JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 20, No. 6, November–December 1997

Identification of a Multistep-Ahead Observer and Its Application to Predictive Control

Ryoung K. Lim* and Minh Q. Phan†
Princeton University, Princeton, New Jersey 08544

State estimation is a fundamental component of modern control theory. In discrete-time format, the standard state estimator (observer) is one step ahead. It provides one-step-ahead estimation of the system state on the basis of information available at the current time step. To obtain multistep-ahead estimation, one can repeatedly propagate the one-step estimation a number of time steps into the future, but this process tends to accumulate errors from one propagation to the next. A multistep observer, which directly estimates the state of the system at some specified time step in the future, is identified directly from I/O data. One possible application of this multistep-ahead observer is in receding-horizon predictive control, which bases its present control action on a prediction of the system response at some time step in the future. It is possible to recover the usual one-step-ahead state-space model of the system from the identified multistep-ahead observer as well, although a stabilizing feedback controller can be designed directly from the identified observer. Numerical examples are used to illustrate the key identification and control aspects of this multistep-ahead observer concept.

Introduction

IN a state-space representation of a dynamical system, the output variables are related to the input variables via an intermediate quantity called the state vector. In modern control, the state information is needed to implement a state feedback controller. Because the state vector usually is not accessible to direct measurement, a state estimator, also known as an observer, can be constructed to provide this information. To design an observer, however, one needs to know a state-space model of the system, and in the case of a Kalman filter, also the statistics of the plant and measurement noise if this information is to be taken into account for optimal design. Because the discrete-time state-space representation is in one-step-ahead form, the associated observer in the standard control theory is also in one-step-ahead form. Such an observer estimates the state of the system at one time step in the future based on current (I/O) information. We examine the notion of a multistep-ahead observer, which provides an estimation of the system state at some future time step, based on current information. For example, let k denote the current time step. A 10-step-ahead observer would yield an estimation of the system state at time-step $k + 10$, given an estimated value of the state at the current time-step k, and the control action (if any) to be carried out in the next nine steps. Of course, starting from the current time step k, one can always repeatedly propagate the standard one-step-ahead observer 10 times to obtain an estimation of the state at time step $k + 10$, but this is done at the expense of accumulated error, especially when the prediction horizon is large and the assumed model (for observer design) is not perfect, as is invariably the case. A multistep observer can overcome this limitation because it can be designed such that the multistep-ahead estimation error is minimized directly. Furthermore, in accordance with the observer identification approach,[1–4] this multistep-ahead observer is not designed from a known model of the system, but it is identified directly from I/O data.

One possible application of a multistep observer is in predictive control. Predictive control is the concept in which the current control action is derived by minimizing the difference between a prediction of the system (controlled) output and the desired output at some time step in the future, and the process is repeated at every time step. It originated from chemical engineering[5] and evolved into several methods including model algorithmic control (MAC),[6] dynamic matrix control (DMC),[7] extended-prediction self-adaptive control (EPSAC),[8] extended-horizon adaptive control,[9] multistep multivariable adaptive regulator (MUSMAR),[10] and generalized predictive control (GPC).[11,12] These controllers are different from each other in forming the cost functions, constraints, and I/O models. Recently, the predictive control concept has made its way into the aerospace control community, e.g., to the tracking control of a NASA 70-m Deep Space Network antenna,[13] to the maneuvering of a modern aircraft (F/A-18 High Alpha Research Vehicle),[14] and to the structural vibration suppression problem.[15]

We first describe the relationship between the coefficients of a multistep-ahead output predictor and those of the standard one-step-ahead state-space model of modern control. This connection was initially explored in an approach referred to as identified predictive control (IPC)[16] to design predictive controllers for systems described by controlled autoregressive integrated moving-average models in which the input has a built-in integrator. However, to be consistent with previous results in observer identification theory, we start with the standard state-space model to derive the output predictor equation, where the current output is expressed in terms of a number of past input and output values. Because some coefficients of this model are the system Markov parameters, this output predictive model belongs to a recently developed class of models referred to as ARMarkov models.[17] Next, we proceed to show explicitly the role of a multistep-ahead observer in this connection. From the coefficients of a predictor model, which can be identified from I/O data, we show how one can obtain a multistep observer in state-space form, and how to use it in the design of a predictive controller. By incorporating system identification, the predictive controller thus is designed from I/O data rather than from an assumed model of the system. Numerical examples are used to illustrate the key identification and control aspects of the formulation.

Mathematical Formulation

An overview of the proposed formulation is as follows. First, a set of prediction parameters that defines an output predictor model is computed from I/O data. Second, a multistep-ahead observer is constructed from these parameters. Third, a predictive controller then is synthesized from the identified observer. Fourth, from the identified prediction parameters, one can obtain a state-space representation of the system, and its associated controllability matrix as well. These can be considered as byproducts if the primary objective is to design a controller for the system.

Received Nov. 19, 1996; presented as Paper 97-0683 at the AIAA 35th Aerospace Sciences Meeting, Reno, NV, Jan. 6–9, 1997; revision received June 27, 1997; accepted for publication July 15, 1997.

*Graduate Research Assistant.

†Assistant Professor, Department of Mechanical and Aerospace Engineering. Member AIAA.

