Proceeding of the 2004 American Control Conference
Boston, Massachusetts June 30 - July 2, 2004

ADAPTIVE-Q WITH LQG STABILIZING FEEDBACK AND REAL TIME COMPUTATION FOR DISK DRIVE SERVO CONTROL

K. Krishnamoorthy* and Tsu-Chin Tsao**
Department of Mechanical and Aerospace Engineering
University of California
Los Angeles, CA 90095, USA

***Abstract*— A method of stabilizing feedback control, known as adaptive-Q control is applied to a LQG disk drive control system to improve its disturbance rejection property. This approach follows the well known result that all stabilizing controllers for a plant can be synthesized by conveniently parametrized augmentations to a nominal controller. The augmentation is parametrized by an arbitrary stable filter Q which is adapted online. The Q filter is restricted to a FIR stable filter which minimizes the H_2 norm of the transfer function from the disturbance to the track following error. Simulations and experimental results show a 15% improvement in the achievable TMR (track misregistration). The Q filter is adapted using a recursive least squares numercially stable adaptive array algorithm. Two algorithms based on inverse QR factorization are considered for implementation. The computation issues involved in updating the Q filter every servo cycle are discussed.**

I. INTRODUCTION

A. Disk Drive Track Following

The fundamental measure of disk drive tracking performance called *track misregistration* (TMR) is the deviation of the center of the read/write head from the center of the data track [1]. TMR is usually reported as a variance parameter per sample of the disk servo e.g., 10 nanometers 3-sigma (3σ). For acceptable performance of the servo system the 3σ value of the TMR must be less than 10% of the track width. A major function of the servo controller for the read-write head is to maintain the head accurately on a selected track of the rotating disk. There are two primary sources of disturbances in the hard disk servo system. The first is *repeatable runout* (RRO) which is a periodic disturbance that stays locked to the disk rotation (both frequency and phase). This disturbance is mainly due to imperfect or eccentric tracks. The second is *nonrepeatable runout* (NRRO) which is the cumulative result of disk drive vibrations, electrical noise in the circuits and the measurement channels. While synchronous excitations may be large, standard practise in the disk drive industry includes using feedforward cancellers to reduce their effect [2].

B. Add-on Adaptive Controller

An add-on adaptive compensator for disk drives based on plant inversion is discussed in [3]. A feedforward scheme to attenuate external vibrations is presented in [4]. This approach requires an accelerometer mounted on the base casting of the disk drive and may not be practically viable. The Adaptive-Q approach has been proposed for vibration suppression in rotor/magnetic bearing systems [5]. The purpose of this paper is to present and evaluate the adaptive-Q approach for rejecting disturbances in a disk drive. The disk drive open loop plant is first stabilized using a fixed LQG compensator. The rest of this paper is organized as follows: Section II details the modeling of the disk drive used for control design and analysis. In section III we derive the class of all stabilizing controllers parametrized by a free parameter Q. Section IV introduces the adaptive-Q algorithm. In section V we derice the RLS update equations for the proposed algorithm. We present simulation and experimental results section VI. We also discuss the computation issues involved in the adaptive-Q filter update followed by conclusions in section VII.

*krishnak@ucla.edu
**ttsao@seas.ucla.edu

II. SYSTEM IDENTIFICATION

A 2 platter (10 GB/platter), 25 kTPI (tracks per inch), 7200 rpm disk drive system was used for experimental verification of the proposed control scheme. The disk drive model was identified in the discrete domain. Since the openloop disk drive plant is marginally stable, a stabilizing *PD* controller was first designed to close the loop (see fig 1). The disk drive sensor output is available at the rate of 9.36 kHz and this is the sampling frequency T used for both system identification and control. We have the control law,

$$u_k = K_p e_k + K_d(e_k - e_{k-1})$$

where K_p and K_d were chosen by trial and error to provide closed loop stability. Then a swept-sine reference r with frequency content from 1 Hz to the Nyquist was used to excite the closed loop plant G. From fig. 1 we have,

$$L = \frac{GC}{1+GC} \quad \Rightarrow \quad G = \frac{L}{(1-L)C}$$

From r and y we get the closed loop frequency response $L(e^{j\omega T})$. Using the above relation we calculate the open loop frequency response $G(e^{j\omega T})$. A fifth order state-space model of the system was obtained by curve-fitting this frequency response (see fig. 2). We could also obtain

0-7803-8335-4/04/$17.00 ©2004 AACC

$G(e^{j\omega T})$ directly from u and y. It turns out that the two procedures lead to almost identical frequency response plots for the open loop plant G.

