Proceedings of IMECE04
2004 ASME International Mechanical Engineering Congress and Exposition
November 13-20, 2004, Anaheim, California USA

IMECE2004-60256

ADAPTIVE CONTROL OF MEMS MIRRORS FOR BEAM STEERING*

Néstor O. Pérez Arancibia Steve Gibson Tsu-Chin Tsao
Mechanical and Aerospace Engineering Department
University of California, Los Angeles
CA 90095-1597, U.S.A.
nestor@seas.ucla.edu, gibson@ucla.edu, ttsao@seas.ucla.edu

ABSTRACT

This paper presents the design and experimental implementation of an adaptive inverse control system for a two axis MEMS tilting mirror used for optical beam steering. The theoretical issues and practical design considerations involved in this task are discussed in detail. The first topic addressed is the system identification of input-output and state-space models of the MEMS mirror. Consistency among the following two system identification methods is verified: identification of a parameterized transfer function and identification of a state-space model by a subspace method. Next, a stabilizing feedback controller and an adaptive inverse control scheme based on an adaptive inverse *QR* recursive least-squares filter are developed. Finally, the experimental implementation of the control loops is described and the performance of the beam steering system is analyzed.

1 INTRODUCTION

Laser beam steering has a wide range of applications in fields such as adaptive optics, wireless communications or manufacturing process. The control problem is to position the centroid of a laser beam at a desired location on a target with minimal beam jitter in the presence of disturbances. In engineering applications, the necessity for a control system arises because targets usually move, light traveling through the atmosphere is affected by turbulence, and the laser source often is subjected to vibration.

This work consists of creating an experiment that allows us to implement a solution based on an adaptive scheme. As it was mentioned earlier, the optic path is under continuous changes in its own dynamics and in the type of disturbance that affects it. Therefore, it is natural to think of a control system that can adapt on time. Among the many possible adaptive control schemes, in this experiment, a so-called adaptive inverse controller [1,2] was developed based on the inverse *QR-RLS* to achieve fast convergence to optimal gains [15–19].

This paper is organized as follows: In section 2, the experiment is described making clear which are the components of the physical implementation and the role that each of this components play in the plant and control systems. In section 3, a system identification is performed using two independent system identification techniques in order to have a reliable plant model. In section 4, the adaptive control is developed and tested. Finally in section 5, the conclusions of this work are given. A novel technical feature of this implementation is that MEMS mirrors are utilized, making possible to have a first practical insight of the capabilities of this arising technology.

2 SYSTEM DESCRIPTION

From a physical viewpoint, this experimental setup was built as a path for a laser beam which travels from point 1 to point 4 in Fig. 1. At point 1 the laser beam is generated. Point 2 marks the position of the first MEMS mirror which changes the direction of the laser beam towards point 3, where a second MEMS mirror changes the direction of the beam towards point 4. At point 4 a position sensor device (PSD) was allocated. The same type of mirror is used at point 2 and at point 3. These mirrors are able to rotate with respect to two orthogonal axis of reference,

*THIS RESEARCH WAS SUPPORTED BY AFOSR GRANT F49620-02-01-0319

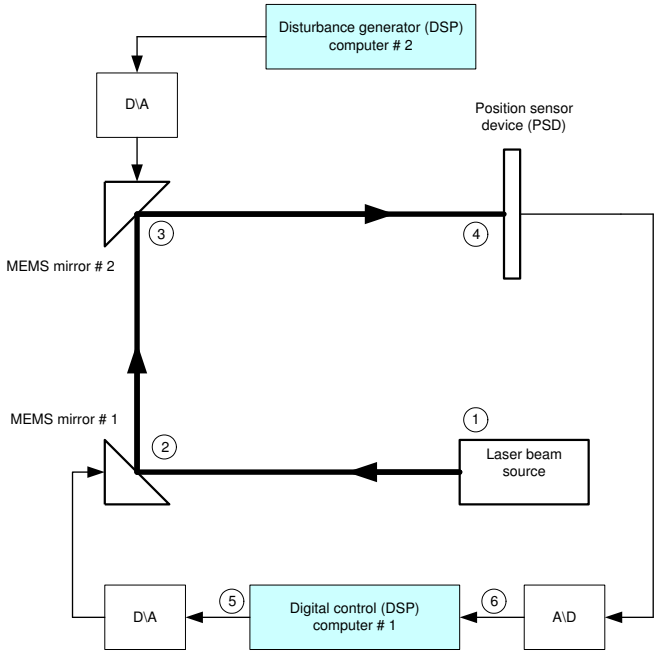

Figure 1. Physical scheme.

between the discrete-time computer and the analog system to be controlled is done by an A/D-D/A board. Fig. 2 shows a picture of the mirrors used for controlling and for generating disturbance in the experimental setup.

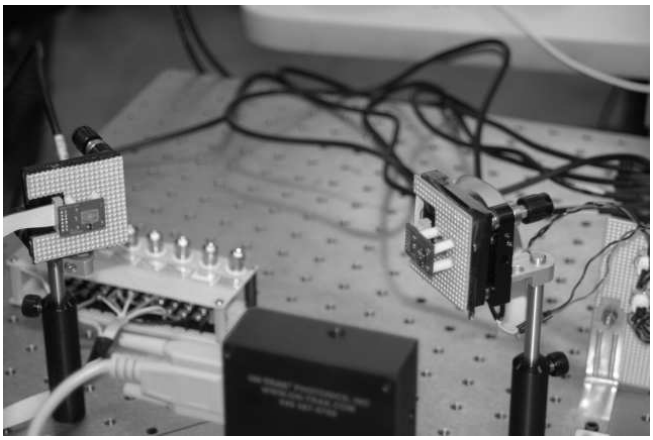

Figure 2. MEMS mirrors view.

which in the context of this paper are labeled as the $h-axis$ and the $v-axis$. Internally the actuation in each axis is realized by two electromagnetic fields with opposing directions created by two coils. The coils are fed using currents of equal magnitude flowing in opposite ways. Nominally the rotation range is ± 5 Degrees, values that are experimentally achieved with a voltage range of approximately ± 1.7 Volts. The reflecting area of the mirrors is $9mm^2$. The position sensor device (PSD) allocated at the end of the beam path measures the position of the laser beam *centriod*. In general the PSD is an optoelectronic device that captures the light intensity distribution, generating current output signals, that are converted to voltage and amplified by an operational-amplifier/feedback circuit. Further electronic processing of these voltage signals yields two final signals, which are the estimates of the centroid coordinates.