Relationship Between State-Space and Output Predictor Models

In this section, we derive a number of key equations that link the usual state-space model used in modern control to an output predictor model for use in predictive control. Consider an nth-order, m-input, q-output discrete-time model of a system in state-space form

$$x(k+1) = Ax(k) + Bu(k), \qquad y(k) = Cx(k) \tag{1}$$

where $A \in R^{n\times n}$, $B \in R^{n\times m}$, $C \in R^{q\times n}$. The state of the system at time step k is denoted by $x(k)$, the control input by $u(k)$, and the output by $y(k)$, $x \in R^{n\times 1}$, $u \in R^{m\times 1}$, $y \in R^{q\times 1}$. By successive substitution, Eq. (1) can be put in an α-step-ahead form as

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ x(k+2) &= A^2x(k) + ABu(k) + Bu(k+1) \\ &= A^2x(k) + [AB, B]\begin{bmatrix} u(k) \\ u(k+1)\end{bmatrix} \\ &\vdots \\ x(k+\alpha) &= A^\alpha x(k) + [A^{\alpha-1}B, \ldots, AB, B]\begin{bmatrix} u(k) \\ \vdots \\ u(k+\alpha-1)\end{bmatrix} \end{aligned} \tag{2}$$

By adding $My(k)$ to and subtracting it from the right-hand side of Eq. (2), we have, respectively,

$$\begin{aligned} x(k+\alpha) &= A^\alpha x(k) + My(k) + [A^{\alpha-1}B, \ldots, AB, B] \\ &\quad \times \begin{bmatrix} u(k) \\ \vdots \\ u(k+\alpha-1)\end{bmatrix} - My(k) \\ &= (A^\alpha + MC)x(k) + [A^{\alpha-1}B, \ldots, AB, B, -M] \\ &\quad \times \begin{bmatrix} u(k) \\ \vdots \\ u(k+\alpha-1) \\ y(k)\end{bmatrix} \end{aligned} \tag{3}$$

The rationale for this operation, which is clarified further in this section, is that it allows us to eliminate the explicit dependence of the state variable in subsequent I/O expressions. Furthermore, we also show that the matrix M, together with A^α, C, and the controllability matrix $\mathcal{C}_\alpha$, are identified directly from I/O data. To simplify the notation, we define $\bar{A}$, $\mathcal{C}_\alpha$, $\bar{\mathcal{C}}_\alpha$, $u_\alpha(k)$, and $v(k)$ as

$$\begin{aligned} \bar{A} &= A^\alpha + MC, \qquad \mathcal{C}_\alpha = [A^{\alpha-1}B, \ldots, AB, B] \\ \bar{\mathcal{C}}_\alpha &= [\mathcal{C}_\alpha, -M], \qquad u_\alpha(k) = \begin{bmatrix} u(k) \\ u(k+1) \\ \vdots \\ u(k+\alpha-1)\end{bmatrix} \\ v(k) &= \begin{bmatrix} u_\alpha(k) \\ y(k)\end{bmatrix} \end{aligned} \tag{4}$$

so that Eq. (3) can be expressed in a simple form,

$$x(k+\alpha) = \bar{A}x(k) + \bar{\mathcal{C}}_\alpha v(k) \tag{5}$$

Note that the matrix $\mathcal{C}_\alpha$ is the system controllability matrix if $\alpha \geq n$. Further successive substitution yields the following $p\alpha$-step-ahead state-space model:

$$x(k+p\alpha) = \bar{A}^p x(k) + \sum_{i=1}^{p} \bar{A}^{i-1}\bar{\mathcal{C}}_\alpha v[k+(p-i)\alpha] \tag{6}$$

At this point, it may be useful to review where α and p enter the equations. The parameter α refers to the number of time steps in a predictive state-space model given by Eq. (2). This parameter raises the power of A in $\bar{A} = A^\alpha + MC$. The parameter p is used to increase the power of $\bar{A}$ itself. We now use the freedom introduced by M and p to impose a deadbeat condition on $\bar{A}$ such that its p-power becomes identically zero:

$$(A^\alpha + MC)^p = 0 \tag{7}$$

where p is such that $qp \geq n$. This special property allows one to write an output predictor model whose coefficients are expressed in terms of the original state-space model as

$$y(k+p\alpha) = \sum_{i=1}^{p} C\bar{A}^{i-1}\bar{\mathcal{C}}_\alpha v[k+(p-i)\alpha], \qquad k \geq 0 \tag{8}$$

Although Eq. (8) is an expression for $y(k+p\alpha)$, it is still α-step predictive because the most recent output measurement involved in the expression for $y(k+p\alpha)$ is $y[k+(p-1)\alpha]$, which is the same as $y(k+p\alpha-\alpha)$. Note that there is no explicit dependence on the state variable $x(k)$ in Eq. (8) because the term $C\bar{A}^p x(k)$ vanishes identically, regardless of $x(k)$. The coefficients for this predictive model are $C\bar{A}^{p-1}\bar{\mathcal{C}}_\alpha$, $C\bar{A}^{p-2}\bar{\mathcal{C}}_\alpha, \ldots, C\bar{\mathcal{C}}_\alpha$. The last group of coefficients in $C\bar{\mathcal{C}}_\alpha$ explicitly involves the system Markov parameters $CA^{\alpha-1}B, \ldots, CAB, CB$ (and $-CM$). Thus, the model in Eq. (8) belongs to the class of ARMarkov models.[17] Because the coefficients $C\bar{A}^{p-1}\bar{\mathcal{C}}_\alpha, \ldots, C\bar{\mathcal{C}}_\alpha$ are the Markov parameters of an α-step-ahead (predictive) state-space model, they are also referred to as the predictor Markov parameters.[18]

The preceding derivation has shown that the coefficients of an output predictor model are related to those of the state-space model A, B, C in a particular fashion, through a special matrix M. We now show that this matrix M has a precise interpretation, namely, it is a gain matrix for an α-step-ahead observer. To this end, consider using the following equation to obtain an estimation of the state at time-step $k+\alpha$, denoted by $\hat{x}(k+\alpha)$, where M is used as a gain on the error between the actual output $y(k)$ of the system and its estimate $\hat{y}(k)$:

$$\begin{aligned} \hat{x}(k+\alpha) &= A^\alpha \hat{x}(k) + \mathcal{C}_\alpha u_\alpha(k) - M[y(k) - \hat{y}(k)] \\ \hat{y}(k) &= C\hat{x}(k) \end{aligned} \tag{9}$$

Let $e(k)$ denote the state estimation error, $e(k) = x(k) - \hat{x}(k)$. From Eqs. (9) and (5), it can be shown that the equation that governs $e(k)$ is $e(k+\alpha) = \bar{A}e(k)$, where $\bar{A} = A^\alpha + MC$, which implies that

$$e(k+p\alpha) = \bar{A}^p e(k) = 0 \tag{10}$$

Equation (10) states that the state estimation error will vanish after $p\alpha$ time steps if the deadbeat condition given by Eq. (7) is imposed. At this point in the formulation, we are concerned only with the existence of such an observer gain matrix because neither A nor C is known. The usual problem is to design M, given A and C. As shown later in this formulation, however, A, C, and M will be identified from I/O data. Note that the condition imposed by Eq. (7) amounts to placing all eigenvalues of $\bar{A} = A^\alpha + MC$ at the origin in the complex plane. By direct analogy with the well-known result in observer pole placement theory, the existence of such an observer gain is ensured if the pair A^α, C is observable. Furthermore, such an observer gain is unique for a single-output system and is not unique for a multiple-output system. In any case, the existence of such a matrix M allows us to conclude that the form of an α-step output predictor given in Eq. (8) is justified.

