Abstract

Jayanth, Vignesh. **Identification and Control of Flexible Structures Supported on Active Magnetic Bearings** (under the guidance of Dr. Gregory Buckner)

The focus of this thesis is the development and demonstration of adaptive control algorithms for Active Magnetic Bearings (AMBs) that account for uncertainties associated with structural vibration modes. A flexible rotor AMB test rig is designed and modeled using Finite Element Analysis (FEA). The flexible modes and equations of motion are obtained using modal analysis techniques. Typically, such analyses accurately predict the low frequency modes, but accuracy at higher frequencies is greatly reduced. Adaptive control algorithms that account for parametric uncertainties and unmodeled dynamics are developed to levitate and control the flexible rotor. Model parameters for the system are identified using recursive least squares, and used to update a self-tuning pole-placement controller. This self-tuning approach dramatically improves the performance and robustness of the magnetic levitation system, as demonstrated through simulations and experimental demonstrations. The flexible rotor AMB test rig is fabricated to experimentally validate the identification and control algorithms. Results clearly demonstrate the performance and limitations of the adaptive control design.

Identification and Control of Flexible Structures Supported on Active Magnetic Bearings

by

Vignesh Jayanth

A thesis submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Master of Science

Mechanical and Aerospace Engineering
Raleigh, North Carolina

September, 2001

APPROVED BY:

____________________________	____________________________
Dr. Larry Silverberg	Dr. Paul I. Ro
(Advisory Committee Member)	(Advisory Committee Member)

____

Dr. Gregory D. Buckner

(Chair of Graduate Advisory Committee)

Biography

Vignesh Jayanth was born in Bangalore, India where he grew up with his parents, Jayanth and Uma, and his younger sister, Aarti. He earned his Bachelor of Engineering in Mechanical Engineering at R.V.C.E, Bangalore in August 1998. He then worked for a year at Jawaharlal Nehru Center for Advanced Scientific Research where he felt the need to get a Masters Degree. In August 1999, he traveled the seas to pursue a Masters Degree in Mechanical Engineering at the North Carolina State University and graduated in December, 2001. Vignesh is currently working with Wyle Laboratories as a development Engineer.

Acknowledgments

I would like to thank my advisor, Dr. Gregory Buckner, for his support and enthusiasm through this research. I am very grateful to him for directing my research study and providing me with guidance whenever I needed it. From him I gained an excellent knowledge of control engineering, a subject that I now feel at least competent in. I am indebted to him for his many suggestions and corrections to this thesis.

I am also thankful to Dr. Paul Ro and Dr. Larry Silverberg, members of my graduate committee, for instilling in me the basics of control engineering and vibration analysis through the courses I had taken under them.

Finally, I wish to thank my family and Lord Ganesha for their unending support, motivation, and spiritual guidance through this research project.

Table of Contents

List of Tables

List of Figures

1.0 Introduction

Active Magnetic Bearings (AMBs) are a slowly emerging technology with tremendous potential for a variety of rotating and translating applications such as flywheel systems, compressors, and linear motors [1, 2, 3]. They have many advantages over conventional bearings, particularly for high-speed applications. AMBs provide non-contacting support of the rotor, eliminating concerns with friction, wear, and lubrication - all commonly associated with conventional bearings. The ultimate speed of AMB-supported rotors can be limited only by the material strength of the rotor [4]. Furthermore, AMBs generate active magnetic forces for damping out shaft vibrations and stabilizing the system.

Industrial applications of AMB's have been hindered by several factors, including challenges associated with control algorithms. AMBs exhibit highly nonlinear dynamics and are open-loop unstable systems. Flexible rotor systems can make the control problem even more challenging by introducing lightly damped vibration modes that must be traversed. Another inherent problem with high-speed rotors is gyroscopic effects, which make the control problem multivariable. Gyroscopic effects increase with rotor speed, causing the system dynamics to change significantly [4, 5].

Simple, fixed gain AMB controllers can be tuned to work well for a limited set of operating conditions [4]. As the operating conditions change (as in the case of high-speed flywheel systems), the system performance can degrade significantly. In order to provide adequate stability margins over a wide range of operating conditions, fixed gain controllers are frequently tuned for worst-case scenarios [4]. Tracking and fixed notch filters can be introduced to deal with synchronous disturbances (such as those caused by mass imbalances) and controller-induced instabilities [5].

Numerous robust control strategies have been proposed to address the control challenges of AMB's [6, 7]. These approaches focus on H_∞ techniques that guarantee theoretical stability for a rotordynamic model subject to modeling uncertainties and disturbances. These techniques take into account the parametric uncertainties associated with variations in material properties and operating conditions, making the controller robust. Most of the published literature, however, has been limited to rigid rotors or fixed operating conditions. Another robust control technique, sliding mode control, has been discussed in [8]. Very little published research has been demonstrated experimentally.

In this research, an adaptive multivariable controller is developed for an AMB-supported flexible rotor. This work relies on theory, simulation and experimental validations. The performance of this algorithm is demonstrated through extensive simulations and experimental demonstrations on a custom AMB test rig. The control strategy adapts to modeling uncertainties and changes in the working environment, and uses this knowledge to improve overall system performance. This adaptation algorithm is very attractive for flexible rotor systems as it eliminates the need for explicitly modeling large numbers of modes and it eliminates spillover effects [9].

To simplify the problem, the flexible structure has only one rotational degree of freedom. The rotor is pinned about its center of mass to allow a “seesaw” pitching motion. This simplification eliminates gyroscopic effects associated with spinning shafts, but retains many of the challenges associated with AMB control. Electromagnetic actuators (AMBs) are located at both ends of the rotor to provide control inputs. The use of electromagnetic actuators illustrates the advantages (mentioned previously) over more conventional actuation methods. Two reflectance-compensated optical sensors are used to measure displacements at each end of the rotor. The rotor and actuators were designed using Finite Element Analysis, and the test rig fabricated specifically for this research.

2. Design and Modeling of the Flexible Rotor

The primary goal of this research was to develop and demonstrate adaptive control algorithms for magnetically supported flexible rotors. For this reason, the design of a suitable rotor was a primary focus. This chapter summarizes the rotor design and modeling efforts.

2.1 Design Objectives

The design of a flexible rotor was guided by several key objectives. First, the rotor needed to exhibit several flexible vibration modes (in this case 4) within the controller bandwidth of 1,000 Hz. The second objective was to dramatically reduce the number of control inputs and outputs from the full-order AMB problem. In the full-order problem, a rotor has six degrees of freedom (3 translational and 3 rotational), five inputs (X and Y radial forces at each bearing, Z thrust forces at one bearing) and five outputs (X and Y radial displacements at each bearing, Z displacement at one bearing). Figure 2.1 shows a typical full-order rotor with AMBs and sensors. For this research, the rotor was required to have two inputs (vertical forces $f(1)$ and $f(2)$ at each bearing) and two outputs (vertical displacement measurements $y(1)$ and $y(2)$ at each sensor). The final design objective was to incorporate adjustable mass imbalances to test experimental controller robustness.

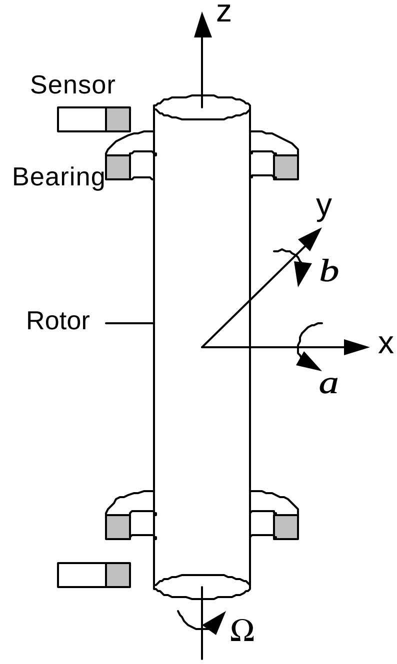


Figure 2.1: Full-order flexible rotor supported by active magnetic bearings

2.2 Finite Element Design and Analysis

The flexible rotor was designed and analyzed using Finite Element Analysis (FEA) such that there were four flexible modes and one rigid body mode within the control bandwidth of 1,000 Hz. ANSYS Version 5.5 was used for the design and analysis. The rotor was hinged to provide a single rigid body degree of freedom (a pitch rotation), and was designed to be symmetric about its axis of rotation. Slots were incorporated (not shown) so that imbalance masses could be added to the system. The FEA rotor geometry is shown in Figure 2.2.

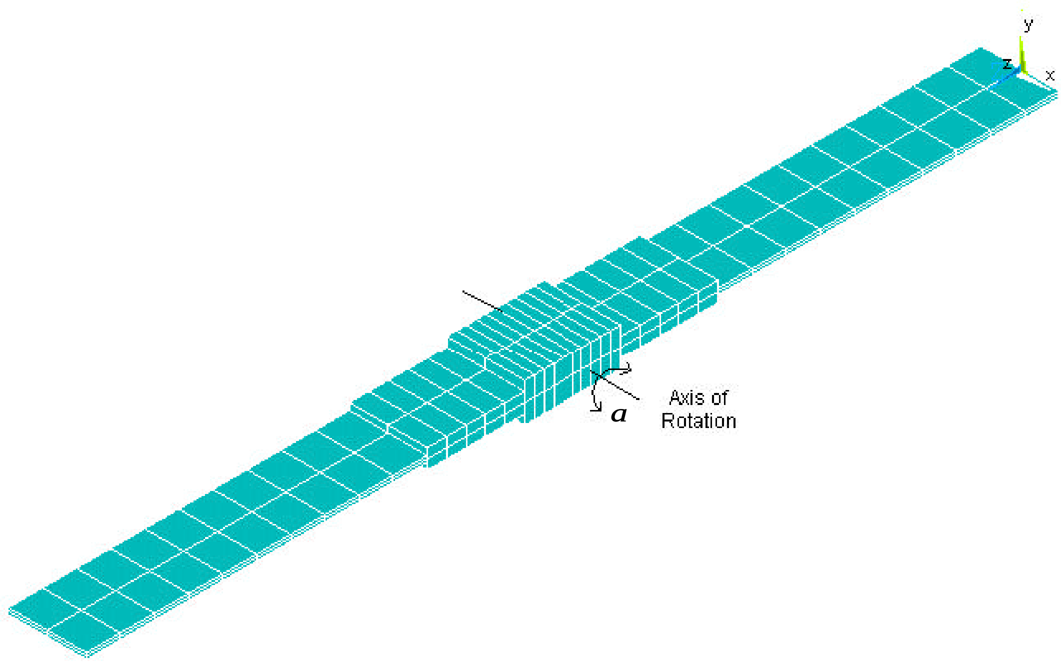


Figure 2.2: Geometry of the flexible rotor

The FEA rotor model was developed using three-dimensional beam elements created from 81 nodes. Each element was constructed of steel, with a density of $7700\ \mathrm{kg/m^3}$ and a Young's modulus of $210\ \mathrm{GPa}$. The overall dimensions were $0.5\ \mathrm{m}$ by $0.04\ \mathrm{m}$ by $0.003\ \mathrm{m}$, and the total rotor mass was 0.9394 kg.

The natural frequencies of the rotor obtained from ANSYS analysis are compiled in Table 2.1:

Table 2.1: Flexible rotor natural frequencies obtained from ANSYS

Mode	Natural Frequency (Hz)
1 (Rigid)	0
2 (Flexible)	78.9
3 (Flexible)	171.4
4 (Flexible)	479.0
5 (Flexible)	622.7

The 0-Hz rigid body mode corresponds to unconstrained pitch rotation, as shown in Figure 2.3 (a). The first four flexible vibration modes are shown in Figure 2.3 (b)–(e). These bending modes occur at frequencies sufficiently separated from each other to ensure accurate identification using on-line parameter estimation algorithms. Because the eigenvalues of the undamped open-loop system are purely imaginary, the associated eigenvectors are purely real, as expected [5].

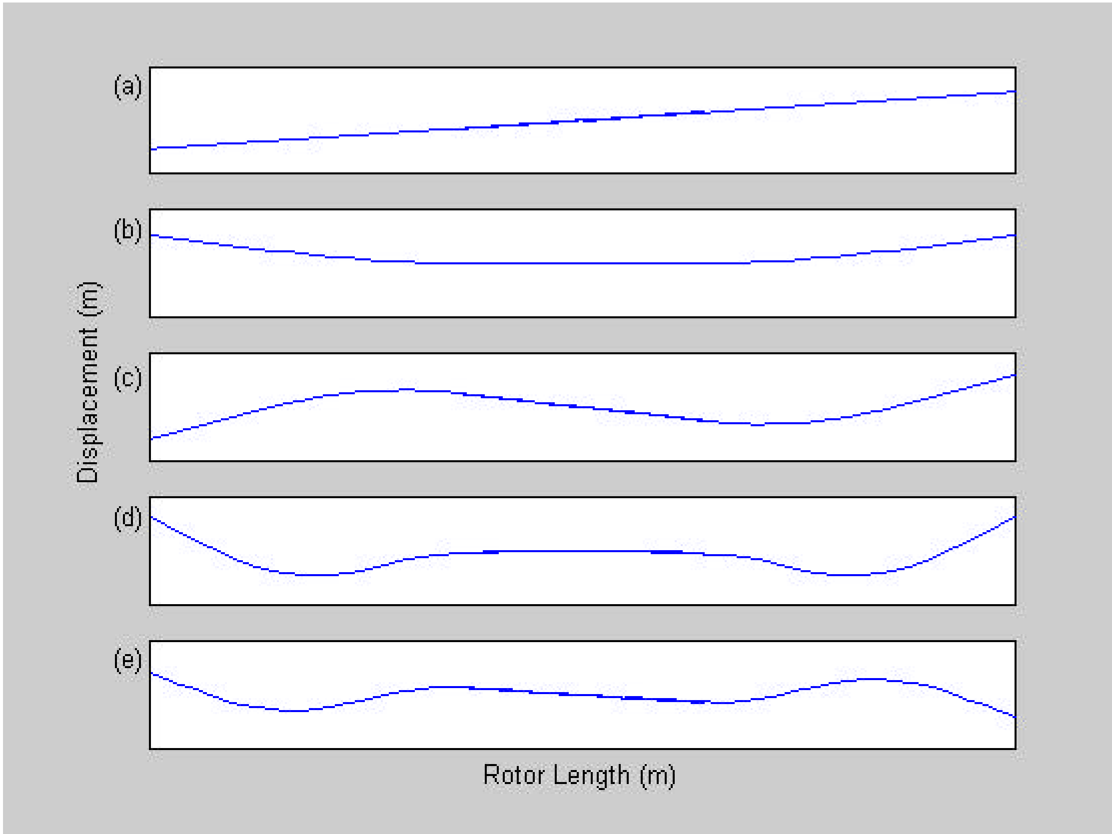


Figure 2.3 - Flexible rotor mode shapes
(a) rigid body bode (0 Hz)
(b) first bending mode (78.9 Hz)
(c) second bending mode (171.4 Hz)
(d) third bending mode (479.0 Hz)
(e) fourth bending mode (622.7 Hz)

2.3 Flexible Rotor Modeling

Mass and stiffness matrices *M* and *K* were obtained from the ANSYS rotor model and exported to Matlab to formulate the state equations. This model was mathematically described using a second order linear differential equation of the form:

$$M\ddot{z} + Kz = f \tag{2.1}$$

where z is the state vector of displacements and rotations defined by:

$$z = \begin{bmatrix} y & \acute{a} \end{bmatrix}^T \tag{2.2}$$

Here y and $\acute{a}$ are vectors of vertical displacements (in the Y direction) and rotations (in the XY plane) of each model element, respectively. Because the FEA model was designed using 81 nodes, and each node has two degrees of freedom, the state vector (2.2) has a total of 162 elements.

The plant has two inputs (vertical control forces f at each bearing) and two outputs (vertical displacement measurements y at each sensor).

The complete state space representation becomes:

$$\begin{aligned} \dot{z}_1 &= Az_1 + Bu \\ y &= Cz_1 + Du \end{aligned} \tag{2.3}$$

where:

$$z_1 = \begin{bmatrix} z & \dot{z} \end{bmatrix}^T \tag{2.4}$$

and the state matrices are:

$$A = \begin{bmatrix} 0 & I \\ -M^{-1}K & 0 \end{bmatrix}_{324 \times 324} \tag{2.5}$$

$$B = \begin{bmatrix} 0 \\ M^{-1} f \end{bmatrix} \tag{2.6}$$

C is a (2×324) null output matrix with ones at the (1,50) and (2,274) elements, and D is a null (2×2) matrix. To verify the accuracy of the state equation derivations, sorted eigenvalues of A were computed and compared to the lowest natural frequencies obtained in the ANSYS analysis (Table 2.1). These eigenvalues compared favorably, as shown in Table 2.2.

Table 2.2: Flexible rotor natural frequencies obtained from MATLAB

Mode	MATLAB Natural Frequency (Hz)	ANSYS Natural Frequency (Hz)
1 (rigid)	0	0
2 (flexible)	78.7	78.7
3 (flexible)	171.1	171.1
4 (flexible)	479.0	479.0
5 (flexible)	622.6	622.6

2.4 Modal Coordinate Transformation

For a vibrating mechanical system with n degrees of freedom, the governing equations of motion are a set of n coupled ordinary differential equations of order two. The analysis and solution of these equations can be simplified by using the modal analysis method. In this technique, the mass displacements are expressed as linear combinations of the system modes. This linear combination uncouples the equations of motion such that n uncoupled differential equations are obtained. The solution of these equations is equivalent to the solution of n single degree of freedom systems [10].

