Robust control techniques require a dynamic model of the plant and bounds on model uncertainty to formulate control laws with guaranteed stability. Although techniques for modeling and identifying dynamic systems are well established, very few procedures exist for estimating uncertainty bounds. In the case of $H_\infty$ control synthesis, a conservative weighting function for model uncertainty is usually chosen to ensure closed-loop stability over the entire operating space. The primary drawback of this conservative, “hard computing” approach is reduced performance due to the number of plants the resulting controller can stabilize. This paper demonstrates a novel “soft computing” approach to estimate bounds of model uncertainty resulting from parameter variations, unmodeled dynamics, and non-deterministic processes in dynamic plants. This approach uses confidence interval networks (CINs), radial basis function neural networks trained using asymmetric bilinear error cost functions, to estimate confidence intervals on the uncertainty associated with nominal linear models for robust control synthesis. This research couples the hard computing features of $H_\infty$ control with the soft computing characteristics of intelligent system identification, and realizes the combined advantages of both. Experimental demonstrations conducted on a multivariable, flexible-rotor active magnetic bearing system confirm these capabilities.
H∞ CONTROL OF ACTIVE MAGNETIC BEARINGS: AN INTELLIGENT UNCERTAINTY MODELING APPROACH

BY

NATHAN SCOTT GIBSON

A DISSERTATION SUBMITTED TO THE GRADUATE FACULTY OF
NORTH CAROLINA STATE UNIVERSITY
IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

DEPARTMENT OF MECHANICAL AND AEROSPACE ENGINEERING

RALEIGH, NORTH CAROLINA

USA

SEPTEMBER 2004

APPROVED BY:

______________________________

CHAIR OF ADVISORY COMMITTEE
Biography

Nathan Gibson was born in Wilmington, NC on September 19, 1977. His academic career began in 1995 at North Carolina State University. He earned a B.S. degree in Textile Engineering in 1999 and a M.S. degree in Textile Engineering in 2000. He began his doctoral work at NCSU in 2001, and earned the Ph.D. degree in 2004. Nathan enjoys spending time with his wife Rachael and his two sons William and Andrew. He is also an avid political junkie and likes a round of golf every now and then. Following his doctoral work, he will take a position as Lead Engineer in the Advanced Controls division of General Electric Aircraft Engines in Cincinnati, Ohio.
I would first like to extend my sincere thanks to my advisor Dr. Gregory Buckner, without whom this work would not be possible. Your time, dedication, and guidance are greatly appreciated and I thank you for taking a chance on me. It has been an honor and a privilege to work with you.

I would also like to thank my committee members, Dr. Fen Wu, Dr. Paul Ro, Dr. Jeff Eischen, and Dr. Helen Zhang for your time, interest, and contributions to this research.

I sincerely appreciate my colleague Heeju Choi. You have been a valuable partner in many aspects of this work, but I also value you as a true friend. I wish you success in all that you endeavor.

To my parents, John and Pat Gibson, I appreciate your love, guidance, and wisdom. Thank you for teaching me that I can accomplish anything.

Finally, I would like to thank my lovely wife Rachael, my greatest supporter. Thank you for believing in me during this time, for your trust that I would one day actually complete this dissertation, and for your love.
# Table of Contents

List of Tables vii

List of Figures viii

1 Introduction 1
   1.1 Background and Motivation ..................................... 1
   1.2 Uncertainty Modeling ........................................... 3
   1.3 Scope of Work .................................................... 4

2 The Active Magnetic Bearing Test Rig 6
   2.1 Overview of Active Magnetic Bearings .......................... 6
   2.2 The MBRotor AMB System ....................................... 7
   2.3 Electromechanical Modeling of AMB systems ................. 9
   2.4 Rotordynamic Modeling of AMB systems ...................... 11

3 Robust $H_\infty$ Control of Active Magnetic Bearings 19
   3.1 $H_\infty$ Control Theory ...................................... 19
   3.2 Presence of Modeling Error .................................... 21
   3.3 Uncertainty Bound Specification .............................. 22
   3.4 Nominal $H_\infty$ Control: Experimental Results .......... 25

4 Estimating Model Uncertainty for Active Magnetic Bearings 31
   4.1 System Identification ........................................... 32
   4.2 Model Error Modeling .......................................... 33
   4.3 Intelligent Model Error Identification ...................... 37

5 Identified $H_\infty$ Control: Experimental Results 44
   5.1 Uncertainty Bound Specification .............................. 45
      5.1.1 MEM bounds ................................................ 45
      5.1.2 CIN bounds ............................................... 50
   5.2 Controller Implementation ..................................... 54
   5.3 Advantages of the Confidence Interval Network for Uncertainty Iden-
        tification ...................................................... 60
6 Conclusions

6.1 Contributions .................................................. 62
6.2 Future Research Directions ................................. 63

List of References .................................................. 65

A Statistical Analysis of Bilinear Cost Functions .......... 69
A.1 Introduction .................................................... 69
A.2 Function Approximation via ANNs .......................... 70
A.3 Quadratic Error Cost Function Training .................. 73
    A.3.1 Proof of convergence .................................. 73
A.4 Bilinear Error Cost Function Training ..................... 75
    A.4.1 Symmetric convergence ................................. 76
    A.4.2 Asymmetric convergence ............................... 76
A.5 Batch Update using the Quadratic Error Cost Function . 80

B CIN Learning: Additional Figures ........................... 85
List of Tables

3.1 Performance indices for nominal $H_\infty$ control .................. 30
5.1 Performance indices for 4 $krpm$ ........................................ 59
5.2 Performance indices for 2 $krpm$ ........................................ 59
5.3 Performance indices for 6 $krpm$ ........................................ 60
5.4 Performance indices for 2 $krpm$ w/ added mass ..................... 60
## List of Figures

2.1 AMB operating principle ........................................ 7
2.2 Radial AMB ....................................................... 8
2.3 Revolve MBRotor AMB system ................................. 9
2.4 AMB differential driving mode ................................. 10
2.5 Generalized rigid rotor supported by two radial bearings ... 12
2.6 Eigenvalues versus rotor speed ................................. 18
3.1 LFT framework .................................................... 20
3.2 $k$-step prediction for $k = 1$ ................................. 22
3.3 $k$-step prediction for $k = 10$ ............................... 23
3.4 Additive model uncertainty .................................. 24
3.5 Multiplicative model uncertainty ............................. 25
3.6 Nominal weighting functions .................................. 26
3.7 $H_{\infty}$ system interconnection .............................. 27
3.8 Rotor orbit - Nominal $H_{\infty}$ controller at 4 $krpm$ ....... 28
3.9 Rotor orbit - Nominal $H_{\infty}$ controller at 2 $krpm$ ....... 28
3.10 Rotor orbit - Nominal $H_{\infty}$ controller at 6 $krpm$ ...... 29
3.11 Rotor orbit - Nominal $H_{\infty}$ controller at 2 $krpm$ w/ added mass ......... 29
3.12 Sum of squared errors (nominal) versus rotational speed ........ 30
4.1 Control and system identification diagram .................. 32
4.2 Control and model error modeling diagram .................. 34
4.3 Model error model, with 99% confidence intervals .......... 36
4.4 Intelligent model error identification using a non-recurrent RBFN 38
4.5 RBFN estimate of average model error magnitude .......... 40
4.6 CIN estimate of uncertainty bound (99.9% confidence interval of model error magnitude) ..................... 43
5.1 Identified MEM bounds for 4 $krpm$ (Nominal speed) ........ 46
5.2 Identified MEM bounds for 2 $krpm$ .......................... 46
5.3 Identified MEM bounds for 6 $krpm$ .......................... 47
5.4 Identified MEM bounds for 2 $krpm$ w/ added mass ....... 47
5.5 Point and cross receptances for MEM bounds ............... 48
5.6 Comparison of MEM $W_{\Delta}$ bounds ........................ 49
5.7 Identified CIN bounds for 4 krpm (Nominal speed) .......................... 51
5.8 Identified CIN bounds for 2 krpm .............................................. 51
5.9 Identified CIN bounds for 6 krpm .............................................. 52
5.10 Identified CIN bounds for 2 krpm w/ added mass ....................... 52
5.11 Point and cross receptances for CIN bounds ................................. 53
5.12 Comparison of CIN $W_\Delta$ bounds .......................................... 53
5.13 Bearing a rotor orbit - Nominal and identified controllers at 4 krpm ... 55
5.14 Bearing b rotor orbit - Nominal and identified controllers at 4 krpm ... 56
5.15 Bearing a rotor orbit - Nominal and identified controllers at 2 krpm ... 56
5.16 Bearing b rotor orbit - Nominal and identified controllers at 2 krpm ... 57
5.17 Bearing a rotor orbit - Nominal and identified controllers at 6 krpm ... 57
5.18 Bearing b rotor orbit - Nominal and identified controllers at 6 krpm ... 58
5.19 Bearing a rotor orbit - Nominal and identified controllers at 2 krpm w/ added mass ........................................ 58
5.20 Bearing b rotor orbit - Nominal and identified controllers at 2 krpm w/ added mass ........................................ 59

A.1 Quadratic loss function ............................................................... 71
A.2 Symmetric bilinear loss function $(c_1 = c_2 = 1)$ ......................... 72
A.3 Asymmetric bilinear loss function $(c_1 = 10, c_2 = 1)$ ................... 72

B.1 CIN learning - E1/U1 for 4 krpm .............................................. 86
B.2 CIN learning - E2/U2 for 4 krpm .............................................. 86
B.3 CIN learning - E3/U3 for 4 krpm .............................................. 87
B.4 CIN learning - E4/U4 for 4 krpm .............................................. 87
B.5 CIN learning - E1/U1 for 2 krpm .............................................. 88
B.6 CIN learning - E2/U2 for 2 krpm .............................................. 88
B.7 CIN learning - E3/U3 for 2 krpm .............................................. 89
B.8 CIN learning - E4/U4 for 2 krpm .............................................. 89
B.9 CIN learning - E1/U1 for 6 krpm .............................................. 90
B.10 CIN learning - E2/U2 for 6 krpm ............................................ 90
B.11 CIN learning - E3/U3 for 6 krpm ............................................ 91
B.12 CIN learning - E4/U4 for 6 krpm ............................................ 91
B.13 CIN learning - E1/U1 for 2 krpm w/ mass .................................. 92
B.14 CIN learning - E2/U2 for 2 krpm w/ mass .................................. 92
B.15 CIN learning - E3/U3 for 2 krpm w/ mass .................................. 93
B.16 CIN learning - E4/U4 for 2 krpm w/ mass .................................. 93
Chapter 1

Introduction

1.1 Background and Motivation

Among the numerous design and analysis tools available to engineers, there exist two major paradigms for computational problem-solving. “Hard computing” denotes rigid mathematical formulation, predictability, and straightforward analysis. “Soft computing”, on the other hand, makes use of subjectivity, ambiguity, and intuition [1]. Soft computing techniques such as fuzzy logic, neural networks, and genetic algorithms achieve tractability and “learning” behavior by exploiting imprecision and approximation [2]. These two approaches have long been separate; recently however, research involving the fusion of hard and soft computing has emerged with the goal of seeing these sets as complementary [1]. By combining the attractive features of both, real-world problems can be solved in more innovative and novel ways.

The goal of this thesis is to show such a synergy applied to an industrial process. This research will show that the soft computing aspects of artificial neural networks can be used in conjunction with the hard computing aspects of robust (optimal)
control to achieve beneficial results in an easier, more efficient fashion.

The artificial neural network (ANN) was developed to model the cognitive process exhibited by the human brain [3]. It uses a structure of neurons (computational nodes), and weighted interconnections to adaptively “learn” based upon a prescribed performance index and a training algorithm that adjusts the network weights to optimize this index [4]. This soft computing tool is essentially a function approximator that is configurable to a wide array of applications.

Robust control is a hard computing design methodology focused on achieving guaranteed stability and performance for uncertain dynamic systems. Robust control synthesis requires an accurate mathematical description of the plant dynamics (a model) and bounds on the uncertainty associated with that model. Such uncertainties may result from parameter variations, under-modeled dynamics, or process disturbances. By specifying a nominal model and a “hard bound” on the uncertainty associated with that model, robust control aims to guarantee robust stability for the actual system, which must lie within the set defined by the model plus the bound. Common linear approaches to robust control (i.e. those requiring linear models of the system dynamics) include $H_\infty$ and $\mu$ – synthesis [5].

Very few procedures exist for estimating the bounds of model uncertainty. For robust control synthesis it is customary to choose uncertainty bounds (uncertainty weighting functions) that are somewhat arbitrary and overly conservative to guarantee stability, usually at the expense of performance. In fact, many textbooks on the subject provide no formal procedures for estimating the modeling uncertainty so critical for robust control synthesis.
1.2 Uncertainty Modeling

The uncertainty associated with linear models can be difficult to quantify. These system models are commonly obtained via analytical approaches, which make use of first principles (Newton’s laws, Maxwell’s equations, principles of thermodynamics, etc.) [6][7], and experimental approaches (modal analysis, etc.)[8]. System identification and parameter estimation approaches are also extensively documented [7], even “intelligent” techniques that incorporate ANNs to retain and generalize adaptive information [9][10][11]. Still, discrepancies remain between actual nonlinear systems and their linear models, and hard bounds on this modeling error need to be quantified.

Recently, research has focused on improving the accuracy of uncertainty bounds associated with dynamic models. Specific approaches include stochastic embedding, which relies on the statistical properties of an identified model [12]; set membership, an identification procedure that estimates a set of feasible models which in turn provides the uncertainty quantification related to the nominal model [13][14][15]; and model error modeling, which identifies a linear mapping from inputs to modeling error [14][16][17].

