Adaptive Feedforward Servo Control Design For Dual-Stage Bit-Patterned Media Recording Hard Disk Drives

by

Behrooz Shahsavari, Ehsan Keikha, Fu Zhang

Computer Mechanics Laboraty Blue Report

University of California, Berkeley

Adviser:

Professor Roberto Horowitz

Spring 2015

Contents

List of Figures

List of Tables

Chapter 1

Abstract

This paper presents an adaptive repetitive controller for active tracking (rejecting) of unknown periodic trajectories (disturbances). The proposed control law is based on a *modified filtered-x least mean squares* (MFX-LMS) algorithm with a novel variable step size that improves the convergence rate, and fades the steady state *excess error* in a stochastic environment. A novel *secondary path modeling* scheme is also proposed to adaptively compensate for the dynamic mismatches between the internal model of the MFX-LMS and the real dynamic system in an online fashion. We further discuss the application of this adaptive controller in servo mechanisms for hard disk drives that use *Bit Patterned Media Recording*, in which, full spectrum tracking of a periodic trajectory is crucial. Finally, comprehensive numerical simulations and experimental implementations are presented for a hard disk drive servo system that is subjected to periodic disturbances known as repeatable runout.

Chapter 2

Introduction

Repetitive control was first introduced in 1980's, and since then, has been widely used in applications in which a task should be performed repeatedly, a periodic disturbance must be attenuated or a periodic trajectory must be tracked. Repetitive control has been applied in many robot manipulators applications, thermal cycling, milling machines and satellite attitude control. This control methodology has also been applied in hard disk drives to follow periodic trajectories (or reject periodic disturbances), in order to read/write data on a magnetic disk surface. This is the application that will be considered in this paper.

Repetitive controllers are typically categorized into two types, namely internal and external model based controllers, based on how the cancellation signal is considered in the algorithm. Internal model based repetitive controllers utilize an embedded signal generator **chew1989digital** whereas external model based controllers view the cancellation signal as being injected from outside of the plant-controller feedback loop **messner1991new**

In this paper, we will use the external model based approach and employ a stochastic gradient descent method, which is known as the *modified filtered-x least mean squares (MFX-LMS)* algorithm, to develop the parameter adaptation algorithm of the repetitive controller. The MFX-LMS is a sophisticated version of the least mean squares (LMS) algorithm **bjarnason1992noise**; **kim1994constraint** The main advantage of the MFX-LMS, as compared to the LMS, is its high stability when the error signal is not correctly "aligned" in time with the reference signal **elliott1985application** Indeed, the LMS algorithm is generally unstable when the control signal and reference signal enters to a dynamic system at distinct points. More formally, in a linear time invariant (LTI) framework, the LMS algorithm diverges if the phase of the closed loop system from the control injection point to the *error* signal exceeds or lags by 90 degrees at the actuation frequency range. On the other hand, the MFX-LMS algorithm uses a secondary path to "align" the control signal and the error signal by using an internal model of the closed loop system. We will be returning to this concept later in section 3.1. Nevertheless, the MFX-LMS algorithm will diverge if the internal model, used in the secondary path, does not match the actual closed loop dynamics very well. This divergence behavior happens when the phase mismatch between the nominal model and real closed loop system exceeds 90 degrees at the excitation frequency range **lopes2004behavior** We propose a self-tuning secondary path identification scheme in section 4 to compensate for these dynamics mismatches in an online fashion, in order to prevent the divergence of the adaptive algorithm.

In many active disturbance cancellation or reference signal tracking applications, the unknown signal that needs to be compensated for or tracked can be decomposed into a product of a regressor vector and an vector of unknown constant parameters that need to be estimated. The adaptive algorithms that identify the unknown parameter vector can be categorized into two types based on whether the regressor signal is known *a-priori*, or has to be generated from measurements in real time. The former type uses a known deterministic regressor to learn and compensate for the disturbances. Such is the case of the repetitive adaptive controller for tracking repeatable runout in hard disk drives (HDDs) that will be presented in this paper, where it is known a-priori what are the frequency contents of the repeatable runout spectrum that should be tracked. In the later type of adaptive algorithms, it is necessary to use a sensor (e.g. a microphone or an accelerometer) to generate the regressor signal in real time. Such is the case of noise cancelation applications and external vibration compensation control systems in HDDs.

The repetitive control application that will be considered in this paper is track-following of high frequency repeatable read/write tracks in computer magnetic hard disk drives (HDDs) that use *Bit Patterned Media Recording.* The head positioning servomechanism in a HDD moves the magnetic read/write head as quickly as possible from one track to another when asked by the host system using track-seeking and track-settling control systems. Once the head reaches the target track, its position relative to the track's center is controlled by a track-following servo system during the data reading and writing process. HDDs generate position feedback signals from special magnetic patterns called servo patterns that are written at designated areas on the disk surface known as servo sectors. The generated feedback signals are called position error signals (PES) and are read by the magnetic read head, as it flies over the servo sectors **Abramovitch2002**; **al2007hard**

In recent years, a potential breakthrough to increase the areal density of HDDs, known as *bit patterned media recording* (BPMR), has been under development. In BPMR each magnetic data bit is recorded on a single-domain magnetic island (one bit per island), as opposed to storing bits in 20-30 magnetic grains within a continuous magnetic film **white1997patterned**; **ross2001patterned** However, BPMR requires that the data tracks be followed with significantly more accuracy than what is required in conventional continuos media recording, since the read/write head has to be accurately positioned over the single-domain magnetic islands to read or write data. The shape of each individual BPMR servo/data track is patterned on the disk using some form of nano-lithography process, and its variations relative to a perfect circular track result in written-in runout which becomes repeatable (periodic) due to the disk spinning. The BPMR written in repeatable runout, which often contains high frequency components, must be accurately tracked by the servo-system.

Section 5 discusses a repetitive adaptive control system that is designed to track BPMR written in repeatable runout and extends the results reported in our previous conference papers **shahsavari2014repeatable**; **shahsavari2014adaptive** The effectiveness of the proposed controller is evaluated through a comprehensive numerical simulation study, as presented in section 6. Furthermore, the controller was is implemented on a 2.5-inch "HGST-a Western Digital company" hard disk drive that is subjected to high-frequency repeatable runout, and the results are discussed in section 7.

Chapter 3

Control Architecture

The adaptive repetitive controller proposed in this work is implemented in a *plug-in* fashion, meaning that the repetitive compensator is used to augment an existing robustly stable closed loop system. To clarify this notion, we use a very common plant-controller interconnection shown in Fig. 3.1(top) as a running example. The blocks G and C_F in the figure respectively denote a linear time invariant (LTI) plant and a nominal LTI feedback compensator that is used to stabilize the plant. This nominal controller can be either continuous or discrete time, and it generally provides disturbance rejection across a broad frequency spectrum. On the other hand, the plug-in adaptive repetitive controller, denoted by C_A, is a non-linear discrete time system that provides specialized compensation for disturbances appearing at selective frequencies, which should be less than the *Nyquist* frequency of the repetitive controller. Since our design does not depend on whether the plant/nominal controller are continuous or discrete time, we assume that both G and C_A are discrete time systems.

We consider a general stochastic environment for the system by appending input disturbance w, output disturbance n, and contaminating measurement noise m to our framework. Generally, the nominal feedback controller is designed to compensate for these input and output noises. The periodic disturbance that should be compensated by the adaptive controller is denoted by r, and without loss of generality, we assume that it is applied to the plant output.

