韓國精密工學會 '95年度 秋季學術大會論文集 pp.348~353

(자기부상 초정밀 고속 공구 서보 시스템의 모델과 제어)

MODELING AND CONTROL OF A MAGNETIC SERVO-LEVITATED FAST-TOOL SERVO SYSTEM

Hector M. Gutierrez
Department of Electrical and Computer Engineering

Paul I. Ro
Department of Mechanical and Aerospace Engineering

Precision Engineering Center
North Carolina State University
Raleigh, North Carolina

ABSTRACT

Magnetic Servo Levitation (MSL) has been proposed as a method to drive a fast-tool servo system. This paper discusses some fundamental control and modeling issues in the development of a long-range high-bandwidth fast-tool servo based on MSL. A recursive linear model is developed to describe the system's dynamics, and further used to discuss controller design. For a given controller architecture, the performance of two controllers is then compared, one based on an approximation to the inverse plant dynamics, the second based on a adaptive neural network.

INTRODUCTION

Fast-tool servos are actuator devices used to provide fine-tune adjustment of the cutting tool position on real time during some machining operations. They are aimed to compensate errors resulting from mechanical misalignment, slide vibration, tool wear, and cutting dynamics that can not be properly corrected by the slide controller; its purpose is therefore to achieve ultra-high precision accuracy, beyond the limitations inherent to the slide controller.

Current designs are mostly based on the use of piezoelectric stacks for exerting the fast-tool servo action. Although they have a fairly high bandwidth, piezo-based FTS systems have very limited range, in the order of tens of microns. Magnetic Servo Levitation (MSL) has been proposed [9] as an alternative who is also free of friction and backlash problems accompanied by conventional mechanical actuators. While MSL should be able to provide an adequate bandwidth within a motion range twenty times larger than a piezo-based FTS, there are a number of fundamental issues that make implementation difficult. An MSL based system is inherently unstable and highly non-linear. Most mathematical models available for magnetic-based systems are only valid for small displacements around a nominal gap, or not accurate enough to achieve nanometer resolution. This paper addresses some of the fundamental issues in design, modeling and control of such type of actuators.

ELECTROMAGNET DESIGN

The basic design concept of the proposed MSL-based actuator is depicted on Figure 1.

The electromagnets have an "E"-shaped geometry, with the coil mounted on the middle leg of the "E". The design equations were derived from basic electromagnetics; the two more important ones are average flux, Φ, estimated as:

$$\Phi = \frac{Ni}{\frac{l}{\mu_r \mu_o A} + \frac{2g}{\mu_o A}} \qquad (1)$$

and force, F, which is given by:

$$F = \left(\frac{N i \mu_r}{l + 2g\mu_r}\right)^2 \mu_o A \qquad (2)$$

where N is the number of turns, i the current in the coil, l the length of the flux path, g the nominal gap, A the pole area, μ_r the relative permeability of the laminations, and μ_o the permeability of air. These equations were used to size the electromagnets and determine overall dimensions of the system. Figure 2 depicts schematically the final design concept. Position feedback will be implemented by using a laser interferometer, the target mirror being attached to the back of the tool holder plate. Several other design issues (structural design, vibrational modes, cooling and heat transfer) have been addressed, but fall beyond the scope of this paper.

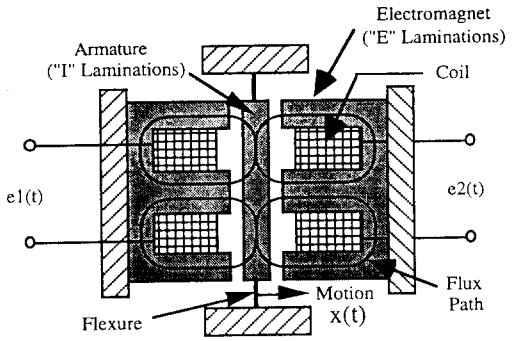

FIGURE 1: ELECTROMAGNET IC ACTUATOR CONCEPT

MATHEMATICAL MODEL OF THE ELECTROMAGNETIC ACTUATOR

Modeling an electromagnetic actuator for long-range high-precision applications requires taking into account several aspects of the system that prevent the use of simplified models, which in general share two basic assumptions: uniform magnetic field density within the magnetic core, and linear magnetic core operation (constant magnetic permeability). Although for short-range low-bandwidth applications (such as magnetic bearings) linear versions of such models seem to work reasonably well [1,2], a long range actuator, where displacement can not be described as small perturbations of a nominal trajectory require a model that is accurate over a wide range of displacement and frequencies.