Identification of Predictor-Model Coefficients

From Eq. (8), the predictor-model coefficients can be computed directly from I/O data with the deadbeat condition in Eq. (7) trivially imposed. Given a set of I/O data, we can arrange it in the form

$$Y = PV \tag{11}$$

where the coefficients of the predictive model are in P:

$$P = \left[C\bar{A}^{p-1}\bar{\mathcal{C}}_\alpha, C\bar{A}^{p-2}\bar{\mathcal{C}}_\alpha, \ldots, C\bar{\mathcal{C}}_\alpha\right] \tag{12}$$

and the data are in Y and V:

$$Y = [y(p\alpha) \quad y(1+p\alpha) \quad y(2+p\alpha) \quad \cdots] \tag{13}$$

$$V = \begin{bmatrix} v(0) & v(1) & v(2) & \cdots \\ \vdots & \vdots & \vdots & \cdots \\ v[(p-2)\alpha] & v[1+(p-2)\alpha] & v[2+(p-2)\alpha] & \cdots \\ v[(p-1)\alpha] & v[1+(p-2)\alpha] & v[2+(p-2)\alpha] & \vdots \end{bmatrix} \tag{14}$$

If a sufficient amount of I/O data is given, P can be computed from Y and V:

$$P = YV^+ \tag{15}$$

where V^+ denotes the pseudoinverse of V. The singular value decomposition method can be used to compute the pseudoinverse. As a matter of good numerical practice, one should always examine the singular values explicitly in this calculation, where the zero (or "practically" zero) singular values (if any) are eliminated to avoid numerical ill-conditioning in the calculation. This practice also allows one to assess the numerical robustness of the solution. In the absence of noise, P can be identified exactly. In the presence of noise, the least-squares solution in Eq. (15) directly minimizes the norm of the α-step-ahead prediction error in the data.

Realization of a Multistep-Ahead Observer

Having identified the predictor coefficients that have A, B, C, and M embedded in a particular fashion, the α-step-ahead observer defined by A^α, $\mathcal{C}_\alpha$, M, and C can be extracted as follows. Defining α_k and β_k as

$$\alpha_k = -C\bar{A}^{k-1}M, \qquad \beta_k = C\bar{A}^{k-1}\mathcal{C}_\alpha, \qquad k = 1, 2, \ldots, p \tag{16}$$

it can be shown algebraically that one can recover the predictor coefficients to produce the following parameters sequences:

$$Y_\mathcal{C}(k) = C(A^\alpha)^{k-1}\mathcal{C}_\alpha = \beta_k + \sum_{i=1}^{k-1} \alpha_i Y_\mathcal{C}(k-i) \tag{17}$$

$$Y_M(k) = C(A^\alpha)^{k-1}M = -\alpha_k + \sum_{i=1}^{k-1} \alpha_i Y_M(k-i) \tag{18}$$

Making use of the deadbeat condition, $\alpha_k = 0$ and $\beta_k = 0$ for $k \geq p$, we can generate $Y_\mathcal{C}(k)$ and $Y_M(k)$ for any k. Also note that the Markov parameters, $Y(k) = CA^{k-1}B$, which are the system unit pulse response samples, appear explicitly in $Y_\mathcal{C}(k)$. For example,

$$\begin{aligned} Y_\mathcal{C}(1) &= [CA^{\alpha-1}B, \ldots, CAB, CB] \\ Y_\mathcal{C}(2) &= [CA^{2\alpha-1}B, \ldots, CA^{\alpha+1}B, CA^\alpha B], \ldots \end{aligned} \tag{19}$$

The dynamics of the system is completely defined by the Markov parameters, which can be further factorized to obtain a realization of A, B, and C. Alternatively, we can obtain A^α, $\mathcal{C}_\alpha$, C, M simultaneously by factorizing the combined sequence

$$Z(k) = [Y_\mathcal{C}(k), Y_M(k)] = C(A^\alpha)^{k-1}[\mathcal{C}_\alpha, M] \tag{20}$$

By direct application of realization theory[19] to the sequence $Z(k)$, $k = 1, 2, \ldots$, a realization of an α-step-ahead observer defined by A^α, $\mathcal{C}_\alpha$, C, M is given as

$$\begin{aligned} A^\alpha &= S_n^{-\frac{1}{2}} U_n^T H(1) V_n S_n^{-\frac{1}{2}} \\ [\mathcal{C}_\alpha, M] &= S_n^{\frac{1}{2}} V_n^T E_{m+q}, \qquad C = E_q^T U_n S_n^{\frac{1}{2}} \end{aligned} \tag{21}$$

where U_n, V_n are n left and right singular vectors of $H(0)$ associated with its n nonzero singular values, $H(0) = USV^T = U_n S_n V_n^T$, and E_k is a selection matrix, $E_k^T = [I_{k\times k}\ \underset{\sim}{0} \cdots \underset{\sim}{0}]$. The Hankel matrices $H(0)$, $H(1)$ are defined as

$$H(k) = \begin{bmatrix} Z(k+1) & Z(k+2) & \cdots & Z(k+s+1) \\ Z(k+2) & Z(k+3) & \cdots & Z(k+s+2) \\ \vdots & \vdots & \vdots & \vdots \\ Z(k+r+1) & Z(k+r+2) & \cdots & Z(k+r+s+1) \end{bmatrix}, \qquad k = 0, 1 \tag{22}$$

where r and s are chosen to be sufficiently large such that $H(k)$ has at least pq singular values. When $\alpha \geq n$, $\mathcal{C}_\alpha = [A^{\alpha-1}B, A^{\alpha-2}B, \ldots, AB, B]$ is the system controllability matrix. If desired, the system matrix A is computed from $\mathcal{C}_\alpha$ as follows:

$$A = [A^{\alpha-1}B, A^{\alpha-2}B, \ldots, AB][A^{\alpha-2}B, A^{\alpha-3}B, \ldots, B]^+ \tag{23}$$