An important point to make here is that our plug-in controller design is not limited to this particular interconnection, and in general, it does not require any details about the individual components of the closed loop system and their interconnections. Rather, our design is based on an abstract LTI dynamics from the adaptive control (u_A) injection point to the *error* signal (e). A succinct representation of this framework is shown in Fig. 3.1(bottom). Indeed, our design is only based on the dynamics of R which is a discrete time transfer function from u_A to e. Returning to our running example, these blocks and signals shown in the abstract

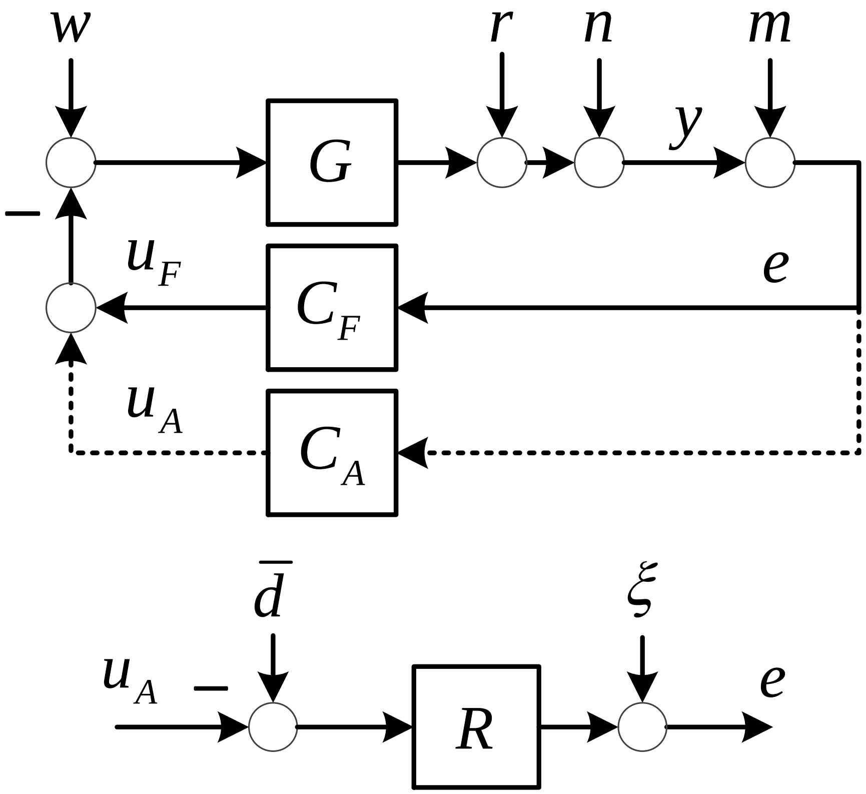

Figure 3.1: Top: Closed loop system augmented by a *plug-in* adaptive controller. Bottom: Succinct representation of the closed loop system.

form are defined by

$$\begin{aligned} S(z) &:= \frac{1}{1 + C_F(z)\,G(z)} \\ R(z) &:= G(z)\,S(z) \\ \xi_k &= R[w_k] + S[n_k + m_k] \cdot \end{aligned} \tag{3.1}$$

Here, the standard discrete-time z variable notation is used for the transfer functions, and the time functionality of the signals is shown by the step index k. For a single-input single-output transfer function $T(z)$ and an input sequence $i_k \in \mathbb{R}^m,\ m \geq 1$, the bracket notation $T[i_k]$ represents the time domain response of the system operating on each element of the input signal individually. For instance, the response of the transfer function $R(z)$ to the input disturbance $w(k)$ is represented by $R[w_k]$. The periodic disturbance in this block diagram is replaced by an equivalent disturbance, $\bar{d}$, which has the same effect as r on the error signal,

$$R\left[\bar{d}_k\right] = S[r_k] \cdot$$

From a control point of view, this replacement is admissible since the disturbance is bounded, periodic, and the system is linear; hence, it is possible to consider the periodic disturbance at any other point, or multiple points in the closed loop system.

3.1 Adaptive Control Synthesis

We aim to synthesis an adaptive control law for generating u_A in order to reduce the effect of disturbance $\bar{d}$ on the error signal e as is shown in Fig. 3.1(bottom). This objective can be ideally achieved when the control signal is equal to the disturbance $\bar{d}$. Since the disturbance is unknown, our control objective is equivalent to learning disturbance in an adaptive manner.

In a spectral analysis framework, we present the disturbance signal as a summation of sine and cosine functions with zero initial phases

$$\begin{aligned}\bar{d}_k &= \sum_{i=1}^{n} \left[a_i \left(\alpha_i \sin\left(\omega_i kT\right)\right) + b_i \left(\alpha_i \cos\left(\omega_i kT\right)\right)\right] \\ &= \theta^T \phi_k\end{aligned} \tag{3.2}$$

where ω_i's and n respectively denote the frequencies and number of those components that are desired to be removed from the spectrum of e, and T is the sampling time of the discrete time system. The vector of parameters, denoted by θ, and regressor vector signal at time step k, represented by ϕ_k, are defined as

$$\theta^T := \left[a_1, \cdots, a_n, b_1, \cdots, b_n\right] \tag{3.3}$$

$$\begin{aligned}\phi_k^T := &\left[\alpha_1 \sin\left(\omega_1 kT\right), \cdots, \alpha_n \sin\left(\omega_n kT\right),\right. \\ &\left.\alpha_1 \cos\left(\omega_1 kT\right), \cdots, \alpha_n \cos\left(\omega_n kT\right)\right]\cdot\end{aligned} \tag{3.4}$$

Here, the regressor ϕ_k is a known signal since the frequencies ω_i's are known and α_i's are positive weighting parameters that should be chosen by the designer. An explicit method for choosing these parameters is given later in the paper. Using the representation given in (3.2), learning the disturbance is equivalent to obtaining an estimation for the parameter vector θ, and constructing the control signal u_A at time step k as

$$u_{A,k} = \hat{\theta}_k^T \phi_k \tag{3.5}$$

where $\hat{\theta}_k$ denotes the estimated parameter vector at time step k. We use a modified filtered-x LMS algorithm that is presented in the following subsection to iteratively identify the parameter vector.

Parameter Adaptation Algorithm

We deploy a modified filtered-x LMS algorithm to adaptively identify the parameter vector θ by using the error signal as the feedback information. We represent the error signal as

$$\begin{aligned} e_k &= R\left[\bar{d}_k - u_{A,k}\right] + \xi_k = R\left[\phi_k^T\theta - u_{A,k}\right] + \xi_k \\ &= \underbrace{R\left[\phi_k\right]}_{\psi_k}{}^T\theta - R\left[u_{A,k}\right] - \underbrace{R[\phi_k]}_{\psi_k}{}^T\hat{\theta}_k + \underbrace{R[\phi_k]}_{\psi_k}{}^T\hat{\theta}_k + \xi_k \\ &= \psi_k^T\theta - \psi_k^T\hat{\theta}_k + \xi_k - R\left[u_{A,k}\right] + \psi_k^T\hat{\theta}_k \cdot \end{aligned} \tag{3.6}$$

We denote the signal $\psi_k := R[\phi_k] \in \mathbb{R}^{2n}$ as the *filtered regressor.* Suppose that the transfer function $R(z)$ is known; then, the filtered regressor is known as well and the error signal is an affine function of our estimated parameter vector, $\hat{\theta}_k$, if the last term on the right hand side of (3.6) is omitted. This suggests defining an *auxiliary error* signal

$$\begin{aligned} \bar{e}_k &:= e_k + R\left[u_{A,k}\right] - \psi_k^T\hat{\theta}_k \\ &= \psi_k^T\theta - \psi_k^T\hat{\theta}_k + \xi_k \,, \end{aligned} \tag{3.7}$$

which is explicitly an affine function of our estimated parameter vector. This *auxiliary error* signal is a good measure of the the original *error* signal because they are equivalent when the parameters converge. Suppose that the estimated parameter vector converges to $\hat{\theta}_{ss}$. The two signals will be equal because

$$\begin{aligned} \bar{e}_k - e_k &= R[\phi_k^T\hat{\theta}_{ss}] - \psi_k^T\hat{\theta}_{ss} \\ &= \psi_k^T\hat{\theta}_{ss} - \psi_k^T\hat{\theta}_{ss} = 0 \,. \end{aligned} \tag{3.8}$$

Accordingly, rather than minimizing the variance of the *error* signal e_k, we use a gradient-based algorithm to minimize the variance of *auxiliary error* signal $\bar{e}_k$.