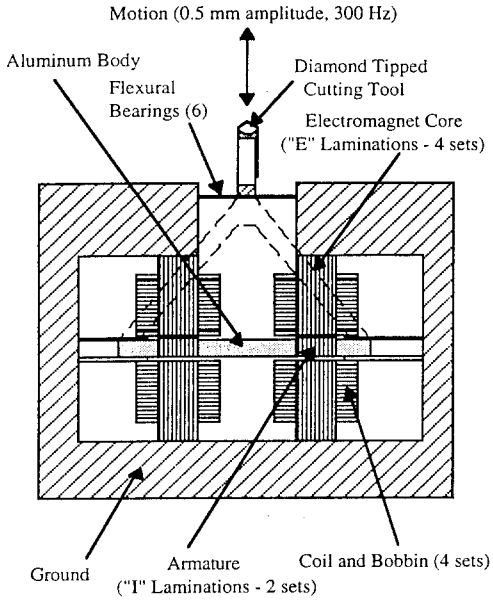

FIGURE 2: ACTUATOR SCHEMATIC

Finite-element based coupled models.

An electromagnetic actuator is a coupled system where motion equations are to be solved simultaneously with electromagnetic and circuit equations. These models share the use of finite-element methods as a tool to solve Maxwell's equations throughout the magnetic circuit.

The electromagnetic field equations can be solved off-line in order to determine magnetic force and flux as static functions of current and position. The finite-element problem is hence solved for several combinations of current and position along the range of motion. The solutions are later used to solve the circuit and motion equations by interpolating parameter values (force, flux) according to measured position and current. Consider a two-magnet levitated system, as depicted in Figure 1. The system's equations are as follows:

$$\frac{d\psi_1}{dt} + Ri_1 = e_1(t) \quad , \quad \frac{d\psi_2}{dt} + Ri_2 = e_2(t) \qquad (3,4)$$

$$m\frac{d^2x_1}{dt^2} + b(t)\frac{dx_1}{dt} + 2K(x_1 - x_o) = F_{mag1} - F_{mag2} + F_e \qquad (5)$$

where i_1, i_2 are coil currents , ψ_1, ψ_2 are total effective flux linkages on coils 1 and 2, respectively, x_1 is the left side gap, x_o is the left side nominal gap (equilibrium position), F_e is an external force applied to the moving mass and F_{mag} is the magnetic force produced by the i^{th} coil:

$$F_{mag_i} = -\frac{1}{2}i_i^2\frac{\partial L}{\partial x} \qquad (6)$$

Notice that these relationships are a quite general description of the system since flux leakage is considered (the magnetic flux is not assumed to link all the turns of a given coil) and uniform field across the core or linear core operation are not assumed. In the general case, inductances are given by:

$$\psi_1 = L_1 i_1 \; ; \quad \psi_2 = L_2 i_2 \qquad (7)$$

Equations (3) to (7) define the coupled model of the electro-magneto-mechanical system:

$$\begin{bmatrix} \frac{d}{dt}L_1 + R & 0 & 0 & 0 \\ 0 & \frac{d}{dt}L_2 + R & 0 & 0 \\ \frac{1}{2}i_1\frac{\partial L_1}{\partial x_1} & -\frac{1}{2}i_2\frac{\partial L_2}{\partial x_2} & m\frac{d^2}{dt^2} + 2K & b \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \\ x_1 \\ \dot{x}_1 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -2kx_o \end{bmatrix} = \begin{bmatrix} e_1 \\ e_2 \\ F_e \end{bmatrix} \qquad (8)$$

The coupled model can be used to simulate the system by integrating (8) by some numerical technique such as the Newton-Raphson method. This requires to previously obtain the parametric surfaces:

$$\psi_1 = \psi_1(i_1, i_2, x_1) \quad ; \quad F_{mag1} = F_{mag1}(i_1, i_2, x_1)$$
$$\psi_2 = \psi_2(i_1, i_2, x_1) \quad ; \quad F_{mag2} = F_{mag2}(i_1, i_2, x_1) \qquad (9)$$

which can be done by interpolation among points provided by the finite element solutions. Although FEM based methods have the appeal of being based on first principles of physics, they all involve a number of approximations and are sensitive to both structural and material parameter uncertainties. Also, they do not consider more complex effects such as Eddy current losses, hysteresis or temperature effects. For long-range, high-bandwidth, high- precision applications, they are not accurate enough.

System identification using parametric models.