III. STABLE Q PARAMETRIZATION

The Disk Drive Assembly is represented by the discrete time state-space model [6] driven by output disturbance signal w

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k \\ y_k &= Cx_k + w_k \end{aligned} \quad (1)$$

The LQG feedback control is represented as [6]

$$\begin{aligned} u_k &= F\hat{x}_k + s_k \\ \hat{x}_{k+1} &= A\hat{x}_k + Bu_k - Hr_k \\ r_k &= y_k - C\hat{x}_k \end{aligned} \quad (2)$$

Let the adaptive filter Q have the time varying dynamic representaion

$$\begin{aligned} z_{k+1} &= A_q z_k + B_q r_k \\ s_k &= C_{q,k} z_k \end{aligned} \quad (3)$$

If we restrict Q to a FIR filter form then we would have

$$\begin{aligned} A_q &= \begin{bmatrix} 0 & & & & \\ 1 & 0 & & & \\ 0 & 1 & 0 & & \\ \vdots & \ddots & \ddots & \ddots & \\ 0 & \dots & 0 & 1 & 0 \end{bmatrix} \\ B_q &= \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix}^T \\ C_{q,k} &= \begin{bmatrix} q_k^0 & q_k^1 & \dots & q_k^{L-1} \end{bmatrix} \end{aligned} \quad (4)$$

We combine the plant with the observer and Q filter dynamics to form the augmented state

$$X_k = \begin{pmatrix} x_k \\ z_k \\ x_k - \hat{x}_k \end{pmatrix}$$

Now we have the augmented system dynamics given by

$$\begin{aligned} X_{k+1} &= \mathscr{A}_k X_k + \mathscr{B} w_k \\ y_k &= \mathscr{C} X_k + w_k \end{aligned} \quad (5)$$

where

$$\begin{aligned} \mathscr{A}_k &= \begin{bmatrix} A+BF & BC_{q,k} & -BF \\ 0 & A_q & -B_qC \\ 0 & 0 & A+HC \end{bmatrix} \\ \mathscr{B} &= \begin{bmatrix} 0 & -B_q & H \end{bmatrix}^T \\ \mathscr{C} &= \begin{bmatrix} C & 0 & 0 \end{bmatrix} \end{aligned}$$

For output disturbance rejection we look at the transfer function from w to y given by

$$y_k = \left(I + \mathscr{C}(zI - \mathscr{A}_k)^{-1}\mathscr{B}\right) w_k \quad (6)$$

Our objective is to make this transfer function *affine* in Q. For conciseness we define the compensator system matrix $A_f = A + BF$ and the observer system matrix $A_c = A + HC$. Now we have from the definition of $\mathscr{A}_k$

$$(zI - \mathscr{A}_k) = \begin{bmatrix} zI - A_f & -BC_{q,k} & BF \\ 0 & zI - A_q & B_qC \\ 0 & 0 & zI - A_c \end{bmatrix}$$

This being a upper triangular matrix one can easily compute the inverse under the assumptions that the diagonal terms are invertible. We have this from the matrices A_f, A_q and A_c being *Hurwitz*.

$$(zI - \mathscr{A}_k)^{-1} = \begin{bmatrix} (zI - A_f)^{-1} & \alpha & \beta \\ 0 & (zI - A_q)^{-1} & \times \\ 0 & 0 & (zI - A_c)^{-1} \end{bmatrix}$$

where

$$\begin{aligned} \alpha &= (zI - A_f)^{-1} BC_{q,k} (zI - A_q)^{-1} \\ \beta &= (zI - A_f)^{-1} B(F + C_{q,k}(zI - A_q)^{-1} B_qC)(zI - A_c)^{-1} \end{aligned}$$