The original state vector (2.4) was transformed to modal coordinates using a modal transformation matrix Y, which was constructed from the sorted real eigenvectors of A [10]. This coordinate transformation results in a modal state vector q:

$$z = Yq \tag{2.7}$$

Equation (2.1) becomes:

$$MY\ddot{q} + KYq = f \tag{2.8}$$

Premultiplying (2.8) by Y^T yields:

$$Y^T MY\ddot{q} + Y^T KYq = Y^T f \tag{2.9}$$

Noting that $Y^T MY = I$, the modal state equations are:

$$\ddot{q} + \boldsymbol{W}q = Q \tag{2.10}$$

where $\boldsymbol{W}$ is a diagonal matrix of squared eigenvalues:

$$\boldsymbol{W} = Y^T KY = \begin{bmatrix} \boldsymbol{w}_{n1}^2 & 0 & & 0 \\ 0 & \boldsymbol{w}_{n2}^2 & & 0 \\ & & \ddots & \\ 0 & 0 & & \boldsymbol{w}_{nn}^2 \end{bmatrix} \tag{2.11}$$

and Q represents the modal force input:

$$Q = Y^T f \tag{2.12}$$

The modal state equations now become:

$$\begin{aligned} \dot{q}_1 &= A_m q_1 + B_m f \\ y &= C_m q_1 + D_m f \end{aligned} \tag{2.13}$$

where:

$$q_1 = \begin{bmatrix} q & \dot{q} \end{bmatrix}^T \tag{2.14}$$

and the state matrices become:

$$A_m = \begin{bmatrix} 0 & I \\ \Omega & 0 \end{bmatrix}_{324\times 324} \tag{2.15}$$

$$B_m = \begin{bmatrix} 0 \\ Y^T \end{bmatrix}_{324\times 2} \tag{2.16}$$

The output modal matrix is:

$$C_m = Y^T C \tag{2.17}$$

$$D_m = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \tag{2.18}$$

3. Design and Modeling of Active Magnetic Bearings

Levitation and control of the AMB test rig is accomplished using electromagnetic actuators at each end of the rotor. The electromechanical design of these actuators was based on magnetic circuit analysis, and was verified using FEA. Each actuator consists of two coils, one above and one below the rotor, to apply bi-directional reluctance control forces. These coils are wired differentially to reduce the effects of nonlinearities. This section outlines the electromechanical design of these actuators.

3.1 Design Objectives

The actuators are required to exert high-bandwidth, bi-directional, non-contacting control forces to each end of the ferromagnetic rotor. Regulation of a perfectly balanced rotor in the absence of disturbances requires relatively small actuation forces. Tracking requirements and disturbance rejection require larger actuator forces. For this project, the peak force requirement of each actuator was selected to equal the rotor weight:

$$f_{peak} = m_r g = 9.22N \tag{3.1}$$

where m_r represents the rotor's mass (0.94 kg) and g represents the gravitational acceleration (9.81 m/s^2).

The bandwidth of the actuator depends on both the electrical and magnetic characteristics of the actuator. The electrical bandwidth $\boldsymbol{w}_{be}$ can be approximated as the corner frequency of the first-order electrical dynamics. The inductance L (H) and resistance R (Ω) of the actuator's copper coils determine this bandwidth:

$$w_{be} = \frac{R}{2pL} \tag{3.2}$$

The magnetic bandwidth can be limited by eddy currents in the actuator core material [11], which are generated when the magnetic flux density reverses directions within the core, and are proportional to the frequency of these reversals. The actuators are designed using M-19 laminated electrical steel to reduce these eddy current losses, hence magnetic bandwidth is not considered critical.

Actuator heating is a critical design criterion. The peak actuator currents and number of turns must be selected appropriately so that the required magnetomotive force Ni can be generated without overheating the coils. Although increasing the number of turns can reduce peak currents, space limitations on the wire diameter (and hence the current ratings) and inductance concerns make this a design tradeoff.

3.2 Actuator Design

The electromechanical actuators employ magnetic fields to create reluctance forces in the ferromagnetic rotor. These actuators require ferromagnetic cores for guiding and concentrating the magnetic flux. The magnetic permeability of core materials (laminated electrical steel) is typically 3 to 4 orders of magnitude higher than air, hence most of the magnetic flux is confined to definite paths determined by the core geometry. Consequently, air gaps constitute the only significant reluctance to the magnetic flux in such devices.

Because reluctance forces are attractive only, two actuators are required for each bearing to exert bi-directional control forces, as shown in Figure 3.1.

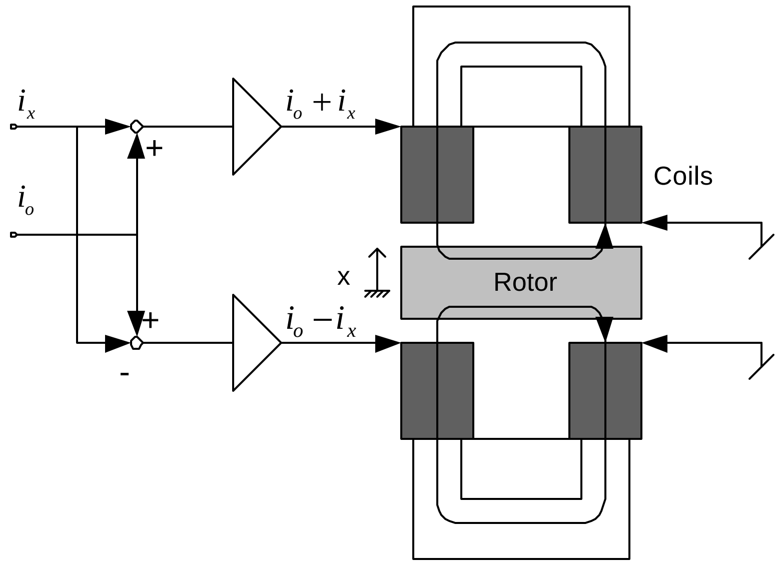
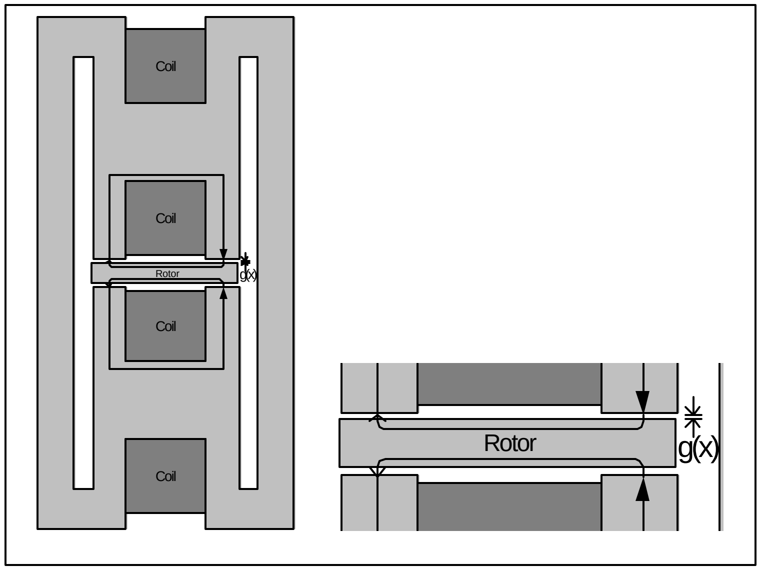


Figure 3.1: Bi-directional control forces exerted by the actuator

In the simplest design, two independent actuators are used at each end of the rotor to generate counteracting positive and negative forces. This scheme results in highly nonlinear force/current and force/displacement relationships, however [5]. If the actuators are wired in a *differential* mode, the bearing nonlinearities are significantly reduced [4]. In differential mode, one actuator is powered with the sum of a bias current and control current and the other with the difference of these currents (Figure 3.1).

The actuator design process resulted in the basic AMB geometry shown in Figure 3.2. Coils above and below the rotor (wired differentially) create the magnetic fluxes that produce reluctance control forces. The design specifications for this bearing (dimensions, materials, number of turns, etc.) required extensive analysis, as outlined in the following sections.



(a) **(b)**

Figure 3.2: (a) basic actuator geometry, (b) close-up of magnetic flux lines

3.3 Magnetic Circuit Analysis

Magnetic circuit analysis is a straightforward technique for designing and analyzing electromechanical devices. The assumptions made during the design process are [11]:

- Infinite core permeability (no flux losses in the ferromagnetic core)
- No fringing fluxes in the air gaps
- No leakage fluxes around the core
- No flux saturation in the core
- No eddy currents
- Lossless magnetic fields

The electromagnetic actuators were designed using M-19 laminated electrical steel, which has high permeability, low residual flux density, and reduces eddy currents. Each coil winding is comprised of 350 turns ($N=350$) of AWG 20 magnet wire (solid copper conducted coated with insulating varnish). Currents in these coils create magnetic flux that passes through the actuator core, through the variable air gap g (this gap depends on rotor position), through the ferromagnetic rotor, and returns similarly to close the path (Figure 3.2).

The reluctance of this magnetic circuit for a nominal air gap of 0.002m can be calculated:

$$R = \frac{2g}{\mu_o A} = \frac{2 \cdot 0.002m}{\left(4\pi 10^{-7}\ H/m\right) \cdot \left(10^{-4}\ m^2\right)} = \ 3.18 \cdot 10^{7}\ A \cdot t/Wb \tag{3.3}$$

where μ_o is the permeability of air and A represents the core's cross-sectional area. The inductance of each coil is:

$$L = \frac{N^2}{R} = \frac{N^2 \mu_o A}{2g} = \frac{\left(350t\right)^2}{3.18 \cdot 10^{7}\ A \cdot t/Wb} = 3.8mH \tag{3.4}$$

The magnetic flux density in the air gap $B(g)$ depends on the gap dimension:

$$B(g) = \frac{Ni\mu_o}{2g} \tag{3.5}$$

The resulting flux linkage is equal to the surface integral of the normal component of the magnetic flux density integrated over any surface spanned by the coil. In this case, the flux linkage is:

$$\lambda(g) = \frac{N^2 \mu_o A}{2g} i \tag{3.6}$$

Assuming that the magnetic field is lossless, the co-energy of the system is defined to be:

$$CoE_f = \int \lambda(g)di = -\frac{N^2 \mu_o A}{4g} i^2 \tag{3.7}$$

The electromagnetic force is derived from the derivative of co-energy with respect to the state-dependent air gap:

$$F = \frac{dCoE_f}{dg} = -\frac{N^2 \mu_o A}{4g^2} i^2 \tag{3.8}$$

The coils selected for this design have a peak current rating of 3.1 A. Hence, the peak force capability of this actuator is:

$$F = -\frac{N^2 \mu_o A}{4g^2} i^2 = -\frac{350^2 \cdot 4\pi 10^{-7} \cdot 10^{-4}}{4 \cdot 0.002^2} \cdot 3.1^2 = -9.2N \tag{3.9}$$

The electromagnetic force for each coil is observed to be negative, indicating that the actuator cannot generate repulsive forces. Note that this force relationship is highly nonlinear, depending on the square of coil current and the inverse square of air gap.

In the differential driving mode (Figure 3.1), the net actuator force F_{net} is simply the difference of actuator forces above and below the rotor. These forces depend on the sum ($i_o + i_x$) and the difference ($i_o - i_x$) of bias and control currents. The upper and lower air gaps depend on the rotor displacement according to $(g+x)$ and $(g-x)$. Equation (3.8) becomes:

$$F = F_{net} = F_{above} - F_{below} \tag{3. 10}$$

$$F = k[\frac{(i_o + i_x)^2}{(g-x)^2} - \frac{(i_o - i_x)^2}{(g+x)^2}] \tag{3. 11}$$

where:

$$k = \frac{1}{4} m_o N^2 A \qquad (3.\ 12)$$

To gain a better understanding of the force/current and force/displacement relationships, this force can be linearized about a nominal operating point g_o, i_o [2]:

$$\begin{aligned} F &= \frac{N^2 m_o A i_o}{g^2} i + \frac{N^2 m_o A i_o^2}{g^3} x \\ F &= K_i i + K_s x \end{aligned} \qquad (3.\ 13)$$

where K_i is the *force-current factor* and K_s is the *force-displacement factor*. For a nominal gap of 0.002 m and a bias current of 0.65 A, the nominal force-current and force-displacements for this actuator are:

$$\begin{aligned} K_i &= 1177 \;\; N/A \\ K_s &= -18000\, N/m \end{aligned} \qquad (3.\ 14)$$

The resistance of 350 turns of copper wire, based on the actuator dimensions, is $1.1\,\Omega$. The electrical bandwidth of this actuator can be calculated from equations (3.2) and (3.4):

$$w_{be} = \frac{R}{2pL} = \frac{1.1}{2p\left(3.8 \cdot 10^{-3}\right)} = 46 Hz \qquad (3.\ 15)$$

Obviously, this bandwidth is far below the controller bandwidth of 1000 Hz. This value represents a design tradeoff, however, and should not adversely affect control performance. The inductance could be significantly reduced (and the bandwidth increased) with fewer turns of larger diameter wire (a preliminary design utilized **100** turns of AWG 16 copper wire). However, it was extremely difficult to manually wind this larger wire. Since the actuator was designed to be operated in current control mode (meaning a high-bandwidth power amplifier would produce enough voltage to achieve the desired current), it was decided that bandwidth could be sacrificed for ease of fabrication, and AWG 20 gage wire was selected. The current requirements for control are below the maximum current ratings for the actuator coils (3.1 A). Hence, the actuator can be run for extended periods of time without overheating the coils.

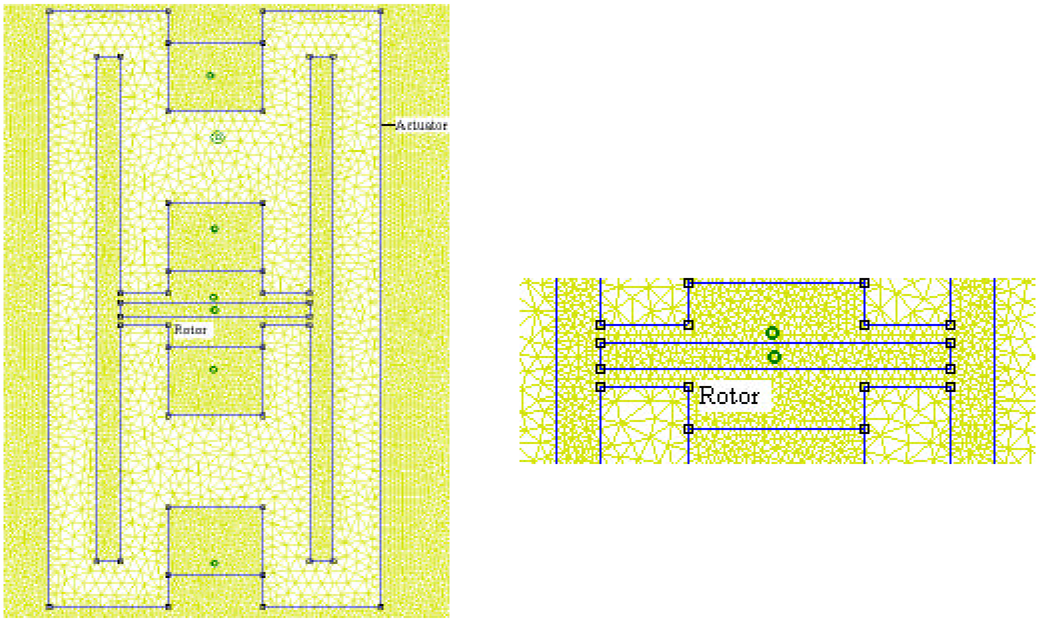
3.4 Finite Element Analysis

Magnetic circuit analysis provides quick, accurate, intuitive results for simple machines. However, it neglects fringing fluxes, leakage fluxes, and eddy currents and is difficult to use in the nonlinear operating regions. FEA is a powerful tool that overcomes many of these limitations and can provide accurate validations of the magnetic circuit design.

The actuator was designed using FEMM® 2-D FEA software. Coils were modeled as solid copper conductors with appropriate current densities and packing factors. The materials used to model the actuator and coils are summarized in Table 3.1. Figure 3.3 shows the nodes and elements generated for this model.

Table 3.1: Materials used in the FEMM actuator model

Model	Materials Used	Relative Permeability
Actuator Core	Laminated M19 Steel	4416
Rotor	1018 Steel	529
Coils	Copper	1



(a) **(b)**

Figure 3.3: (a) Nodes and elements generated in FEMM
(b) A close up view

For the FEMM analyses, the actuator was specified to have 1070.5 amp-turns, corresponding to 350 turns and a current of 3.1 A. The magnetic flux generated in the actuator is shown in Figure 3.4.

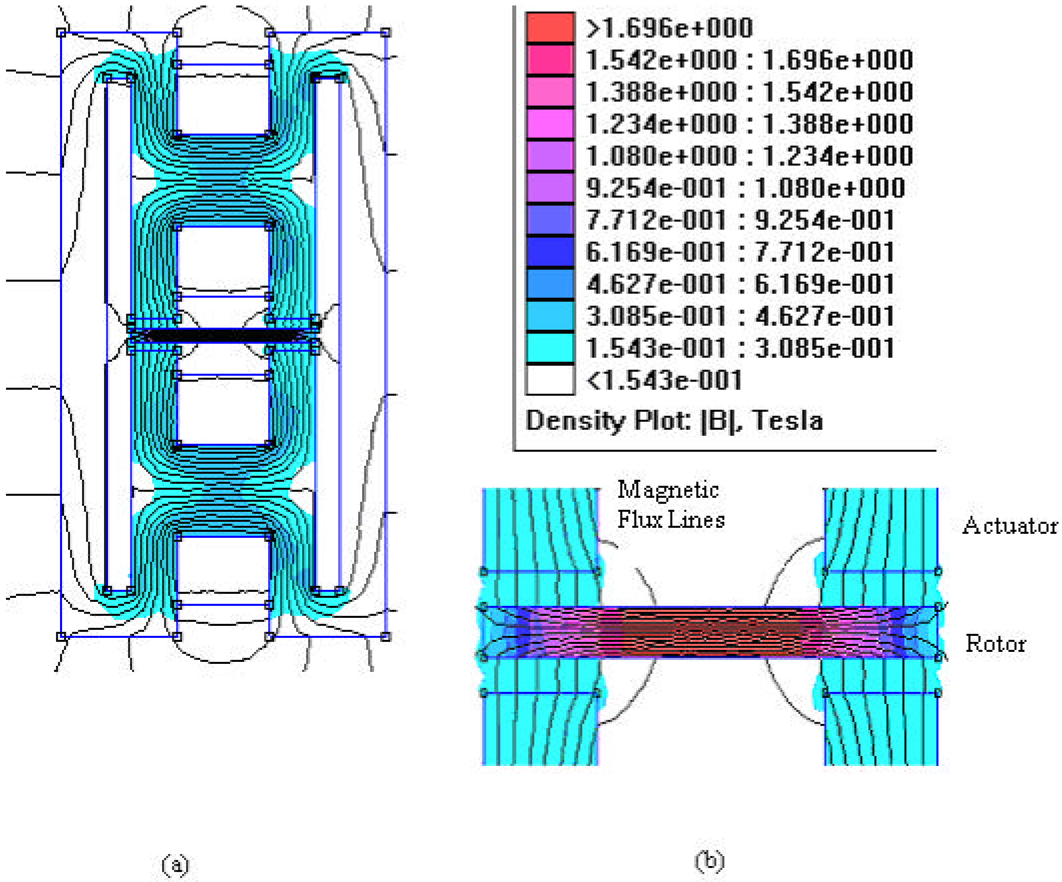


Figure 3.4: Magnetic flux lines through the actuator

Magnetic flux lines from the top half of the actuator flow through the iron rotor and back to the top half of the actuator. This provides the attractive force required to levitate and control the rotor. Similarly, magnetic flux lines generated in the lower half of the actuator flow through the iron rotor and return. A closer look at Figure 3.4 (b) shows these lines of magnetic flux.

