MODELING AND CONTROL OF ROTATING STALL IN HIGH SPEED MULTI-STAGE AXIAL COMPRESSORS

by

Matthew Roger Feulner

B.S., Purdue University, 1988
M.S., University of Texas at Austin, 1990

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

DOCTOR OF PHILOSOPHY

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 1994

Signature of Author ______________________________
Department of Aeronautics and Astronautics
April 27, 1994

Certified by ______________________________
Professor James D. Paduano
Thesis Supervisor, Department of Aeronautics and Astronautics

Certified by ______________________________
Dr. Lena Valavani
Chief Scientist, Systems Engineering
Volpe National Transportation Systems Center

Certified by ______________________________
John R. Dowdle
Leader, Control and Dynamical Systems Group
Charles Stark Draper Laboratory, Inc.

Certified by ______________________________
Professor Richard H. Battin
Department of Aeronautics and Astronautics

Accepted by ______________________________
Professor Harold Y. Wachman
Chairman, Department Graduate Committee

Modeling and Control of Rotating Stall in High Speed Multi-Stage Axial Compressors

by

Matthew Roger Feulner

Submitted to the Department of Aeronautics and Astronautics
April 27, 1994 in partial fulfillment of the requirements
for the Degree of Doctor of Philosophy

Abstract

Using a two-dimensional, compressible flow representation of axial compressor dynamics, a control-theoretic input-output model is derived which is of general utility in rotating stall and surge active control studies. Geometry of a three stage research compressor is used in this model to perform control configuration studies using operating range extension and mean-square costs in the comparisons.

The derivation presented begins with a review of the fluid dynamic model, which is a 2D stage stacking technique that accounts for blade row pressure rise, loss and deviation as well as blade row and inter-blade row compressible flow. This model is extended to include the effects of the upstream and downstream geometry and boundary conditions, then manipulated into a transfer function form that dynamically relates actuator motion to sensor measurements.

Transcendental functions in this input-output form are then approximated using rational polynomials. Further manipulations yield an approximate state-space model which is in standard form for studying active control of rotating stall and surge.

Specifications on the control system are proposed which are standard restrictions in modern control theory. Cost functions are proposed to compare control configurations. Two control design methods are presented to design the compensators. The linear quadratic-Gaussian optimal control minimizes a mean-square cost function of the perturbations and the control activity. The H_∞ optimal control minimizes the H_∞-norm related to the specifications.

The control design methods are then applied to the geometry of a three stage research compressor. As examples of high current relevance, the transfer functions from an array of jet actuators to arrays of either static pressure, stagnation pressure, and velocity sensors are examined. It is found that using a jet actuator that reduces the momentum of the flow can be as good as a jet actuator which adds equivalent momentum. It is also found that velocity and stagnation pressure sensors are superior to static pressure sensors.

Thesis Supervisor: Professor James Paduano
Assistant Professor of Aeronautics and Astronautics

Acknowledgments

The work described in this thesis would not have been possible without the contributions of many different people.

First, I would like to thank Dr. Lena Valavani for her support and enthusiasm. I would still be looking for a thesis topic if not for her ideas and her willingness to spend time with students.

I would like to thank Professor James Paduano for many things. He helped me keep things in perspective and shifted me to fruitful directions. He spent a lot of time explaining the concepts of rotating stall and endured an uncountable number of equations.

I would like to thank Dr. John Dowdle, who was willing to learn about rotating stall, and agreed to advise me on the matters of control theory. He always made me think about what exactly I was trying to do. Also, I'd like to thank Dr. Brent Appleby, who was always willing to discuss finer points of control theory. He didn't always think he was helping, but he was.

I would like to thank Professor Battin for his support throughout my stay at MIT, from the qualifying exams through the thesis defense.

I would like to thank Dr. Gavin Hendricks. He was my first point of contact with the fluids community and helped me understand the dynamics. It was often tough talking to him because of the communication gap between our fields, but it was definitely worth the effort.

I would like to thank Professors Epstein and Greitzer in the Gas Turbine Lab for their leadership and the weekly smart engines meetings.

At the Charles Stark Draper Laboratory, Inc., I would like to thank George Schmidt, Ron Harris, Dale Landis, John Sweeney and Joan Chiffer for providing me with support, even though I ended up doing my research for the GTL.

Now, those people who kind of hung around and made life bearable. First, my officemates: Bryan K. (who was always available to talk to about Ph. D.-type stuff), Lt. Brad Mai (who introduced me to Jimmy Buffett concerts, among other things), Ensign Richard Needham (Sweet!) and J. Chris Stoll (Swe-e-e-et!). And the others: Tony, Debbie, Eric, John D., Jeff, John G., Lisa and Wendy, who all made graduate student life just a bit easier to tolerate.

Most important are my family and my wife Zonda, who put their support behind me 110%. I could never have done this without them.

This thesis was prepared at The Charles Stark Draper Laboratory, Inc.

Publication of this thesis does not constitute approval by Draper or the Massachusetts Institute of Technology of the findings or conclusions contained herein. It is published for the exchange and stimulation of ideas.

Matthew R. Feulner

Table of Contents

List of Figures

List of Tables

Nomenclature

Chapter 1

ϕ	Flow Coefficient
ψ	Pressure Rise

Chapter 2

a	Speed of sound
j	$\sqrt{-1}$
$\dot{m}$	Mass flow rate
r	Radius
s	Laplace transform variable ($s=j\omega$)
t	Time
u	Control variable
x, x'	Axial distance, distance along blade row
$B,C,D,E(s)$	Coefficients in inter-blade row gap solution
$\tilde{B},\tilde{C},\tilde{E}(s)$	Coefficients in blade row solution
M_x, M_θ	Axial and Circumferential Mach numbers (V/a)
S_{entr}	Entropy
V_x, V_θ	Axial and circumferential velocities
V_{plen}	Volume of plenum in Section 2.2.5.2
W	Velocity in blade row
$\alpha_n,\beta_n(s)$	Pressure exponents in inter-blade row gap solutions
$\tilde{\alpha}_n,\tilde{\beta}_n(s)$	Pressure exponents in blade row solutions
γ	Specific heat ratio
$\chi_n(s)$	Entropy and vorticity exponent in inter-blade row gap solutions
$\tilde{\chi}_n(s)$	Entropy exponent in blade row solutions
∂	Partial Derivative operator
δ	Variation operator
ρ	Density

θ	Circumferential angle
ξ	Stagger angle of blade row
ω	Frequency
Ω	Rotor speed
Ω_{vort}	Vorticity
$\mathbf{b}$	Actuator boundary condition matrix
$\mathbf{A}(s)$	Transmission matrix
$\mathbf{B}(x,s)$	Blade row solution matrix
$\mathbf{B}_L, \mathbf{B}_T$	Boundary condition matrices
$\mathbf{D}$	Boundary condition matrix
$\mathbf{J}$	Actuator boundary condition matrix
$\mathbf{N}(s)$	Inlet condition matrix
$\mathbf{P}$	Boundary condition matrix
$\mathbf{S}$	Sensor matrix
$\mathbf{V}(x,s)$	Inter-blade row gap solution matrix
$\mathbf{V}_L, \mathbf{V}_T$	Boundary condition matrices
$\mathbf{X}(s)$	Exit condition matrix

Subscripts:

a	Actuator parameter
k	Blade row and inter-blade row gap number
ka	Inter-blade row gap number of actuator location
ks	Inter-blade row gap number of sensor location
n	Fourier series index (harmonic number)
s	Sensor parameter
t	Total quantity
L,LE	Leading edge
T,TE	Trailing edge

Chapter 3

$\delta\alpha$	Flow deviation angle

Chapter 4

c	Free parameter in LQG problem
d, $\mathbf{d}$	Disturbance vector
e, $\mathbf{e}$	Error vector

q	Free parameter in LQG problem
v, w	Process and measurement noise
y, $\mathbf{y}$	Output variable
γ	H_∞ design parameter
$C(s)$	Complementary sensitivity function
$G(s)$	Plant transfer function
$K(s)$	Compensator transfer function
$S(s)$	Sensitivity function
$W_2(s)$	Complementary sensitivity weighting function
$W_1(s)$	Sensitivity weighting function
$\mathbf{A}$, $\mathbf{B}$, $\mathbf{N}$	State weightings in LQG problem
$\mathbf{F}$, $\mathbf{G}$, $\mathbf{H}$, $\mathbf{D}$, $\mathbf{P}$	State-space matrices
$\mathbf{L}$	Controllability Gramian
$\mathbf{Q}$, $\mathbf{R}$, $\mathbf{T}$	Noise intensities
$\mathbf{S}$, $\mathbf{P}$	Solutions to LQG control and observation Ricatti equations
$\mathbf{X}_\infty$, $\mathbf{Y}_\infty$	Solutions to H_∞ control and observation Ricatti equations

Superscripts:

H	Complex Conjugate Transpose

Chapter 1

Introduction

1.1 Background and Motivation

The operating range of axial compressors is limited at low mass flows by two types of aerodynamic instabilities: *rotating stall* and *surge*. Rotating stall is a non-axisymmetric disturbance which travels around the compressor annulus at about 30% to 70% of the rotor speed. When this phenomenon was first encountered, it was observed that there were regions, or *cells*, where the flow was separated from the compressor blade rows, hence the name 'rotating stall'. Surge, on the other hand, is an axisymmetric disturbance which not only interacts with the compression system, but also that of the plenum volume downstream. Surge is characterized by potentially damaging large amplitude oscillations at low frequency. In many instances, though, rotating stall has been identified as a precursor to surge in an engine. Moreover, blade fatigue considerations will not allow a compressor to operate for prolonged periods in a fully developed rotating stall mode.

Until recently, the research into these phenomena has been focused on the ability to predict the onset of rotating stall and surge and to characterize the inception and fully developed behavior. Within the last five years, there has been increased research on active control of compressor instabilities incorporating ever increasing complexities in the modeling. For example, early proof-of concept studies in surge control [1,2] assumed one dimensional flow through the compression system. Rotating stall control studies [3,4,5] however, required at least a two dimensional model. These rotating stall studies used a two dimensional incompressible flow model which characterized the low-speed experimental apparatus. As the technology of rotating stall and surge control advances, and as the experiments conducted to verify concepts become more sophisticated and more realistic, the compressor models must become more complex.

In high speed compressors, nonlinearity and compressibility must be addressed. Nonlinearity includes not only the nonlinear dynamics of rotating stall and surge, but the coupling between the two phenomena as well. Nonlinear compressible numerical models [6,7,8] have been used to study the inception and characteristics of rotating stall, but these have little use for active control. Nonlinear theoretical models have been developed [9,10,11,12,13] which may prove important to the stabilization efforts.

Nonlinear compressible surge models have been developed by Badmus et al [14] and Escuret and Elder [15], each of which was linearized to find a linear controller for the nonlinear system. Badmus et al applied their controller to a single stage research compressor, while Escuret and Elder simulated their linear controller with their nonlinear model. Both of these models use an axial spatial discretization across blade rows or across the whole compressor and used steady state or quasi-steady state information to find key relationships in the model. Badmus et al report some success at stabilizing surge near neutral stability, while Escuret and Elder report very small operating range extension in simulation. The results of the latter could be attributed to the linear controller which was implemented; Badmus et al used a control design method to achieve robustness to stable additive perturbations in the plant, while Escuret and Elder used a Linear Quadratic-Gaussian (LQG) controller. LQG optimal controllers do not provide explicit robustness properties unless the free parameters are 'tuned' to achieve them.

The only linearized two dimensional compressible theoretical model described to the author's knowledge is that of Bonnaure [16] and Hendricks et al [17]. This model describes an axially continuous two dimensional compressible model in the inter-blade row gaps and upstream and downstream ducts and a one dimensional compressible model within the blade rows to form the total compressor model. Experimentally or theoretically determined pressure losses and deviations are used to find key relationships in their model. This model is used as the starting point of this thesis.

This 2D compressible model predicts the existence of lightly damped 'higher frequency' eigenvalues of the compressor. These are sometimes referred to as 'acoustic' modes, although they may originate from sources such as static pressure delays, entropy and vorticity delays, or interactions of these two. These higher frequency eigenvalues may be considered analogous to higher frequency eigenvalues of structures. Frequently, these predicted modes in structures have uncertainty associated with their locations. This may also be the case with higher frequency modes in compressors, but there is little data as of yet. This thesis assumes that these higher frequency mode locations are well known.

The control methodologies employed in this thesis are: LQG optimal control, with some 'tuning' of the free parameters to meet specified frequency domain robustness properties, and H_∞ optimal control, where frequency domain specifications are used directly in the design procedure.

1.2 High Speed Multi-Stage Axial Compressors

The compressor is the first of three parts of a gas turbine engine. After air is compressed in the compressor, it is used for burning the propellant in the combustion chamber, and the exhaust is fed through the turbine to drive the compressor. An *axial* compressor differs from other compressors in the direction of airflow through the compressor; air flows axially through an annulus in the compressor, as opposed to radially, to achieve a pressure rise. A *high speed* compressor is one where the Mach number of the rotors is close to 1 and where compressibility effects are expected to contribute significantly to the dynamics of the compressor. There are two basic elements of an axial compressor: *stators*, which are stationary sets of blades around the annulus, used to change the circumferential velocity of the flow as well as diffuse the flow; and *rotors*, which are moving sets of blades, used to impart energy by accelerating the airflow (see Figure 1.1). A *stage* of a compressor is the combination of a rotor in front of a stator. Aircraft engines typically have between 8 and 20 stages. Frequently, there will be a set of *inlet guide vanes* (*IGV*s), which is a stator row in the front of the compressor to give the airflow an initial circumferential velocity before the first rotor. Other needed definitions are: the *hub* is the inner radius of the compressor annulus, the *casing* is the outer radius of the compressor annulus, and the *tip* is the outer radius of the rotor.

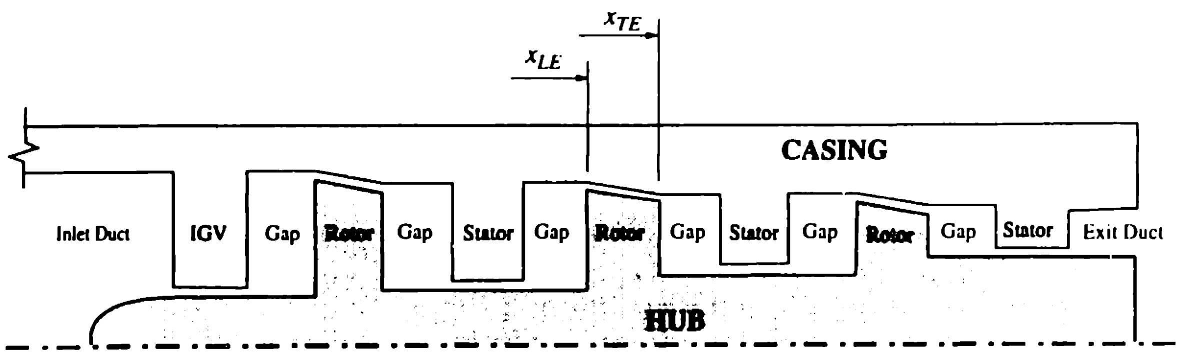

Figure 1.1 - Compressor diagram.

The operating condition of the compressor is determined by the corrected mass flow, or *flow coefficient*, which is a non-dimensional mass flow through the compressor. Compressor performance is characterized by a *speed line* or *characteristic*, which is the functional dependence of the pressure rise on the flow coefficient. A sample characteristic is shown in Figure 1.2. In multi-stage compressor models, each blade row has its own characteristic which can be combined to determine the overall characteristic. Generally speaking, the compressor is in the stable regime when the slope of the characteristic is negative, that is, when a decrease in flow coefficient leads to an increase in pressure rise. When the slope of the characteristic is positive, small deviations from the mean flow tend to grow into nonlinear rotating stall or surge.

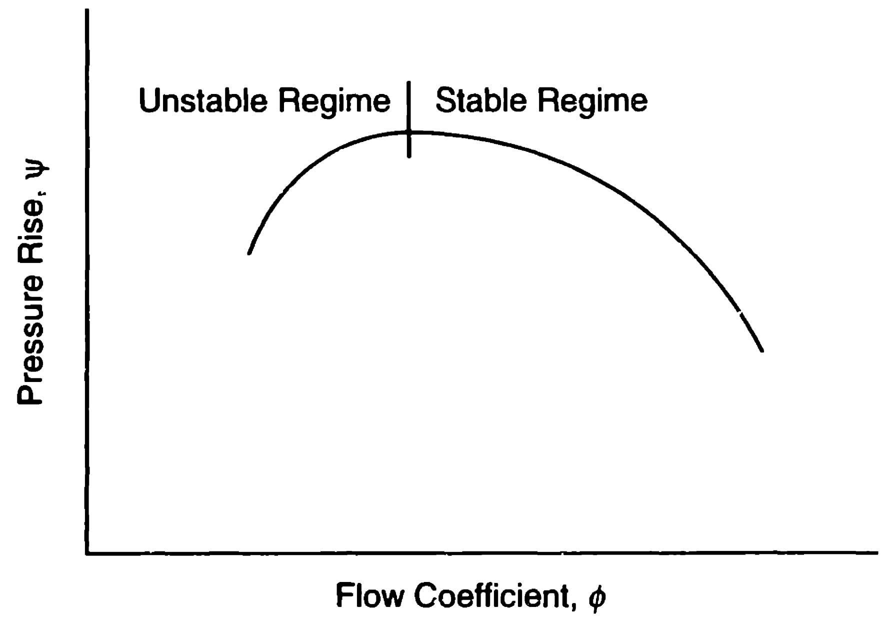

Figure 1.2 - Pressure rise characteristic.

1.3 Summary

The primary purpose of this research is twofold: to present a model of a high speed multi-stage axial compressor in input-output form, and to present control law configuration and design methods to extend the operating range of compressors.

The linear two-dimensional compressible model, developed by Bonnaure [16], is extended and recast into input-output form. Cancellation of branch cuts in the inter-blade row gap solutions is identified; this realization leads to the addition of more realistic end conditions necessary to cancel the branch cuts arising from the inlet and exit duct solutions. These new end conditions also result in a new eigenvalue problem. Similar to the branch cuts, singularities in the inter-blade row gap solutions are shown to cancel in the same manner. Actuator and sensor models are developed, which allow the solution to be recast into input-output form using the same technique that is used to formulate the eigenvalue problem. Another contribution to the modeling is the identification of higher frequency (higher than the rotor frequency) modes of the compressor.

The distributed model for each spatial Fourier coefficient is approximated by a finite dimensional state-space approximation. This is accomplished using Padé approximations of exponentials (time delays) and Taylor series expansions of transcendental functions. Two approximate models are developed: a model ignoring the dynamics of the inter-blade row gaps, and a model including the gap dynamics. The former has the advantage of having on the order of 40% fewer states than the latter, but is less accurate. Long inlet and exit ducts are not easily modeled using this method due to the very high order approximations necessary.

Frequency domain specifications and cost functions on the state and control variables are developed to compare control configurations (i.e. sensor and actuator placement and type). A bandwidth constraint is used to provide robustness to high frequency errors, and a sensitivity constraint is used to provide specified gain and phase margins. A state cost is defined to measure how well a compensator can keep the static pressure perturbations small, while a control cost is defined to measure the amount of control activity necessary to achieve the state cost.

The Linear Quadratic-Gaussian (LQG) and H_∞ design procedures are reviewed for use in control design and configuration comparison. In the LQG design method, colored measurement noise is used to penalize bandwidths greater than the specified constraint, and other free parameters are chosen to provide a good trade-off between bandwidth and low sensitivity. This design also results in an 'LQG' cost function which is used in addition to the state and control costs to compare control configurations. The H_∞ design method uses the frequency domain specifications directly in the design procedure to find which configurations can meet the specifications at the lowest flow coefficients. This information is used with the state and control cost functions to find the 'best' control configuration.

1.4 Structure of Thesis

A review of the two dimensional (2D) compressible model, the *truth* model, is presented in Chapter 2. The truth model is that of Bonnaure [16], with some modifications, reformulated into a frequency (Laplace transform) domain input-output representation. To apply modern control theory to the truth model, the transcendental functions of the truth model must be approximated by rational polynomials in the Laplace variable, s. The approximation method is presented in Chapter 3 with a section analyzing the accuracy of the approximate model for use in stability analysis and in control law design. The approximate model of Chapter 3 is then used in modern control design methodologies. Chapter 4 presents desired frequency domain specifications on closed loop transfer functions of a compensator as well as cost functions which can be used to compare different control configurations. Also presented are two design methodologies which will be used to compare sensor and actuator configurations and to design a compensator to meet the frequency domain specifications. These control design methodologies are used in Chapter 5 to find the 'best' control configuration in terms of the cost functions and operating range extension which also meet the frequency domain specifications. Chapter 6 presents a summary, conclusions and recommendations for further research.

Chapter 2

Fluid Dynamic Model and Analytical Solution

This chapter begins by presenting the motivation for finding the input-output form of the model in Section 2.1. In Section 2.2, a review of the 2D compressible model described by Bonnaure [16] is presented with some modification. The primary modification is the use of new end conditions that model finite length ducts, which are believed to be more realistic than the original end conditions. A new eigenvalue equation is derived which is free of some of the mathematical difficulties of the original. Section 2.3 presents additional modifications to the model, namely, the addition of an actuator model and a sensor model. These are applied to the same procedure used to find the eigenvalue equation to yield the desired input-output form. Section 2.3 comments on some of the general and specific results of the model.

2.1 Motivation for Input-Output Form

The primary application of the original 2D compressible model eigenvalue equation, and that derived in Section 2.2, is stability analysis. It can be used to study the effects of various design parameters on compressor operating range and stall inception behavior [17]. In its current form, however, it has no provision to analyze effects of feedback control. A few modifications are therefore necessary.

Both classical and modern control theory rely upon a description of the open-loop transfer function from an actuator to a sensor, either in the frequency domain, or in state-space form. Using this description, feedback control laws can be designed and evaluated by 'closing the loop'. Stability, performance, and robustness of the closed-loop system can then be evaluated in a number of ways. Section 2.3 will present modifications and the technique to find the input-output form of the 2D compressible model.

2.2 Solution of Partial Differential Equations

This section presents the solution of the original [16] linear, time and space invariant partial differential equations (PDEs). There are a few differences here from the original. The first is the minor change of taking a Laplace transform in time instead of assuming the form of a complex exponential, $e^{j\omega t}$. This is the standard method to produce transfer functions from linear, time invariant ordinary differential equations (ODEs), which also applies to PDEs. Additional steps are also included to aid a person with only a control systems background to understand the usage of the complex Fourier series and to understand each part of the solution. New notation is introduced for organization when the system is approximated. New end conditions, which are considered more realistic than the original, are proposed and derived to eliminate difficulties associated with the original end conditions. Finally, a new eigenvalue equation is derived using the new end conditions.

2.2.1 Inter-Blade Row Gap and Duct Equations

Presented first are the original time and space invariant PDEs describing the two dimensional (2D) fluid flow in an annulus. The two parts of the compressor, where this model is assumed to be valid, are in the annular space between the blade rows, which will be referred to as *inter-blade row gaps* or *gaps*, and the annular space upstream and downstream of the compressor in the ductwork, which will be referred to as *ducts* (see Figure 2.1). These equations apply separately to each gap and duct, with all the constant mean flow variables dependent on the particular gap or duct to be modeled. The index indicating the gap or duct number, k, (see Figure 2.1) is omitted in most places wherever

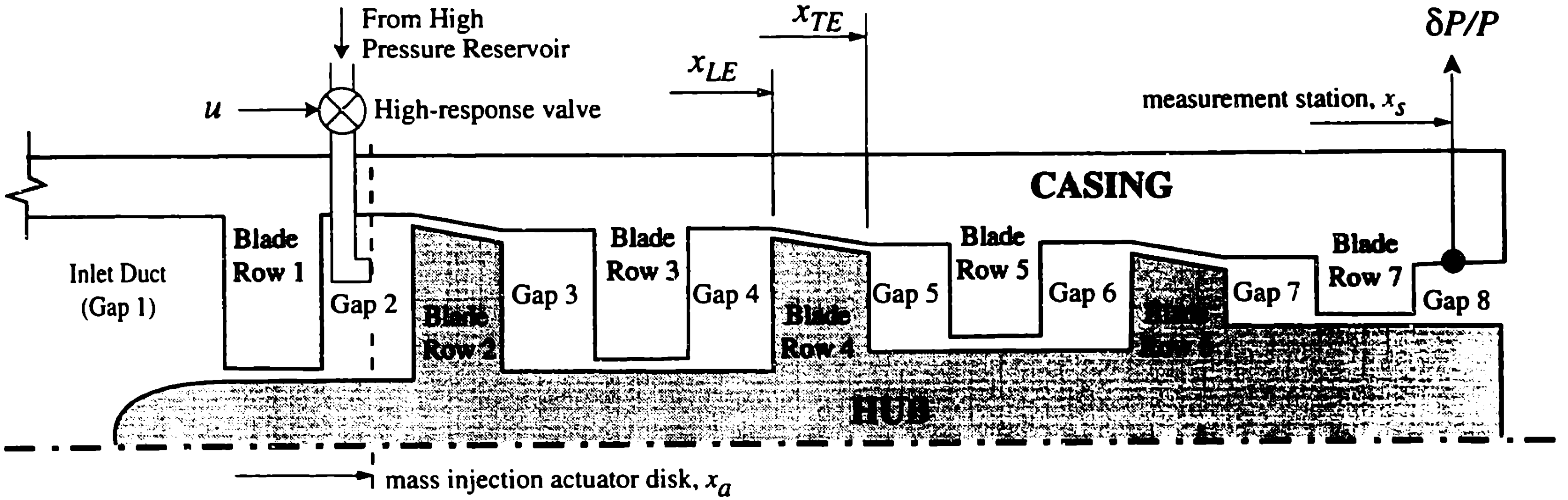

Figure 2.1 – Compressor Diagram

it is understood by context that the equations apply to each gap or duct separately. The four PDEs are

Mass Continuity Equation

$$\frac{\partial \delta\rho}{\partial t} + \rho \frac{\partial \delta V_x}{\partial x} + V_x \frac{\partial \delta\rho}{\partial x} + \frac{1}{r}\left(\rho \frac{\partial \delta V_\theta}{\partial \theta} + V_\theta \frac{\partial \delta\rho}{\partial \theta}\right) = 0 \qquad (2.1)$$

Axial Momentum Equation

$$\rho \frac{\partial \delta V_x}{\partial t} + \rho V_x \frac{\partial \delta V_x}{\partial x} + \frac{\rho V_\theta}{r} \frac{\partial \delta V_x}{\partial \theta} = -\frac{\partial \delta P}{\partial x} \qquad (2.2)$$

Circumferential Momentum Equation

$$\rho \frac{\partial \delta V_\theta}{\partial t} + \rho V_x \frac{\partial \delta V_\theta}{\partial x} + \frac{\rho V_\theta}{r} \frac{\partial \delta V_\theta}{\partial \theta} = -\frac{1}{r} \frac{\partial \delta P}{\partial \theta} \qquad (2.3)$$

Energy Equation (for a perfect gas)

$$\frac{\partial \delta P}{\partial t} + V_x \frac{\partial \delta P}{\partial x} + \frac{V_\theta}{r} \frac{\partial \delta P}{\partial \theta} = a^2\left(\frac{\partial \delta\rho}{\partial t} + V_x \frac{\partial \delta\rho}{\partial x} + \frac{V_\theta}{r} \frac{\partial \delta\rho}{\partial \theta}\right) \qquad (2.4)$$

where the dependent variables $(\delta P, \delta\rho, \delta V_x, \delta V_\theta)$ are the perturbations of static pressure, density, axial velocity and circumferential velocity, respectively. The independent variables, (x, θ, t) are axial distance, circumferential angle, and time, respectively. All other terms in the PDEs are assumed constant in the annulus.

A standard method of solving PDEs is to use a transform or series expansion to eliminate derivatives with respect to independent variables. The result is then either an algebraic expression, or an ordinary differential equation. Equations 2.1-2.4 have three independent variables, so a Laplace transform will be applied to the time variable and a complex Fourier series will be applied to the circumferential angle. What remains will be an ODE in the axial coordinate. First, however, the dependent variables must be eliminated.

The PDEs are first solved for static pressure by differentiating Equation 2.2 with respect to x, differentiating Equation 2.3 with respect to θ and dividing by r, adding the results, using Equation 2.1 to eliminate the velocity differentials by substitution, and using Equation 2.4 to eliminate the density by substitution to obtain

$$\frac{\partial^2 \delta P}{\partial x^2}+\frac{1}{r^2}\frac{\partial^2 \delta P}{\partial \theta^2}=\frac{1}{a^2}\left(\frac{\partial}{\partial t}+V_x\frac{\partial}{\partial x}+\frac{V_\theta}{r}\frac{\partial}{\partial \theta}\right)^2 \delta P \tag{2.5}$$

This is a two dimensional wave equation in Cartesian coordinates, $x = x, y = r\theta$, with the fluid medium moving relative to the reference frame. Taking the Laplace transform in time yields

$$\frac{\partial^2 \delta P}{\partial x^2}+\frac{1}{r^2}\frac{\partial^2 \delta P}{\partial \theta^2}=\frac{1}{a^2}\left(s+V_x\frac{\partial}{\partial x}+\frac{V_\theta}{r}\frac{\partial}{\partial \theta}\right)^2 \delta P \tag{2.6}$$

where $\delta P = \delta P(x,\theta,s)$. This can be rearranged by expanding the RHS and collecting terms:

$$\begin{aligned}&\left(1-M_x^2\right)\frac{\partial^2 \delta P}{\partial x^2}-2M_x\left(\frac{s}{a}+\frac{M_\theta}{r}\frac{\partial}{\partial \theta}\right)\frac{\partial \delta P}{\partial x}+\\&\qquad\left(-\frac{s^2}{a^2}\delta P+\left(1-M_\theta^2\right)\frac{1}{r^2}\frac{\partial^2 \delta P}{\partial \theta^2}-2M_\theta\frac{s}{ra}\frac{\partial \delta P}{\partial \theta}\right)=0\end{aligned} \tag{2.7}$$

where M_x and M_θ are the Mach numbers (velocity divided by speed of sound) in the axial and circumferential directions, respectively. Assuming the static pressure has the form of a Fourier series in the circumferential direction:

$$\delta P(x,\theta,s)=\sum_{n=0}^{\infty}\delta P_s(x,n,s)\sin n\theta+\delta P_c(x,n,s)\cos n\theta, \tag{2.8}$$

Equation 2.7 results in

$$\sum_{n=-\infty}^{\infty}\left\{\begin{bmatrix}\left(1-M_x^2\right)\dfrac{\partial^2\delta P_s}{\partial x^2}-2M_x\left(\dfrac{s}{a}\dfrac{\partial\delta P_s}{\partial x}-M_\theta\dfrac{n}{r}\dfrac{\partial\delta P_c}{\partial x}\right)+\\ \left(-\dfrac{s^2}{a^2}\delta P_s-\left(1-M_\theta^2\right)\dfrac{n^2}{r^2}\delta P_s+2M_\theta\dfrac{n}{r}\dfrac{s}{a}\delta P_c\right)\end{bmatrix}\sin n\theta+\begin{bmatrix}\left(1-M_x^2\right)\dfrac{\partial^2\delta P_c}{\partial x^2}-2M_x\left(\dfrac{s}{a}\dfrac{\partial\delta P_c}{\partial x}+M_\theta\dfrac{n}{r}\dfrac{\partial\delta P_s}{\partial x}\right)+\\ \left(-\dfrac{s^2}{a^2}\delta P_c+\left(1-M_\theta^2\right)\dfrac{n^2}{r^2}\delta P_c-2M_\theta\dfrac{n}{r}\dfrac{s}{a}\delta P_s\right)\end{bmatrix}\cos n\theta\right\}=0. \quad (2.9)$$

For the sum to equal zero, each coefficient of the sines and cosines must be identically equal to zero. The equations for the sine and cosine coefficients can be reduced to a single equation if a new variable is defined as

$$\delta P_z=\delta P_c+j\delta P_s. \quad (2.10)$$

The result is one complex constant coefficient ODE in x for each *harmonic* (i.e. term of the Fourier series):

$$\begin{aligned}&\left(1-M_x^2\right)\frac{\partial^2\delta P_z}{\partial x^2}-2M_x\left(\frac{s}{a}-jM_\theta\frac{n}{r}\right)\frac{\partial\delta P_z}{\partial x}+\\ &\qquad\left(-\frac{n^2}{r^2}-\left(\frac{s}{a}-jM_\theta\frac{n}{r}\right)^2\right)\delta P_z=0\end{aligned} \quad (2.11)$$

There are still two states of interest for each harmonic, the sine and cosine coefficients, but they have been lumped into one complex variable for convenience. This same ODE could have been obtained using the complex Fourier series:

$$\delta P(x,\theta,s) = \sum_{n=-\infty}^{\infty} \delta P(x,n,s) e^{jn\theta} \tag{2.12}$$

in Equation 2.7. The only difference is that the ODE would have n replaced by $-n$. The ODE in x (Equation 2.11 with n replaced by $-n$) becomes

$$\begin{gathered}\left(1-M_x^2\right)\frac{\partial^2 \delta P}{\partial x^2} - 2M_x\left(\frac{s}{a} + jM_\theta \frac{n}{r}\right)\frac{\partial \delta P}{\partial x} + \\ \left(-\frac{n^2}{r^2} - \left(\frac{s}{a} + jM_\theta \frac{n}{r}\right)^2\right)\delta P = 0\end{gathered} \tag{2.13}$$

where $\delta P = \delta P(x,n,s)$.

To understand the dynamics of Equation 2.13, it is instructive to look first at the case where the mean velocities are zero. Equation 2.13 becomes

$$\frac{\partial^2 \delta P}{\partial x^2} - \left(\frac{n^2}{r^2} + \frac{s^2}{a^2}\right)\delta P = 0 \tag{2.14}$$

This can be easily solved by assuming the form of exponentials in x and solving for the exponential constants:

$$\delta P(x,n,s) = B_n(s) e^{x\sqrt{\frac{n^2}{r^2}+\frac{s^2}{a^2}}} + C_n(s) e^{-x\sqrt{\frac{n^2}{r^2}+\frac{s^2}{a^2}}}. \tag{2.15}$$

Examining this equation, we see that there is a cut-on frequency, $|\omega| = na/r$, above which the character of the solution changes. The solution consists of exponentials below the cut-on frequency, and sinusoids above the cut-on frequency. The exponential solutions are analogous to the axially decaying solutions of the incompressible static pressure solution. The sinusoidal solutions are non-decaying propagating waves in the axial direction. The two exponents are not continuous in the frequency domain, however; they represent two Riemann sheets of the solution separated by branch cuts arising from the square root of the frequency parameter, s. If the boundary conditions are stated in terms of the PDE dependent variables, then these branch cuts do not appear either in the eigenvalue equation or in the open-loop transfer functions.

The corresponding solution to Equation 2.13 with non-zero velocities is

$$\delta P(x,n,s) = B_n(s)e^{\alpha_n(s)x} + C_n(s)e^{\beta_n(s)x}$$

where the exponents are

$$\alpha_n,\beta_n(s) = \frac{M_x\left(\frac{sr}{a} + jnM_\theta\right) \pm \sqrt{n^2\left(1 - M_x^2\right) + \left(\frac{sr}{a} + jnM_\theta\right)^2}}{r\left(1 - M_x^2\right)} \tag{2.16}$$

with $\alpha_n(s)$ taking the positive sign. Again, branch cuts appear in the case with the moving fluid relative to the reference frame.

The solution of Equation 2.13 will be modified by changing the definition of the frequency and harmonic dependent coefficients to allow for simpler solutions to the rest of the dependent variables, shown in Equation 2.17 below. Specifically, if the coefficients were not modified, the velocity perturbation solutions would have denominator functions of s. There is no change in the final eigenvalue equations or transfer functions due to this modification, but it makes the solutions easier to manipulate. The solutions for density and velocity perturbations are easily found, using Equation 2.17, by solving Equations 2.4, 2.2 and 2.3, respectively. The solution is

$$\frac{\delta P}{P}(x,\theta,s)=\sum_{n=-\infty}^{\infty}\gamma\begin{bmatrix}\left(r\alpha_n(s)M_x+\left(\frac{sr}{a}+jnM_\theta\right)\right)B_n(s)e^{\alpha_n(s)x}+\\ \left(r\beta_n(s)M_x+\left(\frac{sr}{a}+jnM_\theta\right)\right)C_n(s)e^{\beta_n(s)x}\end{bmatrix}e^{jn\theta} \tag{2.17}$$

$$\frac{\delta\rho}{\rho}(x,\theta,s)=\sum_{n=-\infty}^{\infty}\begin{bmatrix}\left(r\alpha_n(s)M_x+\left(\frac{sr}{a}+jnM_\theta\right)\right)B_n(s)e^{\alpha_n(s)x}+\\ \left(r\beta_n(s)M_x+\left(\frac{sr}{a}+jnM_\theta\right)\right)C_n(s)e^{\beta_n(s)x}+E_n(s)e^{\chi_n(s)x}\end{bmatrix}e^{jn\theta} \tag{2.18}$$

$$\frac{\delta V_x}{a}(x,\theta,s)=\sum_{n=-\infty}^{\infty}\begin{bmatrix}-r\alpha_n(s)B_n(s)e^{\alpha_n(s)x}-r\beta_n(s)C_n(s)e^{\beta_n(s)x}+\\ jnM_xD_n(s)e^{\chi_n(s)x}\end{bmatrix}e^{jn\theta} \tag{2.19}$$

$$\frac{\delta V_\theta}{a}(x,\theta,s)=\sum_{n=-\infty}^{\infty}\begin{bmatrix}-jnB_n(s)e^{\alpha_n(s)x}-jnC_n(s)e^{\beta_n(s)x}+\\ \left(\frac{sr}{a}+jnM_\theta\right)D_n(s)e^{\chi_n(s)x}\end{bmatrix}e^{jn\theta} \tag{2.20}$$

where

$$\chi_n(s)=-\frac{\frac{sr}{a}+jnM_\theta}{rM_x}.$$

When n=0, the zeroth harmonic, Equations 2.17-2.20 have a special form, similar to the blade row equation solution in the next section:

$$\frac{\delta P}{P}(x,\theta,s)=\sum_{n=-\infty}^{\infty}\gamma\left[\tfrac{1}{1-M_x}B_n(s)e^{\alpha_n(s)x}+\tfrac{1}{1+M_x}C_n(s)e^{\beta_n(s)x}\right]e^{jn\theta} \tag{2.21}$$

$$\frac{\delta\rho}{\rho}(x,\theta,s)=\sum_{n=-\infty}^{\infty}\left[\tfrac{1}{1-M_x}B_n(s)e^{\alpha_n(s)x}+\tfrac{1}{1+M_x}C_n(s)e^{\beta_n(s)x}+E_n(s)e^{\chi_n(s)x}\right]e^{jn\theta} \tag{2.22}$$

$$\frac{\delta V_x}{a}(x,\theta,s)=\sum_{n=-\infty}^{\infty}\left[-\tfrac{1}{1-M_x}B_n(s)e^{\alpha_n(s)x}+\tfrac{1}{1+M_x}C_n(s)e^{\beta_n(s)x}\right]e^{jn\theta} \tag{2.23}$$

$$\frac{\delta V_\theta}{a}(x,\theta,s)=\sum_{n=-\infty}^{\infty}D_n(s)e^{\chi_n(s)x}e^{jn\theta} \tag{2.24}$$

Equations 2.17-2.20 or 2.21-2.24 can be written in matrix form as

$$\begin{bmatrix} \frac{\delta P}{P} \\ \frac{\delta \rho}{\rho} \\ \frac{\delta V_x}{a} \\ \frac{\delta V_\theta}{a} \end{bmatrix}(x,\theta,s) = \sum_{n=-\infty}^{\infty} e^{in\theta}\mathbf{V}_n(x,s)\begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_n(s) \tag{2.25}$$

where $\mathbf{V}_n(x,s)$ contains the axial distance, frequency and harmonic dependent terms of the solution multiplying the coefficients. This condensed form of the solution will get much use throughout this chapter and the next.

2.2.2 Blade Row Equations

Presented here are the time and space invariant PDEs describing the one dimensional (1D) fluid flow in the blade row directly from Bonnaure [16]. The fluid is constrained to one dimension by the individual blades and the annulus' inner and outer walls (i.e. the hub and casing). These equations apply separately to each blade row, with all the constant mean flow variables dependent on the particular blade row to be modeled. The index indicating the blade row number, k, (see Figure 2.1) is omitted wherever it is understood by context that the equations apply to each blade row separately. The three PDEs are:

Mass Continuity Equation

$$\frac{\partial \delta\rho}{\partial t} + W\frac{\partial \delta\rho}{\partial x'} = -\rho\frac{\partial \delta W}{\partial x'} \tag{2.26}$$

Momentum Equation

$$\frac{\partial \delta W}{\partial t} + W\frac{\partial \delta W}{\partial x'} = -\frac{1}{\rho}\frac{\partial \delta P}{\partial x'} \tag{2.27}$$

Energy Equation (for a perfect gas)

$$\frac{\partial \delta P}{\partial t} + W\frac{\partial \delta P}{\partial x'} = a^2\left(\frac{\partial \delta \rho}{\partial t} + W\frac{\partial \delta \rho}{\partial x'}\right) \tag{2.28}$$

where ξ is the blade stagger angle, $x' = x / \cos\xi$ is the distance along a particular blade row, the dependent variables $(\delta P, \delta\rho, \delta W)$ are the static pressure, density and velocity perturbations, respectively, and the independent variables are again (x, θ, t). The circumferential angle does not appear in these equations since perturbations are constrained to move along the blades, but the solution can still be assumed to be in the form of a Fourier series. The only implication is that the harmonic number, n, does not affect the dynamics. These three equations can be manipulated like the inter-blade row gap equations to yield

$$(\frac{\partial}{\partial t} + W\frac{\partial}{\partial x'})^2 \delta P = a^2 \frac{\partial^2 \delta P}{\partial x'^2} \tag{2.29}$$

This is a one dimensional wave equation along the length of the blades with the fluid medium moving with respect to the reference frame. This can also be solved with the Laplace transform and the complex Fourier series, yielding the total solution:

$$\frac{\delta P}{P}(x,\theta,s) = \gamma \sum_{n=-\infty}^{\infty}\left[\tilde{B}_n(s)e^{\tilde{\alpha}_n(s)x} + \tilde{C}_n(s)e^{\tilde{\beta}_n(s)x}\right]e^{jn\theta} \tag{2.30}$$

$$\frac{\delta \rho}{\rho}(x,\theta,s) = \sum_{n=-\infty}^{\infty}\left[\tilde{B}_n(s)e^{\tilde{\alpha}_n(s)x} + \tilde{C}_n(s)e^{\tilde{\beta}_n(s)x} + \tilde{E}_n(s)e^{\tilde{\chi}_n(s)x}\right]e^{jn\theta} \tag{2.31}$$

$$\frac{\delta W}{a}(x,\theta,s) = \sum_{n=-\infty}^{\infty}\left[-\tilde{B}_n(s)e^{\tilde{\alpha}_n(s)x} + \tilde{C}_n(s)e^{\tilde{\beta}_n(s)x}\right]e^{jn\theta} \tag{2.32}$$

where

$$\tilde{\alpha}_n(s) = \frac{1}{\cos\xi}\left(-\frac{jn}{r}\sin\xi + \frac{s + jn\Omega}{a - W}\right)$$
$$\tilde{\beta}_n(s) = \frac{1}{\cos\xi}\left(-\frac{jn}{r}\sin\xi - \frac{s + jn\Omega}{a + W}\right)$$
$$\tilde{\chi}_n(s) = \frac{1}{\cos\xi}\left(-\frac{jn}{r}\sin\xi - \frac{s + jn\Omega}{W}\right)$$

As was stated previously, the harmonic number does not affect the dynamics of the blade row solution. It *does*, however, affect the transformation between the reference frame of the blade row and the compressor reference frame. The two effects are due to the stagger of the blade rows and the rotation of the rotors. For example, if the flow is rotated with respect to the compressor an amount $\Delta\theta$, the phase of a harmonic is changed by $n\Delta\theta$. Equations 2.30, 2.31 and 2.32 can be written in matrix form as

$$\begin{bmatrix} \frac{\delta P}{P} \\ \frac{\delta\rho}{\rho} \\ \frac{\delta W}{a} \end{bmatrix}(x,\theta,s) = \sum_{n=-\infty}^{\infty} e^{jn\theta}\mathbf{B}_n(x,s)\begin{bmatrix} \tilde{B} \\ \tilde{C} \\ \tilde{E} \end{bmatrix}_n(s) \tag{2.33}$$

This form of the solution will get much use throughout this chapter and the next.

Through the rest of the thesis, the subscript n will be dropped with the understanding that each equation will apply separately to all harmonics. The only exception to this will be separate end conditions for n=0.

2.2.3 Boundary Conditions

At each interface between a blade row leading or trailing edge and a gap or duct, there must be an appropriate number of boundary conditions to connect the system. The total number of boundary conditions plus end conditions must equal the total number of unknowns in the system, namely, the coefficients in Equations 2.25 and 2.33. For each combination of blade row and adjacent gap, then, there should be seven boundary conditions for the seven unknowns of the two solutions. However, the total number of unknowns for a compressor with K blade rows and K+1 gaps (see Figure 2.1) is 7*K+4, so there must be a total of four end conditions.

2.2.3.1 Leading Edge Conditions:

There are three leading edge boundary conditions: mass continuity, total temperature conservation, and a total pressure loss. These conditions are described in Appendix A, along with the definitions of all the new matrices. The matrix notation for the leading edge boundary condition is

$$(\mathbf{V}_{Lk} + \tfrac{1}{1+s\tau}\mathbf{P}_k)\mathbf{V}_k(x_{LEk},s)\begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_k = \mathbf{B}_{Lk}\mathbf{B}_k(x_{LEk},s)\begin{bmatrix} \tilde{B} \\ \tilde{C} \\ \tilde{E} \end{bmatrix}_k \qquad (2.34)$$

This allows us to solve for the kth blade row unknowns in terms of the preceding gap or duct unknowns, gap k.

2.2.3.2 Trailing Edge Conditions:

The other four boundary conditions for each blade row are the trailing edge boundary conditions: mass continuity, total temperature conservation, total pressure conservation and a flow angle deviation. These conditions are described in Appendix A, along with the definitions of all the new matrices. The matrix notation for the trailing edge boundary condition is

$$\mathbf{B}_{Tk}\mathbf{B}_k(x_{TEk},s)\begin{bmatrix} \tilde{B} \\ \tilde{C} \\ \tilde{E} \end{bmatrix}_k = \mathbf{V}_{Tk+1}\mathbf{V}_{k+1}(x_{TEk},s)\begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_{k+1} + \tfrac{1}{1+s\tau}\mathbf{D}_k\mathbf{V}_k(x_{LEk},s)\begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_k \qquad (2.35)$$

This allows us to solve for the gap k+1 unknowns in terms of the blade k unknowns *as well as* the gap k unknowns.

2.2.4 Stacking Method

Now that the boundary conditions have been defined for each leading and trailing edge, these can be combined into a transmission matrix across the entire blade row, from the gap k unknowns to the gap k+1 unknowns.

Combining Equations 2.34 and 2.35, the transmission matrix across a blade row can be written as

$$\begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_{k+1} = \mathbf{V}_{k+1}^{-1}(x_{TEk},s)\mathbf{V}_{Tk+1}^{-1}\begin{pmatrix} \mathbf{B}_{Tk}\mathbf{B}_k(x_{TEk},s)\mathbf{B}_k^{-1}(x_{LEk},s)\mathbf{B}_{Lk}^{-1}\left(\mathbf{V}_{Lk}+\frac{1}{1+s\tau}\mathbf{P}_k\right) \\ -\frac{1}{1+s\tau}\mathbf{D}_k \end{pmatrix}\mathbf{V}_k(x_{LEk},s)\begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_k$$
$$= \mathbf{A}_k(s)\begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_k \qquad (2.36)$$

Using Equation 2.36, the total transmission matrix relating the inlet to the exit gap unknowns can be found by multiplying, or "stacking", successive stages:

$$\begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_{K+1} = \mathbf{A}_K(s)\mathbf{A}_{K-1}(s)\cdots\mathbf{A}_2(s)\mathbf{A}_1(s)\begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_1 = \mathbf{A}(s)\begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_1 \qquad (2.37)$$

2.2.5 End Conditions

As was stated previously, there must be four end conditions to properly specify the system. Due to physical reasons beyond the scope of this thesis, the four end conditions will be divided into one exit condition and three inlet conditions. In Bonnaure [16], the assumption of infinite length ducts was used. This works well for low speed, incompressible theory but, unfortunately, causes mathematical difficulties in the compressible case. Specifically, branch cuts in the Laplace domain are introduced into the problem which cannot be dealt with easily in standard control theory. This was the motivation for finding more realistic end conditions which eliminate the problem. Indeed, modeling finite length ducts with appropriate end conditions removes this complication.

2.2.5.1 Inlet Conditions:

The inlet will be treated in this thesis as an open end having clean and smooth flow. This translates into variations in total pressure, entropy and vorticity equal to zero, or:

$$\begin{bmatrix} \delta P_t \\ \delta S_{entr} \\ \delta \Omega_{vort} \end{bmatrix} = 0 \tag{2.38}$$

applied at the interface of the free stream and the inlet duct. The specifics of the end condition are derived in Appendix A, and the following condensed form will be used:

$$\mathbf{N}(s) \begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_1 = 0 \tag{2.39}$$

The matrix $\mathbf{N}(s)$ is 3x4.

2.2.5.2 Exit Conditions:

The exit condition model depends on what is downstream of the compressor.. This thesis will assume that the flow dumps into a plenum, so a boundary condition modeling this will be used throughout. The open end condition for non-zeroth harmonics is:

$$\frac{\delta P}{P} = 0 \tag{2.40}$$

applied at the interface of the exit duct and the plenum. In the condensed notation of Equation 2.25, this can be written:

$$\begin{bmatrix}1 & 0 & 0 & 0\end{bmatrix}\mathbf{V}_{K+1}(x_{pled}, s)\begin{bmatrix}B\\C\\D\\E\end{bmatrix}_{K+1} = 0 \tag{2.41}$$

where x_{pled} is the location of the plenum/exit duct interface. The modeling of the exit for zeroth harmonic perturbations includes the open end condition plus the following equation modeling the plenum dynamics (modified Equation 3.60 from Bonnaure):

$$\left(\frac{V_{plen}P_{plen}}{a^2\dot{m}_{ex}}s + \frac{\gamma+1}{2\gamma}\right)\frac{\delta P}{P} - \frac{\delta\rho}{\rho} - \frac{\delta V_x}{V_x} = 0 \tag{2.42}$$

where V_{plen} is the plenum volume. In the condensed notation of Equation 2.25, this can be written:

$$\left[\left(\frac{V_{plen}P_{plen}}{a^2\dot{m}_{ex}}s+\frac{\gamma+1}{2\gamma}\right) \quad -1 \quad -1 \quad 0\right]\mathbf{V}_{K+1}(x_{pled},s)\begin{bmatrix}B\\C\\D\\E\end{bmatrix}_{K+1}=0. \tag{2.43}$$

Equations 2.41 and 2.43 will both be condensed into the general matrix form:

$$\mathbf{X}(s)\begin{bmatrix}B\\C\\D\\E\end{bmatrix}_{K+1}=0 \tag{2.44}$$

where the matrix $\mathbf{X}(s)$ is 1x4.

2.2.6 Eigenvalue Equation

An eigenvalue equation is a function of the Laplace variable, s, that ,when solved, gives the unforced natural frequency modes of the system. One use for the eigenvalue equation is to determine whether or not a system is stable. This was the primary use of the 2D compressible model used by Bonnaure [16] and by Hendricks et al [17]. The eigenvalue equation is also important with respect to open-loop transfer functions, because it will always appear in the denominator, causing eigenvalues to appear as poles in the transfer functions.

To review Bonnaure's eigenvalue equation, the infinite duct assumption was translated into the following equation:

$$\begin{bmatrix}0\\C\\D\\E\end{bmatrix}_{K+1}=\mathbf{A}(s)\begin{bmatrix}B\\0\\0\\0\end{bmatrix}_{1} \tag{2.45}$$

or

$$\mathbf{A}_{(1,1)}(s) = 0 \tag{2.46}$$

where one end condition has been applied at the exit, gap $K+1$, and three conditions have been applied at the inlet, gap 1.

A new eigenvalue problem, using the more realistic end conditions, can be formulated. Combining the end conditions, Equations 2.39 and 2.44, with the total transmission matrix, Equation 2.37, we get

$$\begin{bmatrix} \mathbf{X}(s)\mathbf{A}(s) \\ \mathbf{N}(s) \end{bmatrix} \begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \tag{2.47}$$

or, using the dependent variables of the PDEs in the inlet duct as the unknowns:

$$\begin{bmatrix} \mathbf{X}(s)\mathbf{A}(s) \\ \mathbf{N}(s) \end{bmatrix} \mathbf{V}_1^{-1}(x_{in}, s) \begin{bmatrix} \frac{\delta P}{P} \\ \frac{\delta \rho}{\rho} \\ \frac{\delta V_x}{a} \\ \frac{\delta V_\theta}{a} \end{bmatrix}_{1, x_{in}} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{2.48}$$

The new eigenvalue equation can be stated as

$$\det \begin{vmatrix} \mathbf{X}(s)\mathbf{A}(s)\mathbf{V}_1^{-1}(x_{in}, s) \\ \mathbf{N}(s)\mathbf{V}_1^{-1}(x_{in}, s) \end{vmatrix} = 0 \tag{2.49}$$

where changing the unknowns to the dependent variables serves to eliminate the branch cuts in the eigenvalue equation. There are two singularities in the eigenvalue equation, one in the right half plane (RHP) and one in the left half plane (LHP), due to the vorticity inlet condition (Appendix A). These appear because the vorticity condition is a derivative condition and can not be expressed as a constant linear combination of the dependent variables of the original PDEs (Equations 2.1-2.4). However, these singularities in the eigenvalue equation do not appear at all when transfer functions are formed.

2.2.7 Discussion

It is important to note that the eigenvalue equation (2.49) is a general representation which applies for all harmonics, n. In other words, changing the value of n will yield different solutions.

In the definition of the transmission matrix, Equation 2.36, we see the inverse of the matrix $\mathbf{V}(x_{LE},s)$. The determinant of this matrix is a function of s, and indeed has four roots: two on the imaginary axis of the s domain at the branch points of $\alpha(s)$ and $\beta(s)$, one in the RHP of the s-domain, and one in the LHP of the s-domain. This matrix inverse $\mathbf{V}^{-1}(x_{TE},s)$ appears separately in the transmission matrix of each blade row, so it was expected that all of the roots would show up as singularities in the original eigenvalue equation, Equation 2.46, since they are in the denominator. When the eigenvalue equation was examined, though, only one singularity in the LHP appeared. The lack of multiple singularities is due to the matrix product $\mathbf{V}(x_{LE},s)\mathbf{V}^{-1}(x_{TE},s)$, which appears when two adjacent transmission matrices are multiplied (Equation 2.37). Forming this matrix product cancels the singularities analytically. Appendix B expands the matrix product $\mathbf{V}(x_{LE},s)\mathbf{V}^{-1}(x_{TE},s)$, and Section 3.3.1 will show how these singularities cancel out analytically in the matrix product. Therefore, the only remaining singularities in the *original* eigenvalue equation are due to the downstream exit duct, where in the original eigenvalue equation, the matrix $\mathbf{V}^{-1}(x_{TE},s)$ of $\mathbf{A}_K(s)$ was not multiplied by another $\mathbf{V}(x_{LE},s)$ matrix.

When the *new* end conditions are applied, singularities still exist due to the vorticity inlet condition. This condition is a derivative condition and is not expressible as a constant linear combination of the dependent variables of the gap PDEs. Therefore, there are two singularities in the eigenvalue equation, but when a transfer function is formed, these artifacts are removed completely.

The same cause of the singularities in the original eigenvalue equation also causes branch cuts to appear, but in this case due to both the $\mathbf{V}^{-1}(x_{TE},s)$ matrix of $\mathbf{A}_K(s)$ and the $\mathbf{V}(x_{LE},s)$ matrix of $\mathbf{A}_1(s)$. Appendix B shows the matrix product $\mathbf{V}(x_{LE},s)\mathbf{V}^{-1}(x_{TE},s)$ which is free of branch cuts. Therefore, in the original eigenvalue equation, only two sets of branch cuts remained due to the inlet and exit duct solutions.

The reason for these remaining branch cuts lies in the end conditions. Bonnaure's assumptions about the end conditions was that they were infinite length ducts. The problem was not in the assumption, but in how it was implemented: one part of the static pressure solution was chosen to be zero in each of the inlet and exit ducts. While this implementation will work for 1D flow or incompressible 2D flow models, it causes problems in the 2D compressible flow model. The reason is that, in the 2D compressible flow model, there is no clear distinction between up- and downstream traveling (or decaying) static pressure perturbations as there is in 1D flow or incompressible 2D flow. The separation of the two parts of the static pressure perturbation in the original end conditions causes the branch cuts to appear, since the mathematical form of the boundary conditions does not provide a matrix product $\mathbf{V}(x_{LE},s)\mathbf{V}^{-1}(x_{TE},s)$. Instead, one of these matrices is left alone on each end of the compressor (one for the inlet and one for the exit). The addition of realistic end conditions in Section 2.2.5 results in the cancellation of these artifacts.

While it is not uncommon for solutions to linear constant coefficient PDEs to contain branch cuts, it is inconvenient to have them show up in input-output forms of the solution, especially when performing approximations. However, if all terms of the solution are included, and if boundary conditions are posed in terms of the PDE dependent variables, then the branch cuts tend not to appear in input-output forms of the model. In fact, they do not appear in the 1D diffusion equation (as described below), the 1D Bernoulli-Euler beam model [18], and the present 2D compressible flow model when one dimension is reduced by Fourier series expansion. For example, take the 1D diffusion equation:

$$\alpha^2 u_{xx} = u_t .$$

The general solution, by Laplace transform in time, and solving the resulting ODE in x, is

$$u(x,s) = Ae^{\frac{x\sqrt{s}}{\alpha}} + Be^{-\frac{x\sqrt{s}}{\alpha}}$$

which contains branch cuts in the form of $\sqrt{s}$. Using this solution, the transmission matrix between axial locations can be found by writing the dependent variables of the PDE in terms of the coefficients:

$$\begin{bmatrix} u(x,s) \\ u_x(x,s) \end{bmatrix} = \begin{bmatrix} e^{\frac{x\sqrt{s}}{\alpha}} & e^{-\frac{x\sqrt{s}}{\alpha}} \\ \frac{\sqrt{s}}{\alpha} e^{\frac{x\sqrt{s}}{\alpha}} & -\frac{\sqrt{s}}{\alpha} e^{-\frac{x\sqrt{s}}{\alpha}} \end{bmatrix} \begin{bmatrix} A \\ B \end{bmatrix}$$

and eliminating the coefficients using two axial locations:

$$\begin{bmatrix} u(x_2,s) \\ u_x(x_2,s) \end{bmatrix} = \begin{bmatrix} \cosh\left(\frac{\Delta x\sqrt{s}}{\alpha}\right) & \frac{\alpha}{\sqrt{s}}\sinh\left(\frac{\Delta x\sqrt{s}}{\alpha}\right) \\ \frac{\sqrt{s}}{\alpha}\sinh\left(\frac{\Delta x\sqrt{s}}{\alpha}\right) & \cosh\left(\frac{\Delta x\sqrt{s}}{\alpha}\right) \end{bmatrix} \begin{bmatrix} u(x_1,s) \\ u_x(x_1,s) \end{bmatrix}.$$

All of the elements of the transmission matrix are even functions of $\sqrt{s}$, which means the branch cuts of the solution do not appear. If end conditions were applied to each axial location in terms of $u(x,s)$ and $u_x(x,s)$, then an eigenvalue equation could be formulated without branch cuts. If one of the ends was then taken to infinity as the limit, the transmission matrix would still not contain branch cuts, but the eigenvalues would all move toward the origin, approaching a continuum of eigenvalues, which is different than a branch cut.

In the 2D problem, the PDE solutions containing branch cuts have arbitrary coefficients. When these coefficients are chosen properly, such as in the previous example (A and B), where they were solved in terms of the PDE dependent variables, the branch cuts do not appear. However, when the coefficients are chosen by other methods, such as setting one to zero, the discontinuity of the branch cuts will appear. Two conditions which may cause branch cuts to appear in input-output forms of structural dynamic systems [19] are when approximations to infinite length end conditions are used, and the dereverberated (ignoring reflected or delayed parts of the solution) input-output form is used. In both of these cases, the choice of coefficients causes branch cuts to appear.

2.3 Input-Output Form

Now that we have a complete description of the system, to solve for the open loop response, or transfer function, we only need to model the sensor and actuator and perform some manipulations using the stage stacking technique. The sensors and actuators physically will be arrays around the annulus to measure and affect different harmonics of the perturbations. The discrete nature of the sensors and actuators will limit the number of harmonics that we can control, but experimental results [4] and results of using the new eigenvalue equation (2.49) indicate that the first three harmonics are the most unstable and thus the most important to control. The placement of sensors and actuators will be restricted to the gaps or inlet and exit ducts due to physical constraints. As actuators for this system, we choose injection of high momentum air (jets), because injectors are predicted to be effective in controlling rotating stall [20]. As sensors, we choose static pressure probes since these are likely candidates for sensing in high speed compressors, although others are easily incorporated and will be studied in Chapter 5. If the sensors and actuators are in the same gap, the sensors will be assumed to be in front of the actuators. The reason for this is the discrete nature of the sensors and actuators; if the sensors are downstream of the jets, they may or may not be impinged upon by the jets. If they are not impinged, then the resulting transfer functions would look like the sensors are upstream of the jets. If they are impinged, the distance is so short that the jets would not have time to spread out in the flow properly, and the measurement would not be valid.

2.3.1 Actuator Model

An actuator can be modeled by specifying the change in dependent variables across the actuator disk. The four equations necessary to determine the change in the quartet of unknowns across the actuator disk are continuity, x momentum, θ momentum and energy conservation. The conservation equations and their linearizations are in Appendix A. The result of the linearization is the following equation:

$$\mathbf{J}_{ku}\mathbf{V}(x_u,s)\begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_{ku,downstream} = \mathbf{J}_{ku}\mathbf{V}(x_u,s)\begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_{ku,upstream} + \mathbf{b}_{ku}u(s). \tag{2.50}$$

where the control term, u, is the ratio of injected mass flow to the mean mass flow and $\mathbf{J}_{ku}$ and $\mathbf{b}_{ku}$ are results of the linearization of the conservation equations.

If we write

$$\begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_{K+1} = \mathbf{A}_a(s) \begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_{ka,downstream} \tag{2.51}$$

where $\mathbf{A}_a(s)$ is the part of the $\mathbf{A}(s)$ matrix going from the actuator location to the exit using the stage stacking method:

$$\mathbf{A}_a(s) = \mathbf{A}_K(s)\mathbf{A}_{K-1}(s)\cdots\mathbf{A}_{ka}, \tag{2.52}$$

then Equation 2.50 becomes

$$\begin{aligned} \begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_{K+1} &= \mathbf{A}_a(s)\left(\begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_{ka,upstream} + \left(\mathbf{J}_{ka}\mathbf{V}_{ka}(x_a,s)\right)^{-1}\mathbf{b}_{ka}u(s)\right) \\ &= \underbrace{\mathbf{A}_a(s)\mathbf{A}_{ka-1}(s)\cdots\mathbf{A}_1(s)}_{\mathbf{A}(s)}\begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_1 + \mathbf{A}_a(s)\left(\mathbf{J}_{ka}\mathbf{V}_{ka}(x_a,s)\right)^{-1}\mathbf{b}_{ka}u(s) \end{aligned} \tag{2.53}$$

Combining the end conditions, Equations 2.39 and 2.44, with Equation 2.53 results in

$$\begin{bmatrix} \mathbf{X}(s)\mathbf{A}(s) \\ \mathbf{N}(s) \end{bmatrix}\begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_1 = \begin{bmatrix} -\mathbf{X}(s)\mathbf{A}_a(s)\left(\mathbf{J}_{ka}\mathbf{V}_{ka}(x_a,s)\right)^{-1}\mathbf{b}_{ka} \\ 0 \end{bmatrix}u(s). \tag{2.54}$$

This equation can be used to solve for the response of the unknowns in the inlet duct, gap 1, from the input of the actuator. Notice the eigenvalue equation will turn up in the denominator when the inverse is taken.

2.3.2 Sensor Model

It now remains to define the sensor measurement and relate it to the unknowns in the inlet duct, gap 1.

The state where the sensor is located can be defined by stacking stages from the inlet and including the effect of the actuator if it is between the inlet and the sensor location:

$$\begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_{ks} = \mathbf{A}_s(s) \begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_1 + \mathbf{A}_{as}\left(\mathbf{J}_{ku}\mathbf{V}_{ku}(x_a,s)\right)^{-1}\mathbf{b}_{ku}u(s) \tag{2.55}$$

where $\mathbf{A}_s(s)$ is the part of the $\mathbf{A}(s)$ matrix going from the inlet to the sensor location,

$$\mathbf{A}_s(s) = \mathbf{A}_{ks-1}(s)\cdots\mathbf{A}_2(s)\mathbf{A}_1(s) \tag{2.56}$$

and $\mathbf{A}_{as}(s)$ is the part of the $\mathbf{A}(s)$ matrix going from the actuator location to the sensor location,

$$\mathbf{A}_{as}(s) = \mathbf{A}_{ks-1}(s)\cdots\mathbf{A}_{ku}(s) \tag{2.57}$$

which is zero if the sensor is upstream of the actuator.

The measurement of the sensor gap unknowns is then written as

$$\frac{\delta P}{P}(s)=\begin{bmatrix}1 & 0 & 0 & 0\end{bmatrix}\mathbf{V}(x_s,s)\begin{bmatrix}B\\C\\D\\E\end{bmatrix}_{ks}=\left(\mathbf{S}_{ks}\mathbf{V}(x_s,s)\right)\begin{bmatrix}B\\C\\D\\E\end{bmatrix}_{ks} \tag{2.58}$$

2.3.3 Transfer Function Description

Using Equations 2.54, 2.55 and 2.58 together, the transfer function from actuator to sensor can be written as

$$\frac{\delta P}{P}(s)=\left(\mathbf{S}_{ks}\mathbf{V}_{ks}(s)\right)\left(\begin{matrix}\mathbf{A}_s(s)\begin{bmatrix}\mathbf{X}(s)\mathbf{A}(s)\\ \mathbf{N}(s)\end{bmatrix}^{-1}\begin{bmatrix}-\mathbf{X}(s)\mathbf{A}_u(s)\\ 0\end{bmatrix}\\ +\mathbf{A}_{us}(s)\end{matrix}\right)\left(\mathbf{J}_{ku}\mathbf{V}_{ku}(s)\right)^{-1}\mathbf{b}_{ku}u(s) \tag{2.59}$$

Note that the matrix inverse in Equation 2.59 is the same matrix in the eigenvalue equation (2.47). At the solutions of the eigenvalue equation, this matrix is singular. The result is that the eigenvalues of the system show up as poles in the transfer function, just as in linear, time invariant ODE systems.

2.3.4 Discussion

This open-loop transfer function, Equation 2.59, is an important step toward the design of an active control system. Unfortunately, it is so complex, that we can only apply *ad hoc* techniques to stabilize the compressor or to improve the performance.

Equation 2.59 represents all the possible location combinations of sensors and actuators throughout the compressor as well as all of the harmonics. It would be overwhelming to apply *ad hoc* techniques to each possible combination available for study. For example, a three stage compressor plus IGVs has 8 gaps in which to place sensors and actuators resulting in 64 different single-input/single-output transfer functions for each harmonic and for each sensor and actuator type.

The next chapter derives an approximation to this model which will allow us to write a familiar state-space description to represent the system. This is desired because there are many modern automated procedures to design control systems for state-space representations. We will use Equation 2.59 to check the accuracy of the approximation, and ultimately apply the control system to it to check stability and robustness.

The technique used in Sections 2.2 and 2.3 to yield the input-output model can be generalized to many different PDE-described systems. If one can write an eigenvalue equation of the system by combining solutions within the system with the boundary and end conditions, then writing the open-loop transfer function is only two steps away. First, the description of the effect of an actuator and the measurement of the sensor is needed, then the control variable is left as a free parameter in the eigenvalue equation, and the measurement variable is solved for in terms of the control variable.

