,1)+(k-1)/NELEM*(C(i,2)-C(i,1)))/2;
        b2 = (C(i,1)+k/NELEM*(C(i,2)-C(i,1)))/2;
        m1 = (b1*2)^2*ar*rho; m2 = (b2*2)^2*ar*rho;
        x0 = sum(L(1:k,i))-l;
        eik = ei*c^4; gjk = gj*c^4; Isk = Is*c^4/l^2; ...
        Ink = In*c^4/l^2; Qnk=Qn*c^3/l; Qsk = Qs*c^3/l;
        [q, Mff, Kff, Maerr, Mae, Bae, Kae, Dae, e1, e2, fp, ...
        I1((((i-1)*NELEM+k)*2-1):((i-1)*NELEM+k)*2),:], ...
        IBAR0((((i-1)*NELEM+k)*6-5):((i-1)*NELEM+k)*6),:], ...
        SBAR((((i-1)*NELEM+k)*2-1):((i-1)*NELEM+k)*2),:], ...
        MFF1((((i-1)*NELEM+k)*6-5):((i-1)*NELEM+k)*6),:], ...
        SBAR1((((i-1)*NELEM+k)*2-1):((i-1)*NELEM+k)*2),:], ...
        STIL1((((i-1)*NELEM+k)*2-1):((i-1)*NELEM+k)*2),:] ...
        = aesfunflex(m,m1,m2,l,a,b1,b2,rhoa,U,x0, ...
        eik,gjk,Ink,Isk,Qnk,Qsk,k,MHINGE(i));
    B = zeros(Ndoff,6);
    B((k-1)*3+1:(k-1)*3+6,:) = eye(6);
    if k == 1
        MFF((i-1)*Ndoff+1:i*Ndoff,:) = ...

```

```

        MFF((i-1)*Ndoff+1:i*Ndoff,:)+B*(Mff+Mhinge)*B';
    else
        MFF((i-1)*Ndoff+1:i*Ndoff,:) = ...
        MFF((i-1)*Ndoff+1:i*Ndoff,:)+B*Mff*B';
    end
    KFF((i-1)*Ndoff+1:i*Ndoff,:) = ...
    KFF((i-1)*Ndoff+1:i*Ndoff,:)+B*Kff*B';
    Q((i-1)*NELEM*6+(k-1)*6+1:(i-1)*NELEM*6+(k-1)*6+6,:) = q;
    MAE((i-1)*NELEM*4+(k-1)*4+1:(i-1)*NELEM*4+(k-1)*4+4,:) = Mae;
    KAE((i-1)*NELEM*4+(k-1)*4+1:(i-1)*NELEM*4+(k-1)*4+4,:) = Kae;
    BAE((i-1)*NELEM*4+(k-1)*4+1:(i-1)*NELEM*4+(k-1)*4+4,:) = Bae;
    DAE((i-1)*NELEM*4+(k-1)*4+1:(i-1)*NELEM*4+(k-1)*4+4,:) = Dae;
    E1((i-1)*(NELEM+1)*2+(k-1)*2+1:(i-1)*(NELEM+1)*2+(k-1)*2+2,:) = e1;
    E2((i-1)*(NELEM+1)*2+(k-1)*2+1:(i-1)*(NELEM+1)*2+(k-1)*2+2,:) = e2;
    Fp((i-1)*(NELEM+1)*2+(k-1)*2+1:(i-1)*(NELEM+1)*2+(k-1)*2+2,:) = fp;
    LIFT((i-1)*(NELEM+1)*2+(k-1)*2+1:(i-1)*(NELEM+1)*2+(k-1)*2+2,:)... 130
    = .5*rhoa*U^2.*[2*b1*(C10+Cla*alpharoot); ...
    (2*b1)^2*(Cm0+Cma*alpharoot)];
    MAERRs(i,k) = Maerr;
end
[q, Mff, Kff, Maerr, Mae, Bae, Kae, Dae, e1, e2, fp, ...
    I1((((i-1)*NELEM+k)*2-1):(i-1)*NELEM+k)*2,:),...
    IBAR0((((i-1)*NELEM+k)*6-5):((i-1)*NELEM+k)*6,:),...
    SBAR((((i-1)*NELEM+k)*2-1):(i-1)*NELEM+k)*2,:),...
    trash, SBAR1((((i-1)*NELEM+k)*2-1):(i-1)*NELEM+k)*2,:),...
    STIL1((((i-1)*NELEM+k)*2-1):(i-1)*NELEM+k)*2,:)] =... 140
aesfunflex(m,m2,m2,l,a,b2,b2,rhoa,U,x0,eik,gjk,Ink,...
    Isk,Qnk,Qsk,k,MHINGE(i));
E1((i-1)*(NELEM+1)*2+(NELEM)*2+1:...
    (i-1)*(NELEM+1)*2+(NELEM)*2+2,:) = e1;
E2((i-1)*(NELEM+1)*2+(NELEM)*2+1:...
    (i-1)*(NELEM+1)*2+(NELEM)*2+2,:) = e2;
Fp((i-1)*(NELEM+1)*2+(NELEM)*2+1:...
    (i-1)*(NELEM+1)*2+(NELEM)*2+2,:) = fp;
LIFT((i-1)*(NELEM+1)*2+(NELEM)*2+1:...
    (i-1)*(NELEM+1)*2+(NELEM)*2+2,:) = ... 150
.5*rhoa*U^2.*[2*b2*(C10+Cla*alpharoot); ...
    (2*b2)^2*(Cm0+Cma*alpharoot)];
end
ltip = C(NBODY,3);
Y0=zeros(2*N+Np,1);
nind = N-4-3*(NBODY-1);
qi = zeros(nind,1);
Y0(1:N) = findy(qi);

```