3.5 Actuator Design Summary

The actuator was designed to achieve peak forces of -9.22 N. As shown in Table 3.2, the peak forces achieved, evaluated using magnetic circuit analysis and FEA, were very close this target.

Table 3.2: Comparison of calculated actuator peak forces

Source	Peak Force (N)
Design Requirement	-9.22
Magnetic Circuit Analysis	-9.20
Finite Element Analysis	-9.01

The bandwidth of the actuator (46 Hz) was far below the controller bandwidth of 1000 Hz. However, as explained in the Section 3.3, this bandwidth was a design tradeoff that should not significantly degrade controller performance. The current requirements for control are below the maximum current ratings for the actuator coils (3.1 A). Hence, the actuator can be run for extended periods of time without overheating the coils. The actuator was baked in varnish to prevent any short circuits due to kinks in the actuator windings.

4. Adaptive Control Development

Self-tuning control is an adaptive control strategy that is well suited for magnetic levitation of flexible rotors with uncertain dynamics. The rotor described in Chapter 2 has several lightly damped flexible vibration modes within the controller bandwidth. Although a rotordynamic model was derived using FEA (Equation 2.1), modeling uncertainties were expected, especially for higher-order modes [5]. Parametric uncertainties due to variations in material properties and unmodeled plant dynamics can reduce the performance and stability of non-adapting control algorithms. In the more general case of a spinning rotor, the natural frequencies of these uncertain modes would vary with rotor speed [12]. Hence, LTI models covering large speed ranges cannot accurately model the true rotordynamics, and simple control strategies may not provide adequate stability and performance for the AMB system.

An adaptive controller can accommodate these variations and uncertainties in dynamic behavior by adjusting the controller gains. The development of a self-tuning controller for the magnetic levitation system is the focus of this chapter.

4.1 Self-Tuning Control

In adaptive control systems, the controller gains successfully adapt to plant changes and parameter variations such that the overall system performance improves over time. One adaptive approach uses parameter estimation to update the control law online. Such controllers are called *self-tuning*, since they continually tune the controller for improved performance. The *self-tuning regulator* (STR) separates the estimation of unknown parameters from the design of the controller, as shown in Figure 4.1.

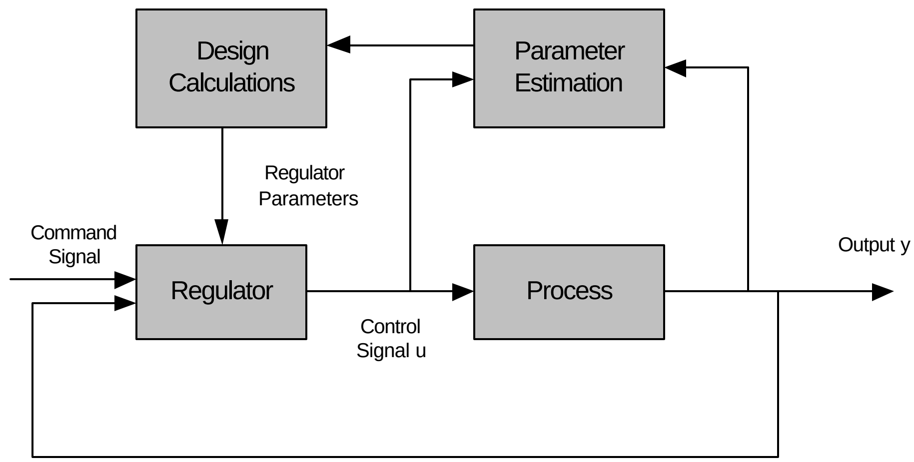


Figure 4.1: Block diagram of a self-tuning regulator

The unknown model parameters are estimated on-line, using recursive least squares or other estimation techniques. The estimated model parameters are then used to update the controller gains at each time step (the "Design Calculations" block)... the so-called *underlying design problem* [13]. The self-tuning regulator is an example of an *indirect adaptive algorithm*. The regulator parameters are not updated directly, but rather indirectly via the estimation of the process model. Kalman first described this type of adaptive controller based on least squares estimation and deadbeat control, in 1958 [14].

4.2 The Two-Input, Two-Output Process Model

For system identification, the process model needs to be represented in discrete time because data acquisition is inherently discrete. The continuous time, multivariable plant dynamics (2.10) can be discretized into an ARMAX (AutoRegressive, Moving Average with eXogenous inputs) model structure [13]:

$$A(q)\,y(t) = B(q)u(t) + e(t) \tag{4.1}$$

Here $u(t)$ is the sampled control input and T is the sampling interval:

$$\begin{aligned} u(t) &= u(kT) \\ k &= 0 \cdots \infty \end{aligned} \tag{4.2}$$

$y(t)$ is the sampled plant output, and $e(t)$ is a discrete sequence of independent, equally distributed Gaussian variables. $A(q)$ and $B(q)$ are polynomials in the forward shift operator q, the standard MATLAB format for discrete time systems. It is usually convenient to express the process model using the delay operator q^{-1}:

$$A(q) = q^n A(q^{-1}) \tag{4.3}$$

where n = deg A.

The model (4.1) can now be expressed as:

$$A(q^{-1})y(t) = B(q^{-1})u(t - d_0) + e(t) \tag{4.4}$$

where:

$$d_0 = deg\ A - deg\ B \tag{4.5}$$

For a two-input, two-output case:

$$A(q^{-1}) = \begin{bmatrix} A_{11}(q^{-1}) & A_{12}(q^{-1}) \\ A_{21}(q^{-1}) & A_{22}(q^{-1}) \end{bmatrix} \tag{4.6}$$

$$B(q^{-1}) = \begin{bmatrix} B_{11}(q^{-1}) & B_{12}(q^{-1}) \\ B_{21}(q^{-1}) & B_{22}(q^{-1}) \end{bmatrix} \tag{4.7}$$

A_{11}, A_{12}, A_{21}, and A_{22} are the open-loop polynomials of the process model and their roots are the open-loop eigenvalues. For the two-input, two-output process model, $A_{11} = A_{12} = A_{21} = A_{22} = A_0$, and A_0 and B_{ij} have the following structure:

$$A_0(q^{-1}) = 1 + a_1 q^{-1} + ... + a_{(na)} q^{-na} \tag{4.8}$$

$$B_{ij}(q^{-1}) = b_{ij1} q^{-nk} + ... + b_{ij(nb)} q^{-nk-nb+1} \tag{4.9}$$

where $na = deg\ A_0$ and $nb = deg\ B_{ij}$. In this case, however $na = nb$ and hence $d_o = 0$. The component of the smallest system pure time delay that is an integer multiple of the sampling time is modeled by the term q^{-nk}. The system transfer function matrix is:

$$\mathbf{H}\left(q^{-1}\right) = \frac{y(t)}{u(t)} = \begin{bmatrix} \frac{B_{11}\left(q^{-1}\right)}{A_0\left(q^{-1}\right)} & \frac{B_{12}\left(q^{-1}\right)}{A_0\left(q^{-1}\right)} \\ \frac{B_{21}\left(q^{-1}\right)}{A_0\left(q^{-1}\right)} & \frac{B_{22}\left(q^{-1}\right)}{A_0\left(q^{-1}\right)} \end{bmatrix} \tag{4.10}$$

The basic relationship between the inputs and outputs is given by the following two *linear difference equations*:

$$\begin{aligned} y_1(t) = &-a_1 y_1(t-1) - ...... - a_n y_1(t-n) + b_{11(1)} u_1(t-1) + ... + b_{11(m)} u_1(t-m) \\ &+ b_{12(1)} u_2(t-1) + ... + b_{12(m)} u_2(t-m) \end{aligned} \tag{4.11}$$

$$\begin{aligned} y_2(t) = &-a_1 y_2(t-1) - ...... - a_n y_2(t-n) + b_{21(1)} u_1(t-1) + ... + b_{21(m)} u_1(t-m) \\ &+ b_{22(1)} u_2(t-1) + ... + b_{22(m)} u_2(t-m) \end{aligned} \tag{4.12}$$

where $y_i(t)$ represents the displacement at the i^{th} sensor and $u_j(t)$ represents the control input at the j^{th} actuator.

4.3 System Identification

In the "Parameter Estimation" block of Figure 4.1, coefficients of a pre-defined model structure are adjusted to minimize prediction errors between the model and the measured plant outputs. This system identification process is depicted in Figure 4.2.

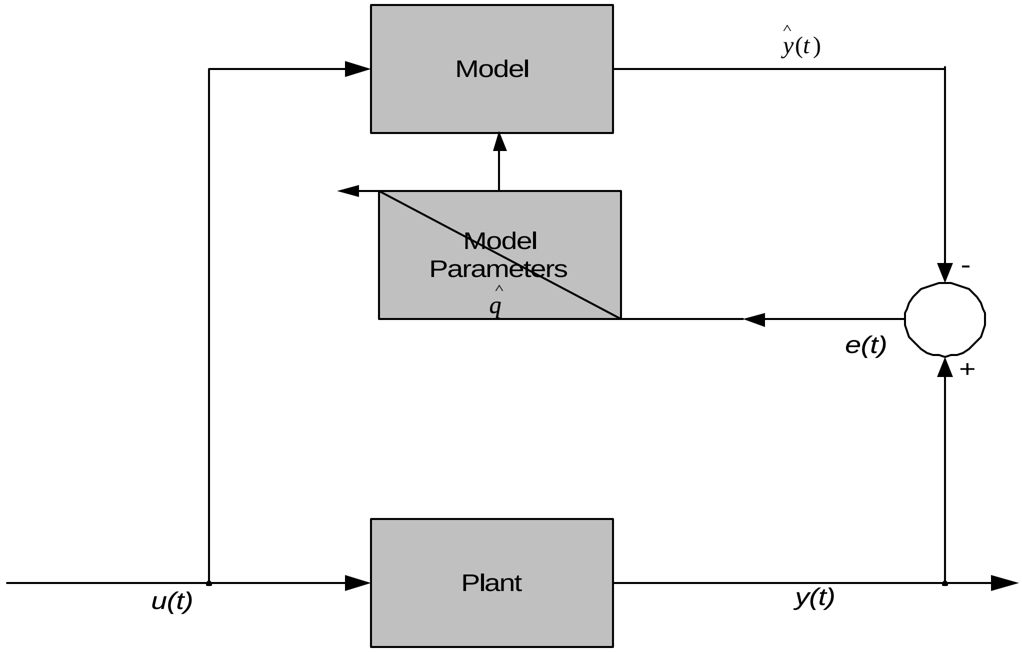


Figure 4.2: On-line parameter estimation

Several system identification strategies can be used for on-line parameter estimation. Some of the traditional approaches are recursive least squares, recursive instrumental variables, and recursive maximum likelihood [15]. A recursive least squares method was selected for this application for the following reasons [16]:

- The RLS algorithm in a deterministic case gives convergence in a finite number of steps, learning quickly under changing conditions
- This algorithm requires less computational effort, which is significant when estimating the parameters online

Equations (4.11) and (4.12) are called *regression models* and can be described by *regression vectors* $\boldsymbol{f}_1(t)$ and $\boldsymbol{f}_2(t)$ and *parameter vectors* $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$:

$$\begin{aligned}\boldsymbol{f}_1(t) = [&-y_1(t-1) \quad . \quad . \quad -y_1(t-n) \\ &u_1(t-1) \quad . \quad . \quad u_1(t-m) \quad u_2(t-1) \quad . \quad . \quad u_2(t-m)]^T\end{aligned} \tag{4.13}$$

$$\boldsymbol{q}_1 = [a_1 \quad . \quad . \quad a_n \quad b_{11(1)} \quad . \quad . \quad b_{11(m)} \quad b_{12(1)} \quad . \quad . \quad b_{12(m)}]^T \tag{4.14}$$

Equation (4. 11) can now be expressed:

$$y_1(t) = \boldsymbol{f}_1^T(t) \cdot \boldsymbol{q}_1 \tag{4.15}$$

Similarly:

$$\begin{aligned}\boldsymbol{f}_2(t) = [&-y_2(t-1) \quad . \quad . \quad -y_2(t-n) \\ &u_1(t-1) \quad . \quad . \quad u_1(t-m) \quad u_2(t-1) \quad . \quad . \quad u_2(t-m)]^T\end{aligned} \tag{4.16}$$

$$\boldsymbol{q}_2 = [a_1 \quad . \quad . \quad a_n \quad b_{21(1)} \quad . \quad . \quad b_{21(m)} \quad b_{22(1)} \quad . \quad . \quad b_{22(m)}]^T \tag{4.17}$$

Equation (4. 12) can now be expressed:

$$y_2(t) = \boldsymbol{f}_2^T(t) \cdot \boldsymbol{q}_2 \tag{4.18}$$

Vectors $\hat{\boldsymbol{q}}_1$ and $\hat{\boldsymbol{q}}_2$ are estimates of the parameter vectors given in equations (4.14) and (4.17), respectively. The *estimation errors* or *residual errors* are given by:

$$\boldsymbol{e}_1(t) = y_1(t) - \boldsymbol{f}_1^T(t)\hat{\boldsymbol{q}}_1 \tag{4.19}$$

$$\boldsymbol{e}_2(t) = y_2(t) - \boldsymbol{f}_2^T(t)\hat{\boldsymbol{q}}_2 \tag{4.20}$$

The parametric fit is based on experimental data, where the outputs y_1 and y_2 and the regression vectors $\boldsymbol{f}_1$ and $\boldsymbol{f}_2$ are known. The parameter estimates are then adjusted such that a quadratic error cost function is minimized with respect to $\hat{\boldsymbol{q}}_1$ and $\hat{\boldsymbol{q}}_2$:

$$J = \frac{1}{2}\sum_{t=1}^{n}\boldsymbol{e}^2(t) \tag{4.21}$$

The recursive least squares algorithm is summarized below [15,17]. Given initial parameter estimates $\hat{\boldsymbol{q}}_1(t_0)$ and $\hat{\boldsymbol{q}}_2(t_0)$:

$$\hat{\boldsymbol{q}}_1(t) = \hat{\boldsymbol{q}}_1(t-1) + mP_1(t)^{-1}\boldsymbol{f}_1(t)\boldsymbol{e}_1(t) \tag{4.22}$$

$$P_1(t) = P_1(t-1) + m\left[\boldsymbol{f}_1(t)\boldsymbol{f}_1(t)^T - P_1(t-1)\right] \tag{4.23}$$

Similarly,

$$\hat{\boldsymbol{q}}_2(t) = \hat{\boldsymbol{q}}_2(t-1) + mP_2(t)^{-1}\boldsymbol{f}_2(t)\boldsymbol{e}_2(t) \tag{4.24}$$

$$P_2(t) = P_2(t-1) + m\left[\boldsymbol{f}_2(t)\boldsymbol{f}_2(t)^T - P_2(t-1)\right] \tag{4.25}$$

P_1 and P_2 are initialized as identity matrices and are updated at each time step as shown in equations (4.23) and (4.25). m is a constant that determines the rate of parameter convergence. The parameters estimated online converge to their true values within a finite number of time steps. These identified parameters can then be used to calculate the control parameters.

4.4 Pole Placement Control Design

The controller design method is selected based on the specifications of the closed-loop system. Examples of linear design methods for similar STR problems include pole-placement controllers [18,19], Linear Quadratic Regulators (LQRs) [13], and minimum variance regulators [16]. More sophisticated linear and nonlinear control approaches have also been used.

The pole-placement design method places the closed-loop system poles at desirable locations. This is a very attractive method, since the closed-loop system performance is primarily determined by the location of these poles. Pole placement is also well suited to implementation on discrete systems.

