IN-39
8206
p 13

IDENTIFICATION OF LINEAR MULTIVARIABLE SYSTEMS FROM A SINGLE SET OF DATA BY IDENTIFICATION OF OBSERVERS WITH ASSIGNED REAL EIGENVALUES

Minh Phan, Jer-Nan Juang, and Richard Longman

March 1991

NASA

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23665

IDENTIFICATION OF LINEAR MULTIVARIABLE SYSTEMS FROM A SINGLE SET OF DATA BY IDENTIFICATION OF OBSERVERS WITH ASSIGNED REAL EIGENVALUES

Minh Phan[1], Jer-Nan Juang[2] , and Richard W. Longman[3]
NASA Langley Research Center, Hampton, Virginia 23665

ABSTRACT

This paper presents a formulation for identification of linear multivariable systems from a single set of input-output data. The identification method is formulated with the mathematical framework of learning identification, by extension of the repetition domain concept to include shifting time intervals. This approach contrasts with existing learning approaches that require data from multiple experiments. In this method, the system input-output relationship is expressed in terms of an observer, which is made asymptotically stable by an embedded real eigenvalue assignment procedure. Through this relationship, the Markov parameters of the observer are identified. The Markov parameters of the actual system are recovered from those of the observer, and then used to obtain a state space model of the system by standard realization techniques. The basic mathematical formulation is derived, and numerical examples presented to illustrate the proposed method.

INTRODUCTION

The aim of learning system identification is to provide methods to improve identification of system parameters as new data in the form of input-output measurements are available. New information regarding the system characteristics may come from multiple experiments. For system identification of flexible structures, multiple experiments are usually performed to develop or improve a mathematical representation. Due to structural complexity and data irregularities such as slight non-linearities, instrumentation errors, background noises, and repetitive disturbances, multiple tests are used to reduce the irregularity effects on the identified model parameters. The conventional approach is to average data sets from multiple experiments with the hope that the averaged data will reduce the irregularity effects. In

learning identification, new information from successive experiments are used to effectively improve current identification result. In fact, current work in learning identification falls within this conceptual framework. They require the availability of different input-output sets of data from different experiments of generally short duration. However, new information about the system need not come from new experiments, but rather it may be derived from a single experiment of extended duration. This motivates the development of an identification algorithm using a single set of input-output data. Originally motivated by the learning problem, the algorithm is derived using the mathematical framework and techniques of learning control and system identification, Refs. [1-9], and related in general concepts to identification methods proposed in Refs. [10,11]. In particular, this requires the extension of the concept of the repetition domain to shifting time intervals.

In usual identification techniques, the system time domain parameters are determined from input-output data. In learning system identification, however, the parameters of interest to be identified are the Markov parameters. Once this step is completed, standard realization procedures can then be used to realize the system time domain parameters. This shifting emphasis on identification of the Markov parameters offers some advantages. First, there is no ambiguity in the dimensions the Markov parameters. Second, in the learning formulation, the Markov parameters are related to input-output data by a simple linear relation, hence many existing techniques can be applied. Third, for a given linear system, the Markov parameters are unique and invariant with respect to any coordinate transformation of the state vectors.

In this paper, a treatment of this problem is presented to identify a linear multivariable system in state space format by first identifying its Markov parameters. From a single set of input-output data, direct solution of the system state space matrices is non-trivial for a general system of completely unknown characters. However, when additional information about the system is imposed, the mathematical problem becomes simplified. For clarity of exposition, identification procedures are presented for the following three cases of increasing complexities. First, the order of the system is known, and is equal to the number of outputs. Second, the order of the system is

not known, but the system is known to be asymptotically stable. Third, the order is known, but no assumption on the stability of the system is made. The resultant scheme is recursive coupled with an eigenvalue assignment procedure, and is based on techniques developed in the first two cases. The algorithm has an embedded observer structure with pole placement. It is emphasized here that the role of the observer structure is not to provide estimates of the system states for identification, but rather to provide by design a set of asymptotically stable auto-regressive moving average equations whose parameters can be identified. These parameters contain in them the desired information about the actual system. The initial assumption regarding the system order can be later removed by an iterating process, or by knowledge of an upper bound on the effective order of the system.

The Markov parameters of a linear system in state space format are related to the system impulse response functions which can be used in modal identification. From the identified Markov parameters, the modal parameters of the system such as natural frequencies, modal damping, and mode shapes can be deduced by standard realization procedures, e.g., the Eigensystem Realization Algorithm (ERA), Refs. [12,13].

STATEMENT OF THE PROBLEM

Consider a general discrete multivariable linear system expressed in state space format as

$$x(k+1) = Ax(k) + Bu(k)$$
$$y(k) = Cx(k)$$
(1)

where $x \in R^n$, $y \in R^q$, $u \in R^m$. The number of inputs m, and of outputs q are known, the order of the system n is in general assumed not known, and neither are the system matrices A, B, C. Starting from some arbitrary, possibly unknown initial state $x(0)$, the system (1) yields a sequence of outputs $y(i)$, when driven by some known sequence of inputs $u(i-1)$, $i = 1, 2, 3, \dots$ for an extended amount of time. The main objective of the problem is to recover the set of Markov parameters of the system CB, CAB, ..., $CA^{p-1}B$ for a given value of p. For purpose of identification, knowledge of a sufficient number of the system Markov parameters is adequate to deduce a state variable description of the system, hence completely characterizes the system of interest.

MATHEMATICAL FORMULATION

First, the set of known input-output data is divided into intervals of p time steps each. For $i = 1, 2, ..., p$, the solution to (1) is

$$x(i) = A^i x(0) + \sum_{\tau=0}^{i-1} A^{i-\tau-1} Bu(\tau)$$
(2)

For the next interval, $i = p+1, p+2, ..., 2p$, the solution is written as

$$x(i) = A^{i-p} x(p) + \sum_{\tau=0}^{i-(p+1)} A^{i-\tau-(p+1)} Bu(\tau+p)$$
(3)

Introduce a repetition variable j, and a new time step variable k, so that the general solution to Eq. (1) for the first, second, and all subsequent intervals can be expressed as

$$x(i) = A^{i-jp} x(jp) + \sum_{\tau=0}^{i-(jp+1)} A^{i-\tau-(jp+1)} Bu(\tau+jp)$$
(4)

for $i = jp + k$, $j = 0, 1, 2, ..., k = 1, 2, ..., p$. At any time step $i = jp + k$, the state vector $x(i)$ is written as

$$x(jp+k) = x_j(k) \quad and \quad x(jp) = x_j(0)$$
(5)

and similarly for $y(jp+k)$. Associated with each state vector $x(i)$ is an input vector $u(i-1)$ which is also rewritten in terms of j and k as

$$u(i-1) = u_j(k-1)$$
(6)