```

%This finds the mass and constraint matrices for
%a system with zero body rotation angles
[Ms, phi] = flexmass2(Y0);
Qh = [ 0 HINGE(4,:)];
Dh = DAMP.*[0 (HINGE(3,:))];
N = NBODY;
Ndofb = Ndofb-2;
Kh = zeros(N); Bh = Kh;
m = sum(M);
l = sum(L);
Q = [1/3 0 1/6 0;
     0 1/3 0 1/6;
     0 0 0 0;
     1/6 0 1/3 0;
     0 1/6 0 1/3;
     0 0 0 0];
Q = Q';
Q1 = [1 0 0 0 0; 0 1 0 0 0; 0 0 0 1 0; 0 0 0 0 1];
Q2 = [1/3 1/6; 1/6 1/3]; Q3 = eye(4);
Kae = zeros(N*Ndofb); Bae = Kae; Mae = Kae; Ks = Kae; Bs = Kae;
Dae = zeros(N*Ndofb,N*NELEM*2);
Fp2 = zeros(N*NELEM*2);
E12 = zeros(N*NELEM*2,N*Ndofb); E22 = E12;
%The hinge loads are added to the system
for i = 1:N
    if i == 1
        Kh(1,1) = Qh(1)+Qh(2);
        Bh(1,1) = Dh(1)+Dh(2);
        Kh(1,2) = -Qh(2);
        Bh(1,2) = -Dh(2);
    elseif i == N
        Kh(N,N-1) = -Qh(N);
        Kh(N,N) = Qh(N);
        Bh(N,N-1) = -Dh(N);
        Bh(N,N) = Dh(N);
    else
        Kh(i,i-1) = -Qh(i);
        Kh(i,i) = Qh(i)+Qh(i+1);
        Kh(i,i+1) = -Qh(i+1);
        Bh(i,i-1) = -Dh(i);
        Bh(i,i) = Dh(i)+Dh(i+1);
        Bh(i,i+1) = -Dh(i+1);
    end
end
% The aero loads are added

```