We introduce the transfer functions

$$\begin{aligned} T_{11} &= I - C(zI - A_f)^{-1} BF(zI - A_c)^{-1} H \\ T_{12} &= C(zI - A_f)^{-1} B \\ T_{21} &= I + C(zI - A_c)^{-1} H \\ Q &= C_{q,k}(zI - A_q)^{-1} B_q \end{aligned} \quad (7)$$

We substitute for $(zI - \mathscr{A}_k)^{-1}$ in (6) and after some manipulations arrive at

$$\begin{aligned} I + \mathscr{C}(zI - \mathscr{A}_k)^{-1}\mathscr{B} &= I - C\alpha B_q + C\beta H \\ &= I - C(zI - A_f)^{-1} BQ \\ &- C(zI - A_f)^{-1} BF(zI - A_c)^{-1} H \\ &- C(zI - A_f)^{-1} BQC(zI - A_c)^{-1} H \\ &= T_{11} - T_{12} Q T_{21} \end{aligned}$$

which gives us the well known result characterizing the class of all stabilizing controllers in terms of a stable transfer function $Q \in RH_\infty$,

$$y_k = (T_{11} - T_{12} Q T_{21}) w_k \quad (8)$$

The Q filter is adapted online so as to minimize y in a least mean squares sense as shown in Fig. 4. In the LQG framework, the input to the Q filter is the estimation residual r and the output s is summed with the state feedback to generate the control signal u (see fig. 3). Also it is noted that the LQG control system is optimal if we have accurate plant and disturbance models. In this ideal situation, the adaptive filter should remain at zero ($Q \equiv 0$) because it cannot improve upon the optimal !

IV. ADAPTIVE-Q FILTER

The nominal LQG controller ($Q \equiv 0$) provides stability but does not attain satisfactory disturbance rejection. The Q filter is introduced to enhance the disturbance rejection whilst still maintaining closed loop system stability. Hence we restrict the filter to an FIR form which gaurantees stability *apriori* provided the filter coefficients remain bounded.

From fig. 4 we see that the adaptive filter output s goes through the block T_{12} before it can affect the disturbance response. In adaptive filtering terminology, T_{12} is referred to as the secondary path [7]. It needs to be modeled offline and used to filter the input to the adaptive filter update. Hence a Filtered-X RLS adaptive array algorithm [7] is used to update the coefficients of Q. The transfer function from s to y given by $T_{12} = C(zI - A - BF)^{-1}B$ is readily available given the plant model G and the LQG state feedback matrix F. The Q filter adaptation loop (see fig. 4) can be executed offline if we have the system models T_{11}, T_{12}, T_{21} and also a disturbance model to generate w. Since the disturbances in a disk drive have both repeatable and random content, it is better to adapt Q online. Also this eliminates the need to model T_{11}, T_{21}. In essence, we have an adaptive filter which minimizes the disturbance response despite having no *apriori* knowledge of its dynamics. Online adaptation requires the Q filter to be updated every servo cycle. Since the servo update rate in disk drives is around 10kHz, the RLS algorithm needs to be computationally efficient (fast update).

V. RLS UPDATE

We denote the FIR filter Q of length L and the secondary path T_{12} (modeled as FIR filter of length M) by

$$\mathbf{q}_k = [\ q_k^0 \quad \dots \quad q_k^{L-1}\ ]^T \tag{9}$$

$$\mathbf{t}_{12} = [\ t^0 \quad \dots \quad t^{M-1}\ ]^T \tag{10}$$

We shall use the recursive prediction error adaptive algorithm [8] to update the filter. This algorithm is based on minimization of the cost function $\xi = E[e_k^2]$. Since ξ is generally unknown or the signals are nonstationary, the algorithm is designed to minimize at each instant of time the mean square error, $\zeta_k = e_k^2$ [9]. The algorithm updates the coefficient vector in (9) along the negative gradient of ζ_k, defined as