Consider the system in Figure 4.3, with transfer functions defined in (4.10) and (4.26):

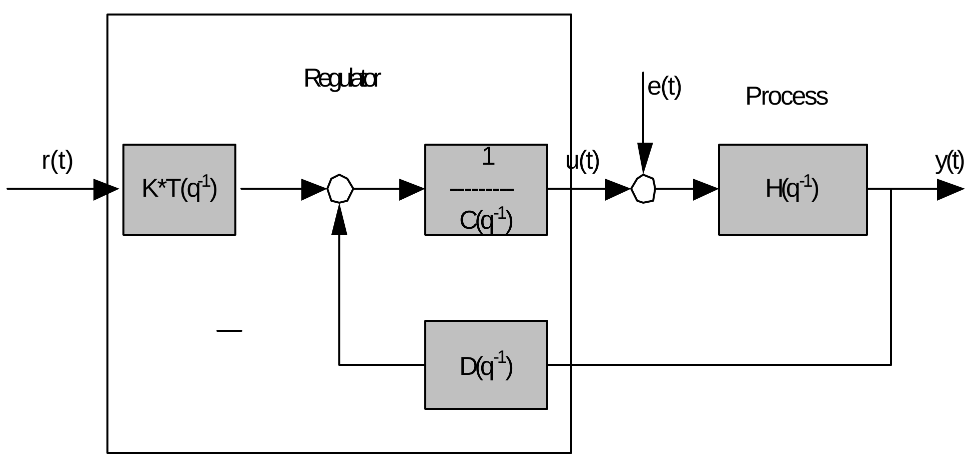


Figure 4.3: Pole placement controller structure

Neglecting disturbances at the plant input, the closed loop transfer function from the reference input r to the output y is:

$$\frac{Y(q^{-1})_{2\times 1}}{R(q^{-1})_{2\times 1}} = \frac{KT(q^{-1})B(q^{-1})}{A(q^{-1})C(q^{-1}) + B(q^{-1})D(q^{-1})} \tag{4.26}$$

where:

$$T(q^{-1}) = \begin{bmatrix} T_{11}(q^{-1}) & T_{12}(q^{-1}) \\ T_{21}(q^{-1}) & T_{22}(q^{-1}) \end{bmatrix} \tag{4.27}$$

$$C(q^{-1}) = \begin{bmatrix} C_{11}(q^{-1}) & C_{12}(q^{-1}) \\ C_{21}(q^{-1}) & C_{22}(q^{-1}) \end{bmatrix} \tag{4. 28}$$

$$D(q^{-1}) = \begin{bmatrix} D_{11}(q^{-1}) & D_{12}(q^{-1}) \\ D_{21}(q^{-1}) & D_{22}(q^{-1}) \end{bmatrix} \tag{4. 29}$$

The three control parameters $C(q^{-1})$, $D(q^{-1})$, and $T(q^{-1})$ must be designed to obtain a suitable regulator. The orders of polynomials $C(q^{-1})$ and $D(q^{-1})$ are nc and nd respectively, and:

$$\begin{aligned} C_{ij}(q^{-1}) &= 1 + c_{ij(1)}q^{-1} + \ldots\ldots + c_{ij(nc)}q^{-nc} \\ nc &= (nk + nb - 1) - 1 \\ i &= 1,2 \quad j = 1,2 \end{aligned} \tag{4.30}$$

$$\begin{aligned} D_{ij}(q^{-1}) &= d_{ij(0)} + d_{ij(1)}q^{-nk} + \ldots\ldots + d_{ij(nd)}q^{-nd} \\ nd &= na - 1 \\ i &= 1,2 \quad j = 1,2 \end{aligned} \tag{4.31}$$

$$T_{ij}(q^{-1}) = 1 + t_{ij(1)}q^{-1} + ... + t_{ij(nt)}q^{-nt} \qquad i = 1,2 \quad j = 1,2 \tag{4.32}$$

To simplify the problem, the desired closed-loop transfer function matrix can also be expressed as**:**

$$\boldsymbol{H}_d\left(q^{-1}\right) = \begin{bmatrix} \dfrac{KT_{11}\left(q^{-1}\right)B_{11}\left(q^{-1}\right)}{P\left(q^{-1}\right)} & \dfrac{KT_{12}\left(q^{-1}\right)B_{12}\left(q^{-1}\right)}{P\left(q^{-1}\right)} \\ \dfrac{KT_{21}\left(q^{-1}\right)B_{21}\left(q^{-1}\right)}{P\left(q^{-1}\right)} & \dfrac{KT_{22}\left(q^{-1}\right)B_{22}\left(q^{-1}\right)}{P\left(q^{-1}\right)} \end{bmatrix} \tag{4.33}$$

Where $P(q^{-1})$ defines the desired locations of the closed-loop system poles:

$$P\left(q^{-1}\right) = (1 - p_1q^{-1})(1 - p_2q^{-2})...........(1 - p_{np}q^{-1}) \qquad np = na + nc \tag{4.34}$$

The polynomials $C(q^{-1})$ and $D(q^{-1})$ are calculated to obtain the desired closed loop poles. The polynomial identity in (4. 34) has to be satisfied.

$$P(q^{-1}) = A_{ij}(q^{-1})C_{ij}(q^{-1}) + B_{ij}(q^{-1})_{ij}D_{ij}(q^{-1}) \tag{4.35}$$

From (4.33), it can be seen that the polynomial $T_{ij}(q^{-1})$ is used to introduce extra zeros, if needed, into the closed loop system.

The controller gain is chosen such that the low frequency gain is one, i.e. $Y(1)=R(1)$.

$$\frac{Y(1)}{R(1)} = \frac{KT_{ij}(1)B_{ij}(1)}{P(1)} \tag{4.36}$$

If the order of $T(q^{-1})$ is chosen to be zero, i.e. $T(q^{-1}) = 1$:

$$K = \frac{P(1)}{B_{ij}(1)} \tag{4.37}$$

The process is identified at each sample period. The control parameters are calculated using these identified process parameters. The control signal $u(t)$ can be determined from Figure 4.3:

$$u(t)_{2\times 1} = \frac{1}{C(q^{-1})}(Kr(t) - D(q^{-1})y(t)_{2\times 1}) \tag{4.38}$$

Eliminating $u(t)$ from (4.1) and (4.38) gives the following equations for the closed-loop system [12,15]:

$$y(t)_{2\times 1} = \frac{B(q^{-1})T(q^{-1})}{A(q^{-1})C(q^{-1}) + B(q^{-1})D(q^{-1})} r(t)_{2\times 1} \tag{4.39}$$

$$u(t)_{2\times 1} = \frac{A(q^{-1})T(q^{-1})}{A(q^{-1})C(q^{-1}) + B(q^{-1})D(q^{-1})} r(t)_{2\times 1} \tag{4.40}$$

Combining (4.39) and (4.40) and solving for $u(t)$:

$$P(q^{-1})u(t)_{2\times 1} = [A(q^{-1})T(q^{-1}) + B(q^{-1})T(q^{-1})]r(t) - P(q^{-1})y(t)_{2\times 1} \tag{4.41}$$

$$u(t)_{2\times 1} = \frac{1}{P(q^{-1})}[\{A(q^{-1})T(q^{-1}) + B(q^{-1})T(q^{-1})\}r(t) - P(q^{-1})y(t)_{2\times 1}] \tag{4.42}$$

$P(q^{-1})$ is obtained from equation (4.35). The control law (4.42) is similar to the law developed in (4.38). An indirect self-tuning regulator can hence be developed using this knowledge.

5. Simulations

Ultimately, the performance of adaptive AMB control strategies for flexible rotors was demonstrated experimentally as part of this research. Computer simulations, however, were used extensively to develop and test these algorithms. Adaptive control strategies were developed and tuned without the additional complexities of real-time implementations. Some benefits provided by a simulation environment include:

- Controllers can be developed and tested more efficiently in simulations
- Controller performance can be explored without sensor noise, modeling uncertainties, or real-time constraints
- The rotor can be positioned accurately and measured perfectly in simulations

The details of simulation-based design of adaptive control algorithms are presented in this section.

5.1 Model Reduction

The dynamic models used for simulation-based controller design must accurately represent the rotor and actuator dynamics so that real-time experimental demonstrations will be successful. For this reason, the *plant* dynamics were represented using the full-order rotor model (2.1) augmented with actuator dynamics (3.13). The resulting plant was a state-space model containing 324 states, and it captured all the dynamics within and beyond the controller bandwidth. These plant dynamics are summarized in the augmented state-space formulation:

$$\begin{aligned} \dot{z}_1 &= Az_1 + Bu \\ y &= Cz_1 + Du \end{aligned} \tag{5.1}$$

where:

$$z_1 = \begin{bmatrix} z & \dot{z} \end{bmatrix}^T \tag{5.2}$$

and the state matrices are:

$$A = \begin{bmatrix} 0 & I \\ -M^{-1}(K - K_s) & 0 \end{bmatrix}_{324\times 324} \tag{5.3}$$

$$B = \begin{bmatrix} 0 \\ M^{-1}K_i \end{bmatrix} \tag{5.4}$$

The actuators were placed under node 13 and node 69. Hence, C is a null (2×324) output matrix, with ones at the (1,25) and (2,137) elements, and D is a null (2×2) matrix.

The adaptive control structure of Figure 5.1 requires a reduced-order *model* to update the controller based on input-output measurements from the full-order *plant*. This *model* must include frequency components up to the controller bandwidth of 1000 Hz, but higher-order dynamics may be neglected for tractable implementation. Consequently, the *plant* had to be reduced such that the *model* contained only the dynamics summarized in Table 5.1.

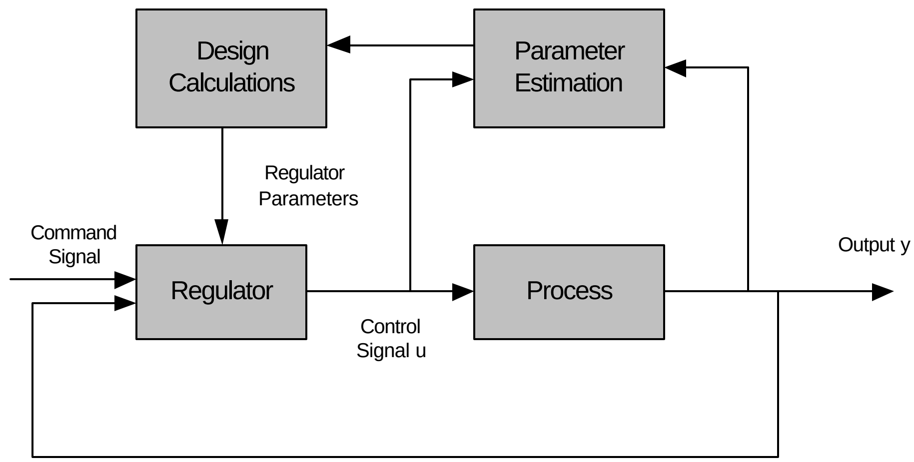


Figure 5.1: Self-tuning Regulator

Table 5.1: Natural frequencies of the plant

MODE	FREQUENCY (Hz)
1 (rotor rigid body)	0.0
2 (actuator electrical)	42.1
3 (first rotor flexible)	82.9
4 (second rotor flexible)	174.4
5 (third rotor flexible)	484.3
6 (fourth rotor flexible)	624.8

The second mode in Table 5.1 corresponds to the actuator electrical mode and is slightly less than the calculated frequency of 46 Hz shown in (3.15). The remaining frequencies correspond to flexible vibration modes of the rotor, as shown graphically in Figures 5.2 (a)-(e).

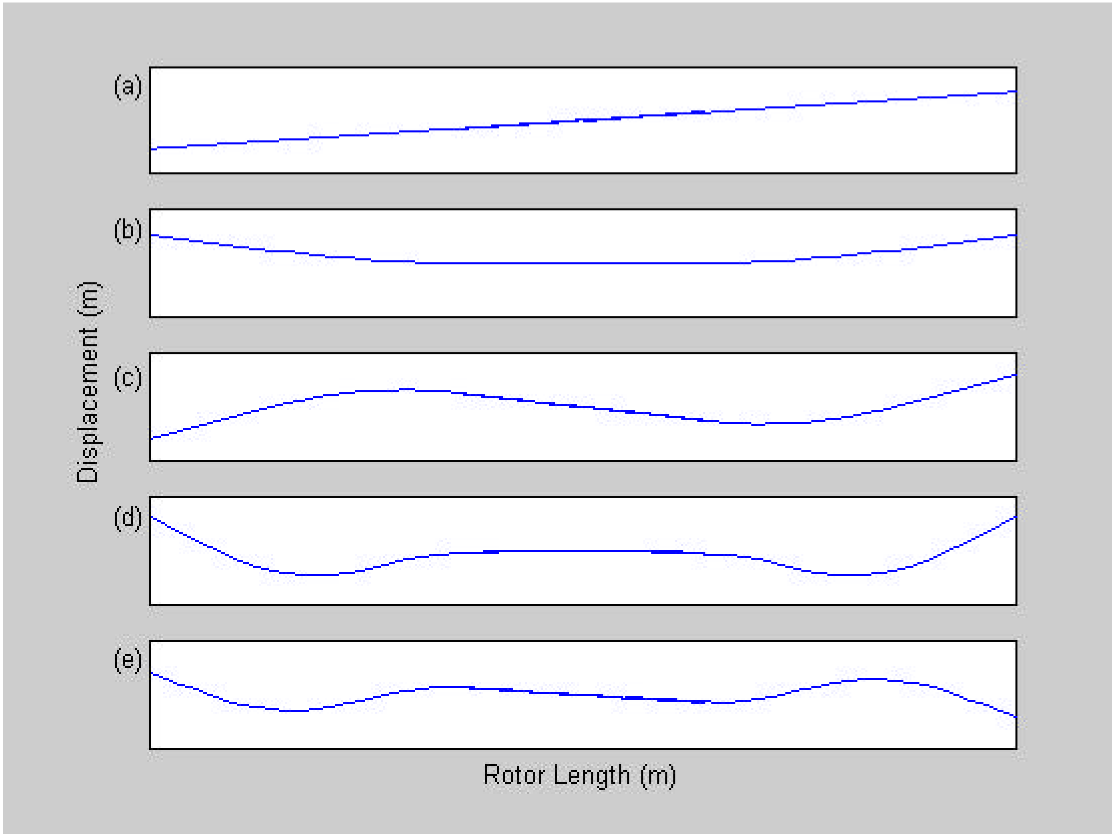


Figure 5.2 - Flexible rotor mode shapes
(a) Rigid body bode (0 Hz)
(b) First bending mode (82.9 Hz)
(c) Second bending mode (174.4 Hz)
(d) Third bending mode (484.3 Hz)
(e) Fourth bending mode (624.8 Hz)

To eliminate the high-frequency dynamics of the plant, a standard modal reduction technique was applied to the full-order state equations of equations (5.1–5.4), as outlined in Section 2.4 [10]. The resulting reduced-order dynamics are described by the following equations:

$$\dot{q}_1 = A_m q_1 + B_m u$$
$$y = C_m q_1 + D_m u \tag{5.5}$$

where:

$$q_1 = [q \quad \dot{q}]^T \tag{5.6}$$

and the state matrices become:

$$A_m = \begin{bmatrix} 0 & I \\ \boldsymbol{W} & 0 \end{bmatrix}_{324\times 324} \tag{5.7}$$

$$B_m = \begin{bmatrix} 0 \\ Q \end{bmatrix}_{324\times 2} \tag{5.8}$$

where $\boldsymbol{W}$ is a diagonal matrix of squared eigenvalues:

$$\boldsymbol{W} = \begin{bmatrix} \boldsymbol{w}_{n1}^2 & 0 & & 0 \\ 0 & \boldsymbol{w}_{n2}^2 & & 0 \\ & & \ddots & \\ 0 & 0 & & \boldsymbol{w}_{nn}^2 \end{bmatrix} \tag{5.9}$$

and Q represents the modal force input:

$$Q = Y^T M^{-1} K_i \tag{5.10}$$

The output modal matrix is:

$$C_m = Y^T C \tag{5.11}$$

To demonstrate the effectiveness of this model reduction process, impulse response simulations were conducted using the full-order *plant* (5.1) and the reduced order *model* (5.5). Impulsive voltages were applied to both actuators, and the undamped response was visually compared for both models. Figure 5.3 (a) shows the impulse response of the full-order plant (324 states). The impulse response of the reduced-order model (10 states) is shown in Figure 5.3 (b). It should be stressed that that these simulations provide only a qualitative visualization of dynamics of the rotor system. However, it is clear that the elimination of 314 high frequency states did not drastically affect the dynamics of the rotor model.

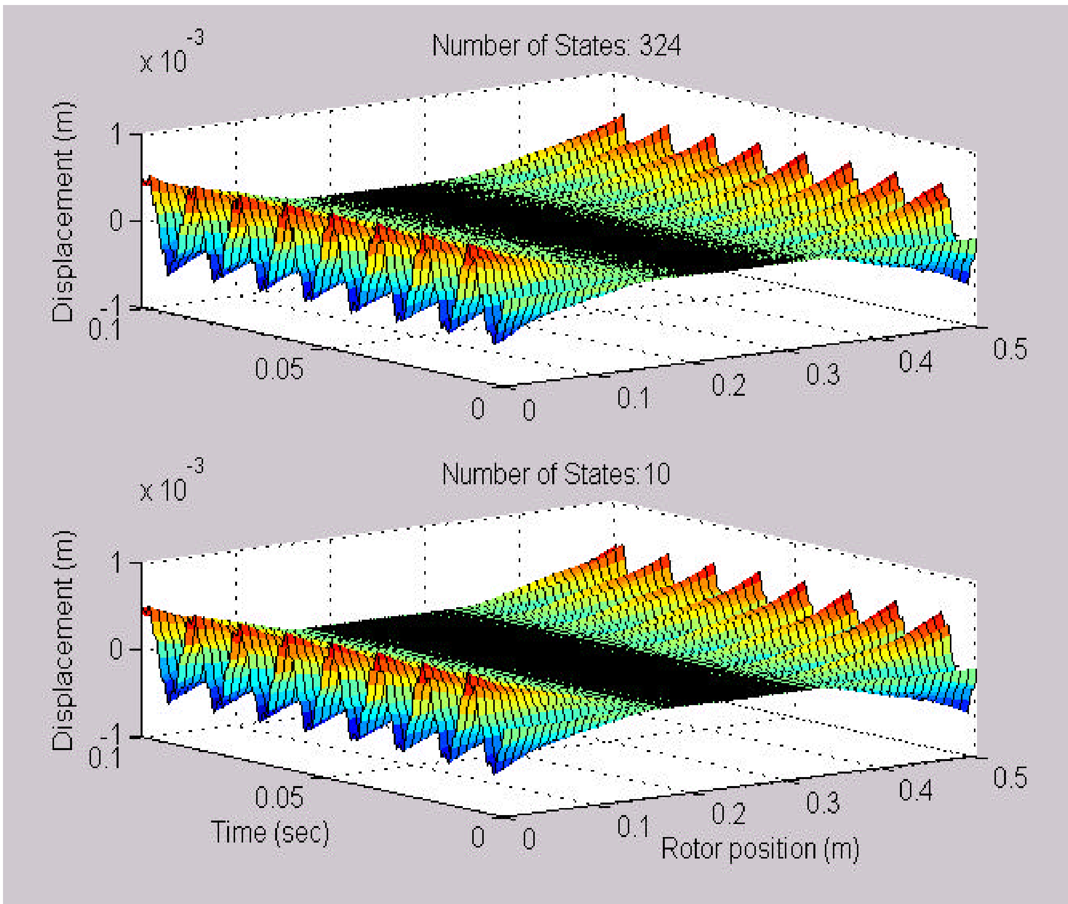


Figure 5.3: (a) Waterfall plot of the *plant*
(b) Waterfall plot of the reduced order *model*

Figure 5.4 compares the frequency responses of the *plant* and *model*. Because this system is multivariable, the maximum singular values are plotted for comparison. It can be observed that the frequencies retained in the reduced order model coincide with the first five frequencies of the *plant*, after which the magnitudes begin to roll off. The frequencies that coincide are the frequencies required to design the control law.