In addition, there have been approaches more focused on obtaining error bounds for their explicit use in robust control uncertainty weighting functions. A mixed-sensitivity approach is proposed in [18] in which the basic form of the weighting function is supplied, allowing for trade-offs in optimization. A fuzzy logic-based genetic algorithm was used in [19] to search for suitable uncertainty weighting functions by
optimizing the trade-off between sensitivity and complementary functions in $H_\infty$ design. Parametric gain-delay uncertainties were transformed into frequency-dependent weights with complex norm-bounding sets in [20]. Optimization techniques for $\mu$ – synthesis and $H_\infty$ loop-shaping were used to select appropriate frequency-dependent weights in [21].

1.3 Scope of Work

This dissertation will demonstrate the fusion of hard and soft computing paradigm by using artificial neural networks to estimate appropriate uncertainty weighting functions for the synthesis of multivariable robust controllers. This work introduces confidence interval networks (CINs), a soft computing variation of the model error modeling approach used to estimate stochastic uncertainty bounds associated with linear dynamic models. Experimental validations on a flexible-rotor active magnetic bearing (AMB) system demonstrate robust stability and acceptable tracking performance due to the accuracy of these intelligent uncertainty bounds.

This dissertation is structured as follows:

Chapter 2 - The Active Magnetic Bearing Test Rig. The AMB system is introduced and modeled analytically.

Chapter 3 - Robust $H_\infty$ control of Active Magnetic Bearings. A detailed description of the $H_\infty$ synthesis process is presented. Nominal performance results motivate the need for the quantification of modeling error.

Chapter 4 - Estimating Model Uncertainty for Active Magnetic Bearings.
A conceptually new approach for bounding model uncertainty incorporating CINs is presented. This (CIN) approach is compared with the model error modeling (MEM) method.

Chapter 5 - Identified $H_\infty$ control – Experimental Results. The uncertainty weighting functions developed using the CIN approach are experimentally validated on the AMB system. Stability and performance results are presented and compared with those of the MEM method.

Chapter 6 - Conclusions. Summarizes the contributions of this dissertation and comments on future research directions extending from it.

Appendices A and B. In Appendix A, a detailed statistical proof of stability and convergence of the CIN is presented. Appendix B contains CIN bounding figures that were consolidated in Chapter 5.
Chapter 2

The Active Magnetic Bearing Test Rig

2.1 Overview of Active Magnetic Bearings

Active magnetic bearings (AMBs) are electromagnetic devices that provide non-contacting support of high speed rotors. AMB systems eliminate traditional concerns with bearing friction and lubrication. In addition, the same electromechanical forces that support the rotor can be used to damp out shaft vibrations and reduce transmitted forces [22]. Thus, AMBs have tremendous potential for many high speed industrial applications.

AMBs also have significant drawbacks, however. The dynamics of these systems are highly nonlinear and inherently unstable, requiring sophisticated real-time control algorithms. During high-speed operation, gyroscopic effects such as nutation and precession (forward and backward ‘whirling’ of the rotor axis of rotation), rotational speed-dependent eigenvalues, static and dynamic unbalances, and excitation of rotor bending modes arise, further complicating control issues [22]. AMBs can also be quite expensive due to their complexity.
A representation of a single-input, single-output (SISO) magnetic bearing is shown in Figure 2.1. The control objective is to manipulate the coil current \( i(t) \) so that the vertical position of the rotor \( y(t) \) tracks the desired trajectory. When multiple actuators of this type are arranged circumferentially, as shown in Figure 2.2, the assembly is a radial magnetic bearing, which controls the rotor motion in the radial direction. This bearing does the majority of the stabilization and vibration suppression work. Typical AMB implementations also include a thrust bearing, which controls axial rotor motion. The thrust bearing ensures that the rotor stays axially centered so that the radial bearings can transmit forces appropriately.
2.2 The MBRotor AMB System

The experimental research described in this thesis required several key AMB system capabilities, the most important of which included the ability to implement and test customized controllers (access to all input and output signals). The MBRotor Research Test Stand from Revolve Magnetic Bearings, Inc. was the only commercial AMB system to satisfy these requirements. Figure 2.3 shows this system with the following system components labelled [23]:

1. Two radial magnetic bearings with silicon-iron rotors and 80 N static capacity

2. Thrust bearing with 120 N static capacity

3. 10000 rpm DC motor

4. Two adjustable balance disks mounted on a 0.25” diameter steel shaft
5. Real-time control injection capability

The Revolve AMB test rig employs a common actuation mode known as *differential driving mode*, or *Class A tuning*. The basic principles of this method are illustrated in Figure 2.4. Opposing electromagnets generate attractive forces on the mass for the same degree of freedom. Each magnet coil is supplied with the same bias current $i_{bias}$, but the control current $i_{control}$ is added to the top coil and subtracted from the bottom coil.

### 2.3 Electromechanical Modeling of AMB systems

The force generated by an electromagnetic actuator such as in Figure 2.1 can be derived using magnetic circuit analysis and conservation of energy techniques [24].

$$ f_{em} = \frac{1}{4} \mu_0 N^2 A_g \frac{i^2}{x^2} $$

(2.1)
The resulting force equation (2.1) is highly nonlinear. The magnetic force is proportional to the square of the coil current $i$ and inversely proportional to the square of the air gap between the actuator and the rotor $x$. In addition, $\mu_0$ is the permeability of free space ($4\pi \cdot 10^{-7} H/m$), $N$ is the number of turns in the coil, and $A_g$ is the area of the air gap [24].

In order to use linear control methods, it is highly desirable to approximate the electromagnetic force by linearizing (2.1) about a nominal operating point $(x_0,i_0)$. For small deviations from this operating point, a linear approximation of the electromagnetic force is

$$f_{em} - f_{em_0} \approx \frac{\partial f_{em}}{\partial x}|_{x_0,i_0}(x - x_0) + \frac{\partial f_{em}}{\partial i}|_{x_0,i_0}(i - i_0)$$

(2.2)

or

$$\Delta f_{em} \approx \frac{\partial f_{em}}{\partial x}|_{x_0,i_0}\Delta x + \frac{\partial f_{em}}{\partial i}|_{x_0,i_0}\Delta i$$

(2.3)
So then,
\[ \Delta f_{em} = k_x \cdot \Delta x + k_i \cdot \Delta i \]  
(2.4)

where \( k_x \) is the force-displacement factor or position stiffness and \( k_i \) is the force-current factor or current stiffness. The position stiffness is the rate of change in force with respect to the change in position at \( x = 0 \), and the current stiffness is the rate of change in force with respect to control current at \( x = 0 \). The position and current stiffness values depend only on the physical bearing characteristics and are independent of the control implementation.

### 2.4 Rotordynamic Modeling of AMB systems

In order to develop model-based controllers for the AMB system (Figure 2.3), accurate dynamic models must be formulated. The state equations for this multivariable, fully coupled system can be derived using an approach similar to the 2 degree-of-freedom procedure presented in [22], but is augmented to consider 4 degrees of freedom.

Consider the generalized rigid rotor in Figure 2.5 which is supported by two radial magnetic bearings, represented by the four bearing forces

\[
\mathbf{F}_B = \begin{bmatrix}
F_{ax} \\
F_{bx} \\
F_{ay} \\
F_{by}
\end{bmatrix}
\]  
(2.5)

at bearings \( a \) and \( b \) along axes \( x \) and \( y \). The rotor has a center of mass at \( g \) and is constrained at the bearing \( b \) end by a motor coupling that can be modeled as linear springs \( k_{cx} \) and \( k_{cy} \).
Figure 2.5: Generalized rigid rotor supported by two radial bearings

The equations of motion for this system can be derived using Lagrange’s equation [25]
\[
\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}_n} \right) - \frac{\partial L}{\partial \theta_n} = Q_n \quad 1 \leq n \leq N
\] (2.6)
where $\theta_n$ is the $n^{th}$ generalized coordinate, $Q_n$ is the $n^{th}$ generalized input, $L$ is the Lagrangian function $L = T - V$, $T$ is the kinetic energy expression, and $V$ is the potential energy expression.

For the generalized AMB rotor (Figure 2.5), the kinetic energy due to translation is
\[
T_t = \frac{1}{2} m \left( \dot{x}^2 + \dot{y}^2 \right)
\] (2.7)
assuming no translation in $z$. The kinetic energy due to rotation is [25]
\[
T_r = \frac{1}{2} \left( I_x \omega_x^2 + I_y \omega_y^2 + I_z \omega_z^2 \right)
\] (2.8)
where $\omega_x$, $\omega_y$, and $\omega_z$ are the angular velocities of the rotor about its own principal
inertia axes. However, the kinetic energy expressions for the Lagrangian equations need to be expressed as time derivatives of angular positions. By transforming the angular coordinates of the rigid body to fixed Euler coordinates, the Lagrangian approach will be valid. This procedure is outlined in [25], with the following result:

\[
\begin{align*}
\omega_x &= \dot{\beta} \sin(\phi) \cos(\alpha) + \dot{\alpha} \cos(\phi) \\
\omega_y &= \dot{\beta} \cos(\phi) \cos(\alpha) - \dot{\alpha} \sin(\phi) \\
\omega_z &= -\dot{\beta} \sin(\alpha) + \dot{\phi}
\end{align*}
\]  

(2.9)

Applying the simplifying assumptions \( \cos(\alpha) \simeq 1, \sin(\alpha) \simeq \alpha, \) and \( \sin^2(\alpha) \ll \alpha \) yields the following expression for the rotational kinetic energy of the rotor in terms of Euler angles.

\[
T_r = \frac{1}{2} \left( I_x \dot{\alpha}^2 + I_y \dot{\beta}^2 + I_z \dot{\phi}^2 \right) + \frac{1}{2} \left( I_z \dot{\phi}^2 + 2 \dot{\phi} \dot{\alpha} \beta \right)
\]

(2.10)

Combining the translational and rotational kinetic energy expressions (2.7) and (2.10) with the rotor’s potential energy \( V = mgh \), the Lagrangian function \( L \) can be expressed as

\[
\frac{1}{2} \left[ m (\dot{x}^2 + \dot{y}^2) + I_x \dot{\alpha}^2 + I_y \dot{\beta}^2 + I_z \dot{\phi}^2 + 2 \dot{\phi} \dot{\alpha} \beta + mgh \right]
\]

(2.11)

where the rotor translation in the direction of gravity \( h \) is

\[
h = \cos \left(45 - \arcsin \left(\frac{y}{\sqrt{x^2 + y^2}}\right)\right) \sqrt{x^2 + y^2}
\]

(2.12)

Substituting (2.11) into (2.6) yields the equations of motion

\[
\begin{align*}
m \ddot{x} &= F_{ax} + F_{bx} + F_{cx} = \Sigma F_x \\
m \ddot{y} &= F_{ay} + F_{by} + F_{cy} = \Sigma F_y \\
I_y \ddot{\beta} - I_z \Omega \dot{\alpha} &= F_{ax} l_a + F_{bx} l_b + F_{cx} l_c = \Sigma M_y \\
I_x \ddot{\alpha} + I_z \Omega \dot{\beta} &= F_{ay} l_a + F_{by} l_b + F_{cy} l_c = \Sigma M_x
\end{align*}
\]

(2.13)
where $m$ is the rotor mass, $I_x$, $I_y$, and $I_z$ are the moments of inertia about the $x$-axis, $y$-axis, and $z$-axis, respectively. The terms containing $I_z$ are the gyroscopic (Coriolis) effects. These system dynamics can be represented in matrix notation

$$M \ddot{z} + G \dot{z} = f$$

where

$$M = \begin{bmatrix} m & 0 & 0 & 0 \\ 0 & I_y & 0 & 0 \\ 0 & 0 & m & 0 \\ 0 & 0 & 0 & I_x \end{bmatrix}, \quad G = \Omega \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -I_z & 0 \\ 0 & 0 & 0 & 0 \\ 0 & I_z & 0 & 0 \end{bmatrix}, \quad z = \begin{bmatrix} x \\ \beta \\ y \\ \alpha \end{bmatrix}, \quad f = \begin{bmatrix} \Sigma f_x \\ \Sigma M_y \\ \Sigma f_y \\ \Sigma M_x \end{bmatrix}$$

To use this dynamic model for control, transforming from Euler coordinates $(x, \beta, y, \alpha)$ to bearing coordinates $(x_a, x_b, y_a, y_b)_B$ will simplify the use of position sensor and actuator measurements.