From a control point of view this system is composed fundamentally by four subsystems: the digital controller, the actuator, the sensor (PSD) and the disturbance generator. The controller is implemented using C with single precision floating point arithmetic in a digital signal processor (DSP). The sampling frequency was set at 2 KHz. The control system consists of two subsystems, a stabilizing feedback control and a so-called inverse control [1,2] adapted every step using the inverse *QR-RLS* algorithm. Given that the controller is digital, all the analysis from this point onwards is done in the discrete-time domain and the MIMO plant $P(z)$ has its input signal u at point 6 and it has its output signal y at point 5. As shown in Fig. 1 the interface

3 SYSTEM IDENTIFICATION

Most adaptive control schemes use a plant model as a fundamental part of the adaptive controller. Therefore the control performance and the closed loop stability will be limited by the accuracy of this plant model. Considering that there are many physical parameters that are difficult to measure, the obvious choice is to find an identified model for the mentioned plant. In order to find a reliable model, two independent identification methods are employed. First, a parametric identification is performed and next, a subspace algorithm is used. In the parametric search an ARX dynamics is proposed and the parameters for it are found by means of the solution to the classical least-squares problem. This model description has the advantage of being very simple and furthermore is intuitively consistent with the idea of a discrete transfer function in the SISO case, however the order of the system has to be assumed. One way to overcome this problem is to write a state-space representation and then, to use the well known technique of model reduction based on a balanced truncation [3–5]. Using this empirical. example it is possible to appreciate the principal advantage of the subspace techniques for which there is no need for assuming, *a priori*, the order of the identified system or the structure of a transfer function, since it yields a space-state representation immediately.

3.1 Parametric Identification of a Transfer Function

In general, parametric identification methods assume that the dynamics of the system to be identified can be captured by a difference equation and the goal is to estimate the unknown coefficients of it. The order and form of the equation has to be assumed *a priori*. In this case an ARX [8] structure

$$y_k = -\sum_{i=1}^{n} y_{k-i}A_i + \sum_{i=0}^{n} u_{k-i}B_i, \quad k = 0,1,2,... \tag{1}$$

was chosen, where the output sequence y_k and the input sequence u_k could be either scalars or row vectors. It is clear that (1) and $y_k = \phi_k\theta$ are equivalent, where $\phi_k = (-y_{k-1} \ \ldots \ -y_{k-n} \ \ u_k \ \ldots \ u_{k-n})$ and $\theta^T = (A_1^T \ \ A_2^T \ \ldots \ A_n^T \ \ B_0^T \ \ B_1^T \ \ldots \ B_n^T)$. Thus, we can formulate the multiple-input/multuple-output least-squares (*MIMO-LS*) problem

$$\min_{\theta} \ \|Y_N - \Phi_N\theta\|_F, \tag{2}$$

where $\|\cdot\|_F$ denotes the matrix Frobenius norm. It is straightforward to show that the solution to (2) is

$$\hat{\theta} = \Phi_N^+ Y_N, \tag{3}$$

where Φ_N^+ is the pseudoinverse of Φ_N [13–15], $Y_N^T = (y_o^T \ \ y_1^T \ \ \ldots \ \ y_N^T)$, $\Phi_N^T = (\phi_o^T \ \ \phi_1^T \ \ \ldots \ \ \phi_N^T)$ and N is the number of times the system is sampled. In this case, it is clear that $\hat{\theta}$ is unique and (3) holds, since in practical applications N has magnitude of various thousands and the input data to the physical system is chosen to be white noise.

3.2 Subspace State-Space System Identification

The subspace methods for identification of dynamical systems intend to find matrices A, B, C and D for a state-space realization

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k \\ y_k &= Cx_k + Du_k \end{aligned}, \tag{4}$$

through solving two consecutive problems. First an observability matrix

$$O_{CA} = \begin{pmatrix} C \\ CA \\ \vdots \\ CA^{M-1} \end{pmatrix} \tag{5}$$

is found, where M is a positive integer larger than $n = number\ of\ states$. Next, matrices A, C, B and D are computed based on the information contained in O_{CA}.

During the last decade many works expanded on this topic and an unified theoretical solution was published [9]. However, in this case, following some simple key ideas [10, 11] it is possible to find a state-space realization which is consistent with the previous parametric solution.

There exist two principal reasons for using a subspace algorithm. The first is that there is no need for assuming *a priori* the order of the system to be identified and the second is that this technique is indifferent to the number of inputs and outputs, thus to have a MIMO system is no longer an issue.

To begin, notice that $y_k = CA^k x_0 + \sum_{i=0}^{k-1} CA^i Bu_i + Du_k$ is the solution to (4) and therefore it is possible to write

$$Y = O_{CA}X + HU, \tag{6}$$

where

$$Y = \begin{pmatrix} y_0 & y_1 & \ldots & y_{N-1} \\ y_1 & y_2 & \ldots & y_N \\ \vdots & \vdots & \ldots & \vdots \\ y_{M-1} & y_M & \ldots & y_{M+N-2} \end{pmatrix}, \tag{7}$$

$$X = \begin{pmatrix} x_0 & x_1 & \ldots & x_{N-1} \end{pmatrix}, \tag{8}$$

$$H = \begin{pmatrix} D & 0 & \ldots & \ldots & \ldots & 0 \\ CB & D & 0 & \ldots & \ldots & 0 \\ CAB & CB & D & \ldots & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ CA^{M-2}B & CA^{M-3}B & \ldots & \ldots & CB & D \end{pmatrix}, \tag{9}$$

which is referenced sometimes as the impulse response matrix of the system. And

$$U = \begin{pmatrix} u_0 & u_1 & \ldots & u_{N-1} \\ u_1 & u_2 & \ldots & u_N \\ \vdots & \vdots & \ldots & \vdots \\ u_{M-1} & u_M & \ldots & u_{M+N-2} \end{pmatrix}, \tag{10}$$

with $N + M - 2 = number\ of\ times\ the\ system\ is\ sampled$. In general $N \gg M$ and $N \cong number\ of\ samples$.