A model for the magnetically-levitated servo system can be formulated in such a way that non-linearities are separable and additive:

$$a_1\frac{d^2x}{dt^2} + a_2\frac{dx}{dt} + a_3\ddot{x} = F_{mag2} - F_{mag1} \qquad (10)$$

where each magnetic force can be described as some polynomial of coil current and the corresponding air gap. For instance, for the right magnet:

$$F_{mag2} = \sum_{j=1}^{n}\left(K_{1j}\frac{i_2}{(x_o - x)^{0.5j}} + K_{2j}\frac{i_2^2}{(x_0 - x)^{0.5j}}\right) \qquad (11)$$

Notice that in these equations x is measured from the equilibrium position, i.e. in Figure 1, $x = x_1 - x_o$, where x_o is the nominal left-side air gap. Although there are some references in the literature [9] about the use of polynomial expressions such as (11) to correlate magnetic force, current and position on a static sense, the problem addressed here is to find the unknown coefficients involved in (10) and (11) such that experimental input-output sequences can be fitted by equation (10) in a minimum mean-square error sense.

Several techniques to solve this problem have been described for linear systems [10,11]. In order to use linear parametric techniques to find optimal fits for the parameters in eqs. (10,11), each rational term in (11) can be considered a separate input to the system, in such a way that (10) can be rewritten as:

$$\frac{d}{dt}\begin{bmatrix} x \\ \frac{dx}{dt} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{a_3}{a_1} & -\frac{a_2}{a_1} \end{bmatrix}\begin{bmatrix} x \\ \frac{dx}{dt} \end{bmatrix} + \begin{bmatrix} 0 & 0 & \dots & 0 \\ K_{11} & K_{21} & \dots & K_{4n} \end{bmatrix}\begin{bmatrix} u_1 \\ u_2 \\ \dots \\ u_{4n} \end{bmatrix} \qquad (12)$$

On the other hand, consider the general case of the parametric estimation problem for linear multiple-input single-output

systems. This can be formulated as the estimation of the coefficients of polynomials A,B,C,D,E,F in :

$$A(q)y(t)=\frac{B_1(q)}{F_1(q)}u_1(t-mk_1)+...+\frac{B_n(q)}{F_n(q)}u_n(t-mk_n)+ +\frac{C(q)}{D(q)}e(t) \quad (13)$$

where y(t) is the current output prediction, e(t) is an unmodeled disturbance and q is the delay operator, defined as: q^{-1} f(t) = f(t-T), T being the sampling rate. The parametric methods to be considered are called prediction error methods and are based on the minimization of a cost function which is the sum of the squared prediction errors. Different mathematical techniques can be used for that purpose.

Consider the auto-regressive moving average model with exogenous input (ARMAX model), which is an special case of (13) with $F_1(q) =...= F_n(q) = D(q) = 1$. The coefficients of the other polynomials can be grouped as a vector of unknown parameters (θ), namely:

$$\theta=[\alpha_1,...,\alpha_{n_a},b_1^1,...,b_{n_{b1}}^1,...,b_1^{nu},...,b_{n_{bnu}}^{nu},...,c_1,...,c_{n_c}]^T \quad (14$$

where nu is the number of inputs. The Bayesian prediction error is defined as:

$$\varepsilon(t|\theta)=y(t)-\hat{y}(t|\theta) \quad (15)$$

The ARMAX one-step predictor can be shown to be [11]:

$$\hat{y}(t|\theta)=B_{o_1}(q)u_1(t)+...B_{o_{nu}}(q)u_{nu}(t)+[1-A(q)]y(t)+ +[C(q)-1]\varepsilon(t|\theta) \quad (16)$$

where the polynomials B_{oj} include the delay terms on (13). Defining the one-step ahead regression vector:

$$\phi(t|\theta)=[-y(t-1),...,-y(t-n_a),u_1(t-1),...,u_1(t-n_{b1}),... ...,u_{nu}(t-1),...,u_{nu}(t-n_{bnu}),...,\varepsilon(t-1|\theta),...,\varepsilon(t-n_c|\theta)]^T \quad (17)$$

