JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 21, No. 5, September–October 1998

Predictive Controllers for Feedback Stabilization

Minh Q. Phan*
Princeton University, Princeton, New Jersey 08544
and
Jer-Nan Juang†
NASA Langley Research Center, Hampton, Virginia 23681

Predictive control is a strategy where the current control action is based on a prediction of the system response at some number of time steps into the future. This approach allows one to work directly with an input–output model of the system, in contrast to a state-space based approach where the state variable is involved explicitly in the controller design and implementation. A connection is provided between the two design approaches by showing that simple predictive controllers can be thought of as generalizations of the observer-based deadbeat (minimum-time) controllers in discrete-time state-space theory. The excessive and, hence, impractical control effort demanded by the standard deadbeat controllers is removed in this predictive extension. By varying the prediction horizon, controllers that range from those exhibiting the extreme deadbeat behavior to those approximating the minimum-energy solution are subsumed in a common framework. Although these predictive controllers can be understood as observer based, no explicit observer is actually involved in the implementation. Instead, these controllers can be derived directly from the coefficients of an identified input–output model, which has been recently shown to subsume an implicit observer. It is also shown that the two primary parameters of a predictive controller can be understood in the state-space framework as those governing the speed of an implicit observer and the speed of an implicit state-feedback controller. This understanding allows one to select optimal choices for these parameters in practice.

Introduction

THE state-space model has long been a fundamental element of modern control theory. In a state-space model, the relationship between the input and output variables is described in terms of an intermediate quantity called the state vector. The needed state-space models can be derived analytically from the equations of motions or identified from experimental input–output data using system identification. Concurrent with the development of state-space based control methods are adaptive and predictive control, which are based on input–output models. There is a very large body of literature on the subject of adaptive and predictive control.[1–10] Application of the predictive concept to aerospace control problems is recent.[11–13] A typical input–output model describes the current output as a linear combination of past input and output measurements. One such model is the autoregressive moving average model with exogenous input (ARX), which is the most commonly used model in discrete-time adaptive control. An attractive feature of an ARX model is that its coefficients can be identified from input and output measurements; the identification process can be carried out recursively in real time if necessary. The predictive control concept can be thought of as an extension of the one-step ahead approach of adaptive control theory,[7] which calls for one-step ahead inversion of the input–output model to produce the control action. This simple inversion approach is not suitable for a nonminimum phase system, which will cause the control input to grow unbounded while the controlled output remained bounded. Note that the same difficulty is also encountered if one inverts an identified state-space model instead.[14] By introducing proper dynamics into the controller structure, this problem can be overcome. Indeed, in state feedback control approach, suppression of the state variables ensures that the control input remains bounded, such as the case of a linear quadratic regulator (LQR) or any pole placement controller.

To better understand the connection between the state-space and input–output models-based control approaches, it is important to understand how an input–output model relates to the state-space model and vice versa. It is well known that starting with an input–output model, such as an ARX model, one can convert it to an equivalent observable or controllable canonical state-space form. The reverse connection from a state-space model to an equivalent ARX model can also be carried out via canonical forms. Recently, it was found that the relationship between the state-space matrices and the coefficients of the equivalent ARX model can be explained in terms of an (implicit) observer gain matrix.[15–18] Specifically, the coefficients of an equivalent ARX model are the Markov parameters of an associated observer for the state-space model under consideration. Depending on the order of the ARX model and the characteristics of the noise in the system, the implicit observer in the model has different meanings. In the absence of noise, an ARX model with the smallest model order corresponds to the fastest observer possible, which is a deadbeat observer. In the presence of noise, an ARX model of a sufficiently large order subsumes an observer whose prediction error is minimized in the least-squares sense. In the special case where the process and measurement noises are white, Gaussian, and uncorrelated, and the order of the ARX model is sufficiently large, then the implicit observer in the ARX model approximates an optimal Kalman filter for the given noise statistics embedded in the system.

Motivated by the system identification results, we look for controllers that can make use of the implicit observer information embedded in an identified input–output model directly. Furthermore, we would like to exploit this understanding in the relationship between the state-space model and an ARX model to derive controllers capable of retaining the key benefits of the state-space-based approach in not requiring the system to be minimum phase nor the number of inputs to be equal to the number of outputs. We show that this can be done by extending the standard deadbeat controllers in state-space theory to produce predictive controllers that are analogous to those derived from an input–output model. This predictive extension overcomes the unrealistic demand on the control energy by the standard deadbeat approach. Instead, a wide range of solutions from minimum time to those approximating the minimum energy solution can be obtained by simply varying the prediction horizon and the order of the input–output model. Furthermore, we show that although these predictive controllers can be understood within the observer-based structure of state-space theory, they can be implemented without an explicit observer. Instead, the state-estimation step is built into the input–output model. Numerical results will be

Received Nov. 30, 1995; revision received April 4, 1997; accepted for publication Jan. 14, 1998.

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

†Principal Scientist, Structural Dynamics Branch. Fellow Member AIAA.

used to illustrate the basic features of these simple and intuitively appealing predictive controller designs.

Mathematical Formulation

We will first describe a number of results that will be put together to synthesize the proposed controllers. These include open loop and feedback deadbeat and predictive control. In the following development, both the single-input/single-output (SISO) and the multi-input/multi-output (MIMO) cases are treated simultaneously. Whenever appropriate, the matrix dimensions will be pointed out explicitly to avoid possible confusion.