On the other hand, after computing its singular value decomposition (SVD), the matrix U can be expanded as

$$U = P_U \begin{pmatrix} \Sigma_U & 0 \end{pmatrix} Q_U^T = P_U \begin{pmatrix} \Sigma_U & 0 \end{pmatrix} \begin{pmatrix} Q_{U_1}^T \\ Q_{U_2}^T \end{pmatrix}. \quad (11)$$

The form of matrices P_U, Σ_U and Q_U in (11) comes from the facts that $N >> M$ and that in practice the sequence input u_k is chosen to be white noise, therefore it is reasonable to consider that $rank(U) = M$. Additionally, from standard SVD properties [12], it follows that Q_U is orthogonal. Thus $Q_{U_1}^T Q_{U_2} = 0$, and

$$YQ_{U_2} = O_{CA} X Q_{U_2} \quad (12)$$

Some authors regard matrices playing the function of Q_{U_2} as annihilators, since they allows us to isolate the relevant matrices by eliminating from the original expression what is not needed.

From (12), it follows that the columns of the matrix YQ_{U_2} are linear combinations of the observability matrix O_{CA}. Then the columns of the matrix YQ_{U_2} span a linear subspace which is contained in the subspace spanned by the columns of O_{CA}. Therefore if YQ_{U_2} and O_{CA} have the same rank, then their columns span the same subspace, i.e., $range(YQ_{U_2}) = range(O_{CA})$.

In this way the task can be reduced to finding a set of linearly independent vectors that span the subspace of the columns of YQ_{U_2}. Fortunately there exist a well known linear algebra theorem that allows this to be done.

Theorem 1: Let P, Σ and Q be matrices such that the SVD decomposition for a generic matrix $\Psi \in \Re^{m \times n}$ is

$$\begin{aligned} \Psi &= P\Sigma Q^T \\ &= \begin{pmatrix} p_1 & \cdots & p_r & \cdots & p_m \end{pmatrix} \left(\begin{array}{ccc|c} \sigma_1 & & 0 & \\ & \ddots & & 0 \\ 0 & & \sigma_r & \\ \hline & 0 & & 0 \end{array}\right) Q^T \end{aligned} \quad (13)$$

and let $\sigma_1, ..., \sigma_r$ be the nonzero singular values of Ψ, then:

1. $rank(\Psi) = r$.
2. $\{p_1, \ldots, p_r\}$ is an unitary basis for the linear subspace spanned by the columns of Ψ.
3. $\{p_{r+1}, \ldots, p_m\}$ is an unitary basis for the orthogonal complement of the subspace spanned by the columns of Ψ.

Proof: See [12].

Consequently, $\Psi = YQ_{U_2}$ can be decomposed as

$$\Psi = P_\Psi \begin{pmatrix} \Sigma_\Psi & 0 \\ 0 & 0 \end{pmatrix} Q_\Psi^T = \begin{pmatrix} P_{\Psi_1} & P_{\Psi_2} \end{pmatrix} \begin{pmatrix} \Sigma_\Psi & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} Q_{\Psi_1}^T \\ Q_{\Psi_2}^T \end{pmatrix} \quad (14)$$

and applying property 2 of Theorem 1 it is possible to conclude that the matrix P_{Ψ_1} is actually an observability matrix for some state-space realization which is related for a similarity transformation to the system (4), i.e., $P_{\Psi_1} = O_{CA} T^{-1}$. is an observability matrix for the system given by $\tilde{A} = TAT^{-1}$, $\tilde{B} = TB$, $\tilde{C} = CT^{-1}$ and $\tilde{D} = D$, where T is assumed to be a full rank matrix.

A fundamental result from Linear System Theory assures us that, a linear system can be represented by infinity many state-space realizations [4–7]. Nevertheless, these realizations are related by the similarity transformations, and equally important, the input-output relationship is unique. Thus, from here onwards it is possible to consider $O_{CA} = P_{\Psi_1}$.

At this point, what is left is to compute matrices for (4) using the information contained in O_{CA}. In order to do that let us consider the following relations [11]:

$$E_0 = \begin{pmatrix} C \\ CA \\ \vdots \\ CA^{M-2} \end{pmatrix}, \quad E_1 = \begin{pmatrix} CA \\ CA^2 \\ \vdots \\ CA^{M-1} \end{pmatrix}. \quad (15)$$

From (15) it is obvious that

$$E_1 = E_0 A, \quad (16)$$

which in this case is solved for A multiplying (16) by E_0^+, i.e., $A = E_0^+ E_1$, where E_0^+ is the pseudoinverse of E_0 [12–14]. At the same time it is clear that C is equal to the first n_o rows of the matrix E_0, where n_o is the number of the outputs of the system to be identified. It is important to notice that (14) gives us the order of the system being identified.

Once matrices A and C have been found, the last task is to look for the matrices B and D. In some sense it is surprising that this could be achieved using the collection of inputs and outputs and the observability matrix of this system. To accomplish this, the key idea is to isolate the matrix H in (6) using the proper annihilator.

From property 3 of Theorem 1 it is clear that the subspace which the columns of P_{Ψ_2} span is orthogonal to the subspace that the columns of YQ_{U_2} span. At the same time from (12) it is known that the columns of YQ_{U_2} are linear combinations of the columns of O_{CA}, and recalling that T was assumed to be a full rank matrix, it is possible to conclude that $P_{\Psi_2}^T O_{CA} = 0$, thus the desired annihilator is $P_{\Psi_2}^T$. Consequently premultiplying the

equation (6) by $P_{\Psi_2}^T$ and postmultiplying by U^+, it is possible to see that

$$P_{\Psi_2}^T Y U^+ = P_{\Psi_2}^T H. \tag{17}$$

Now, we define $K_{DB} = P_{\Psi_2}^T Y U^+$ and we partition K_{DB} and $P_{\Psi_2}^T$ in M blocks of equal size, i.e., $K_{DB} = (K_1 \quad K_2 \quad \dots \quad K_M)$ and $P_{\Psi_2}^T = (P_1 \quad P_2 \quad \dots \quad P_M)$ [11]. This allows us to write

$$\begin{aligned}(K_1 \quad K_2 \quad ... \quad K_M) = (P_1 \quad P_2 \quad ... \quad P_M) \\ \cdot \begin{pmatrix} D & 0 & ... & ... & ... & 0 \\ CB & D & 0 & ... & ... & 0 \\ CAB & CB & D & ... & ... & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ CA^{M-2}B & CA^{M-3}B & ... & ... & CB & D \end{pmatrix}.\end{aligned} \tag{18}$$

