ADAPTIVE DISTURBANCE REJECTION AND STABILIZATION FOR

ROTOR SYSTEMS WITH INTERNAL DAMPING

Except where reference is made to the work of others, the work described in this dissertation is my own or was done in collaboration with my advisory committee. This dissertation does not include proprietary or classified information.

_____

András Simon

Certificate of Approval:

_____

David G. Beale
Professor
Mechanical Engineering

_____

George T. Flowers, Chair
Professor
Mechanical Engineering

_____

Dan B. Marghitu
Professor
Mechanical Engineering

_____

John Y. Hung
Professor
Electrical and Computer Engineering

_____

Mark J. Balas
Professor
Electrical and Computer Engineering
University of Wyoming

_____

George T. Flowers
Dean
Graduate School

ADAPTIVE DISTURBANCE REJECTION AND STABILIZATION FOR

ROTOR SYSTEMS WITH INTERNAL DAMPING

András Simon

A Dissertation

Submitted to

the Graduate Faculty of

Auburn University

in Partial Fulfillment of the

Requirements for the

Degree of

Doctor of Philosophy

Auburn, Alabama
May 9, 2009

ADAPTIVE DISTURBANCE REJECTION AND STABILIZATION FOR

ROTOR SYSTEMS WITH INTERNAL DAMPING

András Simon

_____

Signature of Author

_____

Date of Graduation

András Simon, son of Károly Simon and Éva Titz, was born on November 19, 1974 in Budapest, Hungary. He entered the Budapest University of Technology and Economics (formerly known as Technical University of Budapest) in September 1994 where he graduated with a Master of Science degree in Mechanical Engineering in 1999. After working as a Project Engineer at General Electric Lighting TUNGSRAM in Budapest, he joined the graduate program of Auburn University in 2000 to continue the research program on active magnetic bearings and rotordynamics. He received his Master of Science degree in Mechanical Engineering in 2003 and continued the research for the Ph.D. degree from early 2005. His research interests are nonlinear controls, mechatronics, rotordynamics and active magnetic bearings.

Dissertation Abstract

ADAPTIVE DISTURBANCE REJECTION AND STABILIZATION FOR

ROTOR SYSTEMS WITH INTERNAL DAMPING

András Simon

Doctor of Philosophy, May 9, 2009
(M.Sc., Auburn University, 2003)
(M.Sc., Technical University of Budapest, 1999)

180 Typed Pages

Directed by George T. Flowers

Advanced rotor systems today consist of a lightweight rotor supported by radial active magnetic bearings (AMB). These systems are widely used in flywheel applications and in other fields where high rotational speeds are essential. Flywheels are most often made of composite materials to ensure low weight and sufficient structural strength. It has been shown in previous works that composite materials have high energy dissipation characteristics, mainly due to internal damping. In applications where the rotor speed is subcritical, this property is of low concern, whereas at supercritical speeds the effect of internal damping should not be ignored due to instability effects described in detail later. Therefore, it is essential that one must have a detailed understanding of the sources and effects of internal damping in these structures. This work addresses the application of Adaptive Disturbance Rejection (ADR) control method to deal with the rotordynamic instability caused by internal damping and synchronous vibrations caused by mass imbalance in rotor systems operating at supercritical speeds. The three modeled systems are 1) a simple Jeffcott-rotor, 2) a rigid shaft with flexible hub and 3) a slim, flexible shaft. A detailed description of

the problems and the strategies for addressing them are discussed. A fixed-gain controller was also developed for each model to better compare the results to the ones from the adaptive controllers. Simulation modeling and analysis results are presented and discussed to illustrate the method and demonstrate its effectiveness.

Style manual or journal used <u>Journal of Approximation Theory (together with the style</u>
<u>known as "auphd"). Bibliograpy follows van Leunen's *A Handbook for Scholars.*</u>

Computer software used <u>The document preparation package TEX (specifically LATEX)</u>
<u>together with the departmental style-file `auphd.sty`.</u>

Table of Contents

LIST OF FIGURES

A_g	Electromagnet face area, general AMB model	$[m^2]$
$A_{g,1}$	Electromagnet face area on right side AMB	$[\mathrm{m}^2]$
$A_{g,2}$	Electromagnet face area on left side AMB	$[\mathrm{m}^2]$
$\mathbf{A}$	State matrix of model without internal damping	
$\mathbf{A_C}$	State matrix of model with internal damping	
$\mathbf{A_p}$	Plant state matrix	
$\mathbf{A_m}$	Model state matrix	
$\mathbf{B}$	Input matrix	
$\mathbf{B_p}$	Plant input matrix	
$\mathbf{B_m}$	Model input matrix	
$\mathbf{C}$	Output matrix	
$\mathbf{C_p}$	Plant output matrix	
$\mathbf{C_I}$	Internal damping matrix	
$\mathbf{C_m}$	Model output matrix	
$\mathbf{C_x}$	Output matrix for ADR	
$\mathbf{D}$	Damping matrix of model without internal damping	
$\mathbf{D_C}$	Damping matrix of model with internal damping	
c	Damping in Jeffcott rotor	[Ns/m]
c_c	Critical damping coefficient	[Ns/m]
d	Airgap in general AMB model	[m]
d_1	Radial clearance at right side AMB	[m]

d_2	Radial clearance at left side AMB	[m]
$\mathbf{e_y}$	Error vector	
$\mathbf{F_x}$	Magnetic bearing force vector in x direction	[N]
$\mathbf{F_{x,imb}}$	Imbalance force vector in x direction	[N]
$\mathbf{F_y}$	Magnetic bearing force vector in y direction	[N]
$\mathbf{F_{y,imb}}$	Imbalance force vector in y direction	[N]
$\mathbf{G_p}$	Adaptive stabilizing gain	
$\mathbf{H_p}$	Adaptive disturbance rejection gain	
h_1	Power amplifier bandwidth constant	
h_2	Power amplifier bandwidth constant	
$\mathbf{I_{nx}}$	Rotor polar mass moment of inertia matrix	
$\mathbf{I_{1*}^B}$	Bias currents at right side AMB	[A]
$\mathbf{I_{2*}^B}$	Bias currents at left side AMB	[A]
i_c	Control current in general AMB force equation	[A]
I_B	Bias current in nonlinear AMB force equation	[A]
i_{1*}	Control currents at right side AMB	[A]
i_{2*}	Control currents at left side AMB	[A]
k	Jeffcott-rotor shaft stiffness	[N/m]
$k_{A,1}$	Amplifier voltage-to-current gain at right side AMB	[V/A]
$k_{A,2}$	Amplifier voltage-to-current gain at left side AMB	[V/A]
$\mathbf{K}$	Stiffness matrix	
$\mathbf{K_A}$	Amplifier bandwidth matrix	
$\mathbf{K_B}$	Bias current matrix	

K_x	Position stiffness in linear AMB force equation	[N/m]
K_i	Current stiffness in linear AMB force equation	[N/A]
$\mathbf{K_C}$	Stiffness matrix	
$\mathbf{K_I}$	Current stiffness matrix	
$\mathbf{K_F}$	State feedback gain matrix	
$\mathbf{M}$	Mass matrix	
m	Mass of disk for Jeffcott rotor	[kg]
m_R	Mass of rim for flexible hub model	[kg]
m_S	Mass of shaft for flexible hub model	[kg]
N_T	Number of turns in general AMB model	
$N_{T,1}$	Number of turns on electromagnets at right side AMB	
$N_{T,2}$	Number of turns on electromagnets at left side AMB	
$nb1$	Node number at right side magnetic bearing location	
$nb2$	Node number at left side magnetic bearing location	
$np1$	Node number at right side position sensor location	
$np2$	Node number at left side position sensor location	
$\mathbf{q}$	State vector	
$\mathbf{q_x}$	Rotor modal coordinate vector in x direction	
$\mathbf{q_{x,i}}$	Rotor modal x coordinate for the i-th mode shape	
$\mathbf{q_y}$	Rotor modal coordinate vector in y direction	
$\mathbf{q_{y,i}}$	Rotor modal y coordinate for the i-th mode shape	
$\mathbf{S_{12}}$	Adaptive gain for $\mathbf{x_m}$	
$\mathbf{S_{22}}$	Adaptive gain for $\mathbf{u_m}$	

v_{1*}	Control voltages at right side AMB	[V]
v_{2*}	Control voltages at left side AMB	[V]
$\mathbf{u}$	State-space model input vector	
u	Static imbalance	[kgm]
$\mathbf{u_d}$	Disturbance function	
$\mathbf{u_m}$	Model input for ADR	
$\mathbf{u_p}$	Plant input for ADR	
x	Displacement for general AMB model	[m]
x_{p1}	Rotor physical coordinate in x direction at right side position sensor	[m]
x_{p2}	Rotor physical coordinate in x direction at left side position sensor	[m]
x_{b1}	Rotor physical coordinate in x direction at right side AMB	[m]
x_{b2}	Rotor physical coordinate in x direction at left side AMB	[m]
$\mathbf{x_m}$	Model state vector	
$\mathbf{x_p}$	Plant state vector	
y_{p1}	Rotor physical coordinate in y direction at right side position sensor	[m]
y_{p2}	Rotor physical coordinate in y direction at left side position sensor	[m]
y_{b1}	Rotor physical coordinate in y direction at right side AMB	[m]
y_{b2}	Rotor physical coordinate in y direction at left side AMB	[m]
$\mathbf{y}$	State-space model output	
$\mathbf{y_m}$	Model output	
$\mathbf{y_p}$	Plant output	
$\mathbf{\Delta_1}$	Gain for $\mathbf{S_{12}}$	
$\mathbf{\Delta_2}$	Gain $\mathbf{S_{22}}$	

$\mathbf{\Delta_G}$	Gain for $\mathbf{G_p}$	
$\mathbf{\Delta_H}$	Gain for $\mathbf{H_p}$	
$\mathbf{\Phi}$	Imbalance vector	[kgm]
$\mathbf{\Psi}$	Rotor free-free modal rotation matrix	
$\mathbf{\Gamma}$	Gyroscopic matrix	
$\mathbf{\Gamma_p}$	Disturbance mapping matrix	
τ_1	Right side magnetic bearing amplifiers' time constant	[s]
τ_2	Left side magnetic bearing amplifiers' time constant	[s]
Ω	Rotor speed	[rad/s]
$\omega_\mathbf{n}$	Critical speed vector	[rad/s]
ω_n	Undamped natural frequency	[rad/s]
ω_d	Damped natural frequency	[rad/s]
ϕ	Rotor free-free modal displacement vector	
$\psi_\mathbf{d}$	Disturbance vector	
μ_0	Permeability of air	[H/m]
ζ	Internal damping coefficient, flexible shaft model	

CHAPTER 1

INTRODUCTION

1.1 Background

High-speed machinery is used in a wide variety of applications, ranging from turbomachines in power generation settings to industrial tools. Uncontrolled and undesired vibrations in these systems can lead to catastrophic failures meaning extra costs due to downtime and repair. Of particular interest to this work are the rotor systems engineered using composite materials. Composites are often used in applications where high strength and low weight must be combined in order to reach a certain engineering goal. Composite flywheels represent the common area shared by high-speed machines and composite materials. Advantages of composites also bring certain other properties which are of low concern in other applications using composite structures but pose major questions in high-speed applications. One of these properties is internal (material) damping. Internal damping in composites stems from several sources, the main one being the interaction between the fibers and matrix. While in subcritical speeds the internal damping helps reduce the vibrations, in supercritical speed it causes rotordynamic instability, i.e. a subsynchronous vibration which grows unbounded over time. Since this instability can occur in an otherwise perfectly balanced rotor, it is difficult to design a static control law to suppress it. Instead, some sort of adaptive method has to be considered which would help stabilize the system.

1.2 Motivation for research

Rotordynamic instability due to internal damping is a more and more pressing problem in high-speed applications when combined with shafts made of composite materials. With traditional (metallic) materials internal damping is of low concern but as engineering solutions move toward stronger and lighter materials, composites represent a whole new area in rotordynamics due to their unique characteristics. This research aims to show how Adaptive Disturbance Rejection method can suppress the rotordynamic instability as well as the synchronous vibrations resulting from static mass imbalance.

1.3 Organization of Dissertation

The research goal is to study and show the effectiveness of Adaptive Disturbance Rejection for reducing of rotordynamic instability as well as its ability to suppress synchronous vibrations resulting from mass imbalance. Specifically, the work includes:

- Development of mathematical models

- Development of simulation models

- Simulation results and discussion

The dissertation is divided into seven major parts; Chapter 1 introduces the reader to the argument which the research is based as well as describes the motivation for present research. Chapter 2 gives details about the source, effect and nature of internal damping in general as well as how it applies specifically to this work. Chapter 3 describes the active magnetic bearings used in this work and includes background on Active Magnetic Bearings.

Chapter 4 introduces the adaptive control method used in this work as well as contains detailed development steps on how the controllers were constructed. Chapter 5 shows the development of the three simulation models in great detail; the Jeffcott-rotor model with internal damping, the rigid rotor model with a flexible hub and the slim, flexible shaft model. Chapter 6 discusses the results and findings for each simulated model considered. Chapter 7 concludes the dissertation by giving an overview of the results, findings and the conclusions drawn.

CHAPTER 2

INTERNAL DAMPING

Damping is present in almost every engineering system in some form. It can be classi-
fied into three major groups; Coulomb, viscous, or hysteretic damping, depending on vari-
ous sources. Each describes different method of dissipation of vibration energy. Coulomb
damping is experienced as kinetic friction between two dry surfaces. Hysteretic damping
is caused by internal friction or hysteresis during deformation of solids. Viscous damping,
the most common form of damping, originates from fluid dynamics associated with flow
although it is widely used to describe damping which is proportional to velocity. In general,
having damping present in a system is an advantage as it lowers the vibration amplitude.
On the other hand, in some scenarios damping is a disadvantage, such as damping forces
in moving parts or frictional forces between moving parts which can generate self-excited
vibrations. Consider a simple mass-damper-spring system with no external excitation as
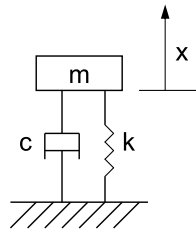
Figure 2.1: Simple mass-damper-spring system

shown on Figure 2.1, with the effects of gravity ignored. The equation of motion is:

$$m\ddot{x} + c\dot{x} + kx = 0 \tag{2.1}$$

4

The system response from an arbitrary initial position x_0 is described by Equation 2.2:

$$x(t) = x_0 e^{-\zeta \omega_n t} \cos(\omega_d t + \alpha) \tag{2.2}$$

where ζ is the damping ratio, expressed as $\zeta = \frac{c}{c_c}$ where c is the actual damping in the system and c_c is the critical damping coefficient, defined as $c_c = 2\sqrt{km}$ for the system described by Equation 2.1. The undamped and damped natural frequencies are ω_n and ω_d, respectively. Expressing them with the help of k, m and ζ (Equation 2.3):

$$\omega_n = \sqrt{\frac{k}{m}} \qquad \omega_d = \omega_n \sqrt{1 - \zeta^2} \tag{2.3}$$

As Equation 2.2 shows, the response of the system from an initial position is of decaying amplitude, the rate of decay is governed by the damping ratio. In this scenario the presence of damping is an advantage as it helps drive the response to zero over some time t. As it is explained later, internal damping has the opposite effect in structures and would result in rotordynamic instability i.e. the system response would grow infinitely large over time (in real engineering systems this would equal to catastrophic failure).

There is a vast body of research on vibration damping, as well as on material damping. The major works include Linacre's discussion of material damping in steels [43, 44], the published results on the investigation of the nature and characteristics of vibration damping on some idealized models [42, 12] from Lazan and Crandall. Lazan's work also contains a detailed list of models for damping characterization as well as levels of damping for most common materials along with the nature of testing. Sources on the most recent discussion on the rotordynamic effects of internal damping include Wettergren's work which provides a

5

good starting point and discussion on the subject [75]. Other valuable sources also include Crane's paper [13] on damping in composite materials, Genin and Maybee's work on external and internal damping [21] and Gabrys' overview on composite materials testing and design [21]. The ASME handbook [16] and Chandra's work also a good resource on damping in composites [9].

Internal damping and material damping, in general are quite complex phenomena and their properties are difficult to describe. As it has been pointed out in prior research [75], internal damping and material damping, represents a pressing problem in high-speed applications employing composite materials. It is well known that internal damping in rotating assemblies may lead to unstable behavior. The sources of internal damping in composites can be due to several sources, which include but not limited to thermoelastic damping due to heat flow [78], friction generated between the matrix and fiber, energy dissipation at cracks and delaminations [22], as well as non-linear damping at large amplitude vibrations [1]. The basic instability driver is the tangential follower force that results from the internal damping. This results in a cross-coupling that produces a vibration mode that becomes less stable with increasing rotor speed and will result in unstable behavior for super-critical operation.

There has been considerable work in this area that is well-documented in the literature, particularly with regard to internal friction arising from micro-slip at shrink-fit interconnections. The earliest published discussion on the topic was by Newkirk [57], who investigated the failure of compressors and observed violent whirling at speeds above the first critical. Kimball identified the source of this instability as being internal friction [38]. Gunter developed a linear rotordynamic model in which internal friction was modeled as a cross-coupled

force. He was able to explain the instability behavior using this model as well as provide insights into how external damping and support (foundation) flexibility can be used to stabilize such systems [25]. Black investigated a variety of internal friction models (viscous, Coulomb and hysteretic) for internal friction and differentiated between the various models with regard to their ability to accurately predict the onset of instability [6]. Lund also investigated internal friction models, specifically due to micro-slip at axial splines and shrink fit joints [47]. More recently, Jafri conducted further investigations into internal friction effects in built-up rotors [33]. He demonstrated that the cross-coupled stiffness model often used for such systems is incorrect and that the correct model uses cross-coupled internal moments instead. Additionally, Childs pointed out that for many traditional applications the practical value of the linear (viscous damping) model for internal friction is limited because of (1) the dominance of friction from shrink-fit rubbing for internal damping, as compared to material hysteresis and (2) the practical success of minimizing shrink-fit rubbing [11].

However, the system that is considered in this work is not a traditional built-up rotor. Rather, one of the models in this investigation is specifically a flywheel system with a flexible composite hub serving as the interface between the rim and shaft. Baseline experiments conducted by the authors using a simple rotor with a flexible hub/rim interface have demonstrated that the primary unstable vibration mode is a relative displacement in which the hub material is stretched [54]. The primary source of internal dissipation for this system comes from the damping and flexibility of this composite interface between the rotor shaft and the flywheel unit.

If the damping is considered viscous, the basic governing equations are expressed in Equations 2.4 and 2.5:

$$m\ddot{x} + c\dot{x} + kx - c\Omega y = 0 \tag{2.4}$$

$$m\ddot{y} + c\dot{y} + ky + c\Omega x = 0 \tag{2.5}$$

The internal damping within composite rotors has been estimated to be from ten to a hundred times greater than that within a shaft built from conventional materials [75]. Energy dissipating behavior of composite materials can be caused by frictional damping in the matrix and fiber or by hysteretic damping at the fiber-matrix interface. For polymer matrix composites (PMC) with ceramic fibers, the damping capacity of the matrix is expected to be much higher than that of fibers. The damping capacity is a function of fiber volume fraction, fiber orientation and lay-up, load frequency, amplitude and temperature [78]. Thus, in general the true functional form of composite material damping can be quite complex. Most investigators content themselves with identifying an equivalent viscous damping that is used for transient and steady-state response analyses. A detailed characterization is certainly beyond the scope of the present investigation. Rather, the primary concern is evaluating the effectiveness of an adaptive control method for stabilizing rotor systems with internal damping. So, for simplicity in this investigation, a linear (equivalent viscous) model for the internal damping is used in the models developed in Sections 6.1 to 6.3.

8

CHAPTER 3

ACTIVE MAGNETIC BEARINGS

To make high rotational speeds possible, the rotor is often supported by active magnetic bearings. These bearings provide contactless and maintenance-free support of the rotor, although they require active control for stable levitation. Active magnetic bearings (AMB) have gone through remarkable development over the past 40 years. These electro-mechanical devices replace conventional and oil-film bearings and provide non-contact support of the rotor. These devices are most often used in high-speed applications [64] (such as turbo-machines, high-speed electric motors, gas turbines, machine tools) and in situations where the unique non-contact, lubrication-free nature of these devices can be utilized in hostile environments. Due to necessary control, magnetic bearings also provide controllable bearing parameters such as stiffness and damping and are successfully used in several other rotordynamic fields for balancing and vibration suppression.

First report on magnetic levitation dates back from 1842, when Earnshaw performed experiments [15] and has shown that magnetic levitation is inherently unstable. The first practical implementation was performed by Beams in 1937 [5]. Later research in the field of the magnetic bearing control and active magnetic bearings in general, has gained more momentum in the second half of the 20th century. The first analog controllers were soon replaced with digital ones as well as more and more sophisticated control methods have been presented to tackle various imbalance, vibration and rotordynamic stability problems. There have been several successful attempts to make these bearings simpler, cheaper to manufacture and maintain by eliminating the need for position sensors; there were several

papers published on the self-sensing AMB [49, 73]. As the geometrical, electromechanical and magnetic properties of AMBs in general are very well known, the focus of research has turned towards control methods to address several pressing rotordynamic problems. Use of magnetic bearings to suppress rotor vibrations was first published by Humphris [30] using analog controllers. Later, Allaire [2] has developed analog and digital controllers for the same. Rotor vibration suppression using active magnetic bearings was also discussed in [63, 59]. Nonami et al. [58] have compared the use of electromechanical actuators vs. active magnetic bearings. In Knospe's work [40], the use of magnetic bearing control robustness to address rotor vibrations was investigated.

Most of the earlier works on rotor vibration suppression were geared towards fixed-gain methods although later gain scheduling was also considered. Application of modern control methods were also examined on magnetic bearings, such as fuzzy logic control [31], vibration suppression using fuzzy sets [69], robust control, and nonlinear control. Reinig and Desrochers have used a method called Adaptive Feedforward Compensation (AFC) to estimate the disturbance force and use that information to drive the magnetic bearings accordingly [61]. Lum also describes a similar method called adaptive autocentering [46]. Knospe have used different disturbance rejection methods [39].

Thanks to the advances in control theory and magnetic bearing technology, active magnetic bearing-supported rotors have become successful candidates for adaptive control methods for disturbance rejection. In previous works it was assumed that vibrations are caused solely due to mass imbalance. This work however, also deals with the associated internal damping of the shaft material which can result in rotordynamic instability at high speeds.

Operating principle and modeling

The levitated rotor is generally supported by four, symmetrically placed electromagnets. The electromagnetic levitation is achieved by using the attractive force exerted from the electromagnets. The position of the levitated rotor is controlled by continuously adjusting the current in the individual electromagnets. The general structure of a typical AMB is shown on Figure 3.1. The electromagnetic force exerted by a single two-pole electromagnet is described by Equation 3.1 [35]:

$$F = -\mu_0 A_g N_T^2 \frac{i^2}{x^2} \tag{3.1}$$

where i is the total current in the electromagnets and x is the airgap. Expressing the same with the bias current description of the AMB is in Equation 3.2. The control current i_c is being superimposed on the constant bias current I_B.

$$F = \mu_0 A_g N_T^2 \left(\frac{(I_B + i_c)^2}{(d - x)^2} - \frac{(I_B - i_c)^2}{(d + x)^2} \right) \tag{3.2}$$

Most applications of magnetic bearings, however, use the simpler, linearized approach to express the electromagnetic force, which is sufficient when the expected movement of the rotor is small compared to the airgap and the rotor is expected to stay in the close vicinity of the center. This linear representation of the magnetic bearing assumes the force is proportional to the control current and the actual position of the rotor, as shown in Equation 3.10. The linearization of Equation 3.2 is possible by using the first terms of its Taylor-series expansion as:

$$F \approx \frac{\partial F}{\partial x} x + \frac{\partial F}{\partial i_c} i_c \qquad x = 0, i_c = 0 \tag{3.3}$$

The partial derivative of Equation 3.2 with respect to the position x is K_x, called position stiffness.

$$K_x = \frac{\partial}{\partial x}\left(\mu_0 A_g N_T^2 \left(\frac{(I_B + i_c)^2}{(d-x)^2} - \frac{(I_B - i_c)^2}{(d+x)^2}\right)\right) \qquad x = 0, i_c = 0 \qquad (3.4)$$

$$K_x = 2\mu_0 A_g N_T^2 \left(\frac{I_B^2}{d^3} + \frac{I_B^2}{d^3}\right) \qquad (3.5)$$

$$K_x = 4\mu_0 A_g N_T^2 \frac{I_B^2}{d^3} \qquad (3.6)$$

Similarly, the partial derivative of Equation 3.2 with respect to the control current i_c is the current stiffness, K_i.

$$K_i = \frac{\partial}{\partial i_c}\left(\mu_0 A_g N_T^2 \left(\frac{(I_B + i_c)^2}{(d-x)^2} - \frac{(I_B - i_c)^2}{(d+x)^2}\right)\right) \qquad x = 0, i_c = 0 \qquad (3.7)$$

$$K_i = 2\mu_0 A_g N_T^2 \left(\frac{I_B}{d^2} + \frac{I_B}{d^2}\right) \qquad (3.8)$$

$$K_i = 4\mu_0 A_g N_T^2 \frac{I_B}{d^2} \qquad (3.9)$$

The magnetic bearings used in this work have been modeled using the linear expression in Equation 3.10, with the parameters from Equations 3.6 and 3.9.

$$F = K_x x + K_i i_c \qquad (3.10)$$
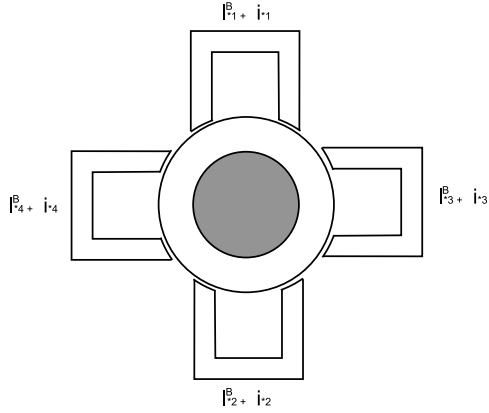
Figure 3.1: Schematic of a radial active magnetic bearing

The shaft position is measured by the proximity sensors placed symmetrically around the shaft. The control system adjusts the amount of current in the individual electromagnets to provide stable levitation. This allows unparalleled flexibility in terms of on-line adjustment of bearing parameters. As shown in Equation 3.2, the relationship between the electromagnetic force, the airgap and the necessary coil current is non-linear. In this case a constant (bias) current I_B is applied to each coil and the control current i_c is being added to or subtracted from. Using this approach, the rotor is supported by the difference of the two counteracting forces. For this layout the net force is the difference of the two attractive forces. More detailed analysis of magnetic bearing types and operating principles can be found in Bleuler's work [7], Schweitzer's book [64] or Maslen's work on AMB sizing [49], among many others. Figure 3.2 shows the logical layout of a typical AMB controller.

13
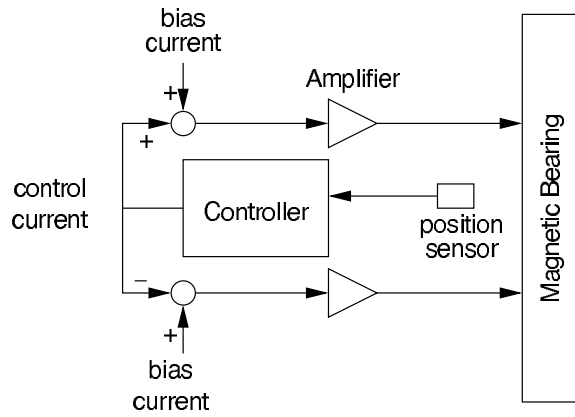
Figure 3.2: AMB control scheme for 1 degree-of-freedom

CHAPTER 4

ADAPTIVE DISTURBANCE REJECTION

The development of high-performance aircraft and the requirements for autopilots was one of the early motivations for research on adaptive control in the 1950-ies [3]. As aircrafts operate over a large range of velocities and flight altitudes their flight dynamics are nonlinear and generally time-varying. The complex dynamics of the aircraft can be approximated by a linear (state-space) model with a continuously varying operating point. The linear model for a given operating point i can be expressed as:

$$\frac{dx}{dt} = A_i x + B_i u \qquad x(0) = x_0 \qquad (4.1)$$

$$y = C_i x + D_i u \qquad (4.2)$$

where A_i, B_i, C_i and D_i are the functions of the operating point, which - in turn - is a function of the actual flight conditions. An adaptive controller should be able to 'learn' the strategy for adjusting the gains for a given operating point. This argument has led to the development of adaptive controllers as shown on Figure 4.1 [32]. The various adaptive schemes differ in the strategy the gains of the controller are adjusted.

An adaptive controller is made up of a parameter estimator and a control law which governs how the controller is adjusted according to the estimator. In indirect adaptive control, the plant parameters are used to calculate the controller parameters. In case of direct adaptive control the plant model is parameterized in terms of the controller parameters. As the detailed introduction and analysis of adaptive control in general is not the scope of this

15
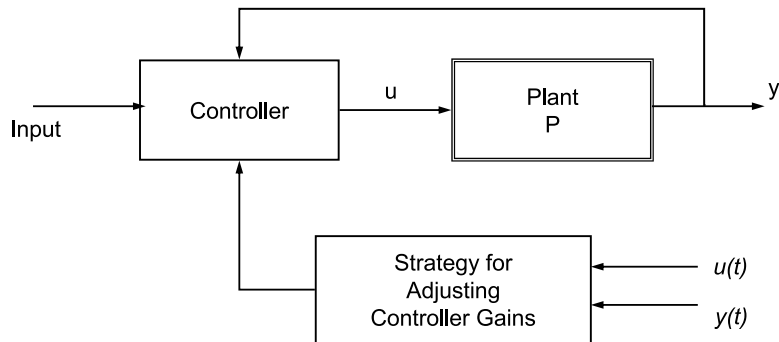
Figure 4.1: Adaptive control

work, the interested reader should consult Astrom's book [4] or Iannoau's work [32], among many others.

Adaptive control in rotordynamics

One of the most pressing problems in rotordynamics has been mass imbalance and the synchronous vibrations as a result of imbalance. There have been several works addressing this problem for rotors supported by rolling element or fluid film bearings as discussed in the works of Gosiewski [23, 24] and Van de Vegte [71]. Adaptive techniques were applied in Shi and Ni's work to address the imbalance problem on flexible rotors [66]. Before the use of magnetic bearings became widespread in rotordynamic applications, adaptive controls were applied to adjustment of balancing weight(s) attached to the rotating shaft. As the limitations of this balancing method were identified the focus of research has turned towards using magnetic bearings' adjustable properties to eliminate or reduce synchronous rotor vibrations due to imbalance. Magnetic bearings can be used to compensate for the vibration of the shaft or to cancel the force transmitted to the bearing foundations. First Knospe and others [39, 40, 41, 59, 17] discussed an adaptive open-loop control scheme for

16

the imbalance vibration control. Williams and colleagues have presented a discussion on analog and digital methods in rotor vibration reduction with AMBs [76]. Shafai and colleagues presented an adaptive method for magnetic bearings to deal with imbalance-induced vibrations [65] as well as other authors [66]. Adaptive feed-forward method was used in a large number of works published, as in Na and Park's discussion [56]. Adaptive methods were often combined with other nonlinear controllers, such as fuzzy logic control. Recently, Huang and Lin presented a fuzzy logic control-based output-feedback solution for imbalance control with magnetic bearings [29]. Another fuzzy controller was used to deal with harmonic disturbances in a rotor system supported by magnetic bearings [28]. Adaptive vibration and imbalance control of rigid rotors supported by magnetic bearings were discussed in Hirschmanner's works [26, 27]. With more detailed analysis and more powerful computers becoming available, flexible rotor models have been developed. Shida et al. [67] and Markert et al. [48] have presented two separate methods for unbalance compensation and vibration control of flexible rotors in magnetic bearings. Some works discussed the importance of actuator and sensor placement on rotor imbalance control [36]. Other works focused on varying speed rotors due to the importance of crossing of critical speeds [68] or multiple-plane balancing [14]. A survey from Zhou and Shi [77] presents the major milestones of vibration control in rotating machinery, including adaptive vibration control, active magnetic bearings and other active balancing methods.