State-Feedback Deadbeat and Predictive Control

Consider a nth-order state-space model of the system of the form

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

with r inputs and m outputs. From the state equations, the state of the system at each successive time steps can be written as

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

$$x(k+2) = A^2x(k) + ABu(k) + Bu(k+1) = A^2x(k) + [AB, B]\begin{bmatrix} u(k) \\ u(k+1) \end{bmatrix}$$

$$x(k+n-r+1) = A^{n-r+1}x(k) + [A^{n-r}B, \ldots, AB, B] \times \begin{bmatrix} u(k) \\ u(k+1) \\ \vdots \\ u(k+n-r) \end{bmatrix}$$

Starting with any $x(k) \neq 0$, the control input time history that will bring the entire state $x(k)$ to zero is the solution of

$$[A^{n-r}B, \ldots, AB, B]\begin{bmatrix} u(k) \\ u(k+1) \\ \vdots \\ u(k+n-r) \end{bmatrix} = -A^{n-r+1}x(k) \tag{2}$$

For a single-input (controllable) system, the minimum-time deadbeat control time history will bring the state $x(k)$ to zero in n steps. For a multi-input system, it is possible to bring the state to the origin sooner. In terms of state-feedback solution, it is known that the deadbeat controller takes the form

$$u(k) = G_D x(k) \tag{3}$$

where the deadbeat gain G_D places all of the eigenvalues of the closed-loop system matrix $A + BG_D$ at the origin in the complex plane. In practice, unconstrained minimum-time (deadbeat) solution is not useful because it requires excessive control effort. Therefore, we require that the state of the system be suppressed in q steps where $q > n - r + 1$. The corresponding equations are

$$x(k+q) = A^q x(k) + [A^{q-1}B, \ldots, AB, B]\begin{bmatrix} u(k) \\ \vdots \\ u(k+q-2) \\ u(k+q-1) \end{bmatrix} \qquad q > n \tag{4}$$

Because there are more than a number of minimum time steps to bring the state vector to zero, the control input time history that can accomplish this task is not unique. We are interested in the minimum control energy solution, which can be obtained from

$$\begin{bmatrix} u(k) \\ \vdots \\ u(k+q-2) \\ u(k+q-1) \end{bmatrix} = -[A^{q-1}B, \ldots, AB, B]^+ A^q x(k) \tag{5}$$

where $[\,]^+$ denotes the pseudoinverse of the matrix in the bracket. This solution motivates a feedback control law of the form

$$u(k) = Gx(k) \tag{6}$$

where G is taken to be the first r-row partition of $-[A^{q-1}B, \ldots, AB, B]^+A^q$, where r is the number of inputs. This state feedback control law, thus, has the interpretation that, at each time step, it is the first of a minimum-norm control sequence that will bring state of the system to zero in q steps. This general idea of computing the control action by looking ahead is well established in predictive and adaptive control theory. Equation (6) is a predictive control law in state-feedback form, where q is the prediction horizon. A smaller value of q implies a shorter prediction horizon, and more control energy is needed to bring the state of the system to zero in shorter time. A larger value of q implies a longer prediction horizon and less control energy is required. This concept is somewhat analogous to driving a car where the driver looks at some distance ahead to determine his immediate driving action. As a limiting case we have the deadbeat control situation considered earlier. For a single-input system, the state of the system can be brought to zero in n steps. For a multiple-input system, the minimum number of time steps can be less than n. It is the minimum value of q such that $q_{\min} r \geq n$.

Dynamic Output Feedback Form of the Deadbeat and Predictive Controller

The preceding consideration examines the deadbeat and predictive controller in state-feedback form. This state-feedback control law can be used only when the state information is known. Normally, one requires a state-space model of the system from which an observer (state estimator) is designed to provide the state information for feedback purpose. It will be much more convenient to avoid state estimation altogether if linear combinations of past input and output measurements are used as the states. If this can be done, then the controller will assume the general form where the current control input is a linear combination of past input and past output measurements. It will be shown in the following that an identified ARX representation of the system can be used for this purpose. We will first briefly describe an ARX model and the identification of its coefficients from input–output data and then explain how this model can be used to convert the predictive controller from state-feedback form into dynamic output feedback form.

An ARX model has the following form:

$$y(k) = \alpha_1 y(k-1) + \alpha_2 y(k-2) + \cdots + \alpha_p y(k-p) + \beta_1 u(k-1) + \beta_2 u(k-2) + \cdots + \beta_p u(k-p) \tag{7}$$

For SISO systems, the coefficients α_i and β_i are scalars. For MIMO systems, each α_i is an $m \times m$ matrix, and each β_i is an $m \times r$, where m and r are the number of outputs and inputs, respectively. The coefficients of an ARX model can be identified directly from input and output data as follows. For a set of ℓ data points, writing the output of the ARX model at every time step yields

$$\mathsf{Y} = \mathsf{PV} \tag{8}$$

where P is made up of the ARX coefficients,

$$\mathsf{P} = [\beta_1, \alpha_1, \beta_2, \alpha_2, \ldots, \beta_p, \alpha_p] \tag{9}$$

and the data matrices Y and V are given as

$$\mathsf{Y} = [y(p), y(p+1), \ldots, y(\ell)] \tag{10}$$

$$\mathsf{V} = \begin{bmatrix} u(p-1) & u(p) & \cdots & u(\ell-1) \\ y(p-1) & y(p) & \cdots & y(\ell-1) \\ \vdots & \vdots & \vdots & \vdots \\ u(0) & u(1) & \cdots & u(\ell-p) \\ y(0) & y(1) & \cdots & y(\ell-p) \end{bmatrix} \tag{11}$$

Therefore, the ARX parameters can be computed from

$$\mathsf{P} = \mathsf{YV}^+ \tag{12}$$

where V^+ is the pseudoinverse of V, which can be computed via the singular value decomposition.