Note also that the input influence matrix B can be obtained directly from last m columns of $\mathcal{C}_\alpha$. The output influence matrix C comes directly from the realization given in Eq. (21). As mentioned, one can apply realization theory to the system Markov parameters in Eq. (19) as well.

Once the α-step-ahead observer is identified from Eq. (21), the system state at some future time step can be computed from information available at the current time step in one of two ways. One way is to use the observer equation given in Eq. (9) with A^α, $\mathcal{C}_\alpha$, C, and the observer gain M. Another way is to make use of the condition in Eq. (7), so that there is no explicit dependence on the current state estimate after $p\alpha$ time steps:

$$\begin{aligned} \hat{x}(k+p\alpha) &= \sum_{i=1}^{p} \bar{A}^{i-1}\mathcal{C}_\alpha u_\alpha[k+(p-i)\alpha] \\ &\quad - \sum_{i=1}^{p} \bar{A}^{i-1} M y[k+(p-i)\alpha], \qquad k \geq 0 \end{aligned} \tag{24}$$

Computational Procedure

In this section, we review the key equations involved in the identification of an α-step-ahead observer from a given set of I/O data.

1) Choose a value of p such that $qp \geq n$, where n is the order of the system and q is the number of (independent) outputs. Form the data matrices Y and V as given in Eqs. (13) and (14), respectively, and compute the α-step-ahead observer parameters P as given by Eq. (15).

2) Use Eqs. (17) and (18) to compute $Y_\mathcal{C}(k)$ and $Y_M(k)$, and group them together to form $Z(k)$, $k = 1, 2, \ldots$ as in Eq. (20). Note that the system pulse response samples (Markov parameters) appear explicitly in $Y_\mathcal{C}(k)$.

3) A realization of A^α, C, an α-step-ahead observer gain matrix M, and the associated controllability matrix $\mathcal{C}_\alpha$ are given by Eq. (21). The Hankel matrices $H(0)$ and $H(1)$ involved in these expressions are defined in Eq. (22). For a chosen value of p, the largest order of the realization that can be obtained is qp. The identified α-step-ahead observer can be implemented via Eq. (9) or, equivalently, via Eq. (24).

Receding-Horizon Predictive Control

Having obtained an α-step-ahead observer, we now can use it to derive a predictive controller as follows. Consider a receding quadratic cost function that penalizes the α-step-ahead state estimate excursions with a weighting Q, and control excursions with a weighting R:

$$J_{k+\alpha} = \hat{x}(k+\alpha)^T Q \hat{x}(k+\alpha) + u_\alpha(k)^T R u_\alpha(k) \tag{25}$$

where Q is positive definite and R is nonnegative definite. Substituting Eq. (9) into the above cost function, and performing the minimization of J with respect to $u_\alpha(k)$, produces

$$u_\alpha(k) = -\left(\mathcal{C}_\alpha^T Q \mathcal{C}_\alpha + R\right)^{-1} \mathcal{C}_\alpha^T Q[\bar{A}\hat{x}(k) - My(k)] \tag{26}$$

Letting G denote the first m rows of $-(\mathcal{C}_\alpha^T Q \mathcal{C}_\alpha + R)^{-1}\mathcal{C}_\alpha^T Q \bar{A}$ and H the first q rows of $(\mathcal{C}_\alpha^T Q \mathcal{C}_\alpha + R)^{-1}\mathcal{C}_\alpha^T Q M$, we have a receding-horizon predictive control law of the simple form

$$u(k) = G\hat{x}(k) + Hy(k) \tag{27}$$

which is a combination of estimated state and output feedback. The estimated state $\hat{x}(k)$ is computed from Eq. (9), where k is replaced by $k-\alpha$, which involves $\hat{x}(k-\alpha)$, $u(k-\alpha)$ up to $u(k-1)$, and $y(k-\alpha)$, i.e.,

$$\begin{aligned}\hat{x}(k) &= A^\alpha \hat{x}(k-\alpha) + \mathcal{C}_\alpha u_\alpha(k-\alpha) - M[y(k-\alpha) - \hat{y}(k-\alpha)] \\ &= \left(A^\alpha + M\mathcal{C}_\alpha\right)\hat{x}(k-\alpha) + \mathcal{C}_\alpha u_\alpha(k-\alpha) - My(k-\alpha)\end{aligned}$$

An extreme case is to set $\hat{x}(k+\alpha)$ in Eq. (9) to zero and then solve for $u_\alpha(k)$:

$$u_\alpha(k) = -\mathcal{C}_\alpha^+[\bar{A}\hat{x}(k) - My(k)] \tag{28}$$

This is a special case of minimizing the cost function J in Eq. (25) with the control weighting R set to zero because

$$\begin{aligned}u_\alpha(k) &= -\left(\mathcal{C}_\alpha^T Q \mathcal{C}_\alpha + R\right)^{-1}\mathcal{C}_\alpha^T Q[\bar{A}\hat{x}(k) - My(k)] \\ &= -\left(\mathcal{C}_\alpha^T Q \mathcal{C}_\alpha\right)^{-1}\mathcal{C}_\alpha^T Q \mathcal{C}_\alpha \mathcal{C}_\alpha^+[\bar{A}\hat{x}(k) - My(k)] \\ &= -\mathcal{C}_\alpha^+[\bar{A}\hat{x}(k) - My(k)] \end{aligned} \tag{29}$$

An alternative form of the controller in Eq. (27) can be obtained by first shifting the time indices in Eq. (24) from $k + p\alpha$ to k, and then substituting the time-shifted Eq. (24) into Eq. (27), so that the explicit dependence on the estimated state variable can be eliminated:

$$u(k) = \sum_{i=1}^{p} G_1 \bar{A}^{i-1}\mathcal{C}_\alpha u_\alpha(k - i\alpha) - \sum_{i=1}^{p} G_1 \bar{A}^{i-1} M y(k - i\alpha) + H_1 y(k) \tag{30}$$

In this form, the current control action is directly expressed as a linear combination of measured input and output data, whereas the form given by Eq. (27) must be used in conjunction with Eq. (9) with k replaced by $k-\alpha$, as mentioned earlier. The two forms, however, are algebraically identical.