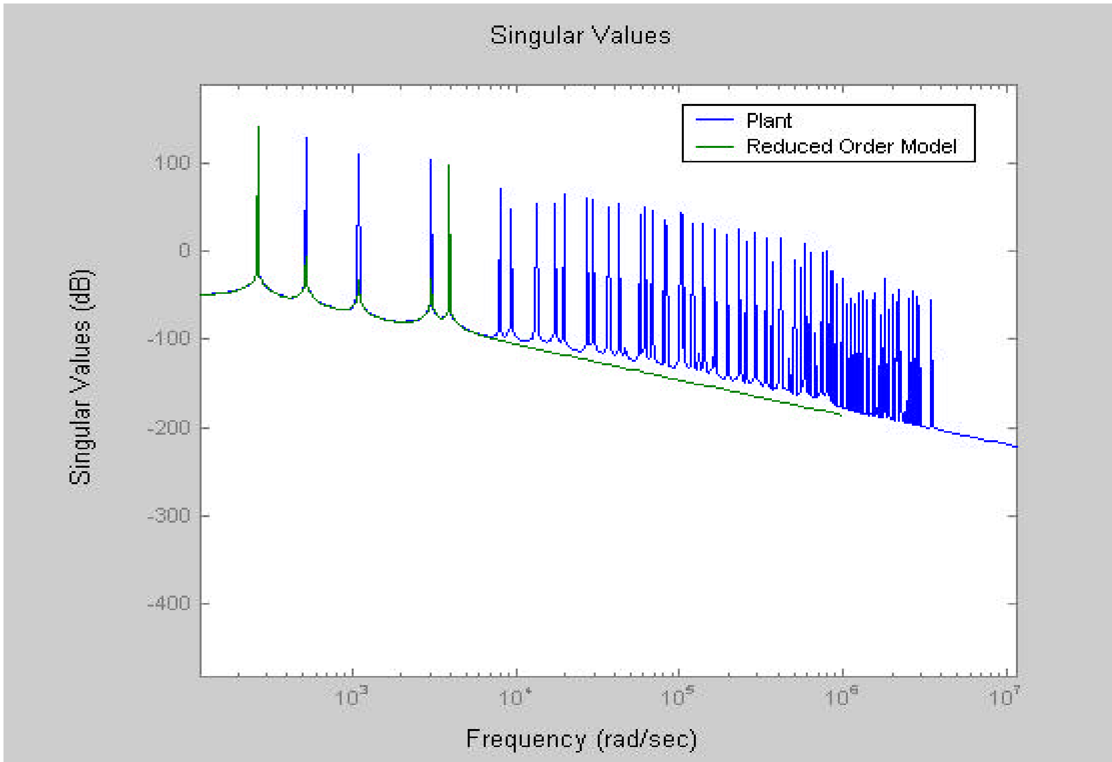


Figure 5.4: Maximum singular values for the plant and reduced-order model

5.2 Control Implementation

The self-tuning, pole placement controller described in Section 4.4 was implemented for simulations in MATLAB and Simulink. Figure 5.5 shows the Simulink implementation of the self-tuning regulator. This figure reveals one of the assets of graphical programming: the program corresponds directly to the STR structure of Figure 5.1. The pole-placement controller (4.38) is contained in the "Controller" subsystem, the full-order plant (5.1) is contained in the "Discrete State Space" subsystem, and the system identification algorithms (Section 4.3) are contained in the "SysID" subsystem. Reference inputs and disturbance inputs are included as shown.

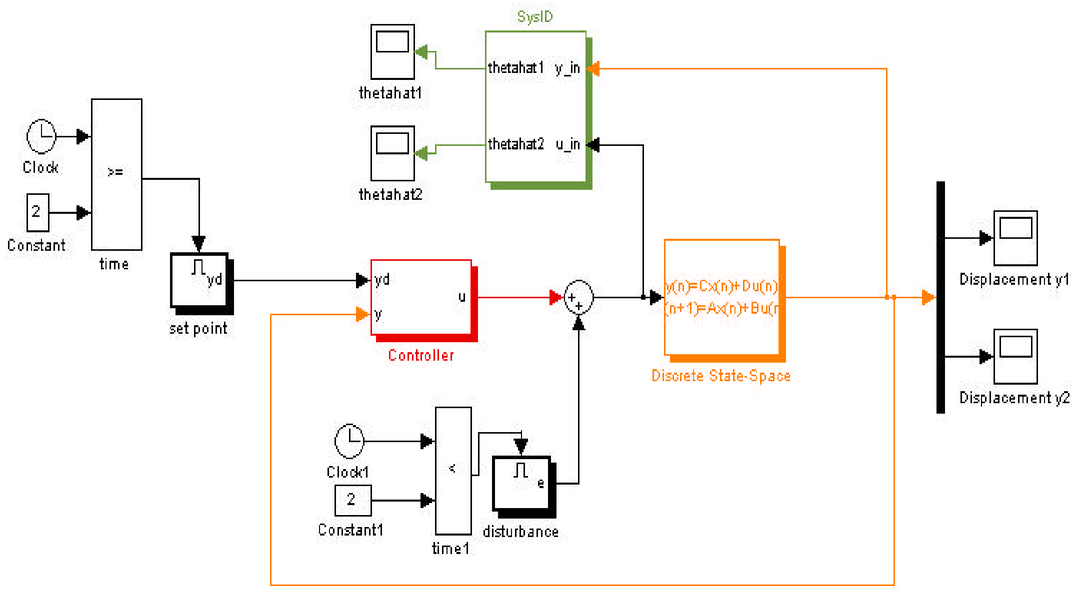


Figure 5.5: Simulink implementation of the self-tuning regulator

The reduced-order model (5.7) was discretized to accommodate data acquisition using the *C2D* command of MATLAB. The sample time, in this case 0.0008 sec, was chosen such that the natural frequencies of the discretized model were the same as the natural frequencies of the continuous time reduced-order model.

5.3 Simulation Results

The primary objective of the computer simulations was to demonstrate improving tracking performance despite a low-order model with significant parametric uncertainties. For this reason, the model parameters were initialized with +/- 10% random errors (normally distributed). The tracking setpoint was a square wave with amplitude 0.001m and a frequency 0.1 Hz. For the first two seconds of each simulation, however, the setpoint was selected to be random noise to ensure persistent excitation for accurate system identification.

Figure 5.6 shows results for a typical simulation *without adaptation* (fixed-gain control). Figure 5.6 (a) shows the setpoint and tracking response of $y_1(t)$, the measured displacement at the left optical sensor. Figure 5.6 (b) shows the setpoint and tracking response of $y_2(t)$, the measured displacement at the right optical sensor. The parameter estimates $\hat{q}_1(t_0)$ and $\hat{q}_2(t_0)$ were initialized with +/-10% random errors (normally distributed), resulting in slightly asymmetrical responses. Both tracking responses are good, though obviously the unmodeled high-frequency modes are not being controlled.

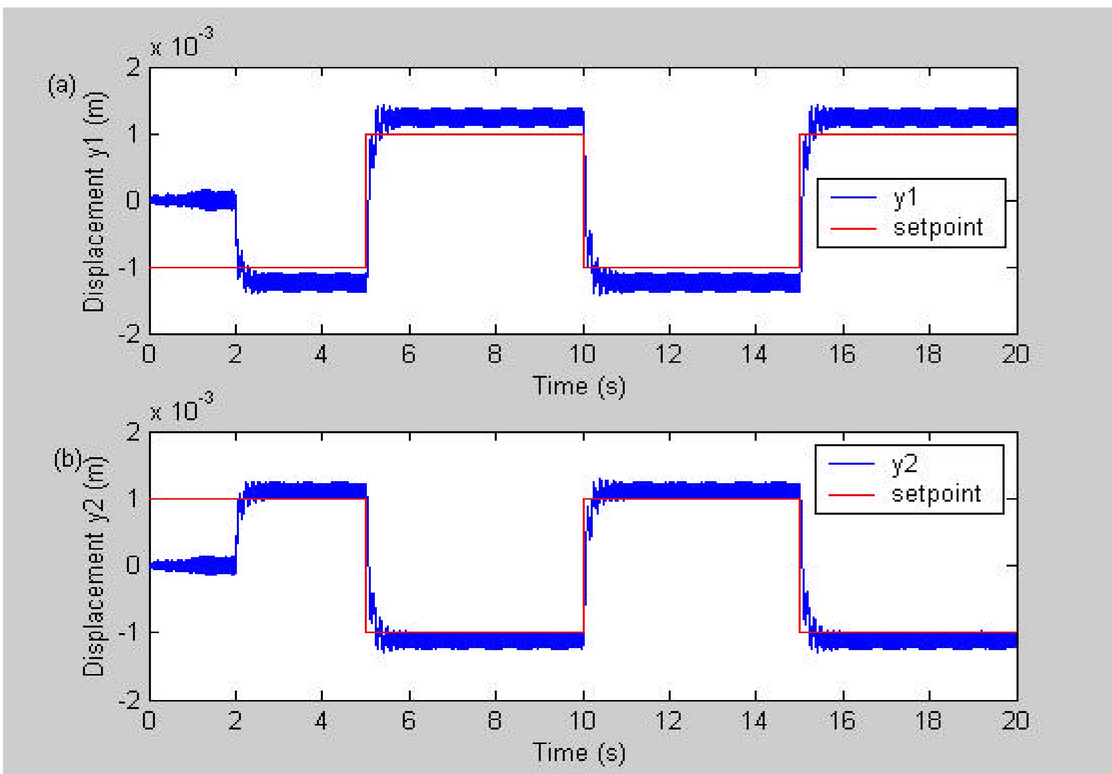


Figure 5.6: Tracking Responses for a non-adapting controller
(a) Displacement y_1(t) at the left optical sensor
(b) Displacement y_2(t) at the right optical sensor

The tracking performance for this non-adapting simulation, evaluated using a sum of squared tracking errors, is:

$$
\begin{aligned}
&T_i = \sum_{t=1}^{n}\left(y_{di}(t) - y_i(t)\right)^2 \\
&T_1 = 4.51e\text{-}011 \\
&T_2 = 9.10e\text{-}011
\end{aligned}
\tag{5.12}
$$

where $y_{di}(t)$ is the desired trajectory, or setpoint, for the i^{th} output and $y_i(t)$ is the measured output. Figure 5.7 shows results for a typical simulation *with adaptation* (self-tuning control). Figure 5.7(a) shows the setpoint and tracking response of $y_1(t)$, the measured displacement at the left optical sensor. Figure 5.7(b) shows the setpoint and tracking response of $y_2(t)$, the measured displacement at the right optical sensor. The parameter estimates $\hat{q}_1(t_0)$ and $\hat{q}_2(t_0)$ were initialized with +/-10% random errors (normally distributed), resulting in slightly asymmetrical responses. Both tracking responses are significantly better than before, though again the unmodeled high-frequency modes are not being controlled.

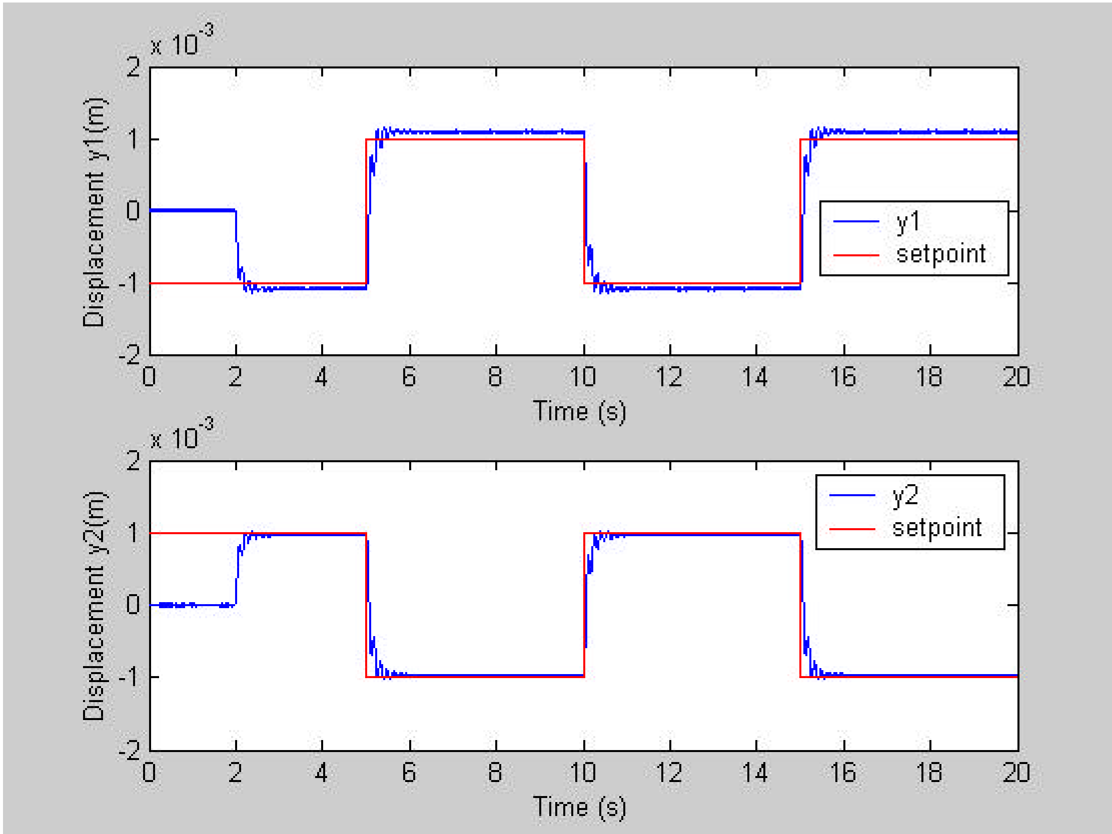


Figure 5.7: Tracking Responses for the adapting controller
(a) Displacement $y_1(t)$ at the left optical sensor
(b) Displacement $y_2(t)$ at the right optical sensor

The tracking performance for this adaptive simulation, again evaluated using the sum of squared tracking errors, is:

$$\begin{aligned} T_i &= \sum_{t=1}^{n} \left(y_{di}(t) - y_i(t)\right)^2 \\ T_1 &= 8.95e\text{-}013 \\ T_2 &= 4.66e\text{-}013 \end{aligned} \tag{5.13}$$

The actuator control currents were limited to a range of 1.5 A, as shown in Figure 5.8.

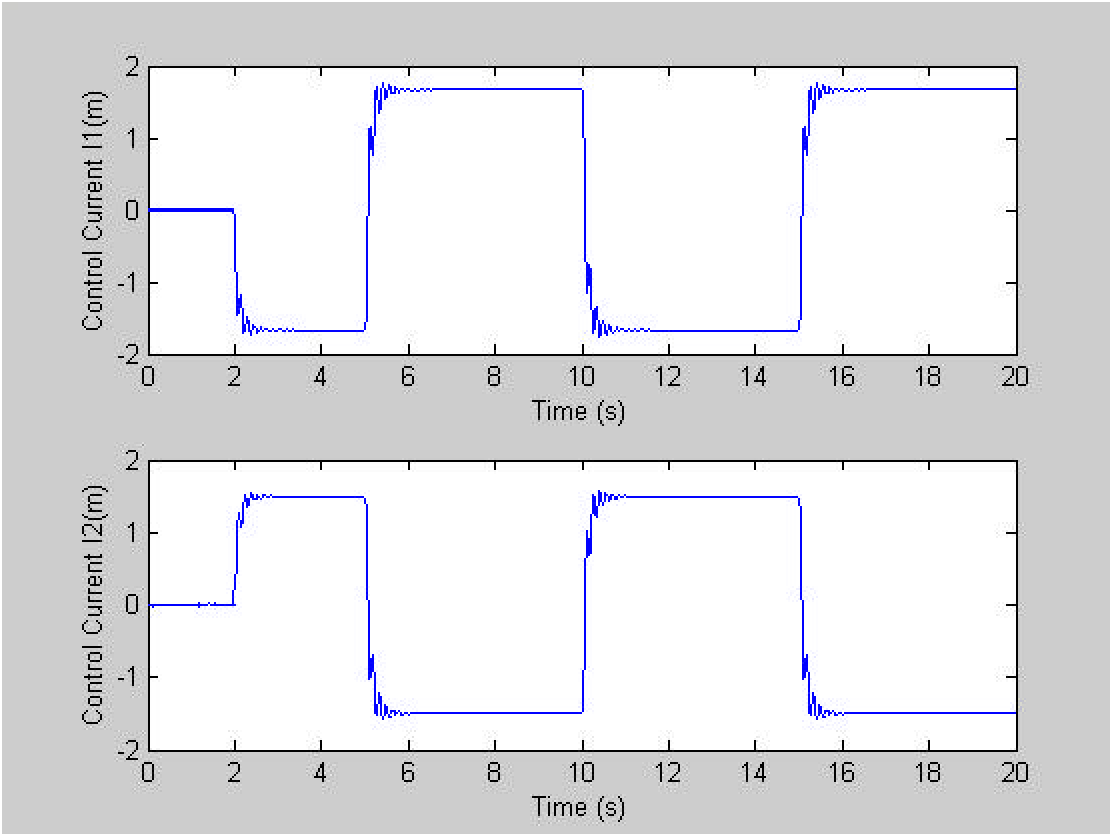


Figure 5.8: Control Currents
(a) Actuator 1
(b) Actuator 2

A total of 60 parameters were estimated "on-line" during the course of each simulation, and each of these parameters converged rapidly to within 0.0001% of their "true" values. Figure 5.9 shows the parameter convergence for two of the 60 model parameters during a typical simulation.