Model reference adaptive control (MRAC) was derived from the model reference problem (MRC). In the latter a good understanding of the plant is required in order to create a good model. The scheme known as Adaptive Disturbance Rejection (ADR) has been developed first to reduce the effects of persistent disturbances in large, complex structures. The

general form of this approach has been presented by Fuentes and Balas [19]. This method is based on Model Reference Adaptive Control (MRAC), in which the disturbance rejection is achieved through an internal reference model. The main advantage of this approach is that the exact shape of the disturbance function does not need to be known in advance.

This work relies heavily on the results obtained by Alex Matras in his thesis and later in his doctoral dissertation. Similar to this work, Matras used a radial active magnetic bearing supported rotor system to simulate and to implement the ADR control scheme. As described in [51], an existing magnetic bearing setup driven by an electric motor was used along with an external simulated sinusoidal disturbance signal to investigate the effectiveness of the ADR method. The rotor was modeled as a rigid-body with 2+2 degrees-of-freedom. Simulations and experiments were carried out while taking into account the bandwidth limitations of the power amplifiers and the effects of rotor speed on the disturbance rejection method's effectiveness. A simple PD-type controller was combined with the adaptive controller. It has been shown that Strict Positive Realness (SPR) could only be guaranteed with current feedback, which was not implemented at the time. Despite the lack of theoretical sufficient conditions for stability the simulation and experimental results have shown that stability can be achieved up to a certain disturbance frequency limit.

In his doctoral dissertation [52] and other works [53], Matras has extended the previous research on Adaptive Disturbance Rejection. Similar to the previous work, a rigid rotor with 2+2 degrees-of-freedom was supported by a pair of radial active magnetic bearings. Amplifier dynamics were also taken into account. A simplified model was also developed with the amplifier dynamics omitted. A linear PID-controller was used to achieve stable levitation. Base motion was also taken into account in the simulations. While the simulated

and experimental model was not SPR, stable adaptive control was achieved using output redefinition. It was found that using redefined outputs the range of rejected frequencies has increased. Experiments have also shown the effectiveness of the output redefinition combined with adaptive disturbance rejection control on the spinning rotor. Similar to Matras' findings, it is also shown in this work that stability can be achieved by using the ADR method despite the system being non-SPR.

Since this work discusses the application of Adaptive Disturbance Rejection, the following section contains the desription of the method as presented in [19].

Adaptive Disturbance Rejection

Consider the bounded vector disturbance function $\mathbf{u_d}(t)$, i.e. the norm of $\mathbf{u_d}$ must be finite, as expressed by Equation 4.3.

$$\|\mathbf{u_d}\| = sup\{\|\mathbf{u_d}\|_p\} \qquad 1 \leq 0 \leq \infty \tag{4.3}$$

The disturbance function can be a combination of several scalar disturbance functions ϕ_k, with amplitude α_k and unit vector e_k, where

$$\|\phi_k\|_\infty = 1 \tag{4.4}$$

It follows that the disturbance function $\mathbf{u_d}$ with l separate disturbance functions accounted for is described as in Equation 4.5:

$$\mathbf{u_d} = \sum_{k=1}^{l} \mathbf{e_k}\phi_k\alpha_k \tag{4.5}$$

19

Also, the basis function $\phi_\mathbf{d}$ is defined as shown in Equation 4.6:

$$\phi_\mathbf{d} = [\phi_1 \quad \phi_2 \quad \dots \quad \phi_k]^T \tag{4.6}$$

The plant's state space model with persistent disturbance $\mathbf{u_d}$ is described by Equations 4.7–4.9.

$$\frac{d\mathbf{x_p}}{dt} = \mathbf{A_p x_p} + \mathbf{B_p u_p} + \mathbf{\Gamma_p u_d} \tag{4.7}$$

$$\mathbf{y_p} = \mathbf{C_p x_p} \tag{4.8}$$

$$\mathbf{x_p^0} = \mathbf{x_p(0)} \in \Re^{\mathbf{N_p}} \tag{4.9}$$

In Equation 4.7 $\mathbf{\Gamma_p}$ is a matrix operator mapping the disturbance $\mathbf{u_d}$ into the state space system. The reference model is described by Equations 4.10 - 4.12, which represent the model of the plant with no disturbances.

$$\frac{d\mathbf{x_m}}{dt} = \mathbf{A_m x_m} + \mathbf{B_m u_m} \tag{4.10}$$

$$\mathbf{y_m} = \mathbf{C_m x_m} \tag{4.11}$$

$$\mathbf{x_m^0} = \mathbf{x_m(0)} \in \Re^{\mathbf{N_m}} \qquad N_m \leq N_p \tag{4.12}$$

The disturbance rejection is achieved by forcing the plant to follow the output of the reference model by modifying the input $\mathbf{u_p}$ of the plant as:

$$\mathbf{u_p} = \mathbf{S_{21} x_m} + \mathbf{S_{22} u_m} + \mathbf{G_p e_y} + \mathbf{H_p}\phi_\mathbf{d} \tag{4.13}$$

20
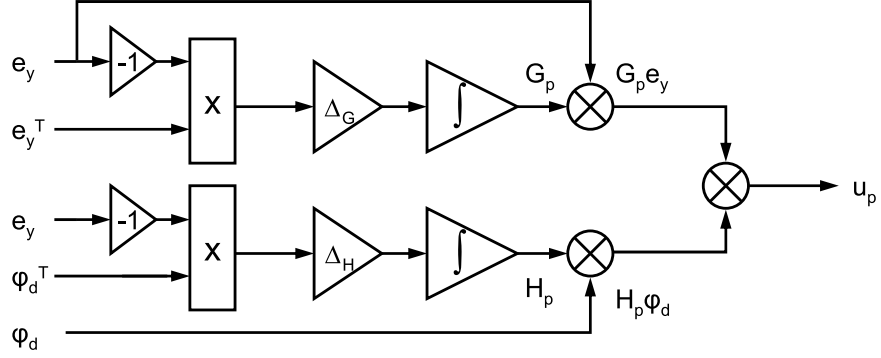
Figure 4.2: Adaptive Disturbance Rejection schematic

with

$$\mathbf{e_y} = \mathbf{y_p} - \mathbf{y_m} \tag{4.14}$$

$$\frac{d\mathbf{S_{21}}}{dt} = -\mathbf{e_y}\mathbf{x_m^T}\mathbf{\Delta_1} \tag{4.15}$$

$$\frac{d\mathbf{S_{22}}}{dt} = -\mathbf{e_y}\mathbf{u_m^T}\mathbf{\Delta_2} \tag{4.16}$$

$$\frac{d\mathbf{G_p}}{dt} = -\mathbf{e_y}\mathbf{e_y^T}\mathbf{\Delta_G} \tag{4.17}$$

$$\frac{d\mathbf{H_p}}{dt} = -\mathbf{e_y}\phi_\mathbf{d}^\mathbf{T}\mathbf{\Delta_H} \tag{4.18}$$

with $\mathbf{\Delta_1}$, $\mathbf{\Delta_2}$, $\mathbf{\Delta_G}$ and $\mathbf{\Delta_H}$ positive definite. In this work the model is following a zero reference, that is

$$
\begin{aligned}
\mathbf{y_m} &= \mathbf{0} \\
\mathbf{x_m} &= \mathbf{0} \\
\mathbf{S_{21}} &= \mathbf{0} \\
\mathbf{S_{22}} &= \mathbf{0}
\end{aligned}
\tag{4.19}
$$

therefore the original expression for $\mathbf{u_p}$ reduces to

$$\mathbf{u_p} = \mathbf{G_p}\mathbf{e_y} + \mathbf{H_p}\phi_{\mathbf{d}} \tag{4.20}$$

From Equation 4.14 the driving error $\mathbf{e_y}$ of the model will also reduce to:

$$\mathbf{e_y} = \mathbf{y_p} \tag{4.21}$$

$$\mathbf{y_p} = \mathbf{C_x}\mathbf{q} \tag{4.22}$$

$$\frac{d\mathbf{G_p}}{dt} = -\left(\mathbf{C_x}\mathbf{q}\right)\left(\mathbf{C_x}\mathbf{q}\right)^T \mathbf{\Delta_G} \tag{4.23}$$

$$\frac{d\mathbf{H_p}}{dt} = -\mathbf{e_y}\phi_{\mathbf{d}}^{\mathbf{T}} \mathbf{\Delta_H} \tag{4.24}$$

The persistent disturbance due to mass imbalance is sinusoidal in nature, therefore a straightforward choice for $\phi_{\mathbf{d}}$ - considering the only synchronous components of the disturbance - is:

$$\phi_{\mathbf{d}} = [\sin\left(\Omega t\right) \quad \cos\left(\Omega t\right) \quad 1]^T \tag{4.25}$$

With this choice of $\phi_{\mathbf{d}}$ and $\mathbf{e_y}$ the main task becomes to obtain an output matrix $\mathbf{C_x}$ which would give the desired response by modifying $\mathbf{G_p}$ and $\mathbf{H_p}$ appropriately.

$$\frac{d\mathbf{q}}{dt} = \left(\mathbf{A} - \mathbf{B}\mathbf{K_F}\right)\mathbf{q} + \mathbf{G_p}\mathbf{e_y} + \mathbf{H_p}\phi_{\mathbf{d}} \tag{4.26}$$

$$\mathbf{u} = \mathbf{K_F}\mathbf{q} + \mathbf{G_p}\mathbf{e_y} + \mathbf{H_p}\phi_{\mathbf{d}} \tag{4.27}$$

$$\mathbf{u} = \left(\mathbf{K_F} + \mathbf{G_p}\mathbf{C_x}\right)\mathbf{q} + \mathbf{H_p}\phi_{\mathbf{d}} \tag{4.28}$$

The ADR control scheme used in this configuration has two adaptive gains, $\mathbf{G_p}$ and $\mathbf{H_p}$, each used for different purposes. $\mathbf{G_p}$ is a 'stabilizing' gain, and as shown in Equation 4.23, it is driven by the deviation of the plant output from the reference. On the other hand, $\mathbf{H_p}$ is used with the disturbance basis function $\phi_\mathbf{d}$, so that the disturbance is canceled out. The number of elements of $\phi_\mathbf{d}$ defines what disturbance functions are accounted for. As $\mathbf{H_p}$ and $\phi_\mathbf{d}$ are used to suppress the effects of persistent disturbances, in this work this disturbance is due to mass imbalance. It can also be shown that $\mathbf{H_p}$ is a matrix gain; its size is defined by the number of outputs of the plant and the number of elements of $\phi_\mathbf{d}$. In general, by choosing to include more elements in $\phi_\mathbf{d}$ - to account for disturbances of sub- and supersynchronous frequencies, for example - more accurate disturbance rejection can be achieved. The weighting matrices $\boldsymbol{\Delta_G}$ and $\boldsymbol{\Delta_H}$ are positive definite and their value acts as a 'gain' to $\mathbf{G_p}$ and $\mathbf{H_p}$, controlling the overall speed and accuracy of adaptation.

In order to make the adaptive scheme work, the open loop system must be Almost Strictly Positive Real (ASPR) i.e. the $(\mathbf{A_p} + \mathbf{B_p G_p C_p}, \mathbf{B_p}, \mathbf{C_p})$ triple must be Strictly Positive Real (SPR). According to the Kalman-Yakubovich lemma, a triple $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ is SPR, if all of the three conditions are true: 1) $\mathbf{A}$ is stable, 2) $(\mathbf{A}, \mathbf{B})$ is controllable and 3) there exist symmetric positive definite matrix operators $\mathbf{Q}$ and $\mathbf{P}$ such that

$$\mathbf{A^T P} + \mathbf{PA} = -\mathbf{Q} \tag{4.29}$$

$$\mathbf{PB} = \mathbf{C^T} \tag{4.30}$$

In other words, for Almost Strict Positive Realness (ASPR), the product $\mathbf{C_p B_p}$ must be positive definite and there must be no unstable zeros in the open-loop transfer function [74].

23

Each simulation model is verified against these requirements later in Chapter 5. It is found that the adaptive controller works even if these theoretical conditions are not met.

Chapter 5

Simulation Models

There are three models used in this work, ranging from the simple Jeffcott-rotor model to a very detailed flexible shaft model. In all three cases the synchronous disturbance is due to a static mass imbalance and the rotordynamic instability is caused by internal damping in the shaft material or in the flexible hub. The development of each model, complete with equations of motion, controller design is presented below.

5.1 Jeffcott-rotor model

The first published analysis of a two-degrees-of-freedom model of a rigid rotor (viewed as a spring-mass system) was performed by Rankine [60] to explain the critical speed behavior of such systems. He has predicted - incorrectly - that rotating machines would never be able to rotate faster than their first critical speed. As a result of his work, engineers have been focusing on designing rotors with high critical speed; meaning heavy shafts with large diameters. Rankine used a model similar to the one on Figure 5.1, a uniform shaft with the effects of friction ignored. The equations of motions developed for that model were incorrectly missing the Coriolis-terms.

Jeffcott's analysis [34] of a flexible rotor has shown that stable operation above the critical speed was possible provided the system was properly balanced. Taylor's work [70] has verified Jeffcott's findings. This prompted engineers to focus on higher operating speeds.
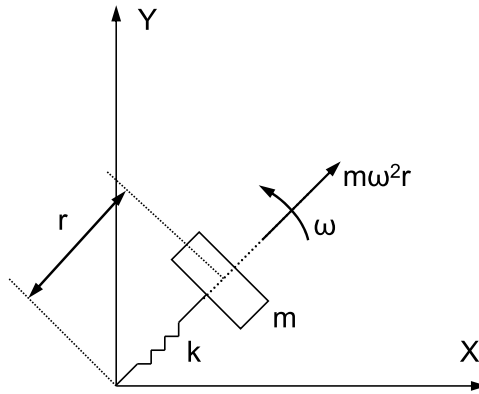
Figure 5.1: Rankine rotor model
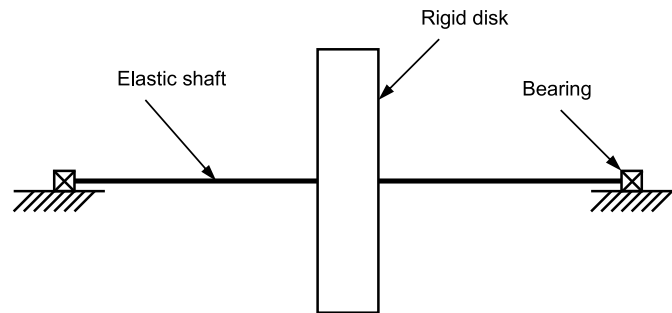
Figure 5.2: Jeffcott rotor model

Since it is practically impossible to build a perfectly balanced rotor in real engineering applications, rotating imbalance is the most often observed source for synchronous vibrations. The Jeffcott-rotor, pictured in Figure 5.2, is the most often used model for rotordynamic analysis. It consists of a large, unbalanced disk mounted on a flexible shaft, placed symmetrically between the bearings. The equations of motion for this model are:

$$m\ddot{x} + c\dot{x} + kx = m\Omega^2 u \cos(\Omega t) \tag{5.1}$$

$$m\ddot{y} + c\dot{y} + ky = m\Omega^2 u \sin\left(\Omega t\right) \tag{5.2}$$

where u is the amount of static imbalance. Gravity loads are omitted as they are often insignificant compared to the rest of the loads in turbomachines. Assuming constant speed Ω and disk mass m, shaft bending stiffness k, the viscous damping resulting from air drag is denoted by c. The amplitude of the synchronous whirl is increasing as the running speed is approaching the critical speed and then decreases at supercritical speeds until reaching the value of static imbalance. Near the critical speed damping is the single most important property which limits the amplitude of the synchronous whirl. There are three major solutions to minimize the amplitude of the synchronous whirl [72]; balancing the rotor; changing the rotor speed or to add damping to the system. Adding damping to the system is possible only by the bearings as the Jeffcott-rotor does not include any damping terms other than air drag. Internal damping, on the other hand, would cause non-synchronous whirl.

Sub- and super-synchronous rotor whirl is often caused by shaft misalignment, rotor-stator rubbing, and loose bearing housings, among others. The solution for these issues is to reduce the underlying mechanical issues, such as alignment of the shaft, eliminate the rub, etc. Rotordynamic instability has its history of causing expensive machine failures, especially in centrifugal compressors used by the process industries. The core of the instability is the fact that while damping is positive (see Equations 5.1 and 5.2 above) it has stabilizing effect. On the other hand, if the damping is negative, it results in rotordynamic instability. In rotating machines this instability is most often caused by forces which are tangential to the whirl path and acting in the same direction as the instantaneous motion. The simulated Jeffcott-rotor is supported by two radial AMB, represented by their linear

model. The effects of internal damping were added to the equations of motion as well as the static mass imbalance.

The equations of motion for the Jeffcott-rotor model are shown in Equations 5.3 and 5.4. The equations were obtained by replacing the spring force in Equations 5.1 and 5.2 with the force from the magnetic bearings and adding the expressions for the cross-coupled force due to internal damping. The force provided by the magnetic bearings is described by the linear AMB model (discussed earlier) in Equation 5.6. The control currents i_{cx} and i_{cy} are calculated by using a PD-type controller as shown in Equation 5.7.

$$m\ddot{x} + c_i\dot{x} + \Omega c_i y = F_{x,AMB} + F_{x,imb} \tag{5.3}$$

$$m\ddot{y} + c_i\dot{y} - \Omega c_i x = F_{y,AMB} + F_{y,imb} \tag{5.4}$$

$$F_{x,imb} = m\Omega^2 u \cos\left(\Omega t\right) \qquad F_{y,imb} = m\Omega^2 u \sin\left(\Omega t\right) \tag{5.5}$$

$$F_{x,AMB} = K_x x + K_i i_{cx} \qquad F_{y,AMB} = K_x y + K_i i_{cy} \tag{5.6}$$

$$i_{cx} = K_p x + K_d \dot{x} \qquad i_{cy} = K_p y + K_d \dot{y} \tag{5.7}$$

Expressing the equations of motion in state-space (with state vector in Equation 5.9):

$$\begin{aligned} \frac{d\mathbf{q}}{dt} &= \mathbf{Aq} + \mathbf{Bu} \\ \mathbf{y} &= \mathbf{Cq} \end{aligned} \tag{5.8}$$

$$\mathbf{q} = \left\{ \begin{array}{c} x \\ y \\ \dot{x} \\ \dot{y} \end{array} \right\} \tag{5.9}$$

The state matrix $\mathbf{A}$ with the state-feedback:

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{K_x + K_i K_p}{m} & -\frac{\Omega c_i}{m} & \frac{K_i K_d - c_i}{m} & 0 \\ \frac{\Omega c_i}{m} & \frac{K_x + K_i K_p}{m} & 0 & \frac{K_i K_d - c_i}{m} \end{bmatrix} \tag{5.10}$$

$$\mathbf{B} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{5.11}$$

The control input $\mathbf{u}$ modified with the adaptive controller:

$$\mathbf{u} = \left\{ \begin{array}{c} i_{cx} \\ i_{cy} \end{array} \right\} + \mathbf{G_p e_{yG}} + \mathbf{H_p} \phi_{\mathbf{d}} \tag{5.12}$$

where $\mathbf{e_{yG}}$ is the error signal driving $\mathbf{G_p}$ to stabilize the system, in this case

$$\mathbf{e_{yG}} = \mathbf{C_G q} \tag{5.13}$$

$$\frac{d\mathbf{G_p}}{dt} = -\mathbf{e_{yG} e_{yG}}^T \mathbf{\Delta_G} \tag{5.14}$$

29

where $\mathbf{C_G}$ has been selected as:

$$\mathbf{C_G} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \qquad (5.15)$$

Also, $\mathbf{e_{yH}}$ is the error signal driving $\mathbf{H_p}$ to reject the synchronous disturbances as:

$$\mathbf{e_{yH}} = \mathbf{C_H q} \qquad (5.16)$$

$$\frac{d\mathbf{H_p}}{dt} = -\mathbf{e_{yH}}{\phi_\mathbf{d}}^T \boldsymbol{\Delta_H} \qquad (5.17)$$

for the best results, $\mathbf{C_H}$ has been selected as:

$$\mathbf{C_H} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \qquad (5.18)$$

The disturbance function $\phi_\mathbf{d}$ has been chosen to include the synchronous components as shown in Equation 5.19:

$$\phi_\mathbf{d} = [\sin(\Omega t) \quad \cos(\Omega t) \quad 1]^T \qquad (5.19)$$

Verification of sufficient conditions for this model:

The product $\mathbf{C_G B}$ (or $\mathbf{C_H B}$) must be positive definite and there must be no unstable zeros in the open-loop transfer function [74]. The $\mathbf{A}$ matrix is expressed in Equation 5.10 and the $\mathbf{B}$ matrix is expressed in Equation 5.11 and the MATLAB script in Appendix A.10

Name	Value	Dimension
N	330	–
A_g	3.22e^{-4}	mm^2
d	0.002	m
i_b	1	A
μ_0	$4\pi\,\text{e}^{-7}$	H/m
m	10	kg
u	0.05	kgm
K_p	-1000	–
K_d	-5	–
c_i	60	Ns/m

Table 5.1: Jeffcott rotor model parameters

shows that both these conditions are met for this model, as

$$\mathbf{C_G B} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{5.20}$$

and the zeros of the state-space model built with the $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C_G}$ (or $\mathbf{C_H}$) matrices are all positive. This model satisfies the sufficient conditions for Strict Positive Realness according to [74].
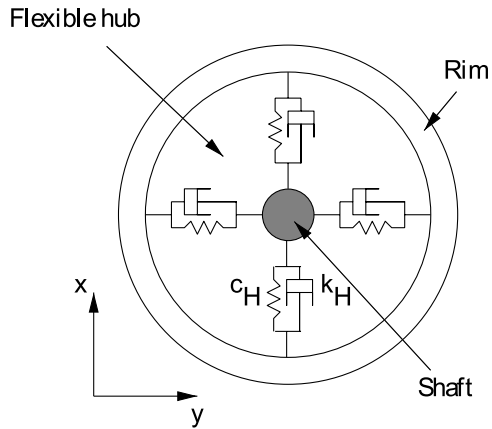
Figure 5.3: Flexible hub model, front view

5.2 Flexible hub model

The simulation model consists of two radial magnetic bearings, current amplifiers, proximity sensors, a horizontally mounted rotor and the flexibly connected outer rim. The effects of gravity were omitted in the simulations, as the weight of the rotor assembly can be offset by applying constant current-offset in the top and bottom electromagnets in each magnetic bearing. The simulation model was developed using MATLAB. For the sake of simplicity, a rigid rotor is considered and, due to the small displacements, only lateral motion (horizontal and vertical) have been considered. The model has 2+2 (shaft and rim) DOF for lateral motion of the shaft and the rim in vertical and in horizontal directions, respectively. These assumptions were based on previously experimentally observed and verified results [50]. The model of the flexible hub flywheel is shown on Figures 5.3 and 5.4.

The shaft and the rim are treated as two discrete masses (m_S and m_R) connected by the flexible hub, which is represented by two sets of springs and dampers in the vertical and
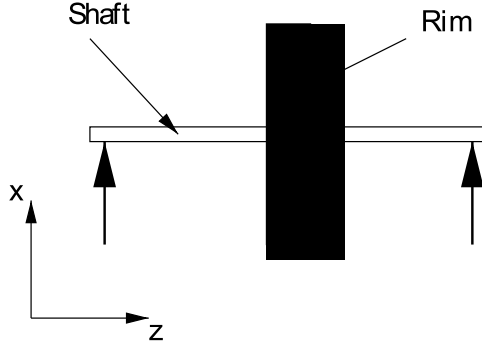
Figure 5.4: Flexible hub model, side view

horizontal directions. The damper and spring constants have been obtained from previous experimental results [54, 55]. The magnetic bearings are described by their linearized model and the forces exerted by the individual magnetic bearings (called bearing A and B) in x and y direction is denoted by F_{xA}, F_{yA} and F_{xB}, F_{yB}, respectively. The equations of motion of the entire system are described by Equations 5.37–5.40. For the sake of simplicity the hub stiffness k_H and damping c_H terms are assumed to be equal in both vertical and horizontal directions (isotropic case), although this is not always true, due to the inherent manufacturing inaccuracies of composites or other geometrical and material imperfections of the rotor assembly. The model development is presented in detail below. The development starts with expressions for the potential energy T and kinetic energy V:

$$T = \frac{1}{2}m_S \left(\dot{x}_S^2 + \dot{y}_S^2 \right) + \frac{1}{2}m_R \left(\dot{x}_R^2 + \dot{y}_R^2 \right) \tag{5.21}$$

$$V = \frac{1}{2}k_{HX} \left(x_R - x_S \right)^2 + \frac{1}{2}k_{HY} \left(y_R - y_S \right)^2 + \frac{1}{2}k_{BX} x_S^2 + \frac{1}{2}k_{BY} y_S^2 \tag{5.22}$$

The dissipation force $\mathbf{F_D}$ acting on the hub-rim interface, expressed in the frame fixed to the rotating assembly:

$$\mathbf{F_D} = -c_{HX}\left[\left(\dot{x}_R^{rot} - \dot{x}_S^{rot}\right)\mathbf{x}\right] - c_{HY}\left[\left(\dot{y}_R^{rot} - \dot{y}_S^{rot}\right)\mathbf{y}\right] \tag{5.23}$$

The dissipation force $\mathbf{F_B}$ due to the bearings:

$$\mathbf{F_B} = -c_{BX}\dot{x}_S\hat{x} - c_{BY}\dot{y}_S\hat{y} \tag{5.24}$$

The coordinate transformation from rotating system to the fixed:

$$\left\{\begin{array}{c} x^{rot} \\ y^{rot} \end{array}\right\} = \left[\begin{array}{cc} \cos\left(\Omega t\right) & \sin\left(\Omega t\right) \\ -\sin\left(\Omega t\right) & \cos\left(\Omega t\right) \end{array}\right]\left\{\begin{array}{c} x \\ y \end{array}\right\} \tag{5.25}$$

Expressing the unit vectors $\bar{x}$ and $\bar{y}$:

$$\left\{\begin{array}{c} \bar{x} \\ \bar{y} \end{array}\right\} = \left[\begin{array}{cc} \cos\left(\Omega t\right) & \sin\left(\Omega t\right) \\ -\sin\left(\Omega t\right) & \cos\left(\Omega t\right) \end{array}\right]\left\{\begin{array}{c} \hat{x} \\ \hat{y} \end{array}\right\} \tag{5.26}$$

The equations of motion for the system can be obtained by expressing the Lagrangian and solving the following set of equations:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} \qquad \mathbf{q} = \{x_S \quad y_S \quad x_R \quad y_R\}^T \qquad i = 1\ldots4 \tag{5.27}$$

which gives:

$$m_S\ddot{x}_S - k_{HX}\left(x_R - x_S\right) + k_{BX}x_S = Q_1 \tag{5.28}$$

$$m_S \ddot{x}_S - k_{HY}(y_R - y_S) + k_{BY} y_S = Q_2 \tag{5.29}$$

$$m_R \ddot{x}_R + k_{HX}(x_R - x_S) = Q_3 \tag{5.30}$$

$$m_R \ddot{x}_R + k_{HY}(y_R - y_S) = Q_4 \tag{5.31}$$

Expressing $\mathbf{F_D}$ in the stationary coordinate system:

$$\mathbf{F_D} = ((c_{HX}\cos^2(\Omega t) + c_{HY}\sin^2(\Omega t))((\dot{x}_S - \dot{x}_R) + \Omega(y_S - y_R)) +$$

$$\sin(\Omega t)\cos(\Omega t)((\dot{y}_R - \dot{y}_S) - \Omega(x_R - x_S))(c_{HY} - c_{HX}))\hat{x} + \tag{5.32}$$

$$((c_{HX}\sin^2(\Omega t) + c_{HY}\cos^2(\Omega t))((\dot{y}_S - \dot{y}_R) - \Omega(x_S - x_R)) +$$

$$\sin(\Omega t)\cos(\Omega t)((\dot{x}_R - \dot{x}_S) + \Omega(y_R - y_S))(c_{HY} - c_{HX}))\hat{y}$$

Next, let us take into consideration that $\mathbf{Q}$ is composed of $\mathbf{F_D}$ and $\mathbf{F_B}$:

$$Q_1 = -\left(c_{HX}\cos^2(\Omega t) + c_{HY}\sin^2(\Omega t)\right)((\dot{x}_S - \dot{x}_R) + \Omega(y_S - y_R)) -$$

$$\sin(\Omega t)\cos(\Omega t)((\dot{y}_R - \dot{y}_S) - \Omega(x_R - x_S))(c_{HY} - c_{HX}) - c_{BY}\dot{x}_S \tag{5.33}$$

$$Q_2 = -\left(c_{HX}\sin^2(\Omega t) + c_{HY}\cos^2(\Omega t)\right)((\dot{y}_S - \dot{y}_R) - \Omega(x_S - x_R)) -$$

$$\sin(\Omega t)\cos(\Omega t)((\dot{x}_R - \dot{x}_S) + \Omega(y_R - y_S))(c_{HY} - c_{HX}) - c_{BY}\dot{y}_S \tag{5.34}$$

$$Q_3 = \left(c_{HX}\cos^2(\Omega t) + c_{HY}\sin^2(\Omega t)\right)((\dot{x}_S - \dot{x}_R) + \Omega(y_S - y_R)) +$$

$$\sin(\Omega t)\cos(\Omega t)((\dot{y}_R - \dot{y}_S) - \Omega(x_R - x_S))(c_{HY} - c_{HX}) \tag{5.35}$$

$$Q_4 = \left(c_{HX} \sin^2\left(\Omega t\right) + c_{HY} \cos^2\left(\Omega t\right)\right) \left(\left(\dot{y}_S - \dot{y}_R\right) - \Omega\left(x_S - x_R\right)\right) +$$

$$\sin\left(\Omega t\right) \cos\left(\Omega t\right) \left(\left(\dot{x}_R - \dot{x}_S\right) + \Omega\left(y_R - y_S\right)\right) \left(c_{HY} - c_{HX}\right) \qquad (5.36)$$

Now all four equations (Equations 5.33 – 5.36) can be simplified by assuming the isotropic case (i.e. $k_B=k_{BX}=k_{BY}$, $c_B=c_{BX}=c_{BY}$, $k_H=k_{HX}=k_{HY}$ and $c_H=c_{HX}=c_{HY}$). The bearing forces have been replaced by the forces from the linearized magnetic bearing models for each magnetic bearing (F_{xA}, F_{xB}, F_{yA} and F_{yB}).

$$m_S \ddot{x}_S + F_{xA} + F_{xB} + F_{x,imb} + c_H\left(\dot{x}_S - \dot{x}_R\right) + k_H\left(x_S - x_R\right) = -\Omega c_H\left(y_S - y_R\right) \qquad (5.37)$$

$$m_S \ddot{y}_S + F_{yA} + F_{yB} + F_{y,imb} + c_H\left(\dot{y}_S - \dot{y}_R\right) + k_H\left(y_S - y_R\right) = \Omega c_H\left(x_S - x_R\right) \qquad (5.38)$$

$$m_R \ddot{x}_R + c_H\left(\dot{x}_R - \dot{x}_S\right) + k_H\left(x_R - x_S\right) = -\Omega c_H\left(y_R - y_S\right) \qquad (5.39)$$

$$m_R \ddot{y}_R + c_H\left(\dot{y}_R - \dot{y}_S\right) + k_H\left(y_R - y_S\right) = \Omega c_H\left(x_R - x_S\right) \qquad (5.40)$$

The imbalance forces due to the static imbalance u ($F_{x,imb}$ and $F_{y,imb}$) are expressed as

$$F_{x,imb} = u\Omega^2 \sin\left(\Omega t\right) \qquad (5.41)$$

$$F_{y,imb} = u\Omega^2 \cos\left(\Omega t\right) \qquad (5.42)$$

The cross-coupled stiffness terms are due to the transformation of the damping forces from the coordinate system fixed to the rotor to the stationary coordinate system of the rim. These cross-coupling terms are causing instability of the system as soon as the running speed becomes sufficiently high (see Figure 5.5). The controller is a typical PD-type, with

K_p and K_d values obtained from previous experiments [18, 69] (see Equations 3.5 and 3.9) and known to give stable levitation with the magnetic bearings used in this work. It is also assumed that both magnetic bearings have the same electrical, magnetic and geometrical parameters, shown in Table 5.2. The currents in the coils are provided by switching amplifiers of moderate bandwidth. Direct measurement of the rim coordinates (x_R, y_R) is impractical due to the flexible attachment to the shaft and the expected range of motion can be quite large. Additional proximity sensors and their wiring would also increase the overall mechanical and electrical complexity of the setup. For these reasons a reduced-order observer has been designed to estimate the rim coordinates. These estimates, as well as the measured shaft coordinates, were used in the calculation of the adaptive gains of the controller.