2.4 Comments on the Partial Differential Equation Modeling

The appearance of branch cuts in a system model should be a clue to the use of a set of improper end or boundary conditions, unless a physical reason can be put forth to describe them. Transfer function in the s-domain are generally analytic functions, with the exception of a set of poles and a set of zeros, possibly infinite sets. The appearance of branch cuts in Bonnaure's eigenvalue equation and in the resulting transfer functions is the result of misinterpreting the two coefficients of the static pressure solution, $B(s)$ and $C(s)$ in the solutions, Equations 2.17-2.20. These were interpreted as up- and downstream decaying solutions, respectively, like those that appear in the incompressible model. However, the two parts of the static pressure solution cannot be interpreted separately. It seems, at first glance, that they behave like the incompressible solution at low frequencies, as decaying solutions, but the compressible model is valid at higher frequencies as well, which is where the similarity ends and the branch cuts appear.

To ensure that there are no problems such as this, boundary and end conditions should be stated in terms of the dependent variables of the PDEs. As the limit of the duct length approaches infinity, the use of boundary conditions in this form may or may not yield a simple approximation. In the current problem, the length of the ducts can play an important role in determining the frequency and stability of the dominant eigenvalues, so it is best to use a duct length and end conditions consistent with the experimental apparatus.

Finally, physical systems which do not have energy feeding them, should not display unstable eigenvalues. If some do appear, as in the case with the 2D gap solutions, then they must be artifacts which will cancel out with appropriate boundary conditions. If the transfer functions across the inter-blade row gaps were not examined in detail to find these cancellations, the inclusion of these eigenvalues could have resulted in severe degradation of the approximate model.

Chapter 3

State-Space Approximation of Analytical Model

This chapter presents approximations to the analytical model derived in the preceding chapter. First, however, some motivation for performing the approximation and choosing the method of approximation is given in Section 3.1. Section 3.2 presents steps necessary to approximate the model ignoring the dynamics of the inter-blade row gaps. This loses some accuracy, but will result in a smaller model. Section 3.2.1 presents the blade row approximations, Section 3.2.2 presents the boundary conditions with actuation, Section 3.2.3 presents the end condition approximations, and Section 3.2.4 compiles the approximations into the complete state-space model. Section 3.3 presents steps necessary to approximate the truth model including the dynamics of the gaps, resulting in a higher order, more accurate model. Sections 3.3.1 and 3.3.2 present boundary conditions and gap approximations for non-zeroth and the zeroth harmonic, respectively. These are used with the blade row and end condition approximations to compile the complete state-space models in Sections 3.3.3 and 3.3.4 for non-zeroth and the zeroth harmonic, respectively. Finally, Section 3.4 presents an accuracy analysis of the two approximate models in terms of their potential use in stability analysis and control law design.

3.1 Motivation for State-Space Model

Now that we have an input-output frequency domain representation of the dynamics of a compressor, we could apply classical control theory to stabilize it, if it is unstable, or increase its stability margins if it is stable. Unfortunately, even a three stage compressor with a set of inlet guide vanes has 8 gaps and ducts in which one can place sensors and actuators (see Figure 2.1). The result is 64 single-input/single-output pairs which are possible for sensor and actuator placement. If we want to find the best possible combination, we have to examine 64 control designs.

Using classical control methods, this would be a tedious procedure which would yield qualitative, not quantitative, results, making it difficult to determine the best sensor and actuator locations. Other disadvantages to classical control theory are: there is no allowance for multiple noise sources and their effect on the resulting actuator motion, and there is only a limited amount of freedom in designing a controller for multiple inputs or multiple outputs. There can be some allowance for robustness to plant parameter uncertainty, but only to a limited degree.

Using modern optimal control design methods, such as Linear Quadratic-Gaussian (LQG) or H_∞ design with a state-space model, one can automate the design procedure to examine 64 control designs in much less time than using the classical methods. Moreover, these procedures result in cost functions, such as a sum of the mean-squares of the states and the control signals, or the H_∞-norm of a particular transfer function. These are inherently multiple-input/multiple-output design methods allowing for multiple noise sources. There are also ways of designing for robustness to different types of uncertainty. Applying these techniques requires a state-space model, which in this case requires an approximation of the input-output form.

Different methods can be used to approximate a transfer function originating from a set of PDEs. First, the frequency domain data (a set of points of the transfer function evaluated on the imaginary axis) can be used directly in a numerical scheme to provide a state-space description. Or, if the transfer function is simple enough, one can directly approximate the transfer function by a rational polynomial expression in the Laplace variable, s. In this thesis, the latter approach is taken to approximate each blade row and each gap and duct separately by rational polynomials. The advantage of this method over the numerical method is that all 64 transfer functions are found at the same time in one state-space system, with physically meaningful states, in less time than it would take to evaluate the analytical model at enough frequencies to feed to the numerical procedure. The disadvantage of this method is the relatively high order in the resulting state-space model.

The first approximate model, derived in Section 3.2, ignores the dynamics of the inter-blade row gaps. This assumption is valid for compressors with short gaps with respect to the length of the blade rows. However, since the total length of the compressor affects the frequency of the eigenvalues, the missing gap length is absorbed into the preceding blade row to keep the total length constant. Modeling the blade row dynamics is easy since the solutions are simple time delays as perturbations travel through the blade row. These delays can be thought of as having an 'input' at one edge of the blade row, and an 'output' at the other edge, depending on whether the perturbation is traveling up- or downstream. All of the delay outputs at the interface (the perturbations traveling toward the interface) between adjacent blade rows are then combined to solve for the inputs of the other delays at the same interface (the perturbations traveling away from the interface). This is the basis for the model without gaps.

The second approximate model, derived in Section 3.3, models the dynamics of the gaps which results in a higher order, more accurate approximation. The gap dynamics have no identifiable 'inputs' or 'outputs' as the blade row solutions do, so the dynamics are effectively lumped into transfer functions between parts of the blade row solution. For example, the 'inputs' of the delays in a blade row now depend on the 'outputs' at the interface filtered through the lumped gap dynamics. This is the basis for the model including the gaps as well as the basis for modeling the inlet and exit ducts.

3.2 State-Space Model without Gaps

This section derives the model approximation ignoring the inter-blade row gap dynamics. Section 3.2.1 describes the approximations to the delays in the blade row equations. Section 3.2.2 describes the boundary conditions, now applied with zero gap length, including actuation. Section 3.2.3 describes the end condition approximations including actuation. The blade row approximations and end condition approximations will also be used in the approximation including gap dynamics in Section 3.3. Section 3.2.4 shows how the separate parts of the approximation fit together to form the complete state-space compressor model.

3.2.1 Blade Row Approximations

A few definitions will be made for notation convenience. First, starting with Equations 2.30, 2.31 and 2.32, the following definitions will be made:

$$
\begin{aligned}
\tilde{B}_{LE}(s) &= \tilde{B}(s)e^{\tilde{\alpha}(s)x_{LE}} \\
\tilde{C}_{LE}(s) &= \tilde{C}(s)e^{\tilde{\beta}(s)x_{LE}} \\
\tilde{E}_{LE}(s) &= \tilde{E}(s)e^{\tilde{\chi}(s)x_{LE}}
\end{aligned}
\tag{3.1}
$$

and similarly

$$
\begin{aligned}
\tilde{B}_{TE}(s) &= \tilde{B}(s)e^{\tilde{\alpha}(s)x_{TE}} \\
\tilde{C}_{TE}(s) &= \tilde{C}(s)e^{\tilde{\beta}(s)x_{TE}} \,. \\
\tilde{E}_{TE}(s) &= \tilde{E}(s)e^{\tilde{\chi}(s)x_{TE}}
\end{aligned}
\tag{3.2}
$$

The relations between Equations 3.1 and 3.2 become

$$\boxed{\begin{aligned}\tilde{B}_{LE}(s) &= e^{-\tilde{\alpha}(s)\Delta x}\tilde{B}_{TE}(s)\\ \tilde{C}_{TE}(s) &= e^{\tilde{\beta}(s)\Delta x}\tilde{C}_{LE}(s)\\ \tilde{E}_{TE}(s) &= e^{\tilde{\chi}(s)\Delta x}\tilde{E}_{LE}(s)\end{aligned}} \tag{3.3}$$

These relations are now all in the form of a complex constant multiplying a time delay, $e^{j\Delta\theta}e^{-s\Delta T}$. Notice the $\tilde{B}$ equation is a time delay with a negative Δx, which means it is an upstream traveling perturbation. The time delays of Equation 3.3 are, respectively

$$\begin{aligned}\Delta T_B &= \frac{\Delta x}{(a-W)\cos\xi}\\ \Delta T_C &= \frac{\Delta x}{(W+a)\cos\xi}\\ \Delta T_E &= \frac{\Delta x}{W\cos\xi}\end{aligned} \tag{3.4}$$

The phases of the complex constants of Equation 3.3 are, respectively

$$\begin{aligned}\Delta\theta_B &= -n\Omega\Delta T_B + n\tan\xi\frac{\Delta x}{r}\\ \Delta\theta_C &= -n\Omega\Delta T_C - n\tan\xi\frac{\Delta x}{r}\\ \Delta\theta_E &= -n\Omega\Delta T_E - n\tan\xi\frac{\Delta x}{r}\end{aligned} \tag{3.5}$$

These time delays are approximated using Padé approximations. For example, the upstream pressure perturbation can be written using a second order Padé approximation [21] as

$$\frac{\tilde{B}_{LE}(s)}{\tilde{B}_{TE}(s)} = e^{j\Delta\theta_B - s\Delta T_B} \cong e^{j\Delta\theta_B}\frac{1-\frac{1}{2}\Delta T_B s+\frac{1}{12}\Delta T_B^2 s^2}{1+\frac{1}{2}\Delta T_B s+\frac{1}{12}\Delta T_B^2 s^2}. \tag{3.6}$$

Note that the blade row solution has three independent delays associated with it. These delays are functions of axial distance, relative speed of the perturbation and stagger angle of the blade row. The complex constant is also a function of these values, as well as harmonic number and rotor angular velocity (zero for a stator). Appendix C provides an example of a first order Padé approximation in state-space form.

3.2.2 Boundary Conditions with Actuation

Now that we have blade row approximations to the solutions, we need to connect these to the adjacent blade rows, since the gaps will be assumed to have negligible length. This will be accomplished through the boundary conditions, Equations 2.34 and 2.35.

The boundary conditions, Equations 2.34 and 2.35, using the actuation effect, Equation 2.50, at the trailing edge, combined are

$$\mathbf{B}_{Lk}\mathbf{B}_k(x_{LEk},s)\begin{bmatrix}\tilde{B}\\ \tilde{C}\\ \tilde{E}\end{bmatrix}_k = (\mathbf{V}_{Lk} + \tfrac{1}{1+s\tau}\mathbf{P}_k)\mathbf{V}_k(x_{LEk},s)\mathbf{V}_k^{-1}(x_{TEk-1},s)\bullet$$
$$\left(\mathbf{V}_{Tk}^{-1}\left(\mathbf{B}_{Tk-1}\mathbf{B}_{k-1}(x_{TEk-1},s)\begin{bmatrix}\tilde{B}\\ \tilde{C}\\ \tilde{E}\end{bmatrix}_{k-1} - \tfrac{1}{1+s\tau}\mathbf{D}_{k-1}\mathbf{V}_{k-1}(x_{LEk-1},s)\begin{bmatrix}B\\ C\\ D\\ E\end{bmatrix}_{k-1}\right) + \mathbf{J}_k^{-1}\mathbf{b}_k u_k\right) \tag{3.7}$$

The assumption that the gap has no dynamics, i.e. $x_{TEk-1} = x_{LEk}$, allows the reduction of this to

$$\mathbf{B}_{Lk}\mathbf{B}_k(x_{LEk},s)\begin{bmatrix}\tilde{B}\\ \tilde{C}\\ \tilde{E}\end{bmatrix}_k = (\mathbf{V}_{Lk} + \tfrac{1}{1+s\tau}\mathbf{P}_k)\bullet$$

$$\left(\begin{array}{l}\mathbf{V}_{Tk}^{-1}\left(\begin{array}{l}\mathbf{B}_{Tk-1}\mathbf{B}_{k-1}(x_{TEk-1},s)\begin{bmatrix}\tilde{B}\\ \tilde{C}\\ \tilde{E}\end{bmatrix}_{k-1}\\ -\tfrac{1}{1+s\tau}\mathbf{D}_{k-1}\mathbf{V}_{k-1}(x_{LEk-1},s)\begin{bmatrix}B\\ C\\ D\\ E\end{bmatrix}_{k-1}\end{array}\right)\\ +\mathbf{J}_k^{-1}\mathbf{b}_k u_k\end{array}\right) \tag{3.8}$$

Using Equations 3.1, and the blade row solutions (Equations 2.30, 2.31 and 2.32), we can write this as

$$\mathbf{B}_{Lk}\mathbf{M}\begin{bmatrix}\tilde{B}_{LE}\\ \tilde{C}_{LE}\\ \tilde{E}_{LE}\end{bmatrix}_k = (\mathbf{V}_{Lk} + \tfrac{1}{1+s\tau}\mathbf{P}_k)\bullet$$

$$\left(\begin{array}{l}\mathbf{V}_{Tk}^{-1}\left(\begin{array}{l}\mathbf{B}_{Tk-1}\mathbf{M}\begin{bmatrix}\tilde{B}_{TE}\\ \tilde{C}_{TE}\\ \tilde{E}_{TE}\end{bmatrix}_{k-1}\\ -\tfrac{1}{1+s\tau}\mathbf{D}_{k-1}\mathbf{V}_{k-1}(x_{LEk-1},s)\begin{bmatrix}B\\ C\\ D\\ E\end{bmatrix}_{k-1}\end{array}\right)\\ +\mathbf{J}_k^{-1}\mathbf{b}_k u_k\end{array}\right) \tag{3.9}$$

where

$$\mathbf{M} = \begin{bmatrix} \gamma & \gamma & 0 \\ 1 & 1 & 1 \\ -1 & 1 & 0 \end{bmatrix} \tag{3.10}$$

A definition is in order to remove the s-dependencies in Equation 3.9:

$$\delta\alpha_{k-1} = -\tfrac{1}{1+s\tau}\mathbf{D}_{4r,k-1}\mathbf{V}_{k-1}(x_{LEk-1},s)\begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_{k-1} \tag{3.11}$$

where $\mathbf{D}_{4r,k-1}$ is the fourth row of the matrix $\mathbf{D}_{k-1}$, the only non-zero row. Unfortunately, this still has some s-dependencies, but this can be resolved by using Equation 3.11 recursively in the trailing edge boundary condition (Equation 2.35), Equation 3.2, the definition for $\mathbf{M}$ (Equation 3.10), and the actuation effect, Equation 2.50:

$$\boxed{\delta\alpha_{k-1} = -\tfrac{1}{1+s\tau}\mathbf{D}_{4r,k-1}\left(\mathbf{V}_{Tk-1}^{-1}\begin{bmatrix} & & 0 \\ \mathbf{B}_{Tk-2}\mathbf{M} & & 0 \\ & & 0 \\ & & 1 \end{bmatrix}\begin{bmatrix} \tilde{B}_{TE} \\ \tilde{C}_{TE} \\ \tilde{E}_{TE} \\ \delta\alpha \end{bmatrix}_{k-2} + \mathbf{J}_{k-1}^{-1}\mathbf{b}_{k-1}u_{k-1}\right)}. \tag{3.12}$$

Using Equation 3.11, Equation 3.9 reduces to

$$\mathbf{B}_{Lk}\mathbf{M}\begin{bmatrix} \tilde{B}_{LE} \\ \tilde{C}_{LE} \\ \tilde{E}_{LE} \end{bmatrix}_k = (\mathbf{V}_{Lk} + \tfrac{1}{1+s\tau}\mathbf{P}_k)\left(\begin{matrix} \mathbf{V}_{Tk}^{-1}\begin{bmatrix} & & 0 \\ \mathbf{B}_{Tk-1}\mathbf{M} & & 0 \\ & & 0 \\ & & 1 \end{bmatrix}\begin{bmatrix} \tilde{B}_{TE} \\ \tilde{C}_{TE} \\ \tilde{E}_{TE} \\ \delta\alpha \end{bmatrix}_{k-1} \\ +\mathbf{J}_k^{-1}\mathbf{b}_k u_k \end{matrix}\right) \tag{3.13}$$

Now, another definition is in order to remove the last s-dependence from Equation 3.13:

$$\boxed{P_{loss,k} = \frac{1}{1+s\tau}\mathbf{P}_{3r,k}\left(\mathbf{V}_{Tk}^{-1}\begin{bmatrix} \mathbf{B}_{Tk-1}\mathbf{M} & \begin{matrix}0\\0\\0\end{matrix} \\ & 1 \end{bmatrix}\begin{bmatrix}\tilde{B}_{TE}\\ \tilde{C}_{TE}\\ \tilde{E}_{TE}\\ \delta\alpha\end{bmatrix}_{k-1} + \mathbf{J}_k^{-1}\mathbf{b}_k u_k\right)} \tag{3.14}$$

where $\mathbf{P}_{3r,k}$ is the third row of $\mathbf{P}_k$, the only non-zero row. Equations 3.12 and 3.14 are simple time lags from variables that are results of rational polynomial expressions, with the exception of the $\tilde{B}_{TE}$ term, which is an input to a time lag, but will be solved for shortly. Now, a constant transformation between the states of the blade rows, the states of the lags, and the control can be realized at the boundary between adjacent blade rows using Equations 3.13 and 3.14:

$$\begin{bmatrix}\tilde{B}_{LE}\\ \tilde{C}_{LE}\\ \tilde{E}_{LE}\end{bmatrix}_k = \mathbf{M}^{-1}\mathbf{B}_{Lk}^{-1}\mathbf{V}_{Lk}\mathbf{V}_{Tk}^{-1}\begin{bmatrix} \mathbf{B}_{Tk-1}\mathbf{M} & \begin{matrix}0\\0\\0\end{matrix} \\ & 1 \end{bmatrix}\begin{bmatrix}\tilde{B}_{TE}\\ \tilde{C}_{TE}\\ \tilde{E}_{TE}\\ \delta\alpha\end{bmatrix}_{k-1} + \mathbf{M}^{-1}\mathbf{B}_{Lk}^{-1}\begin{bmatrix}0\\0\\1\end{bmatrix}P_{loss,k} + \mathbf{M}^{-1}\mathbf{B}_{Lk}^{-1}\mathbf{V}_{Lk}\mathbf{J}_k^{-1}\mathbf{b}_k u_k \tag{3.15}$$

This constant transformation equation will be solved for the set $(\tilde{B}_{TEk-1}, \tilde{C}_{LEk}, \tilde{E}_{LEk})$ since these are the 'inputs' to the time delays (Equation 3.3):

$$\boxed{\begin{bmatrix}\tilde{B}_{TEk-1}\\ \tilde{C}_{LEk}\\ \tilde{E}_{LEk}\end{bmatrix} = \mathbf{K}_k\begin{bmatrix}\tilde{B}_{LEk}\\ \tilde{C}_{TEk-1}\\ \tilde{E}_{TEk-1}\\ P_{loss,k}\\ \delta\alpha_{k-1}\\ u_k\end{bmatrix}} \tag{3.16}$$

which is a constant transformation. The variables on the right hand side are all 'outputs', or results of Padé approximations or time lags.

3.2.3 End Condition Approximations with Actuation

The end condition modeling begins with stating the specified end conditions and joining these with the 2D duct solutions of Chapter 2. In the inlet condition, pressure loss and deviation states are identified to remove their dynamics. Then, the 'inputs' to the adjacent blade row are identified and solved for in terms of the 'outputs' and the pressure and deviation states and the control input. The results are transcendental equations in the frequency variable s, which are then approximated using Taylor series expansions.

3.2.3.1 Inlet Condition

The inlet conditions, Equation A.33, can be reduced to

$$e^{\alpha(s)x_{in}} B_1(s) + e^{\beta(s)x_{in}} C_1(s) = D_1(s) = E_1(s) = 0 \tag{3.17}$$

These conditions are used to find transfer functions from $\tilde{B}_{LE1}(s)$ and $u_1(s)$ (if control is applied at the inlet) to $\tilde{C}_{LE1}(s), \tilde{E}_{LE1}(s), \delta\alpha_1(s), P_{loss,1}(s)$. The inlet conditions, Equation 3.17, along with the effect of an actuator, Equation 2.50, used in the leading edge boundary conditions, Equation 2.34, can be written

$$\begin{bmatrix} \tilde{B}_{LE} \\ \tilde{C}_{LE} \\ \tilde{E}_{LE} \end{bmatrix}_1 = \mathbf{B}_{L1}^{-1}(\mathbf{V}_{L1} + \tfrac{1}{1+s\tau}\mathbf{P}_1) \left(\mathbf{V}_1(x_{LE1}, s) \begin{bmatrix} 1 \\ -e^{(\alpha-\beta)x_{in}} \\ 0 \\ 0 \end{bmatrix} B_1 + \mathbf{J}_1^{-1}\mathbf{b}_1 u_1 \right) \tag{3.18}$$

where the control has been applied in the gap at the leading edge of the first blade row. A pressure loss state can again be introduced to eliminate some of the s-dependencies:

$$P_{loss,1} = \frac{1}{1+s\tau}\mathbf{P}_{1,3r}\left(\mathbf{V}_1(x_{LE1},s)\begin{bmatrix} 1 \\ -e^{(\alpha-\beta)x_m} \\ 0 \\ 0 \end{bmatrix}B_1 + \mathbf{J}_1^{-1}\mathbf{b}_1u_1\right). \tag{3.19}$$

Equation 3.18 can then be written

$$\begin{bmatrix} \tilde{B}_{LE} \\ \tilde{C}_{LE} \\ \tilde{E}_{LE} \end{bmatrix}_1 = \mathbf{B}_{L1}^{-1}\mathbf{V}_{L1}\mathbf{V}_1(x_{LE1},s)\begin{bmatrix} 1 \\ -e^{(\alpha-\beta)x_m} \\ 0 \\ 0 \end{bmatrix}B_1 + \mathbf{B}_{L1}^{-1}\mathbf{V}_{L1}\mathbf{J}_1^{-1}\mathbf{b}_1u_1 + \mathbf{B}_{L1}^{-1}\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}P_{loss,1}. \tag{3.20}$$

A few definitions will be made for convenience, replacing groups of constant matrices with single matrices in Equation 3.20:

$$\begin{bmatrix} \tilde{B}_{LE} \\ \tilde{C}_{LE} \\ \tilde{E}_{LE} \end{bmatrix}_1 = \mathbf{X}_1\mathbf{V}_1(x_{LE1},s)\begin{bmatrix} 1 \\ -e^{(\alpha-\beta)x_m} \\ 0 \\ 0 \end{bmatrix}B_1 + \mathbf{s}_1u_1 + \mathbf{w}_1P_{loss,1} \tag{3.21}$$

where the definitions of $\mathbf{X}_1$, $\mathbf{s}_1$, and $\mathbf{w}_1$ are obvious by comparing Equations 3.20 and 3.21. Expanding the first term for non-zeroth harmonics yields

$$\begin{bmatrix} \tilde{B}_{LE} \\ \tilde{C}_{LE} \\ \tilde{E}_{LE} \end{bmatrix}_1 = \mathbf{X}_1\begin{bmatrix} \gamma\left(\left(\frac{sr}{a} + jnM_\theta + M_x\alpha\right)e^{\alpha x_{LE1}} - \left(\frac{sr}{a} + jnM_\theta + M_x\beta\right)e^{\alpha x_m}e^{\beta\Delta x_1}\right) \\ \left(\left(\frac{sr}{a} + jnM_\theta + M_x\alpha\right)e^{\alpha x_{LE1}} - \left(\frac{sr}{a} + jnM_\theta + M_x\beta\right)e^{\alpha x_m}e^{\beta\Delta x_1}\right) \\ -\alpha e^{\alpha x_{LE1}} + \beta e^{\alpha x_m}e^{\beta\Delta x_1} \\ -jne^{\alpha x_{LE1}} + jne^{\alpha x_m}e^{\beta\Delta x_1} \end{bmatrix}B_1 + \tag{3.22}$$

$$\mathbf{s}_1u_1 + \mathbf{w}_1P_{loss,1}$$

and for the zeroth harmonic,

$$\begin{bmatrix} \tilde{B}_{LE} \\ \tilde{C}_{LE} \\ \tilde{E}_{LE} \end{bmatrix}_1 = \mathbf{X}_1 \begin{bmatrix} \gamma\left(\frac{1}{1-M_x} e^{\alpha x_{LE,1}} - \frac{1}{1+M_x} e^{\alpha x_m} e^{\beta \Delta x_1} \right) \\ \left(\frac{1}{1-M_x} e^{\alpha x_{LE,1}} - \frac{1}{1+M_x} e^{\alpha x_m} e^{\beta \Delta x_1} \right) \\ -\frac{1}{1-M_x} e^{\alpha x_{LE,1}} - \frac{1}{1+M_x} e^{\alpha x_m} e^{\beta \Delta x_1} \\ 0 \end{bmatrix} B_1 + \mathbf{s}_1 u_1 + \mathbf{w}_1 P_{loss,1} \tag{3.23}$$

or in a general form, Equations 3.22 and 3.23 can be written as

$$\begin{bmatrix} \tilde{B}_{LE} \\ \tilde{C}_{LE} \\ \tilde{E}_{LE} \end{bmatrix}_1 = \begin{bmatrix} u_{11}(s) \\ u_{21}(s) \\ u_{31}(s) \end{bmatrix} B_1 + \mathbf{s}_1 u_1 + \mathbf{w}_1 P_{loss,1} \tag{3.24}$$

This can be rearranged through the following steps to solve for the transfer functions of interest, namely $\tilde{C}_{LE1}(s), \tilde{E}_{LE1}(s), \delta\alpha_1(s), P_{loss,1}(s)$:

$$\begin{bmatrix} -u_{11}(s) & 0 & 0 \\ -u_{21}(s) & 1 & 0 \\ -u_{31}(s) & 0 & 1 \end{bmatrix} \begin{bmatrix} B_1 \\ \tilde{C}_{LE1} \\ \tilde{E}_{LE1} \end{bmatrix} = \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix} \tilde{B}_{LE1} + \mathbf{s}_1 u_1 + \mathbf{w}_1 P_{loss,1}$$

$$\begin{bmatrix} B_1 \\ \tilde{C}_{LE1} \\ \tilde{E}_{LE1} \end{bmatrix} = \begin{bmatrix} \frac{1}{u_{11}(s)} \\ \frac{u_{21}(s)}{u_{11}(s)} \\ \frac{u_{31}(s)}{u_{11}(s)} \end{bmatrix} \tilde{B}_{LE1} + \begin{bmatrix} -s_{11} \frac{1}{u_{11}(s)} \\ -s_{11} \frac{u_{21}(s)}{u_{11}(s)} + s_{21} \\ -s_{11} \frac{u_{31}(s)}{u_{11}(s)} + s_{31} \end{bmatrix} u_1 + \begin{bmatrix} -w_{11} \frac{1}{u_{11}(s)} \\ -w_{11} \frac{u_{21}(s)}{u_{11}(s)} + w_{21} \\ -w_{11} \frac{u_{31}(s)}{u_{11}(s)} + w_{31} \end{bmatrix} P_{loss,1} \tag{3.25}$$

where w_{ij} and s_{ij} are elements of $\mathbf{w}_1$ and $\mathbf{s}_1$, respectively. The pressure loss state can now be written using Equation 3.19 and Equation 3.24:

$$P_{loss,1} = \tfrac{1}{1+s\tau}\mathbf{P}_{1,3r}\mathbf{V}_1(x_{LE1},s)\begin{bmatrix}1\\ -e^{(\alpha-\beta)x_m}\\ 0\\ 0\end{bmatrix}B_1 + \tfrac{1}{1+s\tau}\mathbf{P}_{1,3r}\mathbf{J}_1^{-1}\mathbf{b}_1 u_1 \qquad (3.26)$$

$$\equiv \tfrac{1}{1+s\tau}u_p(s)B_1 + \tfrac{1}{1+s\tau}\mathbf{P}_{1,3r}\mathbf{J}_1^{-1}\mathbf{b}_1 u_1$$

Similarly, the deviation state can be defined using Equation 3.11, the inlet conditions (Equation 3.17) the actuation effect (Equation 2.50) and Equation 3.24:

$$\delta\alpha_1 = \tfrac{1}{1+s\tau}\mathbf{D}_{1,4r}\mathbf{V}_1(x_{LE1},s)\begin{bmatrix}1\\ -e^{(\alpha-\beta)x_m}\\ 0\\ 0\end{bmatrix}B_1 + \tfrac{1}{1+s\tau}\mathbf{D}_{1,4r}\mathbf{J}_1^{-1}\mathbf{b}_1 u_1 \qquad (3.27)$$

$$\equiv \tfrac{1}{1+s\tau}u_d(s)B_1 + \tfrac{1}{1+s\tau}\mathbf{D}_{1,4r}\mathbf{J}_1^{-1}\mathbf{b}_1 u_1$$

The four transfer functions can be written as

$$\boxed{\begin{bmatrix}\tilde{C}_{LE1}\\ \tilde{E}_{LE1}\\ P_{loss,1}\\ \delta\alpha_1\end{bmatrix} = \frac{1}{u_{11}(s)}\begin{bmatrix}u_{21}(s)\\ u_{31}(s)\\ \tfrac{1}{1+s\tau}u_p(s)\\ \tfrac{1}{1+s\tau}u_d(s)\end{bmatrix}(\tilde{B}_{LE1} - w_{11}P_{loss,1} - s_{11}u_1) + \begin{bmatrix}w_{21}\\ w_{31}\\ 0\\ 0\end{bmatrix}P_{loss,1} + \begin{bmatrix}s_{21}\\ s_{31}\\ \tfrac{1}{1+s\tau}\mathbf{P}_{1,3r}\mathbf{J}_1^{-1}\mathbf{b}_1\\ \tfrac{1}{1+s\tau}\mathbf{D}_{1,4r}\mathbf{J}_1^{-1}\mathbf{b}_1\end{bmatrix}u_1} . \qquad (3.28)$$

The transfer functions have now been separated into a frequency dependent term with a single input and two 'feed-through' terms. The last step is to expand the frequency dependent terms and approximate them using Taylor series expansions. An example of the approximation will now be given.

The expansion of the first transfer function of Equation 3.27, for non-zeroth harmonics is

$$\frac{u_{21}(s)}{u_{11}(s)}=\frac{\begin{array}{c}(\gamma x_{21}+x_{22})\left[(\frac{sr}{a}+jnM_\theta+r\alpha M_x)e^{\alpha x_{LE1}}-(\frac{sr}{a}+jnM_\theta+r\beta M_x)e^{\alpha x_{in}}e^{\beta\Delta x_1}\right]+\\ x_{23}(-r\alpha e^{\alpha x_{LE1}}+r\beta e^{\alpha x_{in}}e^{\beta\Delta x_1})+x_{24}(-jne^{\alpha x_{LE1}}+jne^{\alpha x_{in}}e^{\beta\Delta x_1})\end{array}}{\begin{array}{c}(\gamma x_{11}+x_{12})\left[(\frac{sr}{a}+jnM_\theta+r\alpha M_x)e^{\alpha x_{LE1}}-(\frac{sr}{a}+jnM_\theta+r\beta M_x)e^{\alpha x_{in}}e^{\beta\Delta x_1}\right]+\\ x_{13}(-r\alpha e^{\alpha x_{LE1}}+r\beta e^{\alpha x_{in}}e^{\beta\Delta x_1})+x_{14}(-jne^{\alpha x_{LE1}}+jne^{\alpha x_{in}}e^{\beta\Delta x_1})\end{array}} \tag{3.29}$$

and for the zeroth harmonic,

$$\frac{u_{21}(s)}{u_{11}(s)}=\frac{\begin{array}{c}(\gamma x_{21}+x_{22})\left[\frac{1}{1-M_x}e^{\alpha x_{LE1}}-\frac{1}{1+M_x}e^{\alpha x_{in}}e^{\beta\Delta x_1}\right]+\\ -x_{23}(\frac{1}{1-M_x}e^{\alpha x_{LE1}}+\frac{1}{1+M_x}e^{\alpha x_{in}}e^{\beta\Delta x_1})\end{array}}{\begin{array}{c}(\gamma x_{11}+x_{12})\left[\frac{1}{1-M_x}e^{\alpha x_{LE1}}-\frac{1}{1+M_x}e^{\alpha x_{in}}e^{\beta\Delta x_1}\right]+\\ -x_{13}(\frac{1}{1-M_x}e^{\alpha x_{LE1}}+\frac{1}{1+M_x}e^{\alpha x_{in}}e^{\beta\Delta x_1})\end{array}}. \tag{3.30}$$

If the numerator and denominator are divided by an exponential and rearranged, for non-zeroth harmonics, Equation 3.29 becomes

$$\frac{u_{21}(s)}{u_{11}(s)}=\frac{\begin{array}{c}\left[(\gamma x_{21}+x_{22})(\frac{sr}{a}+jnM_\theta+r\alpha M_x)-x_{23}r\alpha-x_{24}jn\right]e^{\alpha\Delta x_1}-\\ \left[(\gamma x_{21}+x_{22})(\frac{sr}{a}+jnM_\theta+r\beta M_x)-x_{23}r\beta-x_{24}jn\right]e^{\beta\Delta x_1}\end{array}}{\begin{array}{c}\left[(\gamma x_{11}+x_{12})(\frac{sr}{a}+jnM_\theta+r\alpha M_x)-x_{13}r\alpha-x_{14}jn\right]e^{\alpha\Delta x_1}-\\ \left[(\gamma x_{11}+x_{12})(\frac{sr}{a}+jnM_\theta+r\beta M_x)-x_{13}r\beta-x_{14}jn\right]e^{\beta\Delta x_1}\end{array}} \tag{3.31}$$

and for the zeroth harmonic, Equation 3.30 becomes

$$
\frac{u_{21}(s)}{u_{11}(s)} = \frac{\begin{matrix}\left[(\gamma x_{21} + x_{22}) - x_{23}\right]\frac{1}{1-M_x} e^{\alpha\Delta x} - \\ \left[(\gamma x_{21} + x_{22}) + x_{23}\right]\frac{1}{1+M_x} e^{\beta\Delta x}\end{matrix}}{\begin{matrix}\left[(\gamma x_{11} + x_{12}) - x_{13}\right]\frac{1}{1-M_x} e^{\alpha\Delta x} - \\ \left[(\gamma x_{11} + x_{12}) + x_{13}\right]\frac{1}{1+M_x} e^{\beta\Delta x}\end{matrix}}. \tag{3.32}
$$

Equation 3.31, for non-zeroth harmonics, can be written as

$$
\frac{u_{21}(s)}{u_{11}(s)} = \frac{f_2(\alpha(s)) - f_2(\beta(s))}{f_1(\alpha(s)) - f_1(\beta(s))} \tag{3.33}
$$

It is clear to see why the branch cut does not appear in this transfer function. When the branch cut in the frequency domain is crossed, $\alpha(s)$ and $\beta(s)$ are discontinuous, but $\beta(s)$ on one side is continuous with $\alpha(s)$ on the other side, and vice versa, so the transfer function is continuous. In fact, in the process of expanding Equation 3.33, the branch cuts will be analytically removed.

Expanding Equation 3.33 further, we get

$$
f_1(\alpha(s)) = \begin{bmatrix} \left((\gamma x_{11} + x_{12})(1 + \frac{M_x^2}{1-M_x^2}) - x_{13}\frac{M_x}{1-M_x^2}\right)(\frac{sr}{a} + jnM_\theta) - \\ x_{14}jn + \left((\gamma x_{11} + x_{12})M_x - x_{13}\right)\frac{\sqrt{y(s)}}{1-M_x^2} \end{bmatrix} e^{\frac{M_x(\frac{sr}{a} + jnM_\theta)}{1-M_x^2}\frac{\Delta x_1}{r}} e^{\frac{\sqrt{y(s)}}{1-M_x^2}\frac{\Delta x_1}{r}}
\tag{3.34}
$$

where

$$
y(s) = n^2(1 - M_x^2) + \left(\frac{sr}{a} + nM_\theta\right)^2.
$$

Noting the similarity between $\alpha(s)$ and $\beta(s)$, simply the sign of the radical, we can write Equation 3.34 as

$$
\begin{aligned}
f_1(\alpha(s)) &= \left[g_1(s) + c_1\sqrt{y(s)}\right]h(s)e^{b\sqrt{y(s)}} \\
f_1(\beta(s)) &= \left[g_1(s) - c_1\sqrt{y(s)}\right]h(s)e^{-b\sqrt{y(s)}}
\end{aligned}
\tag{3.35}
$$

where

$$
\begin{aligned}
h(s) &= e^{bM_x\left(\frac{sr}{a} + jnM_\theta\right)} \\
b &= \frac{1}{1 - M_x^2}\frac{\Delta x_1}{r}
\end{aligned}
$$

The denominator of Equation 3.33 can be written as

$$
\begin{aligned}
f_1(\alpha(s)) - f_1(\beta(s)) &= \left[g_1(s) + c_1\sqrt{y(s)}\right]h(s)e^{b\sqrt{y(s)}} - \left[g_1(s) - c_1\sqrt{y(s)}\right]h(s)e^{-b\sqrt{y(s)}} \\
&= g_1(s)h(s)(e^{b\sqrt{y(s)}} - e^{-b\sqrt{y(s)}}) + h(s)c_1\sqrt{y(s)}(e^{b\sqrt{y(s)}} + e^{-b\sqrt{y(s)}}) \\
&= 2g_1(s)h(s)\sinh\left(b\sqrt{y(s)}\right) + 2h(s)c_1\sqrt{y(s)}\cosh\left(b\sqrt{y(s)}\right) \\
&= 2h(s)\sqrt{y(s)}\left(g_1(s)\frac{\sinh\left(b\sqrt{y(s)}\right)}{\sqrt{y(s)}} + c_1\cosh\left(b\sqrt{y(s)}\right)\right)
\end{aligned}
\tag{3.36}
$$

where the term in the parentheses is an even function in the square root, so the branch cut only appears in the square root out front. When the transfer function is formed, the square roots cancel, and the result is

$$
\frac{u_{21}(s)}{u_{11}(s)} = \frac{g_2(s)\dfrac{\sinh\left(b\sqrt{y(s)}\right)}{\sqrt{y(s)}} + c_2\cosh\left(b\sqrt{y(s)}\right)}{g_1(s)\dfrac{\sinh\left(b\sqrt{y(s)}\right)}{\sqrt{y(s)}} + c_1\cosh\left(b\sqrt{y(s)}\right)}
\tag{3.37}
$$

which no longer contains a branch cut. The zeroth harmonic, from Equation 3.32, can be similarly written

$$\frac{u_{21}(s)}{u_{11}(s)} = \frac{d_2 e^{\frac{\Delta x_1}{a-V_s}s} - f_2 e^{-\frac{\Delta x_1}{a+V_s}s}}{d_1 e^{\frac{\Delta x_1}{a-V_s}s} - f_1 e^{-\frac{\Delta x_1}{a+V_s}s}}. \tag{3.38}$$

The other transfer functions have the same form as Equations 3.37 and 3.38. The terms $g(s)$, c, d and f can be written in a general form which simply use different rows of $\mathbf{X}$ from Equation 3.21. This was implemented in the code in Appendix F.

The hyperbolic sines and cosines in the non-zeroth harmonics, Equation 3.37, can be expanded using Taylor series expansions to find approximations in terms of pole and zero polynomials:

$$\frac{\sinh\left(b\sqrt{y(s)}\right)}{\sqrt{y(s)}} \cong \left.\frac{\sinh\left(b\sqrt{y(s)}\right)}{\sqrt{y(s)}}\right|_{s=0} + \frac{1}{2y(s)}\left(b\cosh\left(b\sqrt{y(s)}\right) - \frac{\sinh\left(b\sqrt{y(s)}\right)}{\sqrt{y(s)}}\right)\left.\frac{dy(s)}{ds}\right|_{s=0} s + \cdots$$

$$\cosh\left(b\sqrt{y(s)}\right) \cong \cosh\left(b\sqrt{y(0)}\right) + \frac{b}{2}\frac{\sinh\left(b\sqrt{y(0)}\right)}{\sqrt{y(0)}}\left.\frac{dy(s)}{ds}\right|_{s=0} s + \cdots$$

The zeroth harmonic terms are simple exponentials in s, which can be easily expanded using Taylor series expansions without worrying about branch cuts.

For the state-space realization of Equation 3.28, we can use the controllable canonical state-space form for a two state approximation:

$$\left[\begin{array}{cc|c} 0 & 1 & 0 \\ -p_0 & -p_1 & 1 \\ \hline a_0 & a_1 & a_2 \\ b_0 & b_1 & b_2 \\ c_0 & c_1 & c_2 \\ d_0 & d_1 & d_2 \end{array}\right] \tag{3.39}$$

for the input $(\tilde{B}_{LE1} - w_{11}P_{loss,1} - s_{11}u_1)$, from Equation 3.28. To complete the inlet condition approximation, the pressure loss and control feed-through terms of Equation 3.28 must be added.

3.2.3.2 Exit Condition

The exit condition from Section 2.2.5 is used to find transfer functions from $\tilde{C}_{TEK}(s), \tilde{E}_{TEK}(s), \delta\alpha_K(s)$ and $u_{K+1}(s)$ (when control is applied at the exit) to $\tilde{B}_{TEK}(s)$. This is a dual of the inlet conditions. The exit condition can be written in a general vector form to accommodate both the non-zeroth and the zeroth harmonics:

$$\mathbf{T}(s)\mathbf{V}_{K+1}(x_{ex},s)\begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_{K+1} = 0. \tag{3.40}$$

If control is applied in the exit duct at the trailing edge of the last blade row, we can use the trailing edge boundary condition (Equation 2.35), the actuation equation (Equation 2.50) and the deviation state (Equation 3.11) to yield

$$\mathbf{T}(s)\mathbf{V}_{K+1}(x_{ex},s)\mathbf{V}_{K+1}^{-1}(x_{TEK},s)\left(\mathbf{V}_{TK+1}^{-1}\begin{bmatrix} \mathbf{B}_{TK}\mathbf{M} & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} \tilde{B}_{TE} \\ \tilde{C}_{TE} \\ \tilde{E}_{TE} \\ \delta\alpha \end{bmatrix}_K + \mathbf{J}_{K+1}^{-1}\mathbf{b}_{K+1}u_{K+1}\right) = 0$$

This can be simplified by defining two additional matrices ($\mathbf{Y}$ and $\mathbf{r}$):

$$\mathbf{T}(s)\mathbf{V}_{K+1}(x_{ex},s)\mathbf{V}_{K+1}^{-1}(x_{TEK},s)\begin{bmatrix}\mathbf{Y} & \mathbf{r}\end{bmatrix}\begin{bmatrix}\tilde{B}_{TEK}\\ \tilde{C}_{TEK}\\ \tilde{E}_{TEK}\\ \delta\alpha_K\\ u_{K+1}\end{bmatrix}=0. \tag{3.41}$$

For non-zeroth harmonics, the matrix product, $\mathbf{T}(s)\mathbf{V}_{K+1}(x_{ex},s)\mathbf{V}_{K+1}^{-1}(x_{TEK},s)$, is simply the matrix product, $\mathbf{V}_{1r,K+1}(x_{ex},s)\mathbf{V}_{K+1}^{-1}(x_{TEK},s)$, which contains four elements that can be found in Appendix B. We can now solve Equation 3.41 for $\tilde{B}_{TEK}(s)$ in a general form:

$$\boxed{\tilde{B}_{TEK}=\frac{1}{u_{11}(s)}\begin{bmatrix}u_{12}(s) & u_{13}(s) & u_{14}(s) & u_{15}(s)\end{bmatrix}\begin{bmatrix}\tilde{C}_{TEK}\\ \tilde{E}_{TEK}\\ \delta\alpha_K\\ u_{K+1}\end{bmatrix}} \tag{3.42}$$

where the elements u_{1m} are not the same as in the preceding section.

One of the transfer functions for the non-zeroth harmonics, using Appendix B, is

$$\frac{\tilde{B}_{TEK}(s)}{\tilde{C}_{TEK}(s)}=-\frac{y_{12}\cosh\left(b\sqrt{y(s)}\right)+\left(-y_{32}\gamma\left(\frac{sr}{a}+jnM_\theta\right)+y_{42}\gamma jnM_x\right)\frac{\sinh\left(b\sqrt{y(s)}\right)}{\sqrt{y(s)}}}{y_{11}\cosh\left(b\sqrt{y(s)}\right)+\left(-y_{31}\gamma\left(\frac{sr}{a}+jnM_\theta\right)+y_{41}\gamma jnM_x\right)\frac{\sinh\left(b\sqrt{y(s)}\right)}{\sqrt{y(s)}}}. \tag{3.43}$$

These functions can be expanded as before using Taylor series expansions to lead to a state-space approximation. One of the transfer functions for the zeroth harmonic, using Appendix B, is

$$\frac{\tilde{B}_{TEK}(s)}{\tilde{C}_{TEK}(s)} = -\frac{g_{b1}(s)e^{\frac{\Delta x_K}{a-V_t}s} + g_{b2}(s)e^{-\frac{\Delta x_K}{a+V_t}s} + g_{b3}(s)e^{-\frac{\Delta x_K}{V_t}s}}{g_{c1}(s)e^{\frac{\Delta x_K}{a-V_t}s} + g_{c2}(s)e^{-\frac{\Delta x_K}{a+V_t}s} + g_{c3}(s)e^{-\frac{\Delta x_K}{V_t}s}}. \tag{3.44}$$

The dual of the controllable canonical state-space realization, Equation 3.39, is then used for one output and four inputs of Equation 3.42:

$$\left[\begin{array}{cc|cccc} 0 & -p_0 & a_0 & b_0 & c_0 & d_0 \\ 1 & -p_1 & a_1 & b_1 & c_1 & d_1 \\ \hline 0 & -1 & -a_2 & -b_2 & -c_2 & -d_2 \end{array}\right] \tag{3.45}$$

where a two state approximation has been shown as an example. These transfer functions have different poles than the inlet conditions.

3.2.4 Formation of the Model

The separate pieces of the model are now be connected together to form the state-space matrices. The delayed states, Equation 3.3, and their approximations represented by Equation 3.6, the lag states, Equations 3.12 and 3.14, and the boundary conditions, Equation 3.16 can be used to connect the interior of the compressor. The inlet and exit conditions, Equation 3.28 and 3.42, respectively, and their approximations can then be used to complete the model. Appendix C gives an example of the intermediate state-space matrices (of this section) using first order approximations.

First, an output vector will be defined for each blade row and inlet and exit duct. This is the set of variables which are the result of the delays and the lags for the blade rows and the outputs of the approximations of the end ducts:

$$\mathbf{y}_k = \begin{bmatrix} \tilde{B}_{LE} \\ \tilde{C}_{TE} \\ \tilde{E}_{TE} \\ P_{loss} \\ \delta\alpha \end{bmatrix}_k , \quad \mathbf{y}_{out} = \left[\tilde{B}_{TE}\right]_K , \quad \mathbf{y}_{in} = \begin{bmatrix} \tilde{C}_{LE} \\ \tilde{E}_{LE} \\ (1+s\tau)P_{loss} \\ (1+s\tau)\delta\alpha \end{bmatrix}_1 \tag{3.46}$$

where two outputs of the inlet duct are inputs to the pressure loss and deviation lags. The approximations to Equations 3.3, 3.12, 3.14 and 3.16 can be collected into the form:

$$\begin{aligned} \dot{\mathbf{x}}_k &= \mathbf{A}_k\mathbf{x}_k + \mathbf{B}_{k-}\mathbf{y}_{k-1} + \mathbf{B}_k\mathbf{y}_k + \mathbf{B}_{k+}\mathbf{y}_{k+1} + \mathbf{B}_{uk}\mathbf{u}_k + \mathbf{B}_{uk+}\mathbf{u}_{k+1} \\ \mathbf{y}_k &= \mathbf{C}_k\mathbf{x}_k + \mathbf{D}_{k-}\mathbf{y}_{k-1} + \mathbf{D}_k\mathbf{y}_k + \mathbf{D}_{k+}\mathbf{y}_{k+1} + \mathbf{D}_{uk}\mathbf{u}_k + \mathbf{D}_{uk+}\mathbf{u}_{k+1} \end{aligned} \tag{3.47}$$

where the state vector $\mathbf{x}$ for each blade row contains the Padé states and the lag states. The equations can be written in this form because the only effects on the states of a particular blade row are from the states of the adjacent blade rows and the control in the adjacent gaps. Once the matrices of Equation 3.47 are formed, and similar representations of the inlet and exit duct equations (Equations 3.28 and 3.42) are included, the entire system of output equations must be solved simultaneously:

$$\tilde{\mathbf{D}} \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \vdots \\ \mathbf{y}_K \\ \mathbf{y}_{out} \\ \mathbf{y}_{in} \end{bmatrix} = \tilde{\mathbf{C}} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_K \\ \mathbf{x}_{out} \\ \mathbf{x}_{in} \end{bmatrix} + \tilde{\mathbf{D}}_u \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_K \\ u_{K+1} \end{bmatrix} \tag{3.48}$$

where

$$\tilde{\mathbf{C}} = \begin{bmatrix} \mathbf{C}_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \mathbf{C}_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \ddots & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{C}_K & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{C}_{out} & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathbf{C}_{in} \end{bmatrix}$$

$$\tilde{\mathbf{D}} = \begin{bmatrix} \mathbf{I}-\mathbf{D}_1 & -\mathbf{D}_{1+} & 0 & 0 & 0 & -\mathbf{D}_{1-} \\ -\mathbf{D}_{2-} & \mathbf{I}-\mathbf{D}_2 & \ddots & 0 & 0 & 0 \\ 0 & \ddots & \ddots & -\mathbf{D}_{K-1+} & 0 & 0 \\ 0 & 0 & -\mathbf{D}_{K-} & \mathbf{I}-\mathbf{D}_K & -\mathbf{D}_{K+} & 0 \\ 0 & 0 & 0 & -\mathbf{D}_{out-} & \mathbf{I}-\mathbf{D}_{out} & 0 \\ -\mathbf{D}_{in+} & 0 & 0 & 0 & 0 & \mathbf{I}-\mathbf{D}_{in} \end{bmatrix}$$

$$\tilde{\mathbf{D}}_u = \begin{bmatrix} \mathbf{D}_{u1} & \mathbf{D}_{u1+} & 0 & 0 & 0 \\ 0 & \mathbf{D}_{u2} & \mathbf{D}_{u2+} & 0 & 0 \\ 0 & 0 & \ddots & \ddots & 0 \\ 0 & 0 & 0 & \mathbf{D}_{uK} & \mathbf{D}_{uK+} \\ 0 & 0 & 0 & 0 & \mathbf{D}_{uout} \\ \mathbf{D}_{uin} & 0 & 0 & 0 & 0 \end{bmatrix}$$

The solution to Equation 3.48 is:

$$\mathbf{y} = \tilde{\mathbf{D}}^{-1}\tilde{\mathbf{C}}\mathbf{x} + \tilde{\mathbf{D}}^{-1}\tilde{\mathbf{D}}_u\mathbf{u} \tag{3.49}$$

Now, the differential equation in Equation 3.47 can be resolved:

$$\begin{bmatrix} \dot{\mathbf{x}}_1 \\ \dot{\mathbf{x}}_2 \\ \vdots \\ \dot{\mathbf{x}}_K \end{bmatrix} = \tilde{\mathbf{A}} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_K \end{bmatrix} + \tilde{\mathbf{B}} \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \vdots \\ \mathbf{y}_K \end{bmatrix} + \tilde{\mathbf{B}}_u \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_K \\ u_{K+1} \end{bmatrix} \tag{3.50}$$

where

$$\tilde{\mathbf{A}} = \begin{bmatrix} \mathbf{A}_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \mathbf{A}_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \ddots & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{A}_K & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{A}_{out} & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathbf{A}_{in} \end{bmatrix}$$

$$\tilde{\mathbf{B}} = \begin{bmatrix} \mathbf{B}_1 & \mathbf{B}_{1+} & 0 & 0 & 0 & \mathbf{B}_{1-} \\ \mathbf{B}_{2-} & \mathbf{B}_2 & \ddots & 0 & 0 & 0 \\ 0 & \ddots & \ddots & \mathbf{B}_{K-1+} & 0 & 0 \\ 0 & 0 & \mathbf{B}_{K-} & \mathbf{B}_K & \mathbf{B}_{K+} & 0 \\ 0 & 0 & 0 & \mathbf{B}_{out-} & \mathbf{B}_{out} & 0 \\ \mathbf{B}_{in+} & 0 & 0 & 0 & 0 & \mathbf{B}_{in} \end{bmatrix}$$

$$\tilde{\mathbf{B}}_u = \begin{bmatrix} \mathbf{B}_{u1} & \mathbf{B}_{u1+} & 0 & 0 & 0 \\ 0 & \mathbf{B}_{u2} & \mathbf{B}_{u2+} & 0 & 0 \\ 0 & 0 & \ddots & \ddots & 0 \\ 0 & 0 & 0 & \mathbf{B}_{uK} & \mathbf{B}_{uK+} \\ 0 & 0 & 0 & 0 & \mathbf{B}_{uout} \\ \mathbf{B}_{uin} & 0 & 0 & 0 & 0 \end{bmatrix}$$

so the solution to Equation 3.50 is

$$\dot{\mathbf{x}} = \left(\tilde{\mathbf{A}} + \tilde{\mathbf{B}}\tilde{\mathbf{D}}^{-1}\tilde{\mathbf{C}}\right)\mathbf{x} + \left(\tilde{\mathbf{B}}_u + \tilde{\mathbf{B}}\tilde{\mathbf{D}}^{-1}\tilde{\mathbf{D}}_u\right)\mathbf{u}. \tag{3.51}$$

Finally, the measurement outputs are defined to be the static pressure in each gap including the static pressure at the leading edge of the first blade row and the static pressure at the trailing edge of the last blade row. This set is a subset of the previous total output vector:

$$\frac{\delta \mathbf{P}}{\mathbf{P}} = \mathbf{C}_y \mathbf{y} = \mathbf{C}_y \tilde{\mathbf{D}}^{-1}\tilde{\mathbf{C}}\mathbf{x} + \mathbf{C}_y \tilde{\mathbf{D}}^{-1}\tilde{\mathbf{D}}_u \mathbf{u} \tag{3.52}$$

3.3 State-Space Model Including Gaps

3.3.1 Boundary Conditions and Gap Approximations for Non-Zeroth Harmonics

The gaps are approximated starting with Equation 3.7, using the definitions in Equations 3.1 and 3.2:

$$\mathbf{B}_{Lk}\mathbf{M}\begin{bmatrix}\tilde{B}_{LE}\\ \tilde{C}_{LE}\\ \tilde{E}_{LE}\end{bmatrix}_k = (\mathbf{V}_{Lk}+\tfrac{1}{1+s\tau}\mathbf{P}_k)\mathbf{V}_k(x_{LEk},s)\mathbf{V}_k^{-1}(x_{TEk-1},s)\bullet$$
$$\left(\mathbf{V}_{Tk}^{-1}\left(\begin{matrix}\mathbf{B}_{Tk-1}\mathbf{M}\begin{bmatrix}\tilde{B}_{TE}\\ \tilde{C}_{TE}\\ \tilde{E}_{TE}\end{bmatrix}_{k-1}\\ -\tfrac{1}{1+s\tau}\mathbf{D}_{k-1}\mathbf{V}_{k-1}(x_{LEk-1},s)\begin{bmatrix}B\\ C\\ D\\ E\end{bmatrix}_{k-1}\end{matrix}\right)+\mathbf{J}_k^{-1}\mathbf{b}_k u_k\right). \tag{3.53}$$

A definition for the deviation state can be written

$$\delta\alpha_k = -\tfrac{1}{1+s\tau}\mathbf{D}_{k,4r}\mathbf{V}_k(x_{LEk},s)\mathbf{V}_k^{-1}(x_{TEk-1},s)\bullet$$
$$\left(\mathbf{V}_{Tk}^{-1}\left(\begin{matrix}\mathbf{B}_{Tk-1}\mathbf{M}\begin{bmatrix}\tilde{B}_{TE}\\ \tilde{C}_{TE}\\ \tilde{E}_{TE}\end{bmatrix}_{k-1}\\ -\tfrac{1}{1+s\tau}\mathbf{D}_{k-1}\mathbf{V}_{k-1}(x_{LEk-1},s)\begin{bmatrix}B\\ C\\ D\\ E\end{bmatrix}_{k-1}\end{matrix}\right)+\mathbf{J}_k^{-1}\mathbf{b}_k u_k\right) \tag{3.54}$$

This can be manipulated into a recursion to yield

$$\delta\alpha_k = -\tfrac{1}{1+s\tau}\mathbf{D}_{k,4r}\mathbf{V}_k(x_{LEk},s)\mathbf{V}_k^{-1}(x_{TEk-1},s)\bullet \left(\mathbf{V}_{Tk}^{-1}\begin{bmatrix} & 0 \\ \mathbf{B}_{Tk-1}\mathbf{M} & 0 \\ & 0 \\ & 1 \end{bmatrix}\begin{bmatrix}\tilde{B}_{TE} \\ \tilde{C}_{TE} \\ \tilde{E}_{TE} \\ \delta\alpha\end{bmatrix}_{k-1} + \mathbf{J}_k^{-1}\mathbf{b}_k u_k\right) \tag{3.55}$$

A matrix definition will be made to simplify things:

$$\mathbf{Y}_{k-1} = \mathbf{V}_{Tk}^{-1}\begin{bmatrix} & 0 \\ \mathbf{B}_{Tk-1}\mathbf{M} & 0 \\ & 0 \\ & 1 \end{bmatrix}. \tag{3.56}$$

Equation 3.53, using the deviation equation (Equation 3.54), and Equation 3.56, becomes

$$\mathbf{B}_{Lk}\mathbf{M}\begin{bmatrix}\tilde{B}_{LE} \\ \tilde{C}_{LE} \\ \tilde{E}_{LE}\end{bmatrix}_k = (\mathbf{V}_{Lk} + \tfrac{1}{1+s\tau}\mathbf{P}_k)\mathbf{V}_k(x_{LEk},s)\mathbf{V}_{k-1}^{-1}(x_{TEk-1},s)\bullet \left(\mathbf{Y}_{k-1}\begin{bmatrix}\tilde{B}_{TE} \\ \tilde{C}_{TE} \\ \tilde{E}_{TE} \\ \delta\alpha\end{bmatrix}_{k-1} + \mathbf{J}_k^{-1}\mathbf{b}_k u_k\right). \tag{3.57}$$

Equation 3.55, using Equation 3.56, becomes

$$\delta\alpha_k = -\tfrac{1}{1+s\tau}\mathbf{D}_{k,4r}\mathbf{V}_k(x_{LEk},s)\mathbf{V}_k^{-1}(x_{TEk-1},s)\bullet \left(\mathbf{Y}_{k-1}\begin{bmatrix}\tilde{B}_{TE} \\ \tilde{C}_{TE} \\ \tilde{E}_{TE} \\ \delta\alpha\end{bmatrix}_{k-1} + \mathbf{J}_k^{-1}\mathbf{b}_k u_k\right). \tag{3.58}$$

Similar to the deviation state, Equation 3.58, the pressure loss state is defined as

$$P_{loss,k} = \tfrac{1}{1+s\tau}\mathbf{P}_{k,3r}\mathbf{V}_k(x_{LEk},s)\mathbf{V}_k^{-1}(x_{TEk-1},s)\bullet \left(\mathbf{Y}_{k-1}\begin{bmatrix}\tilde{B}_{TE}\\ \tilde{C}_{TE}\\ \tilde{E}_{TE}\\ \delta\alpha\end{bmatrix}_{k-1} + \mathbf{J}_k^{-1}\mathbf{b}_k u_k\right). \tag{3.59}$$

Using the definitions:

$$\mathbf{X}_k = \mathbf{M}^{-1}\mathbf{B}_{Lk}^{-1}\mathbf{V}_{Lk} \tag{3.60}$$

$$\mathbf{w}_k = \mathbf{M}^{-1}\mathbf{B}_{Lk}^{-1}\begin{bmatrix}0\\0\\1\end{bmatrix} \tag{3.61}$$

Equations 3.57, 3.58 and 3.59 become

$$\begin{bmatrix}\tilde{B}_{LE}\\ \tilde{C}_{LE}\\ \tilde{E}_{LE}\\ (1+s\tau)P_{loss}\\ (1+s\tau)\delta\alpha\end{bmatrix}_k = \begin{bmatrix}\mathbf{X}_k\\ \mathbf{P}_{k,3r}\\ \mathbf{D}_{k,4r}\end{bmatrix}\mathbf{V}(x_{LEk},s)\mathbf{V}^{-1}(x_{TEk-1},s)\begin{bmatrix}\mathbf{Y}_{k-1} & \mathbf{J}_k^{-1}\mathbf{b}_k\end{bmatrix}\begin{bmatrix}\tilde{B}_{TEk-1}\\ \tilde{C}_{TEk-1}\\ \tilde{E}_{TEk-1}\\ \delta\alpha_{k-1}\\ u_k\end{bmatrix} + \begin{bmatrix}\mathbf{w}_k\\ 0\\ 0\end{bmatrix}P_{loss,k} \tag{3.62}$$

This can be rewritten as

$$\begin{bmatrix} \tilde{B}_{LE} \\ \tilde{C}_{LE} \\ \tilde{E}_{LE} \\ (s+\frac{1}{\tau})P_{loss} \\ (s+\frac{1}{\tau})\delta\alpha \end{bmatrix}_k = \mathbf{U}_k(s) \begin{bmatrix} \tilde{B}_{TEk-1} \\ \tilde{C}_{TEk-1} \\ \tilde{E}_{TEk-1} \\ P_{loss,k} \\ \delta\alpha_{k-1} \\ u_k \end{bmatrix} \tag{3.63}$$

where each element of the $\mathbf{U}_k(s)$ matrix is a linear combination of elements of $\mathbf{V}_k(x_{LEk},s)\mathbf{V}_k^{-1}(x_{TEk-1},s)$ which are expanded in Appendix B, except for the terms multiplying $P_{loss,k}$. Equation 3.63 will be manipulated to solve for the variables which are inputs to the blade row Padé approximations and lags:

$$\boxed{\begin{bmatrix} \tilde{B}_{TEk-1} \\ \tilde{C}_{LEk} \\ \tilde{E}_{LEk} \\ (s+\frac{1}{\tau})P_{loss,k} \\ (s+\frac{1}{\tau})\delta\alpha_k \end{bmatrix} = \mathbf{H}_{U,k}(s) \begin{bmatrix} \tilde{B}_{LEk} \\ \tilde{C}_{TEk-1} \\ \tilde{E}_{TEk-1} \\ P_{loss,k} \\ \delta\alpha_{k-1} \\ u_k \end{bmatrix}} \tag{3.64}$$

where

$$\mathbf{H}_{U,k}(s) = \frac{1}{u_{11}} \begin{bmatrix} -1 & 0 & 0 & 0 & 0 \\ -u_{21} & u_{11} & 0 & 0 & 0 \\ -u_{31} & 0 & u_{11} & 0 & 0 \\ -u_{41} & 0 & 0 & u_{11} & 0 \\ -u_{51} & 0 & 0 & 0 & u_{11} \end{bmatrix} \begin{bmatrix} -1 & u_{12} & u_{13} & w_{11} & u_{15} & u_{16} \\ 0 & u_{22} & u_{23} & w_{21} & u_{25} & u_{26} \\ 0 & u_{32} & u_{33} & w_{31} & u_{35} & u_{36} \\ 0 & u_{42} & u_{43} & 0 & u_{45} & u_{46} \\ 0 & u_{52} & u_{53} & 0 & u_{55} & u_{56} \end{bmatrix}. \tag{3.65}$$

The elements u_{lm} are not the same as those of Section 3.2.3.1. Equation 3.64 can be written in the form:

$$
\begin{aligned}
\tilde{B}_{TEk-1} &= f_b(\tilde{B}_{LEk} - w_{11}P_{loss,k}, \tilde{C}_{TEk}, \tilde{E}_{TEk}, \delta\alpha_{k-1}, u_k) \\
\tilde{C}_{LEk} &= f_c(\tilde{B}_{LEk} - w_{11}P_{loss,k}, \tilde{C}_{TEk}, \tilde{E}_{TEk}, \delta\alpha_{k-1}, u_k) + w_{21}P_{loss,k} \\
\tilde{E}_{LEk} &= f_e(\tilde{B}_{LEk} - w_{11}P_{loss,k}, \tilde{C}_{TEk}, \tilde{E}_{TEk}, \delta\alpha_{k-1}, u_k) + w_{31}P_{loss,k} \\
(s+\tfrac{1}{\tau})P_{loss,k} &= f_p(\tilde{B}_{LEk} - w_{11}P_{loss,k}, \tilde{C}_{TEk}, \tilde{E}_{TEk}, \delta\alpha_{k-1}, u_k) \\
(s+\tfrac{1}{\tau})\delta\alpha_k &= f_\alpha(\tilde{B}_{LEk} - w_{11}P_{loss,k}, \tilde{C}_{TEk}, \tilde{E}_{TEk}, \delta\alpha_{k-1}, u_k)
\end{aligned}
\tag{3.66}
$$

This is a five input-five output system. From Equation 3.65 we can see that the transfer function denominators are the upper left element of $\mathbf{U}_k(s)$, and the numerators are either linear combinations of elements or linear combinations of two element products of $\mathbf{U}_k(s)$. As was stated before, the elements of $\mathbf{U}_k(s)$ (defined in Equation 3.62) are linear combinations of elements of $\mathbf{V}_k(x_{LEk}, s)\mathbf{V}_k^{-1}(x_{TEk-1}, s)$ which are expanded in Appendix B. Each of these elements can be approximated by Taylor series expansions. When products of elements occur in the numerator of these transfer functions, the individual Taylor series expansions are simply multiplied to get the approximation. As for the pressure loss state and the deviation state, the resulting transfer function is fed directly into the lag state.

Some of the elements in Appendix B.1 (the non-zeroth harmonics) have frequency dependent denominators which arise from the matrix inverse. This denominator contains an unstable zero which must cancel out in each case, since the duct is not a physically unstable system by itself. The zeros can be removed by taking the Taylor series in the axial coordinate term, b (see Equation 3.35). The (3,1) element will be used as an example. If we take the (3,1) element and rearrange, we get

$$
\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(3,1)} = \frac{h(s)}{\gamma\left(y(s)-n^2\right)}\left\{\begin{array}{l} n^2 M_x \cosh\left(b\sqrt{y(s)}\right) \\ -(\frac{sr}{a} + jnM_\theta)\sqrt{y(s)}\sinh\left(b\sqrt{y(s)}\right) \\ -n^2 M_x e^{-\frac{(sr/a + jnM_\theta)}{M_x}b} \end{array}\right\}
\tag{3.67}
$$

In this notation, it is not clear how the denominator function $y(s)-n^2$ is canceled. The bracketed term is a function of Δx, or b, and doesn't have the denominator as a factor. The motivation for the Taylor series in b is simply to have terms with separate dependence on b and s. So, if we expand the bracketed terms in b, then we will see the cancellation:

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(3.1)} = \frac{h(s)}{\gamma\left(y(s)-n^2\right)}\left\{\begin{array}{l} n^2M_x\left(1+\frac{1}{2}y(s)b^2+\frac{1}{24}y^2(s)b^4+\cdots\right) \\ -(\frac{sr}{a}+jnM_\theta)\left(y(s)b+\frac{1}{6}y^2(s)b^3+\frac{1}{120}y^3(s)b^5+\cdots\right) \\ -n^2M_x\left(1-\phi(s)b+\frac{1}{2}\phi^2(s)b^2-\frac{1}{6}\phi^3(s)b^3+\frac{1}{24}\phi^4(s)b^4\right) \end{array}\right\} \tag{3.68}$$

where

$$\phi(s) = \frac{sr/a + jnM_\theta}{M_x}. \tag{3.69}$$

In Equation 3.68, we see that each term of each series has a function of s multiplying a function of b, the desired separation. Combining terms of similar order of b, and doing some algebraic reduction, we get

$$
\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(3,1)} = \frac{h(s)}{\gamma\left(y(s)-n^2\right)} \left\{ \begin{array}{l} -\left(\dfrac{sr}{a}+jnM_\theta\right)\left(y(s)-n^2\right)b \\[2ex] -\dfrac{n^2(1-M_x^2)}{M_x^2}\left(y(s)-n^2\right)\dfrac{b^2}{2} \\[2ex] -\dfrac{\left(\dfrac{sr}{a}+jnM_\theta\right)}{M_x^2}\left[M_x^2\left(\dfrac{sr}{a}+jnM_\theta\right)^2 - n^2(1-M_x^2)^2\right]\bullet \\[1ex] \qquad\qquad\qquad\qquad\qquad\qquad \left(y(s)-n^2\right)\dfrac{b^3}{6} \\[2ex] + \quad \cdots \end{array} \right\}
$$

where we now see the cancellation of $y(s)-n^2$ term by term. This allows us to expand all the terms of the matrix product $\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}$ in Taylor series expansions of s without denominator terms.