To convert the state-feedback predictive controller into a dynamic output feedback form, first write the ARX model as

$$y(k) = \alpha_1 y(k-1) + \beta_1 u(k-1) + z_1(k-1) \tag{13}$$

where

$$\begin{aligned} z_1(k-1) &= \alpha_2 y(k-2) + \beta_2 u(k-2) + z_2(k-2) \\ z_2(k-2) &= \alpha_3 y(k-3) + \beta_3 u(k-3) + z_3(k-3) \\ &\vdots \\ z_{p-1}(k-p+1) &= \alpha_p y(k-p) + \beta_p u(k-p) \end{aligned} \tag{14}$$

One can shift the time indices to obtain

$$\begin{aligned} y(k+1) &= \alpha_1 y(k) + \beta_1 u(k) + z_1(k) \\ z_1(k+1) &= \alpha_2 y(k) + \beta_2 u(k) + z_2(k) \\ z_2(k+1) &= \alpha_3 y(k) + \beta_3 u(k) + z_3(k) \\ &\vdots \\ z_{p-1}(k+1) &= \alpha_p y(k) + \beta_p u(k) \end{aligned} \tag{15}$$

This operation simply puts the system in an observable canonical form

$$\begin{aligned} z(k+1) &= A_p z(k) + B_p u(k) \\ y(k) &= C_p z(k) \end{aligned} \tag{16}$$

where

$$z(k) = \begin{bmatrix} y(k) \\ z_1(k) \\ z_2(k) \\ \vdots \\ z_{p-1}(k) \end{bmatrix}, \qquad A_p = \begin{bmatrix} \alpha_1 & I & 0 & \cdots & 0 \\ \alpha_2 & 0 & I & \ddots & \vdots \\ \alpha_3 & 0 & 0 & \ddots & 0 \\ \vdots & \vdots & \vdots & \ddots & I \\ \alpha_p & 0 & 0 & \cdots & 0 \end{bmatrix}$$

$$B_p = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \vdots \\ \beta_p \end{bmatrix}, \qquad C_p = [I \quad 0 \quad 0 \quad \cdots \quad 0] \tag{17}$$

Other canonical forms for MIMO systems can be found in Refs. 7 and 18. A q-step predictive controller based on an ARX representation of order p for this system is

$$u(k) = u_{(p,q)}(k) = Gz(k) \tag{18}$$

where the $r \times pm$ matrix G is the first r-row partition of $-[A_p^{q-1}B_p, \ldots, A_pB_p, B_p]^+A_p^q$. In the state-space domain, the controlled system is governed by the closed-loop equation

$$z(k+1) = (A_p + B_pG)z(k) \tag{19}$$

To convert Eq. (18) to dynamic output feedback form, let the p partitions of G be defined as

$$G = [g_1, g_2, g_3, \ldots, g_p]$$

where the dimension of each g_i is $r \times m$. Then the predictive controller becomes

$$u_{(p,q)}(k) = g_1 y(k) + g_2 z_1(k) + g_3 z_2(k) + \cdots + g_p z_{p-1}(k) \tag{20}$$

The expression for $y(k)$ is given in Eq. (13) and expressions $z_i(k)$, $i = 1, 2, \ldots, p-1$ are given in Eqs. (14):

$$\begin{aligned} z_1(k) &= \alpha_2 y(k-1) + \beta_2 u(k-1) + z_2(k-1) \\ &= \alpha_2 y(k-1) + \beta_2 u(k-1) + \alpha_3 y(k-2) + \beta_3 u(k-2) \\ &\quad + \cdots + \alpha_p y(k-p+1) + \beta_p u(k-p+1) \\ z_2(k) &= \alpha_3 y(k-1) + \beta_3 u(k-1) + \alpha_4 y(k-2) + \beta_4 u(k-2) \\ &\quad + \cdots + \alpha_p y(k-p+2) + \beta_p u(k-p+2) \\ &\vdots \\ z_{p-1}(k) &= \alpha_p y(k-1) + \beta_p u(k-1) \end{aligned}$$

The controller now assumes the compact dynamic output feedback form

$$\begin{aligned} u_{(p,q)}(k) &= G_1 y(k-1) + G_2 y(k-2) + \cdots + G_p y(k-p) \\ &\quad + H_1 u(k-1) + H_2 u(k-2) + \cdots + H_p u(k-p) \end{aligned} \tag{21}$$

where the controller gain matrices are given by

$$\begin{aligned} G_1 &= [g_1 \quad g_2 \quad \cdots \quad g_p] \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_p \end{bmatrix} \\ G_2 &= [g_1 \quad g_2 \quad \cdots \quad g_{p-1}] \begin{bmatrix} \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_p \end{bmatrix}, \ldots, G_p = g_1\alpha_p \\ H_1 &= [g_1 \quad g_2 \quad \cdots \quad g_p] \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{bmatrix} \\ H_2 &= [g_1 \quad g_2 \quad \cdots \quad g_{p-1}] \begin{bmatrix} \beta_2 \\ \beta_3 \\ \vdots \\ \beta_p \end{bmatrix}, \ldots, H_p = g_1\beta_p \end{aligned} \tag{22}$$

Because the dimension of each g_i is $r \times m$, each α_i is $m \times m$ and each β_i is $m \times r$; the dimension of each G_i is $r \times m$ and each H_i is $r \times r$ as expected. Clearly, these controllers are applicable to the multi-input and multi-output cases.