The levitation is achieved by using a PD-type controller with the appropriate K_p and K_d values as shown in Equations 5.45 and 5.46. Assuming a rigid shaft the basic equation of motion for the stationary shaft is

$$m\ddot{x} = F_x \tag{5.43}$$

where $m = m_S + m_R$ and based on the linearized AMB model and F_x can be expressed as

$$F_x = K_x x + K_i i_c \tag{5.44}$$

also, assuming a PD controller for i_c, the complete model for the levitation is

$$m\ddot{x} = K_x x + K_i \left(K_p x + K_d \dot{x} \right) \qquad i_c = K_p x + K_d \dot{x} \tag{5.45}$$

therefore the state space model with state feedback is

$$
\left\{ \begin{array}{c} \dot{x} \\ \ddot{x} \end{array} \right\} = \left[\begin{array}{cc} 0 & 1 \\ \left(\frac{(K_x + K_i K_p)}{m} \right) & \left(\frac{K_i K_d}{m} \right) \end{array} \right] \left\{ \begin{array}{c} x \\ \dot{x} \end{array} \right\} \tag{5.46}
$$

which results in $K_p < -500$ and $K_d < -1$ as minimum values for stable levitation using the simulation parameters listed in Table 5.2.

State estimator development

Since the direct measurement of the rim coordinates is a somewhat impractical and unreliable solution, it became apparent that those coordinates should be estimated instead. Therefore a reduced-order state-estimator has been developed which estimates the rim position and - in turn - the rim velocity.

Consider the general form of a mathematical model expressed in the state-space form where x and y is the state vector and output vector, respectively:

$$
\begin{aligned}
\frac{d\mathbf{x}}{dt} &= \mathbf{Ax} + \mathbf{Bu} \\
\mathbf{y} &= \mathbf{Cx}
\end{aligned} \tag{5.47}
$$

For the model discussed here the matrices are as follows:

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ \frac{-2K_x-k_H}{m_S} & -\frac{\Omega c_H}{m_S} & \frac{k_H}{m_S} & \frac{\Omega c_H}{m_S} & -\frac{c_H}{m_S} & 0 & \frac{c_H}{m_S} & 0 \\ \frac{\Omega c_H}{m_S} & \frac{-2K_x-k_H}{m_S} & -\frac{\Omega c_H}{m_S} & \frac{k_H}{m_S} & 0 & -\frac{c_H}{m_S} & 0 & \frac{c_H}{m_S} \\ \frac{k_H}{m_R} & \frac{\Omega c_H}{m_R} & -\frac{k_H}{m_R} & -\frac{\Omega c_H}{m_R} & \frac{c_H}{m_R} & 0 & -\frac{c_H}{m_R} & 0 \\ -\frac{\Omega c_H}{m_R} & \frac{k_H}{m_R} & \frac{\Omega c_H}{m_R} & -\frac{k_H}{m_R} & 0 & \frac{c_H}{m_R} & 0 & -\frac{c_H}{m_R} \end{bmatrix} \tag{5.48}$$

$$\mathbf{B} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ -\frac{2K_i}{m_S} & 0 \\ 0 & -\frac{2K_i}{m_S} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \tag{5.49}$$

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{5.50}$$

With an appropriate state feedback $(\mathbf{u} = -\mathbf{KKx})$ the system is modified as

$$
\begin{aligned}
\frac{d\mathbf{x}}{dt} &= (\mathbf{A} - \mathbf{BKK})\,\mathbf{x} \\
\mathbf{y} &= \mathbf{Cx}
\end{aligned}
\tag{5.51}
$$

The state feedback matrix $\mathbf{KK}$ contains the gains of the PD-controller described earlier:

$$
\mathbf{KK} = \left[
\begin{array}{cccccccc}
K_p & 0 & 0 & 0 & K_d & 0 & 0 & 0 \\
0 & K_p & 0 & 0 & 0 & K_d & 0 & 0
\end{array}
\right]
\tag{5.52}
$$

For the state feedback to work properly it is necessary that all states are available. In this case some of the states are unavailable due to the limitations mentioned above so the missing states would be replaced with estimated ones. After obtaining the solution for the Lyapunov-equation $(\mathbf{TA} - \mathbf{FT} = \mathbf{LC})$, one can proceed with the calculation of the estimated states. The detailed development is not shown here, as it follows the standard method which developed by Luenberger [45] as well as presented in many control text books, such as [8, 10, 37]

$$
\mathbf{L} = \left[
\begin{array}{cccc}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{array}
\right]
\tag{5.53}
$$

$$
\mathbf{F} = \begin{bmatrix} -1000 & 0 & 0 & 0 \\ 0 & -2000 & 0 & 0 \\ 0 & 0 & -3000 & 0 \\ 0 & 0 & 0 & -4000 \end{bmatrix} \tag{5.54}
$$

The estimate of the state vector $\mathbf{x}$ is $\hat{\mathbf{x}}$ which is obtained by solving Equation 5.56:

$$
\hat{\mathbf{x}} = \begin{bmatrix} \mathbf{C} \\ \mathbf{T} \end{bmatrix}^{-1} \left\{ \begin{array}{c} \mathbf{y} \\ \mathbf{z} \end{array} \right\} \tag{5.55}
$$

$$
\frac{d\hat{\mathbf{x}}}{dt} = (\mathbf{A} - \mathbf{LC})\,\hat{\mathbf{x}} + \mathbf{Bu} + \mathbf{Ly} \tag{5.56}
$$

The dynamics for $\mathbf{z}$ are described by Equation 5.57:

$$
\frac{d\mathbf{z}}{dt} = \mathbf{Fz} + \mathbf{TBu} + \mathbf{Ly} \tag{5.57}
$$

After obtaining the estimates states the overall state matrix for the model is

$$
\mathbf{x} = \left\{ \begin{array}{c} x_S \\ y_S \\ \hat{x}_R \\ \hat{y}_R \\ \dot{x}_S \\ \dot{y}_S \\ \dot{\hat{x}}_R \\ \dot{\hat{y}}_R \end{array} \right\} \tag{5.58}
$$

Name	Value	Dimension		Name	Value	Dimension
A_g	$3.28\mathrm{e}^{-4}$	m^2		K_d	-5	
N	330			K_i	11.02	N/A
d	2	mm		K_x	5508.1	N/m
μ_0	$4\pi\mathrm{e}^{-7}$	H/m		m_S	0.238	kg
i_b	1	A		m_R	0.567	kg
k_H	2000	N/m		Δ_G	1500	
c_H	4.84	Ns/m		Δ_H	100	
K_p	-1000			u	0.005	kgm

Table 5.2: Flexible hub model parameters

The error signal $\mathbf{e_{yG}}$ which is used to calculate the adaptive part of controller is made of the estimated rim coordinates for $\mathbf{G_p}$ and the shaft velocity for $\mathbf{H_p}$, as shown in Equations 5.59 and 5.60:

$$\frac{d\mathbf{G_p}}{dt} = -\mathbf{e_{yG}}\mathbf{e_{yG}^T}\boldsymbol{\Delta_G} \tag{5.59}$$

$$\frac{d\mathbf{H_p}}{dt} = -\mathbf{e_{yH}}\phi_{\mathbf{d}}^{\mathbf{T}}\boldsymbol{\Delta_H} \tag{5.60}$$

where the error inputs $\mathbf{e_{yG}}$ and $\mathbf{e_{yH}}$ are expressed as

$$\mathbf{e_{yG}} = \left\{ \begin{array}{c} \hat{x}_R \\ \hat{y}_R \end{array} \right\} \tag{5.61}$$

$$\mathbf{e_{yH}} = \left\{ \begin{array}{c} \dot{x}_S \\ \dot{y}_S \end{array} \right\} \tag{5.62}$$
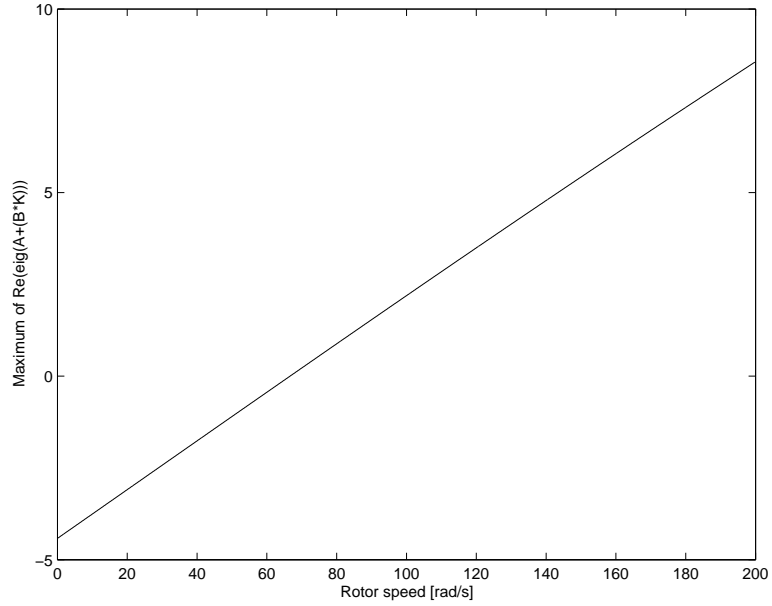
Figure 5.5: Maximum of real part of eigenvalue as a function of rotor speed

Verification of sufficient conditions for this model:

As stated before, the product of $\mathbf{C_x B}$ matrices must be positive definite and there must be no unstable zeros in the open-loop transfer function [74]. The state matrix for this model is $(\mathbf{A} - \mathbf{BKK})$ as expressed in Equations 5.48 and 5.52. The input matrix $\mathbf{B}$ is expressed in Equation 5.49. The MATLAB script in Appendix A.11 shows that one of the two conditions are met for this model, as the eigenvalues of the $\mathbf{C_x B}$ product are all positive.

$$\mathbf{C_x B} = \begin{bmatrix} 92.57 & 0 \\ 0 & 92.57 \end{bmatrix} \tag{5.63}$$

Some zeros of the state-space model built with the $(\mathbf{A} - \mathbf{BKK})$, $\mathbf{B}$ and $\mathbf{C_x}$ matrices are 0, suggesting that the model does not satisfy the sufficient theoretical conditions.

43
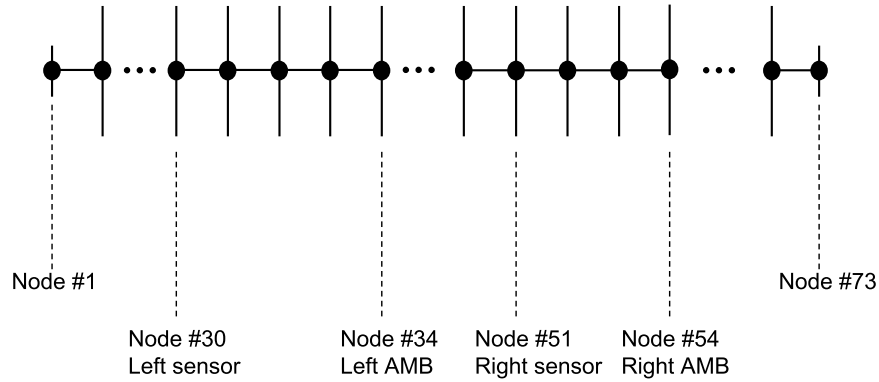
Figure 5.6: Nodal distribution for flexible shaft model

5.3 Flexible shaft model

The simulation model consists of a slender, flexible shaft supported by two radial active magnetic bearings at both ends. The rotor is modeled by using the free-free bending mode shapes and natural frequencies obtained from finite element (FE) model. The FE model uses 73 nodes and the first four mode shapes (two rigid body modes and two flexible modes) are included in the simulation. Although the model includes the flexible shaft modes, due to the insufficient number of position sensor locations those flexible modes can not be used for controller design. The laminated parts of the shaft at the nodes where the magnetic bearings are located ($nb1$, $nb2$) as well as at the position sensors ($np1$, $np2$) are represented by additional masses. The imbalance is modeled as a single point-mass acting at the midspan of the bearings - defined by the imbalance vector ψ - although it is not included in the finite element analysis. Figure 5.6 shows the representation of the finite element model which was developed using ANSYS. Figures 5.7 to 5.10 show each of the four mode shapes obtained from the FE analysis.

44
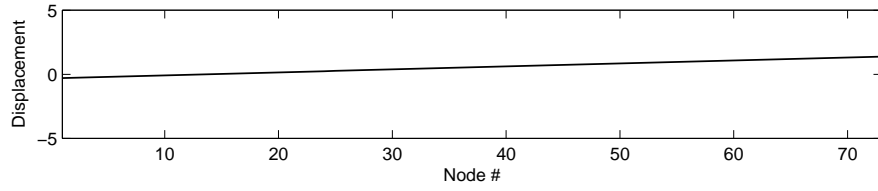
Figure 5.7: First mode shape of flexible shaft model (1st rigid body mode)
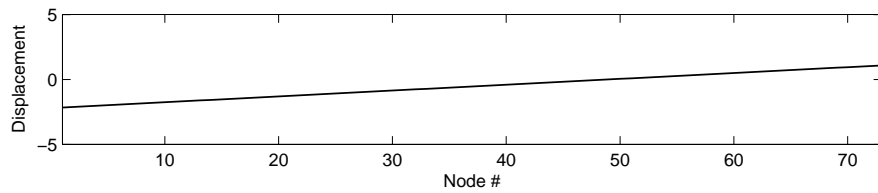
Figure 5.8: Second mode shape of flexible shaft model (2nd rigid body mode)
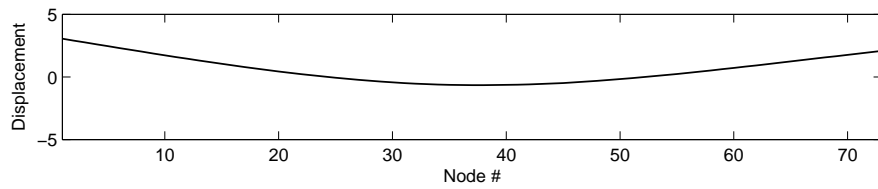
Figure 5.9: Third mode shape of flexible shaft model (1st flexible mode)
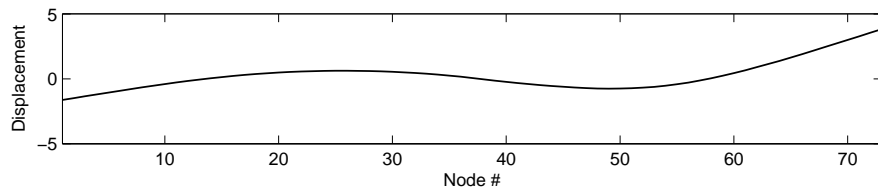
Figure 5.10: Fourth mode shape of flexible shaft model (2nd flexible mode)

45

According to Ryan [62], the equations of motion for a flexible shaft in nodal coordinates can be expressed as:

$$\ddot{\mathbf{q}}_{\mathbf{x}} = -\omega_n^2 \mathbf{q}_{\mathbf{x}} + \Omega \boldsymbol{\Gamma} \dot{\mathbf{q}}_{\mathbf{y}} + \mathbf{F}_{\mathbf{horiz}} \tag{5.64}$$

$$\ddot{\mathbf{q}}_{\mathbf{y}} = -\omega_n^2 \mathbf{q}_{\mathbf{y}} - \Omega \boldsymbol{\Gamma} \dot{\mathbf{q}}_{\mathbf{x}} + \mathbf{F}_{\mathbf{vert}} \tag{5.65}$$

The horizontal and vertical forcing is provided by the magnetic bearings and the imbalance:

$$\mathbf{F}_{\mathbf{horiz}} = \phi^T \mathbf{F}_{\mathbf{x}} + \phi^T \mathbf{F}_{\mathbf{x,imb}} \tag{5.66}$$

$$\mathbf{F}_{\mathbf{vert}} = \phi^T \mathbf{F}_{\mathbf{y}} + \phi^T \mathbf{F}_{\mathbf{y,imb}} \tag{5.67}$$

In Equations 5.66 and 5.67, $\mathbf{F}_{\mathbf{x}}$ and $\mathbf{F}_{\mathbf{y}}$ are the forces from the magnetic bearings in horizontal and vertical direction, respectively. Similarly, $\mathbf{F}_{\mathbf{x,imb}}$ and $\mathbf{F}_{\mathbf{y,imb}}$ are the imbalance forces in horizontal and vertical direction as shown in Equations 5.68 and 5.69.

$$\mathbf{F}_{\mathbf{x,imb}} = \psi \Omega^2 \cos\left(\Omega t\right) \tag{5.68}$$

$$\mathbf{F}_{\mathbf{y,imb}} = \psi \Omega^2 \sin\left(\Omega t\right) \tag{5.69}$$

$$\mathbf{F}_{\mathbf{x}} = \mathbf{F}_{\mathbf{x1}} + \mathbf{F}_{\mathbf{x2}} \tag{5.70}$$

$$\mathbf{F}_{\mathbf{y}} = \mathbf{F}_{\mathbf{y1}} + \mathbf{F}_{\mathbf{y2}} \tag{5.71}$$

The force provided by the magnetic bearings can be expressed in terms of the right and left side bearing as shown in Equations 5.72 – 5.75:

$$\mathbf{F_{x1}} = \mu_0 A_{g,1} N_{T,1}^2 \left(\frac{I_{13}^2}{(d_1 - x_{b1})^2} - \frac{I_{14}^2}{(d_1 + x_{b1})^2} \right) \tag{5.72}$$

$$\mathbf{F_{x2}} = \mu_0 A_{g,2} N_{T,2}^2 \left(\frac{I_{23}^2}{(d_2 - x_{b2})^2} - \frac{I_{24}^2}{(d_2 + x_{b2})^2} \right) \tag{5.73}$$

$$\mathbf{F_{y1}} = \mu_0 A_{g,1} N_{T,1}^2 \left(\frac{I_{11}^2}{(d_1 - y_{b1})^2} - \frac{I_{12}^2}{(d_1 + y_{b1})^2} \right) \tag{5.74}$$

$$\mathbf{F_{y2}} = \mu_0 A_{g,2} N_{T,2}^2 \left(\frac{I_{21}^2}{(d_2 - y_{b2})^2} - \frac{I_{22}^2}{(d_2 + y_{b2})^2} \right) \tag{5.75}$$

In the simulation model, however, the bearing forces ($\mathbf{F_{x1}}$, $\mathbf{F_{x2}}$, $\mathbf{F_{y1}}$ and $\mathbf{F_{y2}}$), are represented by the linearized expression as described in Equations 3.6 and 3.10. The simple model described by Equations 5.64 and 5.65 is now being extended to account for the effects of internal damping and imbalance, as shown in Equations 5.76 and 5.77.

$$\ddot{\mathbf{q}}_\mathbf{x} = -\omega_n^2 \mathbf{q_x} + \Omega \mathbf{C_I} \mathbf{q_y} + \Omega \mathbf{\Gamma} \dot{\mathbf{q}}_\mathbf{y} - \mathbf{C_I} \dot{\mathbf{q}}_\mathbf{x} + \phi^T \mathbf{F_x} + \phi^T \mathbf{F_{x,imb}} \tag{5.76}$$

$$\ddot{\mathbf{q}}_\mathbf{y} = -\omega_n^2 \mathbf{q_y} - \Omega \mathbf{C_I} \mathbf{q_x} - \Omega \mathbf{\Gamma} \dot{\mathbf{q}}_\mathbf{x} - \mathbf{C_I} \dot{\mathbf{q}}_\mathbf{y} + \phi^T \mathbf{F_y} + \phi^T \mathbf{F_{y,imb}} \tag{5.77}$$

The individual coil currents in the magnetic bearings are the sum of the control currents i and bias currents I^B:

$$I_{1k} = I_{1k}^B + i_{1k} \qquad (k = 1\ldots 4) \tag{5.78}$$

$$I_{2k} = I_{2k}^B + i_{2k} \qquad (k = 1\ldots 4) \tag{5.79}$$

The bias currents are assumed to be linearly proportional to the bias voltages:

$$I_{1i}^B = k_{A1} V_{1i}^B \qquad (i = 1 \ldots 4) \tag{5.80}$$

$$I_{2i}^B = k_{A2} V_{2i}^B \qquad (i = 1 \ldots 4) \tag{5.81}$$

In this work it is assumed that the internal damping terms are in the form of $\zeta \omega_i$, therefore the damping matrix takes the form as expressed by Equation 5.82:

$$\mathbf{C_I} = \zeta \begin{bmatrix} \omega_1 & 0 & 0 & 0 \\ 0 & \omega_2 & 0 & 0 \\ 0 & 0 & \omega_3 & 0 \\ 0 & 0 & 0 & \omega_4 \end{bmatrix} \tag{5.82}$$

The simulation model used in this work is described by 20 states which are defined as follows: four states for each mode-shape for the displacement of the shaft in vertical (x) and in horizontal (y) direction as well as their time-derivatives, respectively, and the four augmented current states. The current states are a combination of the bias currents I^B and control currents i.

$$
\mathbf{q} = \begin{Bmatrix} \mathbf{q_x} \\ \mathbf{q_y} \\ \dot{\mathbf{q}}_\mathbf{x} \\ \dot{\mathbf{q}}_\mathbf{y} \\ I_{11}^B i_{11} - I_{12}^B i_{12} \\ I_{13}^B i_{13} - I_{14}^B i_{14} \\ I_{21}^B i_{21} - I_{22}^B i_{22} \\ I_{23}^B i_{23} - I_{24}^B i_{24} \end{Bmatrix}
\tag{5.83}
$$

$$
\mathbf{q_\bullet} = \begin{Bmatrix} q_{\bullet,1} \\ q_{\bullet,2} \\ q_{\bullet,3} \\ q_{\bullet,4} \end{Bmatrix}
\qquad
\dot{\mathbf{q}}_\bullet = \begin{Bmatrix} \dot{q}_{\bullet,1} \\ \dot{q}_{\bullet,2} \\ \dot{q}_{\bullet,3} \\ \dot{q}_{\bullet,4} \end{Bmatrix}
\tag{5.84}
$$

The individual coil currents are not observable. Following the results from [18], with some manipulation the system can be made observable by introducing the new current state variables:

$$
\begin{aligned}
I_{11}^B i_{11} - I_{12}^B i_{12} \\
I_{13}^B i_{13} - I_{14}^B i_{14} \\
I_{21}^B i_{21} - I_{22}^B i_{22} \\
I_{23}^B i_{23} - I_{24}^B i_{24}
\end{aligned}
\tag{5.85}
$$

The amplifiers providing current to the active magnetic bearings are of pulse width modulation (PWM) type with moderate bandwidth. Each power amplifier's bandwidth is dependent upon the actual airgap between the rotor and electromagnet surface in a particular direction. Based on earlier experimental results [18] the coil and amplifier dynamics can be described by Equations 5.86 - 5.93.

$$\frac{di_{11}}{dt} = (k_{A,1}v_{11} - i_{11})(h_1 - h_2(d_1 - y_{b1})) \tag{5.86}$$

$$\frac{di_{12}}{dt} = (k_{A,1}v_{12} - i_{12})(h_1 - h_2(d_1 + y_{b1})) \tag{5.87}$$

$$\frac{di_{13}}{dt} = (k_{A,1}v_{13} - i_{13})(h_1 - h_2(d_1 - x_{b1})) \tag{5.88}$$

$$\frac{di_{14}}{dt} = (k_{A,1}v_{14} - i_{14})(h_1 - h_2(d_1 + x_{b1})) \tag{5.89}$$

$$\frac{di_{21}}{dt} = (k_{A,2}v_{21} - i_{21})(h_1 - h_2(d_2 - y_{b2})) \tag{5.90}$$

$$\frac{di_{22}}{dt} = (k_{A,2}v_{22} - i_{22})(h_1 - h_2(d_2 + y_{b2})) \tag{5.91}$$

$$\frac{di_{23}}{dt} = (k_{A,2}v_{23} - i_{23})(h_1 - h_2(d_2 - x_{b2})) \tag{5.92}$$

$$\frac{di_{24}}{dt} = (k_{A,2}v_{24} - i_{24})(h_1 - h_2(d_2 + x_{b2})) \tag{5.93}$$

The displacement of the rotor at the magnetic bearing locations (x_{b1}, y_{b1}, x_{b2} and y_{b2}) can be calculated by using the following coordinate transformations (Equation 5.94):

$$x_{b1} = \sum_{i=1}^{4} \phi_{nb1,i}q_{x,i} \quad y_{b1} = \sum_{i=1}^{4} \phi_{nb1,i}q_{y,i} \quad x_{b2} = \sum_{i=1}^{4} \phi_{nb2,i}q_{x,i} \quad y_{b2} = \sum_{i=1}^{4} \phi_{nb2,i}q_{y,i} \tag{5.94}$$

Similarly, the displacements at the proximity sensors $(x_{p1}, y_{p1}, x_{p2}$ and $y_{p2})$ can be obtained by using the expressions in Equation 5.95:

$$x_{p1} = \sum_{i=1}^{4} \phi_{np1,i} q_{x,i} \quad y_{p1} = \sum_{i=1}^{4} \phi_{np1,i} q_{y,i} \quad x_{p2} = \sum_{i=1}^{4} \phi_{np2,i} q_{x,i} \quad y_{p2} = \sum_{i=1}^{4} \phi_{np2,i} q_{y,i}$$

$$(5.95)$$

The state-space representation of the system, in general, is expressed by Equation 5.96:

$$\frac{d\mathbf{q}}{dt} = \mathbf{Aq} + \mathbf{Bu}$$

$$\mathbf{y} = \mathbf{Cq} \qquad (5.96)$$

The model with no internal damping is described by Equations 5.97 and 5.98. Similarly, the model with internal damping considered is described by Equations 5.99 and 5.100. Due to the nature of the FE analysis the mass matrix $\mathbf{M}$ is normalized so that in Equations 5.97 and 5.99 the inverse of the mass matrix $(\mathbf{M^{-1}})$ is a unit matrix.

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{M^{-1}K} & -\mathbf{M^{-1}D} & \mathbf{M^{-1}K_I} \\ \mathbf{0} & \mathbf{0} & \mathbf{K_A} \end{bmatrix} \qquad (5.97)$$

$$\mathbf{B} = \begin{bmatrix} \mathbf{0} \\ \mathbf{K_B} \end{bmatrix} \qquad (5.98)$$

$$\mathbf{A_C} = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{M^{-1}K_C} & -\mathbf{M^{-1}D_C} & \mathbf{M^{-1}K_I} \\ \mathbf{0} & \mathbf{0} & \mathbf{K_A} \end{bmatrix} \qquad (5.99)$$

$$\mathbf{B_C} = \begin{bmatrix} \mathbf{0} \\ \mathbf{K_B} \end{bmatrix} \tag{5.100}$$

The uncontrolled system (either $\mathbf{A}$ or $\mathbf{A_C}$) is unstable. Therefore, a basic state feedback controller has been developed by changing all the positive real parts of the eigenvalues of $\mathbf{A}$ negative. This step was necessary, since the uncontrolled system is unstable due to the unstable dynamics of the active magnetic bearings. Also, the system represented by $\mathbf{A_C}$ is unstable as well, due to the presence of internal damping terms. The controller designed to stabilize $\mathbf{A}$, however, is unable to counter the effects of internal damping as it has been developed using the system model which did not take internal damping into consideration. The state feedback matrix $\mathbf{K_F}$ is obtained from MATLAB simulation. The model with state feedback is expressed in Equation 5.101. With the state feedback to the original system, the state-space model can be described as

$$\begin{aligned} \frac{d\mathbf{q}}{dt} &= (\mathbf{A} - \mathbf{B}\mathbf{K_F})\,\mathbf{q} \\ \mathbf{u} &= \mathbf{K_F}\mathbf{q} \end{aligned} \tag{5.101}$$

The state feedback matrix designed for the system with no internal damping ($\mathbf{A}$) will be used on the actual system ($\mathbf{A_C}$) which includes the effects of internal damping. Based on the simulation results the state feedback designed for the original system is not able to suppress the instability resulting from the effects of internal damping. For this reason the Adaptive Disturbance Rejection method is being considered to help the controller deal with rotordynamic instability resulting from internal damping.

It has been found that by using the terms which were used to describe internal damping in Equation 5.99, the output matrix can be successfully be laid out to have the desired effect on $\mathbf{G_p}$, as shown in Equation 5.102. As it is readily apparent, the output matrix used in conjunction with $\mathbf{G_p}$ has little resemblance to an actual output, therefore this matrix could be called a 'pseudo'-output matrix.

$$\mathbf{C_G} = \zeta \begin{bmatrix} 0 & 0 & \Omega\omega_3 & \Omega\omega_4 & 0 & 0 & -\Omega\omega_3 & -\Omega\omega_4 \\ 0 & 0 & -\Omega\omega_3 & -\Omega\omega_4 & 0 & 0 & -\Omega\omega_3 & -\Omega\omega_4 \\ 0 & 0 & -\Omega\omega_3 & -\Omega\omega_4 & 0 & 0 & \Omega\omega_3 & \Omega\omega_4 \\ 0 & 0 & \Omega\omega_3 & \Omega\omega_4 & 0 & 0 & \Omega\omega_3 & \Omega\omega_4 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 & \omega_3 & \omega_4 & 0 & 0 & -\omega_3 & -\omega_4 & 0 & 0 & 0 & 0 \\ 0 & 0 & \omega_3 & \omega_4 & 0 & 0 & \omega_3 & \omega_4 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\omega_3 & -\omega_4 & 0 & 0 & \omega_3 & \omega_4 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\omega_3 & -\omega_4 & 0 & 0 & -\omega_3 & -\omega_4 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{5.102}$$

The output matrix ($\mathbf{C_H}$) used with $\mathbf{H_p}$ is shown in Equation 5.103. This matrix appears as a 'normal' output matrix would in this configuration except that it contains both the position and the velocity components. The argument for the selection of output matrices as described above is the following; the effect of internal damping has been acting on the system in an indirect fashion, therefore a similarly indirect method (i.e. indirect output matrix) has been chosen to properly address the unstable eigenvalues in the system matrix. The layout of the $\mathbf{C_G}$ matrix helps drive the gains of $\mathbf{G_p}$ to a form which is very similar to a fixed state-feedback. The values of $\mathbf{C_G}$ were actually determined from the static state-feedback values acting as guidelines.