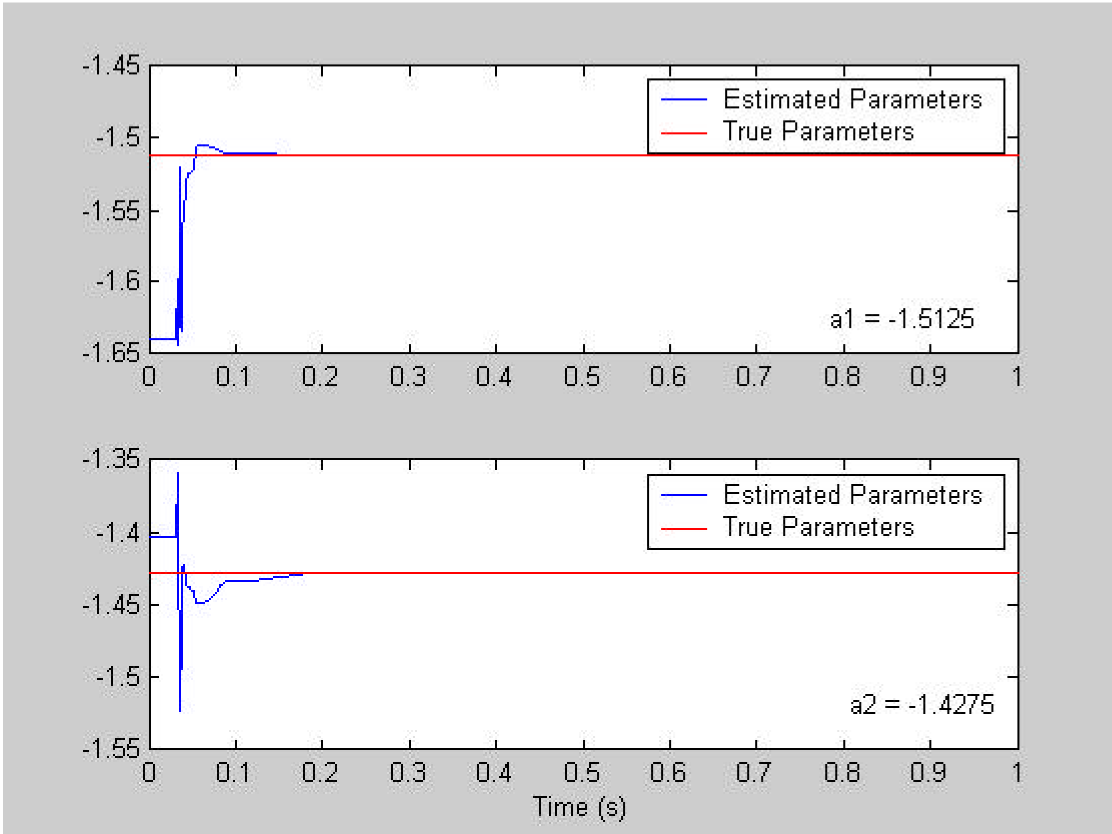


Figure 5.9: Convergence of the first two coefficients of polynomial A with time

Another measure of the parameter estimation effectiveness is the prediction error cost function shown in Figure 5.9. This cost function is defined as the sum of squared errors between the "true" model parameters and the instantaneous estimates at each simulation time:

$$J_i = \frac{1}{2}\sum_{t=1}^{n}\left(y_i(t) - \hat{y}_i(t)\right)^2 \tag{5.14}$$

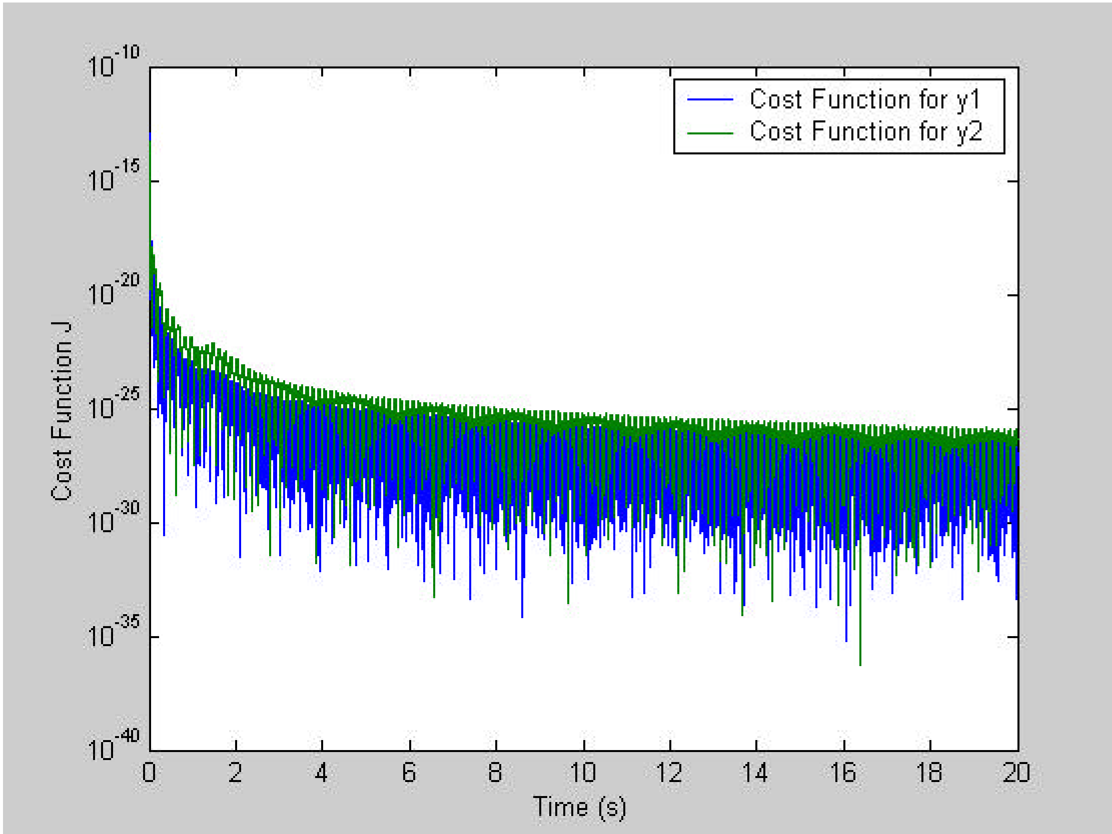


Figure 5.10: Plot of the parameter estimation cost function J

Table 5.2 compares all 60 estimated model parameters to their "true" values after 30 seconds of simulation.

Although the model parameters of Table 5.2 were updated at each time step (0.0008 sec), the controller gains were updated every 3.0 seconds. The initial controller gains were based on a model with +/- 10% parametric uncertainties, thus these controller gains "improved" as they were updated. This improved the overall performance of the control system, as indicated in Figure 5.8. Tables 5.3-5.6 show the "true" and final controller gains from these simulations.

Table 5.2: Comparison of estimated and "true" model parameters

	Parameters	True Parameters	Estimated Parameters		Parameters	True Parameters	Estimated Parameters
A	a1	-1.5125	-1.5125	A	a1	-1.5125	-1.5125
	a2	-1.4275	-1.4275		a2	-1.4275	-1.4275
	a3	3.502	3.502		a3	3.502	3.5021
	a4	0.44975	0.44975		a4	0.44975	0.44975
	a5	-3.9427	-3.9427		a5	-3.9427	-3.9427
	a6	0.44975	0.44975		a6	0.44975	0.44975
	a7	3.502	3.502		a7	3.502	3.5021
	a8	-1.4275	-1.4275		a8	-1.4275	-1.4275
	a9	-1.5125	-1.5125		a9	-1.5125	-1.5125
	a10	1	1		a10	1	1
B11	b1	3.52E-05	3.52E-05	B12	b1	-5.98E-07	-5.98E-07
	b2	-2.60E-05	-2.60E-05		b2	-2.57E-05	-2.57E-05
	b3	-2.80E-05	-2.80E-05		b3	3.16E-06	3.16E-06
	b4	6.14E-05	6.14E-05		b4	7.37E-05	7.37E-05
	b5	-2.65E-05	-2.65E-05		b5	-6.22E-05	-6.22E-05
	b6	-2.65E-05	-2.65E-05		b6	-6.22E-05	-6.22E-05
	b7	6.14E-05	6.14E-05		b7	7.37E-05	7.37E-05
	b8	-2.80E-05	-2.80E-05		b8	3.16E-06	3.16E-06
	b9	-2.60E-05	-2.60E-05		b9	-2.57E-05	-2.57E-05
	b10	3.52E-05	3.52E-05		b10	-5.98E-07	-5.98E-07
B13	b1	-5.98E-07	-5.98E-07	B14	b1	2.80E-05	2.80E-05
	b2	-2.57E-05	-2.57E-05		b2	-1.39E-05	-1.39E-05
	b3	3.16E-06	3.16E-06		b3	-3.41E-05	-3.41E-05
	b4	7.37E-05	7.37E-05		b4	3.82E-05	3.82E-05
	b5	-6.22E-05	-6.22E-05		b5	-5.00E-06	-5.00E-06
	b6	-6.22E-05	-6.22E-05		b6	-5.00E-06	-5.00E-06
	b7	7.37E-05	7.37E-05		b7	3.82E-05	3.82E-05
	b8	3.16E-06	3.16E-06		b8	-3.41E-05	-3.41E-05
	b9	-2.57E-05	-2.57E-05		b9	-1.39E-05	-1.39E-05
	b10	-5.98E-07	-5.98E-07		b10	2.80E-05	2.80E-05

Table 5.3: Comparison of controller parameters for transfer function (1,1)

Polynomial C	True Parameters	Updated Parameters	Polynomial D	True Parameters	Updated Parameters
C1	-3.6365	-3.6365	D0	1.2578e+005	1.2578e+005
C2	17.257	17.257	D1	-5.4328e+005	-5.4328e+005
C3	-18.404	-18.404	D2	7.3261e+005	7.3261e+005
C4	-3.8008	-3.8008	D3	2.6959e+005	2.6959e+005
C5	12.584	12.584	D4	-1.0067e+006	-1.0067e+006
C6	-18.29	-18.29	D5	-41854	-41855
C7	-6.2043	-6.2043	D6	1.019e+006	1.019e+006
C8	14.325	14.325	D7	-2.0882e+005	-2.0882e+005
C9	-20.325	-20.325	D8	-8.5799e+005	-8.5799e+005
			D9	5.7566e+005	5.7566e+005

Table 5.4: Comparison of controller parameters for transfer function (1,2)

Polynomial C	True Parameters	Updated Parameters	Polynomial D	True Parameters	Updated Parameters
C1	0.25094	0.25094	D0	-2.1095e+006	-2.1095e+006
C2	-50.225	-50.225	D1	3.731e+006	3.731e+006
C3	23.682	23.682	D2	2.4274e+005	2.4274e+005
C4	110.2	110.2	D3	-4.3296e+006	-4.3296e+006
C5	-137.71	-137.71	D4	9.6257e+005	9.6257e+005
C6	13.099	13.099	D5	4.0991e+006	4.0991e+006
C7	48.317	48.317	D6	-1.8504e+006	-1.8504e+006
C8	-15.006	-15.006	D7	-2.9599e+006	-2.9599e+006
C9	-0.37425	-0.37425	D8	2.7732e+006	2.7732e+006
			D9	-6.2581e+005	-6.2581e+005

Table 5.5: Comparison of controller parameters for transfer function (2,1)

Polynomial C	True Parameters	Updated Parameters	Polynomial D	True Parameters	Updated Parameters
C1	0.25094	0.25094	D0	-2.1095e+006	-2.1095e+006
C2	-50.225	-50.225	D1	3.731e+006	3.731e+006
C3	23.682	23.682	D2	2.4274e+005	2.4274e+005
C4	110.2	110.2	D3	-4.3296e+006	-4.3296e+006
C5	-137.71	-137.71	D4	9.6257e+005	9.6257e+005
C6	13.099	13.099	D5	4.0991e+006	4.0991e+006
C7	48.317	48.317	D6	-1.8504e+006	-1.8504e+006
C8	-15.006	-15.006	D7	-2.9599e+006	-2.9599e+006
C9	-0.37425	-0.37425	D8	2.7732e+006	2.7732e+006
			D9	-6.2581e+005	-6.2581e+005

Table 5.6: Comparison of controller parameters for transfer function (2,2)

Polynomial C	True Parameters	Updated Parameters	Polynomial D	True Parameters	Updated Parameters
C1	-8.1025	-8.1022	D0	3.1793e+005	3.1792e+005
C2	28.473	28.473	D1	-1.2853e+006	-1.2853e+006
C3	-31.034	-31.034	D2	1.881e+006	1.881e+006
C4	-22.353	-22.353	D3	5.2174e+005	5.2173e+005
C5	22.742	22.742	D4	-2.5401e+006	-2.5401e+006
C6	-31.1	-31.099	D5	80860	80877
C7	4.8925	4.8927	D6	2.527e+006	2.527e+006
C8	15.559	15.559	D7	-7.209e+005	-7.2091e+005
C9	-43.396	-43.396	D8	-2.1379e+006	-2.1379e+006
			D9	1.5491e+006	1.5492e+006

In each case, the controller gains converge to within 0.003% of their “true” values, thus improving the performance of the rotor bearing system.

5.4 Simulation Conclusions

Computer simulations demonstrated that the self-tuning, pole placement controller was very effective in controlling the full-order plant. The parameter estimation algorithms were highly effective, rapidly converging to within 0.0001% of the “true” model parameters. Conclusions and lessons learned from these simulations included:

- The estimated model parameters converged rapidly and stabily to the true parameters in less than 1.0 second
- The system performance improved each time the controller gains were updated
- The self-tuning controller demonstrated robustness to extreme variations in initial model parameters and disturbances
- Persistent excitation was critical for accurate system identification; this necessitated the random setpoint for the first two seconds of each simulation

6. Experimental Demonstrations

To validate the results obtained from computer simulations, an experimental test rig was fabricated based on the rotor design work of Section 2.2 and the actuator design work of Section 3.4. Real-time experiments were conducted to demonstrate self-tuning control of the flexible structure supported by active magnetic bearings. These experiments validated the findings of Section 5, especially for cases where mass imbalances were included.

There were many challenges faced while implementing the control law in real-time, including:

- The system identification algorithms are extremely taxing on the real-time hardware; it was unknown if any parameter estimation algorithm could be implemented successfully in real-time
- Heating and eddy current losses were expected to reduce the performance of the actuator
- Sensor noise and disturbances were expected to reduce the performance of the controller

The details of these experimental validations are presented in this section.

6.1 Flexible Rotor AMB Test Rig

The flexible rotor described in Section 6.1 was machined from 1040 mild steel, and the test rig base and fixtures were machined from aluminum 6061 T6. Slots were machined into the rotor to accommodate mass imbalances at different locations. The reconfigurable base was designed to mount the rotor and actuators (AMBs) at various locations. The complete test rig is shown in Figure 6.1. Figure 6.2 shows a dimensioned schematic of the base and test fixtures.

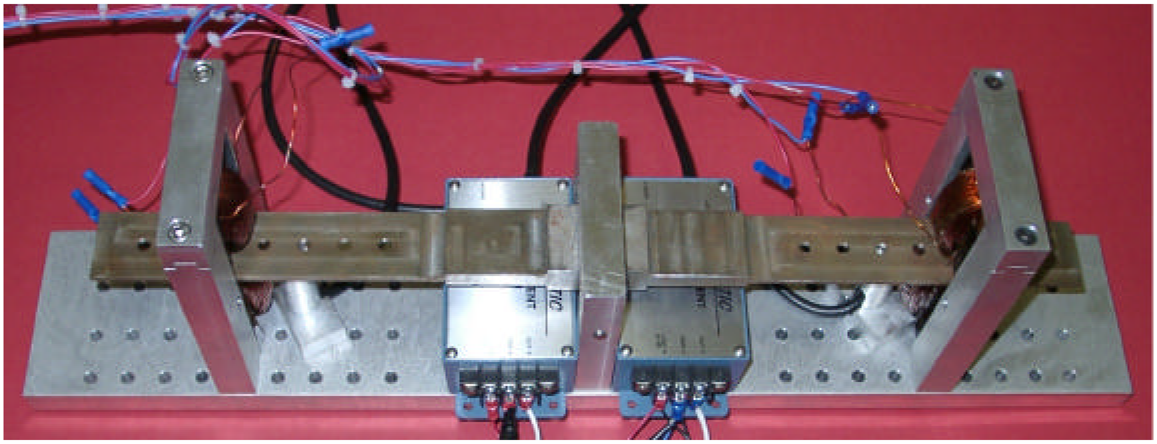


Figure 6.1: Flexible Rotor AMB Test Rig

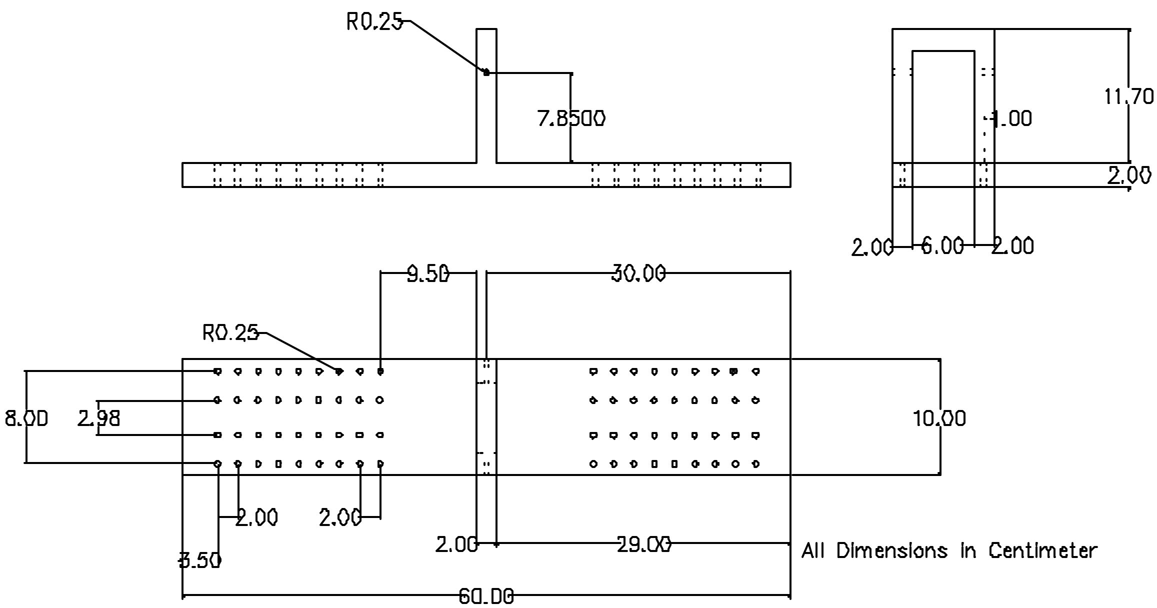


Figure 6.2: Base and Fixture Geometry

Each actuator was constructed of M-19 laminated steel to reduce eddy currents and hysteresis losses, as described in Section 3.3. AWG 20 copper magnet wire was used to wind the actuators. The actuators were baked in a protective varnish to prevent any short circuits due to kinks in the actuator windings. Figure 6.3 shows a dimensioned schematic of the actuator.