$$f = T_F F_B \quad \text{where}$$

$$f = \begin{bmatrix} \Sigma f_x \\ \Sigma M_y \\ \Sigma f_y \\ \Sigma M_x \end{bmatrix}, \quad T_F = \begin{bmatrix} 1 & 1 & 0 & 0 \\ l_a & -l_b & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & -l_a & l_b \end{bmatrix}, \quad F_B = \begin{bmatrix} F_{ax} \\ F_{bx} \\ F_{ay} \\ F_{by} \end{bmatrix}$$

$$z = T_Z z_B \quad \text{where}$$

$$z = \begin{bmatrix} x \\ \beta \\ y \\ \alpha \end{bmatrix}, \quad T_Z = \begin{bmatrix} 1 \\ l_a & l_a & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & l_b & l_a \\ 0 & 0 & -1 & 1 \end{bmatrix}, \quad z_B = \begin{bmatrix} x_a \\ x_b \\ y_a \\ y_b \end{bmatrix}$$
Substituting the bearing force transformation matrix $T_F$ (2.15) and the bearing displacement transformation matrix $T_Z$ (2.16) into (2.14) yields

$$MT_Z \ddot{z}_B + GT_Z \dot{z}_B = T_F F_B$$

$$T_F^{-1}MT_Z \ddot{z}_B + T_F^{-1}GT_Z \dot{z}_B = F_B$$

$$M_B \ddot{z}_B + G_B \dot{z}_B = F_B \quad (2.17)$$

where $M_B = T_F^{-1}MT_Z$

and $G_B = T_F^{-1}GT_Z$

Now assuming linear AMB electromechanics, the bearing forces are

$$F_B = K_s z + K_i u$$

\[
\begin{bmatrix}
  k_{sax} & 0 & 0 & 0 \\
  0 & k_{sbx} - k_{cx} & 0 & 0 \\
  0 & 0 & k_{say} & 0 \\
  0 & 0 & 0 & k_{sby} - k_{cy}
\end{bmatrix}
\begin{bmatrix}
  x_a \\
  x_b \\
  y_a \\
  y_b
\end{bmatrix}
+\begin{bmatrix}
  k_{iax} & 0 & 0 & 0 \\
  0 & k_{ibx} & 0 & 0 \\
  0 & 0 & k_{iay} & 0 \\
  0 & 0 & 0 & k_{iby}
\end{bmatrix}
\begin{bmatrix}
  i_{ax} \\
  i_{bx} \\
  i_{ay} \\
  i_{by}
\end{bmatrix}
\]

(2.18)

where $K_s$ and $K_i$ contain the force-displacement and force-current relations for each actuator, which are

$$k_{sax} = k_{sbx} = k_{say} = k_{sby} = k_s = 1.0 \times 10^5 \text{ N/m}$$

$$k_{iax} = k_{ibx} = k_{iay} = k_{iby} = k_i = 3.0 \times 10^1 \text{ N/A}$$

$$k_c = 2.6 \times 10^3 \text{ N/m} \quad (2.19)$$

The current stiffness $k_i$ was measured by first levitating the uncoupled (and non-rotating) rotor using PID control. A known mass (and thus a known force) was
suspended from the rotor near one of the bearings. Integral control resulted in \( f \approx 0 \), so \( K_s \) made a negligible contribution to the bearing force. By neglecting \( K_s, K_i \) was directly evaluated from (2.18). Then, by eliminating integral control action (using only PD control), the rotor could be made to levitate away from the bearing center. This enabled \( k_s \) to be solved using (2.18) and the experimental \( k_i \) measurement. The value of the coupling stiffness \( k_c \) that opposes \( k_s \) at the ‘b’ bearing end, as shown in Figure 2.5, was experimentally determined by displacing the end of the coupling to a known distance \( l \) from the undeformed position \( l_0 \) with a known force \( F_s \), and solving for the spring stiffness via the well known relation \( F_s = k_c(l - l_0) \).

By assuming that the electromagnetic nonlinearities of each bearing are radially centered about a nominal equilibrium point (\( z = 0 \)), the nominal currents \( u_0 \) can be solved by performing a simple static experiment. By summing actuator and gravitational forces and summing their moments about \( g \), the total force necessary to offset the weight of the rotor was determined to be 7.84 N at bearing \( a \) and 7.26 N at bearing \( b \). Assuming that the angle between gravitational force and each actuation axis is \( \theta = 45^\circ \), the nominal currents necessary to offset the gravitational forces on the rotor at the center of the bearings can be found. The sum of forces in the direction of gravity are

\[
\Sigma F_g = F_{ax} \cos(\theta) + F_{ay} \cos(\theta) + F_{bx} \cos(\theta) + F_{by} \cos(\theta) - mg = 0 \tag{2.20}
\]

Now \( x = 0 \) at the center of the bearing, so

\[
i_{ax \, nom} = i_{ay \, nom} = \frac{F_{ax} + F_{ay}}{2 \cos(\theta) \cdot k_i} = 0.186 A
\]

\[
i_{bx \, nom} = i_{by \, nom} = \frac{F_{bx} + F_{by}}{2 \cos(\theta) \cdot k_i} = 0.173 A \tag{2.21}
\]
The nominal current for each actuation axis is added to the output of the stabilizing feedback controller to formulate $i_{\text{control}}$.

The complete AMB system dynamics can be represented in the state space using a new state vector $\mathbf{x}$, composed of the displacements $\mathbf{z}$ and their time derivatives

$$\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}$$
$$\mathbf{y} = C\mathbf{x}$$

where

$$\mathbf{x} = \begin{bmatrix} \mathbf{z}_B \\ \dot{\mathbf{z}}_B \end{bmatrix}, A = \begin{bmatrix} 0 & \mathbf{I} \\ \mathbf{M}_B^{-1}\mathbf{K}_s & -\mathbf{M}_B^{-1}\mathbf{G}_B \end{bmatrix}, B = \begin{bmatrix} 0 \\ \mathbf{M}_B^{-1}\mathbf{K}_i \end{bmatrix}, C = \begin{bmatrix} \mathbf{I} & 0 \end{bmatrix}$$

(2.22)

When the system parameters $m = 1.54 \, kg$, $I_x = I_y = 2.39 \times 10^{-2} \, kg \cdot m^2$, $I_z = 4.12 \times 10^{-4} \, kg \cdot m^2$, $a = 0.153 \, m$, $b = 0.170 \, m$, and $\dot{\phi} = 418 \, rad/sec \ (4000 \, rpm)$ are included in the model (2.22), a state-space formulation of the analytical model of the MBRotor AMB
system is

\[
\dot{x} = \begin{bmatrix}
\begin{array}{cccc|cccc}
0 & 4 & I & 0 & 0 & -3.5 & 3.5 \\
1.7 \times 10^5 & -5.0 \times 10^4 & 0 & 0 & 0 & 0 & 3.7 & -3.7 \\
-5.0 \times 10^4 & 1.8 \times 10^5 & 0 & 0 & 0 & 0 & 3.7 & 3.7 \\
0 & 0 & 1.7 \times 10^5 & -5.0 \times 10^4 & 3.5 & -3.5 & 0 & 0 \\
0 & 0 & -5.0 \times 10^4 & 1.8 \times 10^5 & -3.7 & 3.7 & 0 & 0 \\
\end{array}
\end{bmatrix}x + 
\end{bmatrix}
\]

This eighth-order continuous-time model is unstable, with eigenvalues \(\lambda_i\) at

\[
\lambda = -471 \pm 3.6i, -351 \pm 0.006i, 351 \pm 0.006i, 471 \pm 3.6i
\]

Figure 2.6 illustrates the dependence of system eigenvalues on rotor speed due to the gyroscopic terms when rotor speed is varied from 0 to 10,000 rpm.
Figure 2.6: Eigenvalues versus rotor speed
Chapter 3

Robust $H_\infty$ Control of Active Magnetic Bearings

3.1 $H_\infty$ Control Theory

Minimizing the adverse effects of modeling uncertainty, steady-state error, noise, and disturbances is a fundamental goal of control synthesis. $H_\infty$ control is a multivariable robust control technique that seeks to calculate a controller $K$ such that these signals are minimized according to performance specifications. $H_\infty$ control allows for frequency-dependent bounds to be placed on each of these signals during controller synthesis to ensure admissible levels of these undesirable effects [5]. The interaction of these signals are commonly analyzed using the Linear Fractional Transformation (LFT) framework, shown in Figure 3.1. $P$ is the interconnection matrix (including the plant), $K$ is the controller, $\Delta$ is the set of all possible uncertainty, $w$ denotes noises, disturbances, and reference signals, $z$ contains all controlled signals and tracking errors, $u$ is the control signal, and $y$ is the output measurement [5].
Figure 3.1: LFT framework

Figure 3.1 corresponds to (3.1).

\[
\begin{bmatrix}
\nu \\
z \\
y
\end{bmatrix} = P \begin{bmatrix}
\eta \\
w \\
u
\end{bmatrix}
\]

(3.1)

\[\eta = \Delta \nu\]

\[u = Ky\]

If $T_{zw}$ is the transfer matrix from $w$ to $z$, the $H_\infty$ norm $\|T_{zw}\|_\infty$ is defined as

\[
\|T_{zw}\|_\infty = \sup_w \sigma(T_{zw}(jw))
\]

(3.2)

An interpretation of this is that the $H_\infty$ norm describes the maximum effect that noises, disturbances, and uncertainties can have upon controlled signals in some frequency range. This corresponds to the peak of the Bode magnitude plot of $|T_{zw}(jw)|$.
which is the maximum amplification of the steady state value of $z$ due to a sinusoidal excitation $w$.

Assuming that $\Delta$ satisfies $\bar{\sigma}(\Delta) < 1/\gamma_u$ for some $\gamma_u > 0$, the $H_\infty$ synthesis problem is to design a controller $K$ that stabilizes the closed-loop system for all admissible $\Delta$ and $\|T_{zw}\|_\infty \leq \gamma_u$.

For stability considerations in $H_\infty$ control, the synthesis objective is to specify a nominal linear model and a bound (weighting function) on the uncertainty between that model and the actual system. The $H_\infty$ controller will then be guaranteed to internally stabilize the set of all plants bounded by the nominal model plus the uncertainty. Then the closed loop feedback system has robust stability. For a plant under unstructured perturbations (model uncertainty is generally treated as unstructured), the basis for the robust stability criteria is the small-gain theorem.

**Theorem 1** Suppose $M \in RH_\infty$ and let $\gamma > 0$. The interconnected system $(P, \Delta)$ is well-posed and internally stable for all $\Delta(s) \in RH_\infty$ with $\|\Delta\|_\infty \leq 1/\gamma$ if and only if $\|M(s)\|_\infty \leq \gamma$.

The small gain theorem and its proof can be found in [5].

### 3.2 Presence of Modeling Error

The system model (2.23) is based upon a linearization of electromagnetic nonlinearities about a nominal setpoint (the center of the bearings). As such, this model can never fully capture the system dynamics. This modeling error can be revealed by comparing the outputs of the actual system with those of its linearized model,
given the same inputs. However, because the model is unstable, it is not possible to generate a stable simulated output using the measured control input. It is, however, possible to validate the model using k-step ahead prediction [26]. Figures 3.2 and 3.3 compare the measured outputs (the orbit of bearing $b$ at 4 $krpm$) $r(\theta)$ to the 1-step ahead and 10-steps ahead modeled predictions $r_p(\theta)$, respectively. The prediction performance is noticeably reduced with the larger horizon (10 steps ahead), indicating the presence of modeling uncertainty. This uncertainty must be accounted for in the design of robust controllers.
Figure 3.3: $k$-step prediction for $k = 10$

3.3 Uncertainty Bound Specification

For $H_{\infty}$ control synthesis, a representation of model uncertainty is required to bound the differences between the actual plant (which has an infinite number of states) and the linear model used for control design (which has a small, finite number of states). These unmodeled dynamics typically result in high-frequency uncertainties that are best formulated as unstructured uncertainties because they are difficult to express in highly-structured parameterized forms [5].

Two common unstructured representations of modeling uncertainty are additive uncertainty (Figure 3.4)

$$G_{\Delta a}(s) = G_0(s) + W_a(s)\Delta_a(s)$$

$$\Delta_a(s) = G(s) + G_0(s)$$

(3.3)
Chapter 3. Robust $H_{\infty}$ Control of Active Magnetic Bearings

Figure 3.4: Additive model uncertainty

and multiplicative uncertainty (Figure 3.5)

\[
G_{\Delta m}(s) = G_0(s)(1 + W_m(s)\Delta_a(s))
\]

\[
\Delta_m(s) = \frac{G(s) + G_0(s)}{G_0(s)}
\]  

(3.4)

Here $W_{\Delta a}$ and $W_{\Delta m}$ (generalized by $W_{\Delta}$) are weighting functions that describe the spatial and frequency characteristics of the uncertainty and define a neighborhood about the nominal model $G_0$ inside which the actual infinite-order plant resides [5]. For the AMB system, the additive case will be used to represent model uncertainty.

Typically, uncertainty weighting functions are chosen arbitrarily or through lengthy trial and error procedures that can result in overly conservative error bounds to guarantee stability. Textbooks on robust control provide general guidelines for specifying bounds on modeling uncertainty $W_{\Delta}$, tracking performance $W_p$, and control input $W_{in}$ as functions of frequency.

For this application, all three weighting functions were constructed based on a
priori knowledge of the AMB system dynamics and trial and error adjustments. The shape of $W_\Delta$ (Figure 3.6), a high-pass filter, was selected based on the assumption that unmodeled dynamics are more prevalent at high frequencies, and its magnitude was adjusted to ensure that $|W_\Delta(j\omega)| \geq \Delta_a(j\omega)$ for all frequencies. $W_p$ penalizes steady-state error, which generally has low-frequency properties. By making its shape a low-pass filter, the controller will focus on minimizing low-frequency errors. Similarly, it is desirable to reduce the high-frequency content of the control signal, so $W_{in}$ has the shape of a high-pass filter (Figure 3.6).

### 3.4 Nominal $H_\infty$ Control: Experimental Results

The AMB system (Figure 2.3) was used to experimentally validate the robust stability and performance of $H_\infty$ control. A nominal $H_\infty$ controller was designed using MATLAB’s LMI Control Toolbox and the system interconnection in Figure 3.7. This
Chapter 3. Robust $H_\infty$ Control of Active Magnetic Bearings

controller was implemented using MATLAB’s xPC Target at a fixed timestep $0.1$ \textit{msec}. The rotor was spun to the \textit{nominal} speed $4 \text{ krpm}$, at which gyroscopic effects were modeled. Nominal orbit plots for both bearings are shown in Figure 3.8.