We deploy the least mean square algorithm – the most widely used method in adaptive filtering – because it has low computational complexity, it converges in stationary environment, and under independence theory assumption – which is satisfied here – the parameters converge in the mean to the Wiener solution. Moreover, this algorithm is suitable for the specific application that we will be discussing in section 5 since it is shown that the LMS algorithm has stable behavior when implemented with finite-precision arithmetic. Let $\hat{\theta}_k^i$ represent the i-th element of $\hat{\theta}_k$. The LMS update equation enriched with a variable step size μ_k and a scheduling binary variable γ_k^i, that will be both discussed in the following subsections, is as follows:

$$\begin{bmatrix} \hat{\theta}_{k+1}^i \\ \hat{\theta}_{k+1}^{i+n} \end{bmatrix} = \begin{bmatrix} \hat{\theta}_k^i \\ \hat{\theta}_k^{i+n} \end{bmatrix} + \gamma_k^i \mu_k \begin{bmatrix} \psi_k^i \\ \psi_k^{i+n} \end{bmatrix} \bar{e}_k, \quad 1 \le i \le n \cdot \tag{3.9}$$

Variable Step Size

Although the coefficient vector on average converges to Wiener solution, the instantaneous deviation in the parameter vector, caused by the noise ξ_k, generate an *excess* mean squared error (MSE) appearing in the variance of $\bar{e}_k$. More important, this parameter oscillation prevents the *auxiliary error* signal from converging to the *error* signal, the equality that was shown in (3.8) under a steady state assumption.

We propose an adaptive law to adjust the step size based on an estimation of the total mean squared error. The key idea behind this scheme is that, as the estimated parameters get closer to the real ones, the step size becomes smaller and the parameters will be frozen in time when a certain desired performance (in terms of the mean squared error) is attained. This removes the excess error from the output and results in smaller steady state errors. However, in a practical situation, the system dynamics or disturbance $\bar{d}$ may be subjected to variations, and it is required that the step size activates the adaptation whenever the error becomes "large" due to these variations.

We use a moving average with a window width of h to estimate the *auxiliary error* power at time step k

$$V_k^h = V_{k-1}^h + \frac{1}{h}\left[(\bar{e}_k)^2 - (\bar{e}_{k-h+1})^2\right]. \tag{3.10}$$

For a given *desired MSE* value, say V^d, we define the step size law as

$$\bar{\mu}_k = \rho\left(V_k^h - V^d\right) \tag{3.11}$$

$$\mu_k = \begin{cases} min(\bar{\mu}_k, \mu_{max}) & \text{if} \quad (\bar{\mu}_k > 0 \ \wedge \ \mu_{k-1} > 0) \\ & \quad \vee(\bar{\mu}_k > \mu^{ub}) \\ 0 & \quad \text{otherwise} \end{cases} \tag{3.12}$$

Here, constant ρ is a positive scalar gain, and the variable $\bar{\mu}_k$ determines how far the current error power is from the desired value V^d. Since V_k^h is not exactly equal to the auxiliary error variance, the system may show *chattering* behavior around the switch line (i.e. $V_k^h = V^d$) if $\bar{\mu}_k$ is used directly as the step size. To avoid this, we add a hysteresis behavior to the step size, which is defined by (3.12). The logic condition represented in (3.12) define a *dead-band* $[V^d,\ V^{ub}]$ on the MSE surface, and represent a hysteresis behavior for the step size. That is, the adaptation is active as long as the estimated error power is above the *dead-band* ($V_k^h > V^{ub}$), and it is inactive whenever the error power falls behind the *dead-band* ($V_k^h < V^d$). Moreover, if the approximated error power enters the *dead-band* from the above, it stays active until it exits from the bottom (values smaller than V^d), and if the power error enters the *dead-band* from the bottom, it stays inactive as long it does not exceed the upper limit V^{ub}. To guarantee the convergence of the second moment of error, an upper limit μ_{max} is considered on the step size. It is well known that a sufficient condition for guaranteeing MSE convergence is to choose $\mu_{max} \leq \frac{2}{3\psi_k^T\psi_k}$ for all values of k **feuer1985convergence** A schematic for the step size hysteresis behavior is shown in Fig. 3.3.

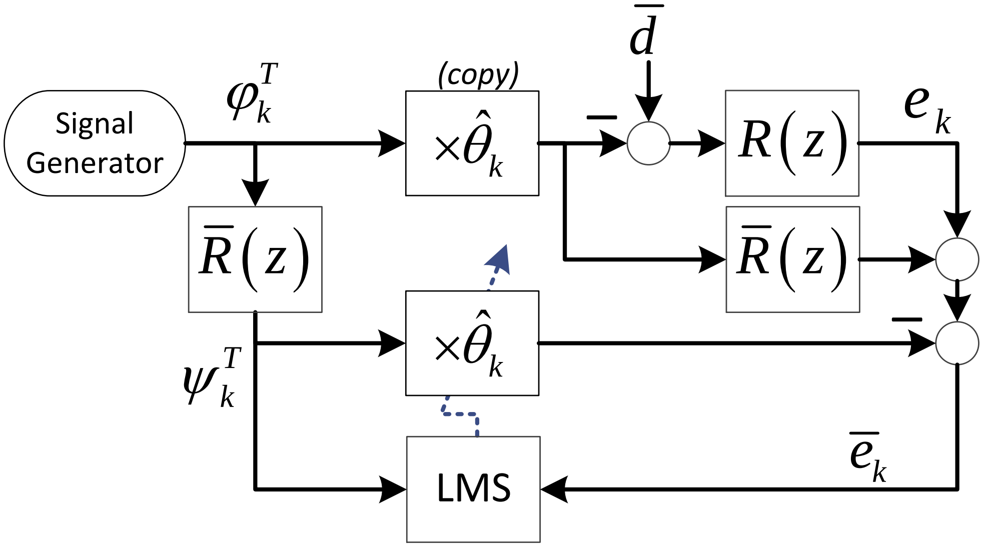

Figure 3.2: Adaptive repetitive controller block diagram.