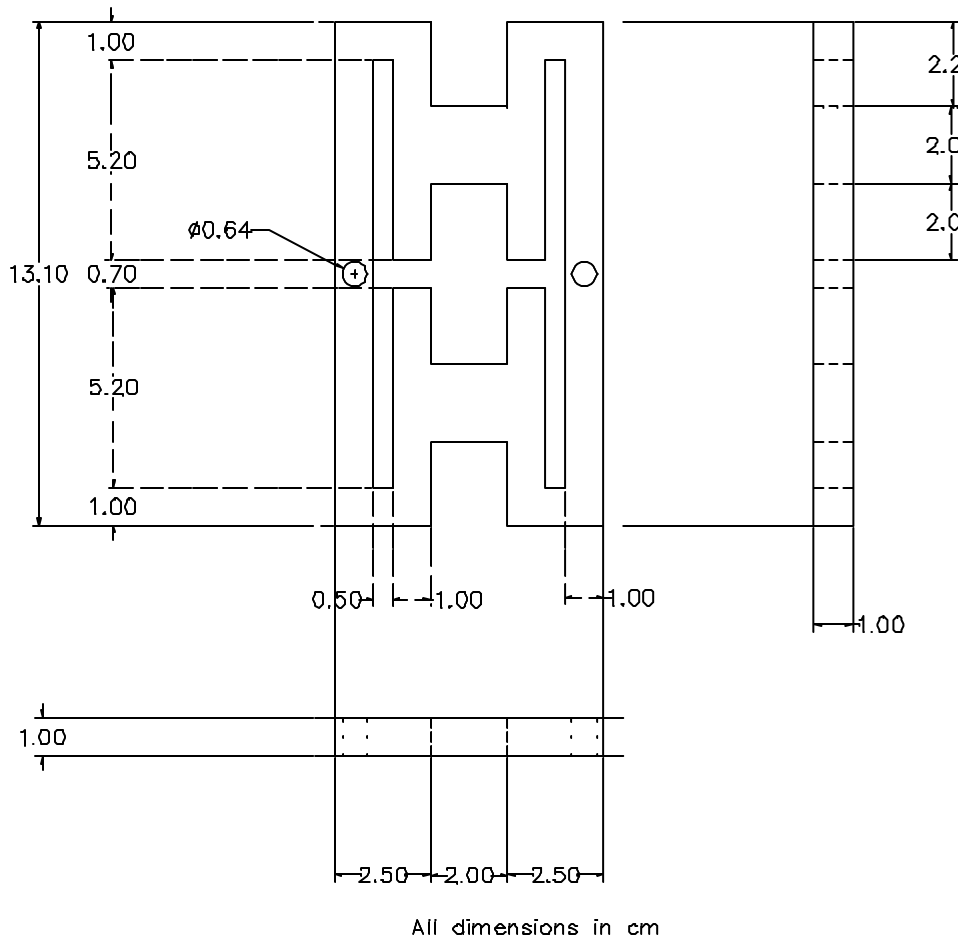


Figure 6.3: Actuator Geometry

PhilTec RC-89 reflectance-compensated optical transducers were used to measure the rotor displacements. These sensors were selected so that the 3mm of total travel was approximately within the linear measurement range. Figure 6.4 shows calibration curves for each sensor.

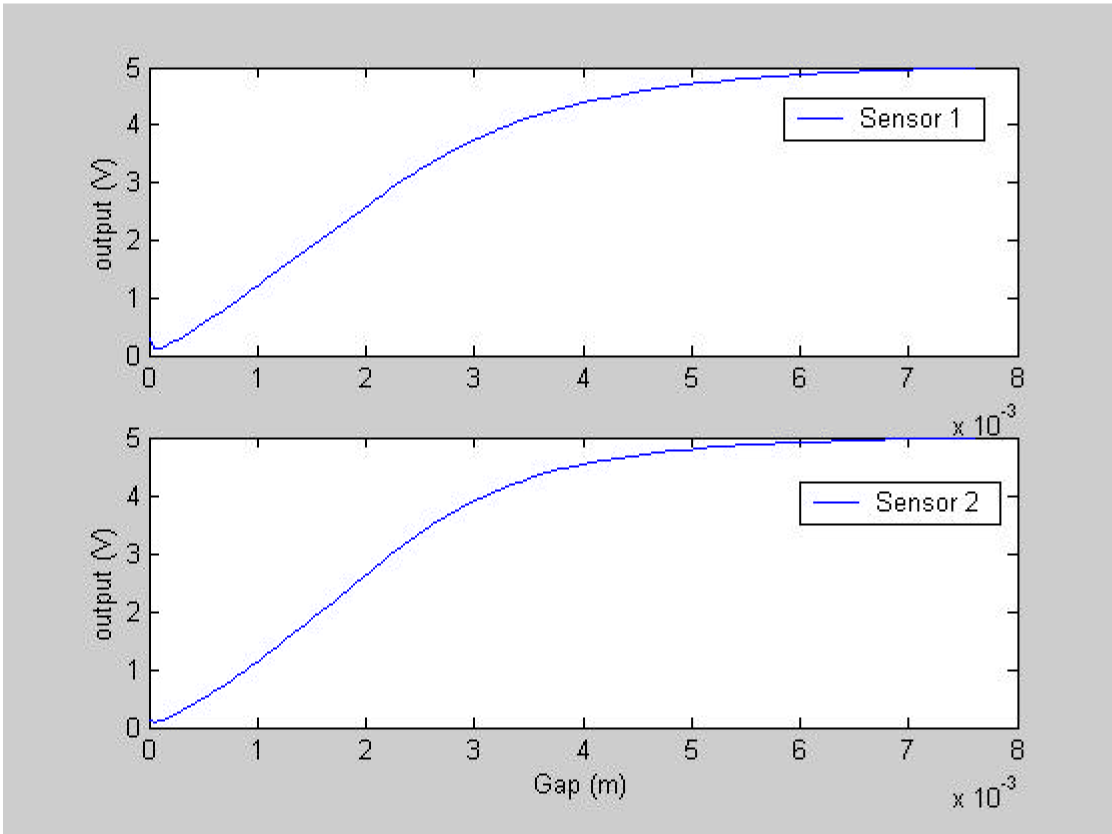


Figure 6.4: Calibration Curves for the Optical Displacement Sensors

6.2 Real-Time Control Platform

Real-time, "hardware-in-the-loop" testing provides the greatest insight into how well a control design will perform under "real world" conditions (with noisy sensors, heating actuators and realistic test conditions). A dSPACE (digital Signal Processing And Control Engineering) 1102 ACE kit was used for embedded real-time system identification and control. This hardware platform was programmed using Simulink©

and the Real Time Workshop©, which interfaces with dSPACE's real time system to form an integrated development and testing environment.

The pole-placement control law (4.42) derived in Section 4.4, along with system identification algorithms for a reduced-order model (4.13 - 4.25), were programmed in Simulink and implemented into dSPACE. Figure 6.5 shows the highest-level Simulink diagram used to implement data acquisition, system identification, and control.

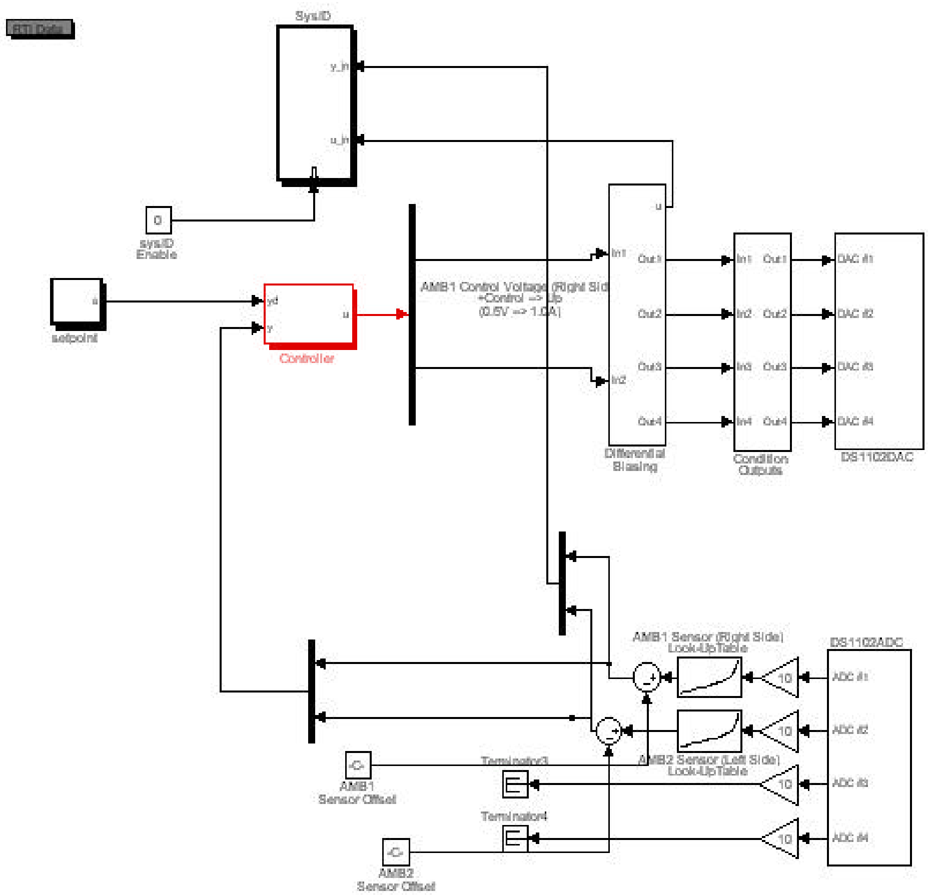


Figure 6.5: Simulink diagram for real-time system identification and control

6.3 Adaptive Control Evaluations

Real-time experiments were conducted to demonstrate self-tuning control of the flexible structure supported by AMBs. The setpoint for these tests was a series of square waves with frequencies ranging from 10 rad/sec to 100 rad/sec and a fixed amplitude of 0.2 mm. A fixed sample time of 0.0015 seconds was used for these tests, resulting in a controller bandwidth of 666.67 Hz.

Although 2^{nd} through 10^{th}-order models were evaluated in the computer simulations of Section 5.2, limitations in the real-time control hardware restricted the experimental tests to 2^{nd}-order models only. The parameters of a second-order model were estimated in the system identification block of Figure 6.5. However, these parameters were not used to update the controller online. The controller parameters were updated offline and the performances were compared.

Case 1 - 10 rad/s, 0.2 mm setpoint, no mass imbalance

The output responses to a 10 rad/sec, 0.2mm amplitude square wave setpoint, using initial controller parameters and updated control parameters, are shown in Figure 6.6. In each case, the output tracking is very good. In fact, the controller performance (measured using a sum of squared tracking errors cost function) actually decreased slightly when the controller parameters were updated:

$$\begin{aligned} T_i &= \sum_{t=1}^{n} \left(y_{di}(t) - y_i(t) \right)^2 \\ T_1 &= 4.28e\text{-}010 \\ T_2 &= 4.65e\text{-}010 \end{aligned} \tag{6.1}$$

The real-time parameter estimation converged rapidly, stably, and consistently to the same model parameters for each experimental test. The predicted rotor displacements (the green traces in Figure 6.6) are nearly indistinguishable from the measured displacements (the blue traces in Figure 6.6). Figure 6 .7 shows the control input currents for the 10 rad /sec setpoint for both actuators.

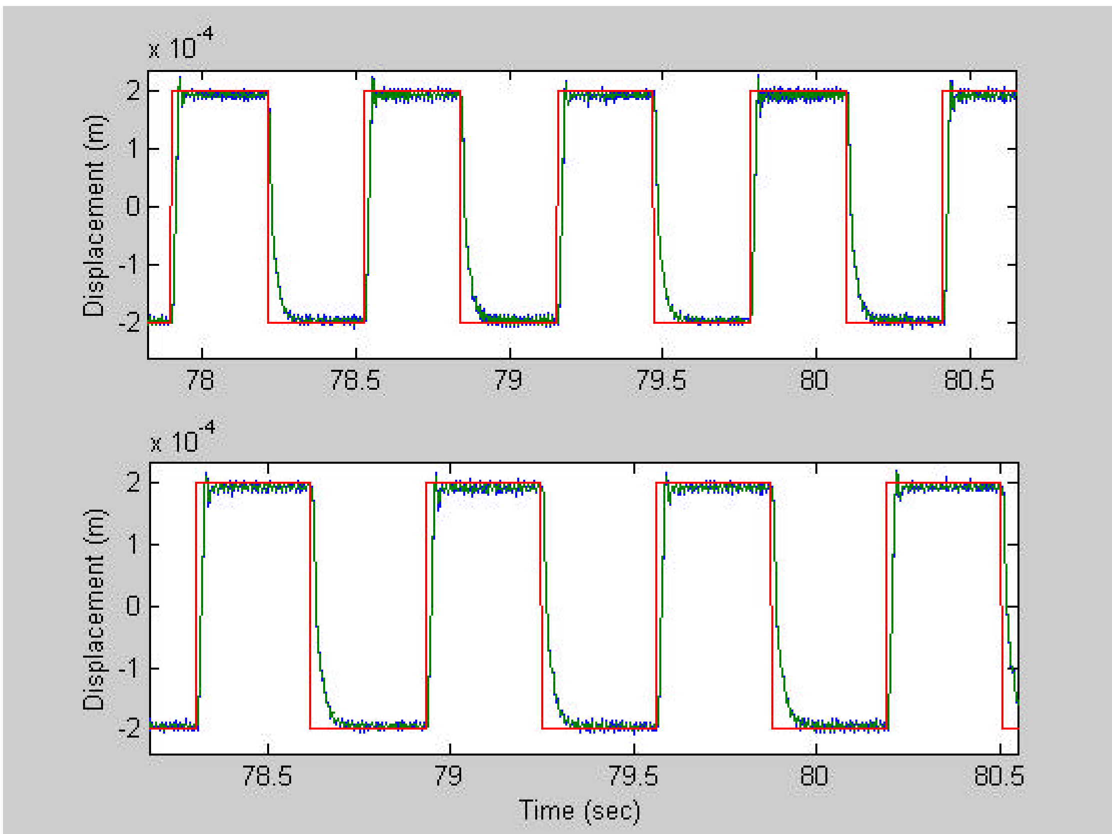


Figure 6.6: Tracking performance with initial controller gains (top) and updated controller gains (bottom) - setpoint 0.2mm, 10 rad/s

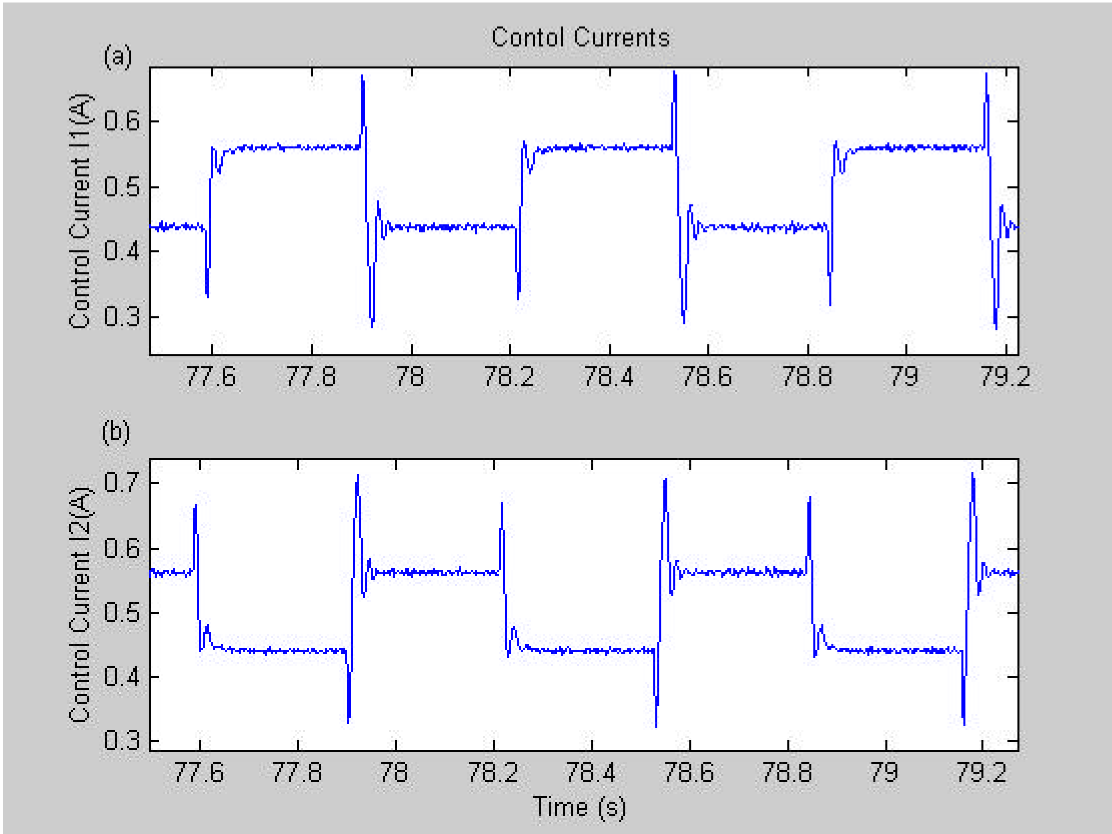


Figure 6.7: Actuator control currents for AMB #1 (top) and AMB #2 (bottom) - setpoint 0.2mm, 10 rad/s

<u>Case 2 - 10 rad/s, 0.2 mm setpoint, with mass imbalance</u>

A mass imbalance was added to the right side of the rotor to study the effect of offline adaptive control. This imbalance had a mass of approximately 0.73 kg and was positioned approximately 4 cm outside the right AMB. The initial model parameters and controller gains were obtained from FEA analyses that did not include this mass imbalance.

For the same tracking setpoint used in Case 1, the initial controller performance was very poor, as shown in Figure 6.8 (a). The large tracking offset error (not seen in Figure 6.6 (a)) can be directly attributed to the mass imbalance. Despite the poor tracking performance, the parameter estimation algorithm was again highly effective. The predicted rotor displacements (the green traces in Figure 6.8) are nearly indistinguishable from the measured displacements (the blue traces in Figure 6.8).