Clearly, this nominal controller stabilized the plant dynamics and provided reasonable tracking performance, as quantified by the sum of squared errors (SSE) for both bearings:

$$SSE = \sum_{k=1}^{N} (r_{a_d}(kT) - r_a(kT))^2 + (r_{b_d}(kT) - r_b(kT))^2$$ \hspace{1cm} (3.5)

where the desired orbital position $r_d(kT)$ is zero, since the control objective is to regulate the rotors to the centers of the bearings. Here $T$ is the discrete sample time, $0.1 \text{ msec}$. The SSE for Figure 3.8, with $N = 10,000$, is $3.02 \times 10^{-5} \text{m}^2$. 

**Figure 3.6**: Nominal weighting functions
To investigate the robustness of this nominal $H_\infty$ controller, the rotational speed was varied away from the nominal value of 4 $krpm$. Since the system model (2.23) is dependent upon rotor speed, the AMB system would then be operating away from where the nominal $H_\infty$ controller was designed (2.23). Figures 3.9 and 3.10 show that at a speed lower than nominal, 2 $krpm$, and higher than nominal, 6 $krpm$, respectively, stability was maintained, but tracking performance deteriorated. The SSE ($N = 10,000$) for Figure 3.9 is $6.35 \times 10^{-5} m^2$, a 110% increase over the nominal case, while the SSE for Figure 3.10 is $7.05 \times 10^{-5} m^2$, a 133% increase. The performance indices for these and other speeds are shown in Table 3.1. For an additional test of robust stability and performance, model uncertainty is further increased at the 2 $krpm$ speed by adding a small mass ($3.0 \times 10^{-3} kg$) to one of the balance disks on the rotor (Figure 2.3). The result was a continued increase in tracking error, as shown in Figure 3.11. The associated SSE was $1.09 \times 10^{-4} m^2$, a 261% increase over the
There is a clear trend, as shown in Figure 3.12. As the rotational speed of the system moves away from the nominal speed 4 \textit{krpm}, tracking error tends to increase. These reductions in tracking performance illustrate the benefits of operating the AMB system near the location of the linear model, as well as the need for model uncertainty to be quantified pertaining to robust control.

\textbf{Figure 3.8}: Rotor orbit - Nominal $H_\infty$ controller at 4 \textit{krpm}
Chapter 3. Robust $H_\infty$ Control of Active Magnetic Bearings

Figure 3.9: Rotor orbit - Nominal $H_\infty$ controller at 2 krpm

Figure 3.10: Rotor orbit - Nominal $H_\infty$ controller at 6 krpm
Figure 3.11: Rotor orbit - Nominal $H_\infty$ controller at 2 krpm w/ added mass

Figure 3.12: Sum of squared errors (nominal) versus rotational speed
Table 3.1: Performance indices for nominal $H_\infty$ control

<table>
<thead>
<tr>
<th>Rotor speed $(krpm)$</th>
<th>SSE $(m^2)$</th>
<th>% change from Nominal SSE</th>
<th>Peak Error $(m)$</th>
<th>% change from Nominal PE</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>$6.35 \times 10^{-5}$</td>
<td>+110.3</td>
<td>$1.51 \times 10^{-4}$</td>
<td>+36.4</td>
</tr>
<tr>
<td>3</td>
<td>$5.05 \times 10^{-5}$</td>
<td>+67.2</td>
<td>$1.45 \times 10^{-4}$</td>
<td>+31.8</td>
</tr>
<tr>
<td>4 (Nominal)</td>
<td>$3.02 \times 10^{-5}$</td>
<td>–</td>
<td>$1.10 \times 10^{-4}$</td>
<td>–</td>
</tr>
<tr>
<td>5</td>
<td>$3.77 \times 10^{-5}$</td>
<td>+24.8</td>
<td>$1.23 \times 10^{-4}$</td>
<td>+11.8</td>
</tr>
<tr>
<td>6</td>
<td>$7.05 \times 10^{-5}$</td>
<td>+133.4</td>
<td>$1.58 \times 10^{-4}$</td>
<td>+43.6</td>
</tr>
<tr>
<td>7</td>
<td>$7.15 \times 10^{-5}$</td>
<td>+136.8</td>
<td>$1.85 \times 10^{-4}$</td>
<td>+68.1</td>
</tr>
<tr>
<td>2 (w/ added mass)</td>
<td>$1.09 \times 10^{-4}$</td>
<td>+260.9</td>
<td>$1.99 \times 10^{-4}$</td>
<td>+80.9</td>
</tr>
</tbody>
</table>
Chapter 4

Estimating Model Uncertainty for Active Magnetic Bearings

The nominal $H_\infty$ controller utilized a somewhat arbitrary uncertainty weighting function for synthesis. The shape of this weighting function was based on the assumption that unmodeled dynamics are more prevalent at high frequencies, and its magnitude was set with the benefit of extensive \textit{a priori} information. Experimental results appear to indicate that this weighting function was conservative enough (defined an adequately large neighborhood about the nominal model $G_0$ inside which the actual infinite-order plant resides) to provide robust stability for a reasonable range of rotational speeds.

However, as the size of the set of stabilizable plants increases, tracking performance tends to decrease, as confirmed by the SSEs of Table 3.1. Thus, it is advantageous to select uncertainty bounds that are accurate, but not overly conservative, for robust stability and improved performance. By using identification techniques to estimate the modeling error and construct an uncertainty weighting function, the plant should
lie just inside the bound of stabilizable model sets, thus minimizing uncertainty bound conservativeness and optimizing controller performance. Two methods for estimating the uncertainty weighting function are presented here: a parametric approach based on recursive least squares (RLS) techniques and a non-parametric approach based on artificial neural networks.

### 4.1 System Identification

System identification is the process of estimating a linear mapping from system input(s) to system output(s). The resulting model is then a description of the system dynamics, useful for purposes such as control design or determination of system parameters. Classical approaches to system identification use linear regression techniques to estimate a parametric system model using a finite data history of system inputs and outputs [26]. A typical “black box” approach is shown in Figure 4.1. To develop the parametric system model, the system inputs and outputs can be related

![Figure 4.1: Control and system identification diagram](image-url)
with a linear ARX (autoregressive with exogenous input) model

\[ y(t) + a_1 y(t-1) + a_2 y(t-2) + \ldots + a_n y(t-n) = b_1 u(t-1) + b_2 u(t-2) + \ldots + b_m u(t-m) \]  

(4.1)

There exists an optimal parameter vector

\[ \theta = [a_1, a_2, \ldots, a_n, b_1, b_2, \ldots, b_m]^T \]  

(4.2)

that when used in (4.1), will best predict the current system output \( y(t) \) given a history of inputs and outputs, or \textit{regressors}, expressed as

\[ \psi(t) = [-y(t-1), -y(t-2), \ldots, -y(t-n), u(t-1), u(t-2), \ldots, u(t-m)]^T \]  

(4.3)

Then (4.1) can be written

\[ \hat{y}(t) = \psi^T(t) \theta \]  

(4.4)

The recursive least squares (RLS) estimate of the optimal system parameter vector \( \hat{\theta}(t) \) can be determined using the algorithm [26]

\[
\hat{\theta}(t) = \hat{\theta}(t-1) + \mu_t R_t^{-1} \psi(t, \hat{\theta}(t-1)) \nu(t, \hat{\theta}(t-1)) \\
\nu(t, \theta) = y(t) - \hat{y}(t|\theta) \\
R_t = R_{t-1} + \mu_t \left[ \psi(t, \hat{\theta}(t-1)) \psi^T(t, \hat{\theta}(t-1)) - R_{t-1} \right]
\]

(4.5)

4.2 Model Error Modeling

The black box identification approach outlined above can be easily tailored to estimate model uncertainty. Instead of estimating the linear mapping from input(s) to
output(s), the mapping is estimated from inputs to modeling error. This process is called model error modeling (MEM) [16], and is illustrated in Figure 4.2.

Parametric system identification equations (4.1) to (4.5), which are commonly used to identify dynamic models, can be modified to identify the modeling error between the actual nonlinear plant and the identified linear model. These modifications are:

1. The signal to be predicted is the difference between the output of the plant and the nominal model $e(t) = y(t) - \hat{y}(t)$

2. The system to be identified, the model error model, represents the unmodeled dynamics, and has input $u(t)$ and output $\hat{e}(t)$
The error model inputs and outputs can be related with a linear ARX (autoregressive with exogenous input) model

\[ e(t) + a_1 e(t-1) + a_2 e(t-2) + \ldots + a_n e(t-n) = b_1 u(t-1) + b_2 u(t-2) + \ldots + b_m u(t-m) \]  

(4.6)

There exists an optimal parameter vector

\[ \theta = [a_1, a_2, \ldots, a_n, b_1, b_2, \ldots, b_m]^T \]  

(4.7)

that when used in (4.6), will best predict the current system output \( e(t) \) given a history of inputs and outputs expressed as

\[ \phi(t) = [-e(t-1), -e(t-2), \ldots, -e(t-n), u(t-1), u(t-2), \ldots, u(t-m)]^T \]  

(4.8)

Then (4.6) can be written

\[ \hat{e}(t) = \phi^T(t) \theta \]  

(4.9)

The recursive least squares (RLS) estimate of the optimal system parameter vector \( \hat{\theta}(t) \) can be determined using the algorithm [26]

\[ \hat{\theta}(t) = \hat{\theta}(t-1) + \mu_t R_t^{-1} \phi(t, \hat{\theta}(t-1)) \nu(t, \hat{\theta}(t-1)) \]

\[ \nu(t, \theta) = e(t) - \hat{e}(t|\theta) \]  

(4.10)

\[ R_t = R_{t-1} + \mu_t \left[ \phi(t, \hat{\theta}(t-1)) \phi^T(t, \hat{\theta}(t-1)) - R_{t-1} \right] \]

For the AMB test rig, the goal is to bound model uncertainty associated with each output. Thus, the model error modeling procedure described above can be used to provide linear mappings from each of the four system inputs to their associated error outputs. To illustrate this, nominal H_{\infty} control was used to regulate the rotor
to the center of the bearings, whereupon the rotor was spun to the nominal speed 4 krpm. Using the technique outlined in (4.1) to (4.5) and Figure 4.2, identification of a 6th-order ARX error model \((n = m = 6)\) was initiated using input-output data for the \(x\)-axis of bearing \(a\).

For use in an \(H_\infty\) uncertainty weighting function, the frequency response of the identified error model is required. However, since the model was identified from finite data, some reasonable guarantee of the model’s accuracy must be obtained. The statistical properties of the data can be used to calculate upper and lower confidence bounds, which enclose a confidence region about the frequency response of the model [17]. For example, a 99.9% confidence region represents the interval in which there is a 99.9% probability that the actual system error response lies. Figure 4.3 shows a Bode plot of the identified model error model with its 99.9% confidence region. Since the nominal model lies inside the uncertainty region defined by these bounds, both the model and MEM confidence bounds are suitable for \(H_\infty\) control synthesis.

### 4.3 Intelligent Model Error Identification

For this research application, intelligent model error identification can be viewed as a non-parametric regression problem in the frequency domain involving feedforward neural networks. One specific type of network, the Radial Basis Function Network (RBFN), is well suited to the task of real-time system identification for two reasons. First, the network uses localized activation functions (radial basis functions), and thus learns information in a localized fashion [9]. As a result, parameter estimates
obtained from a small region of the plant operating space do not adversely affect estimates from other regions. Second, the interconnection weights (which are self-adjusted during learning) are applied linearly on the output side of the network. This feature significantly simplifies the training equations and reduces the computational requirements of RBFNs, making them well suited for real-time implementation.

The goal of intelligent model error identification is to estimate the upper magnitude bound (99.9% confidence interval) of the model error frequency response function

\[ |G_e(j\omega)| = \frac{|E(j\omega)|}{|U(j\omega)|} \]  \hspace{1cm} (4.11)

Stochastic estimates of this frequency response function (FRF) can be obtained by dividing the Fast Fourier Transform (FFT) of prediction error \( e(t) \) by the FFT of control input \( u(t) \). Note that there are plant nonlinearities and non-deterministic effects
Figure 4.4: Intelligent model error identification using a non-recurrent RBFN that result in frequency dependent confidence intervals associated with $|G_e(j\omega)|$.

Figure 4.4 shows one non-parametric approach for estimating the model error FRF $|G_e(j\omega)|$ using sampled input-output data and a single RBFN. The RBFN provides a regression estimate of the model error magnitude $\hat{G}_e(\omega, \mathbf{w})$ that is conditioned on the input frequency $\omega$ and the network weights $\mathbf{w}$. Using this approach, the estimated magnitude corresponding to frequency $\omega$ is:

$$\hat{G}_e(\omega, \mathbf{w}) = \mathbf{w} \cdot e^{-\left(\frac{\omega - c}{s}\right)^2}$$  \hspace{1cm} (4.12)

where $c$ represents the normalized activation center vector and $s$ represents the activation width. The RBFN (4.12) can be “trained” to minimize a cost function of prediction errors by adjusting the network weights $\mathbf{w}$. The standard quadratic error
cost function for $P$ input-output training patterns is:

$$e(i) \equiv |G_e(j\omega(i))| - \hat{G}_e(\omega(i), w)$$

$$J(e) \equiv \frac{1}{2} \sum_{i=1}^{P} e(i)^2$$ \hspace{1cm} (4.13)

The optimal weight vector $w$ could be determined using a closed-form, off-line approach (where complete “batches” of input-output data are available). For on-line implementation, a “backpropagation of error” approach is used to update the weights according to the negative error gradient [27]:

$$w_{new} = w_{old} - \eta \cdot \frac{\partial J(e)}{\partial w}$$

$$\frac{\partial J(e)}{\partial w} = -\sum_{i=1}^{P} \left(|G_e(j\omega(i))| - \hat{G}_e(\omega(i), w)\right) \cdot e^{-\left(\frac{\omega - \omega_c}{s}\right)^2}$$ \hspace{1cm} (4.14)

The training gain $\eta$ is selected to provide rapid, yet stable convergence during training.

