Designing Active Control Laws in a Computational Aeroelasticity Environment

Jerry R. Newsom

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirement for the degree of

Doctor of Philosophy
in
Mechanical Engineering

Dr. Harry S. Robertshaw, Co-Chair
Dr. Rakesh K. Kapania, Co-Chair
Dr. Michael W. Hyer
Dr. Daniel J. Inman
Dr. Donald J. Leo

April 11, 2002
Blacksburg, Virginia

Keywords: Aeroelasticity, Computational Fluid Dynamics, Flutter
Active Controls

Designing Active Control Laws in a
Computational Aeroelasticity Environment

Jerry R. Newsom

(ABSTRACT)

The purpose of this dissertation is to develop a methodology for designing active control laws in a computational aeroelasticity environment. The methodology involves employing a systems identification technique to develop an explicit state-space model for control law design from the output of a computational aeroelasticity code. The particular computational aeroelasticity code employed in this dissertation solves the transonic small disturbance equation using a time-accurate, finite-difference scheme. Linear structural dynamics equations are integrated simultaneously with the computational fluid dynamics equations to determine the time responses of the structural outputs. These structural outputs are employed as the input to a modern systems identification technique that determines the Markov parameters of an "equivalent linear system". The eigensystem realization algorithm is then employed to develop an explicit state-space model of the equivalent linear system. Although there are many control law design techniques available, the standard Linear Quadratic Guassian technique is employed in this dissertation. The computational aeroelasticity code is modified to accept control laws and perform closed-loop simulations. Flutter control of a rectangular wing model is chosen to demonstrate the methodology. Various cases are used to illustrate the usefulness of the methodology as the nonlinearity of the computational fluid dynamics system is increased through increased angle-of-attack changes.

Dedication

To my wife, Ellen, thank you and I love you.

Acknowledgements

First, I want to thank the NASA LaRC for giving me the time and support to conduct this research. Many LaRC researchers have provided help and advice. Dr. David Seidel provided the CAP-TSD math model of the BACT wing. He also helped me understand many of the subtle nuances in running the CAP-TSD code. Dr. Jer-Nan Juang provided the Matlab routines that implement the OKID system identification technique. Both Dr. Juang and Dr. Lucas Horta provided expert advice on understanding OKID. Dr. Walt Silva provided many excellent comments on the first draft of this dissertation. Thank you all. A special thanks to Dr. Jeremiah Creedon, the LaRC Director, for pushing and encouraging me to finish this dissertation. Finally, I would like to thank my co-advisors, Dr. Harry Robertshaw and Dr. Rakesh Kapania, for their guidance and support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Aeroelasticity is the mutual interaction between aerodynamics and a flexible body. Aeroelasticity has been and continues to be an extremely important consideration in many aircraft designs [1]. Static aeroelasticity effects are many times important considerations in such areas as control surface effectiveness, stability, divergence, and overall load distributions. Dynamic aeroelasticity effects, such as flutter, many times provide constraints on the performance of high-performance aircraft. To address undesirable aeroelastic effects or phenomena, the stiffness of the wing is usually increased, adding weight to the aircraft and decreasing the overall performance. Aeroelastic tailoring, through the use of advanced composite materials, is an alternative to adding material (i.e. weight) to increase stiffness.

The control of aeroelastic response through feedback to control surfaces or more recently through feedback to active materials, is an alternative to "passive control" through increased stiffness. Noll [2] presents a review of active control methods, wind-tunnel experiments, and flight experiences associated with feedback control and aeroelasticity. Currently, most aeroelasticity considerations are routinely addressed using linear aeroelastic (i.e. linear aerodynamic and linear elasticity) models. However, within the last few decades, a significant increase in advancing methods to consider nonlinear aeroelasticity, especially nonlinear aerodynamics in the transonic region, has taken place.

Dowell [3] presents an excellent overview of nonlinear aeroelasticity and major problems where nonlinear effects should be considered.

1.1 Structural Nonlinearity in Aeroelasticity

Of all the nonlinear aeroelasticity problems, panel flutter has probably received the most sustained attention. Panel flutter is a dynamic instability, primarily in supersonic flow, of a thin plate or shell where the nonlinearity is geometric involving the bending and extension of a plate. Other types of structural nonlinearities involve freeplay and other mechanisms, especially with control surfaces, which give rise to nonlinear stiffness, especially in torsion. Many times this type of nonlinearity can have a pronounced effect on control surface flutter. It usually leads to a limited amplitude flutter which is not explosive but may cause fatigue problems. Breitbach [4] describes several of these structural types of nonlinear aeroelastic problems. Murty [5] shows that a limit cycle oscillation can occur prior to the linear flutter point for cases involving a nonlinear torsional stiffness. Murty employed a two-dimensional structural model and incompressible linear aerodynamics. References 6 - 8 also present studies of nonlinear aeroelastic response due to structural nonlinearities.

As new unconventional aircraft configurations appear, the potential for other nonlinear aeroelastic issues increases. An example is the extremely high aspect ratio (on the order of 20-30) wing configurations that are characteristic of some uninhabited aircraft vehicle (UAV) designs [9]. Because of the large elastic deflections of this type of wing, geometric nonlinearities are important and must be considered in both static and dynamic aeroelastic analyses [10,11].

Research studies focusing on the control of aeroelastic systems with structural nonlinearity have been limited. Scott and Weishaar [12] examine the use of active materials to control panel flutter. Both piezoelectrics and shape memory alloys were employed as actuators. The study indicated that controlling the in-plane panel forces provided the most effective

control of panel flutter. References 13 and 14 are other examples of studies that examine control of structurally nonlinear aeroelastic systems. Both of these studies employ the 2-dimensional typical section model of an airfoil. Reference 14 also includes an aerodynamic nonlinearity that accounts for a nonlinear pitching moment and employs a neural network control law design method.

1.2 Aerodynamic Nonlinearity in Aeroelasticity

Examples of aerodynamic nonlinearities are separated flow at high angle of attack that can cause wing and tail buffet [15] and transonic flow with shocks, which can have a significant impact on flutter [16]. Cunningham [17] describes several "real-world" examples where nonlinear aerodynamics provided unanticipated problems in flight.

References 18-20 provide studies of nonlinear aeroelastic behavior in subsonic flow. The aerodynamics are computed using the general unsteady vortex-lattice method and can account for aerodynamic nonlinearities associated with angles of attack and vortex dominated flow. Both the structural equations and aerodynamic equations are numerically integrated, in the time domain, simultaneously to solve for the wing motions.

For transonic flow, shock waves over wings and bodies occur and can be extremely important in calculating aerodynamic forces. The transonic region is of significant interest since an aircraft critical flutter point is usually encountered in this region. Ashley [21] discusses the role of shocks in the flutter problem and states that "unsteady shock effects are often quite pronounced and should not be omitted from flutter calculations". Computational fluid dynamics (CFD) methods have been extensively developed, especially for transonic flow, over the last 3 decades and primarily applied to steady aerodynamic applications. Computational fluid dynamics models are available for solving increasingly more accurate fluid flow equations including the nonlinear potential equation (both transonic small disturbance and full potential equation), the Euler equations, and the Navier-Stokes equations. Although the accuracy of the fluid flow increases from the

transonic small disturbance solution to the Navier-Stokes solution, the computational burden increases significantly as well.

With the maturity of CFD codes, their incorporation into aeroelastic analyses, both static and dynamic, is beginning to occur, primarily in the research community. This relatively new field has been termed "computational aeroelasticity" and involves coupling structural elasticity (static and/or dynamic) and computational fluid dynamics together to perform time domain analyses where the aerodynamics are a nonlinear function of the deformation of the aircraft. Most of the work in computational aeroelasticity has employed solutions to either the nonlinear potential equation or the Euler equations. Very little computational aeroelasticity work has employed the Navier-Stokes equations. Bennett and Edwards [22] provide a status of recent developments in computational aeroelasticity, with an emphasis on unsteady transonic flow, and results of some applications.

1.3 Computational Aeroelasticity in Transonic Flow

The first applications of CFD for aeroelastic analyses occurred in the late 1970s. One of the earliest applications is by Rizzetta [23]. Rizzetta integrated solutions of the unsteady low-frequency small-disturbance transonic potential equation with a three-degree-of-freedom NACA 64A010 airfoil structural model to calculate time responses of the aeroelastic system. Flutter analyses were conducted at two Mach numbers (0.72 and 0.80) and two angles of attack (0 and 1 deg.). He noted the decrease of flutter speed with increasing Mach number. Yang, Guruswamy, and Striz [24] also used the same technique to compare flutter results for both a NACA 64A006 and NACA 64A010 airfoil. They also conducted parametric studies for different values of mass-ratio, frequency ratio, mass center location, and elastic axis location. Eastep and Olsen [25] conducted one of the first applications of a 3-dimensional CFD code to transonic flutter. The wing was rectangular with constant mass and stiffness properties. Transonic flutter analyses were performed at two Mach numbers and compared to the analyses using linear aerodynamics. The decrease in flutter speed with Mach number was greater when transonic aerodynamics

were used rather than linear aerodynamics. In these studies, the low-frequency limit was determined to be at a reduced frequency of approximately 0.2. Edwards [26] employed an improved version of the low-frequency CFD code (2-d) that eliminated the low-frequency limit. He made a comparison of transonic flutter boundaries using four nonlinear transonic codes. He considered the effect of angle of attack and aeroelastic twisting from a steady pitching moment. His results showed that a change of 1.5 degrees in angle of attack can cause a 50% decrease in flutter speed index for a NACA 64A010 airfoil and a 60% decrease for a supercritical MBB A-3 airfoil.

References 27-30 are studies that incorporate solutions to the Euler equations for aeroelastic analyses. The Euler equations allow for large amplitude motion and resulting large shock motions. These results indicated that shocks on the surface of the airfoils interact with the airfoil motion to force the system into a limit cycle.

Batina [31] developed a transonic unsteady aerodynamic and aeroelastic code called Computational Aero-elasticity Program — Transonic Small Disturbance (CAP-TSD) for application to realistic aircraft configurations. The CAP-TSD code uses a time-accurate approximate factorization algorithm for the solution of the unsteady transonic small-disturbance equation. CAP-TSD has been used for a variety of aircraft configurations for flutter analyses [32-36]. Edwards [37] coupled an interactive boundary layer modeling technique with the CAP-TSD code to evaluate the viscous effects on self-excited shock-induced oscillations and transonic flutter. One of the conclusions was that for Mach numbers below and very near unity, viscous modeling is required on thin wings to achieve good comparison with experiment. For a thicker wing, the requirement for viscous modeling extends to the lower transonic Mach numbers.

An overview of an extensive study to evaluate three computational aeroelasticity codes for loads and flutter is given in Ref. 38. The three codes are the CAP-TSD code with boundary layer modeling [37] and two Euler/Navier-Stokes codes. Comparisons are made

with available wind-tunnel and flight test data of realistic aircraft. One of the conclusions was that the modeling and computational efficiency of CAP-TSD makes it an excellent first choice for computing transonic phenomena.

1.4 Reduced Order Models

Because of the high computational burden of CFD analyses during computational aeroelasticty or computational aeroservoelasticity analyses, there is a motivation to develop techniques to calculate reduced-order-models (ROMs). The development of reduced-order-models is often employed especially when there is a need to perform repetitive analyses. Indeed, control law design is one of the major motivations for developing ROMs. References 39-49 describe some of the techniques for developing ROMs. All of these techniques focus on developing a ROM for the unsteady aerodynamic forces.

As stated in Ref. 39, "a ROM for a computational aeroelasticity system is a functional condensation as opposed to the traditional decreased order of a matrix". One of the techniques for developing a ROM is to apply an exponential pulse to excite the aeroelastic system, one mode at a time, to calculate time histories of the generalized aerodynamic forces. The generalized aerodynamic force matrix can then be transformed into the frequency domain (using Fourier Transformation) and then be used in standard linear aeroelastic analyses. A recent technique for developing ROMs is based on Volterra theory of nonlinear systems [41-45]. Volterra ROMs are based on the computation of unsteady aerodynamic impulse responses that are used in a convolution algorithm to provide the responses to arbitrary inputs. Reference 43 extends the Volterra approach to the computation of state-space models of the aerodynamic forces. Reduced-order models based on Volterra theory have been applied successfully from Transonic Small Disturbance to Navier-Stokes models of nonlinear unsteady aerodynamic systems.

Cowan, Arena, and Gupta [46, 47] developed a technique that employs system identification to fit the time responses of the generalized forces from a CFD code with constant coefficients of a linear system model. An autoregressive moving average model is used and describes the aerodynamic forces as a sum of scaled previous outputs and inputs. This linear model of the aerodynamic forces is then coupled with the structural dynamics model to form an overall ROM of an aeroelastic system. Results have shown that this technique accurately represents the input-output relationship of an unsteady CFD code.

Another technique is to determine the eigenvalues and eigenvectors of the fundamental fluid equations [48, 49]. The eigenvalues and eigenvectors can then be employed to develop a "modal" model of the aerodynamic forces. A state-space representation of the aerodynamic model can then be constructed and used in standard linear aeroelastic analyses.

1.5 Previous Transonic Computational Aeroservoelasticity Work

Computational aeroservoelasticity involves coupling structural dynamics, computational fluid dynamics, and active control systems together. Batina and Yang [50] were perhaps the first researchers to examine control of an aeroelastic system in a computational aeroelasticity environment for transonic flow. They conducted studies with a 2-d airfoil and a 2-d small-disturbance transonic CFD code. The effect of a simple constant gain control law utilizing displacement, velocity, and acceleration feedback on the time responses was determined. Comparison with linear theory indicated that the frequency and damping values were significantly different for transonic and linear subsonic theory results.

References 51-55 are other examples of research in control of aeroelastic systems within a computational aeroelasticity environment. Similar to Batina and Yang, these studies also

only involve varying the gains of simple feedback control laws to study their effect on the response of the aeroelastic system. Two-dimensional and three-dimensional small-disturbance and Euler CFD codes are used in these studies. The studies show that feedback control can be effective in suppressing transonic flutter.

Guillot and Friedman [56,57] employ adaptive control theory to design control laws using a CFD technique. A 2-dimensional airfoil model, with a trailing-edge control surface, is used with an Euler CFD code to perform the computational aeroelastic solutions. An adaptive control law was used because of the assumed nonlinear behavior of the system in the presence of nonlinear transonic flow with large shock motions. The adaptive control law involves identifying a linear auto-regressive moving average (ARMA) model and then determining an optimal full-state control law. A random control surface excitation is used for the first 24 time steps to provide a learning period for the adaptive control law (i.e. determine the initial ARMA model). At the end of the learning period, the control law is engaged and the ARMA model and control law are then updated at each subsequent time step during the computational aeroelasticity simulation. An adaptive control law was shown to be quite effective in suppressing transonic flutter with strong shocks.

1.6 Objective

The major objective of this dissertation is to develop a general methodology to design control laws in the context of a computational aeroelasticity environment. A brief description of the present state-of-the-art for active control design is given followed by a description of the technical approach developed in this dissertation. The technical approach involves employing a systems identification technique to develop an explicit state-space model for control law design from the output of a computational aeroelasticity code. Although there are many control law design techniques available, the standard Linear Quadratic Guassian technique is employed in this dissertation. The computational aeroelasticity code is modified to accept control laws and perform closed-loop

simulations. Numerical results for flutter suppression of the Benchmark Active Control Technology wind-tunnel model are given to illustrate the approach. Finally, conclusions and recommended future research are discussed.

Chapter 2

State-of-the-Art Modeling for Control Design

In order to employ nearly all control law design methods available, a mathematical model of the system to be controlled is required. The form of the mathematical model required, especially for modern control law design methods, is a set of constant coefficient linear first-order differential equations (i.e. state-space form). However, the present-day state-of-the-art aeroelastic equations are second-order and contain frequency-dependent terms that represent linear unsteady aerodynamics. Furthermore, these frequency-dependent terms are calculated at discrete frequencies and therefore are available in a tabular format. Over the last 15-20 years, there have been several methods developed for transforming the classical frequency dependent aeroelastic equations into state-space form. All of these methods involve some type of rational function approximation to the tabular frequency-dependent linear aerodynamic terms.

2.1 Equations of Motion for a Flexible System

The commonly accepted way of deriving the equations of motion for an aeroelastic system is to describe the motion as a linear combination of the undamped natural

vibration modes of the system. Employing energy methods and virtual work, the equations of motion for a flexible system can be written as:

$$[M]\{\ddot{q}\}+[C]\{\dot{q}\}+[K]\{q\}=\{F\} \qquad (2.1)$$

where

$[M]$ is the generalized mass matrix

$[C]$ is the generalized damping matrix

$[K]$ is the generalized stiffness matrix

$\{F\}$ is the generalized aerodynamic force vector

By virtue of using the natural vibration modes of the system, the $M, C, and K$ matrices are diagonal where each element of the diagonal is defined as:

$$M_{ii}=\iint m(x,y)\, Z_i^{\,2}(x,y)\, ds$$

$$C_{ii}=2\xi_i M_{ii}\omega_i$$

$$K_{ii}=M_{ii}\omega_i^{\,2}$$

and

$m(x,y)$ is the mass distribution, $Z_i(x,y)$ is the ith vibration mode shape, ω_i is the ith vibration mode frequency, and ξ_i is the ith mode viscous damping coefficient.

The elements of the aerodynamic force vector are:

$$F_i=\iint_S \Delta p(x,y,t) Z_i(x,y) dS$$

When small amplitudes are assumed, the total pressure distribution $\Delta p(x,y,t)$ can be expressed as the sum of the contributions due to each flexible mode, control surface motions, and external forces such as a gust. The present state-of-the-art methods for computing unsteady aerodynamics only compute aerodynamics for simple oscillatory motion. Therefore, the equations of motion for an aeroelastic system are usually expressed in the frequency domain as:

$$-\omega^2[M]\{q(i\omega)\} + i\omega[C]\{q(i\omega)\} + [K]\{q(i\omega)\} + [Q(ik)]\{q(i\omega)\} = \{f(ik)\} \quad (2.2)$$

where $[Q(ik)]$ is the tabular-valued unsteady aerodynamic force matrix, $\{f(ik)\}$ is the control input and any disturbance force (e.g. gust), and k is reduced frequency defined as k=c/(2 U)ω ; where c is a reference length and U is the free-stream velocity. The computation of the aerodynamic force matrix is well established through such common methods as the Doublet Lattice and Kernel Function techniques. These techniques compute the unsteady aerodynamics for a specific Mach number and for a user-defined set of reduced frequencies k.

2.2 Rational Function Approximations

Rational function approximations of the frequency dependent unsteady aerodynamic forces are employed to permit the second-order aeroelastic equations of motion to be transformed to state-space form. Basically, rational function approximation techniques use the tabular frequency-dependent oscillatory unsteady aerodynamic forces to generate approximations capable of continuation into the complex frequency or Laplace s-plane [2,58]. The basic technique is to assume a rational function approximation such as:

$$\hat{Q}_{ij}(ik) = (A_0)_{ij} + (A_1)_{ij} ik + (A_2)_{ij} (ik)^2 + \sum_{l=1}^{n_l} (A_{(l+2)})_{ij} \frac{ik}{ik + \beta_l} \quad (2.3)$$

and find the values of A_i *and* β_i that minimize the difference between the tabular values and the rational function approximation over the reduced frequency range that the unsteady aerodynamics have been computed as shown in Fig. 2.1.

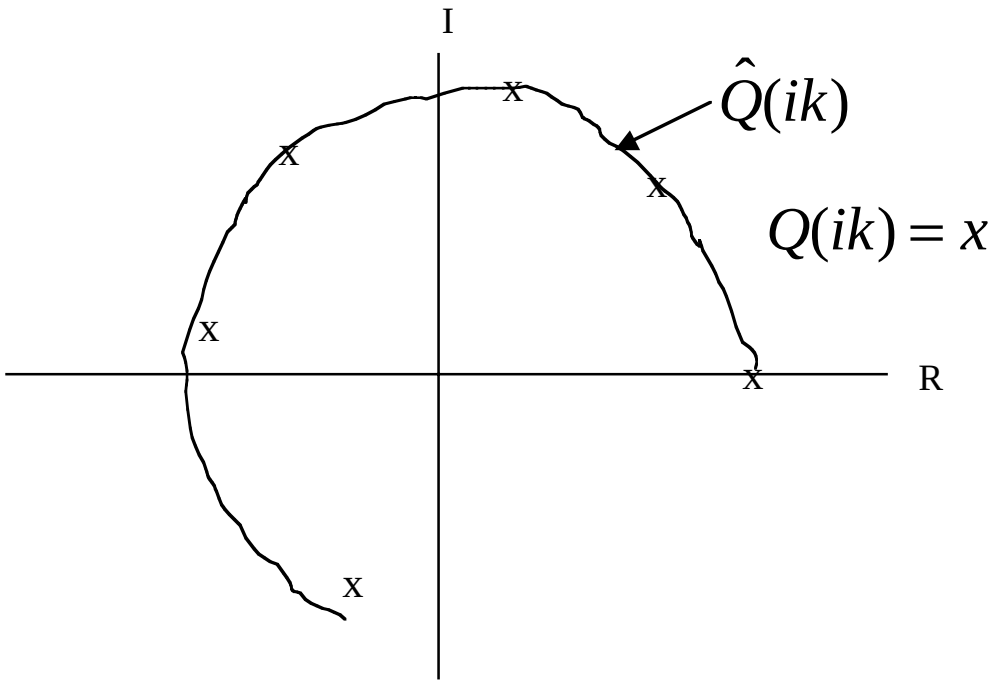

Figure 2.1 Rational Function Approximations to Linear Unsteady Aerodynamics

There are several other commonly used rational function approximations, however the basic overall technique is the same. After the values of A_l *and* β_l are determined, *ik* is replaced with the nondimensional Laplace operator p=(c/2U)s employing the concept of analytic continuation. By identifying derivatives with the powers of s, a set of state-space equations can be formulated of the form:

$$\begin{aligned} \{\dot{X}\} &= [A]\{X\} + [B]\{u\} \\ \{Y\} &= [C]\{X\} + [D]\{u\} \end{aligned} \tag{2.4}$$

where

$$\{X\} = \begin{Bmatrix} q \\ \dot{q} \\ x_a \end{Bmatrix}$$

$\{q\}$ *generalized coordinate displacements*

$\{\dot{q}\}$ *generalized coordinate velocities*

$\{x_a\}$ *aerodynamic states*

By employing a rational function approximation, the overall size of the state vector of an aeroelastic system model can be quite large. This is primarily due to the size of the number of aerodynamic states required to accurately model the unsteady aerodynamics. There is always a trade-off between how well the rational function approximation approximates the tabular values and the desire to minimize the number of aerodynamic states. After the equations have been transformed to state-space form, linear systems analysis techniques and various control law design techniques can be employed.

The next chapter will present a technical description of the methodology developed in this dissertation.

Chapter 3

Description of Technical Approach

This chapter begins with a description of computational aeroelasticity simulations. The development of a control law design model from the outputs of the computational aeroelasticity simulation is presented and a brief description of the control law design technique is given. Finally, a description of the computer implementation of the overall methodology is presented.

The overall methodology for active control law design in a computational aeroelasticity environment, as developed in this dissertation, is illustrated in Fig. 3.1. The process begins with performing a computational aeroelasticity simulation (uncontrolled) with prescribed control surface inputs to obtain a set of corresponding output time histories. The next step is to employ a system identification technique, using the time histories of outputs and inputs from the first step, to determine an "equivalent linear system" for use as a control law design model. Next, design of a control law design can be performed using any control law design technique. Finally, the control law is evaluated in the computational aeroelasticity simulation. If the control law performance is not adequate, the control law can be redesigned and evaluated again until the desired performance is obtained.

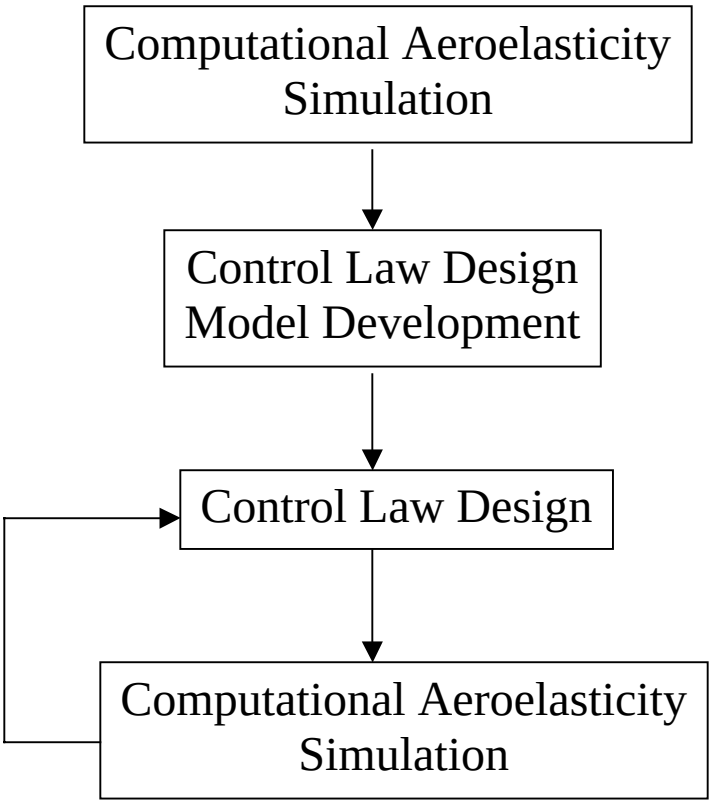

Figure 3.1 Overall process for control design

3.1 Computational Aeroelasticity Simulation

Computational aeroelasticity simulation involves integrating the structural, aerodynamic (CFD), and for this research, control equations simultaneously. This dissertation focuses on the transonic case where the aerodynamic fluid flow equations contain nonlinear terms. The particular code that is employed is the CAP-TSD [31] code that has been developed at the NASA Langley Research Center. The CAP-TSD code is a finite-difference code

which solves the transonic small-disturbance equation. The primary outputs of the CAP-TSD code are time histories of the pressures and the generalized coordinate displacements, velocities, and accelerations. It has been used on a wide variety of configurations for steady and unsteady pressure distribution calculations and for calculating transonic flutter characteristics, including nonlinear limit-cycle instabilities. In order to use the CAP-TSD code, the user provides the following general information:

A. Problem definition data such as Mach number, dynamic pressure, boundary conditions, number of time steps, etc.
B. Geometric data such as the number of horizontal and vertical surfaces, control surfaces, and bodies and the mesh definition, etc.
C. Modal data such as the number of modes, generalized mass matrix, frequencies, structural dampings, mode-shape displacements and slopes, initial conditions, etc.

The modifications to the CAP-TSD code to incorporate a feedback control law will be described later.

3.1.1 Equations of Motion

The basic equations of motion are the same as previously given in Chapter 2.1 (eq. 2.1).

$$[M]\{\ddot{q}\} + [C]\{\dot{q}\} + [K]\{q\} = \{F\} \tag{3.1}$$

The primary difference between state-of-the-art linear aeroelasticity methods and computational aeroelasticity is in the computation of the aerodynamic pressure ΔC_p that is used in computing the generalized aerodynamic force vector $\{F\}$. As stated earlier, CAP-TSD solves the transonic small disturbance equation by a finite difference technique to determine the aerodynamic pressure ΔC_p. The solution of the equations of motion will be described in a subsequent section.

3.1.2 Transonic Small Disturbance Equation

The CAP-TSD code is a finite difference program that solves the general-frequency modified TSD potential equation

$$M_\infty^2 (\phi_t + 2\phi_x)_t = [(1-M_\infty^2)\phi_x + F\phi_x^2 + G\phi_y^2]_x + (\phi_y + H\phi_x\phi_y)_y + (\phi_z)_z \qquad (3.2)$$

where M_∞ is the freestream Mach number, ϕ is the disturbance velocity potential, and the subscripts of ϕ represent partial derivatives.

Several choices are available for the coefficients F, G, and H, depending upon the assumptions used in deriving the TSD equation. In this dissertation, the coefficients are defined as

$$F = -\frac{1}{2}(\gamma+1) M_\infty^2$$
$$G = \frac{1}{2}(\gamma-3) M_\infty^2$$
$$H = -(\gamma-1) M_\infty^2$$

where γ is the ratio of specific heats of the aerodynamic fluid. The linear potential equation can be solved by simply setting F, G, and H equal to zero.

Equation 3.2 is solved within CAP-TSD by a time-accurate approximate factorization (AF) algorithm developed by Batina [59]. The algorithm consists of a Newton linearization procedure coupled with an internal iteration technique. The CAP-TSD code is capable of treating configurations with multiple lifting surfaces and bodies. A relatively simple Cartesian grid is input along with the coordinates defining the geometry

of the configuration and the corresponding surface slopes. After the potential is calculated at each time step, the pressure coefficient is calculated by

$$C_p = -2\phi_x - 2/U_\infty \phi_t$$

The pressure coefficient is employed to calculate the generalized force vector at each time step.

3.1.3 Time-Marching Solutions to the Equations of Motion

Equation 3.1 can be rewritten in state-space form as

$$\{\dot{X}\} = [A]\{X\} + [B]\{u\} \tag{3.3}$$

where

$$\{X\} = (q, \dot{q})^T$$

$$[A] = \begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}C \end{bmatrix}$$

$$[B] = \begin{bmatrix} 0 \\ M^{-1} \end{bmatrix}$$

$$\{u\} = \{F\}$$

The numerical algorithm [26], employed in CAP-TSD, for solving equation 3.3 is

$$X_{n+1} = \phi X_n + \theta B(3u_n - u_{n-1})/2$$

where

$$\phi = e^{A\Delta T}$$

$$\theta = \int_0^{\Delta T} e^{A(\Delta T - \tau)} d\tau$$

The calculation of the ϕ and θ matrices is given in Appendix A.

3.2 Control Law Design Model Development

Most control law design methods require an explicit mathematical model of the system to be controlled. A computational aeroelasticity simulation provides time histories of the variables of an aeroelastic system, but does not generate an explicit mathematical model of the system. Computational aeroelasticity simulations are analogous to performing experimental investigations where the only direct outputs are time responses. Therefore, to employ the various control law design methods that are available, a control design mathematical model of a "computational aeroelasticity system" (CAS) must be developed. System identification techniques are widely employed for developing a mathematical model given experimental data. Therefore, since a CAS simulation is analogous to performing an experiment, system identification is a logical choice. Juang [60] provides a description of many of the advances in system identification for vibration modal testing and control of space structures. The Observer/Kalman filter Identification (OKID) technique [61,62] was developed primarily for development of a mathematical model for control law design. The OKID technique has been applied to space structures, such as the Hubble Space Telescope [60] and the Shuttle Remote Manipulator System [63], and to the identification of linear rigid aircraft models [64]. Because the OKID technique was developed primarily for identifying models for control law design, it is the technique employed in this dissertation.

3.2.1 Description of Observer/Kalman Filter Identification Technique

Given a set of input and output data, the Observer/Kalman Filter Identification (OKID) algorithm is shown in Fig. 3.2. One of the keys to the OKID algorithm is the introduction of an observer into the identification process. The fist step of the process is the calculation of the observer Markov parameters. Then the system Markov parameters are obtained.

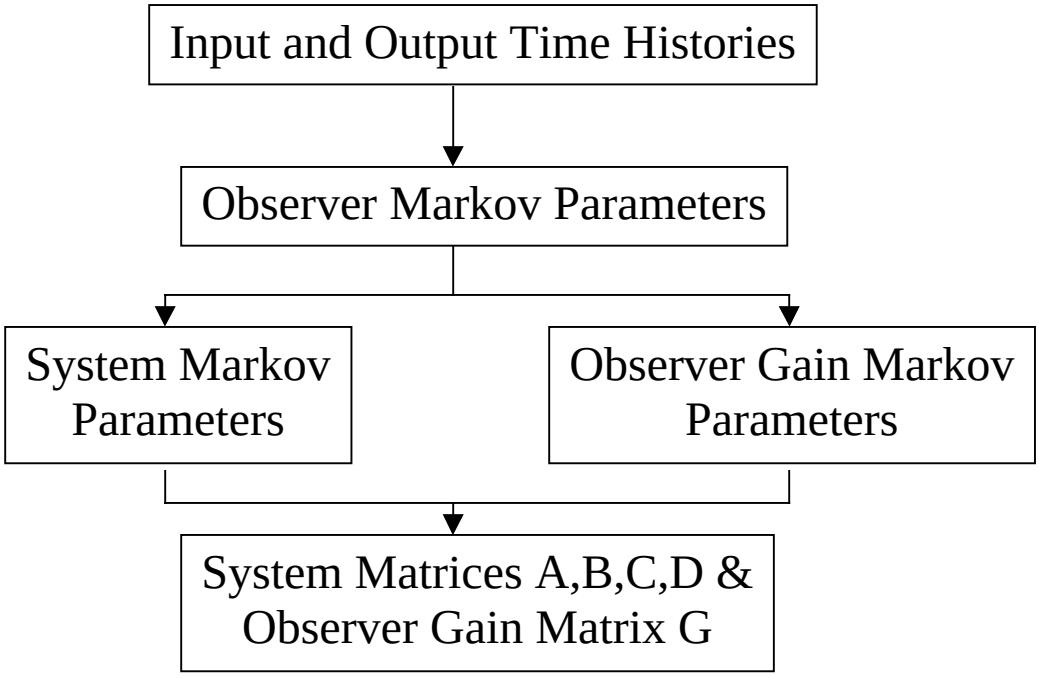

Figure 3.2 Flow Chart for OKID

3.2.2 Markov Parameters

Consider a discrete time state-space model of a system described by a set of first order difference equations of the form

$$x(k+1) = Ax(k) + Bu(k)$$
$$y(k) = Cx(k) + Du(k) \tag{3.4}$$

Solving for the output $y(k)$ in terms of the previous inputs, with the assumption that the system is initially at rest , i.e. $x(0) = 0$, yields

$$y(k) = \sum_{i=0}^{k} h_i u(k-i) \tag{3.5}$$

where the parameters

$$h_0 = D, \quad h_k = CA^{k-1}B, \quad k = 1,2,3,....$$

are the system Markov parameters, which are also the system pulse response samples. To reduce the number of Markov parameters needed to adequately model a system, an observer is introduced into the OKID technique. Adding and subtracting the term $\bar{K}y(k)$ to the right hand side of equation (3.4) yields

$$x(k+1) = (A + \bar{K}C)x(k) + (B + \bar{K}D)u(k) - \bar{K}y(k)$$
$$y(k) = Cx(k) + Du(k) \tag{3.6}$$

The matrix $\bar{K}$ can be interpreted as an observer gain. The parameters defined as

$$\bar{Y}(k) = C(A + \bar{K}C)^{k-1}[B + \bar{K}D, -\bar{K}]$$
$$= [\beta_k, \quad \alpha_k] \tag{3.7}$$

are the Markov parameters of an observer system. Consider the special case where $\bar{K}$ is a deadbeat observer gain such that all eigenvalues of $A + \bar{K}C$ are zero, the observer

Markov parameters will become identically zero after a finite number of terms. For lightly damped systems, this means that the system can be described by a reduced number of observer Markov parameters. Furthermore, an unstable system can be represented using this technique. This is obviously a major advantage for this research.