```

Kaei = zeros(Ndoff+2); Daei = zeros(Ndoff+2,(NELEM+1)*2);
Baei = Kaei; Maei = Kaei;
E1i = zeros((NELEM)*2,Ndoff+2); E2i = E1i; Fpi = zeros((NELEM)*2);
for k = 1:NELEM
    kae = KAE((i-1)*4*NELEM+(k-1)*4+1:(i-1)*4*NELEM+(k)*4,:);
    bae = BAE((i-1)*4*NELEM+(k-1)*4+1:(i-1)*4*NELEM+(k)*4,:);
    mae = MAE((i-1)*4*NELEM+(k-1)*4+1:(i-1)*4*NELEM+(k)*4,:);
    dae = DAE((i-1)*4*NELEM+(k-1)*4+1:(i-1)*4*NELEM+(k)*4,:);
    e1 = E1((i-1)*2*(1+NELEM)+(k)*2+1:(i-1)*2*(1+NELEM)+(k)*2+2,:);
    e2 = E2((i-1)*2*(1+NELEM)+(k)*2+1:(i-1)*2*(1+NELEM)+(k)*2+2,:);
    B = zeros(8,Ndoff+2);
    B(1,1) = 1;
    B(2,2) = 1;
    B(3:8,3+(k-1)*3:8+(k-1)*3) = eye(6);
    B1 = zeros(4,(NELEM+1)*2);
    B1(1:4,(k-1)*2+1:(k-1)*2+4) = eye(4);
    B11 = zeros(2,(NELEM)*2);
    B11(:,(k-1)*2+1:k*2) = eye(2);
    x1 = sum(L(1:k,i))-L(k,i); x2 = sum(L(1:k,i));
    T = [[.5 (2*x1+x2)/6; 0 0; .5 (2*x2+x1)/6; 0 0] Q];
    T1 = [[1 x1; 0 0; 1 x2; 0 0] Q1];
    T2 = T1*B;
    T3 = T2(3:4,:);
    Kaei = Kaei+L(k,i).*B'*T'*kae*T2;
    Baei = Baei+L(k,i).*B'*T'*bae*T2;
    Maei = Maei+L(k,i).*B'*T'*mae*T2;
    Daei = Daei+L(k,i).*B'*T'*dae*Q3*B1;
    E1i = E1i+B11'*e1*T3;
    E2i = E2i+B11'*e2*T3;
    Fpi((k-1)*2+1:k*2,(k-1)*2+1:k*2) = ...
        Fp((i-1)*2*(1+NELEM)+(k)*2+1:(i-1)*2*(1+NELEM)+(k)*2+2,:);
end
if i == 1
    %This is for the first cantilvered segement
    B2 = zeros(Ndofb,Ndoff+2);
    B4 = zeros(Ndofb,Ndoff);
    B2(1:2,1:2) = eye(2);
    B2(3:Ndofb,6:Ndoff+2) = eye(Ndoff-3);
    B4(3:Ndofb,4:Ndoff) = eye(Ndoff-3);
    B3 = zeros(2*(NELEM+1),2*(NELEM));
    B3(3:2+2*NELEM,:) = eye(2*NELEM);
    Dae(1:Ndofb,1:2*NELEM) = B2*Daei*B3;
    Ks((i-1)*Ndofb+1:i*Ndofb,(i-1)*Ndofb+1:i*Ndofb) = ...
        B4*KFF((i-1)*Ndofb+1:i*Ndoff,:)*B4';

```



```

Kae((i-1)*Ndofb+1:i*Ndofb,(i-1)*Ndofb+1:i*Ndofb) = B2*Kaei*B2';
Bae((i-1)*Ndofb+1:i*Ndofb,(i-1)*Ndofb+1:i*Ndofb) = B2*Baei*B2';
Mae((i-1)*Ndofb+1:i*Ndofb,(i-1)*Ndofb+1:i*Ndofb) = B2*Maei*B2';