Implicit Observer and Kalman Filter in ARX Model

The relationship between the parameter p, which is the order of the ARX model, to the state-space model is now examined. We will show in the following that this relationship can be explained in terms of a special matrix M_p:

$$M_p = \begin{bmatrix} -\alpha_1 \\ -\alpha_2 \\ -\alpha_3 \\ \vdots \\ -\alpha_p \end{bmatrix} \tag{23}$$

which turns the combination $\bar{A}_p = A_p + M_pC_p$ into a special matrix

$$\bar{A}_p = A_p + M_pC_p = \begin{bmatrix} 0 & I & 0 & \cdots & 0 \\ 0 & 0 & I & \ddots & \vdots \\ 0 & 0 & 0 & \ddots & 0 \\ \vdots & \vdots & \vdots & \ddots & I \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix} \tag{24}$$

This matrix has the property that its p power becomes identically zero:

$$(\bar{A}_p)^p = (A_p + M_p C_p)^p = 0 \tag{25}$$

This special property allows one to derive an ARX model from the state-space model by adding and subtracting the term $M_p y(k)$ to the right-hand side of the state equation

$$\begin{aligned} z(k+1) &= A_p z(k) + B_p u(k) + M_p y(k) - M_p y(k) \\ &= (A_p + M_p C_p) z(k) + B_p u(k) - M_p y(k) \end{aligned} \tag{26}$$

In Eq. (26) the state of the system is not only a function of the system input but also its output. Together with the output equation $y(k) = C_p z(k)$, the current output can be expressed as a linear combination of past input and past output measurements:

$$\begin{aligned} y(1) &= C_p \bar{A}_p z(0) + C_p B_p u(0) - C_p M_p y(0) \\ y(2) &= C_p \bar{A}_p^2 z(0) + C_p \bar{A}_p B_p u(0) + C_p B_p u(1) \\ &\quad - C_p \bar{A}_p M_p y(0) - C_p M_p y(1) \\ &\ \vdots \end{aligned}$$

In general, for $k \geq p$,

$$\begin{aligned} y(k) &= C_p \bar{A}_p^{p-1} B_p u(k-p) + \cdots + C_p B_p u(k-1) \\ &\quad - C_p \bar{A}_p^{p-1} M_p y(k-p) - \cdots - C_p M_p y(k-1) \end{aligned} \tag{27}$$

Note that for $k \geq p$ only p past input and p past output measurements are involved in the expression because $(\bar{A}_p)^p = (A_p + M_p C_p)^p = 0$ identically. For the same reason, the initial state $z(0)$ no longer contributes to the input–output relationship for $k \geq p$. Note that this result is independent of the actual system dynamics, which may still be in the transient portion after p time steps. Equation (27) is exactly the same as the ARX model described earlier in Eq. (7). Comparing Eq. (27) with Eq. (7) reveals that the coefficients α_k and β_k of the ARX model are related to those of the state-space model by the following relationship ($k = 1, 2, \ldots, p$):

$$\begin{aligned} \alpha_k &= -C_p (A_p + M_p C_p)^{k-1} M_p \\ \beta_k &= C_p (A_p + M_p C_p)^{k-1} B_p \end{aligned} \tag{28}$$

Another observation is that Eq. (26) has the exact form of an observer that estimates $z_p(k)$ from input and output measurements. Specifically, the associated observer has the form

$$\begin{aligned} \hat{z}(k+1) &= A_p \hat{z}(k) + B_p u(k) + M_p [y(k) - \hat{y}(k)] \\ &= (A_p + M_p C_p) \hat{z}(k) + B_p u(k) - M_p y(k) \\ \hat{y}(k) &= C_p \hat{z}(k) \end{aligned} \tag{29}$$

If one propagates Eq. (29) forward in time, one again obtains Eq. (27) because $\hat{y}(k)$ will converge to $y(k)$ in exactly p time steps because of Eq. (25). For this reason, the combination $\bar{A}_p = A_p + M_p C_p$ is the observer system matrix, and M_p is a deadbeat observer gain in the observable canonical coordinates. If the state-space model is in a different set of coordinates, then there is a corresponding deadbeat observer gain in that set of coordinates that will relate the state-space matrices to the coefficients of the ARX model by the same relationship as given in Eq. (28). It is now clear that an ARX model of order p subsumes a deadbeat observer of order p.

For a single-output system, it is known that the order of an ARX model and its equivalent state-space model is the same, $p = n$. However, for a multiple-output state-space model of order n, the preceding consideration reveals that the minimum order of the equivalent ARX model can be less than n. It corresponds to the minimum value of p such that $p_{\min} q \geq n$. If p is chosen such that the product pq exceeds the true order n of the system, and an observable canonical state-space realization is constructed from an identified ARX model of order p according to Eq. (12), then the pq-order observable canonical realization necessarily contains uncontrollable states, which corresponds to linearly dependent rows in the controllability matrix

$$\mathsf{C}_q = \left[(A_p)^{q-1} B_p, \ldots, A_p B_p, B_p \right] \tag{30}$$

Recall that the computation of G involves the pseudoinverse of the controllability matrix C_q. This step can be easily handled by a singular value decomposition of C_q, where the zero singular values associated with this uncontrollable subspace are eliminated to compute the pseudoinverse of C_q.