Furthermore,

$$\begin{aligned}\begin{pmatrix} K_1 \\ K_2 \\ \vdots \\ K_M \end{pmatrix} = \begin{pmatrix} P_1 & P_2 & P_3 & ... & ... & P_M \\ P_2 & P_3 & ... & ... & P_M & 0 \\ P_3 & P_4 & ... & ... & ... & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ P_{M-1} & P_M & 0 & ... & ... & 0 \\ P_M & 0 & ... & ... & ... & 0 \end{pmatrix} \\ \cdot \begin{pmatrix} I & 0 \\ 0 & E_0 \end{pmatrix} \begin{pmatrix} D \\ B \end{pmatrix}.\end{aligned} \tag{19}$$

In this way a relationship that permits us to solve for B and D has been found. It is clear that in order to have a unique solution, M has to be chosen large enough. In this work, this problem was solved using the least-squares (*LS*) solution.

3.3 Model Reduction

For the case when using a parametric identification, once a state-space model has been found, it is desired to have a minimal realization [4–7], which by definition is a realization that is both controllable and observable. In reality the identified realizations are going to be minimal, nevertheless one can think intuitively of states that are more observable than others and states that are more controllable than others. Thus, the idea would be to eliminate the states that are both comparatively not very observable and not very controllable at the same time. Unfortunately, it is known that the so-called controllability and observability ellipsoids in general are not aligned [4]. From linear system theory it is well known that a way to see how controllable a system is, is to look at the discrete time controllability gramian defined as $W_C = C_{AB} C_{AB}^T$, with C_{AB} being the controllability matrix of the system defined as $C_{AB} = (B \quad AB \quad A^2B \quad ... \quad A^{n-1}B)$ and with $n = numberofstates$. From the definition it is easy to see that W_C is a nonnegative definite matrix and that $range(W_C) = range(C_{AB})$.

Additionally it can be shown that if A is stable then W_C solves $W_C - AW_CA^T = BB^T$ and the pair (A,B) is controllable if and only if W_C is positive definite ($W_C > 0$). Furthermore the eigenvalues of $W_C^{1/2}$, defined as in [4], are a measure of the controllability of their corresponding state (unit-norm eigenvector). In other words if a state has a corresponding comparatively small eigenvalue with respect to the others, then this state is comparatively less controllable [4].

Similarly, the observability gramian is defined as $W_O = O_{CA}^T O_{CA}$, which in the case that A is stable, solves the equation $W_O - A^T W_O A = C^T C$ and the pair (C,A) is observable if an only if W_O is positive definite ($W_O > 0$). Analogous to the controllability case, each eigenvalue of $W_O^{1/2}$, defined as in [4], is a measure of the observability of its corresponding state.

A realization is called balanced if its controllability and observability gramians are diagonal and equal and therefore the controllability and observability ellipsoids are aligned. Under the assumption that the matrix A is stable, the theoretical and computational issues for finding a balanced realization were solved during the eighties [3] for the continuous time case. Nevertheless as it is noted in [3], the continuous time method applies to the discrete time case, keeping in mind the intrinsic differences between discrete and continuous time systems; for instance the differences in the Lyapunov equations that their gramians solve.

This method can be applied to either case, because it is based on a so called congruence transformation such that

$$\begin{aligned}TW_CT^T = T^{-T} W_O T^{-1} = \Sigma \\ (\tilde{A}, \tilde{B}, \tilde{C}, \tilde{D}) = (TAT^{-1}, TB, CT^{-1}, D)\end{aligned} \tag{20}$$

where T is a nonsingular matrix and where Σ is a diagonal positive definite matrix if (A,B,C) is both controllable and observable or where Σ is a diagonal semipositive definite matrix if (A,B,C) is not both controllable and observable. The eigenvalues of Σ are ordered in a decreasing manner.

It is interesting to note that the previous transformation is possible because the eigenvalues of the system matrix, A, are invariant under similarity transformation, but on the other hand the eigenvalues of the gramians are not. For rigorous technical details see [4] and for algorithm implementation see [3].

Once the balanced realization has been obtained, arbitrarily the states corresponding to comparatively small singular values are discharged. This process is called balanced truncation. In general a way to see the effect of this truncation is to compare the frequency responses of the original and the truncated system. There are some works that support technically the validness of

this algorithm for the continuous formulation [4]. In this case it was used in a discrete time case and empirically it worked as desired.

In theory model reduction is not needed if using a subspace algorithm, nevertheless in practice the last element of the matrix Σ_Ψ is always a very small number but larger than zero. Thus, one option could be to get a system of order M and after to realize a balanced truncation. In this case, the identification of the order was performed during the computing process by including a function able to notice jumps in the elements of Σ_Ψ.

3.4 Computing the Plant Model

In this section the task of finding a model for the open-loop plant is discussed. Both axis are considered as SISO systems that are labeled as $P_1(z)$ in the $h-axis$ case, and as $P_2(z)$ in the $v-axis$ case. In the parametric identification of both axis, the ARX model was considered as having order 96 and the input data and output data sequences were sampled 20,000 times each. Next, state-space realizations were computed. For the subspace identification of both axis, the corresponding constants were set as $M = 96$ and $N = 12,000$.