The Markov parameters are solved using a least squares technique. The observer state equations (3.6) can be rewritten as

$$x(k+1) = \overline{A}x(k) + \overline{B}v(k)$$
$$y(k) = Cx(k) + Du(k)$$

where

$$\overline{A} = A + \overline{K}C$$
$$\overline{B} = [B + \overline{K}D, -\overline{K}]$$
$$v(k) = \begin{bmatrix} u(k) \\ y(k) \end{bmatrix}$$

Similar to (3.5), but in matrix form, a data set of $(N+1)$ sampled outputs can be represented as

$$\overline{y} = \overline{Y}\,\overline{V}$$

where

$$\overline{y} = [y(0)\ y(1)\ .....\ y(N)]$$

$$\overline{Y} = [\beta_0,\ \beta_1,\ \alpha_1,\ \beta_2,\ \alpha_2,\ \ldots\ \beta_p,\ \alpha_p]$$

$$\overline{V} = \begin{bmatrix} u(0) & u(1) & \dots & u(p) & u(p+1) & \dots & u(N) \\ & v(0) & \dots & v(p-1) & v(p) & \dots & v(N-1) \\ & & . & . & . & . & . \\ & & & v(0) & v(1) & \dots & v(N-p) \end{bmatrix}$$

The observer Markov parameters can be solved by

$$\overline{Y} = \overline{y}\ \overline{V}^{+}$$

where $\overline{V}^{+}$ is the pseudo-inverse of $\overline{V}$. The actual system Markov parameters are determined from the observer Markov parameters using a recursive formula. By partitioning $\overline{Y}$ and using the definition of the system Markov parameters (equation 3.7), the system Markov parameters Y are recovered by

$$Y_0 = \overline{Y}_0 = D$$

$$Y_k = \beta_k - \sum_{i=1}^{k} \alpha_i Y_{k-i} \quad for\ k = 1,2,\dots,p$$

$$Y_k = -\sum_{i=1}^{p} \alpha_i Y_{k-i} \quad for\ k = p+1,\dots,\infty$$

3.2.3 State-Space Model System Matrices

A state-space model of a system is developed, employing the system Markov parameters, using the Eigensystem Realization Algorithm (ERA) [65]. The ERA begins with determining the singular value decomposition of a matrix with entries that are the Markov parameters.

$$H(k-1)=\begin{bmatrix} Y_k & Y_{k+1} & \cdots & Y_{k+\beta-1} \\ Y_{k+1} & Y_{k+2} & \cdots & Y_{k+\beta} \\ \cdot & \cdot & \cdot & \cdot \\ Y_{k+\alpha-1} & Y_{k+\alpha} & \cdots & Y_{k+\alpha+\beta-2} \end{bmatrix}$$

where α *and* β are integers.

The order of the system is determined by the singular value decomposition of $H(0)$

$$H(0)=U\Sigma V^T$$

where the columns of U and V are orthonormal, Σ is an nxn diagonal matrix of positive singular values, and n is the order of the system.

A discrete-time minimal-order realization of the system is

$$A=\Sigma^{-1/2}U^T H(1)V\Sigma^{-1/2}$$
$$B=\textit{ first m columns of }\Sigma^{1/2}V^T$$
$$C=\textit{ first r rows of }U\Sigma^{1/2}$$

where m is the number of inputs and r is the number of outputs.

3.3 Control Law Design

There are many control law design methods available. These range from classical control law design to the LQG method to H_∞ robust control law design methods to nonlinear control law design methods. Because of its ease of use, the LQG design method is employed in this dissertation. Basically the LQG design process involves minimizing a cost function of the form

$$J = \int_0^\infty [y^T Q y + u^T R u] dt$$

where y is an output vector (e.g. accelerations, loads) of the system and u is the control input. The resulting control law is u=-Gx where G is the state feedback gain matrix that minimizes the cost function and x is the state vector. Since the states of a system are generally not all available for feedback, a Kalman filter is employed to estimate the states. The resulting control law is of the form:

$$\{\dot{\hat{x}}\} = [\overline{A}]\{\hat{x}\} + [L]\{y\}$$
$$\{u\} = [-G]\{\hat{x}\}$$

where $\{\hat{x}\}$ is the estimate of the state vector x and L is the Kalman filter gain matrix.

3.4 Computer Implementation of Methodology

The computations for the methodology are performed on two different computers. The CAP-TSD simulations are performed on a Silicon Graphics Incorporated (SGI) computer and the other computations are performed on a Macintosh G-4 computer. Data is transferred between the computers using FTP.

The first step in the methodology is to calculate the aeroelastic outputs (uncontrolled) for a prescribed control surface input using the CAP-TSD program. Time histories of the outputs (modal displacements, velocities, and accelerations) and the control surface input are saved to a file on the SGI computer. The file is transferred (by FTP) to a Mac G-4 for system identification and control law design. The system identification and the control law design are performed in Matlab. The Matlab m-files used for the overall process are given in Appendix B.

After the outputs and inputs are loaded into Matlab, the OKID Toolbox [62] is used to perform the system identification. The system ID m-file provides for plotting the output of the system ID model (state-space model) compared against the CAP-TSD outputs. The primary design variable for the system ID process is the maximum order of the model. The maximum order is a product of an input parameter times the number of outputs. The input parameter is varied until a desired match between the system ID model outputs and the CAP-TSD outputs is obtained. After the system ID state-space model is obtained, a full-order LQG control law is designed. The Matlab m-file provides for simulating (using the state-space model) the controlled response and comparing to the uncontrolled response. The m-file also provides for calculating the open-loop Bode diagram for determining stability margins. After a control law is designed that provides the desired characteristics (controlled response and stability margins), it is discretized at the CAP-TSD integration time interval and written to a file on the Mac G-4. This file is then transferred to the SGI computer and is used as input to CAP-TSD to perform a control law evaluation.

The modifications to CAP-TSD to accept a control law and perform a controlled simulation will be described next.

3.4.1 Modifications to CAP-TSD to Perform Controlled Simulations

The major modifications to the CAP-TSD computer program to perform controlled (closed-loop feedback control) simulations are input to describe the control equations and sensor information, integration of the control equations, and modification of the control surface downwash to account for the feedback control.

Most of the input to CAP-TSD is in the form of namelists. Therefore, a new namelist called CONSYS was added that reads the following variables.

NSENSOR — number of feedback sensors

NCST — number of states defining the control law

CSEN(i,j) — modal amplitude at the sensor locations,

i=1,NSENSOR, j=1,NMODES

ACONTRL — control law state matrix (size NCST x NCST)

BCONTRL — control law input matrix (size NCST x NSENSOR)

CCONTRL — control law output matrix (size 2 x NCST)

DCONTRL — control law direct input matrix (size 2 x NSENSOR)

IREAD — parameter that indicates how the control law matrices are read

=0 read from NAMELIST CONSYS

=1 read from file fort.77

```
read(77,*) ((ACONTRL(i,j),I=1,NCST),j=1,NCST)
read(77,*) ((BCONTRL(i,j),I=1,NCST),j=1,NSENSOR)
read(77,*) ((CCONTRL(i,j),I=1,2),j=1,NCST)
read(77,*) ((DCONTRL(i,j),I=1,2),j=1,NSENSOR)
```

maximum dimensions:

NCST=20

NSENSOR=5

In this implementation, there are the following assumptions:

1. Control law matrices ACONTRL, BCONTRL, CCONTRL, and DCONTRL have been discretized at the CAP-TSD sample rate dt (real time).
2. Assumes only one control surface.
3. The second row of the control law output equations is the rate of the control surface rotation command (the first equation is the control surface rotation command).

The control law equations are of the form:

$$\{xc\ (k+1)\} = [ACONTRL]\ \{xc\ (k)\} + [BCONTRL]\ [Y]$$

$\{\delta \; \& \; \delta\text{-dot}\,(k)\} = [\text{CCONTRL}]\,\{xc\,(k)\} + [\text{DCONTRL}]\,[Y]$

δ = feedback control surface rotation command

δ-dot = feedback control surface rotation command rate

Y = vertical acceleration (for this research)

The δ and δ-dot are propagated in time using the above equations and used in the downwash subroutine to modify the control surface downwash at both the predictor and corrector stage of the structural integration within CAP-TSD. Acceleration at the sensor locations are computed at each time increment and used as input to the control law equations.

The next chapter describes the application of the methodology to flutter control of a wind-tunnel model wing. Numerical results are given to illustrate the advantages of this methodology over designing control laws from state-of-the-art linear models.

Chapter 4

Results and Discussion

The example employed in this dissertation to demonstrate the overall control law design methodology is active flutter suppression for the Benchmark Active Controls Technology (BACT) wind-tunnel model shown in Fig. 4.1 [66]. Selecting this model has the advantages of available analytical and experimental data and control law design results for linear models. In addition, several of the researchers who have considered control law design for a CAS have used the BACT model [53,54,56,57].

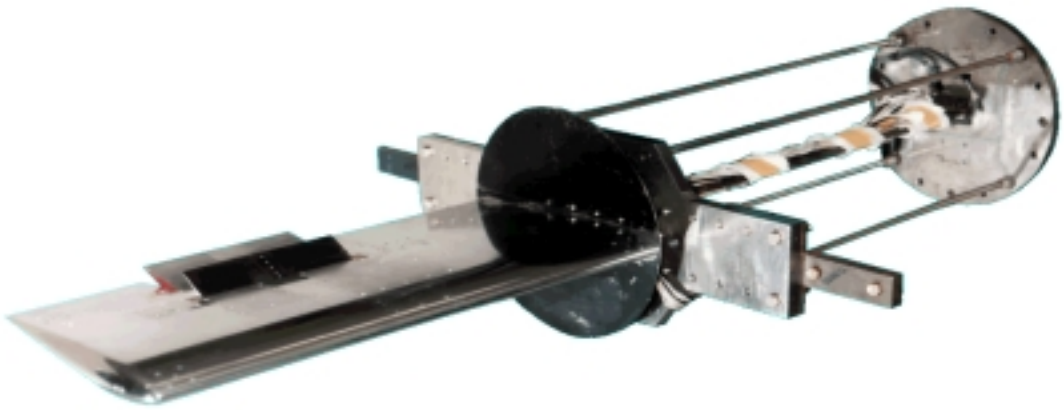

Figure 4.1 Photograph of BACT Wind-Tunnel Model

4.1 Description of Physical and Mathematical Models

The BACT model is a rigid, rectangular wing with a NACA 0012 airfoil section. The rectangular wing has a span of 0.812 m and a chord of 0.406 m and therefore an aspect ratio of 2. It is equipped with a trailing-edge control surface and upper and lower surface spoilers that are controlled independently by hydraulic actuators. Only the trailing-edge control surface is employed in this dissertation. The span of the control surfaces is 30% of the wing span, centered about the wing 60% span station. The trailing-edge control surface has a chord of 25% of the wing chord. For this study, actuator dynamics are ignored. For the control law designs, an accelerometer located near the outboard trailing edge is the assumed sensor employed for feedback. The wing is mounted to a device called the Pitch and Plunge Apparatus (PAPA) which is designed to permit motion in principally two modes – pitching and vertical translation (plunge). Therefore, the BACT has structural dynamic behavior very similar to the classical two degree-of-freedom problem in aeroelasticity. The vibration frequencies, computed from a NASTRAN model of the BACT, are 3.4 Hz (plunge) and 5.2 Hz (pitch). Structural damping is assumed to be zero. There are two CAP-TSD aerodynamic representations of the BACT wind-tunnel model used in this dissertation. The first one, shown in Fig. 4.2, is a 3-d model whose CFD grid contains 140 points in the chordwise direction, 40 points in the spanwise direction, and 92 points in the vertical direction for a total of 515,200 grid points. The second one is an equivalent 2-d model whose CFD grid contains 140 points in the chordwise direction, only 2 points in the spanwise direction, and 92 points in the vertical direction for a total of 25,760 grid points.

Although some aerodynamic data and basic flutter calculations will be presented for the 3-d model, most of the control law design and evaluation process results will be demonstrated with the 2-d model. However, the exact same process would be applied to

the 3-d case as illustrated with a few examples near the end. The results will begin with some basic steady aerodynamic data and then proceed to uncontrolled flutter calculations, and finally to controlled flutter calculations.

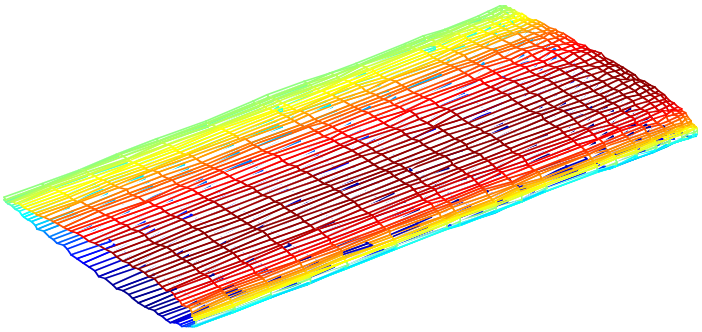

Figure 4.2 CAP-TSD Mesh of BACT Model

4.2 Basic CAP-TSD Results Compared with Experiment

This section compares the output of the CAP-TSD code with some experimental data. Comparisons of aerodynamic pressure distributions for both angle of attack variations and control surface deflection variations will be presented. Finally, the flutter dynamic pressure predicted using the CAP-TSD code is compared with the experimental value.

4.2.1 Rigid Aerodynamic Results

In order to calculate rigid aerodynamic pressures, equation 3.2 is integrated in time by CAP-TSD without coupling the structural dynamics equations (i.e. no aeroelastic effects). Figure 4.3 shows the chordwise pressure distribution at the 60% span location at M=0.77 for 0, 2, and 5 degrees angles of attack. Results from CAP-TSD are shown with the experimental results of Ref. 66. In addition, linear CAP-TSD results are shown for the 2 degrees angle of attack case (Fig. 4.3 (b)).

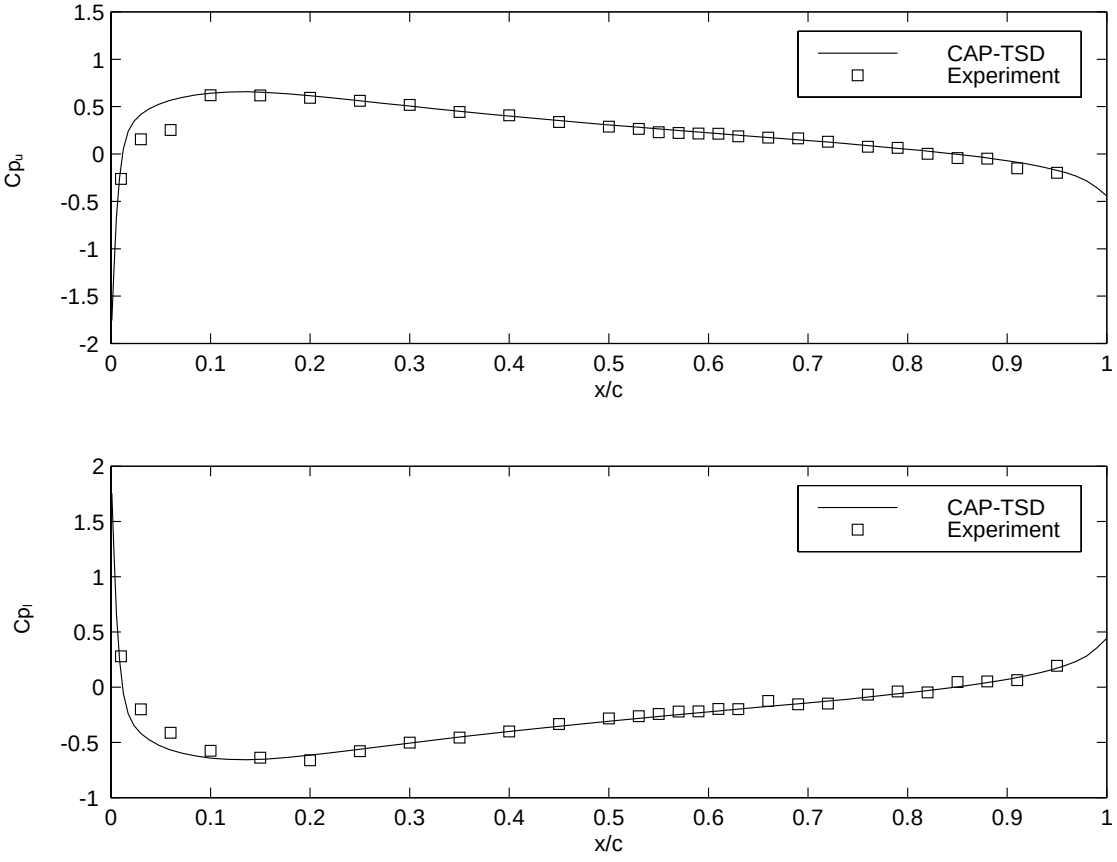

Figure 4.3(a). Aerodynamic results α=0 degree, M=0.77, 60% semi-span

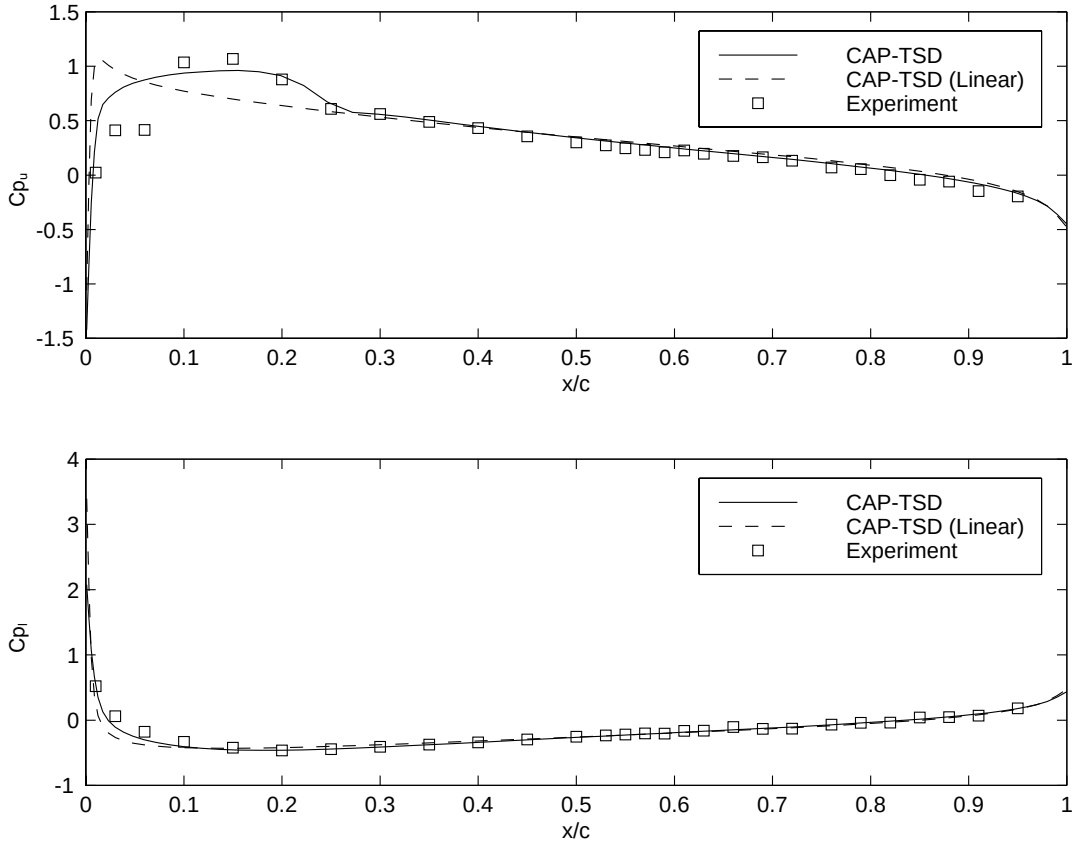

Figure 4.3(b). Aerodynamic results α=2 degrees, M=0.77, 60% semi-span

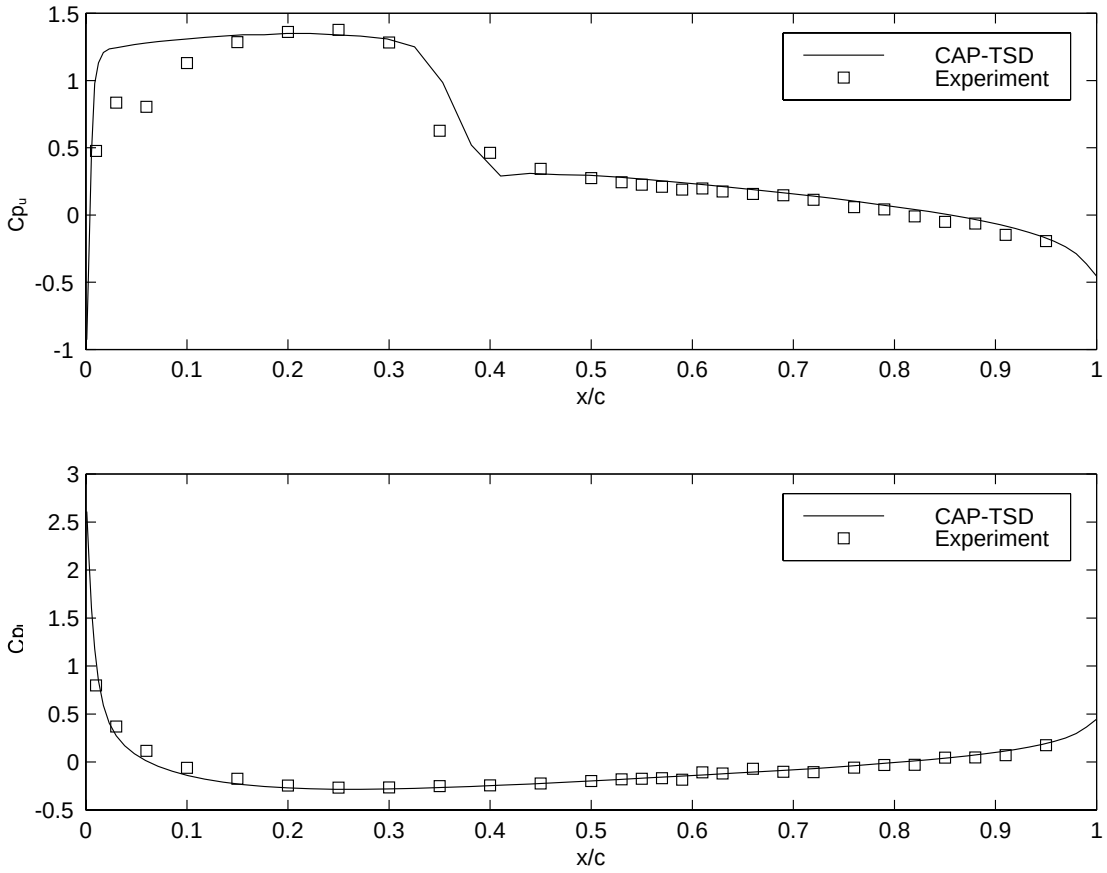

Figure 4.3(c). Aerodynamic results α=5 degrees, M=0.77, 60% semi-span

Beginning at 2 degrees angle of attack, a weak shock near the 20% chord has developed indicative of transonic flow. At 5 degrees angle of attack, the shock has moved aft to approximately the 35% chord and become moderately strong. The CAP-TSD results capture the shock location and strength reasonably well. Of course, the linear CAP-TSD results do not indicate a shock and are not in agreement with the experimental data on the upper surface forward of the shock. However, both the CAP-TSD and linear CAP-TSD results compare very well with the experimental data aft of the 40% chord on the upper surface and compare well along the entire chord for the lower surface. The apparent dip in the experimental upper surface pressures near the leading edge has been attributed to

the transition strip at that location [66]. These results provide some degree of confidence in the aerodynamic predictions of CAP-TSD for angle-of-attack changes.

Figure 4.4 shows the chordwise pressure distribution at the 60% span location at M=.77 for trailing-edge control surface deflections of 2 and 5 degrees. Again, results from CAP-TSD are shown with the experimental results of Ref. 66.

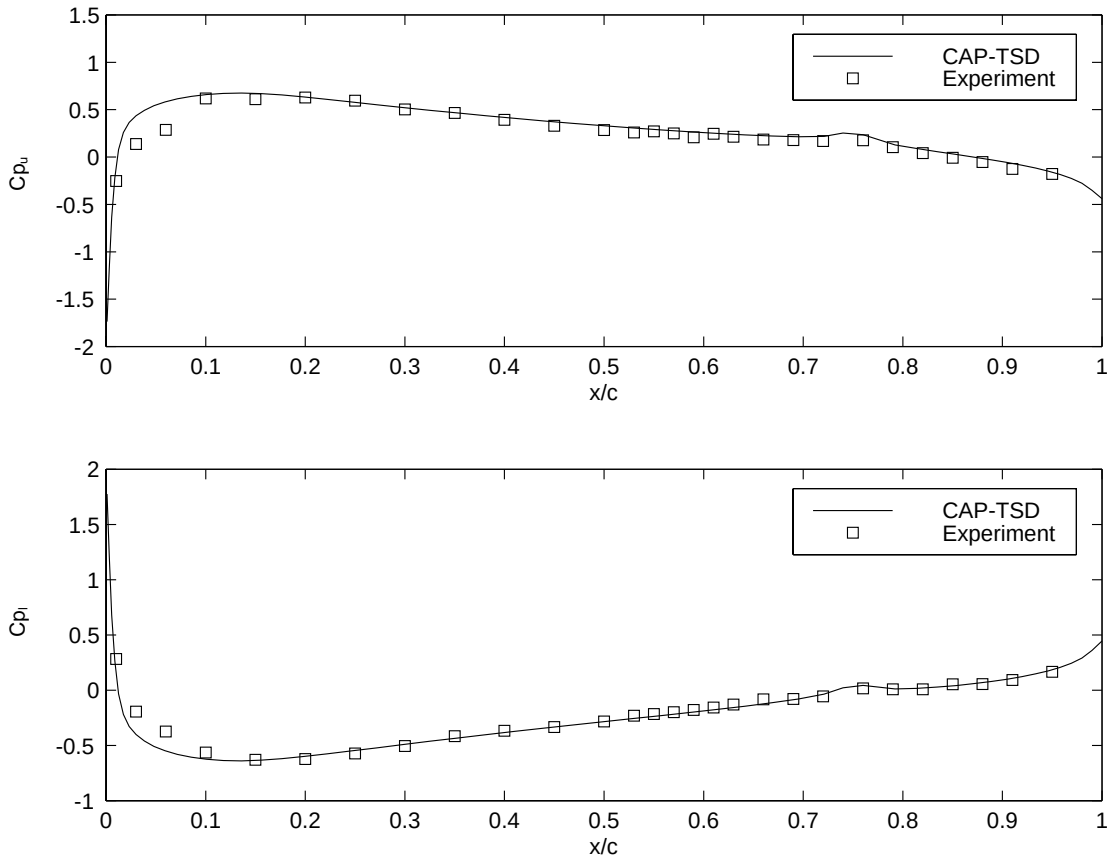

Figure 4.4(a). Aerodynamic results δ=2 degrees, M=0.77, 60% semi-span

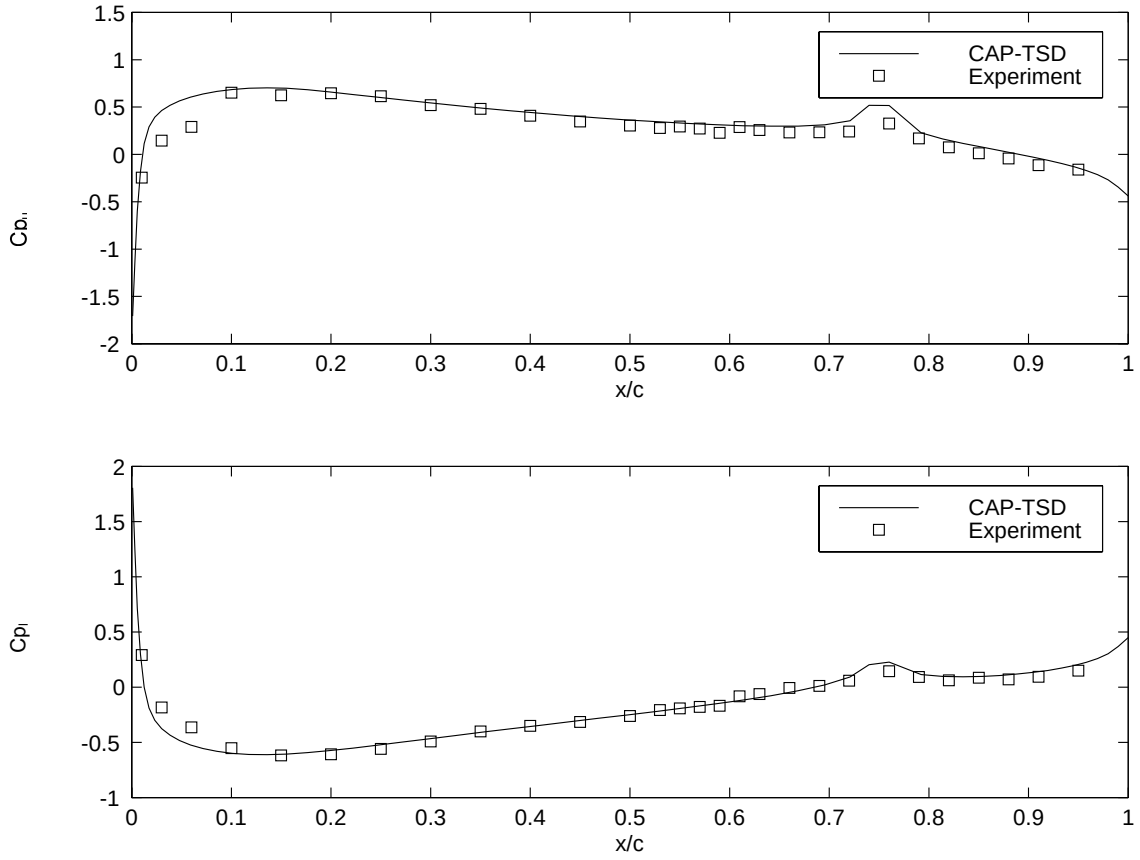

Figure 4.4(b). Aerodynamic results δ=5 degrees, M=0.77, 60% semi-span

CAP-TSD slightly overpredicts the supper surface pressure peak near the control surface hinge line for δ=2 degrees. CAP-TSD considerably (approximately 50%) overpredicts the upper surface pressure peak for δ=5 degrees (Fig. 4.4 (b)). Overprediction of control surface forces is typical for inviscid codes. Therefore, care should be used in real applications to account for this overprediction when using an inviscid code.

4.2.2 Uncontrolled Flutter Results

For dynamic aeroelastic analyses in a computational aeroelasticty environment, two steps are employed in performing the calculations [33]. In the first step, a static aeroelastic deformation is calculated to provide the initial flowfield for the dynamic aeroelastic

solution. The dynamic solution is a perturbation about a converged static aeroelastic solution for each Mach number and dynamic pressure of interest. The procedure for calculating static aeroelastic solutions is to allow the structure and aerodynamics to interact with no initial condition on the modal displacements and velocities and no external excitation. A very large value (0.99) for structural damping is used to prevent divergence of the solution and to accelerate convergence. This method results in convergence of the steady-state plunge and pitch displacements, velocities, and accelerations. Figure 4.5 shows a typical time history of the plunge and pitch displacements for a static aeroelastic solution.

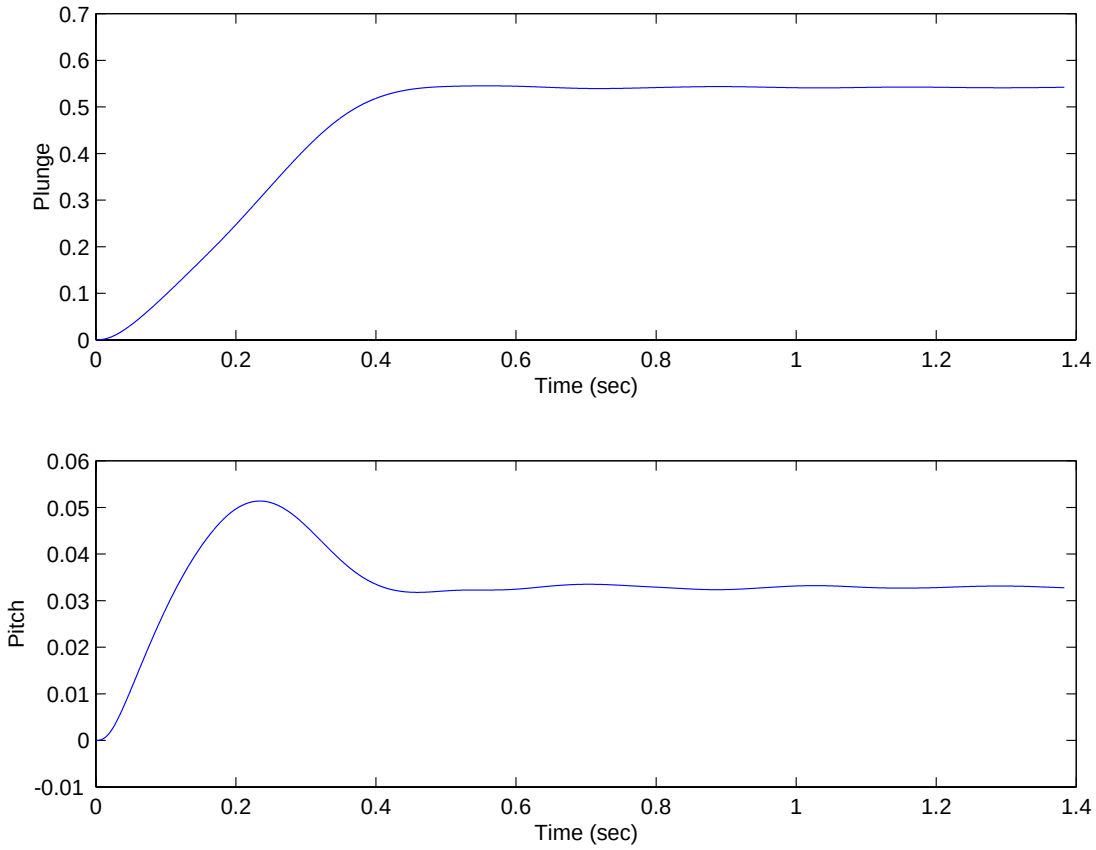

Figure 4.5 Typical static aeroelastic solution, α=2 deg

The plunge displacement rises smoothly to its steady-state value in the first 0.4 seconds. The pitch displacement has an overshoot at approximately 0.2 seconds and then also reaches its steady state value at approximately 0.4 seconds.

Once a static aeroelastic solution is computed, the next step is to prescribe either an initial condition on the displacements or velocities or an external input. For flutter calculations, initial conditions on the velocities are used to begin the dynamic structural integration. CAP-TSD was run using a value of 1 on both the plunge and pitch velocity initial conditions and then allowed to decay freely. Figure 4.6 shows typical time histories of the plunge and pitch displacement generalized coordinates.

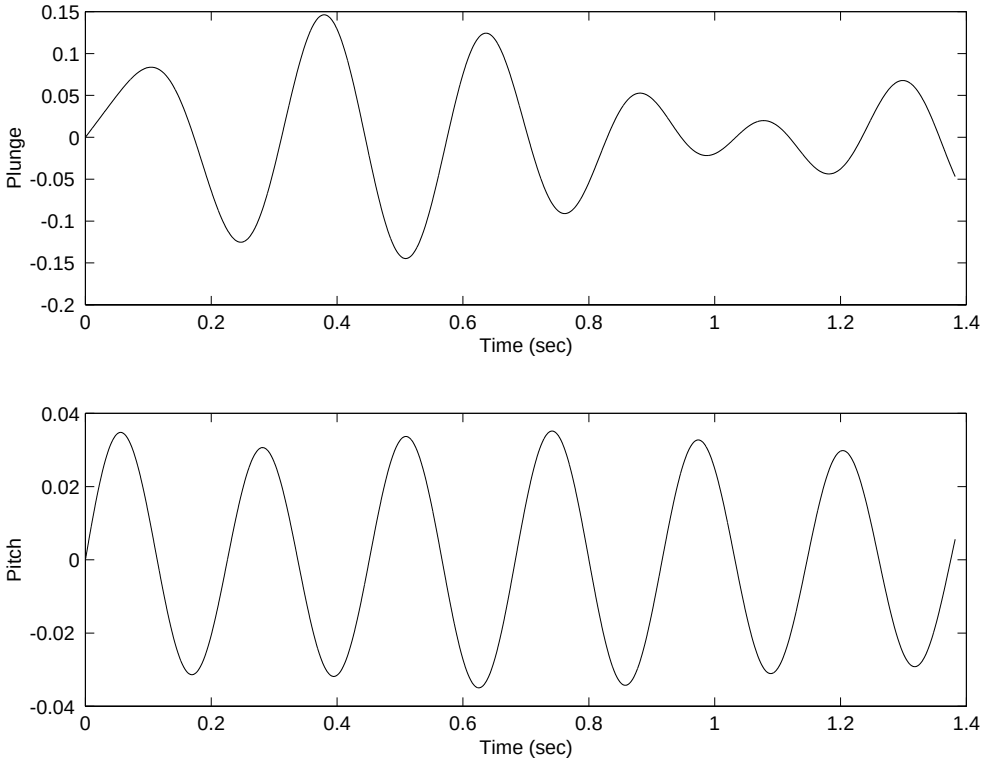

Figure 4.6 Typical plunge and pitch displacement time histories

The ERA system ID method of Ref. 65 was employed to estimate modal dampings and frequencies at various values of dynamic pressure. Figure 4.7 shows the dampings as a function of dynamic pressure (M=0.77) for a Doublet Lattice linear model, the CAP-TSD 3-d linear model, and CAP-TSD 3-d model.

