

Available online at www.sciencedirect.com



Control Engineering Practice 11 (2003) 261-278

CONTROL ENGINEERING PRACTICE

www.elsevier.com/locate/conengprac

A data driven subspace approach to predictive controller design

Ramesh Kadali^{a,1}, Biao Huang^{a,*}, Anthony Rossiter^b

^a Department of Chemical and Materials Engineering, 536 Chemical–Materials Eng Building, University of Alberta, Edmonton, Alberta, Canada T6G 2G6

^b Department of Automatic Control and Systems Engineering, Sheffield University, Mappin Street, Sheffield, SI 3JD, UK

Received 7 June 2001; accepted 13 May 2002

Abstract

This paper shows the design of predictive controllers using the predictor, designed from the subspace matrices, obtained directly from the input/output data. The model-free design approach presented in the literature so far does not include all the important predictive control features such as inclusion of an integrator for offset-free control, constraint handling, feedforward option and a means of tuning the controllers through the disturbance model; these features are important for practical applications and hence, amongst other issues, addressed in this paper. The proposed predictive controller is demonstrated on multivariate systems using MATLAB simulations and an application on a pilot scale process.

© 2002 Elsevier Science Ltd. All rights reserved.

Keywords: Model predictive control; Data driven approach; Model free; Subspace identification; Subspace matrices; Generalized predictive control; Feedforward control; Constraint handling; Noise model

1. Introduction

Predictive controllers have been widely used in process industries for more than two decades (Camacho & Bordons, 1999; Lee & Cooley, 1996; Morari, 1994; Qin & Badgwell, 1996). Several forms of predictive controllers such as IDCOM (Richalet, Rault, Testud, & Papon, 1978), DMC (Cutler & Ramaker, 1979, 1980; Lee & Cooley, 1996) QDMC (Cutler, Morshedi, & Haydel, 1983; Garcia & Morshedi, 1986) GPC (Clarke, Mohtadi, & Tuffs, 1987a, b), etc. have been proposed and successfully implemented in the process industries through the years. The term predictive control does not designate a specific control strategy but a wide range of control algorithms which make an explicit use of a process model in a cost function minimization to obtain the control signal (Camacho & Bordons, 1999; Garcia, Prett, & Morari, 1989). Hence a model of the process is the basic requirement for the design of predictive

E-mail address: biao.huang@ualberta.ca (B. Huang).

controllers; this is first identified using plant input and output data. From the process model, predictor matrices can be obtained (for example, the dynamic matrix constructed using step response coefficients in DMC (Cutler & Ramaker, 1979, 1980)). The predictor matrices are used to obtain predictions for the process output which are used in the controller design. However, it has been found recently that these predictor matrices can be obtained directly from the input/output data by using the subspace matrices (identified as a first step in the subspace identification methods), eliminating the intermediate step of process model identification and providing a means for designing a predictive controller, in the generalized predictive controller (GPC) framework (e.g. Rossiter & Kouvaritakis, 2001; Rossiter & Yao, 2000), without a parametric model. Since no traditional parametric model is required for the controller design this approach is also referred to as the "model-free approach", and this term has been adopted in the literature (for example see Favoreel, De Moor, Van Overschee, & Gevers, 1999; Stenman, 1999). The idea is to obtain the controller matrices used in the predictive controllers directly from the data without the intermediate parametric model identification step. Hence this approach can also be considered as a direct

^{*}Corresponding author. Tel.: +1-403-492–9016; fax: +1-403-492-2881.

¹Currently working for Suncor Energy Inc., Fort McMurray, AB, Canada.

data driven approach. Moreover, subspace matrices identification involves minimizing the summation of multi-step ahead prediction errors, making the subspace matrices based design approach appropriate for predictive control.