The effect of static imbalance appears as a direct term in the equations of motion, thus it is sufficient to use a direct output matrix, i.e. one that very closely resembles the original one. The output matrix for the disturbance rejection part is $\mathbf{C_H}$ and is built by combining the position and velocity terms for the original output matrix $\mathbf{C}$.

$$
\mathbf{C_H} = \left[\begin{array}{cccccccc}
\phi_{np1,1} & \phi_{np1,2} & \phi_{np1,3} & \phi_{np1,4} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \phi_{np1,1} & \phi_{np1,2} & \phi_{np1,3} & \phi_{np1,4} \\
\phi_{np2,1} & \phi_{np2,2} & \phi_{np2,3} & \phi_{np2,4} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \phi_{np2,1} & \phi_{np2,2} & \phi_{np2,3} & \phi_{np2,4}
\end{array}\right.
$$

$$
\left.\begin{array}{cccccccccccc}
\phi_{np1,1} & \phi_{np1,2} & \phi_{np1,3} & \phi_{np1,4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \phi_{np1,1} & \phi_{np1,2} & \phi_{np1,3} & \phi_{np1,4} & 0 & 0 & 0 & 0 \\
\phi_{np2,1} & \phi_{np2,2} & \phi_{np2,3} & \phi_{np2,4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \phi_{np2,1} & \phi_{np2,2} & \phi_{np2,3} & \phi_{np2,4} & 0 & 0 & 0 & 0
\end{array}\right]
$$

$$(5.103)$$

Verification of sufficient conditions for the flexible shaft model:

As stated before, the product of $\mathbf{C_G B}$ (or $\mathbf{C_H B}$) matrices must be positive definite and there must be no unstable zeros in the open-loop transfer function [74]. The state matrix for this model is $\mathbf{A_C}$ as expressed in Equations 5.99. The input matrix $\mathbf{B}$ is expressed in Equation 5.100. The MATLAB script in Appendix A.11 shows that one of the two conditions are met for this model, as the eigenvalues of both the $\mathbf{C_G B}$ product and the

Name	Value	Dim.		Name	Value	Dim.
$N_{T,1}$	160			$k_{A,1}$	0.45	V/A
$N_{T,2}$	160			$k_{A,2}$	0.45	V/A
$A_{g,1}$	$3.23\mathrm{e}^{-4}$	m^2		$nb1$	54	
$A_{g,2}$	$3.23\mathrm{e}^{-4}$	m^2		$nb2$	34	
d_1	$1.25\mathrm{e}^{-3}$	m		$np1$	51	
d_2	$1.25\mathrm{e}^{-3}$	m		$np2$	30	
h_1	$2.87\mathrm{e}^6$			τ_1	$2.53\mathrm{e}^{-5}$	s
h_2	2700			τ_2	$2.53\mathrm{e}^{-5}$	s
$I_{1\bullet}^B$	0.5	A		$I_{2\bullet}^B$	0.5	A
μ_0	$4\pi\mathrm{e}^{-7}$	H/m		ω_3	39.6	rad/s
$\omega_{1,2}$	0	rad/s		ω_4	95.6	rad/s

Table 5.3: Flexible shaft model parameters

$\mathbf{C_H B}$ are all positive.

$$
\mathbf{C_G B} = \begin{bmatrix} 0.05 & 0 & 0 & 0 \\ 0 & 0.05 & 0 & 0 \\ 0 & 0 & 0.05 & 0 \\ 0 & 0 & 0 & 0.05 \end{bmatrix}
\tag{5.104}
$$

$$
\mathbf{C_H B} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\tag{5.105}
$$

Some zeros of the state-space model built with the $(\mathbf{A_C} - \mathbf{B K_F})$, $\mathbf{B}$ and $\mathbf{C_G}$ matrices are

$\mathbf{0}$, proving that the flexible shaft model does not satisfy the sufficient theoretical conditions.

As shown later, despite the fact that the model does not pass these requirements, it performs well in the face of persistent disturbance and internal damping induced instability.

Chapter 6

Simulation Results

This section presents the simulation results for all three models used in this work. First the simple Jeffcott-rotor is being considered with added internal damping (c_i) and static mass imbalance (u). The combined effects of both imbalance and internal damping are also discussed. It is shown how the ADR control scheme can be utilized to suppress synchronous vibrations and rotordynamic imbalance at the same time. Next, the same analysis is presented for the rigid rotor with a rim attached to it by a flexible hub in Section 6.2. In that model, the flexible hub simulates the effect of internal damping with added static mass imbalance. Finally, a flexible shaft model is considered with a mass imbalance and internal damping in Section 6.3. A fixed-gain controller was also developed for each model to better illustrate the effectiveness of the adaptive controller.

6.1 Jeffcott-rotor model results

First, the model for the Jeffcott-rotor has been developed which is a rigid disk on a flexible shaft. The effect of internal damping was added to the equations of motion as well as the static mass imbalance. As the model presented here is a purely theoretical one, the values for various parameters such as internal damping, mass, etc. were selected to present the argument.

As Figure 6.1 shows the uncontrolled response of the model grows unstable at supercritical speed when internal damping is present. Furthermore, as Figure 6.2 shows the combined effect of internal damping and mass imbalance further complicates the problem

by driving the system unstable even faster. When considering only internal damping, the stabilizing part of the ADR scheme ($\mathbf{G_p}$) drives the response stable as shown on Figure 6.3. To counter the combined effect of mass imbalance and internal damping, both the stabilizing and disturbance rejection parts of the ADR are used, as shown on Figure 6.4 the response is stable. It is shown that the stabilizing part of ADR has successfully dealt with the rotordynamic instability due to internal damping and the disturbance rejection part ($\mathbf{H_p}$) has also reduced the amplitude of the synchronous vibration.
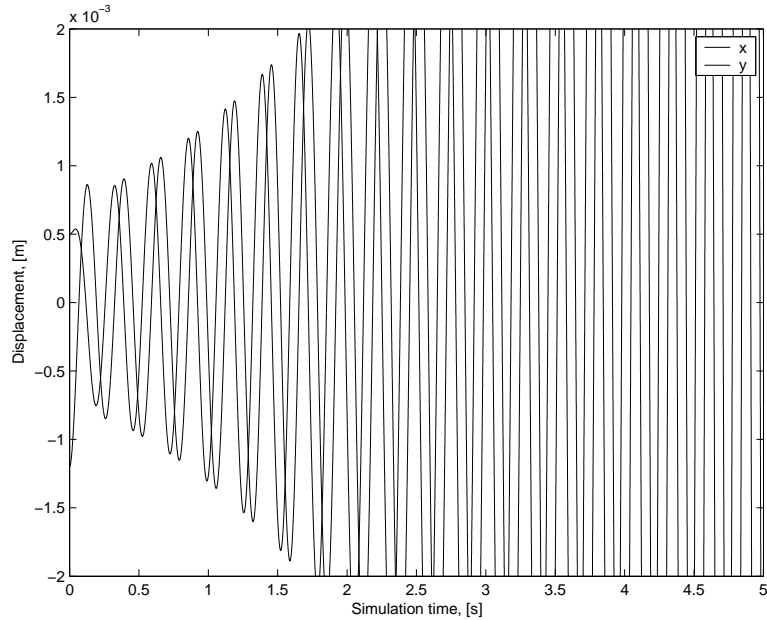
Figure 6.1: Jeffcott-rotor response, Ω=50 rad/s, u=0, Δ_G=0, Δ_H=0

Fixed-gain controller results

The fixed-gain baseline controller has been developed to better illustrate the ADR scheme's effectiveness in face of varying model parameters, such as internal damping, mass imbalance. As expected, the response is stable as shown on Figure 6.5, when the simulation
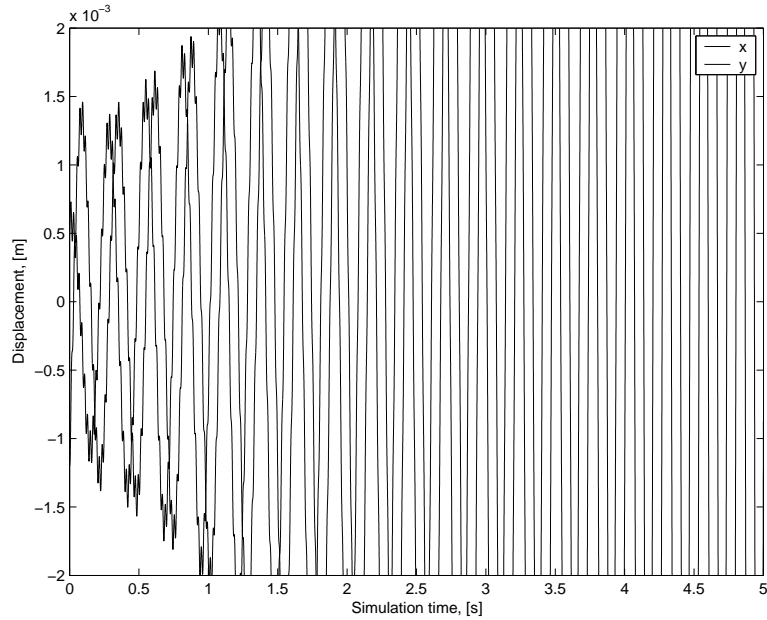
Figure 6.2: Jeffcott-rotor response, Ω=50 rad/s, u=0.005, Δ_G=0, Δ_H=0

is run with the same parameters as the fixed-gain controller is designed for. On the other hand, the controller is not able to stabilize the system when the internal damping is increased (Figure 6.6) or combined with persistent disturbance, such as mass imbalance, as shown on Figure 6.7.

6.2   Flexible hub model results

This section discusses the results observed from the simulation on the model described previously. The simulations were performed with no external disturbance (i.e. mass imbalance) taken into consideration. Thus, the only source of rotordynamic instability was the effect of internal damping. The first simulations were performed to test how a simple PD-type controller would deal with the rotordynamic instability as a result of internal damping.
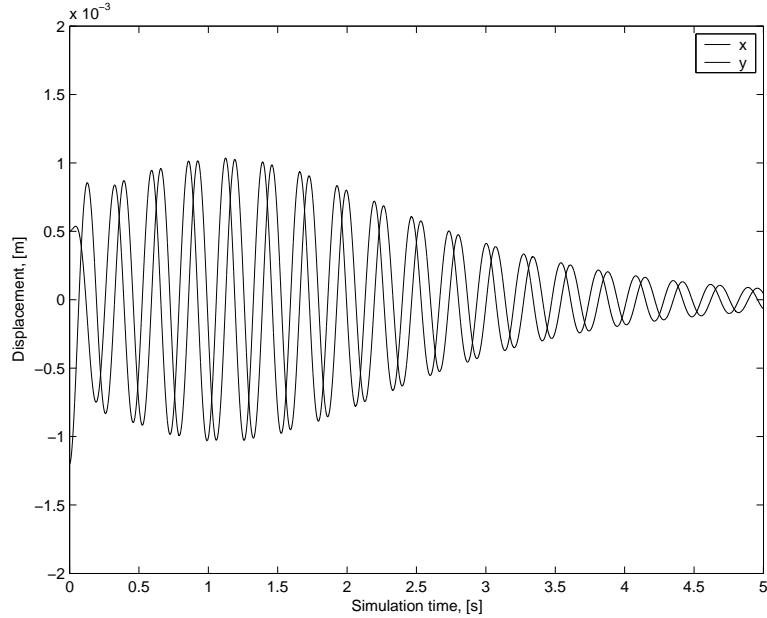
Figure 6.3: Jeffcott-rotor response, Ω=50 rad/s, u=0, Δ_G=5000, Δ_H=0

Therefore, it is first assumed that $\mathbf{H_p} = \mathbf{0}$ and $\Delta_H = \mathbf{0}$. Figure 6.8 shows the results of a test run while the rotor speed is subcritical; in this case the internal damping has a stabilizing effect on the system. The response is stable, as expected from the steady-state solution of Equations 5.28–5.31.

Internal damping induced instability with no external disturbance

According to simulations the critical speed of this model with the presented parameters is expected around the Ω=70 rad/s speed as shown on Figure 5.5. While the rotor speed is below the critical value, the response is stable, as shown on Figure 6.8. Increasing running speed Ω results in longer settling time and eventually unstable behavior as the critical speed is exceeded. The rotordynamic instability shown on Figure 6.9 takes effect and the controller is not able to compensate for the increasing destabilizing forces. The
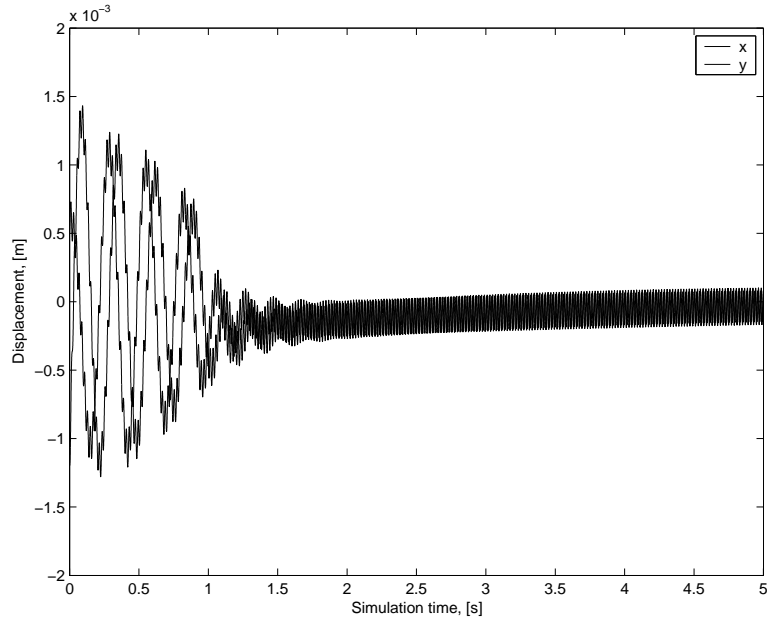
Figure 6.4: Jeffcott-rotor response, Ω=50 rad/s, u=0.005, Δ_G=5000, Δ_H=500

results on the actual experiments would result in severe mechanical damage to the structure. As shown on Figure 6.9 the plant response becomes unstable at Ω=80 rad/s due to the internal damping induced destabilizing forces. Using the adaptive controller described above combined with the reduced order state estimator the response becomes stable and balanced despite the destabilizing forces. By choosing an appropriate value for Δ_G the adaptive disturbance rejection scheme is successfully rejecting the destabilizing effects. Figure 6.10 shows the stable response which was achieved with Δ_G=500. The settling time is longer than with Δ_G=1500 (see Figure 6.8), although with proper adjustment of Δ_G this can be fine-tuned, striking a balance between slower stabilization (with smaller Δ_G) or shorter settling time (using higher Δ_G value), which is, on the other hand, limited by the physical limitations of the plant (i.e. current amplifier and proximity sensor bandwidth, amplifier
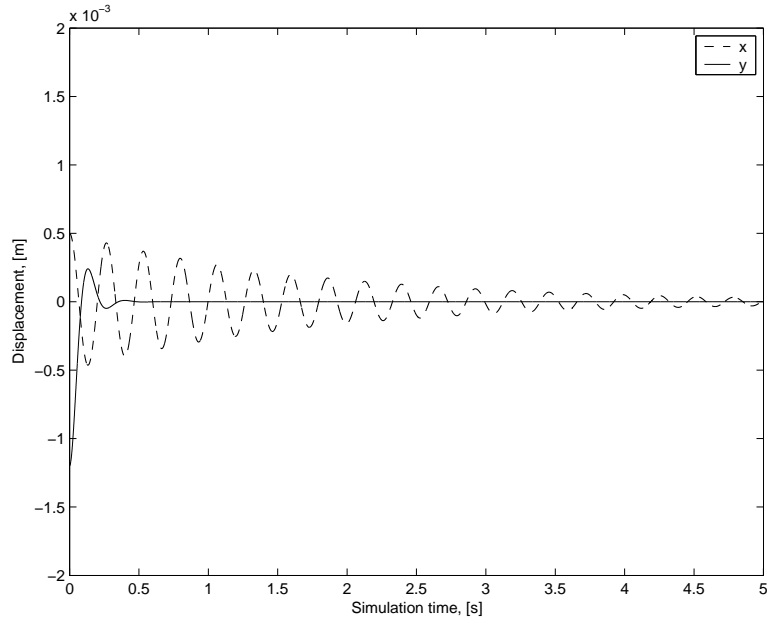
Figure 6.5: Jeffcott-rotor model, fixed gain, $\Omega=50$ rad/s, $u=0$

slew rate, etc.). Increase of Δ_G results in shorter settling time, as shown on Figure 6.11. The suppression of the internal damping induced instability with no external disturbance represents a meaningful finding.   In order to fully take advantage of ADR, the effect of external disturbances, such as mass imbalance should also be considered. This combined problem is presented in the next section.

Internal damping induced instability with persistent synchronous disturbance

In this section the effect of a persistent disturbance is discussed and the relevant simulation results are being presented. The simulation model is the same one used in the previous subsection. In this case, however, a static mass imbalance is also considered, which results in synchronous sinusoidal disturbance. A series of simulations were performed to assess this

62
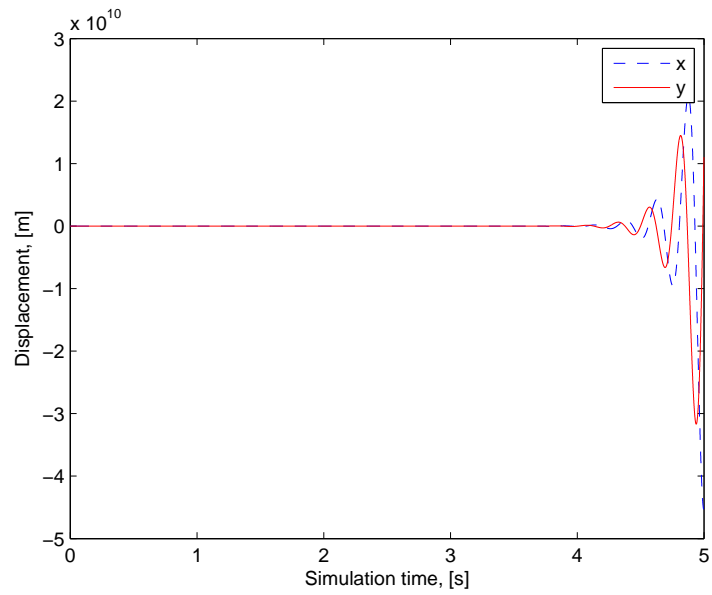
Figure 6.6: Jeffcott-rotor model, fixed gain, increased c_i, Ω=50 rad/s, u=0
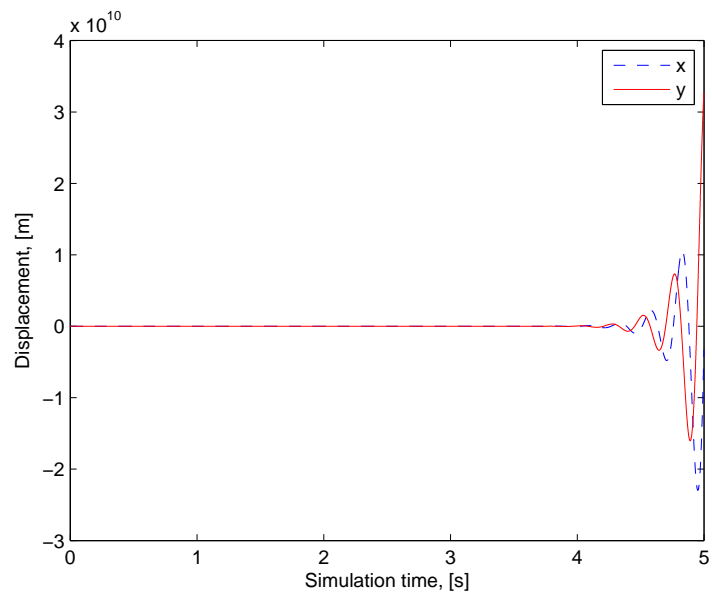
Figure 6.7: Jeffcott-rotor model, fixed gain, Ω=50 rad/s, u=0.005

Figure 6.8: Flexible hub model response at Ω=60 rad/s, u=0, Δ_G=0, Δ_H=0

Figure 6.9: Flexible hub model response at Ω=80 rad/s, u=0, Δ_G=0, Δ_H=0

Figure 6.10: Flexible hub model response at Ω=80 rad/s, u=0, Δ_G=500, Δ_H=0

Figure 6.11: Flexible hub model response at Ω=80 rad/s, u=0, Δ_G=1500, Δ_H=0

65

method's ability to deal with the synchronous disturbance in the plant. The mass imbalance u is located at the midspan of the shaft. Figure 6.12 shows the effect of the mass imbalance on a system with Ω=70 rad/s, i.e. in the sub-critical speed region. Next, on Figure 6.13 the response is shown at Ω=80 rad/s (supercritical region), with the ADR scheme inactive, that is, with Δ_G=0 and Δ_H=0. The unstable response is due to the rotordynamic instability as a result of internal damping. Figure 6.11 shows how the synchronous disturbance is canceled with the ADR scheme by using the $\mathbf{H_p}$ part of the ADR. Note that on Figures 6.14 and 6.15 the ADR controller is activated at 1 second in order to avoid effects from the initial transients. The same method (i.e. $\mathbf{H_p}$ only) is used when the rotor is operating in the supercritical region, as shown on Figure 6.15; the internal damping still drives the response unstable. Figures 6.14 and 6.15 show the disturbance rejection part of the controller is successful in rejecting the synchronous disturbance. Also, Figures 6.10 and 6.11 show that the stabilization component of the controller is also effective. The combined effect of both $\mathbf{G_p}$ and $\mathbf{H_p}$ is shown on Figures 6.16 and 6.17, which reflect stable response even at rotor speeds well above the critical value. Here both internal damping as well as mass imbalance is acting on the model; hence the need for both stabilization and disturbance rejection. The response is stable at the supercritical speed of Ω=80 rad/s and, additionally, the same gains (Δ_G and Δ_H) can be successfully used at even higher speed (Ω=120 rad/s) showing the robustness of the adaptive control scheme.

Fixed-gain controller results

The fixed-gain controller was developed with known model parameters, such as running speed, internal damping. The stable response is shown on Figure 6.18. Increased internal damping is driving the model unstable as the fixed gain controller is not able to stabilize
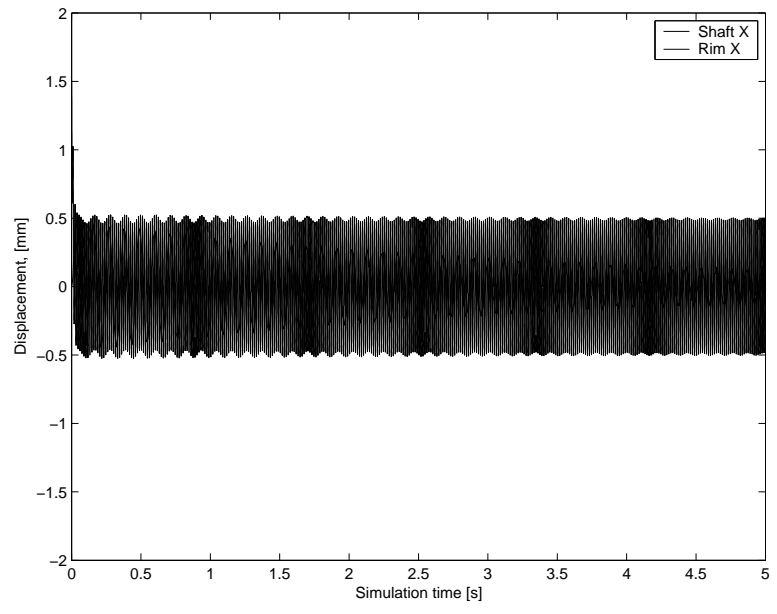
Figure 6.12: Flexible hub model response at Ω=70 rad/s, u=0.005, Δ_G=0, Δ_H=0
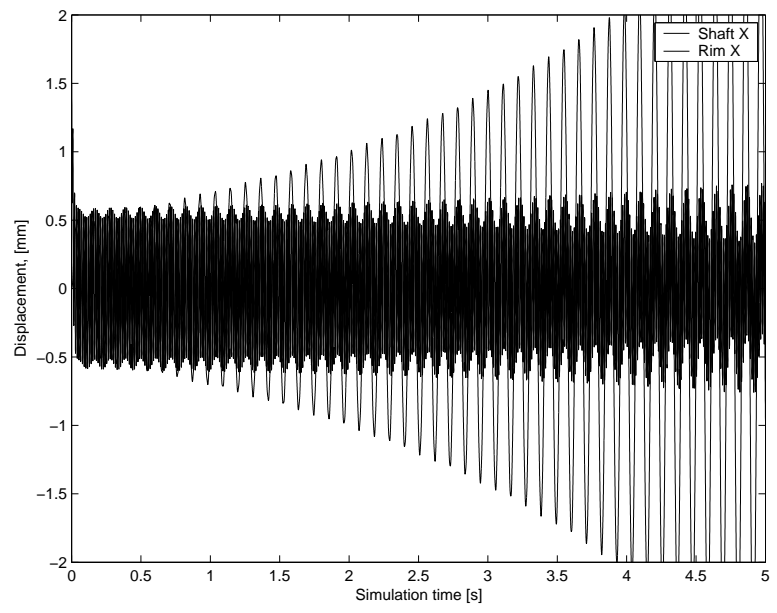
Figure 6.13: Flexible hub model response at Ω=80 rad/s, u=0.005, Δ_G=0, Δ_H=0
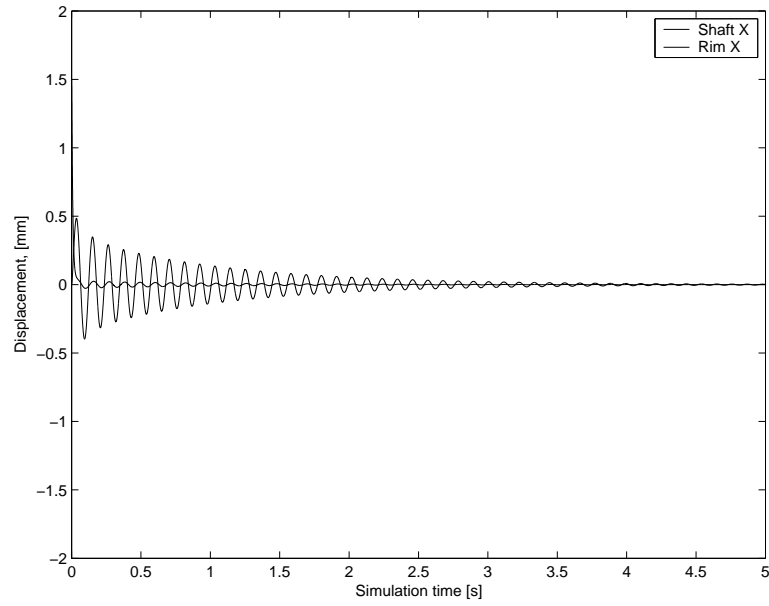
67
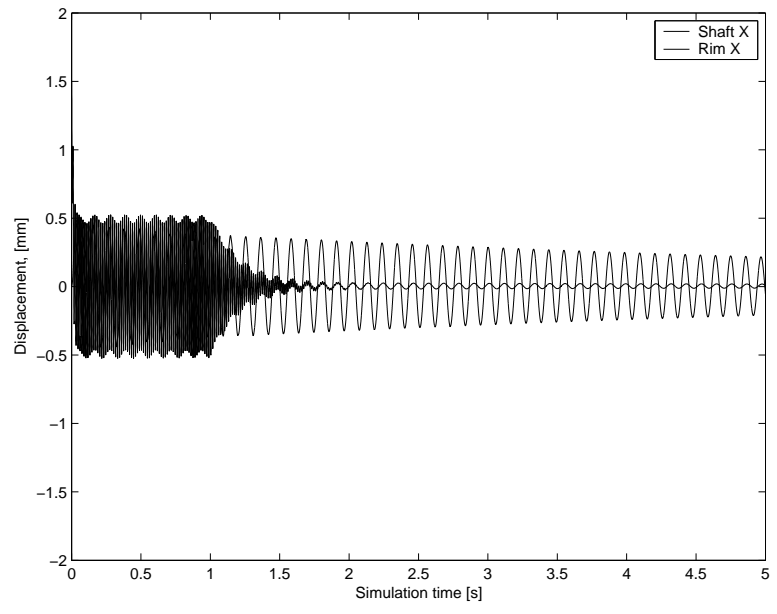
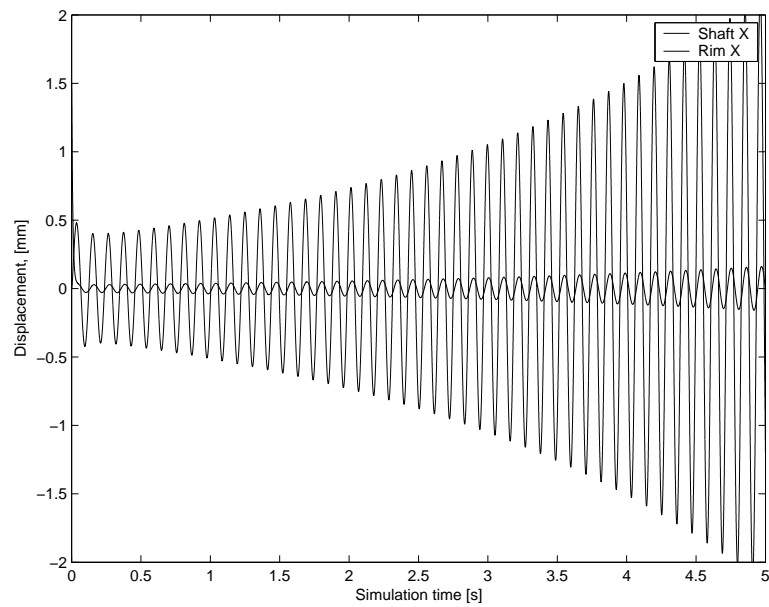Figure 6.14: Flexible hub model response at Ω=70 rad/s, u=0.005, Δ_G=0, Δ_H=100
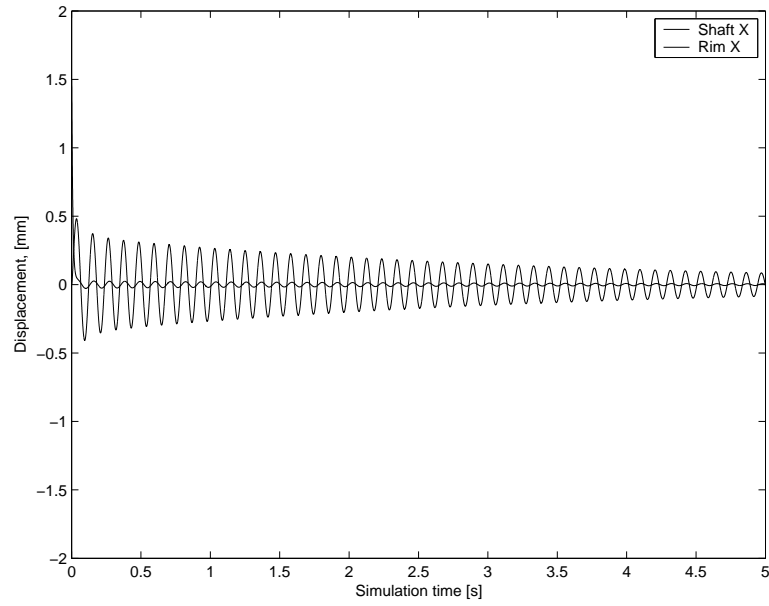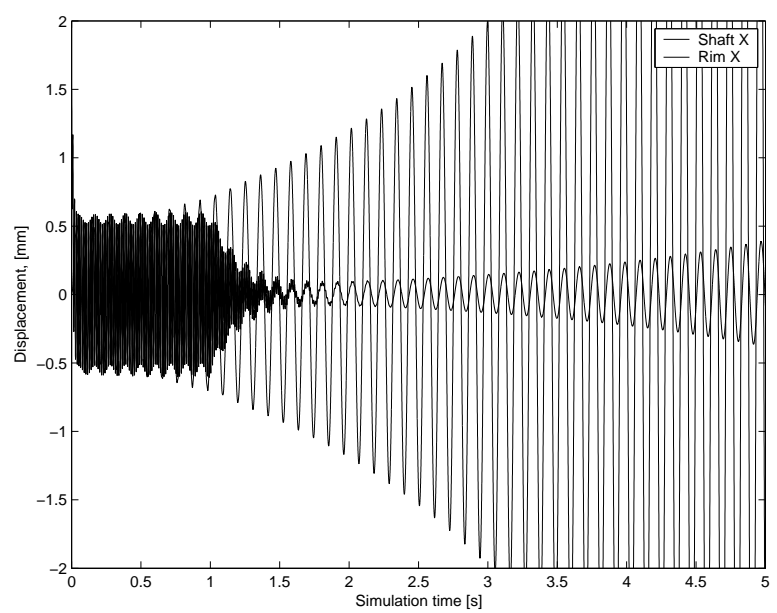