In the presence of noise, if p is chosen to be sufficiently large, then the structure of the identified ARX model computed from Eqs. (12) subsumes an optimal Kalman filter. This can be shown as follows. Consider the case where the state-space equations given in Eq. (1) are extended to include process and measurement noise

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + w_1(k) \\ y(k) &= Cx(k) + w_2(k) \end{aligned} \tag{31}$$

We make the standard assumption that the process noise $w_1(k)$ and measurement noise $w_2(k)$ are two statistically independent, zero-mean, stationary white noise processes. If A is stable and (A, C) is an observable pair, the same system can also be expressed in the form of a Kalman filter

$$\begin{aligned} \hat{x}(k+1) &= A\hat{x}(k) + Bu(k) - K\varepsilon(k) \\ y(k) &= C\hat{x}(k) + \varepsilon(k) \end{aligned} \tag{32}$$

where the optimal residual $\varepsilon(k)$ is white and minimized. The Kalman filter in Eq. (32) can also be expressed as

$$\begin{aligned} \hat{x}(k+1) &= (A + KC)\hat{x}(k) + Bu(k) - Ky(k) \\ y(k) &= C\hat{x}(k) + \varepsilon(k) \end{aligned} \tag{33}$$

If p is sufficiently large such that

$$\tilde{A}^k = (A + KC)^k \approx 0, \qquad k \geq p \tag{34}$$

then the input–output description can be approximated by

$$\begin{aligned} y(k) &= \tilde{\alpha}_1 y(k-1) + \cdots + \tilde{\alpha}_p y(k-p) \\ &\quad + \tilde{\beta}_1 u(k-1) + \cdots + \tilde{\beta}_p u(k-p) + \varepsilon(k) \end{aligned} \tag{35}$$

where for $k = 1, 2, \ldots, p$,

$$\tilde{\alpha}_k = -C(A + KC)^{k-1} K, \qquad \tilde{\beta}_k = C(A + KC)^{k-1} B$$

Observe that for a sufficiently large p [such that the condition given in Eq. (34) holds] this model has the same internal structure as the ARX model considered earlier in Eq. (28), where the Kalman filter gain K now plays the role of the observer gain M_p. An important advantage of the structure in Eq. (35) is that the residual $\varepsilon(k)$ appears as an additive term. In the presence of noise, the identification of the coefficients $\tilde{\alpha}_k$ and $\tilde{\beta}_k$, $k = 1, 2, \ldots, p$, by the least-squares solution given in Eq. (12) minimizes this residual directly. The implication of this in our present control problem is that a large value of p will make the predictive controller more optimal in the sense that the identified ARX model now subsumes a Kalman filter. Explicit recovery of the state-space model and the corresponding Kalman filter gain is also possible as shown in Refs. 16–18, although this step is not necessary in the present predictive control approach. Again, to obtain this benefit, the coefficients of the ARX models must be identified directly from noise-contaminated data from Eq. (12).

Computational Procedure

In this section, we review the key steps that are involved in the design of these predictive controllers from a set of input–output data as follows.

1) Select a value p governing the speed of the implicit observer such that $pm \geq n$, where n is the assumed order of the system and

m is the number of outputs. Form the data matrices Y and V as specified in Eqs. (10) and (11). Compute the $m \times m$ coefficients α_i and the $m \times r$ coefficients β_i, $i = 1, 2, \ldots, p$, by Eq. (12).

2) Select a value q governing the speed of the implicit controller such that $qr \geq n$, where r is the number of inputs. Extract an $r \times pm$ matrix G from the first r rows of

$$-\left[A_p^{q-1}B_p, \ldots, A_pB_p, B_p\right]^+ A_p^q$$

where A_p, B_p, and C_p are defined in Eq. (17). The singular value decomposition should be used to compute the pseudoinverse, where the zero singular values are eliminated.

3) Let the p partitions of the $r \times pm$ matrix G be denoted by g_i, where each g_i is an $r \times m$ matrix, $i = 1, 2, \ldots, p$. Compute the controller gain matrices G_i and H_i from Eqs. (22). The dimension of each G_i is $r \times m$, and each H_i is $r \times r$. The control law is given by Eq. (21).

Discussion

In this section, we discuss the effect of choosing different values for p and q. The combination $(p_{\min}, q_{\min})$ produces a minimum-time controller that is equivalent to a state-feedback controller that has a deadbeat (minimum-time) observer and a deadbeat (minimum-time) state-feedback controller. This is a theoretical extreme and should not be used in practice. When p is minimum, the identification is sensitive to noise. Furthermore, when q is minimum, not only the controller requires unrealistic control effort, the extremely large gain makes the controller susceptible to noise and modeling errors. It is well known that identification accuracy improves as p increases, and for sufficiently large p, the identification results become optimal as the implicit observer converges to an optimal Kalman filter.[17–19] Also, as q increases, one also moves away from the deadbeat (minimum-time) solution toward the minimum-energy solution, making it less susceptible to noise. Thus, the combination $(p > p_{\min}, q > q_{\min})$ offers the advantage of having the implicit observer in the ARX model being more optimal and, at the same time, reducing the demand on the control energy. Note that in the predictive control context, the control input at any time step is part of a sequence of minimum-norm control actions that would bring the system states to zero in the next q steps. Therefore, the resultant control sequence is only an approximation of the truly energy-optimal solution. Generally speaking, it is best to choose a large p (for example, 2 or 3 times the assumed order of the system or more), and then to vary q to find a compromise between the speed at which the system states are driven to zero and the maximum allowable output of the control actuator. It has been our observation that suitable values of p and q can usually be achieved after a few iterations. In the noise-free case, of course, the minimum value for p can be used.