The predictive controller based on subspace matrices uses the same cost-function as GPC and hence an important question is how one obtains the predictions utilized within the cost function. One of the key aspects in GPC is the assumption of an ARIMAX model for the process (Bitmead, Gevers, & Wertz, 1990; Clarke et al., 1987a). This requires pre-specification of the order and structure of the model to be identified for controller design. Typically one uses reduced complexity models which frequently introduces bias errors. Moreover, the model is usually identified in a nonlinear, iterative manner, and in general Diophantine equations need to be solved to obtain the prediction matrices. On the other hand, the predictive controller designed using subspace matrices makes no pre-assumptions about the structure and order of the process model (alleviating some bias errors). Moreover the prediction matrices are obtained through a single matrix algebraic calculation. In summary, the subspace approach to predictive control has the key features of GPC (Clarke et al., 1987a, b) such as: (1) long-range prediction over a finite horizon; (2) inclusion of weighting on outputs and control moves in the costfunction and (3) choice of a prediction horizon and a control horizon after which projected control moves are taken to be zero. It combines these with the added advantages of: (1) no pre-assumptions about model order or structure; (2) parametric matrices obtained in a single iteration and (3) not having to solve Diophantine equations. We also note that the extension to the multivariate case is straightforward with the subspace approach.

Although the idea of designing predictive controllers using the subspace matrices, such as model-free LQG and subspace predictive controller (Favoreel & De Moor, 1998; Favoreel, De Moor, Gevers, & van Overschee, 1998; Favoreel et al., 1999; Rossiter & Yao, 2000), or using the state space model identified through subspace approach (Ruscio, 1997b, c; Ruscio & Foss, 1998), has been around for a few years, designing a predictive controller from subspace matrices with all the features of the traditional predictive controller has not been investigated fully. The equivalence of finite horizon LQG to GPC is well known (Bitmead et al., 1990) however there are several other important issues that need to be addressed in the subspace predictive control framework and they form the main contribution of this paper. The following issues are considered: (1) derivation of a predictive control law in the GPC framework (with systematic inclusion of integral action, an issue ignored in previous works); (2) extension of the

predictive control law to include feedforward control to compensate for measured disturbances; (3) inclusion of a constraint handling facility and (4) tuning of the noise model.

The paper is arranged as follows. Section 2 gives an overview of GPC design followed by a discussion of important features of the open loop subspace identification method in Section 3. Subspace approach to the predictive controller design with enhanced features is explained in Section 4. Inclusion of the independent noise model for tuning is discussed in Section 5. Results from the simulation and actual implementation on a pilot scale plant using the proposed predictive control scheme are presented in Section 6 and Section 7 respectively. The conclusions are presented in Section 8.

2. Revisit of GPC

GPC design (Bitmead et al., 1990; Clarke et al., 1987a, b) starts by first identifying an ARIMAX model for the process, expressed as

$$A(z^{-1})y_t = B(z^{-1})u_{t-1} + \frac{C(z^{-1})}{\triangle}e_t,$$
(1)

where A, B and C are polynomials in the back shift operator, z^{-1} , with A and C being monic. $\Delta = (1 - z^{-1})$ is the differencing operator. The role of the Δ is to ensure integral action in the controller by including an internal disturbance model of typical load perturbations arising in the process control industry (Bitmead et al., 1990). A popular quadratic cost function to be minimized is

$$J = \sum_{k=N_1}^{N_2} (r_{t+k} - \hat{y}_{t+k|t})^2 + \sum_{k=1}^{N_u} \lambda (\Delta u_{t+k-1})^2$$
(2)

with N_2 and N_u being the prediction and control horizons respectively and λ being the weighting on the control effort. N_1 is usually chosen as 1 or the process time delay t_d . r_{t+k} is the future setpoint for the time instant t + k. For a discussion on the selection of values for N_1 , N_2 , N_u and λ , readers are referred to (Clarke et al., 1987a, b). Using the Diophantine equations

$$\frac{C(z^{-1})}{A(z^{-1})\Delta} = E_k + q^{-k} \frac{F_k}{A(z^{-1})\Delta},$$
(3)

$$E_k B = G_k C + q^{-k} \Gamma_k, \tag{4}$$

along with Eq. (1) (and ignoring the term $E_k e_{t+k}$) we obtain the k-step ahead output prediction equation

$$\hat{y}_{t+k} = \frac{F_k}{C} y_t + \frac{\Gamma_k}{C} \bigtriangleup u_{t-1} + G_k \bigtriangleup u_{t+k-1}$$
$$= F_k y_t^f + \Gamma_k \bigtriangleup u_{t-1}^f + G_k \bigtriangleup u_{t+k-1}$$
$$= f(k) + G_k \bigtriangleup u_{t+k-1}, \tag{5}$$

where

$$\triangle u_t^f = C^{-1} \triangle u_t; \quad y_t^f = C^{-1} y_t$$

and $f(k) = F_k y_t^f + \Gamma_k u_{t-1}^f$ is the free response of the process. Define the vectors of predictions