Because of the symmetric nature of the error cost function (4.13), a network trained using standard backpropagation (4.14) should converge to the average FRF magnitude. To illustrate this point, an RBFN was trained to identify the average error magnitude of the linear system model across the frequency range 0.0 - 5000 Hz (5000 Hz represents the Nyquist frequency for data collected at 0.1 msec). A simple network pruning approach indicated that 94 network activation functions provided both estimation accuracy and computational efficiency. The 94 activation centers $c$ were distributed uniformly across the input domain (0 - 5000 Hz) and the activation width $s$ was set to 56 Hz. The network weights $w$ were initialized to zero. Input-output data for the $x$-axis of bearing $a$ was collected from the AMB system and used to generate sequential 16384-point FRFs of model error $G_e(j\omega)$. Using a training gain of $1.0 \times 10^{-4}$, the error cost function $J(e)$ dropped from $1.05 \times 10^{-3}$ to $4.14 \times 10^{-5}$ in
100 training epochs, as a smooth approximation of the average FRF magnitude was achieved (Figure 4.5).

Also plotted in Figure 4.5 is the computed average magnitude across 100 FRFs. Note that the RBFN provides a much smoother estimate of this mean, and requires far less computational resources to approximate it (only 94 network weights, as opposed to an 819,200 element double precision array).

If the standard quadratic error cost function (4.13) is replaced with an asymmetric, bilinear error cost function, the learning characteristics of the RBFN will be dramatically changed [28]. The quadratic cost function causes network outputs to converge to the statistical mean of the model error magnitudes because positive and negative errors are equally weighted. Using an asymmetric bilinear cost function, it is
possible to weight positive and negative errors differently. Consider the asymmetric cost function described by

\[ e(i) \equiv \left( |G_e(j\omega(i))| - \hat{G}_e(\omega(i), \mathbf{w}) \right) \]

\[ c_H(i) \equiv J(e) = \begin{cases} 
 a \cdot e(i) & e(i) \geq 0 \\
 1 \cdot e(i) & e(i) < 0 
\end{cases} \]

\[ J_H(e) \equiv \sum_{i=1}^{p} c_H(i) \]  

As before, this network can be trained by updating the weights in the negative error gradient:

\[ \mathbf{w}_{\text{new}} = \mathbf{w}_{\text{old}} - \eta \cdot \frac{\partial J_H(e)}{\partial \mathbf{w}} \]

\[ \frac{\partial J_H(e)}{\partial \mathbf{w}} = -\sum_{i=1}^{p} \begin{pmatrix} e^{-(\frac{\omega - e}{a})^2} \cdot \begin{cases} a & e(i) \geq 0 \\
 1 & e(i) < 0 \end{cases} \end{pmatrix} \]  

By selecting a unit cost constant \( a = 1 \), positive and negative prediction errors will be evenly weighted, and an RBFN will converge to the statistical median of the training data (if the data is normally distributed, this will coincide with the mean) [28]. This occurs because each weight update depends not on the magnitude of error, but on its \( \pm \) sign. Thus, each weight update is of a consistent magnitude, with a direction depending on the \( \pm \) sign of the prediction error. Whenever the likelihood of positive prediction errors equals the likelihood of negative prediction errors, the weight update process will chatter back and forth about a stable equilibrium point. If
the training rate $\eta$ is then reduced so $\eta \to 0$, the chattering will reduce likewise. The weights will settle and the learning will converge upon the median of the distributed training data. A formal statistical proof of this process can be found in Appendix 1.

By selecting a larger cost constant $a > 1$, an RBFN trained using this error cost function will consistently overestimate the median of distributed training data. Similarly, if $a < 1$, an RBFN will underestimate the median of distributed training data. In fact, it is possible to select this cost constant so that an RBFN will approximate a desired confidence interval associated with randomly distributed training data, resulting in a Confidence Interval Network (CIN) [28]. To estimate the 99.9% confidence interval, the cost constant $a$ must be selected to equally weight 999 negative prediction errors and 1 positive prediction error, thus $a = 999$.

Recall the RBFN used to estimate the average error magnitude of the nominal linear model (Figure 4.4). Now, however, let the quadratic error cost function (4.13) be replaced with the asymmetric bilinear error cost function (4.15), with $a = 999$. Using this modified weight update equation, a CIN was trained to approximate the upper uncertainty bound (99.9% confidence interval) between the linear model (2.23) and the actual plant (Figure 2.3). As before, the 94 activation centers $c$ were distributed uniformly across the input domain (0.0 - 5000 Hz), and all activation widths $s$ were set to 56 Hz. The network weights $w$ were initialized to zero. Input-output data collected from the AMB system was used to generate sequential 16384-point FRFs of model error. Using a training gain of $1.0 \times 10^{-6}$, the error cost function $J_H(e)$ dropped from $11.32 \times 10^1$ to $8.9 \times 10^{-1}$ in 100 training epochs, resulting in a smooth approximation of the 99.9% confidence interval, as shown in Figure 4.6.
As an aside, the stochastic FRF data in Figures 4.5 and 4.6 contain noise-induced, periodic “anti-nodal” behavior most visible at 67 Hz but reoccurring every 67 Hz thereafter. Though undesirable, this noise effect illustrates another benefit of the CIN bounding technique. The learned 99.9% confidence interval does not fall into the local minima induced by the noise, but generalizes over it to consistently bound the data, especially at higher frequencies. However, the calculated confidence interval does follow the local minima, providing an erroneous and less smooth uncertainty bound.

**Figure 4.6**: CIN estimate of uncertainty bound (99.9% confidence interval of model error magnitude)
Chapter 5

Identified $H_\infty$ Control: Experimental Results

To investigate the benefits of identified uncertainty weighting functions on robust stability and performance, experiments were conducted at a variety of operating conditions. As shown in Chapter 3, the performance of the AMB system (Figure 2.3) degrades at rotational speeds away from the one used in the nominal model (2.23), due in part to the model’s gyroscopic dependency on rotor speed $\dot{\phi}$. By applying the MEM and CIN uncertainty bounding techniques outlined in Chapter 4, appropriate modeling error bounds were identified using the nominal experimental data for 4 krpm, 2 krpm, 6 krpm, and 2 krpm with the added mass ($3.0 \times 10^{-3}$ kg added to one of the balance disks on the rotor, as described in Chapter 3). The performance of the controllers resulting from the identified bounds for each case will be assessed and compared with that of the nominal cases.
Chapter 5. Identified $H_\infty$ Control: Experimental Results

5.1 Uncertainty Bound Specification

5.1.1 MEM bounds

The additive uncertainty weighting function ($W_\Delta$) required for $H_\infty$ control synthesis can be derived directly from the identified confidence intervals like those in Figures 4.3 and 4.6, instead of being constructed arbitrarily. To assess the robust control benefits associated with identified bounds, two additional $H_\infty$ controllers were synthesized for each rotational speed. The uncertainty weighting function for the MEM controller was a second-order transfer function derived using MATLAB’s fitsys command which was designed to minimally cover the four MEM uncertainty bounds (with 99.9% confidence intervals), one for each input-output pair in the AMB system. This enabled the identified bounds to be expressed in a low-order framework more suitable for $H_\infty$ synthesis while still capturing the identified frequency-dependent uncertainty at each rotational speed.

Figures 5.1 - 5.4 show the MEM uncertainty bounds and their associated fitsys bounds ($W_\Delta$) for identification at 4 krpm (Nominal speed), 2 krpm, 6 krpm, and 2 krpm with added mass, respectively. Here, only the mappings to the FFT of the error from the FFT of its corresponding input (point receptance, such as $E_1/U_1$) are shown, while the mappings to FFT of error from other inputs (cross receptance, such as $E_1/U_3$) are not shown because their associated uncertainty bounds are significantly of less magnitude than the point receptances, exemplified in Figure 5.5 for the $x$ axis of bearing $a$ at 4 krpm.

To illustrate how the amount of model uncertainty varies across the range of operating speeds, Figure 5.6 compares the $W_\Delta$ for each case (obtained by covering
Figure 5.1: Identified MEM bounds for 4 krpm (Nominal speed)

Figure 5.2: Identified MEM bounds for 2 krpm
Chapter 5. Identified $H_{\infty}$ Control: Experimental Results

Figure 5.3: Identified MEM bounds for 6 krpm

Figure 5.4: Identified MEM bounds for 2 krpm w/ added mass
Figure 5.5: Point and cross receptances for MEM bounds

all identified bounds using MATLAB’s fitsys command) from Figures 5.1 - 5.4. The relative magnitudes of these MEM-identified uncertainty bounds show that, generally, as the rotational speed of the system moves away from the nominal speed 4 krpm used in modeling, the amount of model uncertainty increases. The 2 krpm and 2 krpm with added mass cases are nearly the same at high frequencies, and the 6 krpm case is slightly higher; all three, however, are higher than the 4 krpm case.
Figure 5.6: Comparison of MEM $W_\Delta$ bounds
5.1.2 CIN bounds

The uncertainty weighting function for the CIN controller was a second-order transfer function constructed similarly to minimally cover the four 99.9% confidence intervals derived from intelligent model error identification (Figure 4.4) for each speed. Figures 5.7 - 5.10 show the CIN uncertainty bounds and their associated \( \text{fitsys} \) bounds \((W_{\Delta})\) from identification at 4 krpm (Nominal), 2 krpm, 6 krpm, and 2 krpm with added mass, respectively. As with the MEM case, only the point receptances are used, due to their higher magnitudes compared to the cross receptances, exemplified in Figure 5.11 for the \( x \) axis of bearing \( a \) at 4 krpm.

To illustrate how the amount of model uncertainty varies across the range of operating speeds, Figure 5.12 compares the \( W_{\Delta} \) for each case (obtained by covering all identified bounds using MATLAB’s \text{fitsys} \) command) from Figures 5.7 - 5.10. The relative magnitudes of these CIN-identified uncertainty bounds generally seem to indicate that as the rotational speed of the system moves away from the nominal speed 4 krpm used in modeling, the amount of model uncertainty increases. The 4 krpm and 2 krpm cases are nearly the same at high frequencies, while the 2 krpm with added mass and 6 krpm cases are significantly higher.
Figure 5.7: Identified CIN bounds for 4 krpm (Nominal speed)

Figure 5.8: Identified CIN bounds for 2 krpm
Figure 5.9: Identified CIN bounds for 6 krpm

Figure 5.10: Identified CIN bounds for 2 krpm w/ added mass
Figure 5.11: Point and cross receptances for CIN bounds

Figure 5.12: Comparison of CIN $W_\Delta$ bounds
5.2 Controller Implementation

The AMB system (Figure 2.3) was used to compare the robust stability and performance of the nominal and identified $H_\infty$ controllers. When calculating the identified controllers, the performance and control input weighting functions ($W_p$ and $W_{in}$) were the same as those used for nominal $H_\infty$ synthesis, shown in Figure 3.6. Then, the appropriate identified uncertainty weighting functions $W_\Delta$ were used for their respective cases. These identified $H_\infty$ controllers were designed using MATLAB’s LMI Control Toolbox as before.

Figures 5.13 and 5.14 show the tracking performance of the AMB system at 4 $krpm$ for bearings $a$ and $b$, respectively, using the nominal controller, the MEM controller, and the CIN controller. These figures appear to show modest improvements in tracking performance due to the use of the identified uncertainty weighting functions ($W_\Delta$). Table 5.1 quantifies these improvements – the MEM controller reduced the SSE index by 48.7%, and the CIN controller by 29.1%.

When the AMB system operates away from 4 $krpm$, the uncertainty bounding techniques have a greater influence upon the system performance. Figures 5.15 and 5.16 show the tracking performance of the AMB system slower than the nominal speed, at 2 $krpm$, for bearings $a$ and $b$, respectively. Here, the tracking performance using the identified controllers is greatly improved over the nominal case – the MEM controller reduced the SSE index by 87.9%, and the CIN controller by 96.8%.

Similarly, as the AMB system is operated faster than the nominal speed, at 6 $krpm$, the identified controllers improve the system performance over the nominal case (Figures 5.17 and 5.18). The MEM controller reduced the SSE index by 23.8%,
and the CIN controller by 43.7%. While the orbits were larger with the identified controllers, the tracking was centered in the bearings rather than oscillating away from the bearing centers as in the nominal case.

The remaining case places additional tests on the ability of the MEM and CIN identification techniques to bound uncertainty. Here the system was operated at 2 krpm, away from the nominal model, but there is also an unmodeled mass added to the rotor shaft. The performance resulting from the use of the identified error bounds in controller synthesis can be seen in Figures 5.19 and 5.20. Both identified controllers greatly improve the tracking performance over the nominal case – the MEM controller reduced the SSE index by 87.3%, and the CIN controller by 90.6%.

Table 5.1 shows both the SSE and the peak tracking error indices for all cases.