Applying the definitions in (5) and (6) to (4), one obtains the following description of the system in the repetition domain

$$\underline{x}_{j+1} = \underline{A} x_j(0) + P \underline{u}_{j+1}$$
(7)

where

$$\underline{x} = \left[x^T(1) \quad x^T(2) \quad ... \quad x^T(p) \right]^T$$

$$\underline{u} = \left[u^T(0) \quad u^T(1) \quad ... \quad u^T(p-1) \right]^T$$

$$\underline{A} = \left[A^T \quad \left(A^2 \right)^T \quad ... \quad \left(A^p \right)^T \right]^T$$

$$P = \begin{bmatrix} B & & & O \\ AB & B & & \\ \vdots & \vdots & \ddots & \\ A^{p-1}B & A^{p-2}B & ... & B \end{bmatrix}$$

using $y_j(k) = Cx_j(k)$, the equivalent output description to (7) is

$$\underline{y}_{j+1} = \underline{CA} x_{j+1}(0) + P_o \underline{u}_{j+1}$$
(8)

where

$$\underline{y} = \left[y^T(1) y^T(2) ... y^T(p) \right]^T$$

$$\underline{C} = diag \left[C \quad C \quad ... \quad C \right] \qquad P_o = \underline{C} \ P$$

2

$$P_o = \begin{bmatrix} CB & & & O \\ CAB & CB & & \\ \vdots & \vdots & \ddots & \\ CA^{P-1}B & CA^{P-2}B & \cdots & CB \end{bmatrix}$$

Note that in the repetition domain, the unknown sequence of Markov parameters to be determined, CB, CAB, CA^2B, ... appears naturally. One can make the following general observations before proceeding to the respective algorithms.

Equation (8), in a succinct form, relates the known input-output data in terms of the system Markov parameters, and necessarily the quantities $CAx_{j+1}(0)$. If the states $x_{j+1}(0)$ are known, then given a sufficiently long sequence of input-output data, subject to some appropriate conditions on the input sequence, the system Markov parameters can be uniquely recovered. In general, however, the states $x_{j+1}(0)$ are not known. Equation (8) when written for all available input-output data, represents a set of under-determined equations, with the quantities $x_{j+1}(0)$ as additional unknowns. Therefore, the system Markov parameters cannot be uniquely determined without imposing additional constraints to the set of equations, i.e., without assuming additional knowledge about the system to be identified. In particular, we consider the following cases.

If the order of the system is equal to the number of outputs, and the system is observable, i.e., if C is square and full rank, then with $A_0 = CAC^{-1}$, the output description (8) can be rewritten as

$$y_{j+1} = A_0 \, y_{j+1}(0) + P_0 \, u_{j+1} \tag{9}$$

From Eq. (9) it is clear that subject to some usual condition on the richness of input sequence, from a given set of input-output data of sufficiently long duration, the Markov parameters can be uniquely determined.

There are various methods that can be employed to solve for the Markov parameters from (9). Here, we are particularly concerned with recursive algorithms. Recursive algorithms offer some fundamental advantages for this particular problem. Namely, they are efficient in processing a large amount of data on-line. Furthermore, recursive methods are essentially approximation methods to solving the problem. As such they can be tailored to solve for certain aspect of the problem without solving the full problem as would be required by any exact method, which is some cases, may require exact solution to a difficult non-linear problem. To solve (9) recursively, first let the rows of A_0, P_0 be defined as

$$a_l(k) = \begin{bmatrix} a_{(k-1)n+l,1} & a_{(k-1)n+l,2} & \cdots & a_{(k-1)n+l,n} \end{bmatrix}^T$$

$$p_l(k) = \begin{bmatrix} p_{(k-1)n+l,1} & p_{(k-1)n+l,2} & \cdots & p_{(k-1)n+l,mk} \end{bmatrix}^T$$

where the subscripts on a and p denote the position of the elements in A_0 and P_0 respectively, $l = 1, 2, ..., q$, $k = 1, 2,..., p$. The mk-dimensional input vector is defined as

$$u_{m,j+1}(k-1) = \begin{bmatrix} u_{1,j+1}(0) & ... & u_{m,j+1}(0) & ... & u_{m,j+1}(k-1) \end{bmatrix}^T$$

Making use of the above definitions, Eq. (9) can be expressed as

$$y_{l,j+1}(k) = a_l^T(k)y_{j+1}(0) + p_l^T(k)u_{m,j+1}(k-1) \tag{10}$$

for all (l,k) pairs. Equation (10) represents a set of single-output, multiple-input models in the repetition domain. Grouping the unknown parameters together, Eq. (10) can be rewritten as

$$y_{l,j+1}(k) = p_l^+(k)^T u_{m,j+1}^+(k-1) \tag{11}$$

where

$$p_l^+(k) = \begin{bmatrix} a_l^T(k) & p_l^T(k) \end{bmatrix}^T$$

$$u_{m,j+1}^+(k-1) = \begin{bmatrix} y_{j+1}^T(0) & u_{m,j+1}^T(k-1) \end{bmatrix}^T$$

The parameter estimates of $p_l^+(k)$ at repetition j, denoted by $\hat{p}_{l,j}^+(k)$ can be updated recursively by various methods, for example,

i) The Projection Algorithm:

$$\hat{p}_{l,j}^+(k) = \hat{p}_{l,j-1}^+(k) + a_j u_{m,j}^+(k-1)\left[\frac{y_{l,j}(k) - u_{m,j}^{+T}(k-1)\hat{p}_{l,j-1}^+(k)}{1 + u_{m,j}^{+T}(k-1)u_{m,j}^+(k-1)}\right]$$

for all (l, k) pairs, $0 < a_j < 2$.

ii) The Least Squares Algorithm:

$$\hat{p}_{l,j}^+(k) = \hat{p}_{l,j-1}^+(k) + R_{j-2}(k)u_{m,j}^+(k-1)\Delta\hat{p}_{l,j}^+(k)$$

$$\Delta\hat{p}_{l,j}^+(k) = \left[\frac{y_{l,j}(k) - u_{m,j}^{+T}(k-1)\hat{p}_{l,j-1}^+(k)}{1 + u_{m,j}^{+T}(k-1)R_{j-2}(k)u_{m,j}^+(k-1)}\right]$$

$$R_{j-1}(k) = R_{j-2}(k) - \left[\frac{R_{j-2}(k)u_{m,j}^+(k-1)u_{m,j}^{+T}(k-1)R_{j-2}(k)}{1 + u_{m,j}^{+T}(k-1)R_{j-2}(k)u_{m,j}^+(k-1)}\right]$$

$R_{-1}(k) = \alpha I_{mk \times mk}$, for some large $\alpha > 0$. In actual implementation of this algorithm, covariance resetting may be employed to speed up the convergence of the parameter estimates.

Other recursive algorithms can be easily incorporated within this framework, and are not considered here for the moment. One can analyze the above algorithms to establish the necessary condition on the input sequence to guarantee that the estimated parameters do indeed converge to the true values. For the present development, another class of problems where recursive approximation methods can be used to recover the system Markov parameters is considered.