3.3.2 Boundary Conditions and Gap Approximations for the Zeroth Harmonic

For the zeroth harmonic, we can directly approximate the dynamics in the gap because there is no problem with branch cuts. The solution is in the form of time delays. Therefore, we will start with some definitions in the gap:

$$
\begin{aligned}
B_{LE}(s) &= B(s)e^{\alpha(s)x_{LE}} \\
C_{LE}(s) &= C(s)e^{\beta(s)x_{LE}} \\
D_{LE}(s) &= D(s)e^{\chi(s)x_{LE}} \\
E_{LE}(s) &= E(s)e^{\chi(s)x_{LE}}
\end{aligned} \tag{3.71}
$$

and similarly

$$
\begin{aligned}
B_{TE}(s) &= B(s)e^{\alpha(s)x_{TE}} \\
C_{TE}(s) &= C(s)e^{\beta(s)x_{TE}} \\
D_{TE}(s) &= D(s)e^{\chi(s)x_{TE}} \\
E_{TE}(s) &= E(s)e^{\chi(s)x_{TE}}
\end{aligned}
\tag{3.72}
$$

The relations between Equations 3.71 and 3.72 become

$$
\boxed{
\begin{aligned}
B_{TE}(s) &= e^{-s\Delta T_{B_k}} B_{LE}(s) \\
C_{LE}(s) &= e^{-s\Delta T_{C_k}} C_{TE}(s) \\
D_{LE}(s) &= e^{-s\Delta T_{D_k}} D_{TE}(s) \\
E_{LE}(s) &= e^{-s\Delta T_{E_k}} E_{TE}(s)
\end{aligned}
}
\tag{3.73}
$$

where the time delays are

$$
\begin{aligned}
\Delta T_{B_k} &= \tfrac{\Delta x}{a-V_x} \\
\Delta T_{C_k} &= \tfrac{\Delta x}{a+V_x} \\
\Delta T_{D_k} &= \tfrac{\Delta x}{V_x} \\
\Delta T_{E_k} &= \tfrac{\Delta x}{V_x}
\end{aligned}
.
\tag{3.74}
$$

The right hand side of Equation 3.73 will be the 'inputs' to the gap solutions and the left hand side will be the 'outputs'. The leading edge boundary condition, Equation 2.34, using Equation 3.71 is

$$
\left(\mathbf{V}_{Lk} + \tfrac{1}{1+s\tau}\mathbf{P}_k\right)\mathbf{G}_k \begin{bmatrix} B_{LE} \\ C_{LE} \\ D_{LE} \\ E_{LE} \end{bmatrix}_k = \mathbf{B}_{Lk}\mathbf{M} \begin{bmatrix} \tilde{B}_{LE} \\ \tilde{C}_{LE} \\ \tilde{E}_{LE} \end{bmatrix}_k
\tag{3.75}
$$

where $\mathbf{M}$ is defined in Equation 3.10, and $\mathbf{G}$ is the analogous matrix for the gaps:

$$\mathbf{G}_k = \begin{bmatrix} \frac{\gamma}{1-M_t} & \frac{\gamma}{1+M_t} & 0 & 0 \\ \frac{1}{1-M_t} & \frac{1}{1+M_t} & 0 & 1 \\ -\frac{1}{1-M_t} & \frac{1}{1+M_t} & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{3.76}$$

The pressure loss state is defined, to remove the s-dependence in 3.75, as

$$\boxed{P_{loss,k} = \tfrac{1}{1+s\tau}\mathbf{P}_{k,3r}\mathbf{G}_k \begin{bmatrix} B_{LE} \\ C_{LE} \\ D_{LE} \\ E_{LE} \end{bmatrix}_k} \tag{3.77}$$

The deviation state will be defined similarly as

$$\boxed{\delta\alpha_k = -\tfrac{1}{1+s\tau}\mathbf{D}_{k,4r}\mathbf{G}_k \begin{bmatrix} B_{LE} \\ C_{LE} \\ D_{LE} \\ E_{LE} \end{bmatrix}_k} \tag{3.78}$$

so Equation 3.75 with 3.77 and 3.78 can be rewritten as

$$\begin{bmatrix} & 0 \\ \mathbf{V}_{Lk}\mathbf{G}_k & 0 \\ & 1 \\ \frac{1}{\tau}\mathbf{P}_{k,3r}\mathbf{G}_k & 0 \\ \frac{1}{\tau}\mathbf{D}_{k,4r}\mathbf{G}_k & 0 \end{bmatrix} \begin{bmatrix} B_{LE} \\ C_{LE} \\ D_{LE} \\ E_{LE} \\ P_{loss} \end{bmatrix}_k = \begin{bmatrix} \mathbf{B}_{Lk}\mathbf{M} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \begin{bmatrix} \tilde{B}_{LE} \\ \tilde{C}_{LE} \\ \tilde{E}_{LE} \\ (s+\frac{1}{\tau})P_{loss} \\ (s+\frac{1}{\tau})\delta\alpha \end{bmatrix}_k . \tag{3.79}$$

This constant transformation can be manipulated to solve for the 'inputs' of the blade row, the gap and the lags:

$$\boxed{\begin{bmatrix} B_{LE} \\ \tilde{C}_{LE} \\ \tilde{E}_{LE} \\ (s+\frac{1}{\tau})P_{loss} \\ (s+\frac{1}{\tau})\delta\alpha \end{bmatrix}_k = \mathbf{K}_{LEk} \begin{bmatrix} \tilde{B}_{LE} \\ C_{LE} \\ D_{LE} \\ E_{LE} \\ P_{loss} \end{bmatrix}_k} \tag{3.80}$$

The trailing edge boundary condition, Equation 2.35, can be written using Equation 3.72, the definitions for **M** and **G** (Equations 3.10 and 3.75), and the actuation relation, Equation 2.50, (for actuation at the trailing edge) as

$$\mathbf{B}_{Tk}\mathbf{M}\begin{bmatrix} \tilde{B}_{TE} \\ \tilde{C}_{TE} \\ \tilde{E}_{TE} \end{bmatrix}_k + \mathbf{J}^{-1}_{k+1}\mathbf{b}_{k+1}u_{k+1} = \mathbf{V}_{Tk+1}\mathbf{G}_{k+1}\begin{bmatrix} B_{TE} \\ C_{TE} \\ D_{TE} \\ E_{TE} \end{bmatrix}_{k+1} + \frac{1}{1+s\tau}\mathbf{D}_k\mathbf{G}_k\begin{bmatrix} B_{LE} \\ C_{LE} \\ D_{LE} \\ E_{LE} \end{bmatrix}_k \tag{3.81}$$

The deviation state defined in Equation 3.78 removes the s-dependence in Equation 3.81 to yield

$$\begin{bmatrix} \mathbf{B}_{Tk}\mathbf{M} & \begin{matrix} 0 \\ 0 \\ 0 \\ 1 \end{matrix} & \mathbf{J}^{-1}_{k+1}\mathbf{b}_{k+1} \end{bmatrix} \begin{bmatrix} \tilde{B}_{TEk} \\ \tilde{C}_{TEk} \\ \tilde{E}_{TEk} \\ \delta\alpha_k \\ u_{k+1} \end{bmatrix} = \mathbf{V}_{Tk+1}\mathbf{G}_{k+1}\begin{bmatrix} B_{TE} \\ C_{TE} \\ D_{TE} \\ E_{TE} \end{bmatrix}_{k+1} \tag{3.82}$$

This constant transformation can also be manipulated to solve for the 'inputs' to the blade row and the gap:

$$\boxed{\begin{bmatrix} \tilde{B}_{TEk} \\ C_{TEk+1} \\ D_{TEk+1} \\ E_{TEk+1} \end{bmatrix} = \mathbf{K}_{TEk} \begin{bmatrix} B_{TEk+1} \\ \tilde{C}_{TEk} \\ \tilde{E}_{TEk} \\ \delta\alpha_k \\ u_{k+1} \end{bmatrix}}. \tag{3.83}$$

Equation 3.73 defines the transfer functions within a gap for the zeroth harmonic. These will be approximated like the blade row solutions with Padé approximations. Equations 3.77 and 3.81 define the pressure loss and deviation lag states within the blade row. Equations 3.79 and 3.83 define how these states connect with the adjacent blade row states.

3.3.3 Formation of the Model for Non-Zeroth Harmonics

The separate pieces of the non-zeroth harmonic model will now be connected together to form the state-space matrices. The approximated equations (also repeated in Appendix C) are the blade row solutions, Equations 3.3, the end conditions, Equations 3.28 and 3.42, and the gap approximations (including lag states), Equation 3.64.

Similar to the model without gaps, a solution will be sought in the form of Equation 3.47, except with a larger set of states and outputs for each blade row. An example of all of the intermediate state-space matrices (of this section) is given in Appendix C. The gap states will be added to the previous blade row states:

$$
\mathbf{y}_k = \begin{bmatrix} \tilde{B}_{LE} \\ \tilde{C}_{TE} \\ \tilde{E}_{TE} \\ P_{loss} \\ \delta\alpha \\ \tilde{B}_{TE} \\ \tilde{C}_{LE} \\ \tilde{E}_{LE} \\ (1+s\tau)P_{loss} \\ (1+s\tau)\delta\alpha \end{bmatrix}_k, \quad \mathbf{y}_{out} = \left[\tilde{B}_{TE}\right]_K, \quad \mathbf{y}_{in} = \begin{bmatrix} \tilde{C}_{LE} \\ \tilde{E}_{LE} \\ (1+s\tau)P_{loss} \\ (1+s\tau)\delta\alpha \end{bmatrix}_1 \tag{3.84}
$$

The same procedure as without the gaps is then followed to get the state-space matrices.

3.3.4 Formation of the Model for the Zeroth Harmonic

The separate pieces of the zeroth harmonic model will now be connected together to form the state-space matrices. The approximated equations (also repeated in Appendix C) are: the blade row solutions, Equation 3.3, the end conditions, Equations 3.28 and 3.42, the gap solutions, Equation 3.73, the lag states, Equations 3.77 and 3.78, and the boundary conditions, Equations 3.80 and 3.83.

Similar to the model for non-zeroth harmonics, a solution will be sought in the form of Equation 3.47, except with a larger set of states and outputs for each blade row. An example of all of the intermediate state-space matrices (of this section) is given in Appendix C. The gap states will be added to the previous blade row states:

$$
\mathbf{y}_k = \begin{bmatrix} \tilde{B}_{LE} \\ \tilde{C}_{TE} \\ \tilde{E}_{TE} \\ P_{loss} \\ \delta\alpha \\ B_{TE} \\ C_{LE} \\ D_{LE} \\ E_{LE} \end{bmatrix}_k , \quad \mathbf{y}_{out} = \left[\tilde{B}_{TE} \right]_K , \quad \mathbf{y}_{in} = \begin{bmatrix} \tilde{C}_{LE} \\ \tilde{E}_{LE} \\ (1+s\tau)P_{loss} \\ (1+s\tau)\delta\alpha \end{bmatrix}_1 \tag{3.85}
$$

The same procedure as without the gaps is then followed to get the state-space matrices.

3.4 Validation of the Approximate Models

Two models have been derived for approximating the analytical input-output model of Chapter 2. The first model, assuming no gap dynamics, is expected to result in a smaller, but less accurate model than the second including gap dynamics. In fact, the model including the gap dynamics should match the analytical model perfectly at zero frequency and lose accuracy slowly at higher frequencies, since all of the dynamics are approximated and all the approximations are expanded about s=0.

3.4.1 Compressor Geometry and Data

The compressor to be modeled throughout this thesis is a 3 stage high speed axial compressor with a row of inlet guide vanes (IGVs), such as shown in Figure 2.1. The correlations and geometry were supplied by the manufacturer, and a mean-line prediction code was used to get the steady state flow conditions in each blade row and gap (i.e. mean pressures, velocities, etc.). The inlet and exit duct lengths were not known, so these were chosen to be lengths of .3 times the radius of the compressor. The plenum geometry downstream of the compressor was not known, either, so a value of $V_{plen} / \dot{m}_{ex} = .5$ was assumed.

As the flow coefficient, ϕ, is decreased at 100% design speed, the first harmonic to reach neutral stability is $n=1$ at a frequency of about 40% of the rotor frequency, Ω. The next *harmonics* to reach neutral stability are the zeroth, then the second and third, etc. The first *higher frequency* modes to reach neutral stability are a pair of modes of the zeroth harmonic, which reach neutral stability before any of the third harmonic modes.

3.4.2 Order of Approximation for Different Length Elements

The 'incompressible' rotating stall mode for the first harmonic has a frequency less than the rotor frequency, Ω. Similarly, the higher harmonics usually have frequencies on the order of n times the frequency of the first harmonic. If it is desired to control the first three harmonics, then the approximate model should have small errors below three times the rotor frequency. If the entire model needs a specific accuracy, then certainly each element should retain that accuracy. An example for each type of element, blade row, inter-blade row gap, and exit duct, will be given to examine the required order of approximation.

In the blade row solution, there are three separate delays which need to be approximated. The longest delay is the entropy mode since it travels with the fluid velocity. The pressure modes travel at the sound velocity plus or minus the fluid velocity, hence the associated delays are shorter. The longest delay, which incorporates the longest blade length with the slowest velocity, in the 100% design speed case at flow coefficient of .4506 is .52 when normalized to the rotor speed, Ω. Figure 3.1 shows a comparison of phases for different order Padé approximations versus the pure time delay. It is clear that in the frequency range less than four times the rotor frequency, a second order approximation is sufficient, yielding only a few degrees of error for this worst case delay, with the error growing to 75 degrees at 10Ω.

In the exit duct, the order of approximation will depend heavily on the length of the duct. In the absence of information about the true compressor geometry, a length of

.3 times the radius was chosen. There are a number of transfer functions related to the exit duct, all with the same poles. Figure 3.2 shows one of these transfer functions compared with different orders of Taylor series approximations for numerator and denominator. In this case, a third order approximation may be sufficient, but a fourth order approximation is clearly better at frequencies up to 8Ω, so it will be used. Figure 3.3 shows the same transfer functions when the length of the duct is assumed to be equal to the radius. The fourth order approximation is only good up to about 2.5Ω, which steadily gets worse as the duct length is increased.

Each gap, like the end ducts, is modeled with a number of transfer functions all having the same poles. The gaps differ in length, so Figure 3.4 shows one of these transfer functions versus different order approximations in the longest gap. The magnitudes of a constant approximation and the first order approximation are both fairly good, with about .2 dB in error at 10Ω, but the phase of the constant approximation is off by 5 degrees at 3Ω, while the first order approximation is nearly perfect up to 10Ω. The first order approximation will be used in the model including the gaps.

3.4.3 Eigenvalue Comparisons

One of the potential application of the approximate model is to compute eigenvalues of the compressor to determine the neutral stability operating point. Using the orders of approximation from the preceding section, the eigenvalues of the system can be computed as the eigenvalues of the state matrix for the approximate model, or at the zeros of the eigenvalue equation, Equation 2.49, of the truth model. Finding the zeros of the eigenvalue equation requires a numerical procedure which has difficulties away from the frequency axis of the s-domain. For this reason, the comparisons to follow will only examine eigenvalues near the s-domain frequency axis.

The model without gaps is compared against the analytical, or 'truth', model in Figures 3.5, 3.6, and 3.7 for the zeroth, first and second harmonics. This shows good

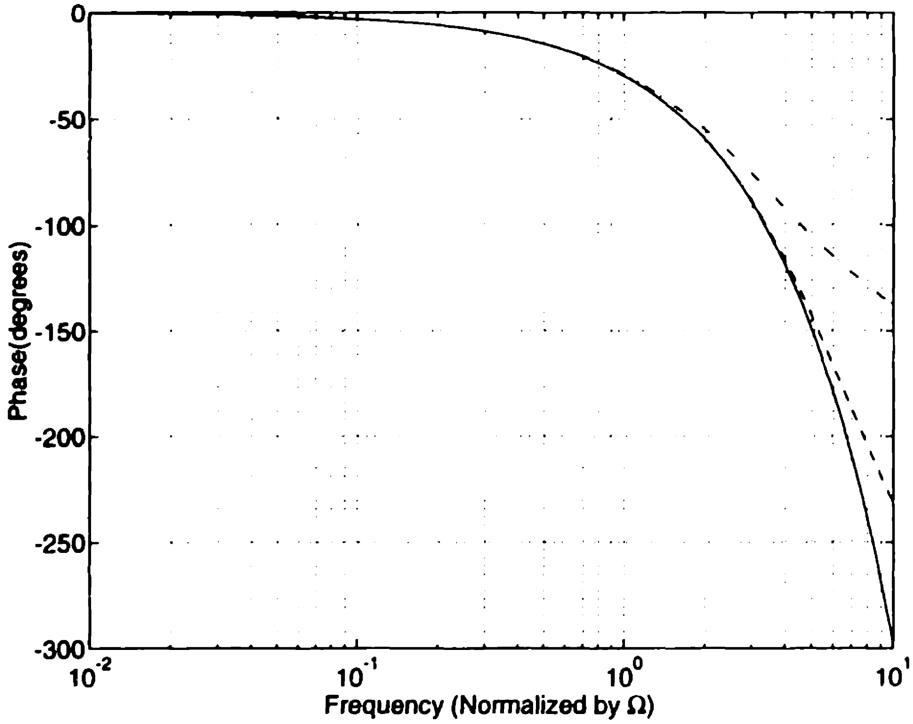

Figure 3.1 - Phases of pure delay (solid) and first (dash-dotted) and second (dashed) order Padé approximations.

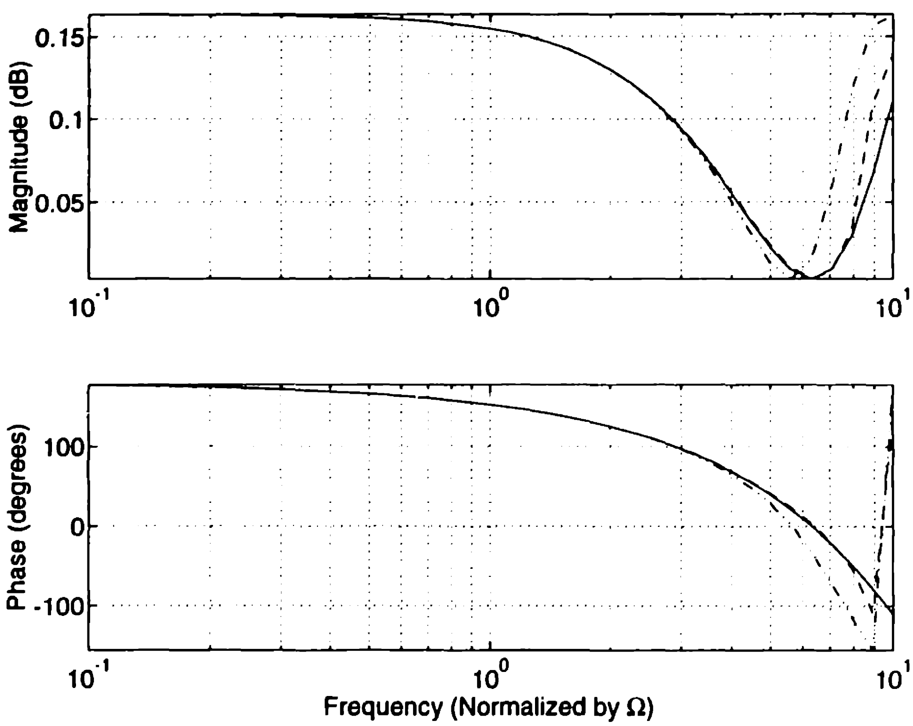

Figure 3.2 - Magnitude and phase of truth model (solid) exit duct transfer function and third (dash-dotted) and fourth (dashed) order approximations for $\Delta x_e = .3r$.

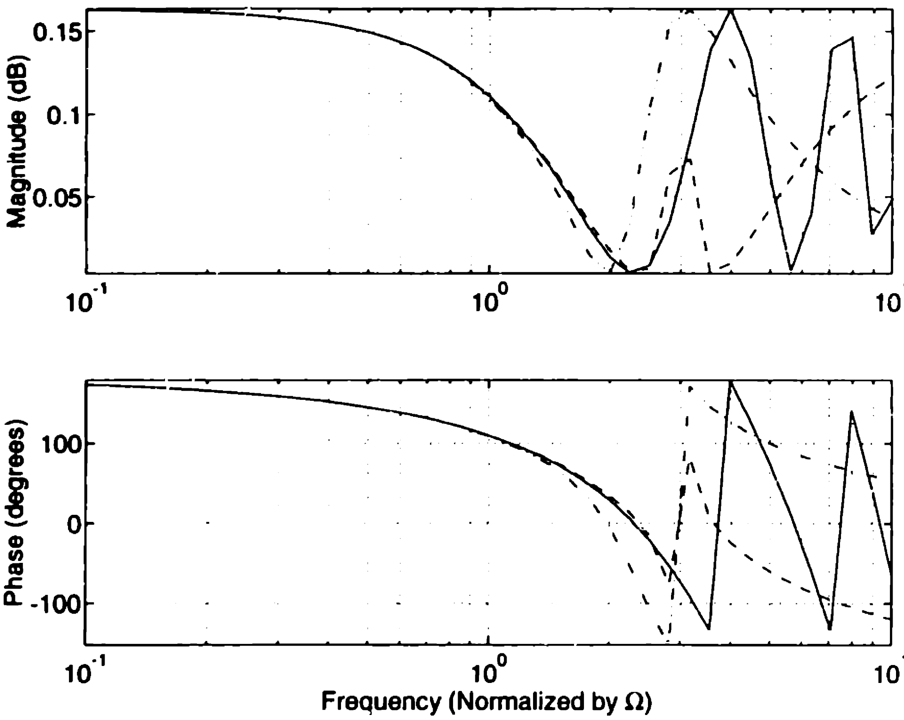

Figure 3.3 - Magnitude and phase of truth model (solid) exit duct transfer function and third (dash-dotted) and fourth (dashed) order approximations for $\Delta x_e = r$

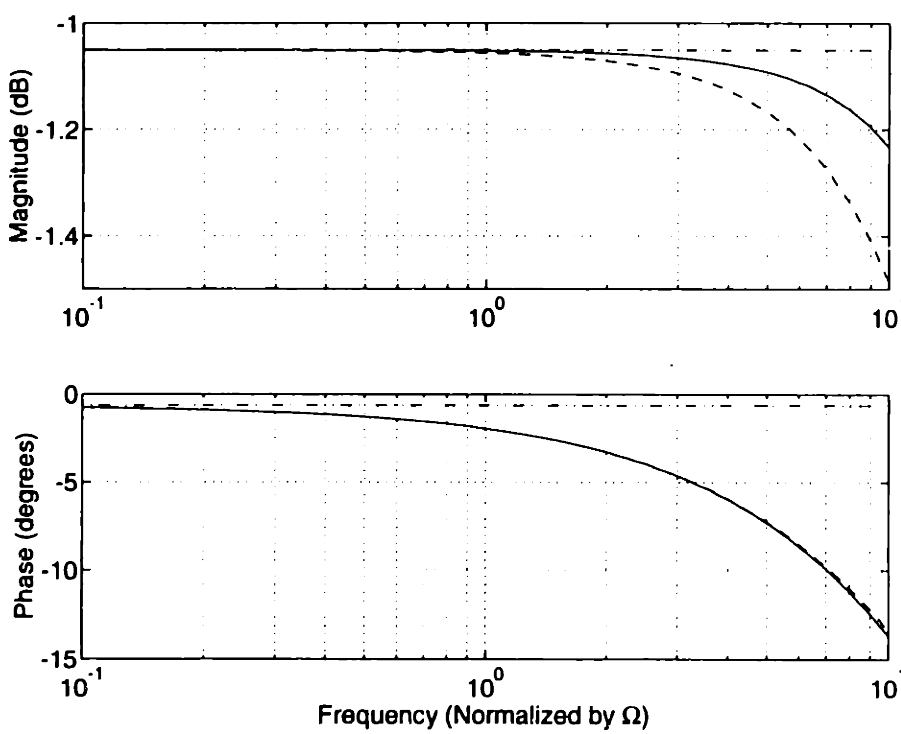

Figure 3.4 - Magnitude and phase of truth model (solid) gap transfer function and constant (dash-dotted) and first order (dashed) approximations.

is a pole near neutral stability. These errors validate the use of control systems based on this model for application to the truth model.

is a pole near neutral stability. These errors validate the use of control systems based on this model for application to the truth model.

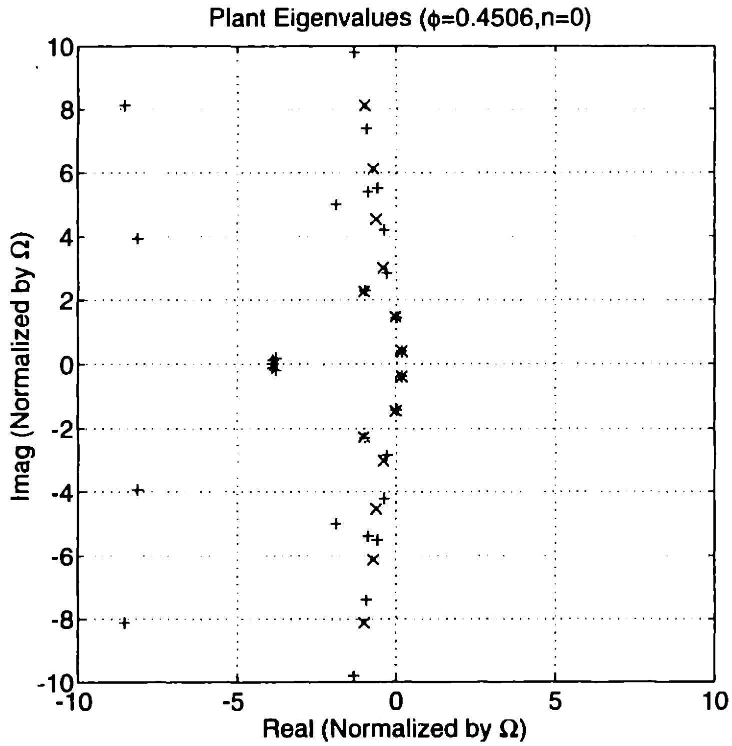

Figure 3.5 - Eigenvalue comparison between truth model (x) and approximate model (+), n=0, no gap approximation.

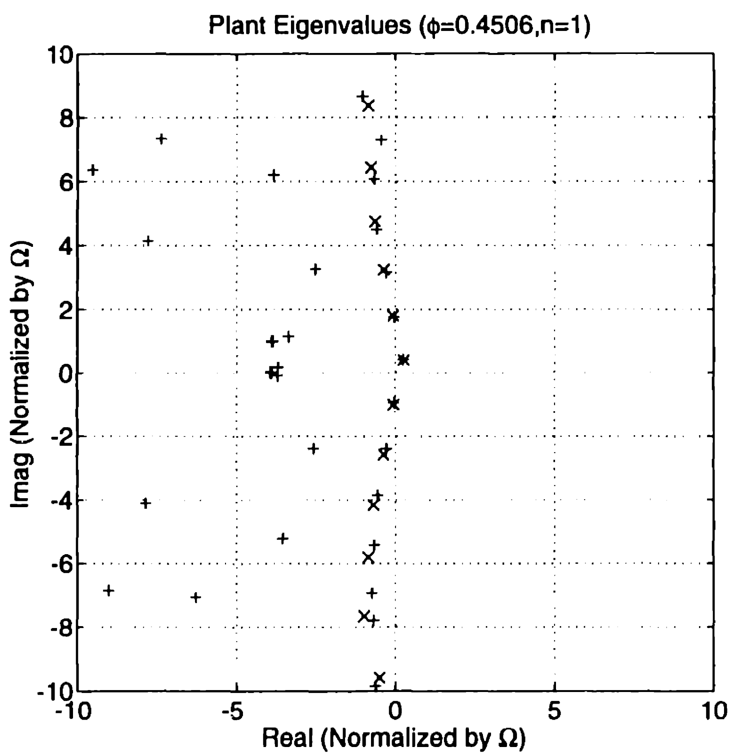

Figure 3.6 - Eigenvalue comparison between truth model (x) and approximate model (+), n=1, no gap approximation.

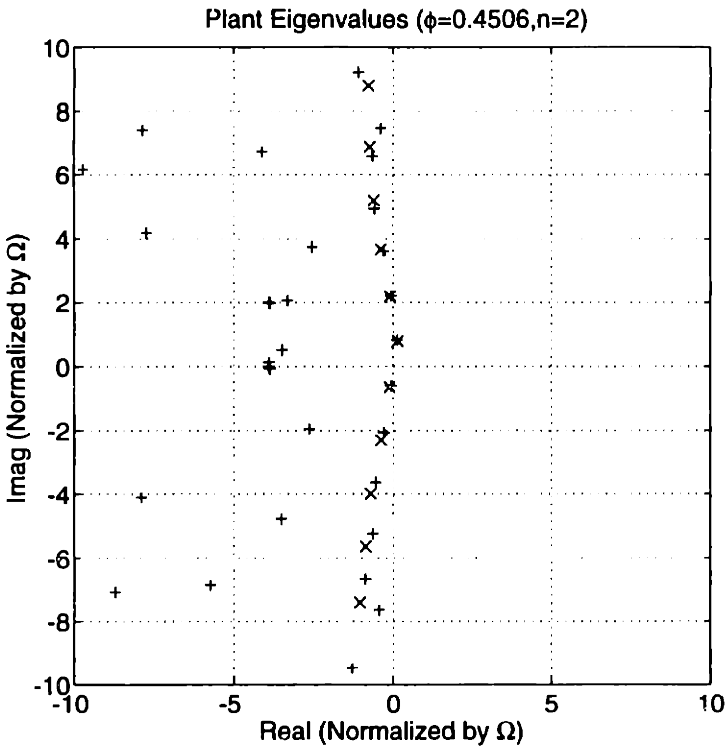

Figure 3.7 - Eigenvalue comparison between truth model (x) and approximate model (+), n=2, no gap approximation.

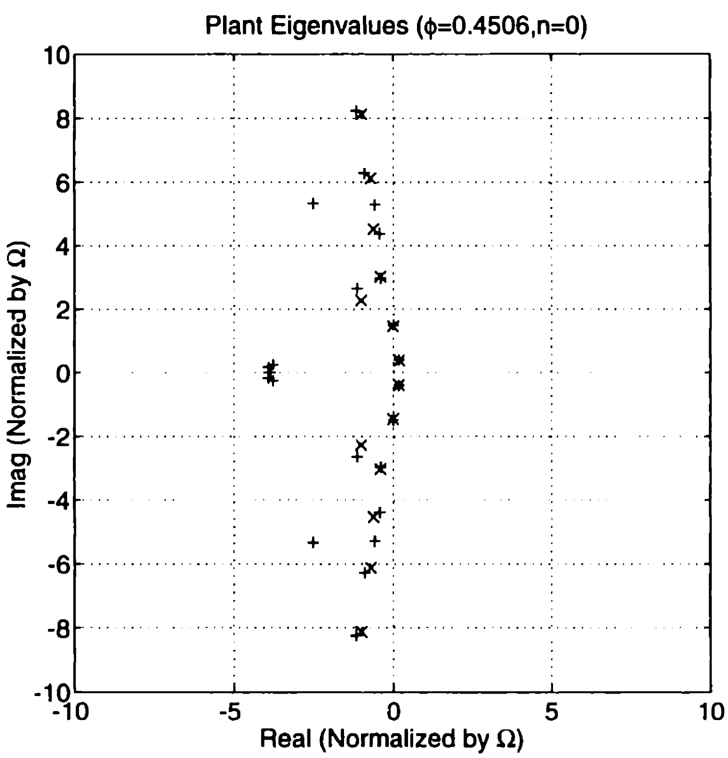

Figure 3.8 - Eigenvalue comparison between truth model (x) and approximate model (+), n=0, gap approximation.

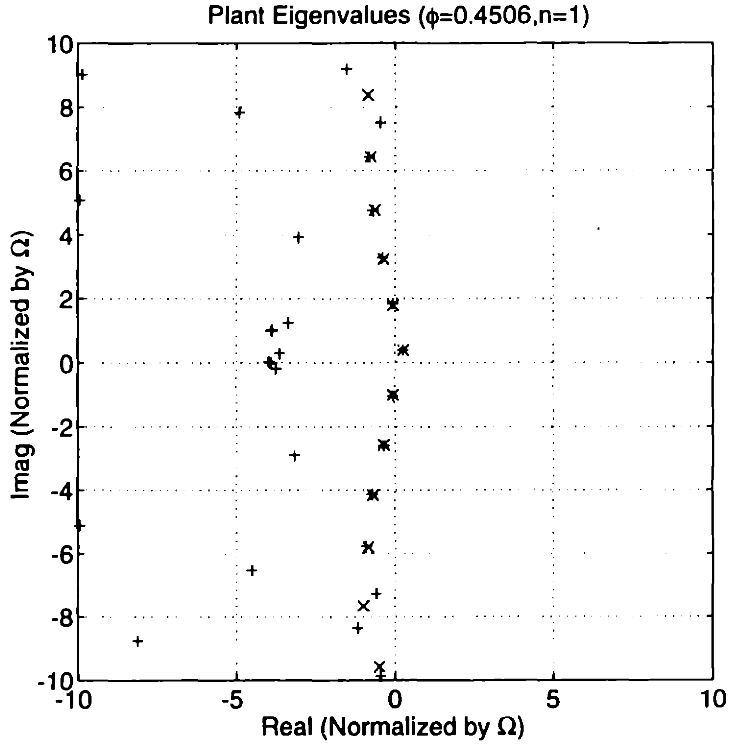

Figure 3.9 - Eigenvalue comparison between truth model (x) and approximate model (+), n=1, gap approximation.

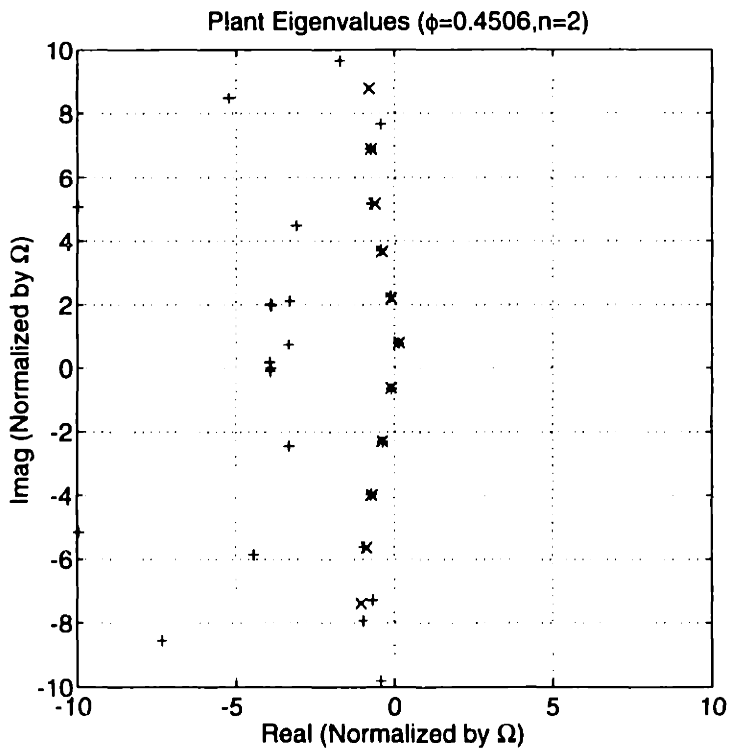

Figure 3.10 - Eigenvalue comparison between truth model (x) and approximate model (+), n=2, gap approximation.

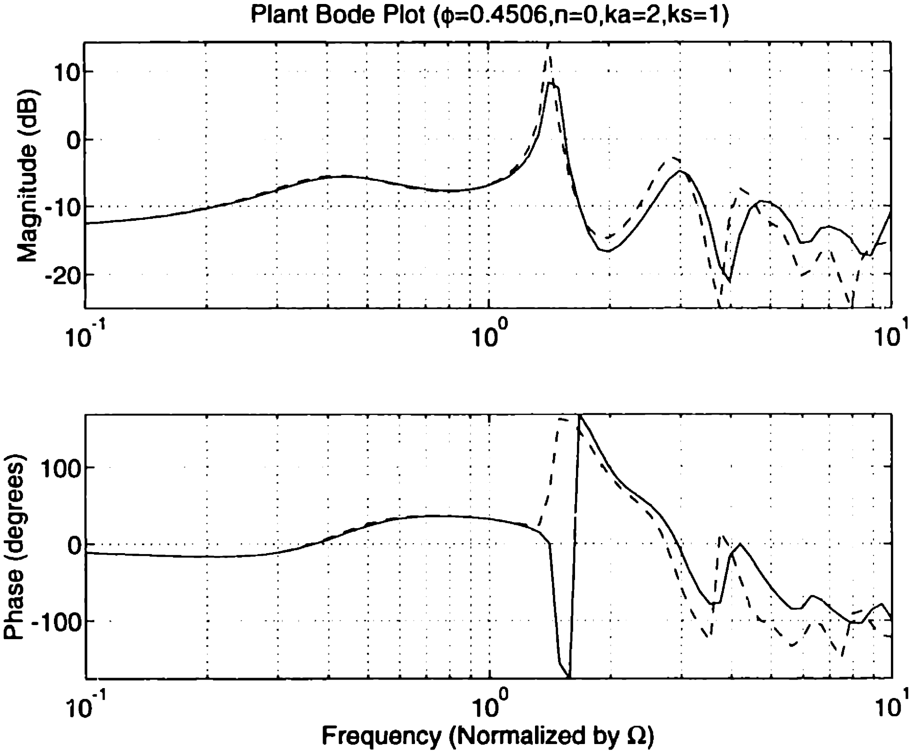

Figure 3.11 - Open-loop frequency response comparison between truth model (solid) and approximate model (dashed), n=0, no gap approximation.

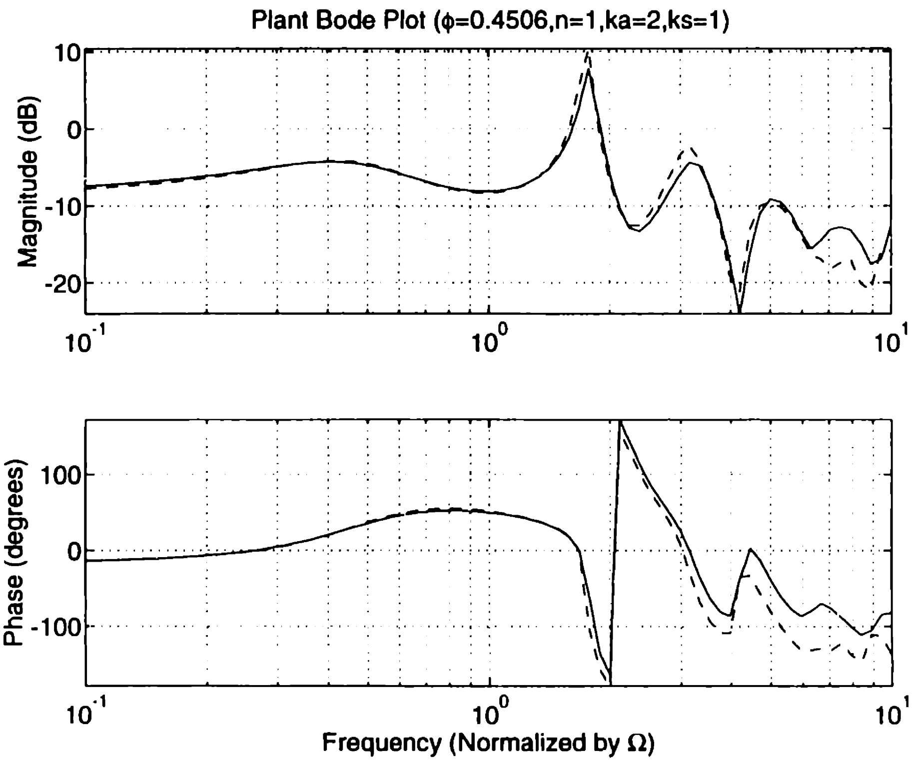

Figure 3.12 - Open-loop frequency response comparison between truth model (solid) and approximate model (dashed), n=1, no gap approximation.

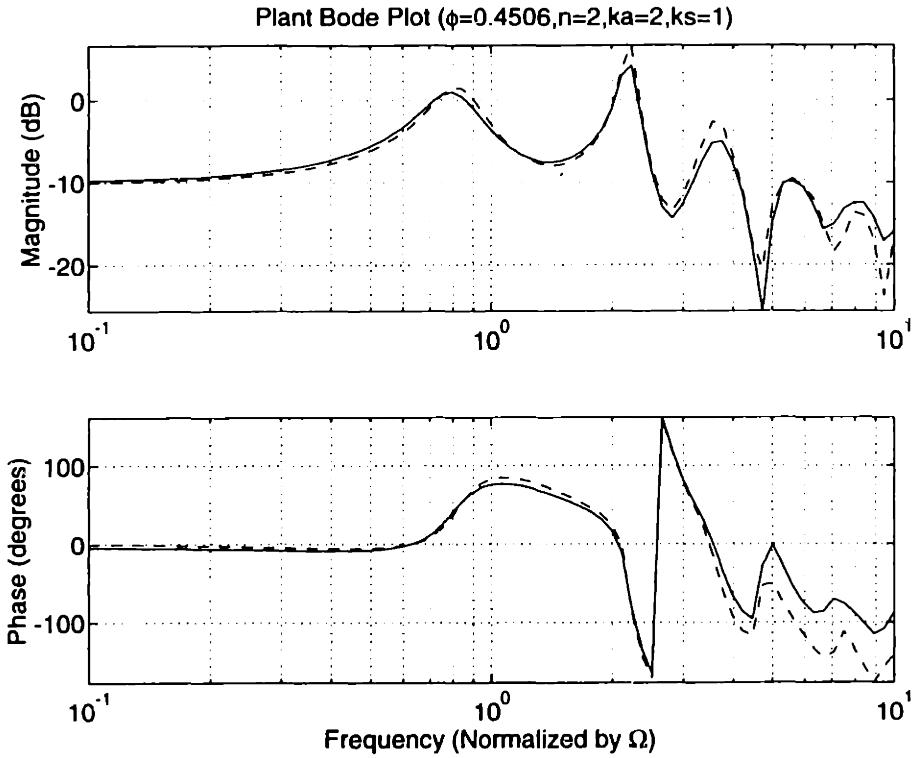

Figure 3.13 - Open-loop frequency response comparison between truth model (solid) and approximate model (dashed), n=2, no gap approximation.

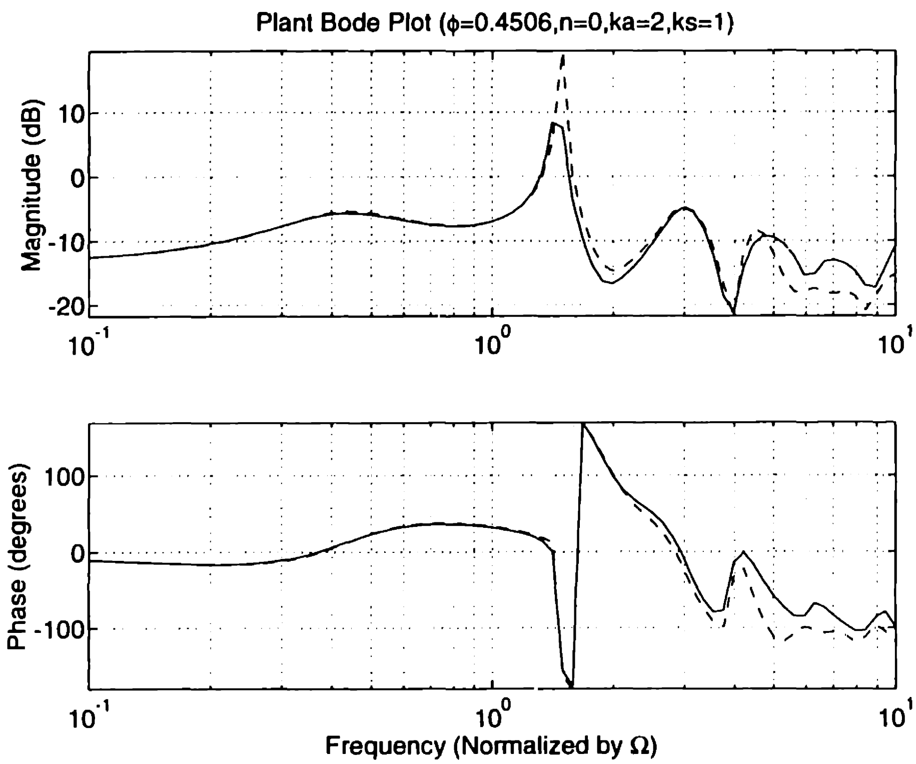

Figure 3.14 - Open-loop frequency response comparison between truth model (solid) and approximate model (dashed), n=0, gap approximation.

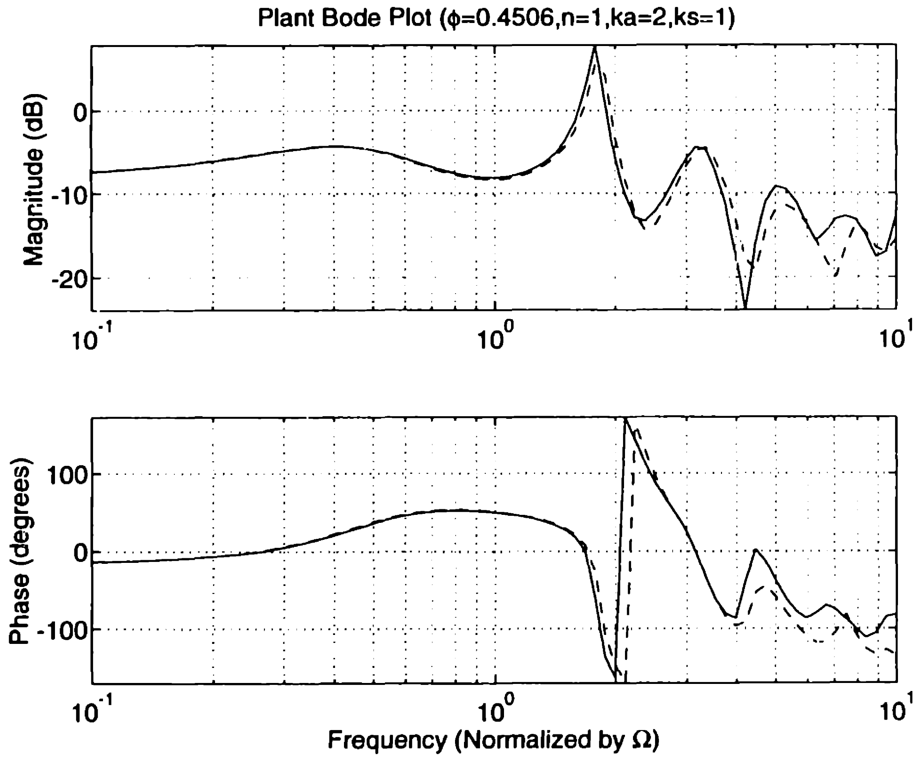

Figure 3.15 - Open-loop frequency response comparison between truth model (solid) and approximate model (dashed), n=1, gap approximation.

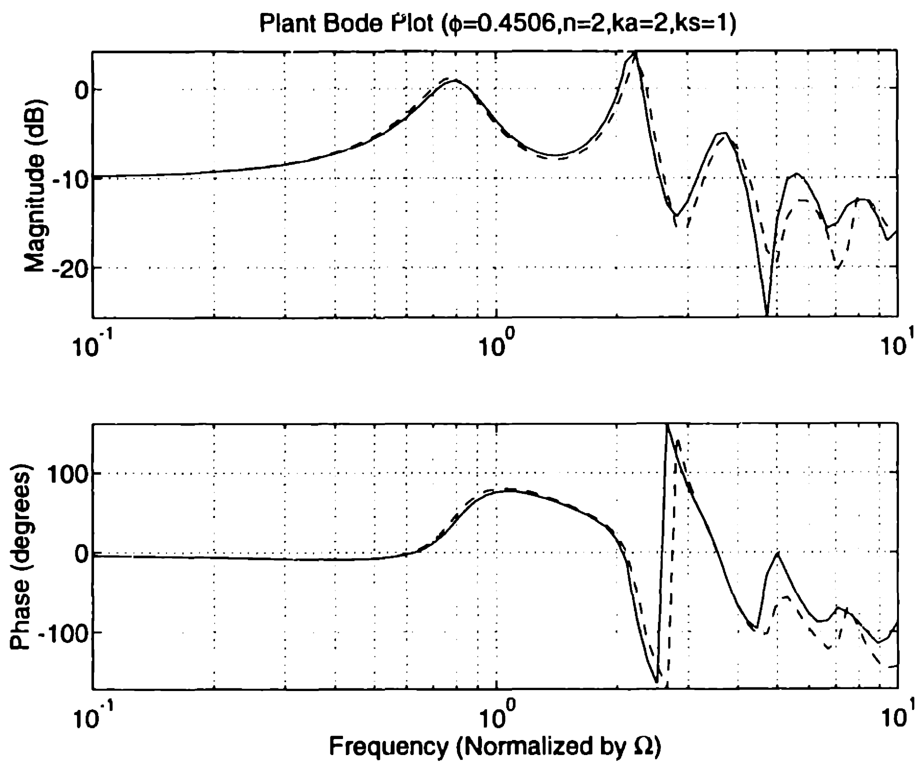

Figure 3.16 - Open-loop frequency response comparison between truth model (solid) and approximate model (dashed), n=2, gap approximation.

Chapter 4

Control Law Specifications and Design Methodologies

This chapter develops frequency domain specifications for a control system design and associated mean-square performance measures. These are described in Section 4.1. Two different control system design methodologies are presented to accomplish these goals: the *Linear Quadratic-Gaussian* (LQG) design technique (described in Section 4.2), which can be used to minimize a mean-square cost function, but will be modified to try to satisfy the frequency domain specifications; and the H_∞ design technique (described in Section 4.3), which can be used directly to satisfy the specifications (if they are feasible for a configuration), but has no provision for minimizing mean-square costs.

These two design techniques are applied to single-input/single-output systems in Chapter 5 to find the 'best' control configuration in terms of minimum cost functions and minimum flow coefficients at which the specifications can still be met. The configurations to be tested are sensor and actuator placement and types.

4.1 Design Specifications and Cost Functions

There are two types of specifications which the control system design will have to meet. The first type is frequency domain bounds on the complementary sensitivity and the sensitivity transfer functions, described in Sections 4.1.1 and 4.1.2, respectively. A controller meeting these specifications will have robustness to high frequency errors, such as model errors, insensitivity to noise at the rotor frequency, and guaranteed gain and phase margins from classical control theory.

The second type of specification, described in Section 4.1.3, and perhaps the most difficult for which to account, is robustness to operating point changes. The range of operating points contains both stable and unstable plants, so specifying bounds on the system to account for this is not an easy proposition. Fortunately, experience in designing compensators has shown that many control configurations are 'naturally' robust to operating point changes in the following way: if a control law is designed for a particular unstable operating point, then the system will remain stable for any more stable operating point (i.e. a higher flow coefficient) than the designed point. This feature of many configurations is the motivation for designing a compensator at the lowest possible flow coefficient for which the frequency domain specifications can still be met. This robustness will have to be examined on a case by case basis. If a configuration is not naturally robust to operating point changes, then *ad-hoc* bounds can be used to try to improve robustness.

The cost function to be minimized, described in Section 4.1.4, is the mean-squared sum of the static pressure perturbations in each inter-blade row gap and duct. Another factor affecting the configuration selection is the mean-squared actuator activity. Driving these functions is measurement noise, which is white, and process noise, which is white except for a spike at the rotor frequency, Ω.

4.1.1 Complementary Sensitivity Bounds

This section develops a frequency domain bound on the complementary sensitivity function that the compensator must satisfy. First, a general form of an H_∞ complementary sensitivity constraint is given, then a particular function is chosen as a specification.

The specification on the complementary sensitivity function is the closed-loop *bandwidth*. The bandwidth of the system is commonly defined as the highest frequency where the complementary sensitivity function is equal to -3 dB. Although there are many reasons to desire a high bandwidth, it is desired to have the bandwidth of the system limited for practical purposes. One such reason to have the bandwidth limited is modeling error at high frequencies. A reduced order compensator will be implemented in practice, so it is desired for the system to have small response at higher frequencies so that a low-order compensator model can accurately represent the full order compensator. A second reason is the impact of sensor and actuator dynamics. If particular sensors and actuators are under consideration for use in active control, then their dynamics will impact the closed-loop behavior of the system. Sensor and actuator dynamics are not modeled in the present study, but their effect can be minimized if the bandwidth of the system is kept below the frequency of all sensor and actuator dynamics.

The general bandwidth constraint can be specified as follows: for a particular cross-over ω_c and a 20 dB/decade roll-off, yielding a bandwidth of 1.4 ω_c, the bound on the complementary sensitivity can be expressed as

$$\left|\frac{j\omega}{\omega_c} C(j\omega)\right| < 1 \qquad \forall \omega \tag{4.1}$$

where the complementary sensitivity, $C(s)$, is defined in terms of the plant, $G(s)$, and the compensator, $K(s)$:

$$C(s) = \frac{G(s)K(s)}{1 + G(s)K(s)}. \tag{4.2}$$

Equation 4.1 is equivalent to the H_∞-norm constraint:

$$\|W_2 C\|_\infty < 1 \tag{4.3}$$

where the weighting function, $W_2(s)$, will be defined as the complementary sensitivity weighting function.

The specified bandwidth must be chosen high enough to allow stabilization of all the spatial Fourier harmonics of interest. The highest harmonic to be considered is the third harmonic, which has a rotating stall frequency of about 1.1Ω. Therefore, a cross-over of 2Ω should be adequate. The weighting function, $W_2(s)$, therefore becomes

$$W_2(s) = \frac{s / 2\Omega}{(1 + s / 100\Omega)} \tag{4.4}$$

with a high frequency pole included to make the transfer function proper. If robustness to operating point changes (described in Section 4.1.3) is included as a complementary sensitivity constraint, then $W_2(s)$ will still have this cross-over frequency, but its low frequency characteristics will be modified.

4.1.2 Sensitivity Bounds

This section will develop a frequency domain bound on the sensitivity function that the compensator must satisfy. First, some limits on this bound will be described to motivate the idea that not any sensitivity weighting function can be satisfied by any plant. Next, the relation between sensitivity and gain and phase margins will be described, and a specification will be chosen based on these ideas.

The magnitude of the sensitivity transfer function

$$S(s) = \frac{1}{1 + G(s)K(s)} \tag{4.5}$$

is the inverse of the distance from the critical point of the Nyquist plot. So, the desire to keep the distance from the critical point large translates into keeping the sensitivity small. For example, if the sensitivity is always less than 2, then the Nyquist contour will always remain at least .5 away from the critical point. The margin specified must be based upon practical design considerations.

4.1.2.1 Limits to Achievable Sensitivity

The first limit on sensitivity arises from its relation to the complementary sensitivity function. As a result of the relation

$$C(s) + S(s) = 1, \tag{4.6}$$

when one function approaches zero, then the other must approach unity. For instance, beyond the bandwidth of the system defined in Section 4.1.1, the complementary sensitivity approaches zero and, from Equation 4.6, the sensitivity must approach unity. Therefore, the sensitivity cannot be constrained to be less than unity for frequencies that are well beyond the bandwidth.

Other restrictions on the sensitivity weighting function are dependent on the characteristics of the plant [22]. For example, if the specifications on the sensitivity is

$$\|W_1 S\|_\infty < 1 \tag{4.7}$$

where $W_1(s)$ is defined as the sensitivity weighting function, then an upper bound on the weighting function for a plant with a non-minimum phase zero at z, is

$$|W_1(z)| \le \|W_1 S\|_\infty < 1. \tag{4.8}$$

This restricts how far the sensitivity function can be reduced at frequencies near the non-minimum phase zero. Furthermore, if the plant has an unstable pole at p and a non-minimum phase zero at z, an upper bound on the weighting function is

$$|W_1(z)| \leq \|W_1 S\|_\infty \leq \left|\frac{z-p}{z+p}\right| \tag{4.9}$$

When the pole and zero are close to each other, this severely restricts how far the sensitivity function can be reduced at frequencies near the non-minimum phase zero.

As the operating point changes, the pole and zero locations will change, so the restrictions on the sensitivity function will change. For instance, at a particular unstable flow coefficient, it may not be possible to meet the specified sensitivity bound; increasing the flow coefficient (toward the stable regime) tends to relax Equation 4.9 such that the bound will eventually be able to be satisfied.

4.1.2.2 Gain and Phase Margin Concepts

In classical control, the gain and phase margin of the design are typically specified. Graphically, these margins are shown in Figure 4.1. The gain margins are stated as the inverse of the gain when the phase is 180 degrees. When expressed in dB, the upper gain margin, g_u, is positive and the lower gain margin, g_l, is negative. These gain margins correspond to how much the system gain can vary in magnitude before the system becomes unstable - the upper gain margin placing an upper limit, and the lower gain margin placing a lower limit. For an unstable system, the critical point, -1, must be encircled at least once, so there are both upper and lower gain margins. For stable systems, the critical point is not encircled, so there usually only exists an upper gain margin. The phase margin, θ_m, is the difference in phase from 180 degrees of the Nyquist plot when the magnitude is equal to unity. Typical gain and phase margins are 6 dB and 40 degrees, respectively. An upper gain margin of 6 dB translates into a distance

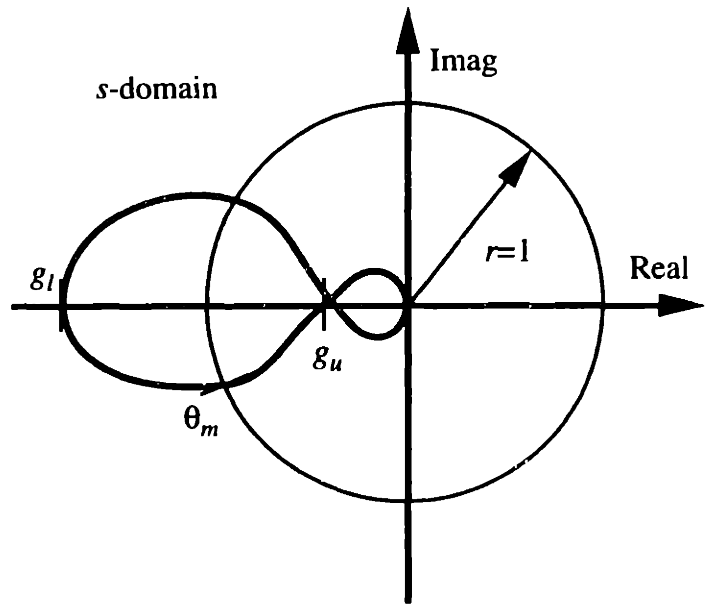

Figure 4.1 - Gain and phase margins on a complex Nyquist plot.

away from the critical point of about .5. If the sensitivity is bounded less than 2, this gives an upper gain margin of 6 dB, a lower gain margin of -3.5 dB and a phase margin of 28.9 degrees. The weighting is

$$W_1(s) = \frac{1}{2}. \tag{4.10}$$

These margins are lower limits for typical designs, but will be adequate for the present design study.

4.1.3 Robustness to Operating Point Changes

This section will first present the general construction of an H_∞ constraint for a nominal model with a multiplicative perturbation in the frequency domain. Then, the 'truth' model will be examined at different flow coefficients using different sensor and actuator locations to understand why this type of robustness is difficult to specify and also why some configurations are naturally robust to operating point changes. For the cases without this natural stability robustness, part of the frequency domain bounds described may be of use.

The main characteristic of operating point changes is the movement of the incompressible rotating stall mode between the stable and unstable regimes. Other factors, though, include the movement of the zeros of particular transfer functions. If a zero is near the rotating stall mode, then the configuration tends to have worse robustness to operating point changes.

First, some notation will be introduced. Let $G_N(s)$ represent the nominal system used for design purposes, and $G_A(s)$ represent the actual plant. The relation between the two can be expressed as a multiplicative error:

$$G_A(s) = G_N(s)\left(1 + \Delta(s)W_2(s)\right) \tag{4.11}$$

where $\Delta(s)$ has unity magnitude with arbitrary phase and $W_2(s)$ represents the magnitude of the error between plants. Rearranging Equation 4.11 yields

$$|W_2(s)| = \left|\frac{G_A(s)}{G_N(s)} - 1\right|. \tag{4.12}$$

This relation between the nominal and the actual plant model can be formulated into restrictions on the complementary sensitivity transfer function [22]:

$$\|W_2 C\|_\infty < 1. \tag{4.13}$$

This is actually a result of the small gain theorem stating that, if the nominal closed-loop system and weighting function are stable, then a feedback loop with a stable Δ(s) having an H_∞ norm less than unity is guaranteed to be stable, if Equation 4.13 is satisfied. This is shown in block diagram form in Figure 4.2.

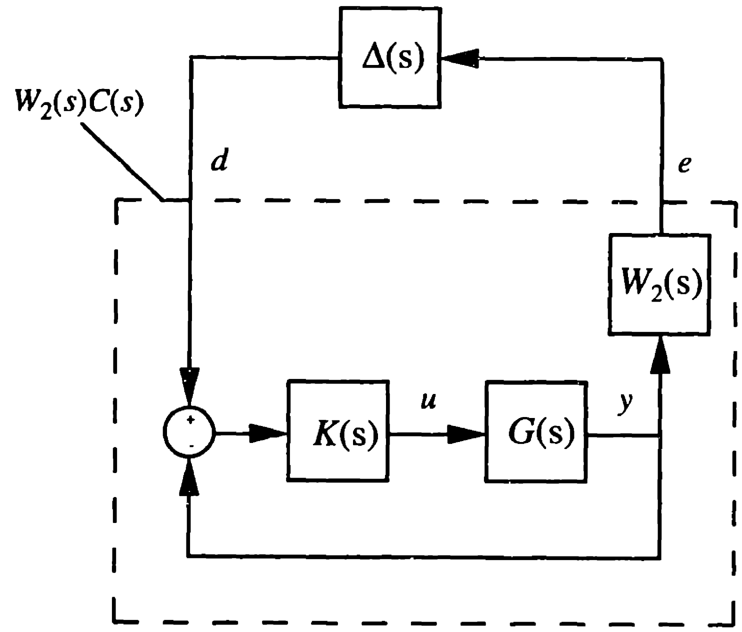

Figure 4.2 - Block diagram of small gain theorem.

Table 4.1 - Model data for operating point changes study.

Harmonic	1
Flow Coefficient	.4506-.4721
Rotor Speed	100% of design speed
Measurement	$\delta P/P$
Jet Velocity	2 * gap mean axial velocity
Jet Total Temperature	mean total temperature in the gap
Correlations	supplied by manufacturer

For the cases without natural stability robustness, as describe in the introduction to Section 4.1, it is necessary to obtain some theoretical data on how the plant changes when the operating point is changed. The nominal conditions are described in Table 4.1. Since some configurations will have natural robustness to increases in the flow coefficient, the design point will be the lowest flow coefficient, ϕ, which can satisfy the constraints. Therefore, Equation 4.12 needs to be examined using the nominal plant defined at the design point, and actual plants defined at higher flow coefficients. Four separate sensor/actuator pairs will be examined, $(k_a, k_s) = (2,1), (2,8), (8,3)$, and $(5,6)$, using the numbering scheme of Figure 2.1.

First, Figures 4.3 and 4.4 show the open-loop transfer functions for a particular sensor/actuator pair $(k_a, k_s) = (2,1)$ for 2 stable, 2 unstable, and 1 near neutrally stable operating points. Figure 4.3 shows the positive frequency magnitudes and phases, while Figure 4.4 shows the negative frequency magnitudes and phases. These give an idea of how the plant differs for different flow coefficients. In terms of the Nyquist diagram, a compensator designed at an unstable flow coefficient will have one loop around the critical point as shown in Figure 4.1. Generally, the rotating stall frequency (the lowest positive frequency eigenvalue) will correspond to the point on the diagram labeled the lower gain margin. As the flow coefficient is increased, this loop tends to expand (following the increases in magnitude in Figure 4.3), increasing the distance from the critical point, hence decreasing the sensitivity. As the neutral stability point is crossed, the Nyquist diagram 'flips' to the right hand side of Figure 4.1, thereby decreasing the sensitivity. However, the point on the diagram of Figure 4.1 labeled the upper gain margin may also increase in magnitude, thus decreasing the sensitivity. The control configurations which display a decrease in sensitivity over all frequencies as the flow coefficient is increased are said to have 'natural' stability robustness to operating point changes, while others may only have increases in sensitivity near the frequency of the upper gain margin, and still others may exhibit worse behavior.

Figures 4.5-4.8 show $|G_A(j\omega) / G_N(j\omega) - 1|$ for four different sensor/actuator pair locations for a nominal case of ϕ=.4506 and the actual case of ϕ=.4721. If a transfer function, $W_2(s)$, is fit to the figures, then the constraint (Equation 4.13) can be used in a design procedure. However, this bound is conservative since the phase of the multiplicative error is not arbitrary. Indeed, at the rotating stall frequency, these bounds would require the complementary sensitivity function to approach zero. This is clearly not necessary since, if the phase of the open-loop system is 180 degrees at this frequency, then increasing the flow coefficient would move the Nyquist contour away from the critical point - not towards it. Having said this, one expects worse robustness properties from sensor/actuator pair (k_a, k_s) = (2,8) shown in Figure 4.6, because it shows a higher magnitude at low frequencies than the others. Examining further deviations in flow coefficient, Figures 4.9 and 4.10 show the two pairs (k_a, k_s) = (2,1), (2,8) for the same nominal flow coefficient, but an actual case of ϕ=.4935, which is 4.9% greater than the neutral stability point. The case of (k_a, k_s) = (2,8) again shows a higher magnitude at low frequencies which is expected to cause poor stability robustness to operating point changes.

4.1.4 State and Control Cost Functions

The two cost functions of the system, to be used for evaluating control configurations, will be a *state cost* and a *control cost*. The state cost is the ultimate objective to keep small, but not at the expense of unacceptably large actuator activity.

The state cost function will be the sum of the mean-square static pressure perturbations, $\delta P/P$, in each of the inter-blade row gaps and ducts. The reason for this choice is that, if the static pressure perturbations are kept small, then the velocity perturbations are also kept small. Keeping perturbations small helps to avoid non-linear effects. The plant to which the state cost is applied will be driven by process noise and measurement noise as described below. Similarly, the mean-squared control activity, $\delta\dot{m}/\dot{m}$, required to control the plant will be defined as the control cost.

Driving the mean square states are process noise and sensor noise. Little data are available for process and sensor noise, but process noise is expected to dominate the measurement. The dominant frequency of the process noise affecting compressors is at the rotor frequency, due to various imperfections (asymmetries) in the compressor. It is desired to keep the response of the control system small at the rotor frequency so that unnecessary effort is not expended trying to reduce these disturbances. In the operating

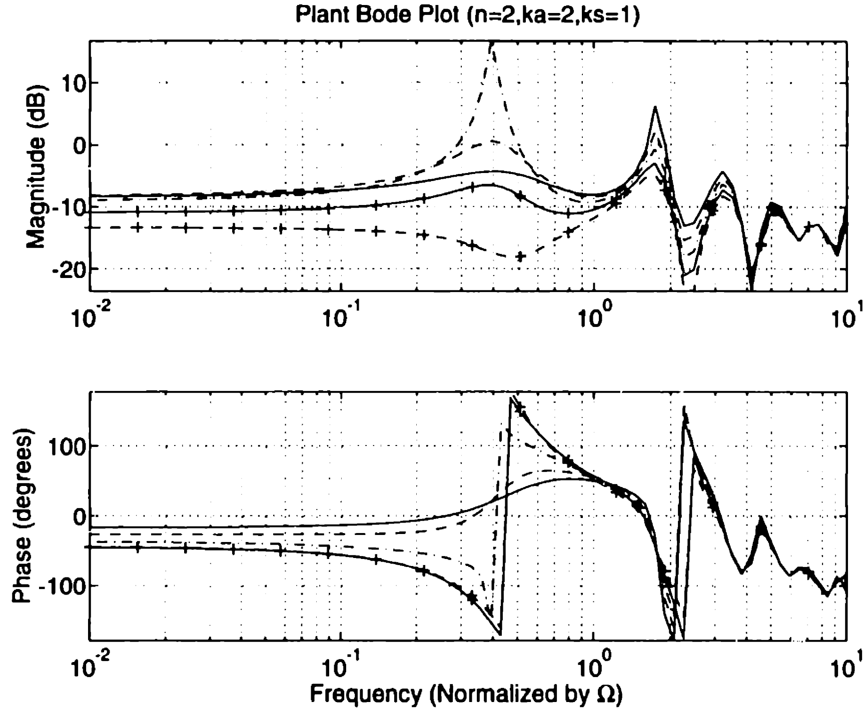

Figure 4.3 - Open-loop positive frequency response plots for unstable [ϕ=.4506 (solid),ϕ=.461 (dashed)] and stable [ϕ=.4713 (dash-dotted),ϕ=.4816 (solid-plus),ϕ=.4919 (dashed-plus)] flow coefficients.