Weighting Parameters, α_i

Since the transfer function $R(z)$ is LTI, the filtered regressor in a steady state, ψ_k, can be simply calculated by knowing the frequency response of $R(z)$

$$\begin{gathered}\begin{bmatrix}\psi_{i,k}\\ \psi_{i+n,k}\end{bmatrix} = m_i\alpha_i \begin{bmatrix}\sin\left(\omega_i kT+\delta_i\right)\\ \cos\left(\omega_i kT+\delta_i\right)\end{bmatrix}, i \in \{1,\cdots,n\}\\ m_i = |R\left(e^{j\omega_i T}\right)|, \quad \delta_i = \angle R\left(e^{j\omega_i T}\right).\end{gathered} \tag{3.13}$$

Here, $|x|$ and $\angle x$ denote the magnitude and phase of a complex number x. It is well known that the convergence rate of the LMS algorithm depends on the eigenvalue spread of the regressor correlation matrix **ungerboeck1972theory** This fact suggests that the values of α_i's in (3.3) should be chosen such that the amplitude of all sinusoidal elements in (3.13) are equal – i.e. $\alpha_i = \frac{c}{m_i}$, where c is a constant scalar. This constant can be chosen to be one, because the update equation (3.9) is related to the term ρc, and accordingly, it is possible to fix one of these two variables.

Scheduling Parameters, γ_k^i

In the lack of a priori knowledge about the parameter values, the transient error may be large if many parameters are being updated simultaneously. To solve this issue, the adaptation of different parameters can be scheduled in time by using γ_k^i, as a binary variable, that is one when the ith parameter should be updated and zero otherwise. An example for choosing these parameters is given in section (4).

Table 3.1: Adaptive Control Algorithm

1. Apply $u_{A,k}$ to the VCM
2. Calculate the auxiliary error: $\bar{e}_k = e_k - \tilde{e}_k$
3. Calculate the step size

$$V_k^h = V_{k-1}^h + \frac{1}{h}\bar{e}_k^2 - \frac{1}{h}\bar{e}_{k-h+1}^2$$

$$\bar{\mu}_k = \rho\left(V_k^h - V^d\right)$$

$$\mu_k = \begin{cases} min(\bar{\mu}_k, \mu_{max}) & \text{if } \ (\bar{\mu}_k > 0 \ \wedge \ \mu_{k-1} > 0) \\ & \vee(V_k^h > V^{ub}) \\ 0 & \text{otherwise} \end{cases}$$

4. Update the estimated parameters $\hat{\theta}_{k+1}^i = \hat{\theta}_k^i + \gamma_k^i \mu_k \psi_k^i \bar{e}_k$
5. Calculate $u_{A,k+1} = \phi_{k+1}^T \hat{\theta}_{k+1}$
6. Calculate $\tilde{e}_{k+1} = R\left[u_{A,k+1}\right] - \psi_{k+1}^T \hat{\theta}_{k+1}$

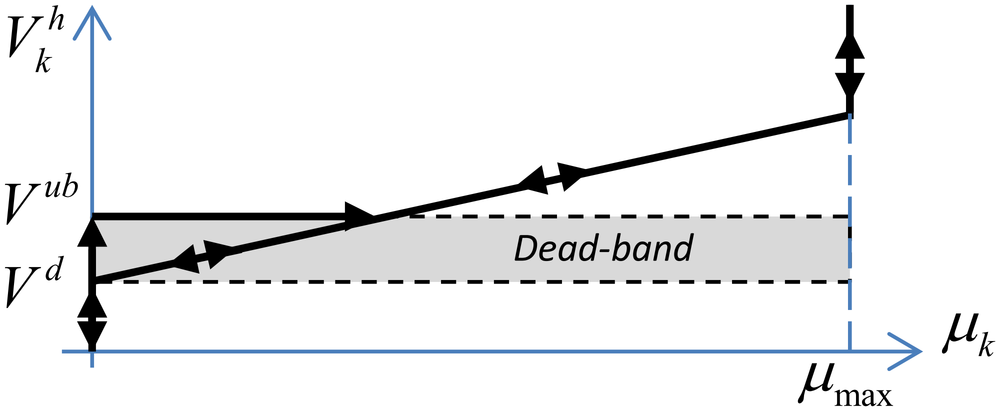

Figure 3.3: Adaptive variable step size with hysteresis behavior.

Uncertain System Dynamics

The adaptive controller proposed in section 3.1 uses the closed loop system transfer function, $R(z)$, to construct the auxiliary error as shown in (3.7). In practical applications, an exact dynamics of the system, in general, is not available because of the uncertainties due to unmodeled dynamics, variation over time and temperature, etc. Hence, we use a nominal dynamics model, say $\bar{R}(z)$, in all occurrences of the actual transfer function $R(z)$ throughout the algorithm

$$\bar{\psi}_k := \bar{R}[\phi_k] \tag{3.14}$$

$$\bar{e}_k := e_k + \bar{R}\left[u_{A,k}\right] - \bar{\psi}_k^T \hat{\theta}_k \tag{3.15}$$

$$\hat{\theta}_{k+1} = \hat{\theta}_k + \gamma_k \mu_k \bar{\psi}_k \bar{e}_k. \tag{3.16}$$

The explicit algorithm iteration is given in Table 3.1, and its block diagram is illustrated in Fig. 3.2.

The robustness and performance of the MFX-LMS algorithm depends on the mismatch between the model used in the secondary path, $\bar{R}\left(z\right)$, and the actual transfer function, $R\left(z\right)$. When the step size is small, a sufficient condition for robustness to the secondary path modeling mismatch is that the phase difference between these two systems should be less than 90 degrees **lopes2004behavior** – i.e. Real $\big(\bar{R}\left(e^{j\omega}\right)\big) / R\left(e^{j\omega}\right)\big)\big) > 0$.

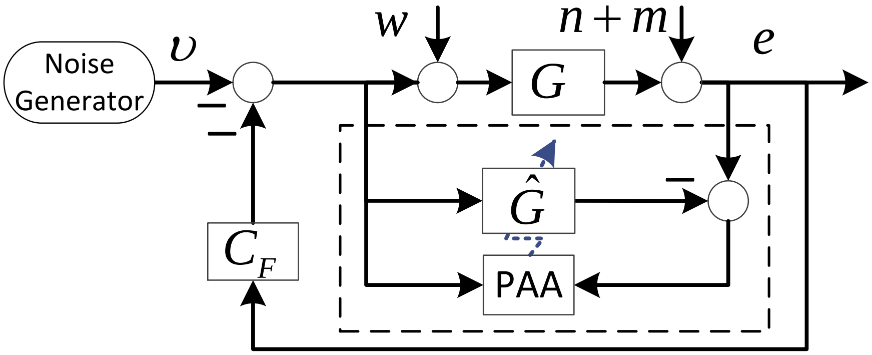

Figure 4.1: something

Chapter 4

Online Secondary Path Modeling

This section proposes an adaptive secondary path modeling architecture to stabilize the adaptive repetitive controller in case the phase mismatch between $R(z)$ and $\bar{R}(z)$ does not satisfy the aforementioned "90 degrees" criteria. Unlike the previous section, that we ignored the internal structure of the closed loop dynamics $R(z)$, we use our knowledge about the nominal feedback controller and the internal interconnections to identify the uncertain plant. This is an important distinction between this architecture and previous works **eriksson1989use**; **akhtar2006new** in which, modeling of the closed loop system $R\left(z\right)$ is studied. The order of the closed loop dynamics is equal to the summation of plant and controller orders when no pole-zero-cancellation occurs. Hence, the number of parameters required to identify the plant is less than the closed loop system in general.