The controller formulated here is an interesting blend of both a feedforward and a feedback control. On the one hand, the controller is feedforward in that at each time step it determines its action to bring the future state to zero in a finite number of time steps. The predictive component is a feedforward action, which takes advantage of the knowledge of the system to guide its control action by looking ahead. On the other hand, the form of the controller is clearly feedback because the current control input is a linear combination of actual (past) input and output measurements. This feedback feature gives the controller the ability to handle unexpected disturbance as well as a certain degree of robustness with respect to both noise and modeling error. This feedback action compensates for the inherent weakness (sensitivity) of the feedforward action.

This design of predictive controllers involves some fine tuning of the parameters p and q. Because the meanings of p and q are clearly defined, however, workable designs can be produced with minimal effort. These controllers are essentially designed from an input–output model, which can be identified directly from input–output data. The calculation can be carried out recursively in real time if necessary. The fine tuning of p and q can also be done in real time as well. Also, these controllers make use of the implicit observer that is embedded in the identified model without requiring an explicit observer that one has to design separately. Strictly speaking, however, these controllers are not optimal in the sense of traditional state-space dynamic compensator designs satisfying a variety of design objectives and constraints. This type of design is, therefore, useful in engineering applications where tradeoff between design simplicity vs optimality is an issue.

Numerical Examples

This section illustrates various behaviors of the controlled system for different combinations of p and q. A sixth-order system will be used first to illustrate certain theoretical extremes, followed by a more realistic example of a truss structure. In the first example, we consider a three-degree-of-freedom flexible system

$$\begin{bmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ 0 & 0 & m_3 \end{bmatrix}\begin{bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \\ \ddot{x}_3 \end{bmatrix} + \begin{bmatrix} c_1 + c_2 & -c_2 & 0 \\ -c_2 & c_2 + c_3 & -c_3 \\ 0 & -c_3 & c_3 \end{bmatrix}\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} + \begin{bmatrix} k_1 + k_2 & -k_2 & 0 \\ -k_2 & k_2 + k_3 & -k_3 \\ 0 & -k_3 & k_3 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}$$

where m_i, k_i, and c_i, $i = 1, 2, 3$, are the mass, spring stiffness, and damping coefficients, respectively. The control force applied to each mass is denoted by u_i, $i = 1, 2, 3$. In the simulation, we use $m_1 = m_2 = m_3 = 1$ kg, $k_1 = k_2 = k_3 = 1000$ N/m, $c_1 = c_2 = c_3 = 0.1$ N s/m. The sampling interval is 0.01 s. Let the measurements y_i be the positions of the three masses, $y_i = x_i$, $i = 1, 2, 3$.

Example 1

Let us first consider a SISO case, where the input to the system is the force to the first mass and the output is the position of the third mass (noncollocated actuator–sensor). The smallest order of the ARX model p is $p_{\min} = 6$ corresponding to a deadbeat observer, and the smallest value for q is $q_{\min} = 6$ corresponding to a deadbeat controller. Using these values of $p_{\min}$ and $q_{\min}$, the controller $u_{(6,6)}(t)$ for this case is

$$\begin{aligned} u_{(6,6)}(k) &= u_1(k) \\ &= -5.51u_1(k-1) - 17.09_1u(k-2) \\ &\quad - 31.80u_1(k-3) - 22.92u_1(k-4) - 3.93u_1(k-5) \\ &\quad - 0.065_1u(k-6) + 10^5 \times [-0.992y_1(k-1) \\ &\quad + 3.88y_1(k-2) - 6.51y_1(k-3) + 5.79y_1(k-4) \\ &\quad - 2.71y_1(k-5) + 0.53y_1(k-6)] \end{aligned}$$

As an illustration, the system is driven by a random excitation for 2 s, after which the controller is turned on. The vibration is suppressed exactly in six time steps or 0.06 s, as shown in Figs. 1a and 1b. The solid curve is the system response with control. The dashed curve is the open-loop response without control. This example illustrates one theoretical extreme of the formulation. Clearly, the deadbeat solution is not practical because the control input is excessive and large gains are involved in the controller.

If we increase the prediction horizon, then the control energy will be reduced. This is the case where the controller is computed with $p_{\min} = 6$ and $q = 50$:

$$\begin{aligned} u_{(6,50)}(k) &= u_1(k) \\ &= -0.101u_1(k-1) - 0.081u_1(k-2) \\ &\quad - 0.0498u_1(k-3) - 0.0186u_1(k-4) \\ &\quad - 0.0025u_1(k-5) + 5.65y_1(k-1) + 5.01y_1(k-2) \\ &\quad - 73.83y_1(k-3) + 142.57y_1(k-4) \\ &\quad - 112.68y_1(k-5) + 33.58y_1(k-6) \end{aligned}$$

The corresponding input and output histories are shown in Figs. 2a and 2b. Note that it takes longer for the vibration to decay but the

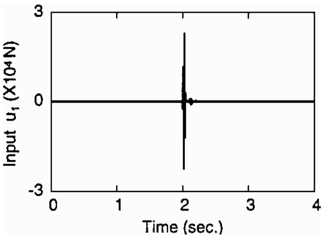

Fig. 1a Control input time history: $p = q = 6$.