E12((i-1)*2*NELEM+1:i*2*NELEM,(i-1)*Ndofb+1:i*Ndofb) = E1i*B2';    250
E22((i-1)*2*NELEM+1:i*2*NELEM,(i-1)*Ndofb+1:i*Ndofb) = E2i*B2';
Fp2((i-1)*2*NELEM+1:i*2*NELEM,(i-1)*2*NELEM+1:i*2*NELEM) = Fpi;
% Back to accounting for hinge loads
Ks(i*Ndofb,(i-1)*Ndofb+2) =Ks(i*Ndofb,(i-1)*Ndofb+2) +Qh(i+1);
Ks(i*Ndofb,(i)*Ndofb+2) =Ks(i*Ndofb,(i)*Ndofb+2) -Qh(i+1);
Ks(i*Ndofb,i*Ndofb) =Ks(i*Ndofb,i*Ndofb) +Qh(i+1);
Ks(i*Ndofb,i*Ndofb+3) =Ks(i*Ndofb,i*Ndofb+3) -Qh(i+1);
Ks((i-1)*Ndofb+2,i*Ndofb) =Ks((i-1)*Ndofb+2,i*Ndofb) +Qh(i+1);
Ks((i-1)*Ndofb+2,i*Ndofb+3) =Ks((i-1)*Ndofb+2,i*Ndofb+3)-Qh(i+1);
Bs(i*Ndofb,(i-1)*Ndofb+2) =Bs(i*Ndofb,(i-1)*Ndofb+2) +Dh(i+1);    260
Bs(i*Ndofb,(i)*Ndofb+2) =Bs(i*Ndofb,(i)*Ndofb+2) -Dh(i+1);
Bs(i*Ndofb,i*Ndofb) =Bs(i*Ndofb,i*Ndofb) +Dh(i+1);
Bs(i*Ndofb,i*Ndofb+3) =Bs(i*Ndofb,i*Ndofb+3) -Dh(i+1);
Bs((i-1)*Ndofb+2,i*Ndofb) =Bs((i-1)*Ndofb+2,i*Ndofb) +Dh(i+1);
Bs((i-1)*Ndofb+2,i*Ndofb+3) =Bs((i-1)*Ndofb+2,i*Ndofb+3)-Dh(i+1);
else
% for all the pinned segments
B2 = zeros(Ndofb+2, Ndoff+2);
B4 = zeros(Ndofb+2,Ndoff);
B5 = zeros(Ndofb, Ndoff+2);    270
B2(1,4) = 1;
B2(3:4,1:2) = eye(2);
B2(5:Ndofb,5:Ndoff-1) = eye(Ndoff-5);
B2(Ndofb+1:Ndofb+2,Ndoff+1:Ndoff+2) = eye(2);
B4(1,2) = 1;
B4(5:Ndofb,3:Ndoff-3) = eye(Ndoff-5);
B4(Ndofb+1:Ndofb+2,Ndoff-1:Ndoff) = eye(2);
B5(1:2,1:2) = eye(2);
B5(3:Ndofb-2,5:Ndoff-1) = eye(Ndoff-5);
B5(Ndofb-1:Ndofb,Ndoff+1:Ndoff+2) = eye(2);    280
B3 = eye(2*(NELEM+1));
%Aero loads
Dae((i-1)*Ndofb-1:i*Ndofb,(i-1)*2*NELEM-1:i*2*NELEM) = ...
Dae((i-1)*Ndofb-1:i*Ndofb,(i-1)*2*NELEM-1:i*2*NELEM)+B2*Daei*B3;
Ks((i-1)*Ndofb-1:i*Ndofb,(i-1)*Ndofb-1:i*Ndofb) = ...
Ks((i-1)*Ndofb-1:i*Ndofb,(i-1)*Ndofb-1:i*Ndofb)+ ...
B4*KFF((i-1)*Ndoff+1:i*Ndoff,:)*B4';
Kae((i-1)*Ndofb-1:i*Ndofb,(i-1)*Ndofb-1:i*Ndofb) = ...
Kae((i-1)*Ndofb-1:i*Ndofb,(i-1)*Ndofb-1:i*Ndofb)+B2*Kaei*B2';
Bae((i-1)*Ndofb-1:i*Ndofb,(i-1)*Ndofb-1:i*Ndofb) = ...    290

```