The ARMAX regression model can now be written as:

$$\hat{y}(t|\theta)=\theta^T\phi(t|\theta)=\phi^T(t|\theta)\theta \quad (18)$$

The estimation problem is aimed to minimize the prediction error cost function:

$$J_N(\theta)=\frac{1}{2N}\sum_{t=1}^{N}\varepsilon^2(t,\theta) \quad (19)$$

Defining: $E(\theta)=[\varepsilon(1|\theta),...,\varepsilon(N|\theta)]^T$, and the matrix O(θ) such that $o_{jt}(q) = \partial\varepsilon(t|\theta)/\partial\theta_j$, the gradient g of J_N can be expressed as:

$$g(\theta)=\frac{\partial J(\theta)}{\partial\theta}=\frac{1}{N}O(\theta)E(\theta) \quad (20)$$

and similarly, the Hessian matrix H of J_N is given by:

$$H=\frac{\partial^2 J}{\partial\theta^2}=\frac{1}{N}O(\theta)O(\theta)^T+\frac{1}{N}\frac{\partial O(\theta)}{\partial\theta}E(\theta) \quad (21)$$

This defines the Gauss-Newton algorithm for updating the parameter vector, namely, for a step size s:

$$\theta_{i+1}=\theta_i-sH^{-1}g(\theta_i) \quad (22)$$

Approximating H as the left term at the right side of (21), the algorithm becomes:

$$\theta_{i+1}=\theta_i-s(OO^T)^{-1}O(\theta_i)E(\theta_i) \quad (23)$$

Using (15) and (18), the vector E(θ) is defined as a function of both input-output measurements and the vector of unknown parameters. Then (20), (21) and (23) define the algorithm that updates θ until the minimum value (or a local minimum) of J_N is reached. Notice that previous knowledge or "reasonable guesses" of the system's parameters (e.g. effective mass, estimated spring stiffness) significantly improves the speed of convergence and minimizes the chance of hitting local minima since they provide initial estimates to the algorithm.

Parameters of the magnetically-levitated servo system model (12) were estimated using the Gauss-Newton algorithm (20,21,23) by first converting (12) to the discrete domain, and then expressing the system's equations as the ARMAX model derived from the general form (13). Model parameters were determined from exciting the system with band-limited noise. Model response was then analyzed with different excitation signals; results (actual output and model predicted output) are shown in Figure 3 A and B. It can be noticed that under some excitations (Fig. 3B) the model is not sufficiently damped to properly track the actual response, while in Figure 3A the model seems to be overdamped. This was corroborated on different tests under different ranges and frequencies, and suggested that the damping term or perhaps the weighting coefficients in the force functions, or both, are actually dependent on position and therefore time-varying.

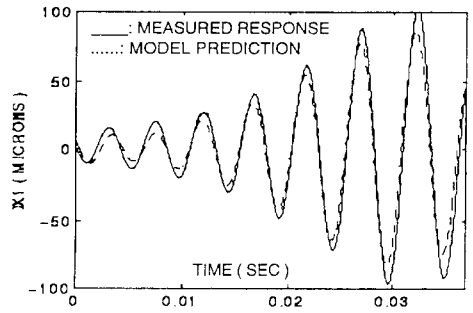

FIGURE 3A. RESPONSE TO SWEPT SINE WAVE

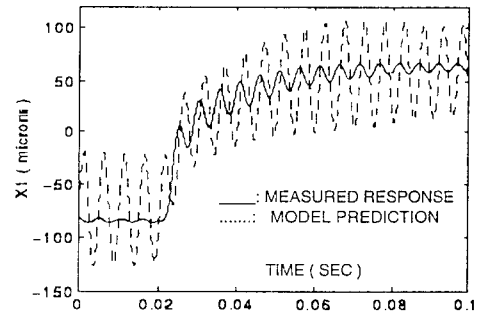

FIGURE 3B. STEP RESPONSE

Since it was not possible to find a given set of parameters that would fit all different sets of data under different experimental conditions, a recursive adjustment of the model parameters was found necessary. Notice that the Gauss-Newton algorithm converges to a unique set of parameters and therefore models the spring-mass system as time-invariant.

There are several alternatives to recursively updating the vector of system's parameters [12]. A typical recursive identification algorithm is:

$$\hat{\theta}(t)=\hat{\theta}(t-1)+K(t)[y(t)-\hat{y}(t)]$$
$$\hat{y}(t)=\phi^T(t)\hat{\theta}(t-1) \quad (24)$$

where y(t) is the observed output at time t and $\hat{y}(t)$ is a prediction of y(t) based on observations up to time t-1. K(t) is typically chosen as a function of the regression vector (eq. 17), which can be conceived as an estimate of the gradient with respect to θ of the one-step ahead predictor (eq.18).