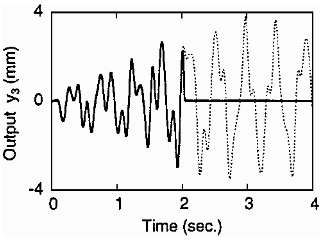

Fig. 1b Open- and closed-loop responses: $p = q = 6$.

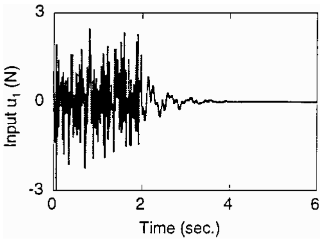

Fig. 2a Control input time history: $p = 6$ and $q = 50$.

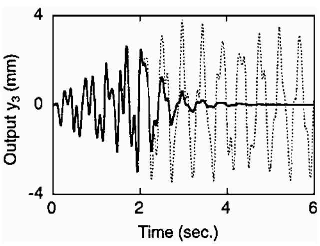

Fig. 2b Open- and closed-loop responses: $p = 6$ and $q = 50$.

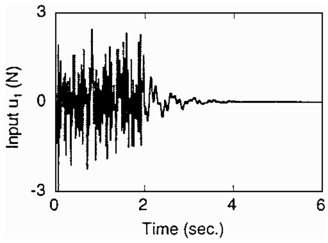

Fig. 3a Control input time history: $p = 2$ and $q = 50$.

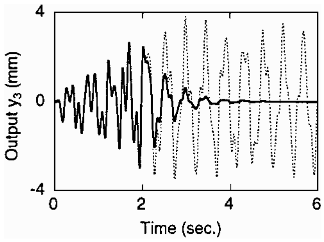

Fig. 3b Open- and closed-loop responses of the third mass: $p = 2$ and $q = 50$.

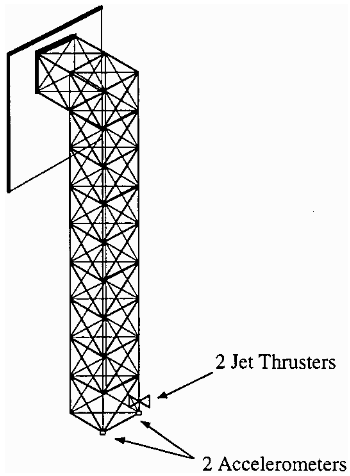

Fig. 4 Truss structure.

control effort is substantially reduced. The magnitudes of the controller gains are also reduced substantially.

Next, we consider the case where there are additional measurements available for feedback control (unequal number of inputs and outputs). In addition to the position of the third mass, position measurements of the two remaining masses are also available. This is a one-input/three-output system. In this case the minimum value of the order of the ARX model can now be reduced to $p_{\min} = 2$. For comparison, the prediction horizon is kept the same at $q = 50$. The controller in this case is

$$u_{(2,50)}(k) = u_1(k)$$

$$= -[0.682 \quad 0.444 \quad 0.740] \begin{bmatrix} y_1(k-1) \\ y_2(k-1) \\ y_3(k-1) \end{bmatrix}$$

$$+ [0.960 \quad 0.332 \quad 0.756] \begin{bmatrix} y_1(k-2) \\ y_2(k-2) \\ y_3(k-2) \end{bmatrix}$$

$$- 0.101 u_1(k-1) - 0.047 u_1(k-2)$$

Note that with the additional measurements, the current control only depends on the input and output in the past two time steps. This is because the implicit observer can estimate the state faster when additional sensors are involved. Figures 3a and 3b show the time histories of the control input to the first mass and the output of the third mass. Responses for the other outputs are similar and, hence, not shown here. In these examples, we have taken p to be minimum for illustration, but any larger value of p can be used as well.

Example 2

As a MIMO example of a more realistic system, the method is applied to a mathematical model of a truss structure, as shown in Fig. 4. The L-shaped 10 in. by 10 in. cross section aluminum truss is oriented such that is longer section is mounted in a vertical direction extending 90 in. The shorter section, 20 in. long, is horizontal and is clamped at one end to a wall-mounted rigid plate. The structure has two inputs, which are two cold air jet thrusters positioned at the tip. Two accelerometers located at a corner of the square cross section provide the in-plane tip acceleration measurements. A highly accurate 40th-order experimental model is used as the plant in the following simulation. This model is used to generate a set of input–output data from which a predictive controller is designed with $p = 5$ and $q = 150$. Typical of a sampled-data control system, the input signals are sampled and held constant in between sampling instants before entering the plant. The sampling rate is 250 Hz. We use the plant to generate input–output data from which the controller is designed; the controller has no explicit knowledge of the plant model itself. Figures 5a–5d show the results obtained by this controller, where the system is excited for 2 s, and then the controller is turned on to suppress the vibration. Again, the dashed curves are the uncontrolled responses, whereas the solid curves are the controlled responses.