The resulting balanced realizations obtained using the parametric method showed clearly, for both axis that a good model would be of order 2, since for the the $v-axis$ case, the third diagonal element of the $W_O = W_C$ matrix was more than a hundred times smaller than the second diagonal element. At the same time for the $h-axis$ case, the third diagonal element of the $W_O = W_C$ matrix was more than a hundred and a half times smaller than the second diagonal element. In the case of the subspace identification, (14) gives us the order of the system being identified, since there exists a clear jump in the magnitude of the diagonal elements of the matrix Σ_Ψ. This yield the following state-space models: $(A_{SS_1}, B_{SS_1}, C_{SS_1}, D_{SS_1},)$, which is the model for $P_1(z)$ using the subspace algorithm, $(A_{ARX_1}, B_{ARX_1}, C_{ARX_1}, D_{ARX_1},)$, which is the model for $P_1(z)$ using the parametric method, $(A_{SS_2}, B_{SS_2}, C_{SS_2}, D_{SS_2},)$, which is the model for $P_2(z)$ using the subspace algorithm and finally $(A_{ARX_2}, B_{ARX_2}, C_{ARX_2}, D_{ARX_2},)$, which is the model for $P_2(z)$ using the parametric method. The identified models are:

$$\left(\begin{array}{c|c} A_{SS_1} & B_{SS_1} \\ \hline C_{SS_1} & D_{SS_1} \end{array}\right) = \left(\begin{array}{cc|c} 0.9227 & 0.3907 & -0.5205 \\ -0.3824 & 0.9198 & -3.2368 \\ \hline -0.1234 & 0.0452 & 0.0281 \end{array}\right)$$

$$\left(\begin{array}{c|c} A_{ARX_1} & B_{ARX_1} \\ \hline C_{ARX_1} & D_{ARX_1} \end{array}\right) = \left(\begin{array}{cc|c} 0.9217 & -0.3867 & -0.4052 \\ 0.3867 & 0.9204 & 0.5238 \\ \hline -0.4052 & -0.5238 & -0.0002934 \end{array}\right)$$

$$\left(\begin{array}{c|c} A_{SS_2} & B_{SS_2} \\ \hline C_{SS_2} & D_{SS_2} \end{array}\right) = \left(\begin{array}{cc|c} 0.9296 & 0.3622 & 0.3848 \\ -0.3622 & 0.9248 & 0.4581 \\ \hline 0.3848 & -0.4581 & 0.0041 \end{array}\right)$$

$$\left(\begin{array}{c|c} A_{ARX_2} & B_{ARX_2} \\ \hline C_{ARX_2} & D_{ARX_2} \end{array}\right) = \left(\begin{array}{cc|c} 0.9229 & -0.3631 & -0.3945 \\ 0.3631 & 0.9225 & 0.4986 \\ \hline -0.3945 & -0.4986 & 0.00002407 \end{array}\right)$$

Figures 3 and 4 show the consistency between the two independent identification methods utilized in this work. The vali-

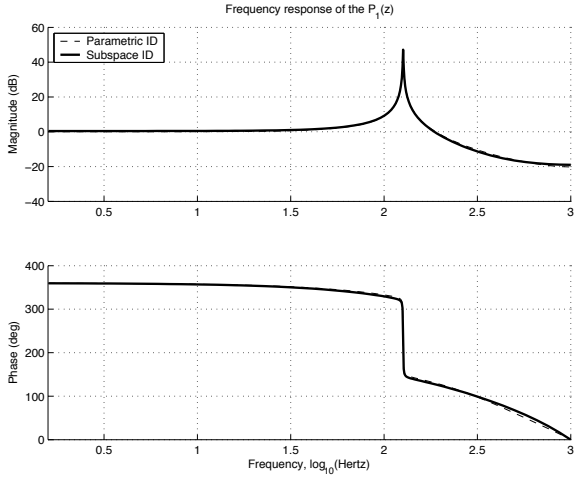

Figure 3. Bode plot of $P_1(z)$.

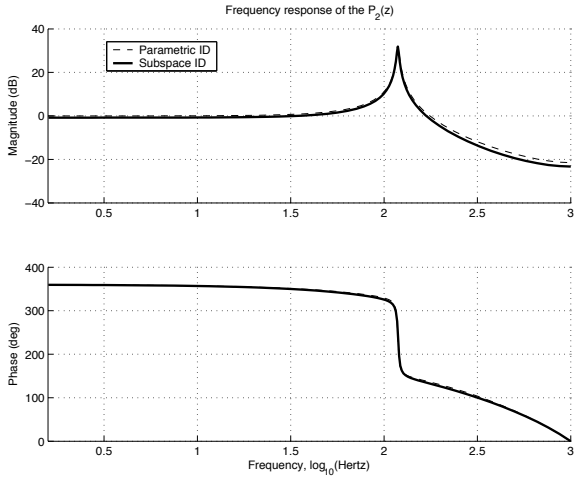

Figure 4. Bode plot of $P_2(z)$.

dation of the identified model was done comparing the physical output of the system with the simulated output of the identified model using a random sequence as input. Not surprisingly for both axis the match is almost perfect. These results are shown in Fig. 5. This plot shows a small time windows, otherwise no difference between both signals could be appreciate.

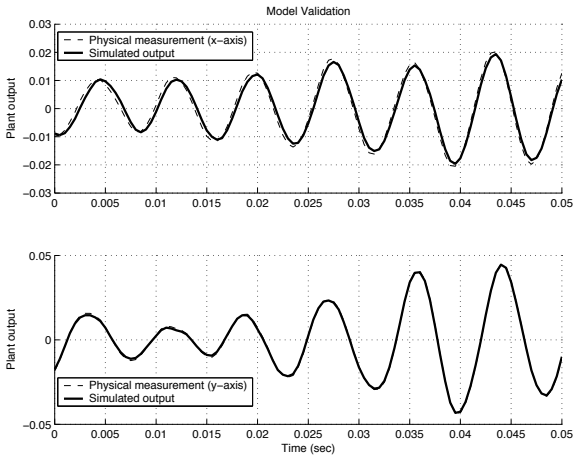

Figure 5. Model validation.

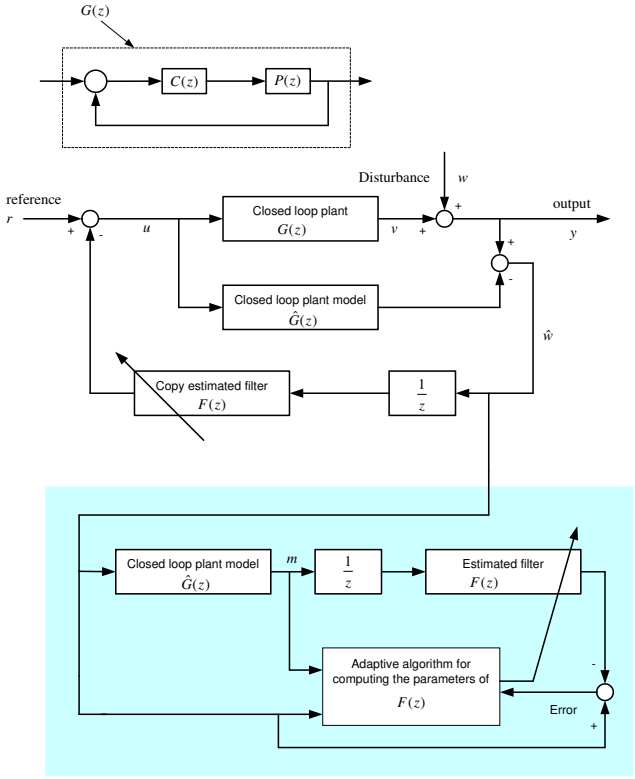

Figure 6. Adaptive inverse control scheme.