When the controller gains were updated based on real-time parameter estimation, the controller performance increased significantly, as shown in Figure 6.8 (b) and evidenced in the tracking cost functions:

$$\begin{aligned} T_i &= \sum_{t=1}^{n} \left(y_{di}(t) - y_i(t)\right)^2 \\ T_1 &= 3.69e-009 \\ T_2 &= 1.74e-011 \end{aligned} \tag{6.2}$$

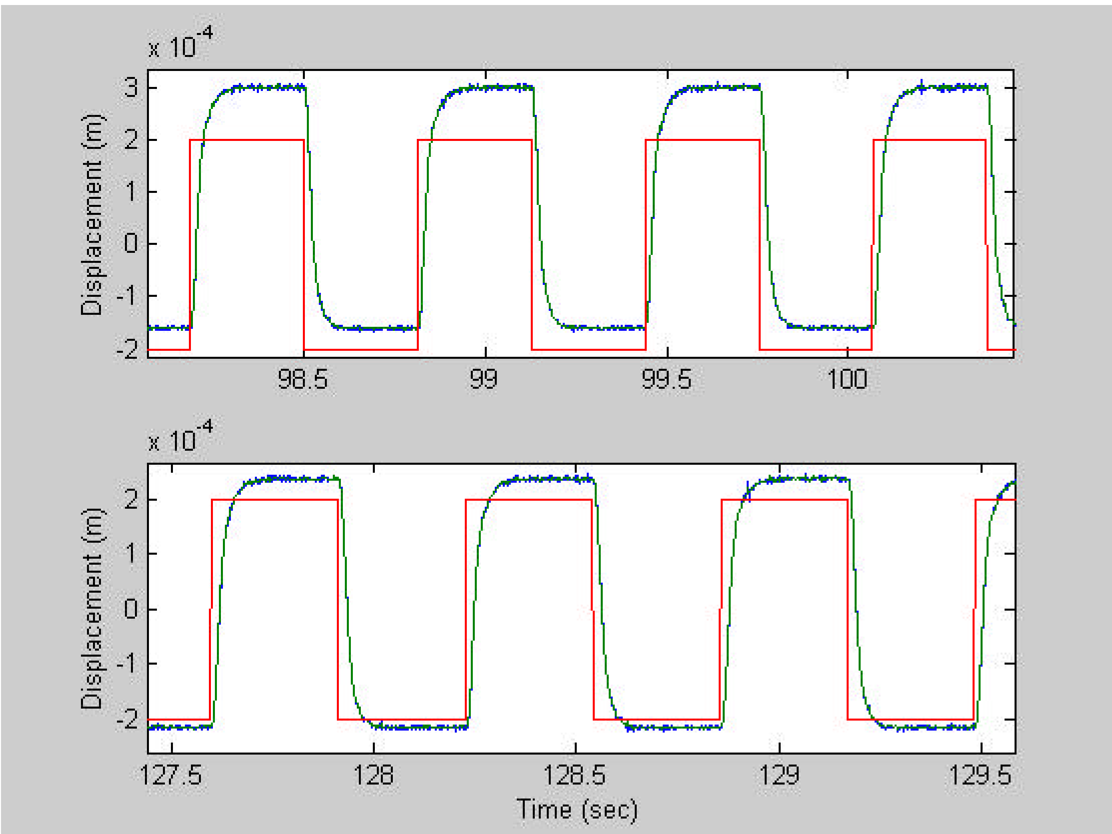


Figure 6.8: Tracking performance when the plant has a mass imbalance with initial controller gains (top) and updated control gains at 10 rad/s

<u>Case 3 - 50 rad/s, 0.2 mm setpoint, no mass imbalance</u>

The output responses to a 50 rad/sec, 0.2mm amplitude square wave setpoint, using initial controller parameters and updated control parameters, are shown in Figure 6.9. In each case, the output tracking is good but clearly inferior to Case 1. The controller performance did improve when the controller parameters were updated:

$$\begin{aligned} T_i &= \sum_{t=1}^{n}\left(y_{di}(t) - y_i(t)\right)^2 \\ T_1 &= 1.41e\text{-}010 \\ T_2 &= 1.40e\text{-}010 \end{aligned} \tag{6.3}$$

The parameter estimation performance was again very good, as shown in Figure 6.9.

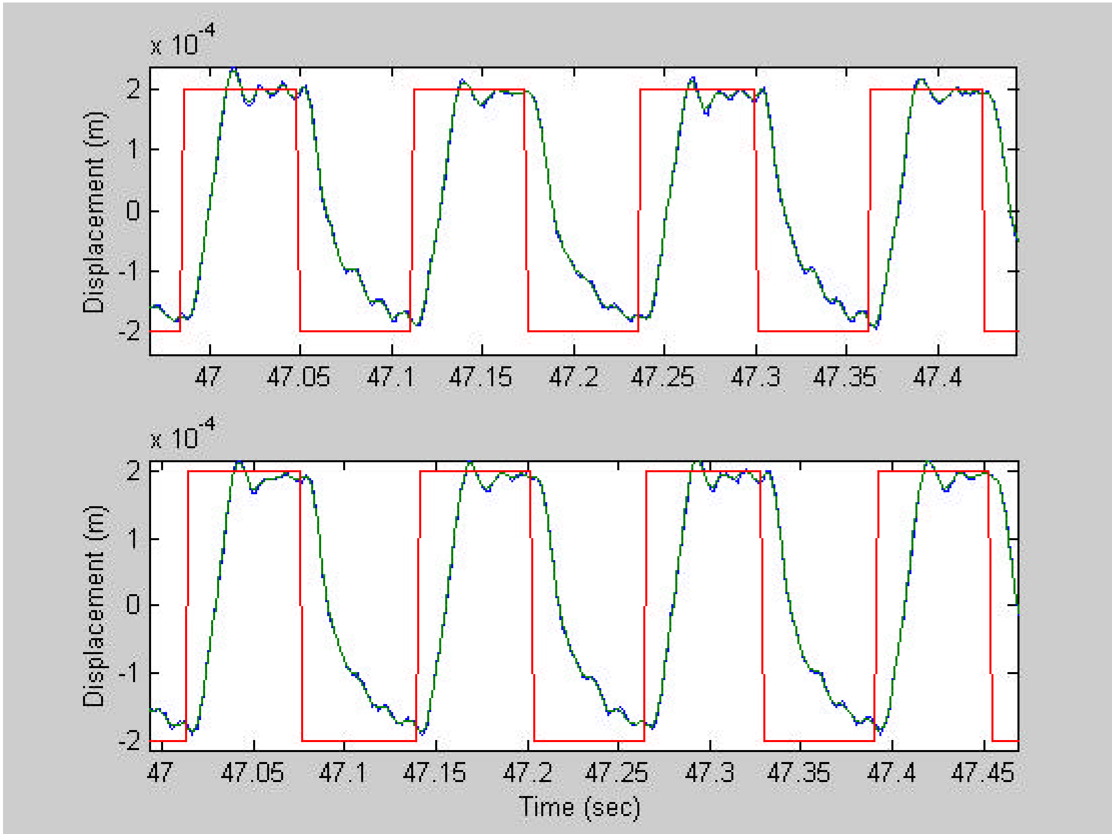


Figure 6.9: Tracking performance with initial controller gains (top) and updated controller gains (bottom) - setpoint 0.2mm, 50 rad/s

<u>Case 4 - 50 rad/s, 0.2 mm setpoint, with mass imbalance</u>

Again the controller robustness to parametric uncertainties was investigated by adding a discrete mass imbalance to the rotor. For the same tracking setpoint used in Case 3, the initial controller performance was very poor, as shown in Figure 6.10 (a). The large tracking offset error can be directly attributed to the mass imbalance. Despite the poor tracking performance, the parameter estimation algorithm was again highly effective. The predicted rotor displacements (the green traces in Figure 6.8) are nearly indistinguishable from the measured displacements (the blue traces in Figure 6.8).

When the controller gains were updated based on real-time parameter estimation, the controller performance increased significantly, as shown in Figure 6.10 (b) and evidenced in the tracking cost functions:

$$\begin{aligned} T_i &= \sum_{t=1}^{n} \left(y_{di}(t) - y_i(t) \right)^2 \\ T_1 &= 4.13e\text{-}009 \\ T_2 &= 3.77e\text{-}011 \end{aligned} \tag{6.4}$$

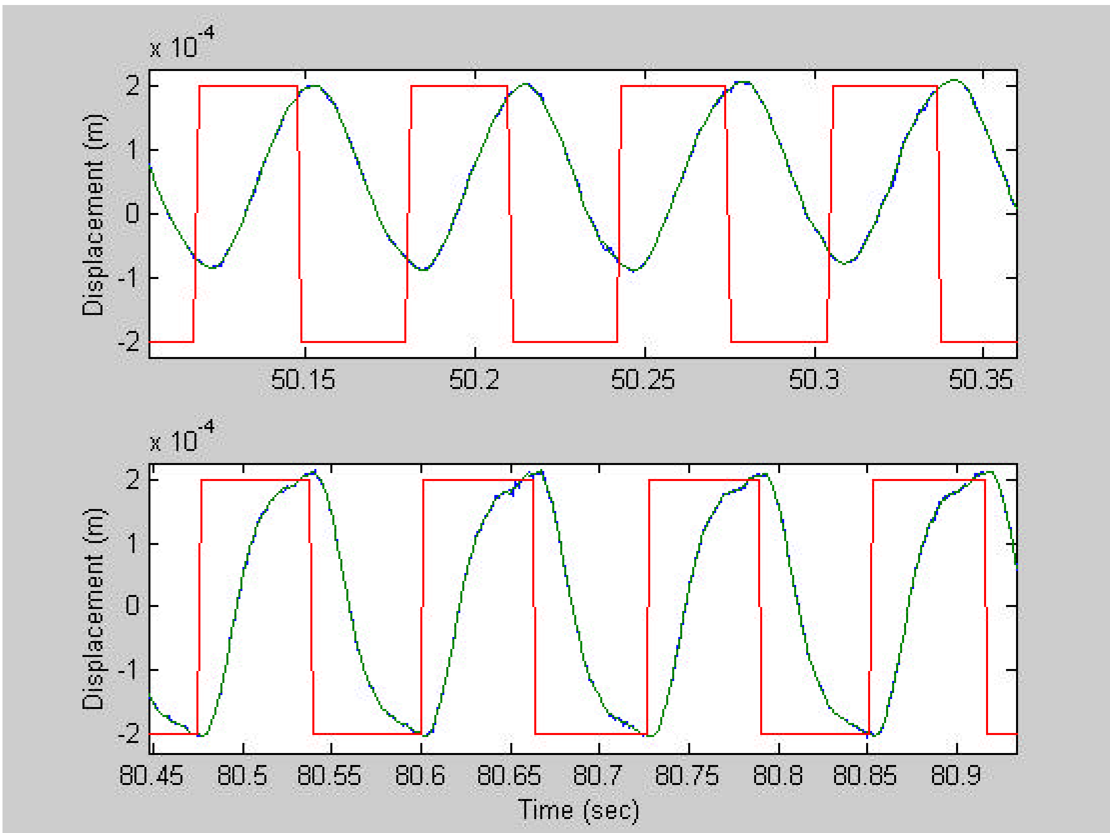


Figure 6.10: Tracking performance when the plant has a mass imbalance with initial controller gains (top) and updated control gains at 50 rad/s

6.4 Experimental Conclusions

Experimental tests using a flexible rotor AMB test rig demonstrated that the self-tuning, pole placement controller was both effective and robust in controlling the full-order plant. The parameter estimation algorithms were highly effective, rapidly and consistently converging to the same model parameters. The tracking performance improved considerably when the controller gains were updated, particularly when mass imbalances were included. Table 6.1 compares the performance for systems with initial controller parameters and updated controller parameters for all cases.

Table 6.1: Performance cost function at different set point frequencies

	Without Mass Imbalance			With Mass Imbalance		
Setpoint Frequency (rad/sec)	Tracking Cost (Initial Controller Parameters)	Tracking Cost (Updated Controller Parameters)	Percent Improve	Tracking Cost (Initial Controller Parameters)	Tracking Cost (Updated Controller Parameters)	Percent Improve
10	4.28e-010	4.65e-010	-8.66	3.69e-009	1.74e-011	+99.53
30	4.05e-010	3.92e-010	+3.21	3.72e-009	1.47e-011	+99.61
50	1.41e-010	1.40e-010	+1.49	4.13e-009	3.77e-011	+99.09
100	1.35e-011	6.29e-012	+5.33	4.51e-009	1.08e-010	+97.60

For plants with no mass imbalances, there were only marginal performance improvements associated with updated controller parameters; the initial controllers and updated controllers both exhibited good model following properties (Figure 6.6). When the plant had an added mass imbalance, system performance increased by almost 100% when the controller was updated. The initial controller showed very poor model following properties (Figures 6.8, 6.10) at all setpoint frequencies. Once the controller parameters were updated, the tracking performance increased significantly.

Conclusions and lessons learned from these experiments included:

- The estimated model parameters converged rapidly and stabily to the true parameters in less than 1.0 second
- The system performance improved each time the controller gains were updated
- The self-tuning controller demonstrated robustness to parameter variations
- The system identification algorithms were extremely taxing on the real-time hardware; although computer simulations were tested for 2^{nd}-10^{th} order models, only 2^{nd}-order models could be evaluated in real-time

7. Conclusions

This thesis has demonstrated the benefits that self-tuning, multivariable control provides for AMB levitation of flexible rotors. These benefits include a reduction in the required *a priori* knowledge of model parameters, and an extension in the range of operating conditions that can be accommodated. The techniques developed and demonstrated in this thesis enable complicated control issues to be resolved in real-time.

The self-tuning, pole-placement controllers developed for reduced-order models were novel in both their application and their structure. The self-tuning regulator was designed to update the controller parameters online, and the improvements in system responses were quite evident when compared to non-adaptive controllers. The controller was designed to have a 1000 Hz bandwidth and to generate sufficient current to generate forces to dampen the flexible modes of the rotor within its operating range.

The development of dynamic models of the flexible rotor and actuator enabled various controllers to be tried and tested in a realistic simulation environment before being applied experimentally. This provided insights as to how the rotor would react when forces were applied on different locations and the implications on position control. The validation of controller performance and robustness was paramount, and every effort was made to ensure that there was close correspondence between the simulation and experimental pole-placement regulators.

The actuators designed and built for real-time implementation proved to be efficient in producing the required control forces, and enabled the adaptive controller to good model following properties. Though the actuator bandwidth was far below the controller bandwidth, these actuators performed well within their limitations. The actuators were well designed from a heating standpoint, as they did not heat up even when used for extended periods in time.

The adaptive control algorithms were implemented in real-time to validate the control law that was developed in Chapter 4. Due to hardware limitations, the control law was developed for only a second-order system, and the controller parameters were updated offline. The system performance improved significantly when the controller parameters were updated, especially for cases involving mass imbalances. Hence, the adaptive algorithms developed for this thesis showed improved performance in both computer simulations and real-time implementations.

References:

[1] M. Komori, N-Akinaga, “A prototype of flywheel energy storage system suppressed by hybrid magnetic bearings with H_∞ controller”, IEEE Transactions on Applied Superconductivity, vol.11, no.1, pt.2, March 2001, pp.1733-6

[2] K.Komori, T.Yamane, “Magnetically levitated micro PM motors by two types of active magnetic bearings”, IEEE/ASME Transactions on Mechatronics, vol.6, no.1, March 2001, pp.43-9

[3] F.Kleiner, B.de-Wit; B.Ponick, “Ecology and economy decide for 50000 hp electric turbocompressor drivers”, Conference Record of the 2000 IEEE Industry Applications Conference, IEEE, Piscataway, NJ, USA, 2000, vol. 4, pp. 2738-48

[4] R. Siegwart, “Design and application of Active Magnetic Bearings for vibration control,” Von Karman Inst. Fluid Dynamics Lecture Series 1992-06, Vibration and Rotor Dynamics, Sept. 21-25, 1992

[5] G. Schweitzer, H. Bleuler, and A. Traxler, Active Magnetic Bearings: Basics, Properties and Applications, vdf Hochschulverlag AG and der ETH, Zurich Germany, 1994

[6] M.Fujita, F.Matsumara and T.Namerikawa, “ m analysis and synthesis of a flexible beam magnetic suspension system”, 3rd International Symposium of Magnetic Bearings, 1992, pp. 495-504

[7] Kenzo Nonami and Takayuki Ito,” m Synthesis of Flexible Rotor-Magnetic Bearing Systems”, IEEE Transactions on Control Systems Technology, vol.4, No. 5, September 1996

[8] Gangbing Song, Ranjan Mukherjee, “Integrated Adaptive Control of Active Magnetic Bearings”, IEEE International Electric Machines and Drives Conference, IEMDC '99 Proceedings (Cat. No.99EX272), IEEE, Piscataway, NJ, USA

[9] I.Bar-Kana, H.Kaufman and M.Balas, “ Model Reference Adaptive Control of Large Structural Systems”, Journal of Guidance, March 1983

[10] Singiresu S.Rao, Mechanical Vibrations, 3rd Edition, Addison-Wesley, Reading, MA, 1995

[11] A.E. Fritzgerald, Charles Kingsley, Jr., and Stephen D. Umans, Electric Machinery, Fifth Edition, McGraw-Hill Series in Electrical Engineering, 1990

[12] J. Schmied, F. Betschon, “Engineering for rotors supported on magnetic bearings: The process and the tools”

[13] K.J. Astrom and B. Wittenmark, Adaptive Control, 2nd Edition, Addison Wesley, 1995

[14] R.E. Kalman, “Design of self-optimizing control systems”, ASME Transactions, vol. 80, pp. 468-478, 1958

[15] L. Ljung, System Identification: Theory for the User, Prentice Hall, Englewood Cliffs, NJ, 1987

[16] K.J. Astrom and B. Wittenmark, “On self tuning regulators”, Automatica, vol. 9, pp. 185-199, 1973

[17] L. Ljung, The Control Handbook, CRC press, Inc., 1996

[18] P.E. Wellstead, J.M. Edmunds, D. Prager and P. Zanker, ”Self-tuning pole/zero assignment regulators”, Int. J. Control, vol. 30, no. 1, pp. 1-26, 1979

[19] D.L. Prager and P.E. Wellstead, “Multivariable pole-assignment self-tuning regulators”, IEE Proceedings, vol. 128, January 1980