![Figure 5.13: Bearing a rotor orbit - Nominal and identified controllers at 4 krpm](image)
Chapter 5. Identified $H_{\infty}$ Control: Experimental Results

Figure 5.14: Bearing $b$ rotor orbit - Nominal and identified controllers at 4 krpm

Figure 5.15: Bearing $a$ rotor orbit - Nominal and identified controllers at 2 krpm
Figure 5.16: Bearing $b$ rotor orbit - Nominal and identified controllers at 2 $krpm$

Figure 5.17: Bearing $a$ rotor orbit - Nominal and identified controllers at 6 $krpm$
Chapter 5. Identified $H_\infty$ Control: Experimental Results

Figure 5.18: Bearing $b$ rotor orbit - Nominal and identified controllers at 6 $krpm$

Figure 5.19: Bearing $a$ rotor orbit - Nominal and identified controllers at 2 $krpm$ w/ added mass
Chapter 5. Identified $H_\infty$ Control: Experimental Results

Figure 5.20: Bearing $b$ rotor orbit - Nominal and identified controllers at 2 krpm w/ added mass

Table 5.1: Performance indices for 4 krpm

<table>
<thead>
<tr>
<th>Controller type</th>
<th>SSE ($m^2$)</th>
<th>% change from Nominal</th>
<th>Peak Error ($m$)</th>
<th>% change from Nominal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nominal</td>
<td>$3.02 \times 10^{-5}$</td>
<td>–</td>
<td>$1.10 \times 10^{-4}$</td>
<td>–</td>
</tr>
<tr>
<td>MEM</td>
<td>$1.55 \times 10^{-5}$</td>
<td>-48.7</td>
<td>$7.62 \times 10^{-5}$</td>
<td>-30.7</td>
</tr>
<tr>
<td>CIN</td>
<td>$2.14 \times 10^{-5}$</td>
<td>-29.1</td>
<td>$8.83 \times 10^{-5}$</td>
<td>-19.7</td>
</tr>
</tbody>
</table>

Table 5.2: Performance indices for 2 krpm

<table>
<thead>
<tr>
<th>Controller type</th>
<th>SSE ($m^2$)</th>
<th>% change from Nominal</th>
<th>Peak Error ($m$)</th>
<th>% change from Nominal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nominal</td>
<td>$6.35 \times 10^{-5}$</td>
<td>–</td>
<td>$1.51 \times 10^{-4}$</td>
<td>–</td>
</tr>
<tr>
<td>MEM</td>
<td>$7.70 \times 10^{-6}$</td>
<td>-87.9</td>
<td>$5.98 \times 10^{-5}$</td>
<td>-60.4</td>
</tr>
<tr>
<td>CIN</td>
<td>$2.01 \times 10^{-6}$</td>
<td>-96.8</td>
<td>$3.04 \times 10^{-5}$</td>
<td>-79.9</td>
</tr>
</tbody>
</table>
5.3 Advantages of the Confidence Interval Network for Uncertainty Identification

The experimental data summarized in Tables 5.1 – 5.4 clearly show that identification of the uncertainty weighting function $W_\Delta$ generally results in better tracking performance compared to an *ad hoc* $W_\Delta$ when applied to $H_\infty$ synthesis. Varying amounts of uncertainty were identified using both the model error modeling and confidence interval network techniques, and the resulting controllers yielded excellent results.

In addition to the performance benefits, there is another significant advantage to identifying model uncertainty. For initial $H_\infty$ control of the AMB system, the nominal uncertainty weighting function was developed by lengthy trial and error experimentation, approximately 900 man-hours. The *a priori* assumption was that its shape should be high-pass, but much adjusting was required to first achieve stable
levitation, then to obtain acceptable performance for a range of rotational speeds. After initial uncertainty identification with both the MEM and CIN methods, which use only system data and do not consider *a priori* information, stability and improved performance were obtained extremely quickly relative to the nominal case, in approximately 40 man-hours.

There are fundamental differences in the implementation of the two identification techniques, however. The MEM method is highly computationally intensive due to the operations required in the RLS equations (4.5). In addition, the quality of the fit is directly related to the order of the identified model, easily resulting in lengthy identification. Alternatively, the CIN method can provide the benefits of identified uncertainty while obtaining the bounds in relatively short time with computational ease. This is due to linear application of a relatively small number of network weights and simple weight update (network training) equations (4.14). In short, the confidence interval network can provide results comparable to current model error modeling techniques, except with a faster and more versatile implementation.
Chapter 6

Conclusions

The primary purpose of this work was to provide an innovative, intelligent, and experimentally-validated technique to identify model uncertainty for use in robust $H_\infty$ control.

6.1 Contributions

A novel soft computing approach has been demonstrated to estimate bounds of model uncertainty resulting from parameter variations, unmodeled dynamics, and non-deterministic processes in dynamic plants. A variation on model error modeling (MEM) techniques, this approach uses confidence interval networks (CINs), radial basis function neural networks trained using asymmetric bilinear error cost functions, to estimate confidence intervals associated with linear models. This research couples the hard computing features of multivariable $H_\infty$ control (rigid mathematics, predictable stability) with the soft computing characteristics of intelligent system identification (generalization, tractability) and realizes the combined advantages of both. Better control performance can be obtained by intelligently “learning” uncertainty bounds
for robust control synthesis rather than making them overly conservative to guarantee system stability. Experimental demonstrations conducted on an active magnetic bearing system confirm these capabilities.

6.2 Future Research Directions

Some possible directions for future research are outlined:

**CIN bounding of stochastic data to design other weighting functions:** This work utilized the CIN technique to bound stochastic uncertainty data and supplied that bound to $H_{\infty}$ synthesis as an uncertainty weighting function. Robust controller synthesis requires two additional weighting functions, *performance* and *control input*. The performance weighting function penalizes steady-state error to improve performance, and the control input weighting function penalizes the control signal to achieve a desired frequency content. Much like the uncertainty weighting function, these are usually designed by using a frequency-dependent shape based upon factors known *a priori*, then using trial and error to tweak them until satisfactory results are obtained. The CIN bounding technique could be extended to intelligently learn appropriate performance and control input weighting functions for robust control.

**Improved cost functions for CIN learning:** The CINs used to bound data with a desired confidence interval require that the learning rate be decreased as learning progresses so the network output will converge to an acceptable level.
This is due to a discontinuity in the bilinear cost function. As learning converges to a minimum solution, the prediction error tends to “chatter” about the discontinuity, and reducing the training rate simply lowers this chatter to an acceptable level. A continuous cost function that is able to weight positive and negative training errors differently would not require a decreasing learning rate for convergence; rather, prediction error would settle in a minimum.

**Improved modeling of the AMB system:** This research used a low-order, analytical, rigid-body, linear model for control synthesis. Various techniques exist to model the flexible nature of the rotor, such as system identification, finite element analysis, and experimental modal analysis. Applying these methods to model the AMB system more accurately would decrease modeling error and would likely improve control performance.

**Gain-scheduled $H_\infty$ control of the AMB system:** The AMB system used for this research has a variable speed range, and gyroscopic effects influence its performance. A gain-scheduling approach using rotational speed as the scheduling variable would implement a different controller depending on the instantaneous speed of the system. By optimizing the control in this fashion, overall system performance, especially near critical speeds, could improve.
List of References


Appendix A

Statistical Analysis of Bilinear Cost Functions

(excerpted from [28])

A.1 Introduction

The Artificial Neural network (ANN) is a highly intensive processing and analysis tool for many scientific problems, including pattern recognition, prediction, and function approximation [29][30]. The most common method to train ANNs is backpropagation of error (BPE), a supervised learning procedure that uses input and output pairs to iteratively adjust the network interconnection weights [31][32]. Traditional BPE seeks to minimize an error cost function of the difference between an actual reference output and the output of the network (prediction errors), given actual inputs. The prediction errors are backpropagated through the output layer and hidden layers, and connection weights are updated to drive prediction error down the cost function curve [30]. When the prediction error cost function is minimized, a correct mapping of
the presented input/output data has been estimated. Quadratic error cost functions are most common, and error minimization is usually performed using such a gradient descent method.

This paper introduces a novel bilinear cost function designed to estimate confidence intervals of stochastic data. The statistical behavior of these confidence interval networks (CINs) using both symmetric and asymmetric bilinear cost functions is analyzed, and their stability and convergence properties are proven. The capabilities of CINs to estimate confidence intervals is validated using experimental stochastic data.

### A.2 Function Approximation via ANNs

Given $x$, assume $y(x)$ is a stochastic function following some distribution $F$, with $E[y|x] = \mu(x)$. The estimated function (network output) is denoted by $\hat{y}(x)$, and the corresponding prediction error by $e(x) = y(x) - \hat{y}(x)$. Using the simplest possible ANN structure, a single network weight $w$ with no hidden layer neurons, the estimate is simply $\hat{y}(x) = w$. BPE seeks to minimize some cost function $J(e)$ by applying gradient descent methods to update the weights. The adaptation rule for $w$ is

$$w = w - \eta \frac{\partial J}{\partial w}, \quad (A.1)$$

where $0 < \eta < 1$ is the learning rate. Traditional BPE utilizes the quadratic error cost function

$$J(e) = \frac{1}{2} e^2 \quad (A.2)$$

illustrated in Figure A.1. The adaptation rule (A.1) becomes

$$w = w + \eta \cdot e \quad (A.3)$$
This paper introduces a new error cost function, which has a bilinear form

\[ J(e) = \begin{cases} 
  c_1 e & \text{if } e \geq 0 \\
  -c_2 e & \text{if } e < 0.
\end{cases} \tag{A.4} \]

Positive constants \( c_1 \) and \( c_2 \) represent penalties for positive and negative prediction errors, respectively. When \( c_1 = c_2 \), the cost function is “symmetric”, as illustrated in Figure A.2. When \( c_1 \neq c_2 \), it is “asymmetric”, as illustrated in Figure A.3.

A BPE adaptation rule for \( w \) with this bilinear cost is given by

\[ w = \begin{cases} 
  w + c_1 \eta & \text{if } e > 0 \\
  w - c_2 \eta & \text{if } e < 0.
\end{cases} \tag{A.5} \]

A fundamental difference between the quadratic and bilinear error cost functions is that BPE converges to a distribution \( F \) depending on constants \( c_1 \) and \( c_2 \). The quadratic error cost function causes convergence to the mean \( \mu(x) \), while bilinear error
Figure A.2: Symmetric bilinear loss function \((c_1 = c_2 = 1)\)

Figure A.3: Asymmetric bilinear loss function \((c_1 = 10, c_2 = 1)\)
cost function causes convergence to some quantile of $F$. In particular, the symmetric bilinear error cost function causes BPE to converge to the median of $F$.

A.3 Quadratic Error Cost Function Training

A.3.1 Proof of convergence

For the simple ANN structure (A.3) the first observation $y_1(x)$ and estimate $\hat{y}_1(x)$ can be readily obtained. The prediction error $y_1 - \hat{y}_1$ can be used to implement the weight update equation (A.5). Another observation $y_2(x)$ and an updated value $w_2$ gives $\hat{y}_2(x)$. This procedure is continued by taking sequential observations $y_2(x), ..., y_n(x)$, then updating the weight $w$ for each new observation, until some stopping criteria is met. The following theorem shows that as $n$ goes to infinity, $\hat{y}_n(x)$ approaches $\mu(x)$.

Without loss of generality, the network weight can be initialized to zero ($w_1 = 0$). For simplicity, denote the approximation at the $n^{th}$ iteration $\hat{y}_n \equiv \hat{y}_n(x)$. The estimation error associated with the $n^{th}$ update at $x$ is $e_n = y_n - \hat{y}_n$. When using the quadratic error cost function $J(e) = \frac{1}{2}e_n^2$, the adaptation rule for $w_{n+1}$ can be obtained via the chain rule

$$w_{n+1} = w_n - \eta \frac{\partial J}{\partial w} \bigg|_{w_n} = w_n - \eta \frac{\partial J}{\partial e_n} \frac{\partial e_n}{\partial \hat{y}_n} \frac{\partial \hat{y}_n}{\partial w_n} = w_n - \eta e_n(-1) = w_n + \eta e_n,$$  
(A.6)

where $\eta$ is a small positive constant. The following lemma is needed.

**Lemma 1** For $n \geq 1$, the $n^{th}$ update of $\hat{y}$ at the design point $x$ has the expression

$$\hat{y}_n = \eta \sum_{i=1}^{n-1} y_i (1 - \eta)^{n-1-i} + (1 - \eta)^{n-1} w_1.$$  
(A.7)
Appendix A. Statistical Analysis of Bilinear Cost Functions

Proof:

For $n = 1$, $\hat{y}_1 = w_1, e_1 = y_1 - \hat{y}_1$, and $w_2 = w_1 + \eta e_1 = w_1 + \eta y_1 - \eta \hat{y}_1$. When $n = 2$,

$$\hat{y}_2 = w_2 = w_1 + \eta y_1 - \eta \hat{y}_1 = \hat{y}_1 + \eta y_1 - \eta \hat{y}_1 = \eta y_1 + \hat{y}_1 (1 - \eta) = \eta y_1 + (1 - \eta) w_1.$$

Therefore (A.7) is true when $n = 1, 2$. Next it will be shown that, if (A.7) holds for any $n \geq 2$ then it also holds for $n + 1$. Using the equations $e_n = y_n - \hat{y}_n, w_{n+1} = w_n + \eta e_n = w_n + \eta y_n - \eta \hat{y}_n$,

$$\hat{y}_{n+1} = w_{n+1} = \hat{y}_n + \eta y_n - \eta \hat{y}_n = \eta y_n + (1 - \eta) \hat{y}_n \quad \text{(by assumption)}$$

$$= \eta y_n + \eta \sum_{i=1}^{n-1} y_i (1 - \eta)^{n-i} + (1 - \eta)^n w_1 = \eta \sum_{i=1}^{n} y_i (1 - \eta)^{n-i} + (1 - \eta)^n w_1.$$

Therefore (A.7) holds for the $n + 1$th update. The lemma is true.