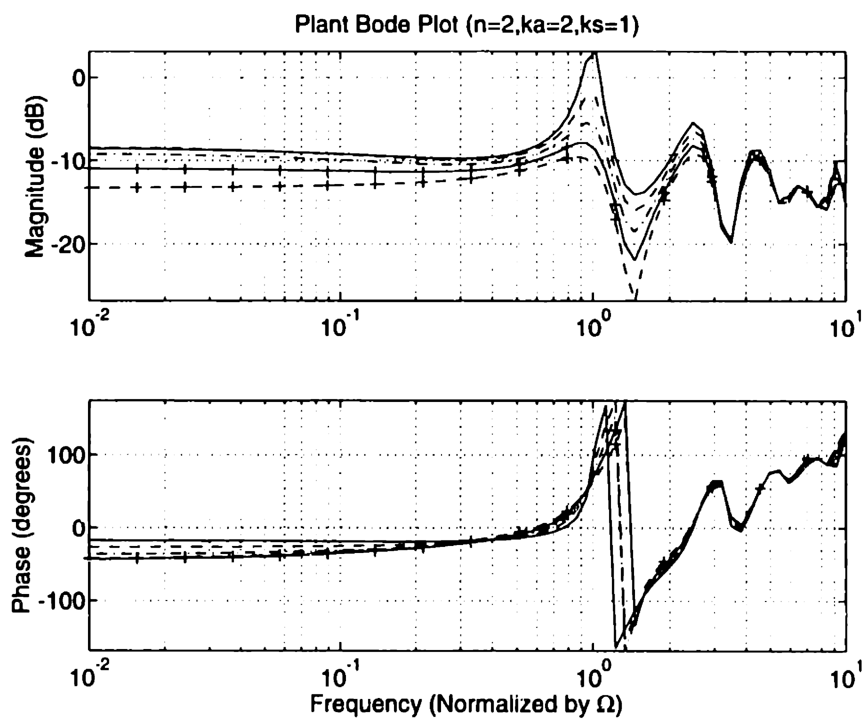

Figure 4.4 - Open-loop negative frequency response plots for unstable [ϕ=.4506 (solid),ϕ=.461 (dashed)] and stable [ϕ=.4713 (dash-dotted),ϕ=.4816 (solid-plus),ϕ=.4919 (dashed-plus)] flow coefficients.

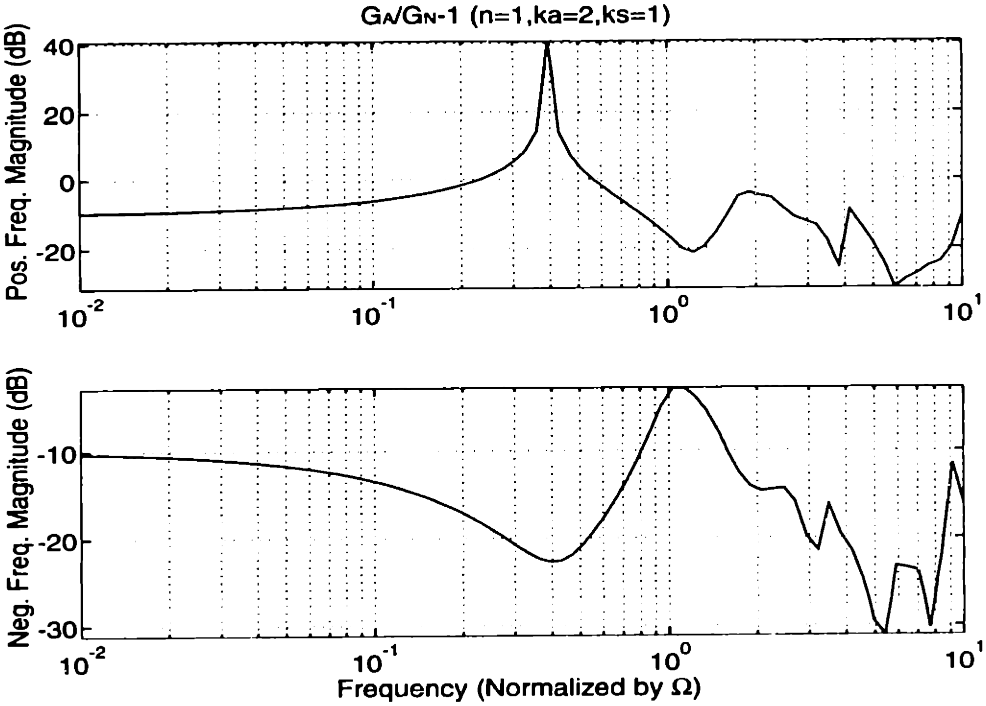

Figure 4.5 - Positive and negative frequency magnitude responses of $G_A/G_N - 1$
$(k_a, k_s) = (2,1)$, ϕ_N=.4506, ϕ_A=.4721.

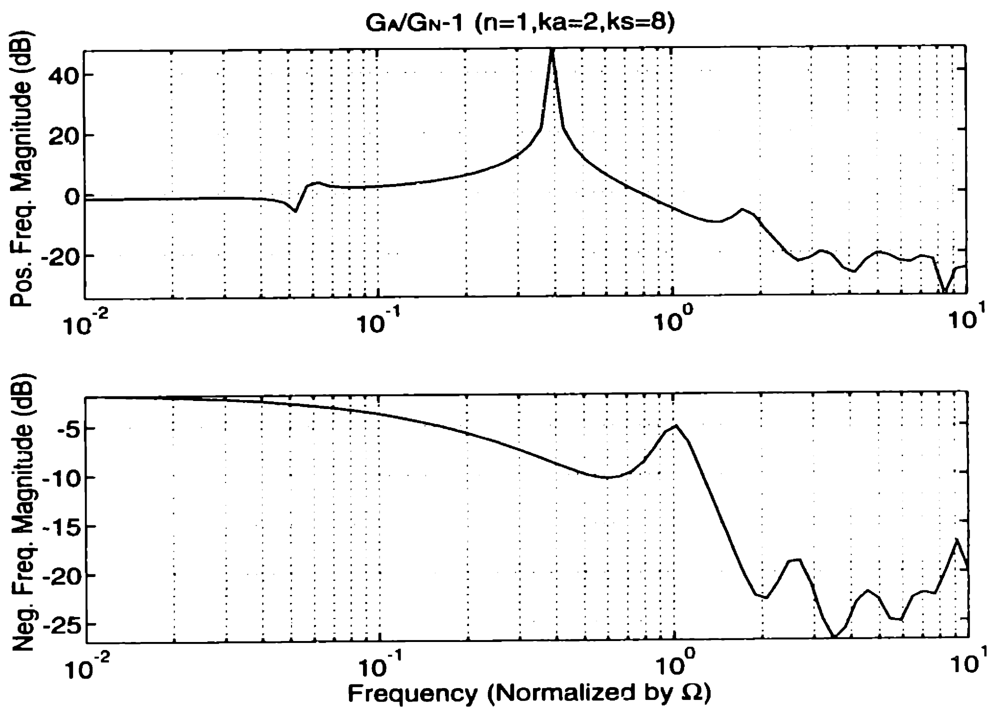

Figure 4.6 - Positive and negative frequency magnitude responses of $G_A/G_N - 1$
$(k_a, k_s) = (2,8)$, ϕ_N=.4506, ϕ_A=.4721.

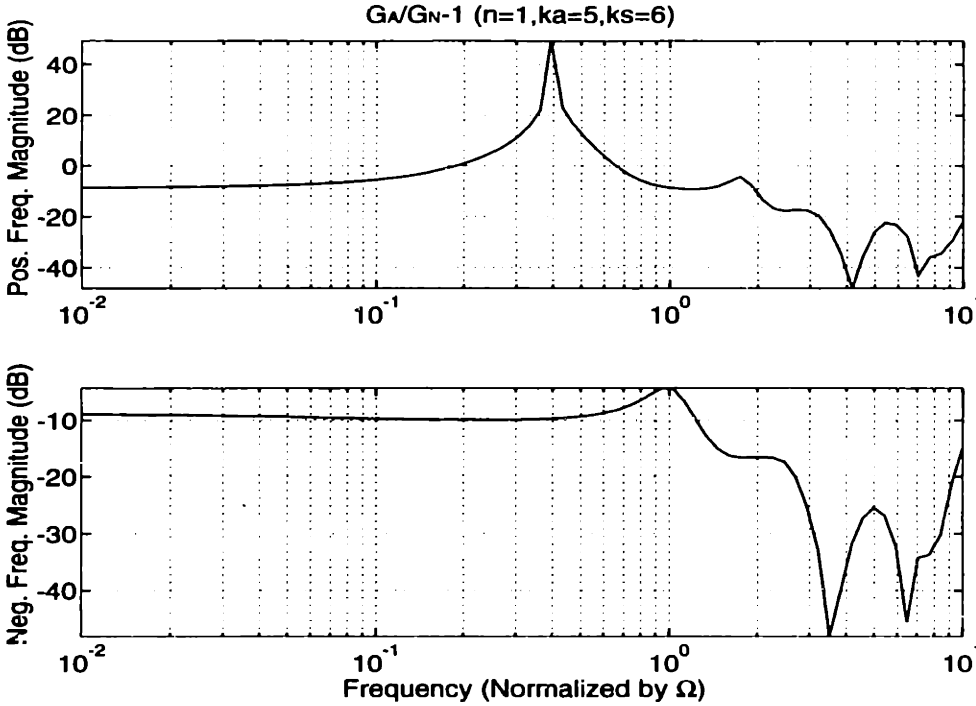

Figure 4.7 - Positive and negative frequency magnitude responses of $G_A/G_N - 1$ $(k_a, k_s) = (5,6)$, ϕ_N=.4506, ϕ_A=.4721.

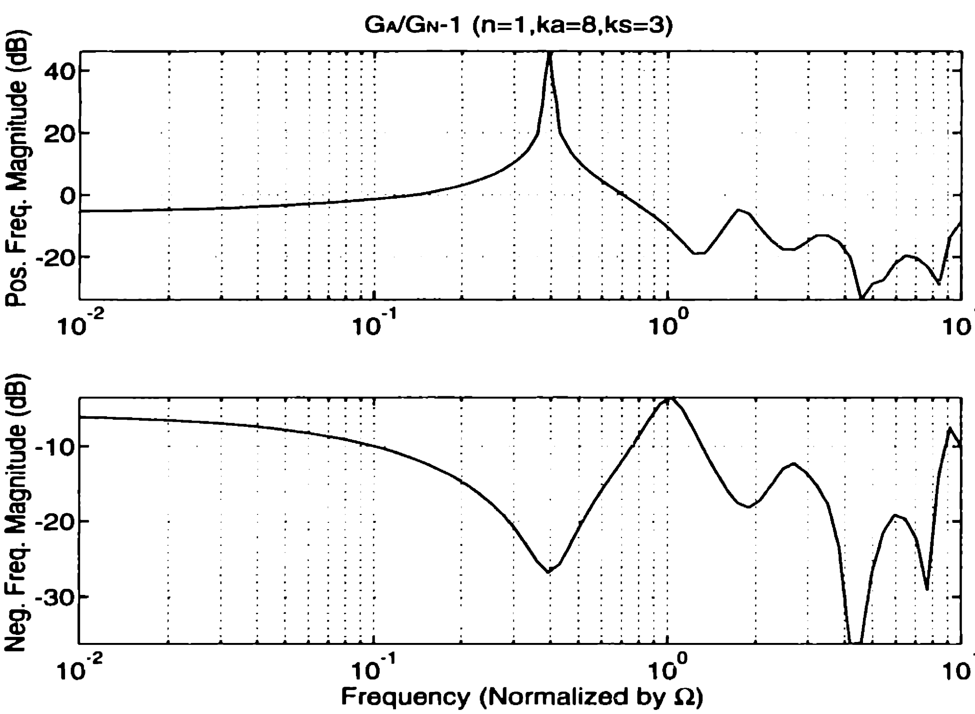

Figure 4.8 - Positive and negative frequency magnitude responses of $G_A/G_N - 1$ $(k_a, k_s) = (8,3)$, ϕ_N=.4506, ϕ_A=.4721.

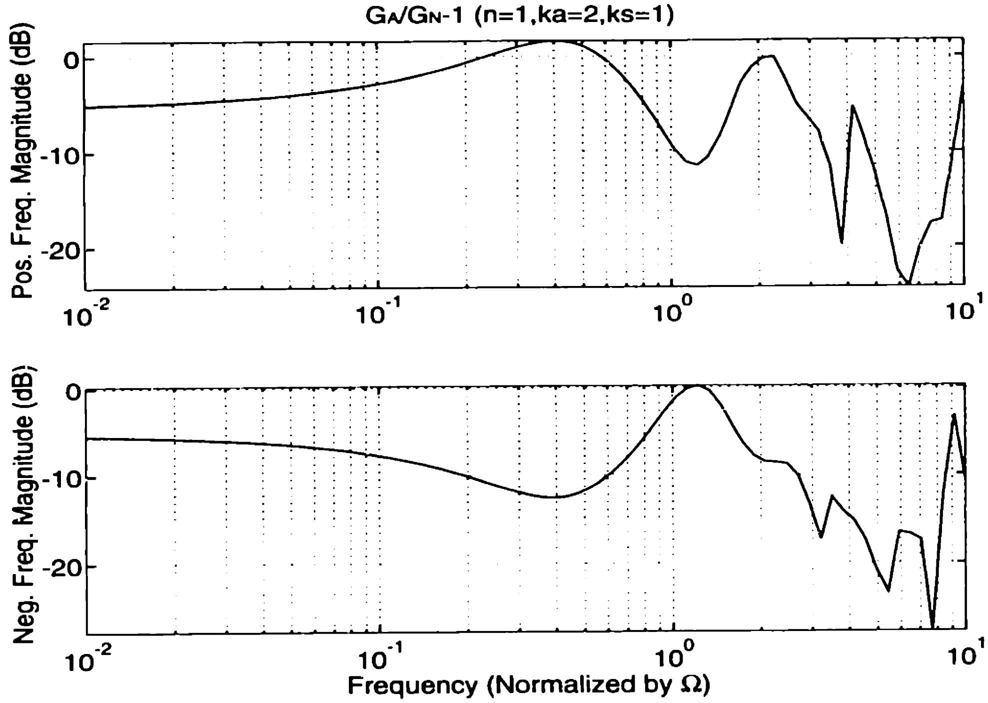

Figure 4.9 - Positive and negative frequency magnitude responses of $G_A/G_N - 1$
$(k_a, k_s) = (2,1)$, ϕ_N=.4506, ϕ_A=.4935.

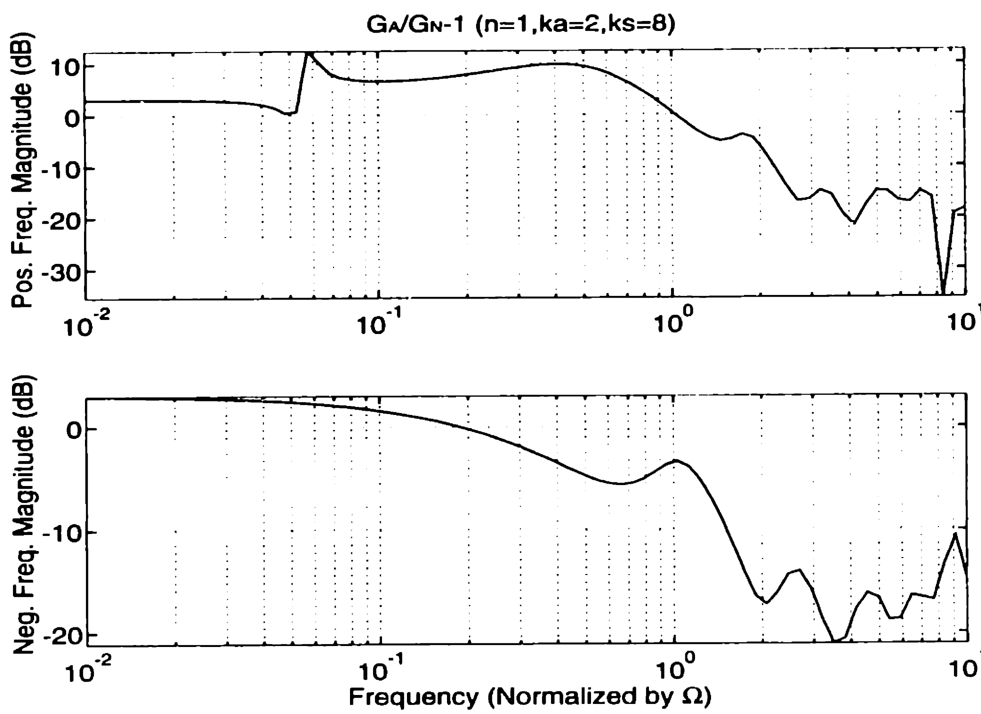

Figure 4.10 - Positive and negative frequency magnitude responses of $G_A/G_N - 1$
$(k_a, k_s) = (2,8)$, ϕ_N=.4506, ϕ_A=.4935.

range which is being examined, in this compressor, and for the first harmonic, all of the eigenvalues are stable, except for the rotating stall mode which appears at a frequency less than the rotor frequency. Thus, it is possible for the compensator to reduce the sensitivity to noise at the rotor frequency without affecting stability properties of the system. Other compressors may exhibit dominant eigenvalues at or near the rotor frequency. For instance, the third harmonic has a dominant eigenvalue near the rotor frequency. In this case, there is a conflict between keeping the noise response low and keeping the gain high for stability robustness at the rotor frequency. This is certainly an undesirable situation which may not have a good solution.

Anticipating the H_∞ robust design, the rotor frequency noise can be incorporated into the design procedure by placing a spike in the weighting of the complementary sensitivity function at the rotor frequency. This will cause the compensator to have a notch at that frequency. The weighting function becomes

$$W_2'(s) = \frac{s + (.001 - j\Omega)}{s + (.00001 - j\Omega)} W_2(s) \tag{4.14}$$

for a spike of 40 dB.

The cost functions will be calculated using a system driven by a process noise of unity magnitude with a spike of 40 dB at the rotor frequency in each pressure loss state. In other words, the process noise is a white process noise of unity intensity fed through a 'spike' filter. The sensor noise will be white with an intensity of 0.01, since it is expected to have less of an effect on the measurement. A few definitions will be made before the cost functions can be calculated. First the system is defined as

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{F}\mathbf{x} + \mathbf{G}u + \mathbf{P}w \\ y &= \mathbf{H}\mathbf{x} + \mathbf{D}u + v \end{aligned} \tag{4.15}$$

where the state representing the spike has been appended to the state vector, and the process noise feeds into that state. The compensator, with negative feedback, is defined as

$$\begin{aligned} \dot{\mathbf{z}} &= \mathbf{F}_f\mathbf{z} + \mathbf{G}_f y \\ u &= -\mathbf{H}_f\mathbf{z} \end{aligned} \tag{4.16}$$

The state space model from the noise inputs to the states is then

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\mathbf{z}} \end{bmatrix} = \begin{bmatrix} \mathbf{F} & -\mathbf{G}\mathbf{H}_f \\ \mathbf{G}_f\mathbf{H} & \mathbf{F}_f - \mathbf{G}_f\mathbf{D}\mathbf{H}_f \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{z} \end{bmatrix} + \begin{bmatrix} \mathbf{P} & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_f \end{bmatrix} \begin{bmatrix} w \\ v \end{bmatrix} = \mathbf{F}_c \begin{bmatrix} \mathbf{x} \\ \mathbf{z} \end{bmatrix} + \mathbf{G}_c \begin{bmatrix} w \\ v \end{bmatrix} \tag{4.17}$$

If an output vector is defined to be the set of static pressure perturbations in each gap and duct:

$$\mathbf{y} = \mathbf{H}_c \begin{bmatrix} \mathbf{x} \\ \mathbf{z} \end{bmatrix} \tag{4.18}$$

then using the solution of the Lyapunov equation

$$\mathbf{F}_c\mathbf{L} + \mathbf{L}\mathbf{F}_c^H + \mathbf{G}_c\mathbf{G}_c^H = \mathbf{0} \tag{4.19}$$

the sum of the mean-square outputs, the state cost, is [22]

$$J_x = Tr\left[\mathbf{H}_c\mathbf{L}\mathbf{H}_c^H\right] \tag{4.20}$$

Similarly, the control cost is the mean-square control activity:

$$J_u = Tr\left[\begin{bmatrix}\mathbf{0} & -\mathbf{H}_f\end{bmatrix}\mathbf{L}\begin{bmatrix}\mathbf{0}^H \\ -\mathbf{H}_f^H\end{bmatrix}\right] \tag{4.21}$$

4.2 Performance Design Methodology

This section will specify a cost function for the compensator to minimize. It will not be exactly the cost functions described in Section 4.1.4, but it will be related. The Linear Quadratic-Gaussian (LQG) design technique will be used for the design procedure.

The LQG design technique optimizes the mean-square of the states and the control action under specified noise inputs [23]. If the weights of the cost function and noise input parameters (defined below) are fixed in advance, then the resulting minimum cost index can be used to compare control configurations, with lower costs being better. These costs will be derived at only one flow coefficient, but it is likely that a sensor/actuator pair with the lowest cost among all configurations at the flow coefficient will have a low cost at different flow coefficients (this will be examined in Chapter 5). In the present case, this means that the best sensor/actuator location, in terms of LQG cost, can be found by examining the LQG costs at one flow coefficient for each sensor/actuator pair location. This technique is easier than that used for H_∞ designs, which requires performing designs at multiple flow coefficients, as will be seen in Section 4.3.

The LQG optimal control assumptions and solution will be reviewed in Section 4.2.1. The number of free parameters in the LQG design procedure will be reduced by making assumptions on the cost index and the noise sources in Section 4.2.2. Section 4.2.3 will present a study to determine the free parameters in the LQG design procedure.

4.2.1 Review of Linear Quadratic-Gaussian (LQG) Design and Cost Definition

The LQG optimal control design procedure operates on a linear system of the form:

$$
\begin{aligned}
\dot{\mathbf{x}} &= \mathbf{F}\mathbf{x} + \mathbf{G}u + \mathbf{w} \\
y &= \mathbf{H}\mathbf{x} + \mathbf{D}u + v
\end{aligned} \tag{4.22}
$$

using the notation of Bryson and Ho [23], where $\mathbf{w}$ and v are white noise with intensities

$$
E\left\{\begin{bmatrix}\mathbf{w}(t)\\ v(t)\end{bmatrix}\begin{bmatrix}\mathbf{w}^H(t_1) & v^H(t_1)\end{bmatrix}\right\} = \begin{bmatrix}\mathbf{Q}(t) & \mathbf{T}(t)\\ \mathbf{T}^H(t) & \mathbf{R}(t)\end{bmatrix}\delta(t-t_1) \tag{4.23}
$$

and where E is the expectation operator, δ is the delta function and ^{H} denotes the complex conjugate transpose. The cost function to optimize is quadratic in the states and the control:

$$
J = E\left\{\frac{1}{t_f - t_0}\int_{t_0}^{t_f}\begin{bmatrix}\mathbf{x}^H(t) & \mathbf{u}^H(t)\end{bmatrix}\begin{bmatrix}\mathbf{A}(t) & \mathbf{N}(t)\\ \mathbf{N}^H(t) & \mathbf{B}(t)\end{bmatrix}\begin{bmatrix}\mathbf{x}(t)\\ \mathbf{u}(t)\end{bmatrix}dt\right\}. \tag{4.24}
$$

In the present case, all of the matrices are constant, and the statistical steady state solution is desired. The minimization [23] involves solving two Ricatti equations, the control and filter Ricatti equations:

$$
\begin{aligned}
\dot{\mathbf{S}} &= -\mathbf{S}\mathbf{F} - \mathbf{F}^H\mathbf{S} + (\mathbf{S}\mathbf{G} + \mathbf{N})\mathbf{B}^{-1}\left(\mathbf{G}^H\mathbf{S} + \mathbf{N}^H\right) - \mathbf{A} \\
\dot{\mathbf{P}} &= \mathbf{F}\mathbf{P} + \mathbf{P}\mathbf{F}^H - \left(\mathbf{P}\mathbf{H}^H + \mathbf{T}\right)\mathbf{R}^{-1}\left(\mathbf{H}\mathbf{P} + \mathbf{T}^H\right) + \mathbf{Q}
\end{aligned} \tag{4.25}
$$

The minimized cost is

$$
\min J_{LQG} = Tr\left\{\frac{1}{t_f - t_0}\mathbf{S}(t_0)\mathbf{X}(t_0) + \frac{1}{t_f - t_0}\int_{t_0}^{t_f}\left[\mathbf{S}\mathbf{Q} + (\mathbf{S}\mathbf{G} + \mathbf{N})\mathbf{B}^{-1}\left(\mathbf{G}^H\mathbf{S} + \mathbf{N}^H\right)\mathbf{P}\right]dt\right\} \tag{4.26}
$$

where Tr is the trace operator. The steady state cost, referred to here as the 'LQG' cost, can be found by letting $t_f \rightarrow \infty$ in Equation 4.26, resulting in:

$$\min J_{LQG}^{SS} = Tr\left\{\mathbf{SQ} + (\mathbf{SG} + \mathbf{N})\mathbf{B}^{-1}\left(\mathbf{G}^H\mathbf{S} + \mathbf{N}^H\right)\mathbf{P}\right\} \tag{4.27}$$

4.2.2 Free Parameter Reduction

Now that the solution of the LQG problem has been stated, free parameters in the procedure must be chosen. The weights in the cost function, Equation 4.24, and the noise intensities in Equation 4.23 can be viewed as design parameters which can be chosen to shape the frequency response to meet the specifications of Section 4.1.

To penalize the complementary sensitivity function at frequencies greater than the desired bandwidth, the measurement noise can be modeled to be significant at frequencies higher than the bandwidth. Since the transfer function from the sensor noise to the states is related to the complementary sensitivity function, the complementary sensitivity will be penalized for being large at frequencies greater than the desired bandwidth. To achieve this, the measurement noise is chosen to be 'colored' with a first order filter whose white noise input is uncorrelated with the process noise. The shape of the filter has a low frequency level of 0.1 (corresponding to the 0.01 intensity of the sensor noise of the cost functions in Section 4.1.4) with increasing gain beyond the desired bandwidth of the system.

To achieve an insensitive response to process noise at the rotor frequency, a first order filter can be used to color the process noise. In compressors, there is really only one source of rotor frequency noise, the rotor itself, so only one noise term will be fed into all the different blade rows. This approach avoids redundant states. The disadvantage of this is that all of the phases of the noise entering each blade row are the same, but this is not expected to be important. The compensator will simply contain a zero at the rotor frequency to cancel this noise.

For example, if the process noise filter is

$$G_x(s) = \frac{s + (.001 - j\Omega)}{s + (.000001 - j\Omega)} \tag{4.28}$$

and the measurement noise filter is

$$G_y(s) = \frac{(s / 2\Omega + .1)}{1 + s / 100\Omega} \tag{4.29}$$

with state-space representations

$$G_y = \left[\begin{array}{c|c} \mathbf{F}_y & \mathbf{G}_y \\ \hline \mathbf{H}_y & \mathbf{D}_y \end{array}\right] \qquad G_x = \left[\begin{array}{c|c} \mathbf{F}_x & \mathbf{G}_x \\ \hline \mathbf{H}_x & \mathbf{D}_x \end{array}\right] \tag{4.30}$$

then the state-space system becomes

$$\begin{aligned} \begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\mathbf{z}}_x \\ \dot{\mathbf{z}}_y \end{bmatrix} &= \begin{bmatrix} \mathbf{F} & \mathbf{PH}_x & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_x & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{F}_y \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{z}_x \\ \mathbf{z}_y \end{bmatrix} + \begin{bmatrix} \mathbf{G} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \mathbf{u} + \begin{bmatrix} \mathbf{PD}_x & \mathbf{0} \\ \mathbf{G}_x & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_y \end{bmatrix} \begin{bmatrix} w \\ v \end{bmatrix} \\ y &= \begin{bmatrix} \mathbf{H} & \mathbf{0} & \mathbf{H}_y \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{z}_x \\ \mathbf{z}_y \end{bmatrix} + \mathbf{Du} + \mathbf{D}_y v \end{aligned} \tag{4.31}$$

where $\mathbf{P}$ directs the noise into the proper states. If w and v are white noise with $E[w(t)v(t_1)] = 0$, the noise intensities becomes

$$\begin{bmatrix} \tilde{\mathbf{Q}} & \tilde{\mathbf{T}} \\ \tilde{\mathbf{T}}^H & \tilde{\mathbf{R}} \end{bmatrix} = \left[\begin{array}{c|c} \mathbf{PD}_x & \mathbf{0} \\ \mathbf{G}_x & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_y \\ \hline \mathbf{0} & \mathbf{D}_y \end{array}\right] \begin{bmatrix} \mathbf{Q} & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \end{bmatrix} \left[\begin{array}{c|c} \mathbf{PD}_x & \mathbf{0} \\ \mathbf{G}_x & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_y \\ \hline \mathbf{0} & \mathbf{D}_y \end{array}\right]^H \tag{4.32}$$

If we want to weight the following error states:

$$\mathbf{e} = \begin{bmatrix} c\mathbf{H}_e & c\mathbf{D}_e \\ \mathbf{0} & \mathbf{B} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ u \end{bmatrix} \tag{4.33}$$

with c an arbitrary weighting, the LQG weights become

$$\begin{bmatrix} \tilde{\mathbf{A}} & \tilde{\mathbf{N}} \\ \tilde{\mathbf{N}}^H & \tilde{\mathbf{B}} \end{bmatrix} = \left[\begin{array}{ccc|c} c\mathbf{H}_e & \mathbf{0} & \mathbf{0} & c\mathbf{D}_e \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{B} \end{array}\right]^H \left[\begin{array}{ccc|c} c\mathbf{H}_e & \mathbf{0} & \mathbf{0} & c\mathbf{D}_e \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{B} \end{array}\right] \tag{4.34}$$

4.2.3 Free Parameter Selection

The LQG compensator resulting from the minimization process has no guaranteed robustness properties, so these have to be checked after the design is done. Frequently, though, the free parameters in the performance index, $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{N}$, and the noise parameters, $\mathbf{Q}$, $\mathbf{R}$, and $\mathbf{T}$, are manipulated in an iterative design procedure to satisfy a set of constraints not directly related to the minimization procedure. In the present case, it is desired to fix as many parameters as possible by examining their impact on properties of the system, such as the sensitivity and complementary sensitivity constraints, and use these parameters for each separate sensor/actuator location design for performance index comparison.

To standardize the procedure, the error states of Equation 4.33 will be chosen to be the static pressure perturbations in each gap, which are the desired errors to minimize in the state cost, with a variable weight of c, and a unity weight on the control, $\mathbf{B}=\mathbf{I}$. The noise intensities in Equation 4.32 are chosen as unity for the measurement noise, $\mathbf{R}=\mathbf{I}$, and a variable intensity for the process noise, $\mathbf{Q}=q$. $\mathbf{B}$ can be specified as the identity because only the relative values between state and control weights is important. Any scaling of these parameters simply changes the cost without changing the resultant

Table 4.2 - Model data for LQG free parameter selection.

Harmonic	1
Flow Coefficient	.461 (2.1% below neutral stability)
Rotor Speed	100% of design speed
Measurement	$\delta P/P$
Jet Velocity	2 * mean axial velocity in the gap
Jet Total Temperature	mean total temperature in the gap
Correlations	supplied by manufacturer

compensator. Similarly, **R** can be specified as identity because only the relative value of the noise intensities is important.

A few sensor/actuator pair locations will be examined to get a general idea of how the free parameters affect the complementary sensitivity and the sensitivity functions. An unstable operating point is chosen because it is expected that the compensator will be designed at the lowest possible flow coefficient from the arguments in Section 4.1.3.

The four sensor/actuator location pair combinations are (k_a, k_s) = (2,1), (2,8), (8,3), and (5,6). These were chosen as a representative sample of the input-output pairs. The configuration is described in Table 4.2. In the designs that follow, if the sensor and actuator are in the same gap, the sensor is assumed to be upstream of the actuator. This is a practical consideration due to the discrete nature of the sensors and actuators; if the jet actuator is upstream of the sensor, the jet may or may not impinge upon the sensor, depending on their relative alignment. If the jet does not impinge upon the sensor, then the sensor may act as if it is upstream of the actuator; if the jet impinges upon the sensor, then the jet will not have enough axial distance to spread out properly, and the measurement may not reflect reality. The jet velocity is chosen to be high, but not supersonic. The jet total temperature is chosen to be equal to the mean because tests have shown that total temperature is not a significant variable.

Sample sensitivity and complementary sensitivity functions from LQG designs using various parameter selections are shown in Appendix E and summarized here. The results of the designs are the transfer functions of interest and how the parameters affect them. For instance, as the process noise level is reduced, the sensitivity increases at low frequencies and the complementary sensitivity decreases near the desired bandwidth. As the state weights are increased, the complementary sensitivity decreases near the desired bandwidth, although the marginal effect is diminished at state weights greater than 1. The results also show the effect of the rotor frequency noise, which is essentially to place a zero in the compensator to cancel the noise.

The choice of parameters will be $c=1$ and $q=1$. With these choices, 3 of the 4 sensor/actuator pairs meet the constraints, and of the one which does not, pair (8,3), this choice of parameters is in the middle of possibilities. The best parameters for (8,3) is $c=1$, $q=.01$. However, all of the other pairs have poor sensitivities using these parameters.

It is not possible to choose one set of LQG parameters to meet the specifications all sensor/actuator placements. One choice of parameters can be made based on the most placements which satisfy the constraints. But, if we accept the fact that robustness can not be guaranteed with one set of parameters, then the parameters can be chosen to reflect the sum of the state and control costs, ignoring robustness. The LQG cost will then be the minimum sum of state and control costs applied to a system having colored measurement noise. In this particular case, the two ways to choose parameters, in fact, coincide.

4.3 Robust Design Methodology

This section will review the general H_∞ design procedure and then apply the constraints of Section 4.1 in the H_∞ design procedure. This design procedure, in general, iterates to find the minimum H_∞-norm of a transfer function. In the present case, though, specifications have been developed that the system needs to satisfy. No iteration on the norm will be performed; rather, the operating point will be iterated on to find the lowest flow coefficient at which the system can satisfy the requirements. The resulting compensator will then be used to find the state and control costs of the system.

4.3.1 Review of H_∞ Design Method

4.3.1.1 Definition and Motivation

The H_∞ optimal control problem can be stated as minimizing the H_∞-norm of a transfer function from a set of disturbances to a set of errors,

$$\left\| \frac{\mathbf{e}(s)}{\mathbf{d}(s)} \right\|_\infty \tag{4.35}$$

where the H_∞-norm is the maximum singular value, $\bar{\sigma}$, of the transfer function over all frequencies,

$$\left\| \frac{\mathbf{e}(s)}{\mathbf{d}(s)} \right\|_\infty = \max_\omega \bar{\sigma}\left(\frac{\mathbf{e}(j\omega)}{\mathbf{d}(j\omega)} \right). \tag{4.36}$$

In the current problem, the H_∞-norm is exactly the type of specification developed in Section 4.1. For example, if the frequency domain constraint on a particular transfer function is

$$\left| \frac{e(j\omega)}{d(j\omega)} \right| < \left| W^{-1}(j\omega) \right| \qquad \forall \omega, \tag{4.37}$$

then this will be satisfied if the H_∞-norm of $\left\| W \frac{e}{d} \right\|_\infty$ is less than unity.

The weighting function can arise from different specifications. One such source, as stated in Section 4.1.3, is a bound on the multiplicative modeling error of the system.

Another source can be a performance specification, such as decreasing the sensitivity at low frequencies to achieve low frequency disturbance rejection. A third source, as stated in Section 4.1.2, is to guarantee specified gain and phase margins.

4.3.1.2 Solution Technique

This section will present a simplified version of the H_∞ design solution. The complete solution complicates the equations considerably, but all of the important concepts are contained in this problem. The solution to the H_∞ design procedure is described in Doyle et al., 1988 [24].

If a state-space system has a disturbance vector, $\mathbf{d}$, and an error vector, $\mathbf{e}$, appended to the system as follows:

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \mathbf{e} \\ \mathbf{y} \end{bmatrix} = \left[\begin{array}{c|cc} \mathbf{F} & \mathbf{G}_1 & \mathbf{G}_2 \\ \hline \mathbf{H}_1 & \mathbf{0} & \mathbf{D}_{12} \\ \mathbf{H}_2 & \mathbf{D}_{21} & \mathbf{0} \end{array}\right] \begin{bmatrix} \mathbf{x} \\ \mathbf{d} \\ \mathbf{u} \end{bmatrix} \tag{4.38}$$

with the restrictions

$$\begin{aligned} \mathbf{D}_{12}{}^{H}\begin{bmatrix} \mathbf{H}_1 & \mathbf{D}_{12} \end{bmatrix} &= \begin{bmatrix} \mathbf{0} & \mathbf{I} \end{bmatrix} \\ \mathbf{D}_{21}\begin{bmatrix} \mathbf{G}_1{}^{H} & \mathbf{D}_{21}{}^{H} \end{bmatrix} &= \begin{bmatrix} \mathbf{0} & \mathbf{I} \end{bmatrix} \end{aligned} \tag{4.39}$$

then the H_∞ design procedure will produce a compensator, if one exists, such that

$$\left\| \frac{\mathbf{e}(s)}{\mathbf{d}(s)} \right\|_\infty < \gamma \tag{4.40}$$

The compensator is found by solving the following two Ricatti equations:

$$\mathbf{F}^H\mathbf{X}_\infty + \mathbf{X}_\infty\mathbf{F} + \mathbf{X}_\infty\left(\gamma^{-2}\mathbf{G}_1\mathbf{G}_1^H - \mathbf{G}_2\mathbf{G}_2^H\right)\mathbf{X}_\infty + \mathbf{H}_1\mathbf{H}_1^H = \mathbf{0}$$
$$\mathbf{F}\mathbf{Y}_\infty + \mathbf{Y}_\infty\mathbf{F}^H + \mathbf{Y}_\infty\left(\gamma^{-2}\mathbf{H}_1\mathbf{H}_1^H - \mathbf{H}_2\mathbf{H}_2^H\right)\mathbf{Y}_\infty + \mathbf{G}_1\mathbf{G}_1^H = \mathbf{0} \quad (4.41)$$

If positive semidefinite solutions, $\mathbf{X}_\infty$ and $\mathbf{Y}_\infty$, exist, and $\rho(\mathbf{X}_\infty\mathbf{Y}_\infty) < \gamma^2$, where ρ is the spectral radius, then a compensator can be found which satisfies Equation 4.40. One such compensator is

$$\mathbf{K}(s) = \left[\begin{array}{c|c} \mathbf{F}_\infty & -\mathbf{Z}_\infty\mathbf{L}_\infty \\ \hline \mathbf{A}_\infty & \mathbf{0} \end{array}\right] \quad (4.42)$$

where

$$\mathbf{F}_\infty = \mathbf{F} + \gamma^{-2}\mathbf{G}_1\mathbf{G}_1^H\mathbf{X}_\infty + \mathbf{G}_2\mathbf{A}_\infty + \mathbf{Z}_\infty\mathbf{L}_\infty\mathbf{H}_2$$
$$\mathbf{A}_\infty = -\mathbf{G}_2^H\mathbf{X}_\infty$$
$$\mathbf{L}_\infty = -\mathbf{Y}_\infty\mathbf{H}_2^H$$
$$\mathbf{Z}_\infty = \left(\mathbf{I} - \gamma^{-2}\mathbf{Y}_\infty\mathbf{X}_\infty\right)^{-1}$$

To find the minimum H_∞-norm, $\gamma_{\min}$, this procedure must be iterated to find the smallest γ for which a solution still exists.

4.3.2 Application to Current Problem

The general H_∞ design procedure is adapted for the current problem in this section. The current problem has two frequency domain constraints on the sensitivity and complementary sensitivity functions. If the two weighting functions are used from Sections 4.1.1, 4.1.2 and 4.1.4:

$$|S(j\omega)| < |W_1^{-1}(j\omega)| \qquad \forall\omega$$
$$|C(j\omega)| < |W_2^{-1}(j\omega)| \qquad \forall\omega \quad (4.43)$$

and a disturbance and error outputs are defined as

$$\begin{aligned} e_1(s) &= W_1(s)S(s)d(s) \\ e_2(s) &= W_2(s)C(s)d(s) \end{aligned} \tag{4.44}$$

then the H_∞ optimal control using $\gamma=1$, if it exists, will guarantee that the specifications are met.

To perform the H_∞ design, it is necessary to integrate the constraints of Sections 4.1.1 and 4.1.2 into the state-space model and define the error and disturbance functions. The state-space descriptions of the constraints on the sensitivity and complementary sensitivity functions will be defined as, respectively

$$W_1(s) = \left[\begin{array}{c|c} \mathbf{F}_{W1} & \mathbf{G}_{W1} \\ \hline \mathbf{H}_{W1} & \mathbf{D}_{W1} \end{array}\right] \qquad W_2(s) = \left[\begin{array}{c|c} \mathbf{F}_{W2} & \mathbf{G}_{W2} \\ \hline \mathbf{H}_{W2} & \mathbf{D}_{W2} \end{array}\right]. \tag{4.45}$$

It is then necessary to find one input and two outputs which will yield the two desired transfer functions, the sensitivity and complementary sensitivity. One way to do this is to add a fictitious disturbance to the measurement equation:

$$\mathbf{y} = \mathbf{Hx} + \mathbf{Du} + \mathbf{d}. \tag{4.46}$$

The sensitivity is then the transfer function from this disturbance to the measurement. The complementary sensitivity is the transfer function from this disturbance to the disturbance-free measurement:

$$S(s) = \frac{y(s)}{d(s)} \tag{4.47}$$

$$C(s) = 1 - \frac{y(s)}{d(s)} \tag{4.48}$$

The weights of Equation 4.45 must be applied to these outputs to produce the desired 'error' output:

$$\mathbf{e}(s) = \begin{bmatrix} e_1(s) \\ e_2(s) \end{bmatrix} = \begin{bmatrix} W_1(s)S(s) \\ W_2(s)C(s) \end{bmatrix} \mathbf{d}(s) \tag{4.49}$$

One additional error output must be defined to ensure that the control is directly weighted in the error output. This is a technical requirement of the H_∞ design procedure (and the LQG design procedure) which bounds the magnitude of the control signal. This error output is not necessary if one of the other error outputs contains a non-zero weighting of the control term, but it is prudent to include it. In state-space form, this is written

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\mathbf{z}}_1 \\ \dot{\mathbf{z}}_2 \\ \mathbf{e} \\ \mathbf{y} \end{bmatrix} = \left[\begin{array}{ccc|c|c} \mathbf{F} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{G} \\ \mathbf{G}_{W1}\mathbf{H} & \mathbf{F}_{W1} & \mathbf{0} & \mathbf{G}_{W1} & \mathbf{G}_{W1}\mathbf{D} \\ \mathbf{G}_{W2}\mathbf{H} & \mathbf{0} & \mathbf{F}_{W2} & \mathbf{0} & \mathbf{G}_{W2}\mathbf{D} \\ \hline \mathbf{D}_{W1}\mathbf{H} & \mathbf{H}_{W1} & \mathbf{0} & \mathbf{D}_{W1} & \mathbf{D}_{W1}\mathbf{D} \\ \mathbf{D}_{W2}\mathbf{H} & \mathbf{0} & \mathbf{H}_{W2} & \mathbf{0} & \mathbf{D}_{W2}\mathbf{D} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \varepsilon \\ \hline \mathbf{H} & \mathbf{0} & \mathbf{0} & 1 & \mathbf{D} \end{array}\right] \begin{bmatrix} \mathbf{x} \\ \mathbf{z}_1 \\ \mathbf{z}_2 \\ \mathbf{d} \\ \mathbf{u} \end{bmatrix} \tag{4.50}$$

This description will be used in the H_∞ designs of Chapter 5.

Chapter 5

Control Configuration and Design

5.1 Procedure

The design techniques presented in the preceding chapter can be used to aid in the selection of the 'best' control system configuration (i.e. sensor and actuator placement and type). In Section 5.2, a 3 stage compressor is studied. Section 5.2.1 presents a nominal case to examine sensor and actuator locations using the LQG, state and control costs for the LQG design and the state and control costs for the H_∞ design. The state and control costs of the LQG designs are essentially a breakdown of the LQG cost, although the LQG cost will be greater than the sum due to its different assumed system noise. The nominal case has certain assumptions and free parameters, hence, it is necessary to test variations from the nominal to validate the selection of the nominal case as a representative case which can be used for sensor and actuator placement studies. These results are presented in Section 5.2.2 In Section 5.2.3, other sensor types and actuator variations are examined for comparison to the nominal case and to each other. Section 5.2.4 presents designs for the 'best' sensor and actuator locations for the nominal case and the cases of Section 5.2.3. Section 5.2.5 presents a discussion and summary of the 3 stage compressor results. Section 5.3 presents the design procedure and conclusions based on the configuration comparisons and designs.

Table 5.1 - Data for the nominal case.

Harmonic	1
Flow Coefficient	.461 (2.1% below neutral stability)
Rotor Speed	100% of design speed
Measurement	$\delta P/P$
Jet Velocity	2 * mean axial velocity in the gap
Jet Total Temperature	mean total temperature in the gap
Correlations	supplied by manufacturer

5.2 LQG and H_∞ Designs on a 3 Stage Compressor

Using the control system design methodologies developed in Chapter 4, different control system configurations for the 3 stage compressor will be examined in an attempt to find the 'best' set. The 3 stage compressor is described in Section 3.4.1. The supporting tables of this section are in Appendix E.

The state-space model with no gaps will be used for all of the designs because of its smaller size.

5.2.1 Nominal Case

This section will present a nominal case for which a best sensor/actuator location pair can be chosen. The first harmonic was chosen since it is usually the harmonic to reach neutral stability first as the flow coefficient is lowered. At this flow coefficient for the first harmonic, the rotating stall eigenvalue (i.e. the lowest frequency eigenvalue in the direction of the rotor) is the only unstable eigenvalue. The data for the nominal case is shown in Table 5.1. The supporting tables of this section are in Appendix E.

From the LQG and H_∞ designs, the sensor/actuator pairs 'near' $(k_a, k_s) = (2,1)$ are the best choices in terms of all the costs. Except for the pair (1,1), they meet the specifications.

5.2.2 Variations from Nominal

This section will present cases which vary from the nominal to examine whether or not the best sensor/actuator locations hold up under different assumptions. The cases to be examined are: the zeroth and second harmonics, a 2.3% change in the flow coefficient, a rotor speed change to 70%, a different set of correlations to predict the mean flow, and a different definition of LQG and state costs using velocity perturbations instead of static pressure perturbations. For each of these cases, new designs will be done and new performance indexes will be computed using both the LQG and the H_∞ design procedures.

5.2.2.1 Case 2: Zeroth Harmonic

It is necessary to determine if the best sensor/actuator location pair for the first harmonic is the best location pair for the zeroth harmonic. The dynamics of the zeroth harmonic are expected to be somewhat different from the non-zero harmonics due to the interaction of the plenum volume with the flow. For the flow coefficient of the nominal case, the two conjugate symmetric surge eigenvalues (i.e. the lowest frequency eigenvalues) are unstable at frequencies near the first harmonic rotating stall frequency, and two higher frequency eigenvalues are stable, but very nearly neutrally stable. The data for the second case is shown in Table 5.2. The supporting tables of this section are in Appendix E.

The results show that pairs around $(k_u, k_s) = (8,8)$ and $(8,1)$ are the best selections, except for those which do not meet the specifications. The best locations of the nominal case, around $(2,1)$, are not able to meet the specifications.

5.2.2.2 Case 3: Second Harmonic

It is necessary to determine if the best sensor/actuator pair location for the first harmonic is also good for the higher harmonics. Checking the second harmonic should be sufficient, since it is the second of the non-zero harmonics to reach neutral stability as

Table 5.2 - Data for case 2.

Harmonic	0
Flow Coefficient	.461 (2.1% below neutral stability)
Rotor Speed	100% of design speed
Measurement	$\delta P/P$
Jet Velocity	2 * mean axial velocity in the gap
Jet Total Temperature	mean total temperature in the gap
Correlations	supplied by manufacturer

Table 5.3 - Data for case 3.

Harmonic	2
Flow Coefficient	.461 (2.1% below neutral stability)
Rotor Speed	100% of design speed
Measurement	$\delta P/P$
Jet Velocity	2 * mean axial velocity in the gap
Jet Total Temperature	mean total temperature in the gap
Correlations	supplied by manufacturer

the flow coefficient is lowered. The dynamics of the second and higher harmonics are similar to the first harmonic dynamics since, as opposed to the zeroth harmonic, the exit duct conditions are the same. The data for the third case is shown in Table 5.3. The supporting tables of this section are in Appendix E.

The LQG design results show that the pairs near $(k_a, k_s) = (2,1)$ have the lowest costs. The H_∞ design results show that the control does not have to work very hard to meet the constraints for any pair, so all the state costs are very similar. However, the control costs near (2,1) are the lowest.

Table 5.4 - Data for case 4.

Harmonic	1
Flow Coefficient	.4506 (4.5% below neutral stability)
Rotor Speed	100% of design speed
Measurement	$\delta P/P$
Jet Velocity	2 * mean axial velocity in the gap
Jet Total Temperature	mean total temperature in the gap
Correlations	supplied by manufacturer

5.2.2.3 Case 4: 2.3% Flow Coefficient Change

It s necessary to check whether the costs obtained at one flow coefficient can be generalized over a range of flow coefficients, so a lower flow coefficient will be examined. The data for the fourth case is shown in Table 5.4. The supporting tables of this section are in Appendix E.

The LQG design results show that the pairs around $(k_a, k_s) = (2,1)$ have the lowest costs. However, since the flow coefficient was reduced from the nominal, none of these pairs were able to meet the constraints in the H_∞ design results. All of the pairs which satisfy the constraint at this flow coefficient have an unacceptably high control cost. This is an important well-known result which shows that small mean-square cost functions do not necessarily imply good robustness.

5.2.2.4 Case 5: Rotor Speed Change to 70%

It may be necessary to control rotating stall when the compressor is operating at a different rotor speed than the design speed. This section will present an example with the rotor speed at 70% design speed. The data for the fifth case is shown in Table 5.5. The supporting tables of this section are in Appendix E.

Table 5.5 - Data for case 5.

Harmonic	1
Flow Coefficient	.2644 (1.2% below neutral stability)
Rotor Speed	100% of design speed
Measurement	$\delta P/P$
Jet Velocity	2 * mean axial velocity in the gap
Jet Total Temperature	mean total temperature in the gap
Correlations	supplied by manufacturer

Table 5.6 - Data for case 6.

Harmonic	1
Flow Coefficient	.3898 (7.5% below neutral stability)
Rotor Speed	100% of design speed
Measurement	$\delta P/P$
Jet Velocity	2 * mean axial velocity in the gap
Jet Total Temperature	mean total temperature in the gap
Correlations	supplied by manufacturer

The LQG design results show that the pairs near (k_u, k_s) = (2,1) are the best overall, while the H_∞ design results show that a smaller set near (2,1) are able to meet the specifications with similarly low costs.

5.2.2.5 Case 6: Predicted Correlations

The correlations supplied by the manufacturer may not capture the actual conditions exactly. It is therefore necessary to examine a case in which an independent set of correlations are used to predict the mean-line. The predicted correlations are described by Bonnaure [16]. The data for the sixth case is shown in Table 5.6. The supporting tables of this section are in Appendix E.

Table 5.7 - Data for case 7.

Harmonic	1
Flow Coefficient	.461 (2.1% below neutral stability)
Rotor Speed	100% of design speed
Measurement	$\delta P/P$
Jet Velocity	2 * mean axial velocity in the gap
Jet Total Temperature	mean total temperature in the gap
Correlations	supplied by manufacturer

The LQG results show that the pairs around $(k_a, k_s) = (2,1)$ have the best costs. The H_∞ design results show that a smaller set of pairs around (2,1) are able to meet the constraints with similarly low costs.

5.2.2.6 Case 7: Axial Velocity Weightings

One of the assumptions on the state and LQG costs is that static pressure should be used in the cost functions. However, axial velocity is another option. This section uses axial velocity in the LQG and state cost functions which can be compared against the nominal case which uses static pressure in the LQG and state cost functions. The data for the seventh case is shown in Table 5.7. The supporting tables of this section are in Appendix E.

The results are for this case are essentially equal to the results for the nominal case in terms of relative costs between placements.

5.2.2.7 Summary of Variation Results

All of the cases presented in this section, with the exception of case 2, the zeroth harmonic, resulted in the same group of sensor/actuator pairs as having the lowest state and control costs while still able to meet the specification as the nominal case. This validates the use of a nominal case to choose sensor and actuator locations without the need to check all of the variations above.

The zeroth harmonic, as stated above, has a different interaction with the plenum than the non-zero harmonics. This is most likely the reason why the results differ for this case. It may be necessary to implement an independent control scheme to stabilize surge.

5.2.3 Sensor and Actuator Variations

A few special cases will be examined which do not fall into the category of variations on the nominal to verify the nominal case results. The cases to be examined are: using a zero axial velocity jet actuator, using a total pressure sensor, using an axial velocity sensor, and using velocity weightings in the LQG and state cost functions, instead of static pressure weightings, while again using velocity measurements.

5.2.3.1 Case 8: Zero Velocity Jet Injection

It has been suggested [20] that the dominant factor of a jet injector is the momentum change which it gives to the flow. This case will examine taking away momentum rather than adding it, by injecting air with zero axial velocity rather than twice the mean axial velocity. The advantage of this is that it may be easier to implement. The data for the eighth case is shown in Table 5.8. The supporting tables of this section are in Appendix E.

The LQG results show that the pairs near $(k_u, k_s) = (8,1)$ and $(1,1)$ have the lowest costs. The H_∞ design results show that a smaller set of pairs near (8,1) are able to meet the constraints with the lowest costs. These costs can be compared directly against the nominal case. The pair (8,1), for the LQG design, has 8.7, 3.5 and 3.5 LQG, state and control costs, respectively, and for the H_∞ design, has 4.3 and 3.5 state and control costs, respectively. Comparing this to the nominal case, the pair (2,1), for the LQG design, has 9.9, 5.7 and 2.4 LQG, state and control costs, respectively, and for the H_∞ design, has 6.3 and 2.1 state and control costs, respectively. In terms of these costs, the best pair for the

Table 5.8 - Data for case 8.

Harmonic	1
Flow Coefficient	.461 (2.1% below neutral stability)
Rotor Speed	100% of design speed
Measurement	$\delta P/P$
Jet Velocity	0
Jet Total Temperature	mean total temperature in the gap
Correlations	supplied by manufacturer

Table 5.9 - Data for case 9.

Harmonic	1
Flow Coefficient	.461 (2.1% below neutral stability)
Rotor Speed	100% of design speed
Measurement	$\delta P_t/P_t$
Jet Velocity	2 * mean axial velocity in the gap
Jet Total Temperature	mean total temperature in the gap
Correlations	supplied by manufacturer

zero velocity jet injection is at least as good as the best pair for the nominal case of injection at twice the mean flow velocity.

5.2.3.2 Case 9: Total Pressure Sensor

Currently under consideration for sensor type is a total pressure sensor. This section will examine the cost functions when measuring total pressure perturbations. The data for the ninth case is shown in Table 5.9. The supporting tables of this section are in Appendix E.

The LQG results show that the pairs around (k_u, k_s) = (2,1) and (2,8) have the lowest costs. The H_∞ design results show that a smaller set of pairs around (2,8) are able to meet the constraints with the lowest costs. These costs can be compared directly

Table 5.10 - Data for case 10.

Harmonic	1
Flow Coefficient	.461 (2.1% below neutral stability)
Rotor Speed	100% of design speed
Measurement	$\delta V_X/a$
Jet Velocity	2 * mean axial velocity in the gap
Jet Total Temperature	mean total temperature in the gap
Correlations	supplied by manufacturer

against the nominal case. The pair (2,8), for the LQG design, has 6.6, 1.9 and 3.2 LQG state and control costs, respectively, and for the H_∞ design, has 1.8 and 3.0 state and control costs, respectively. Comparing this to the nominal case, the pair (2,1), for the LQG design, has 9.9, 5.7 and 2.4 LQG state and control costs, respectively, and for the H_∞ design, has 6.3 and 2.1 state and control costs, respectively. In terms of these costs, the best pair for total pressure measurement is at least as good as the best pair for the nominal case of measuring static pressure perturbations.

5.2.3.3 Case 10: Axial Velocity Sensor

Previous results on low speed compressors show that velocity sensors have advantages over static pressure sensors [20]. This case will examine the cost functions when measuring axial velocity perturbations. The data for the tenth case is shown in Table 5.10. The supporting tables of this section are in Appendix E.

The LQG results show that the pairs near (k_u, k_s) = (2,1) and (2,8) have the lowest costs. The H_∞ design results show that a smaller set of pairs near (2,1) are able to meet the constraints with the lowest costs. These costs can be compared directly against the nominal case. The pair (2,1), for the LQG design, has 10.1, 6.0 and 2.5 LQG state and control costs, respectively, and for the H_∞ design, has 6.3 and 2.1 state and control costs, respectively. Comparing this to the nominal case, the pair (2,1), for the LQG design, has

9.9, 5.7 and 2.4 LQG state and control costs, respectively, and for the H_∞ design, has 6.3 and 2.1 state and control costs, respectively. In terms of these costs, the best pair for axial velocity measurement is at least as good as the best pair for the nominal case of measuring static pressure perturbations.

One comment about axial velocity measurements is that in terms of these costs, the sensors are less sensitive to placement than static or total pressure sensors.

5.2.4 Examination of Best Sensor/Actuator Pairs

Now that a best sensor and actuator pair has been determined for each type of sensor and actuator parameter, these can be compared to find which one can stabilize the compressor to the lowest flow coefficient while still meeting the specifications. Not only do the specifications have to be met at the design point, but also at increased flow coefficients. In each of the cases to be presented, the neutral stability operating point is the worst case in terms of plant changes due to operating point changes, so the designs will only be tested off of the design point at neutral stability. Finally, the compensators in each case will be reduced to a size amenable to implementation.

5.2.4.1 Minimum Flow Coefficient

The sensor/actuator locations to be examined in each case are: nominal case $(k_u, k_s) = (2,1)$, case 8 $(k_u, k_s) = (8,1)$, case 9 $(k_u, k_s) = (2,8)$, and case 10 $(k_u, k_s) = (2,8)$. While in case 10, the pairs (2,1) and (2,8) seem to be equal in terms of costs, the pair (2,8) has a lower minimum flow coefficient. The results are summarized in Table 5.11. These are coarse numbers which are the minimum flow coefficients as the flow coefficient was decreased by approximately 0.01. Case 4, one of the variations of the nominal, in fact found that only six placements of the static pressure sensor could meet the constraints at ϕ=.4506, but all with unacceptably high control costs.

Table 5.11 - Minimum flow coefficients for different cases.

	Jet Velocity	Measurement	Minimum Flow Coefficient	% Decrease
Uncontrolled	-	-	.4703	-
Nominal Case	$V_j=2V_x$	$\delta P/P$	.4610	2.0%
Case 8	$V_j=0$	$\delta P/P$	.4610	2.0%
Case 9	$V_j=2V_x$	$\delta P_t/P_t$	.4403	6.4%
Case 10	$V_j=2V_x$	$\delta V_x/a$	.4403	6.4%

One note about the results is that cases 9 and 10 were able to control the compressor to flow coefficients where two 'acoustic', or higher frequency, modes were unstable in addition to the rotating stall mode.

Some insight into the results are presented in Figures 5.1-5.4 for the nominal case and cases 8-10, respectively. Clearly the discussion in Section 4.1.2 applies to the nominal case in Figure 5.1. It has a non-minimum phase zero fairly near the unstable eigenvalue and it has one of the highest (worst) flow coefficients. However, Figures 5.3 and 5.4, for cases 9 and 10, show no non-minimum phase zeros within the imposed bandwidth of the system and have the lowest (best) flow coefficients. Figure 5.2, for case 8, however, shows only one minimum phase zero anywhere near the unstable eigenvalue and two distant non-minimum phase zeros, but still has one of the highest (worst) flow coefficients. This case may suffer due to the lack of interlaced, minimum phase zeros, such as in Figures 5.3 and 5.4.

The complementary sensitivity and sensitivity transfer functions at the flow coefficients of Table 5.11 are shown in Figures 5.5-5.8 for the nominal case and cases 8-10, respectively. All of these show that the sensitivity function is very close to its bound at frequencies less than the bandwidth.

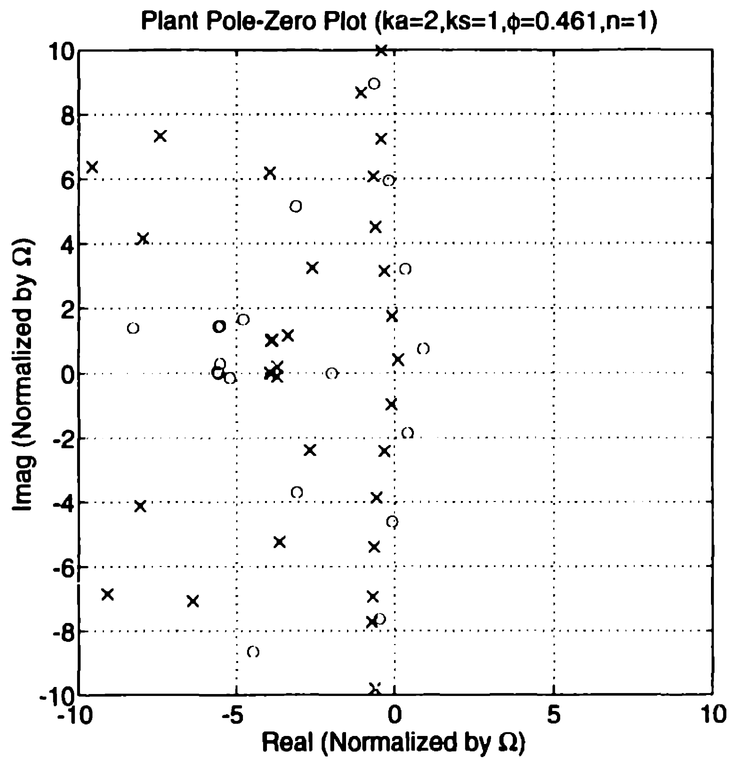

Figure 5.1 - Pole-zero plot for nominal case - (k_a, k_s) = (2,1).

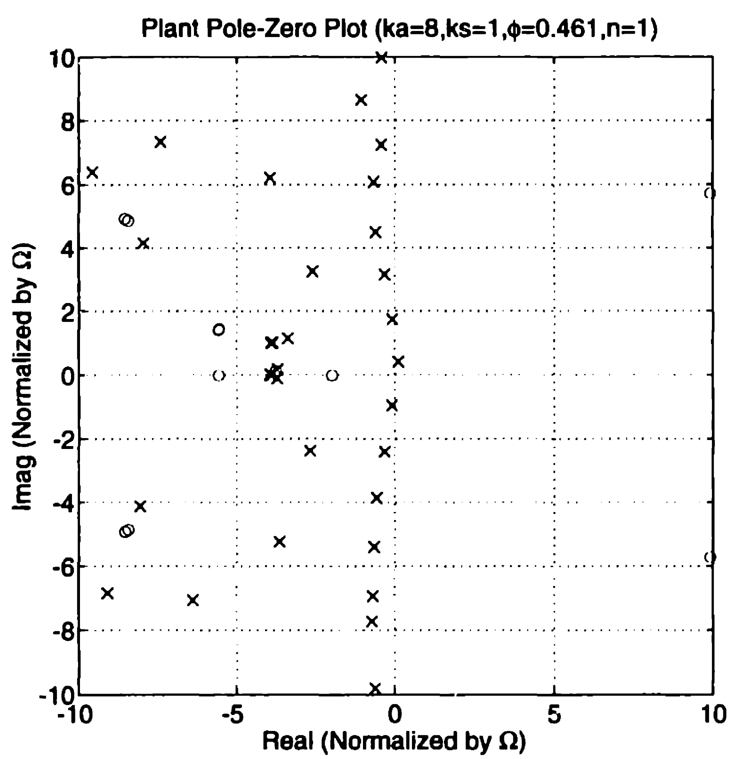

Figure 5.2 - Pole-zero plot for case 8 - (k_a, k_s) = (8,1).

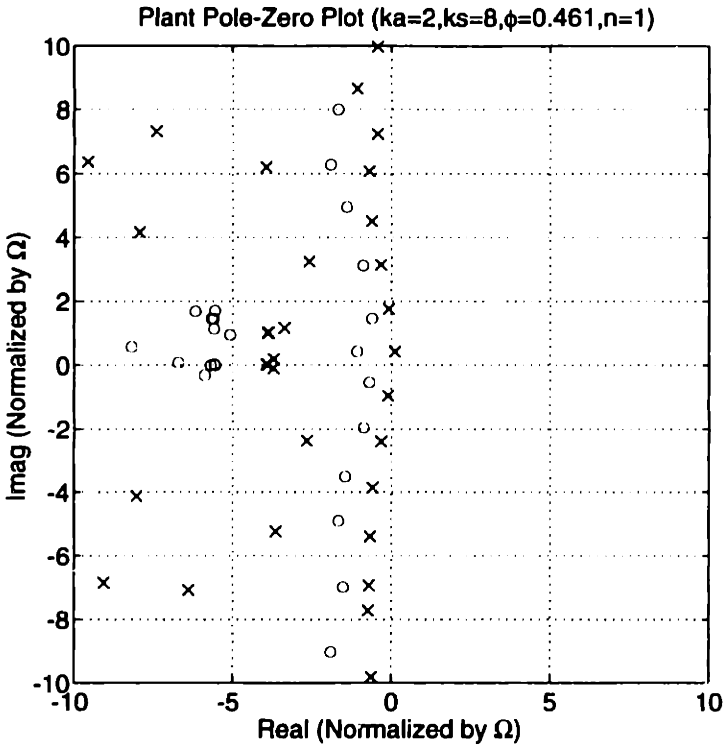

Figure 5.3 - Pole-zero plot for case 9 - $(k_a, k_s) = (2,8)$.

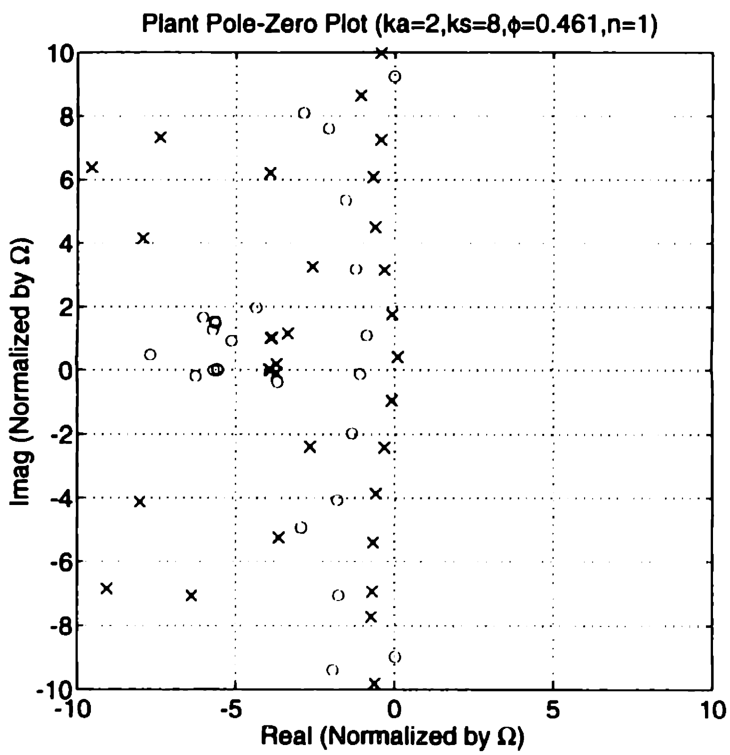

Figure 5.4 - Pole-zero plot for case 10 - $(k_a, k_s) = (2,8)$.