In predictive control, the prediction horizon α governs the speed at which the state is to be driven to zero. A controller with a small α drives the system states to zero quickly whereas a controller with a large α brings the states to zero slowly. The minimum value of α, denoted by $\alpha_{\min}$, is the smallest value of α such that $m\alpha_{\min} \geq n$, where m is the number of (independent) inputs and n is the order of the system. The minimum value of p, denoted by $p_{\min}$, is the smallest value of p such that $qp_{\min} \geq n$, where q is the number of (independent) outputs. Controllers designed with various combinations of p and α possess different properties.

The combination of $(\alpha_{\min}, p_{\min})$ produces a deadbeat controller that is associated with a minimum-time controller and a minimum-time observer. This combination is only of theoretical interest because it requires an excessive amount of control energy. The control energy can be reduced by increasing the control weighting or the receding horizon $\alpha > \alpha_{\min}$, which will cause the state to be driven to zero more slowly. In the absence of noise, there is no benefit in selecting $p > p_{\min}$ in the sense that the identified observer is overparameterized, hence containing redundant information. In the presence of noise, however, the redundancy in selecting $p > p_{\min}$ is useful in that it will improve the identification accuracy by averaging out the effect of noise. Thus, the combination of $(\alpha > \alpha_{\min}, p > p_{\min})$ is more practical in that it does not require excessive control input and the identified parameters are more accurate in the presence of noise.

Numerical Examples

Consider a five-degree-of-freedom system consisting of five masses connected in a series, where $m_i, c_i, k_i, i = 1, 2, \ldots, 5$ denote the mass, damping, and stiffness constants, respectively. The variables x_i, $i = 1, 2, \ldots, 5$ are the positions of the five masses measured from their equilibrium positions. The equations of motion are given as

$$M\ddot{x} + C\dot{x} + Kx = B_f u$$

where

$$M = \begin{bmatrix} m_1 & 0 & 0 & 0 & 0 \\ 0 & m_2 & 0 & 0 & 0 \\ 0 & 0 & m_3 & 0 & 0 \\ 0 & 0 & 0 & m_4 & 0 \\ 0 & 0 & 0 & 0 & m_5 \end{bmatrix}, \quad x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix}, \quad B_f = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$C = \begin{bmatrix} c_1 + c_2 & -c_2 & 0 & 0 & 0 \\ -c_2 & c_2 + c_3 & -c_3 & 0 & 0 \\ 0 & -c_3 & c_3 + c_4 & -c_4 & 0 \\ 0 & 0 & -c_4 & c_4 + c_5 & -c_5 \\ 0 & 0 & 0 & -c_5 & c_5 \end{bmatrix}$$

$$K = \begin{bmatrix} k_1 + k_2 & -k_2 & 0 & 0 & 0 \\ -k_2 & k_2 + k_3 & -k_3 & 0 & 0 \\ 0 & -k_3 & k_3 + k_4 & -k_4 & 0 \\ 0 & 0 & -k_4 & k_4 + k_5 & -k_5 \\ 0 & 0 & 0 & -k_5 & k_5 \end{bmatrix}$$

The input to the system is force to the first mass, and the output is position of the last mass, $y = x_5$ (a noncollocated actuator–sensor case). In this simulation, we use $m_i = 1$ kg, $k_i = 1000$ N/m, $c_i = 1$ N $\cdot$ s/m, $i = 1, 2, 3, 4, 5$. The sampling interval is 0.01 s. Figure 1 shows the system output ($y = x_5$) to a random input excitation for 10 s. These data are used to identify a multipstep-ahead observer, and to use it in the design of a stabilization feedback controller.

First, to verify the validity of the developed equations, we first consider the noise-free case (Fig. 1). A multistep-ahead observer with $p = p_{\min} = 10$ and $\alpha = \alpha_{\min} = 10$ is identified from the available I/O data. Note that we have a 10th-order system with one input and one output. Thus, the minimum value for both p and α is 10. We use this minimum value for p and α in this example to illustrate a theoretical extreme of the formulation. Equation (15) is used to identify

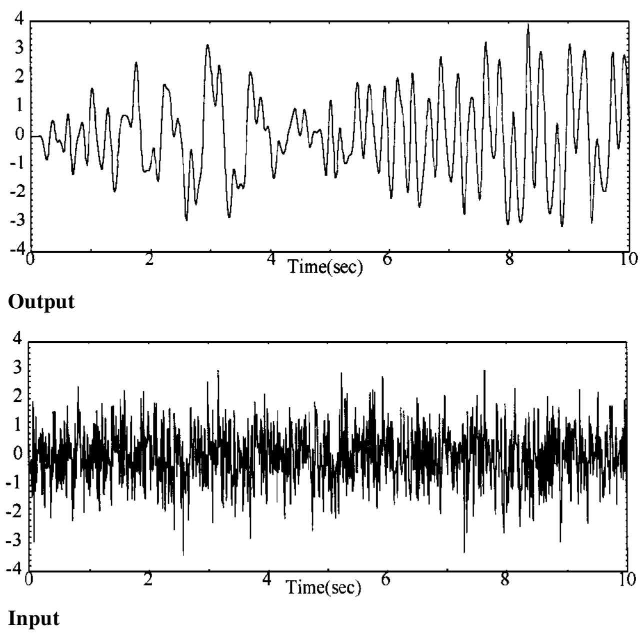

Fig. 1 Open-loop response to random excitation.