**Theorem 2** Assume $y_1(x), y_2(x), \ldots$ are independently distributed from the same distribution, with mean $\mu(x)$ and variance $\nu(x)^2$. Then the limiting distribution of $\hat{y}_n(x)$ is $N(\mu(x), \frac{n}{2 - \eta} \nu(x)^2)$ as $n \to \infty$.

**Proof:**

For any $n \geq 1$, $E[y_n] = \mu(x), V(y_n) = \nu(x)$. Then

$$E[\hat{y}_n] = \eta \sum_{i=1}^{n-1} \mu(x) (1 - \eta)^{n-1-i} + (1 - \eta)^{n-1} w_1$$

$$= \eta \mu(x) \frac{1 - (1 - \eta)^{n-1}}{1 - (1 - \eta)} + (1 - \eta)^{n-1} w_1$$

$$= \mu(x) [1 - (1 - \eta)^{n-1}] + (1 - \eta)^{n-1} w_1.$$

Since $\lim_{n \to \infty} (1 - \eta)^{n-1} = 0$,

$$\lim_{n \to \infty} E[\hat{y}_n] = \mu(x)$$
Similarly, by the independence of $y_n(x)$’s,
\[
V[\hat{y}_n] = \eta^2 \sum_{i=1}^{n-1} \nu(x)^2 (1 - \eta)^{2(n-1-i)} = \nu(x)^2 \eta^2 \frac{1 - (1 - \eta)^{2(n-1)}}{1 - (1 - \eta)^2}
\]
\[
= \nu(x)^2 [1 - (1 - \eta)^{2(n-1)}] \frac{\eta}{2 - \eta}.
\]
Thus,
\[
\lim_{n \to \infty} V[\hat{y}_n] = \frac{\eta}{2 - \eta} \nu(x)^2.
\]

**A.4 Bilinear Error Cost Function Training**

Consider the bilinear error cost function in (A.4). The constants $c_1 > 0$ and $c_2 > 0$ are respectively the penalty for positive and negative errors. When $c_1 = c_2$, the cost function is “symmetric”. When $c_1 \neq c_2$, the cost function is “asymmetric”.

Given $x$, assume $Y_x(\cdot)$ is a stochastic function following some cumulative distribution function $F_x$ which has the probability density function $f_x$. Using the simplest possible ANN with a single weight $w$, the quantiles of $F_x$ are estimated based on the multiple observations $y_1, ..., y_n, ...$ taken at $x$. The error back-propagation (BP) algorithm minimizes the cost function and updates the weight by $w = w - \eta \frac{\partial J}{\partial w}$.

Starting with some initial value $w = w_0$, each update $\hat{y}_n = w$ is performed as each new observation $y_{n+1}$ is taken. Defining prediction error $e = y_{n+1} - \hat{y}_n$ and using BPE, the adaptation rule of $w$ with bilinear cost is
\[
w = \begin{cases} 
  w + c_1 \eta & \text{if } e > 0 \\
  w - c_2 \eta & \text{if } e < 0.
\end{cases}
\]
where $\eta > 0$ is the unit step length (learning rate).
Appendix A. Statistical Analysis of Bilinear Cost Functions

Let the $p$th quantile of $F_x$ be $\nu_x$, which satisfies $F_x(\nu_x) = p$, or $\nu_x = F_x^{-1}(p), 0 < p < 1$. It will be shown that as $n \to \infty$, the limiting distribution of $\hat{y}_n$ is approximately centered around the quantile $F_x^{-1}\left(\frac{c_1}{c_2+c_1}\right)$. The variance of the limiting distribution depends linearly on $\eta$, thus it can be made arbitrarily small. In particular, if $c_1 = c_2$, the asymptotical mean of $\hat{y}_n$ is approximately the median $m_x = F_x^{-1}(0.5)$.

A.4.1 Symmetric convergence

When $c_1 = c_2$, we have approximately

$$E(\hat{y}) = F_x^{-1}(0.5), \quad (A.8)$$

$$Var(\hat{y}) = \eta \frac{c_1}{4f(\nu_0)}. \quad (A.9)$$

A.4.2 Asymmetric convergence

Moments of Limiting Distribution

Without loss of generality, assume that $c_1$ and $c_2$ are positive integers. The network weight is initialized to $\hat{y}_0 = w_0 = 0$. At the $n$th step, $\hat{y}_n$ is updated by either adding $c_1\eta$ or subtracting $c_2\eta$ with some random scheme. Thus the possible values of $\hat{y}_n$ are the $\eta$-grid points

$$\eta\mathbb{Z} = \{-2\eta, -\eta, 0, \eta, 2\eta, \ldots\} \quad (A.10)$$

over the real line. After the new observation $y_n$ is taken, the state of $\hat{y}_n$ is $\hat{y}_{n-1} + c_1\eta$ if $y_n > \hat{y}_{n-1}$; and is $\hat{y}_{n-1} - c_2\eta$ otherwise. Since the state of $y_n$ depends only on the state of its closet ancestor $y_{n-1}$, the whole process $\{\hat{y}_n, n = 1, 2, \ldots\}$ is a Markov
process with the state space (A.10) and the following transition probability

\[ \text{Prob}(\hat{y}_n = (k + c_1)\eta|\hat{y}_{n-1} = k\eta) = \text{Prob}(y_n > k\eta) = 1 - F(k\eta) \]
\[ \text{Prob}(\hat{y}_n = (k - c_2)\eta|\hat{y}_{n-1} = k\eta) = \text{Prob}(y_n < k\eta) = F(k\eta). \quad (A.11) \]

From here, \( x \) will be omitted from the notation since all the observations and estimations are taken on the fixed \( x \). The stationary distribution \( \pi \) of \( \hat{y} = \lim_{n \to \infty} \hat{y}_n \) will now be investigated.

Define a random variable \( U \) by \( \hat{y} = \eta U \), which implies that

\[ \pi_k = \text{Prob}(\hat{y} = k\eta) = \text{Prob}(U = k). \]

For simplicity, define

\[ q_k \equiv F(k\eta), \quad p_k \equiv 1 - q_k = 1 - F(k\eta). \]

From (A.11), the stationary distribution \( \pi \) of \( \hat{y} \) satisfies

\[ \pi_k = p_{k-c_1} \pi_{k-c_1} + q_{k+c_2} \pi_{k+c_2}, \quad \forall k \in \mathbb{Z} \]

Let \( \phi_U(t) = E(e^{itU}) \) be the characteristic function of \( U \) and \( \phi_\hat{y}(t) \) be that of \( \hat{y} \). For any \( t \in \mathbb{R} \),

\[ \phi_U(t) = \sum_{k=-\infty}^{\infty} e^{ikt} \pi_k = \sum_{k=-\infty}^{\infty} e^{ikt}(p_{k-c_1} \pi_{k-c_1} + q_{k+c_2} \pi_{k+c_2}) \]
\[ = e^{ic_1 t} \sum_{k=-\infty}^{\infty} e^{ikt} p_k \pi_k + e^{-ic_2 t} \sum_{k=-\infty}^{\infty} e^{ikt} q_k \pi_k \]
\[ = (e^{ic_1 t} - e^{-ic_2 t}) \sum_{k=-\infty}^{\infty} e^{ikt} p_k \pi_k + e^{-ic_2 t} \phi_U(t), \]
then
\[
\frac{1 - e^{-ic_2 t}}{e^{ic_1 t} - e^{-ic_2 t}} \tilde{\phi}_U(t) = \sum_{k=-\infty}^{\infty} e^{ikt} p_k \pi_k
\]
(A.12)

Consider the first-order Taylor expansion of \( F \) at \( \nu_0 = F^{-1}(c_1/c_1+c_2) \),
\[
F(k\eta) \equiv F(\nu_0) + f(\nu_0)(k\eta - \nu_0) = \frac{c_1}{c_1 + c_2} + f(\nu_0)(k\eta - \nu_0).
\]

Then
\[
p_k \equiv 1 - \frac{F(k\eta)}{F(\nu_0)} \equiv \left[ \frac{c_2}{c_1 + c_2} + \nu_0 f(\nu_0) \right] + \left[ -f(\nu_0) \eta \right] k \equiv a + bk
\]
\[
q_k \equiv F(k\eta) \equiv (1 - a) - bk,
\]

where \( a = \frac{c_2}{c_1 + c_2} + \nu_0 f(\nu_0) \) and \( b = -f(\nu_0) \eta \). Then (A.12) becomes
\[
\frac{1 - e^{-ic_2 t}}{e^{ic_1 t} - e^{-ic_2 t}} \tilde{\phi}_U(t) = \sum_{k=-\infty}^{\infty} e^{ikt} (a + bk) \pi_k = a \phi(t) + b \sum_{k=-\infty}^{\infty} ke^{ikt} \pi_k = a \phi_U(t) - bi \phi_U'(t),
\]

which leads to a differential equation system of \( \phi_U(t) \)
\[
\frac{\phi_U'(t)}{\phi_U(t)} = \frac{-i}{b} \left( \frac{a - 1 - e^{ic_2 t}}{1 - e^{i(c_1+c_2)t}} \right)
\]
\[
\phi_U(0) = 1.
\]
(A.13)

**First and Second Moments**

In order to derive the mean and variance of the stationary distribution, we need to find \( \phi'_y(t)|_{t=0} \) and \( \phi''_y(t)|_{t=0} \). Define \( A(t) = \frac{1 - e^{ic_2 t}}{1 - e^{i(c_1+c_2)t}} \). Then using (A.13) yields
\[
\phi'_U(0) = \phi_U(0) \left( \frac{-i}{b} \right) [a - A(0)] = \frac{i}{b} [a - A(0)].
\]

Notice \( A(0) \) can be computed using L’Hospital’s rule
\[
A(0) = \lim_{t \to 0} \frac{1 - e^{ic_2 t}}{1 - e^{i(c_1+c_2)t}} = \lim_{t \to 0} \frac{-ic_2 e^{ic_2 t}}{-i(c_1 + c_2) e^{i(c_1+c_2)t}} = \frac{c_2}{c_1 + c_2}.
\]
Appendix A. Statistical Analysis of Bilinear Cost Functions

Recall that $\nu_0 = F^{-1}\left(\frac{c_1}{c_1+c_2}\right)$, $a = \frac{c_2}{c_1+c_2} + \nu_0 f(\nu_0)$ and $b = -f(\nu_0)\eta$. Then

$$\phi'_U(0) = -\frac{i}{b}[a - A(0)] = \frac{i}{f(\nu_0)\eta}\left[\frac{c_2}{c_1} + \nu_0 f(\nu_0) - \frac{c_2}{c_1+c_2}\right] = \frac{i\nu_0}{\eta}.$$ 

Therefore, approximately,

$$\phi''_y(0) = \eta\phi'_U(0) = i\nu_0$$

$$E(\hat{y}) = \frac{\phi''_y(0)}{i} = \nu_0.$$ 

The first equation of (A.13) implies that $\phi'_U(t) = \phi_U(t)(-\frac{i}{b})[a - A(t)]$. Differentiating this equation yields

$$\phi''_U(t) = \phi'_U(t)(-\frac{i}{b})[a - A(t)] + \phi_U(t)(-\frac{i}{b})[-A'(t)].$$

Furthermore,

$$\phi''_U(0) = \phi'_U(0)(-\frac{i}{b})[a - A(0)] + \phi_U(0)(-\frac{i}{b})[-A'(0)] = -\frac{\nu_0^2}{\eta^2} + \frac{i}{b}A'(0).$$

By definition of $A(t)$

$$A'(t) = \frac{-i\nu_2 e^{i\nu_2 t} + i\nu_1 e^{i(\nu_2 + \nu_1)t} + i(c_1 + c_2)e^{i(c_1 + c_2)t} - i(c_1 + c_2)e^{i(2\nu_2 + \nu_1)t}}{(1 - e^{i(c_1 + c_2)t})^2}.$$ 

Applying L’Hospital’s rule twice, we get

$$A'(0) = \lim_{t \to 0} A'(t) = \frac{c_2^2 e^{i\nu_2 t} - (2c_2 + c_1)c_1 e^{i(2\nu_2 + \nu_1)t} - (c_1 + c_2)^2 e^{i(c_1 + c_2)t} + (c_1 + c_2)(2c_2 + c_1)e^{i(2\nu_2 + \nu_1)t}}{2(1 - e^{i(c_1 + c_2)t})(-i)(c_1 + c_2)e^{i(c_1 + c_2)t}}$$

$$= \lim_{t \to 0} \frac{c_2^2 e^{i\nu_2 t} + (2c_2 + c_1)c_1 e^{i(2\nu_2 + \nu_1)t} - (c_1 + c_2)^2 e^{i(c_1 + c_2)t}}{(-i)2(c_1 + c_2)[e^{i(c_1 + c_2)t} - e^{i(2\nu_2 + \nu_1)t}]}$$

$$= \lim_{t \to 0} \frac{ic_2^2 e^{i\nu_2 t} + i(2c_2 + c_1)^2 c_1 e^{i(2\nu_2 + \nu_1)t} - i(c_1 + c_2)^3 e^{i(c_1 + c_2)t}}{(-i)2(c_1 + c_2)[i(c_1 + c_2)e^{i(2\nu_2 + \nu_1)t} - i2(c_1 + c_2)e^{i(2\nu_2 + \nu_1)t}]}$$

$$= \frac{c_3^3 + (2c_1 + c_2)^2+c_2 - (c_1 + c_2)^3}{(-i)2(c_1 + c_2)[(c_1 + c_2) - 2(c_1 + c_2)]} = (-i)\frac{c_1 c_2(c_1 + c_2)}{2(c_1 + c_2)^2} = (-i)\frac{c_1 c_2}{2(c_1 + c_2)}.$$
Then

\[
\phi''_U(0) = -\frac{\nu_0^2}{\eta^2} + \frac{i}{b} (-i) \frac{c_1c_2}{2(c_1 + c_2)} \\
= -\frac{\nu_0^2}{\eta^2} - \frac{1}{\eta f(\nu_0)} \frac{c_1c_2}{2(c_1 + c_2)}.
\]