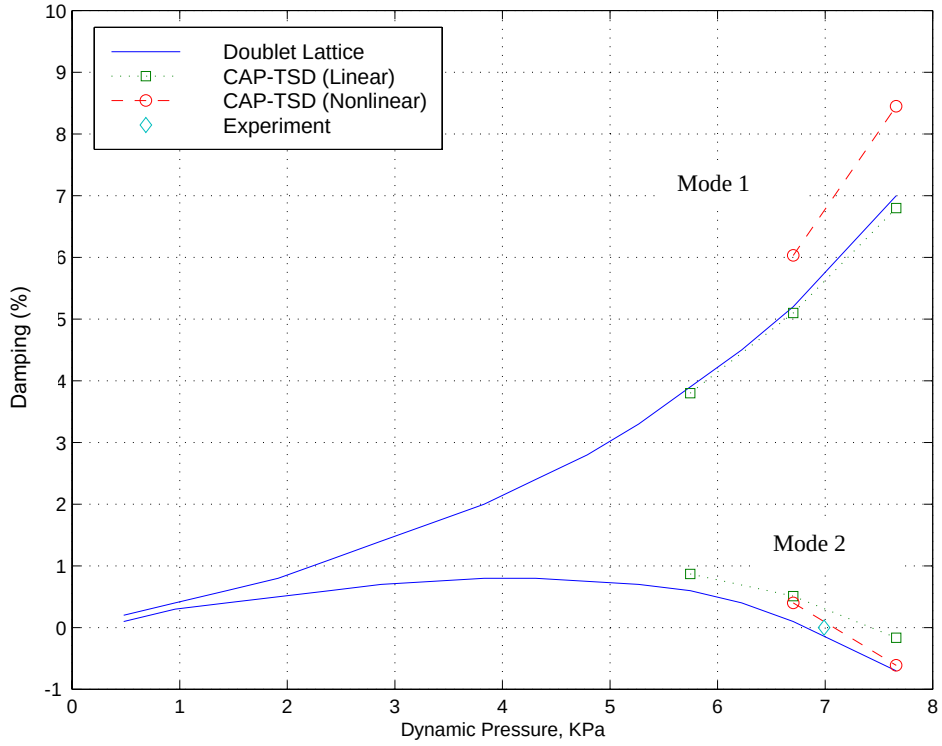

Figure 4.7 Dampings as a function of dynamic pressure, 3-d

The CAP-TSD 3-d linear model shows approximately a 9% greater flutter dynamic pressure than the Doublet Lattice linear model. The Doublet Lattice linear model shows approximately a 3% lower flutter dynamic pressure than the experimental value. The CAP-TSD linear model shows approximately a 6% greater flutter dynamic pressure than the experimental value. The CAP-TSD model shows approximately a 2% greater flutter dynamic pressure than the experimental value.

4.3 Basic CAP-TSD 2-d Results

This section presents steady aerodynamic results, uncontrolled flutter results, initial system identification results, and results illustrating the sensitivity of the system ID model. All of the results are for a Mach number of 0.77 and employ the CAP-TSD 2-d model previously described.

4.3.1 Steady Aerodynamics

Figure 4.8 shows the upper surface chordwise pressure distribution for 0, 0.3, and 0.6 degrees angles of attack. Pressure distributions are given for both the rigid and flexible case. The flexible case is for a dynamic pressure of 5.75 kPa and is calculated using the static aeroelastic solution technique previously described. As the angle of attack is increased from 0 to 0.6 degrees, a shock forms at approximately the 25% chord for the rigid case. The shock forms earlier and moves further aft (approximately 45% chord) for the flexible case. This is due to the fact that the effective angle of attack increases for the flexible case.

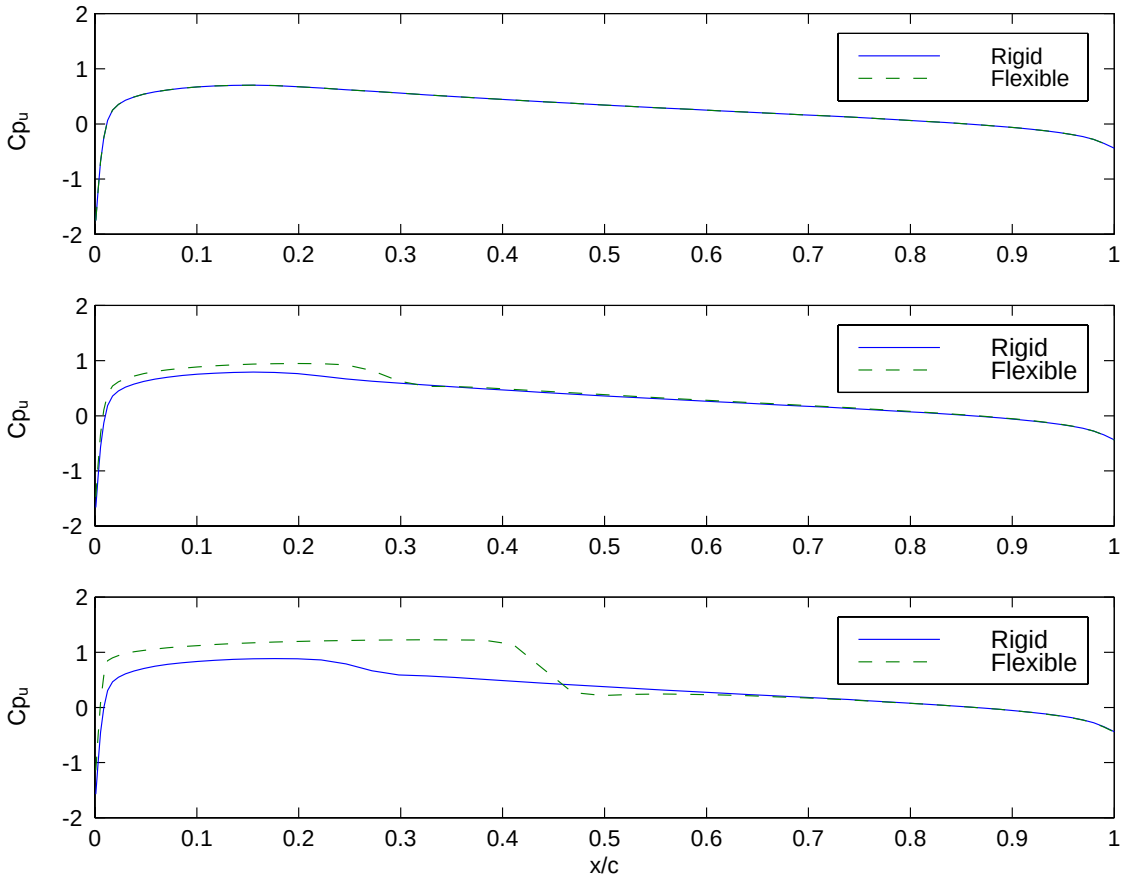

Figure 4.8 Rigid and flexible chordwise pressure distributions, 2-d

4.3.2 Uncontrolled Flutter Results

The ERA system ID method was also employed to estimate modal dampings and frequencies for the 2-d cases. Damping, as a function of dynamic pressure, is shown in Fig. 4.9 for the 2-d case. The overall character of the curves is very similar to the 3-d case. The CAP-TSD linear 2-d model shows a flutter dynamic pressure which is approximately 19% lower than the corresponding linear 3-d case. The CAP-TSD 2-d model shows a flutter dynamic pressure that is approximately 21% lower than the

corresponding nonlinear 3-d case. This magnitude of reduction between 3-d and 2-d is similar to that described in Ref. 67.

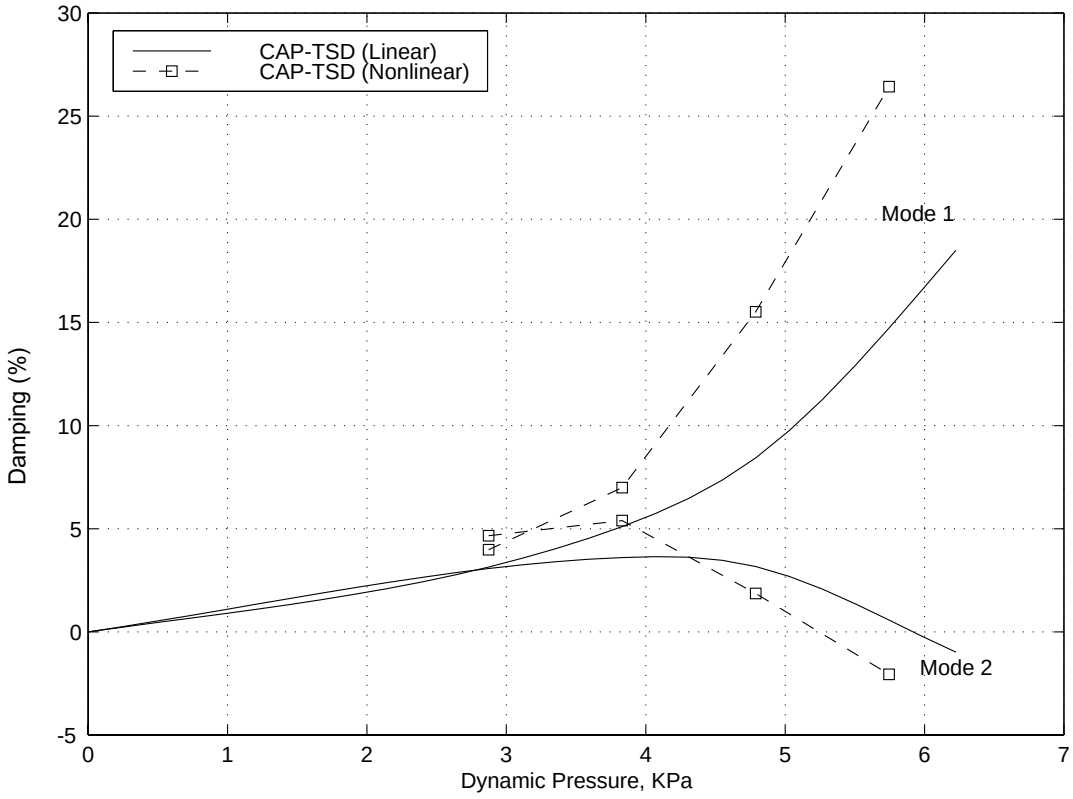

Figure 4.9 Damping as a function of dynamic pressure, 2-d

4.3.3 Initial System ID Results

In order to perform a system identification (ID), an appropriate control surface input signal must be used to excite the system modes. After exploring several input signals

(sinusoid sweep and random), the exponential pulse signal shown in Fig. 4.10 provided a good input signal for identifying an equivalent linear model of the CAP-TSD outputs.

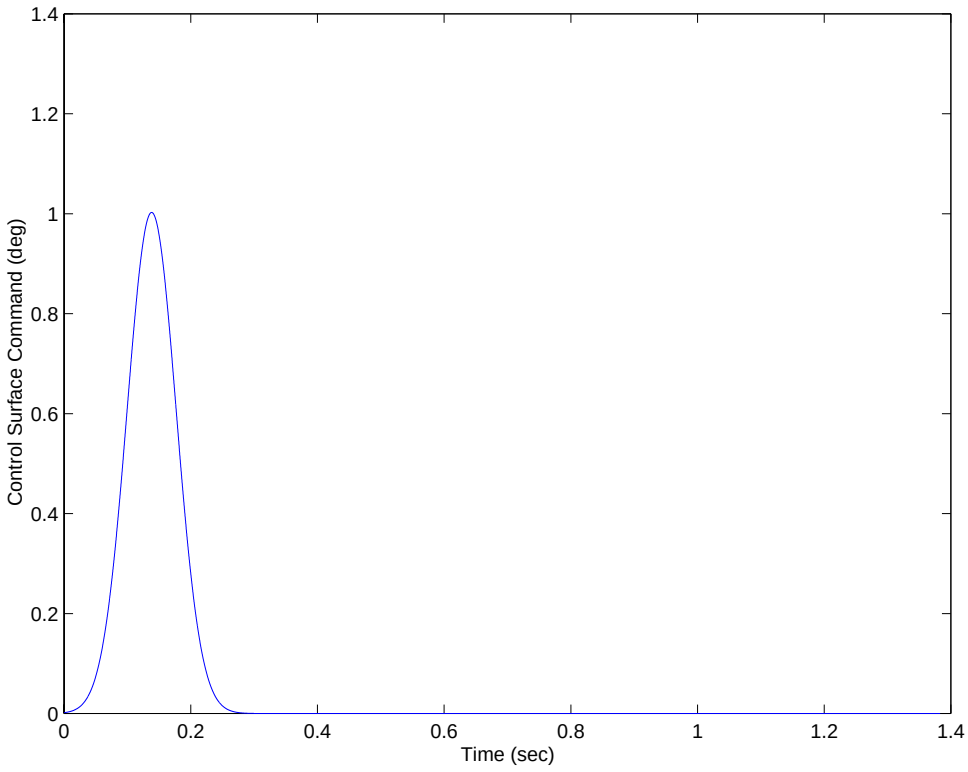

Figure 4.10 Exponential pulse control surface input

In order to test the system ID technique, CAP-TSD was run in the linear mode for several of the initial results that will be shown. A dynamic pressure of 5.75 kPa was chosen to identify all of the 2-d system ID models. This dynamic pressure is between the flutter dynamic pressure predicted by CAP-TSD (linear) and the flutter dynamic pressure predicted by CAP-TSD (nonlinear). The linear system ID models were identified using the plunge and pitch displacements, velocities, and accelerations as the measurement signals. Figure 4.11 shows the results from the system ID procedure in terms of a

comparison between the CAP-TSD outputs (linear) and two system ID model outputs (using the same exponential pulse input).

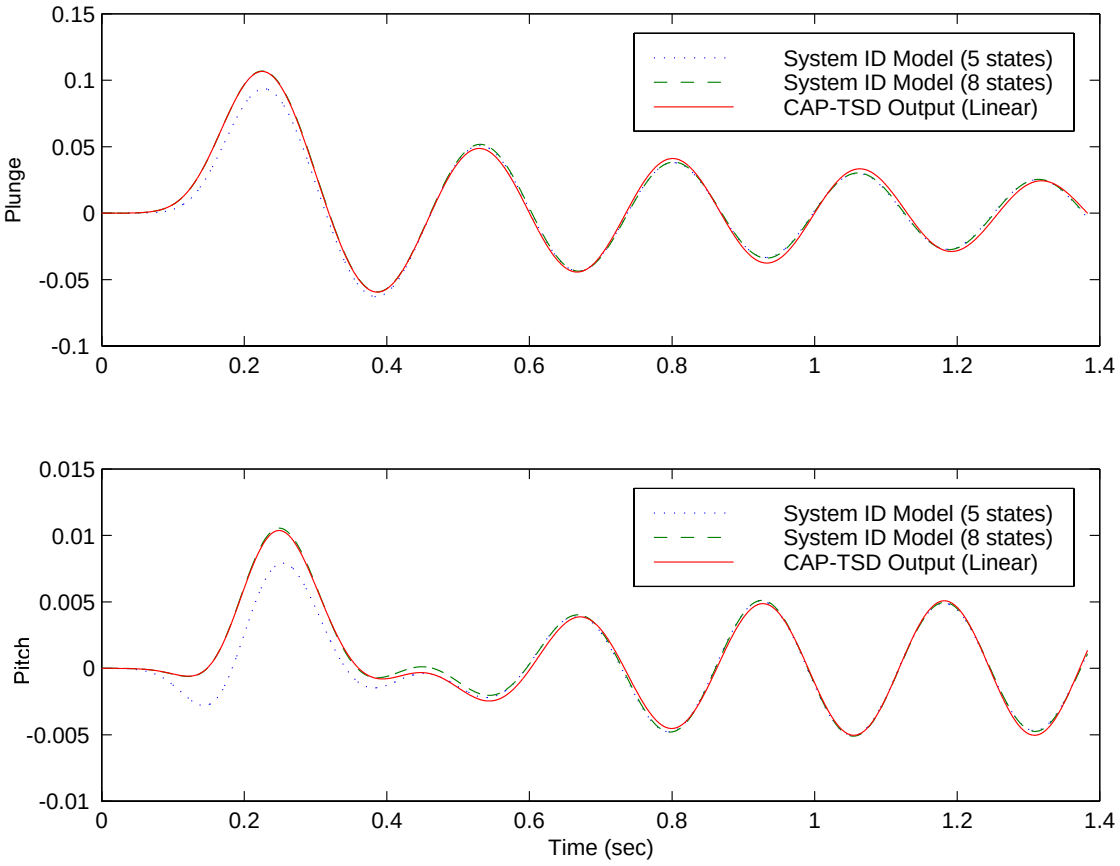

Figure 4.11(a). System ID results for displacements

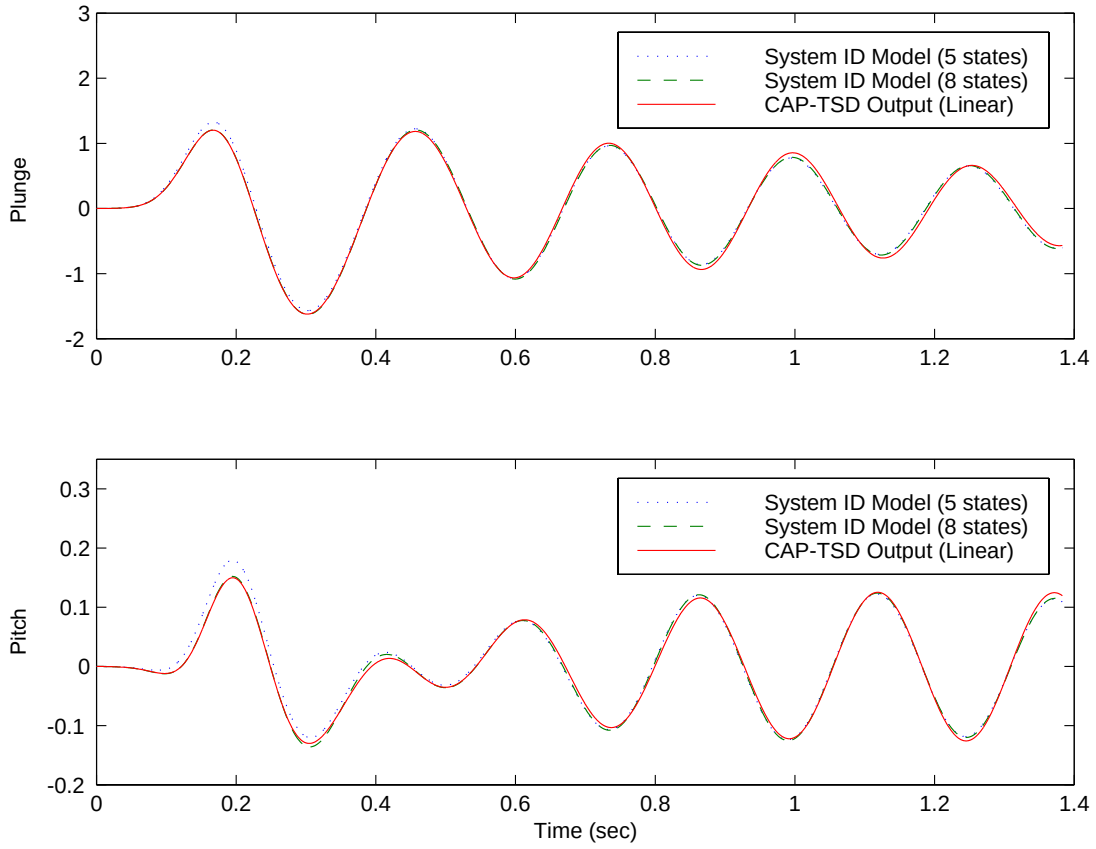

Figure 4.11(b). System ID results for velocities

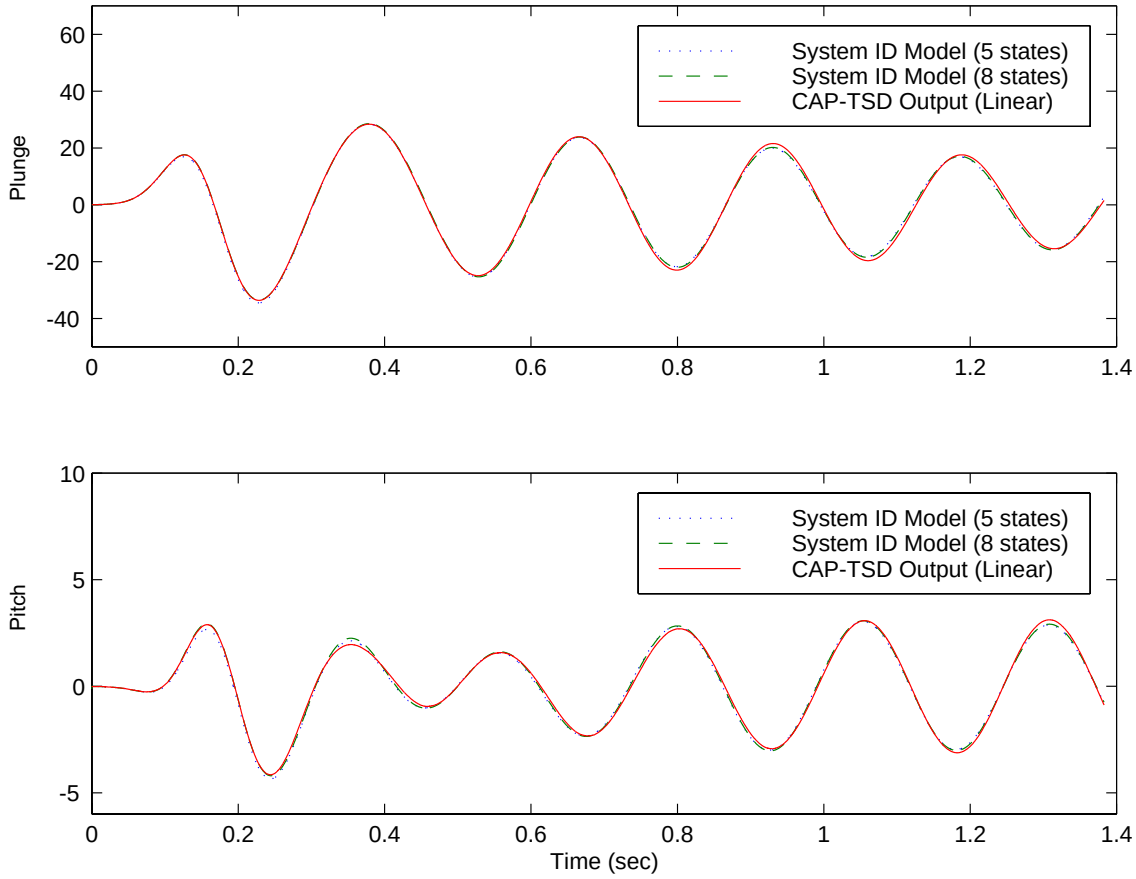

Figure 4.11(c). System ID results for accelerations

A 5-state system ID model was the first one identified. This 5-state model matches the CAP-TSD plunge responses well. However, it does not match the CAP-TSD pitch response well during the exponential pulse time period. The order of the system ID model was increased and an 8-state model was identified. As can be seen in Fig. 4.11, the 8-state system ID model provides a very good representation of the CAP-TSD results for both the plunge and pitch responses over the entire time history. Several additional system ID models with higher orders were identified, but no overall improvement (over the 8-state model) in the ability to match the CAP-TSD results were seen. Most of the subsequent results employ an 8-state system ID model.

In order to assess the sensitivity of the system ID model to changes in the control input signal, the system ID model (derived from the exponential pulse input) was used in a simulation where the control surface input was a sine wave. This result employed the outputs of CAP-TSD run in the nonlinear mode (including the aerodynamic nonlinear terms). Both the outputs used as input to system ID procedure and the simulation for the sine wave input were for the nonlinear CAP-TSD mode. Figure 4.12 shows results of the simulation using a 5 Hz and 10Hz sine wave input in terms of a comparison of the system ID model with the CAP-TSD output.

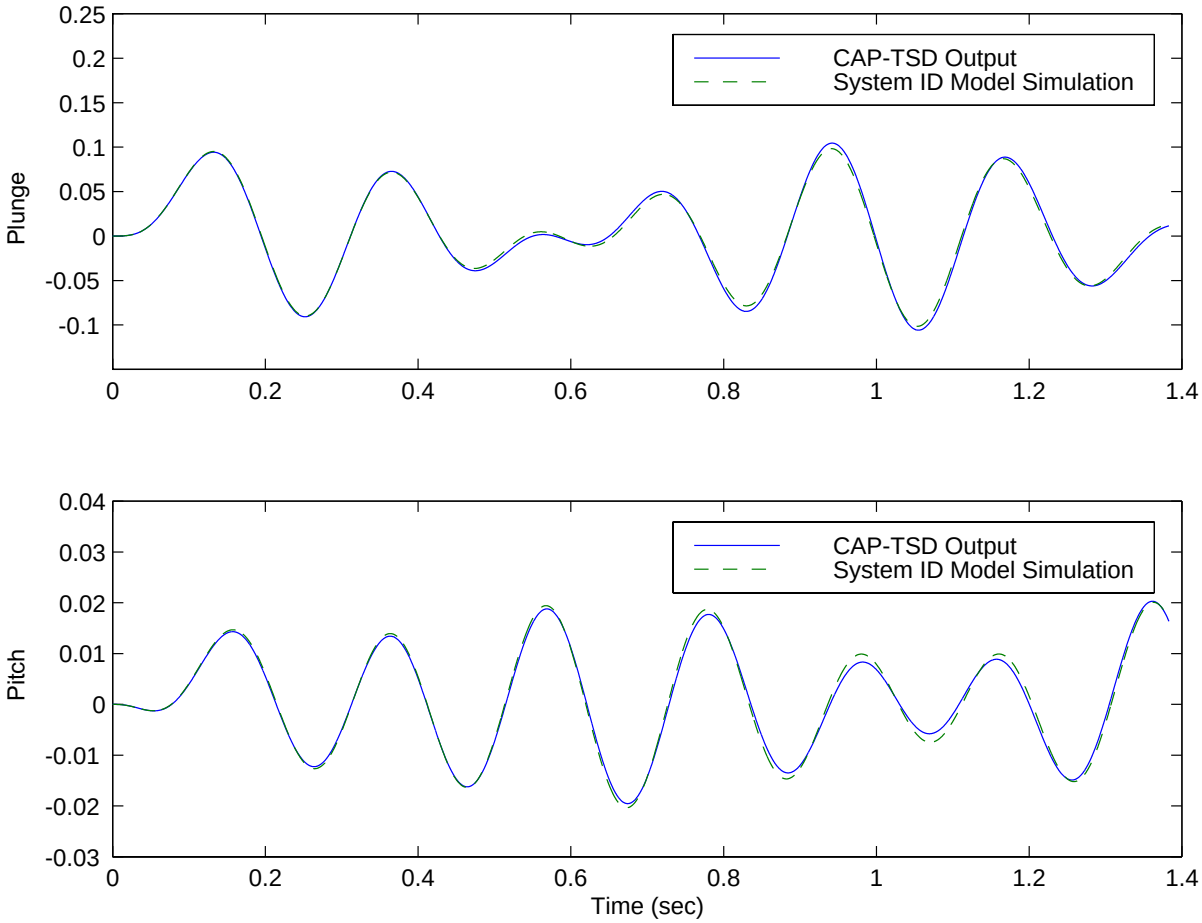

Figure 4.12(a). Comparison of displacement responses for a 5 Hz sine wave input

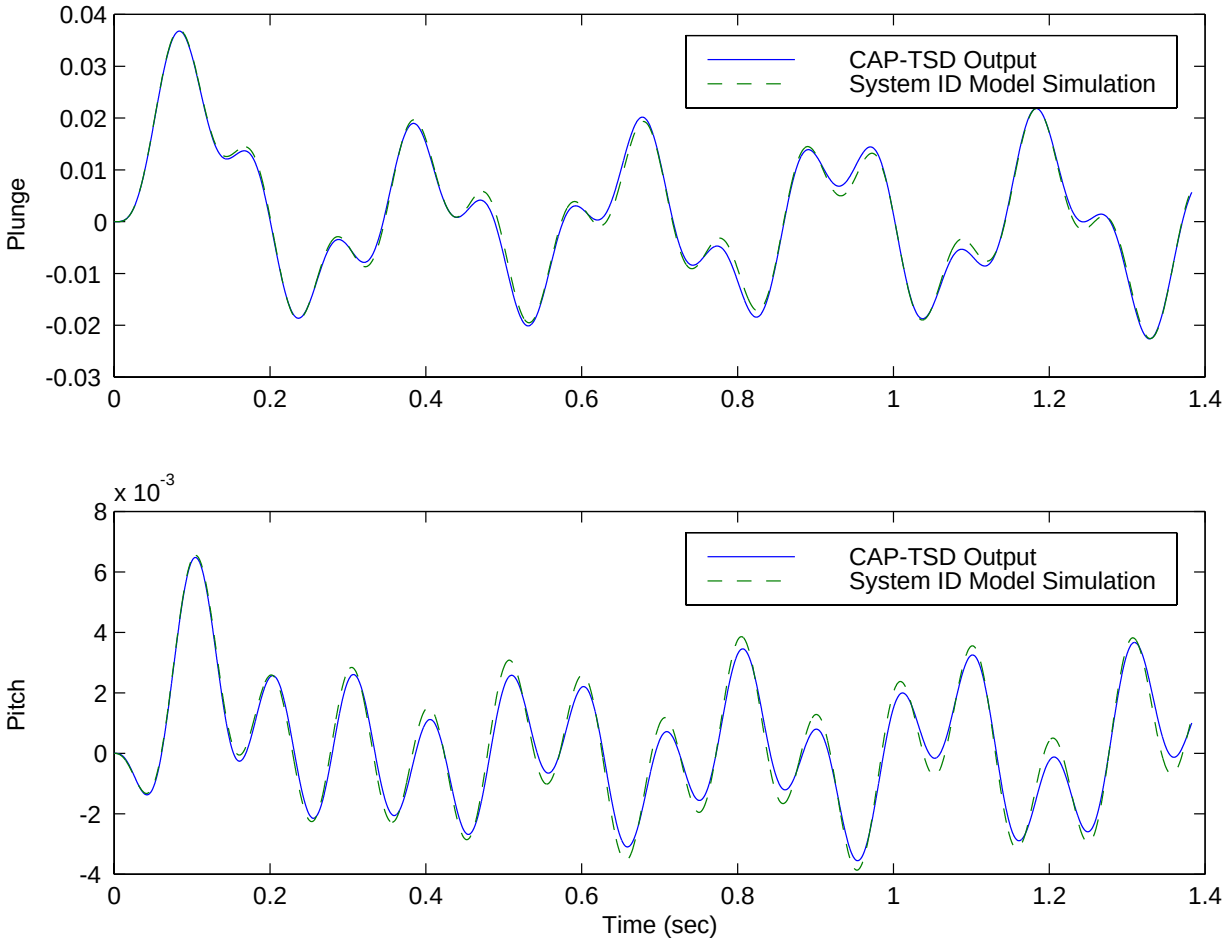

Figure 4.12(b). Comparison of displacement responses for a 10 Hz sine wave input

Only the plunge and pitch displacements are shown. The agreement between the system ID model and the CAP-TSD outputs are excellent for the 5 Hz case. There is some slight mismatch of the peaks (especially for the pitch displacement) between the system ID model and the CAP-TSD outputs for the 10 Hz case. However, the overall ability of the system ID model to match the CAP-TSD outputs for these different inputs is quite good.

4.3.4 Angle of Attack and Control Surface Amplitude Effects

These results are also computed at a dynamic pressure of 5.75 kPa and employ the outputs of CAP-TSD in the nonlinear mode (including the aerodynamic nonlinear terms).

The plunge and pitch displacement responses (using the exponential pulse control input) for the three angles of attack that were used for the steady aerodynamics calculations are shown in Fig. 4.13.

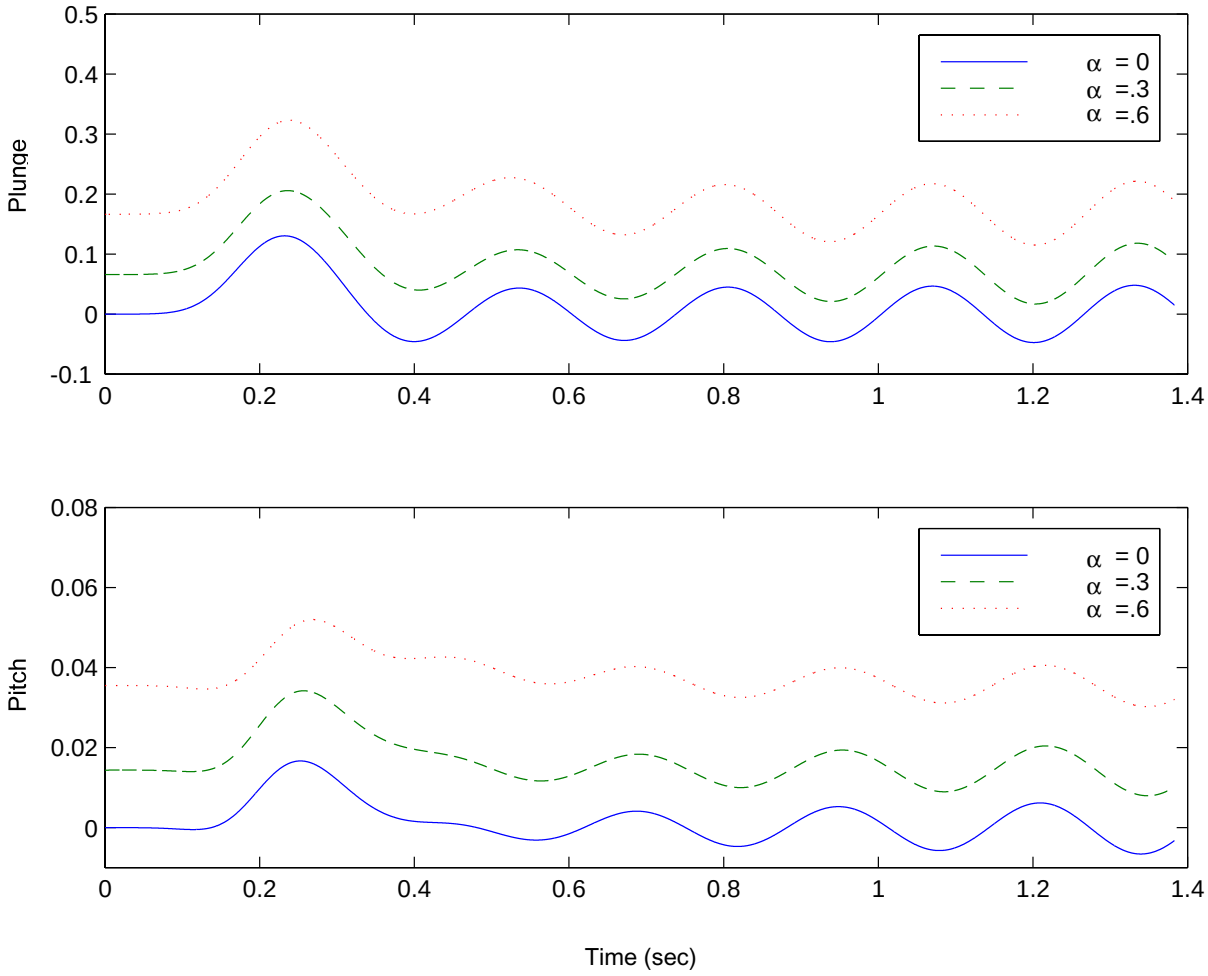

Figure 4.13 Plunge and pitch displacement responses for angle of attack changes

The effect of the static aeroelastic solution (nonzero value at time=0) is seen for the 0.3 and 0.6 degrees angle of attack results. As angle of attack is increased, the static aeroelastic values of the plunge and pitch displacements increase. The dynamic

aeroelastic results for the plunge displacement are very similar. The pitch displacements are very similar for the 0 and 0.3 degrees angle of attack. However, the oscillatory nature of the response begins earlier in time for the 0.6 degrees angle of attack. Modal damping and frequencies for all three angles of attack will be presented in Section 4.4.