$$\nabla_{\mathbf{q}}\zeta_k \equiv \frac{\partial \zeta_k}{\partial \mathbf{q}} = e_k \nabla_{\mathbf{q}} e_k = -e_k \nabla_{\mathbf{q}} \tilde{s}_k, \quad \tilde{s}_k = \sum_{j=0}^{M-1} t^j s_{k-j}$$

$$\begin{aligned} \nabla_{\mathbf{q}}\tilde{s}_k &= \left[\ \frac{\partial \tilde{s}_k}{\partial q_k^0} \quad \dots \quad \frac{\partial \tilde{s}_k}{\partial q_k^{L-1}}\ \right]^T \\ &= \sum_{j=0}^{M-1} t^j \left[\ \frac{\partial s_{k-j}}{\partial q_k^0} \quad \dots \quad \frac{\partial s_{k-j}}{\partial q_k^{L-1}}\ \right]^T \end{aligned} \tag{11}$$

Noting that the adaptive filter output is given by

$$s_{k-j} = \sum_{l=0}^{L-1} q_{k-j}^l r_{k-j-l}$$

we have

$$\nabla_{\mathbf{q}}\tilde{s}_k = \sum_{j=0}^{M-1} t^j \sum_{l=0}^{L-1} r_{k-j-l} \left[\ \frac{\partial q_{k-j}^l}{\partial q_k^0} \quad \dots \quad \frac{\partial q_{k-j}^l}{\partial q_k^{L-1}}\ \right]^T \tag{12}$$

Now assuming that the filter coefficients change slowly

$$\mathbf{q}_k \approx \mathbf{q}_{k-j}, \quad 1 \le j \le L-1$$

we have

$$\begin{aligned} \nabla_{\mathbf{q}}\tilde{s}_k &= \sum_{j=0}^{M-1} t^j \sum_{l=0}^{L-1} r_{k-j-l} \left[\ \frac{\partial q_k^l}{\partial q_k^0} \quad \dots \quad \frac{\partial q_k^l}{\partial q_k^{L-1}}\ \right]^T \\ &= \sum_{j=0}^{M-1} t^j [\ r_{k-j} \quad \dots \quad r_{k-j-L+1}\ ]^T \\ &= [\ \tilde{r}_k \quad \dots \quad \tilde{r}_{k-L+1}\ ]^T \\ &= \tilde{\mathbf{r}}_k \end{aligned} \tag{13}$$

where we define the filtered input

$$\tilde{\mathbf{r}}_k = [\ \tilde{r}_k \quad \dots \quad \tilde{r}_{k-L+1}\ ]^T, \quad \tilde{r}_k = \sum_{j=0}^{M-1} t^j r_{k-j}$$

Finally we have the update equation

$$\begin{aligned} \mathbf{P}_{k+1}^{-1} &= \frac{1}{\lambda}\left(\mathbf{P}_k^{-1} - \frac{\mathbf{P}_k^{-1}\tilde{\mathbf{r}}_k\tilde{\mathbf{r}}_k^T\mathbf{P}_k^{-1}}{\lambda + \tilde{\mathbf{r}}_k^T\mathbf{P}_k^{-1}\tilde{\mathbf{r}}_k}\right) \\ \mathbf{q}_{k+1} &= \mathbf{q}_k + \mathbf{P}_{k+1}^{-1}\tilde{\mathbf{r}}_k e_k \end{aligned} \tag{14}$$

with initial conditions, $\mathbf{P}_0 = \varepsilon I$, $\mathbf{q}_0 \equiv 0$ where ε is a small positive scalar and the forgetting factor satisfies $0 << \lambda < 1$. If we choose $\mathbf{P}_k^{-1} = \mu I$, then we get the Least mean squares (LMS) algorithm given by