Therefore, approximately,

\[
E(\hat{y}^2) = -\phi''_y(0) = \nu_0^2 + \eta \frac{c_1c_2}{2(c_1 + c_2)} \frac{1}{f(\nu_0)},
\]

which implies that

\[
Var(\hat{y}) = \eta \frac{c_1c_2}{2(c_1 + c_2)} \frac{1}{f(\nu_0)}.
\]
A.5 Batch Update using the Quadratic Error Cost Function

In order to estimate $\mu(x)$, a scheme is proposed under a fixed design case. The design set consists of $x_1, \ldots, x_n$ which are ordered non-random numbers in $[a, b]$. Denote one observation across the design by the vector

$$y = [y(x_1) \ y(x_2) \ \ldots \ y(x_n)]^T,$$

Then

$$E(y) \equiv \mu = [\mu(x_1) \ \mu(x_2) \ \ldots \ \mu(x_n)]^T$$

$$Var(y) \equiv V = \sigma^2 \text{diag} [\nu(x_1) \ \nu(x_2) \ \ldots \ \nu(x_n)]$$

In addition, a set of knots $\{c_p, \text{ for } p = 1, \ldots, P\}$ is chosen in $[a, b]$. Usually $c's$ are chosen equally spread on $[a, b]$. Corresponding to each knot $c_p$, there is a basis function

$$\phi_p(x) = e^{-\left(\frac{x-c_p}{\sigma}\right)^2}, \text{ for } p = 1, \ldots, P.$$ 

The estimator for $\mu(x)$ is given by

$$\hat{y}(x) = \sum_{p=1}^{P} w_p \phi_p(x), \quad (A.14)$$

where $w = [w_1, \ldots, w_P]^T$ is obtained using the proposed scheme. Define the evaluation values of $P$ basis functions at $x_i$ by a row vector

$$\Phi_i = [\phi_1(x_i) \ \phi_2(x_i) \ \ldots \ \phi_p(x_i)];$$
and the $n \times p$ matrix

$$
\Phi = \begin{pmatrix}
\Phi_1 \\
\Phi_2 \\
\vdots \\
\Phi_n
\end{pmatrix}.
$$

Then the estimates of (A.14) at the design can be expressed by

$$\hat{y} = \Phi w.$$

Assume there are $K$ (replicate) observations over the design $\{y_k, k = 1, ..., K\}$. Start with the weight vector $w_1$. Usually $w_1 = 0$. Assume at the $k^{\text{th}}$ iteration, our estimate of $\mu(x_i)$ is $\hat{y}_k = \Phi w_k$. Define the estimation error associated with the $k^{\text{th}}$ update by

$$e_k = y_k - \hat{y}_k,$$

and the quadratic error cost function is

$$J(e_k) = \frac{1}{2} e_k^T e_k.$$

The traditional “backpropagation of error” adaptation rule for $w_{k+1}$ is

$$w_{k+1} = w_k - \eta \frac{\partial J}{\partial w_k},$$

(A.15)

where $\eta$ is a small positive constant. The partial derivative can be evaluated using the chain rule:

$$\frac{\partial J}{\partial w_k} = \frac{\partial J}{\partial e_k} \frac{\partial e_k}{\partial \hat{y}_k} \frac{\partial \hat{y}_k}{\partial w_k} = \Phi^T (-I) e_k = -\Phi^T e_k,$$

thus the adaptation rule in (A.15) becomes

$$w_{k+1} = w_k + \eta \Phi^T e_k.$$

(A.16)
**Lemma 2** Let $I_n$ be $n \times n$ identity matrix. For $k \geq 1$, the $k^{th}$ update of $\hat{y}$ has the expression

$$\hat{y}_k = \sum_{m=1}^{k-1} (I_n - \eta \Phi \Phi^T)^{k-1-m} (\eta \Phi \Phi^T) y_m + (I_n - \eta \Phi \Phi^T)^{k-1} \Phi w_1 \quad (A.17)$$

**Proof:**

For $k = 1$, $\hat{y}_1 = \Phi w_1$, and by (A.16)

$$e_1 = y_1 - \hat{y}_1, \quad w_{k+1} = w_k + \eta \Phi^T e_k = w_k + \eta \Phi^T y_1 - \eta \Phi^T \hat{y}_1.$$

For $k = 2$,

$$\hat{y}_2 = \Phi w_2 = \Phi w_1 + \eta \Phi \Phi^T y_1 - \eta \Phi \Phi^T \hat{y}_1 = \hat{y}_1 + \eta \Phi \Phi^T y_1 - \eta \Phi \Phi^T \hat{y}_1$$

$$= \eta \Phi \Phi^T y_1 + (I_n - \eta \Phi \Phi^T) \hat{y}_1 = \eta \Phi \Phi^T y_1 + (I_n - \eta \Phi \Phi^T) \Phi w_1.$$

It was shown that (A.17) holds for $k \leq 2$. Assume (A.17) is true for any $k \geq 1$, i.e.

$$\hat{y}_k = \sum_{m=1}^{k-1} (I_n - \eta \Phi \Phi^T)^{k-1-m} (\eta \Phi \Phi^T) y_m + (I_n - \eta \Phi \Phi^T)^{k-1} \Phi w_1$$

and it also holds for $k+1$. Firstly,

$$e_k = y_k - \hat{y}_k, \quad w_{k+1} = w_k + \eta \Phi^T e_k = w_k + \eta \Phi^T y_k - \eta \Phi^T \hat{y}_k.$$

Thus we have

$$\hat{y}_{k+1} = \Phi w_{k+1} = \Phi w_k + \eta \Phi \Phi^T y_k - \eta \Phi \Phi^T \hat{y}_k$$

$$= \hat{y}_k + \eta \Phi \Phi^T y_k - \eta \Phi \Phi^T \hat{y}_k = \eta \Phi \Phi^T y_k + (I_n - \eta \Phi \Phi^T) \hat{y}_k$$

$$= \eta \Phi \Phi^T y_k + \sum_{m=1}^{k-1} (I_n - \eta \Phi \Phi^T)^{k-1-m} (\eta \Phi \Phi^T) y_m + (I_n - \eta \Phi \Phi^T)^{k} \Phi w_1$$

$$= \sum_{m=1}^{k} (I_n - \eta \Phi \Phi^T)^{k-m} (\eta \Phi \Phi^T) y_m + (I_n - \eta \Phi \Phi^T)^{k} \Phi w_1$$

Thus (A.17) holds for the $k + 1^{st}$ update. The lemma is true.
Lemma 3 The limiting distribution of $\hat{y}_k$ is $N(\mu, \hat{V})$. When $V = \sigma^2 I_n$, we have $\hat{V} = \sigma^2 U A U^T$ with $A = \text{diag}\left[\frac{r_1}{2-r_1}, ..., \frac{r_n}{2-r_n}\right]$, where $r_1, ..., r_n$ are the eigenvalues of $\Phi\Phi^T$.

Proof:

Since $E(y_k) = \mu$,

$$E(\hat{y}_k) = \sum_{m=1}^{k-1} (I_n - \eta \Phi\Phi^T)^{k-1-m}(\eta \Phi\Phi^T)\mu + (I_n - \eta \Phi\Phi^T)^{k-1}w_1$$

$$= \sum_{m=1}^{k-1} (I_n - \eta \Phi\Phi^T)^{k-1-m} - \sum_{m=1}^{k-1} (I_n - \eta \Phi\Phi^T)^{k-1-m} \mu + (I_n - \eta \Phi\Phi^T)^{k-1}w_1$$

$$= \sum_{m'=0}^{k-2} (I_n - \eta \Phi\Phi^T)^{m'} \mu - \sum_{m'=1}^{k-1} (I_n - \eta \Phi\Phi^T)^{m'} \mu + (I_n - \eta \Phi\Phi^T)^{k-1}w_1$$

$$= [I_n - (I_n - \eta \Phi\Phi^T)^{k-1}] \mu + (I_n - \eta \Phi\Phi^T)^{k-1}w_1$$

Assume the matrix $\Phi\Phi^T$ is strictly positive definite. As long as the number of basis functions $p \geq n$ this assumption holds. Define the eigenvalue-eigenvector decomposition $\Phi\Phi^T = U R U^T$, where $UU^T = I$ and $R = \text{diag}[r_1, ..., r_n]$ with $0 < r_i < 1$. Choose a very small $\eta > 0$, such that $0 < \eta << 1$. Then the matrix $\eta \Phi\Phi^T = U(\eta I)U^T$ is also strictly positive definite and

$$I_n - \eta \Phi\Phi^T = I_n - U(\eta I)U^T = U(I_n - \eta R)U^T \equiv USU^T \quad \text{where} \quad S = I_n - \eta R.$$  

Note $(I_n - \eta \Phi\Phi^T)^k = (USU^T)^k = US^{k-1}U^T$. Then

$$E(\hat{y}_k) = [I_n - US^{k-1}U^T] \mu + US^{k-1}U\Phi w_1.$$  

Because $\lim_{k \to \infty} US^{k-1}U^T = 0_n$, where $0_n$ is the $n \times n$ matrix of zeros, we have

$$\lim_{k \to \infty} E(\hat{y}_k) = \mu.$$
Appendix A. Statistical Analysis of Bilinear Cost Functions

Notice the $y$'s are independent with each other and $\text{Var}(y) = V$. In addition, both $\eta \Phi \Phi^T$ and $I_n - \eta \Phi \Phi^T$ are positive definite.

$$
\text{Var}(\hat{y}_k) = \sum_{m=1}^{k-1} (I_n - \eta \Phi \Phi^T)^{k-1-m}(\eta \Phi \Phi^T) V [(I_n - \eta \Phi \Phi^T)^{k-1-m}(\eta \Phi \Phi^T)]^T
\quad = \sum_{m=1}^{k-1} (I_n - \eta \Phi \Phi^T)^{k-1-m}(\eta \Phi \Phi^T) V (\eta \Phi \Phi^T)(I_n - \eta \Phi \Phi^T)^{k-1-m}
\quad = \sum_{m=1}^{k-1} US^{k-1-m} U^T (U \eta R U^T) V (\eta R U^T) US^{k-1-m} U^T
\quad = U \left[ \sum_{m=1}^{k-1} S^{k-1-m} \eta R U^T V U R S^{k-1-m} \right] U^T
$$

Consider the special case: $\text{Var}(y) = \sigma^2 I_n$.

$$
\text{Var}(\hat{y}_k) = \sigma^2 U \left[ \sum_{m=1}^{k-1} S^{k-1-m} \eta R U^T U R S^{k-1-m} \right] U^T = \sigma^2 U \left[ \sum_{m=1}^{k-1} S^{k-1-m} (\eta R)^2 S^{k-1-m} \right] U^T
\quad = \sigma^2 U \left[ \sum_{m=1}^{k-1} (I - \eta R)^2 (k-1-m)(\eta R)^2 \right] U^T = \sigma^2 U A_k U^T
$$

where $A_k = \text{diag}[a_{k1}, ..., a_{kn}]$, with

$$
a_{ki} = \sum_{m=1}^{k-1} (1 - \eta r_i)^2 (k-1-m)(\eta r_i)^2 = (1 - (1 - \eta r_i)^2 (k-1)) \frac{\alpha_i}{2 - \alpha_i}, \quad \text{with} \quad \alpha_i = \eta r_i.
$$

Since $\lim_{k \to \infty} a_{ki} = \frac{\alpha_i}{2 - \alpha_i}$, we have

$$
\lim_{k \to \infty} \text{Var}(\hat{y}_k) = \sigma^2 U A U^T, \quad \text{with} \quad A = \text{diag}\left[\frac{\alpha_1}{2 - \alpha_1}, ..., \frac{\alpha_n}{2 - \alpha_n}\right].
$$
Appendix B

CIN Learning: Additional Figures
Figure B.1: CIN learning - E1/U1 for 4 krpm

Figure B.2: CIN learning - E2/U2 for 4 krpm
Figure B.3: CIN learning - E3/U3 for 4 krpm

Figure B.4: CIN learning - E4/U4 for 4 krpm
Figure B.5: CIN learning - E1/U1 for 2 krpm

Figure B.6: CIN learning - E2/U2 for 2 krpm
Figure B.7: CIN learning - E3/U3 for 2 krpm

Figure B.8: CIN learning - E4/U4 for 2 krpm
Appendix B. CIN Learning: Additional Figures

Figure B.9: CIN learning - E1/U1 for 6 krpm

Figure B.10: CIN learning - E2/U2 for 6 krpm
Figure B.11: CIN learning - E3/U3 for 6 krpm

Figure B.12: CIN learning - E4/U4 for 6 krpm
Figure B.13: CIN learning - E1/U1 for 2 krpm w/ mass

Figure B.14: CIN learning - E2/U2 for 2 krpm w/ mass
Figure B.15: CIN learning - E3/U3 for 2 krpm w/ mass

Figure B.16: CIN learning - E4/U4 for 2 krpm w/ mass