Figure 6.15: Flexible hub model response at Ω=80 rad/s, u=0.005, Δ_G=0, Δ_H=100

Figure 6.16: Flexible hub model response at Ω=80 rad/s, u=0.005, Δ_G=1500, Δ_H=100
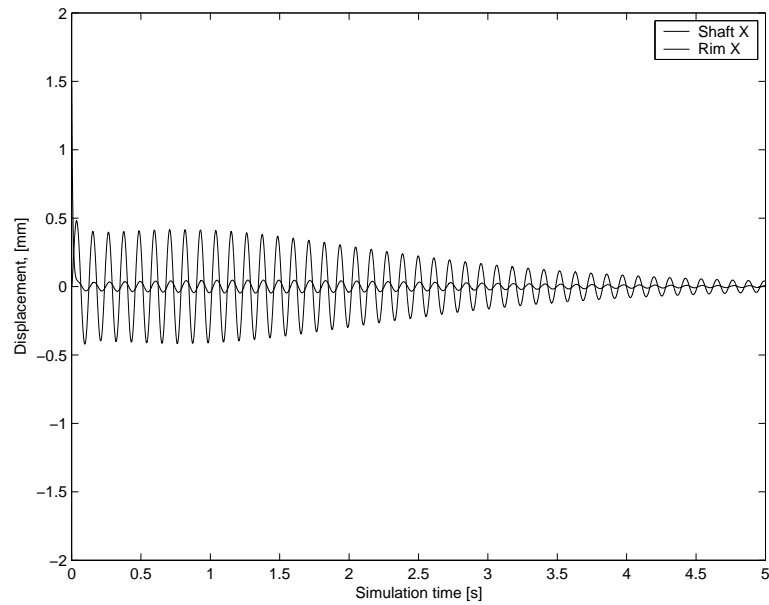

Figure 6.17: Flexible hub model response at Ω=120 rad/s, u=0.005, Δ_G=1500, Δ_H=100
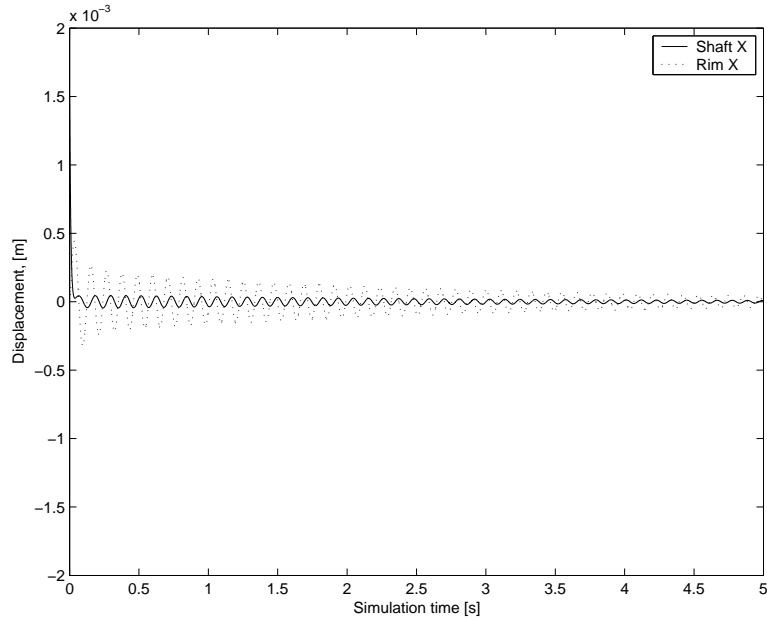
Figure 6.18: Flexible hub model, fixed gain, at Ω=80 rad/s, u=0.0

the response, shown on Figure 6.19. Also, when disturbance is introduced, such as mass imbalance, the fixed-gain controller can't stabilize the system, as shown on Figure 6.20.

6.3 Flexible shaft model results

Internal damping

Figure 6.21 shows the stable response when the rotor speed is below the first critical speed and no internal damping is acting on the model. Similarly, Figure 6.22 also shows a stable response although in this case internal damping is present but the rotor speed is subcritical. Since the rotor speed is below the first critical, the internal damping is not causing rotordynamic instability. As shown on Figure 6.23 the increasing speed enables the internal damping to destabilize the system. In order to counter the rotordynamic
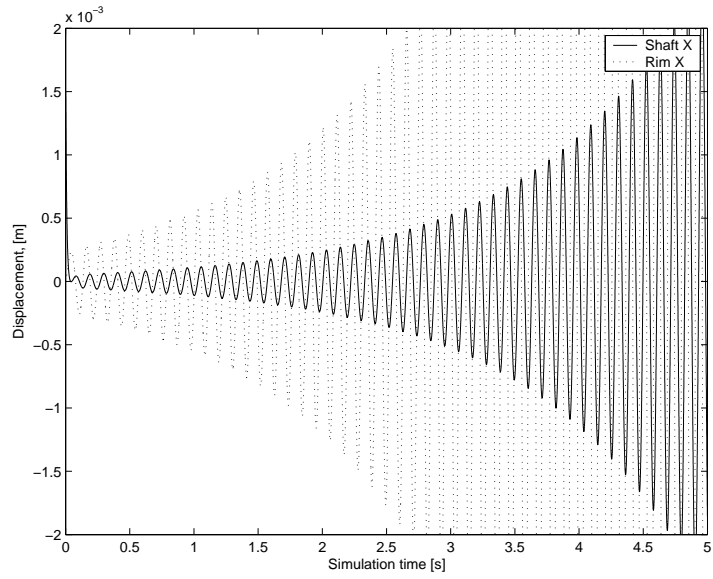
Figure 6.19: Flexible hub model, fixed gain, at Ω=80 rad/s, u=0.0, increased c_H
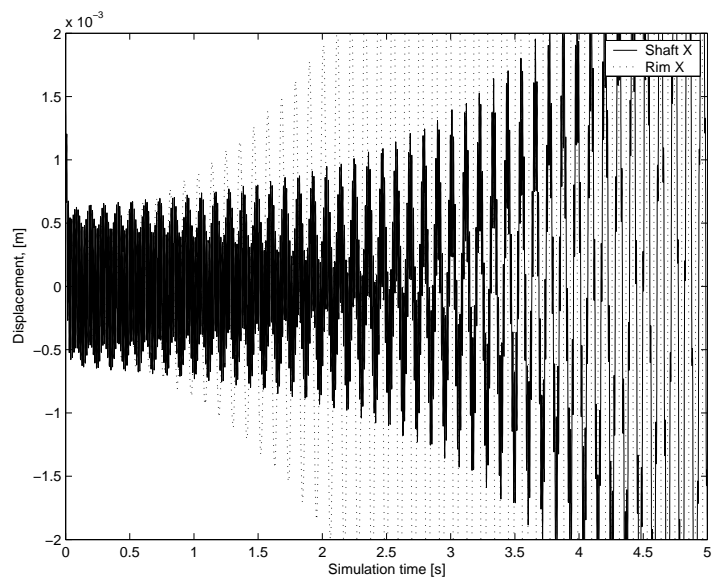
Figure 6.20: Flexible hub model, fixed gain, at Ω=80 rad/s, u=0.005, increased c_H

instability, the ADR scheme has been developed and applied as described earlier. With a choice of Δ_G=50 or Δ_G=100 stable response can be observed, which is shown on Figures 6.24 and 6.25, respectively. The exact value of Δ_G is of secondary importance in this case, as it can be 'fine-tuned' to give the desired response parameters (overshoot, settling time, etc.). In this case the range of appropriate gain is quite large, in the order of magnitudes.

Mass imbalance

Next, the effect of mass imbalance is being considered on the basic rotor system. A constant mass imbalance is simulated. The mass imbalance is located in the bearing mid-span, at node 44. The model is capable of handling several separate mass imbalances, each located at different nodes of the model and at different relative phase angles. In this work a single mass imbalance is being considered.

As it is widely known, the effect of a static mass imbalance is a sinusoidal force of synchronous frequency. The first run of simulation is shown on Figure 6.26 and 6.27 with no adaptive control (Δ_G=0 and Δ_H=0), where the effect of mass imbalance can be observed for various imbalance values. The result is a persistent disturbance which does not decay over time. Figure 6.28 shows the suppressed synchronous response after the ADR disturbance rejection part ($\mathbf{H_p}$) has been activated at Ω=150 rad/s with no internal damping acting on the system.

Mass imbalance and internal damping

As the internal damping terms are included in the model output ($\mathbf{C_x}$ and in turn, $\mathbf{e_y}$, as shown in Eqns. 4.21 to 4.24), it is difficult to separate the two simple cases (i.e. internal damping or mass imbalance alone) therefore the most general case will be considered next. Internal damping (ζ=0.01) and mass imbalance (ϕ_{44}=0.005) is being combined in the next
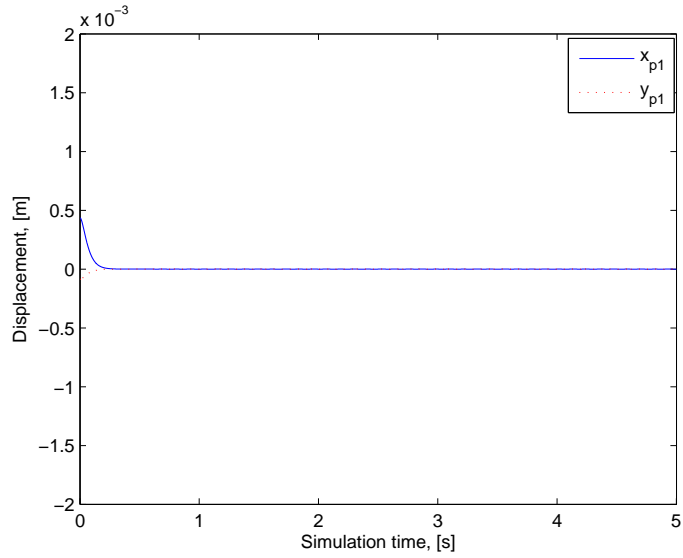
Figure 6.21: Flexible shaft model response at Ω=120 rad/s, ζ=0, u=0, Δ_G=0, Δ_H=0
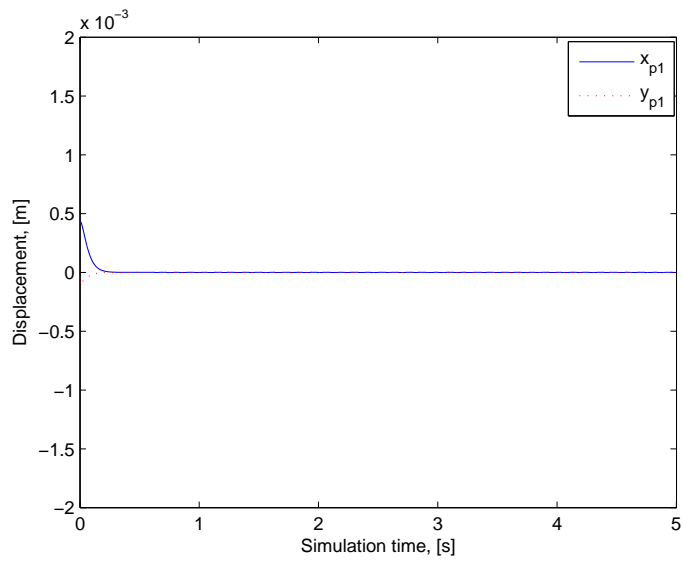
Figure 6.22: Flexible shaft model response at Ω=150 rad/s, ζ=0, u=0, Δ_G=0, Δ_H=0
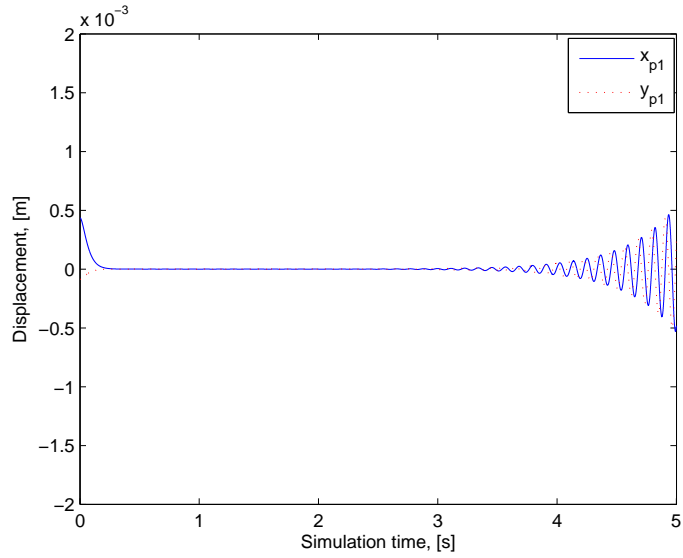
Figure 6.23: Flexible shaft model response at Ω=150 rad/s, ζ=0.01, u=0, Δ_G=0, Δ_H=0
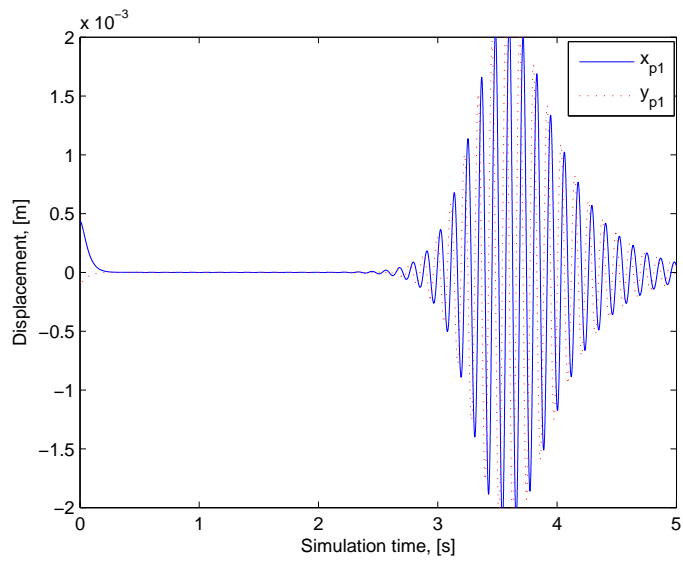
Figure 6.24: Flexible shaft model response at Ω=150 rad/s, ζ=0.01, u=0, Δ_G=50, Δ_H=0

74
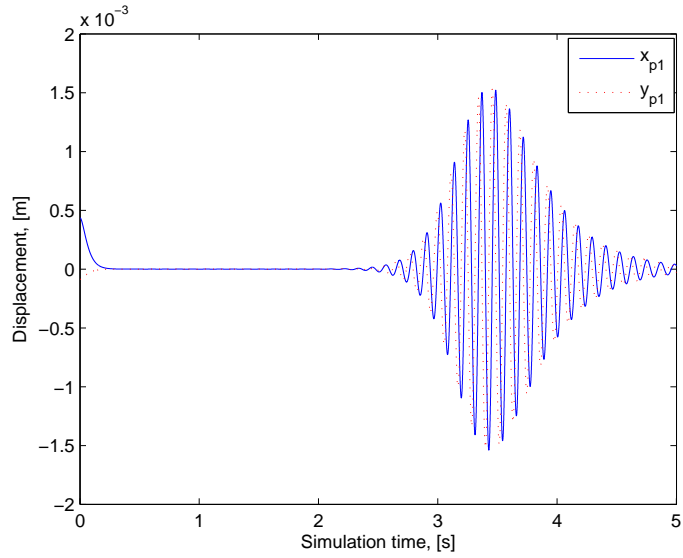
Figure 6.25: Flexible shaft model response at Ω=150 rad/s, ζ=0.01, u=0, Δ_G=100, Δ_H=0
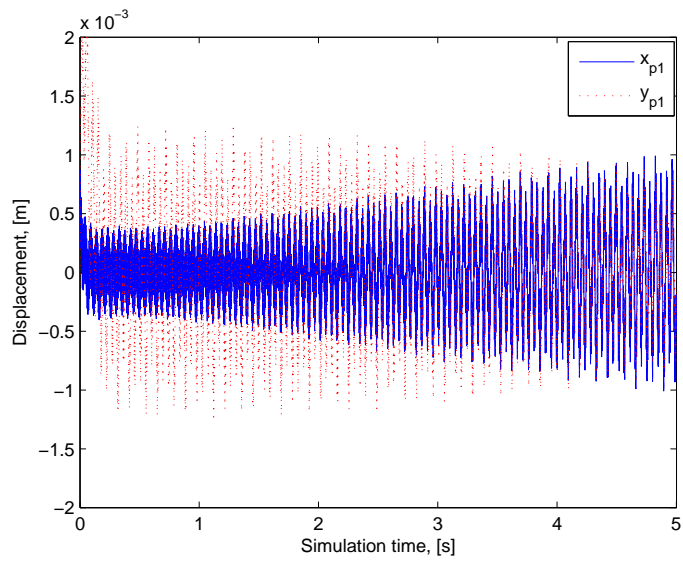
Figure 6.26: Flexible shaft model response at Ω=150 rad/s, ζ=0, u=0.01, Δ_G=0, Δ_H=0

75
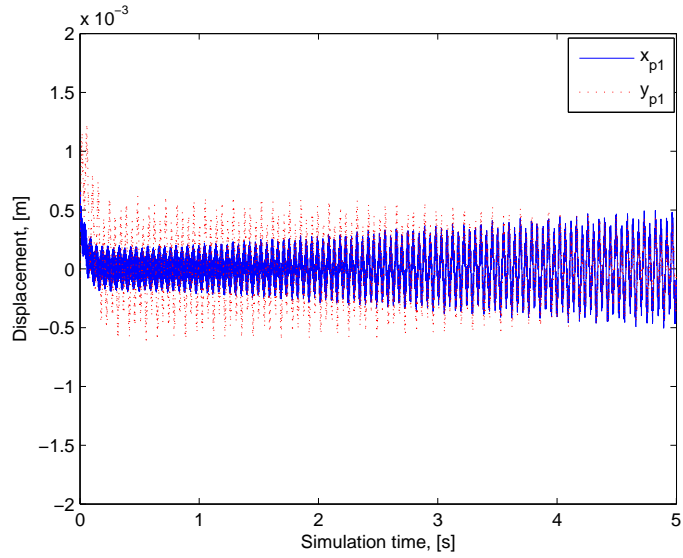
Figure 6.27: Flexible shaft model response at $\Omega=150$ rad/s, $\zeta=0$, $u=0.005$, $\Delta_G=0$, $\Delta_H=0$
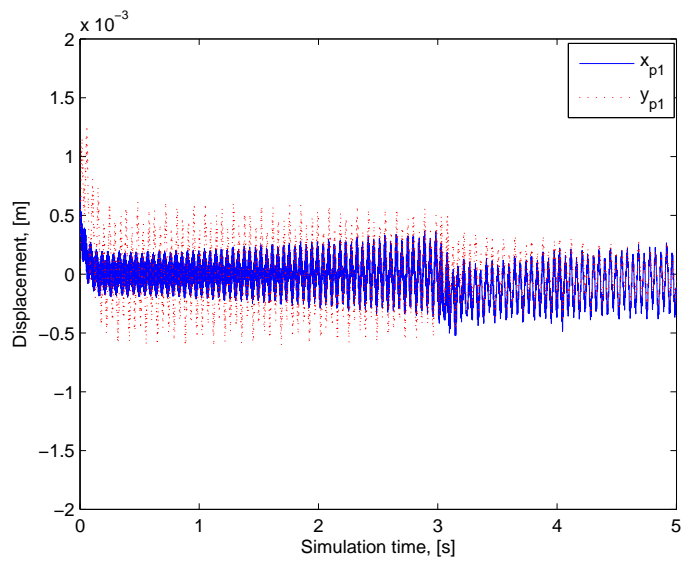
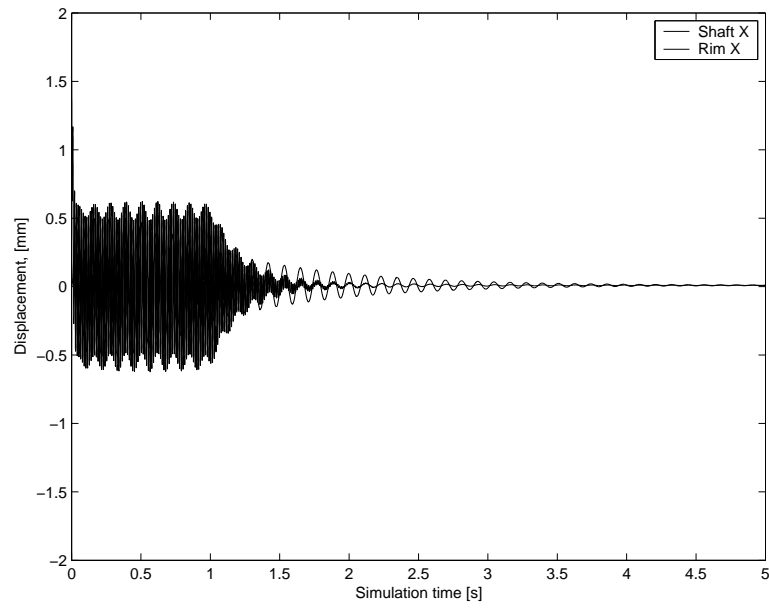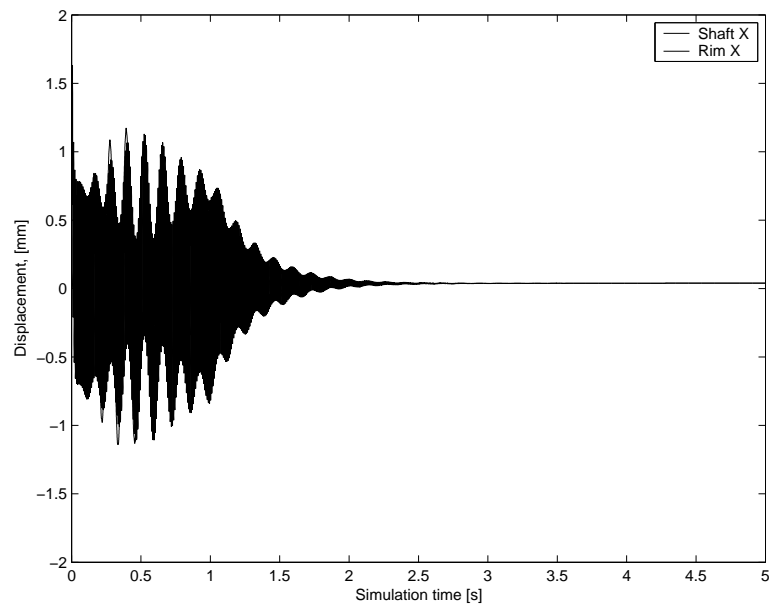Figure 6.28: Flexible shaft model response at $\Omega=150$ rad/s, $\zeta=0$, $u=0.005$, $\Delta_G=0$, $\Delta_H=1000$

76
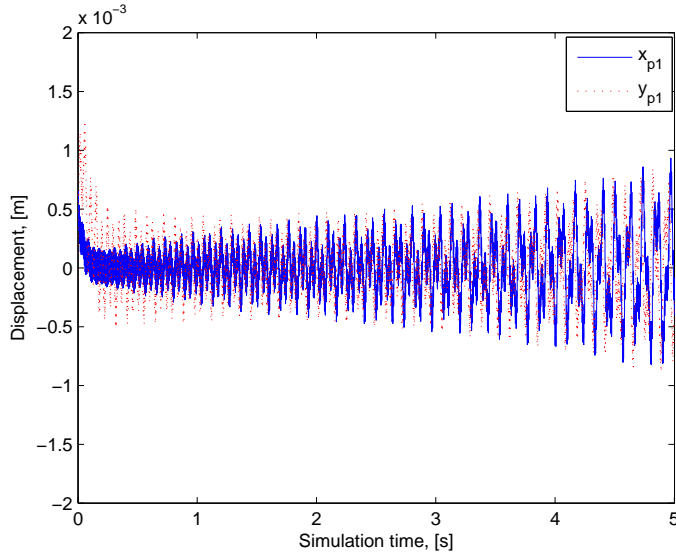
Figure 6.29: Flexible shaft model response at Ω=150 rad/s, ζ=0.01, u=0.005, Δ_G=0, Δ_H=0

series of simulations. As shown on Figure 6.29, the rotordynamic imbalance is very strong when the two effect co-exist; the rotor response grows unstable almost immediately with no adaptive control (Δ_G=0 and Δ_H=0). As Figure 6.30 shows the introduction of both parts of the ADR scheme ($\mathbf{G_p}$ and $\mathbf{H_p}$) helps to reach stable response as well as rejecting some of the persistent disturbances with the proper selection of gains.

Fixed-gain controller results

In order to better emphasize the results from the non-controlled case as well as the adaptive controller, a fixed-gain controller was developed for the flexible shaft model. This output-feedback controller was designed with the internal damping known beforehand. To show how the fixed-gain controller performs in certain scenarios, the internal damping was changed and the controller response is investigated. The fixed-gain controller as developed with the damping ratio ζ=0.01 and the stable response is shown on Figure 6.31. As expected

77
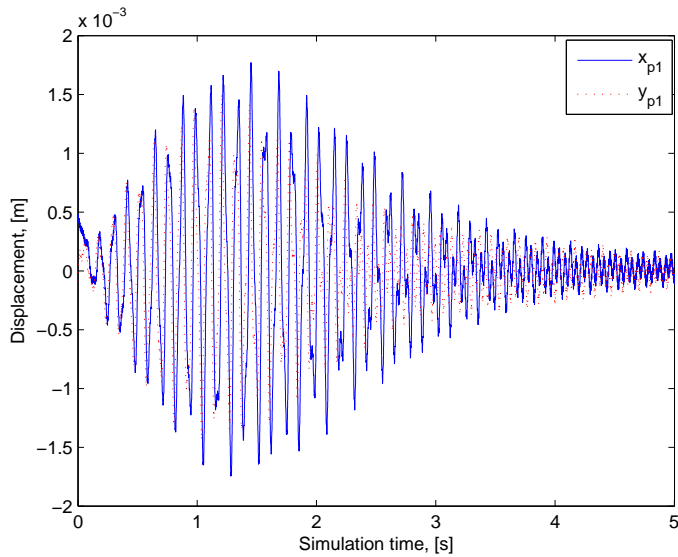
Figure 6.30: Flexible shaft model response at Ω=150 rad/s, ζ=0.01, u=0.001, Δ_G=50, Δ_H=25

the response is stable since the internal damping and the running speed matches the value

this controller was designed for. Figure 6.32 shows the unstable response of the fixed-gain

controller when the internal damping is increased but the gains have not been changed from

the original values. As expected, the response grows unstable after a time. Small increase

in ζ from ζ=0.01 to ζ=0.05 results in unstable response in shorter time. Overall it is shown

that a fixed-gain controller is unable to properly stabilize a system in which the internal

damping is changed; running speed is increased or mass imbalance acting on the system as

shown on Figure 6.33.
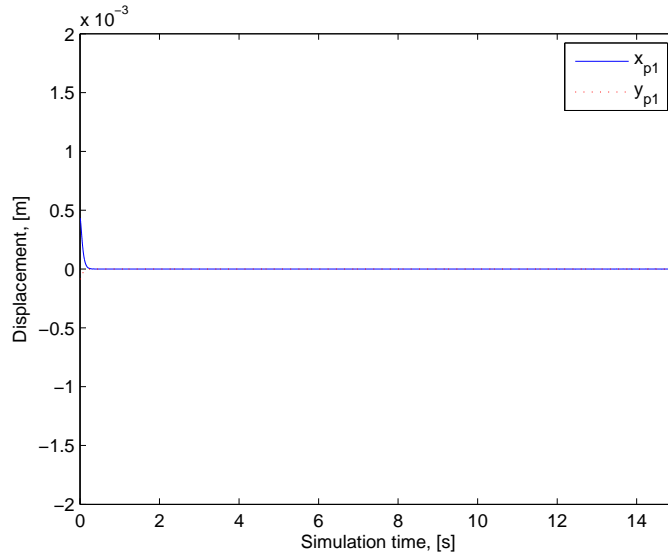
Figure 6.31: Flexible shaft model with fixed gain, Ω=150 rad/s, ζ=0.01, u=0

Figure 6.32: Flexible shaft model with fixed gain, Ω=150 rad/s, ζ=0.05, u=0

79

Figure 6.33: Flexible shaft model with fixed gain, $\Omega=150$ rad/s, $\zeta=0.05$, $u=0.001$

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

The research presented in this dissertation provides an investigation of using adaptive control method to suppress rotordynamic instability and synchronous disturbance. The Adaptive Disturbance Rejection control scheme was used to stabilize rotors with persistent synchronous, sinusoidal disturbance represented by constant mass imbalance. There were three different rotor models developed, each with increasing complexity. First a simple Jeffcott-rotor was investigated, then a rigid rotor with a flexibly connected outer rim and finally a flexible shaft by using finite element analysis results. The internal damping in each of the three models was modeled as a simple viscous damping. The modeled rotors were supported by a pair of radial AMBs described by their linearized model. The power amplifier and circuitry dynamics were also included.