We first consider a simple system identification case that is depicted in Fig. 4.1. Let $G(q^{-1})$ be a finite dimension transfer function of G represented by the *one step delay operator* q^{-1}

$$G\left(q^{-1}\right) = \frac{B\left(q^{-1}\right)}{A\left(q^{-1}\right)} = \frac{b_0 + b_1 q^{-1} + \cdots + b_{n_g} q^{-n_g}}{1 + a_1 q^{-1} + \cdots + a_{n_g} q^{-n_g}}.$$

In the figure, $\hat{G}\left(z\right)$ is the estimated plant model and **PAA** denotes the parameter adaptation

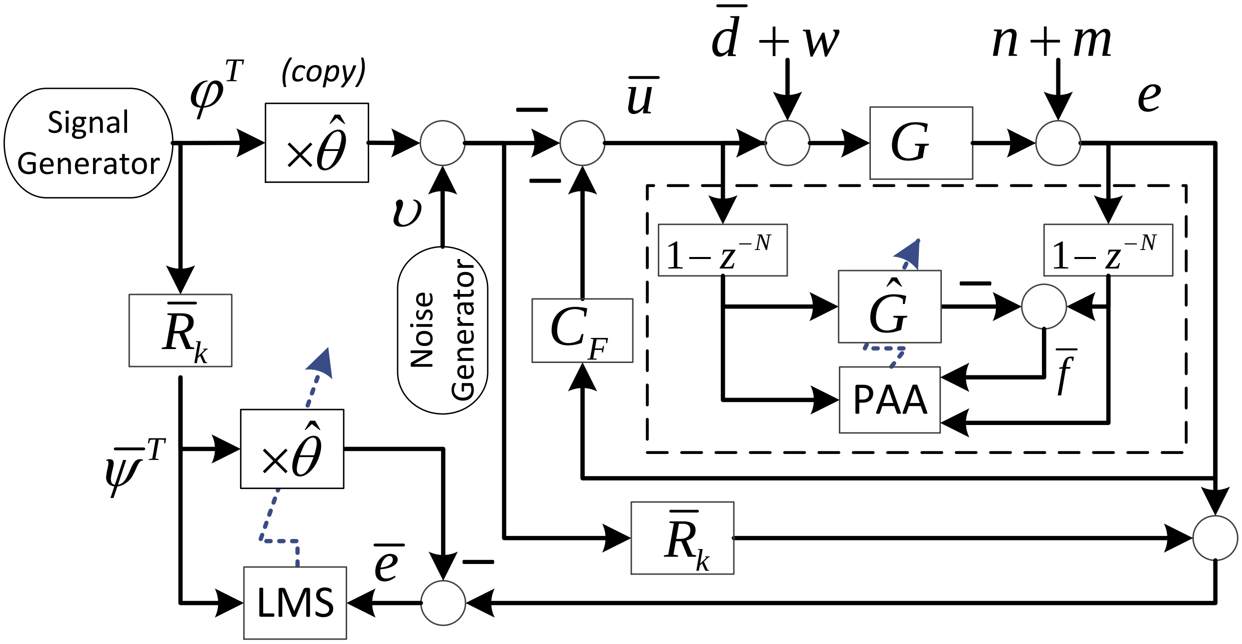

Figure 4.2: Adaptive repetitive controller with secondary path modeling (dashed box)

algorithm. A great deal of research effort has been focused on the design of **PAA** based on the characteristics of input and output noises **regalia1994adaptive**; **ljung1983theory** We will not focus on the design of **PAA**, and rather, make the following assumption.

Assumption 1 *The parameter adaptation algorithm in Fig. 4.1 is converging and an adequate* tap-length *is chosen such that the closed loop system*

$$\bar{R}_k(z) = \frac{\hat{G}_k(z)}{1 + C_F(z)\hat{G}_k(z)}. \tag{4.1}$$

satisfies

$$\lim_{k\to\infty}\left(\max_{0\le\omega\le\pi} \angle\left(\frac{R(e^{j\omega})}{\bar{R}_k(e^{j\omega})}\right)\right) < \frac{\pi}{2}.$$

Based on this assumption we advance to the problem of secondary path modeling for the adaptive repetitive controller. Figure 4.2 illustrates the proposed architecture, that uses the same **PAA** as Fig. 4.1. As shown in the figure, the secondary path of the repetitive controller is now using the estimated model given in (4.1).

Lemma 1 The existence of the repetitive controller in the loop cannot cause diverging behavior in the modeling path. *Since the adaptive control is known, we can think of the* *summation of the feedback control, adaptive control, and the injected noise, as a single excitation signal that is known to us. Therefore, if we ignore filters* $1 - z^{-N}$ *and signal* $\bar{d}$*, the estimated plant parameters in Fig. 4.2 converge if and only if they converge when there is no adaptive repetitive controller in the loop. However, the periodic disturbance* $\bar{d}_k$ *can cause biased parameters convergence, and it is required to be filtered out to validate this statement. The error signal in Fig. 4.2 is* $e_k = \theta_g^T \phi_{g,k} + d_{g,k} + \xi_{g,k}$ *where*

$$\theta_g^T := \left[a_1, \cdots, a_{n_g}, b_0, \cdots, b_{n_g}\right] \tag{4.2}$$

$$\phi_{g,k}^T := \left[e_{k-1}, \cdots, e_{k-n_g}, \bar{u}_k, \cdots, \bar{u}_{k-n_g}\right] \tag{4.3}$$

$$d_{g,k} := G\left[\bar{d}_k\right] \tag{4.4}$$

$$\xi_{g,k} := G\left[w_k\right] + n_k + m_k. \tag{4.5}$$

Note that $d_{g,k}$ *is periodic in steady state since* $\bar{d}_k$ *is periodic and* $G\left(q^{-1}\right)$ *is a LTI system. In the remaining, we ignore the transient state and assume that* $d_{g,k}$ *is periodic. This assumption leads us to filter both* e_k *and* $\phi_{g,k}$ *through* $1 - q^{-N}$ *and define*

$$\begin{aligned} \bar{f}_k : &= \left(e_k - e_{k-N}\right) - \hat{\theta}_{g,k}\left(\phi_{g,k} - \phi_{g,k-N}\right) \\ &= \tilde{\theta}_{g,k}^T \tilde{\phi}_{g,k} + \tilde{\xi}_{g,k} \end{aligned} \tag{4.6}$$

where $\tilde{\phi}_{g,k} = \phi_{g,k} - \phi_{g,k-N}$*,* $\tilde{\theta}_{g,k} = \theta_g - \hat{\theta}_{g,k}$ *and* $\tilde{\xi}_{g,k} = \xi_{g,k} - \xi_{g,k-N}$*. Note that* (4.6) *gives a suitable representation of an* error *variable to be used in an adaptive algorithm (c.f.* (3.7)*).*

Corollary 1 The adaptive repetitive controller parameters converge eventually. *Based on the separation property given in the previous lemma, and our assumption, the phase mismatch between* $\bar{R}_k(z)$ *and* $R(z)$ *can be decreased to less than* 90 *degrees if an adequate tap-length is chosen for the modeling path. Accordingly, the convergence criteria for the adaptive repetitive controller will be satisfied. On the other hand, the excitation noise* υ_k*, is statistically independent of the regressor signal to the adaptive repetitive controller, and it cannot cause unbiased parameter convergence in the repetitive controller.*

An important point to make is that, the adaptive control path does not converge as long as the updated transfer function $\bar{R}\left(z\right)$ has more than 90 degrees phase error relative to the actual transfer function $R\left(z\right)$. As a result when the initial parameters of $\hat{G}$ are not accurate, we expect that the adaptive control parameters diverge quickly. Although the adaptive controller becomes stable eventually – once the plant parameters get close enough (in terms of phase error) – this behavior is not desirable in many applications since the transient error may be very large, and accordingly the adaptive control requires a long time to recover. We thus suggest using an *initialization* period prior to the simultaneous adaptation, in which the adaptive controller is inactive till the secondary path modeling parameters converge.