Consider a special class of problems, where the system matrix A is known to be asymptotically stable. In particular, p can be chosen to be sufficiently large such that A^k can be approximated to be zero for $k \geq p$. Then the sequence of Markov parameters CB, CAB, $CA^P B$, ..., $CA^{p-1}B$ can be recovered by the following method. First, Eq. (8) is rewritten here for clarity and convenience

$$y_{j+1} = \underline{CA}x_{j+1}(0) + P_0 \underline{u}_{j+1} \qquad (12)$$

We seek to write an expression for $x_{j+1}(0)$ from (7). Note that (7) can be rewritten as

$$x_j = \underline{A}x_{j-1}(0) + P \underline{u}_j \qquad (13)$$

which yields

$$x_j(p) = A^P x_{j-1}(0) + \begin{bmatrix} A^{p-1}B & \cdots & AB & B \end{bmatrix} \underline{u}_j \qquad (14)$$

Defining

$$P(p) = \begin{bmatrix} A^{p-1}B & A^{p-2}B & \cdots & AB & B \end{bmatrix}$$

and noting that $x_j(p) = x_{j+1}(0)$, Eq. (14) becomes

$$x_{j+1}(0) = A^P x_j(0) + P(p)\underline{u}_j \approx P(p)\underline{u}_j \qquad (15)$$

after making the approximation $A^P = 0$. Substituting (15) into (12) yields

$$y_{j+1} = \underline{CA}P(p)\underline{u}_j + P_0 \underline{u}_{j+1} \qquad (16)$$
$$= \begin{bmatrix} \underline{CA}P(p) & P_0 \end{bmatrix}\begin{bmatrix} \underline{u}_j \\ \underline{u}_{j+1} \end{bmatrix}$$

Consider the product $\underline{CA}P(p)$

$$\underline{CA}P(p) = \begin{bmatrix} CA \\ CA^2 \\ \vdots \\ CA^{p-1} \\ CA^P \end{bmatrix} \begin{bmatrix} A^{p-1}B & A^{p-2}B & \cdots & AB & B \end{bmatrix} \qquad (17)$$

$$= \begin{bmatrix} CA^P B & CA^{p+1}B & \cdots & CA^2B & CAB \\ CA^{p+1}B & CA^P B & \cdots & CA^3B & CA^2B \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ CA^{2p-1}B & CA^{2p-2}B & \cdots & CA^{p+1}B & CA^P B \end{bmatrix}$$

$$= \begin{bmatrix} 0 & CA^{p-1}B & \cdots & CA^2B & CAB \\ 0 & 0 & CA^{p-1}B & \cdots & CA^2B \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 \end{bmatrix}$$

after neglecting all terms involving p or higher powers of A. Then

$$\begin{bmatrix} CAP(p) & P_0 \end{bmatrix} =$$

$$\begin{bmatrix} 0 & CA^{p-1}B & \cdots & CA^2B & CAB & CB & 0 \\ & 0 & CA^{p-1}B & \cdots & CA^2B & CAB & \ddots & 0 \\ & & \ddots & \cdots & \vdots & \vdots & \vdots & CB & 0 \\ & 0 & & 0 & CA^{p-1}B & CA^{p-2}B & \cdots & CB & 0 \\ & & & & 0 & CA^{p-1}B & \cdots & \cdots & CB \end{bmatrix}$$

Defining

$$P_0(p) = \begin{bmatrix} CA^{p-1}B & \cdots & CA^2B & CB \end{bmatrix} \qquad (18)$$

to be the matrix of Markov parameters to be identified, Eq. (16) thus becomes

$$y_{j+1}(k) = P_0(p)\underline{u}^+_{j+1}(k-1), \quad k = 1, 2, ..., p \qquad (19)$$

where $\underline{u}^+_{j+1}(k-1)$ is a mp-dimensional input vector defined as

$$\underline{u}^+_{j+1}(k-1) = \qquad (20)$$

$$\begin{bmatrix} u^T_j(k) & \cdots u^T_j(p-1), & u^T_{j+1}(0) & u^T_{j+1}(1) & \cdots & u^T_{j+1}(k-1) \end{bmatrix}^T$$

To estimate $P_0(p)$ recursively, the rows of $P_0(p)$ can be recursively updated in parallel. Let $y_l(k)$ denote the l-th output at time step k, and p_l denote the column vector formed by the l-th row of $P_0(p)$, Eq. (19) becomes

$$y_{l,j+1}(k) = p^T_l \underline{u}^+_{j+1}(k-1) \qquad (21)$$

for $l = 1, 2, ..., q$. Each l-th row of $P_0(p)$ is now updated from input-output data associated with the l-th output at time step k, $k = 1, 2, ..., p$ of repetition $j+1, j+2, ...$. The resultant identified parameters are smoothed both in time and in repetition. Hence, the identification scheme makes use of all available input-output data to arrive at one single set of parameter estimates. Once the identification is put into the form as in Eq. (21), a recursive solution is simple, for example, the recursive least squares estimate is

$$\hat{p}^+_{l,j}(k) = \hat{p}^+_{l,j-1}(k) + R_{j-2}(k)\underline{u}^+_j(k-1) \Delta\hat{p}^+_{l,j}(k)$$

$$\Delta\hat{p}^+_{l,j}(k) = \begin{bmatrix} \dfrac{y_{l,j}(k) - \underline{u}^{+^T}_j(k-1)\hat{p}^+_{l,j-1}(k)}{1 + \underline{u}^{+^T}_j(k-1)R_{j-2}(k)\underline{u}^+_j(k-1)} \end{bmatrix}$$

4

$$R_{j-1}(k) = R_{j-2}(k) - \left[\frac{R_{j-2}(k)u_j^+(k-1)u_j^{+T}(k-1)R_{j-2}(k)}{1 + u_j^{+T}(k-1)R_{j-2}(k)u_j^+(k-1)}\right]$$

Note that in the above algorithm, the identified system is required to be asymptotically stable. In the following development, this restriction will be removed.

Consider the system in Eq. (1). It has an observer of the form

$$\hat{x}(i+1) = A\hat{x}(i) + Bu(i) - M [y(i) - \hat{y}(i)] \quad (22)$$
$$= (A+MC)\hat{x}(i) + Bu(i) - My(i)$$

Defining the state estimation error $\tilde{x}(i) = x(i) - \hat{x}(i)$, the equation that governs $\tilde{x}(i)$ is

$$\tilde{x}(i+1) = (A + MC) \tilde{x}(i) \quad (23)$$

If M is chosen such that $A+MC$ is asymptotically stable, then the estimated state will converge to the true state as i tends to infinity. If the system (1) is observable, then M may be chosen so as to place the eigenvalues of $A+MC$ in any desired (symmetric) configuration. The above is a Luenberger observer, a well-known result. However, in our problem, the system matrices A, B, C, and hence M are not known. Multiplying C to both sides of (22) yields

$$\hat{y}(i+1) = C\overline{A}\tilde{x}(i) + CBu(i) - CMy(i) \quad (24)$$

where for simplicity,

$$\overline{A} = A + MC$$

If $\overline{A}$ is asymptotically stable, then for large i, $\hat{x}(i)$ tends to $x(i)$, hence $\hat{y}(i)$ tends to $y(i)$, then Eq. (24) becomes

$$y(i+1) = C\overline{A}x(i) + CBu(i) - CMy(i) \quad (25)$$
$$= C\overline{A}x(i) + C\overline{B}v(i)$$

where

$$\overline{B} = [B \quad -M] \quad \text{and} \quad v(i) = \begin{bmatrix} u(i) \\ y(i) \end{bmatrix}$$

If $\overline{A}$ is asymptotically stable, then by the method developed above for asymptotically stable system, the Markov parameters of the observer equation, $C\overline{B}$, $C\overline{A}\overline{B}$, ..., $C\overline{A}^{p-1}\overline{B}$ can be identified. This can be easily seen by noting that from (22) and (25) for asymptotically stable $\overline{A}$ and large i, the system in (22) becomes

$$x(i+1) = \overline{A}x(i) + \overline{B}v(i) \quad (26)$$
$$y(i) = Cx(i)$$

which when compared to (1), $\overline{A}$, $\overline{B}$, $v(i)$ play the same roles as A, B, $u(i)$, respectively.