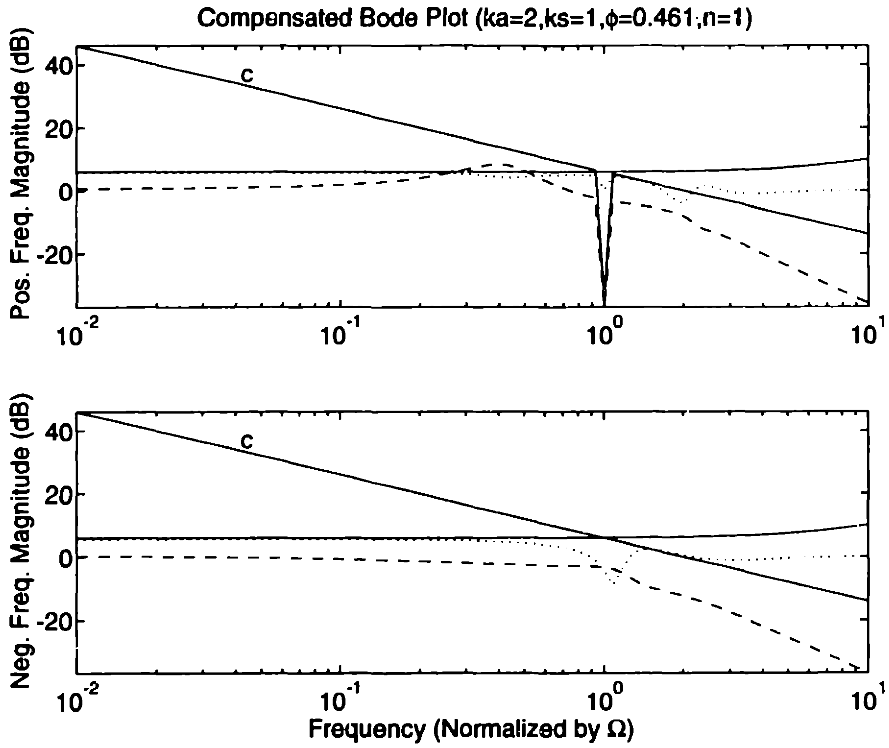

Figure 5.5 - Comp. sens. (dashed) with constraint (solid-c) and sens. (dotted) with constraint (solid) for nominal case - $(k_a, k_s) = (2,1)$, ϕ=.461.

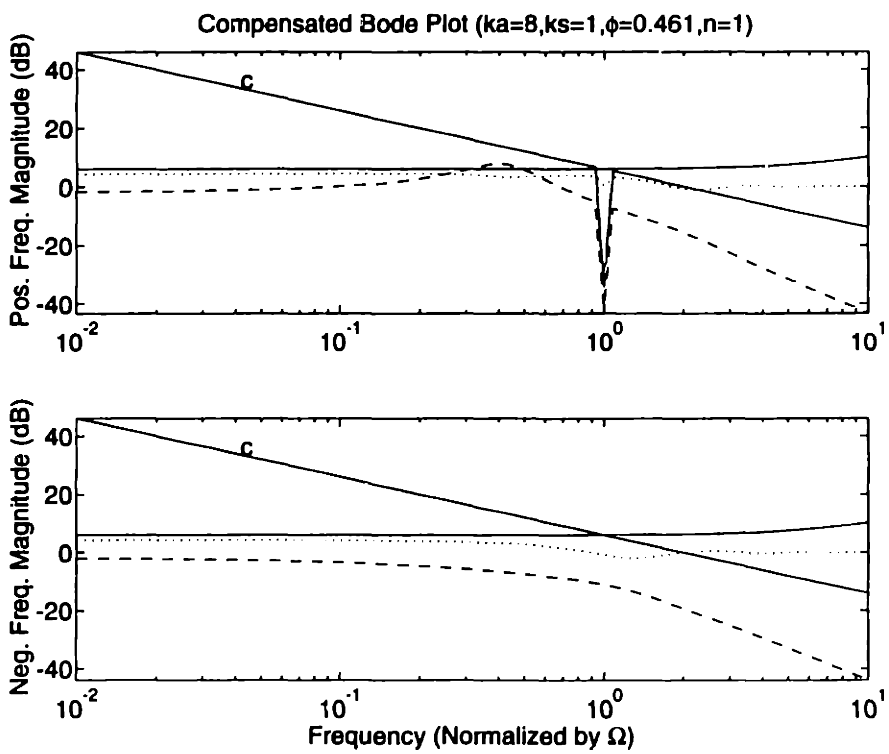

Figure 5.6 - Comp. sens. (dashed) with constraint (solid-c) and sens. (dotted) with constraint (solid) for case 8 - $(k_a, k_s) = (8,1)$, ϕ=.461.

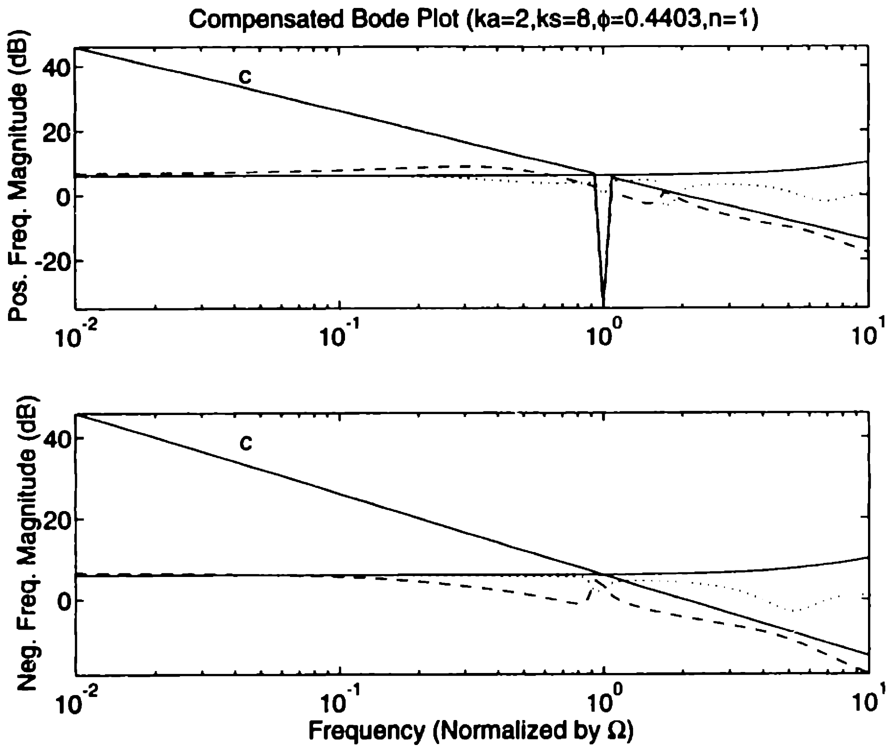

Figure 5.7 - Comp. sens. (dashed) with constraint (solid-c) and sens. (dotted) with constraint (solid) for case 9 - (k_a, k_s) = (2,8), ϕ=.4403.

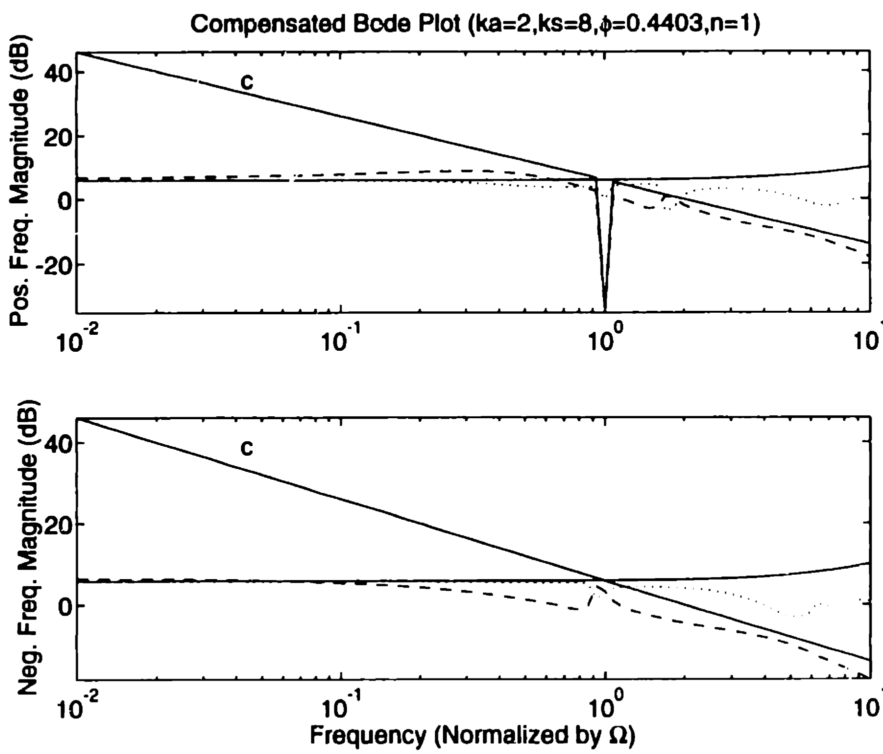

Figure 5.8 - Comp. sens. (dashed) with constraint (solid-c) and sens. (dotted) with constraint (solid) for case 10 - (k_a, k_s) = (2,8), ϕ=.4403.

5.2.4.2 Robustness To Operating Point Changes

Now that control designs have been done at the lowest flow coefficients for each case, it is necessary to determine whether these designs are robust to operating point changes. To be robust to operating point changes, the system must remain stable and the specifications must still be met as the flow coefficient is raised. As was stated above, the neutral stability operating point is the worst case variation, so this operating point will be shown, although the results apply to all flow coefficients above this one.

Figures 5.9-5.12 show the complementary sensitivity and sensitivity transfer functions when the compensator was designed for the minimum attainable flow coefficient, but the compressor is operating at neutral stability. For the nominal case and case 8, which could only decrease the flow coefficient by about 2%, the specifications are still met at neutral stability. For cases 9 and 10, however, which could decrease the flow coefficient by more than 6%, the specifications are not met at neutral stability. This is due to a larger plant deviation than the nominal case and case 8, rather than worse compensators. Referring to Section 4.1.3, the frequencies at which the specifications are violated are those near the upper gain margin.

The compensators of cases 9 and 10 must be altered to achieve robustness to operating point changes. Examining the frequencies where the specifications are violated, a modified constraint on the complementary sensitivity can be imposed to account for these frequencies. However, the result of imposing more restrictive constraints is that the constraints are no longer able to be satisfied at the flow coefficients of Table 5.11. The flow coefficient has then been raised by about 0.005 (about 1%) to account for the modified bounds. Figures 5.13 and 5.14 show the modified bounds and the complementary sensitivity and sensitivity functions for cases 9 and 10, respectively. Figures 5.15 and 5.16 show the original specifications and the complementary sensitivity and sensitivity functions for cases 9 and 10, respectively, when the compensators were designed at the new minimum and the compressor is operating at neutral stability. A summary of the minimum flow coefficients which are robust to operating point changes is presented in Table 5.12.

Table 5.12 - Minimum flow coefficients for different cases with robustness to operating point changes.

	Jet Velocity	Measurement	Minimum Flow Coefficient	% Decrease
Uncontrolled	-	-	.4703	-
Nominal Case	$V_j=2V_x$	$\delta P/P$	.4610	2.0%
Case 8	$V_j=0$	$\delta P/P$	.4610	2.0%
Case 9	$V_j=2V_x$	$\delta P_t/P_t$	.4455	5.3%
Case 10	$V_j=2V_x$	$\delta V_x/a$	.4455	5.3%

5.2.4.3 Compensator Model Reduction

For implementation, it is desired to have a low order compensator due to computational constraints. This section will present the lowest order compensators (using a balance and truncate algorithm [25]) which can still satisfy the specifications of the previous section at the lowest flow coefficient (i.e. the original specifications for the nominal and case 8, and the modified specifications for cases 9 and 10). The results are summarized in Table 5.13.

Table 5.13 - Minimum compensator order for different cases with robustness to operating point changes.

	Jet Velocity	Measurement	Minimum Order
Nominal Case	$V_j=2V_x$	$\delta P/P$	10
Case 8	$V_j=0$	$\delta P/P$	10
Case 9	$V_j=2V_x$	$\delta P_t/P_t$	10
Case 10	$V_j=2V_x$	$\delta V_x/a$	16

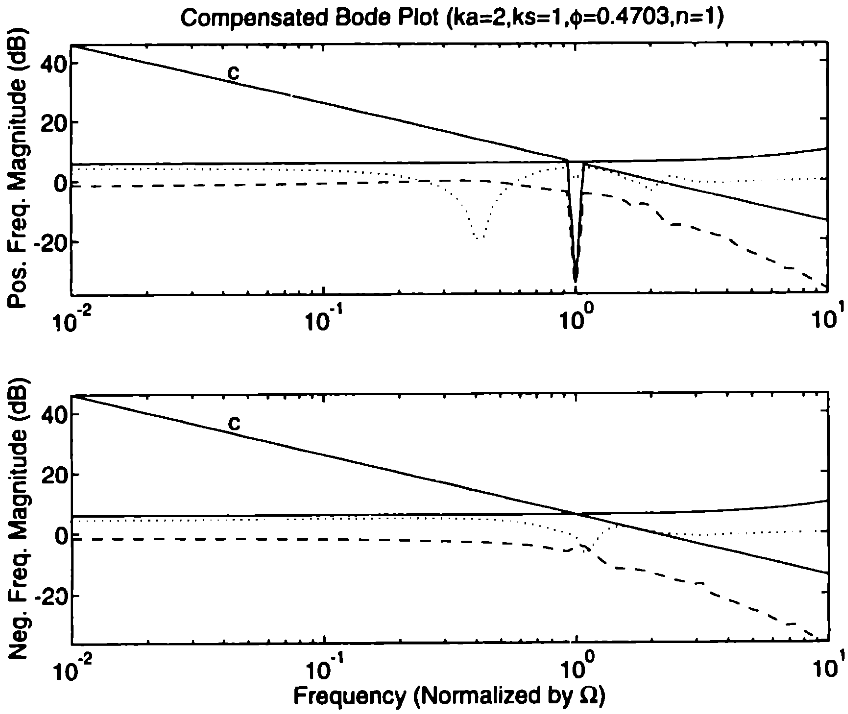

Figure 5.9 - Comp. sens. (dashed) with constraint (solid-c) and sens. (dotted) with constraint (solid) at neutral stability using compensator design at ϕ=.461 for nominal case - (k_a, k_s) = (2,1).

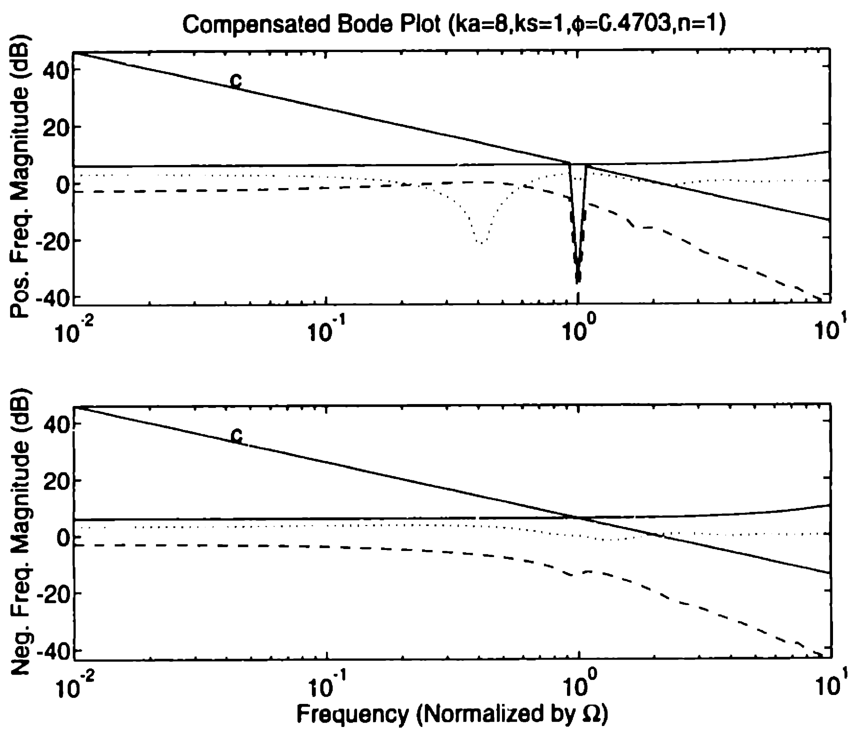

Figure 5.10 - Comp. sens. (dashed) with constraint (solid-c) and sens. (dotted) with constraint (solid) at neutral stability using compensator design at ϕ=.461 for case 8 - (k_a, k_s) = (8,1).

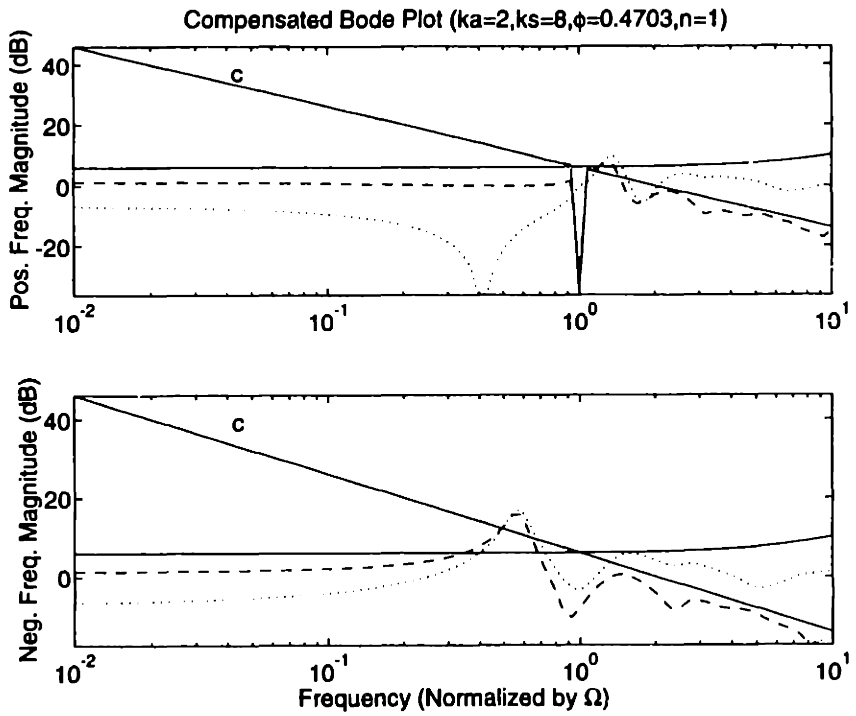

Figure 5.11 - Comp. sens. (dashed) with constraint (solid-c) and sens. (dotted) with constraint (solid) at neutral stability using compensator design at ϕ=.4403 for case 9 - $(k_a, k_s) = (2,8)$.

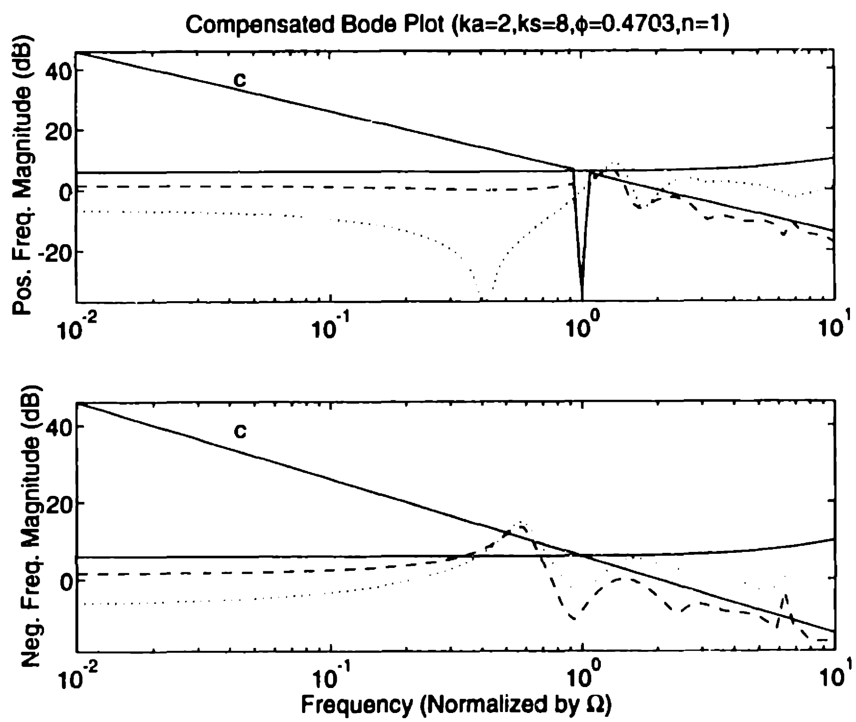

Figure 5.12 - Comp. sens. (dashed) with constraint (solid-c) and sens. (dotted) with constraint (solid) at neutral stability using compensator design at ϕ=.4403 for case 10 - $(k_a, k_s) = (2,8)$.

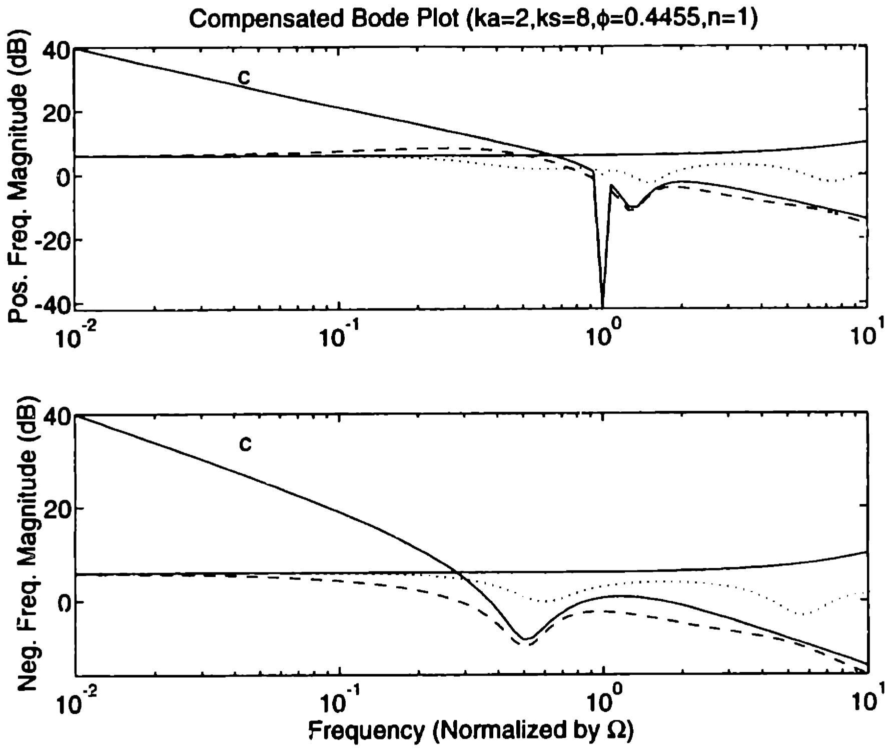

Figure 5.13 - Comp. sens. (dashed) with modified constraint (solid-c) and sens. (dotted) with constraint (solid) for case 9 - (k_a, k_s) = (2,8), ϕ=.4455.

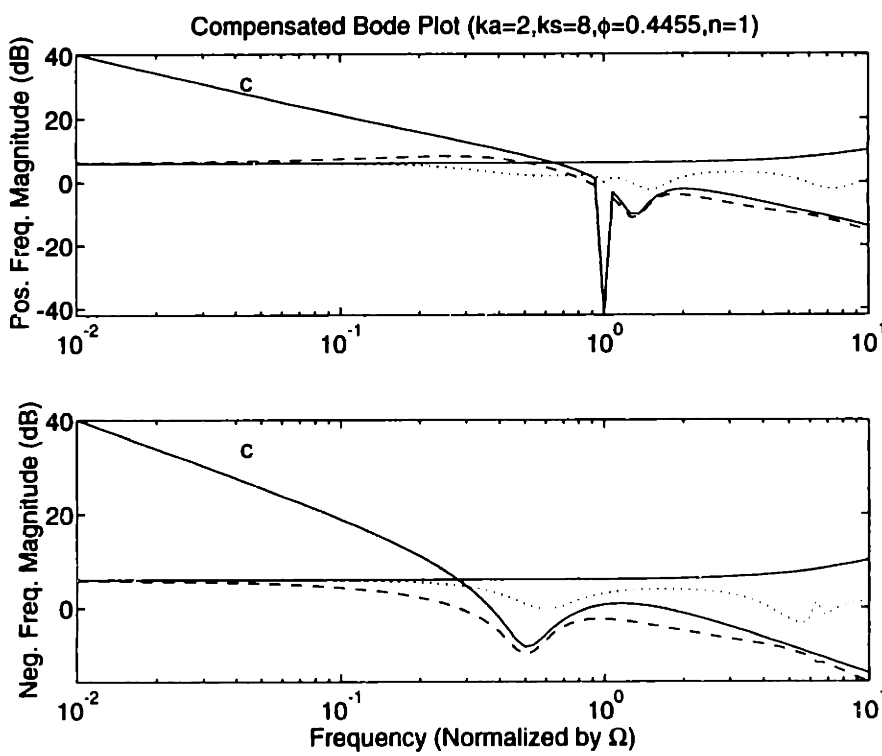

Figure 5.14 - Comp. sens. (dashed) with modified constraint (solid-c) and sens. (dotted) with constraint (solid) for case 10 - (k_a, k_s) = (2,8), ϕ=.4455.

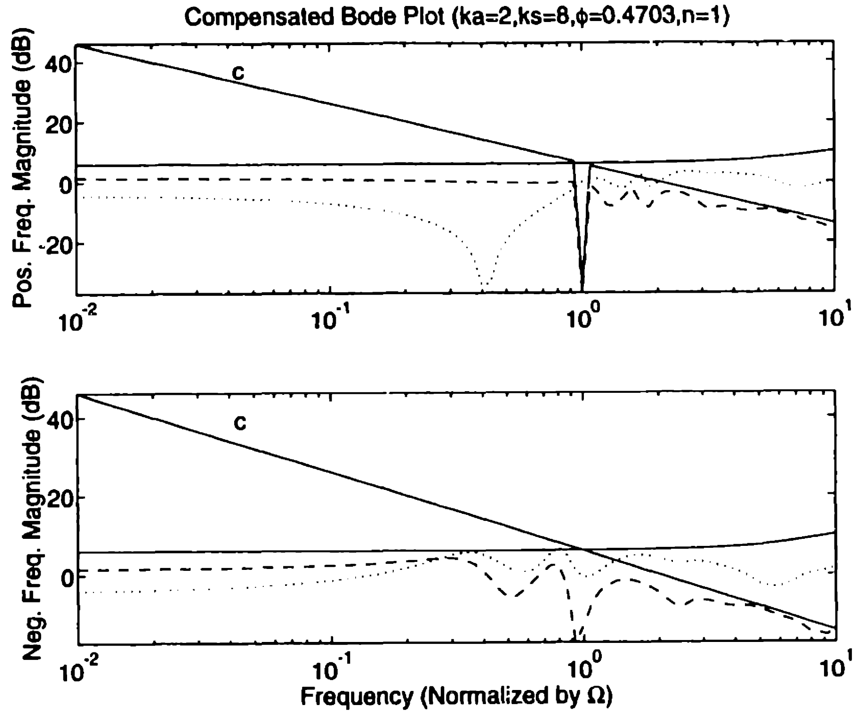

Figure 5.15 - Comp. sens. (dashed) with constraint (solid-c) and sens. (dotted) with constraint (solid) at neutral stability using compensator design at ϕ=.4455 for case 9 - (k_a, k_s) = (2,8).

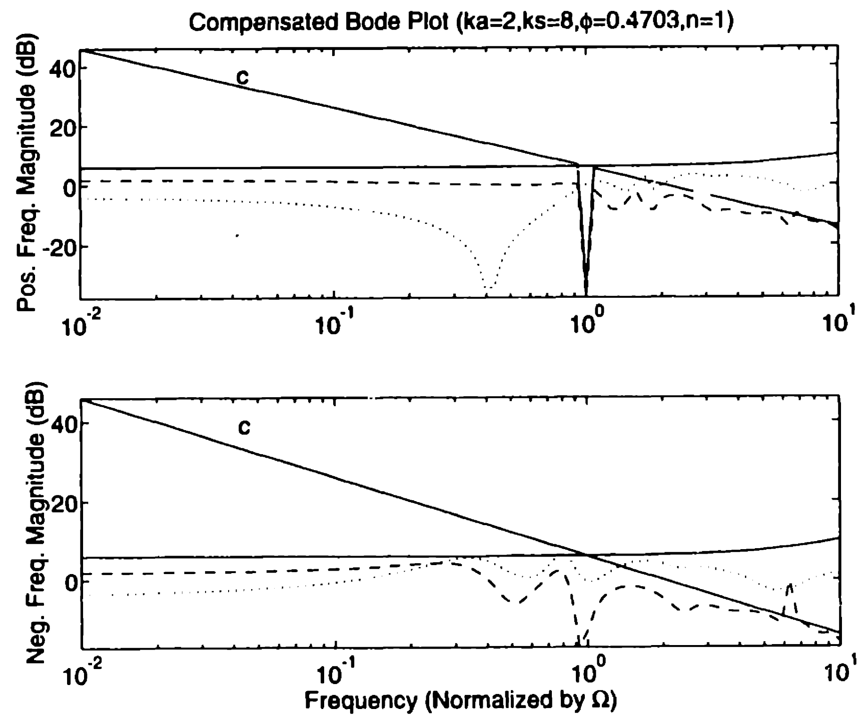

Figure 5.16 - Comp. sens. (dashed) with constraint (solid-c) and sens. (dotted) with constraint (solid) at neutral stability using compensator design at ϕ=.4455 for case 10 - (k_a, k_s) = (2,8).

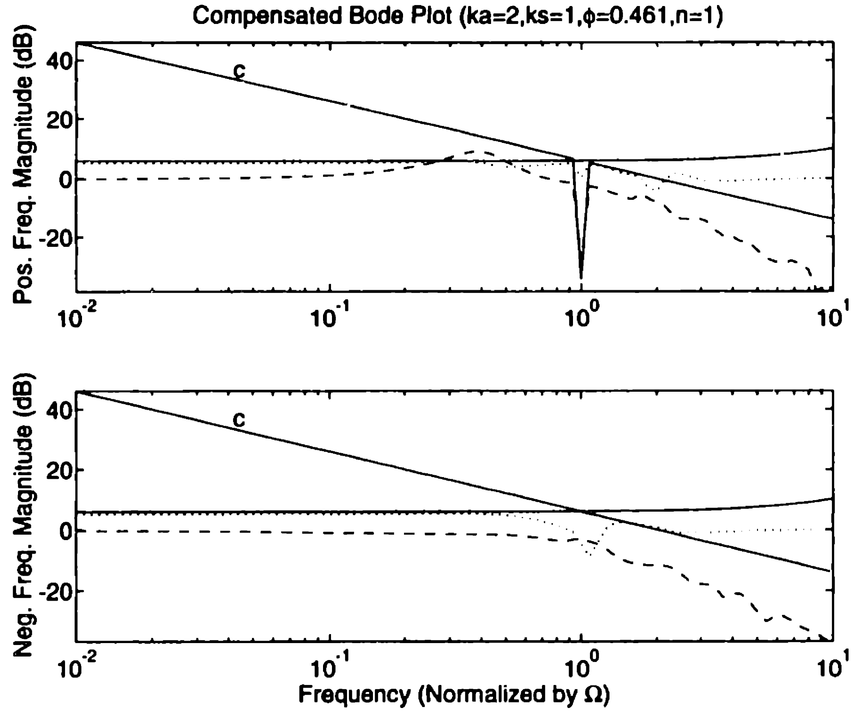

Figure 5.17 - Comp. sens. (dashed) with constraint (solid-c) and sens. (dotted) with constraint (solid) for nominal case - $(k_a, k_s) = (2,1)$, ϕ=.461, compensator order = 10.

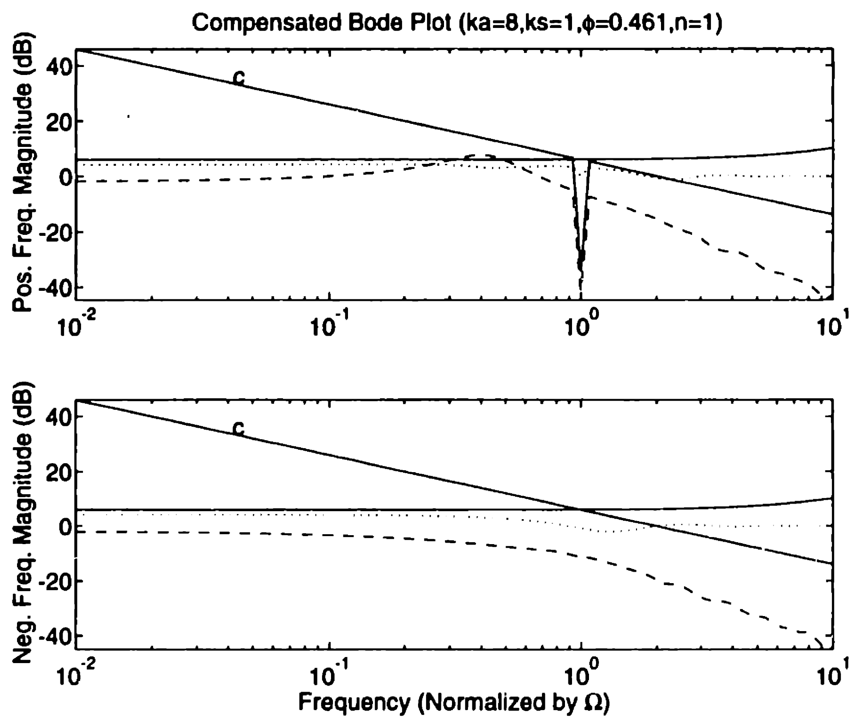

Figure 5.18 - Comp. sens. (dashed) with constraint (solid-c) and sens. (dotted) with constraint (solid) for case 8 - $(k_a, k_s) = (8,1)$, ϕ=.461, compensator order = 10.

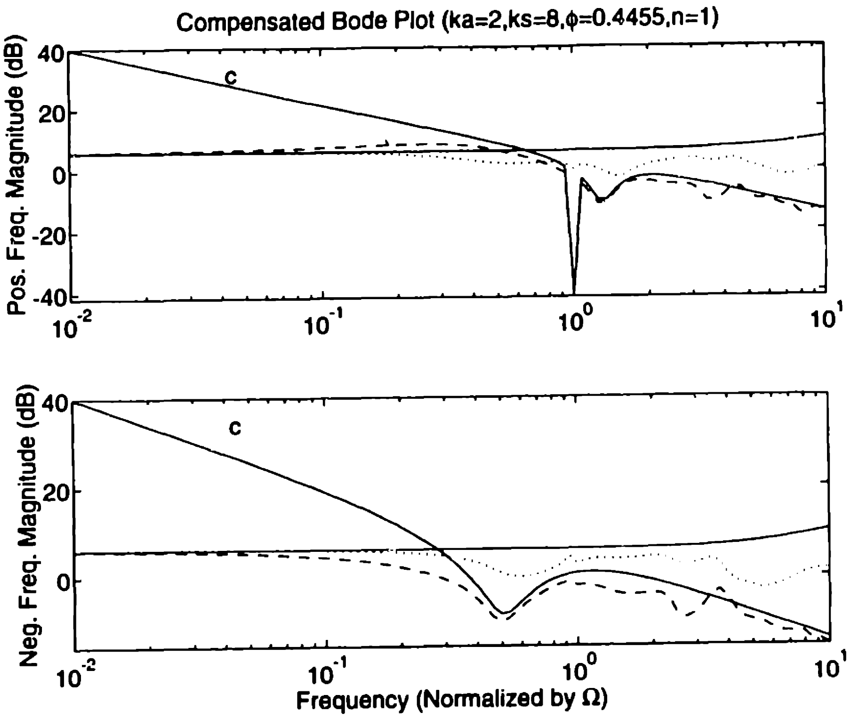

Figure 5.19 - Comp. sens. (dashed) with constraint (solid-c) and sens. (dotted) with constraint (solid) for case 9 - $(k_a, k_s) = (2,8)$, ϕ=.4455, compensator order = 10.

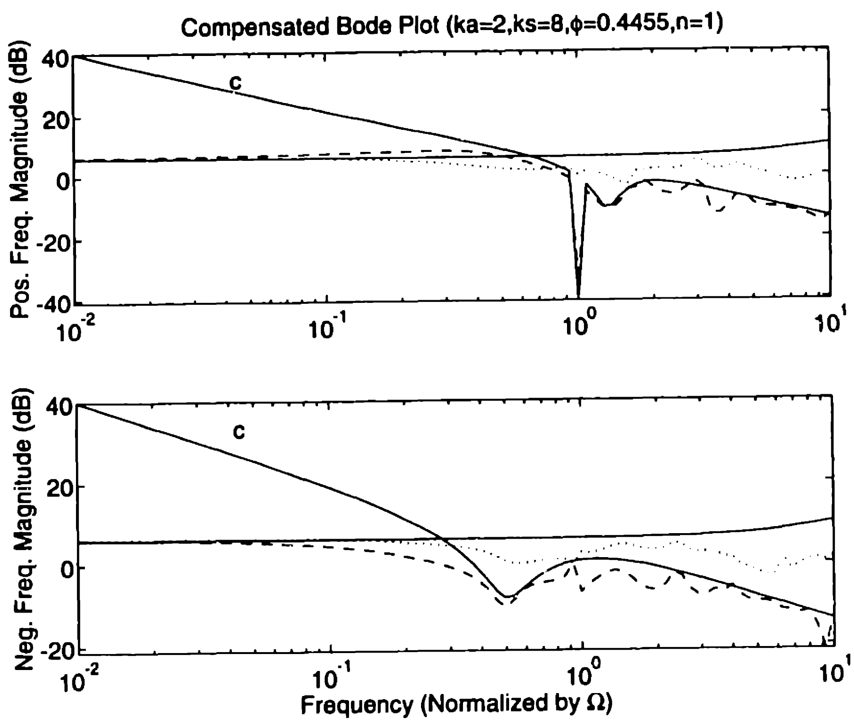

Figure 5.20 - Comp. sens. (dashed) with constraint (solid-c) and sens. (dotted) with constraint (solid) for case 10 - $(k_a, k_s) = (2,8)$, ϕ=.4455, compensator order = 16.

front of the compressor. One reason it is bad in the back of the compressor is that the exit condition specifies that the static pressure be constant. This condition causes static pressure 'nodes' in the exit duct, so particular frequencies cannot be observed. The best location for the total pressure sensor is in the back of the compressor. One reason it is bad in the front is that one of the inlet conditions specifies that the total pressure be constant. This condition causes total pressure nodes in the inlet duct. The best location for the axial velocity sensor is in the back of the compressor, although the variations in position are not as severe as the other sensors. One reason for the insensitivity is that the velocity perturbations throughout the compressor tend to be uniform; they are not specified to be zero at any point.

5.3 Conclusions of Configuration and Design Results

The specifications and state and control costs described in Chapter 4 were used to compare control configurations using the control design procedures also described in Chapter 4. The LQG design procedure was able to distinguish between good and bad configurations, in terms of state and control costs, using only the LQG cost, but did not provide information on which configuration would yield a lower minimum flow coefficient. The H_∞ designs generally yielded the same results with respect to state and control costs as the LQG designs, but also found the minimum flow coefficient attainable for each configuration.

The design procedure is:

1) Choose state and control cost functions and the noise affecting the system to which they are applied. The state cost in this study was shown to be insensitive to whether it used static pressure or velocity in the definition. The noise affecting the system, however, can play an important role.

2) Choose frequency domain specifications. The specifications in this study were a bandwidth constraint and a constant sensitivity constraint. If it is desired, the sensitivity weighting can be frequency dependent to force the sensitivity lower at low frequencies.

3) Choose free parameters in the LQG design method. The free parameters in this study were 'tuned' to a degree in an attempt to satisfy the constraints, although it

may be better just to define the LQG cost as the sum of the state and control costs since this design will not be implemented. The LQG design will then yield the minimum sum of the two costs.

4) Choose an unstable flow coefficient as a nominal case. A flow coefficient of 2% below neutral stability was chosen in this study.

For each sensor and actuator type under consideration, repeat steps (5)-(9) for the first harmonic:

5) Perform LQG designs at each sensor/actuator location for this nominal case. These results will yield LQG, state and control costs, which can be used as a guide as to which locations are expected to be better than others.

6) Perform H_∞ designs at each sensor/actuator location for the nominal case using the frequency domain specifications. These will result in state and control costs as well as some information as to which pairs can or can not meet the specifications at the nominal conditions. If the nominal case results in very tight bounds so that only a few pairs can satisfy them, then the LQG costs can verify whether or not the pairs which meet the constraints are among the lowest cost pairs. If they are not, then the nominal flow coefficient must be raised and this step must be repeated, but the LQG designs do not since they are valid for a range of operating points.

7) Choose a sensor and actuator placement based on the H_∞ designs.

8) Perform H_∞ designs at lower flow coefficients to find the minimum attainable.

9) Using the design at the minimum flow coefficient, test the design at higher flow coefficients. If it only has small robustness problems at a few frequencies, then the constraints can be modified and step (8) must be repeated. If it has large robustness problems, such as those caused by an unstable compensator, then a new sensor/actuator pair must be chosen in step (7).

10) Choose the sensor and actuator type based on the minimum flow coefficient attainable in step (9) as well as the state and control cost from step (6), as long as the nominal flow coefficient is constant for each case.

Chapter 6

Summary and Conclusions

6.1 Summary

The linear two-dimensional compressible model, developed by Bonnaure [16], is extended and recast into input-output form. Cancellation of branch cuts in the inter-blade row gap solutions is identified; this realization leads to the addition of realistic boundary conditions necessary to cancel the branch cuts arising from the inlet and exit duct solutions. These new boundary conditions also result in a new eigenvalue problem. Similar to the branch cuts, the singularities in the inter-blade row gap solutions are shown to cancel in the same manner. Actuator and sensor models are developed which allow the solution to be recast into input-output form using the same technique that is used to formulate the eigenvalue problem. Another contribution to the modeling is the identification of higher frequency (higher than the rotor frequency) modes of the compressor.

The distributed model for each spatial Fourier coefficient is approximated by a finite dimensional state-space approximation. This is accomplished using Padé approximations of exponentials (time delays) and Taylor series expansions of transcendental functions. Two approximate models are developed: a model ignoring the dynamics of the inter-blade row gaps, and a model including the gap dynamics. The former has the advantage of having on the order of 40% fewer states than the latter, but is less accurate. Long inlet and exit ducts are not easily modeled using this method due to the very high order approximations necessary.

Frequency domain specifications and cost functions on the state and control variables are developed to compare control configurations (i.e. sensor and actuator placement and type). A bandwidth constraint is used for robustness to high frequency errors, and a sensitivity constraint is used to provide specified gain and phase margins. A state cost is defined to measure how well a compensator can keep the static pressure perturbations small, while a control cost is defined to measure the amount of control activity necessary to achieve the state cost.

The Linear Quadratic-Gaussian (LQG) and H_∞ design procedures are reviewed for use in control design and configuration comparison. In the LQG design method, colored measurement noise is used to penalize bandwidths greater than the specified constraint, and other free parameters are chosen to provide a good trade-off between bandwidth and low sensitivity. This design also results in an 'LQG' cost function which is used in addition to the state and control costs to compare control configurations. The H_∞ design method uses the frequency domain specifications directly in the design procedure to find which configurations can meet the specifications at the lowest flow coefficients. This information is used with the state and control cost functions to find the 'best' control configuration.

6.2 Conclusions

The results of Chapter 5 provide a framework for comparing control configurations using realistic frequency domain specifications and cost functions. The procedure proposed is listed in Section 5.3.

Specific configuration results for the 3 stage compressor are:

1) Using a jet actuator with zero axial velocity can be as good as using a jet actuator having twice the mean axial velocity provided the actuator in each

case is placed in its respective 'best' location. The best jet actuator location with a velocity of twice the mean velocity is after the IGVs. The best location for the jet actuator with zero velocity is in the back of the compressor.

2) The static pressure sensor was able to yield control laws which had comparable state and control costs with respect to other sensors, but was not able to meet the specifications to as low of a flow coefficient (~2% unstable) as the other sensors. The best location of the static pressure sensor for use with the jet actuator is in the front of the compressor.

3) The total (stagnation) pressure sensor was able to yield control laws which had comparable state and control costs with respect to other sensors and was able to meet the specifications to the lowest coefficient (~5.3% unstable). The best location for the sensor, when used with a jet actuator having a velocity of twice the mean velocity, is in the back of the compressor.

4) The axial velocity sensor was able to yield control laws which had comparable state and control costs with respect to other sensors and was able to meet the specifications to the lowest coefficient (~5.3% unstable). The best location for the sensor, when used with a jet actuator having a velocity of twice the mean velocity, is in the back of the compressor.

Fortunately, no unstable compensators resulted in the best sensor and actuator locations, possibly because unstable compensators might have higher control costs. Although not discussed, unstable compensators generally show poor robustness to operating point changes, sometimes destabilizing the system at normally stable operating points.

Also fortunate was the fact that the compensators designed for the best sensor and actuator locations did not exhibit severe sensitivity to operating point changes. Only minor modifications were necessary to provide robustness to operating point changes in the two cases in which the compensators were designed at a 5.3% unstable operating point. For the two cases in which the compensators were designed at a 2% unstable operating point, no modifications were necessary.

6.3 Recommended Future Work

The approximate model developed has difficulties when the length of the inlet or exit duct becomes long with respect to the compressor radius due the need of very high order approximations of the duct dynamics. A higher order approximation using a different approximation technique than the Taylor series might solve these problems. Or, a numerical frequency domain fitting algorithm might be used on the analytical input-output system. The drawback to this is that a new approximation will have to be done for each flow coefficient, requiring the analytical model to be evaluated at each frequency for each input-output combination. Using the numerical algorithm will be much slower than using the approximate model derived here, but it can result in a lower order approximation for the same accuracy. The order of the approximate model developed in this thesis can also be reduced, while retaining its accuracy, but this can be slow and must be performed for each flow coefficient.

In the course of this study, a number of parameters were chosen somewhat arbitrarily due to lack of data or lack of knowledge of implementation issues. Some of the specific issues and parameters are: sensor and actuator dynamics and bandwidths, jet actuator velocity, system noise and gain and phase margins.

Also, before an implementation can be performed, validation of the input-output model must be performed experimentally.

It is unknown at this time whether the 2D compressible model accurately predicts the locations of the higher frequency eigenvalues. If it does not predict them accurately, then robust techniques must be applied in order not to destabilize the system.

The addition of a second sensor and a second actuator has the potential to reduce the total mean-square control power as well as increase operating range. The mean-square cost functions can easily be modified to include multiple sensors and actuators, but to analyze operating range extension, however, a modified stability robustness test must be proposed that can be used to compare single-input/single-output systems with two-input/two-output systems.

References

[1] Ffowcs Williams, F. E., and Huang, X., "Active Stabilization of Compressor Surge," *Journal of Fluid Mechanics,* vol. 204, 1989, pp. 245-262.

[2] Pinsley, J. E.., Guenette, G. R., Epstein, A. H., and Greitzer, E. M., "Active Stabilization of Centrifugal Compressor Surge," *Journal of Turbomachinery*, vol. 113, 1991, pp. 723-732.

[3] Epstein, A. H., Ffowcs-Williams, J. E., and Greitzer, E. M., "Active Suppression of Aerodynamic Instabilities in Turbomachines," AIAA Paper 86-1994. Also *Journal of Propulsion*, 1989, Vol. 5, No. 2, pp. 204-211.

[4] Paduano, J. D., Epstein, A. H., Valavani, L., Longley, J. P., Greitzer, E. M., and Guenette, G. R., "Active Control of Rotating Stall in a Low-Speed Axial Compressor" *Journal of Turbomachinery*, Vol. 115, January 1993, pp. 48-56.

[5] Haynes, J. M., Hendricks, G. J., and Epstein, A. H., "Active Stabilization of Rotating Stall in a Three-Stage Axial Compressor," ASME Paper #93-GT-346, presented at the International Gas Turbine and Aeroengine Congress and Exposition, Cincinnati, Ohio, May 24-27, 1993.

[6] Takata, H. and Nagano, S., "Nonlinear Analysis of Rotating Stall," *Journal of Engineering for Power*, vol. 94, no. 4, 1972.

[7] Pandolfi, M. and Colasurdo, G., "Numerical Investigations on the Generation and Development of Rotating Stalls," ASME paper 78-WA/GT-5, 1978.

[8] Neuhoff, H. G. and Grahl, K. G., "Numerical Simulation of Rotating Stall in Axial Compressor Blade Rows and Stages," ???.

[9] Greitzer, E. M., "Surge and Rotating Stall in Axial Flow Compressors, Part I: Theoretical Compression System Model, and Part II: Experimental Results and Comparison with Theory," *ASME Journal of Engineering for Power,* Vol. 98, no. 2, April, 1976, pp. 190-216.

[10] McCaughan, F. E., "Application of Bifurcation Theory to Axial Flow Compressor Stability," *ASME Journal of Turbomachinery*, Vol. 3, October 1989a, pp. 426-433.

[11] McCaughan, F. E., "An Analytical and Numerical Study of Axial Flow Compressor Instability," *ASME Journal of Turbomahinery*, Vol. 3, October 1989b, pp. 434-441.

[12] Adomaitis, R. A., and Abed, E. H., "Local Nonlinear Control of Stall Inception in Axial Flow Compressors," AIAA Paper #93-2230, Presented at the 29th Joint Propulsion Conference, June 28-30, 1993, Monterey, CA.

[13] Mansoux, C., Gysling, D. L., and Paduano J. D., "Distributed Nonlinear Modeling and Stability Analysis of Axial Compressor Stall and Surge," to appear, proceedings of the 1994 American Control Conference, June 29, 1994, Baltimore.

[14] Badmus, O. O., Chowdhury, S., Eveker, K. M., and Nett., C. N., "Control-Oriented High-Frequency Turbomachinery Modeling: Single-Stage Compression System 1D Model," ASME Paper 93-GT-18, Presented at the International Gas Turbine and Aeroengine Congress and Exposition, May 24-27, Cincinnati, OH.

[15] Escuret, J. F. and Elder, R. L., "Active Control of Surge in Multi-Stage Axial-Flow Compressors," ASME Paper 93-GT-39, Presented at the International Gas Turbine and Aeroengine Congress and Exposition, May 24-27, Cincinnati, OH.

[16] Bonnaure, L. P., 1991, "Modelling High Speed Multistage Compressor Stability," Masters Thesis, Massachusetts Institute of Technology, Cambridge, MA.

[17] Hendricks, G. J., Bonnaure, L. P., Longley, J. P., Greitzer E. M., Epstein A. H., "Analysis of Rotating Stall Onset In High-Speed Axial Flow Compressors," AIAA paper 93-2233, Presented at the 29th Joint Propulsion Conference, June 28-30, 1993, Monterey, CA.

[18] Miller, D. W., Hall, S. R. and von Flotow, A. H., "Optimal Control of Power Flow at Structural Junctions," *Journal of Sound and Vibration* **140**, 1990, pp. 475-497.

[19] MacMartin, D. G. and Hall, S. R., "Control of Uncertain Structures Using an H_∞ Power Flow Approach," *AIAA Journal of Guidance, Control, and Dynamics*, Vol. 14, No. 3, May-June 1991, pp. 521-530.

[20] Hendricks, G. J., and Gysling, D. L., "A Theoretical Study of Sensor-Actuator Schemes for Rotating Stall Control," presented at the AIAA/SAE/ASME/ASEE 28th Joint Propulsion Conference and Exhibit, Nashville, TN, 1992

[21] Truxal, *Control Engineers Handbook*, McGraw Hill, 1958.

[22] Doyle, J. C., Francis, B. A., and Tannenbaum, A. R., 1992, *Feedback Control Theory*, Macmillan Publishing Company, New York.

[23] Bryson, A. E. and Ho, Y.-C., 1975, *Applied Optimal Control*, Hemisphere Publishing Corp., New York.

[24] Doyle, J. C., Glover, K., Khargonekar, P. P., and Francis, B. A., "State-Space Solutions to Standard H_2 and H_∞ Control Problems," *IEEE Transactions on Automatic Control*, vol. 34, no. 8, pp. 831-847.

[25] Maciejowski, J. M., *Multivariable Feedback Design*, Addison-Wesley, 1989.

[26] Greitzer, E. M., personal communications.

Appendix A

Linearized Boundary Condition Matrices

This appendix derives all of the linearized boundary condition matrices for the model of Chapter 2. In Section A.1, the linearized leading edge boundary condition matrices are derived. In Section A.2, the linearized trailing edge boundary condition matrices are derived. In Section A.3, the linearized inlet condition matrices are derived. In Section A.4, the linearized actuation effect matrices are derived.

In this appendix, the numbers "1" and "2" are used extensively to denote upstream of the boundary and downstream of the boundary, respectively.

A.1 Leading Edge Boundary Conditions

The first leading edge boundary condition is continuity or mass flow conservation. The equation for mass flow and its linearization are:

$$\begin{gathered} \dot{m}_1 = \dot{m}_2 \\ \delta\dot{m}_1 = \delta\dot{m}_2 \end{gathered} \tag{A.1}$$

$$\begin{gathered} \dot{m} = \rho A V_x \\ \delta\dot{m} = \delta\rho A V_x + \rho A \delta V_x \\ \frac{\delta\dot{m}_2}{\dot{m}} = \frac{\delta\dot{m}_1}{\dot{m}} = \frac{\delta\rho}{\rho} + \frac{a}{V_x}\frac{\delta V_x}{a} \end{gathered} \tag{A.2}$$

The second leading edge boundary condition is relative total temperature conservation, relative meaning in the rotating reference frame of a rotor, or stationary frame of a stator. The equation and its linearization are:

$$
\begin{aligned}
T_{t1} &= T_{t2} \\
\delta T_{t1} &= \delta T_{t2}
\end{aligned}
\tag{A.3}
$$

$$
\begin{aligned}
T_t &= T\left(1+\tfrac{\gamma-1}{2}M^2\right) \\
\delta T_t &= \delta T\left(1+\tfrac{\gamma-1}{2}M^2\right)+T(\gamma-1)M\delta M \\
\frac{\delta T_t}{T_t} &= \frac{\delta T}{T}+\frac{(\gamma-1)M}{\left(1+\frac{\gamma-1}{2}M^2\right)}\delta M
\end{aligned}
\tag{A.4}
$$

Here we need to derive a few more relations before we can get the linearization in terms of the dependent variables:

$$
\begin{aligned}
P &= \rho RT \\
\delta P &= \delta\rho RT+\rho R\delta T \\
\frac{\delta T}{T} &= \frac{\delta P}{P}-\frac{\delta\rho}{\rho}
\end{aligned}
\tag{A.5}
$$

and

$$
\begin{aligned}
M &= \frac{V}{a}=\sqrt{(V_x^2+V_\theta^2)\tfrac{\rho}{\gamma P}} \\
\delta M &= \frac{M}{2}\frac{\delta\rho}{\rho}-\frac{M}{2}\frac{\delta P}{P}+\frac{V_x}{V}\frac{\delta V_x}{a}+\frac{V_\theta}{V}\frac{\delta V_\theta}{a}
\end{aligned}
\tag{A.6}
$$

so combining Equations A.3, A.4, A.5 and A.6, we get

$$
\delta T_{t2}=\delta T_{t1}=\frac{T_{t1}}{1+\frac{\gamma-1}{2}M^2}\left[\frac{\delta P}{P}-\frac{\delta\rho}{\rho}+(\gamma-1)M_x\frac{\delta V_x}{a}+(\gamma-1)M_\theta\frac{\delta V_\theta}{a}\right]
\tag{A.7}
$$

The third leading edge boundary condition is a relative total pressure loss equation. This equation is approximated in Bonnaure [16] as

$$P_{t2} = P_{t1} - \frac{(P_{t1} - P_1)\omega_{loss}(\alpha_1, M_{R1})}{1 + s\tau}$$

The angle is the relative angle of the flow at the leading edge of the blade, the Mach number is the Mach number relative to the blade reference frame, and the loss coefficient partial derivatives need to be supplied as part of the compressor characteristic. The linearization of this equation is

$$\delta P_{t2} = \delta P_{t1} - \frac{1}{1 + s\tau}\begin{bmatrix}(\delta P_{t1} - \delta P_1)\omega_{loss} + \\ (P_{t1} - P_1)\left(\dfrac{\partial \omega_{loss}}{\partial \tan\alpha_1}\delta\tan\alpha_1 + \dfrac{\partial \omega_{loss}}{\partial M_{R1}}\delta M_{R1}\right)\end{bmatrix} \tag{A.8}$$

The total pressure and its linearization are:

$$\begin{aligned} P_t &= P\left(1 + \tfrac{\gamma-1}{2}M^2\right)^{\gamma/\gamma-1} \\ \frac{\delta P_t}{P_t} &= \frac{\delta P}{P} + \frac{\gamma M}{1 + \frac{\gamma-1}{2}M^2}\delta M \end{aligned} \tag{A.9}$$

using Equation A.6, we get

$$\frac{\delta P_{t1}}{P_{t1}} = \frac{1}{1 + \frac{\gamma-1}{2}M^2}\left[\left(1 - .5M^2\right)\frac{\delta P}{P} + .5\gamma M^2\frac{\delta\rho}{\rho} + \gamma M_x\frac{\delta V_x}{a} + \gamma M_\theta\frac{\delta V_\theta}{a}\right] \tag{A.10}$$

We also need another relation at this point:

$$\tan\alpha = \frac{V_\theta}{V_x}$$

$$\frac{\delta \tan\alpha}{\tan\alpha} = -\frac{1}{M_x}\frac{\delta V_x}{a} + \frac{1}{M_\theta}\frac{\delta V_\theta}{a} \tag{A.11}$$

Using Equations A.8, A.9, A.10 and A.11, we get

$$\begin{aligned}\delta P_{t2} = {} & \frac{P_{t1}}{1+\frac{\gamma-1}{2}M^2}\left[\left(1-.5M^2\right)\frac{\delta P}{P} + .5\gamma M^2\frac{\delta\rho}{\rho} + \gamma M_x\frac{\delta V_x}{a} + \gamma M_\theta\frac{\delta V_\theta}{a}\right] + \\ & \frac{1}{1+s\tau}\left[\begin{array}{l} -\left(\frac{P_{t1}}{1+\frac{\gamma-1}{2}M^2}\left[\begin{array}{l}\left(1-.5M^2\right)\frac{\delta P}{P} + .5\gamma M^2\frac{\delta\rho}{\rho} + \\ \gamma M_x\frac{\delta V_x}{a} + \gamma M_\theta\frac{\delta V_\theta}{a}\end{array}\right] - P\frac{\delta P}{P}\right)\omega_{loss} + \\ -\left(P_{t1}-P_1\right)\left(\begin{array}{l}\frac{\partial\omega_{loss}}{\partial\tan\alpha_1}\tan\alpha_1\left(-\frac{1}{M_x}\frac{\delta V_x}{a} + \frac{1}{M_\theta}\frac{\delta V_\theta}{a}\right) + \\ \frac{\partial\omega_{loss}}{\partial M_{R1}}\left(\frac{M}{2}\frac{\delta\rho}{\rho} - \frac{M}{2}\frac{\delta P}{P} + \frac{V_x}{V}\frac{\delta V_x}{a} + \frac{V_\theta}{V}\frac{\delta V_\theta}{a}\right)\end{array}\right)\end{array}\right]\end{aligned} \tag{A.12}$$

Equations A.2, A.7 and A.12 can be put in matrix form as

$$\begin{aligned}\begin{bmatrix}\delta\dot{m}_2/\dot{m}_2 \\ \delta T_{t2} \\ \delta P_{t2}\end{bmatrix} = {} & \begin{bmatrix}1 & 0 & 0 \\ 0 & \frac{T_t}{1+\frac{\gamma-1}{2}M^2} & 0 \\ 0 & 0 & \frac{P_t}{1+\frac{\gamma-1}{2}M^2}\end{bmatrix}\begin{bmatrix}0 & 1 & M_x^{-1} & 0 \\ 1 & -1 & (\gamma-1)M_x & (\gamma-1)M_\theta \\ 1-.5M^2 & .5\gamma M^2 & \gamma M_x & \gamma M_\theta\end{bmatrix}\begin{bmatrix}\frac{\delta P}{P} \\ \frac{\delta\rho}{\rho} \\ \frac{\delta V_x}{a} \\ \frac{\delta V_\theta}{a}\end{bmatrix}_k \\ & + \frac{1}{1+s\tau}\begin{bmatrix}0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ p_{3,1} & p_{3,2} & p_{3,3} & p_{3,4}\end{bmatrix}\begin{bmatrix}\frac{\delta P}{P} \\ \frac{\delta\rho}{\rho} \\ \frac{\delta V_x}{a} \\ \frac{\delta V_\theta}{a}\end{bmatrix}_k = \left(\mathbf{V}_{Lk} + \frac{1}{1+s\tau}\mathbf{P}_k\right)\begin{bmatrix}\frac{\delta P}{P} \\ \frac{\delta\rho}{\rho} \\ \frac{\delta V_x}{a} \\ \frac{\delta V_\theta}{a}\end{bmatrix}_k\end{aligned} \tag{A.13}$$

where

$$p_{3,1} = -\left[\left(\frac{P_{t1}}{1+\frac{\gamma-1}{2}M^2}\left(1-.5M^2\right) - P\right)\omega_{loss} - \left(P_{t1}-P_1\right)\frac{M}{2}\frac{\partial\omega_{loss}}{\partial M_{R1}}\right]$$

$$p_{3,2}=-\left[\frac{P_{t1}}{1+\frac{\gamma-1}{2}M^2}.5\gamma M^2\omega_{loss}+\left(P_{t1}-P_1\right)\frac{M}{2}\frac{\partial\omega_{loss}}{\partial M_{R1}}\right]$$
$$p_{3,3}=-\left[\frac{P_{t1}}{1+\frac{\gamma-1}{2}M^2}\gamma M_x\omega_{loss}+\left(P_{t1}-P_1\right)\left(-\frac{\partial\omega_{loss}}{\partial\tan\alpha_1}\tan\alpha_1\frac{1}{M_x}+\frac{\partial\omega_{loss}}{\partial M_{R1}}\frac{V_x}{V}\right)\right]$$
$$p_{3,4}=-\left[\frac{P_{t1}}{1+\frac{\gamma-1}{2}M^2}\gamma M_\theta\omega_{loss}+\left(P_{t1}-P_1\right)\left(\frac{\partial\omega_{loss}}{\partial\tan\alpha_1}\tan\alpha_1\frac{1}{M_\theta}+\frac{\partial\omega_{loss}}{\partial M_{R1}}\frac{V_\theta}{V}\right)\right]$$

Now, the corresponding relations must be found for the blade row solutions, i.e. state "2". The mass flow is written as

$$\dot{m}_2=\rho AW$$
$$\delta\dot{m}=\delta\rho AW+\rho A\delta W$$
$$\frac{\delta\dot{m}_2}{\dot{m}}=\frac{\delta\rho}{\rho}+\frac{a}{W}\frac{\delta W}{a} \tag{A.14}$$

The relative total temperature is written, from Equation A.4

$$\frac{\delta T_t}{T_t}=\frac{\delta T}{T}+\frac{(\gamma-1)M}{\left(1+\frac{\gamma-1}{2}M^2\right)}\delta M \tag{A.4}$$

The linearized Mach number equation is

$$M=\frac{W}{a}=W\sqrt{\tfrac{\rho}{\gamma P}}$$
$$\delta M=\frac{\delta W}{a}+.5M\left(\frac{\delta\rho}{\rho}-\frac{\delta P}{P}\right) \tag{A.15}$$

Combining Equations A.4, A.5 and A.15, we get

$$\frac{\delta T_{t2}}{T_{t2}}=\frac{1}{1+\frac{\gamma-1}{2}M^2}\left[\frac{\delta P}{P}-\frac{\delta\rho}{\rho}+(\gamma-1)M\frac{\delta W}{a}\right] \tag{A.16}$$

The last equation to derive is the total pressure in the blade row. Combining Equations A.9 and A.15, we get

$$\frac{\delta P_{t2}}{P_{t2}} = \frac{1}{1+\frac{\gamma-1}{2}M^2}\left[\left(1-.5M^2\right)\frac{\delta P}{p}+.5\gamma M^2\frac{\delta\rho}{\rho}+\gamma M\frac{\delta W}{a}\right] \tag{A.17}$$

Equations A.14, A.16 and A.17 can be written in matrix form as

$$\begin{bmatrix} \delta\dot{m}_2/\dot{m}_2 \\ \delta T_{t2} \\ \delta P_{t2} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{T_{t2}}{1+\frac{\gamma-1}{2}M^2} & 0 \\ 0 & 0 & \frac{P_{t2}}{1+\frac{\gamma-1}{2}M^2} \end{bmatrix}\begin{bmatrix} 0 & 1 & M^{-1} \\ 1 & -1 & (\gamma-1)M \\ 1-.5M^2 & .5\gamma M^2 & \gamma M \end{bmatrix}\begin{bmatrix} \frac{\delta P}{P} \\ \frac{\delta\rho}{\rho} \\ \frac{\delta W}{a} \end{bmatrix}_k$$
$$= \mathbf{B}_{Lk}\begin{bmatrix} \frac{\delta P}{P} \\ \frac{\delta\rho}{\rho} \\ \frac{\delta W}{a} \end{bmatrix}_k \tag{A.18}$$

The leading edge boundary condition is written using A.13, A.18, 2.25 and 2.33:

$$\left(\mathbf{V}_{Lk}+\frac{1}{1+s\tau}\mathbf{P}_k\right)\mathbf{V}_k(x_{LEk},s)\begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_k = \mathbf{B}_{Lk}\mathbf{B}_k(x_{LEk},s)\begin{bmatrix} \tilde{B} \\ \tilde{C} \\ \tilde{E} \end{bmatrix}_k \tag{A.19}$$

A.2 Trailing Edge Boundary Conditions

The first trailing edge boundary condition is continuity. This condition has the same form as Equations A.2 and A.14, with the blade solution being upstream and the gap solution being downstream, instead:

$$\frac{\delta\dot{m}_1}{\dot{m}} = \frac{\delta\dot{m}_2}{\dot{m}} = \frac{\delta\rho}{\rho}+\frac{a}{V_x}\frac{\delta V_x}{a} \tag{A.20}$$

$$\frac{\delta \dot{m}_1}{\dot{m}} = \frac{\delta \rho}{\rho} + \frac{a}{W}\frac{\delta W}{a} \tag{A.21}$$

The second condition in the relative total temperature conservation. This condition also has the same form as Equations A.7 and A.16 with the upstream and downstream reversed again:

$$\delta T_{t1} = \delta T_{t2} = \frac{T_{t2}}{1+\frac{\gamma-1}{2}M^2}\left[\frac{\delta P}{P} - \frac{\delta \rho}{\rho} + (\gamma-1)M_x\frac{\delta V_x}{a} + (\gamma-1)M_\theta\frac{\delta V_\theta}{a}\right] \tag{A.22}$$

$$\frac{\delta T_{t1}}{T_{t1}} = \frac{1}{1+\frac{\gamma-1}{2}M^2}\left[\frac{\delta P}{P} - \frac{\delta \rho}{\rho} + (\gamma-1)M\frac{\delta W}{a}\right] \tag{A.23}$$

The third condition is relative total pressure conservation. This condition has the same form as Equations A.10 and A.17 with the upstream and downstream reversed again:

$$\frac{\delta P_{t2}}{P_{t2}} = \frac{1}{1+\frac{\gamma-1}{2}M^2}\left[\left(1-.5M^2\right)\frac{\delta P}{P} + .5\gamma M^2\frac{\delta \rho}{\rho} + \gamma M_x\frac{\delta V_x}{a} + \gamma M_\theta\frac{\delta V_\theta}{a}\right] \tag{A.24}$$

$$\frac{\delta P_{t2}}{P_{t2}} = \frac{1}{1+\frac{\gamma-1}{2}M^2}\left[\left(1-.5M^2\right)\frac{\delta P}{p} + .5\gamma M^2\frac{\delta \rho}{\rho} + \gamma M\frac{\delta W}{a}\right] \tag{A.25}$$