$$\mathbf{q}_{k+1} = \mathbf{q}_k + \mu\tilde{\mathbf{r}}_k e_k \tag{15}$$

where μ is the step size. The adaptive-Q LQG control update is given by

$$u_k = F\hat{x}_k + s_k, \quad s_k = \sum_{j=0}^{L-1} q_k^j r_{k-j} \tag{16}$$

There are many efficient algorithms available to carry out the RLS update [10]. We discuss two such algorithms based on inverse QR decomposition. The computational complexity of both algorithms are $O(L^2)$ operations per iteration. The inverse QR method, also known as the square-root RLS is given by

$$\begin{bmatrix} 1 & \lambda^{-1/2}\tilde{\mathbf{r}}^T\mathbf{P}_k^{1/2} \\ 0 & \lambda^{-1/2}\mathbf{P}_k^{1/2} \end{bmatrix} \Theta_k = \begin{bmatrix} \gamma_{k+1}^{-1/2} & 0 \\ \mathbf{g}_{k+1}\gamma_{k+1}^{-1/2} & \mathbf{P}_{k+1}^{1/2} \end{bmatrix} \tag{17}$$

$$\mathbf{q}_{k+1} = \mathbf{q}_k + [\mathbf{g}_{k+1}\gamma_{k+1}^{-1/2}][\gamma_{k+1}^{-1/2}]^{-1} e_k \tag{18}$$

This algorithm finds a unitary matrix Θ_k that lower-triangulizes the pre-array in (17) and generates a post-array with positive diagonal entries. It then updates the Q filter (18) using the elements read from the first column of the post-array. The matrix Θ_k is computed in an iterative fashion using Householder transformations [10]. Lower-triangulization of the pre-array is not an elemantary operation and it involves considerable computation time. To reduce this time we look at yet another algorithm which updates Θ_k using elementary Givens rotations [11]. This second algorithm exploits the structure of the unitary matrix Θ_k and eliminates the need for lower triangulization. This considerably reduces the complexity in implementing the inverse QR scheme and makes the second scheme faster than the first (see Table I).

VI. SIMULATION AND EXPERIMENTAL RESULTS

A. Simulation

The two algorithms discussed in the previous section were tested on Matlab and implemented on a TMS320C67 floating point DSP. Position error signal (PES) data collected from the 7200 rpm disk drive was used in the simulations to emulate output disturbance. The simulations were run for a time period of 1 sec. Fig. 5 plots the plant output variance under regulation with and without the adaptive-Q control. The error variance is calculated as a moving average according to

$$\sigma_k^2 = \frac{\sigma_{k-1}^2(k-1)+e_k^2}{k}, \quad \sigma_0^2 = 0$$

From fig. 5 we see that adaptive-Q with $L=5$ gives a 15% improvement over the fixed LQG ($Q \equiv 0$) compensator. Fig. 6 shows the plant output spectrum with and without the adaptive-Q filter. We see that the adaptive-Q filter has knocked down some of the peaks in the disturbance response. Since the sampling rate is high and we need to update the adaptive filter every servo cycle, computation time puts a limit on the adaptive-Q filter length L. To calculate the computation time required we tested the two algorithms discussed in the previous section. The CPU loading factor for the two algorithms on the C67 DSP is shown in Table I. The first algorithm does not allow for a filter length greater than 4. Hence the second algorithm is chosen for implementation on the 7200 rpm disk drive.

B. Experimental Results

We used the 7200 rpm disk drive for experimental verification of the proposed control algorithm. The built in control loop was first opened and then we used the DSP to close the loop externally. We have the LQG control setup as the inner loop stabilizing controller and the adaptive-Q sits on top of it. The outer loop is set up with a software toggle switch which can turn on/off the adaptive-Q control. Experiments were performed on all three available disk drive heads and three different track positions on the inside, middle and outer edge of the disk. Table II shows the 3σ value of the position error as a percentage of the track width. We compare the performance between LQG control ($Q \equiv 0$) and three different implementations of the adaptive-Q scheme. The first is an LMS scheme (see eq.15) with step size $\mu = 0.01$ and filter length $L=5$. The second is an LMS scheme with step size $\mu = 0.01$ and filter length $L=10$. The third is the inverse QR based RLS scheme (see eq.14) with filter length $L=5$. As can be seen from Table I it is not possible to run the RLS scheme with a filter length higher than 8. LMS algorithm is not preferred because it requires us to choose a step size (by trial and error) which may or may not be optimal. Table II shows that irrespective of head and track position, the adaptive-Q algorithm improves on the inner loop LQG control system.