In order to assess the nonlinearity of the system from an output-to-input point-of-view, the outputs of CAP-TSD computed with a control input of 2 degrees were compared with the outputs of CAP-TSD with a control input of 1 degree multiplied by 2. For a linear system (over the range from 0 to 2 degrees of control surface deflection), outputs of CAP-TSD computed with a control input of 2 degrees would be equal to the outputs of CAP-TSD with a control input of 1 degree multiplied by 2. The effect of control surface amplitude on the pitch response for the three angles of attack (0, 0.3, and 0.6) is shown in Fig. 4.14. Overall, the system is quite linear over this range of angles of attack. There is some small nonlinear effect for the 0.6 degrees case at values of pitch displacement greater than approximately 0.04 (this corresponds to a flexible angle of attack of approximately 2 degrees).

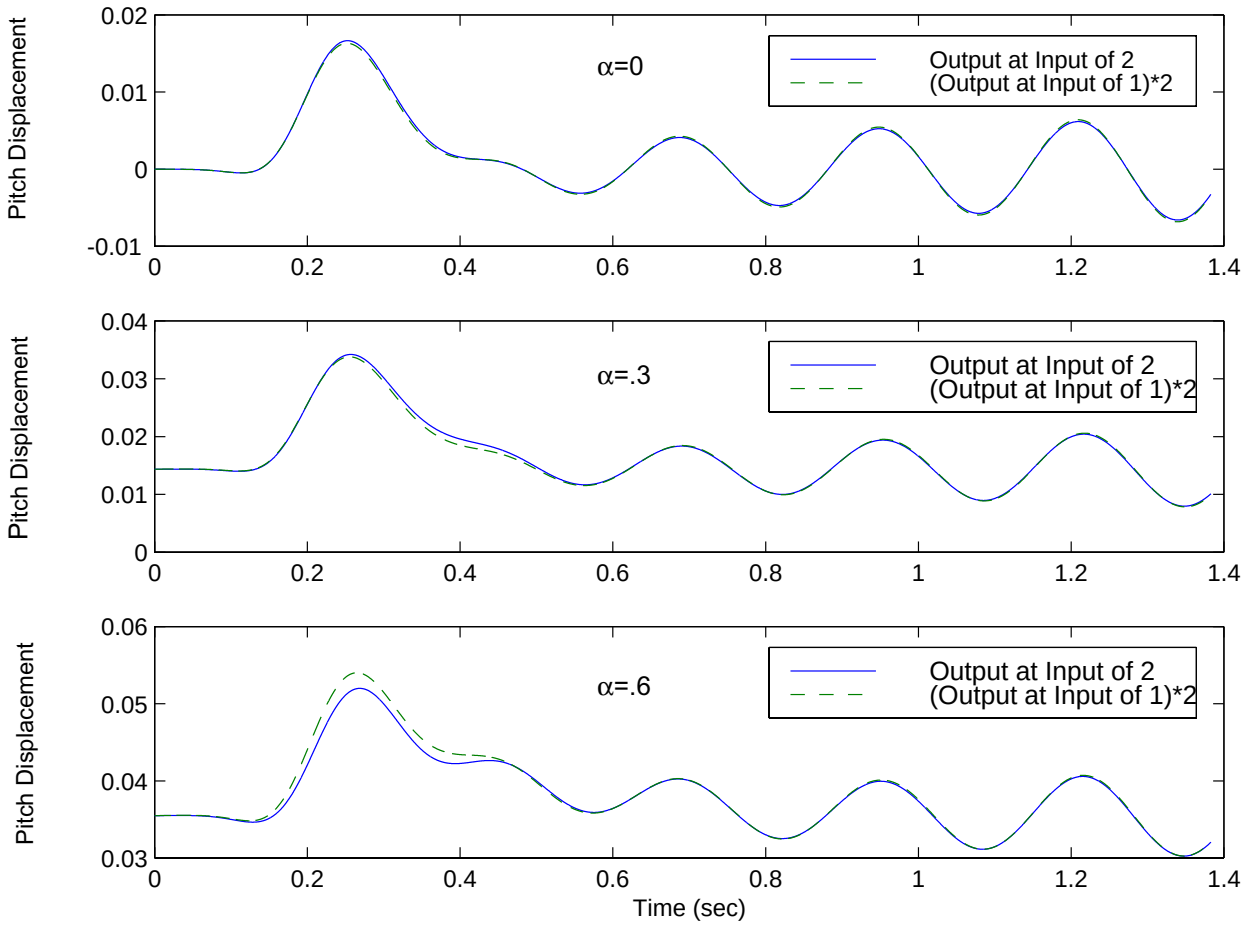

Figure 4.14 Effect of control surface amplitude on pitch response

4.4 Control Law Design and Evaluations

Ten different cases will be studied to illustrate the design methodology. Table 4.1 is a summary of the different cases that will be described. The first seven cases employ the CAP-TSD 2-d model previously described. The last three cases employ the CAP-TSD 3-d model. All of the cases are for a Mach number of 0.77. The 2-d cases are for a dynamic pressure of 5.75 kPa and the 3-d cases are for a dynamic pressure of 7.37 kPa.

Table 4.1. Control Law Design Cases

Case	Mach	Simulation used for System ID	Control Law	Control Law Evaluation Simulation
1	0.77 (2-d)	Linear α=0	LQG-designed	Linear α=0
2	0.77 (2-d)	Nonlinear α=0	Case 1	Nonlinear α=0
3	0.77 (2-d)	Nonlinear α=0	LQG-designed	Nonlinear α=0
4	0.77 (2-d)	Nonlinear α=0.3	Case 1	Nonlinear α=0.3
5	0.77 (2-d)	Nonlinear α=0.3	LQG-designed	Nonlinear α=0.3
6	0.77 (2-d)	Nonlinear α=0.6	Case 1	Nonlinear α=0.6
7	0.77 (2-d)	Nonlinear α=0.6	LQG-designed	Nonlinear α=0.6
8	0.77 (3-d)	Linear α=0	LQG-designed	Linear α=0
9	0.77 (3-d)	Nonlinear α=0	LQG-designed	Nonlinear α=0
10	0.77 (3-d)	Nonlinear α=2	LQG-designed	Nonlinear α=2

The 2-d cases begin with the linear case at 0 degrees angle of attack and then proceed with nonlinear cases at 0 degrees angle of attack, 0.3 degrees angle of attack, and 0.6 degrees angle of attack. For each angle of attack, a system-ID model is developed and the control law designed with the system ID model for that angle of attack is compared with the control law for the linear case. The comparison is in terms of gain and phase margins, acceleration time history (using the exponential control input), and control surface deflection. In addition, a comparison of the eigenvalues for both the uncontrolled and controlled results for each case is presented. The control law designed with the linear model is intended to represent a control law using the state-of-the-art methodology described in Chapter 2. The 3-d cases are for 0 and 2 degrees angles of attack.

The primary design goal for all of the control law designs is to increase the damping of the system while exhibiting at least 6 dB gain margins and 60 degrees phase margins. The gain and phase margins are to account for uncertainty in the model. In addition, the

control surface displacements, due to the feedback command, should be less than 1 degree in order to stay within a somewhat linear range of the control surface displacement. In each of the LQG designs, the weighting on the output and input during the regulator design and the intensities of the noise matrices during the Kalman filter design were varied by trial and error until a design that met the goal was determined.

All of the CAP-TSD simulation results were calculated using the solution technique described in Chapter 3.1.3 with a ΔT of .000173 seconds and 8000 time steps.

4.4.1 Case 1

Case 1 involves developing a system ID model from the CAP-TSD (in a linear mode) outputs, designing an LQG control law, and evaluating the LQG control law using CAP-TSD in the linear mode. The results of the system ID process were previously described in section 4.3.3.

Figure 4.15 shows a Bode plot of the open-loop system (system ID model and LQG control law) that indicates a gain margin of 8.52 dB and a phase margin of 76.28 degrees. Figure 4.16 shows a comparison of CAP-TSD linear and the system ID linear model outputs for the uncontrolled and controlled (using the LQG control law) case. The results using the system ID model and the CAP-TSD simulation for the controlled case are almost identical. These results were one of the many used to verify the implementation of feedback control in CAP-TSD. The controlled case indicates a much higher degree of damping than the uncontrolled case. The values of the modal dampings and frequencies, as determined using the system ID process, are given in Table 4.2. Figure 4.16 also shows the feedback control surface command for Case 1. The maximum control surface displacement is approximately 0.3 degrees and occurs during the exponential pulse excitation.

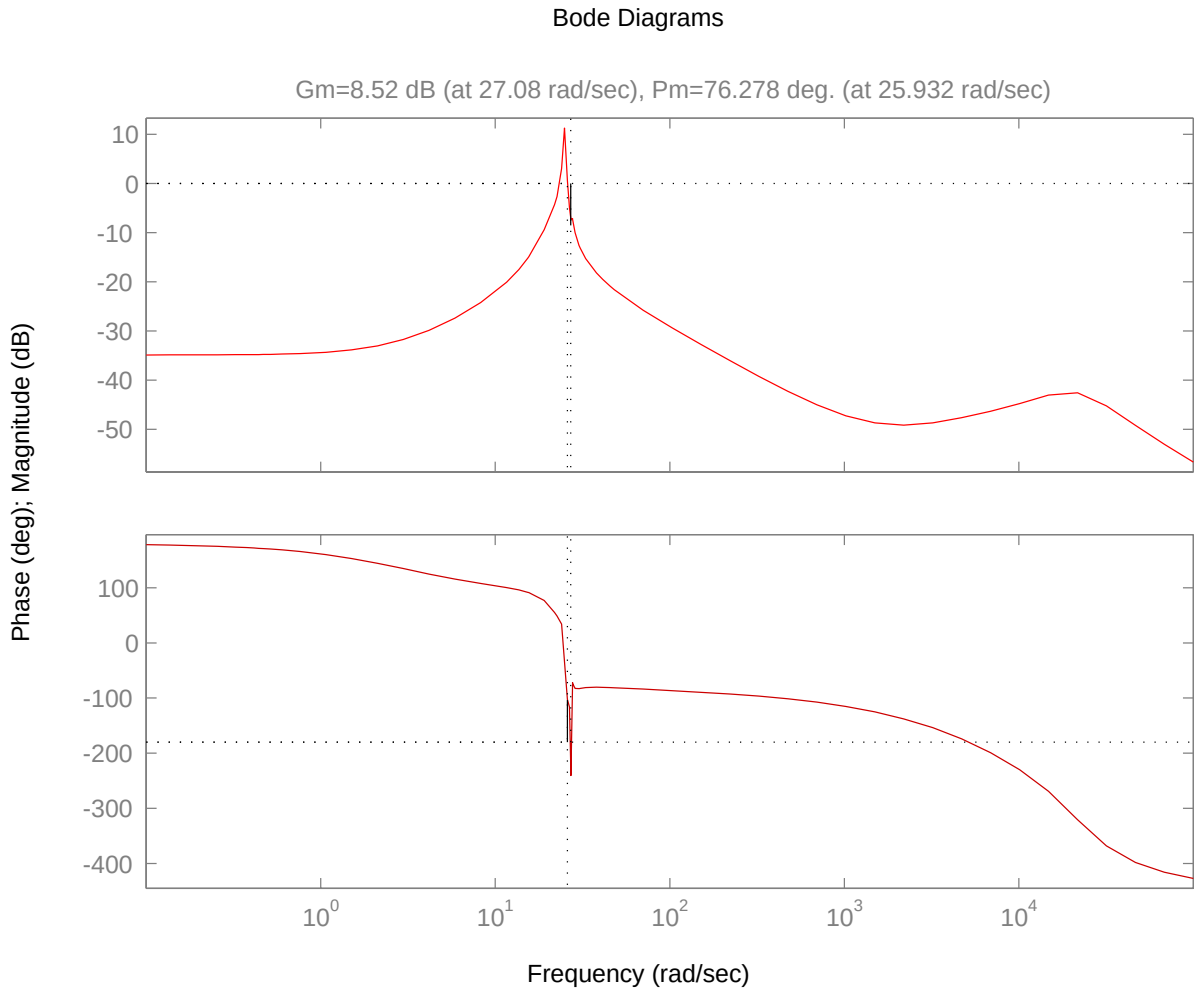

Figure 4.15 Case 1 Open-loop system Bode diagram

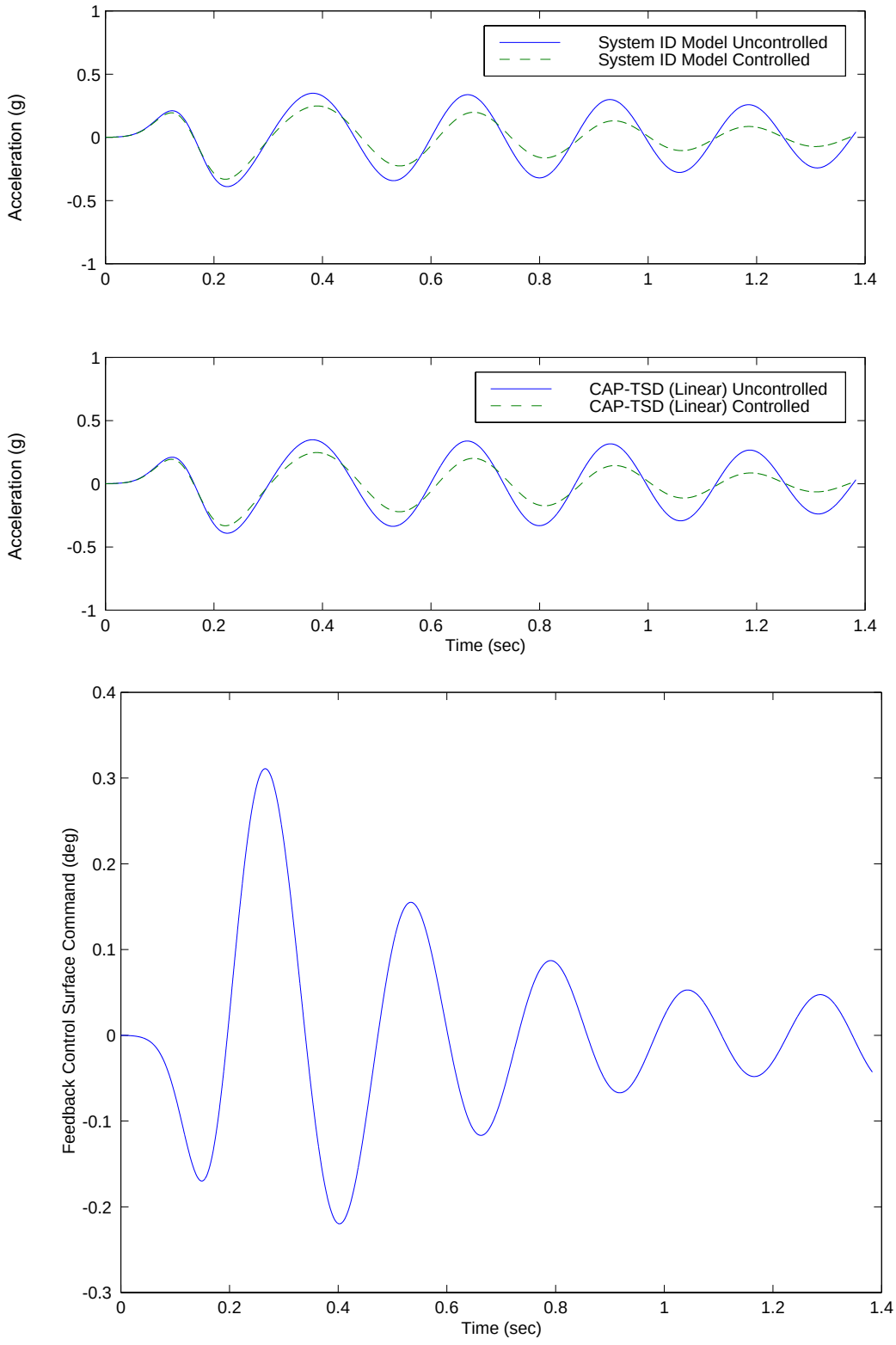

Figure 4.16 Case 1 controlled results

Table 4.2. Dampings and frequencies (less than 5 Hz) for all cases

Case	Uncontrolled Or Controlled	ζ_1 (%)	f_1 (Hz)	ζ_2 (%)	f_2 (Hz)	ζ_3 (%)	f_3 (Hz)
1	Uncontrolled	1.32	3.97	17.77	3.59	100	1.52
1	Controlled	4.40	4.04	23.08	3.57	100	.04
2	Uncontrolled	-1.53	3.82	35.42	3.77	100	3.72
2	Controlled	.999	3.84	44.38	4.07	100	2.26
3	Uncontrolled	-1.53	3.82	35.42	3.77	100	3.72
3	Controlled	4.42	33.82	34.63	4.03	100	2.26
4	Uncontrolled	-1.99	3.78	35.82	4.38	84.58	1.32
4	Controlled	.01	3.81	45.88	4.55	100	1.16
5	Uncontrolled	-1.99	3.78	35.82	4.38	84.58	1.32
5	Controlled	2.72	3.83	42.65	4.24	100	1.51
6	Uncontrolled	-1.85	3.77	23.08	4.57	100	.96
6	Controlled	-.05	3.79	25.47	4.81	89.51	1.11
7	Uncontrolled	-1.85	3.77	23.08	4.57	100	.96
7	Controlled	1.64	3.81	25.52	4.66	91.20	1.27
8	Uncontrolled	.09	4.25	6.53	3.41	100	2.21
8	Controlled	1.96	4.16	17.31	3.37	100	.10
9	Uncontrolled	-.54	4.16	8.25	3.41	-	-
9	Controlled	.66	4.16	8.50	3.41	-	-
10	Uncontrolled	-1.09	4.07	11.13	3.42	-	-
10	Controlled	.81	4.06	11.80	3.41	-	-

4.4.2 Case 2

Case 2 involves evaluating the control law designed in Case 1 using the CAP-TSD nonlinear simulation at an angle of attack of 0 degrees. In order to calculate stability margins and design a control law using the results of the CAP-TSD nonlinear simulation (Case 3), the CAP-TSD (nonlinear) outputs are used as the input to the system ID procedure. Figure 4.17 shows the results from the system ID procedure in terms of a comparison between the CAP-TSD (nonlinear) outputs and the system ID model. Similar to the linear case, the system ID model provides a very good representation of the CAP-TSD outputs. The modal dampings and frequencies for this system ID model are given in Table 4.2.

Figure 4.18 shows a Bode plot of the open-loop system that indicates a gain margin of –4.75 dB and a phase margin of 31.34 degrees. Although stable (see Table 4.2), the control law designed from the linear model does not provide satisfactory stability margins. Figure 4.19 shows a comparison of CAP-TSD and the system ID linear model outputs for the uncontrolled and controlled (using Case 1 control law) case. The modal dampings and frequencies are given in Table 4.2. The controlled response using the CAP-TSD nonlinear model is not as highly damped as the linear model results. Figure 4.19 also shows the feedback control surface command for Case 2. Similar to the Case 1, the maximum control surface displacement is approximately 0.3 degrees and occurs during the exponential pulse excitation.

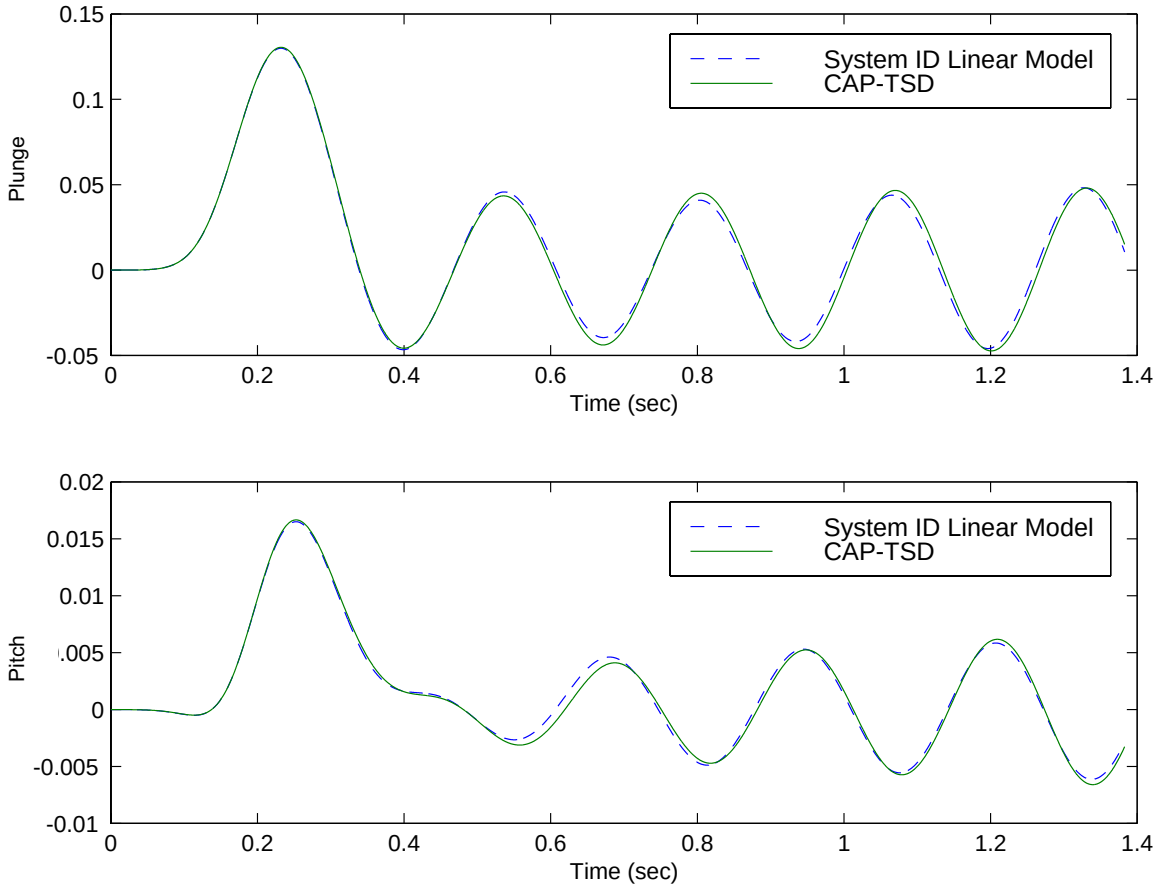

Figure 4.17(a). Case 2 system ID results for displacements

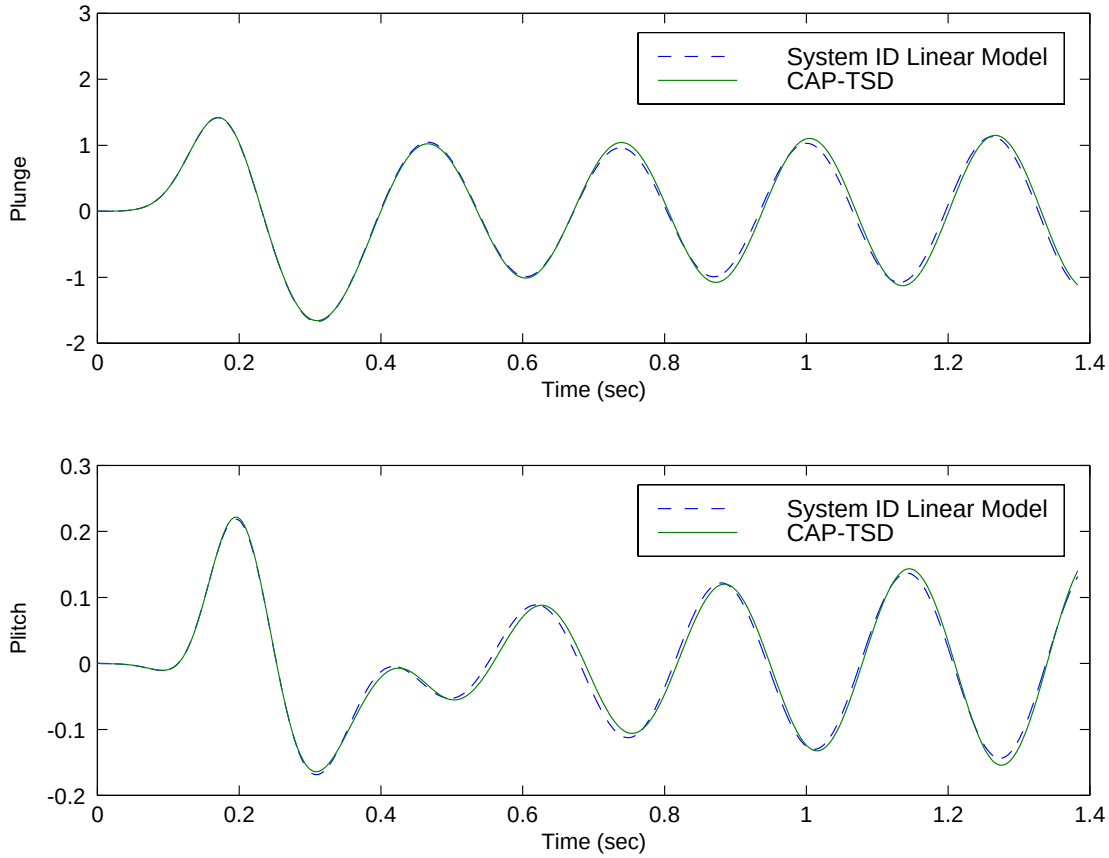

Figure 4.17 (b) Case 2 system ID results for velocities

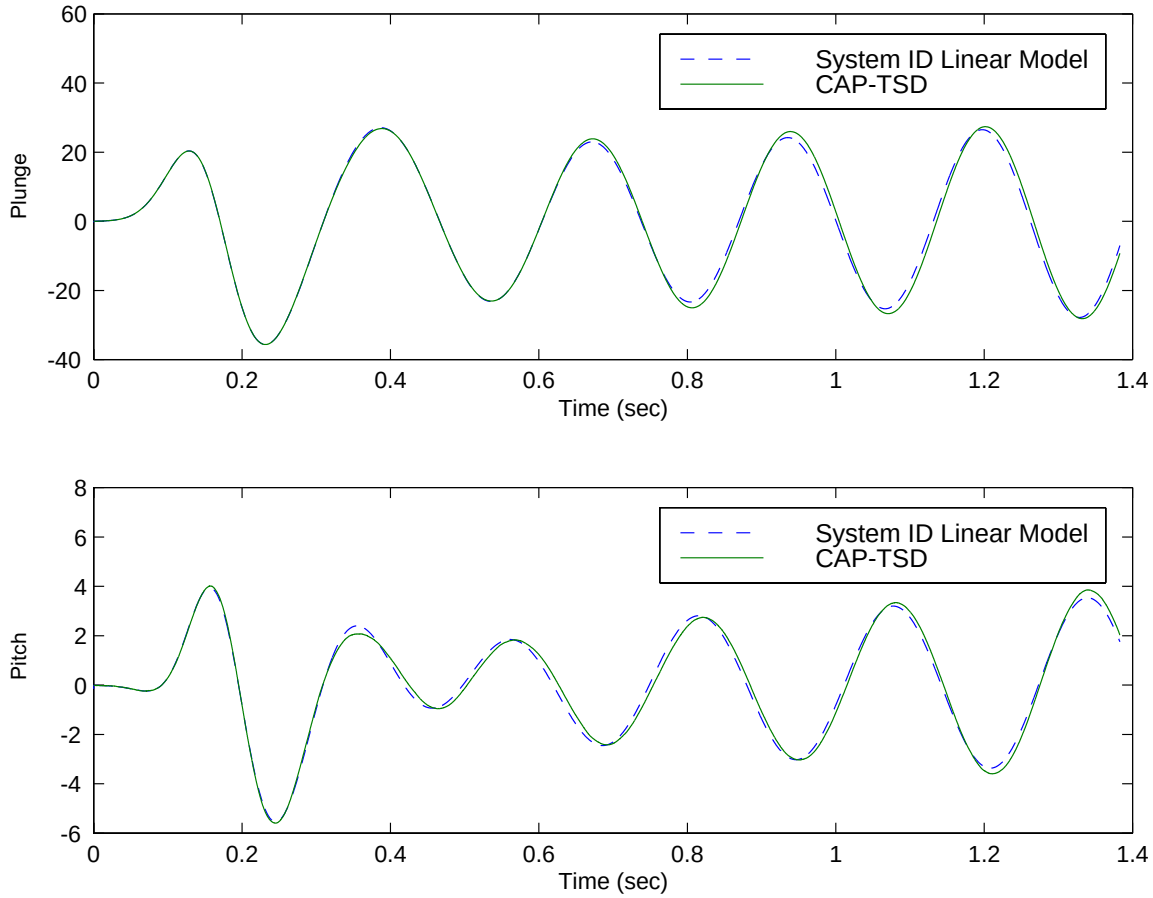

Figure 4.17 (c) Case 2 system ID results for accelerations

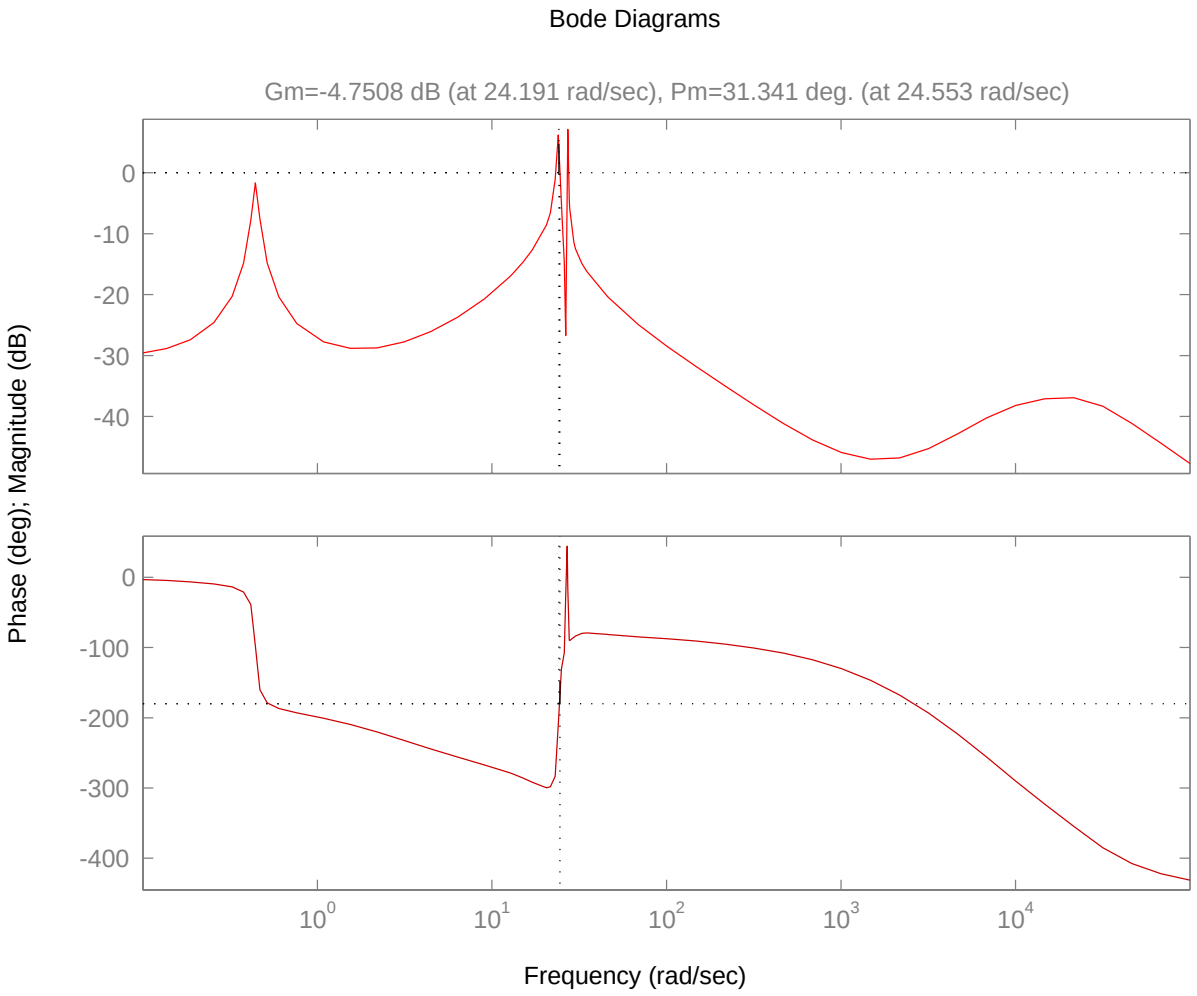

Figure 4.18 Case 2 Open-loop system Bode diagram

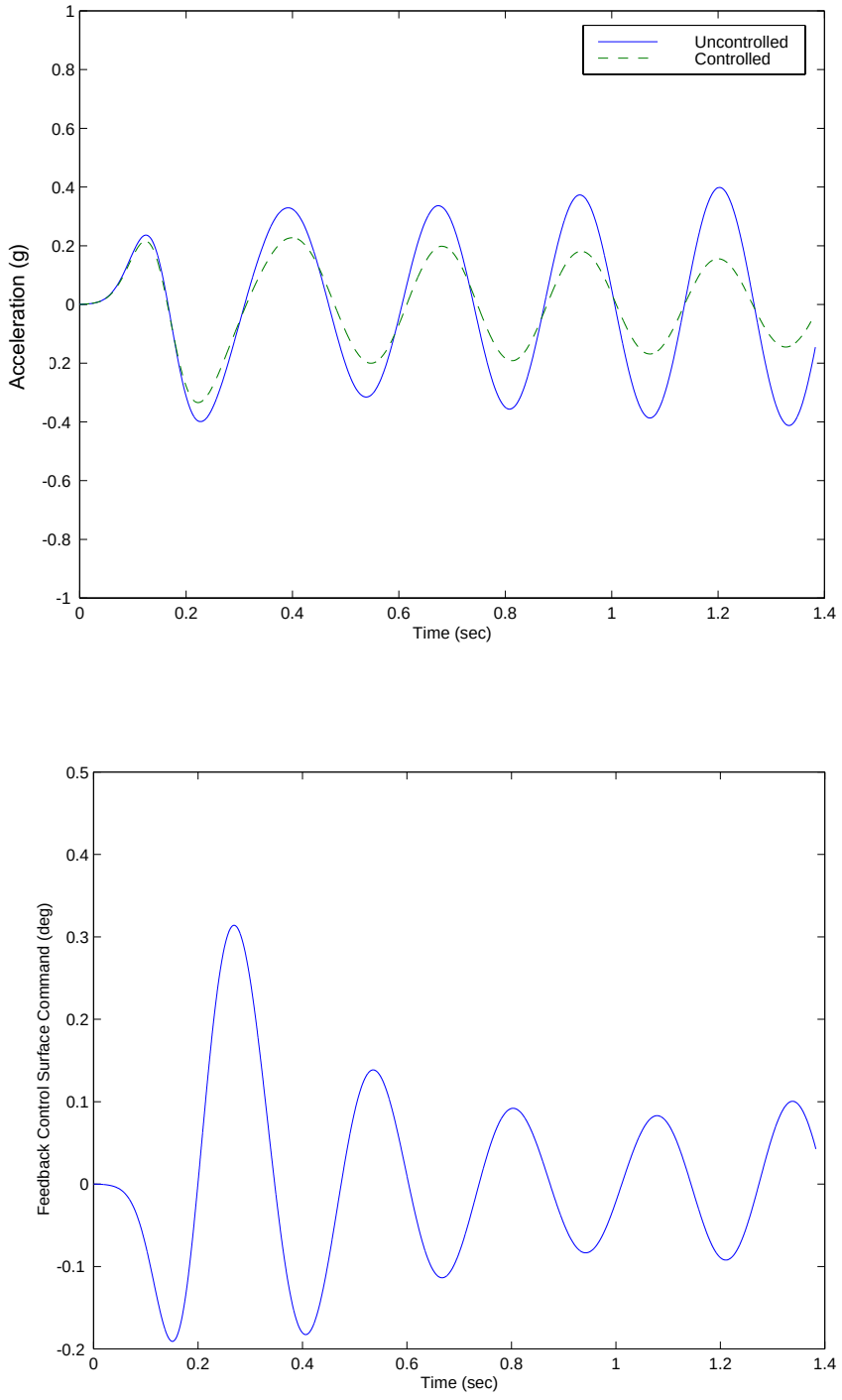

Figure 4.19 Case 2 controlled results

4.4.3 Case 3

Case 3 employs the system ID model derived from the nonlinear CAP-TSD results of Case 2 to design a control law. After some trial and error during the LGQ design process, a satisfactory control law was determined. Figure 4.20 shows a Bode plot of the open-loop system that indicates a gain margin of –10.84 dB and a phase margin of 64.72 degrees.