The fourth condition is a deviation condition. This condition is approximated by Bonnaure [16] as

$$\alpha_2 = \frac{\alpha_{2q.s.}(\alpha_{1,inlet}, M_{R1,inlet})}{1+s\tau}$$

where "inlet" refers to the inlet of the blade row, i.e. the state "1" from the leading edge boundary, and "q.s." means quasi-steady. This condition is linearized as

$$\delta\alpha_2 = \frac{1}{1+s\tau}\left[\frac{\partial\alpha_2}{\partial\tan\alpha_1}\delta\tan\alpha_1 + \frac{\partial\alpha_2}{\partial M_{R1}}\delta M_{R1}\right]$$

and using Equations A.6 and A.11, we get

$$\delta\alpha_2 = \frac{1}{1+s\tau}\left[\begin{array}{l}\dfrac{\partial\alpha_2}{\partial\tan\alpha_1}\tan\alpha_1\left(-\dfrac{1}{M_x}\dfrac{\delta V_x}{a}+\dfrac{1}{M_\theta}\dfrac{\delta V_\theta}{a}\right)+ \\ \dfrac{\partial\alpha_2}{\partial M_{R1}}\left(\dfrac{M}{2}\dfrac{\delta\rho}{\rho}-\dfrac{M}{2}\dfrac{\delta P}{P}+\dfrac{V_x}{V}\dfrac{\delta V_x}{a}+\dfrac{V_\theta}{V}\dfrac{\delta V_\theta}{a}\right)\end{array}\right] \tag{A.26}$$

and the linearized flow angle downstream is

$$\delta\alpha_2 = \frac{\partial\alpha_2}{\partial\tan\alpha_2}\delta\tan\alpha_2$$

$$\delta\alpha_2 = \frac{\partial\alpha_2}{\partial\tan\alpha_2}\tan\alpha_2\left(-\frac{1}{M_x}\frac{\delta V_x}{a}+\frac{1}{M_\theta}\frac{\delta V_\theta}{a}\right) \tag{A.27}$$

Now, Equations A.20, A.22, A.24 and A.27 can be combined in matrix form as

$$\begin{bmatrix}\delta\dot m_2/\dot m_2 \\ \delta T_{t2} \\ \delta P_{t2} \\ \delta\alpha_2\end{bmatrix} = \begin{bmatrix}1 & 0 & 0 & 0 \\ 0 & \frac{T_t}{1+\frac{\gamma-1}{2}M^2} & 0 & 0 \\ 0 & 0 & \frac{P_t}{1+\frac{\gamma-1}{2}M^2} & 0 \\ 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix}0 & 1 & M_x^{-1} & 0 \\ 1 & -1 & (\gamma-1)M_x & (\gamma-1)M_\theta \\ 1-.5M^2 & .5\gamma M^2 & \gamma M_x & \gamma M_\theta \\ 0 & 0 & -M_\theta M_x^{-2} & M_x^{-1}\end{bmatrix}\begin{bmatrix}\frac{\delta P}{P} \\ \frac{\delta\rho}{\rho} \\ \frac{\delta V_x}{a} \\ \frac{\delta V_\theta}{a}\end{bmatrix}_{k+1}$$

$$= \mathbf{V}_{Tk+1}\begin{bmatrix}\frac{\delta P}{P} \\ \frac{\delta\rho}{\rho} \\ \frac{\delta V_x}{a} \\ \frac{\delta V_\theta}{a}\end{bmatrix}_{k+1} \tag{A.28}$$

Also, Equations A.21, A.23, A.25 and A.26 can be combined in matrix form as

$$\begin{bmatrix} \delta \dot{m}_1 / \dot{m}_1 \\ \delta T_{t1} \\ \delta P_{t1} \\ \delta \alpha_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{T_{t2}}{1+\frac{\gamma-1}{2}M^2} & 0 & 0 \\ 0 & 0 & \frac{P_{t2}}{1+\frac{\gamma-1}{2}M^2} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & M^{-1} \\ 1 & -1 & (\gamma-1)M \\ 1-.5M^2 & .5\gamma M^2 & \gamma M \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{\delta P}{P} \\ \frac{\delta \rho}{\rho} \\ \frac{\delta W}{a} \end{bmatrix}_k +$$

$$\frac{1}{1+s\tau} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ d_{4,1} & d_{4,2} & d_{4,3} & d_{4,4} \end{bmatrix} \begin{bmatrix} \frac{\delta P}{P} \\ \frac{\delta P}{P} \\ \frac{\delta P}{P} \\ \frac{\delta P}{P} \end{bmatrix}_k = \mathbf{B}_{Tk} \begin{bmatrix} \frac{\delta P}{P} \\ \frac{\delta \rho}{\rho} \\ \frac{\delta W}{a} \end{bmatrix}_k + \frac{1}{1+s\tau} \mathbf{D}_k \begin{bmatrix} \frac{\delta P}{P} \\ \frac{\delta P}{P} \\ \frac{\delta P}{P} \\ \frac{\delta P}{P} \end{bmatrix}_k \tag{A.29}$$

where

$$d_{4,1} = -\frac{\partial \alpha_2}{\partial M_{R1}} \frac{M}{2}$$

$$d_{4,2} = \frac{\partial \alpha_2}{\partial M_{R1}} \frac{M}{2}$$

$$d_{4,3} = -\frac{\partial \alpha_2}{\partial \tan \alpha_1} \frac{M_\theta}{M_x^2} + \frac{\partial \alpha_2}{\partial M_{R1}} \frac{V_x}{V}$$

$$d_{4,4} = \frac{\partial \alpha_2}{\partial \tan \alpha_1} \frac{1}{M_x} + \frac{\partial \alpha_2}{\partial M_{R1}} \frac{V_\theta}{V}$$

The trailing edge boundary condition is written using Equations A.28, A.29, 2.25 and 2.33:

$$\mathbf{V}_{Tk+1} \mathbf{V}_{k+1}(x_{TEk}, s) \begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_{k+1} = \mathbf{B}_{Tk} \mathbf{B}_k(x_{TEk}, s) \begin{bmatrix} \tilde{B} \\ \tilde{C} \\ \tilde{E} \end{bmatrix}_k + \frac{1}{1+s\tau} \mathbf{D}_k \mathbf{V}_k(x_{LEk}, s) \begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_k \tag{A.30}$$

A.3 Inlet Condition

The three inlet conditions are constant total pressure, zero entropy and zero vorticity [26]. The total pressure equation (A.10) will be used for the first condition. The second condition requires a definition of an isentropic process:

$$P\rho^{-\gamma} = \text{constant}$$

This can be linearized and written as

$$\rho^{-\gamma}\delta P - \gamma P\rho^{-\gamma-1}\delta\rho = 0$$

The result of this, using Equations 2.17 and 2.18 for non-zeroth harmonics (2.21 and 2.22 for the zeroth harmonic), is

$$E_n(s) = 0 \tag{A.31}$$

The last condition also requires a definition of vorticity:

$$\Omega_{vort} = \frac{\partial V_x}{\partial \theta} - \frac{\partial V_\theta}{\partial x}$$

This can be linearized as

$$\delta\Omega_{vort} = \frac{\partial}{\partial \theta}\frac{\delta V_x}{a} - \frac{\partial}{\partial x}\frac{\delta V_\theta}{a} = 0$$

The result of this, using Equations 2.19 and 2.20 for non-zeroth harmonics (2.23 and 2.24 for the zeroth harmonic), is

$$D_n(s) = 0 \tag{A.32}$$

Finally, the inlet condition matrix can be written using Equations A.10, 2.25, A.31 and A.32 as

$$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} N_{1,1} & N_{1,2} & N_{1,3} & N_{1,4} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_1 = \mathbf{N}(s) \begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix}_1 \tag{A.33}$$

where

$$\begin{bmatrix} N_{1,1} & N_{1,2} & N_{1,3} & N_{1,4} \end{bmatrix} = \frac{P_t}{1+\frac{\gamma-1}{2}M^2} \begin{bmatrix} 1-.5M^2 & .5\gamma M^2 & \gamma M_x & \gamma M_\theta \end{bmatrix} \mathbf{V}_1(x_{in}, s)$$

A.4 Jet Actuator Conditions

The four boundary conditions across a jet actuator are continuity, x momentum, θ momentum and energy. The continuity equation can be written, as

$$\frac{\delta \dot{m}_2}{\dot{m}} = \frac{\delta \dot{m}_1}{\dot{m}} + \frac{\rho_j V_j A_j}{\rho V A} \tag{A.34}$$

Here, the control variable will be introduced as the ratio of injected mass flow to mean mass flow:

$$u(s) = \frac{\rho_j V_j A_j(s)}{\rho V A} \tag{A.35}$$

where the physical control will be on the injector area, as indicated by the s dependence. The x momentum equation is written as

$$P_2 A + \rho_2 V_{x2}^2 A = P_1 A + \rho_1 V_{x1}^2 A + \rho_j V_j^2 A_j$$

This can be linearized and written as

$$\frac{1}{\gamma M_x^2}\frac{\delta P}{P}_2 + \frac{\delta\rho}{\rho}_2 + \frac{2}{M_x}\frac{\delta V_x}{a}_2 = \frac{1}{\gamma M_x^2}\frac{\delta P}{P}_1 + \frac{\delta\rho}{\rho}_1 + \frac{2}{M_x}\frac{\delta V_x}{a}_1 + \frac{V_j}{V_x}u(s)$$

where the following relation was used:

$$\frac{P}{\rho V_x^2} = \frac{1}{\gamma M_x^2}$$

This can be simplified a bit by using Equations A.34, A.35 and A.2 to become

$$\frac{1}{\gamma M_x^2}\frac{\delta P}{P}_2 + \frac{1}{M_x}\frac{\delta V_x}{a}_2 = \frac{1}{\gamma M_x^2}\frac{\delta P}{P}_1 + \frac{1}{M_x}\frac{\delta V_x}{a}_1 + \left(\frac{V_j}{V_x} - 1\right)u(s) \tag{A.36}$$

The θ momentum is similar to the x momentum, but the jet velocity will have no component in the θ direction:

$$\rho_2 V_{\theta 2}^2 A = \rho_1 V_{\theta 1}^2 A$$

Also, there is no static pressure difference in the θ direction. The linearization of this is

$$\frac{\delta\rho}{\rho}_2 + \frac{2}{M_\theta}\frac{\delta V_\theta}{a}_2 = \frac{\delta\rho}{\rho}_1 + \frac{2}{M_\theta}\frac{\delta V_\theta}{a}_1 \tag{A.37}$$

The final equation is energy:

$$\rho_2 V_{x2} T_{t2} A = \rho_1 V_{x1} T_{t1} A + \rho_j V_{xj} T_{tj} A_j$$

This can be linearized as

$$\frac{\delta T_t}{T_t}\bigg|_2 + \frac{\delta \rho}{\rho}\bigg|_2 + \frac{1}{M_x}\frac{\delta V_x}{a}\bigg|_2 = \frac{\delta T_t}{T_t}\bigg|_1 + \frac{\delta \rho}{\rho}\bigg|_1 + \frac{1}{M_x}\frac{\delta V_x}{a}\bigg|_1 + \frac{T_{tj}}{T_t}u(s)$$

This can also be simplified a bit by using Equations A.34, A.35 and A.2 to become

$$\frac{\delta T_t}{T_t}\bigg|_2 = \frac{\delta T_t}{T_t}\bigg|_1 + \left(\frac{T_{tj}}{T_t} - 1\right)u(s) \tag{A.38}$$

Now, using A.34 and A.35 with A.2, A.36, A.37 and A.38 with A.7, we get the following relation:

$$\begin{bmatrix} 0 & 1 & M_x^{-1} & 0 \\ \gamma^{-1}M_x^{-2} & 0 & M_x^{-1} & 0 \\ 0 & 1 & 0 & 2M_\theta^{-1} \\ \frac{T_t}{1+\frac{\gamma-1}{2}M^2} & -\frac{T_t}{1+\frac{\gamma-1}{2}M^2} & \frac{(\gamma-1)M_x T_t}{1+\frac{\gamma-1}{2}M^2} & \frac{(\gamma-1)M_\theta T_t}{1+\frac{\gamma-1}{2}M^2} \end{bmatrix} \begin{bmatrix} \frac{\delta P}{P} \\ \frac{\delta \rho}{\rho} \\ \frac{\delta V_x}{a} \\ \frac{\delta V_\theta}{a} \end{bmatrix}_{ka,downstream} =$$

$$\begin{bmatrix} 0 & 1 & M_x^{-1} & 0 \\ \gamma^{-1}M_x^{-2} & 0 & M_x^{-1} & 0 \\ 0 & 1 & 0 & 2M_\theta^{-1} \\ \frac{T_t}{1+\frac{\gamma-1}{2}M^2} & -\frac{T_t}{1+\frac{\gamma-1}{2}M^2} & \frac{(\gamma-1)M_x T_t}{1+\frac{\gamma-1}{2}M^2} & \frac{(\gamma-1)M_\theta T_t}{1+\frac{\gamma-1}{2}M^2} \end{bmatrix} \begin{bmatrix} \frac{\delta P}{P} \\ \frac{\delta \rho}{\rho} \\ \frac{\delta V_x}{a} \\ \frac{\delta V_\theta}{a} \end{bmatrix}_{ka,upstream} + \begin{bmatrix} 1 \\ \frac{V_j}{V_x} - 1 \\ 0 \\ \frac{T_{tj}}{T_t} - 1 \end{bmatrix}_{ka} u(s)$$

(A.39)

or, in matrix notation:

$$\mathbf{J}_{ku}\begin{bmatrix}\frac{\delta P}{P}\\ \frac{\delta\rho}{\rho}\\ \frac{\delta V_x}{a}\\ \frac{\delta V_\theta}{a}\end{bmatrix}_{ku,downstream}=\mathbf{J}_{ku}\begin{bmatrix}\frac{\delta P}{P}\\ \frac{\delta\rho}{\rho}\\ \frac{\delta V_x}{a}\\ \frac{\delta V_\theta}{a}\end{bmatrix}_{ku,upstream}+\mathbf{b}_{ku}u(s) \qquad \text{(A.40)}$$

Appendix B

Elements of $\mathbf{V}(x_{LE},s)\mathbf{V}^{-1}(x_{TE},s)$

This Appendix expands the elements of the matrix product $\mathbf{V}(x_{LE},s)\mathbf{V}^{-1}(x_{TE},s)$ to be approximated by Taylor series'. These terms are necessary to approximate the end conditions and the inter-blade row gaps. The matrix $\mathbf{V}$ is defined in Equation 2.25. Section B.1 shows the elements for non-zeroth harmonics. Section B.2 shows the elements for the zeroth harmonic.

B.1 Non-Zeroth Harmonics

This appendix will show each term of the matrix product for non-zeroth harmonics starting with the following definitions:

$$h(s) = e^{\frac{M_x\left(\frac{sr}{a}+jnM_\theta\right)}{1-M_x^2}\frac{\Delta x}{r}}$$

$$b = \frac{1}{1-M_x^2}\frac{\Delta x}{r}$$

$$y(s) = n^2(1-M_x^2) + \left(\frac{sr}{a} + jnM_\theta\right)^2$$

$$\chi(s) = -\frac{\frac{sr}{a} + jnM_\theta}{rM_x}$$

The elements are:

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(1,1)} = h(s)\cosh(b\sqrt{y(s)})$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(2,1)} = \frac{h(s)}{\gamma}\cosh(b\sqrt{y(s)}) - \frac{e^{\chi(s)\Delta x}}{\gamma}$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(3,1)} = \frac{-h(s)}{\gamma\left(y(s)-n^2\right)}\left(-n^2 M_x \cosh(b\sqrt{y(s)}) + y(s)(\frac{sr}{a} + jnM_\theta)\frac{\sinh(b\sqrt{y(s)})}{\sqrt{y(s)}}\right) - \frac{n^2 M_x}{\gamma\left(y(s)-n^2\right)} e^{\chi(s)\Delta x}$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(4,1)} = \frac{-jnh(s)}{\gamma\left(y(s)-n^2\right)}\left((\frac{sr}{a} + jnM_\theta)\cosh(b\sqrt{y(s)}) - M_x y(s)\frac{\sinh(b\sqrt{y(s)})}{\sqrt{y(s)}}\right) + \frac{jn(\frac{sr}{a} + jnM_\theta)}{\gamma\left(y(s)-n^2\right)} e^{\chi(s)\Delta x}$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(1,2)} = 0$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(2,2)} = e^{\chi(s)\Delta x}$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(3,2)} = 0$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(4,2)} = 0$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(1,3)} = -\gamma(\frac{sr}{a} + jnM_\theta)h(s)\frac{\sinh(b\sqrt{y(s)})}{\sqrt{y(s)}}$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(2,3)} = -(\frac{sr}{a} + jnM_\theta)h(s)\frac{\sinh(b\sqrt{y(s)})}{\sqrt{y(s)}}$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(3,3)} = \frac{(\frac{sr}{a} + jnM_\theta)h(s)}{\left(y(s)-n^2\right)}\left((\frac{sr}{a} + jnM_\theta)\cosh(b\sqrt{y(s)}) - n^2 M_x \frac{\sinh(b\sqrt{y(s)})}{\sqrt{y(s)}} \right) - \frac{n^2 M_x^2}{\left(y(s)-n^2\right)} e^{\chi(s)\Delta x}$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(4,3)} = \frac{jn(\frac{sr}{a} + jnM_\theta)h(s)}{\left(y(s)-n^2\right)}\left(-M_x\cosh(b\sqrt{y(s)}) + (\frac{sr}{a} + jnM_\theta)\frac{\sinh(b\sqrt{y(s)})}{\sqrt{y(s)}} \right) + \frac{jnM_x(\frac{sr}{a} + jnM_\theta)}{\left(y(s)-n^2\right)} e^{\chi(s)\Delta x}$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(1,4)} = \gamma jnM_x h(s)\frac{\sinh(b\sqrt{y(s)})}{\sqrt{y(s)}}$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(2,4)} = jnM_x h(s)\frac{\sinh(b\sqrt{y(s)})}{\sqrt{y(s)}}$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(3,4)} = \frac{-jnM_x h(s)}{\left(y(s)-n^2\right)}\left((\frac{sr}{a} + jnM_\theta)\cosh(b\sqrt{y(s)}) - n^2 M_x \frac{\sinh(b\sqrt{y(s)})}{\sqrt{y(s)}} \right) + \frac{jnM_x(\frac{sr}{a} + jnM_\theta)}{\left(y(s)-n^2\right)} e^{\chi(s)\Delta x}$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(4,4)} = \frac{n^2 M_x h(s)}{\left(y(s)-n^2\right)}\left(-M_x\cosh(b\sqrt{y(s)}) + (\frac{sr}{a} + jnM_\theta)\frac{\sinh(b\sqrt{y(s)})}{\sqrt{y(s)}} \right) + \frac{(\frac{sr}{a} + jnM_\theta)^2}{\left(y(s)-n^2\right)} e^{\chi(s)\Delta x}$$

B.2 Zeroth Harmonic

This appendix will show each term of the matrix product for the zeroth harmonic. The elements are:

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(1,1)} = \frac{1}{2}\left(e^{\frac{\Delta x}{a-V_t}s} + e^{-\frac{\Delta x}{a+V_t}s} \right)$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(2,1)} = \frac{1}{2\gamma}\left(e^{\frac{\Delta x}{a-V_t}s} + e^{-\frac{\Delta x}{a+V_t}s} \right) - \frac{1}{\gamma} e^{-\frac{\Delta x}{V_t}s}$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(3,1)} = \frac{1}{2\gamma}\left(-e^{\frac{\Delta x}{a-V_t}s} + e^{-\frac{\Delta x}{a+V_t}s} \right)$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(4,1)} = 0$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(1,2)} = 0$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(2,2)} = e^{-\frac{\Delta x}{V_t}s}$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(3,2)} = 0$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(4,2)} = 0$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(1,3)} = \frac{\gamma}{2}\left(-e^{\frac{\Delta x}{a-V_t}s} + e^{-\frac{\Delta x}{a+V_t}s} \right)$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(2,3)} = \frac{1}{2}\left(-e^{\frac{\Delta x}{a-V_t}s} + e^{-\frac{\Delta x}{a+V_t}s} \right)$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(3,3)} = \frac{1}{2}\left(e^{\frac{\Delta x}{a-V_s}s} + e^{-\frac{\Delta x}{a+V_s}s} \right)$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(4,3)} = 0$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(1,4)} = 0$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(2,4)} = 0$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(3,4)} = 0$$

$$\mathbf{V}(x_L)\mathbf{V}(x_T)^{-1}{}_{(4,4)} = e^{-\frac{\Delta x}{V_s}s}$$

Appendix C

State-Space Matrices

This appendix presents examples of the state-space matrices of Chapter 3. Section C.1 presents the state-space matrices for the model without gaps. Section C.2 presents the state-space matrices for the model including gaps for the non-zeroth harmonics. Section C.3 presents the state-space matrices for the model including gaps for the zeroth harmonic.

C.1 Model without Gaps

For this example, a first order Padé approximation will be used for the delays. First, Equation 3.12, the deviation lag, will be written as

$$\delta\alpha_k = \frac{\tau_d}{1+s\tau_d}\begin{bmatrix} D_{k1} & D_{k2} & D_{k3} & D_{k4} \end{bmatrix}\begin{bmatrix} \tilde{B}_{TE} \\ \tilde{C}_{TE} \\ \tilde{E}_{TE} \\ \delta\alpha \end{bmatrix}_{k-1} + \tau_d D_{ku} u_k \tag{C.1}$$

Similarly, Equation 3.14, the pressure loss lag, will be written as

$$P_{loss,k} = \frac{\tau_p}{1+s\tau_p}\begin{bmatrix} P_{k1} & P_{k2} & P_{k3} & P_{k4} \end{bmatrix}\begin{bmatrix} \tilde{B}_{TE} \\ \tilde{C}_{TE} \\ \tilde{E}_{TE} \\ \delta\alpha \end{bmatrix}_{k-1} + \tau_p P_{ku} u_k \tag{C.2}$$

where P_{ki} and D_{ki} are scalars. The other equations used in this sections are the time delay approximations:

$$\frac{\tilde{B}_{LE}(s)}{\tilde{B}_{TE}(s)} = e^{j\Delta\theta_B - s\Delta T_B} \cong e^{j\Delta\theta} \frac{1 - \frac{1}{2}\Delta T_B s + \frac{1}{12}\Delta T_B^2 s^2}{1 + \frac{1}{2}\Delta T_B s + \frac{1}{12}\Delta T_B^2 s^2}, \tag{3.6}$$

the boundary condition constant transformation:

$$\begin{bmatrix} \tilde{B}_{TEk-1} \\ \tilde{C}_{LEk} \\ \tilde{E}_{LEk} \end{bmatrix} = \mathbf{K}_k \begin{bmatrix} \tilde{B}_{LEk} \\ \tilde{C}_{TEk-1} \\ \tilde{E}_{TEk-1} \\ P_{loss,k} \\ \delta\alpha_{k-1} \\ u_k \end{bmatrix}, \tag{3.16}$$

the inlet condition:

$$\begin{bmatrix} \tilde{C}_{LE1} \\ \tilde{E}_{LE1} \\ P_{loss,1} \\ \delta\alpha_1 \end{bmatrix} = \frac{1}{u_{11}(s)} \begin{bmatrix} u_{21}(s) \\ u_{31}(s) \\ \frac{1}{1+s\tau} u_p(s) \\ \frac{1}{1+s\tau} u_d(s) \end{bmatrix} (\tilde{B}_{LE1} - w_{11} P_{loss,1} - s_{11} u_1) + \begin{bmatrix} w_{21} \\ w_{31} \\ 0 \\ 0 \end{bmatrix} P_{loss,1} + \begin{bmatrix} s_{21} \\ s_{31} \\ \frac{1}{1+s\tau} \mathbf{P}_{1,3r} \mathbf{J}_1^{-1} \mathbf{b}_1 \\ \frac{1}{1+s\tau} \mathbf{D}_{1,4r} \mathbf{J}_1^{-1} \mathbf{b}_1 \end{bmatrix} u_1, \tag{3.28}$$

where u_{ij} are found using Equations 3.21, 3.26 and 3.27, and the exit condition:

$$\tilde{B}_{TEK} = \frac{1}{u_{11}(s)} \begin{bmatrix} u_{12}(s) & u_{13}(s) & u_{14}(s) & u_{15}(s) \end{bmatrix} \begin{bmatrix} \tilde{C}_{TEK} \\ \tilde{E}_{TEK} \\ \delta\alpha_K \\ u_{K+1} \end{bmatrix}. \tag{3.42}$$

where u_{ij} are found using Equation 3.41.

The state matrix is found by finding the differential equations associated with the Padé approximation and the time lags, using the time delays from Equation 3.4 and the first order form of the Padé approximation from Equation 3.6:

$$\mathbf{A}_k = \begin{bmatrix} \frac{-2}{\Delta T_B} & 0 & 0 & 0 & 0 \\ 0 & \frac{-2}{\Delta T_C} & 0 & 0 & 0 \\ 0 & 0 & \frac{-2}{\Delta T_E} & 0 & 0 \\ 0 & 0 & 0 & -\left(\frac{1}{\tau_p} + jn\Omega\right) & 0 \\ 0 & 0 & 0 & 0 & -\left(\frac{1}{\tau_d} + jn\Omega\right) \end{bmatrix} \tag{C.3}$$

and the rotor frequency, Ω, enters in the lag states because if the blade row is a rotor, it is rotating with respect to the stationary reference frame (s is replaced by $s+jn\Omega$). This has also been done in the delay equations, but the rotor frequency shows up in the complex constant terms (Equation C.9). From Equations 3.16, C.1 and C.2, we get:

$$\mathbf{B}_k = \begin{bmatrix} 0 & K_{k+1}(1,2) & K_{k+1}(1,3) & 0 & K_{k+1}(1,5) \\ K_k(2,1) & 0 & 0 & K_k(2,4) & 0 \\ K_k(3,1) & 0 & 0 & K_k(3,4) & 0 \\ P_{k1}K_k(1,1) & 0 & 0 & P_{k1}K_k(1,4) & 0 \\ D_{k1}K_k(1,1) & 0 & 0 & D_{k1}K_k(1,4) & 0 \end{bmatrix} \tag{C.4}$$

$$\mathbf{B}_{k-} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & K_k(2,2) & K_k(2,3) & 0 & K_k(2,5) \\ 0 & K_k(3,2) & K_k(3,3) & 0 & K_k(3,5) \\ 0 & P_{k2} + P_{k1}K_k(1,2) & P_{k3} + P_{k1}K_k(1,3) & 0 & P_{k4} + P_{k1}K_k(1,5) \\ 0 & D_{k2} + D_{k1}K_k(1,2) & D_{k3} + D_{k1}K_k(1,3) & 0 & D_{k4} + D_{k1}K_k(1,5) \end{bmatrix} \tag{C.5}$$

$$\mathbf{B}_{k+} = \begin{bmatrix} K_{k+1}(1,1) & 0 & 0 & K_{k+1}(1,4) & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{C.6}$$

$$\mathbf{B}_{ku} = \begin{bmatrix} 0 \\ K_k(2,6) \\ K_k(3,6) \\ P_{ku} + P_{k1}K_k(1,6) \\ D_{ku} + D_{k1}K_k(1,6) \end{bmatrix} \tag{C.7}$$

$$\mathbf{B}_{ku+} = \begin{bmatrix} K_{k+1}(1,6) \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{C.8}$$

The output matrices are

$$\mathbf{C}_k = \begin{bmatrix} \frac{4}{\Delta T_B} e^{j\Delta\theta_B} & 0 & 0 & 0 & 0 \\ 0 & \frac{4}{\Delta T_C} e^{j\Delta\theta_C} & 0 & 0 & 0 \\ 0 & 0 & \frac{4}{\Delta T_E} e^{j\Delta\theta_E} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{C.9}$$

and if a matrix is defined as

$$\mathbf{E}_k = \begin{bmatrix} e^{j\Delta\theta_B} & 0 & 0 & 0 & 0 \\ 0 & e^{j\Delta\theta_C} & 0 & 0 & 0 \\ 0 & 0 & e^{j\Delta\theta_E} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{C.10}$$

then the feed-through matrices are

$$
\begin{aligned}
\mathbf{D}_k &= -\mathbf{E}_k\mathbf{B}_k \\
\mathbf{D}_{k+} &= -\mathbf{E}_k\mathbf{B}_{k+} \\
\mathbf{D}_{k-} &= -\mathbf{E}_k\mathbf{B}_{k-} \\
\mathbf{D}_{ku} &= -\mathbf{E}_k\mathbf{B}_{ku} \\
\mathbf{D}_{ku+} &= -\mathbf{E}_k\mathbf{B}_{ku+}
\end{aligned}
\tag{C.11}
$$

Defining these matrices in Equations 3.47 allows the formation of the state-space matrices, Equations 3.49, 3.51 and 3.52.

C.2 Model with Gaps for Non-Zeroth Harmonics

For this example, a first order Padé approximation will be used for the delays, where the delays come from Equation 3.4 and the Padé approximation comes from Equation 3.6. A summary of equations used in this section are: the time delay approximations

$$
\frac{\tilde{B}_{LE}(s)}{\tilde{B}_{TE}(s)} = e^{j\Delta\theta_{||}-s\Delta T_{||}} \cong e^{j\Delta\theta}\frac{1-\frac{1}{2}\Delta T_B s+\frac{1}{12}\Delta T_B^2 s^2}{1+\frac{1}{2}\Delta T_B s+\frac{1}{12}\Delta T_B^2 s^2}, \tag{3.6}
$$

the deviation and pressure loss lags:

$$
\begin{aligned}
\delta\alpha_k = -\tfrac{1}{1+s\tau}\mathbf{D}_{k,4r}\mathbf{V}_k(x_{LEk},s)\mathbf{V}_k^{-1}(x_{TEk-1},s)\bullet \\
\left(\mathbf{Y}_{k-1}\begin{bmatrix}\tilde{B}_{TE}\\ \tilde{C}_{TE}\\ \tilde{E}_{TE}\\ \delta\alpha\end{bmatrix}_{k-1} + \mathbf{J}_k^{-1}\mathbf{b}_k u_k\right),
\end{aligned}
\tag{3.58}
$$

and

$$P_{loss,k} = \tfrac{1}{1+s\tau}\mathbf{P}_{k,3r}\mathbf{V}_k(x_{LEk},s)\mathbf{V}_k^{-1}(x_{TEk-1},s)\bullet \left(\mathbf{Y}_{k-1}\begin{bmatrix}\tilde{B}_{TE}\\ \tilde{C}_{TE}\\ \tilde{E}_{TE}\\ \delta\alpha\end{bmatrix}_{k-1} + \mathbf{J}_k^{-1}\mathbf{b}_k u_k \right), \tag{3.59}$$

the boundary condition transformation:

$$\begin{bmatrix}\tilde{B}_{TEk-1}\\ \tilde{C}_{LEk}\\ \tilde{E}_{LEk}\\ (s+1/\tau)P_{loss,k}\\ (s+1/\tau)\delta\alpha_k\end{bmatrix} = \mathbf{H}_{U,k}(s)\begin{bmatrix}\tilde{B}_{LEk}\\ \tilde{C}_{TEk-1}\\ \tilde{E}_{TEk-1}\\ P_{loss,k}\\ \delta\alpha_{k-1}\\ u_k\end{bmatrix}, \tag{3.64}$$

the inlet condition:

$$\begin{bmatrix}\tilde{C}_{LE1}\\ \tilde{E}_{LE1}\\ P_{loss,1}\\ \delta\alpha_1\end{bmatrix} = \frac{1}{u_{11}(s)}\begin{bmatrix}u_{21}(s)\\ u_{31}(s)\\ \frac{1}{1+s\tau}u_p(s)\\ \frac{1}{1+s\tau}u_d(s)\end{bmatrix}(\tilde{B}_{LE1} - w_{11}P_{loss,1} - s_{11}u_1) + \begin{bmatrix}w_{21}\\ w_{31}\\ 0\\ 0\end{bmatrix}P_{loss,1} + \begin{bmatrix}s_{21}\\ s_{31}\\ \frac{1}{1+s\tau}\mathbf{P}_{1,3r}\mathbf{J}_1^{-1}\mathbf{b}_1\\ \frac{1}{1+s\tau}\mathbf{D}_{1,4r}\mathbf{J}_1^{-1}\mathbf{b}_1\end{bmatrix}u_1, \tag{3.28}$$

and the exit condition:

$$\tilde{B}_{TEK} = \frac{1}{u_{11}(s)}\begin{bmatrix}u_{12}(s) & u_{13}(s) & u_{14}(s) & u_{15}(s)\end{bmatrix}\begin{bmatrix}\tilde{C}_{TEK}\\ \tilde{E}_{TEK}\\ \delta\alpha_K\\ u_{K+1}\end{bmatrix}. \tag{3.42}$$

The gap approximations will use a first order denominator and a constant numerator:

$$H_k(s)(l,m) = \frac{h_k(l,m)}{s+a_k}.$$

The state matrix is then

$$\mathbf{A}_k = \mathrm{diag}\begin{bmatrix} \frac{-2}{\Delta T_B} \\ \frac{-2}{\Delta T_C} \\ \frac{-2}{\Delta T_E} \\ -\left(\frac{1}{\tau_p} + jn\Omega\right) \\ -\left(\frac{1}{\tau_d} + jn\Omega\right) \\ -a_{k+1} \\ -a_k \\ -a_k \\ -a_k \\ -a_k \end{bmatrix}$$

where a_k is the pole of the approximation of the matrix $\mathbf{H}_{U,k}(s)$.

$$\mathbf{B}_k = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & h_{k+1}(1,2) & h_{k+1}(1,3) & 0 & h_{k+1}(1,5) & 0 & 0 & 0 & 0 & 0 \\ h_k(2,1) & 0 & 0 & h_k(2,4) & 0 & 0 & 0 & 0 & 0 & 0 \\ h_k(3,1) & 0 & 0 & h_k(3,4) & 0 & 0 & 0 & 0 & 0 & 0 \\ h_k(4,1) & 0 & 0 & h_k(4,4) & 0 & 0 & 0 & 0 & 0 & 0 \\ h_k(5,1) & 0 & 0 & h_k(5,4) & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{B}_{k-} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & h_k(2,2) & h_k(2,3) & 0 & h_k(2,5) & 0 & 0 & 0 & 0 & 0 \\ 0 & h_k(3,2) & h_k(3,3) & 0 & h_k(3,5) & 0 & 0 & 0 & 0 & 0 \\ 0 & h_k(4,2) & h_k(4,3) & 0 & h_k(4,5) & 0 & 0 & 0 & 0 & 0 \\ 0 & h_k(5,2) & h_k(5,3) & 0 & h_k(5,5) & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{B}_{k+} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ h_{k+1}(1,1) & 0 & 0 & h_{k+1}(1,4) & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{B}_{ku} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ h_k(2,6) \\ h_k(3,6) \\ h_k(4,6) \\ h_k(5,6) \end{bmatrix} \qquad \mathbf{B}_{ku+} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ h_{k+1}(1,6) \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

The output matrices are:

$$
\mathbf{C}_k = \begin{bmatrix}
\frac{4}{\Delta T_B} e^{j\Delta\theta_B} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{4}{\Delta T_C} e^{j\Delta\theta_C} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{4}{\Delta T_E} e^{j\Delta\theta_E} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
$$

and if a matrix is defined as:

$$
\mathbf{E}_k = \begin{bmatrix}
e^{j\Delta\theta_B} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & e^{j\Delta\theta_C} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & e^{j\Delta\theta_E} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

then the feed-through matrices are:

$$
\begin{aligned}
\mathbf{D}_k &= -\mathbf{E}_k \mathbf{B}_k \\
\mathbf{D}_{k+} &= -\mathbf{E}_k \mathbf{B}_{k+} \\
\mathbf{D}_{k-} &= -\mathbf{E}_k \mathbf{B}_{k-} \quad . \\
\mathbf{D}_{ku} &= -\mathbf{E}_k \mathbf{B}_{ku} \\
\mathbf{D}_{ku+} &= -\mathbf{E}_k \mathbf{B}_{ku+}
\end{aligned}
$$

Defining these matrices in Equations 3.47 allows the formation of the state-space matrices, Equations 3.49, 3.51 and 3.52.

C.3 Model with Gaps for the Zeroth Harmonic

For this example, a first order Padé approximation will be used for the delays. First, Equation 3.12, the deviation lag, will be written as

$$\delta\alpha_k = \frac{\tau_d}{1+s\tau_d}\begin{bmatrix} D_{k1} & D_{k2} & D_{k3} & D_{k4} \end{bmatrix}\begin{bmatrix} B_{LE} \\ C_{LE} \\ D_{LE} \\ D_{LE} \end{bmatrix}_k$$

Similarly, Equation 3.14, the pressure loss lag, will be written as

$$P_{loss,k} = \frac{\tau_p}{1+s\tau_p}\begin{bmatrix} P_{k1} & P_{k2} & P_{k3} & P_{k4} \end{bmatrix}\begin{bmatrix} B_{LE} \\ C_{LE} \\ D_{LE} \\ E_{LE} \end{bmatrix}_k$$

The other equations used in this sections are the time delay approximations:

$$\frac{\tilde{B}_{LE}(s)}{\tilde{B}_{TE}(s)} = e^{j\Delta\theta_B - s\Delta T_B} \cong e^{j\Delta\theta}\frac{1 - \frac{1}{2}\Delta T_B s + \frac{1}{12}\Delta T_B^2 s^2}{1 + \frac{1}{2}\Delta T_B s + \frac{1}{12}\Delta T_B^2 s^2}, \tag{3.6}$$

the gap delays:

$$\begin{aligned} B_{TE}(s) &= e^{-s\Delta T_{BK}} B_{LE}(s) \\ C_{LE}(s) &= e^{-s\Delta T_{CK}} C_{TE}(s) \\ D_{LE}(s) &= e^{-s\Delta T_{DK}} D_{TE}(s), \\ E_{LE}(s) &= e^{-s\Delta T_{EK}} E_{TE}(s) \end{aligned} \tag{3.73}$$

the leading edge boundary condition constant transformation:

$$\begin{bmatrix} B_{LE} \\ \tilde{C}_{LE} \\ \tilde{E}_{LE} \\ (s+\frac{1}{\tau})P_{loss} \\ (s+\frac{1}{\tau})\delta\alpha \end{bmatrix}_k = \mathbf{K}_{LEk} \begin{bmatrix} \tilde{B}_{LE} \\ C_{LE} \\ D_{LE} \\ E_{LE} \\ P_{loss} \end{bmatrix}_k , \tag{3.80}$$

the trailing edge boundary condition constant transformation:

$$\begin{bmatrix} \tilde{B}_{TEk} \\ C_{TEk+1} \\ D_{TEk+1} \\ E_{TEk+1} \end{bmatrix} = \mathbf{K}_{TEk} \begin{bmatrix} B_{TEk+1} \\ \tilde{C}_{TEk} \\ \tilde{E}_{TEk} \\ \delta\alpha_k \\ u_{k+1} \end{bmatrix}, \tag{3.83}$$

the inlet condition:

$$\begin{bmatrix} \tilde{C}_{LE1} \\ \tilde{E}_{LE1} \\ P_{loss,1} \\ \delta\alpha_1 \end{bmatrix} = \frac{1}{u_{11}(s)} \begin{bmatrix} u_{21}(s) \\ u_{31}(s) \\ \frac{1}{1+s\tau}u_p(s) \\ \frac{1}{1+s\tau}u_d(s) \end{bmatrix} (\tilde{B}_{LE1} - w_{11}P_{loss,1} - s_{11}u_1) + \begin{bmatrix} w_{21} \\ w_{31} \\ 0 \\ 0 \end{bmatrix} P_{loss,1} + \begin{bmatrix} s_{21} \\ s_{31} \\ \frac{1}{1+s\tau}\mathbf{P}_{1,3r}\mathbf{J}_1^{-1}\mathbf{b}_1 \\ \frac{1}{1+s\tau}\mathbf{D}_{1,4r}\mathbf{J}_1^{-1}\mathbf{b}_1 \end{bmatrix} u_1 , \tag{3.28}$$

and the exit condition:

$$\tilde{B}_{TEK} = \frac{1}{u_{11}(s)} \begin{bmatrix} u_{12}(s) & u_{13}(s) & u_{14}(s) & u_{15}(s) \end{bmatrix} \begin{bmatrix} \tilde{C}_{TEK} \\ \tilde{E}_{TEK} \\ \delta\alpha_K \\ u_{K+1} \end{bmatrix}. \tag{3.42}$$

The state matrix is found by finding the differential equations associated with the Padé approximation and the time lags, using the time constants from Equations 3.4 and 3.74:

$$\mathbf{A}_k = diag\begin{bmatrix} \frac{-2}{\Delta T_B} \\ \frac{-2}{\Delta T_C} \\ \frac{-2}{\Delta T_E} \\ \frac{-1}{\tau_p} \\ \frac{-1}{\tau_d} \\ \frac{-2}{\Delta T_{Bk}} \\ \frac{-2}{\Delta T_{Ck}} \\ \frac{-2}{\Delta T_{Dk}} \\ \frac{-2}{\Delta T_{Ek}} \end{bmatrix}$$

$$\mathbf{B}_k(:,1:5) = \begin{bmatrix} 0 & K_{TEk}(1,2) & K_{TEk}(1,3) & 0 & K_{TEk}(1,4) \\ K_{LEk}(2,1) & 0 & 0 & K_{LEk}(2,5) & 0 \\ K_{LEk}(3,1) & 0 & 0 & K_{LEk}(3,5) & 0 \\ P_{k1}K_{LEk}(1,1) & 0 & 0 & P_{k1}K_{LEk}(1,5) & 0 \\ D_{k1}K_{LEk}(1,1) & 0 & 0 & D_{k1}K_{LEk}(1,5) & 0 \\ K_{LEk}(1,1) & 0 & 0 & K_{LEk}(1,5) & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{B}_k(:,6:9) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & K_{LEk}(2,2) & K_{LEk}(2,3) & K_{LEk}(2,4) \\ 0 & K_{LEk}(3,2) & K_{LEk}(3,3) & K_{LEk}(3,4) \\ 0 & P_{2k}+P_{1k}K_{LEk}(1,2) & P_{3k}+P_{1k}K_{LEk}(1,3) & P_{4k}+P_{1k}K_{LEk}(1,4) \\ 0 & D_{2k}+D_{1k}K_{LEk}(1,2) & D_{3k}+D_{1k}K_{LEk}(1,3) & D_{4k}+D_{1k}K_{LEk}(1,4) \\ 0 & K_{LEk}(1,2) & K_{LEk}(1,3) & K_{LEk}(1,4) \\ K_{TEk-1}(2,1) & 0 & 0 & 0 \\ K_{TEk-1}(3,1) & 0 & 0 & 0 \\ K_{TEk-1}(4,1) & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{B}_{k+} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & K_{TEk}(1,1) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{B}_{k-} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & K_{TEk-1}(2,2) & K_{TEk-1}(2,3) & 0 & K_{TEk-1}(2,4) & 0 & 0 & 0 & 0 \\ 0 & K_{TEk-1}(3,2) & K_{TEk-1}(3,3) & 0 & K_{TEk-1}(3,4) & 0 & 0 & 0 & 0 \\ 0 & K_{TEk-1}(4,2) & K_{TEk-1}(4,3) & 0 & K_{TEk-1}(4,4) & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{B}_{ku} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ K_{TEk-1}(2,5) \\ K_{TEk-1}(3,5) \\ K_{TEk-1}(4,5) \end{bmatrix} \qquad \mathbf{B}_{ku+} = \begin{bmatrix} K_{TEk}(1,5) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

The output matrices are

$$\mathbf{C}_k = diag\begin{bmatrix} \frac{4}{\Delta T_H} \\ \frac{4}{\Delta T_C} \\ \frac{4}{\Delta T_L} \\ 1 \\ 1 \\ \frac{4}{\Delta T_{HR}} \\ \frac{4}{\Delta T_{CR}} \\ \frac{4}{\Delta T_{DR}} \\ \frac{4}{\Delta T_{LR}} \end{bmatrix} \qquad \mathbf{E}_k = diag\begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

then the feed-through matrices are

$$\begin{aligned}
\mathbf{D}_k &= -\mathbf{E}_k\mathbf{B}_k \\
\mathbf{D}_{k+} &= -\mathbf{E}_k\mathbf{B}_{k+} \\
\mathbf{D}_{k-} &= -\mathbf{E}_k\mathbf{B}_{k-} \\
\mathbf{D}_{ku} &= -\mathbf{E}_k\mathbf{B}_{ku} \\
\mathbf{D}_{ku+} &= -\mathbf{E}_k\mathbf{B}_{ku+}
\end{aligned}$$

Defining these matrices in Equations 3.47 allows the formation of the state-space matrices, Equations 3.49, 3.51 and 3.52.

Appendix D

LQG Free Parameter Selection

This appendix presents the supporting figures for the LQG free parameter selection in Chapter 4. For each sensor/actuator pair, six combinations of parameters are shown: c=[0.1,1], q=[0.01,1,100].

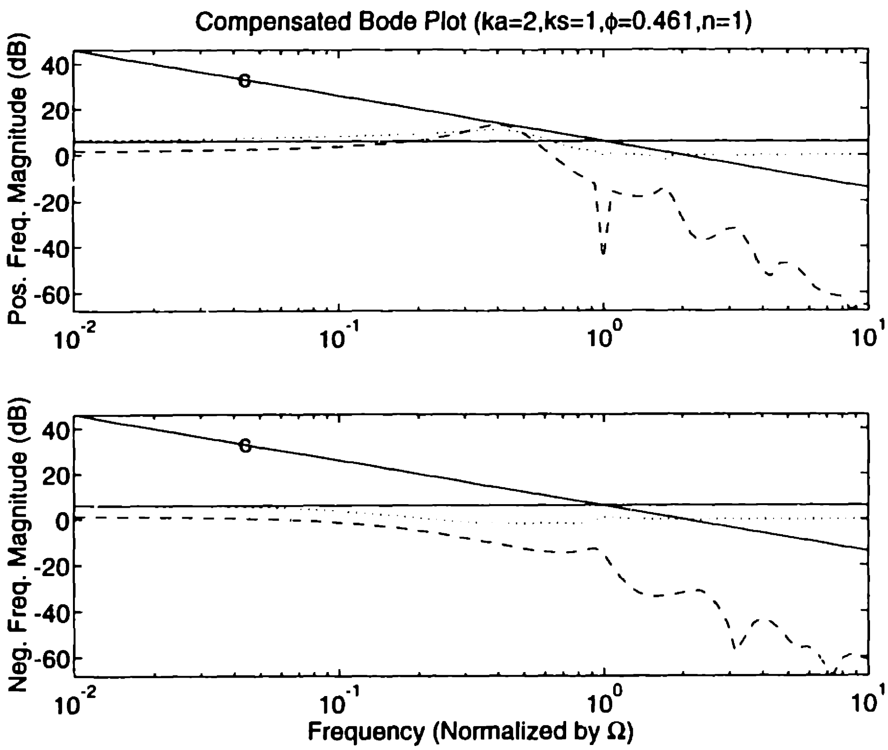

Figure D.1 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = .1$, $q = .01$, $(k_a, k_s) = (2,1)$.

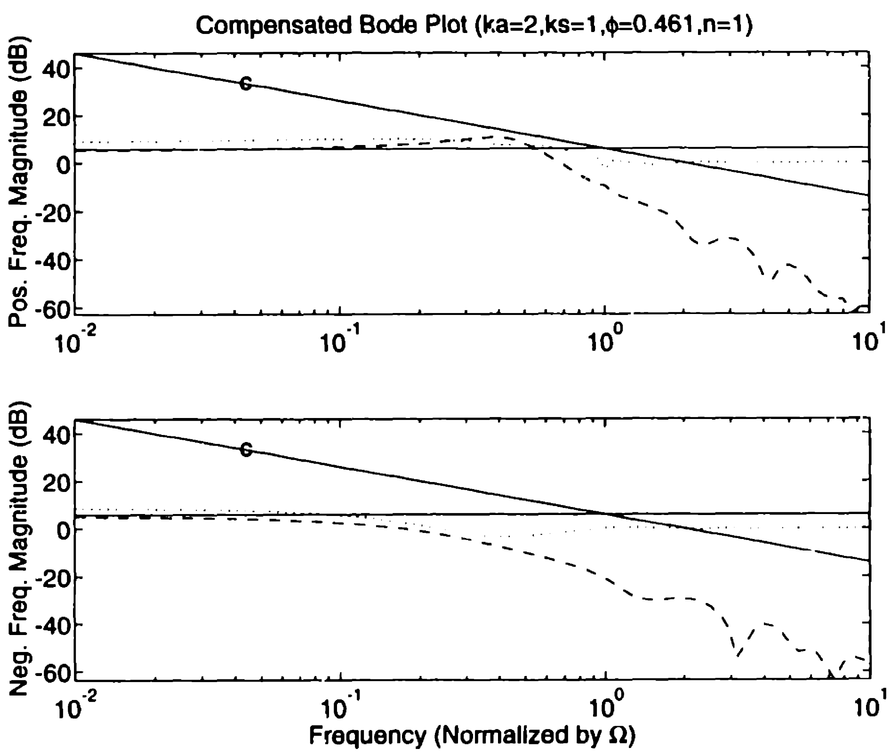

Figure D.2 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = 1$, $q = .01$, $(k_a, k_s) = (2,1)$.

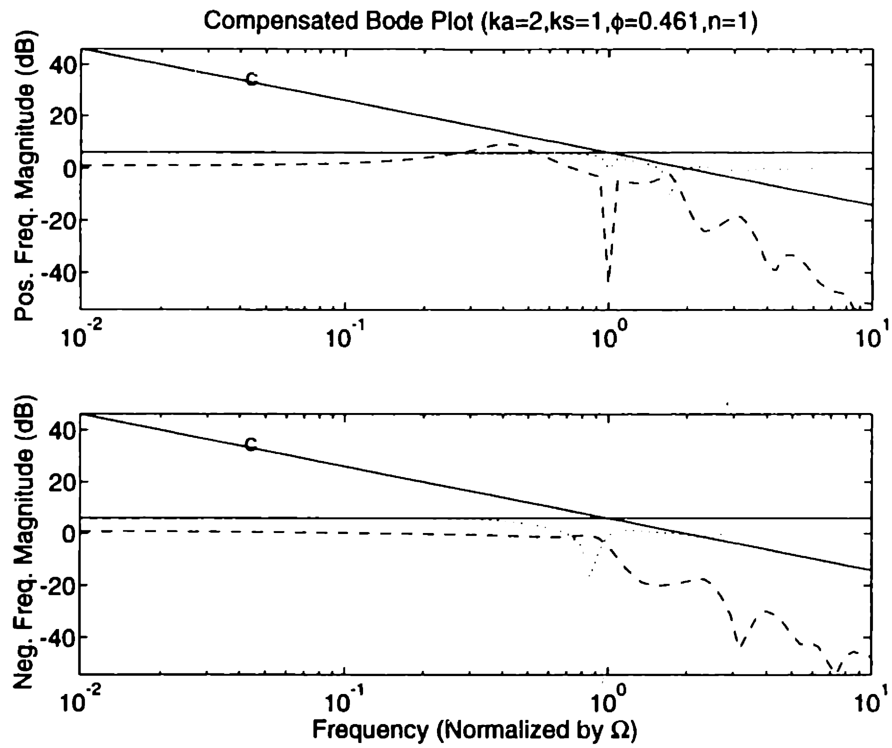

Figure D.3 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = .1$, $q = 1$, $(k_a, k_s) = (2,1)$.

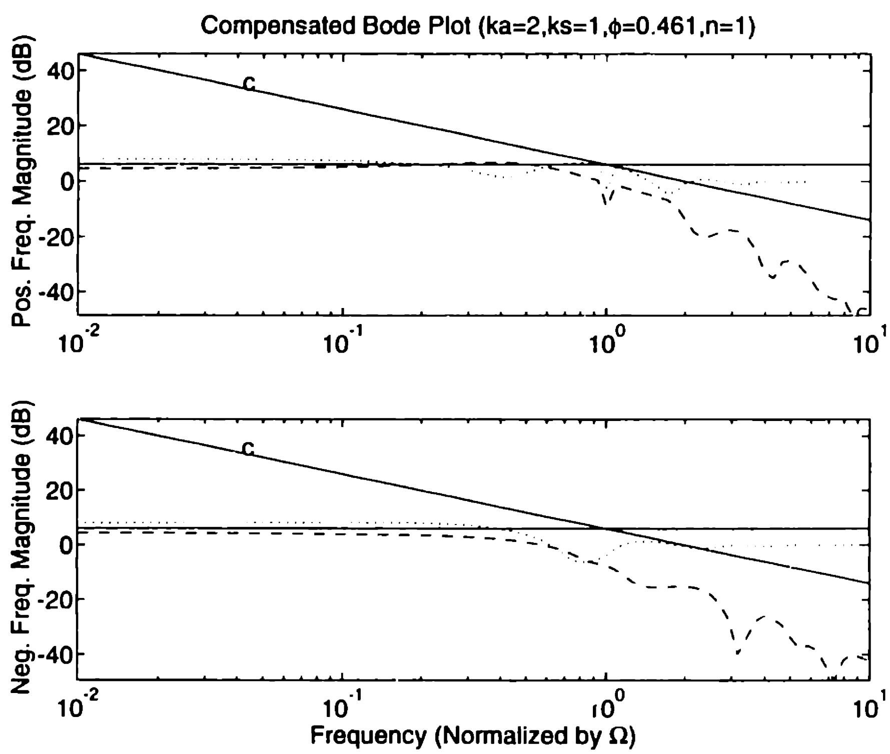

Figure D.4 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = 1$, $q = 1$, $(k_a, k_s) = (2,1)$.

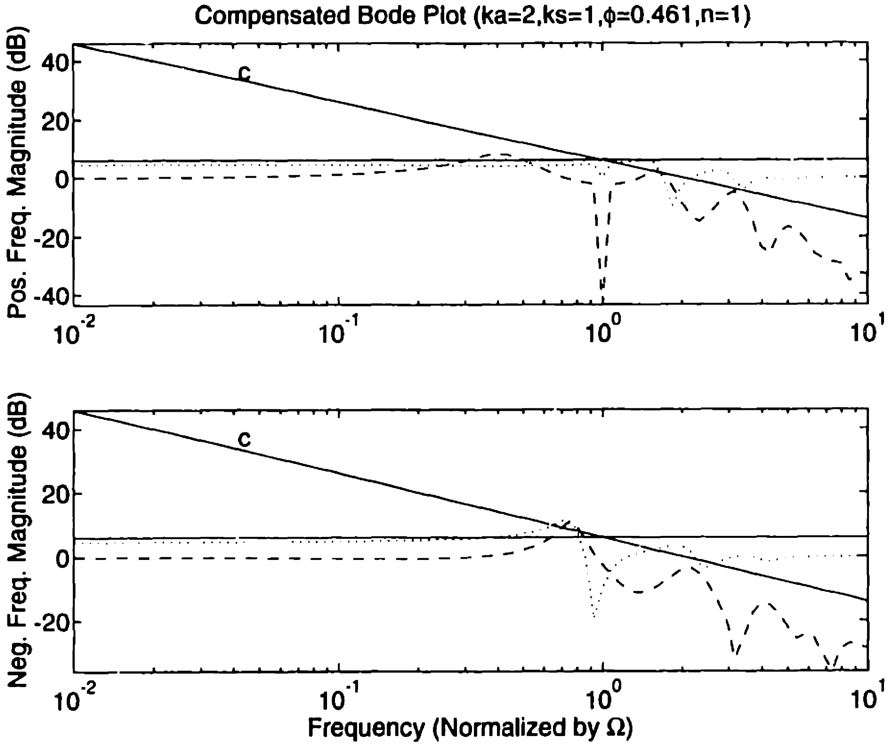

Figure D.5 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = .1$, $q = 100$, $(k_a, k_s) = (2,1)$.

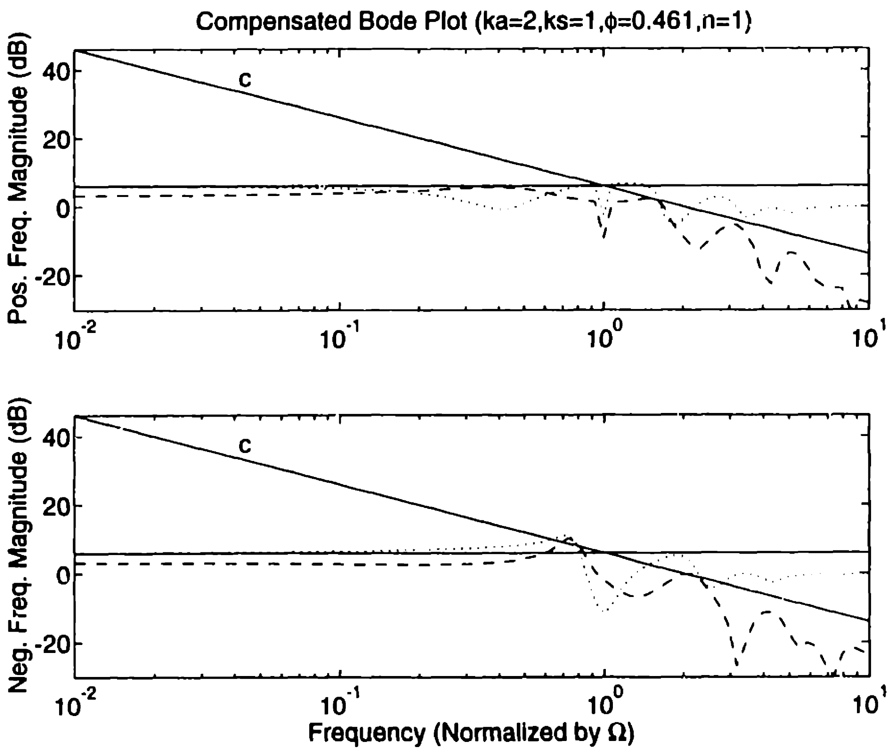

Figure D.6 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = 1$, $q = 100$, $(k_a, k_s) = (2,1)$.

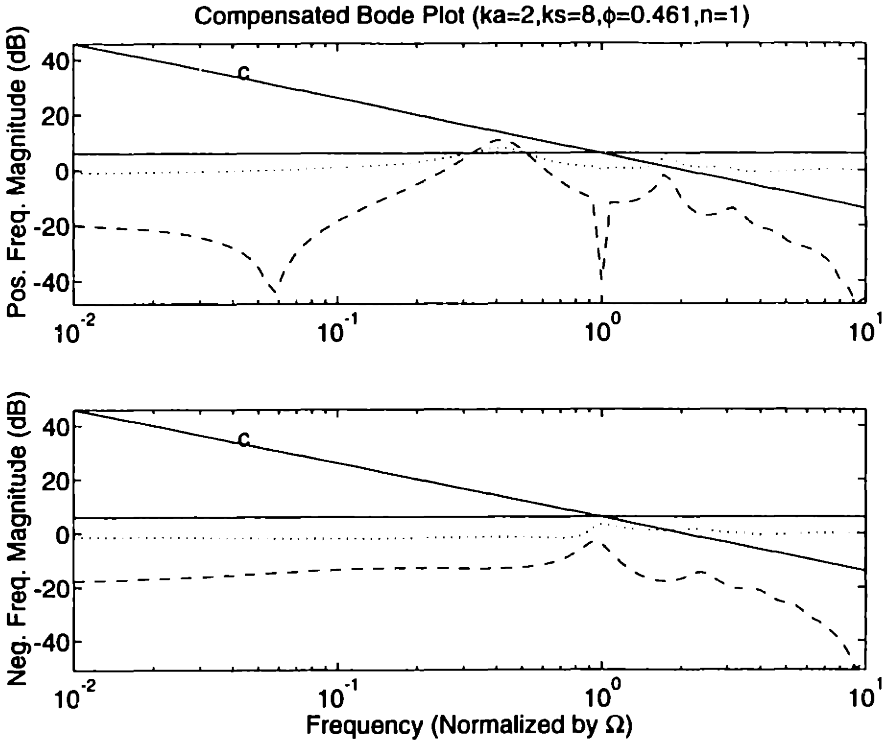

Figure D.7 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = .1$, $q = .01$, $(k_a, k_s) = (2,8)$.

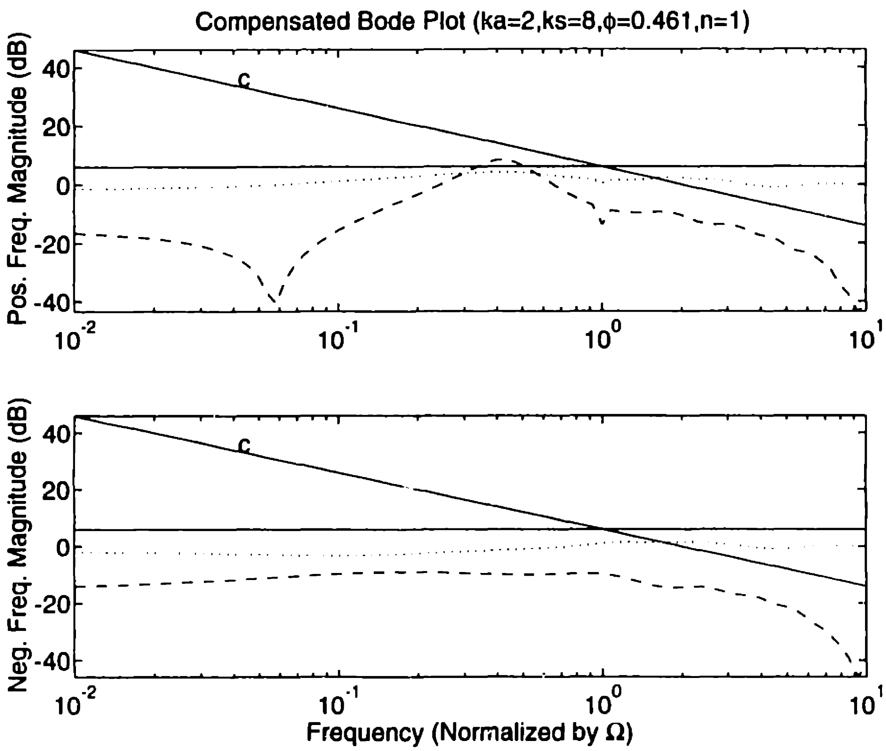

Figure D.8 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = 1$, $q = .01$, $(k_a, k_s) = (2,8)$.

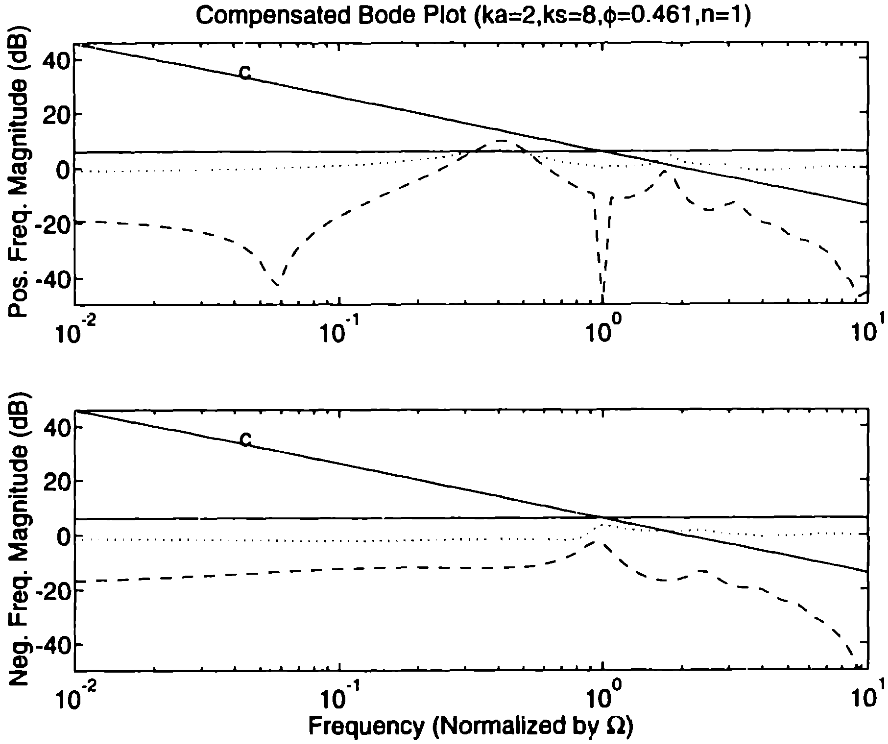

Figure D.9 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = .1$, $q = 1$, $(k_a, k_s) = (2,8)$.

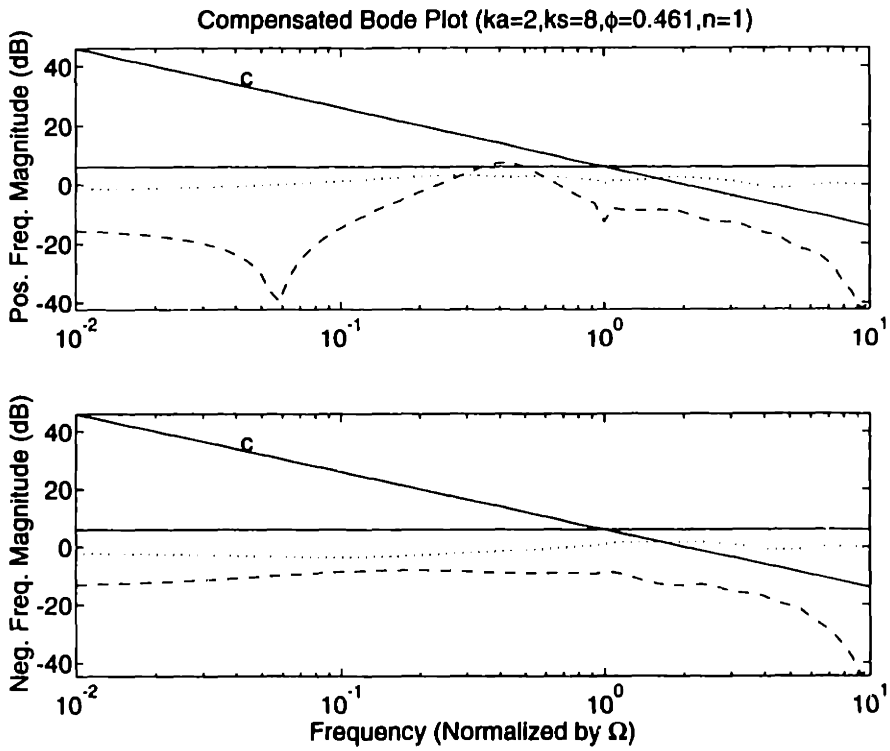

Figure D.10 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = 1$, $q = 1$, $(k_a, k_s) = (2,8)$.

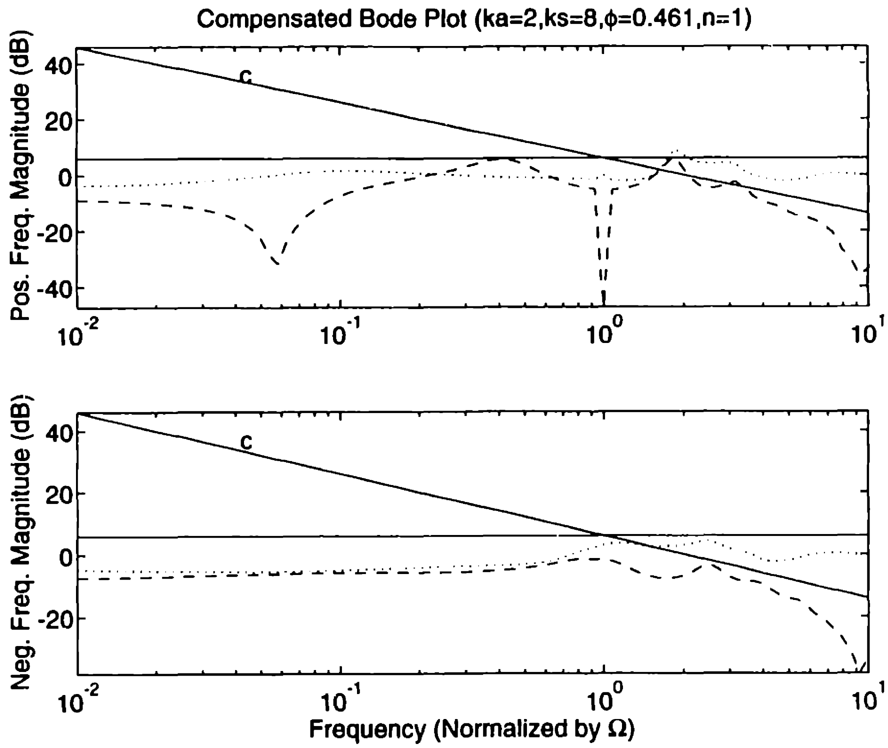

Figure D.11 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = .1$, $q = 100$, $(k_a, k_s) = (2,8)$.