VII. CONCLUSIONS

We have demonstrated the effectiveness of the adaptive-Q control algorithm on a Disk Drive with an existing inner loop stabilizing LQG control. There is a consistent improvement in achievable TMR across different heads and tracks when the add-on adaptive controller is turned on. The simulations clearly indicate that increasing the adaptive filter length improves the disturbance rejection property of the adaptive-Q control system. But this is not viable because of computation time constraints. A inverse QR based adaptive array algorithm is chosen for implementation. Future work will focus on improving the inner loop stabilizing control so that it rejects all known disturbances (RRO). Then we hope that the add-on adaptive controller will take care of the remaining NRRO and other random disturbances.

REFERENCES

[1] William Messner and Rick Ehrlich, “A Tutorial on Controls for Disk Drives” *Proc. of the American Control Conference*, Arlington, VA, June 2001, pp. 408-420.

[2] Alexei H. Sacks, Marc Bodson and William Messner, “Advanced Methods for Repeatable Runout Compensation” *IEEE Trans. on Magnetics*, vol.31(2), March 1995, pp.1031-1036.

[3] R. Horowotz and B. Li, “Adaptive Control for Disk File Actuators”, *Proc. IEEE Conf. on Decision and Control*, vol.1, New Orleans, 1995, pp.655-660.

[4] S. Pannu and R. Horowitz, “Adaptive Accelerometer Feedforward Servo for Disk Drives”, *Proc. IEEE Conf. on Decision and Control*, vol.5, San Diego, Dec. 1997, pp. 4216-4218

[5] M.O.T. Cole, P.S. Koegh and C.R. Burrows, “Adaptive-Q Control of Vibration due to Unkown Disturbances in Rotor/Magnetic Bearing Systems”, *Proc. IEEE Conf. on Control Applications*, Alaska, 2000, pp. 965-970.

[6] Teng-Tiow Tay, Iven Mareels and John B. Moore, *High Performance Control*, Birkhauser Boston, 1998.

[7] Sen M. Kuo and Dennis R. Morgan, *Active Noise Control Systems - Algorithms and DSP Implementations*, John Wiley and Sons, NY, 1996.

[8] L. Ljung and T. Soderstrom, *Theory and Practise of Recurvsive Identification*, MIT Press, Cambridge, 1983.

[9] John J. Shynk, “Adaptive IIR Filtering”, *IEEE ASSP Mag.*, April, 1989, pp. 4-21.

[10] A.H. Sayed, *Fundamentals of Adaptive Filtering*, John Wiley and Sons, NY, 2003.

[11] S. Thomas Alexander and Avinash L. Ghirnikar, “A Method for Recursive Least Squares Filtering Based Upon an Inverse QR Decomposition”, *IEEE Trans. on Signal Processing*, vol.41(1), Jan, 1993, pp. 20-30.