A somewhat simple approach to compute K(t) is to assume a certain model for how the "true" parameters θ_0 change. For instance, this could be described as a white gaussian process with covariance matrix M_1 . A natural choice would then be the use of a Kalman algorithm to find K(t), although it would only be optimal if the underlying description of the observations were a linear regression. The Kalman gain in (24) would be:

$$K(t) = \frac{V(t-1)\phi(t)}{M_2 + \phi(t)^T V(t-1)\phi(t)} \qquad (25)$$

The corresponding covariance matrix of the estimated parameters would be:

$$V(t) = V(t-1) + M_1 - \frac{V(t-1)\phi(t)\phi(t)^T V(t-1)}{M_2 + \phi(t)^T V(t-1)\phi(t)} \qquad (26)$$

where M_1 is the covariance of the true parameters and M_2 is the variance of the innovations: $\varepsilon(t) = y(t) - \phi^T(t)\theta_o(t)$. This approach can be modified if the estimation of M_1, M_2 is difficult, by discounting old measurements exponentially in such way that an observation that is τ samples old carries a weight that is λ^τ of the weight of the most recent observation. The cost function (19) then becomes:

$$J_N(t,\theta) = \frac{1}{2N}\sum_{k=1}^{t}\lambda^{t-k}\varepsilon^2(k,\theta) \qquad (27)$$

In this context, $\tau = 1/(1-\lambda)$ can be conceived as the memory horizon of the approach. The criterion (27) can now be minimized, yielding a different choice for K(t):

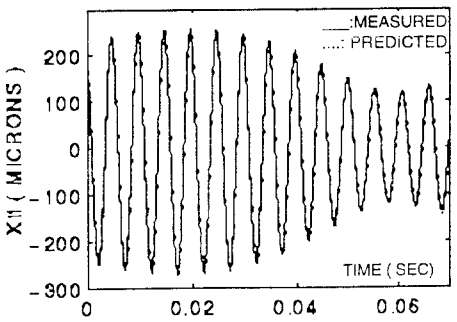

FIGURE 4A. RESPONSE TO SWEPT SINE WAVE

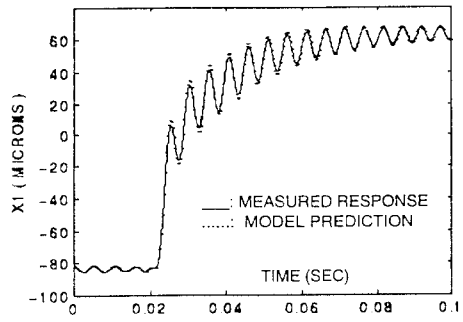

FIGURE 4B. STEP RESPONSE

$$K(t) = \frac{V(t-1)\phi(t)}{\lambda + \phi(t)^T V(t-1)\phi(t)} \qquad (28)$$

$$V(t) = \frac{1}{\lambda}\left(V(t-1) - \frac{V(t-1)\phi(t)\phi(t)^T V(t-1)}{\lambda + \phi(t)^T V(t-1)\phi(t)}\right) \qquad (29)$$

The resulting algorithm (24,28,29) is called the Forgetting Factor approach with forgetting factor λ. This algorithm was implemented to fit experimental input-output data from the magnetically-levitated system, using λ =0.98. The model was initialized by using a vector of parameter estimates and its covariance matrix V(t), generated "off-line" by the Gauss-Newton algorithm, using data from band-limited noise excitation. Results are shown in Figure 4, A and B. It is remarkable that under a wide range of displacements and frequencies, the model thus developed is able to track the measured output with maximum errors ranging from 0.5 to 1 %. Further analysis showed that the auto correlation of the error stays within a 95 % confidence interval of that of a white process. Independence was also analyzed through cross correlating the output with each input, and all combinations showed independence to within 95% confidence.

CONTROLLER DESIGN

The Coordinated Feed-forward Control Method (CFCM) structure.

The development of a model that is both linear and robust opened the possibility of using a controller structure based on an estimation of the inverse plant dynamics. The coordination of feed-forward control method (CFCM) described in [13] and [14] will be used as a basic structure to compare controller performance. The CFCM structure is depicted on Figure 5, and is (in the general case) a MIMO structure, where P represents the plant, B the desired dynamics (model reference system), A is a feed-forward controller, G a feedback loop controller, and y_r the desired system's response.