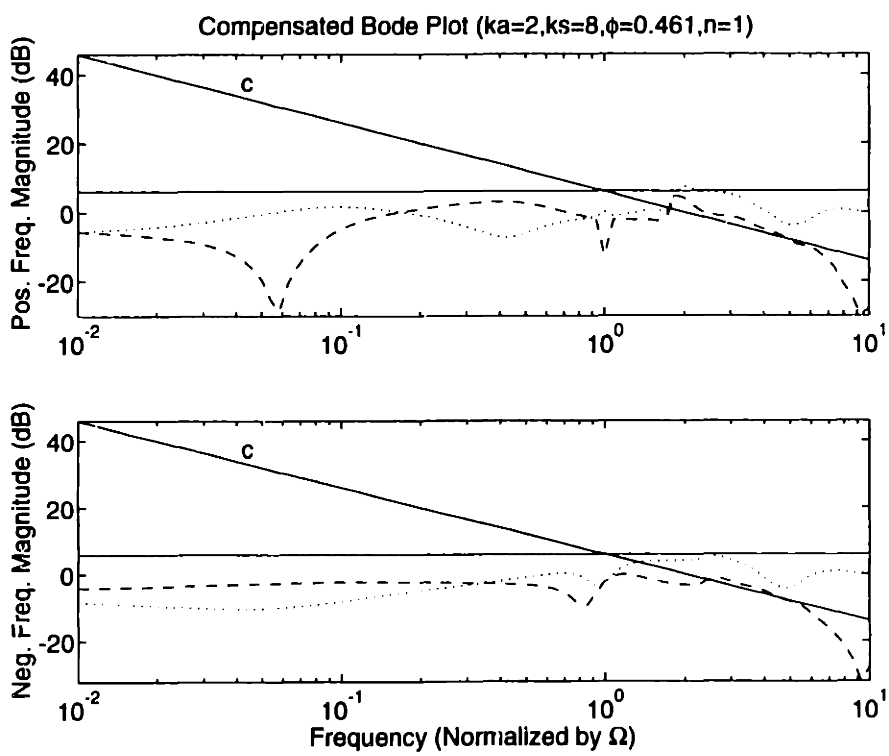

Figure D.12 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = 1$, $q = 100$, $(k_a, k_s) = (2,8)$.

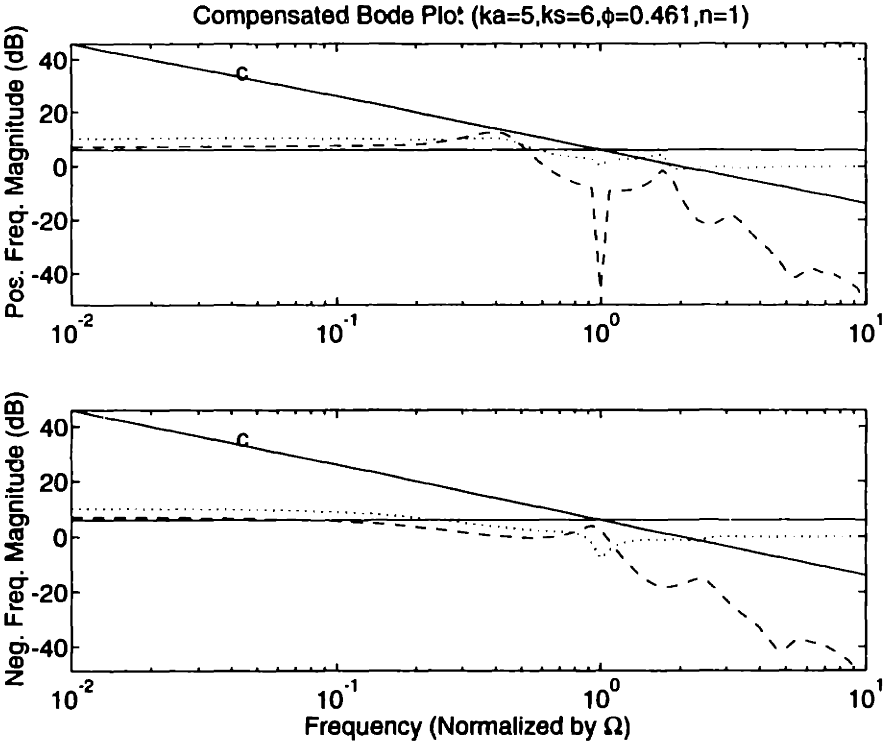

Figure D.13 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = .1$, $q = .01$, $(k_a, k_s) = (5,6)$.

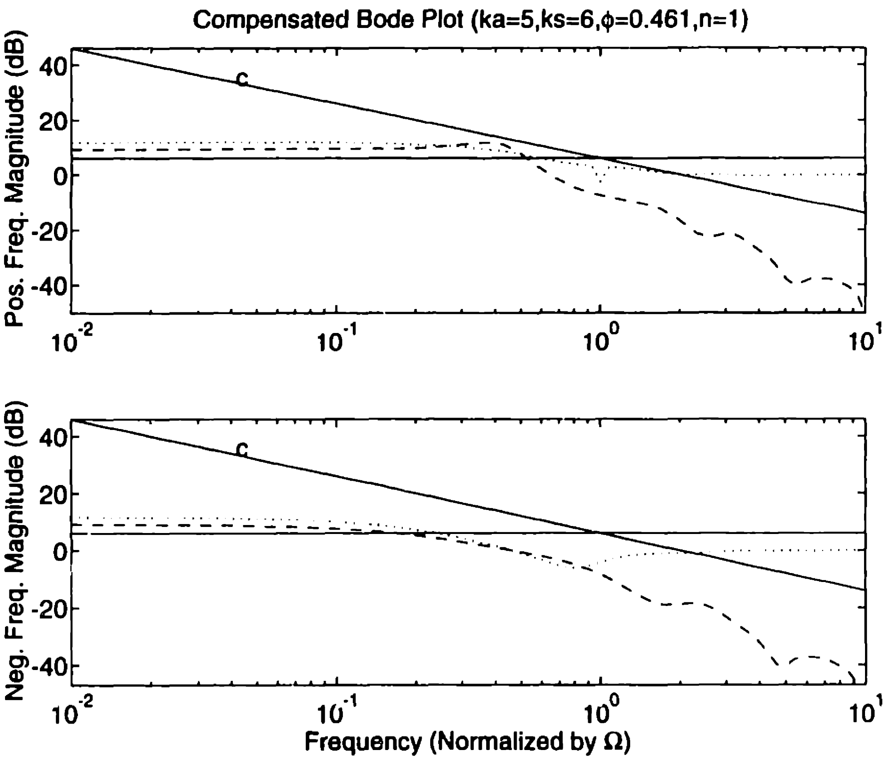

Figure D.14 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = 1$, $q = .01$, $(k_a, k_s) = (5,6)$.

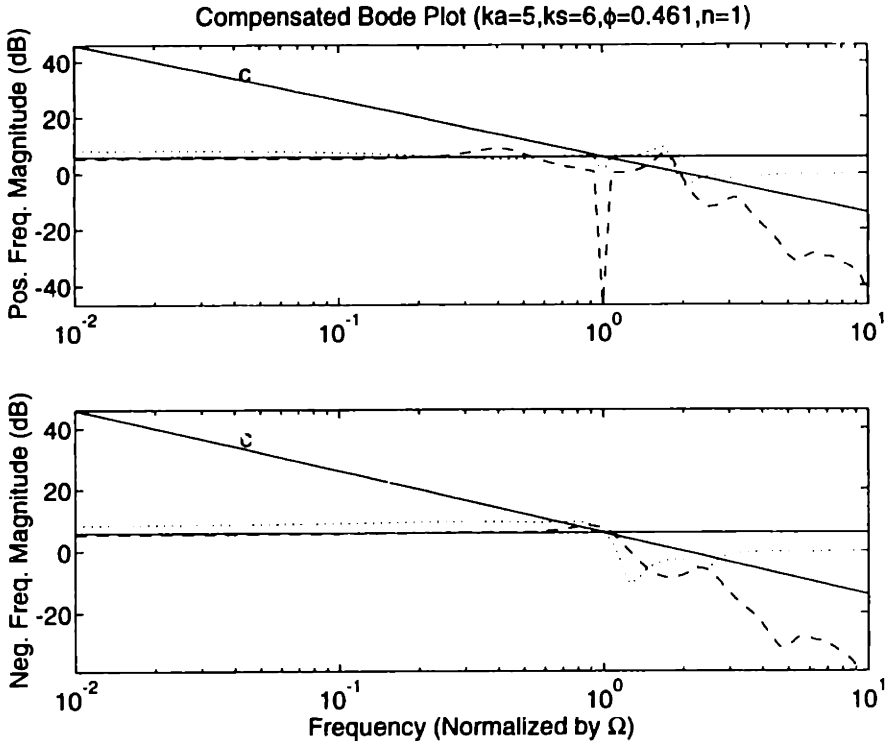

Figure D.15 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = .1$, $q = 1$, $(k_a, k_s) = (5,6)$.

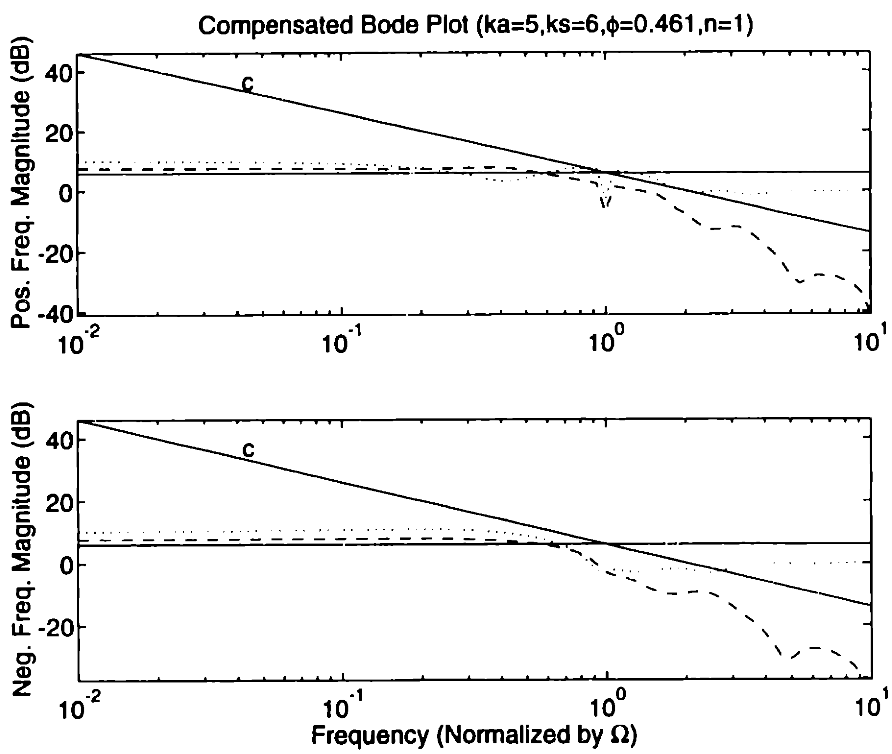

Figure D.16 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = 1$, $q = 1$, $(k_a, k_s) = (5,6)$.

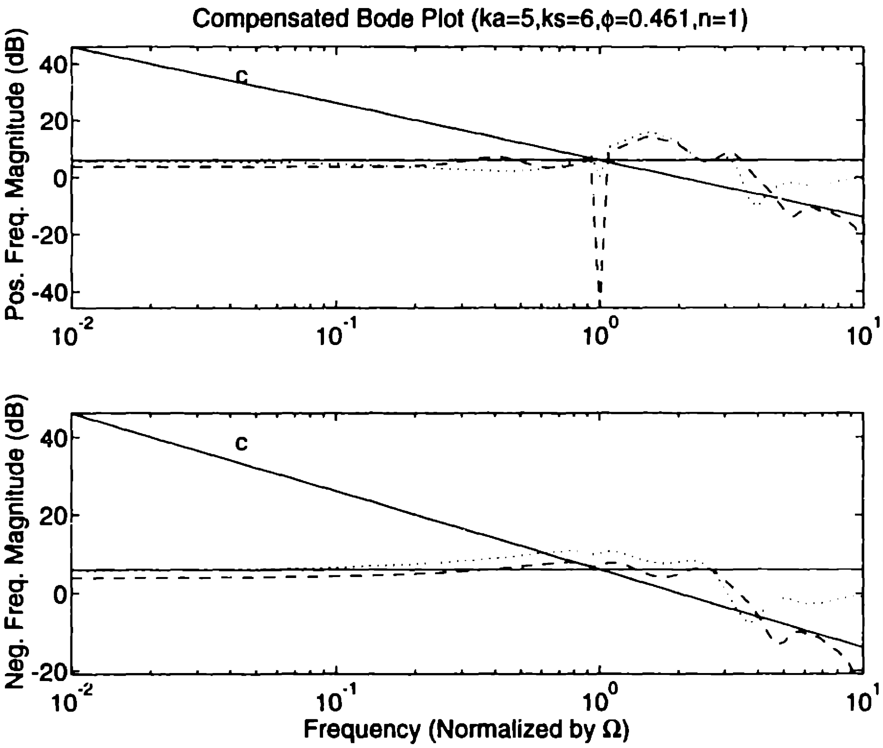

Figure D.17 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = .1$, $q = 100$, $(k_a, k_s) = (5,6)$.

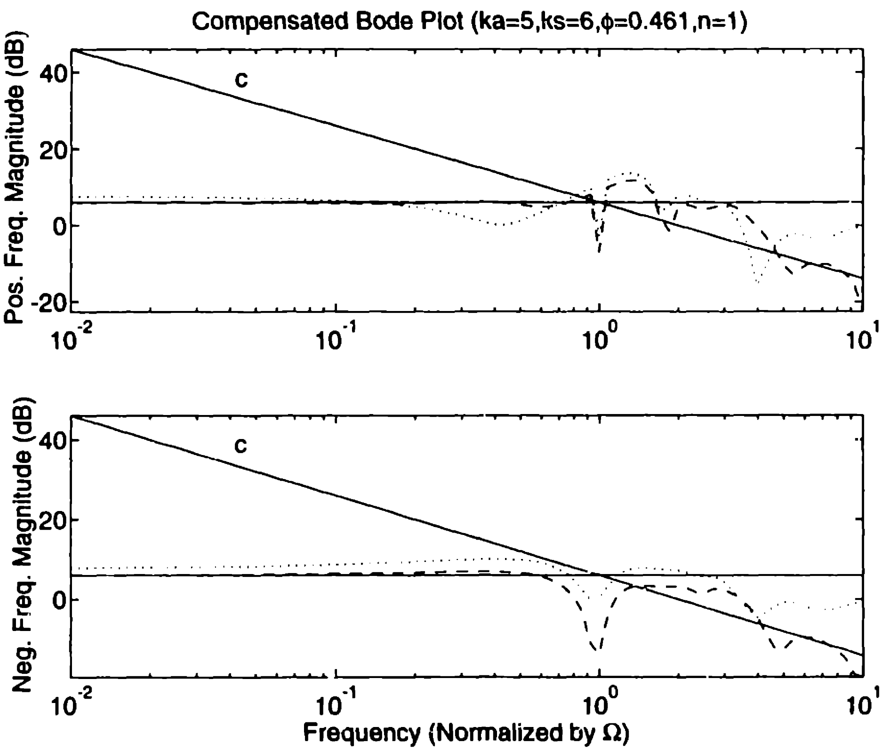

Figure D.18 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = 1$, $q = 100$, $(k_a, k_s) = (5,6)$.

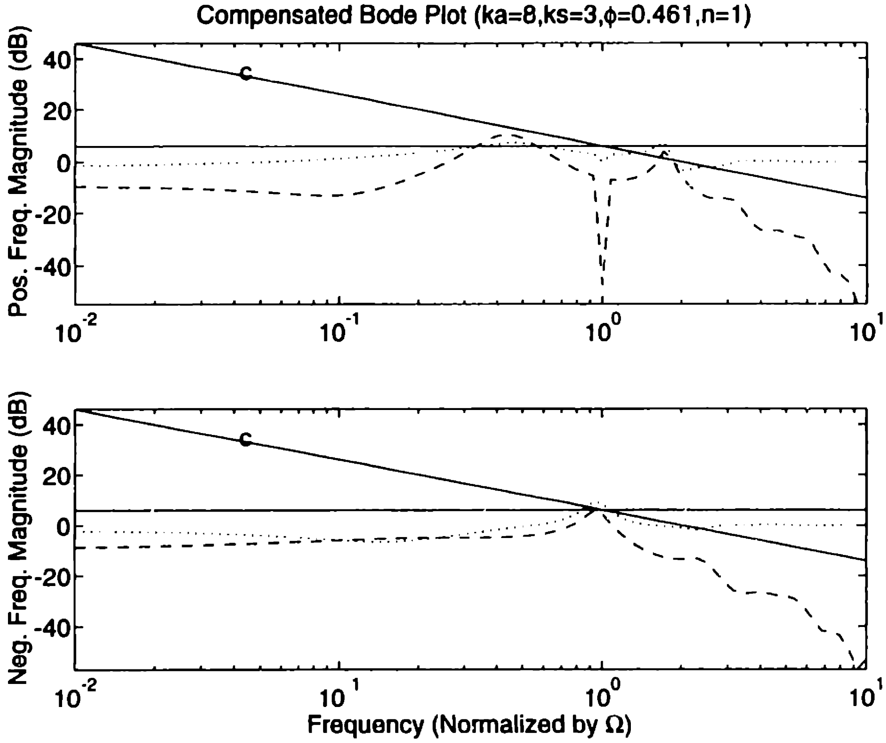

Figure D.19 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = .1$, $q = .01$, $(k_a, k_s) = (8,3)$.

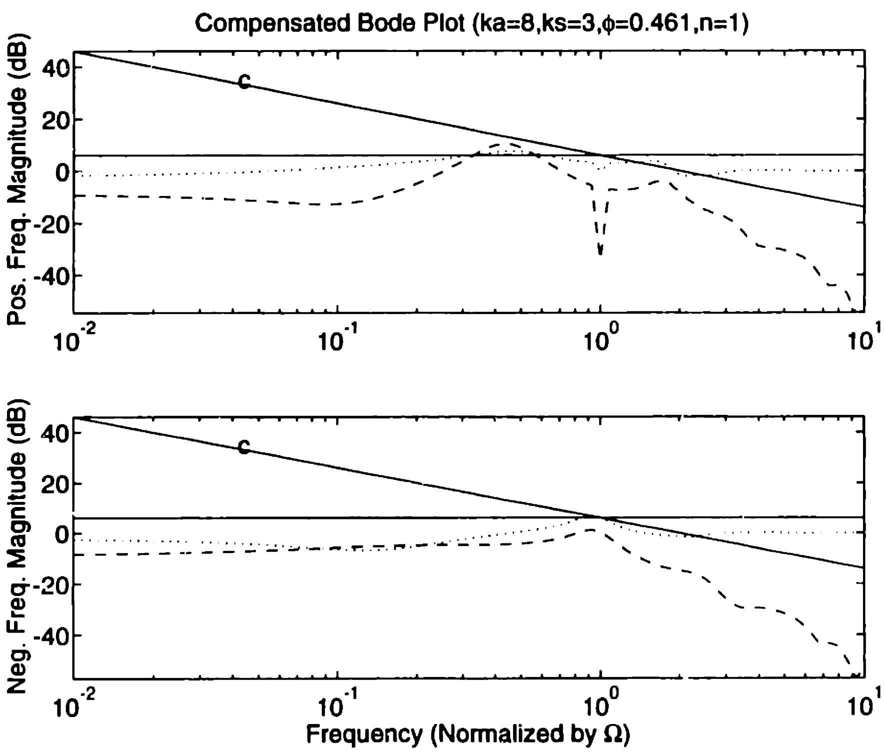

Figure D.20 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = 1$, $q = .01$, $(k_a, k_s) = (8,3)$.

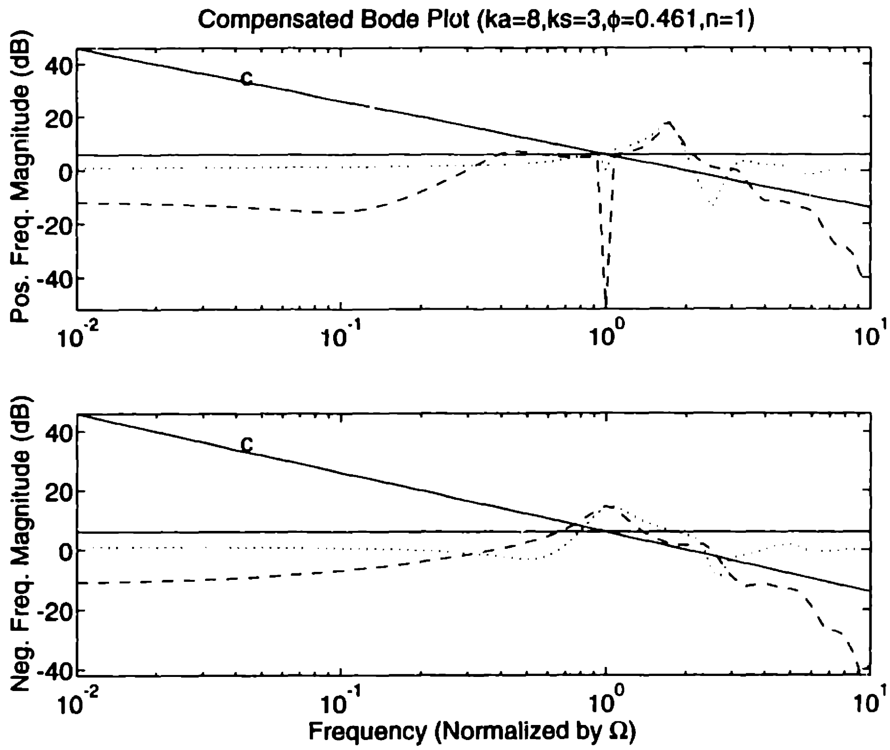

Figure D.21 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = .1$, $q = 1$, $(k_a, k_s) = (8,3)$.

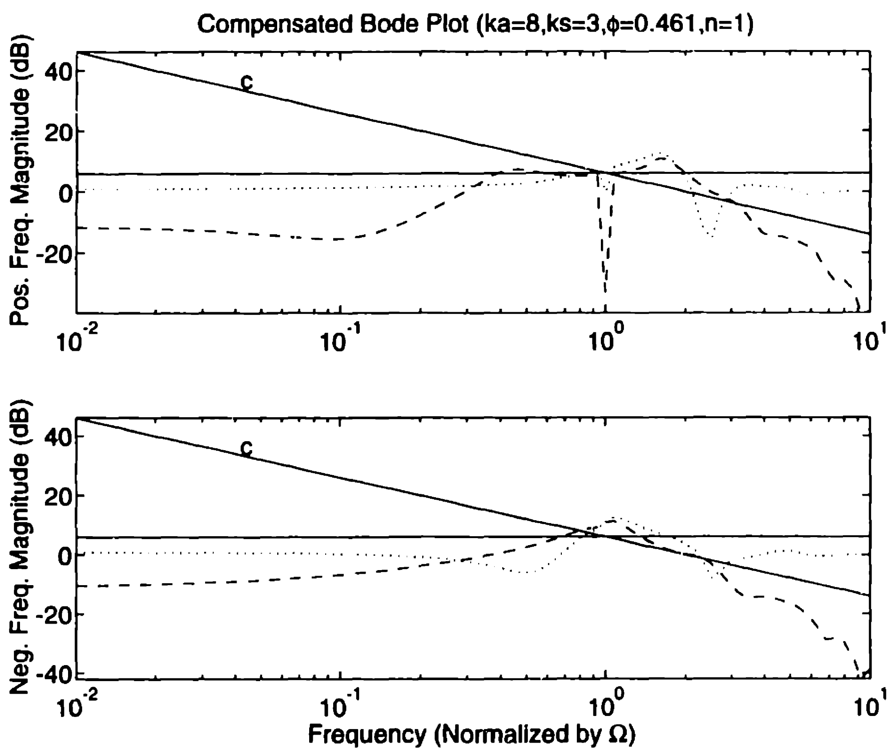

Figure D.22 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = 1$, $q = 1$, $(k_a, k_s) = (8,3)$.

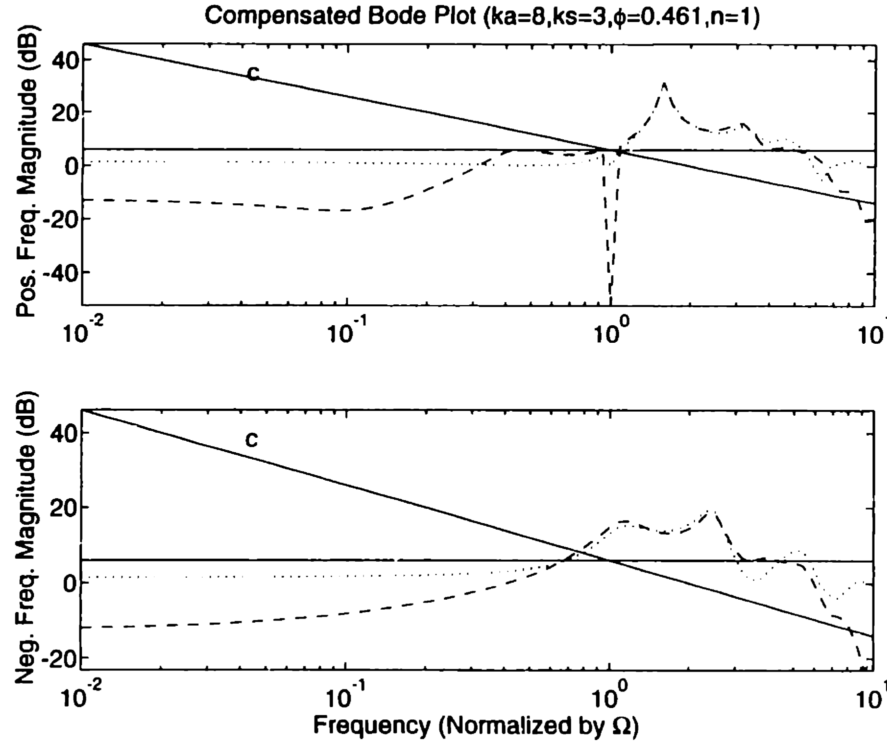

Figure D.23 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = .1$, $q = 100$, $(k_a, k_s) = (8,3)$.

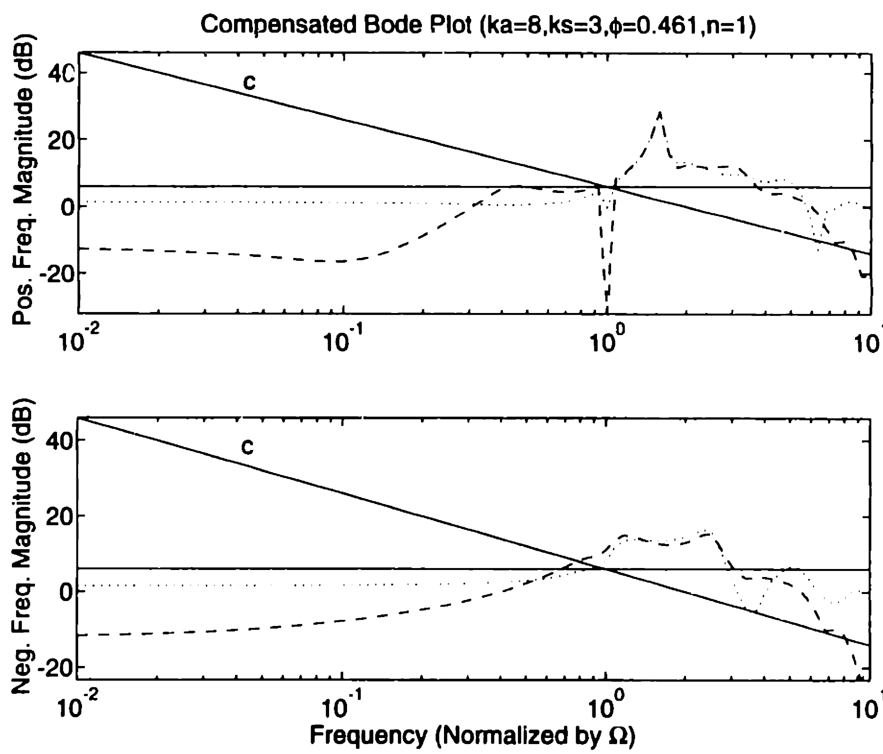

Figure D.24 - Comp. Sens. (dashed) with Constraint (solid-c) and Sens. (dotted) with Constraint (solid), $c = 1$, $q = 100$, $(k_a, k_s) = (8,3)$.

Appendix E

Control Configuration Cost Results

This appendix presents the cost results of the configuration studies in Chapter 5. The first three tables of each case were arrived at using the LQG design procedure described in Chapter 4. The last two tables of each case were arrived at using the H_∞ design procedure described in Chapter 4. No shading indicates values less than 20. Light shading indicates values between 20 and 100, and dark shading indicates values greater than 100. The black entries indicate configurations which were not able to meet the constraints in the H_∞ design procedure. The black entries do not indicate infinite cost, rather, if the constraints were relaxed enough to enable the configuration to meet the specifications, then the values of the black entries would not necessarily be high.

Table E.1 - Nominal case LQG costs (n=1, ϕ= .461, Measure $\delta P/P$).

Sensor Location - k_s

Actuator Location k_a	1	2	3	4	5	6	7	8
1	14.44	15.31	15.97	17.53	18.51	23.82	30.22	116.24
2	9.93	10.58	11.07	12.22	12.94	16.76	21.38	83.65
3	10.56	11.28	11.81	13.03	13.79	17.82	22.67	88.30
4	15.18	16.22	16.94	18.63	19.67	25.26	31.93	122.30
5	15.76	16.70	17.40	19.11	20.17	25.90	32.84	126.00
6	44.22	46.63	48.39	52.80	55.51	70.76	89.13	335.27
7	28.90	30.38	31.58	34.49	36.33	46.43	58.68	221.83
8	414.12	434.13	449.95	488.48	513.27	652.91	820.50	3048.9

Table E.2 - Nominal case state costs for LQG design (n=1, ϕ= .461, Measure $\delta P/P$).

Sensor Location - k_s

Actuator Location k_a	1	2	3	4	5	6	7	8
1	7.03	7.75	8.15	8.95	9.46	11.70	13.94	27.97
2	5.74	6.40	6.76	7.45	7.89	9.77	11.64	23.76
3	5.62	6.32	6.70	7.40	7.85	9.72	11.52	23.49
4	6.56	7.43	7.86	8.65	9.16	11.31	13.32	26.76
5	6.01	6.70	7.08	7.81	8.27	10.23	12.13	24.71
6	9.75	10.84	11.35	12.39	13.00	15.84	18.48	37.25
7	5.71	6.22	6.57	7.24	7.69	9.52	11.40	23.19
8	8.23	8.31	8.74	9.08	9.74	11.43	12.78	23.69

Table E.3 - Nominal case control costs for LQG design (n=1, ϕ=.461, Measure $\delta P/P$).

Sensor Location - k_s

Actuator Location k_a	1	2	3	4	5	6	7	8
1	5.07	5.51	5.70	6.05	6.28	7.32	8.33	16.82
2	2.42	2.64	2.74	2.90	3.01	3.48	3.94	8.13
3	3.05	3.32	3.44	3.65	3.79	4.41	5.02	10.19
4	6.13	6.63	6.86	7.31	7.60	8.93	10.27	20.33
5	7.19	7.75	8.02	8.55	8.90	10.48	12.07	23.92
6	28.71	30.76	31.75	33.90	35.28	41.88	48.65	94.41
7	19.16	20.57	21.26	22.70	23.63	28.02	32.44	63.48
8	360.89	385.23	397.36	423.62	441.03	526.07	612.94	1184.7

Table E.4 - Nominal case state costs for H_∞ design (n=1, ϕ= .461, Measure $\delta P/P$).

		Sensor Location - k_s							
		1	2	3	4	5	6	7	8
	1		8.55	9.93	9.68	10.87	11.38	14.19	731.15
	2	6.27	7.93	8.54	8.46	9.58	9.99	12.78	569.79
Actuator	3	5.74	7.38	8.94	8.21	9.87	9.90	14.05	366.39
Location	4	8.58	7.56	8.98	12.46	12.98	12.37	20.94	263.90
k_a	5	6.38	7.16	8.35	11.49	15.46		25.55	113.33
	6					17.96	102.01		
	7	5.34	6.66	7.86	10.77	14.32	85.27		
	8	5.44	6.77	7.79	10.36	13.70	80.87		

Table E.5 - Nominal case control costs for H_∞ design (n=1, ϕ= .461, Measure $\delta P/P$).

		Sensor Location - k_s							
		1	2	3	4	5	6	7	8
	1		5.06	6.40	5.38	6.03	6.21	8.17	1210.9
	2	2.10	2.38	2.89	2.44	2.75	2.83	3.75	556.10
Actuator	3	2.60	3.02	3.31	3.08	3.67	3.62	5.09	402.92
Location	4	7.67	5.97	6.57	8.14	8.76	8.19	13.52	432.06
k_a	5	7.66	7.67	8.09	9.92	11.69		18.17	189.24
	6					41.34	179.06		
	7	18.36	22.35	23.78	29.26	33.34	143.42		
	8	1241.4	1709.6	1419.3	1557.4	1146.9	3453.3		

Table E.6 - Case 2 LQG costs (n=0, ϕ= .461, Measure $\delta P/P$).

Actuator Location k_a	Sensor Location - k_s: 1	2	3	4	5	6	7	8
1	36.07	39.19	38.64	44.12	52.14	64.77	65.55	63.22
2	24.58	26.93	26.69	32.34	39.42	48.84	48.57	46.18
3	22.59	24.86	25.98	37.47	47.47	56.08	52.37	47.50
4	19.90	21.81	28.90	49.90	59.88	61.97	51.60	43.56
5	16.43	17.89	27.48	47.50	53.20	50.56	39.64	32.61
6	15.30	16.42	27.45	41.15	39.60	32.71	24.31	20.59
7	16.86	15.05	25.02	36.59	34.71	28.17	21.08	18.04
8	14.34	15.69	24.39	31.36	27.42	21.59	17.04	15.72

Table E.7 - Case 2 state costs for LQG design (n=0, ϕ= .461, Measure $\delta P/P$).

Actuator Location k_a	Sensor Location - k_s: 1	2	3	4	5	6	7	8
1	16.36	19.85	20.24	24.48	30.01	37.11	36.35	33.79
2	12.77	15.90	16.47	21.59	27.28	33.29	31.74	28.86
3	10.12	13.00	14.70	22.90	23.39	33.66	29.84	25.72
4	7.79	9.91	14.65	26.58	31.75	31.50	24.65	19.69
5	5.84	7.23	12.87	23.42	25.92	23.34	16.95	13.20
6	6.21	7.09	13.33	20.20	18.62	14.18	9.73	8.17
7	5.46	6.42	12.06	18.02	16.56	12.43	8.67	7.31
8	5.67	6.82	11.67	15.10	12.71	9.38	7.13	6.59

Table E.8 - Case 2 control costs for LQG design (n=0, ϕ= .461, Measure $\delta P/P$).

Actuator Location k_a	Sensor Location - k_s: 1	2	3	4	5	6	7	8
1	10.69	12.66	12.61	16.01	20.08	24.32	23.18	20.87
2	5.10	6.11	6.23	8.40	10.63	12.61	11.72	10.34
3	6.01	7.13	8.10	12.86	16.17	17.78	15.37	12.91
4	7.28	8.35	12.45	21.24	23.90	22.29	17.04	13.41
5	7.28	8.20	13.18	21.10	21.91	18.74	13.70	10.77
6	7.09	7.77	12.10	16.14	14.52	11.12	8.32	7.30
7	6.54	7.16	10.99	14.19	12.51	9.43	7.14	6.43
8	6.61	7.35	10.36	11.77	9.69	7.55	6.50	6.50

Table E.9 - Case 2 state costs for H_∞ design (n=0, ϕ= .461, Measure $\delta P/P$).

Actuator Location k_a \ Sensor Location - k_s	1	2	3	4	5	6	7	8
1								32.54
2							28.83	26.01
3							30.12	28.22
4							30.73	29.55
5						25.32	25.44	25.23
6				18.28	19.42	40.24	31.35	29.17
7			15.77	18.66	19.42	33.89	13.84	19.95
8	8.40	12.57	15.12	17.05	16.36	17.11	9.12	7.35

Table E.10 - Case 2 control costs for H_∞ design (n=0, ϕ= .461, Measure $\delta P/P$).

Actuator Location k_a \ Sensor Location - k_s	1	2	3	4	5	6	7	8
1								21.38
2							9.25	8.49
3							13.40	11.94
4							20.23	17.88
5						18.95	20.26	17.38
6				12.73	15.49	92.46	62.88	49.53
7			9.97	12.00	14.80	81.08	13.04	34.55
8	14.44	10.30	9.94	10.70	11.63	25.66	6.70	6.44

Table E. 11 - Case 3 LQG costs (n=2, ϕ= .461, Measure $\delta P/P$).

		Sensor Location - k_s							
		1	2	3	4	5	6	7	8
	1	10.69	10.80	11.15	11.40	11.79	12.81	13.18	18.50
	2	7.84	7.98	8.32	8.56	8.94	9.93	10.29	15.52
Actuator	3	8.54	8.69	9.04	9.28	9.66	10.65	11.00	16.25
Location	4	10.91	11.17	11.53	11.78	12.17	13.20	13.53	18.80
k_a	5	11.19	11.30	11.63	11.87	12.26	13.26	13.65	19.03
	6	17.82	18.01	18.34	18.62	18.99	20.09	20.49	26.23
	7	13.22	13.25	13.62	13.84	14.25	15.27	15.65	21.03
	8	28.41	28.38	28.84	29.04	29.59	30.89	31.40	37.97

Table E. 12 - Case 3 state costs for LQG design (n=2, ϕ= .461, Measure $\delta P/P$).

		Sensor Location - k_s							
		1	2	3	4	5	6	7	8
	1	4.99	5.20	5.51	5.70	6.06	6.73	7.01	9.62
	2	4.28	4.54	4.83	5.01	5.37	5.98	6.23	8.67
Actuator	3	4.36	4.64	4.95	5.12	5.49	6.11	6.34	8.83
Location	4	4.65	5.05	5.38	5.55	5.94	6.63	6.86	9.40
k_a	5	4.68	4.89	5.19	5.37	5.73	6.39	6.67	9.42
	6	6.60	6.91	7.20	7.42	7.76	8.56	8.85	12.06
	7	4.61	4.68	5.01	5.18	5.56	6.26	6.56	9.46
	8	9.25	9.19	9.58	9.70	10.19	11.13	11.57	15.54

Table E. 13 - Case 3 control costs for LQG design (n=2, ϕ= .461, Measure $\delta P/P$).

		Sensor Location - k_s							
		1	2	3	4	5	6	7	8
	1	1.16	1.22	1.19	1.17	1.16	1.06	1.02	.64
	2	.68	.72	.71	.69	.68	.62	.58	.34
Actuator	3	.81	.86	.84	.82	.82	.74	.71	.43
Location	4	1.25	1.29	1.27	1.24	1.23	1.13	1.10	.70
	5	1.44	1.50	1.48	1.46	1.45	1.34	1.29	.84
	6	3.24	3.31	3.29	3.26	3.25	3.08	3.02	2.24
	7	2.18	2.26	2.23	2.21	2.20	2.06	1.98	1.37
	8	9.97	10.04	10.00	9.97	9.97	9.74	9.71	8.47

Table E.14 - Case 3 state costs for H_∞ design (n=2, ϕ= .461, Measure $\delta P/P$).

Actuator Location k_a \ Sensor Location - k_s	1	2	3	4	5	6	7	8
1	39.34	37.84	37.57	37.62	37.94	38.51	38.09	37.52
2	36.22	38.05	37.43	37.33	37.49	38.08	37.72	37.21
3	35.87	36.19	38.45	38.06	37.72	38.33	38.21	37.67
4	36.02	36.29	37.43	40.14	38.50	39.20	39.01	38.81
5	35.54	35.79	36.70	37.51	39.93	39.96	40.11	39.91
6	36.82	37.08	38.01	38.81	40.59	47.04	45.94	46.94
7	34.30	34.55	35.44	36.17	37.43	41.63	40.09	41.87
8	33.66	33.92	34.80	35.53	36.80	41.04	39.85	62.10

Table E.15 - Case 3 control costs for H_∞ design (n=2, ϕ= .461, Measure $\delta P/P$).

Actuator Location k_a \ Sensor Location - k_s	1	2	3	4	5	6	7	8
1	.21	.21	.21	.20	.21	.21	.21	.68
2	.10	.10	.10	.10	.10	.10	.10	.25
3	.13	.13	.14	.13	.13	.13	.13	.21
4	.24	.24	.24	.25	.26	.25	.25	.31
5	.29	.29	.29	.30	.31	.31	.30	.32
6	1.00	1.01	1.01	1.02	1.03	1.09	1.08	1.06
7	.52	.53	.53	.54	.53	.55	.55	.62
8	5.74	6.38	6.00	6.12	5.67	5.67	5.68	14.11

Table E. 16 - Case 4 LQG costs (n=1, ϕ= .4506, Measure $\delta P/P$).

Actuator Location k_a	Sensor Location - k_s: 1	2	3	4	5	6	7	8
1	33.02	35.53	38.69	47.64	52.09	91.15	126.99	653.35
2	20.19	21.82	23.86	29.46	32.28	56.41	78.55	404.56
3	23.11	25.08	27.49	33.92	37.15	64.82	90.01	462.46
4	41.38	45.02	49.12	60.45	66.10	115.52	160.36	821.66
5	46.11	49.61	54.04	66.54	72.72	127.33	177.22	911.53
6	192.26	206.35	223.16	273.73	298.78	524.16	730.80	3757.2
7	103.28	110.25	119.56	146.79	160.18	281.20	392.22	2018.1
8	1127.4	1203.9	1297.2	1584.0	1724.7	3037.4	4238.7	21733

Table E. 17 - Case 4 state costs for LQG design (n=1, ϕ= .4506, Measure $\delta P/P$).

Actuator Location k_a	Sensor Location - k_s: 1	2	3	4	5	6	7	8
1	10.81	12.68	13.98	16.60	18.13	26.88	32.77	82.24
2	8.61	10.22	11.32	13.45	14.71	21.81	26.58	67.51
3	8.50	10.31	11.50	13.66	14.95	22.10	26.58	67.52
4	11.24	13.93	15.44	18.17	19.87	29.19	34.63	87.99
5	10.48	12.72	14.18	16.77	18.32	26.78	31.57	82.24
6	25.79	31.61	34.66	40.42	43.71	63.51	72.68	196.34
7	10.94	12.83	14.43	17.06	18.67	26.77	30.89	79.77
8	26.48	29.43	32.11	35.58	38.57	54.51	53.90	106.28

Table E. 18 - Case 4 control costs for LQG design (n=1, ϕ= .4506, Measure $\delta P/P$).

Actuator Location	Sensor Location - k_s: 1	2	3	4	5	6	7	8
1	13.96	16.01	17.21	19.93	21.45	31.77	39.24	108.02
2	6.31	7.28	7.85	9.09	9.79	14.47	17.88	50.17
3	8.62	9.86	10.62	12.33	13.29	19.69	25.46	66.76
4	19.93	22.61	24.27	28.23	30.45	45.42	56.70	150.62
5	24.28	27.43	29.47	34.36	37.03	55.30	69.00	183.07
6	122.08	137.12	146.44	170.33	183.0	274.26	343.14	899.00
7	68.07	76.82	82.34	95.85	103.17	154.23	191.78	508.79
8	851.02	956.22	1018.6	1179.5	1268.5	1897.0	2359.0	6231.2

Table E.19 - Case 4 state costs for H_∞ design (n=1, ϕ= .4506, Measure $\delta P/P$).

Actuator Location k_a	Sensor Location - k_s: 1	2	3	4	5	6	7	8
1								3955.3
2								3625.3
3								2058.2
4								1476.8
5								
6								
7								
8				21.29	139.76			

Table E.20 - Case 4 control costs for H_∞ design (n=1, ϕ= .4506, Measure $\delta P/P$).

Actuator Location k_a	Sensor Location - k_s: 1	2	3	4	5	6	7	8
1								7442.4
2								3878.2
3								2759.6
4								23237
5								
6								
7								
8				6118.5	17366			

Table E.21 - Case 5 LQG costs (n=1, ϕ= .2644, Measure $\delta P/P$).

Actuator Location k_a \ Sensor Location - k_s	1	2	3	4	5	6	7	8
1	448.32	317.86	329.70	384.84	460.00	391.62	386.47	893.23
2	44.03	30.72	31.82	37.35	44.99	38.20	37.76	89.94
3	46.18	32.30	33.47	39.28	47.25	40.22	39.74	94.06
4	90.78	64.11	66.47	77.77	93.10	79.51	78.51	182.55
5	93.52	66.05	68.47	80.06	95.86	81.81	80.79	187.96
6	55.24	38.86	40.34	47.32	56.74	48.44	47.81	111.76
7	109.28	77.23	80.12	93.66	112.15	95.52	94.29	219.07
8	2421.8	1717.1	1781.5	2078.8	2485.3	2113.1	2084.8	4816.3

Table E.22 - Case 5 state costs for LQG design (n=1, ϕ= .2644, Measure $\delta P/P$).

Actuator Location k_a \ Sensor Location - k_s	1	2	3	4	5	6	7	8
1	20.30	15.78	16.51	19.75	23.96	20.72	20.22	32.95
2	17.68	13.70	14.25	16.95	20.55	17.68	17.31	28.53
3	16.50	12.72	13.28	15.89	19.31	16.65	16.27	26.84
4	20.58	16.24	16.99	20.23	24.32	21.27	20.72	33.15
5	18.10	14.26	14.92	17.80	21.51	18.77	18.27	29.56
6	13.25	10.12	10.67	12.99	15.90	13.73	13.36	22.11
7	13.46	10.16	10.70	13.00	16.03	13.71	13.35	22.57
8	11.71	10.52	11.91	14.60	19.02	15.46	13.95	20.58

Table E.23 - Case 5 control costs for LQG design (n=1, ϕ= .2644, Measure $\delta P/P$).

Actuator Location k_a \ Sensor Location - k_s	1	2	3	4	5	6	7	8
1	317.95	272.55	282.00	320.45	371.33	328.90	321.13	466.60
2	12.89	11.84	12.23	13.70	15.64	14.17	13.78	18.95
3	15.55	14.09	14.55	16.35	18.71	16.90	16.45	22.81
4	45.12	39.39	40.71	46.10	53.17	47.47	46.35	66.06
5	49.62	43.08	44.52	50.47	58.26	51.96	50.74	72.67
6	25.36	22.35	23.11	26.14	30.10	26.96	26.29	37.15
7	66.22	57.33	59.31	67.28	77.77	69.17	67.51	97.11
8	1832.1	1562.1	1616.0	1837.6	2131.7	1884.5	1840.8	2689.4

Table E.24 - Case 5 state costs for H_∞ design (n=1, $\phi = .2644$, Measure $\delta P/P$).

Sensor Location - k_s

Actuator Location k_a	1	2	3	4	5	6	7	8
1		16.33	17.13			21.81	23.00	201.55
2	13.68	15.12	16.15			21.24	22.66	184.72
3	12.58	14.11	16.26			20.80	22.20	155.02
4				24.26		31.20	34.76	166.39
5	12.33	12.90	15.04	21.96	41.28	31.58	36.96	99.31
6	10.28	11.53	13.19	19.32	36.98	26.41	25.20	86.41
7	10.41	11.40	12.90	18.79	35.63	25.33	34.56	139.33
8	8.81	9.64	10.92	16.01	30.56	21.13	28.48	

Table E.25 - Case 5 control costs for H_∞ design (n=1, $\phi = .2644$, Measure $\delta P/P$).

Sensor Location - k_s

Actuator Location k_a	1	2	3	4	5	6	7	8
1		303.57	318.07			392.58	405.25	10400
2	10.64	10.70	11.29			13.87	14.23	295.78
3	12.74	13.10	14.09			17.14	17.58	311.11
4				49.18		58.04	61.59	680.61
5	44.81	39.59	43.22	54.92	81.61	68.92	75.62	433.10
6	21.98	22.87	24.54	30.80	45.95	36.28	35.08	276.71
7	61.63	63.03	66.77	83.59	120.76	95.91	120.34	537.11
8	16999	17375	16557	19344	9514.6	12154	6926.1	

Table E.26 - Case 6 LQG costs (n=1, ϕ= .3898, Measure $\delta P/P$).

Sensor Location - k_s

Actuator Location k_a	1	2	3	4	5	6	7	8
1	13.07	13.51	20.72	23.99	32.69	119.23	23.62	40.40
2	8.17	8.45	13.06	15.09	20.74	75.53	14.96	25.59
3	234.61	241.95	366.05	426.15	569.10	2088.0	412.75	707.20
4	236.47	244.03	368.97	430.29	576.28	2111.4	417.67	715.02
5	466.55	480.95	731.76	847.73	1136.1	4162.6	823.71	1411.1
6	215.10	219.63	332.81	387.01	516.57	1896.0	374.72	642.26
7	81.46	83.99	128.31	148.29	199.37	729.75	144.47	247.45
8	386.88	398.89	606.61	703.32	942.86	3453.8	683.49	1170.7

Table E.27 - Case 6 state costs for LQG design (n=1, ϕ= .3898, Measure $\delta P/P$).

Sensor Location - k_s

Actuator Location k_a	1	2	3	4	5	6	7	8
1	3.82	4.00	5.83	5.92	8.69	11.24	6.56	8.71
2	3.00	3.15	4.64	4.68	7.02	9.11	5.29	7.01
3	9.46	9.81	10.31	11.75	13.75	32.21	9.92	12.94
4	10.47	10.99	12.27	14.43	18.62	37.31	13.33	17.17
5	16.97	17.33	21.31	20.80	27.56	49.58	19.84	26.09
6	5.71	5.76	6.45	7.38	7.84	18.24	5.50	7.56
7	4.55	4.70	7.01	6.67	9.67	12.39	7.20	9.61
8	5.14	5.18	7.56	7.04	9.34	13.52	6.62	8.88

Table E.28 - Case 6 control costs for LQG design (n=1, ϕ= .3898, Measure $\delta P/P$).

Sensor Location - k_s

Actuator Location k_a	1	2	3	4	5	6	7	8
1	7.68	7.98	10.09	10.31	14.05	22.26	11.02	14.04
2	3.99	4.15	5.20	5.31	7.25	11.91	5.69	7.23
3	206.27	214.06	280.16	286.72	387.76	563.65	303.90	392.57
4	206.61	214.47	280.01	286.82	388.42	565.90	304.46	392.97
5	412.04	427.56	560.01	572.00	774.69	1135.8	606.85	783.26
6	190.24	197.37	257.70	263.56	356.36	522.25	279.24	360.58
7	69.86	72.51	94.49	96.46	130.79	194.73	102.35	131.91
8	350.42	363.63	474.06	484.76	656.49	970.35	513.37	662.56

Table E.29 - Case 6 state costs for H_∞ design (n=1, ϕ= .3898, Measure $\delta P/P$).

Actuator Location - k_a \ Sensor Location - k_s	1	2	3	4	5	6	7	8
1	4.23	5.69			89.77		7.82	162.63
2	3.26	3.80			57.84		6.81	109.37
3	6.80	7.34	18.64					
4			14.82					
5								
6	6.20	6.84	21.27					
7	4.47	5.05	14.29					
8	4.00	4.51	12.59					

Table E.30 - Case 6 control costs for H_∞ design (n=1, ϕ= .3898 Measure $\delta P/P$).

Actuator Location - k_a \ Sensor Location - k_s	1	2	3	4	5	6	7	8
1	7.78	10.35			118.93		14.26	1364.8
2	3.78	4.20			45.54		7.25	665.85
3	294.00	332.22	998.76					
4			582.03					
5								
6	366.12	406.03	1191.8					
7	84.67	100.61	164.74					
8	1671.1	2285.9	1072.2					

Table E.31 - Case 7 LQG costs (n=1, ϕ= .461, Use $\delta V_X/a$ in costs).

Actuator Location k_a	Sensor Location - k_s: 1	2	3	4	5	6	7	8
1	14.08	14.97	15.64	17.27	18.28	23.96	30.75	121.88
2	10.20	10.93	11.47	12.76	13.55	17.98	23.25	94.28
3	11.94	12.79	13.42	14.89	15.79	20.81	26.75	107.18
4	18.21	19.42	20.30	22.37	23.65	30.77	39.19	153.02
5	19.02	20.19	21.08	23.21	24.52	31.87	40.64	158.73
6	50.06	52.81	54.86	59.92	63.05	80.80	102.02	386.69
7	32.70	34.47	35.85	39.23	41.34	53.25	67.53	258.57
8	406.65	427.76	442.28	480.42	504.63	642.19	806.96	3003.6

Table E.32 - Case 7 state costs for LQG design (n=1, ϕ= .461, Use $\delta V_X/a$ in costs).

Actuator Location k_a	Sensor Location - k_s: 1	2	3	4	5	6	7	8
1	6.08	6.77	7.17	8.03	8.56	11.13	13.70	29.12
2	5.32	6.02	6.41	7.21	7.70	10.04	12.33	26.73
3	6.26	7.08	7.54	8.42	8.97	11.54	14.00	29.82
4	8.83	9.88	10.44	11.55	12.26	15.54	18.65	38.54
5	8.55	9.48	10.02	11.10	11.77	14.93	17.99	37.27
6	15.89	17.37	18.19	19.89	20.92	26.01	30.88	61.89
7	9.14	9.96	10.47	11.59	12.29	15.65	19.02	38.87
8	9.80	10.34	10.84	11.76	12.49	15.60	18.48	36.03

Table E.33 - Case 7 control costs for LQG design (n=1, ϕ= .461, Use $\delta V_X/a$ in costs).

Actuator Location k_a	Sensor Location - k_s: 1	2	3	4	5	6	7	8
1	5.05	5.48	5.64	5.94	6.15	7.11	8.04	16.42
2	2.44	2.66	2.73	2.86	2.96	3.37	3.78	7.88
3	3.02	3.28	3.38	3.55	3.67	4.22	4.78	9.88
4	5.95	6.43	6.62	6.98	7.24	8.40	9.59	19.28
5	8.97	7.52	7.75	8.20	8.51	9.91	11.33	22.86
6	27.23	29.22	30.10	31.99	33.24	39.18	45.26	88.66
7	18.55	19.96	20.59	21.87	22.71	26.73	30.75	60.91
8	351.92	376.05	387.88	413.35	430.18	512.55	596.04	1152.8

Table E.34 - Case 7 state costs for H_∞ design (n=1, ϕ= .461, Use $\delta V_X/a$ in costs).

Sensor Location - k_s

		1	2	3	4	5	6	7	8
	1		8.34	9.73	9.80	11.52	11.63	15.64	777.22
	2	7.16	9.33	9.30	9.58	11.43	11.43	15.95	761.09
Actuator	3	7.72	10.01	12.46	10.70	13.74	13.11	20.44	500.80
Location	4	9.12	11.35	14.16	20.37	21.44	19.52	35.92	417.71
k_a	5	9.13	11.67	14.36	20.41	27.78		47.17	186.64
	6					34.20	205.04		
	7	9.85	12.46	15.08	21.19	28.38	175.89		
	8	8.48	10.62	12.92	18.08	25.00	159.57		

Table E.35 - Case 7 control costs for H_∞ design (n=1, ϕ= .461, Use $\delta V_X/a$ in costs).

Sensor Location - k_s

		1	2	3	4	5	6	7	8
	1		5.06	6.40	5.38	6.03	6.21	8.17	1210.9
	2	2.10	2.38	2.89	2.44	2.75	2.83	3.75	556.10
Actuator	3	2.60	3.02	3.31	3.08	3.67	3.62	5.09	402.92
Location	4	7.67	5.97	6.57	8.14	8.76	8.19	13.52	432.06
k_a	5	7.66	7.67	8.09	9.92	11.69		18.17	189.24
	6					41.34	179.06		
	7	18.36	22.35	23.78	29.26	33.34	143.42		
	8	1241.4	1709.6	1419.3	1557.4	1146.9	3453.3		

Table E.36 - Case 8 LQG costs (n=1, ϕ= .461, Measure $\delta P/P$, V_j=0).

Actuator Location k_a	Sensor Location - k_s: 1	2	3	4	5	6	7	8
1	17.97	18.69	19.22	20.85	21.94	28.29	36.19	138.39
2	46.53	48.95	50.55	54.79	57.55	73.68	93.09	348.25
3	617.16	645.89	668.78	725.82	761.94	969.02	1217.8	4530.3
4	17.65	18.69	19.35	20.98	22.03	28.46	36.09	136.71
5	52.93	55.56	57.62	62.48	65.69	83.99	105.84	395.29
6	10.12	10.86	11.37	12.42	13.11	17.04	21.54	82.53
7	18.81	19.97	20.87	22.76	24.03	30.91	38.93	146.65
8	8.67	9.39	9.90	10.91	11.56	15.02	18.92	72.89

Table E.37 - Case 8 state costs for LQG design (n=1, ϕ= .461, Measure $\delta P/P$, V_j=0).

Actuator Location k_a	Sensor Location - k_s: 1	2	3	4	5	6	7	8
1	8.64	9.06	9.23	9.91	10.39	12.93	15.81	32.03
2	12.03	13.03	13.27	14.05	14.65	18.04	21.60	42.68
3	129.04	138.19	142.47	150.39	155.64	181.74	205.39	405.43
4	4.33	4.87	4.99	5.28	5.50	7.06	8.62	18.04
5	5.03	5064	5.85	6.01	6.27	7.51	8.54	17.64
6	2.62	3.10	3.31	3.59	3.84	5.04	6.10	12.69
7	3.05	3.55	3.83	4.17	4.50	5.80	6.91	13.55
8	3.46	4.07	4.39	4.85	5.20	6.63	7.86	15.91

Table E.38 - Case 8 control costs for LQG design (n=1, ϕ= .461, Measure $\delta P/P$, V_j=0).

Actuator Location k_a	Sensor Location - k_s: 1	2	3	4	5	6	7	8
1	6.67	7.13	7.31	7.74	8.02	9.36	10.77	21.75
2	28.81	30.69	31.54	33.54	34.91	41.59	48.55	94.19
3	422.10	448.13	464.94	493.55	514.16	616.43	723.10	1391.3
4	10.76	11.40	11.70	12.42	12.91	15.40	18.04	34.75
5	41.67	44.14	45.47	48.50	50.54	60.72	71.41	137.73
6	5.69	6.07	6.25	6.62	6.89	8.18	9.52	18.51
7	12.98	13.85	14.28	15.17	15.82	18.87	22.02	43.03
8	3.47	3.74	3.87	4.09	4.26	5.02	5.79	11.41

Table E.39 - Case 8 state costs for H_∞ design (n=1, ϕ= .461, Measure $\delta P/P$, V_j=0).

Sensor Location - k_s

Actuator Location k_a	1	2	3	4	5	6	7	8
1	34.37						13.55	1335.7
2	14.24	16.25						4175.2
3								
4	9.30	10.72	12.53	15.73				
5	9.82	11.96	13.58	19.09	21.45			
6	3.88	4.80	5.66	7.77	10.22	61.52		
7	3.87	4.87	5.74	7.88	10.33	63.27		
8	4.30	5.42	6.34	8.66	11.25	67.79		

Table E.40 - Case 8 control costs for H_∞ design (n=1, ϕ= .461, Measure $\delta P/P$, V_j=0).

Sensor Location - k_s

Actuator Location k_a	1	2	3	4	5	6	7	8
1	15.26						9.79	2359.0
2	32.64	37.46						13262
3								
4	18.92	21.11	24.21	28.73				
5	132.28	159.91	176.49	248.18	252.35			
6	6.64	7.64	8.45	10.41	12.41	53.65		
7	15.24	18.57	19.56	24.35	26.52	115.00		
8	3.52	4.12	4.44	5.40	6.23	26.76		

Table E.41 - Case 9 LQG costs (n=1, ϕ= .461, Measure $\delta P_t/P_t$).

Sensor Location - k_s

Actuator Location k_a	1	2	3	4	5	6	7	8
1	13.22	11.19	21.10	15.64	18.62	134.91	10.45	11.66
2	7.38	6.29	17.66	8.67	10.42	74.00	5.93	6.59
3	8.51	7.25	20.49	10.05	12.03	86.03	6.82	7.59
4	14.82	12.61	36.11	17.65	20.91	152.29	11.73	13.10
5	17.56	14.86	42.96	20.86	24.78	180.78	13.84	15.47
6	63.00	53.03	155.32	75.06	88.45	654.86	48.95	54.91
7	43.51	36.56	107.36	51.80	61.03	450.99	33.79	37.92
8	716.51	599.30	1774.6	854.00	997.61	7447.4	549.74	618.73

Table E.42 - Case 9 state costs for LQG design (n=1, ϕ= .461, Measure $\delta P_t/P_t$).

Sensor Location - k_s

Actuator Location k_a	1	2	3	4	5	6	7	8
1	2.19	2.11	4.47	2.56	3.21	8.22	2.14	2.21
2	1.89	1.79	3.69	2.14	2.70	6.56	1.84	1.89
3	1.70	1.65	3.40	1.99	2.48	6.40	1.69	1.74
4	1.55	1.59	3.08	1.89	2.35	6.74	1.62	1.63
5	2.10	2.07	4.41	2.54	3.17	9.09	2.11	2.15
6	3.86	3.90	8.63	4.93	6.04	20.76	3.87	3.86
7	3.89	3.67	8.39	4.62	5.72	15.45	3.69	3.78
8	5.99	4.87	15.02	7.37	8.72	30.30	4.60	4.29

Table E.43 - Case 9 control costs for LQG design (n=1, ϕ= .461, Measure $\delta P_t/P_t$).

Sensor Location - k_s

Actuator Location k_a	1	2	3	4	5	6	7	8
1	7.39	6.82	15.70	8.65	9.84	29.69	6.52	6.83
2	3.44	3.20	7.36	4.06	4.61	14.37	3.06	3.20
3	4.44	4.12	9.44	5.21	5.94	17.89	3.94	4.13
4	9.19	8.49	19.64	10.79	12.25	36.09	8.10	8.51
5	10.64	9.80	22.63	12.43	14.17	41.62	9.37	9.84
6	42.10	38.58	89.43	48.96	55.81	162.12	36.82	38.70
7	27.81	25.54	59.45	32.50	36.96	110.28	24.36	25.58
8	518.53	474.09	1109.8	604.41	687.58	2037.2	451.54	474.24

Table E.44 - Case 9 state costs for H_∞ design (n=1, ϕ= .461, Measure $\delta P_t/P_t$).

Sensor Location - k_s

		1	2	3	4	5	6	7	8
	1					16.43		2.22	2.11
	2	367.29	1196.4			5.52		1.95	1.84
Actuator	3	426.12	1546.6			3.14		1.91	1.78
Location	4	628.30	1694.5					2.00	1.83
k_a	5	614.34	2701.3					3.59	3.12
	6							3.69	4.42
	7	406.74	2477.1						
	8	74.10	320.31		7.38				

Table E.45 - Case 9 control costs for H_∞ design (n=1, ϕ= .461, Measure $\delta P_t/P_t$).

Sensor Location - k_s

		1	2	3	4	5	6	7	8
	1					54.64		6.97	6.50
	2	1747.4	5836.9			10.53		3.23	2.99
Actuator	3	2051.9	7712.0			7.73		4.55	4.15
Location	4	3465.8	1323.1					11.89	10.43
k_a	5	3559.5	16713					17.57	15.09
	6							36.72	50.82
	7	5610.3	32441						
	8	1.26e5	5.79e5		2038.6				

Table E.46 - Case 10 LQG costs (n=1, ϕ= .461, Measure $\delta V_X/a$).

Actuator Location k_a	Sensor Location - k_s:	1	2	3	4	5	6	7	8
	1	14.63	14.85	15.42	16.05	16.76	17.72	19.07	20.35
	2	10.08	10.25	10.70	11.18	11.71	12.40	13.38	14.31
	3	10.71	10.90	11.39	11.91	12.47	13.20	14.24	15.21
	4	15.37	15.64	16.30	17.00	17.78	18.79	20.22	21.58
	5	15.96	16.21	16.83	17.52	18.30	19.34	20.81	22.19
	6	44.68	45.30	46.77	48.45	50.43	53.14	57.02	60.71
	7	29.19	29.58	30.53	31.64	32.94	34.72	37.28	39.71
	8	415.97	421.78	432.97	446.73	463.08	487.41	522.03	555.49

Table E.47 - Case 10 state costs for LQG design (n=1, ϕ= .461, Measure $\delta V_X/a$).

Actuator Location k_a	Sensor Location - k_s:	1	2	3	4	5	6	7	8
	1	7.32	7.50	7.82	8.17	8.55	9.01	9.57	10.12
	2	6.00	6.17	6.48	6.80	7.12	7.51	7.98	8.45
	3	5.87	6.05	6.38	6.71	7.04	7.42	7.89	8.35
	4	6.84	7.07	7.43	7.82	8.19	8.64	9.18	9.70
	5	6.30	6.47	6.77	7.11	7.46	7.87	8.36	8.84
	6	10.26	10.51	10.83	11.28	11.81	12.39	13.10	13.78
	7	6.00	6.11	6.31	6.60	6.94	7.32	7.79	8.23
	8	8.42	8.43	7.36	7.16	7.38	7.72	8.01	8.35

Table E.48 - Case 10 control costs for LQG design (n=1, ϕ= .461, Measure $\delta V_X/a$).

Actuator Location k_a	Sensor Location - k_s:	1	2	3	4	5	6	7	8
	1	5.24	5.37	5.46	5.59	5.72	5.91	6.14	6.40
	2	2.51	2.57	2.62	2.68	2.74	2.83	2.93	3.05
	3	3.15	3.23	3.29	3.38	3.46	3.57	3.71	3.87
	4	6.31	6.44	6.58	6.76	6.93	7.18	7.49	7.83
	5	7.39	7.54	7.71	7.92	8.13	8.43	8.81	9.20
	6	29.44	30.00	30.61	31.40	32.26	33.50	35.08	36.72
	7	19.67	20.06	20.46	20.99	21.56	22.37	23.41	24.48
	8	369.48	376.57	382.61	391.49	401.76	417.05	437.15	457.82

Table E.49 - Case 10 state costs for H_∞ design (n=1, ϕ= .461, Measure $\delta V_X/a$).

Sensor Location - k_s

Actuator Location k_a	1	2	3	4	5	6	7	8
1		9.19	9.30	9.68	10.02	10.59	12.51	12.31
2	6.27	6.77	8.22	8.66	9.07	9.63	10.54	11.33
3	5.75	6.25	6.64	9.25	9.88	10.64	11.88	12.89
4	9.12	7.26	6.90	7.47	14.67	16.12	18.40	20.41
5	7.35	6.08	6.28	6.88	7.41	21.66	27.39	31.97
6					10.55	11.41		
7	5.33	5.69	6.01	6.42	6.74	7.16	7.70	
8	5.41	5.70	5.74	6.00	6.42	6.80	7.27	7.55

Table E.50 - Case 10 control costs for H_∞ design (n=1, ϕ= .461, Measure $\delta V_X/a$).