The Jeffcott-rotor model is a relatively simple one with 2 degrees-of-freedom. The ADR scheme was successfully used to stabilize the system as well as to reduce the synchronous vibrations due to the static imbalance. The Jeffcott-rotor model also satisfies the exact conditions for Strict Positive Realness. The second model is a rigid rim attached to a rigid shaft by a flexible hub, as analyzed next. For practical reasons a reduced-order observer was developed to estimate the rim positions. The ADR scheme was successfully used to stabilize the system as well as to reduce the synchronous vibrations due to the static imbalance. This model also satisfies the conditions for Strict Positive Realness. The third model was a flexible shaft represented by a 73-node lumped-mass parameter model where the flexible mode shapes and eigenfrequencies were obtained from finite element analysis. It was found

that by modifying the ADR algorithm this model can be successfully stabilized. As the internal damping acting on the system dynamics in an indirect fashion, the corresponding output matrix had to be shaped in a way which drives $\mathbf{G_P}$ in the required manner. On the other hand, the effect of static mass imbalance is directly tied into the system dynamics, requiring an output matrix laid out similarly to the original one. It was shown that even though the flexible shaft model was not SPR (some zeros in the open-loop transfer function were positive), the model worked well and the ADR scheme was able to stabilize the system and reject the persistent disturbances by using a modified output matrix. This method of output redefinition is very much similar to Matras' earlier works.

A fixed-gain baseline controller has been developed for each model in order to better illustrate the effectiveness of the adaptive method. As expected, the fixed-gain controller is unable to stabilize the system when the model parameters change or external disturbance is introduced.

In order to gain more insight from the application of ADR to rotordynamic instability and synchronous disturbance, additional research and work should be conducted, such as:

- Investigation of the effectiveness of disturbance rejection with mass imbalance in more than one plane, i.e. several masses mounted on the flexible shaft in different node locations and different relative phase angles.

- Experimental verification of the results obtained from the flexible shaft and flexible hub models.

- Development of finite element model to investigate the effectiveness of ADR on real rotor structures assembled from several parts and replacing the cross-coupled stiffness with cross-coupled moments according to the newest observations.

- Investigation the effect of different linear and nonlinear expressions for the internal damping models.

BIBLIOGRAPHY

[1] Adams, R. D. and Maheri, M. R., "Dynamic Flexural Properties of Anisotropic Fibrous Composite Beams", Composite Science Technology, 50(4), pp. 497-514, (1994).

[2] Allaire, P. E., Mikula, A., Banerjee, B. B., Lewis, D. W. and Imlach, J., "Design and Testing of a Magnetic Thrust Bearing", Journal of the Franklin Institute, 326(6), pp. 831-847, (1989).

[3] Aseltine, J. A., Mancini, A. R. and Sarture, C. W., "A Survey of Adaptive Control Systems", IRE Transactions on Automatic Control, 3(6), pp. 102-108, (1958).

[4] Åstrom, K. J. and Wittenmark, B., Adaptive Control, 2nd Ed., Addison-Wesley, Reading, Massachusetts, USA, (1995).

[5] Beams, J. W. and Black, S. A., "Electrically-Driven Magnetically-Supported Vacuum-Type Ultracentrifuge", Review of Scientific Instruments, 10, pp. 59-63, (1939).

[6] Black, H. F., "The Stabilizing Capacity of Bearings for Flexible Rotors with Hysteresis", Transactions of the ASME, pp. 87-91, (1976).

[7] Bleuler, H., "A Survey of Magnetic Levitation and Magnetic Bearing Types", JSME International Journal Series III, 35(3), pp. 335-342 (1992).

[8] Brogan, W. L., Modern Control Theory, Prentice Hall, Englewood Cliffs. (1991).

[9] Chandra, R., Singh, S. P. and Gupta, K., "Damping Studies in Fiber-Reinforced Composites - A Review", Composite Structures, 46, pp. 41-51, (1999).

[10] Chen, C.-T., Linear System Theory and Design, Oxford University Press, (1999).

[11] Childs, Dara, Turbomachinery Rotordynamics, Wiley, pp. 23-28, (1993).

[12] Crandall, S. H. "The Role of Vibration Damping in Vibration Theory", Journal of Sound and Vibration, 11(1), pp. 3-18, (1970).

[13] Crane, R. M. and Gillespie, J. W. Jr., "Characterization of the Vibration Damping Loss Factor of Glass and Graphite Fiber Composites", Composite Science and Technology, 40, pp. 355-375, (1940).

[14] Dyer, S. W., Shi, J., Ni, J. and Shin, K.-K., "Robust Optimal Influence-Coefficient Control of a Multiple-Plane Active Rotor Balancing System", Journal of Dynamic Systems, Measurement and Control, 124(1), pp. 41-46, (2002).

84

[15] Earnshaw, S., "On the Nature of Molecular Forces", Trans. Camb. Phil. Soc., 7, pp. 97-112, (1842).

[16] Engineered Materials Handbook, 1, ASME International Committee, (1987).

[17] Fittro, R. and Knospe, C. R., "μ-Synthesis Control Design Applied to a High Speed Machining Spindle with Active Magnetic Bearings", Proceedings of the 6th International Conference on Magnetic Bearings, (1998).

[18] Flowers, G. T., Szász, G., Trent, G. S. and Greene, M. E., "A Study of Integrally Augmented State Feedback Control for an Active Magnetic Bearings Supported Rotor System", ASME Journal of Engineering for Gas Turbines and Power, 123, pp. 377-382, (2001).

[19] Fuentes, R. J. and Balas, M. J., "Direct Adaptive Rejection of Persistent Disturbances", Journal of Mathematical Analysis and Applications, 251, pp. 28-39, (2000).

[20] Gabrys, C. W. and Bakis, C. E., Design and Testing of Composite Flywheel Rotors, in Composite Materials: Testing and Design, 13, pp. 3-22, American Society for Testing and Materials, Conshohocken, PA, USA, (1997).

[21] Genin, J. and Maybee, J. S., "External and Material Damping in Three-Dimensional Rotor System", International Journal of Non-Linear Mechanics, 5, pp. 287-297, (1970).

[22] Gibson, R. F., Principles of Composite Material Mechanics, McGraw-Hill, (1994).

[23] Gosiewski, Z., "Automatic Balancing of Flexible Rotors, Part I: Theoretical Background", Journal of Sound and Vibration, 100(4), pp. 551-567, (1985).

[24] Gosiewski, Z., "Automatic Balancing of Flexible Rotors, Part II: Synthesis of System", Journal of Sound and Vibration, 114(2), pp. 103-119, (1987).

[25] Gunter, E. J., Dynamic Stability of Rotor-Bearing Systems, NASA Technical Report, SP-113, (1966).

[26] Hirschmanner, M., Steinschaden, N. and Springer, H., "Adaptive Control of a Rotor Excited by Destabilizing Cross-Coupling Forces", Proceedings of the 6th International Conf. on Rotor Dynamics, Sydney, Australia, pp. 38-45, (2002).

[27] Hirschmanner, M. and Springer, H., "Adaptive Vibration and Unbalance Control of a Rotor Supported by Active Magnetic Bearings", Eighth International Symposium on Magnetic Bearings, Mito, Japan, pp. 483-488, (2002).

[28] Hong, S.-K. and Langari, R., "Robust Fuzzy Control of a Magnetic Bearing System Subject to Harmonic Disturbances", IEEE Transactions on Control Systems Technology, 8(2), pp. 366-371, (2002).

[29] Huang, S.-H. and Lin, L.-C., "Fuzzy Dynamic Output Feedback Control with Adaptive Rotor Imbalance Compensation for Magnetic Bearing Systems", IEEE Transactions on Systems, Man and Cybernetics, Part B, 34(4), pp. 1854-1864, (2004).

[30] Humphris, R. R., Kelm, R. D., Lewis, D. W. and Allaire, P. E., "Effect of Control Algorithms on Magnetic Journal Properties", ASME Journal of Engineering for Gas Turbines and Power, 108, pp. 624-632, (1986).

[31] Hung, J. Y., "Magnetic Bearing Control Using Fuzzy Logic", IEEE Transactions on Industry Applications, 31(6), pp. 1492-1497, (1995).

[32] Iannoau, P. A. and Sun, J., Robust Adaptive Control, Prentice Hall, (1996).

[33] Jafri, S., Shrink Fit Effects On Rotordynamic Stability: Experimental And Theoretical Study, Ph.D. Dissertation in Mechanical Engineering, Texas A&M University, (2007).

[34] Jeffcott, H. H., "The Lateral Vibration of Loaded Shafts in the Neighborhood of a Whirling Speed - The Effect of Want of Balance", Philosophical Magazine, Series 6, 37, pp. 304-314, (1919).

[35] Johnson, D., Alternate Operating Modes for Magnetic Bearing Codes, Ph.D. Dissertation, State University on New York at Buffalo (1995).

[36] Johnson, M. E., Nascimento, L. P., Kasarda, M. and Fuller, C. R., "The Effect of Actuator and Sensor Placement on the Active Control of Rotor Unbalance", Journal of Vibration and Acoustics, 125(3), pp. 365-373, (2003).

[37] Kailath, T., Linear Systems, Prentice Hall, Englewood Cliffs. (1980).

[38] Kimball, A. L. Jr., "Internal Friction Theory of Shaft Whirling", General Electric Review, 27(4), pp. 244-251, (1924).

[39] Knospe, C. R., Hope, W., Fedigan, S. J. and Williams, R. D., "Experiments in the Control of Unbalance Response Using Magnetic Bearings", Mechatronics, 5(4), pp. 385-400, (1985).

[40] Knospe, C. R., Hope, R. W., Tamer, S. M. and Fedigan, S. J., "Robustness of Adaptive Unbalance Control of Rotors with Magnetic Bearings", Journal of Vibration and Control, 2, pp. 33-52, (1996).

[41] Knospe, C. R., Hope, R. W., Fedigan, S. J. and Williams, R. D., "A Multi-Tasking DSP Implementation of Adaptive Magnetic Bearing Control", IEEE Transactions on Control Systems Technology, 5, pp. 230-238, (1997).

[42] Lazan, B. J., Damping of Materials and Members in Structural Mechanics, Pergamon Press, (1968).

[43] Linacre, E. "Damping Capacity, Part I: Introduction and Technique of Measurement", Iron and Steel, May, pp. 153-156, (1950).

[44] Linacre, E. "Damping Capacity, Part II: Effects of Conditions of Measurement", Iron and Steel, June, pp. 285-286, (1950).

[45] Luenberger, D., "An Introduction to Observers", IEEE Transactions on Automatic Control, 16(6), pp. 596-602, (1971).

[46] Lum, K.-Y., Coppola, V. T. and Bernstein, D. S., "Adaptive Autocentering Control for an Active Magnetic Bearing with Unknown Mass Imbalance", IEEE Transactions on Control Systems Technology, 5(5), pp. 587-597, (1996).

[47] Lund, J. W., Destabilization of Rotors from Friction in Internal Joints with Micro-slip, International Conference in Rotordynamics, JSME, pp. 487-491, (1986).

[48] Markert, R., Skrica, N. and Zhang, X., "Unbalance Compensation on Flexible Rotors by Magnetic Bearings Using Transfer Functions", Proc. 8th International Symposium on Magnetic Bearings, Mito, Japan, pp. 417-422, (2002).

[49] Maslen, E. H. and Allaire, P. E., "Magnetic Bearing Sizing for Flexible Rotors", Journal of Tribology, 114, pp. 223-229, (1992).

[50] Matras, A., Flowers, G. T., Fuentes, R., Balas, M. and Fausz, J., "Suppression of Persistent Rotor Vibrations Using Adaptive Techniques", Journal of Vibration and Acoustics, 128(6), pp. 682-689, (2006).

[51] Matras, Alex, Implementation of Adaptive Disturbance Rejection Control in an Active Magnetic Bearing System, M.Sc. Thesis, Department of Mechanical Engineering, Auburn University, Auburn, Alabama, USA, (2003).

[52] Matras, Alex, Applied Adaptive Disturbance Rejection Control Using Output Redefinition on Magnetic Bearings, Ph.D. Dissertation, Department of Aerospace Sciences, University of Colorado, Boulder, Colorado, USA, (2005).

[53] Matras, A. and Balas, M. J., "Modification of Adaptive Disturbance Rejection Control for an Active Magnetic Bearing", Proc. of AIAA Conference on Guidance, Navigation and Control, San Francisco, California, (2005).

[54] Moreira, A., Flowers, G. T., Balas, M., Matras, A. and Fausz, J., "The Influence of Internal Damping on the Rotordynamic Stability of a Flywheel Rotor with Flexible Hub", Proceedings of the IDETC/CIE 2005, (2005).

[55] Moreira, A., Characterization and Dynamic Analysis of Damping Effects in Composite Materials for High-Speed Flywheel Applications, Ph.D. Dissertation, Dept. of Mechanical Engineering, Auburn University, Auburn, Alabama, USA, (2005).

[56] Na, H.-S. and Park, Y., "An Adaptive Feedforward Controller for Rejection of Periodic Disturbances", Journal of Sound and Vibration, 201(4), pp. 427-435, (1997).

[57] Newkirk, B. L., "Shaft Whipping", General Electric Review, 27(3), pp. 169-178, (1924).

[58] Nonami, K., DiRusso, E. and Fleming, D. P., "Vibration and Control of Flexible Rotor Supported by Magnetic Bearings", presented at the 1st International Conference on Magnetic Bearings, co-sponsored by the Swiss Federal Institute of Technology and the Swiss Society of Microtechnics, June 6-8, (1988).

[59] Palazzolo, A. B., Jagannathan, S., Montague, G. T., Kascak, A. F. and Király, L. J., "Hybrid Active vibration Control of Rotorbearing Systems using Piezoelectric Actuators", presented at the 12th Biennial Conference on Mechanical Vibration and noise sponsored by the ASME, Sept. 17-20, (1989).

[60] Rankine, W. A., "On the Centrifugal Force of Rotating Shafts", Engineer, 27, p. 249, (1869).

[61] Reinig, K. D. and Desrochers, A. A., "Disturbance Accommodating Controllers for Rotating Mechanical Systems", ASME Journal of Dynamic Systems, Measurement and Control, 108, pp. 24-31, (1986).

[62] Ryan, S. G., "Limit Cycle Vibrations in Turbomachinery", NASA Technical Paper No. 3181, (1991).

[63] Schweitzer, G., Magnetic Bearings, Rotordynamics 2: Problems in Turbomachinery, Edited by N. F. Reiger, pp. 543-570, Springer-Verlag, (1988).

[64] Schweitzer, G., Bleuler, H. and Traxler, A., Active Magnetic Bearings, vdf Hochschulverlag AG, ETH Zurich, Switzerland, (1994).

[65] Shafai, B., Beale, S., LaRocca, P. and Cusson, E., "Magnetic Bearing Control Systems and Adaptive Forced Balancing", IEEE Control Systems Magazine, 14(2), pp. 4-13, (1994).

[66] Shi, J., Zmood, R. and Qin, L., "Synchronous Disturbance Attenuation in Magnetic Bearing Systems Using Adaptive Compensating Signals", Control Engineering Practice, 12(3), pp. 283-290, (2004).

[67] Shida, H., Ichihara, M., Seto, K. and Saigo, M., "Motion and Vibration Control of Flexible Rotor Using Magnetic Bearings", Proc. 8th International Symposium on Magnetic Bearings, Mito, Japan, pp. 381-386, (2002).

[68] Shin, K.-K. and Ni, J., "Adaptive Control of Active Balancing System for Speed-Varying Rotors Using Feedforward Gain Adaptation Technique", Journal of Dynamic Systems, Measurement and Control, 123(3), pp. 346-352, (2001).

[69] Simon, A. and Flowers, G. T., "Non-Singleton Fuzzy Sets for Disturbance Attenuation", International Journal of Acoustics and Vibration, 12(4), pp. 171-178, (2007).

[70] Taylor, H. D., "Shaft Behavior at Critical Speeds", General Electric Review, 32, pp. 194-200, (1929).

[71] Van de Vegte, J., "Continuous Automatic Balancing of Rotating Systems", Journal of Mechanical Engineering Science, 6, pp. 264-269, (1964).

[72] Vance, J. M. Rotordynamics of Turbomachinery, John Wiley and Sons, Inc., (1988).

[73] Vischer, D. and Bleuler, H., "Self-Sensing Active Magnetic Levitation", IEEE Transactions on Magnetics, 29(2), pp. 1276-1281, (1993).

[74] Wen, J., "Time Domain and Frequency Domain Conditions for Strict Positive Realness", IEEE Transactions on Automatic Control, AC-33 (10), pp. 988-992, (1988).

[75] Wettergren, H. L., Rotordynamic Analysis with Special Reference to Composite Rotors and Internal Damping, Ph.D. Dissertation, Dept. of Mechanical Engineering, Linköping University, Linköping, Sweden, (1994).

[76] Williams, R. D., Keith, F. J. and Allaire, P. E., "A Comparison of Analog and Digital Controls for Rotor Dynamic Vibration Reduction through Active Magnetic Bearings", Journal of Engineering for Gas Turbines and Power, 113, pp. 535-543, (1991).

[77] Zhou, S. and Shi, J., "Active Balancing and Vibration Control of Rotating Machinery: A Survey", The Shock and Vibration Digest, 33(5), pp. 361-371, (2001).

[78] Zinoviev, P. and Ermakov, Y., Energy Dissipation in Composite Materials, Technomic Publishing Co. Inc., Pennsylvania, USA, (1994).

Appendices

FLEXIBLE SHAFT MODEL CALCULATION DETAILS

$$\mathbf{K} = \begin{bmatrix} K_1 - \omega_1^2 & K_2 & 0 & 0 & 0 & 0 & 0 & 0 \\ K_3 & K_4 - \omega_2^2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\omega_3^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\omega_4^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & K_5 - \omega_1^2 & K_6 & 0 & 0 \\ 0 & 0 & 0 & 0 & K_7 & K_8 - \omega_2^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\omega_3^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\omega_4^2 \end{bmatrix}$$

$$\mathbf{K_C} = \begin{bmatrix} K_1 - \omega_1^2 & K_2 & 0 & 0 & -\Omega\zeta\omega_1 & 0 & 0 & 0 \\ K_3 & K_4 - \omega_2^2 & 0 & 0 & 0 & -\Omega\zeta\omega_2 & 0 & 0 \\ 0 & 0 & -\omega_3^2 & 0 & 0 & 0 & -\Omega\zeta\omega_3 & 0 \\ 0 & 0 & 0 & -\omega_4^2 & 0 & 0 & 0 & -\Omega\zeta\omega_4 \\ -\Omega\zeta\omega_1 & 0 & 0 & 0 & K_5 - \omega_1^2 & K_6 & 0 & 0 \\ 0 & -\Omega\zeta\omega_2 & 0 & 0 & K_7 & K_8 - \omega_2^2 & 0 & 0 \\ 0 & 0 & -\Omega\zeta\omega_3 & 0 & 0 & 0 & -\omega_3^2 & 0 \\ 0 & 0 & 0 & -\Omega\zeta\omega_4 & 0 & 0 & 0 & -\omega_4^2 \end{bmatrix}$$

$$\mathbf{C} = \begin{bmatrix} \phi_{np1,1} & \phi_{np1,2} & \phi_{np1,3} & \phi_{np1,4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \phi_{np1,1} & \phi_{np1,2} & \phi_{np1,3} & \phi_{np1,4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \phi_{np2,1} & \phi_{np2,2} & \phi_{np2,3} & \phi_{np2,4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \phi_{np2,1} & \phi_{np2,2} & \phi_{np2,3} & \phi_{np2,4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{D} = \begin{bmatrix} \mathbf{0} & \Omega\boldsymbol{\Gamma} \\ -\Omega\boldsymbol{\Gamma} & \mathbf{0} \end{bmatrix}$$

$$\mathbf{D_C} = \begin{bmatrix} -\mathbf{C_I} & \Omega\boldsymbol{\Gamma} \\ -\Omega\boldsymbol{\Gamma} & -\mathbf{C_I} \end{bmatrix}$$

$$\boldsymbol{\Gamma} = \boldsymbol{\Psi}^{\mathbf{T}}\mathbf{I_{nx}}\boldsymbol{\Psi}$$

$$\mathbf{K_I} = \begin{bmatrix} K_{c,1}^1 & 0 & K_{c,2}^1 & 0 \\ K_{c,1}^2 & 0 & K_{c,2}^2 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & K_{c,1}^1 & 0 & K_{c,2}^1 \\ 0 & K_{c,1}^2 & 0 & K_{c,2}^2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{K_A} = \begin{bmatrix} -\frac{k_{A,1}}{\tau_1} & 0 & 0 & 0 \\ 0 & -\frac{k_{A,1}}{\tau_1} & 0 & 0 \\ 0 & 0 & -\frac{k_{A,2}}{\tau_2} & 0 \\ 0 & 0 & 0 & -\frac{k_{A,2}}{\tau_2} \end{bmatrix}$$

$$\mathbf{K_B} = \begin{bmatrix} k_{A,1}\left(I_{1,1}^B + I_{1,2}^B\right) & 0 & 0 & 0 \\ 0 & k_{A,1}\left(I_{1,3}^B + I_{1,4}^B\right) & 0 & 0 \\ 0 & 0 & k_{A,2}\left(I_{2,1}^B + I_{2,2}^B\right) & 0 \\ 0 & 0 & 0 & k_{A,2}\left(I_{2,3}^B + I_{2,4}^B\right) \end{bmatrix}$$

$$\tau_1 = 2\pi\left(h_1 + h_2 d_1\right)$$

$$\tau_2 = 2\pi\left(h_1 + h_2 d_2\right)$$

$$K_{c,1}^i = \frac{2A_{g,1}\mu_0 N_{T,1}^2}{d_1^2}\phi_{nb1,i} \qquad K_{c,2}^i = \frac{2A_{g,2}\mu_0 N_{T,2}^2}{d_2^2}\phi_{nb2,i}$$

$$K_1 = \left(K_{x,1}^{1,1} + K_{x,1}^{2,1}\right)\phi_{nb1,1} + \left(K_{x,2}^{1,1} + K_{x,2}^{2,1}\right)\phi_{nb2,1}$$

$$K_2 = \left(K_{x,1}^{1,1} + K_{x,1}^{2,1}\right)\phi_{nb1,2} + \left(K_{x,2}^{1,1} + K_{x,2}^{2,1}\right)\phi_{nb2,2}$$

$$K_3 = \left(K_{x,1}^{1,2} + K_{x,1}^{2,2}\right)\phi_{nb1,1} + \left(K_{x,2}^{1,2} + K_{x,2}^{2,2}\right)\phi_{nb2,1}$$

$$K_4 = \left(K_{x,1}^{1,2} + K_{x,1}^{2,2}\right)\phi_{nb1,2} + \left(K_{x,2}^{1,2} + K_{x,2}^{2,2}\right)\phi_{nb2,2}$$

$$K_5 = \left(K_{y,1}^{3,1} + K_{y,1}^{4,1}\right)\phi_{nb1,1} + \left(K_{y,2}^{3,1} + K_{y,2}^{4,1}\right)\phi_{nb2,1}$$

$$K_6 = \left(K_{y,1}^{3,1} + K_{y,1}^{4,1}\right)\phi_{nb1,2} + \left(K_{y,2}^{3,1} + K_{y,2}^{4,1}\right)\phi_{nb2,2}$$

$$K_7 = \left(K_{y,1}^{3,2} + K_{y,1}^{4,2}\right)\phi_{nb1,1} + \left(K_{y,1}^{3,2} + K_{y,1}^{4,2}\right)\phi_{nb2,1}$$

$$K_8 = \left(K_{y,1}^{3,2} + K_{y,1}^{4,2}\right)\phi_{nb1,2} + \left(K_{y,2}^{3,2} + K_{y,2}^{4,2}\right)\phi_{nb2,2}$$

$$K_{x,1}^{1,i} = \frac{2A_{g,1}\mu_0 \left(N_{T,1}I_{1,1}^B\right)^2}{d_1^3}\phi_{nb1,i} \qquad K_{y,1}^{3,i} = \frac{2A_{g,1}\mu_0 \left(N_{T,1}I_{1,3}^B\right)^2}{d_1^3}\phi_{nb1,i}$$

$$K_{x,1}^{2,i} = \frac{2A_{g,1}\mu_0 \left(N_{T,1}I_{1,2}^B\right)^2}{d_1^3}\phi_{nb1,i} \qquad K_{y,1}^{4,i} = \frac{2A_{g,1}\mu_0 \left(N_{T,1}I_{1,4}^B\right)^2}{d_1^3}\phi_{nb1,i}$$

$$K_{x,2}^{1,i} = \frac{2A_{g,2}\mu_0 \left(N_{T,2}I_{2,1}^B\right)^2}{d_2^3}\phi_{nb2,i} \qquad K_{y,2}^{3,i} = \frac{2A_{g,2}\mu_0 \left(N_{T,2}I_{2,3}^B\right)^2}{d_2^3}\phi_{nb2,i}$$

$$K_{x,2}^{2,i} = \frac{2A_{g,2}\mu_0 \left(N_{T,2}I_{2,2}^B\right)^2}{d_2^3}\phi_{nb2,i} \qquad K_{y,2}^{4,i} = \frac{2A_{g,2}\mu_0 \left(N_{T,2}I_{2,4}^B\right)^2}{d_2^3}\phi_{nb2,i}$$

$$K_{c,1}^1 = \frac{2A_{g,1}\mu_0 N_{T,1}^2}{d_1^2}\phi_{nb1,1} \qquad K_{c,2}^1 = \frac{2A_{g,1}\mu_0 N_{T,1}^2}{d_1^2}\phi_{nb1,2}$$

$$K_{c,1}^2 = \frac{2A_{g,2}\mu_0 N_{T,2}^2}{d_2^2}\phi_{nb2,1} \qquad K_{c,2}^2 = \frac{2A_{g,2}\mu_0 N_{T,2}^2}{d_2^2}\phi_{nb2,2}$$

MATLAB CODE LISTING FOR `jeff1.m`

```
%
% jeff1.m - simple Jeffcott rotor with internal damping and
% mass imbalance
%
clc;
clear all;
format compact;
format short g;
diary off;

% AMB parameters
N = 330;
Ag = 3.22e-4;
mu0 = 4*pi*1e-7;
d = 2.0e-3;
ib = 1.0;

% AMB position stiffness
Kx = Ag*N*N*ib*ib*mu0/(d*d*d);

% AMB current stiffness
Ki = Ag*N*N*ib*mu0/(d*d);

% mass [kg]
m=10;

%internal damping
%ci=4.4158; % critical value for w=1000; m=1
ci=60.0; % critical value for w=50; m=10
ci=0.0;%5;

% running speed [rad/s]
w=50;

% static imbalance [kgm]
```

```
imb = 0.001;

% sampling frequency [Hz]
ws = 5000;

dt = 1/ws;

t_end = 5.0;
t1 = 0;

steps = t_end/dt;

q = zeros(4,1);

% initial conditions
% % q(1) = 0.5e-3;
% % q(2) = -1.2e-3;

% simple PD controller
Kp= -1000;
Kd= -5;

icx=0;
icy=0;

xx = zeros(steps,1);
yy = zeros(steps,1);
tt = zeros(steps,1);

A=[zeros(2,2), eye(2,2);
    (Kx+(Ki*Kp))/m, -w*ci/m, ((Ki*Kd)-ci)/m,0;
    w*ci/m, (Kx+(Ki*Kp))/m, 0, ((Ki*Kd)-ci)/m];

eig(A)

nd = 3;

b = [0,0; 0,0; 1,0; 0,1];

Gp = zeros(2,2);
Hp = zeros(2,nd);
```

```
% gain for Gp
DG = 0;

% gain for Hp
qq = 500; %%1000; %250; %5000;

DH = qq; %*eye(nd,nd);

% output matrix
Cx = [1,0,1,0; 0,1,0,1];
%Cx = [.01,0,2,0; 0,.01,0,2];
%Cx = [0,0,1,0; 0,0,0,1];
%Cx = [1,0,0,0; 0,1,0,0];
%Cx = [-1,-1,-1,-1; -1,-1,-1,-1];

u=[0;0];

phid = zeros(nd,steps);
imbx = zeros(steps,1);
imby = zeros(steps,1);

wd=w;
%wd=100;

for k=1:steps
    tt(k)=dt*(k-1);
    imbx(k) = w*w*imb*cos(wd*2.0*pi*dt*k);
    imby(k) = w*w*imb*sin(wd*2.0*pi*dt*k);
end;

%wdp=2*wd/(nd-1);
wdp=wd/(nd-1);

for k=1:steps
    for n=1:((nd-1)/2)
        phid(n,k) = cos(((wd)-((n-1)*wdp))*dt*k*2.0*pi);
        phid(((nd-1)/2)+n,k)= sin(((wd)-((n-1)*wdp))*dt*k*2.0*pi);
    end;
    phid(nd,k) = 1.0;
end;

disp('Simulation is running...');
```

```
Fx=0;
Fy=0;
dx_r=0;
dy_r=0;

tic;

cxx=zeros(2,4);
cxx(:,3:4) = ones(2,2);

for k=1:steps

    x_old=q(1);
    y_old=q(2);

    q(3) = dx_r;
    q(4) = dy_r;

    [t,Q] = ode45(@jeff1_ode,[t1, t1+dt],q,[],...
ci,w,m,u,b,Fx,Fy,imbx(k),imby(k));
    q = Q(size(t,1),:)';
    t1 = t1+dt;

    dx_r = q(3);
    dy_r = q(4);

    % manual differentiation for velocity calculation
    q(3)=(q(1)-x_old)/dt;
    q(4)=(q(2)-y_old)/dt;

    % simple PD control for current to AMB
    icx = (Kp*q(1))+(Kd*q(3));
    icy = (Kp*q(2))+(Kd*q(4));

    Fx = (Kx*q(1))+(Ki*icx);
    Fy = (Kx*q(2))+(Ki*icy);

    xx(k)=q(1);
    yy(k)=q(2);

    ey = Cx*q;
```

```
    Gp = Gp+(-ey*(ey')*dt*DG);

    if(t1>1)
        eyy=Cx*q;
%        eyy = [-q(3); -q(4)];
%        eyy = [q(1); q(2)];
%        eyy =cxx*q;
        Hp = Hp+(-eyy*(phid(:,k)')*dt*DH);
    end;

    u=(Gp*ey)+(Hp*phid(:,k));

    if(mod(k,ws)==0)
        disp(k);
    end;
end;

toc;

figure;
plot(tt(1:k),xx(1:k),'b-',tt(1:k),yy(1:k),'r-');
%axis([0 t1 -d d]);
%axis([0 t1 -0.1 0.1]);
%axis([0 t1 -5e-5 5e-5]);
legend('x','y');
title(strcat('\omega=',num2str(w),' \Delta_{G}=',num2str(DG),...
' \Delta_{H}=',num2str(qq)));

disp('done.');

% A=[zeros(2,2), eye(2,2);
%     (Kx+(Ki*Kp))/m, -w*ci/m, ((Ki*Kd)-ci)/m,0;
%     w*ci/m, (Kx+(Ki*Kp))/m, 0, ((Ki*Kd)-ci)/m];
%
% disp('After adaptation:');
% eig(A+(b*Gp*Cx))

%
% the end
%
```

MATLAB CODE LISTING FOR `jeff1ode.m`

```matlab
%
% jeff1_ode.m  dynamics for the Jeffcott-rotor model
%
function dq = jeff1_ode(t,q,ci,w,m,u,b,Fx,Fy,imbx,imby)
dq = zeros(4,1); %#ok<MFAMB>

dq(1) = q(3);
dq(2) = q(4);
dq(3) = ((Fx-(ci*q(3))-(w*ci*q(2)))/m)+imbx;
dq(4) = ((Fy-(ci*q(4))+(w*ci*q(1)))/m)+imby;

% additional input from ADR
dq = dq+(b*u);

return;
%
% the end
%
```