$$\Delta u_{f} = [\Delta u_{t} \dots \Delta u_{t+k} \dots \Delta u_{t+N_{u}}]^{\mathrm{T}}; \hat{y}_{f} = [\hat{y}_{t+1} \dots \hat{y}_{t+k} \dots \hat{y}_{t+N_{2}}]^{\mathrm{T}} r_{f} = [r_{t+1} \dots r_{t+k} \dots r_{t+N_{2}}]^{\mathrm{T}}; F_{f} = [f(1) \dots f(k) \dots f(N_{2})]^{\mathrm{T}}$$

then the multi step predictor equations can be expressed as

$$\hat{y_f} = G \triangle u_f + F_f, \tag{6}$$

where G is the dynamic matrix containing the step response coefficients of B/A or the impulse response coefficients of $B/A \triangle$. Substituting the above equation in (2) we can derive the GPC control law

$$\Delta u_f = (G^{\mathrm{T}}G + \lambda I)^{-1}G^{\mathrm{T}}(r_f - F_f).$$
⁽⁷⁾

3. Subspace identification method

This section provides the necessary background on the identification of subspace matrices from open loop data, that will be used in the subsequent sections for designing a predictive controller. This section is based on the combined deterministic and stochastic subspace state space identification method for multivariate systems presented in Van Overschee and De Moor (1994), Van Overschee and De Moor (1996). Other variations in subspace identification, for example CVA (Larimore, 1996, 1990), DSR (Ruscio, 1997a, d) and MOESP (Chou & Verhaegen, 1997), and the use of weighting matrices in subspace identification are not presented here. They can be found in the standard book on subspace identification such as Van Overschee and De Moor (1996) and the references therein.

A linear time-invariant system can be described in a state space innovation form as:

$$x_{k+1} = Ax_k + Bu_k + K^{j} e_k, (8)$$

$$y_k = Cx_k + Du_k + e_k,\tag{9}$$

where u_k , y_k and x_k are the process inputs, outputs and states respectively. K^f is the Kalman filter gain and e_k is an unknown innovation sequence with the following covariance matrix:

$$E[e_k e_k^{\mathrm{T}}] = S. \tag{10}$$

For an *l*-input and *m*-output system, *A*, *B*, *C*, *D*, K^f and *S* are $(n \times n)$, $(n \times l)$, $(m \times n)$, $(m \times l)$, $(n \times m)$ and $(m \times m)$ matrices respectively, where *n* is the state order. e_k is

 $(m \times 1)$ and S is the innovations covariance matrix. Suppose the measurements of the inputs and the outputs u_k , y_k for $k \in \{1, 2, ..., 2i + j - 1\}$ are available. The data block Hankel matrices for u_k , represented as U_p and U_f , with *i*-block rows and *j*-block columns are defined as

$$U_{p} = \begin{bmatrix} u_{1} & u_{2} & \dots & u_{j} \\ u_{2} & u_{3} & \dots & u_{j+1} \\ \dots & \dots & \dots & \dots \\ u_{i} & u_{i+1} & \dots & u_{i+j-1} \end{bmatrix};$$

$$U_{f} = \begin{bmatrix} u_{i+1} & u_{i+2} & \dots & u_{i+j} \\ u_{i+2} & u_{i+3} & \dots & u_{i+j+1} \\ \dots & \dots & \dots & \dots \\ u_{2i} & u_{2i+1} & \dots & u_{2i+j-1} \end{bmatrix}.$$
(11)

Each block element in the above data Hankel matrices is a column vector of inputs, i.e., $u_i = [u_{i1} \ u_{i2} \ \dots \ u_{il}]^{\mathrm{T}}$. Similar data Hankel matrices for y_k , represented as Y_p and Y_f , can be written. The past and future state sequences are defined as

$$X_p = [x_0 \quad x_1 \quad \dots \quad x_{j-1}];$$

$$X_f = [x_i \quad x_{i+1} \quad \dots \quad x_{i+j-1}].$$
(12)