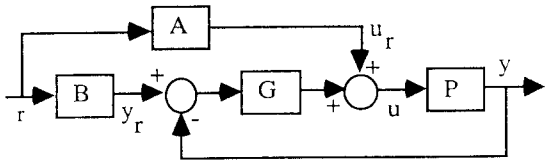

FIGURE 5. THE CFCM CONTROL STRUCTURE

The design equations are:

$$PA = B$$
$$PG = T_L(I-T_L)^{-1}$$

and if P is invertible, this yields:

$$A = P^{-1}B$$
$$G = P^{-1}T_L(I-T_L)^{-1} \qquad (30)$$

The selected feedback loop response (T_L) is the transfer function from y_r to y. If P is unstable, care must be taken when choosing T_L in order to avoid zero-pole cancellations within the loop. If P is stable, we can make $T_L(I-T_L)^{-1} = B$, and it is easy to see that the global transfer function (r to y) becomes B, the desired dynamics.

Approximate inverse-plant dynamics.

The recursive model described in the previous section provides a description of the plant as a MISO system:

$$y(t) = \frac{B_1(q)}{A(q)}u_1(t-mk_1) + \ldots + \frac{B_N(q)}{A(q)}u_N(t-mk_N) + \frac{1}{A(q)}e(t) \qquad (31)$$

where the inputs are delayed rational terms of current and position, and the output y(t) is the next position estimate. Since the plant is not square (number of inputs and outputs are different) the inverse plant problem does not have an exact solution but instead a minimum mean-square error solution $[T_1(q) \ldots T_N(q)]^T$ defined by:

$$y(t)=[G_1(q).....G_N(q)]\begin{bmatrix} u_1(t) \\ \ldots \\ u_N(t) \end{bmatrix}+\frac{1}{A(q)}e(t) \qquad (32)$$

$$\Rightarrow \quad \begin{bmatrix} T_1(q) \\ \ldots \\ T_N(q) \end{bmatrix}[G_1(q).....G_N(q)]\approx I_{NxN}$$

where the delays are now included on the transfer functions G. Although several techniques have been proposed to estimate the inverse plant on a minimum mean square error sense [15], the procedure in general involves the estimation of the impulse response (for each transfer function), and a matrix inversion, where the order of the matrix is related to the number of terms considered when truncating the impulse response. Since model parameters are changing at every iteration, this approach would be impractical from a computational time standpoint for a real time application.

For this reason, a different approach has been undertaken. From a physical point of view, there are only two independent inputs (the coil currents). Since there is only one output, a constraint can be imposed that link both inputs in such a way that the system is invertible. For instance:

$$i_1 = i + i_o \; ; i_2 = -i + i_o \qquad (33)$$

where i_o is some constant bias. This constraint is easy to implement, but more important than that is the fact that both currents have to be somehow be related in order to achieve optimal performance (from a control effort point of view), and therefore even if (33) is replaced by some other relationship, there will always be some expression relating i_1 and i_2.

Now consider (31), where each transfer function is a two-pole model, and all inputs have the same delay (m_k=p for all u). Then (31) can be rewritten as:

$$y(t)=\left(\frac{q^{-p}}{1+a_1q^{-1}+a_2q^{-2}}\right)\left[i_2(t)\sum_{i=1}^{N}\frac{b_i}{(x_o-x)^i}+i_2^2(t)\sum_{j=1}^{N}\frac{b_j}{(x_o-x)^j}+\right.$$
$$\left.+i_1(t)\sum_{k=1}^{N}\frac{b_k}{(x+x_o)^k}+i_1^2(t)\sum_{m=1}^{N}\frac{b_m}{(x+x_o)^m}\right] \qquad (34)$$

where x=x(t)=y(t) is the current position estimate,and x_o is the nominal gap. This can be expressed in the discrete domain as:

$$x_{k+p}+a_1x_{k+p-1}+a_2x_{k+p-2}=i_2(k)f_1(x_k)+i_2^2(k)f_2(x_k)+$$
$$+i_1(k)f_3(x_k)+i_1^2(k)f_4(x_k) \qquad (35)$$

Combining (33) and (35), an estimate for the required current signal i(k) can be obtained, given measurements x(1)...x(k-1) and target locations x(k), x(k+p), x(k+p-1), x(k+p-2). This represents an approximation to the inverse plant dynamics, since (33) is not exact (is limited by hardware). All coefficients in (35) are updated at every iteration.

Since all the corresponding two-pole models in (34) have been found to be stable, this approximate inverse can be used to implement the CFCM controller by taking $T_L(I-T_L)^{-1} = B$, with B given by:

$$B(s)=\frac{\omega_n^2}{s^2+2\xi\omega_n s+\omega_n^2} \qquad (36)$$

where ζ=0.9 and ω_n=1800(2π) Hz. This compensator was tested by computer simulation, tracking a 190 Hz sinusoidal command reference. The sampling rate was 8 KHz, results are shown in Figure 6, for p=2 and N=4.