Remark 1 *In many applications the plant dynamics is well known at a frequency range, say* Ω*. If such information is available, the frequency components of disturbance can be* *categorized into two sets $I_1 = \{i|\omega_i \in \Omega\}$ and $I_2 = \{i|\omega_i \notin \Omega\}$. The adaptation of parameters $\hat{\theta}^i, i \in I_1$ and plant parameters can be done simultaneously since the plant model is exact at that region. This can be done by choosing*

$$\gamma_k^i = \begin{cases} \mathit{0} & i \in I_2 \text{ and } k \leq P \\ \mathit{1} & \text{otherwise} \end{cases} \tag{4.7}$$

where P denotes the length of initialization period required for secondary path modeling.

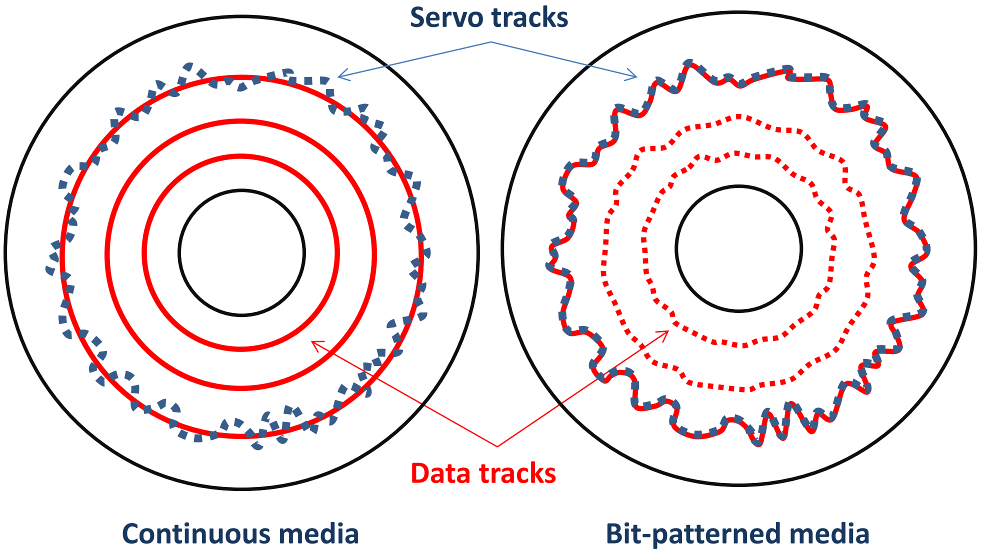

Figure 5.1: Schematic of servo tracks (dotted lines) and data tracks (solid lines) in conventional and bit-patterned media HDDs

Chapter 5

Application to Bit Patterned Media Recording

The HDD servo mechanism, operating in track-follow mode, is a nano-positioning system that controls the read/write head to follow a predetermined data track profile. This system is usually subjected to both repeatable (periodic) and non-repeatable (random) disturbances that are due to the imperfection in both fabrication and assembly processes, as well as windage induced and external vibrations, etc. As discussed in the introduction, the control strategy in conventional HDDs, that use continuous media, is distinct from BPMR. In the former technology, data is written on concentric circular tracks, whereas in BPMR, data should be written on data tracks with unknown shapes, which are created by lithography on the disk. A schematic of the ideal trajectory for these two types of magnetic recording is shown in Fig. 5.1. In the figure, servo tracks determine the desired trajectories to be tracked in BPMR, and their deviation from an ideal circular shape is called *repeatable runout* (RRO). Challenges in regards to control design for BPMR mainly arise due to the following RRO specifications:

- the RRO profile is unknown and its frequency spectrum spreads beyond the bandwidth of the servo system. Therefore, the tracking error will be amplified by the feedback controller at high frequencies.

- the RRO spectrum contains many harmonics – approximately 200 in current HDDs – of the spindle frequency that should be attenuated. In general, this requires computationally intensive control methods.

Accordingly, the servo control methodologies used for conventional drives **kempf1993comparison**; **sacks1995advanced**; **wu2006repeatable**; **chen2006iterative** cannot be applied to BPMR.

Fig. 3.1 (top) can be adopted to abstract the block diagram of a single stage HDD servo system. The blocks G and C_F refer to a voice coil motor (VCM) and the nominal feedback controller respectively. The signals w, r, n and m denote the airflow disturbance known as *windage*, *repeatable runout* (RRO), *non-repeatable runout* (NRRO) and *measurement noise* respectively. The actual *position error signal* (PES) and measured PES are respectively referred by y and e in the figure. In the remaining, the term "PES" is used for referring to the measured PES signal e.

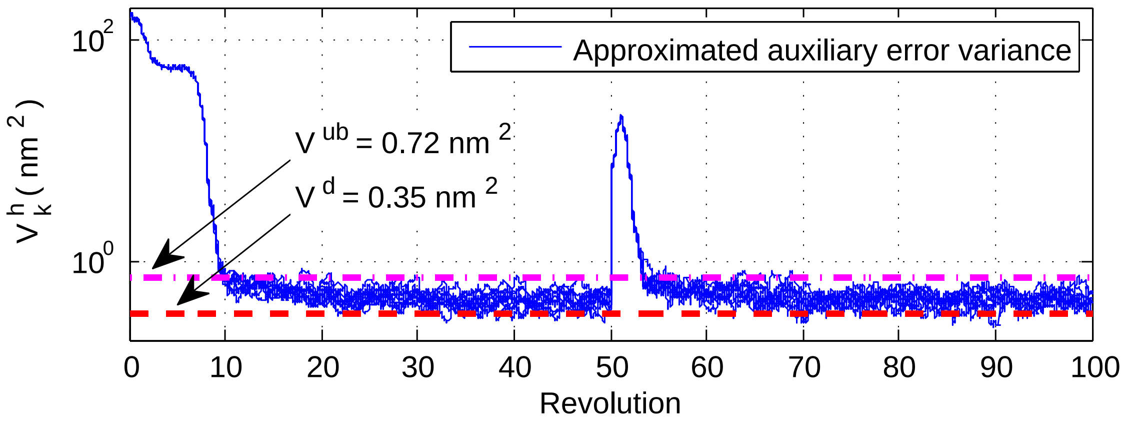

Figure 6.1: Approximated auxiliary error variance for 10 different plants and two different RRO profiles (simulation)

Chapter 6

Simulation Setup

Since BPMR HDDs have not been released yet, we do a realistic simulation study to validate the effectiveness of our proposed algorithm for this type of hard disk drives. We expect that the actuator and contaminating noises will be similar in both types of magnetic recording, and accordingly, model the dynamics (i.e. G and C_F) and noises (i.e. w, n and m) of the system based on a set of measurements on a conventional 2.5-inch hard disk drive that is provided by *HGST, a Western Digital company*. We use the RRO data from a prototype BPMR hard disk drive that is provided by *Seagate Technology* to create a realistic RRO profile for our simulations. The proposed control algorithm is implemented in MATLAB, and simulation results for 10 uncertain plant models and 100 revolutions of the disk are obtained. There are 400 servo sectors on each track that provides position samples, and the control rate is equal to the sampling rate. Therefore, the total number of time steps in our simulation is $40'000$, and the error spectrum can contain up to 200 harmonics of the spinning frequency.