It is noted here that since Eq. (26) is derived from (22), hence it can be interpreted in terms of an observer equation. But in fact it is an exact relation which always holds true for any matrix M. Equation (26) can be derived by simply adding and subtracting the product $My(i)$ to the right hand side of Eq. (1). Recall that if the system (1) is observable, then there exists at least a matrix M such that the eigenvalues of $\overline{A}$ can be assigned in any arbitrary symmetric configuration. Such an M is not necessarily unique, but this poses no restriction to the present problem. In the following development, the additional freedom introduced by the matrix M will now be used to derive a recursive algorithm to identify the Markov parameters of the system in (26), i.e., $C\overline{B}$, $C\overline{A}\overline{B}$, ..., $C\overline{A}^{p-1}\overline{B}$, and at the same time place the eigenvalues of $\overline{A}$ in desired asymptotically stable locations. Note, however, that the identified parameters are the observer system Markov parameters, but they will be used to recover the desired actual system Markov parameters. This will be done in later sections.

Applying the technique developed before for asymptotically stable system to the system in (26), assuming for the moment that $\overline{A}$ is asymptotically stable, yields the corresponding version of (19)

$$y_{j+1}(k) = \overline{P}_o(p)y_{j+1}^+(k+1), \quad k = 1, 2, ..., p \quad (27)$$

$\overline{P}_o(p)$ and $y_{j+1}^+(k-1)$ are defined as

$$\overline{P}_o(p) = \left[C\overline{A}^{p-1}\overline{B} \quad \cdots \quad C\overline{A}\overline{B} \quad C\overline{B} \right] \quad (28)$$

and

$$y_{j+1}^+(k-1) = \quad (29)$$
$$\left[v_j^T(k) \quad \cdots v_j^T(p-1), \quad u_{j+1}^T(0) \quad v_{j+1}^T(1) \quad \cdots \quad v_{j+1}^T(k-1) \right]^T$$

where

$$v_j^T(k) = \left[u_j^T(k) \quad y_j^T(k) \right] \quad (30)$$

The eigenvalue assignment procedure can be derived, first by noting that for desired (real) eigenvalues of $\overline{A}$, we have for some T

$$\overline{A} = T^{-1}\Lambda T \quad (31)$$

where

$$A = \begin{bmatrix} \lambda_1 & & & 0 \\ & \lambda_2 & & \\ & & \ddots & \\ 0 & & & \lambda_n \end{bmatrix} \tag{32}$$

Let $\lambda_1, \lambda_2, ..., \lambda_n$ denote the desired eigenvalues of $\overline{A}$ to be selected a priori. Then the product $C\overline{A}^k\overline{B}$ becomes

$$\begin{aligned} C\overline{A}^k\overline{B} &= CT^{-1}\Lambda^k TB \\ &= C^*\Lambda^k B^* \end{aligned} \tag{33}$$

If the elements of C^* and B^* are written explicitly as

$$C^* = \begin{bmatrix} c_{11}^* & c_{12}^* & \cdots & c_{1n}^* \\ c_{21}^* & c_{22}^* & \cdots & c_{2n}^* \\ \vdots & \vdots & \ddots & \vdots \\ c_{q1}^* & c_{q2}^* & \cdots & c_{qn}^* \end{bmatrix}$$

$$B^* = \begin{bmatrix} b_{11}^* & b_{12}^* & \cdots & b_{1m}^* & -m_{11}^* & -m_{12}^* & \cdots & -m_{1q}^* \\ b_{21}^* & b_{11}^* & \cdots & b_{2m}^* & -m_{21}^* & -m_{22}^* & \cdots & -m_{2q}^* \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ b_{n1}^* & b_{n2}^* & \cdots & b_{nm}^* & -m_{n1}^* & -m_{n2}^* & \cdots & -m_{nq}^* \end{bmatrix}$$

then in general, the l-th row of the matrix product in (33), denoted $\overline{p}_l^{(k)}$, is equal to

$$\overline{p}_l^{(k)} = [\; \sum_{i=1}^{n} \lambda_i^k c_{li}^* b_{i1}^* \;,\; \sum_{i=1}^{n} \lambda_i^k c_{li}^* b_{i2}^* \;,\; \cdots \;,\; \sum_{i=1}^{n} \lambda_i^k c_{li}^* b_{im}^* \;,$$
$$-\sum_{i=1}^{n} \lambda_i^k c_{li}^* m_{i1}^* \;,\; -\sum_{i=1}^{n} \lambda_i^k c_{li}^* m_{i2}^* \;,\; \cdots \;,\; -\sum_{i=1}^{n} \lambda_i^k c_{li}^* m_{iq}^* \;]$$

Let the l-th row of $P_o(p)$ of equation (27) be denoted as $\overline{p}_l^T$, then the l-th output at time step k of repetition $j+1$, denoted by $y_{l,j+1}(k)$, can be expressed as

$$y_{l,j+1}(k) = \overline{p}_l^T \underline{y}_{j+1}^+(k-1) \tag{34}$$

where

$$\overline{p}_l^T = \begin{bmatrix} \overline{p}_l^{(p-1)} & \overline{p}_l^{(p-2)} & \cdots & \overline{p}_l^{(1)} & \overline{p}_l^{(0)} \end{bmatrix}$$

Note that each $\overline{p}_l^{(k)}$ is a row vector, each with $m+q$ sums. Each sum has n terms. Together each $\overline{p}_l^{(k)}$ has $n(m+q)$ terms of the products $c_{ij}^* b_{ik}^*$ and $c_{ij}^* m_{ik}^*$. First, write for m inputs and q outputs

$$u_j^T(k) = \begin{bmatrix} u_{1,j}(k) & u_{2,j}(k) & \cdots & u_{m,j}(k) \end{bmatrix}$$

and

$$y_j^T(k) = \begin{bmatrix} y_{1,j}(k) & y_{2,j}(k) & \cdots & y_{q,j}(k) \end{bmatrix}$$

Note that

$$u_{1,j}(k) \sum_{i=1}^{n} \lambda_i^{p-1} c_{li}^* b_{i1}^* =$$

$$\begin{bmatrix} c_{l1}^* b_{11}^* & c_{l2}^* b_{21}^* & \cdots & c_{ln}^* b_{n1}^* \end{bmatrix} \begin{bmatrix} \lambda_1^{p-1} \\ \lambda_2^{p-1} \\ \vdots \\ \lambda_n^{p-1} \end{bmatrix} u_{1,j}(k)$$

With the following simplifying definitions of the unknown parameters

$$\begin{aligned} \alpha_{lr} &= \begin{bmatrix} c_{l1}^* b_{1r}^* & c_{l2}^* b_{2r}^* & \cdots & c_{ln}^* b_{nr}^* \end{bmatrix} \\ \beta_{lr} &= \begin{bmatrix} c_{l1}^* m_{1s}^* & c_{l1}^* m_{2s}^* & \cdots & c_{ln}^* m_{ms}^* \end{bmatrix} \end{aligned} \tag{35}$$

$$r = 1, 2,..., m \;;\quad s = 1, 2,..., q$$

and a vector of the assigned eigenvalues

$$\lambda^{(p-1)} = \begin{bmatrix} \lambda_1^{p-1} & \lambda_2^{p-1} & \cdots & \lambda_n^{p-1} \end{bmatrix}^T \tag{36}$$