4 ADAPTIVE CONTROL IMPLEMENTATION

This section is based on the idea of adaptive inverse control (AIC) [1,2]. The approach comes from applications on signal processing systems used for noise canceling and in principle intends to find a transfer function $F(z)$ such that it is the inverse transfer function for the plant system $G(z)$. This idea appears naive and of course, in general it is not possible to find a causal function $F(z)$ in the whole range of frequencies, nevertheless empirically it has been seen that this goal is achievable in a limited range of frequencies which is reasonable in many applications with limited bandwidth disturbance.

Having the previous statements in mind, the scheme in Fig. 6 is proposed. In this scheme $G(z)$ is the closed loop plant which has already a stabilizing feedback control incorporated. $P(z)$ is the open loop plant of the system, $C(z)$ is the stabilizing feedback control and $\hat{G}(z)$ is the a closed loop plant model identified off line.

In every step an estimate $\hat{w}$ for the disturbance is computed based on a model of the closed loop plant. For purposes of analysis consider for now that the reference is set $r_k = 0,\ \forall\ k$, thus once $\hat{w}$ is obtained, it is filtered through $F(z)$. Then, ideally the output in $F(z)$ is the required input in $G(z)$ such that the output in $G(z)$ is $v_k = -\hat{w}_k$. Therefore, if v_k is almost equal to the disturbance w_k, then the control error is close to 0 and the noise is canceled.

At this point there are some issues to be mentioned. First, it is evident that this noise canceling scheme could not work if the disturbance has components of high frequency, i.e., close to the Nyquist frequency. Second, stability and the performance of this control scheme is limited by the accuracy of $\hat{G}(z)$, and third, performance and stability of the whole system will depend on $F(z)$. $F(z)$ is chosen to be a finite impulse response filter (FIR) principally for stability reasons, since FIR filters are always stable.

Arguments about stability are described in [1,2] reasoning that the transfer function $H(z) = \frac{Y(z)}{R(z)}$ is

$$H(z) = \frac{G(z)}{1 + z^{-1}G(z)F_k(z) - z^{-1}\hat{G}(z)F_k(z)}, \qquad (21)$$

where the subindex k means that the filter $F(z)$ is updated every step.

Looking at (21) it is possible to notice that when $\hat{G}(z) = G(z)$ the plant dynamics remains unchanged, i.e., $H(z) = G(z)$. Therefore, since the closed loop plant $G(z)$ is stable then $H(z)$ is stable as well. Thus, heuristically it can be said that if the

dynamics of $\hat{G}(z)$ is close to the dynamics of $G(z)$, $H(z)$ will remain stable. Otherwise, there exist coefficients of $F(z)$ that makes $H(z)$ unstable. A more rigorous analysis will be attempted in future works. For further arguments on stability and optimality see [1,2].

In general $F(z)$ can be computed in three ways. This can be done off-line, in which case the control scheme could be called inverse but not adaptive. A second way is implementing a so-called pseudo adaptive inverse control where $F(z)$ is compute on line by solving a typical least-squares problem storing several thousands of points [20]. Third, this scheme can be implemented solving a least-squares problem in a recursive manner (*RLS*), in which the coefficients for $F(z)$ are computed in every step. This last way was chosen for this work, solving the *RLS* problem using the algorithm called inverse *QR-RLS* [15–19]. A future publication will show a solution using a more efficient algorithm with a lattice filter structure.

4.1 Stabilizing Feedback Control Design

The first step in the implementation of the proposed control scheme is finding a feedback control law for the open loop plant $P(z)$. This stabilizing control is needed since the open-loop plant is marginally stable, i.e., the poles of $P_1(z)$ and $P_2(z)$ are very close to the unit circle. From section 3 the state-space models of $P_1(z)$ and $P_2(z)$ are known. The control is implemented in a decoupled manner, therefore we are going to focus on one axis only labeled $P_j(z)$ *with* $j = 1,2$. Given the distribution of poles and zeros of $P_j(z)$ the controller

$$C(z) = K_P \frac{(z-z_1)(z-z_2)}{(z-p_1)(z-p_2)} \tag{22}$$

is proposed, where $z_1 = \rho\zeta_1$, $z_2 = \rho\zeta_2$, $K_p > 0$ and $1 > \rho \simeq 1$, with ζ_1 and ζ_2 being the poles of the plant $P_j(z)$ and with $p_1 = 0$ and $p_2 = -1$. After tuning, using the gain margin and the phase margin as the criteria of robustness, the closed-loop frequency responses shown in Fig. 7 are achieved.

4.2 Adaptive Filter Algorithm

The transfer function $F(z)$ in Fig. 6 is chosen to be an FIR filter. This filter will be adaptive because in every time step the vector of weights $B_i = (\,b_0 \quad b_1 \quad b2 \quad \ldots \quad b_{M-1}\,)^T$ in Fig. 8 is computed using the inverse *QR-RLS* algorithm. This algorithm is a member of the family known as square-root algorithms that solve the recursive least-squares problem in a numerically stable manner. For details about its derivation refer to [15–19]. We say that the filter $F(z)$ is order $M-1$ when $F(z) = b_0 + z^{-1}b_1 + \ldots + z^{-M+1}b_{M-1}$.

In our control scheme of Fig. 6, the goal is to achieve the desired output $\hat{w}_i$ for a given input sequence m_i to $F(z)$. For that

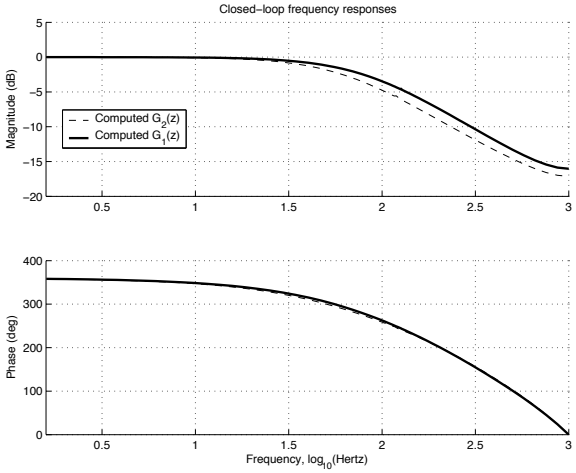

Figure 7. Closed-loop Bode plots.