MATLAB CODE LISTING FOR `flexhub10.m`

```
%
% flexhub10.m
%
% Simulation of flexible hub with internal damping
%

clear all;
clc;

% rotational speed [rad/s]
w = 70; %80; % 80 crit.

% disturbance frequency [Hz]
wd = w;

% static imbalance [kg*m]
imb = 0.005;

DG = 1500*1e6;%-1000; %100;%10000;

nd = 3; % number of members of disturbance vector

qq = -100; %-100;

DH = qq; %*eye(nd,nd);

%
% sampling frequency [Hz]
%
%ws = 10000;
ws = 5000;
dt = (1.0/ws);

% simulation time [s]
t_end = 5.0;
```

```matlab
%
% AMB properties
%
N = 330;
Ag = 3.22e-4;
mu0 = 4*pi*1e-7;
d = 2.0e-3;
i_b = 1.0;


%
% linearized AMB parameters
%
%K = Ag*N*N*mu0/4.0;

% position stiffness
Kd = Ag*N*N*i_b*i_b*mu0/(d*d*d);

% current stiffness
Kc = Ag*N*N*i_b*mu0/(d*d);

%
% PD controller gains
%
Kp = 1000;
Kd = 10;

%
% Rotor properties
%
mS = 0.238;
mR = 0.567;
cH = 4.84; %1.84;
kH = 2000; %26994;

q = zeros(8,1);

%
% initial conditions of shaft (xS,yS)
% (for the shaft it should not be larger than the AMB airgap (2 mm)
%
q(1) = 1.5e-3;
```

```
q(2) = -1.0e-3;

%
% initial conditions for the Rim position (xRr,yRr) and for the
% estimated Rim position (xRe,yRe)
%
xRe = q(1);
yRe = q(2);

Gp = zeros(2,2);
%dGp = zeros(2,2);

Hp = zeros(2,nd);
%dHp = zeros(2,nd);

steps = (t_end/dt);

i_cx = 0;
i_cy = 0;

kk = [(-2*Kc/mS), 0; 0,(-2*Kc/mS)];
B = [zeros(4,2); kk; zeros(2,2)];
clear kk;

KK = [Kp,0,0,0,Kd,0,0,0; 0,Kp,0,0,0,Kd,0,0];

A1 = [(-(2*Kd)-kH)/mS, (-w*cH)/mS, kH/mS, (w*cH)/mS;
      (w*cH)/mS,(-(2*Kd)-kH)/mS, (-w*cH)/mS, kH/mS;
      kH/mR,(w*cH)/mR,(-kH)/mR,(-w*cH)/mR;
      (-w*cH)/mR,kH/mR,(w*cH)/mR,(-kH)/mR];

A2 = [-cH/mS,0,cH/mS,0;
      0,-cH/mS,0,cH/mS;
      cH/mR,0,-cH/mR,0;
      0,cH/mR,0,-cH/mR];

A = [zeros(4,4), eye(4,4); A1, A2];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% reduced order state-estimator for Rim position and velocity
%
```

```matlab
    AA = (A+(B*KK));

    F = eye(4,4);
    %tmp = min(real(eig(AA)));

    F(1,1) = -1000; %2*tmp;
    F(2,2) = -2000; %2.1*tmp;
    F(3,3) = -3000; %2.2*tmp;
    F(4,4) = -4000; %2.3*tmp;

    L = eye(4,4);

    C = [eye(2,2), zeros(2,6);
         zeros(2,4), eye(2,2), zeros(2,2)];

    %disp('Rank of controllability matrix F|L (min.4):');
    %rank(ctrb(F,L))

    %disp('Eigenvalues of A+(B*KK) (state feedback + linear AMB):');
    eig(A+(B*KK))

    %
    % solve the Lyapunov-equation for T in (T*(A+(B*KK)) - F*T = L*C)
    %
    T = lyap(-F,AA,-(L*C));

    %disp('Error in solution of Lyapunov equation (should be small):');
    (T*(AA))-(F*T)-(L*C)

    TB = T*B;

    invCT = inv([C;T]);
    z = zeros(4,1);
    dz = zeros(4,1);

    gp_tmp = [0; 0];
    hp_tmp = [0; 0];

    % set up storage variables for plotting
    DTpos_xS = zeros(steps,1);
    DTpos_yS = zeros(steps,1);
    DTpos_xRr = zeros(steps,1);
```

```
DTpos_yRr = zeros(steps,1);
DTpos_xRe = zeros(steps,1);
DTpos_yRe = zeros(steps,1);
DTgp = zeros(steps,1);
DThp_x = zeros(steps,1);
DThp_y = zeros(steps,1);
DTctr_x = zeros(steps,1);
DTctr_y = zeros(steps,1);
DTphid = zeros(steps,nd);
DTtt = zeros(steps,1);

phid = zeros(nd,steps);

for k=1:steps
    tmpx = wd*2.0*pi*dt*k;
    imbx(k) = w*w*imb*cos(tmpx);
    imby(k) = w*w*imb*sin(tmpx);
end;
clear tmpx;

wdp=wd/(nd-1);

for k=1:steps
    for n=1:((nd-1)/2)
        phid(n,k) = cos((wd-((n-1)*wdp))*dt*k*2.0*pi);
        phid(((nd-1)/2)+n,k)= sin((wd-((n-1)*wdp))*dt*k*2.0*pi);
    end;
    phid(nd,k) = 1.0;
end;

disp('Simulation is running...');

t1 = 0;

xSdot_real = 0;
ySdot_real = 0;
xRrdot_real = 0;
yRrdot_real = 0;

%
% main simulation loop
%
```

```
tic;

for k=1:steps

    %
    % simple PD control
    %
    i_cx = (Kp*q(1))+(Kd*q(5));
    i_cy = (Kp*q(2))+(Kd*q(6));

    i_cx = i_cx+gp_tmp(1)+hp_tmp(1);
    i_cy = i_cy+gp_tmp(2)+hp_tmp(2);

    Fx = (Kd*q(1))+(Kc*i_cx)+imbx(k);
    Fy = (Kd*q(2))+(Kc*i_cy)+imby(k);

    % restore real velocities for ODE()
    q(5) = xSdot_real;
    q(6) = ySdot_real;
    q(7) = xRrdot_real;
    q(8) = yRrdot_real;

    [t,Q] = ode45(@flexhub_dyn,[t1,t1+dt],q,[],mS,mR,cH,kH,w,Fx,Fy);

    xS_old = q(1);
    yS_old = q(2);
    xRr_old = q(3);
    yRr_old = q(4);

    q = Q(size(t,1),:)';
    t1 = t1+dt;

    xS = q(1);
    yS = q(2);


    xRr = q(3);
    yRr = q(4);

    %
    % velocities obtained from ODE solution
    %
```

```
        xSdot_real = q(5);
        ySdot_real = q(6);
        xRrdot_real = q(7);
        yRrdot_real = q(8);


        %
        % velocities calculated manually from measured positions
        %
        xSdot = (q(1)-xS_old)/dt;
        ySdot = (q(2)-yS_old)/dt;
        xRrdot = (q(3)-xRr_old)/dt;
        yRrdot = (q(4)-yRr_old)/dt;


        dz = (F*z)+(TB*[i_cx;i_cy])+(L*[q(1); q(2); q(5); q(6)]);


        z = z+(dz*dt);


        xRe_old = xRe;
        yRe_old = yRe;


        ztx = invCT*[q(1);q(2);q(5);q(6);z];


        xRe = ztx(3);
        yRe = ztx(4);
%        xRedot = ztx(7);
%        yRedot = ztx(8);
        xRedot = (xRe-xRe_old)/dt;
        yRedot = (yRe-yRe_old)/dt;


        %
        % ADR parts calculated here (xRe and yRe are estimated)
        %
%        if(t1>0.5)
%        Gp = Gp+(((-[xRr; yRr])*[xRr; yRr]')*DG)*dt;
%        gp_tmp = Gp*[xRr; yRr];
        Gp = Gp+(((-[xRe; yRe])*[xRe; yRe]')*DG)*dt;
        gp_tmp = Gp*[xRe; yRe];
%        Gp = Gp+(((-[xSdot; ySdot])*[xSdot; ySdot]')*DG)*dt;
%        gp_tmp = Gp*[xSdot; ySdot];
%        end;


        if(t1>1)
```

```
%     Hp = Hp+((-[xRe; yRe])*(phid(:,k)')*DH)*dt;
%     Hp = Hp+((-[xRr; yRr])*(phid(:,k)')*DH)*dt;
%     Hp = Hp+((-[xS; yS])*(phid(:,k)')*DH)*dt;
      Hp = Hp+((-[xSdot; ySdot])*(phid(:,k)')*DH)*dt;
      hp_tmp = Hp*phid(:,k);
      end;


%     u = (Gp*[xRe; yRe])+(Hp*phid(:,k));

      DTpos_xS(k) = q(1)*1000.0;
      DTpos_yS(k) = q(2)*1000.0;
      % real Rim positions from mechanical model
      DTpos_xRr(k) = q(3)*1000.0;
      DTpos_yRr(k) = q(4)*1000.0;
      % estimated Rim positions from estimator
      DTpos_xRe(k) = xRe*1000.0;
      DTpos_yRe(k) = yRe*1000.0;
      DThp_x(k) = hp_tmp(1);
      DThp_y(k) = hp_tmp(2);
      DTctr_x(k) = i_cx;
      DTctr_y(k) = i_cy;
      DTtt(k) = t1;

      if(mod(k,ws)==0)
          disp(k);
      end;
end;

disp('complete');
beep;

toc;

figure;
%plot(DTtt,DTpos_xS,DTtt,DTpos_yS,DTtt,DTpos_xRr,DTtt,DTpos_yRr);
plot(DTtt,DTpos_xS,DTtt,DTpos_xRr);
axis([0, t_end, -(d*1000.0), d*1000.0]);
tit = strcat('\Omega=',num2str(w),' \Delta_{G}=',...
    num2str(DG),' \Delta_{H}=', num2str(qq),' n_{d}=',num2str(nd),...
    ' u=',num2str(imb));
title(tit);
ylabel('Displacement, [mm]');
```

```
xlabel('Simulation time [s]');
legend('Shaft X','Rim X');

%
% the end
%
```

MATLAB CODE LISTING FOR `flexhubdyn.m`

```
%
% Rigid shaft and flexibly attached rim equations of motion
%

function qdot = flexhub_dyn(t,q,mS,mR,cH,kH,w,Fx,Fy)

qdot = zeros(8,1);

%
% 4-DOF model, only lateral movement; assumes symmetric AMB operation
%

a1 = ((q(5)-q(7))+(w*(q(2)-q(4))));
a2 = ((q(6)-q(8))-(w*(q(1)-q(3))));

qdot(1) = q(5);
qdot(2) = q(6);
qdot(3) = q(7);
qdot(4) = q(8);
qdot(5) = ((kH*(q(3)-q(1)))-(2.0*Fx)-(a1*cH))/mS;
qdot(6) = ((kH*(q(4)-q(2)))-(2.0*Fy)-(a2*cH))/mS;
qdot(7) = ((kH*(q(1)-q(3)))+(a1*cH))/mR;
qdot(8) = ((kH*(q(2)-q(4)))+(a2*cH))/mR;

return;
%
% the end
%
```

MATLAB code listing for `FlexDamp.m`

```
%
% George T. Flowers and Andras Simon
%
% Flex_Damp.m
%
%*************************************************************************
% Program to simulate a flexible rotor with internal damping and
% supported by two radial magnetic bearings.
%*************************************************************************
%
% this script is using the FLEX_SHAFT2 Ansys model and generated files
%

clc;
clear all;
format compact;
format short g;
%
% 1st crit. speed: 39 rad/s
% 2nd crit. speed: 96 rad/s
%
% simulation time [s]
%
t_end = 10;

global a1 ac1 b1;
global nm zeta;

% running speed [rad/s]
global w;

w = 200;
wd = w;

% range multipliers for phid[] components (lower and upper limit)
```

```
pll=0.0; %0.05
plu=1.0; %1.0

% ADR gain
% % D3 = 500*eye(4,4);
DG = 150; % 50; %50;

% number of disturbances taken into account (MUST be odd number)
global nd
nd=3; %101;
%nd=51;
nd=71;

qq=400; %<-----

DH = qq;
%DH = qq*eye(nd,nd);
% for k=1:nd
%     DH(k,k)=DH(k,k)*k;
% end;

ss = 4;

% ADR stabilizing part
Gp = zeros(ss,ss);

% ADR disturbance rejection part
Hp = zeros(ss,nd);

% time when Gp is activated [s]
tgp = 1.0;

% time when Hp is activated [s]
thp = 1.0;

% internal damping ratio (best leave at 1%)
zeta = 0.0;
zeta = 0.01;

%
% setting up state-space model and AMB parameters (Model.m)
%
```

```
Model;

q = zeros((4*nm)+4,1);

%initial bearing positions
q(1) = 0.0005;
q(5) = -0.0001;


pcon = eig(a1);


%
% poles for state f/b (change real parts of eig(a1))
%
%
for k=1:((4*nm)+4)
    if(real(pcon(k))>0)
        pcon(k) = -pcon(k);
    end;
    if(abs(real(pcon(k)))<1.0e-5)
        pcon(k) = 0+(imag(pcon(k))*sqrt(-1));
    end;
end;

% state feedback for model *without* internal damping
Kc = place(a1,b1,pcon);

Kc(1:4,(4*nm)+1:(4*nm)+4) = 0;
%Kc(1:4,11:12)=0;
%Kc(1:4,15:16)=0;

a11 = a1-(b1*Kc);
ac1 = ac-(b1*Kc);
% eig(ac1)
% return;
%
% sampling frequency [Hz]
%
%ws = 10000.0;
ws = 5000.0;

dt = 1.0/ws;
```

```
t1 = 0.0;

steps = (t_end-t1)/dt;
%hp_mat = zeros(ss,nd,steps);
%gp_mat = zeros(ss,ss,steps);

cx1 = [0,0, om(3), om(4), 0,0, om(3), om(4);
       0,0,-om(3),-om(4), 0,0, om(3), om(4);
       0,0,-om(3),-om(4), 0,0,-om(3),-om(4);
       0,0, om(3), om(4), 0,0,-om(3),-om(4)];

cx2 = [0,0, om(3), om(4), 0,0,-om(3),-om(4);
       0,0, om(3), om(4), 0,0, om(3), om(4);
       0,0,-om(3),-om(4), 0,0, om(3), om(4);
       0,0,-om(3),-om(4), 0,0,-om(3),-om(4)];

cx = zeta*[w*cx1,cx2,zeros(4,4)]; % pad it with zeros to fit size
cxx = [w*cx1,cx2,zeros(4,4)];

clear cx1 cx2

Kcxx = zeros(4,20);

% how many imbalances are accounted for
ni = 1;

% imbalance node locations (for each imbalance)
iml = ones(ni,1)*37;
% imbalance values (for each imbalance)
imm = zeros(ni,1);
% imbalance phase shifts (for each imbalance)
imph = zeros(ni,1);

% imbalance matrix
phi_imb = zeros(nn,ni);
imphase = zeros(ni,1);

% node locations where imbalances are located (1 per node)
iml(1) = 37;
%iml(2) = 30;

% individual imbalances [kg-m]
```

```
%imm(1) = 2e-3; %1e-2;
%imm(1) = 1e-1; % for w=10
%imm(1) = 1e-2; % for w=1000
imm(1) = 5e-3; %<-----

% phase shift of imbalances [degrees]
imph(1) = 0;
%imph(2) = 90;

for i=1:ni
    phi_imb(iml(i),i)=imm(i); % place imbalances into imbalance matrix
    imph(i) = imph(i)*(pi/180.0); % convert [degree] -> [rad]
end;

imbtmp = (phi')*phi_imb*w*w;

imbx = zeros(nm,steps);
imby = zeros(nm,steps);
imbal_angle_cos = zeros(ni,steps);
imbal_angle_sin = zeros(ni,steps);
phid = zeros(nd,steps);
tt = zeros(steps,1);

for k=1:steps
    tmp_imb = (wd*2.0*pi*dt*k*ones(ni,1))+imph;
    imbal_angle_cos(:,k) = cos(tmp_imb);
    imbal_angle_sin(:,k) = sin(tmp_imb);
    tt(k) = dt*k;

    % imbalance force in X direction
    imbx(:,k) = imbtmp*imbal_angle_cos(:,k);

    % imbalance force in Y direction
    imby(:,k) = imbtmp*imbal_angle_sin(:,k);
end;

wdp=(plu-pll)*wd/(nd-1);

for k=1:steps
    for n=1:((nd-1)/2)
        tmpx = ((pll*wd)+((n-1)*wdp))*dt*k*2.0*pi;
%        tmpx = ((plu*wd)-((n-1)*wdp))*dt*k*2.0*pi;
```

```matlab
        phid(n,k) = cos(tmpx);
        phid(((nd-1)/2)+n,k)= sin(tmpx);
    end;
    phid(nd,k) = 1.0;
end;
clear tmpx;

%QQ = zeros((4*nm)+4,steps);

ey = zeros(4,1);
eyy = zeros(4,1);
eyyd = zeros(4,1);

x_old = zeros(nm,1);
y_old = zeros(nm,1);
dx_real = zeros(nm,1);
dy_real = zeros(nm,1);
dx_calc = zeros(nm,1);
dy_calc = zeros(nm,1);

pos_bear = zeros(nm,steps);
pos_prox = zeros(nm,steps);

u = zeros(4,1);

cvprox = cprox;
cvprox(:,9:16) = cprox(:,1:8);

cvvprox = zeros(4,20);
cvvprox(:,9:16) = cprox(:,1:8);
%ccc = zeros(20,4);

%cvprox(:,1:8) = zeros(4,8);

disp('Simulation is running...');

tic;

cx3 = zeros(4,20);
cx3(:,17:20)=ones(4,4);

for k=1:steps
```

```matlab
        % save previous positions for manual differentiation
        x_old = q(1:4);
        y_old = q(5:8);

        % restore real velocities for ODE45()
        q(9:12) = dx_real;
        q(13:16) = dy_real;

        [t,Q] = ode45(@flex_ode_L1,[t1, t1+dt],q,[],...
            b1,ac1,u,imbx(:,k),imby(:,k));
        q = Q(size(t,1),:)';

        t1 = t1+dt;

        % save real velocities (from ODE solution)
        dx_real = q(9:12);
        dy_real = q(13:16);

        % calculate actual velocities from position inputs
        % by using simple differentiation
        q(9:12) = (q(1:4)-x_old)/dt;
        q(13:16) = (q(5:8)-y_old)/dt;

        % positions at the prox. sensors
        pos_prox(:,k) = cprox*q;

        % positions at the bearings
        pos_bear(:,k) = cbear*q;

        % ADR stabilizing part
        if(t1>tgp)
            %ey = cprox*q;
            ey = cx*q;
            Gp = Gp+((-ey)*(ey')*dt*DG);
%           gp_mat(:,:,k) = Gp;
        end;

        % ADR disturbance rejection part
        if(t1>thp)
            eyy = cvprox*q; %100
%            eyy = cprox*q; %100
```

```matlab
%          eyy = cvvprox*q;
           Hp = Hp+((-eyy)*(phid(:,k)')*dt*DH);
%           hp_mat(:,:,k) = Hp;
    end;

    % actual control input from ADR for next cycle
    u = (Gp*ey)+(Hp*phid(:,k));

    if(mod(k,ws)==0)
        disp(k);
    end;

%    if(max(pos_prox(:,k))>1 || min(pos_prox(:,k))<-1)
%        disp('Oooopssss...'); % if response grows unstable
%        break;
%    end;

end;

toc;

% plot X and Y positions at each bearing on a single figure
figure;
plot(tt(1:k),pos_prox(1,:),'b-',tt(1:k),pos_prox(3,:),'r-',...
    tt(1:k),pos_prox(2,:),'k-',tt(1:k),pos_prox(4,:),'g-');
tit = strcat('\Omega=',num2str(w),' \Delta_{G}=',...
    num2str(DG),' \Delta_{H}=', num2str(qq),' n_{d}=',num2str(nd),...
    ' u=',num2str(imm(1)));
title(tit);
legend('x_{p1}','x_{p2}','y_{p1}','y_{p2}');
axis([0 t1 -0.002 0.002]);

disp('completed.');

%
% the end
%
```

MATLAB code listing for `Model.m`

```
%
% George T. Flowers and Andras Simon
%
% Model.m
%
%***************************************************************************
% This program sets up the state-space model of the flexible shaft with
% internal damping, various imbalances, supported by two radial AMB.
%
% This routine is called from Flex_Damp.m
%
%***************************************************************************
%
% this script is using the FLEX_SHAFT2 Ansys model and generated files
%

% running speed (rad/s)
global w;

% number of mode shapes from ANSYS solution
global nm;
nm = 4;

% internal damping ratio (best leave at 1%)
global zeta;

% node location of AMB 1 (right)
nb1 = 54;

% node location of AMB 2 (left)
nb2 = 34;

% node location of position sensors for AMB 1 (right)
np1 = 51;
```

```matlab
% node location of position sensors for AMB 2 (left)
np2 = 30;

% number of nodes in ANSYS model
nn = 73;

%
% Simulation model
%
% 4 inputs, 4 outputs, 4 modeshapes
%
% modal displacement matrix
phi = zeros(nn,nm); %#ok<NASGU>

% modal rotation matrix
psi = zeros(nn,nm); %#ok<NASGU>

% array of eigenfrequencies
om = zeros(nm,1);

% mass moment of inertia matrix
ainx = zeros(nn,nn);

% mass matrix
am = zeros(nn,nn);

%
% read ANSYS-generated files
%
massdata = load('massdata2.dat');
phi = load('displace2.dat'); % nodal displacement matrix
psi = load('rotate2.dat'); % nodal rotation matrix
omega = load('omega2.dat'); % critical speeds in Hz

% trim psi and phi to the first 4 modeshapes
phi = phi(:,1:nm);
psi = psi(:,1:nm);

% fix ANSYS results for rigid body modes
omega(1,2) = 0.0;
omega(2,2) = 0.0;
```

```matlab
%************************************************************************
% AMB properties begin here
%************************************************************************

% permeability of air, H/m
mu = 4*pi*1e-7;

% parameters for AMB 2 (left)
ld = 1.25e-3; % airgap [m]
lnt = 160; % number of turns on one leg
laa = 3.2258e-4; % face area [m^2]
lka = 0.45; % amplifier gain (Volt to Amps)
% bias voltages/currents for Left AMB
%lvb = [0.85, 0.2, 0.5, 0.5];
lvb = [0.5, 0.5, 0.5, 0.5];
lib = lvb*lka;
% bearing constant
lk1 = (mu*laa*(2*lnt)^2.0)/4.0;
% amplifier bandwidth constants
lba1 = 2.87*1e6;
lba2 = 2700;
% lk2 is 1/tau (amplifier time constant)
lk2 = 2.0*pi*(lba2+(lba1*ld));
lk3 = lka*lk2;

% elastic stiffness, left AMB, at each coil and both modeshapes
% left AMB, for ib1...ib4, mode 1...nm
lKx = zeros(4,nm);
for j=1:4 % 4 currents top, bottom, right, left
    for i=1:2 %...for rigid body modes only
        lKx(j,i) = (2.0*laa*mu*((lnt*lib(j))^2)/(ld^3))*phi(nb2,i);
    end;
end;

% left AMB current stiffness, for mode 1 and 2
lKi = zeros(nm,1);
for i=1:2
    lKi(i) = (2.0*laa*mu*(lnt^2)/(ld^2))*phi(nb2,i);
end;

% parameters for AMB 1 (right)
rd = 1.25e-3; % airgap [m]
```

```matlab
rnt = 160; %number of turns on one leg
raa = 3.2258e-4; % face area [m^2]
rka = 0.45; % amplifier gain (V->A)
% bias voltages/currents
%rvb = [0.85, 0.2, 0.5, 0.5];
rvb = [0.5, 0.5, 0.5, 0.5];
rib = rvb*rka;
% bearing constant
rk1 = (mu*raa*(2*rnt)^2.0)/4.0;
% amplifier bandwidth constants
rba1 = 2.87*1e6;
rba2 = 2700;
% rk2 is 1/tau (amplifier time constant)
rk2 = 2.0*pi*(rba2+(rba1*rd));
rk3 = rka*rk2;

% elastic stiffness, right AMB, at each coil and both modeshapes
% right AMB, for ib1...ib4, mode 1...nm
rKx = zeros(4,nm);
for j=1:4 % 4 currents top, bottom, right, left
    for i=1:2 %...for rigid body modes only
        rKx(j,i) = (2.0*raa*mu*((rnt*rib(j))^2)/(rd^3))*phi(nb1,i);
    end;
end;

%
% right AMB current stiffness, mode 1 and 2
rKi = zeros(nm,1);
for i=1:2
    rKi(i) = (2.0*raa*mu*(rnt^2)/(rd^2))*phi(nb1,i);
end;

%*************************************************************************
% AMB properties end here
%*************************************************************************
%*************************************************************************
% Setting up A, B and C matrices for state-space model
%*************************************************************************

global a1 ac ac1 b1 cprox cbear; %#ok<NUSED,NUSED>

a = zeros((4*nm)+4,(4*nm)+4);
```

```
a1 = zeros((4*nm)+4,(4*nm)+4);
ac = a1; %#ok<NASGU>
a2 = zeros((4*nm)+4,(4*nm)+4);
a3 = a2;
b1 = zeros((4*nm)+4,4);

% mass matrix (normalized)
mass = eye(2*nm,2*nm);

% stiffness matrix (from AMB Kx)
stiffkx = zeros(2*nm,2*nm);

% stiffness matrix (from AMB Ki)
stiffki = zeros(2*nm,4);

% stiffness matrix associated with internal damping
stiffcc = zeros(2*nm,2*nm);

% damping matrix
damp = zeros(2*nm,2*nm);

for i=1:nm,
%    psi(:,i) = rotate(:,i); % extract psi (nodal rotation matrix)
%    phi(:,i) = displace(:,i); % extract phi (nodal displacement matrix)
     om(i) = 2.0*pi*omega(i,2); % critical speeds in rad/s
end

% mass (am) and inertia (ainx) values at each node location
for i=1:nn,
   am(i,i) = massdata(i,2);
   ainx(i,i) = massdata(i,3);
end

% total mass
%m = trace(am);

%clear omega;
%clear rotate;
%clear displace;
%clear massdata;

% form Gamma matrix
```

```
%
gamm = (psi')*ainx*(psi);
%clear ainx;
%clear psi;

% !!! check signs on gamma terms !!!
damp(1:nm,nm+1:(2*nm)) = -gamm(1:nm,1:nm)*w;
damp(nm+1:(2*nm),1:nm) = gamm(1:nm,1:nm)*w;

%
% stiffness matrix from internal damping
%
for i=1:nm,
    stiffcc(i,i+nm) = w*zeta*om(i); %+
    stiffcc(i+nm,i) = -w*zeta*om(i); %-
end

%
% stiffness matrix assoc. with position stiffness (Kx) of AMB
%
% rKx/lKx summed up in one direction, for both AMBs, rigid modes only
%
for i=1:nm
    for j=1:2
tmp = ((rKx(1,i)+rKx(2,i))*phi(nb1,j));
stiffkx(i,j) = tmp + ((lKx(1,i)+lKx(2,i))*phi(nb2,j));
    tmp = ((rKx(3,i)+rKx(4,i))*phi(nb1,j));
stiffkx(i+nm,j+nm) = tmp + ((lKx(3,i)+lKx(4,i))*phi(nb2,j));
    end;
end;

%
% stiffness matrix assoc. with current stiffness (Ki) of AMB
%
for i=1:nm
    stiffki(i,1) = rKi(i);
    stiffki(nm+i,2) = rKi(i);
    stiffki(i,3) = lKi(i);
    stiffki(nm+i,4) = lKi(i);
end;

stiff2 = stiffkx + stiffcc;
```

```
% build A matrix

for i=1:nm,
    a(i+(2*nm),i)= -(om(i)^2);
    a(i+(3*nm),i+nm) = -(om(i)^2);
end;


a2(1:(2*nm),((2*nm)+1):(4*nm)) = eye(2*nm,2*nm);
a2((2*nm)+1:(4*nm),1:(2*nm)) = inv(mass)*stiffkx(1:(2*nm),1:(2*nm));
a2((2*nm)+1:(4*nm),(2*nm)+1:(4*nm)) = -inv(mass)*damp;
a2((2*nm)+1:(4*nm),(4*nm)+1:(4*nm)+4) = inv(mass)*stiffki;


a3(1:(2*nm),((2*nm)+1):(4*nm)) = eye(2*nm,2*nm);
a3((2*nm)+1:(4*nm),1:(2*nm)) = inv(mass)*stiff2(1:(2*nm),1:(2*nm));
a3((2*nm)+1:(4*nm),(2*nm)+1:(4*nm)) = -inv(mass)*damp;
a3((2*nm)+1:(4*nm),(4*nm)+1:(4*nm)+4) = inv(mass)*stiffki;


for i=1:nm
    a3(i+(2*nm),i+(2*nm))= -zeta*om(i);
    a3(i+(3*nm),i+(3*nm))= -zeta*om(i);
end;

% a1 is the state matrix A *without* internal damping
a1 = a2 + a;

% ac is the state matrix Ac *with* internal damping included
ac = a3 + a;

% Current states
a1((4*nm)+1,(4*nm)+1) = -rk3;
a1((4*nm)+2,(4*nm)+2) = -rk3;
a1((4*nm)+3,(4*nm)+3) = -lk3;
a1((4*nm)+4,(4*nm)+4) = -lk3;

ac((4*nm)+1,(4*nm)+1) = -rk3;
ac((4*nm)+2,(4*nm)+2) = -rk3;
ac((4*nm)+3,(4*nm)+3) = -lk3;
ac((4*nm)+4,(4*nm)+4) = -lk3;

% input matrix B
b1((4*nm)+1,1) = rk3*(rib(1)+rib(2));
```

```
b1((4*nm)+2,2) = rk3*(rib(3)+rib(4));
b1((4*nm)+3,3) = lk3*(lib(1)+lib(2));
b1((4*nm)+4,4) = lk3*(lib(3)+lib(4));
%

% output matrix C with proximitor positions (cprox)
cprox = zeros(4,(4*nm)+4);
for i=1:nm
    cprox(1,i) = phi(np1,i);
    cprox(2,nm+i) = phi(np1,i);
    cprox(3,i) = phi(np2,i);
    cprox(4,nm+i) = phi(np2,i);
end;

% output matrix C with bearing positions (cbear)
cbear = zeros(4,(4*nm)+4);
for i=1:nm
    cbear(1,i) = phi(nb1,i);
    cbear(2,nm+i) = phi(nb1,i);
    cbear(3,i) = phi(nb2,i);
    cbear(4,nm+i) = phi(nb2,i);
end;

%***********************************************************************
% A, B and C matrices are now completed for the state-space model
%***********************************************************************

return;

%
% the end
%
```

MATLAB code listing for flexode.m

```
%
% flex_ode.m - describes the dynamics of the rotor system model
%
% called from Flex_Damp.m
%
function dq = flex_ode(t,q,b1,ac1,u,imbx,imby) %#ok<INUSL>

dq = (ac1*q)+(b1*u);
dq(9:16) = dq(9:16)+[imbx;imby];
return;


%
% the end
%
```