Equation (34) can then can be expressed as

$$y_{l,j+1}(k) =$$

$$\alpha_{l1}^T \left[\sum_{\tau=k}^{p-1} \lambda^{(k-\tau+p-1)} u_{1,j}(\tau) + \sum_{\tau=0}^{k-1} \lambda^{(k-\tau-1)} u_{1,j+1}(\tau) \right]$$

$$+ \alpha_{l2}^T \left[\sum_{\tau=k}^{p-1} \lambda^{(k-\tau+p-1)} u_{2,j}(\tau) + \sum_{\tau=0}^{k-1} \lambda^{(k-\tau-1)} u_{2,j+1}(\tau) \right]$$

$$\vdots$$

$$+ \alpha_{lm}^T \left[\sum_{\tau=k}^{p-1} \lambda^{(k-\tau+p-1)} u_{m,j}(\tau) + \sum_{\tau=0}^{k-1} \lambda^{(k-\tau-1)} u_{m,j+1}(\tau) \right]$$

$$- \beta_{l1}^T \left[\sum_{\tau=k}^{p-1} \lambda^{(k-\tau+p-1)} y_{1,j}(\tau) + \sum_{\tau=0}^{k-1} \lambda^{(k-\tau-1)} y_{1,j+1}(\tau) \right]$$

$$- \beta_{l2}^T \left[\sum_{\tau=k}^{p-1} \lambda^{(k-\tau+p-1)} y_{2,j}(\tau) + \sum_{\tau=0}^{k-1} \lambda^{(k-\tau-1)} y_{2,j+1}(\tau) \right]$$

$$\vdots$$

$$- \beta_{lq}^T \left[\sum_{\tau=k}^{p-1} \lambda^{(k-\tau+p-1)} y_{q,j}(\tau) + \sum_{\tau=0}^{k-1} \lambda^{(k-\tau-1)} y_{q,j+1}(\tau) \right]$$

6

or

$$y_{l,j+1}(k) =$$

$$\sum_{\zeta=1}^{m} \alpha_{l\zeta}^T \left[\sum_{\tau=k}^{p-1} \lambda^{(k-\tau+p-1)} u_{\zeta,j}(\tau) + \sum_{\tau=0}^{k-1} \lambda^{(k-\tau-1)} u_{\zeta,j+1}(\tau) \right]$$

$$- \sum_{\xi=1}^{q} \beta_{l\xi}^T \left[\sum_{\tau=k}^{p-1} \lambda^{(k-\tau+p-1)} y_{\xi,j}(\tau) + \sum_{\tau=0}^{k-1} \lambda^{(k-\tau-1)} y_{\xi,j+1}(\tau) \right]$$

This equation can be simplified further by making the following definitions

$$\phi_{r,j+1}(k-1) = \sum_{\tau=k}^{p-1} \lambda^{(k-\tau+p-1)} u_{r,j}(\tau) + \sum_{\tau=0}^{k-1} \lambda^{(k-\tau-1)} u_{r,j+1}(\tau)$$

$$\psi_{s,j+1}(k-1) = \sum_{\tau=k}^{p-1} \lambda^{(k-\tau+p-1)} y_{s,j}(\tau) + \sum_{\tau=0}^{k-1} \lambda^{(k-\tau-1)} y_{s,j+1}(\tau)$$

Then it becomes

$$y_{l,j+1}(k) = \alpha_{l1}^T \phi_{1,j+1}(k-1) + \cdots + \alpha_{lm}^T \phi_{m,j+1}(k-1)$$
$$- \beta_{l1}^T \psi_{1,j+1}(k-1) - \cdots - \beta_{lq}^T \psi_{q,j+1}(k-1)$$ (37)

$$= \begin{bmatrix} \alpha_{l1}^T & \cdots & \alpha_{lm}^T & -\beta_{l1}^T & \cdots & -\beta_{lq}^T \end{bmatrix} \begin{bmatrix} \phi_{1,j+1}(k-1) \\ \vdots \\ \phi_{m,j+1}(k-1) \\ \psi_{1,j+1}(k-1) \\ \vdots \\ \psi_{q,j+1}(k-1) \end{bmatrix}$$

By making obvious definitions for γ_l^T, and $\Gamma_{j+1}(k-1)$, the above equation becomes simply

$$y_{l,j+1}(k) = \gamma_l^T \Gamma_{j+1}(k-1) \qquad (38)$$

for $l = 1, 2, ..., q$. The above set of equations is in linear form, with unknown (time-invariant) observer parameter vector γ_l and known "input" vector $\Gamma_{j+1}(k-1)$, therefore γ_l can be solved for in one step, or recursively. Any appropriate method to solve linear equations can be used. For example, the recursive least squares solution to (38) can be written down immediately for the l-th row of $C^* A^k B^*$, which corresponds to the l-th output, as

$$\hat{\gamma}_{l,j}(k) = \hat{\gamma}_{l,j-1}^+(k) + R_{j-2}(k)\Gamma_j(k-1)\Delta\hat{\gamma}_{l,j}(k) \qquad (39)$$

$$\Delta\hat{\gamma}_{l,j}(k) = \left[\frac{y_{l,j}(k) - \Gamma_j^T(k-1)\hat{\gamma}_{l,j-1}^+(k)}{1 + \Gamma_j^T(k-1)R_{j-2}(k)\Gamma_j(k-1)} \right]$$

$$R_{j-1}(k) = R_{j-2}(k) - \left[\frac{R_{j-2}(k)\Gamma_j(k-1)\Gamma_j^T(k-1)R_{j-2}(k)}{1 + \Gamma_j^T(k-1)R_{j-2}(k)\Gamma_j(k-1)} \right]$$

The above algorithm in (39) identifies the observer parameters γ_l which consists of $\alpha_{l1}, \alpha_{l2}, ..., \alpha_{lm}, \beta_{l1}, \beta_{l2}, ..., \beta_{lq}$. Each $\alpha_{l1}, \alpha_{l2}, \beta_{l1}, \beta_{l2}, ...$ in turn contains certain products of the elements of C^* and B^*, e.g., $c_{l1}^* b_{11}^*, c_{lm}^* m_{11}^*, ...$ .These products together with the assigned eigenvalues $\lambda_1, \lambda_2, ..., \lambda_n$ can be used to construct the l-th row of the observer Markov parameters $\overline{CA^k B}$. Thus any row of the observer Markov parameter for any given k can be computed. Hence a complete set of the parameters $\overline{CB}, \overline{CAB}, ..., \overline{CA^{p-1}B}$ can be identified. Furthermore, note that Eq. (38) is true for all $k = 0, 1, 2, ..., p$; and $j = 1, 2, 3, ...$ hence the identified γ_l, which is then used to recover the l-th row of $CA^k B$ is smoothed over all k and j. Note that Eq. (27) is simply an ARMA model of the system, and the observer Markov parameters are precisely the matrix coefficients of this ARMA model. Through the embedded eigenvalue assignment procedure, the ARMA model is made asymptotically stable by design. It is the stabilizing property of this model that allows a straight forward application of existing parameter estimation techniques to identify its matrix coefficients. What remains to be done is to recover the actual Markov parameters of the open loop system from these identified matrix coefficients.