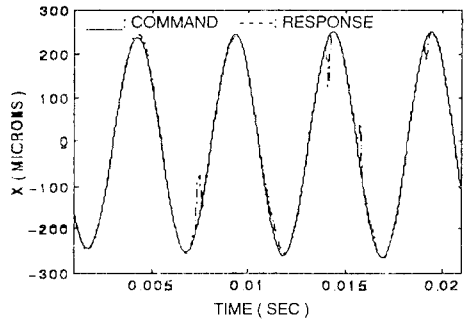

FIGURE 6. CFCM CONTROLLER PERFORMANCE WITH ALGEBRAIC INVERSE PLANT

Inverse plant dynamics on-line estimation by adaptive neural networks.

The inverse plant dynamics in the CFCM control structure can also be estimated by an artificial neural network. However, the results from the modeling section suggest that a multi-layer feed-forward ANN would probably have problems to identify the inverse plant. This is due to the inherent similarity between training a neural net and finding an optimal set of parameters to fit input-output data by a prediction error method: in both cases the sum of squared errors is iteratively evaluated, and the parameters are updated in the direction of the error gradient until the minimum (or a local minimum) of the error surface is reached.

Since we do not expect that a given set of fixed parameters will be able to model all different sets of data (in the same sense that a time-invariant regressive model failed to do so), adaptive neural network identification was attempted.

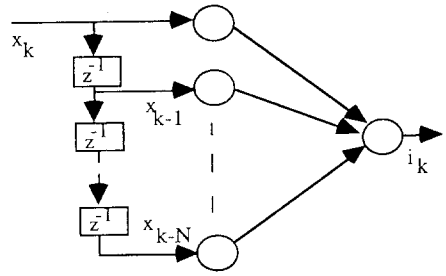

FIGURE 7. SINGLE LAYER LINEAR NETWORK

The network structure is a single layer of linear neurons, where the number of inputs is given by a tab-delayed line, as depicted on Figure 7. The network is trained adaptively (i.e., a single input vector is presented at each time step) using the Widrow-Hoff rule. For a single layer network, the Widrow-Hoff algorithm is defined as:

$$\frac{\partial}{\partial w_{ij}}\left(t_i-b_i-\sum_{j=1}^{R}w_{ij}p_j\right)^2=-2e_ip_j \qquad (37)$$

$$\Delta w_{ij}=\eta e_ip_j; \quad \Delta\vec{b}=\eta\vec{E}, \quad \text{where} \quad \Delta\vec{w}=\eta\vec{E}\vec{p}^T$$

where w_{ij} is the corresponding weight from input j to neuron i, $\mathbf{p}(Rx1)$ is the input vector, **t** the target vector, **e** the network error and η the learning rate. Notice that this type of structure provides a minimum mean-square error linear estimate of the non-linear system. An adaptive linear model will be highly accurate as long as the non-linear system stays near a given operating point. If the operating point changes, it takes a few iterations to the adaptive linear network to adjust to the new operating point. It is easy to see that keeping a high enough sampling rate is critical, since we would like to obtain the linear model of the non-linear plant at the current operating point in the shortest amount of time.

The single layer adaptive linear network was used as an inverse plant predictor within the CFCM structure, and tested by tracking the same 190 Hz sinusoidal command reference as in the previous section. The sampling rate was 8 Khz, results are shown in Figure 8.

Notice the fundamental similarity between both control methods: on both the model-based inverse plant and the adaptive neural net approach, the algorithms are using previous measurements and future desired positions to estimate the next point in the control signal trajectory. Both schemes use adaptive algorithms to adjust model coefficients, based on the optimization of some cost functional. However, the neural network performs better since it estimates the current signal trajectory itself, as opposed to the algebraic inverse method in which the direct plant dynamics is estimated (to predict future values in the output trajectory) and the required current is calculated backwards using (33) and (35).

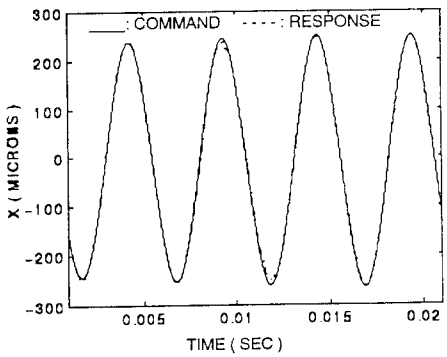

FIGURE 8. CFCM CONTROLLER PERFORMANCE WITH SINGLE LAYER ADAPTIVE LINEAR NETWORK

CONCLUSIONS

Recursive parametric system identification has proven to successfully model the dynamics of the magnetically-levitated fast tool servo system.