The matrix input-output equations used in subspace identification (Van Overschee & De Moor, 1994, 1995, 1996) are obtained by recursive substitution of Eqs. (8)–(9):

$$Y_f = \Gamma_i X_f + H_i U_f + H_i^s E_f.$$
⁽¹³⁾

The prediction expressions of the output, Y_f , can be written, for $i, j \rightarrow \infty$, as

$$\hat{Y}_f = \Gamma_i X_f + H_i U_f \tag{14}$$

$$=L_w W_p + L_u U_f, (15)$$

where $W_p = \begin{bmatrix} Y_p \\ U_p \end{bmatrix}$, Γ_i ($im \times n$) is the extended observability matrix and, H_i ($im \times il$) and H_i^s ($im \times im$) are the lower triangular Toeplitz matrices containing the impulse response coefficients corresponding to the deterministic input u_k and the unknown stochastic input e_k , respectively. p and f denote the past and the future respectively. The subscript i follows from the number of row blocks in the block Hankel matrices as shown above. The transition from Eq. (14) to Eq. (15) is shown in Appendix A.

$$\Gamma_i = [C^{\mathrm{T}} \quad (CA)^{\mathrm{T}} \quad \dots \quad (CA^{i-1})^{\mathrm{T}}]^{\mathrm{T}};$$
 (16)

IR Model of deterministic input

IR Model of stochastic input

Fig. 1. Subspace representation.

$$H_{i} = \begin{bmatrix} D & 0 & \dots & 0 \\ CB & D & \dots & 0 \\ \dots & \dots & \dots & \dots \\ CA^{i-2}B & CA^{i-3}B & \dots & D \end{bmatrix};$$

$$H_{i}^{s} = \begin{bmatrix} I_{m} & 0 & \dots & 0 \\ CK^{f} & I_{m} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ CA^{i-2}K^{f} & CA^{i-3}K^{f} & \dots & \dots \end{bmatrix}.$$
(17)

See Appendix A for more information on subspace matrices and the estimation of state space system matrices from the subspace matrices. See Fig. 1 for the model parameters captured in the subspace matrices. As will be shown in the next section, the state space system matrices are not required for the design of the predictive controller. Hence we are interested in identifying only the subspace matrices.

If (i) the deterministic input u_k is uncorrelated with e_k , (ii) u_k is persistently exciting of the order 2i, and (iii) the measurements go to infinity, $j \rightarrow \infty$, the open loop models can be consistently identified and identification involves finding the prediction of the future outputs Y_f using a linear predictor. The least-squares prediction \hat{Y}_f can be found by solving the least squares problem:

$$\min_{L_w,L_u} \left| Y_f - (L_w \quad L_u) \begin{pmatrix} W_p \\ U_f \end{pmatrix} \right|_F^2$$
(18)

 \hat{Y}_f is found by the orthogonal projection of the row space of Y_f into the row space spanned by W_p and U_f defined as (Van Overschee & De Moor, 1996)

$$\hat{Y}_f = \left. Y_f \right/ \left[egin{array}{c} W_p \\ U_f \end{array}
ight]$$

$$\begin{bmatrix} L_w & L_u \end{bmatrix} = Y_f \begin{bmatrix} W_p \\ U_f \end{bmatrix}^{\mathsf{T}}$$
$$= Y_f [W_p^{\mathsf{T}} & U_f^{\mathsf{T}}] \left(\begin{bmatrix} W_p \\ U_f \end{bmatrix} [W_p^{\mathsf{T}} & U_f^{\mathsf{T}}] \right)^{-1}, \quad (19)$$

where [†] represents the Moore–Penrose pseudo-inverse. This projection can be implemented in a numerically robust way with a QR-decomposition (Van Overschee & De Moor, 1994, 1995, 1996; Ruscio, 1997a; Verhaegen, 1994) or using PLS (Ruscio, 1997d).

4. Predictive controller design from subspace matrices

Consider a controller objective function which is the same as that of GPC. To simplify the notation, assume $N_1 = 1$. The cost function to be minimized becomes

$$J = \sum_{k=1}^{N_2} (r_{t+k} - \hat{y}_{t+k|t})^2 + \sum_{k=1}^{N_u} \lambda (\Delta u_{t+k-1})^2$$
$$= (r_f - \hat{y}_f)^{\mathrm{T}} (r_f - \hat{y}_f) + \Delta u_f^{\mathrm{T}} (\lambda I) \Delta u_f, \qquad (20)$$

where the future outputs are over the prediction horizon, t + 1 to $t + N_2$, and the future incremental inputs are over the control horizon, t to $t + N_u - 1$. For the state space representation (8)–(9), the vector of the optimal prediction of the future outputs can be expressed in terms of the future inputs and current states as