In the following a procedure to recover the actual system Markov parameters is presented for the general case of multiple-input, multiple-output systems. First, recall that $\overline{A} = A + MC, \overline{B} = [B, -M]$, and the Markov parameters of the observer system are now known from the identification algorithm developed above. For ease of presentation, the following definitions are made

$$\overline{Y}_0 = C\overline{B} = \begin{bmatrix} CB, & -CM \end{bmatrix}$$
$$= \begin{bmatrix} \overline{Y}_0^{(1)}, & \overline{Y}_0^{(2)} \end{bmatrix}$$

$$\overline{Y}_1 = C\overline{AB} = \begin{bmatrix} \overline{Y}_1^{(1)}, & \overline{Y}_1^{(2)} \end{bmatrix} \qquad (40)$$

and similarly for $\overline{Y}_2, \overline{Y}_3, ...$ From the first equation in (40), the first Markov parameter of the actual system, $Y_0 = CB$, is simply

$$Y_0 = \overline{Y}_0^{(1)} \qquad (41)$$

Next, consider the product $C\overline{A}\,\overline{B}$

$$C\overline{A}\,\overline{B} = C(A + MC)\begin{bmatrix} B, & -M \end{bmatrix}$$
$$= \begin{bmatrix} CAB + CMCB, & -(CAM + CMCM) \end{bmatrix}$$
$$= \begin{bmatrix} \overline{Y}_1^{(1)}, & \overline{Y}_1^{(2)} \end{bmatrix}$$

7

Hence, the second system Markov parameter $Y_1 = CAB$ can be computed as

$$Y_1 = CAB \tag{42}$$

$$= \overline{Y}_1^{(1)} - \overline{Y}_0^{(2)} Y_0$$

To obtain an expression for $Y_2 = CA^2B$, consider the product $\overline{Y}_2 = C\overline{A}^2\overline{B}$

$$C\overline{A}^2\overline{B} = C(A+MC)^2 [B \ , -M]$$

$$= \left[\overline{Y}_2^{(1)} \ , \ \overline{Y}_2^{(2)} \right]$$

Thus

$$\overline{Y}_2^{(1)} = C(A + MC)^2 B$$

$$= CA^2B + (CM)CAB + (CAM + CMCM)CB$$

Hence

$$Y_2 = CA^2B \tag{43}$$

$$= \overline{Y}_2^{(1)} + \overline{Y}_0^{(2)} Y_1 + \overline{Y}_1^{(2)} Y_0$$

Similarly, an expression for Y_3 can be obtained from $\overline{Y}_3$ as

$$Y_3 = CA^3B \tag{44}$$

$$= \overline{Y}_3^{(1)} + \overline{Y}_0^{(2)} Y_2 + \overline{Y}_1^{(2)} Y_1 + \overline{Y}_2^{(2)} Y_0$$

By induction, the general solution for the actual system Markov parameters $Y_k = CA^kB$ can be written as a convolution product of its previous values and the Markov parameters of the observer system as

$$Y_k = CA^kB \tag{45}$$

$$= \overline{Y}_k^{(1)} + \sum_{i=0}^{k-1} \overline{Y}_i^{(2)} Y_{k-1-i}$$

Once the Markov parameters are identified, a realization procedure such as ERA can be applied to obtain a realization of the system matrices. Physical aspects of the model such as natural frequencies, damping ratios, mode shapes can then be found.

EXAMPLES

In this section numerical examples are presented to illustrate the above developed identification algorithm. Recall that in this formulation, for the multivariable case, all the couplings between the outputs are expressed in the "input" vectors. Hence for each output, the multiple-input multiple-output (MIMO) problem is treated as a multiple-

input single-output (MISO) problem. The results are then combined for MIMO identification. Multiple-input single-output problems, however, are essentially similar to single-input single-output (SISO) problems, therefore the algorithm can be best illustrated by first considering the SISO case.

EXAMPLE 1: Consider a fourth-order single-input single-output system in observable canonical form driven by a random input sequence

$$A = \begin{bmatrix} 0 & 0 & 0 & -0.305 \\ 1 & 0 & 0 & -0.110 \\ 0 & 1 & 0 & -0.110 \\ 0 & 0 & 1 & 0.000 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \end{bmatrix}, \quad C = [0 \ 0 \ 0 \ 1]$$

The first 30 Markov parameters are to be identified. The number of time step in each interval is thus chosen to be $p = 30$. With a 90-time step input-output history, this results in 3 available repetitions for learning. In this example, the eigenvalues chosen are ± 0.6 and ± 0.7. In fact, any real, distinct value λ_i with magnitude less than one can be used as long as λ_i^p is sufficiently small to allow for accurate identification. In this example, the standard least squares method is employed, and convergence of the identified observer parameters is shown in Figure 1. With these parameters, the observer Markov parameters are constructed, and then used to recover correctly the system Markov parameters. This is shown in Table 1.

EXAMPLE 2: The algorithm is applied to identify a state space model of a mass-spring-dashpot system. This is a sixth-order system with three inputs and three outputs with the discrete-time system matrices given as

$$A = \begin{bmatrix} 0.9691 & 0.0154 & 0.0001 & 0.2120 & 0.0014 & 0.0000 \\ 0.0154 & 0.9690 & 0.0155 & 0.0014 & 0.2139 & 0.0014 \\ 0.0000 & 0.0077 & 0.9768 & 0.0000 & 0.0007 & 0.2127 \\ -0.2817 & 0.1395 & 0.0009 & 0.9458 & 0.0180 & 0.0001 \\ 0.1401 & -0.2838 & 0.1412 & 0.0180 & 0.9134 & 0.0182 \\ 0.0005 & 0.0699 & -0.2122 & 0.0000 & 0.0091 & 0.9544 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.0232 & 0.0001 & 0.0000 \\ 0.0001 & 0.0233 & 0.0001 \\ 0.0000 & 0.0000 & 0.0232 \\ 0.2120 & 0.0014 & 0.0000 \\ 0.0014 & 0.2139 & 0.0014 \\ 0.0000 & 0.0007 & 0.2127 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

The first twenty five Markov parameters are to be identified, thus p is chosen to be 25. Using a single time history of 100 time steps under random input excitation, this yields four repetitions for learning. Assuming for the moment, the true order of the system, $n = 6$, is known. The eigenvalues for the observer are placed at ± 0.2, ± 0.3, and ± 0.4. The observer equation parameters corresponding to each output are first identified, and then the results combined to recover the actual system Markov parameters. Figure 2 shows the convergence of some arbitrarily selected observer parameter estimates corresponding to the first output. Results for the two

8

remaining outputs are similar, and not shown here. The algorithm correctly identifies all desired Markov parameters, each is a 3 x 3 matrix.