A number of subtleties are involved in this example, and they are discussed here. First, the plant is 40th order, but the predictive controller is designed with $p = 5$. This is because the response is dominated by a few lightly damped modes so that a two-output ARX model with $p = 5$ is sufficient to capture this dominant behavior. To capture the dynamics of this 40th-order model fully, $p = 20$ should be used instead, but this turns out to be unnecessary, as one might expect. Second, in the discretization of a continuous system or in an experimentally identified model, the resultant discrete-time transfer

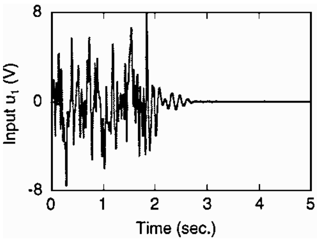

Fig. 5a Control input time history of the first input: $p = 5$ and $q = 150$.

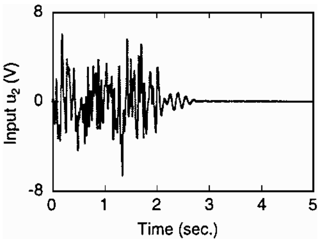

Fig. 5b Control input time history of the second input: $p = 5$ and $q = 150$.

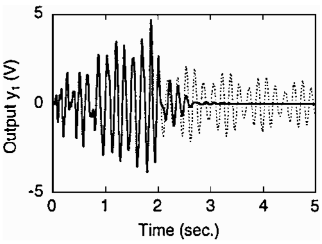

Fig. 5c Open- and closed-loop responses of the first output: $p = 5$ and $q = 150$.

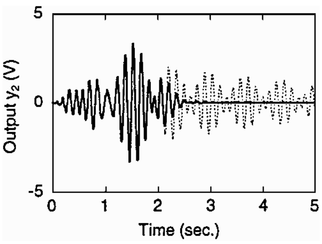

Fig. 5d Open and closed-loop responses of the second output: $p = 5$ and $q = 150$.

function commonly has zeros outside the unit circle in the complex plane (nonminimum phase).[20] This feature is also present in the discrete-time plant used in the simulation. As already discussed, the proposed predictive control design can be explained in terms of an observer-based state-feedback controller. Hence, unlike controllers that involve direct inversion of the dynamic model,[14] this predictive controller does not require the system to be minimum phase. This fact is also illustrated here.

Concluding Remarks

We have derived a class of predictive discrete-time controllers that are formulated from the state-space perspective, yet they do not require a state estimator in its implementation. In the final form, they have a simple linear dynamic feedback structure, where the current control input is a linear combination of past input and past output measurements up to a finite number of time steps. The controllers are designed from the ARX coefficients, which are computed directly from input and output data. The character of these controllers is governed by two parameters. One parameter is the prediction horizon as in predictive control. The other parameter is the order of the ARX model. This parameter is shown to relate to an observer in the equivalent state-space formulation. By varying these two parameters, one separately influences the speed of the implicit observer and the speed of the implicit state-feedback controller. In the final implementation of the control law, there is no explicit state-space model of the dynamic system, and no observer design is required. Because these predictive controllers have clear state-feedback interpretation, they retain the key advantages of a state-feedback controller over simplistic inverse controllers in that they do not require the system to be minimum phase nor do they require the number inputs to be equal to the number of outputs. A recursive version of the developed algorithm can be developed for real-time implementation, which includes adaptive tuning of the design parameters p and q for optimal performance. Inasmuch as recursive estimation will be part of such a scheme, robustness to parameter variation will also be enhanced because such a controller will have the opportunity to adjust itself to the system dynamics, which may be slowly changing.

References

[1]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.

[2]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.

[3]Peterka, V., "Predictor-Based Self-Tuning Control," *Automatica*, Vol. 20, No. 1, 1984, pp. 39–50.

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

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

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

[7]Goodwin, G. C., and Sin, K. S., *Adaptive Filtering, Prediction, and Control*, Prentice–Hall, Englewood Cliffs, NJ, 1984.

[8]Astrom, K. J., and Wittenmark, B., *Adaptive Control*, Addison–Wesley, Reading, MA, 1989.

[9]Soeterboek, R., *Predictive Control: A Unified Approach*, Prentice–Hall, Englewood Cliffs, NJ, 1992.

[10]Mosca, E., *Optimal, Predictive, and Adaptive Control*, Prentice–Hall, Englewood Cliffs, NJ, 1995.

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

[12]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.

[13]Heise, S. A., and Maciejowski, J. M., "Model Predictive Control of a Supermaneuverable Aircraft," *Proceedings of the AIAA Guidance, Navigation, and Control Conference* (San Diego, CA), AIAA, Reston, VA, 1996.

[14]Horta, L. G., and Sandridge, C. A., "On-line Indentification of Forward/Inverse Systems for Adaptive Control Applications," *Proceedings of the Guidance, Navigation, and Control Conference* (Hilton Head, SC), AIAA, Washington, DC, 1992 (AIAA 92-4384).

[15]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.

[16]Chen, C.-W., Huang, J.-K., Phan, M., and Juang, J.-N., "Integrated System Identification and Modal State Estimation for Control of Large Flexible Space Structures," *Journal of Guidance, Control, and Dynamics*, Vol. 15, No. 1, 1992, pp. 88–95.

[17]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.

[18]Juang, J.-N., *Applied System Identification*, Prentice–Hall, Englewood Cliffs, NJ, 1994.

[19]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, April 1995, pp. 232–239.

[20]Narendra, K. S., and Annaswamy, A. M., *Stable Adaptive Systems*, Prentice–Hall, Englewood Cliffs, NJ, 1989, p. 374.