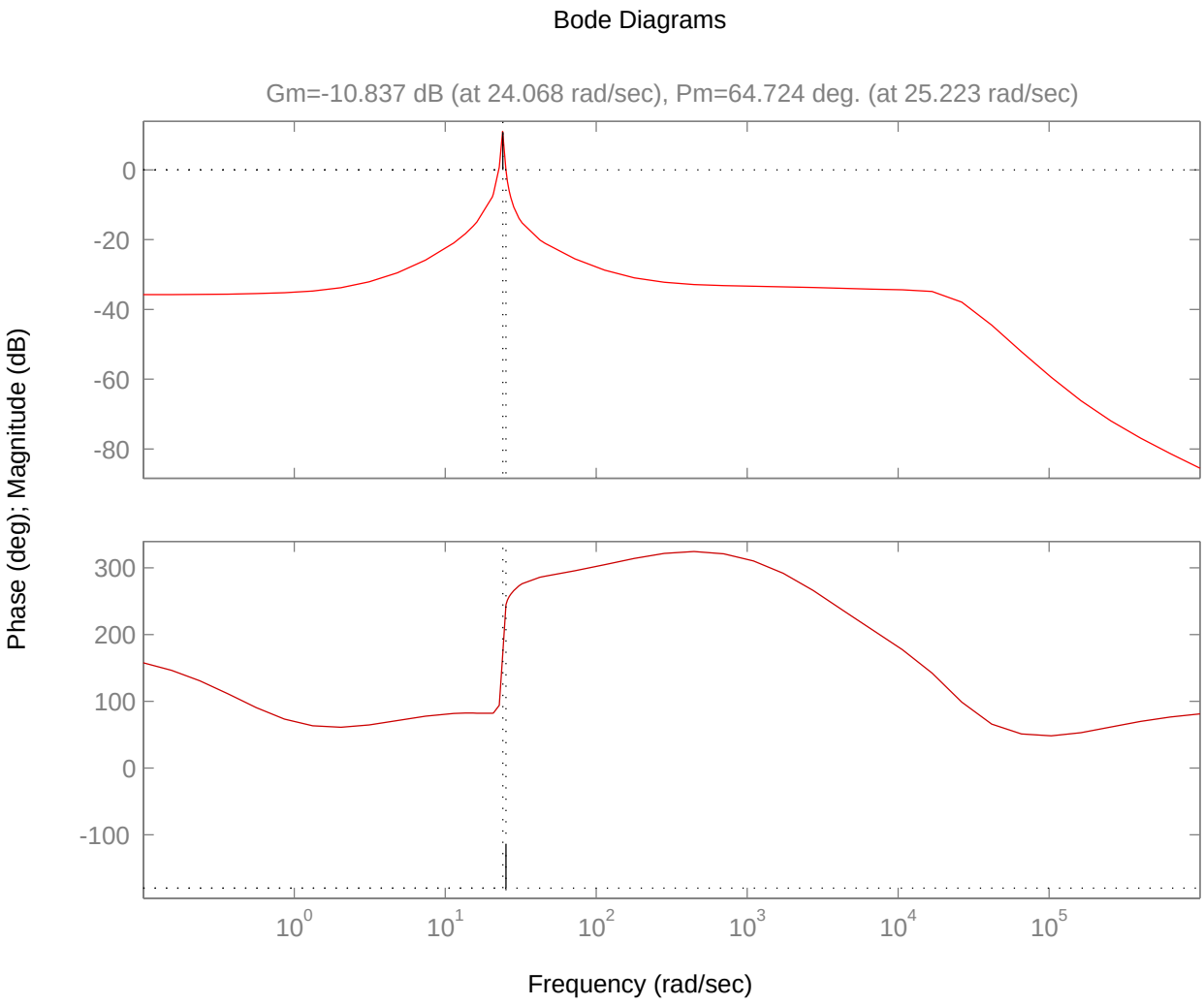

Figure 4.20 Case 3 Open-loop system Bode diagram

Figure 4.21 shows a comparison of CAP-TSD outputs for the uncontrolled and controlled case. The modal dampings and frequencies are given in Table 4.2. Figure 4.21 also shows the feedback control surface command for Case 3.

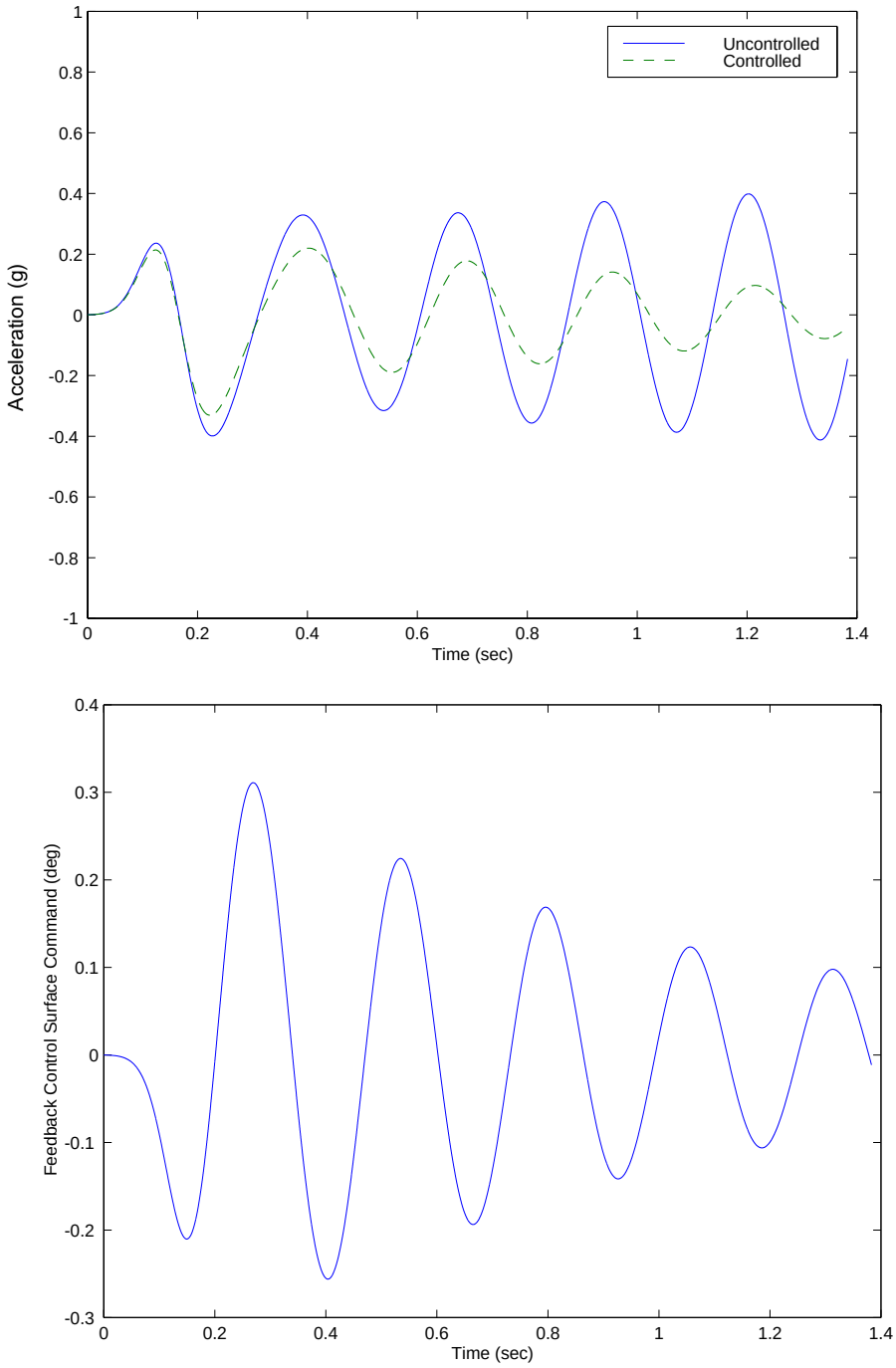

Figure 4.21 Case 3 controlled results

Similar to the Cases 1 and 2, the maximum control surface displacement is approximately 0.3 degrees and occurs during the exponential pulse excitation. The results using this control law clearly show much better results, in particular stability margins, than using the control law designed using the system ID model of the linear CAP-TSD outputs.

4.4.4 Case 4

Case 4 involves evaluating the control law designed in Case 1 using the CAP-TSD nonlinear simulation at an angle of attack of 0.3 degrees. In order to evaluate the stability margins and design a control law for Case 5, the system ID procedure is applied to the outputs of CAP-TSD at 0.3 degrees angle of attack. Obtaining a good system ID model for this case was more of a challenge than the 0 degree angle of attack cases. First, since the plunge and pitch displacement static aeroleastic values are not zero, the plunge and pitch displacement outputs appear to have a bias from the system ID point of view. This bias tends to manifest itself as a pseudo rigid-body-like effect during the system ID procedure. Therefore, the bias is removed from the plunge and pitch displacements by subtracting the value at time=0 from all of the values of plunge and pitch displacements. Secondly, increasing the order of the model greater than 5 results in an unstable observer and nonconvergence. However, for this case, the 5^{th} order system provides a much better system ID model than a 5^{th} order system did for the 0 angle of attack cases. Figure 4.22 shows the results from the system ID procedure (after the bias is removed from the plunge and pitch displacements) in terms of a comparison between the CAP-TSD outputs and the system ID model (5^{th} order model). Similar to Cases 1 and 2, the system ID model provides a very good representation of the CAP-TSD results.

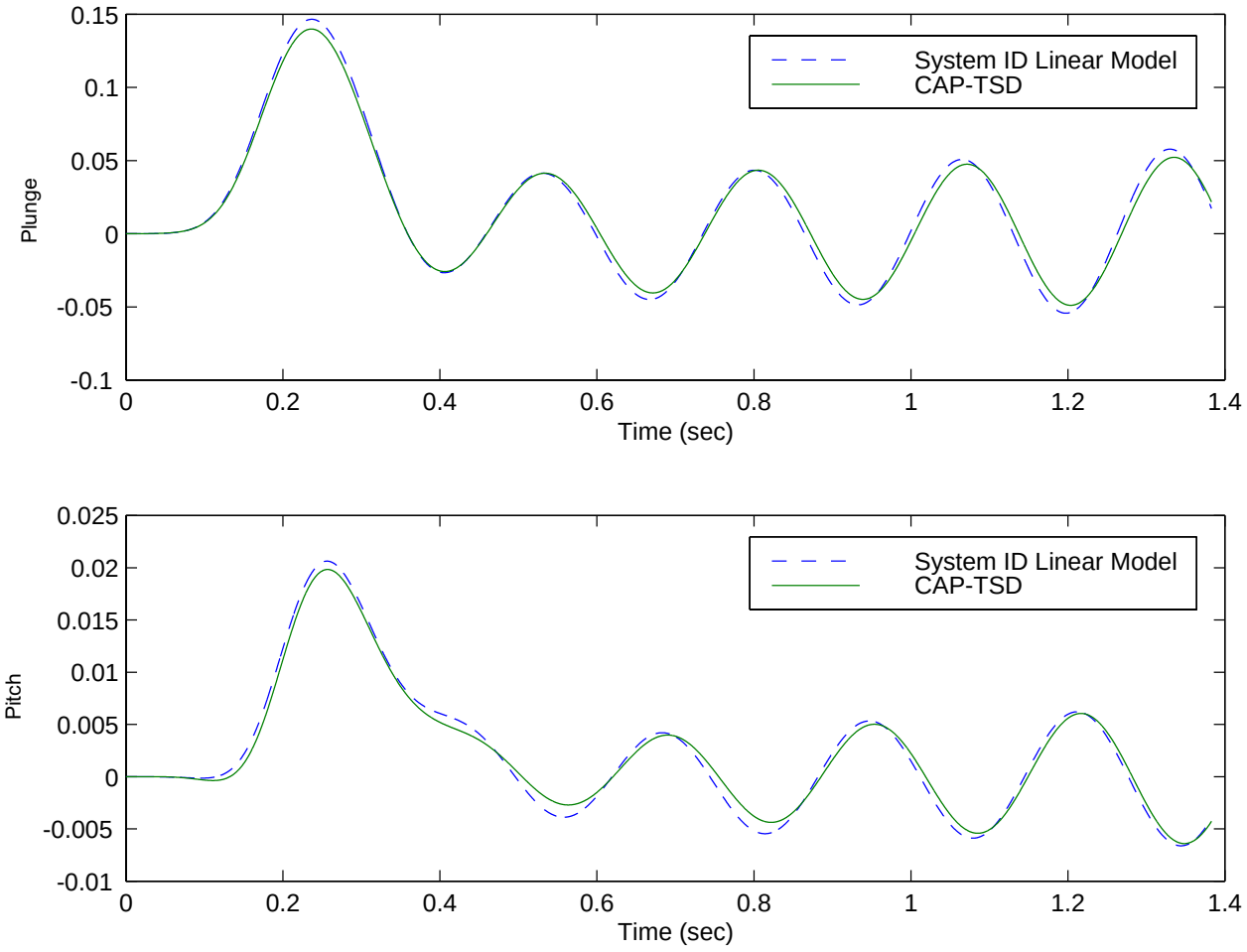

Figure 4.22 (a) Case 4 system ID results for displacements

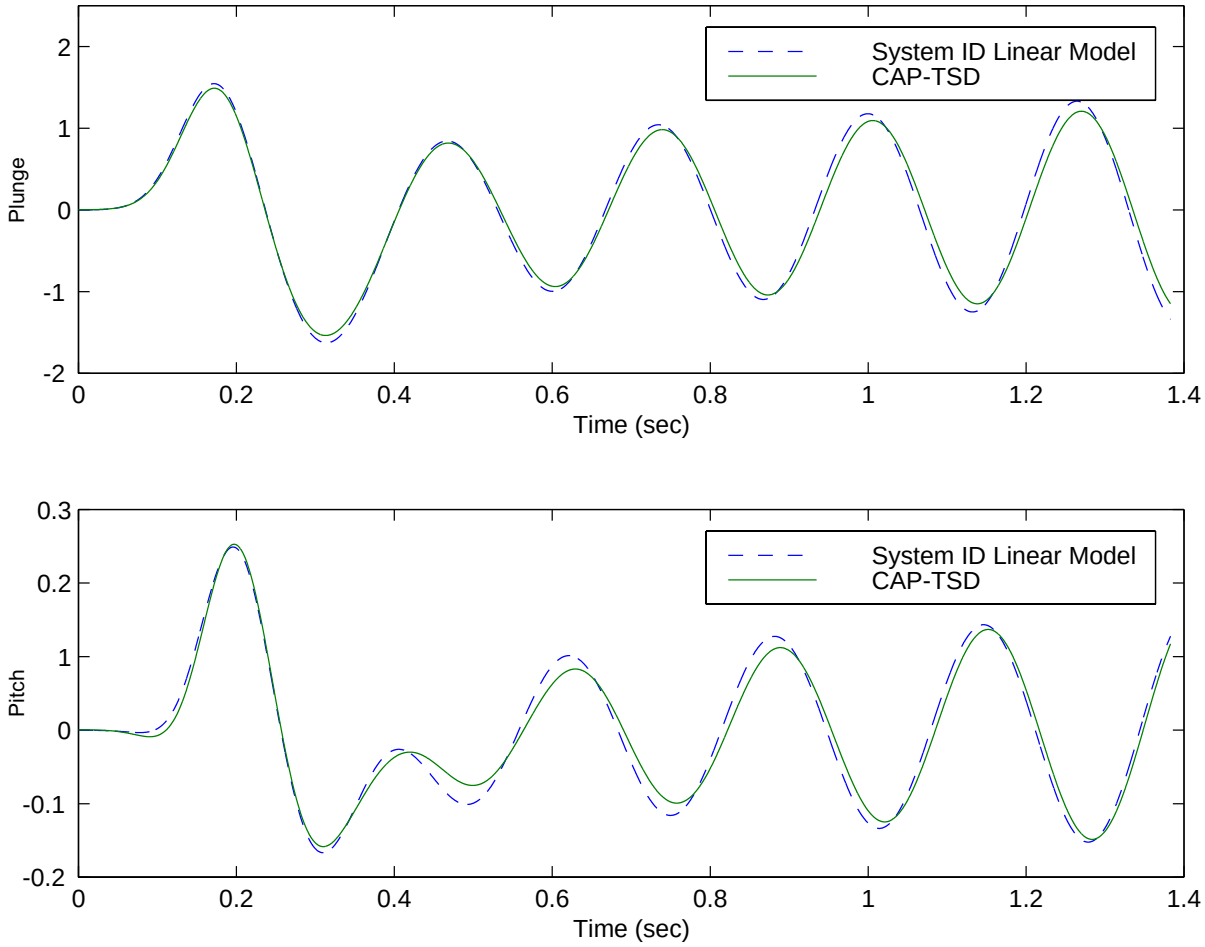

Figure 4.22 (b) Case 4 system ID results for velocities

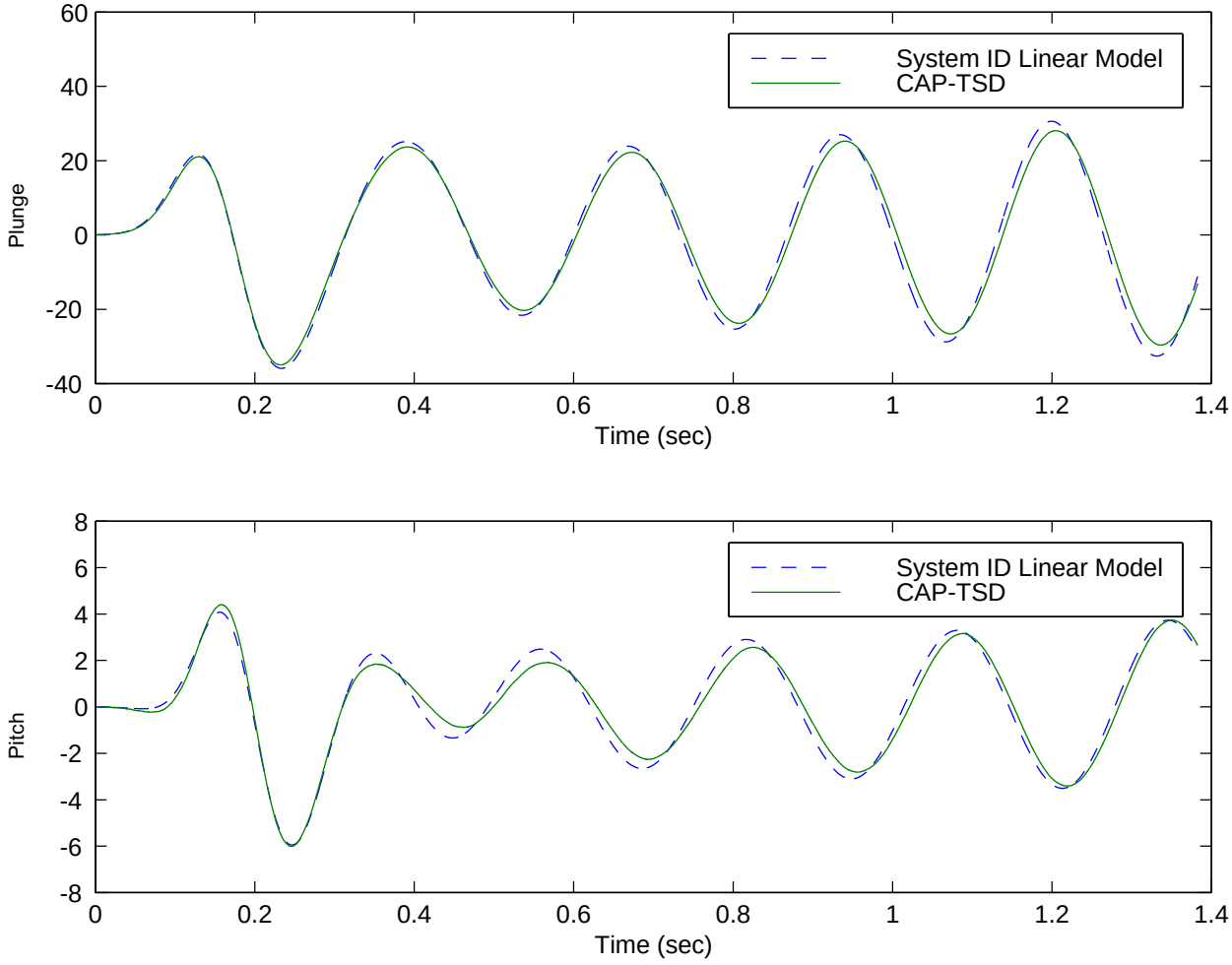

Figure 4.22 (c) Case 4 system ID results for accelerations

Figure 4.23 shows a Bode plot of the open-loop system that indicates a gain margin of –3.19 dB and a phase margin of 35.36 degrees. Similar to Case 2, the control law designed from the linear model does not provide satisfactory stability margins. Figure 4.24 shows a comparison of CAP-TSD outputs for the uncontrolled and controlled (using Case 1 control law) case. The controlled system is close to neutrally stable. The modal dampings and frequencies are given in Table 4.2. Figure 4.24 also shows the feedback

control surface command for Case 4. Similar to the previous cases, the maximum control surface displacement is approximately 0.3 degrees and occurs during the exponential pulse excitation.

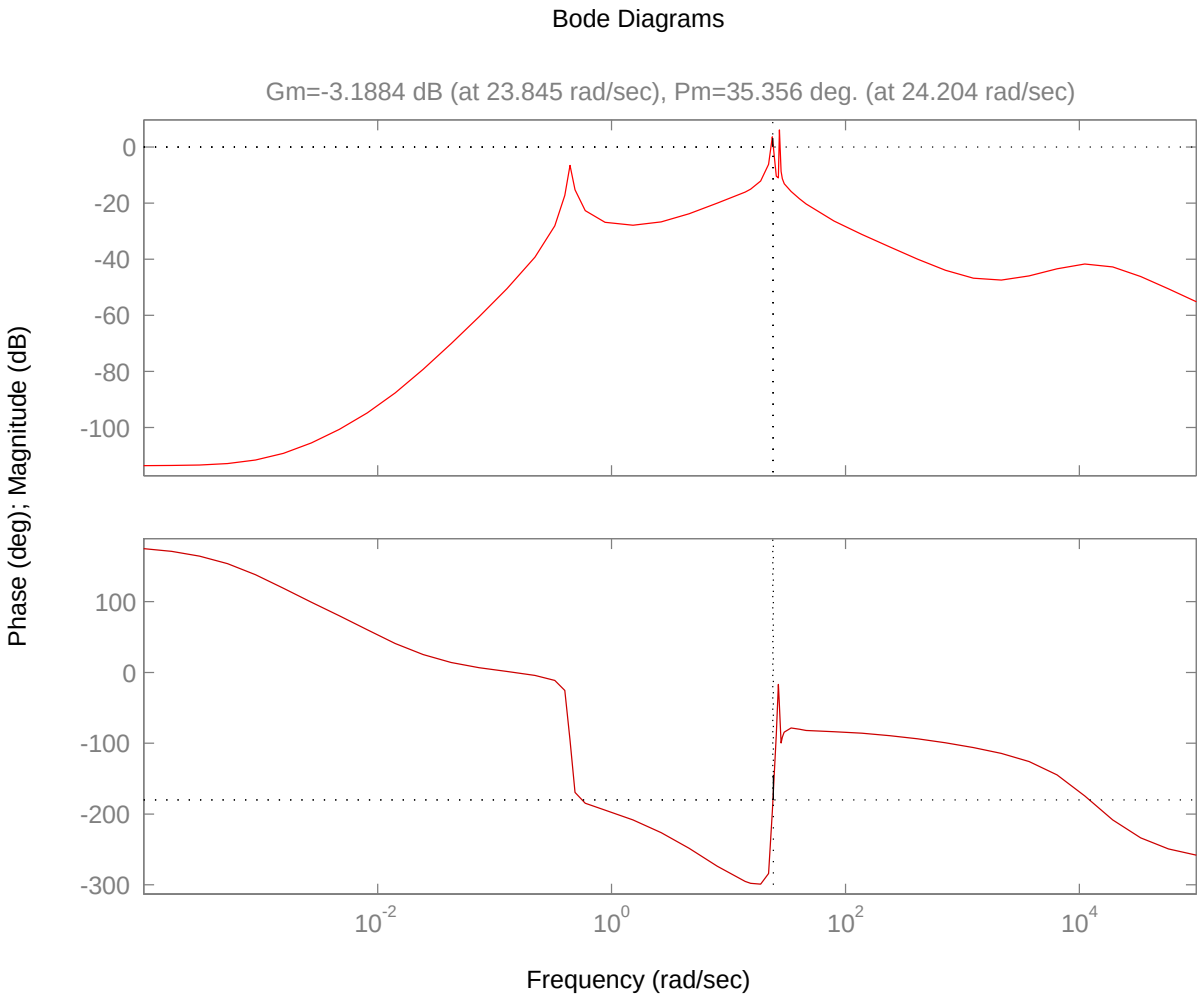

Figure 4.23 Case 4 Open-loop system Bode diagram

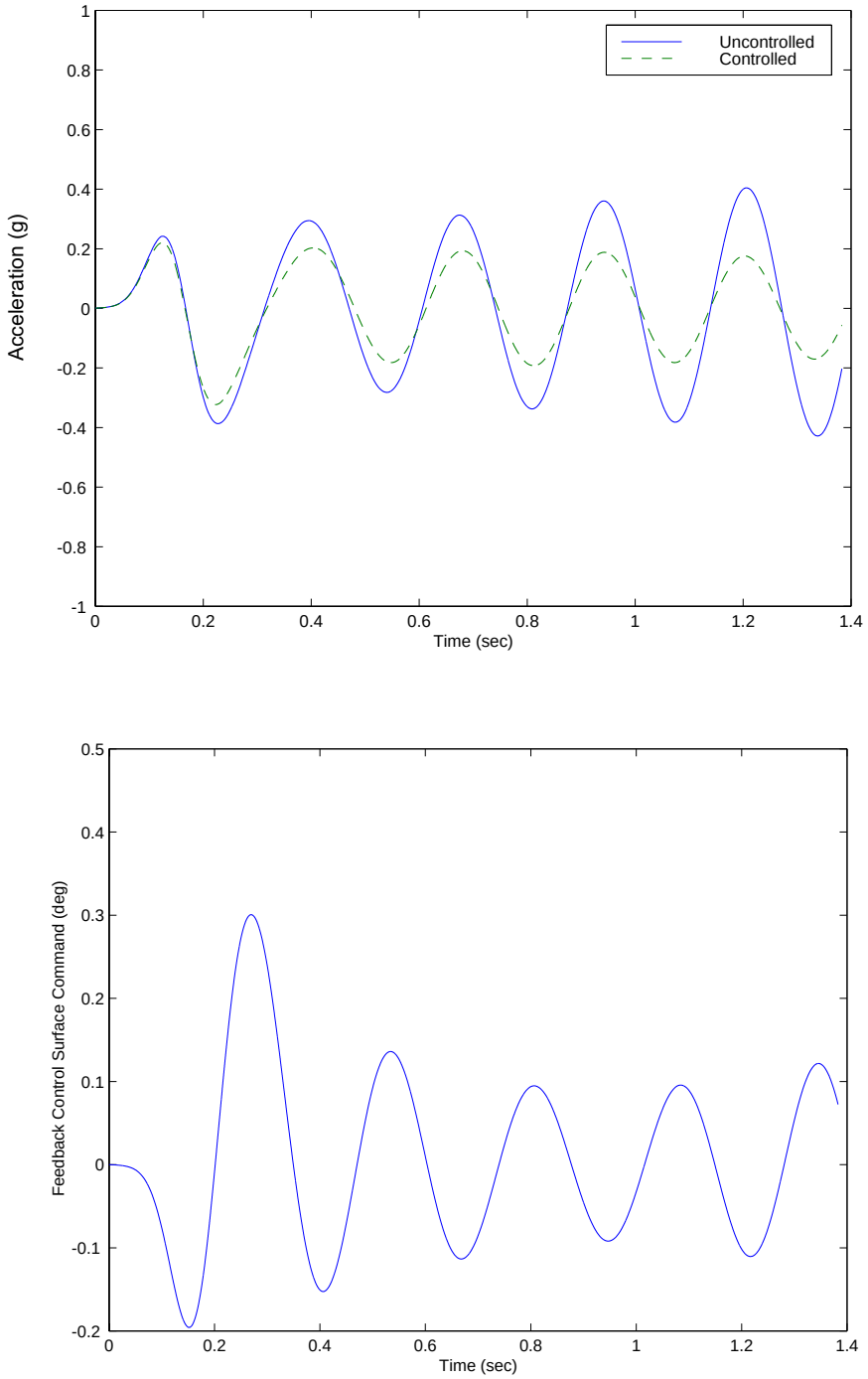

Figure 4.24 Case 4 controlled results

4.4.5 Case 5

Case 5 employs the system ID model derived from the nonlinear CAP-TSD outputs at 0.3 degrees angle of attack to design a control law. Figure 4.25 shows a Bode plot of the open-loop system that indicates a gain margin of –9.62 dB and a phase margin of 63.74 degrees.

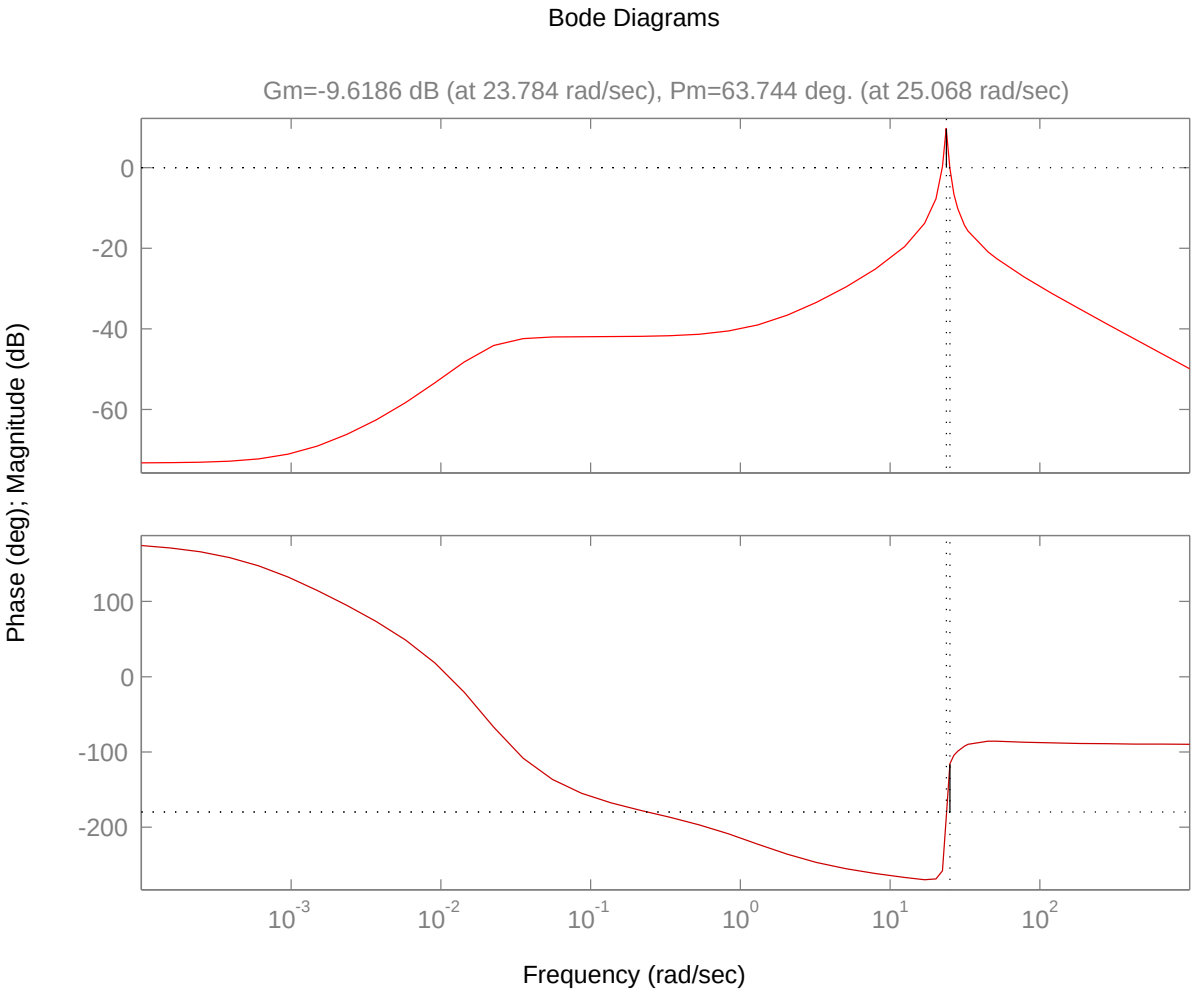

Figure 4.25 Case 5 Open-loop system Bode diagram

Figure 4.26 shows a comparison of CAP-TSD outputs for the uncontrolled and controlled case. There is a significant reduction in the acceleration response with the controlled

case. The modal dampings and frequencies are given in Table 4.2. Figure 4.26 also shows the feedback control surface command for Case 5.