The Eigensystem Realization Algorithm (ERA) is then applied to decompose the identified Markov parameters to obtain a set of realized system matrices which is equivalent to the set (A, B, C) above. They are listed here in that order as follows.

$$
\begin{bmatrix}
0.9971 & 0.0561 & 0.0238 & 0.0606 & -0.0049 & 0.1166 \\
-0.0553 & 0.9406 & 0.1879 & -0.1115 & -0.0291 & 0.0289 \\
-0.0357 & -0.1762 & 0.9878 & -0.0341 & 0.1757 & 0.0753 \\
-0.0497 & 0.1321 & 0.0173 & 0.9725 & 0.2365 & -0.0193 \\
0.0113 & 0.0232 & -0.1827 & -0.2027 & 0.9207 & 0.0923 \\
-0.1178 & 0.0050 & -0.0703 & 0.0326 & -0.1105 & 0.9600
\end{bmatrix}
$$

$$
\begin{bmatrix}
0.2015 & 0.2770 & 0.2140 \\
-0.1835 & -0.0138 & 0.3051 \\
0.0380 & -0.1269 & 0.2732 \\
0.2712 & 0.0430 & -0.1038 \\
0.1925 & -0.2085 & 0.0613 \\
0.1514 & 0.2515 & 0.1813
\end{bmatrix}
$$

$$
\begin{bmatrix}
0.1804 & -0.2296 & 0.2651 & -0.0293 & -0.1461 & -0.1934 \\
0.2757 & -0.1481 & -0.1093 & -0.3034 & 0.1177 & -0.1246 \\
0.2309 & 0.2355 & -0.1408 & 0.1200 & 0.0625 & -0.2810
\end{bmatrix}
$$

Now, consider the identification algorithm used in Example 2, but this time the system order is assumed incorrectly. First, the system is over-estimated to be of eighth-order. This calls for two additional eigenvalues in the algorithm, here chosen to be ± 0.25. Note that since the assumed order of the system is higher, there are more parameters to be identified. However, at the final step, the algorithm correctly identifies the true order of the system and recovers all desired Markov parameters of the open loop system. This means that over-parameterization does not affect the final result. Second, the system is under-estimated to be only of second-order. This reduces the number of parameters to be identified substantially. Yet, again, the true order of the system and all the Markov parameters are recovered correctly at the final step. This indicates that under-estimation of the system order does not lead automatically to incorrect identification. Correct results are also obtained when the order of the system in this example is under-estimated to be 5, 4, or 3. The algorithm fails when the order is assumed to be 1, which is obviously an erroneous over-simplification. The identified observer parameters are different under different order assumptions, yet they yield the same result in the final reconstruction of the actual system Markov parameters.

EXAMPLE 3: In the absence of noise, the identified values are found to be practically identical to the actual values. In the deterministic theory, this error can be made arbitrarily small simply by choosing p to be sufficiently large, and the eigenvalues λ_i sufficiently small such that the approximation in deriving Eq. (27) is valid. With regard to identification accuracy in the presence of noise, more effective recursive identification methods, e.g., the

instrumental variable method, can be used to replace the standard least squares method at one step in this identification procedure. For example, consider the case in Example 1. When the output data is corrupted by about 12% measurement noise, the standard recursive least squares method leads to biased results which can be corrected by using the instrumental variable method instead. Figure 3 shows this improvement in the accuracy of some arbitrarily selected identified observer parameter estimates.

CONCLUSIONS

This paper formulates an algorithm for identification of linear multivariable systems from a one set of input-output data. Instead of identifying the system directly, the proposed scheme identifies an observer for the system, whose poles can be assigned by an embedded real eigenvalue assignment procedure. Recursive techniques are used to estimate the matrix coefficients of an auto-regressive moving average model formed from this observer. These matrix coefficients are precisely its Markov parameters. From the identified observer Markov parameters, the true system Markov parameters are recovered through simple algebraic relations, and then used to obtain a realization of the system of interest. For modal identification, the modal parameters such as natural frequencies, modal damping, and mode shapes of the open loop system can then be found. Preliminary results indicates that the deterministic algorithm is fast and accurate. Stochastic aspects of the procedure is currently under investigation.

REFERENCES

[1] J. Nagumo and A. Noda, "A Learning Method for System Identification," *IEEE Transactions*, Vol. AC-12, pp. 282-287, 1967.

[2] M. Phan and R.W. Longman, "Learning System Identification," *Proceedings of the 20th Annual Pittsburgh Conference on Modelling and Simulation*, May 1989.

[3] J.-N. Juang, L.G. Horta, M. Phan and R.W. Longman, "Learning System Identification for Flexible Structures," *Proceedings of the 2nd USAF/NASA Workshop on System Identification and Health Monitoring of Precision Space Structures*, California Institute of Technology, Pasadena, California, March 1990.

[4] M. Phan and R.W. Longman, "A Mathematical Theory of Learning Control for Linear Discrete Multivariable Systems," *Proceedings of the AIAA/AAS Astrodynamics Specialist Conference*, Minneapolis, Minnesota, August 1988.

[5] M. Phan and R.W. Longman, "Liapunov Based Model Reference Learning Control," *Proceedings of the 26th Annual Allerton Conference on Communication, Control, and Computing*, Monticello, Illinois, September 1988.

[6] M. Phan and R.W. Longman, "Indirect Learning Control with Guaranteed Stability," *Proceedings of the 1989 Johns Hopkins Conference on Information Sciences and Systems*, Baltimore, Maryland, March 1989.

[7] M. Phan, Y. Wei, L.G. Horta, J.-N. Juang, and R.W. Longman, "Learning Control for Slewing of a Flexible Panel," *Proceedings of the 1989 VPI&SU/AIAA Symposium on Dynamics and Control of Large Structures*, May 1989.

[8] M. Phan, *A Mathematical Theory of Learning Control*, Ph.D. Dissertation, Columbia University, New York, New York, May 1989.

[9] M. Phan, J.-N. Juang, and R.W. Longman, "Recent Developments in Learning Control and System Identification for Robots and Structures," *Proceedings of the 1990 Conference on Dynamics of Flexible Structures in Space*, Cranfield, England, May 1990.

[10] M. Phan, R.W. Longman, and J.-N. Juang, "Indirect Repetitive Control for Linear Discrete Multivariable Systems, " *Proceedings of the 27th Annual Allerton Conference on Communication, Control, and Computing*, Monticello, Illinois, September 1989.

[11] C.W. Chen, J.-K. Huang, M. Phan, and J.-N. Juang, "Integrated System Identification and Modal State Estimation for Control of Large Flexible Space Structures," to appear in the *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, Portland, Oregon, August 1990.

[12] B.L. Ho and R.E. Kalman, "Effective Construction of Linear State Variable Models from Input/Output Data," *Proceedings of the 3rd Annual Allerton Conference on Circuits and Systems Theory*, Monticello, Illinois, 1965.

[13] J.-N. Juang and R.S. Pappa, "An Eigensystem Realization Algorithm for Modal Parameter Identification and Model Reduction," *Journal of Guidance, Control, and Dynamics*, Vol. 8, Sept.-Oct. 1985.