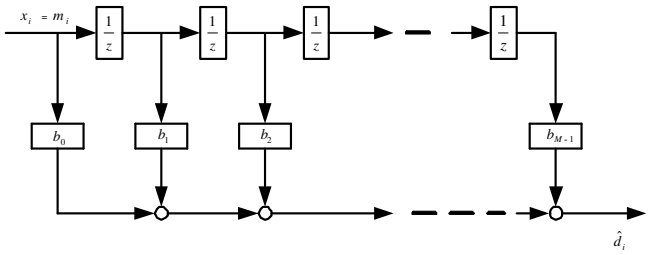

Figure 8. FIR filter.

reason, in this section, we define $x_i = m_i$ and we refer to it as the input sequence. Using the input sequence x_i, the so-called regressor vector is defined as $X_i = (\,x_i \quad x_{i-1} \quad x_{i-2} \quad \ldots \quad x_{i-M+1}\,)$. Finally we define $d_i = \hat{w}_i$, which is referred as the desired output, and $\hat{d}_i = X_i B_{i-1}$, which is the actual computed output from the adaptive filter at time i. Only for completeness we state the formulation of the *RLS* problem and the inverse *QR* solution.

Consider the optimization problem

$$\min_B \left(\lambda^{(N+1)} (B-\bar{B})^* \Pi (B-\bar{B}) + \sum_{j=0}^{N} \lambda^{(N-j)} |d_j - X_j B|^2 \right) \tag{23}$$

and data $\{X_j, d_j\}_{j=0}^N$, where the X_j are $1 \times M$ and the d_j are scalars. Consider also an $M \times 1$ vector $\bar{B}$, an $M \times M$ positive-definite matrix Π, and a positive scalar $\lambda \leq 1$. Usually, Π is referred as the regularization matrix and λ is called the forgetting factor.

The solution, B_N, of the least-squares problem stated as (23) can be recursively computed as follows:

Algorithm 1 (Inverse QR-RLS): Let $\Sigma = \Pi^{-1}$ and introduce the Cholesky decomposition $\Sigma = \Sigma^{1/2}\Sigma^{*/2}$, where $\Sigma^{1/2}$ is lower triangular with positive diagonal entries. Then start with $B_{-1} = \bar{B}, P_{-1}^{1/2} = \Sigma^{1/2}$ and repeat for $i \geq 0$.

1. Find a unitary matrix Θ_i that lower triangularizes the pre-array shown bellow and generates a post-array with positive diagonal entries. Then the entries in the post-array will correspond to

$$\begin{pmatrix} 1 & \lambda^{-1/2} X_i P_{i-1}^{1/2} \\ 0 & \lambda^{-1/2} P_{i-1}^{1/2} \end{pmatrix} \Theta_i = \begin{pmatrix} \gamma_i^{-1/2} & 0 \\ g_i \gamma_i^{-1/2} & P_i^{1/2} \end{pmatrix}. \qquad (24)$$

2. Update the weight vector using

$$B_i = B_{i-1} + [g_i \gamma_i^{-1/2}][\gamma_i^{-1/2}]^{-1}[d_i - X_i B_{i-1}], \qquad (25)$$

where the quantities $\{g_i \gamma_i^{-1/2}, \gamma_i^{-1/2}\}$ are read from the post-array. The computational complexity of this algorithm is $O(M^2)$ operations per iteration. From the original *RLS* algorithm it is known that the relations $\gamma_i = 1 - X_i P_i X_i^*$ and $g_i = P_i X_i^*$ hold [15].

Algorithm 1 was programmed in C and downloaded to the DSP used for real-time control described in section 2. The pre-array is triangularized using the Givens rotation method [14,15].

4.3 Experimental Results

The final test for the adaptive controller developed during this work was done using a disturbance of bandlimited noise. In each sampling step a random number is generated using the minimal random number generator by Park and Miller [21]. This random sequence is filtered through a low-pass filter in order to bandlimit the noise. For disturbance signals with bandwidth smaller than 50 Hz, the adaptive scheme was able to improve the performance achieved by the stabilizing controller described in section 4.1, showing that as expected the system is able to adapt and able to cancel a significant amount of the artificially generated disturbance. It is important to mention that the the disturbance signal w, in Fig. 6, of the physical system is given by $w = (I + PC)^{-1} P_a d_w$, where C denotes the control operator, P denotes the open-loop plant operator, P_a denotes the operator of the actuator and d_w denotes the random sequence out of computer 2 in Fig. 1.

Experimentally, it was observed that ensuring stability in this adaptive scheme became a problem when the disturbance signal was chosen with a bandwidth wider than 50 Hz. A heuristic explanation for this phenomenon is that beyond the bandwidth of the closed-loop frequency response, the identified plant $\hat{G}(z)$ is not accurate enough to model the dynamics of the real system $G(z)$. This issue can be solved by finding certain conditions that ensure robust stability of the real system. The solution will be published in future work.

Table 1. RMS values of output error y.

Axis	Feedback Only	Adaptive Loop
Horizontal ($h-axis$)	0.0148	0.0057
Vertical ($v-axis$)	0.0139	0.0058

Fig. 9 and Fig. 10 show typical error and control sequences in the $h-axis$ and the $v-axis$ respectively. The first 5,000 samples were taken while the system is exited in open loop. The next 5,000 samples were taken while the system was under the feedback control and finally the last 12,000 samples were taken with the adaptive loop on. The numerical speed can be tuned by changing the value of the initial matrix P_{-1} and the forgetting factor λ. As expected, in general the steady-state bound can be achieved very quickly if the order of the adaptive filter is low. In this case the results in the plots were achieved with an adaptive filter of order 5.