ANSYS CODE LISTING FOR FlexShaft2.inp

```
/COM
/COM Flexible shaft model
/COM
/COM Long, slender steel shaft with two radial AMB and proximitors
/COM

/FILNAME,FLEX_SHAFT2,1

!number of nodes
nodes=73

!number of mode shapes to be calculated
shapes=10

! bearing 1 node
nb1=54

!bearing 2 node
nb2=34

! proximitor 1 node
np1=51

!proximitor 2 node
np2=30

!total shaft length [m]
lenT = 1.5

!distance of nodes
lenS=(lenT/(nodes-1))

! Pi constant
pi=acos(-1)
```

```
! steel density [kg/m^3]
rho = 7800

! shaft radius [m]
rad=0.005

mTot = rad*rad*pi*lenT*rho

*DIM,masses,ARRAY,nodes,3
*DIM,phi_array,ARRAY,nodes,shapes
*DIM,psi_array,ARRAY,nodes,shapes
*DIM,w_vector,ARRAY,shapes,2
*DIM,tmp,ARRAY,nodes,1



*DO,var1,1,nodes,1
  masses(var1,1) = var1
*ENDDO

/PREP7
/TITLE,Flexible shaft with 2 AMB

! Rotor properties

DENS,1,0.0

GXY,1,8.0E10
GYZ,1,8.0E10
GXZ,1,8.0E10

EX,1,2.0E11
EY,1,2.0E11
EZ,1,2.0E11

NUXY,1,0.284
NUYZ,1,0.284
NUXZ,1,0.284

! Elastic beams
ET,1,BEAM3
R,221,7.53828E-5,4.52204E-10,1.0
```

```
! Rigid masses
ET,2,MASS21

! axis of symmetry is X

! solid slender shaft (S)
mS=mTot/(nodes-1)
IxS=(mS*rad*rad)/2.0
IyS=(mS*((4*rad*rad)+(lenS*lenS)))/12.0
IzS=IyS

! elements at shaft ends (E)
mE=mS/2.0
IxE=(mE*rad*rad)/2
IyE=(mE*((4*rad*rad)+(lenS*lenS/4)))/12.0
IzE=IyE

! mass at position sensors (P)
mP=0.1211
IxP=0.2072e-4
IyP=0.1196e-4
IzP=IyP

! mass at magnetic bearings (B)
mB=0.5414
IxB=0.2276e-3
IyB=0.1425e-3
IzB=IyB

R,1,mE,mE,mE,IxE,IyE,IzE
R,2,mS,mS,mS,IxS,IyS,IzS
R,3,mP,mP,mP,IxP,IyP,IzP
R,4,mB,mB,mB,IxB,IyB,IzB

! Coordinate system
CSYS,0,0,0,0

! Define nodes
N,1,0,0,0
NGEN,nodes,1,1,nodes,1,lenS,0,0
! End of node print
```

```
! 2D Elastic beam elements
TYPE,1
MAT,1
REAL,221
E,1,2
EGEN,(nodes-1),1,1,(nodes-1),1
! End of beam elements

! Generalized mass elements
TYPE,2
! shaft endpoints
REAL,1
E,1
E,nodes
masses(1,2) = mE
masses(nodes,2) = mE
masses(1,3) = IxE
masses(nodes,3) = IxE

! slender solid shaft (except endpoints)
REAL,2
E,2
EGEN,(nodes-2),1,(nodes+2),((2*nodes)-1),1
*DO,var1,2,(nodes-1),1
  masses(var1,2) = mS
  masses(var1,3) = IxS
*ENDDO

! proximitor assembly at np1 and np2
REAL,3
E,np1
E,np2
masses(np1,2) = masses(np1,2)+mP
masses(np2,2) = masses(np2,2)+mP
masses(np1,3) = masses(np1,3)+IxP
masses(np2,3) = masses(np2,3)+IxP

! laminated bearing assembly at nb1 and nb2
REAL,4
E,nb1
E,nb2
```

```
masses(nb1,2) = masses(nb1,2)+mB
masses(nb2,2) = masses(nb2,2)+mB
masses(nb1,3) = masses(nb1,3)+IxB
masses(nb2,3) = masses(nb2,3)+IxB
! End of mass elements

! Constraints: no axial movement
D,ALL,UX

FINISH
! End of rotor model
! End of element print

/SOLU
ANTYPE,MODAL
MODOPT,REDUC,shapes
TOTAL,shapes
MXPAND,shapes
!LUMPM,ON
SAVE
SOLVE
SAVE
FINISH

/SOLU
EXPASS,ON
MXPAND,shapes
SOLVE
FINISH

! write results into files
/POST1
*DO,var1,1,shapes,1
  SET,1,var1
  *VGET,tmp,NODE,1,U,Y,,0
  *DO,var2,1,nodes,1
    phi_array(var2,var1)=tmp(var2)
  *ENDDO

  *VGET,tmp,NODE,1,ROT,Z,,0
  *DO,var2,1,nodes,1
    psi_array(var2,var1)=tmp(var2)
```

```
  *ENDDO
*ENDDO

! save natural frequencies to w_vector
*DO,var1,1,shapes,1
  w_vector(var1,1) = var1
  *GET,w_vector(var1,2),MODE,var1,FREQ
*ENDDO

! write nodal displacements to DISPLACE2.DAT
*MWRITE,phi_array,displace2,dat,,JIK,shapes,nodes,1
%15.7G %15.7G %15.7G %15.7G %15.7G %15.7G %15.7G %15.7G %15.7G %15.7G

! write nodal rotations to ROTATE2.DAT
*MWRITE,psi_array,rotate2,dat,,JIK,shapes,nodes,1
%15.7G %15.7G %15.7G %15.7G %15.7G %15.7G %15.7G %15.7G %15.7G %15.7G

! write eigenfrequencies to OMEGA2.DAT
*MWRITE,w_vector,omega2,dat,,JIK,2,shapes,1
%D %G

! write nodal masses to MASSDATA2.DAT
*MWRITE,masses,massdata2,dat,,JIK,3,nodes,1
%D %G %G

SAVE
FINISH
```

MATLAB CODE LISTING FOR `jeff1ver.m`

```
%
% jeff1_ver.m - Jeffcott rotor with internal damping and mass imbalance
%
% verification of theoretical conditions for stability
%
clc;
clear all;
format compact;
format short g;
diary off;

% AMB parameters
N = 330;
Ag = 3.22e-4;
mu0 = 4*pi*1e-7;
d = 2.0e-3;
ib = 1.0;

% AMB position stiffness
Kx = Ag*N*N*ib*ib*mu0/(d*d*d);

% AMB current stiffness
Ki = Ag*N*N*ib*mu0/(d*d);

% mass
m=10;

%internal damping
ci=60.0; % critical value for w=50; m=10

% running speed, rad/s
w=50;

% simple PD controller
Kp= -1000;
```

```
Kd= -5;

icx=0;
icy=0;

A=[zeros(2,2), eye(2,2);
    (Kx+(Ki*Kp))/m, -w*ci/m, ((Ki*Kd)-ci)/m,0;
    w*ci/m, (Kx+(Ki*Kp))/m, 0, ((Ki*Kd)-ci)/m];

b = [0,0; 0,0; 1,0; 0,1];

% output matrix
Cx = [1,0,1,0; 0,1,0,1];

disp('C*B matrix:');
Cx*b

rank(Cx*b)

sys = ss(A,b,Cx,zeros(2,2));

[z,p,k] = zpkdata(sys);

disp('Zeros of the OLTF:');
z{1}
z{2}
z{3}
z{4}


%
% the end
%
```

MATLAB code listing for `flexhub11ver.m`

```
% flexhub11_ver.m
%
% Simulation of flexible hub with internal damping
%
% Verification of theoretical conditions
%

clear all;
clc;

% rotational speed [rad/s]
w = 80;

% disturbance frequency [Hz]
wd = w;

% static imbalance [kg*m]
imb = 0.0; %0.005; %0.005;

%
% AMB properties
%
N = 330;
Ag = 3.22e-4;
mu0 = 4*pi*1e-7;
d = 2.0e-3;
i_b = 1.0;

%
% linearized AMB parameters
%
%K = Ag*N*N*mu0/4.0;

% position stiffness
Kd = Ag*N*N*i_b*i_b*mu0/(d*d*d);
```

```
% current stiffness
Kc = Ag*N*N*i_b*mu0/(d*d);

%
% PD controller gains
%
Kp = 1000;
Kd = 10;

%
% Rotor properties
%
mS = 0.238;
mR = 0.567;
cH = 4.84; %1.84;
kH = 2000; %26994;

kk = [(-2*Kc/mS), 0; 0,(-2*Kc/mS)];
B = [zeros(4,2); kk; zeros(2,2)];
clear kk;

KK = [Kp,0,0,0,Kd,0,0,0; 0,Kp,0,0,0,Kd,0,0];

A1 = [(-(2*Kd)-kH)/mS, (-w*cH)/mS, kH/mS, (w*cH)/mS;
    (w*cH)/mS,(-(2*Kd)-kH)/mS, (-w*cH)/mS, kH/mS;
    kH/mR,(w*cH)/mR,(-kH)/mR,(-w*cH)/mR;
    (-w*cH)/mR,kH/mR,(w*cH)/mR,(-kH)/mR];

A2 = [-cH/mS,0,cH/mS,0;
    0,-cH/mS,0,cH/mS;
    cH/mR,0,-cH/mR,0;
    0,cH/mR,0,-cH/mR];

A = [zeros(4,4), eye(4,4); A1, A2];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% reduced order state-estimator for Rim position and velocity
%
AA = (A+(B*KK));
```

```
F = eye(4,4);
%tmp = min(real(eig(AA)));

F(1,1) = -1000; %2*tmp;
F(2,2) = -2000; %2.1*tmp;
F(3,3) = -3000; %2.2*tmp;
F(4,4) = -4000; %2.3*tmp;

L = eye(4,4);

C = [eye(2,2), zeros(2,6);
     zeros(2,4), eye(2,2), zeros(2,2)];

%Cx = [-1,0,0,0, -1,0,0,0; 0,-1,0,0, 0,-1,0,0];
Cx = [-1,0,1,0, -1,0,1,0; 0,-1,0,1, 0,-1,0,1];

%disp('Rank of controllability matrix F|L (min.4):');
%rank(ctrb(F,L))

%disp('Eigenvalues of A+(B*KK) (state feedback + linear AMB):');
eig(A+(B*KK))

disp('C*B matrix:');
Cx*B

eig(Cx*B)

sys = ss(AA,B,Cx,zeros(2,2));
[z,p,k] = zpkdata(sys);

disp('Zeros of the open loop transfer function:');
z{1}
z{2}
z{3}
z{4}


%
% the end
%
```

MATLAB CODE LISTING FOR FlexDampver.m

```
%
% George T. Flowers and Andras Simon
%
% Flex_Damp20_ver.m
%
%*************************************************************************
% Program to simulate a flexible rotor with internal damping and
% supported by two radial magnetic bearings.
%*************************************************************************
%
% this script is using the FLEX_SHAFT2 Ansys model and generated files
%
% verification of theoretical conditions for (A)SPR-ness
%

clc;
clear all;
format compact;
format short g;
%
% 1st crit. speed:  54 rad/s
% 2nd crit. speed: 133 rad/s
%
% simulation time
%
t_end = 5;

global a1 ac1 b1;
global nm zeta;

% running speed [rad/s]
global w;

w = 100;%150; %800;
wd = w;
```

```
%DG=50000;

% internal damping ratio (best leave at 1%)
zeta = 0.01;
%zeta = 0.01;
zeta = 0.05;
%zeta=0;
%
% setting up state-space model and AMB parameters
%
Model;

pcon = eig(a1);


%
% poles for state f/b (change real parts of eig(a1))
%
for k=1:((4*nm)+4)
    if(real(pcon(k))>0)
        pcon(k) = -pcon(k);
    end;
    if(abs(real(pcon(k)))<1.0e-5)
        pcon(k) = 0+(imag(pcon(k))*sqrt(-1));
    end;
end;

% state feedback for model *without* internal damping
Kc = place(a1,b1,pcon);
Kc(:,17:20) = 0;

a11 = a1-(b1*Kc);
ac1 = ac-(b1*Kc);

Kc2 = place(ac,b1,pcon);

ws = 5000.0;

cx1 = [0,0, om(3), om(4), 0,0, om(3), om(4);
       0,0,-om(3),-om(4), 0,0, om(3), om(4);
       0,0,-om(3),-om(4), 0,0,-om(3),-om(4);
       0,0, om(3), om(4), 0,0,-om(3),-om(4)];
```

```
cx2 = [0,0, om(3), om(4), 0,0,-om(3),-om(4);
       0,0, om(3), om(4), 0,0, om(3), om(4);
       0,0,-om(3),-om(4), 0,0, om(3), om(4);
       0,0,-om(3),-om(4), 0,0,-om(3),-om(4)];

cx = zeta*[w*cx1,cx2,eye(4,4)/400];
cxx = [w*cx1,cx2,zeros(4,4)];

clear cx1 cx2

Kcxx = zeros(4,20);

disp('C*B matrix:');
cx*b1

eig(cx*b1)

sys = ss(ac1,b1,cx,zeros(4,4))

disp('Zeros of the open loop t.f.:')

[z,p,k] = zpkdata(sys)

z{1}

%
% the end
```

MATLAB CODE LISTING FOR `jefffixed.m`

```
%
% jeff_fixed.m - simple Jeffcott rotor with internal damping and
% mass imbalance
% fixed-gain version

clc;
clear all;
format compact;
format short g;
diary off;

% AMB parameters
N = 330;
Ag = 3.22e-4;
mu0 = 4*pi*1e-7;
d = 2.0e-3;
ib = 1.0;

% AMB position stiffness
Kx = Ag*N*N*ib*ib*mu0/(d*d*d);

% AMB current stiffness
Ki = Ag*N*N*ib*mu0/(d*d);

% mass
m=10;

%internal damping
%ci=4.4158; % critical value for w=1000; m=1
ci=60.0; % critical value for w=50; m=10
%ci=0.0;%5;

% running speed, rad/s
w=50;
```

```
% static imbalance
imb = 0.005; %0.005;%0.001;

% sampling frequency
ws = 5000;

dt = 1/ws;

t_end = 5.0;
t1 = 0;

steps = t_end/dt;

q = zeros(4,1);

% initial conditions
q(1) = 0.5e-3;
q(2) = -1.2e-3;

% simple PD controller
Kp= -1000;
Kd= -5;

icx=0;
icy=0;

xx = zeros(steps,1);
yy = zeros(steps,1);
tt = zeros(steps,1);

A=[zeros(2,2), eye(2,2);
    (Kx+(Ki*Kp))/m, -w*ci/m, ((Ki*Kd)-ci)/m,0;
    w*ci/m, (Kx+(Ki*Kp))/m, 0, ((Ki*Kd)-ci)/m];

pcon = eig(A);

%
% poles for state f/b (change real parts of eig(a1))
%
%
for k=1:4
    if(real(pcon(k))>0)
```

```
            pcon(k) = -pcon(k);
        end;
        if(abs(real(pcon(k)))<1.0e-5)
            pcon(k) = 0+(imag(pcon(k))*sqrt(-1));
        end;
    end;
end;


eig(A)

b = [0,0; 0,0; 1,0; 0,1];

K = place(A,b,pcon);

AA = A-(b*K);
eig(AA)

imbx = zeros(steps,1);
imby = zeros(steps,1);

wd=w;
%wd=100;

% 20 rad/s : 250
% 50 rad/s : 12500
% 60 rad/s : 15000
for k=1:steps
    tt(k)=dt*(k-1);
    imbx(k) = w*w*imb*cos(wd*2.0*pi*dt*k);
    imby(k) = w*w*imb*sin(wd*2.0*pi*dt*k);
end;

disp('Simulation is running...');

ci = 10;

Fx=0;
Fy=0;
dx_r=0;
dy_r=0;

tic;
```

```
for k=1:steps

    x_old=q(1);
    y_old=q(2);

    q(3) = dx_r;
    q(4) = dy_r;

    [t,Q] = ode45(@jeff_fix_ode,[t1,t1+dt],q,...
[],ci,w,m,K,b,Fx,Fy,imbx(k),imby(k));
    q = Q(size(t,1),:)';
    t1 = t1+dt;

    dx_r = q(3);
    dy_r = q(4);

    % manual differentation for velocity calculation
    q(3)=(q(1)-x_old)/dt;
    q(4)=(q(2)-y_old)/dt;

    % simple PD control for current to AMB
    icx = (Kp*q(1))+(Kd*q(3));
    icy = (Kp*q(2))+(Kd*q(4));

    Fx = (Kx*q(1))+(Ki*icx);
    Fy = (Kx*q(2))+(Ki*icy);

    xx(k)=q(1);
    yy(k)=q(2);

    if(mod(k,ws)==0)
        disp(k);
    end;
end;

toc;

figure;
plot(tt(1:k),xx(1:k),'b--',tt(1:k),yy(1:k),'r-');
axis([0 t1 -d d]);
%axis([0 t1 -0.1 0.1]);
```

```
%axis([0 t1 -5e-5 5e-5]);
legend('x','y');
xlabel('Simulation time, [s]');
ylabel('Displacement, [m]');

disp('done.');

%
% the end
%
```

MATLAB CODE LISTING FOR flexhubfixed.m

```
%
% flexhub_fixed.m
%
% Simulation of flexible hub with internal damping
% Fixed-gain version

clear all;
clc;

% rotational speed [rad/s]
w = 80; %80; % 80 crit.

% disturbance frequency [Hz]
wd = w;

% static imbalance [kg*m]
imb = 0.0; %0.005; %0.005;

%
% sampling frequency [Hz]
%
%ws = 10000;
ws = 5000;
dt = (1.0/ws);

% simulation time [s]
t_end = 5.0;

%
% AMB properties
%
N = 330;
Ag = 3.22e-4;
mu0 = 4*pi*1e-7;
d = 2.0e-3;
```

```matlab
i_b = 1.0;

%
% linearized AMB parameters
%
%K = Ag*N*N*mu0/4.0;

% position stiffness
Kd = Ag*N*N*i_b*i_b*mu0/(d*d*d);

% current stiffness
Kc = Ag*N*N*i_b*mu0/(d*d);

%
% PD controller gains
%
Kp = 1000;
Kd = 10;

%
% Rotor properties
%
mS = 0.238;
mR = 0.567;
cH = 4.84; %1.84;
kH = 2000; %26994;

q = zeros(8,1);

%
% initial conditions of shaft (xS,yS)
% (for the shaft it should not be larger than the AMB airgap (2 mm)
%
q(1) = 1.5e-3;
q(2) = -1.0e-3;

%
% initial conditions for the Rim position (xRr,yRr) and for the
% estimated Rim position (xRe,yRe)
%
xRe = q(1);
yRe = q(2);
```

```
steps = (t_end/dt);

i_cx = 0;
i_cy = 0;

kk = [(-2*Kc/mS), 0; 0,(-2*Kc/mS)];
B = [zeros(4,2); kk; zeros(2,2)];
clear kk;

KK = [Kp,0,0,0,Kd,0,0,0; 0,Kp,0,0,0,Kd,0,0];

A1 = [(-(2*Kd)-kH)/mS, (-w*cH)/mS, kH/mS, (w*cH)/mS;
    (w*cH)/mS,(-(2*Kd)-kH)/mS, (-w*cH)/mS, kH/mS;
    kH/mR,(w*cH)/mR,(-kH)/mR,(-w*cH)/mR;
    (-w*cH)/mR,kH/mR,(w*cH)/mR,(-kH)/mR];

A2 = [-cH/mS,0,cH/mS,0;
    0,-cH/mS,0,cH/mS;
    cH/mR,0,-cH/mR,0;
    0,cH/mR,0,-cH/mR];

A = [zeros(4,4), eye(4,4); A1, A2];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% reduced order state-estimator for Rim position and velocity
%
AA = (A+(B*KK));

pcon = eig(AA);
%
% poles for state f/b
%
%
for k=1:8
    if(real(pcon(k))>0)
        pcon(k) = -pcon(k);
    end;
    if(abs(real(pcon(k)))<1.0e-5)
        pcon(k) = 0+(imag(pcon(k))*sqrt(-1));
    end;
```

```
end;

K2 = place(AA,B,pcon);

AAA = AA-(B*K2);

% recalc matrices with higher speed or higher cH
cH = 10.0; % was 4.84
%w = 100; % was 80

A1 = [(-(2*Kd)-kH)/mS, (-w*cH)/mS, kH/mS, (w*cH)/mS;
    (w*cH)/mS,(-(2*Kd)-kH)/mS, (-w*cH)/mS, kH/mS;
    kH/mR,(w*cH)/mR,(-kH)/mR,(-w*cH)/mR;
    (-w*cH)/mR,kH/mR,(w*cH)/mR,(-kH)/mR];

A2 = [-cH/mS,0,cH/mS,0;
    0,-cH/mS,0,cH/mS;
    cH/mR,0,-cH/mR,0;
    0,cH/mR,0,-cH/mR];

A = [zeros(4,4), eye(4,4); A1, A2];

AA = (A+(B*KK));

F = eye(4,4);
%tmp = min(real(eig(AA)));

F(1,1) = -1000; %2*tmp;
F(2,2) = -2000; %2.1*tmp;
F(3,3) = -3000; %2.2*tmp;
F(4,4) = -4000; %2.3*tmp;

L = eye(4,4);

C = [eye(2,2), zeros(2,6);
    zeros(2,4), eye(2,2), zeros(2,2)];

cc = [-1,0,0,0,-1,0,0,0; 0,-1,0,0,0,-1,0,0];

%disp('Rank of controllability matrix F|L (min.4):');
%rank(ctrb(F,L))
```

```matlab
%disp('Eigenvalues of A+(B*KK) (state feedback + linear AMB):');
%eig(A+(B*KK))

%
% solve the Lyapunov-equation for T in (T*(A+(B*KK)) - F*T = L*C)
%
%T = lyap(-F,AA,-(L*C));
T = lyap(-F,AAA,-(L*C));

%disp('Error in solution of Lyapunov equation (should be small):');
%(T*(AA))-(F*T)-(L*C)

TB = T*B;

invCT = inv([C;T]);
z = zeros(4,1);
dz = zeros(4,1);

% set up storage variables for plotting
DTpos_xS = zeros(steps,1);
DTpos_yS = zeros(steps,1);
DTpos_xRr = zeros(steps,1);
DTpos_yRr = zeros(steps,1);
DTpos_xRe = zeros(steps,1);
DTpos_yRe = zeros(steps,1);
DTctr_x = zeros(steps,1);
DTctr_y = zeros(steps,1);
DTtt = zeros(steps,1);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for k=1:steps
    tmpx = wd*2.0*pi*dt*k;
    imbx(k) = w*w*imb*cos(tmpx);
    imby(k) = w*w*imb*sin(tmpx);
end;
clear tmpx;

disp('Simulation is running...');

t1 = 0;
```

```
xSdot_real = 0;
ySdot_real = 0;
xRrdot_real = 0;
yRrdot_real = 0;

%
% main simulation loop
%
tic;

for k=1:steps

    %
    % simple PD control
    %
    i_cx = (Kp*q(1))+(Kd*q(5));
    i_cy = (Kp*q(2))+(Kd*q(6));

    Fx = (Kd*q(1))+(Kc*i_cx)+imbx(k);
    Fy = (Kd*q(2))+(Kc*i_cy)+imby(k);

    % restore real velocities for ODE()
    q(5) = xSdot_real;
    q(6) = ySdot_real;
    q(7) = xRrdot_real;
    q(8) = yRrdot_real;

    [t,Q] = ode45(@flexhub_dyn2,[t1,t1+dt],q,...
[],mS,mR,cH,kH,w,Fx,Fy,B*K2);

    xS_old = q(1);
    yS_old = q(2);
    xRr_old = q(3);
    yRr_old = q(4);

    q = Q(size(t,1),:)';
    t1 = t1+dt;

    xS = q(1);
    yS = q(2);

    xRr = q(3);
```

```matlab
        yRr = q(4);

        %
        % velocities obtained from ODE solution
        %
        xSdot_real = q(5);
        ySdot_real = q(6);
        xRrdot_real = q(7);
        yRrdot_real = q(8);

        %
        % velocities calculated manually from measured positions
        %
        xSdot = (q(1)-xS_old)/dt;
        ySdot = (q(2)-yS_old)/dt;
        xRrdot = (q(3)-xRr_old)/dt;
        yRrdot = (q(4)-yRr_old)/dt;

        dz = (F*z)+(TB*[i_cx;i_cy])+(L*[q(1); q(2); q(5); q(6)]);

        z = z+(dz*dt);

        xRe_old = xRe;
        yRe_old = yRe;

        ztx = invCT*[q(1);q(2);q(5);q(6);z];

        xRe = ztx(3);
        yRe = ztx(4);
%        xRedot = ztx(7);
%        yRedot = ztx(8);
        xRedot = (xRe-xRe_old)/dt;
        yRedot = (yRe-yRe_old)/dt;

        DTpos_xS(k) = q(1);
        DTpos_yS(k) = q(2);
        DTpos_xRr(k) = q(3); % real Rim positions from mechanical model
        DTpos_yRr(k) = q(4);
        DTpos_xRe(k) = xRe; % estimated Rim positions from estimator
        DTpos_yRe(k) = yRe;
        DTctr_x(k) = i_cx;
        DTctr_y(k) = i_cy;
```

```matlab
    DTtt(k) = t1;

    if(mod(k,ws)==0)
        disp(k);
    end;
end;

disp('complete');
beep;

toc;

dist=1;
msteps = steps/dist;
xmark = zeros(msteps,1);
ymark = xmark;

for k=1:msteps
    xmark(k) = DTpos_xS(dist*k);
    ymark(k) = DTtt(dist*k);
end;

figure;
%plot(DTtt,DTpos_xS,DTtt,DTpos_yS,DTtt,DTpos_xRr,DTtt,DTpos_yRr);
plot(DTtt,DTpos_xS,'b-',DTtt,DTpos_xRr,'r:');
axis([0, t_end, -d, d]);
tit = strcat('\Omega=',num2str(w),' u=',num2str(imb));
title(tit);
ylabel('Displacement, [m]');
xlabel('Simulation time [s]');
legend('Shaft X','Rim X');

%
% the end
%
```

MATLAB code listing for `FixedGain1.m`

```
%
% George T. Flowers and Andras Simon
%
% FixedGain1.m
%
%*************************************************************************
% Program to simulate a flexible rotor with internal damping and
% supported by two radial magnetic bearings.
%*************************************************************************
%
% this script is using the FLEX_SHAFT2 Ansys model and generated files
%
% Fixed gain controller (Baseline)
%

clc;
clear all;
format compact;
format short g;
%
% 1st crit. speed:  54 rad/s
% 2nd crit. speed: 133 rad/s
%
% simulation time
%
t_end = 5;

global a1 ac1 b1;
global nm zeta;

% running speed [rad/s]
global w;

w = 150;%150; %800;
wd = w;
```

```
% internal damping ratio (best leave at 1%)
zeta = 0.01;

%
% setting up state-space model and AMB parameters
%
Model;

pcon = eig(a1);

%
% poles for state f/b (change real parts of eig(a1))
%
%
for k=1:((4*nm)+4)
    if(real(pcon(k))>0)
        pcon(k) = -pcon(k);
    end;
    if(abs(real(pcon(k)))<1.0e-5)
        pcon(k) = 0+(imag(pcon(k))*sqrt(-1));
    end;
end;

% state feedback for model *with* internal damping
Kc2 = place(ac,b1,pcon);
%Kc2(:,17:20) = 0;

ac2 = ac-(b1*Kc2);

%
% change zeta to trick fixed gain controller
%
%zeta = 0.05;

%
% re-calculate system matrices
%
%Model;

q = zeros((4*nm)+4,1);
```

```matlab
%initial bearing positions
q(1) = 0.0005;
q(5) = -0.0001;

% sampling frequency [Hz]
%
ws = 5000.0;

dt = 1.0/ws;

t1 = 0.0;

steps = (t_end-t1)/dt;

% how many imbalances are accounted for
ni = 1;

% imbalance node locations (for each imbalance)
iml = ones(ni,1)*44;
% imbalance values (for each imbal.)
imm = zeros(ni,1);
% imbalance phase shifts (for each imbalance)
imph = zeros(ni,1);

% imbalance matrix
phi_imb = zeros(nn,ni);
imphase = zeros(ni,1);

% node locations where imbalances are located (1 per node)
iml(1) = 44;
%iml(2) = 30;

% individual imbalances [kg-m]
%imm(1) = 5e-3;
imm(1) = 1e-3;

% phase shift of imbalances [degrees]
imph(1) = 0;
%imph(2) = 90;

for i=1:ni
    phi_imb(iml(i),i)=imm(i); % place imbalances into imbalance matrix
```

```matlab
        imph(i) = imph(i)*(pi/180.0); % [degree] -> [rad]
end;

imbtmp = (phi')*phi_imb*w*w;

imbx = zeros(nm,steps);
imby = zeros(nm,steps);
imbal_angle_cos = zeros(ni,steps);
imbal_angle_sin = zeros(ni,steps);

tt = zeros(steps,1);

for k=1:steps
    tmp_imb = (wd*2.0*pi*dt*k*ones(ni,1))+imph;
    imbal_angle_cos(:,k) = cos(tmp_imb);
    imbal_angle_sin(:,k) = sin(tmp_imb);
    tt(k) = dt*k;

    % imbalance force in X direction
    imbx(:,k) = imbtmp*imbal_angle_cos(:,k);

    % imbalance force in Y direction
    imby(:,k) = imbtmp*imbal_angle_sin(:,k);
end;

x_old = zeros(nm,1);
y_old = zeros(nm,1);
dx_real = zeros(nm,1);
dy_real = zeros(nm,1);
dx_calc = zeros(nm,1);
dy_calc = zeros(nm,1);

pos_bear = zeros(nm,steps);
pos_prox = zeros(nm,steps);

u = zeros(4,1);

disp('Simulation is running...');

tic;
for k=1:steps
```

```matlab
    % save previous positions for manual differentation
    x_old = q(1:4);
    y_old = q(5:8);

    % restore real velocities for ODE45()
    q(9:12) = dx_real;
    q(13:16) = dy_real;

     [t,Q] = ode45(@flex_ode20,[t1, t1+dt],q,[],...
         b1,ac2,u,imbx(:,k),imby(:,k));
    q = Q(size(t,1),:)';

    t1 = t1+dt;

    % save real velocities (from ODE solution)
    dx_real = q(9:12);
    dy_real = q(13:16);

    % calculate actual velocities from position inputs
    % by using simple differentiation
    q(9:12) = (q(1:4)-x_old)/dt;
    q(13:16) = (q(5:8)-y_old)/dt;

    % positions at the prox. sensors
    pos_prox(:,k) = cprox*q;

    % positions at the bearings
    pos_bear(:,k) = cbear*q;

    if(mod(k,ws)==0)
        disp(k);
    end;

end;

toc;

% plot X and Y positions at each bearing on a single figure
figure;
plot(tt(1:k),pos_prox(1,:),'b-',tt(1:k),pos_prox(2,:),'r:');
tit = strcat('\Omega=',num2str(w),' u=',num2str(imm(1)));
title(tit);
```

```
legend('x_{p1}','y_{p1}');
axis([0 t1 -0.002 0.002]);

ylabel('Displacement, [m]');
xlabel('Simulation time, [s]');

disp('completed.');

%
% the end
%
```