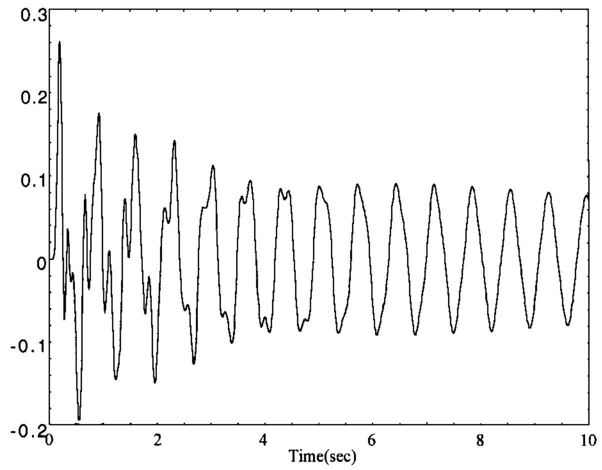

Fig. 2 Actual vs identified pulse responses.

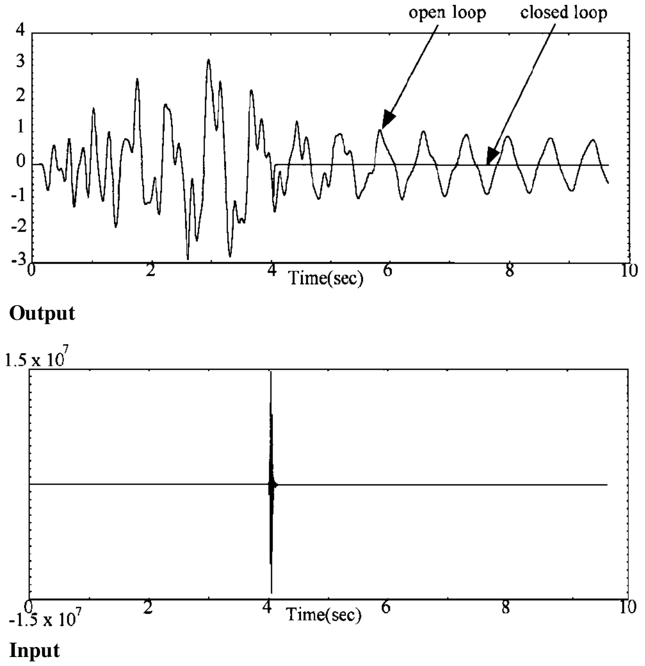

Fig. 3 System responses and control input, $(p, \alpha) = (10, 10)$.

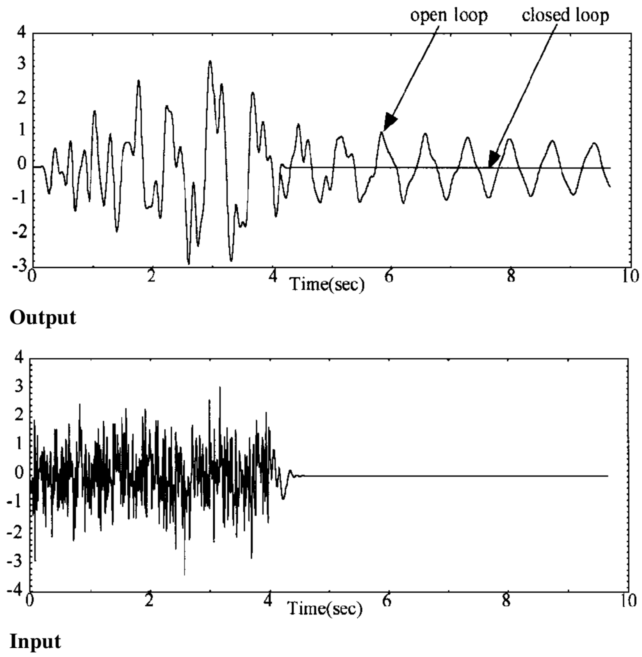

Fig. 4 System responses and control input, $(p, \alpha) = (10, 35)$.

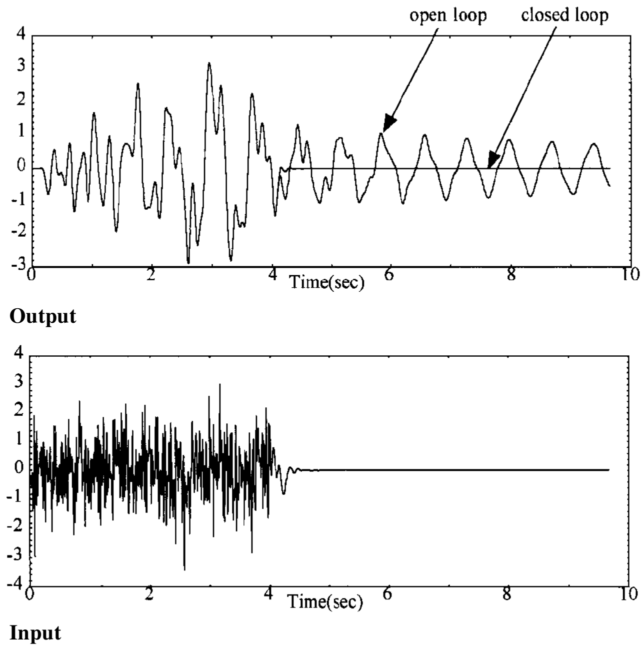

Fig. 5 System responses and control input, $(p, \alpha, Q, R) = (10, 15, I_{10\times 10}, 0.003 I_{15\times 15})$.