```

    Bae((i-1)*Ndofb-1:i*Ndofb,(i-1)*Ndofb-1:i*Ndofb)+B2*Baei*B2' ;
    Mae((i-1)*Ndofb-1:i*Ndofb,(i-1)*Ndofb-1:i*Ndofb) = ...
    Mae((i-1)*Ndofb-1:i*Ndofb,(i-1)*Ndofb-1:i*Ndofb)+B2*Maei*B2' ;
    E12((i-1)*2*NELEM+1:i*2*NELEM,(i-1)*Ndofb+1:i*Ndofb) = Eli*B5' ;
    E22((i-1)*2*NELEM+1:i*2*NELEM,(i-1)*Ndofb+1:i*Ndofb) = E2i*B5' ;
    Fp2((i-1)*2*NELEM+1:i*2*NELEM,(i-1)*2*NELEM+1:i*2*NELEM) = Fpi;
    %Hinge loads
    Ks((i-1)*Ndofb+3,(i-1)*Ndofb+2) =Ks((i-1)*Ndofb+3,(i-1)*Ndofb+2)+Qh(i);
    Ks((i-1)*Ndofb+3,(i-2)*Ndofb+2) =Ks((i-1)*Ndofb+3,(i-2)*Ndofb+2)-Qh(i);
    Ks((i-1)*Ndofb+3,(i-1)*Ndofb+3) =Ks((i-1)*Ndofb+3,(i-1)*Ndofb+3)+Qh(i); 300
    Ks((i-1)*Ndofb+3,(i-1)*Ndofb) =Ks((i-1)*Ndofb+3,(i-1)*Ndofb) -Qh(i);
    Ks((i-1)*Ndofb+2,(i-1)*Ndofb) =Ks((i-1)*Ndofb+2,(i-1)*Ndofb) -Qh(i);
    Ks((i-1)*Ndofb+2,(i-1)*Ndofb+3) =Ks((i-1)*Ndofb+2,(i-1)*Ndofb+3)+Qh(i);
    Bs((i-1)*Ndofb+3,(i-1)*Ndofb+2) =Bs((i-1)*Ndofb+3,(i-1)*Ndofb+2)+Dh(i);
    Bs((i-1)*Ndofb+3,(i-2)*Ndofb+2) =Bs((i-1)*Ndofb+3,(i-2)*Ndofb+2)-Dh(i);
    Bs((i-1)*Ndofb+3,(i-1)*Ndofb+3) =Bs((i-1)*Ndofb+3,(i-1)*Ndofb+3)+Dh(i);
    Bs((i-1)*Ndofb+3,(i-1)*Ndofb) =Bs((i-1)*Ndofb+3,(i-1)*Ndofb) -Dh(i);
    Bs((i-1)*Ndofb+2,(i-1)*Ndofb) =Bs((i-1)*Ndofb+2,(i-1)*Ndofb) -Dh(i);
    Bs((i-1)*Ndofb+2,(i-1)*Ndofb+3) =Bs((i-1)*Ndofb+2,(i-1)*Ndofb+3)+Dh(i);
    if i < N 310
    Ks(i*Ndofb,(i-1)*Ndofb+2) =Ks(i*Ndofb,(i-1)*Ndofb+2) +Qh(i+1);
    Ks(i*Ndofb,(i)*Ndofb+2) =Ks(i*Ndofb,(i)*Ndofb+2) -Qh(i+1);
    Ks(i*Ndofb,i*Ndofb) =Ks(i*Ndofb,i*Ndofb) +Qh(i+1);
    Ks(i*Ndofb,i*Ndofb+3) =Ks(i*Ndofb,i*Ndofb+3) -Qh(i+1);
    Ks((i-1)*Ndofb+2,i*Ndofb) =Ks((i-1)*Ndofb+2,i*Ndofb) +Qh(i+1);
    Ks((i-1)*Ndofb+2,i*Ndofb+3) =Ks((i-1)*Ndofb+2,i*Ndofb+3)-Qh(i+1);
    Bs(i*Ndofb,(i-1)*Ndofb+2) =Bs(i*Ndofb,(i-1)*Ndofb+2) +Dh(i+1);
    Bs(i*Ndofb,(i)*Ndofb+2) =Bs(i*Ndofb,(i)*Ndofb+2) -Dh(i+1);
    Bs(i*Ndofb,i*Ndofb) =Bs(i*Ndofb,i*Ndofb) +Dh(i+1);
    Bs(i*Ndofb,i*Ndofb+3) =Bs(i*Ndofb,i*Ndofb+3) -Dh(i+1); 320
    Bs((i-1)*Ndofb+2,i*Ndofb) =Bs((i-1)*Ndofb+2,i*Ndofb) +Dh(i+1);
    Bs((i-1)*Ndofb+2,i*Ndofb+3) =Bs((i-1)*Ndofb+2,i*Ndofb+3)-Dh(i+1);
    end
    end
    end
    %now everything gets put into the global matrices
    Ks(2:Ndofb:N*Ndofb,2:Ndofb:N*Ndofb) = Ks(2:Ndofb:N*Ndofb,2:Ndofb:N*Ndofb) +Kh;
    Bs(2:Ndofb:N*Ndofb,2:Ndofb:N*Ndofb) = Bs(2:Ndofb:N*Ndofb,2:Ndofb:N*Ndofb) +Bh;
    Mp = Ms-Mae;
    Kp = Ks-Kae; 330
    Bp = Bs-Bae;
    fdsa = size(Mp);
    N2 = fdsa(1);
    fdsa = size(Fp2);

```

```

N3 = fdsa(1);
fdsa = size(phi);
nlam = fdsa(1);
L2 = zeros(2*N2+N3+nlam);
R2 = L2;
L2(1:N2,1:N2) = eye(N2);
L2(N2+1:2*N2,N2+1:2*N2) = Mp;
L2(N2+1:2*N2,2*N2+N3+1:2*N2+N3+nlam) = phi';
L2(2*N2+1:2*N2+N3,2*N2+1:2*N2+N3) = eye(N3);
L2(2*N2+N3+1:2*N2+N3+nlam,N2+1:2*N2) = phi;
R2(1:N2,N2+1:2*N2) = eye(N2);
R2(N2+1:2*N2,1:2*N2+N3) = [-Kp -Bp Dae];
R2(2*N2+1:2*N2+N3,1:2*N2+N3) =[E12 E22 Fp2];
%the eigen values are found
[phi,lam] = eig(L2\R2);
im(:,sti) = (imag(diag(lam)));
re(:,sti) =(real(diag(lam)));
end
%after all the eigen values of all speeds have been found
%they are split into thier real and imaginary parts and sorted
[im2,I] = sort(abs(im));
re2 = zeros(6,sti);
for i = 1:sti
    for k = 1:6
        %Here the real part is normalized to make a damping ratio
        re2(k,i) = re(I(45+k,i),i)/(re(I(45+k,i),i)^2+...
            im(I(45+k,i),i)^2)^.5;
    end
end
re2 = sort(re2);

```