VIII. FIGURES AND TABLES

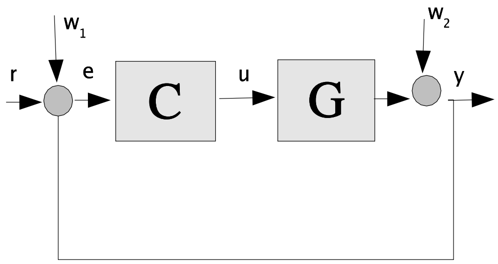

Fig. 1. Closed loop PD

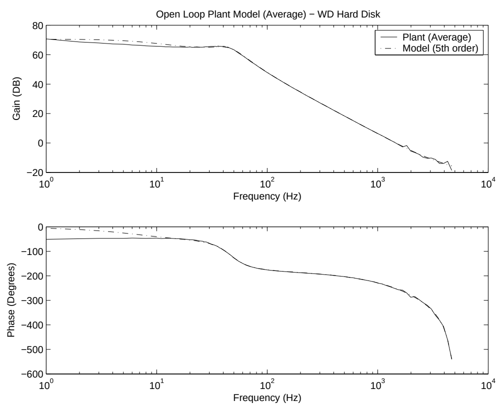

Fig. 2. Disk drive model

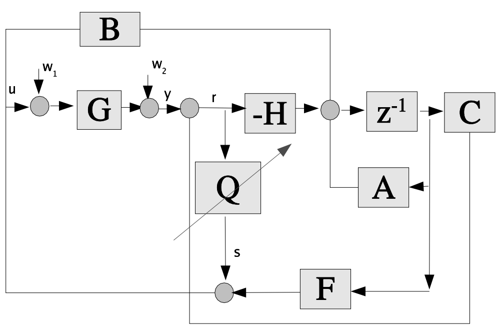

Fig. 3. LQG with stabilizing adaptive-Q control

TABLE I

CPU LOADING CHART

L	algorithm #1	algorithm #2
4	56%	34%
6	111%	61%
8	192%	95%

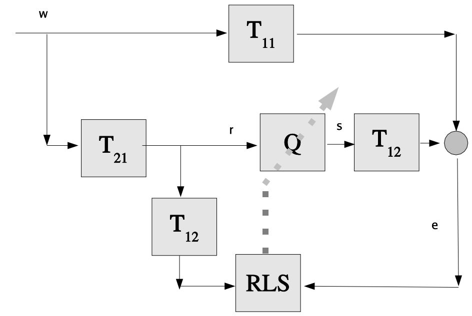

Fig. 4. Filtered-X RLS Adaptive-Q filter

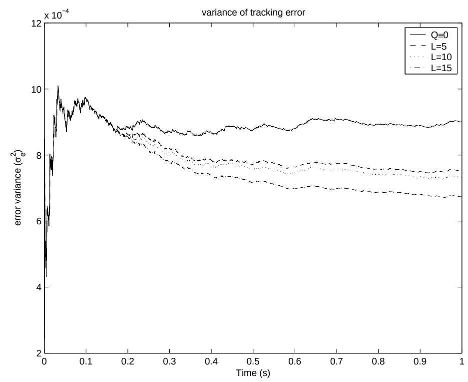

Fig. 5. Error variance curves

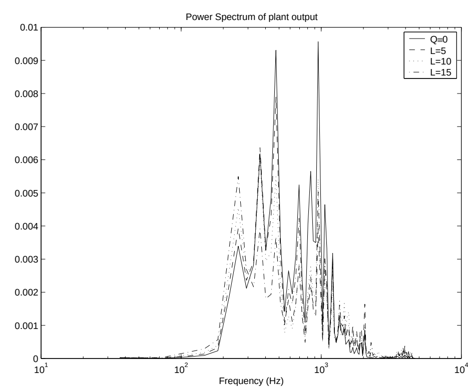

Fig. 6. Plant Output Spectrum

TABLE II

3σ OF ERROR IN TERMS OF % TRACK WIDTH

	Head#0	Head#1	Head#2
	in, mid, out	in, mid, out	in, mid, out
LQG ($Q \equiv 0$)	(9.1, 8.9, 6.3)	(9.9, 9.7, 8.1)	(10.4, 10.4, 8.9)
LMS (L=5)	(7.5, 7.7, 6.3)	(8.1, 7.9, 6.7)	(9.3, 9.5, 8)
LMS (L=10)	(7.2, 7.4, 6.2)	(7.6, 7.4, 6.4)	(9.1, 9.2, 7.8)
RLS (L=5)	(7.5, 7.8, 6.4)	(8.4, 8.3, 6.8)	(9.2, 9.6, 8.3)