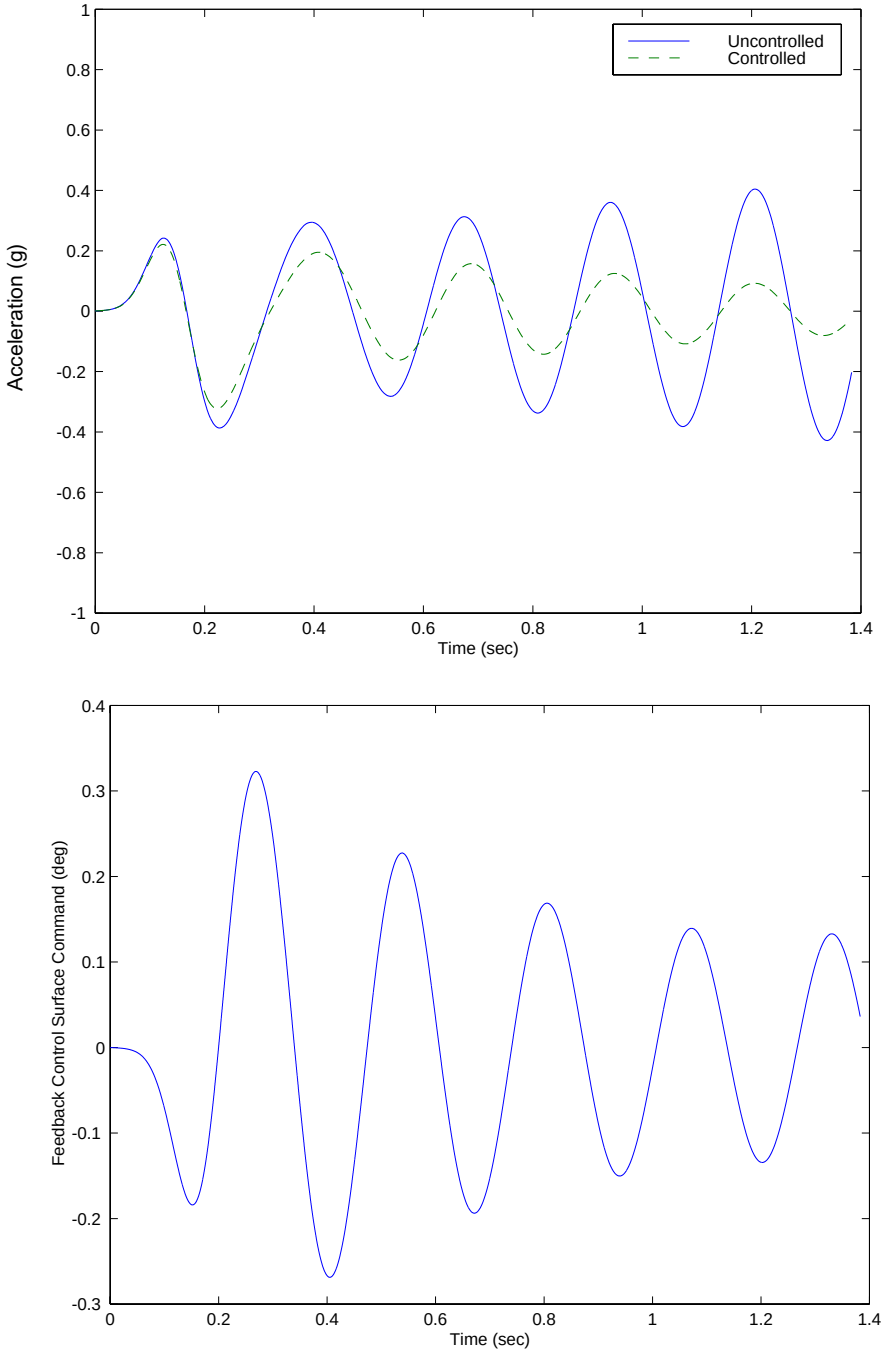

Figure 4.26 Case 5 controlled results

Similar to the previous cases, the maximum control surface displacement is approximately 0.3 degrees and occurs during the exponential pulse excitation. The results using this control law clearly show much better results, in particular stability margins and damping, than using the control law designed using the system ID model of the linear CAP-TSD outputs.

4.4.6 Case 6

Case 6 involves evaluating the control law designed in Case 1 using the CAP-TSD nonlinear simulation at an angle of attack of 0.6 degrees. In order to evaluate the stability margins and design a control law for Case 7, the system ID procedure is applied to the outputs of CAP-TSD at 0.6 degrees angle of attack. Obtaining a good system ID model for this case was even more of a challenge than the 0.3 degrees angle of attack case. Both the bias and increasing order problems were still present in this case. However, an additional problem involving the value of the sampling time was present. In order to obtain a good system ID model, it was necessary to increase the sampling time by a factor of forty (decrease the sample rate by a factor of forty). When the bias was removed, the sample rate was decreased and the model order was 5, a good system ID model was obtained. Figure 4.27 shows the results from the system ID in terms of a comparison between the CAP-TSD outputs and the system ID model. Although the system ID model provides a good representation of the CAP-TSD results, it is not as good as the 0 and 0.3 degrees angle of attack cases.

For this case, the controlled response is unstable with the Case 1 control law (see Table 4.2). Figure 4.28 shows a comparison of CAP-TSD outputs for the uncontrolled and controlled (using Case 1 control law) case. The controlled system can be seen to be slightly unstable. Figure 4.28 also shows the feedback control surface command for Case 6. The unstable character is clearly seen in the feedback control surface command.

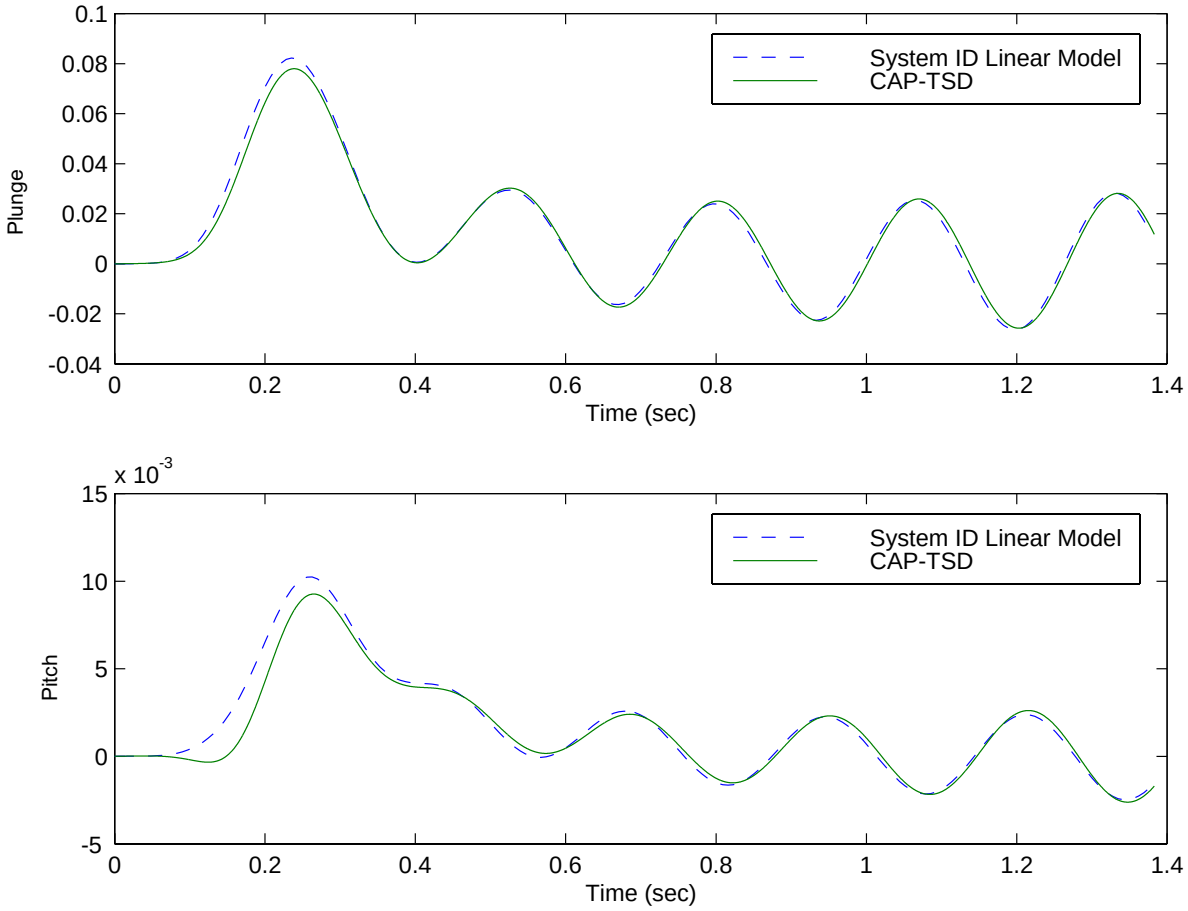

Figure 4.27 (a) Case 6 system ID results for displacements

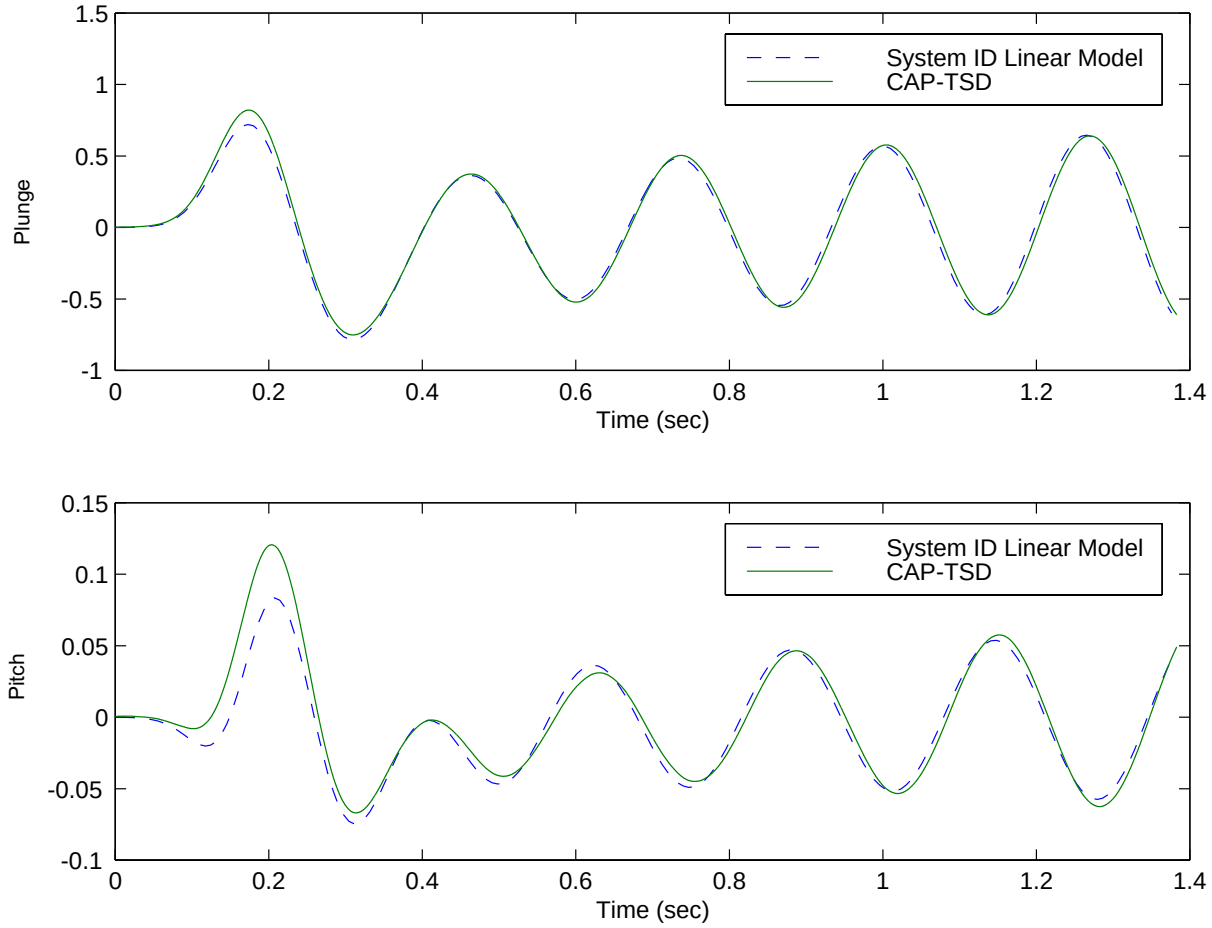

Figure 4.27 (b) Case 6 system ID results for velocities

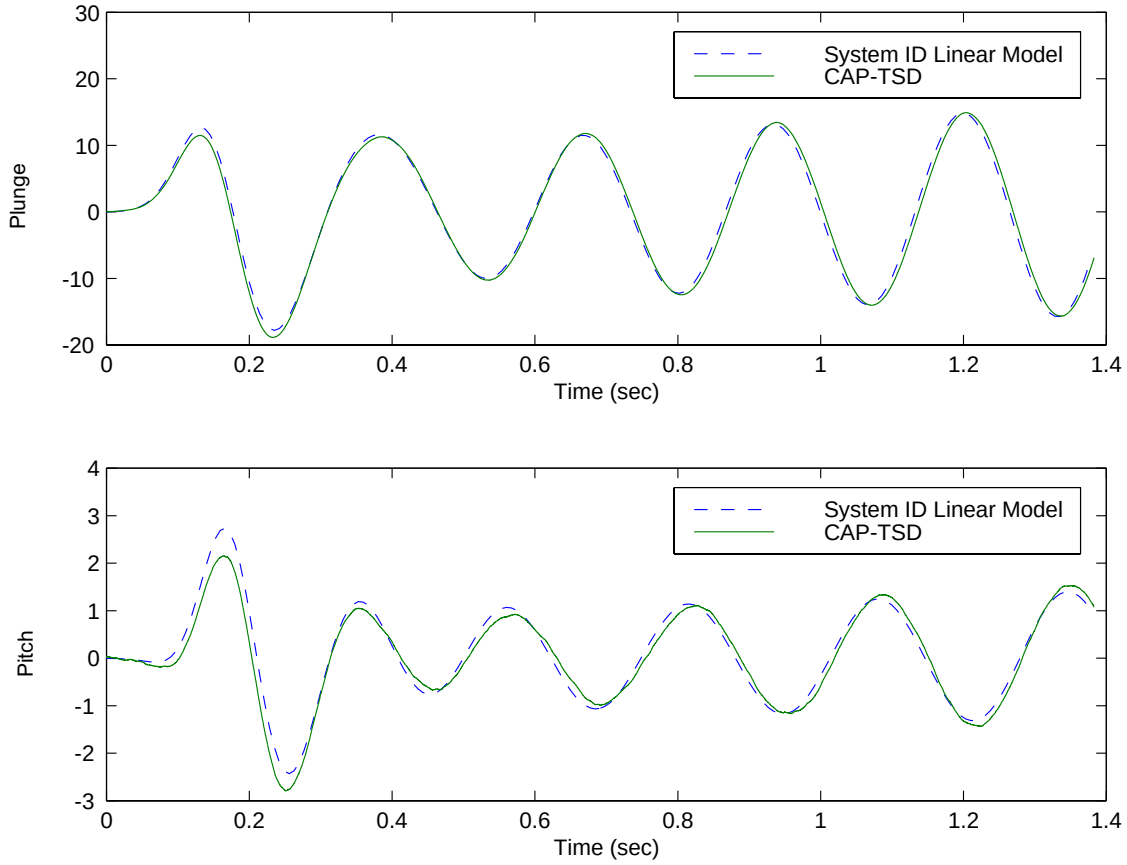

Figure 4.27 (c) Case 6 system ID results for accelerations

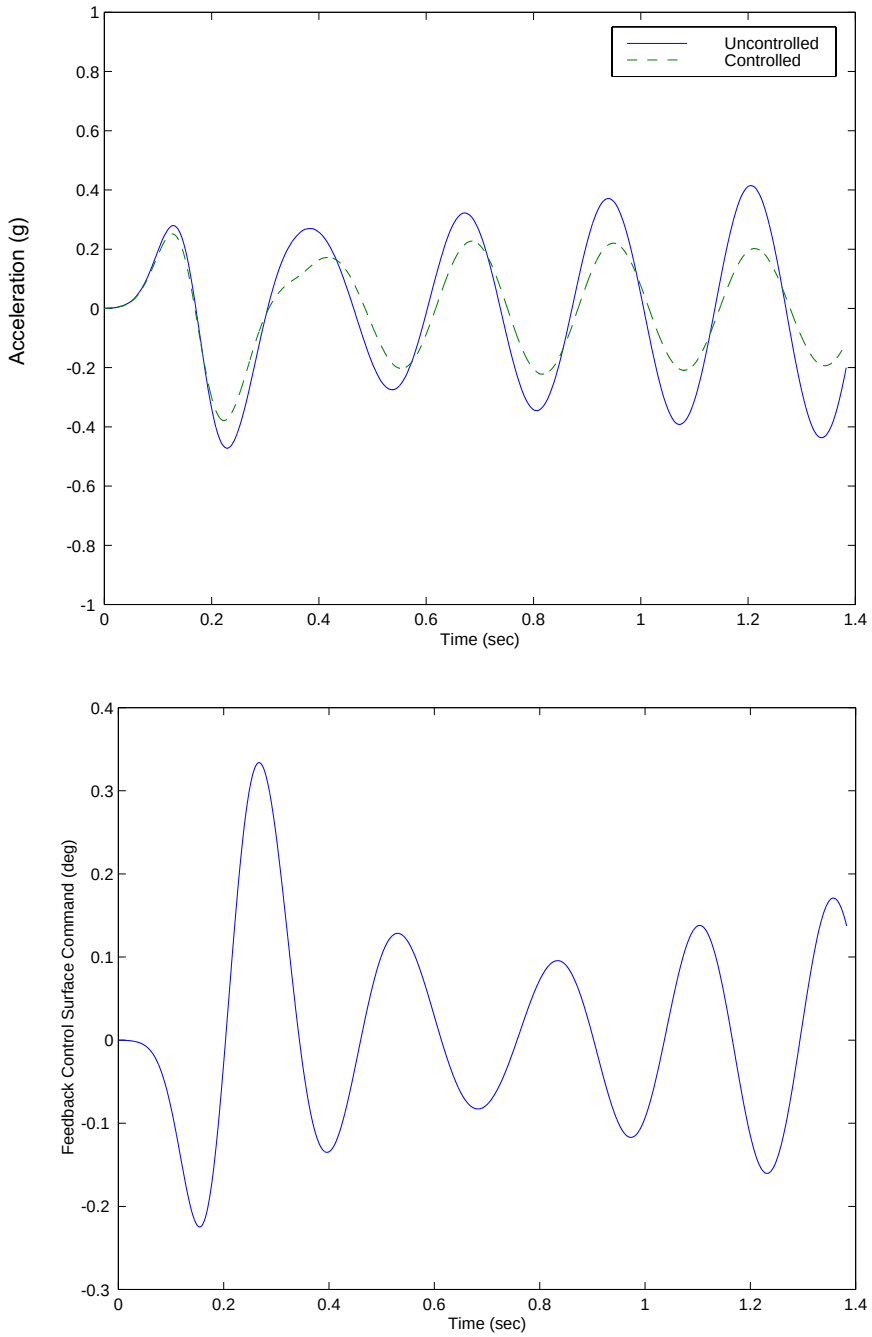

Figure 4.28 Case 6 controlled results

4.4.7 Case 7

Case 7 employs the system ID model derived from the nonlinear CAP-TSD outputs at 0.6 degrees angle of attack to design a control law. Figure 4.29 shows a Bode plot of the

open-loop system that indicates a gain margin of –9.66 dB and a phase margin of 64.17 degrees.

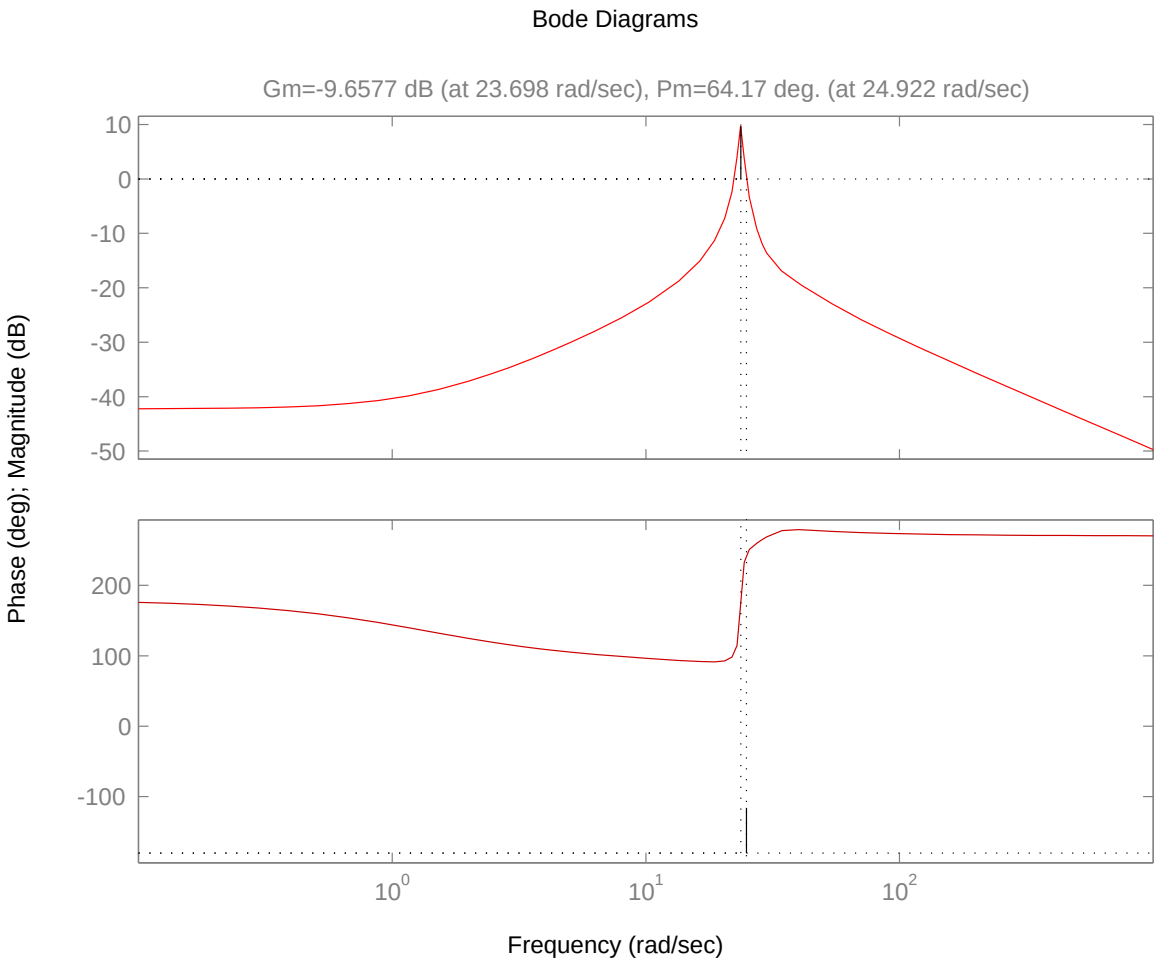

Figure 4.29 Case 7 Open-loop system Bode diagram

Figure 4.30 shows a comparison of CAP-TSD outputs for the uncontrolled and controlled case. There is a significant reduction in the acceleration response with the controlled case. The modal dampings and frequencies are given in Table 4.2. Figure 4.30 also shows the feedback control surface command for Case 7. Similar to the previous cases, the maximum control surface displacement is approximately 0.3 degrees and occurs

during the exponential pulse excitation. The results using this control law clearly show much better results, in particular stability margins and damping, than using the control law designed using the system ID model of the linear CAP-TSD outputs.

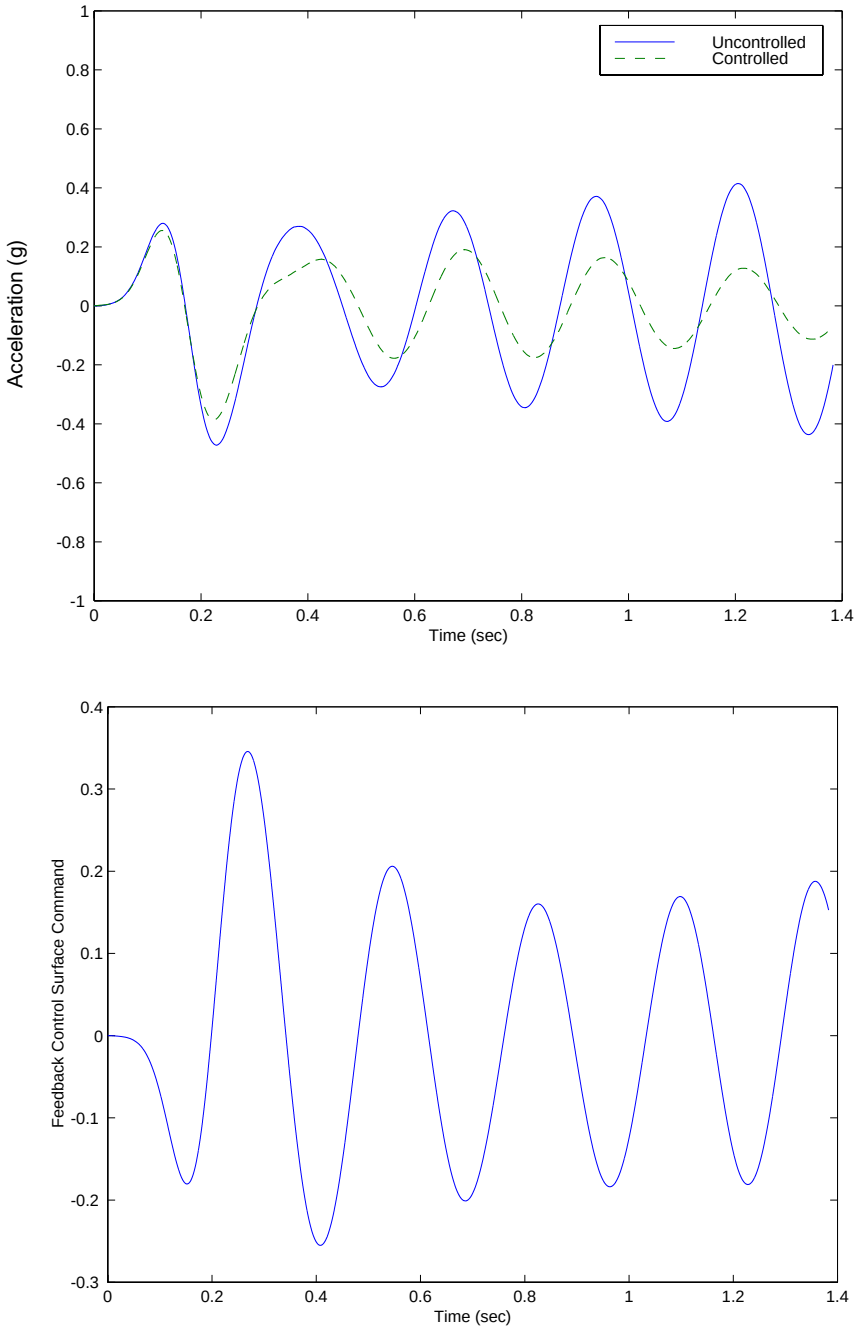

Figure 4.30 Case 7 controlled results

4.4.8 Case 8

Case 8 involves developing a system ID model from the CAP-TSD (in a linear mode) outputs using the 3-d aerodynamic model, designing an LQG control law, and evaluating the LQG control law using CAP-TSD (3-d) in the linear mode. Figure 4.31 shows the results from the system ID procedure in terms of a comparison between the CAP-TSD outputs and the system ID model. Similar to the linear 2-d case, the system ID model provides a very good representation of the CAP-TSD outputs. The modal dampings and frequencies for this system ID model are given in Table 4.2.

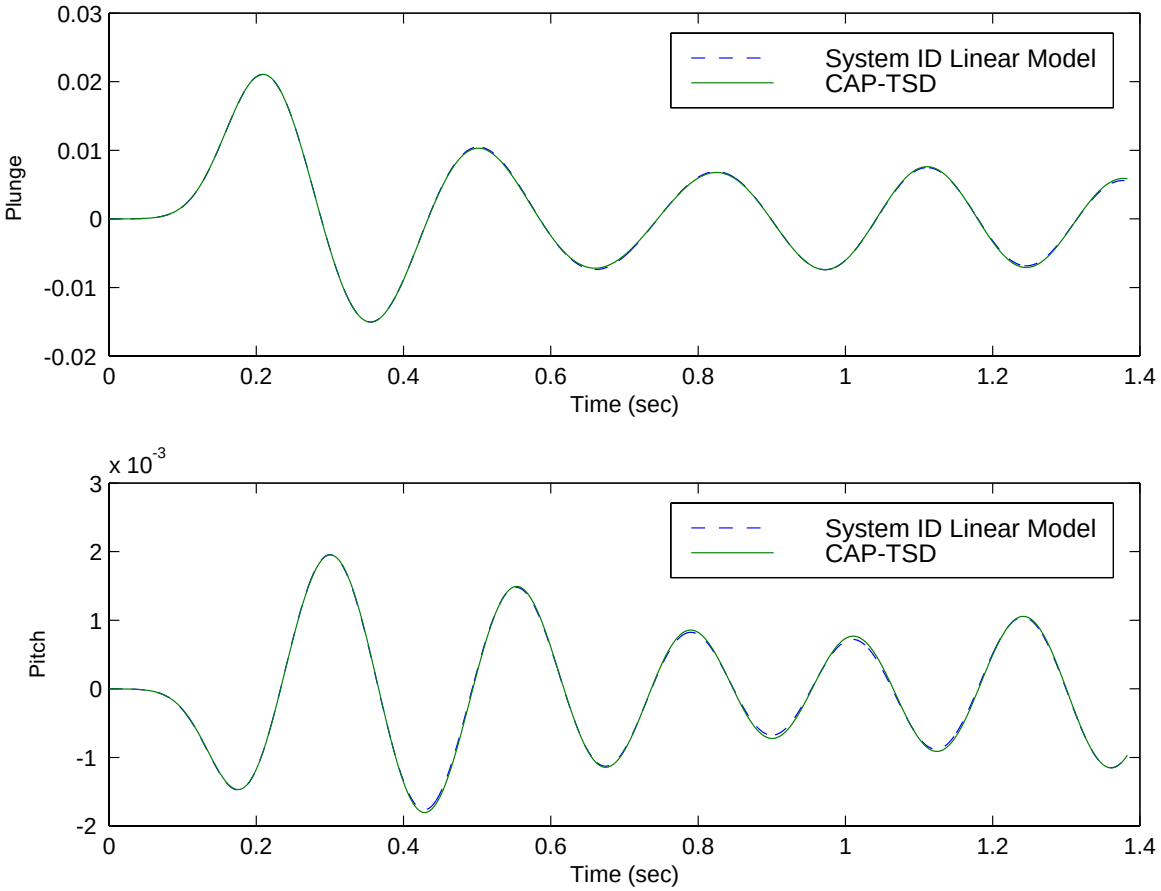

Figure 4.31 (a) Case 8 system ID results for displacements

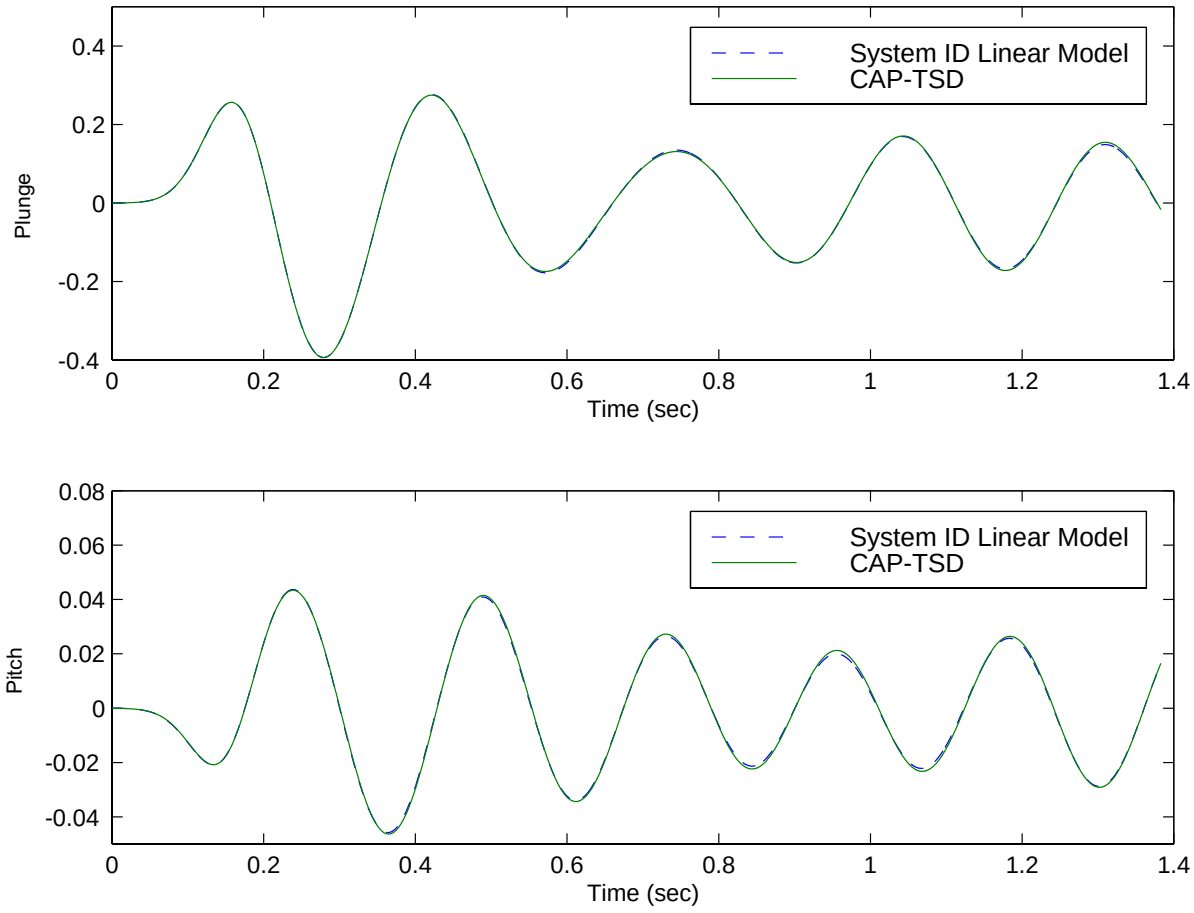

Figure 4.31 (b) Case 8 system ID results for velocities

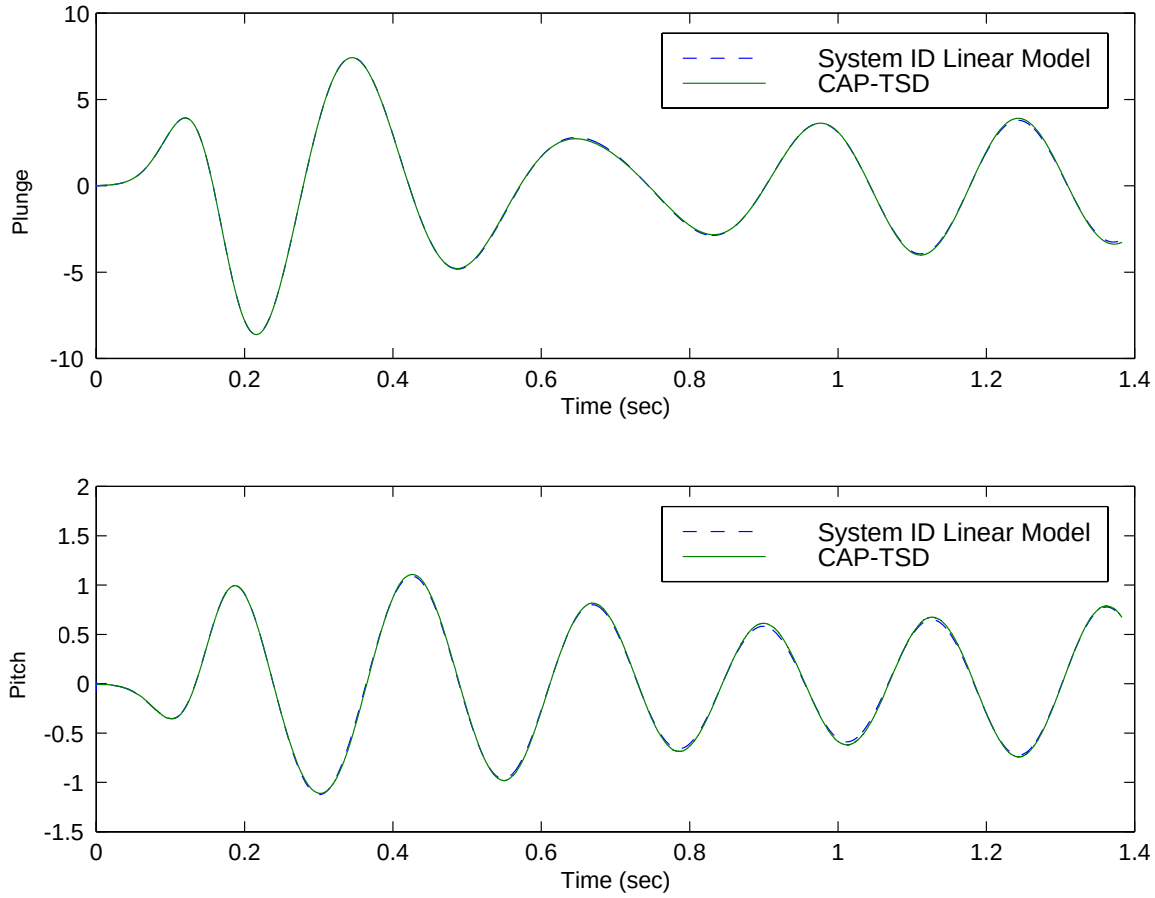

Figure 4.31 (c) Case 8 system ID results for accelerations

Figure 4.32 shows a Bode plot of the open-loop system that indicates a gain margin of 9.22 dB and a phase margin of –61.04 degrees. Figure 4.33 shows a comparison of CAP-TSD outputs for the uncontrolled and controlled case. The controlled case indicates a much higher degree of damping than the uncontrolled case. The values of the modal dampings and frequencies are given in Table 4.2. Figure 4.33 also shows the feedback control surface command for Case 8. The maximum control surface displacement is approximately 0.5 degrees and occurs during the exponential pulse excitation.