Sensor Location - k_s

Actuator Location k_a	1	2	3	4	5	6	7	8
1		5.36	5.29	5.36	5.41	5.61	6.91	6.42
2	2.11	2.17	2.41	2.45	2.49	2.59	2.92	3.00
3	2.62	2.72	2.82	3.32	3.45	3.63	4.07	4.23
4	8.30	6.27	5.79	5.96	8.83	9.51	10.52	11.47
5	9.12	6.90	6.80	7.12	7.63	14.43	17.45	19.93
6					27.29	30.21		
7	18.38	19.25	19.95	20.68	22.41	23.55	27.04	
8	122.97	126.16	129.23	121.31	1389.3	131.26	1773.0	1629.4

Appendix F

State-Space Approximation Code

This appendix presents the code which builds the state-space matrices.

```
%
% init
%
clear
format compact
load blade_data -ascii;

n  = -1                    % Harmonic
nmeas = 1                   % Stat. Pres, Density, Axial Vel, Circ Vel,
Tot. Pres
bb = [1 ; 1 ; 0 ; 0]       % Control Vector
xi = -.3
xe = .3
Vplen = .5;
mdotex = 1;
ka = 2
ks = 1
after_control = 0

[M,Nblades] = size(blade_data);
Nblades

ple = blade_data(1,:);
vle = blade_data(2,:);
ale = blade_data(3,:);
rle = blade_data(4,:);
tt1nd = blade_data(5,:);
pt1nd = blade_data(6,:);
ps1nd = blade_data(7,:);
ut1nd = blade_data(8,:);
ux1nd = blade_data(9,:);
u1nd = blade_data(10,:);
a1nd = blade_data(11,:);
mr1 = blade_data(12,:);
beta1m = blade_data(13,:);
omega = blade_data(14,:);

pte = blade_data(15,:);
vte = blade_data(16,:);
ate = blade_data(17,:);
rte = blade_data(18,:);
tt2nd = blade_data(19,:);
pt2nd = blade_data(20,:);
ps2nd = blade_data(21,:);
ut2nd = blade_data(22,:);
ux2nd = blade_data(23,:);
u2nd = blade_data(24,:);
a2nd = blade_data(25,:);
mr2 = blade_data(26,:);
beta2m = blade_data(27,:);

urownd = blade_data(28,:);
stagg = blade_data(29,:);
pdla = blade_data(30,:);
```

```
pdlm = blade_data(31,:);
pdaa = blade_data(32,:);
pdam = blade_data(33,:);
xle = blade_data(34,:);
xte = blade_data(35,:);
chord = blade_data(36,:);
taup = blade_data(37,1);
taud = blade_data(37,2);

xle(1,Nblades+1)=xte(1,Nblades);
delx = xle(1,2:Nblades+1)-xle(1,1:Nblades);

gam = 1.4;
gam1 = 0.2;

rnd=1;

taup = taup*chord.*cos(stagg)./uxlnd;
taud = taud*chord.*cos(stagg)./uxlnd;;
j = sqrt(-1);

tslndl = ttlnd(1)/(1+gam1*mr1(1)^2);
sphi = mr1(1)/max(urownd(1:2))*sqrt(tslndl);
```

```
%
%   This MATLAB routine sets up the state space matrices to
%   approximate the PDE solution of the high speed axial
%   compressor WITHOUT gaps.  It uses 2nd order Pade
%   approximations for the time delays in the blade passages.
%
%   options
%     n       - Harmonic
%     a_c     - Measure after control flag, otherwise before
%     nmeas   - Quantity to measure (1-Pres,2-Dens,3-Vx,4-Vt)
%     bb      - Control Vector
%     nPade   - Pade approximation order (2 or 3)
%
nPade = 2                    % Pade approximation order (2 or 3)
incomp = 0;                  % Incompressible approx (dtb=dtc=.0001)

if (nPade == 2)
   apde = 8;
   nPpm = 1;
elseif (nPade == 3)
   apde = 11;
   nPpm = -1;
end
nB = nPade;
nC = 2*nPade;
nE = 3*nPade;
nP = apde-1;
nV = apde;

if (incomp == 1)
   a1nd = 1000*a1nd;
   a2nd = 1000*a2nd;
end

gam = 1.4;
gam1 = 0.2;

Ak = zeros(apde,apde);
Bk = zeros(apde,5); Bkm = Bk; Bkp = Bk;
Ck = zeros(5,apde);
Dk = zeros(5,5); Dkm = Dk; Dkp = Dk;
Bku = zeros(apde,1);
Dku = zeros(5,1);
Cky = zeros(1,5);
Dtuu = zeros(Nblades+1,Nblades+1);
q11=0;

Kp = zeros(3,3);
Lp = zeros(3,1);
M = zeros(3,1);
Pp = zeros(1,4);
Dp = zeros(1,4);
N = zeros(3,1);
Pu = 0;
```

```
Du = 0;
Z1r = zeros(1,4);

for k=1:Nblades

   dtb = delx(k)/((ale(k)-vle(k))*cos(stagg(k)));
   ejtb = exp(-n*j*(urownd(k)*dtb-tan(stagg(k))*delx(k)));
   dtc = delx(k)/((vle(k)+ale(k))*cos(stagg(k)));
   ejtc = exp(-n*j*(urownd(k)*dtc+tan(stagg(k))*delx(k)));
   dte = delx(k)/(vle(k)*cos(stagg(k)));
   ejte = exp(-n*j*(urownd(k)*dte+tan(stagg(k))*delx(k)));

   K = Kp;
   L = Lp;
   Mm = M;
   P = Pp;
   D = Dp;
   Nm = N;
   Pum = Pu;
   Dum = Du;
   Z1rm = Z1r;
   k = k+1;
   if (k<Nblades+1)
      Interm_Matrices
   else
      Kn = zeros(3,3);
      Ln = zeros(3,1);
      Mmn = zeros(3,1);
      Pn = zeros(1,4);
      Dn = zeros(1,4);
      Nmn = zeros(3,1);
      Pumn = 0;
      Dumn = 0;
      Z1rmn = zeros(1,4);
   end
   k = k-1;
   Kp = Kn;
   Lp = Ln;
   M = Mmn;
   Pp = Pn;
   Dp = Dn;
   N = Nmn;
   Pu = Pumn;
   Du = Dumn;
   Z1r = Z1rmn;

   if (nPade == 2)
      Ak(1:2,1:2) = [0 1 ; -12/(dtb*dtb) -6/dtb];
      Ak(3:4,3:4) = [0 1 ; -12/(dtc*dtc) -6/dtc];
      Ak(5:6,5:6) = [0 1 ; -12/(dte*dte) -6/dte];
   elseif (nPade == 3)
      Ak(1:3,1:3) = [0 1 0 ;0 0 1 ; -120/dtb^3 -60/dtb^2 -12/dtb];
      Ak(4:6,4:6) = [0 1 0 ;0 0 1 ; -120/dtc^3 -60/dtc^2 -12/dtc];
      Ak(7:9,7:9) = [0 1 0 ;0 0 1 ; -120/dte^3 -60/dte^2 -12/dte];
```

```
end
Ak(nP,nP) = -(1/taup(k)+j*n*urownd(k));
Ak(nV,nV) = -(1/taud(k)+j*n*urownd(k));

Bkm(nC,2) = K(2,2);
Bkm(nC,3) = K(2,3);
Bkm(nC,5) = Mm(2);
Bkm(nE,2) = K(3,2);
Bkm(nE,3) = K(3,3);
Bkm(nE,5) = Mm(3);
Bkm(nP,2) = P(2)+P(1)*K(1,2);
Bkm(nP,3) = P(3)+P(1)*K(1,3);
Bkm(nP,5) = P(4)+P(1)*Mm(1);
Bkm(nV,2) = D(2)+D(1)*K(1,2);
Bkm(nV,3) = D(3)+D(1)*K(1,3);
Bkm(nV,5) = D(4)+D(1)*Mm(1);
Bk(nB,2) = Kp(1,2);
Bk(nB,3) = Kp(1,3);
Bk(nB,5) = M(1);
Bk(nC,1) = K(2,1);
Bk(nC,4) = L(2);
Bk(nE,1) = K(3,1);
Bk(nE,4) = L(3);
Bk(nP,1) = P(1)*K(1,1);
Bk(nP,4) = P(1)*L(1);
Bk(nV,1) = D(1)*K(1,1);
Bk(nV,4) = D(1)*L(1);
Bkp(nB,1) = Kp(1,1);
Bkp(nB,4) = Lp(1);
Bku(nB,1) = N(1);
Bku(nC,1) = Nm(2);
Bku(nE,1) = Nm(3);
Bku(nP,1) = Pum+P(1)*Nm(1);
Bku(nV,1) = Dum+D(1)*Nm(1);

if (nPade == 2)
   Ck(1,2) = -12/dtb*ejtb;
   Ck(2,4) = -12/dtc*ejtc;
   Ck(3,6) = -12/dte*ejte;
elseif (nPade == 3)
   Ck(1,1:3) = [240/dtb^3 0 24/dtb]*ejtb;
   Ck(2,4:6) = [240/dtc^3 0 24/dtc]*ejtc;
   Ck(3,7:9) = [240/dte^3 0 24/dte]*ejte;
end
Ck(4,nP) = 1;
Ck(5,nV) = 1;
Cky(1,1) =           Z1rm(1,1)*K(1,1);
Cky(1,2) = Z1r(1,2)+Z1r(1,1)*Kp(1,2);
Cky(1,3) = Z1r(1,3)+Z1r(1,1)*Kp(1,3);
Cky(1,4) =           Z1rm(1,1)*L(1);
Cky(1,5) = Z1r(1,4)+Z1r(1,1)*M(1);

E = diag([ejtb ejtc ejte 0 0]);
Dkm = [Bkm(nB,:) ; Bkm(nC,:) ; Bkm(nE,:) ; zeros(size(Bkm(1:2,:)))];
```

```
    Dkm = nPpm*E*Dkm;
    Dk  = [Bk(nB,:) ; Bk(nC,:) ; Bk(nE,:) ; zeros(size(Bk(1:2,:)))];
    Dk  = nPpm*E*Dk;
    Dkp = [Bkp(nB,:) ; Bkp(nC,:) ; Bkp(nE,:) ; zeros(size(Bkp(1:2,:)))];
    Dkp = nPpm*E*Dkp;
    Dku = [Bku(nB,1) ; Bku(nC,1); Bku(nE,1) ; 0 ; 0];
    Dku = nPpm*E*Dku;
    if (k<Nblades)
       Dtuu(k+1,k+1) = Z1r(1,1)*N(1);
       if (after_control == 1)
          Dtuu(k+1,k+1) = Dtuu(k+1,k+1)+q11;
       end
    end
%
%
%
    kk1 = (k-1)*apde+1;
    kk2 = kk1+apde-1;
    kk3 = (k-1)*5+1;
    kk4 = kk3+4;
%
%   A tilde
%
    At(kk1:kk2,kk1:kk2) = Ak;
%
%   B tilde
%
    Bt(kk1:kk2,kk3:kk4) = Bk;
    if(k>1)
       Bt(kk1:kk2,kk3-5:kk4-5) = Bkm;
    end
    if(k<Nblades)
       Bt(kk1:kk2,kk3+5:kk4+5) = Bkp;
    end
%
%   Bu tilde
%
    if(k<Nblades)
       Btu(kk1:kk2,k+1) = [zeros(nPade-1,1)
                           Bku(nB,1)
                           zeros(apde-nPade,1)];
    end
    if(k>1)
       Btu(kk1:kk2,k) =   [zeros(nPade*2-1,1)
                           Bku(nC,1)
                           zeros(nPade-1,1)
                           Bku(nE,1)
                           Bku(nP,1)
                           Bku(nV,1)];
    end
%
%   C tilde
%
    Ct(kk3:kk4,kk1:kk2) = Ck;
```

```
%
%  Cy tilde
%
   if(k<Nblades)
      Cty(k+1,kk3:kk4) = [0 Cky(1,2) Cky(1,3) 0 Cky(1,5)];
   end
   if(k>1)
      Cty(k,kk3:kk4) = [Cky(1,1) 0 0 Cky(1,4) 0];
   end
%
%  D tilde
%
   Dt(kk3:kk4,kk3:kk4) = eye(size(Dk))-Dk;
   if(k>1)
      Dt(kk3:kk4,kk3-5:kk4-5) = -Dkm;
   end
   if(k<Nblades)
      Dt(kk3:kk4,kk3+5:kk4+5) = -Dkp;
   end
%
%  Du tilde
%
   if(k<Nblades)
      Dtu(kk3:kk4,k+1) = [ Dku(1,1); 0; 0; 0; 0];
   end
   if(k>1)
      Dtu(kk3:kk4,k) = [ 0; Dku(2,1); Dku(3,1); 0; 0];
   end

end

End_Conditions


IDt = inv(Dt);
A = At + Bt*IDt*Ct;
B = Btu + Bt*IDt*Dtu;
C = Cty*IDt*Ct;
D = Dtuu+Cty*IDt*Dtu;
```

```
%
%  Interm_Matrices
%  Intermediate Matrices
%

  VinvV = eye(4);

%   BUILD T and J MATRIX

M2 = mr2(k-1);
qrmtr5 = 1/(1+gam1*M2*M2);
Gmm = diag([1 ; tt2nd(k-1)*qrmtr5 ; pt2nd(k-1)*qrmtr5 ; ...
          1/(1+tan(beta2m(k-1))^2)]);
Mx = ux2nd(k-1)/a2nd(k-1);
Mt = ut2nd(k-1)/a2nd(k-1);
T = [0 1 1/Mx 0 ; 1 -1 ((gam-1)*Mx) ((gam-1)*Mt) ; ...
     (1-.5*M2*M2) (.5*gam*M2*M2) (gam*Mx) (gam*Mt) ; ...
     0 0 Mt/Mx^2 -1/Mx];
T = Gmm*T;

J = [0 1 1/Mx 0 ; 1/(gam*Mx*Mx) 0  1/Mx 0 ; 0 1 0 2/Mt ; ...
     qrmtr5 -qrmtr5 (gam-1)*Mx*qrmtr5 (gam-1)*Mt*qrmtr5];
JI = inv(J);
JIbb = JI*bb;
vnm = [0 0 0 0];
if nmeas <= 4
   vnm(nmeas) = 1;
elseif nmeas == 5
   Mr = M2;
   vnm = [1-.5*Mr^2 .5*gam*Mr^2 gam*Mx gam*Mt]/(1+(gam-1)/2*Mr^2);
end
q11 = vnm*JIbb(:,1);

%   BUILD ML MATRIX

M1 = mr1(k);
qrmtr5 = 1/(1+gam1*M1*M1);
Gm = diag([1 ; tt1nd(k)*qrmtr5 ; pt1nd(k)*qrmtr5]);
Mx = ux1nd(k)/a1nd(k);
Mt = ut1nd(k)/a1nd(k);
ML = [0 1 1/Mx 0 ; 1 -1 ((gam-1)*Mx) ((gam-1)*Mt) ; ...
     (1-.5*M1*M1) (.5*gam*M1*M1) (gam*Mx) (gam*Mt)];
ML = Gm*ML*VinvV;

%   BUILD MP MATRIX

M1 = mr1(k);
Ut = ut1nd(k);
Ux = ux1nd(k);
Mx = Ux/a1nd(k);
Mt = Ut/a1nd(k);
qrmtr5 = 1/(1+(gam-1)/2*M1*M1);
pdlak = pdla(k)/(1+tan(beta1m(k))^2);
P31 = -omega(k)*(pt1nd(k)*qrmtr5*(1-.5*M1*M1)-ps1nd(k))+...
```

```
         (pt1nd(k)-ps1nd(k))*pdlm(k)*.5*M1;
 P32 = -omega(k)*pt1nd(k)*qrmtr5*.5*gam*M1*M1-...
         (pt1nd(k)-ps1nd(k))*pdlm(k)*.5*M1;
 P33 = -omega(k)*pt1nd(k)*qrmtr5*gam*Mx-...
         (pt1nd(k)-ps1nd(k))*(-
pdlak*Mt/(Mx*Mx)+pdlm(k)*ux1nd(k)/u1nd(k));
 P34 = -omega(k)*pt1nd(k)*qrmtr5*gam*Mt-...
         (pt1nd(k)-ps1nd(k))*(pdlak/Mx+pdlm(k)*ut1nd(k)/u1nd(k));
MP = [P31 P32 P33 P34]*VinvV;

%   BUILD MD MATRIX

pdaak = pdaa(k)/(1+tan(beta1m(k))^2);
MD = pdam(k)*[-.5*M1 .5*M1 ux1nd(k)/u1nd(k) ut1nd(k)/u1nd(k)]+...
     pdaak*[0 0 -Mt/(Mx*Mx) 1/Mx];
MD = -MD*VinvV;  % This is because MVC was defined to be negative
          % with repect to how I am using it.

%   BUILD MB2ip1 MATRIX

M1 = vte(k-1)/ate(k-1);
qrmtr5 = 1/(1+(gam-1)/2*M1*M1);
Gm = diag([1 ; tt2nd(k-1)*qrmtr5 ; gam*pte(k-1)*qrmtr5 ; 1]);
MB2ip1 = [(1-1/M1) (1+1/M1) 1 0 ; (gam-1)*(1-M1) (gam-1)*(1+M1) -1 0 ;
...
        (1-M1) (1+M1) M1*M1/2 0 ; 0 0 0 1];
MB2ip1 = Gm*MB2ip1;

%   BUILD MB2i MATRIX

M1 = vle(k)/ale(k);
qrmtr5 = 1/(1+(gam-1)/2*M1*M1);
Gm = diag([1 ; tt1nd(k)*qrmtr5 ; gam*ple(k)*qrmtr5]);
MB2i = [(1-1/M1) (1+1/M1) 1 ; (gam-1)*(1-M1) (gam-1)*(1+M1) -1 ; ...
        (1-M1) (1+M1) M1*M1/2];
MB2i = Gm*MB2i;

%   BUILD Kn, Ln, Mmn, Pn, Dn MATRICES

Z = inv(T)*MB2ip1;
U = inv(MB2i)*ML*Z;
V = eye(3);
V(:,1) = -U(:,1);
U(:,1) = [-1;0;0];
W = inv(V)*U;
X = inv(V)*inv(MB2i);
Y = X*ML*JIbb;
Kn = W(1:3,1:3);
Mmn = W(1:3,4);
Ln = X(1:3,3);
Pn =  1/taup(k)*MP*inv(T)*MB2ip1;
Dn =  1/taud(k)*MD*inv(T)*MB2ip1;

Nmn = Y;
```

```
Pumn = 1/taup(k)*MP*JIbb;
Dumn = 1/taud(k)*MD*JIbb;

Z1rmn = vnm*Z;
```

```
%
% End_Conditions
%
k=Nblades+1;
Exit_Matrices;
dtb = delx(Nblades)/((ale(Nblades)-vle(Nblades))*cos(stagg(Nblades)));
ejtb      =      exp(-n*j*(urownd(Nblades)*dtb-
tan(stagg(Nblades))*delx(Nblades)));

kk1 = Nblades*apde+1;
kk2 = kk1+3;
kk3 = Nblades*5+1;
kk4 = kk3;
At(kk1:kk2,kk1:kk2) =      [0 0 0 -pp(5);
                            1 0 0 -pp(4);
                            0 1 0 -pp(3);
                            0 0 1 -pp(2)];
Bt(kk1:kk2,kk3-5:kk3-1) = [0 z1(5) z2(5) 0 z3(5);
                            0 z1(4) z2(4) 0 z3(4);
                            0 z1(3) z2(3) 0 z3(3);
                            0 z1(2) z2(2) 0 z3(2)];
Btu(kk1:kk2,Nblades+1) =   [z4(5) ; z4(4) ; z4(3) ; z4(2)];
Ct(kk3,kk1:kk2) =          [0 0 0 1];
Dt(kk3,kk3) =              [1];                   %eye
Dt(kk3,kk3-5:kk3-1) =     -[0 z1(1) z2(1) 0 z3(1)];   %-Dk
Dtu(kk3,Nblades+1) =       [z4(1)];

Bt(kk1-apde+nB-1,kk3) =  [1];
Dt(kk3-5,kk3) = -[ejtb]*nPpm;                         %-Dk

if (after_control == 1)
   Dtuu(Nblades+1,Nblades+1) = q11;
end
Cty(1:Nblades,kk3) = zeros(Nblades,1);
vnm = [0 0 0 0];
if nmeas <= 4
   vnm(nmeas) = 1;
elseif nmeas == 5
   Mr = mr2(Nblades);
   Mx = ux2nd(Nblades)/a2nd(Nblades);
   Mt = ut2nd(Nblades)/a2nd(Nblades);
   vnm = [1-.5*Mr^2 .5*gam*Mr^2 gam*Mx gam*Mt]/(1+(gam-1)/2*Mr^2);
end
Cty(Nblades+1,kk3) = vnm*Y(:,1);
Cty(Nblades+1,kk3-4) = vnm*Y(:,2);
Cty(Nblades+1,kk3-3) = vnm*Y(:,3);
Cty(Nblades+1,kk3-1) = vnm*Y(:,4);



Inlet_Matrices;
dtc = delx(1)/((vle(1)+ale(1))*cos(stagg(1)));
ejtc = exp(-n*j*(urownd(1)*dtc+tan(stagg(1))*delx(1)));
```

```
dte = delx(1)/(vle(1)*cos(stagg(1)));
ejte = exp(-n*j*(urownd(1)*dte+tan(stagg(1))*delx(1)));

kk1 = Nblades*apde+5;
kk2 = kk1+3;
kk3 = Nblades*5+2;
kk4 = kk3+2;

At(kk1:kk2,kk1:kk2) =   [0      1      0      0;
                         0      0      1      0;
                         0      0      0      1;
                         -pp(5) -pp(4) -pp(3) -pp(2)];

Bt(kk2,1:5) =           [1 0 0 -w13 0];
Btu(kk2,1) =            [-s11];

Ct(kk3:kk3+1,kk1:kk2) = [z1(5) z1(4) z1(3) z1(2);
                         z2(5) z2(4) z2(3) z2(2)];
At(nP,kk1:kk2) =        [z3(5) z3(4) z3(3) z3(2)]/taup(1);
At(nV,kk1:kk2) =        [z4(5) z4(4) z4(3) z4(2)]/taud(1);
Ct(kk3+2,kk1:kk2) =     [z5(5) z5(4) z5(3) z5(2)];

Dt(kk3,kk3) =           [1];                                  %eye
Dt(kk3,1:5) =          -[z1(1) 0 0 (-z1(1)*w13+w23) 0];       %-Dk
Dtu(kk3,1) =            [           -z1(1)*s11+s21    ];

Dt(kk3+1,kk3+1) =       [1];                                  %eye
Dt(kk3+1,1:5) =        -[z2(1) 0 0 (-z2(1)*w13+w33) 0];       %-Dk
Dtu(kk3+1,1) =          [           -z2(1)*s11+s31    ];

Bt(nP,1:5) =            [z3(1) 0 0  -z3(1)*w13      0]/taup(1);
Btu(nP,1) =             [           -z3(1)*s11+rij1c ]/taup(1);
Bt(nV,1:5) =            [z4(1) 0 0  -z4(1)*w13      0]/taud(1);
Btu(nV,1) =             [           -z4(1)*s11+dij1c ]/taud(1);

Dt(kk3+2,kk3+2) =       [1];                                  %eye
Dt(kk3+2,1:5) =        -[z5(1) 0 0 (-z5(1)*w13     ) 0];      %-Dk
Dtu(kk3+2,1) =          [           -z5(1)*s11        ];

Bt(nC,kk3) =            [1];
Bt(nE,kk3+1) =          [1];
Bt(nE,kk3+2) =          [0];
Dt(2,kk3) =            -[ejtc]*nPpm;                     %-Dk
Dt(3,kk3+1) =          -[ejte]*nPpm;                     %-Dk

if (after_control == 1)
   Dtuu(1,1) = q11;
end
Cty(1,kk4) = 1;
```

```
%
% Exit_Matrices
%

%   BUILD T and J MATRIX

M2 = mr2(k-1);
qrmtr5 = 1/(1+(gam-1)/2*M2*M2);
Gmm = diag([1 ; tt2nd(k-1)*qrmtr5 ; pt2nd(k-1)*qrmtr5 ; ...
                1/(1+tan(beta2m(k-1))^2)]);
Mx = ux2nd(k-1)/a2nd(k-1);
Mt = ut2nd(k-1)/a2nd(k-1);
T = [0 1 1/Mx 0 ; 1 -1 ((gam-1)*Mx) ((gam-1)*Mt) ; ...
     (1-.5*M2*M2) (.5*gam*M2*M2) (gam*Mx) (gam*Mt) ; ...
     0 0 Mt/Mx^2 -1/Mx];
T = Gmm*T;

J = [0 1 1/Mx 0 ; 1/(gam*Mx*Mx) 0  1/Mx 0 ; 0 1 0 2/Mt ; ...
     qrmtr5 -qrmtr5 (gam-1)*Mx*qrmtr5 (gam-1)*Mt*qrmtr5];
JI = inv(J);
JIbb = JI*bb;
vnm = [0 0 0 0];
if nmeas <= 4
   vnm(nmeas) = 1;
elseif nmeas == 5
   Mr = M2;
   vnm = [1-.5*Mr^2 .5*gam*Mr^2 gam*Mx gam*Mt]/(1+(gam-1)/2*Mr^2);
end

q11 = vnm*JIbb(:,1);

%   BUILD MB2ip1 MATRIX

M1 = vte(k-1)/ate(k-1);
qrmtr5 = 1/(1+(gam-1)/2*M1*M1);
Gm = diag([1 ; tt2nd(k-1)*qrmtr5 ; gam*pte(k-1)*qrmtr5 ; 1]);
MB2ip1 = [(1-1/M1) (1+1/M1) 1 0 ; (gam-1)*(1-M1) (gam-1)*(1+M1) -1 0 ;
...
        (1-M1) (1+M1) M1*M1/2 0 ; 0 0 0 1];
MB2ip1 = Gm*MB2ip1;

%   BUILD extra MATRICES

Y = inv(T)*MB2ip1;
a = a2nd(k-1);
rnd = 1;

if (n ~= 0)
  t0 = 1-Mx*Mx;
  cm = [j 0 0 0];
  kk = xe/t0;
  gw = [0 0 gam 0;0 0 0 -gam*n*Mx];
    y = n*n*(t0-Mt*Mt);
    ry    = sqrt(y);
```

```
yp    = -2*n*Mt;
ypp   = -2;
z     = kk*ry;
zp    = .5*kk/ry*yp;
zpp   = -.25*kk/ry^3*yp^2+.5*kk/ry*ypp;
zppp  = .375*kk/ry^5*yp^3-.75*kk/ry^3*yp*ypp;
zpppp = -15/16*kk/ry^7*yp^4+9/4*kk/ry^5*yp^2*ypp-.75*kk/ry^3*ypp^2;
cz    = cosh(z);
sz    = sinh(z);
f     = sz;
fp    = cz*zp;
fpp   = sz*zp^2+cz*zpp;
fppp  = cz*zp^3+3*sz*zp*zpp+cz*zppp;
fpppp = sz*zp^4+6*cz*zp^2*zpp+3*sz*zpp^2+4*sz*zp*zppp+cz*zpppp;
g     = z*fp-f*zp;
gp    = z*fpp-f*zpp;
gpp   = z*fppp+zp*fpp-fp*zpp-f*zppp;
gppp  = z*fpppp+2*zp*fppp-2*fp*zppp-f*zpppp;
h     = z*gp-2*g*zp;
hp    = z*gpp-gp*zp-2*g*zpp;
hpp   = z*gppp-3*gp*zpp-2*g*zppp;
q     = z*hp-3*h*zp;
qp    = z*hpp-2*hp*zp-3*h*zpp;
r     = z*qp-4*q*zp;
coshw = [ 0,...

(cz*zp^4+6*sz*zp^2*zpp+3*cz*zpp^2+4*cz*zp*zppp+sz*zpppp)/24,...
          (sz*zp^3+3*cz*zp*zpp+sz*zppp)/6,...
          (cz*zp^2+sz*zpp)/2,...
           sz*zp,...
           cz                      ];
 sinhbryw = kk*[ r/z^5/24,...
                 q/z^4/6,...
                 h/z^3/2,...
                 g/z^2,...
                 f/z                ];

 convw = diag([0,(rnd/a)^4,j*(rnd/a)^3,-(rnd/a)^2,-j*(rnd/a),1]);
 gw(2,:) = gw(2,:)+n*Mt*gw(1,:);
 pp = -(cm*Y(:,1)*coshw+conv(gw*Y(:,1),sinhbryw))*convw;
 pp = pp(2:6);
 z1 = (cm*Y(:,2)*coshw+conv(gw*Y(:,2),sinhbryw))*convw;
 z1 = z1(2:6);
 z2 = (cm*Y(:,3)*coshw+conv(gw*Y(:,3),sinhbryw))*convw;
 z2 = z2(2:6);
 z3 = (cm*Y(:,4)*coshw+conv(gw*Y(:,4),sinhbryw))*convw;
 z3 = z3(2:6);
 z4 = (cm*JIbb*coshw+conv(gw*JIbb,sinhbryw))*convw;
 z4 = z4(2:6);

z1 = z1/pp(1);
z2 = z2/pp(1);
z3 = z3/pp(1);
z4 = z4/pp(1);
```

```
  pp = pp/pp(1);
  z1(2:5) = z1(2:5)-pp(2:5)*z1(1);
  z2(2:5) = z2(2:5)-pp(2:5)*z2(1);
  z3(2:5) = z3(2:5)-pp(2:5)*z3(1);
  z4(2:5) = z4(2:5)-pp(2:5)*z4(1);
elseif (n == 0)
  kk1 = Vplen*ps2nd(k-1)/(a2nd(k-1)^2*mdotex);
  kk2 = (gam+1)/(2*gam);
  Ts = [kk1 0 0 0];
  Tc = [kk2 -1 -1 -1];
  tau1 = abs(xe)/(a-ux1nd(1));
  tau2 = -abs(xe)/(a+ux1nd(1));
  tau3 = -abs(xe)/(ux1nd(1));
  et1 = [tau1^3/6 tau1^2/2 tau1 1];
  et2 = [tau2^3/6 tau2^2/2 tau2 1];
  et3 = [tau3^3/6 tau3^2/2 tau3 1];
  G = [gam gam 0 0;1 1 0 1;-1 1 0 0;0 0 1 0];
  TGs = Ts*G;
  TGc = Tc*G;
  vtemp(1,:) = TGs(1)*[et1 0]+TGc(1)*[0 et1];
  vtemp(2,:) = TGs(2)*[et2 0]+TGc(2)*[0 et2];
  vtemp(3,:) = TGs(3)*[et3 0]+TGc(3)*[0 et3];
  vtemp(4,:) = TGs(4)*[et3 0]+TGc(4)*[0 et3];
  vtemp = vtemp.';
  vGYr = vtemp*(G\[Y JIbb]);
  vtemp = vtemp.';
  pp = -vGYr(:,1).';
  z1 = vGYr(:,2).';
  z2 = vGYr(:,3).';
  z3 = vGYr(:,4).';
  z4 = vGYr(:,5).';
  z1 = z1/pp(1);
  z2 = z2/pp(1);
  z3 = z3/pp(1);
  z4 = z4/pp(1);
  pp = pp/pp(1);
  z1(2:5) = z1(2:5)-pp(2:5)*z1(1);
  z2(2:5) = z2(2:5)-pp(2:5)*z2(1);
  z3(2:5) = z3(2:5)-pp(2:5)*z3(1);
  z4(2:5) = z4(2:5)-pp(2:5)*z4(1);

end
```

```
%
% Inlet_Matrices
%

clear Q pp z1 z2 z3 z4 z5

%   BUILD ML MATRIX

M1 = mr1(1);
qrmtr5 = 1/(1+(gam-1)/2*M1*M1);
Gm = diag([1 ; tt1nd(1)*qrmtr5 ; pt1nd(1)*qrmtr5]);
Mx = ux1nd(1)/a1nd(1);
Mt = ut1nd(1)/a1nd(1);
ML = [0 1 1/Mx 0 ; 1 -1 ((gam-1)*Mx) ((gam-1)*Mt) ; ...
     (1-.5*M1*M1) (.5*gam*M1*M1) (gam*Mx) (gam*Mt)];
ML = Gm*ML;

J = [0 1 1/Mx 0 ; 1/(gam*Mx*Mx) 0  1/Mx 0 ; 0 1 0 2/Mt ; ...
     qrmtr5 -qrmtr5 (gam-1)*Mx*qrmtr5 (gam-1)*Mt*qrmtr5];
if (Mt == 0)
   J(3,4) = 10000.;
   disp('ok')
end
JI = inv(J);
JIbb = JI*bb;
vnm = [0 0 0 0];
if nmeas <= 4
   vnm(nmeas) = 1;
elseif nmeas == 5
   Mr = M1;
   vnm = [1-.5*Mr^2 .5*gam*Mr^2 gam*Mx gam*Mt]/(1+(gam-1)/2*Mr^2);
end
q11 = vnm*JIbb(:,1);

%   BUILD MP MATRIX

M1 = mr1(1);
Mx = ux1nd(1)/a1nd(1);
Mt = ut1nd(1)/a1nd(1);
qrmtr5 = 1/(1+(gam-1)/2*M1*M1);
pdlak = pdla(1)/(1+tan(beta1m(1))^2);
P31 = -omega(1)*(pt1nd(1)*qrmtr5*(1-.5*M1*M1)-ps1nd(1))+...
        (pt1nd(1)-ps1nd(1))*pdlm(1)*.5*M1;
P32 = -omega(1)*pt1nd(1)*qrmtr5*.5*gam*M1*M1-...
        (pt1nd(1)-ps1nd(1))*pdlm(1)*.5*M1;
P33 = -omega(1)*pt1nd(1)*qrmtr5*gam*Mx-...
        (pt1nd(1)-ps1nd(1))*(-
pdlak*Mt/(Mx*Mx)+pdlm(1)*ux1nd(1)/u1nd(1));
P34 = -omega(1)*pt1nd(1)*qrmtr5*gam*Mt-...
        (pt1nd(1)-ps1nd(1))*(pdlak/Mx+pdlm(1)*ut1nd(1)/u1nd(1));
MP = [P31 P32 P33 P34];

%   BUILD MD MATRIX
```

```
pdaak = pdaa(1)/(1+tan(betalm(1))^2);
MD = pdam(1)*[-.5*M1 .5*M1 uxlnd(1)/ulnd(1) utlnd(1)/ulnd(1)]+...
     pdaak*[0 0 -Mt/(Mx*Mx) 1/Mx];

%   BUILD MB2i MATRIX

M1 = vle(1)/ale(1);
qrmtr5 = 1/(1+(gam-1)/2*M1*M1);
Gm = diag([1 ; ttlnd(1)*qrmtr5 ; gam*ple(1)*qrmtr5]);
MB2i = [(1-1/M1) (1+1/M1) 1 ; (gam-1)*(1-M1) (gam-1)*(1+M1) -1 ; ...
         (1-M1) (1+M1) M1*M1/2];
MB2i = Gm*MB2i;

%   BUILD extra MATRICES

X = inv(MB2i)*ML;
W = inv(MB2i);
w13 = W(1,3);
w23 = W(2,3);
w33 = W(3,3);
S = X*JIbb;
s11 = S(1,1);
s21 = S(2,1);
s31 = S(3,1);

pij = MP*JIbb;
pij1c = pij(1,1);

dij = MD*JIbb;
dij1c = dij(1,1);

rnd = 1;
a = alnd(1);

%   BUILD coefficients

if (n ~= 0)

  t0 = 1-Mx*Mx;
  cm = -j/t0*[gam*rnd*Mx;rnd*Mx;-rnd;0];
  kk = -xi/t0;
  gw = [gam*(1+rnd*Mx*Mx/t0) 0
          (1+rnd*Mx*Mx/t0) 0
     -rnd*Mx/t0            0
     0                     -n];

  y = n*n*(t0-Mt*Mt);
  ry    = sqrt(y);
  yp    = -2*n*Mt;
  ypp   = -2;
  z     = kk*ry;
  zp    = .5*kk/ry*yp;
  zpp   = -.25*kk/ry^3*yp^2+.5*kk/ry*ypp;
  zppp  = .375*kk/ry^5*yp^3-.75*kk/ry^3*yp*ypp;
```

```
zpppp = -15/16*kk/ry^7*yp^4+9/4*kk/ry^5*yp^2*ypp-.75*kk/ry^3*ypp^2;
cz    = cosh(z);
sz    = sinh(z);
f     = sz;
fp    = cz*zp;
fpp   = sz*zp^2+cz*zpp;
fppp  = cz*zp^3+3*sz*zp*zpp+cz*zppp;
fpppp = sz*zp^4+6*cz*zp^2*zpp+3*sz*zpp^2+4*sz*zp*zppp+cz*zpppp;
g     = z*fp-f*zp;
gp    = z*fpp-f*zpp;
gpp   = z*fppp+zp*fpp-fp*zpp-f*zppp;
gppp  = z*fpppp+2*zp*fppp-2*fp*zppp-f*zpppp;
h     = z*gp-2*g*zp;
hp    = z*gpp-gp*zp-2*g*zpp;
hpp   = z*gppp-3*gp*zpp-2*g*zppp;
q     = z*hp-3*h*zp;
qp    = z*hpp-2*hp*zp-3*h*zpp;
r     = z*qp-4*q*zp;
coshw = [ 0,...

(cz*zp^4+6*sz*zp^2*zpp+3*cz*zpp^2+4*cz*zp*zppp+sz*zpppp)/24,...
          (sz*zp^3+3*cz*zp*zpp+sz*zppp)/6,...
          (cz*zp^2+sz*zpp)/2,...
           sz*zp,...
           cz                     ];
sinhbryw = kk*[ r/z^5/24,...
                q/z^4/6,...
                h/z^3/2,...
                g/z^2,...
                f/z                ];

convw = diag([0,(rnd/a)^4,j*(rnd/a)^3,-(rnd/a)^2,-j*(rnd/a),1]);
gw(:,2) = gw(:,2)+n*Mt*gw(:,1);
pp = (X(1,:)*cm*coshw+conv(X(1,:)*gw,sinhbryw))*convw;
pp = pp(2:6);
z1 = (X(2,:)*cm*coshw+conv(X(2,:)*gw,sinhbryw))*convw;
z1 = z1(2:6);
z2 = (X(3,:)*cm*coshw+conv(X(3,:)*gw,sinhbryw))*convw;
z2 = z2(2:6);
z3 = (MP*cm*coshw+conv(MP*gw,sinhbryw))*convw;
z3 = z3(2:6);
z4 = (MD*cm*coshw+conv(MD*gw,sinhbryw))*convw;
z4 = z4(2:6);
z5 = (vnm*cm*coshw+conv(vnm*gw,sinhbryw))*convw;
z5 = z5(2:6);
z1 = z1/pp(1);
z2 = z2/pp(1);
z3 = z3/pp(1);
z4 = z4/pp(1);
z5 = z5/pp(1);
pp = pp/pp(1);
z1(2:5) = z1(2:5)-pp(2:5)*z1(1);
z2(2:5) = z2(2:5)-pp(2:5)*z2(1);
z3(2:5) = z3(2:5)-pp(2:5)*z3(1);
```

```
  z4(2:5) = z4(2:5)-pp(2:5)*z4(1);
  z5(2:5) = z5(2:5)-pp(2:5)*z5(1);

elseif (n == 0)
  tau1 = abs(xi)/(a-ux1nd(1));
  tau2 = -abs(xi)/(a+ux1nd(1));
  et1 = [tau1^4/24 tau1^3/6 tau1^2/2 tau1 1];
  et2 = [tau2^4/24 tau2^3/6 tau2^2/2 tau2 1];
  tempvd = [gam;1;-1;0]/(1-Mx);
  tempvf = [gam;1; 1;0]/(1+Mx);
  d = [X;MP;MD]*[gam;1;-1;0]/(1-Mx);
  d(6) = vnm*tempvd;
  f = [X;MP;MD]*[gam;1; 1;0]/(1+Mx);
  f(6) = vnm*tempvf;
  pp = d(1)*et1-f(1)*et2;
  z1 = d(2)*et1-f(2)*et2;
  z2 = d(3)*et1-f(3)*et2;
  z3 = d(4)*et1-f(4)*et2;
  z4 = d(5)*et1-f(5)*et2;
  z5 = d(6)*et1-f(6)*et2;
  z1 = z1/pp(1);
  z2 = z2/pp(1);
  z3 = z3/pp(1);
  z4 = z4/pp(1);
  z5 = z5/pp(1);
  pp = pp/pp(1);
  z1(2:5) = z1(2:5)-pp(2:5)*z1(1);
  z2(2:5) = z2(2:5)-pp(2:5)*z2(1);
  z3(2:5) = z3(2:5)-pp(2:5)*z3(1);
  z4(2:5) = z4(2:5)-pp(2:5)*z4(1);
  z5(2:5) = z5(2:5)-pp(2:5)*z5(1);

end
```

```
%
%   This MATLAB routine sets up the state space matrices to
%   approximate the PDE solution of the high speed axial
%   compressor WITH gaps.  It uses 2nd order Pade
%   approximations for the time delays in the blade passages.
%
%   options
%     n       - Harmonic
%     a_c     - Measure after control flag, otherwise before
%     nmeas   - Quantity to measure (1-Pres,2-Dens,3-Vx,4-Vt)
%     bb      - Control Vector
%     nPade   - Pade approximation order (2 or 3)
%
nPade = 2                        % Pade approximation order (2 or 3)
incomp = 0;                      % Incompressible approx (dtb=dtc=.0001)

if (nPade == 2)
   apde = 8;
   nPpm = 1;
elseif (nPade == 3)
   apde = 11;
   nPpm = -1;
end
nB = nPade;
nC = 2*nPade;
nE = 3*nPade;
nP = apde-1;
nV = apde;
ng = nV+1;
nt = nV+5;
nf = 5;

if (incomp == 1)
   a1nd = 1000*a1nd;
   a2nd = 1000*a2nd;
end

gam = 1.4;
gam1 = 0.2;

Ak = zeros(nt,nt);
Bk = zeros(nt,10); Bkm = Bk; Bkp = Bk;
Ck = zeros(10,nt);
Dk = zeros(10,10); Dkm = Dk; Dkp = Dk;
Bku = zeros(nt,1);
Dku = zeros(10,1);
Cky = zeros(1,10);
Dtuu = zeros(Nblades+1,Nblades+1);
q11=0;

HB = zeros(5,5);
HD = HB;
u11c = 0;
BLi = zeros(3,3);
```

```
for k=1:Nblades

   dx = xte(k)-xle(k);
   dtb = dx/((ale(k)-vle(k))*cos(stagg(k)));
   ejtb = exp(-n*j*(urownd(k)*dtb-tan(stagg(k))*dx));
   dtc = dx/((vle(k)+ale(k))*cos(stagg(k)));
   ejtc = exp(-n*j*(urownd(k)*dtc+tan(stagg(k))*dx));
   dte = dx/(vle(k)*cos(stagg(k)));
   ejte = exp(-n*j*(urownd(k)*dte+tan(stagg(k))*dx));

   HBo = HB;
   HDo = HD;
   u11co = u11c;
   BLio = BLi;
   if (k<Nblades)
      Gap_Matrices
   end

   if (nPade == 2)
      Ak(1:2,1:2) = [0 1 ; -12/(dtb*dtb) -6/dtb];
      Ak(3:4,3:4) = [0 1 ; -12/(dtc*dtc) -6/dtc];
      Ak(5:6,5:6) = [0 1 ; -12/(dte*dte) -6/dte];
   elseif (nPade == 3)
      Ak(1:3,1:3) = [0 1 0 ;0 0 1 ; -120/dtb^3 -60/dtb^2 -12/dtb];
      Ak(4:6,4:6) = [0 1 0 ;0 0 1 ; -120/dtc^3 -60/dtc^2 -12/dtc];
      Ak(7:9,7:9) = [0 1 0 ;0 0 1 ; -120/dte^3 -60/dte^2 -12/dte];
   end
   Ak(nP,nP) = -(1/taup(k)+j*n*urownd(k));
   Ak(nV,nV) = -(1/taud(k)+j*n*urownd(k));

   Ak(ng:nt,ng:nt) = -diag([u11c u11co u11co u11co u11co]);

   Bk(nB,6)  = 1;
   Bk(nC,7)  = 1;
   Bk(nE,8)  = 1;
   Bk(nP,9)  = 1;
   Bk(nV,10) = 1;


   Bk(ng,1:5)      = [0 HB(1,2:3) 0 HB(1,4)];
   Bkp(ng,1:5)     = HB(1,1)*[1 0 0 -BLi(1,3) 0];
   Bku(ng,1)       = HB(1,5);
   Dk(6,1:5)       = [0 HD(1,2:3) 0 HD(1,4)];
   Dkp(6,1:5)      = HD(1,1)*[1 0 0 -BLi(1,3) 0];
   Dku(6,1)        = HD(1,5);

   Bk(ng+1,1:5)  = HBo(2,1)*[1 0 0 -BLio(1,3) 0];
   Bkm(ng+1,1:5) = [0 HBo(2,2:3) 0 HBo(2,4)];
   Bku(ng+1,1)   = HBo(2,5);
   Dk(7,1:5)     = [HDo(2,1) 0 0 -HDo(2,1)*BLio(1,3)+BLio(2,3) 0];
   Dkm(7,1:5)    = [0 HDo(2,2:3) 0 HDo(2,4)];
   Dku(7,1)       = HDo(2,5);
```

```
Bk(ng+2,1:5)   = HBo(3,1)*[1 0 0 -BLio(1,3) 0];
Bkm(ng+2,1:5) = [0 HBo(3,2:3) 0 HBo(3,4)];
Bku(ng+2,1)   = HBo(3,5);
Dk(8,1:5)     = [HDo(3,1) 0 0 -HDo(3,1)*BLio(1,3)+BLio(3,3) 0];
Dkm(8,1:5)    = [0 HDo(3,2:3) 0 HDo(3,4)];
Dku(8,1)      = HDo(3,5);

Bk(ng+3,1:5)  = HBo(4,1)*[1 0 0 -BLio(1,3) 0];
Bkm(ng+3,1:5) = [0 HBo(4,2:3) 0 HBo(4,4)];
Bku(ng+3,1)   = HBo(4,5);
Dk(9,1:5)     = HDo(4,1)*[1 0 0 -BLio(1,3) 0];
Dkm(9,1:5)    = [0 HDo(4,2:3) 0 HDo(4,4)];
Dku(9,1)      = HDo(4,5);

Bk(ng+4,1:5)  = HBo(5,1)*[1 0 0 -BLio(1,3) 0];
Bkm(ng+4,1:5) = [0 HBo(5,2:3) 0 HBo(5,4)];
Bku(ng+4,1)   = HBo(5,5);
Dk(10,1:5)    = HDo(5,1)*[1 0 0 -BLio(1,3) 0];
Dkm(10,1:5)   = [0 HDo(5,2:3) 0 HDo(5,4)];
Dku(10,1)     = HDo(5,5);

if (nPade == 2)
   Ck(1,2) = -12/dtb*ejtb;
   Ck(2,4) = -12/dtc*ejtc;
   Ck(3,6) = -12/dte*ejte;
elseif (nPade == 3)
   Ck(1,1:3) = [240/dtb^3 0 24/dtb]*ejtb;
   Ck(2,4:6) = [240/dtc^3 0 24/dtc]*ejtc;
   Ck(3,7:9) = [240/dte^3 0 24/dte]*ejte;
end
Ck(4,nP) = 1;
Ck(5,nV) = 1;
Ck(6:10,ng:nt) = eye(5);

Cky(1,1:10) = [0 Z1rmn(:,2:3) 0 Z1rmn(:,4) Z1rmn(:,1) 0 0 0 0];

E = diag([ejtb ejtc ejte 0 0]);
     Dkm(1:5,:)   =   [Bkm(nB,:)   ;   Bkm(nC,:)   ;   Bkm(nE,:)   ;
zeros(size(Bkm(1:2,:)))];
Dkm(1:5,:) = nPpm*E*Dkm(1:5,:);
     Dk(1:5,:)    =   [Bk(nB,:)   ;   Bk(nC,:)   ;   Bk(nE,:)   ;
zeros(size(Bk(1:2,:)))];
Dk(1:5,:)  = nPpm*E*Dk(1:5,:);
     Dkp(1:5,:)   =   [Bkp(nB,:)   ;   Bkp(nC,:)   ;   Bkp(nE,:)   ;
zeros(size(Bkp(1:2,:)))];
Dkp(1:5,:) = nPpm*E*Dkp(1:5,:);
Dku(1:5,1) = [0;0;0;0;0];
if (k<Nblades)
   if (after_control == 1)
      Dtuu(k+1,k+1) = q11;
   else
      Dtuu(k+1,k+1) = 0;
   end
end
```

```
%
%
%
   if (k==1)
      kk1 = 1;
      kk2 = apde+1;
      kk3 = 1;
      kk4 = 6;
   elseif (k==Nblades)
      kk1 = (Nblades-2)*nt+apde+2;
      kk2 = kk1+apde+3;
      kk3 = (Nblades-2)*10+7;
      kk4 = kk3+8;
   else
      kk1 = (k-2)*nt+apde+2;
      kk2 = kk1+nt-1;
      kk3 = (k-2)*10+7;
      kk4 = kk3+9;
   end
%
%  A tilde
%
   if (k==1)
      At(kk1:kk2,kk1:kk2) = Ak(1:apde+1,1:apde+1);
   elseif (k==Nblades)
      At(kk1:kk2,kk1:kk2) = Ak([1:apde,ng+1:nt],[1:apde,ng+1:nt]);
   else
      At(kk1:kk2,kk1:kk2) = Ak;
   end
%
%  B tilde
%
   if (k==1)
      Bt(kk1:kk2,kk3:kk4) = Bk(1:apde+1,1:6);
   elseif (k==Nblades)
      Bt(kk1:kk2,kk3:kk4) = Bk([1:apde,ng+1:nt],[1:5,7:10]);
   else
      Bt(kk1:kk2,kk3:kk4) = Bk;
   end
   if (k>1)
      if (k==2)
         Bt(kk1:kk2,kk3-6:kk3-1) = Bkm(1:kk2-kk1+1,1:6);
      elseif (k==Nblades)
         Bt(kk1:kk2,kk3-10:kk3-1) = Bkm([1:apde,ng+1:nt],:);
      else
         Bt(kk1:kk2,kk3-10:kk3-1) = Bkm;
      end
   end
   if (k<Nblades)
      if (k==1)
         Bt(kk1:kk2,kk4+1:kk4+10) = Bkp(1:apde+1,:);
      elseif (k==Nblades-1)
         Bt(kk1:kk2,kk4+1:kk4+9) = Bkp(:,[1:5,7:10]);
      else
```

```
        Bt(kk1:kk2,kk4+1:kk4+10) = Bkp;
      end
   end
%
%  Bu tilde
%
   if (k<Nblades)
      if (k==1)
         Btu(kk1:kk2,k+1) = [zeros(apde,1);Bku(ng,1)];
      else
         Btu(kk1:kk2,k+1) = [zeros(apde,1);Bku(ng,1);zeros(4,1)];
      end
   end
   if (k>1)
      if (k==Nblades)
         Btu(kk1:kk2,k) =   [zeros(apde,1);Bku(ng+1:nt,1)];
      else
         Btu(kk1:kk2,k) =   [zeros(apde,1);0;Bku(ng+1:nt,1)];
      end
   end
%
%  C tilde
%
   if (k==1)
      Ct(kk3:kk4,kk1:kk2) = Ck(1:6,1:apde+1);
   elseif (k==Nblades)
      Ct(kk3:kk4,kk1:kk2) = Ck([1:5,7:10],[1:apde,ng+1:nt]);
   else
      Ct(kk3:kk4,kk1:kk2) = Ck;
   end
%
%  Cy tilde
%
   if (k<Nblades)
      if (k==1)
         Cty(k+1,kk3:kk4) = Cky(1,1:6);
      else
         Cty(k+1,kk3:kk4) = Cky;
      end
   end
%
%  D tilde
%
   if (k==1)
      Dt(kk3:kk4,kk3:kk4) = eye(6)-Dk(1:6,1:6);
   elseif (k==Nblades)
      Dt(kk3:kk4,kk3:kk4) = eye(9)-Dk([1:5,7:10],[1:5,7:10]);
   else
      Dt(kk3:kk4,kk3:kk4) = eye(size(Dk))-Dk;
   end
   if(k>1)
      if (k==2)
         Dt(kk3:kk4,kk3-6:kk3-1)  = -Dkm(1:kk4-kk3+1,1:6);
      elseif (k==Nblades)
```

```
            Dt(kk3:kk4,kk3-10:kk3-1) = -Dkm([1:5,7:10],:);
         else
            Dt(kk3:kk4,kk3-10:kk3-1) = -Dkm;
         end
      end
      if (k<Nblades)
         if (k==1)
            Dt(kk3:kk4,kk4+1:kk4+10) = -Dkp(1:6,:);
         elseif (k==Nblades-1)
            Dt(kk3:kk4,kk4+1:kk4+9)  = -Dkp(:,[1:5,7:10]);
         else
            Dt(kk3:kk4,kk4+1:kk4+10) = -Dkp;
         end
      end
%
%  Du tilde
%
      if (k<Nblades)
         if (k==1)
            Dtu(kk3:kk4,k+1) = [0;0;0;0;0;Dku(6,1)];
         else
            Dtu(kk3:kk4,k+1) = [0;0;0;0;0;Dku(6,1);0;0;0;0];
         end
      end
      if (k>1)
         if (k==Nblades)
            Dtu(kk3:kk4,k) = [0;0;0;0;0;Dku(7:10,1)];
         else
            Dtu(kk3:kk4,k) = [0;0;0;0;0;0;Dku(7:10,1)];
         end
      end

end

End_Condgap

A = At + Bt/Dt*Ct;
B = Btu + Bt/Dt*Dtu;
C = Cty/Dt*Ct;
D = Dtuu+Cty/Dt*Dtu;
```

```
%
% Gap_Matrices
%

%   BUILD T and J MATRIX

M2 = mr2(k);
qrmtr5 = 1/(1+gam1*M2*M2);
Gmm = diag([1 ; tt2nd(k)*qrmtr5 ; pt2nd(k)*qrmtr5 ; ...
          1/(1+tan(beta2m(k))^2)]);
a = a2nd(k);
Mx = ux2nd(k)/a;
Mt = ut2nd(k)/a;
T = [0 1 1/Mx 0 ; 1 -1 ((gam-1)*Mx) ((gam-1)*Mt) ; ...
     (1-.5*M2*M2) (.5*gam*M2*M2) (gam*Mx) (gam*Mt) ; ...
     0 0 Mt/Mx^2 -1/Mx];
T = Gmm*T;

J = [0 1 1/Mx 0 ; 1/(gam*Mx*Mx) 0  1/Mx 0 ; 0 1 0 2/Mt ; ...
     qrmtr5 -qrmtr5 (gam-1)*Mx*qrmtr5 (gam-1)*Mt*qrmtr5];
JI = inv(J);
JIbb = JI*bb;
vnm = [0 0 0 0];
if nmeas <= 4
   vnm(nmeas) = 1;
elseif nmeas == 5
   Mr = M2;
   vnm = [1-.5*Mr^2 .5*gam*Mr^2 gam*Mx gam*Mt]/(1+(gam-1)/2*Mr^2);
end
q11 = vnm*JIbb(:,1);

%   BUILD Gap Matrices

if n~=0
  Mt = (ut2nd(k)+urownd(k))/a;
  dx = xle(k+1)-xte(k);
  kk = dx/(1-Mx*Mx);
  h0 = exp(j*n*Mt*kk*Mx);
  y0 = n*n*(1-Mx*Mx-Mt*Mt);
  ry0 = sqrt(y0);
  cy0 = cosh(kk*ry0);
  sy0 = sinh(kk*ry0);
  st0 = j*n*Mt;
  dy0 = 2*st0*rnd/a;

  eo0 = exp(-st0/Mx*kk);
  ec0 = exp(-j*n*Mt*dx/Mx);
  ectay = [ec0 -rnd/(Mx*a)*dx];
  htay = [h0 kk*Mx*rnd/a*h0];
  ctay = [cy0 (sy0/ry0*kk*j*n*Mt*rnd/a)];
  srtay = [sy0/ry0 j*n*Mt*rnd/a*(cy0/y0*kk-sy0/(y0*ry0))];
  vv11 = conv(htay,ctay);
  vv21 = 1/gam*(vv11(1:2)-ectay);
  ttem(1,1) = (n*n*Mx*cy0-st0*ry0*sy0-n*n*Mx*eo0)/(y0-n*n);
```

```
   ttem(1,2) = (-ttem(1,1)*dy0+n*n*Mx*sy0*.5*kk/ry0*dy0-rnd/a*ry0*sy0-...
              .5*st0/ry0*sy0*dy0-st0*cy0*.5*kk*dy0+...
              n*n*Mx*rnd*kk/(a*Mx)*eo0)/(y0-n*n);
   vv31 = conv(htay,ttem)/gam;
   ttem(1,1) = (-st0*cy0+Mx*ry0*sy0+st0*eo0)/(y0-n*n);
   ttem(1,2) = (-ttem(1,1)*dy0-rnd/a*cy0-st0*sy0*.5*kk/ry0*dy0+...
              .5*Mx/ry0*sy0*dy0+Mx*cy0*.5*kk*dy0+rnd/a*eo0-...
              rnd*kk/(Mx*a)*st0*eo0)/(y0-n*n);
   vv41 = conv(htay,ttem)*j*n/gam;
   vv22 = ectay;
   vv13 = -gam*conv(conv([j*n*Mt rnd/a],htay),srtay);
   vv23 = vv13/gam;
   ttem(1,1) = (st0*st0*cy0-n*n*Mx*st0*sy0/ry0-n*n*Mx*Mx*eo0)/(y0-n*n);
   ttem(1,2) = 0;
   vv33 = conv(htay,ttem);
   ttem(1,1) = (-Mx*cy0+st0*sy0/ry0+Mx*eo0)/(y0-n*n);
   ttem(1,2) = 0;
   vv43 = conv(htay,conv([j*n*Mt rnd/a],ttem))*j*n;
   vv14 = gam*j*n*Mx*conv(htay,srtay);
   vv24 = vv14/gam;
   ttem(1,1) = (-st0*cy0+n*n*Mx*sy0/ry0+st0*eo0)/(y0-n*n);
   ttem(1,2) = (-ttem(1,1)*dy0-rnd/a*cy0-st0*sy0*.5*kk/ry0*dy0+...
              n*n*Mx*cy0*.5*kk/y0*dy0-n*n*Mx*sy0*.5/(y0*ry0)*dy0+...
              rnd/a*eo0-st0*rnd*kk/(Mx*a)*eo0)/(y0-n*n);
   vv34 = conv(htay,ttem)*j*n*Mx;
   ttem(1,1) = (-n*n*Mx*Mx*cy0+n*n*Mx*st0*sy0/ry0+st0*st0*eo0)/(y0-n*n);
   ttem(1,2) = (-ttem(1,1)*dy0-n*n*Mx*Mx*sy0*.5*kk/ry0*dy0+...
              n*n*Mx*rnd/a*sy0/ry0+n*n*Mx*j*n*Mt*cy0*.5*kk/y0*dy0-...
              n*n*Mx*j*n*Mt*sy0*.5/(y0*ry0)*dy0+2*j*n*Mt*rnd/a*eo0+...
              n*n*Mt*Mt*rnd*kk/(a*Mx)*eo0)/(y0-n*n);
   vv44 = conv(htay,ttem);
else
   dx = xle(k+1)-xte(k);
   kk = dx/(1-Mx*Mx);
   ectay = [1 -rnd/(Mx*a)*dx];
   coshtay = [1 0];
   sinhtay = [0 kk*rnd/a];
   vv11 = coshtay;
   vv21 = coshtay/gam-ectay/gam;
   vv31 = -sinhtay/gam;
   vv41 = [0 0];
   vv22 = ectay;
   vv13 = -gam*sinhtay;
   vv23 = -sinhtay;
   vv33 = coshtay;
   vv43 = [0 0];
   vv14 = [0 0];
   vv24 = [0 0];
   vv34 = [0 0];
   vv44 = coshtay;
end
   Wc = [vv11(1)    0     vv13(1) vv14(1)
         vv21(1) vv22(1) vv23(1) vv24(1)
         vv31(1)    0     vv33(1) vv34(1)
```

```
        vv41(1)    0      vv43(1) vv44(1)];
  VVs = [vv11(2)   0      vv13(2) vv14(2)
        vv21(2) vv22(2) vv23(2) vv24(2)
        vv31(2)    0      vv33(2) vv34(2)
        vv41(2)    0      vv43(2) vv44(2)];


%   BUILD ML MATRIX

M1 = mr1(k+1);
qrmtr5 = 1/(1+gam1*M1*M1);
Gm = diag([1 ; tt1nd(k+1)*qrmtr5 ; pt1nd(k+1)*qrmtr5]);
Mx = ux1nd(k+1)/a1nd(k+1);
Mt = ut1nd(k+1)/a1nd(k+1);
L = [0 1 1/Mx 0 ; 1 -1 ((gam-1)*Mx) ((gam-1)*Mt) ; ...
     (1-.5*M1*M1) (.5*gam*M1*M1) (gam*Mx) (gam*Mt)];
L = Gm*L;

%   BUILD P3r MATRIX

M1 = mr1(k+1);
Ut = ut1nd(k+1);
Ux = ux1nd(k+1);
Mx = Ux/a1nd(k+1);
Mt = Ut/a1nd(k+1);
qrmtr5 = 1/(1+(gam-1)/2*M1*M1);
pdlak = pdla(k+1)/(1+tan(beta1m(k+1))^2);
P31 = -omega(k+1)*(pt1nd(k+1)*qrmtr5*(1-.5*M1*M1)-ps1nd(k+1))+...
         (pt1nd(k+1)-ps1nd(k+1))*pdlm(k+1)*.5*M1;
P32 = -omega(k+1)*pt1nd(k+1)*qrmtr5*.5*gam*M1*M1-...
         (pt1nd(k+1)-ps1nd(k+1))*pdlm(k+1)*.5*M1;
P33 = -omega(k+1)*pt1nd(k+1)*qrmtr5*gam*Mx-...
         (pt1nd(k+1)-ps1nd(k+1))*(-
pdlak*Mt/(Mx*Mx)+pdlm(k+1)*ux1nd(k+1)/u1nd(k+1));
P34 = -omega(k+1)*pt1nd(k+1)*qrmtr5*gam*Mt-...
         (pt1nd(k+1)-
ps1nd(k+1))*(pdlak/Mx+pdlm(k+1)*ut1nd(k+1)/u1nd(k+1));
P3r = [P31 P32 P33 P34]/taup(k+1);

%   BUILD Dr MATRIX

pdaak = pdaa(k+1)/(1+tan(beta1m(k+1))^2);
Dr   =   pdam(k+1)*[-.5*M1   .5*M1   ux1nd(k+1)/u1nd(k+1)
ut1nd(k+1)/u1nd(k+1)]+...
     pdaak*[0 0 -Mt/(Mx*Mx) 1/Mx];
Dr = -Dr/taud(k+1);  % This is because MVC was defined to be negative
          % with repect to how I am using it.

%   BUILD BT MATRIX

M1 = vte(k)/ate(k);
qrmtr5 = 1/(1+(gam-1)/2*M1*M1);
Gm = diag([1 ; tt2nd(k)*qrmtr5 ; gam*pte(k)*qrmtr5 ; 1]);
BT = [(1-1/M1) (1+1/M1) 1 0 ; (gam-1)*(1-M1) (gam-1)*(1+M1) -1 0 ; ...
```

```
        (1-M1) (1+M1) M1*M1/2 0 ; 0 0 0 1];
BT = Gm*BT;

%   BUILD BL MATRIX

M1 = vle(k+1)/ale(k+1);
qmtr5 = 1/(1+(gam-1)/2*M1*M1);
Gm = diag([1 ; ttlnd(k+1)*qmtr5 ; gam*ple(k+1)*qmtr5]);
BL = [(1-1/M1) (1+1/M1) 1 ; (gam-1)*(1-M1) (gam-1)*(1+M1) -1 ; ...
        (1-M1) (1+M1) M1*M1/2];
BL = Gm*BL;

%   BUILD Kn, Ln, Mmn, Pn, Dn MATRICES

  Uc = [BL\L;P3r;Dr]*VVc*[T\BT JIbb];
  Us = [BL\L;P3r;Dr]*VVs*[T\BT JIbb];
  BLi = inv(BL);
  Hnc = Uc(1,1)*eye(4);
  Hc1 = [-1 0 0 0 0;
        -Uc(2,1) Uc(1,1) 0 0 0
        -Uc(3,1) 0 Uc(1,1) 0 0
        -Uc(4,1) 0 0 Uc(1,1) 0
        -Uc(5,1) 0 0 0 Uc(1,1)];
  Hc2 = [[-1;0;0;0;0] Uc(:,2:5)];
  Hc = Hc1*Hc2;
  Hs1 = [0 0 0 0 0;
        -Us(2,1) Us(1,1) 0 0 0
        -Us(3,1) 0 Us(1,1) 0 0
        -Us(4,1) 0 0 Us(1,1) 0
        -Us(5,1) 0 0 0 Us(1,1)];
  Hs2 = [[0;0;0;0;0] Us(:,2:5)];
  Hs = Hc1*Hs2+Hs1*Hc2;
  u11c = Uc(1,1);
  u11s = Us(1,1);
  Hc = Hc/u11s;
  Hs = Hs/u11s;
  u11c = u11c/u11s;
  u11s = 1;
  HB = Hc-Hs*u11c;
  HD = Hs;
%
 constant_numerator=1
 HB = Hc;
 HD = zeros(size(Hs));
%
  Z = T\BT;
  Z1rmn = vnm*Z;
```

```
%
% End_Condgap
%
k=Nblades+1;
Exit_Matrices;
dx = xte(Nblades)-xle(Nblades);
dtb = dx/((ale(Nblades)-vle(Nblades))*cos(stagg(Nblades)));
ejtb = exp(-n*j*(urownd(Nblades)*dtb-tan(stagg(Nblades))*dx));

kk1 = Nblades*nt-5+1;
kk2 = kk1+3;
kk3 = Nblades*10-5+1;
kk4 = kk3;
At(kk1:kk2,kk1:kk2) =      [0 0 0 -pp(5);
                             1 0 0 -pp(4);
                             0 1 0 -pp(3);
                             0 0 1 -pp(2)];
Bt(kk1:kk2,kk3-9:kk3-5) = [0 z1(5) z2(5) 0 z3(5);
                             0 z1(4) z2(4) 0 z3(4);
                             0 z1(3) z2(3) 0 z3(3);
                             0 z1(2) z2(2) 0 z3(2)];
Btu(kk1:kk2,Nblades+1) =   [z4(5) ; z4(4) ; z4(3) ; z4(2)];
Ct(kk3,kk1:kk2) =          [0 0 0 1];
Dt(kk3,kk3) =              [1];                          %eye
Dt(kk3,kk3-9:kk3-5) =     -[0 z1(1) z2(1) 0 z3(1)];     %-Dk
Dtu(kk3,Nblades+1) =       [z4(1)];

Bt(kk1-apde-4+nB-1,kk3) =  [1];
Dt(kk3-9,kk3) = -[ejtb]*nPpm;                          %-Dk

if (after_control == 1)
   Dtuu(Nblades+1,Nblades+1) = q11;
end
Cty(1:Nblades,kk3) = zeros(Nblades,1);
vnm = [0 0 0 0];
if nmeas <= 4
   vnm(nmeas) = 1;
elseif nmeas == 5
   Mr = mr2(Nblades);
   Mx = ux2nd(Nblades)/a2nd(Nblades);
   Mt = ut2nd(Nblades)/a2nd(Nblades);
   vnm = [1-.5*Mr^2 .5*gam*Mr^2 gam*Mx gam*Mt]/(1+(gam-1)/2*Mr^2);
end
Cty(Nblades+1,kk3) = vnm*Y(:,1);
Cty(Nblades+1,kk3-8) = vnm*Y(:,2);
Cty(Nblades+1,kk3-7) = vnm*Y(:,3);
Cty(Nblades+1,kk3-5) = vnm*Y(:,4);



Inlet_Matrices;
dx = xte(1)-xle(1);
dtc = dx/((vle(1)+ale(1))*cos(stagg(1)));
```

```
ejtc = exp(-n*j*(urownd(1)*dtc+tan(stagg(1))*dx));
dte = dx/(vle(1)*cos(stagg(1)));
ejte = exp(-n*j*(urownd(1)*dte+tan(stagg(1))*dx));

kk1 = Nblades*nt;
kk2 = kk1+3;
kk3 = Nblades*10-3;
kk4 = kk3+2;

At(kk1:kk2,kk1:kk2) =   [0      1      0      0;
                         0      0      1      0;
                         0      0      0      1;
                         -pp(5) -pp(4) -pp(3) -pp(2)];

Bt(kk2,1:5) =           [1 0 0 -w13 0];
Btu(kk2,1) =            [-s11];

Ct(kk3:kk3+1,kk1:kk2) = [z1(5) z1(4) z1(3) z1(2);
                         z2(5) z2(4) z2(3) z2(2)];
At(nP,kk1:kk2) =        [z3(5) z3(4) z3(3) z3(2)]/taup(1);
At(nV,kk1:kk2) =        [z4(5) z4(4) z4(3) z4(2)]/taud(1);
Ct(kk3+2,kk1:kk2) =     [z5(5) z5(4) z5(3) z5(2)];

Dt(kk3,kk3) =           [1];                                   %eye
Dt(kk3,1:5) =          -[z1(1) 0 0 (-z1(1)*w13+w23) 0];        %-Dk
Dtu(kk3,1) =            [           -z1(1)*s11+s21    ];

Dt(kk3+1,kk3+1) =       [1];                                   %eye
Dt(kk3+1,1:5) =        -[z2(1) 0 0 (-z2(1)*w13+w33) 0];        %-Dk
Dtu(kk3+1,1) =          [           -z2(1)*s11+s31    ];

Bt(nP,1:5) =            [z3(1) 0 0  -z3(1)*w13      0]/taup(1);
Btu(nP,1) =             [           -z3(1)*s11+pij1c ]/taup(1);
Bt(nV,1:5) =            [z4(1) 0 0  -z4(1)*w13      0]/taud(1);
Btu(nV,1) =             [           -z4(1)*s11+dij1c ]/taud(1);

Dt(kk3+2,kk3+2) =       [1];                                   %eye
Dt(kk3+2,1:5) =        -[z5(1) 0 0 (-z5(1)*w13    ) 0];        %-Dk
Dtu(kk3+2,1) =          [           -z5(1)*s11        ];

Bt(nC,kk3) =            [1];
Bt(nE,kk3+1) =          [1];
Bt(nE,kk3+2) =          [0];
Dt(2,kk3) =            -[ejtc]*nPpm;                   %-Dk
Dt(3,kk3+1) =          -[ejte]*nPpm;                   %-Dk

if (after_control == 1)
   Dtuu(1,1) = q11;
end
Cty(1,kk4) = 1;
```