APPENDIX

Table 1: *Identification of Markov Parameters of a Single-Input Single-Output System*

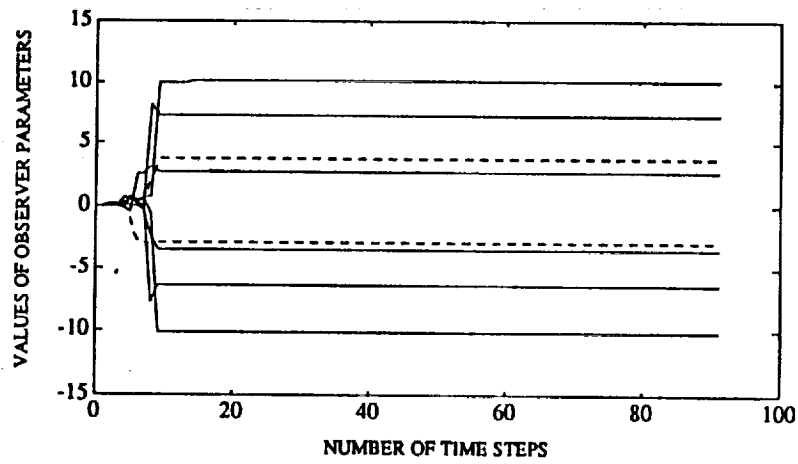| IDENTIFIED OBSERVER PARAMETERS	RECONSTRUCTED OBSERVER MARKOV PARAMETERS		IDENTIFIED MARKOV PARAMETERS	ACTUAL MARKOV PARAMETERS
-10.1021	1.0000	0.0000	0.0003	0.0003
7.3337	1.0000	-0.9600	-0.0005	-0.0005
10.0710	0.8499	-0.1100	-0.0012	-0.0012
-6.3025	1.8500	-0.9446	-0.0015	-0.0015
-3.4628	0.5460	-0.0935	0.0000	0.0000
2.6168	1.3961	-0.6336	0.0023	0.0023
3.7143	0.3142	-0.0601	0.0030	0.0030
-2.8682	0.8603	-0.3719	0.0031	0.0031
	0.1707	-0.0346	-0.0022	-0.0022
	0.4850	-0.2044	-0.0080	-0.0080
	0.0897	-0.0188	-0.0062	-0.0062
	0.2605	-0.1081	-0.0051	-0.0051
	0.0461	-0.0099	0.0114	0.0114
	0.1359	-0.0558	0.0239	0.0239
	0.0234	-0.0051	0.0077	0.0077
	0.0695	-0.0284	0.0054	0.0054
	0.0117	-0.0026	-0.0422	-0.0422
	0.0351	-0.0143	-0.0652	-0.0652
	0.0059	-0.0013	0.0135	0.0135
	0.0176	-0.0071	0.0009	0.0009
	0.0029	-0.0006	0.1331	0.1331
	0.0088	-0.0035	0.1654	0.1654
	0.0014	-0.0003	-0.1519	-0.1520
	0.0043	-0.0018	-0.0079	-0.0079
	0.0007	-0.0002	-0.3787	-0.3787
	0.0021	-0.0009	-0.4029	-0.4029
	0.0003	-0.0001	0.7800	0.7800
	0.0011	-0.0004	-0.1100	-0.1101
	0.0002	-0.0000	1.0000	1.0000
	0.0005	-0.0002	1.0000	1.0000

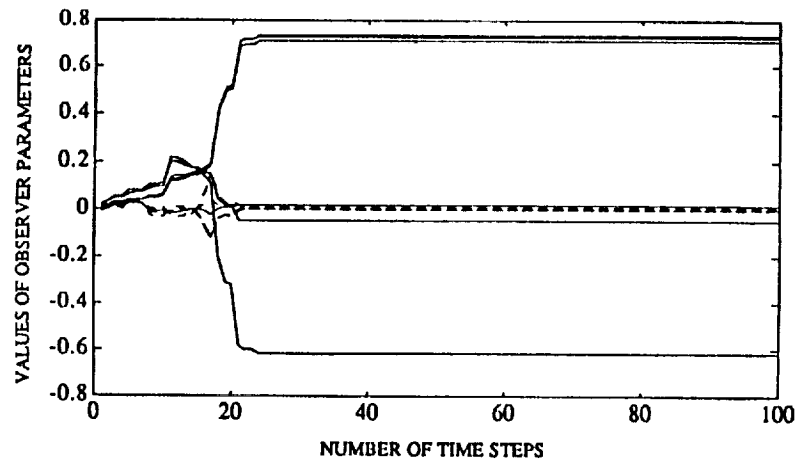**Figure 1:** Convergence of identified observer parameters of a single-input single-output system

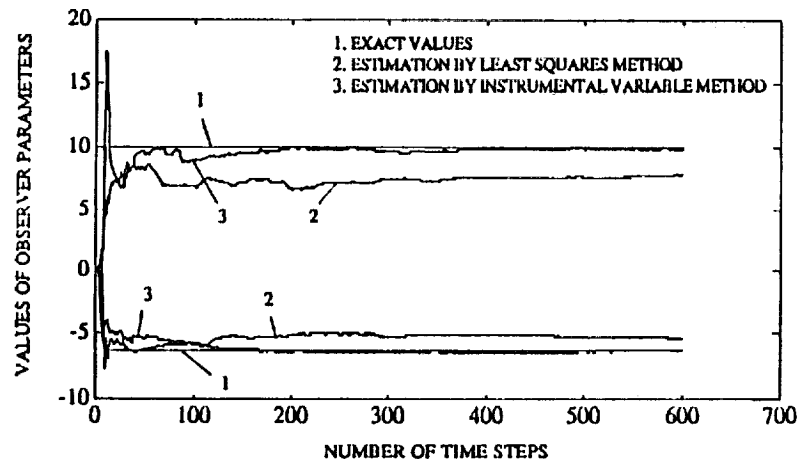**Figure 2:** Convergence of identified observer parameters of a multiple-input multiple-output system

1. EXACT VALUES
2. ESTIMATION BY LEAST SQUARES METHOD
3. ESTIMATION BY INSTRUMENTAL VARIABLE METHOD

Figure 2: Convergence of observer parameters in the presence of noise using standard least-squares and instrumental variable methods

NASA
National Aeronautics and
Space Administration

Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA TM-102669		

4. Title and Subtitle	5. Report Date
Identification of Linear Multivariable Systems From a Single Set of Data by Identification of Observers With Assigned Real Eigenvalues	March 1991
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
Minh Phan, Jer-Nan Juang, and Richard W. Longman	
	10. Work Unit No.
	590-14-21-01

9. Performing Organization Name and Address	11. Contract or Grant No.
NASA Langley Research Center Hampton, VA 23665-5225	
	13. Type of Report and Period Covered

12. Sponsoring Agency Name and Address	Technical Memorandum
National Aeronautics and Space Administration Washington, DC 20546-0001	14. Sponsoring Agency Code

15. Supplementary Notes

Presented at the 32nd Structures, Structural Dynamics and Materials Conference Baltimore, MD, April 8-10, 1991.
Minh Phan and Richard Longman: National Research Council Research Associates at Langley Research Center.

16. Abstract

This paper presents a formulation for identification of linear multivariable from a single set of input-output data. The identification method is formulated with the mathematical framework of learning identification, by extension of the repetition domain concept to include shifting time intervals. This method contrasts with existing learning approaches that require data from multiple experiments. In this method, the system input-output relationship is expressed in terms of an observer, which is made asymptotically stable by an embedded real eigenvalue assignment procedure. Through this relationship, the Markov parameters of the ovserver are identified. The Markov parameters of the actual system are recovered from those of the observer, and then used to obtain a state space model of the system by standard realization techniques. The basic mathematical formulation is derived, and numerical examples presented to illustrate.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
System Identification Observer Identification Observer Pole Placement	Unclassified--Unlimited Subject Category 39

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of pages	22. Price
Unclassified	Unclassified	12	A03