the coefficients of a 10-step predictor model, and Eq. (17) is used to recover the system unit pulse response samples. The computed and actual pulse responses are identical, as shown in Fig. 2, which verifies that the identification result is perfect. Next, a controller is designed from the identified observer. Because $\alpha = \alpha_{\min} = 10$ is the minimum value, this controller is expected to suppress the system vibration in the shortest amount of time possible. As an illustration, the system is excited randomly for another 4 s, after which the controller is turned on. Figure 3 shows the closed-loop responses of this deadbeat controller. The system vibration is suppressed in exactly 0.1 s (10 steps), but the required control effort is extremely large. The oscillatory curve is the open-loop response without control. The excessive control effort can be reduced by increasing the prediction horizon. Increasing α from 10 to 35 reduces the maximum control amplitude from the impractical 1.5×10^7 to 1.5, as shown in Fig. 4, although it takes longer time to suppress the vibration. Another way to decrease the maximum control is by choosing a positive control weighting. A controller with a shorter prediction horizon and a "small" control weighting ($\alpha = 15$, $R = 0.003 I_{15\times 15}$) performs similarly to a controller with a larger prediction horizon ($\alpha = 35$) and no control weighting (Fig. 5) in that the maximum control amplitudes in both cases are almost the same ($u_{\max} \approx 1.5$). The example shows the expected result that the control effort decreases with increasing control weighting, or longer prediction horizon.

In the presence of noise, it is observed that increasing p generally improves both the identification and control results. This numerical observation is consistent with known theoretical understanding that system identification can be performed more accurately when a larger order is assumed in the identification model.[1–3] As an illustration, Fig. 6 shows the case in which the controller is designed with a larger value of p, $p = 15$, and $\alpha = 15$, $Q = I_{10\times 10}$, $R = 0.005 I_{15\times 15}$. The controller is designed from a 15-step-ahead observer computed from data contaminated by measurement noise. The controller then is implemented to the actual (true) system, with measurement noise present. The steady-state fluctuation seen in Fig. 6 is due almost entirely to the rather significant additive noise at the output.

Finally, if the primary goal is to obtain a multistep-ahead prediction, e.g., for prediction control, Figs. 7 and 8 illustrate the benefit of identifying an α-step-ahead observer directly, rather than identifying a one-step-ahead observer and then repeatedly propagating it α

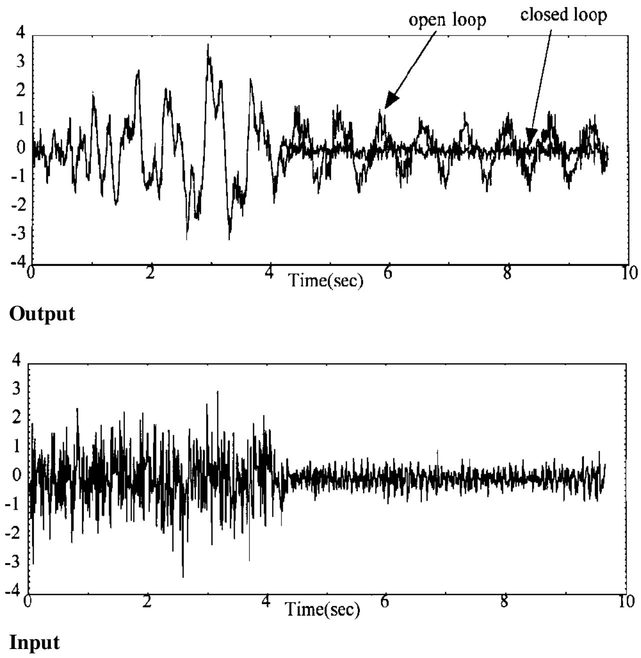

Fig. 6 System responses and control input in the presence of noise, $(p, \alpha, Q, R) = (15, 15, I_{10\times 10}, 0.005 I_{15\times 15})$.

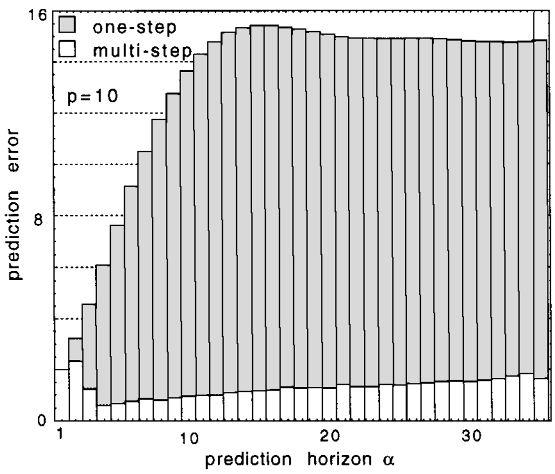

Fig. 7 Prediction from identified one-step and multistep observers, $p = 10$.

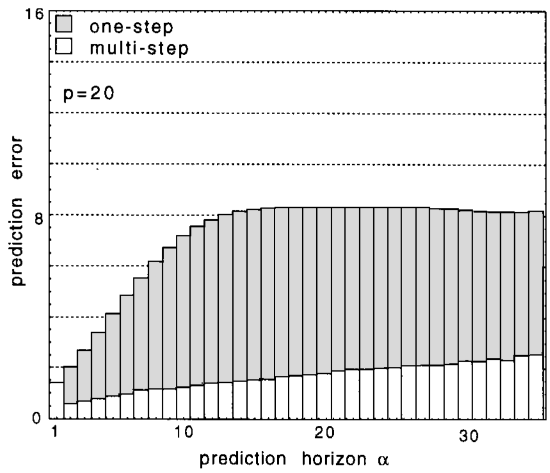

Fig. 8 Prediction from identified one-step and multistep observers, $p = 20$.

times to obtain an α-step-ahead prediction. The norms of the α-step-ahead prediction errors over a 6-s interval of one element of the state vector (position of the fifth mass) are plotted against the prediction horizons α. Note that the α-step-ahead prediction computed by the α-step-ahead observer is consistently more accurate than that obtained from the one-step approach. This result is expected because the α-step-ahead approach minimizes the α-step-ahead prediction error directly, whereas the one-step-ahead approach minimizes the one-step-ahead prediction error. Both observers are identified from the same noise-contaminated I/O data considered in the preceding example. In the presence of noise, increasing p improves the quality of the prediction by the one-step approach, but the α-step approach is still significantly better. In the absence of noise, of course, both observers identify the system exactly; hence, both can predict the response exactly without error.