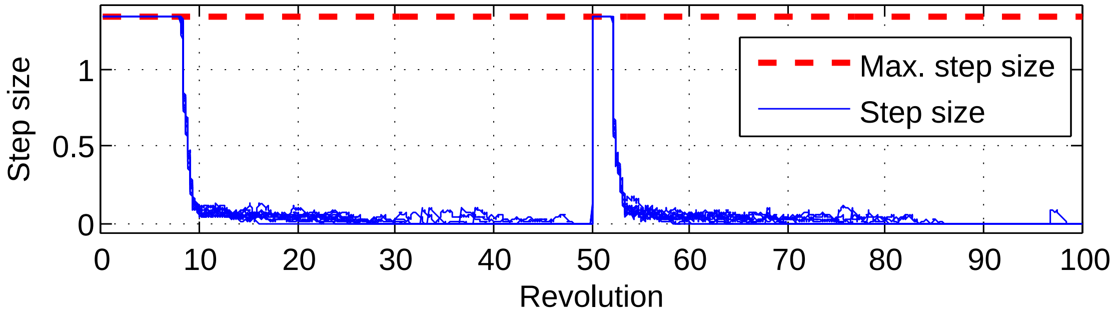

Figure 6.2: Variable step size for 10 different plants and two different RRO profiles(simulation)

Simulation Results

The *baseline error variance*, attained by perfect tracking, is $0.25nm^2$ in our simulations. We chose a desired error variance, $V^d := 0.35\ nm^2$, slightly larger than this limit of performance. Approximated *auxiliary error* variance, V_k^h, and variable step size, μ_k, for the 10 uncertain systems are shown in Fig. 6.1 and Fig. 6.2. The RRO profile in the first and last 50 revolutions, respectively, belong to a *mid-diameter* (MD) and an *outer-diameter* (OD) track on the disk. As can be seen in these two figures, the *auxiliary error* variance enters the *dead-band* in less than 10 revolutions. Figure 3.3 shows that he adaptation is automatically stopped when V_k^h reaches the desired value, and this along with (3.8) implies that the *PES* variance is in the same level. In order to decrease the transient error, the adaptation for harmonics $1-80$ starts first, and after 7 revolutions the adaptation for harmonics $81-190$ begins (c.f. Fig.6.1).

The spectrum of the steady state error is shown in Fig. 6.4. Since we chose a desired performance, V^d, larger than the baseline error variance, there is some leftover mostly located between harmonic 1 and 40.

Lastly, we simulated the proposed secondary path modeling scheme to identify the plant dynamics of VCM in an online fashion. Fig. 6.3 shows the phase mismatch between a fairly inaccurate nominal closed loop plant model and the actual one, which exceeds 90 degrees (in magnitude) at high frequencies. In order to identify the plant parameters, an extended recursive least squares algorithm with the proposed filtering method was implemented. Figure 6.3 shows that the identified system phase matches the actual plant very well, guaranteeing the adaptive repetitive controller convergence.

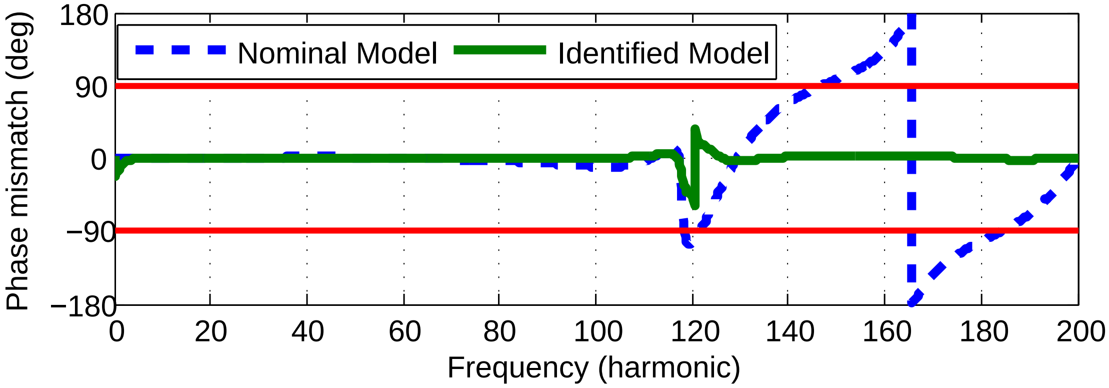

Figure 6.3: Phase mismatch between the nominal plant models and actual plant (simulation)

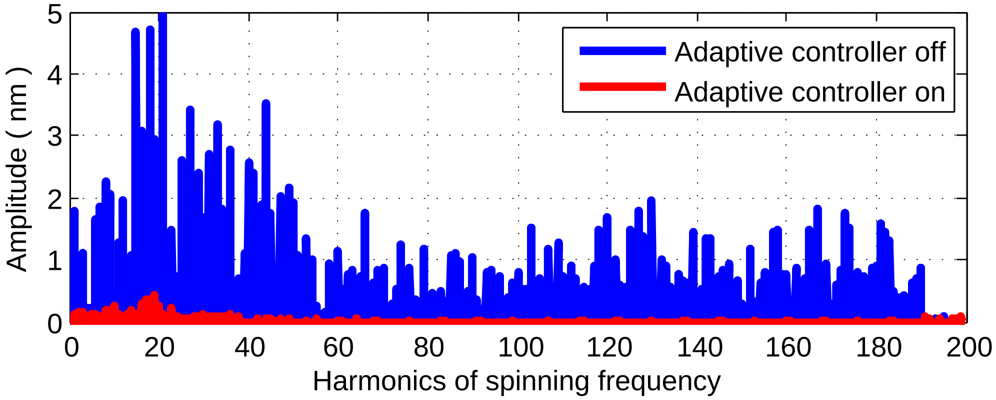

Figure 6.4: Closed loop error spectrum in BPMR simulation

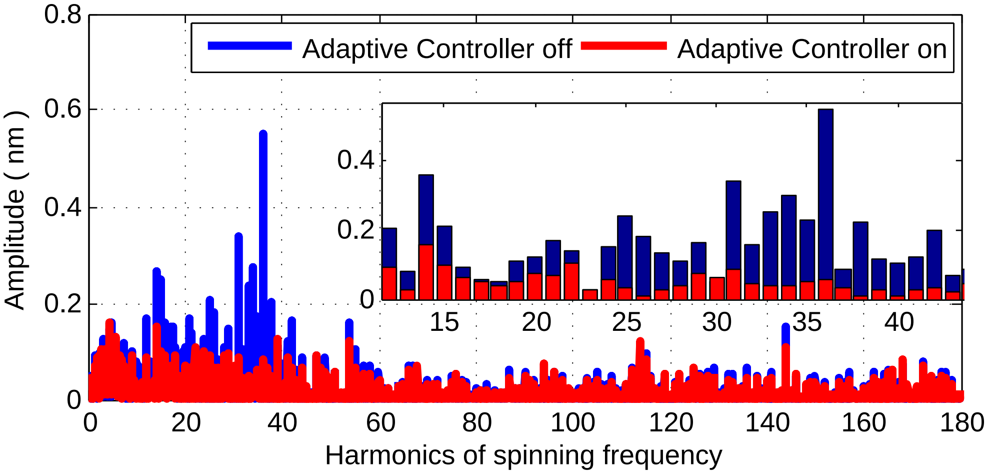

Figure 7.1: Closed loop error spectrum in experiments on a conventional HDD

Chapter 7

Implementation Results

The proposed control and modeling algorithms are tested on a conventional 2.5” HDD to validate the presented analytical and simulation results. Modifications were made in the HDD firmware to send out the PES to an OMAP-L138 development kit that runs our adaptive control algorithm, and applies the control signal to the VCM through a digital to analog converter (DAC).