$$\hat{y}_f = [\hat{y}_{t+1} \quad \dots \quad \hat{y}_{t+N_2}]^{\mathrm{T}}$$
 (21)

$$= \begin{bmatrix} C \\ CA \\ ... \\ CA^{N_2-1} \end{bmatrix} x_t + \begin{bmatrix} D & 0 & 0 & ... \\ CB & D & 0 & ... \\ ... & ... & ... \\ CA^{N_2-2}B & ... & D & ... \end{bmatrix} \times \begin{bmatrix} u_t \\ u_{t+1} \\ ... \\ u_{t+N_u-1} \end{bmatrix}$$
(22)

$$=\Gamma_{N_2}x_t + H(1:mN_2,1:lN_u)u_f$$
(23)

$$= L_w(1:mN_2,:)w_p + L_u(1:mN_2,1:lN_u)u_f,$$
(24)

where $w_p = [y_{t-i+1} \dots y_t \ u_{t-i} \dots u_{t-1}]^T$ and $u_f = [u_t \dots u_{t+N_u-1}].$

The predictor equation in Eq. (24) is used in minimizing the objective function

$$J = \sum_{k=1}^{N_2} (r_{t+k} - \hat{y}_{t+k|t})^2 + \sum_{k=1}^{N_u} \lambda (u_{t+k-1})^2$$
(25)

to derive, for a finite $\{N_2, N_u\}$, the 'subspace predictive control (SPC)' law presented in Favoreel and De Moor (1998), which computes the future control moves as

$$u_f = (\lambda I + L_u^{\rm T} L_u)^{-1} L_u^{\rm T} (r_f - L_w w_p),$$
(26)

where $w_p = W_p(:, 1)$. As $\{N_2, N_u\} \rightarrow \infty$, the above SPC becomes an LQG-controller presented in Favoreel et al. (1998), Favoreel et al. (1999). However, for implementation on real processes the controller should have an integrator since the objective function (25) does not admit zero static error in the case of non-zero constant reference unless the open loop process contains an integrator (Bitmead et al., 1990). Hence we need to use the GPC objective function, with incremental inputs Δu_f , shown in Eq. (20). One of the several subspace matrices based predictive controller design approaches presented in Ruscio (1997b, c), Ruscio and Foss (1998) has also included an integrator. In their method, to get an integrator in the predictor, Eq. (24) is multiplied on both sides with a difference operator, $\triangle = 1 - z^{-1}$, where z^{-1} is the back shift operator, and then rearranged to get a predictor equation with incremental inputs and outputs. A slightly different subspace identification method called DSR (Ruscio, 1997a, d) is used in their approaches.

In the next section we present a different approach to get incremental variables in the predictor equation. The new approach uses an integrated noise model. As the subspace model in Eq. (15) is I/O based, it is logical to use a similar technique to that adopted in conventional GPC (Clarke et al., 1987a, b). As will become clear later on, the new approach is equivalent to the original GPC design since the innovations form state space representation in Eqs. (8)–(9) combined with integrated noise assumption is equivalent to ARIMAX representation.

4.1. Inclusion of integral action through integrated noise model

Consider the noise input e_t as an integrating noise, which is common in the process industries. Therefore,

$$e_{k+1} = e_k + a_k, \tag{27}$$

$$e_k = \frac{a_k}{\Delta},\tag{28}$$

where a_k is a white noise signal and $\triangle (= 1 - z^{-1})$ is a differentiating operator. Note that the system considered in Eqs. (8)–(9) together with (28) is equivalent to an ARIMAX representation, as in Eq. (1), considered in the GPC design. Substituting Eq. (27) in (8)–(9), we obtain

$$z_{k+1} = Az_k + B \triangle u_k + K' a_k, \tag{29}$$

$$\Delta y_k = C z_k + D \Delta u_k + a_k, \tag{30}$$

where $z_k = x_k - x_{k-1}$. The subspace matrix input–output expressions for the system (29)–(30) are

$$\Delta Y_f = \Gamma_i Z_f + H_i \Delta U_f + H_i^s A_f, \tag{31}$$

and

$$\Delta \hat{y}_f = \Gamma_i z_k + H_i \Delta u_f \tag{32}$$

$$= L_w \begin{bmatrix} \Delta y_p \\ \Delta u_p \end{bmatrix} + L_u \Delta u_f.$$
(33)