The plots depicted in Fig. 9 and Fig. 10 demonstrate that the idea of adaptive inverse control works in this experimentally controlled environment and these results show the applicability of this control scheme to real systems under band-limited frequency disturbances but with changing statistics such as those faced in adaptive optics. The RMS values in the feedback control case and in the adaptive control case are shown in Table 1. These RMS values quantify the effect of the adaptive loop with respect to this specific stabilizing control and it permits us to conclude that this control strategy works as expected. Nevertheless it is necessary to say that possibly there exists a feedback strategy which has a performance comparable to the adaptive loop performance.

5 CONCLUSIONS

This paper has presented the development of a so-called adaptive inverse controller, which was successfully implemented in a real time optical experiment. The potentials of this control scheme for canceling noise have been demonstrated, specially for optical cases, where systems are under disturbance produced by changes in the optical path, atmospheric conditions for instance.

It is possible to conclude that the key issues involved in this solution are the system identification and the adaptive filter algorithm. It is important to mention that the coefficients of the filter $F(z)$ are updated each sampling time using a numerically stable version of the *RLS* algorithm.

During this experimental exploration many issues have arisen opening the door for further research, in particular those

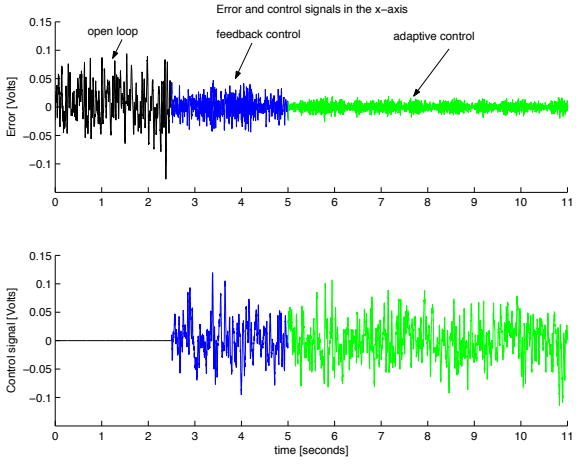

Figure 9. Top: Output error. Bottom: Control command sequence .

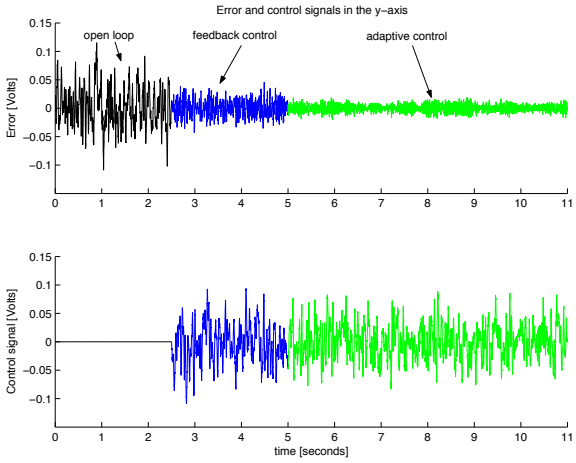

Figure 10. Top: Output error. Bottom: Control command sequence .

relating to stability robustness.

ACKNOWLEDGMENT

The authors are indebted to K. Krishnamoorthy for sharing his experience in C/C++ programming and DSP implementation, and to Y. L. Shue for his useful comments.

REFERENCES

[1] B. Widrow and E. Walach, *Adaptive Inverse Control,* Prentice-Hall, 1996.

[2] B. Widrow and S. D. Stearns, *Adaptive Signal Processing,* Prentice-Hall, 1985.

[3] A.J. Laub, M.T. Heath, C.C. Paige and R.C. Ward, "Computation of system balancing transformations and other applications of simultaneous diagonalization algorithms", *IEEE Transactions on Automatic Control*, vol AC-32, NO. 2, pp. 115-122, Feb. 1987.

[4] G.E. Dullerud and F. Paganini, *A Course in Robust Control Theory,* Springer-Verlag, 2000.

[5] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control,* John Wiley & Sons, 1996

[6] T. Kailath, *Linear Systems,* Prentice-Hall, 1980.

[7] C-T. Chen, *Linear System Theory and Design,* Oxford, 1999.

[8] L. Ljung, *System Identification,* Prentice-Hall, 1987.

[9] P. Van Overschee and B. De Moor, "A unifying theorem for three subspace system identification algorithms", *Automatica*, vol 31, NO. 12, pp. 1853-1864, Dec. 1995.

[10] B. De Moor, M. Moonen, L. Vandenberghe, and J. Vandewalle, "A geometrical approach for the identification of state space models with singular value decomposition", in *Proc. IEEE ICASSP*, vol 4, New York, 1988, pp. 2244-2247.

[11] Y.M. Cho, G. Xu and T. Kailath, "Fast identification of state-space models via exploitation of displacement structure", *IEEE Transactions on Automatic Control*, vol 39, NO. 10, pp. 2004-2017, Oct. 1994.

[12] G. Nakos and D. Joyner, *Linear Algebra with Applications,* Brooks-Cole, 1998.

[13] T. Kailath, A. H. Sayed and B. Hassibi, *Linear Estimation,* Prentice-Hall, 2000.

[14] G.H. Golub and C.F. Van Loan, *Matrix Computations,* The Johns Hopkins University Press, 1989.

[15] A.H. Sayed, *Fundamentals of Adaptive Filtering,* John Wiley & Sons, 2003.

[16] S.T. Alexander, *Adaptive Signal Processing,* Springer-Verlag, 1986.

[17] S.T. Alexander and A.L. Ghirnikar, "A method for recursive least squares filtering based upon an inverse QR decomposition," *IEEE Transactions on Signal Processing*, vol 41, NO. 1, pp. 20-30, Jan. 1993.

[18] S. Haykin, *Adaptive Filter Theory,* Prentice-Hall, 1996.

[19] A.H. Sayed and T. Kailath, "A state-space approach to adaptive RLS filtering", *IEEE Signal Processing Magazine*, vol 11, NO. 3, pp. 18-60, Jul. 1994.

[20] B-S. Kim, S. Gibson and T-C. Tsao, "Adaptive control of a tilt mirror for laser beam steering", in *Proc. American Control Conference*, Boston, MA, 2004, pp. 3417-3421.

[21] W.H. Press, S.A. Teukolsky, W.T. Vetterling and B.P. Flannery, *Numerical Recipes in C,* Cambrige, 1988.