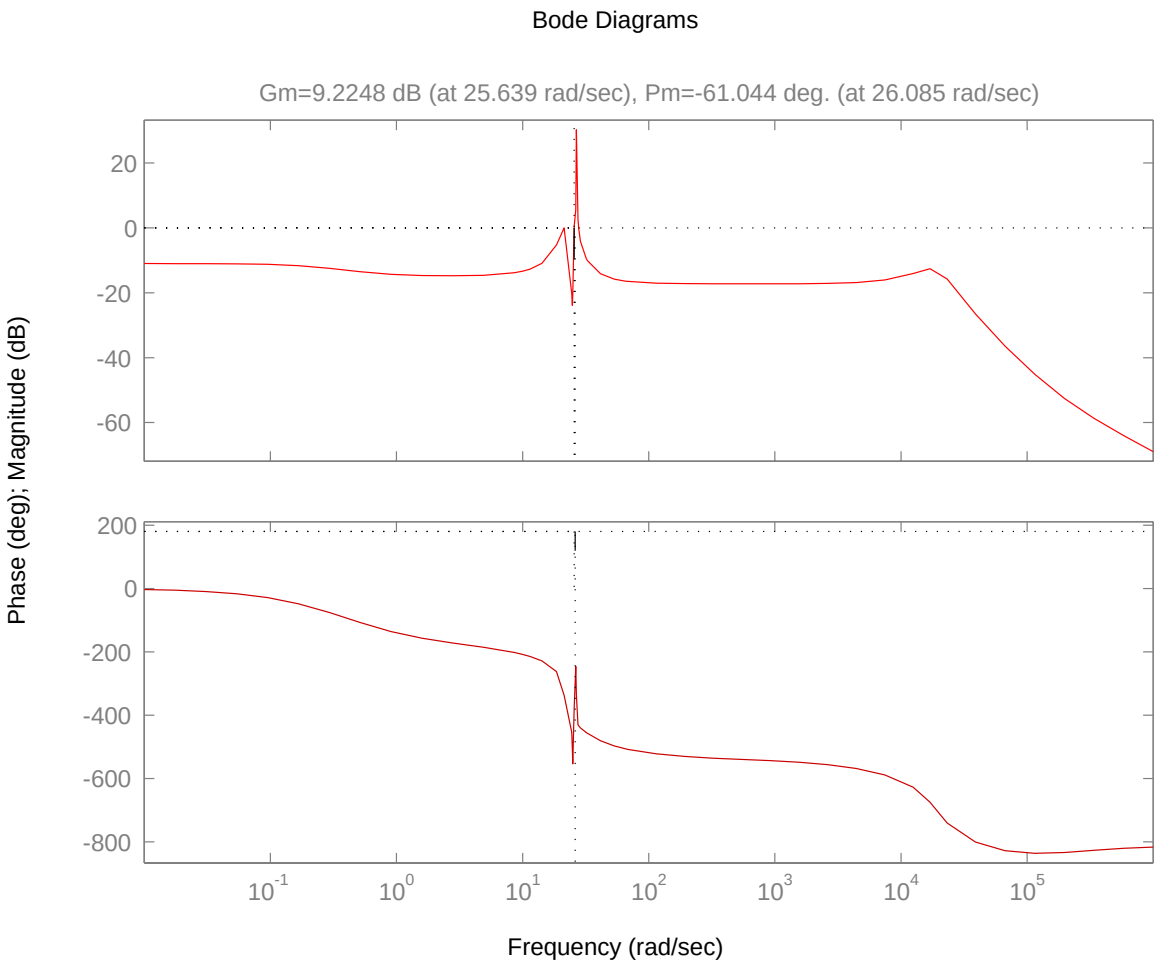

Figure 4.32 Case 8 Open-loop system Bode diagram

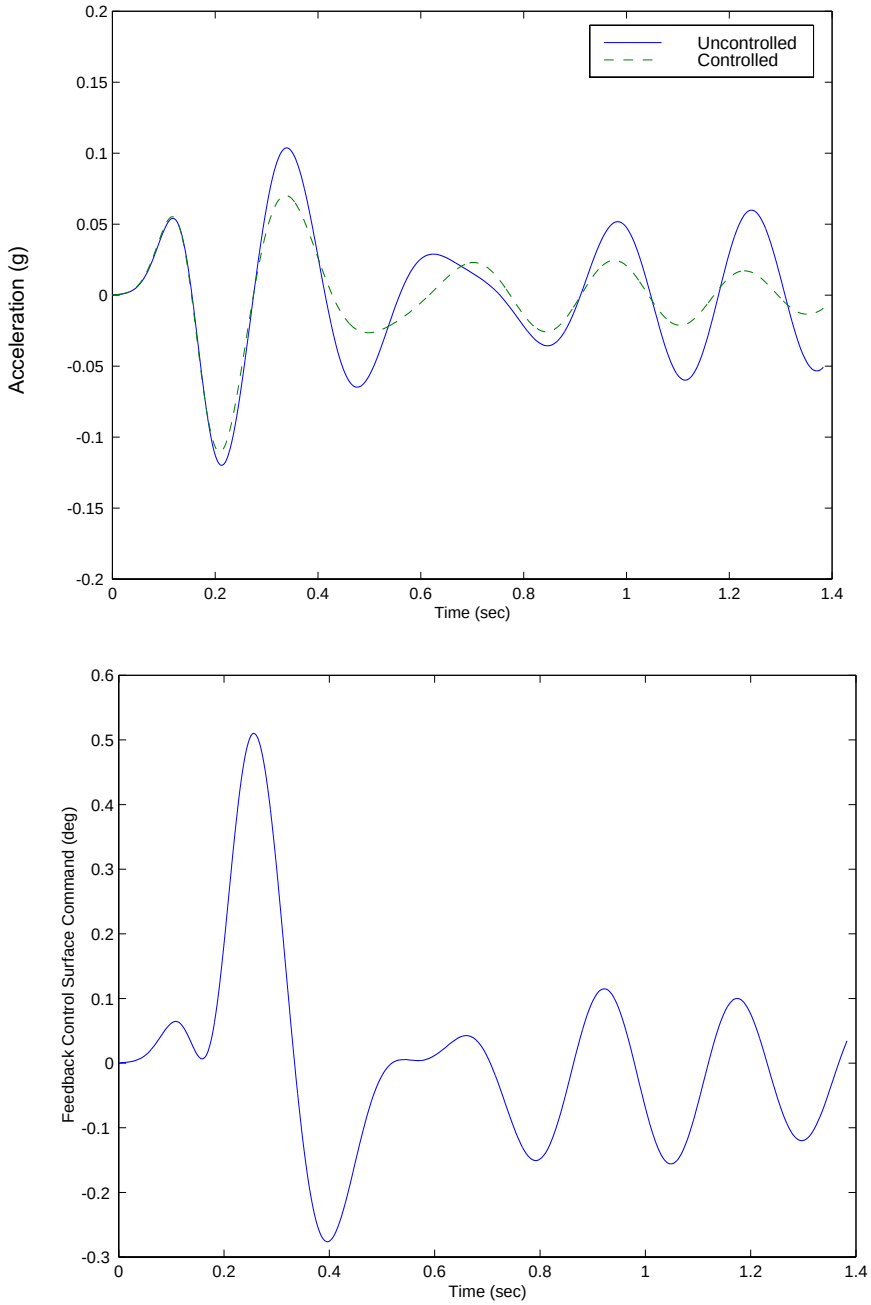

Figure 4.33 Case 8 controlled results

4.4.9 Case 9

Case 9 involves developing a system ID model from the CAP-TSD outputs (including the nonlinear terms) using the 3-d aerodynamic model at 0 degree angle of attack, designing an LQG control law, and evaluating the LQG control law using CAP-TSD (3-d). Figure 4.34 shows the results from the system ID procedure in terms of a comparison between the CAP-TSD outputs and the system ID model. Similar to the linear 2-d case, the system ID model provides a very good representation of the CAP-TSD outputs. The modal dampings and frequencies for this system ID model are given in Table 4.2.

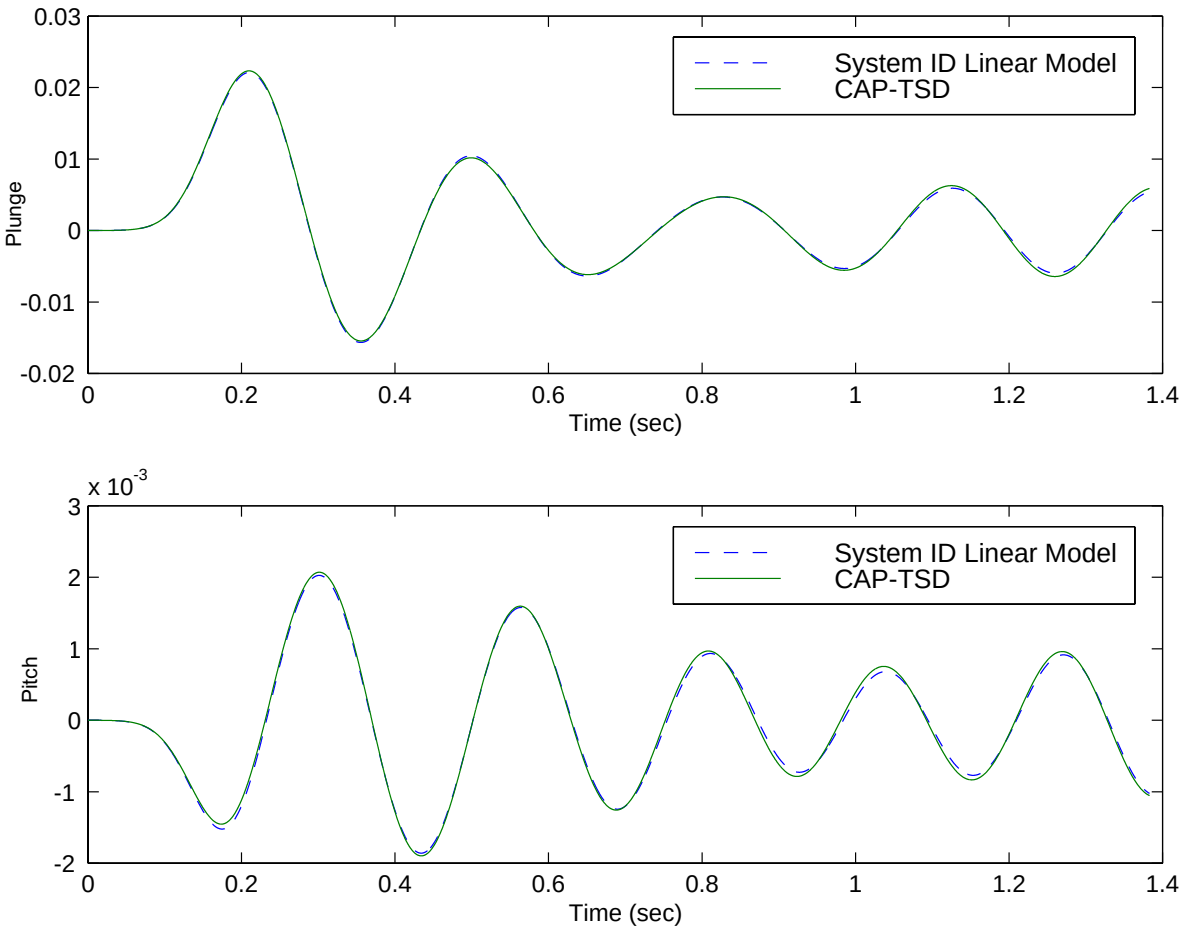

Figure 4.34 (a) Case 9 system ID results for displacements

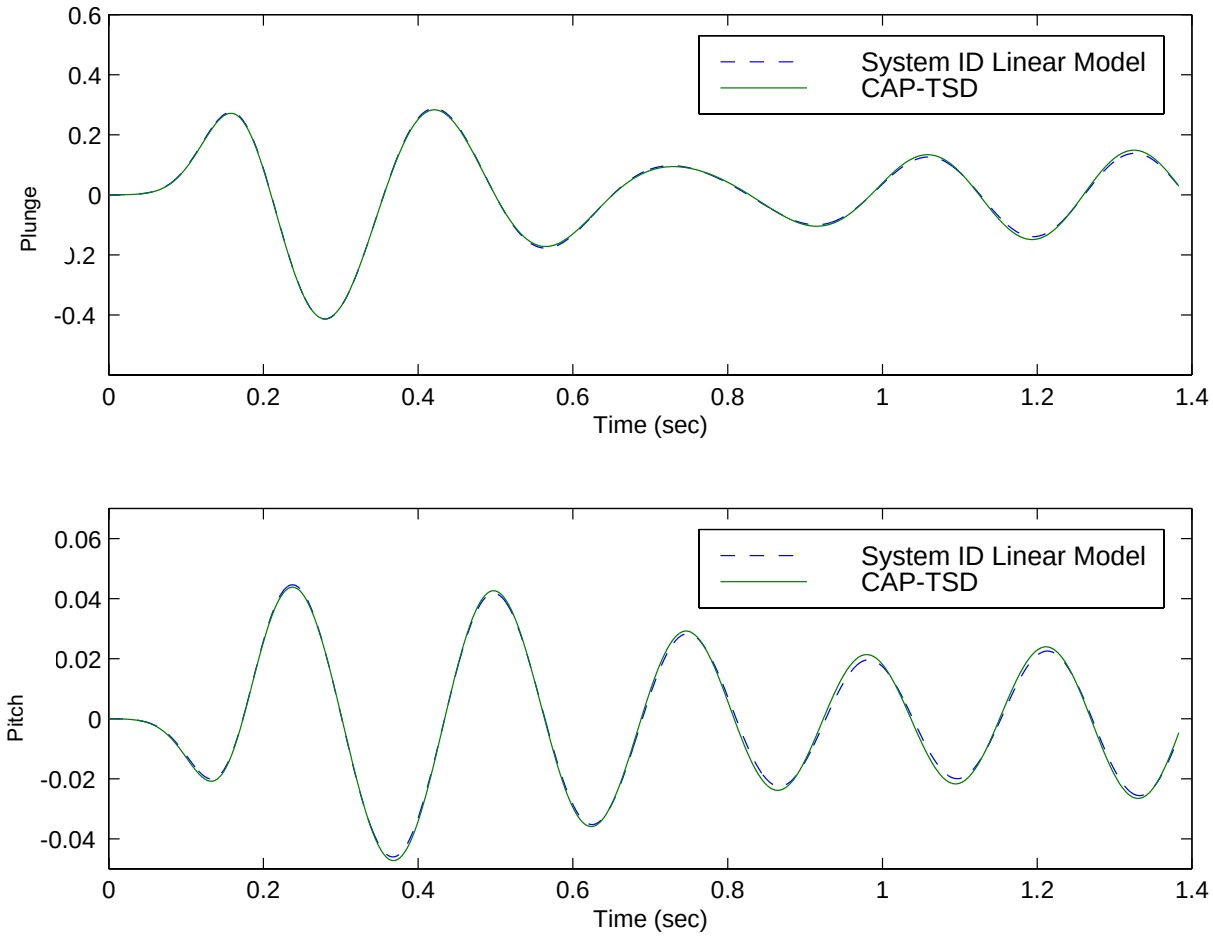

Figure 4.34 (b) Case 9 system ID results for velocities

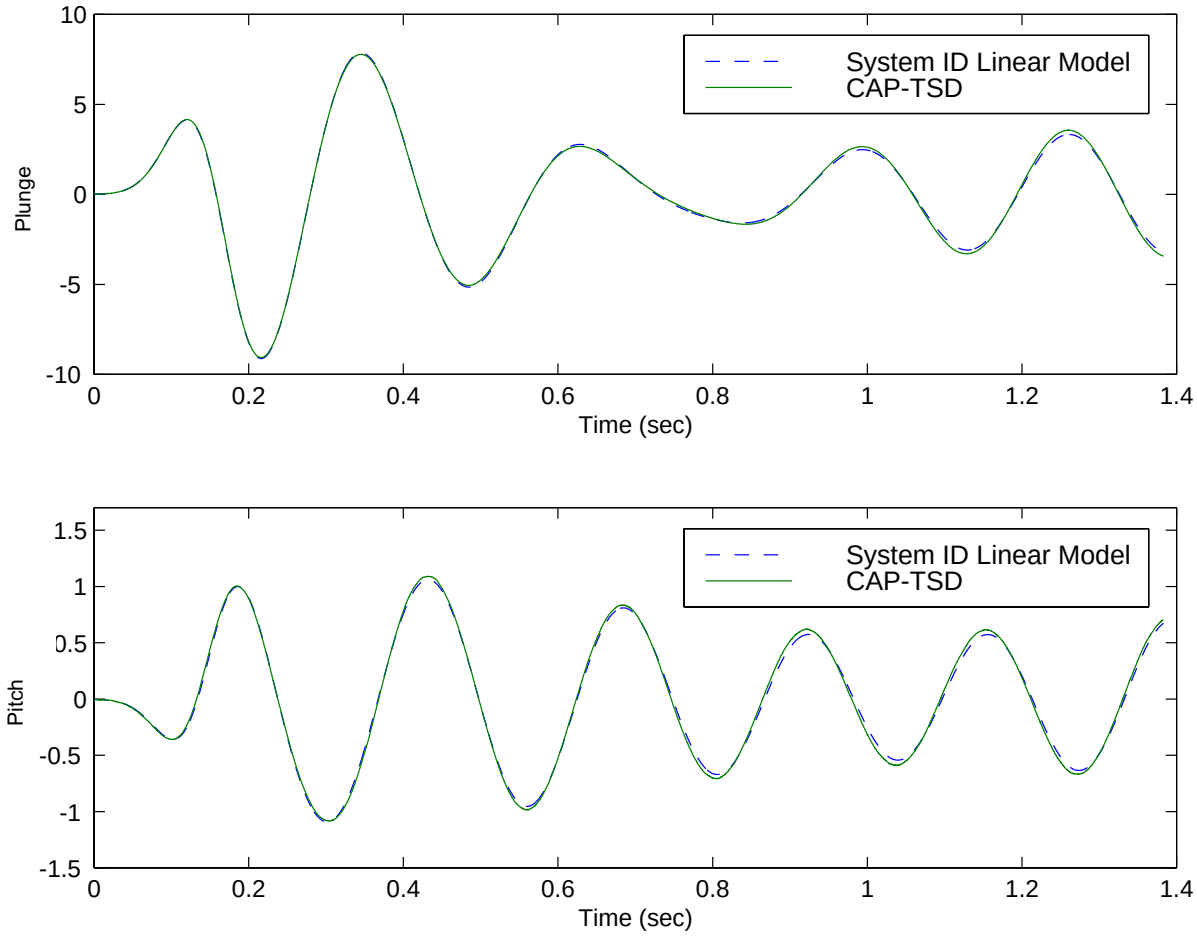

Figure 4.34 (c) Case 9 system ID results for accelerations

Figure 4.35 shows a Bode plot of the open-loop system that indicates a gain margin of –7.16 dB and a phase margin of –60.13 degrees. Figure 4.36 shows a comparison of CAP-TSD outputs for the uncontrolled and controlled case. The controlled case indicates

only a small decrease in response compared to the uncontrolled case. The small decrease is a result of the weights employed for the LQG control design. The weights were chosen to approximately reflect the unstable eigenvalue about the imaginary axis. This control law is the control law that stabilizes the system with the least amount of control input.

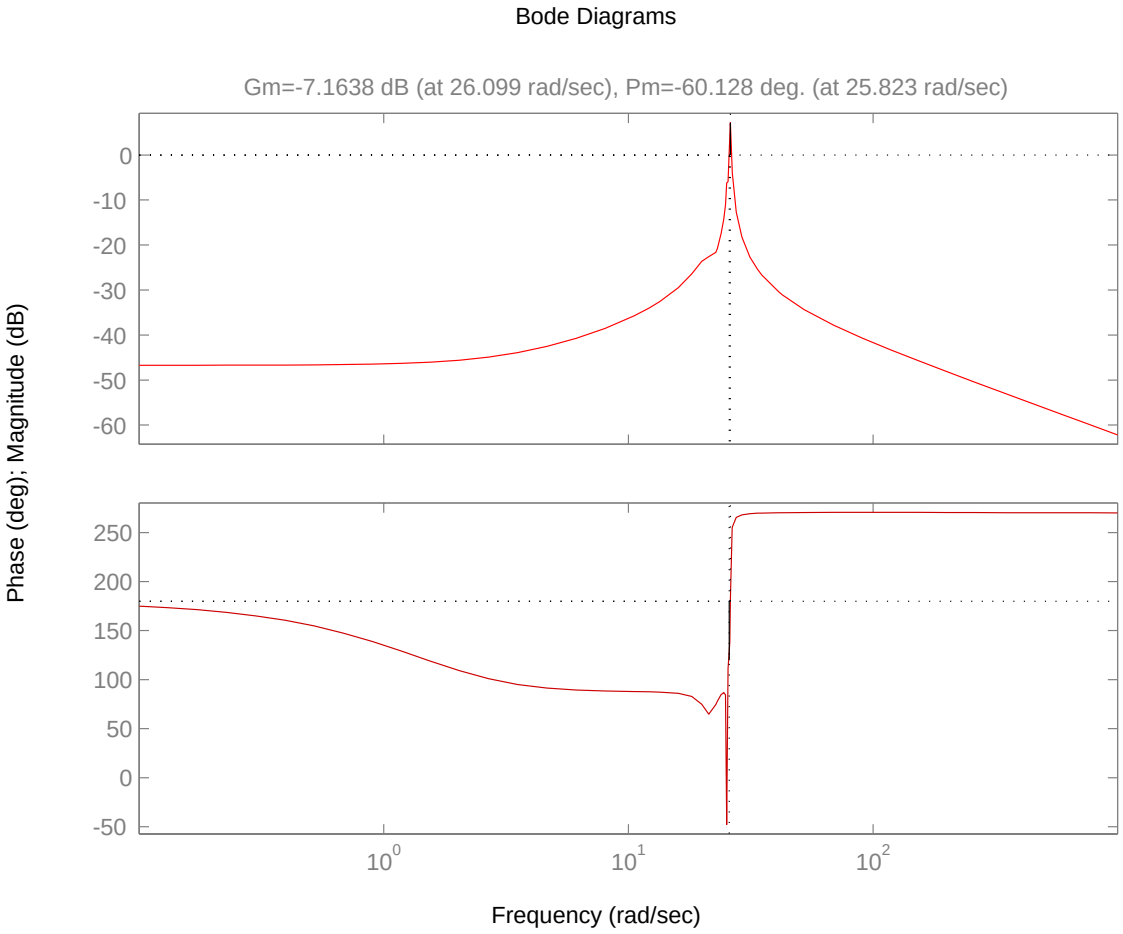

Figure 4.35 Case 9 Open-loop system Bode diagram

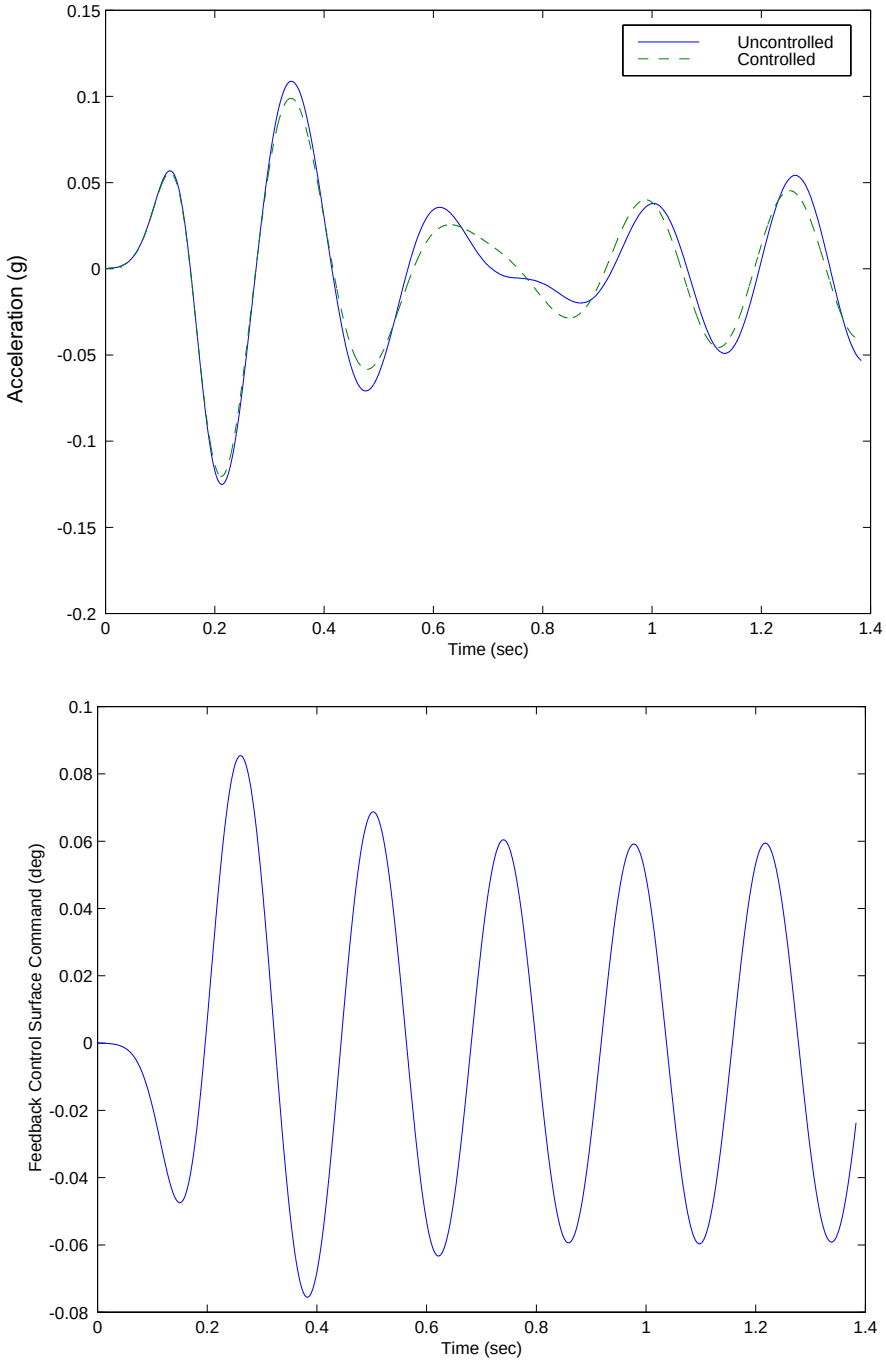

Figure 4.36 Case 9 controlled results

The values of the modal dampings and frequencies are given in Table 4.2. Figure 4.36 also shows the feedback control surface command for Case 9. The maximum control

surface displacement is approximately 0.08 degrees and occurs during the exponential pulse excitation.

4.4.10 Case 10

Case 10 involves developing a system ID model from the CAP-TSD outputs (including the nonlinear terms) using the 3-d aerodynamic model at 2 degrees angle of attack, designing an LQG control law, and evaluating the LQG control law using CAP-TSD (3-d). Figure 4.37 shows the results from the system ID procedure in terms of a comparison between the CAP-TSD outputs and the system ID model.

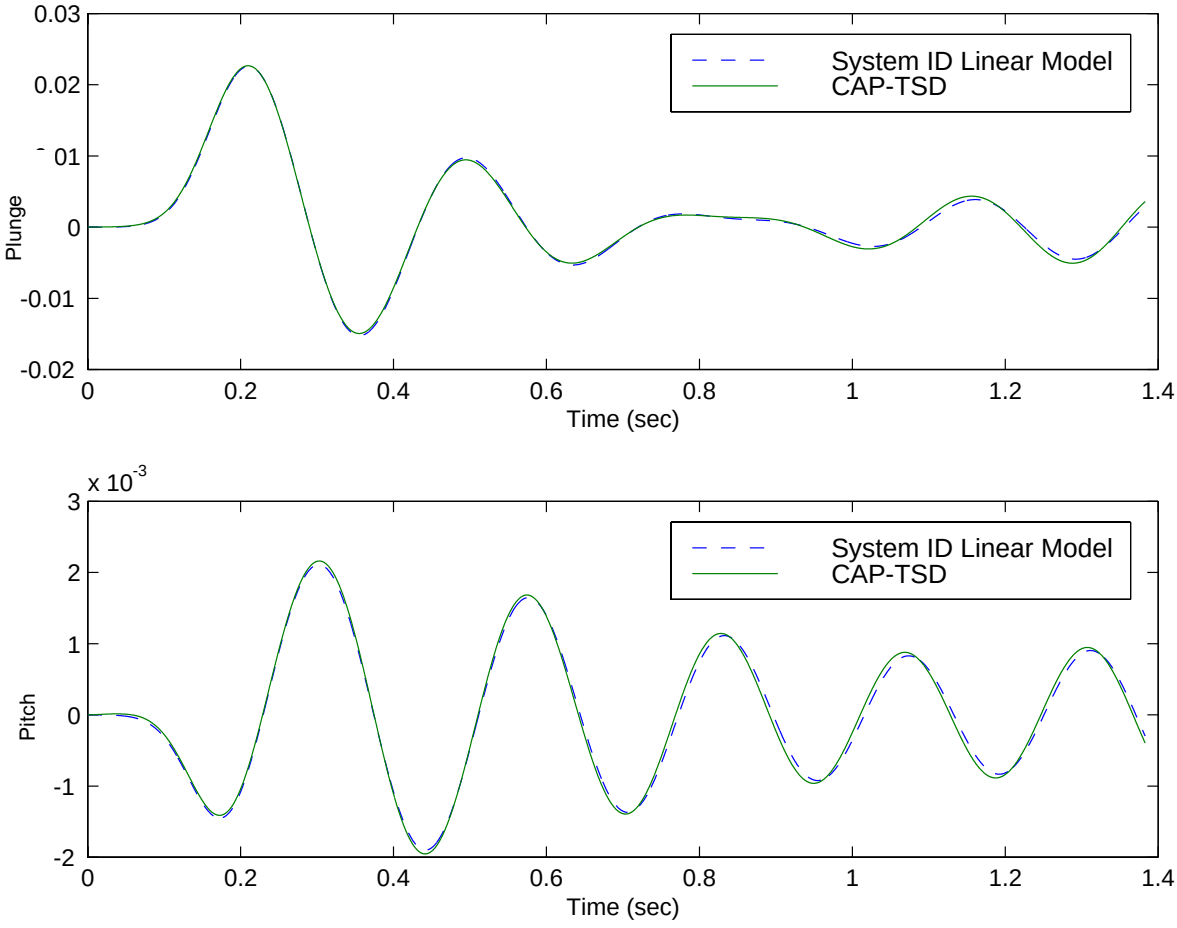

Figure 4.37 (a) Case10 system ID results for displacements

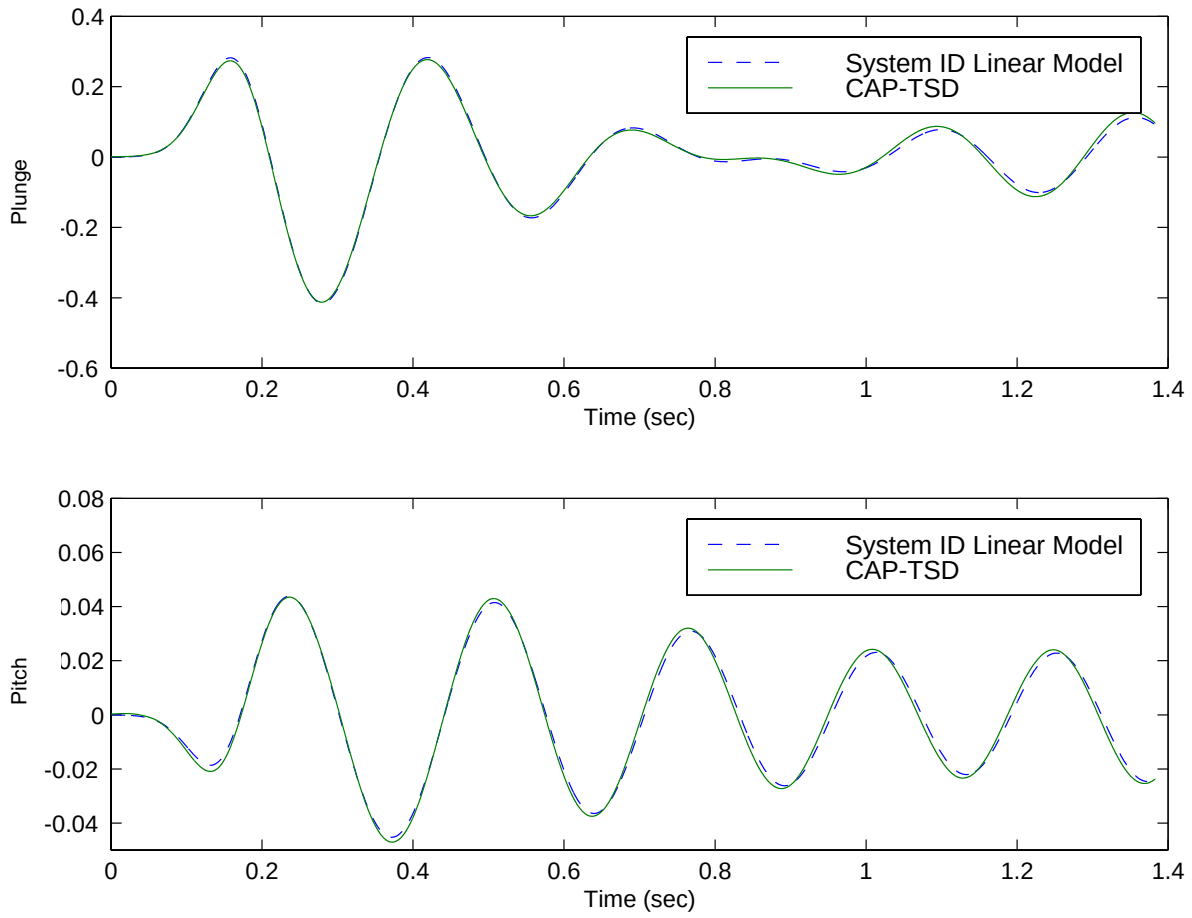

Figure 4.37 (b) Case 10 system ID results for velocities

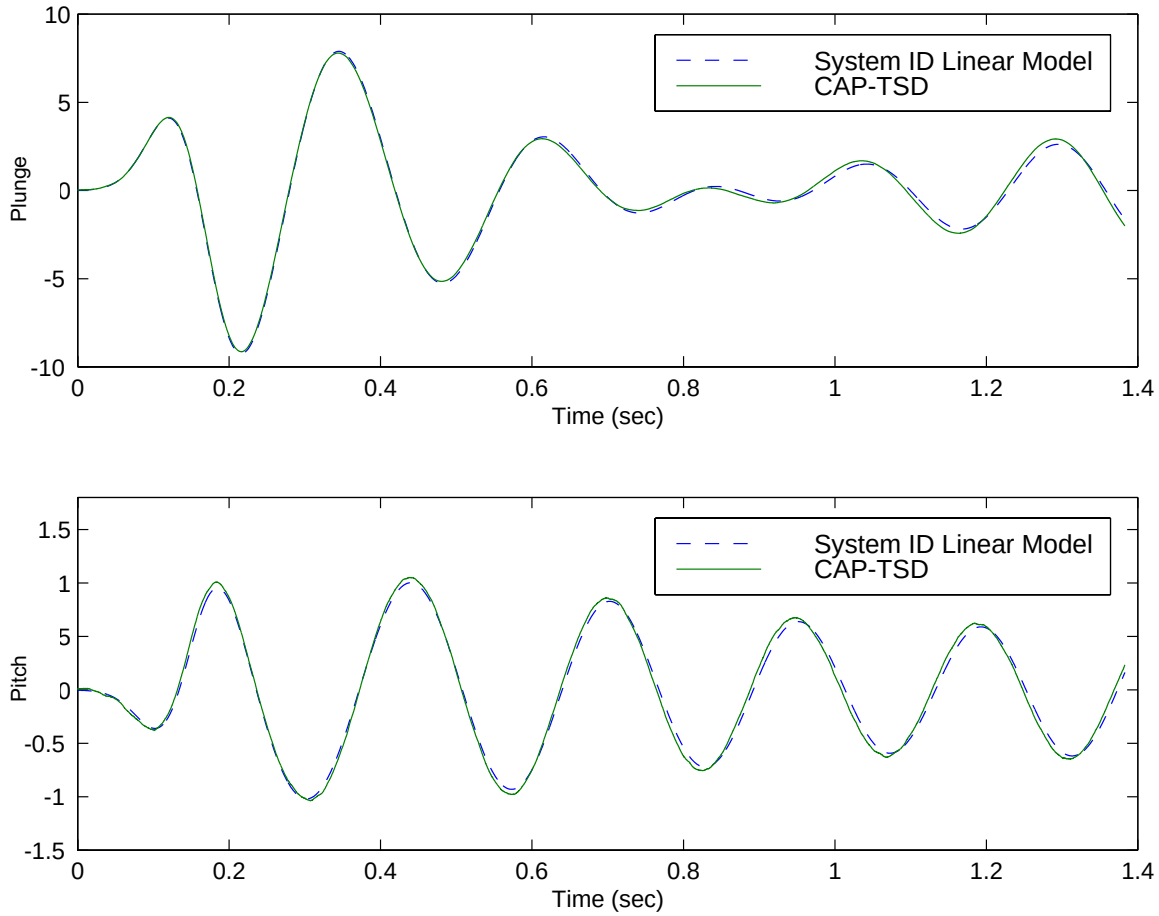

Figure 4.37 (c) Case 10 system ID results for accelerations

Similar to the 2-d case, the system ID model provides a very good representation of the CAP-TSD outputs. The modal dampings and frequencies for this system ID model are given in Table 4.2.