Some remarkable features of the model here presented include the fact that prior knowledge of the system can be incorporated, eliminating some fundamental system identification problems such as determination of model order and model structure. Initial guesses of some model parameters given by physical characteristics of the system facilitates convergence of the algorithms and help avoid local minima. The system has been found to be time-varying, which illustrates that the main difference between this fast-tool servo and purely levitated devices (e.g. magnetic bearings) is given by parametric uncertainties in the mechanical structure.

The model is linear and exact (no need for approximations or further assumptions), and performs well under a wide range of displacements and frequencies. On the other hand, the calculations involved in recursively estimating model parameters are fairly simple (equations 22,26,27) and can be performed within the control loop without major sacrifice in closed-loop bandwidth. Since the resulting structure is linear time-varying, several linear controller design techniques are readily available to use with this model.

The coordination of feed-forward control method (CFCM) has been presented and evaluated with two different inverse plant dynamics estimators. The results look very promising, although final implementation will indeed bring a number of new problems. The neural network-based estimator seem to perform better than the algebraic inverse plant, although this may not be so if the final version is forced to run at smaller sampling rates due to hardware limitations. On the other hand, the type of command signal used in this particular experiment (sinusoid) may be making particularly easy to the neural network to adapt to successive operating points . At any rate, the results strongly suggest the use of adaptive techniques both to model and control the magnetically-levitated fast-tool servo.

REFERENCES

[1] Youcef-Toumi,K.,Reddy,S.,Vithiananthan,I.: "Digital time delay controller for active magnetic bearings". *2nd Intl. Simp. Mag.Bearings*, Tokyo, Japan, 1990.

[2] Higuchi,T., Mizuno,T., Tsukamoto,M.: "Digital control system for magnetic bearings with automatic balancing". *2nd Intl.Simp. Mag.Bearings*, Tokyo,Japan, 1990.

[3] Ren,Z.and Razek,A.:"Modelling of dynamical behaviors of electro-magneto-mechanical coupled systems". *Proc.2nd Intl. Conf. Computation in Electromagnetics*, UK, April 1994.

[4] Bedrosian,G.: "New method for coupling finite-element field solutions with external circuits and kinematics". *IEEE Transactions on Magnetics*, Vol.29 No.2,1993.

[5] Piriu,F. and Razek,A.: "Simulation of electromagnetic systems by coupling of magnetic and electric equations". *Mathematics and Computers in Simulation*, Vol.31, 1989.

[6] Godfrey,K.: Perturbation Signals for System Identification. Prentice Hall Intl., London, UK, 1993.

[7] Barker,H. and Davy,R.: "Measurement of second-order Volterra kernels using pseudo-random ternary signals". *International Journal of Control*, Vol. 27,277-291.

[8] Saha, D. and Rao, G.: Identification of Continous Dynamical Systems: The Poisson Moment Functional Approach. Lecture Notes in Control Inf.Sciences, Springer-Verlag, 1983.

[9] Higuchi,T. and Tsuda,M.: "Design and control of magnetic servo levitation". Report of the Institute of Industrial Science. University of Tokyo, Japan, 1992.

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

[11] Van den Bosch, P. and Van der Klauw, A.: Modeling Identification and Simulation of Dynamical Systems. CRC Press, Boca Raton, FL, 1994.

[12] Ljung, L. and Soderstrom T. : Theory and Practice of Recursive Identification. The MIT Press, Cambridge, MA. 1983.

[13] Kuschewski, J. G., Hui, S. and Zak, S. H. : "Application of feed-forward neural networks to dynamical system identification and control". *IEEE Trans. Control Sys. Tech.* Vol. 1, No.1, 1993.

[14] Peczowski, J. L. and Sain, M. K. : "Design of nonlinear multivariable feedback control by total synthesis". Dep. Elec. Eng. , Control Sys. Tech. Rep. No. 36, Univ. Notre Dame, Notre Dame, IN, 1985.

[15] Proakis, J. G. and Manolakis, D. G. : Digital Signal Processing: Principles, Algorithms and Applications. Mac Millan Publishing Co., New York, NY, 1992.