Since the internal HDD controller already compensates the low frequency part of the RRO, the major observed RRO peaks are located in mid frequency range. Considering the limited available control in our setup, we focus on attenuating the 31 major PES frequency components that are between the 12^{th} and 42^{nd} harmonics of the spinning frequency. In our experimental implementation, a simple second order model is identified to approximate the actual transfer function $R(z)$ in the frequency range of interest. It is validated by an accurate frequency analyzer that this model has less than 90 degrees phase mismatch with $R(z)$ in that frequency interval.

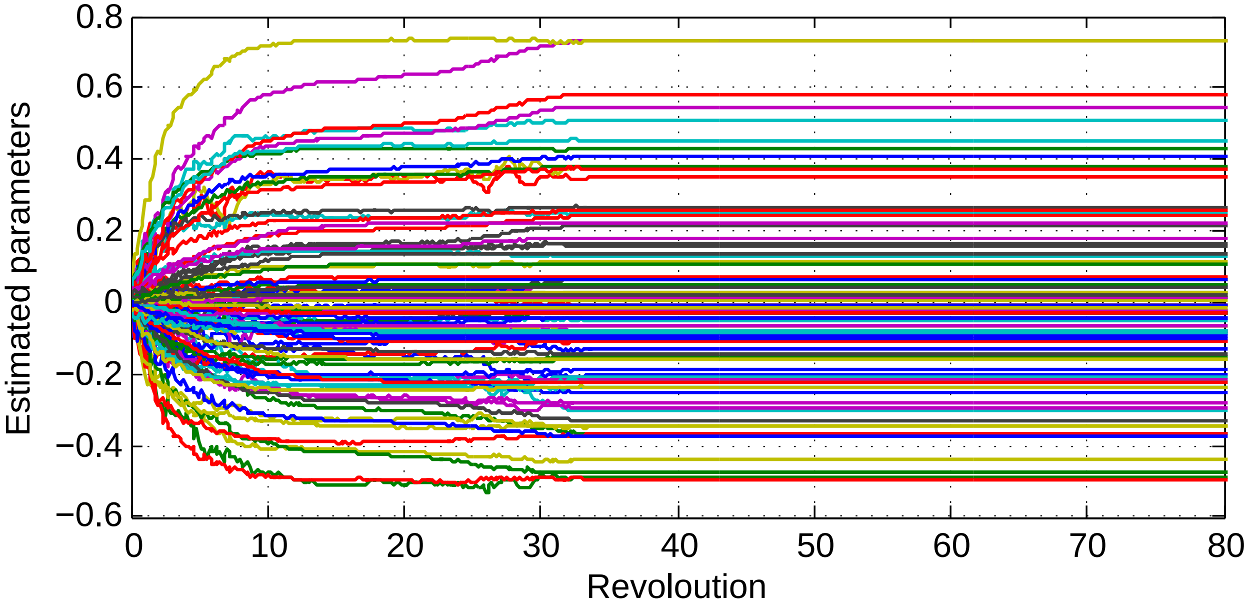

Figure 7.2: EVOLUTION OF THE ESTIMATED PARAMETERS

Fig. 7.1 compares the error spectrum before and after applying the adaptive control. The mid frequency range of spectrum is magnified in a box inside the figure, which demonstrates that utilizing the adaptive controller results in a considerable attenuation of the RRO frequency components in the excitation interval. The PES variance was computed for both cases and the results show 26.63% improvement.

Fig. 7.2 shows the evolution of the estimated parameters. As can be seen in the figure, most of the estimated parameters have converged after 16 revolutions of the disk. However to reach the "desired" performance level, that is $V^d := 1.36nm^2$ in this particular drive, the adaption remains active for approximately 26 revolutions with a small step size to "fine-tune" the parameters. This becomes more clear by considering the "approximated auxiliary error variance" V_k^h, and the variable step size μ_k, which are both plotted in Fig. 7.3. As shown in the figure, the adaptive algorithm remains active until V_k^h meets the "desired" performance level $(i.e.V_k^h \leq V^d)$ and the adaption stops automatically by setting $\mu_k = 0$.

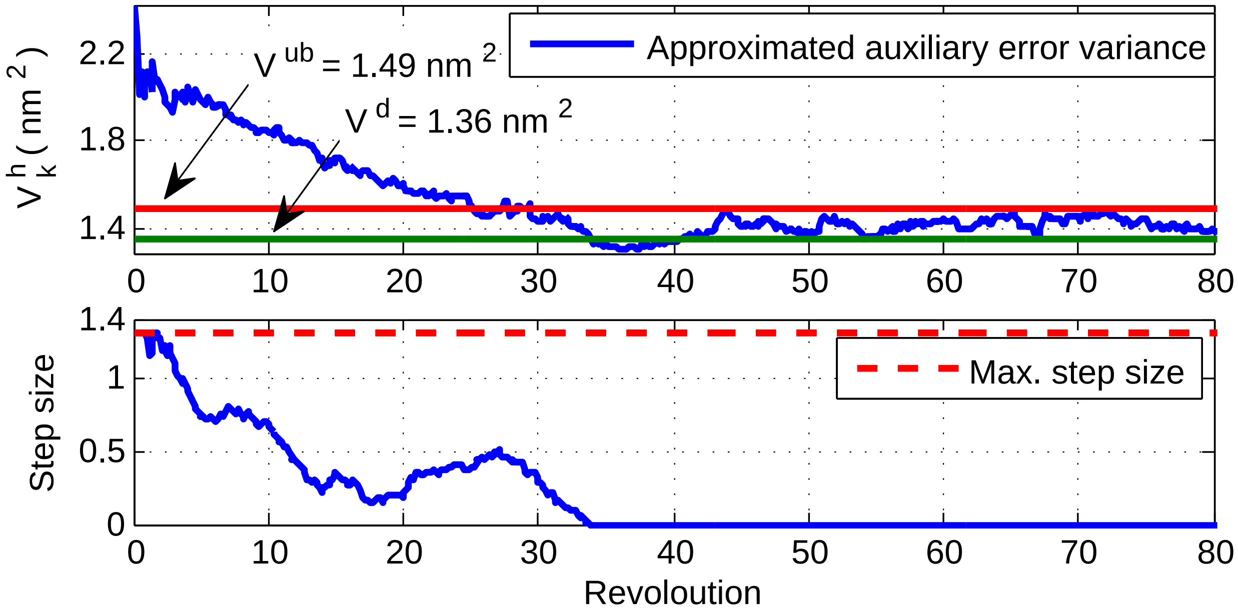

Figure 7.3: top: Approximated auxiliary error variance. bottom: variable step size. (experiments)

Chapter 8

Conclusions

An adaptive repetitive controller based on a modified filtered-x LMS algorithm is proposed in this paper. We have illustrated how this method can be effectively used for rejecting periodic disturbances or selective frequency attenuation by using a deterministic regressor. Furthermore, an adaptive secondary path modeling scheme is proposed to identify and compensate the dynamics mismatch between the actual and nominal closed loop systems. This online modeling scheme prevents divergence of the repetitive controller at the frequencies that the phase mismatch exceeds 90 degrees.

The application of the proposed algorithm in bit-patterned media recording is presented and illustrative simulation and implementation results are provided to show the effectiveness of this method.