Figure 4.38 shows a Bode plot of the open-loop system that indicates a gain margin of –6.69 dB and a phase margin of –60.06 degrees. Figure 4.39 shows a comparison of CAP-TSD outputs for the uncontrolled and controlled case.

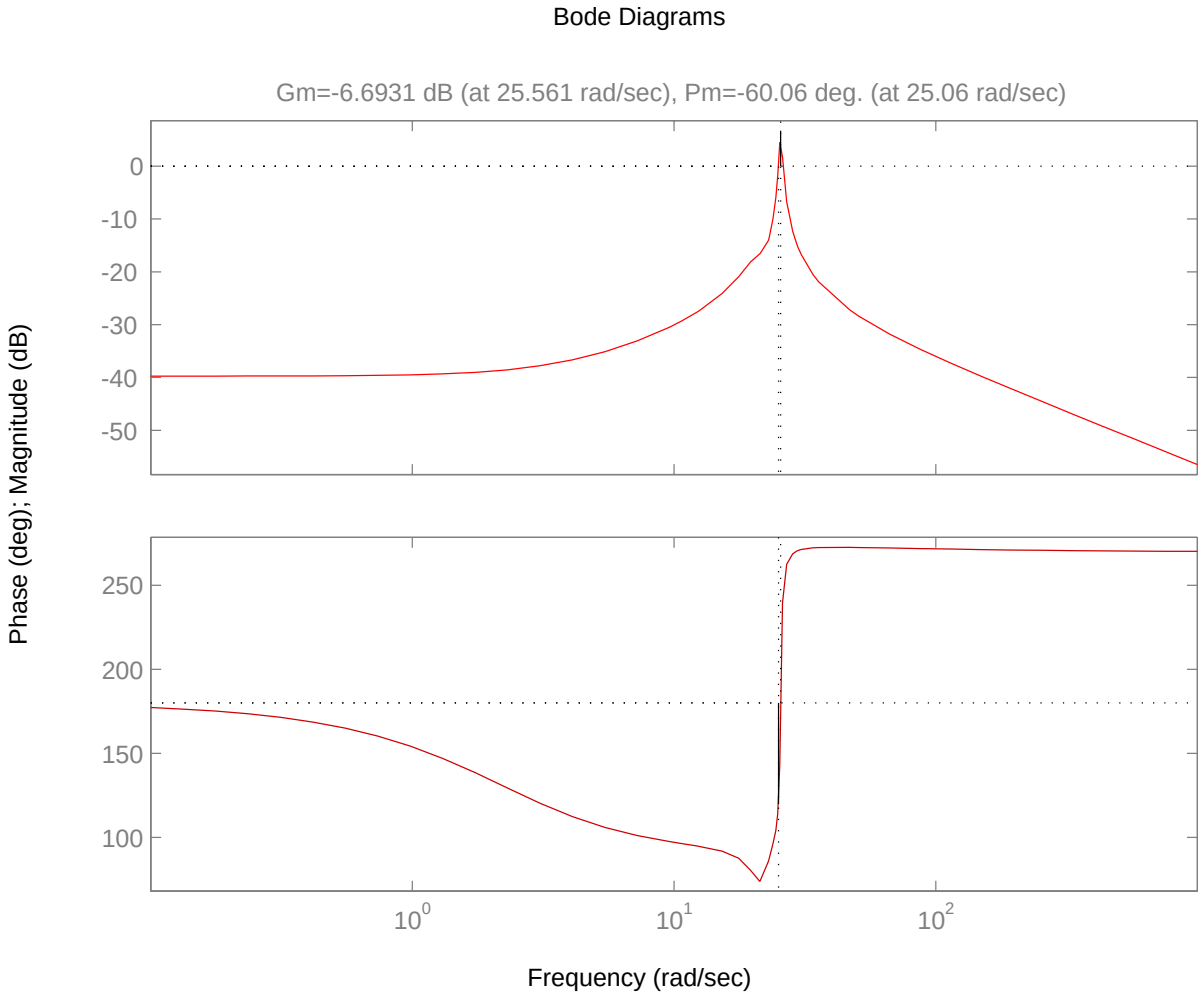

Figure 4.38 Case 10 Open-loop system Bode diagram

Similar to Case 9, the controlled case indicates only a small decrease in response compared to the uncontrolled case. The small decrease is again a result of the weights

employed for the LQG control design The values of the modal dampings and frequencies are given in Table 4.2. Figure 4.39 also shows the feedback control surface command for Case 10. The maximum control surface displacement is approximately 0.17 degrees and occurs during the exponential pulse excitation.

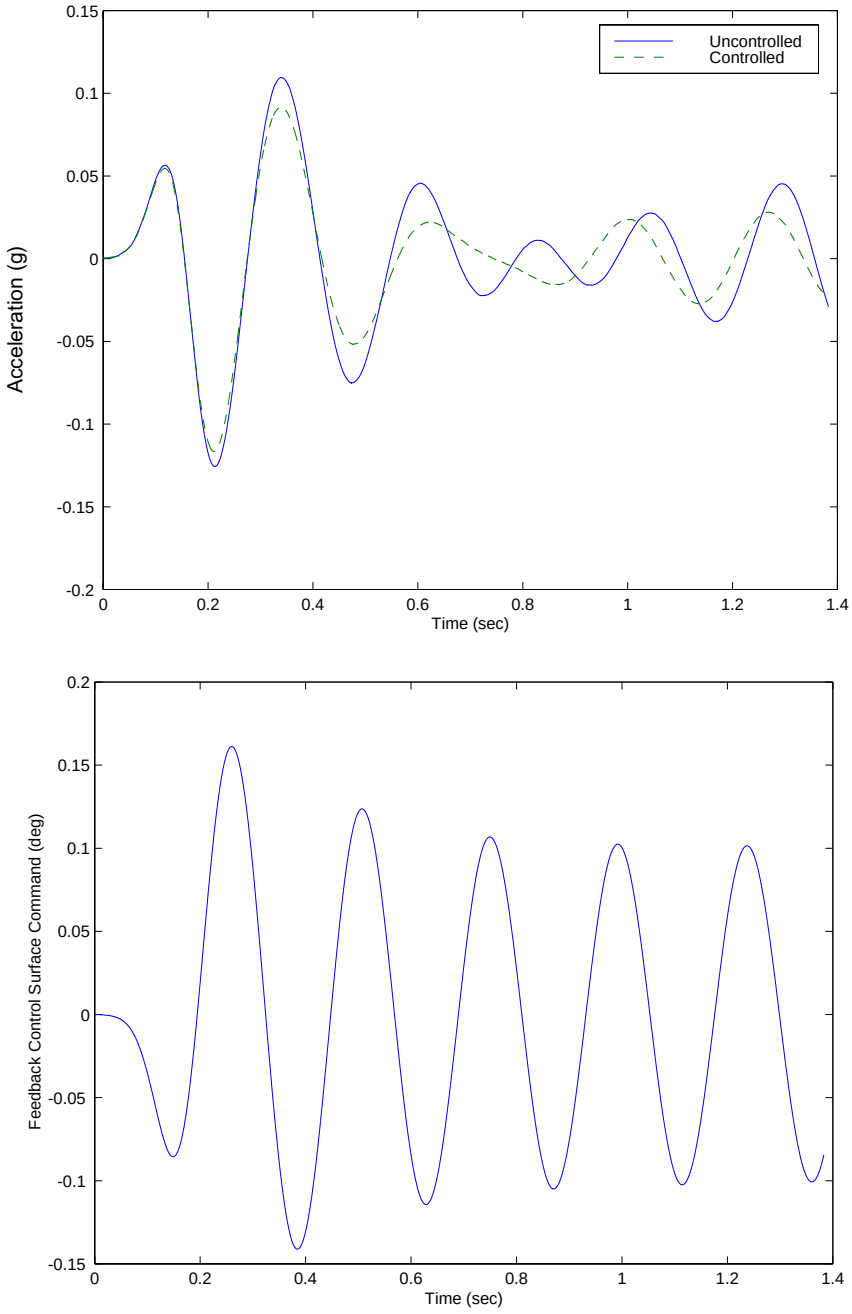

Figure 4.39 Case 10 controlled results

The results of this chapter indicate the capabilities and limitations of the methodology. The next chapter will summarize the major conclusions derived from the numerical results.

Chapter 5

Conclusions and Suggested Further Research

In summary, the objective of this dissertation was the development of a general methodology for designing active control laws in a computational aeroelasticity environment. The methodology involves using a modern system identification technique to develop an equivalent linear model from the nonlinear simulation results. Standard control law design techniques are then used to design control laws. The resulting control laws are then incorporated into the nonlinear computational aeroelasticity simulation for evaluation. Results of a numerical study applying this methodology to flutter control of the BACT wind-tunnel model were presented.

The major conclusions of this research are:

1. Equivalent linear models developed by employing a system identification technique can represent the input-output relationship of a computational aeroelasticity simulation very well.

2. For the BACT model used in this study, the system ID model represents the input-output relationship very well until the transonic flow conditions cause the shock on the upper surface to move aft of the 40% chord. At this point, extreme care must be used to obtain a good system ID model.
3. A control law designed using a system ID model developed from a nonlinear simulation can control the nonlinear model better than a control designed using a system ID model developed from a linear computational aeroelasticity simulation.

The major contributions of this research are:

1. Development of a methodology that can be used to design active control laws when employing a computational aeroelasticity simulation model.
2. Development of the modifications to a computational aeroelasticity code that allow the incorporation of feedback control laws during the aeroelasticity simulation.
3. Review of the state-of-the-art for designing active control laws using linear aeroelasticity models and comparison against the technique developed during this research.
4. Demonstrated utility of employing modern system identification techniques to estimate modal parameters from a computational aeroelasticity simulation.
5. Numerical results that illustrate the applicability of the present methodology for aeroelastic systems with nonlinear unsteady flow.

Several suggested further research topics are:

1. Use the methodology with higher-order computational fluid dynamics codes that incorporate viscous effects, especially for control surfaces.

2. Conduct studies on the effects of sample time, number of sampled data points, and system ID model order in using the OKID technique on an unstable nonlinear computational aeroelasticity simulation.
3. Explore other system identification techniques.
4. Explore other control law design techniques.
5. Explore the use of active structures for developing the control forces.

References

[1] Friedman, P. P.: "The Renaissance of Aeroelasticity and Its Future," SDM Lecture, 38th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, April 1997.

[2] Noll, T. E.: "Flight-Vehicle Materials, Structures, and Dynamics - Assessment and Future Directions", Volume 5, 1993, pp. 179 - 212.

[3] Dowell, E. H.: "Nonlinear Aeroelasticity," AIAA-90-1031-CP, April 1990.

[4] Breitbach, E.: "Effects of Structural Non-Linearities on Aircraft Vibration and Flutter," AGARD Report No. 665, January 1978.

[5] Murty, H. S.: "Aeroelastic Stability Analysis of an Airfoil with Structural Nonlinearities Using a State-space Unsteady Aerodynamic Model," AIAA-95-1293-CP, April 1995.

[6] ONeil, T. and Strganac, T. W.: "Nonlinear Aeroelastic Response - Analyses and Experiments," AIAA-95-1404-CP, April 1995.

[7] Yang, Z. C. and Zhao, L. C.: "Analysis of Limit Cycle Flutter of an Airfoil in Incompressible Flow," Journal of Sound and Vibration 123(1), 1988.

[8] Murty, H. S.: "Aeroelastic Stability Analysis of an Airfoil With Structural Nonlinearities Using a State-space Unsteady Aerodynamic Model", AIAA 95-1293-CP, April 1995.

[9] Baer-Riedhart, J.: "ERAST Program Marks Early Successes," Aerospace America, January 1998, pp. 36-41.

[10] Drela, M.: "Method for Simultaneous Wing Aerodynamic and Structural Load Prediction," AIAA Journal of Aircraft, Vol. 27, August 1990.

[11] Schoor, M. C., Zerweckh, S. H., and von Flotow, A. H.: "Aeroelastic Stability and Control of a Highly Flexible Aircraft," AIAA-89-1187-CP, April 1989.

[12] Scott, R. C. and Weisshaar, T. A.: "Panel Flutter Suppression Using Adaptive Material Actuators," Journal of Aircraft, Vol. 31, Number 1, Jan.- Feb. 1994, pp. 213-222.

[13] Ko, J., Kurdila, A. J., and Strganac, T. W.: "Nonlinear Dynamics and Control for a Structurally Nonlinear Aeroelastic System," AIAA 97-1024-CP, April 1997.

[14] Ku, C. S. and Hajela, P.: "Optimized Neural Network Based Controller for a Nonlinear Aeroelastic System," AIAA 97-1182-CP, April 1997.

[15] Moses, R. W.: "Vertical Tail Buffeting Alleviation Using Piezoelectric Actuators - Some Results of the Actively Controlled Response of Buffet-Affected Tails (ACROBAT) Program," NASA TM 110336, April 1997.

[16] Edwards, J. W. and Malone, J. B.: "Current Status of Computational Methods for Transonic Aerodynamics and Aeroelastic Applications," Paper No. 1 in Transonic Unsteady Aerodynamics and Aeroelasticity, AGARD CP 507, March 1992.

[17] Cunningham, A. M.: "The Role of Non-Linear Aerodynamics In Fluid-Structure Interaction," AIAA 98-2423, June 1998.

[18] Strganac, T. W. and Mook, D. T.: "Application of the Unsteady Vortex-Lattice Method to the Nonlinear Two-Degree-of-Freedom Aeroelastic Equations," AIAA 86-0867, May 1986.

[19] Strganac, T. W. and Mook, D. T.: "Numerical Model of Unsteady Subsonic Aeroelastic Behavior," AIAA Journal, Vol. 28, No. 5, May 1990, pp. 903-909.

[20] Preidikman, S. and Mook, D. T.: "Time-Domain Simulations of Linear and Nonlinear Aeroelastic Behavior," Journal of Vibration and Control, Vol. 6, June 2000, pp. 1135-1175.

[21] Ashley, H.: "On the Role of Shocks in the Sub-Transonic Flutter Phenomenom," AIAA 79-0765, April 1979.

[22] Bennett, R. M. and Edwards, J. W.: "An Overview of Recent Developments in Computational Aeroelasticity," AIAA 98-2421, June 1998.

[23] Rizzetta, D. P.: "Time-Dependent Response of a Two-Dimensional Airfoil in Transonic Flow," AIAA Journal, Vol. 17, January 1979, pp. 26-32.

[24] Yang, T. Y.; Guruswamy, P.; and Striz, A. G.: "Flutter Analysis of a NACA 64A006 Airfoil in Small Disturbance Transonic Flow," Journal of Aircraft, Vol. 17, No. 4, April 1980, pp. 225-232.

[25] Eastep, F. E. and Olsen, J. J.: "Transonic Flutter Analysis of a Rectangular Wing with Conventional Airfoil Sections," AIAA Journal, Vol. 18, No. 10, October 1980, pp. 1159-1164.

[26] Edwards, J. W.; Bennett, R. M.; Whitlow, W.; and Seidel, D. A.: "Time-Marching Transonic Flutter Solutions Including Angle-of-Attack Effects," Journal of Aircraft, Vol. 20, No. 11, November 1983, pp. 899-906.

[27] Kousen, K. A. and Bendiksen, O. O.: "Nonlinear Aspects of the Transonic Aeroelastic Stability Problem," AIAA 88-2306-CP, April 1988.

[28] Guruswamy, G. P.: "Interaction of Fluids and Structures for Aircraft Applications", Computers & Structures, Vol. 30, No. 1/2, 1988, pp. 1-13.

[29] Kousen, K. A. and Bendiksen, O. O.: "Limit Cycle Phenomena in Computational Transonic Aeroelasticity," AIAA 89-1185-CP, April 1989.

[30] Lee-Rausch, E. M. and Batina, J. T.: "Wing Flutter Boundary Prediction Using Unsteady Euler Aerodynamic Method," AIAA 93-1422-CP, April 1993.

[31] Batina, J. T.; Seidel, D. A.; Bland, S. R.; and Bennett, R. M.: "Unsteady Transonic Flow Calculations for Realistic Aircraft Configurations," Journal of Aircraft, Vol.26, No. 1, January 1989, pp. 21-28.

[32] Bennett, R. M.; Batina, J. T.; and Cunningham, H. J.: "Wing-Flutter Calculations with the CAP-TSD Unsteady Transonic Small-Disturbance Program," Journal of Aircraft, Vol. 26, No. 9, September 1989, pp. 876-882.

[33] Silva, W. A. and Bennett, R. M.: "Using Transonic Small Disturbance Theory for Predicting the Aeroelastic Stability of a Flexible Wind-Tunnel Model," NASA TM 102617, March 1990.

[34] Silva, W. A. and Bennett, R. M.: "Investigation of the Aeroelastic Stability of the AFW Wind-Tunnel Model Using CAP-TSD", NASA TM 104142, September 1991.

[35] Silva, W. A. and Bennett, R. M.: "Application of Transonic Small Disturbance Theory to the Active Flexible Wing Model", Journal of Aircraft, Vol. 32, No. 1, January - February 1995, pp 16-22.

[36] Jacobson, S. B.; Britt, R. T.; Dreim, D. R., and Kelly, P. D.: "Residual Pitch Oscillation (RPO) Flight Test and Analysis on the B-2 Bomber," AIAA 98-1805, April 1998.

[37] Edwards, J. W.: "Calculated Viscous and Scale Effects on Transonic Aeroelasticity," Paper No. 1, AGARD Structures and Materials Panel Workshop on Numerical Unsteady Aerodynamics and Aeroelastic Simulation, October 1997.

[38] Huttsell, L., Schuster, D., Volk, J., Giesing, J., and Love, M.: "Evaluation of Computational Aeroelasticity Codes for Loads and Flutter," A01-16435, January 2001.

[39] Silva, W., et.al,: "Reduced-Order Modeling: Cooperative Research and Development at the NASA Langley Research Center," International Forum on Aeroelasticity and Structural Dynamics 2001-008, June 2001.

[40] Ueda, T. and Dowell, E. H.: "Flutter Analysis Using Nonlinear Aerodynamic Forces", AIAA 82-0728-CP, April 1982.

[41] Silva, W. A.: "A Methodology for Using Nonlinear Aerodynamics in Aeroservoelastic Analysis and Design," NASA TM 104087, May 1991.

[42] Silva, W. A.: "Application of Nonlinear Systems Theory to Transonic Unsteady Aerodynamic Responses," Journal of Aircraft, Vol. 30, No. 5, September - October 1993, pp. 660-668.

[43] Silva, W. A.: "Discrete-Time Linear and Nonlinear Aerodynamic Impulse Responses for Efficient CFD Analyses," Ph.D. Dissertation, College of William and Mary, October 1997.

[44] Silva, W. and Raveh, D.: "Development of Unsteady Aerodynamic State-Space Models from CFD-Based Pulse Responses," International Forum on Aeroelasticity and Structural Dynamics 2001-1213, June 2001

[45] Marzocca, P.; Librescu, L; and Silva, W. A.: "Volterra Series Approach for Nonlinear Aeroelastic Response of 2-D Lifting Surfaces," AIAA-2001-1459, April 2001.

[46] Cowan, T. J.; Arena, A. S.; and Gupta, K. K.: "Accelerating CFD-Based Aeroelastic Predictions Using System Identification," AIAA-98-4152, January 1998.

[47] Cowan, T. J.; Arena, A. S.; and Gupta, K. K.: Development of a Discrete-Time Aerodynamic Model for CFD-Based Aeroelastic Analysis," AIAA-99-16614, January 1999.

[48] Dowell, E. H.; Hall, K. C.; and Romanowski, M. C.: "Eigenmode Analysis in Aerodynamics: Reduced Order Models," Applied Mechanics Reviews, Vol. 50, No. 6, June 1997.

[49] Tang, D.; Kholodar, D; Juang, J. N.; and Dowell, E. H.: "System Identification and Pod Method Applied to Unsteady Aerodynamics," NASA TM-2001-211243, December 2001.

[50] Batina, J. T. and Yang, T. Y.: "Transonic Calculation of Airfoil Stability and Response With Active Controls", AIAA 84-0873, April 1984.

[51] Guruswamy, G. P., Tu, E. L., and Goorjian, P. M.: "Transonic Aeroelasticity of Wings with Active Control Surfaces", AIAA 87-0709-CP, April 1987.

[52] Ominsky, D. and Ide, H.: "An Effective Flutter Control Method Using Fast, Time-Accurate CFD Codes", AIAA 89-3468-CP, April 1989.

[53] Bendiksen, O., Hwang, G., and Piersol, J.: "Nonlinear Aeroelastic and Aeroservoelastic Calculations for Transonic Wings," AIAA 98-1898, April 1998.

[54] Stephens, C, Arena, A., and Gupta, K.: "CFD-Based Aeroservoelastic Predictions with Comparisons to Benchmark Experimental Data," AIAA 99-0766, January 1999.

[55] Djayapertapa, L. and Allen, C. B.: "Aeroservoelastic Computations in Unsteady Transonic Flow," AIAA 2000-4226, August 2000.

[56] Guillot, D. M. and Friedmann, P. P.: "A Fundamental Aeroservoelastic Study Combining Unsteady CFD With Adaptive Control", AIAA 94-1721-CP, April 1994.

[57] Friedmann, P. P., Guillot, D. M., and Presente, E.: "Adaptive Control of Aeroelastic Instabilities in Transonic Flow and Its Scaling", AIAA 97-0581-CP, January 1997.

[58] Tiffany, S.H. and Karpel, M.:"Aeroservoelastic Modeling and Applications Using Minimum-State Approximations of the Unsteady Aerodynamics," NASA TM-101574, April 1989.

[59] Batina, J. T.: "Efficient Algorithm for Solution of the Unsteady Transonic Small-Disturbance Equation", Journal of Aircraft, Volume 25, July 1988, pp. 598-605.

[60] Juang, J.: "An Overview of Recent Advances in System Identification," AIAA-93-1664, April 1993.

[61] Phan, M Q.;Horta, L. G., Juang, J., and Longman, R. W.: "Identification of Linear Systems by an Asymptotically Stable Observer", NASA TP 3164, June 1992.

[62] Juang, J., Horta, L. G., and Phan, M.: "System/Observer/Controller Identification Toolbox", NASA TM 107566, February 1992.

[63] Scott, M; Gilbert, M;, and Demeo, M.: "Active Vibration Damping of the Space Shuttle Remote Manipulator System," Proceedings of the AIAA GN&C Conference, August 1991.

[64] Chen, W. and Valasek, J.: "Observer/Kalman Filter Identification for On-line System Identification of Aircraft," AIAA-99-4173, August 1999.

[65] Juang, J.N. and Pappa, R.S.:"An Eigensystem Realization Algorithm for Modal Parameter Identification and Model Reduction," Journal of Guidance, Control, and Dynamics, Vol. 8, No. 5, 1985, pp. 620-627.

[66] Scott, R. C., Hoadley, S. T., Wieseman, C. D., and Durham, M. H.: "The Benchmark Active Controls Technology Model Aerodynamic Data", AIAA 97-0829-CP, January 1997.

[67] Bisplinghoff, R., Ashley, H., and Halfman, R.: Aeroelasticity, Addison Wesley Publishing Company, Reading, Massachusetts, 1954.

Appendix A

State Transition Matrix (ϕ) and Integral of the State Transition Matrix (θ)

Both the state transition matrix (ϕ) and the integral of the state transition matrix (θ) are block diagonal matrices of the form

$$\phi = \begin{bmatrix} \phi^1 & 0 & .... & 0 \\ 0 & \phi^2 & .... & 0 \\ . & .. & . & . \\ 0 & .. & .. & \phi^N \end{bmatrix}$$

$$\theta = \begin{bmatrix} \theta^1 & 0 & .... & 0 \\ 0 & \theta^2 & .... & 0 \\ . & .. & . & . \\ 0 & .. & .. & \theta^N \end{bmatrix}$$

where each ϕ^i and θ^i are 2x2 sub-matrices and N is the number of vibration modes employed in eq. 3.1. Each element of the 2x2 sub-matrices is expressed in terms of σ_i and γ_i defined as

$\sigma_i = -\xi_i \omega_i$

$\gamma_i = \omega_i \sqrt{1-\xi_i^2}$

where ξ_i is the damping of the i[th] mode, ω_i is the natural frequency of the i[th] mode and ΔT is the integration time.

$$\phi_{11}^i = e^{\sigma_i \Delta T}(\cos\gamma_i \Delta T - \frac{\sigma_i}{\gamma_i}\sin\gamma_i \Delta T)$$

$$\phi_{12}^i = e^{\sigma_i \Delta T}(\frac{1}{\gamma_i}\sin\gamma_i \Delta T)$$

$$\phi_{21}^i = -e^{\sigma_i \Delta T}((\gamma_i + \frac{\sigma_i^2}{\gamma_i})\sin\gamma_i \Delta T)$$

$$\phi_{22}^i = e^{\sigma_i \Delta T}(\cos\gamma_i \Delta T + \frac{\sigma_i}{\gamma_i}\sin\gamma_i \Delta T)$$

$$\theta_{11}^i = \frac{(2\sigma_i e^{\sigma_i \Delta T}\cos\gamma_i \Delta T + (\gamma_i - \frac{\sigma_i^2}{\gamma_i})e^{\sigma_i \Delta T}\sin\gamma_i \Delta T - 2\sigma_i)}{\sigma_i^2 + \gamma_i^2}$$

$$\theta_{12}^i = \frac{(\frac{\sigma_i e^{\sigma_i \Delta T}\sin\gamma_i \Delta T}{\gamma_i} - e^{\sigma_i \Delta T}\cos\gamma_i \Delta T + 1)}{\sigma_i^2 + \gamma_i^2}$$

$$\theta_{21}^i = \frac{-\sigma_i e^{\sigma_i \Delta T}\sin\gamma_i \Delta T}{\gamma_i} + e^{\sigma_i \Delta T}\cos\gamma_i \Delta T - 1$$

$$\theta_{22}^i = \frac{e^{\sigma_i \Delta T}\sin\gamma_i \Delta T}{\sigma_i}$$

Appendix B

Matlab m-files

```
%  This m-file reads the outputs and control inputs from a
%    CAP-TSD simulation and also computes the time vector
%
load sysid3d2035   %  read in the outputs from file 'sysid3d2035'
x=sysid3d2035;
clear sysid3d2035;
load upulse     %   read in the control surface command input from file 'upulse'
u=upulse(:,6);
clear upulse
nout=6;          %  number of outputs
ninp=1;          %  number of inputs
capdt=.05;       %  nondimensional CAP-TSD delta-t
reflen=1.333333; %  CAP-TSD reference length
nsteps=8000;     % number of time steps used in the CAP-TSD simulation
vel=385.6;       %    velocity
dt=capdt*reflen/vel;    %  compute real delta-t
t=[0:dt:(nsteps-1)*dt]; %  construct time vector
clear capdt reflen vel;
```

```
% This m-file identifies a discrete state-space model given "measurements"
% x and control input u from CAP-TSD.
%
% x - matrix of generalized coordinate displacements, their rates, and their
%      accelerations
% dt - CAP-TSD integration time (real time)
%
%
```

```
%  remove biases on displacements
x(:,1)=x(:,1)-x(1,1);
x(:,2)=x(:,2)-x(1,2);
pp=1;% parameter for increasing assumed system order
nt=1;% nt=time subsampling interval
dt=nt*dt;
u=u([1:nt:nsteps],1);
x=x([1:nt:nsteps],:);
t=t(1,[1:nt:nsteps]);
nsteps=nsteps/nt;
[a,b,c,d,g]=okid(nout,ninp,dt,u,x,'batch_oc',pp);
sys=ss(a,b,c,d,dt);
[xh]=lsim(sys,u,t);
plot(t,x(:,1),t,xh(:,1),'--')
ylabel('Plunge Displacement')
xlabel('Time (sec)')
legend('CAP Output,Linear System ID Model')
pause
plot(t,x(:,2),t,xh(:,2),'--')
ylabel('Pitch Displacement')
xlabel('Time (sec)')
legend('CAP Output,Linear System ID Model')
pause
plot(t,x(:,3),t,xh(:,3),'-- ')
ylabel('Plunge Velocity')
xlabel('Time (sec)')
legend('CAP Output,Linear System ID Model')
pause
plot(t,x(:,4),t,xh(:,4),'--')
ylabel('Pitch Velocity')
xlabel('Time (sec)')
legend('CAP Output,Linear System ID Model')
pause
plot(t,x(:,5),t,xh(:,5),'--')
ylabel('Plunge Acceleration')
xlabel('Time (sec)')
legend('CAP Output,Linear System ID Model')
pause
plot(t,x(:,6),t,xh(:,6),'--')
ylabel('Pitch Acceleration')
xlabel('Time (sec)')
legend('CAP Output,Linear System ID Model')
```

```
% This m-file calculates an LQG controller given a discrete system
% identified from CAP-TSD output (using sysid.m)
%
% The controller is computed in continuous time and then discretized
% at the CAP-TSD integration time dt (real time)
%
%
phi=[0 0 0 0 .4171  -.40369];  % matrix of modal coefficients at sensor
sysd=ss(a,b,phi*c,phi*d,dt); % Trailing edge acceleration feedback
q=.15         %  output weight
r=200000.     %  input weight
sysc=d2c(sysd);
[acc,bcc,ccc,dcc]=ssdata(sysc);
k=lqry(sysc,q,r);
p=ss(acc,[bcc bcc],ccc,[dcc dcc]);
noise1=1;     %  process noise
noise2=.000001;  %  sensor noise
kest=kalman(p,noise1,noise2);
f=lqgreg(kest,k);
gain=1.
[atemp,btemp,ctemp,dtemp]=ssdata(f);
ctemp=gain*ctemp;
dtemp=gain*dtemp;
f=ss(atemp,btemp,ctemp,dtemp);
clsys=feedback(sysc,f,+1);
damp(sysc)
damp(clsys)
[ycl,t,xcl]=lsim(clsys,u,t);
[yol]=lsim(sysc,u,t);
plot(t,yol(:,1),t,ycl(:,1),'--')
pause
[ac,bc,cc,dc]=ssdata(f);
dc=[dc;cc*bc];
cc=[cc;cc*ac];
f=ss(ac,bc,cc,dc);
fd=c2d(f,dt/nt);
[acd,bcd,ccd,dcd]=ssdata(fd);
[ac1,bc1,cc1,dc1]=ssdata(f);
sysol=series(-sysc,ss(ac1,bc1,cc1(1,:),dc1(1,:)));
margin(sysol)
```

```
%  This m-file writes the controller matrices acd, bcd, ccd, dcd
%   (discretized at the CAP-TSD integration time dt)
%    computed using m-file cdesign.m
%   to a file (fort.77) to transfer to THEO to read into CAP
%    for closed-loop simulations
%
%
fid=fopen('fort.77','w')
fprintf(fid,'%15.6e\n',acd);
fprintf(fid,'%15.6e\n',bcd);
fprintf(fid,'%15.6e\n',ccd);
fprintf(fid,'%15.6e\n',dcd);
status=fclose(fid)
```

Vita

Mr. Jerry Russell Newsom is currently the Director of the Project Implementation Office at the NASA Langley Research Center (LaRC) in Hampton, Virginia.

Previously, Mr. Newsom has held several positions at LaRC including Director, Earth and Space Science Program Office, Assistant Division Chief of the Atmospheric Sciences Division, Leader of the Spacecraft Thrust Office, and Manager of the Controls-Structures Interaction Office.

From December 1986 to June 1988, Mr. Newsom was assigned to the Administrator's Office at NASA Headquarters as the Administrator's Technical Assistant. Prior to the Headquarters assignment, he was Head of the Aeroservoelasticity Branch at Langley. Mr. Newsom earned his Bachelors degree in Engineering Science and Mechanics from VPISU in 1973 and his Masters degree in Aerospace Engineering from George Washington University in 1978.

Mr. Newsom has been the author or co-author of over 25 publications and has received several NASA awards, including the NASA Outstanding Leadership Medal in 1992.