Conclusions

We have explored the notion of a multistep-ahead observer as an extension to the standard one-step-ahead observer in modern control theory. A multistep-ahead observer directly predicts the state of the system at some future time step, given the current estimated state and possible future input to be applied to the system. This avoids the need to repeatedly propagate the one-step-ahead model, which tends to accumulate the estimation error, especially when the prediction horizon is large and the assumed model is imperfect. Furthermore, we have shown that such an observer is subsumed in an output predictor model whose coefficients can be identified directly from I/O data. Relevant expressions that obtain this observer from the predictor coefficients are developed. The system controllability matrix is identified along with this multistep observer, from which the standard one-step-ahead state-space model can be recovered as well if desired. We also explore one possible application of the identified multistep observer in the context of predictive control. In fact, one can design a predictive controller directly from the identified multistep observer without the need to obtain the usual (one-step-ahead) state-space model of the system.

Because of the simplicity of the steps involved in this formulation, it is possible to implement the design in real time. One can envision a system that generates a controller design (and a model of the system as a byproduct) directly from input and output data. The identification step involved is not to identify the system per se, but rather it identifies the needed parameters for successful design of a controller. For the predictive control design considered in this paper, it is a multistep-ahead observer. As in any other control design approaches, there are design parameters that influence the controller performance. In the case of predictive control, the two primary parameters are the prediction horizon and the order of the identified observer. Adaptive strategies can be developed so that these parameters can be tuned in real time. Investigation of these issues in the stochastic case as well as actual implementation of the developed identification and control strategy in a laboratory setting will be pursued in future work.

References

[1]Phan, M., Horta, L. G., Juang, J.-N., and Longman, R. W., "Linear System Identification via an Asymptotically Stable Observer," *Journal of Optimization Theory and Applications*, Vol. 79, No. 1, 1993, pp. 59–86.

[2]Juang, J.-N., Phan, M., Horta, L. G., and Longman, R. W., "Identification of Observer/Kalman Filter Markov Parameters: Theory and Experiments," *Journal of Guidance, Control, and Dynamics*, Vol. 16, No. 2, 1993, pp. 320–329.

[3]Phan, M., Horta, L. G., Juang, J.-N., and Longman, R. W., "Improvement of Observer/Kalman Filter Identification (OKID) by Residual Whitening," *Journal of Vibrations and Acoustics*, Vol. 117, No. 2, 1995, pp. 232–239.

[4]Juang, J.-N., *Applied System Identification*, Prentice–Hall, Englewood Cliffs, NJ, 1994, pp. 175–227.

[5]Richalet, J., Rault, A., Testud, J. L., and Papon, J., "Model Predictive Heuristic Control: Applications to Industrial Processes," *Automatica*, Vol. 14, No. 5, 1978, pp. 413–428.

[6]Rouhani, R., and Mehra, R. K., "Model Algorithmic Control (MAC), Basic Theoretical Properties," *Automatica*, Vol. 18, No. 1, 1982, pp. 401–414.

[7]Cutler, C. R., and Ramaker, B. L., "Dynamic Matrix Control—A Computer Control Algorithm," *Proceedings of the JACC*, WP5-B, San Francisco, 1980.

[8]De Keyser, R. M. C., and Van Cauwenberghe, A. R., "A Self-Tuning Multi-Step Predictor Application," *Automatica*, Vol. 17, No. 1, 1979, pp. 167–174.

[9]Ydstie, B. E., "Extended Horizon Adaptive Control," *Proceedings of the 9th IFAC World Congress*, Vol. 7, Budapest, Hungary, 1984, pp. 133–138.

[10]Menga, G., and Mosca, E., "MUSMAR: Multivariable Adaptive Regulators Based on Multistep Cost Functionals," *Advances in Control*, edited by D. G. Lainiotis and N. S. Tzannes, Reidel, Dordrecht, The Netherlands, 1980, pp. 334–341.

[11]Clarke, D. W., Mohtadi, C., and Tuffs, P. S., "Generalized Predictive Control, Part 1. The Basic Algorithm," *Automatica*, Vol. 23, No. 2, 1987, pp. 137–148.

[12]Clarke, D. W., Mohtadi, C., and Tuffs, P. S., "Generalized Predictive Control, Pt. 2. Extensions and Interpretations," *Automatica*, Vol. 23, No. 2, 1987, pp. 149–160.

[13]Gawronski, W., "Predictive Controller and Estimator for NASA Deep Space Network Antennas," *Journal of Dynamic Systems, Measurement and Control*, Vol. 116, No. 2, 1994, pp. 241–248.

[14]Heise, S. A., and Maciejowski, J. M., "Model Predictive Control of a Supermaneuverable Aircraft," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Washington, DC, 1996.

[15]Phan, M., and Juang, J.-N., "Predictive Feedback Controllers for Stabilization of Linear Multivariable Dynamic Systems," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, Washington, DC, 1996.

[16]Bialasiewicz, J. T., Horta, L. G., and Phan, M., "Identified Predictive Control," *Proceedings of the American Control Conference*, Vol. 3, American Automatic Control Council, Green Valley, AZ, 1994, pp. 3243–3247.

[17]Hyland, D. C., "Adaptive Neural Control (ANC) Architecture—a Tutorial," *Proceedings of the Industry, Government, and University Forum on Adaptive Neural Control for Aerospace Structural Systems*, Harris Corp., Melbourne, FL, 1993.

[18]Phan, M. Q., Juang, J.-N., and Hyland, D. C., "On Neural Networks in Identification and Control of Dynamic Systems," *Wave Motion, Intelligent Structures, and Nonlinear Mechanics*, edited by A. Guran and D. J. Inman, World Scientific, Singapore, 1995, pp. 194–225.

[19]Juang, J.-N., and Pappa, R. S., "An Eigensystem Realization Algorithm for Model Parameter Identification and Model Reduction," *Journal of Guidance, Control, and Dynamics*, Vol. 8, No. 5, 1985, pp. 620–627.