Using the system representation (29)–(30) we can write a k-step ahead predictor as

$$y_{t+k-1} - y_{t-1} = (CA^{k-1} + \dots + CA + C)z_t + [(CA^{k-2}B + \dots + D) \triangle u_t + \dots + D \triangle u_{t+k-1}] + [a_t + a_{t+1} + \dots + a_{t+k-1}]$$
(34)

and Eqs. (21)-(22) change to

$$\hat{y}_f = [\hat{y}_{t+1} \ \hat{y}_{t+2} \ \dots \ \hat{y}_{t+N_2}]^{\mathrm{T}}$$
 (35)

$$= \mathbf{y}_{\mathbf{t}} + \Gamma_{N_2}^{\circ} z_t + S_{N_2,N_u} \Delta u_f \tag{36}$$

$$= \mathbf{y}_{\mathbf{t}} + L_{w}^{\circ}(1:N_{2}m,:) \begin{bmatrix} \Delta y_{p} \\ \Delta u_{p} \end{bmatrix} + S_{N_{2},N_{u}} \Delta u_{f}$$
(37)

$$=F+S_{N_2,N_u} \bigtriangleup u_f, \tag{38}$$

where $\Gamma_{N_2}^{\circ}$ is the modified extended observability matrix and S_{N_2,N_u} is the $(N_2m \times N_ul)$ dynamic matrix containing the step response coefficients/Markov parameters and formed from L_u .

$$\mathbf{y}_{\mathbf{t}} = \begin{bmatrix} y_t & y_t & \dots & y_t \end{bmatrix}^{\mathrm{T}},\tag{39}$$

$$\Gamma_{N_{2}}^{\circ} = \begin{bmatrix} C \\ CA + C \\ ... \\ CA^{N_{2}-1} + \dots + C \end{bmatrix},$$
(40)

 $S_{N_2,N_u} =$

$$\begin{bmatrix} D & 0 & 0 & \cdots & 0 \\ CB+D & D & 0 & \cdots & 0 \\ CAB+CB+D & CB+D & D & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ CA^{N_2-2}B+\cdots+CB+D & CA^{N_2-3}B+\cdots+CB+D & \cdots & \cdots \end{bmatrix}$$

$$= L_{u}(1:N_{2}m,1:N_{u}l) \begin{bmatrix} I_{l} & 0 & \dots & 0\\ I_{l} & I_{l} & \dots & 0\\ \dots & \dots & \dots & \dots\\ I_{l} & I_{l} & \dots & I_{l} \end{bmatrix}.$$
 (41)

 L_w° is constructed from L_w as

$$L_{w}^{\circ}(m(k-1)+1:mk,:) = \sum_{i=1}^{k} L_{w}(m(i-1)+1:mi,:)$$

for $1 \le k \le N_{2}$ (42)

and F is the free response of the process output.

$$F = \mathbf{y}_{\mathbf{t}} + L_{w}^{\circ}(1:N_{2}m,:) \begin{bmatrix} \Delta y_{p} \\ \Delta u_{p} \end{bmatrix}.$$
(43)

Note that the matrices L_w° and S_{N_2,N_u} are related in a simple manner to L_w and L_u . Even though L_w° and S_{N_2,N_u} can be alternatively directly identified from the differentiated data, it is difficult to design an identification signal for such an identification. Hence, a simple strategy is to identify L_w and L_u and use these matrices to form L_w° and S_{N_2,N_u} .

The objective function in Eq. (20) can be expanded as

$$J = (r_f - F - S_{N_2, N_u} \bigtriangleup u_f)^{\mathrm{T}} (r_f - F - S_{N_2, N_u} \bigtriangleup u_f) + \bigtriangleup u_f^{\mathrm{T}} (\lambda I) \bigtriangleup u_f.$$
(44)

Differentiating J with respect to $\triangle u_f$ and equating it to zero gives the control law

$$\Delta u_f = (S_{N_2,N_u}^{\rm T} S_{N_2,N_u} + \lambda I)^{-1} S_{N_2,N_u}^{\rm T} (r_f - F).$$
(45)

Only $\triangle u_f(1)$ is implemented and the calculation is repeated at each time instant. Hence at time instant *t*, we only calculate

$$\Delta u_t = \Delta u_f(1) = m_l(r_f - F), \tag{46}$$

where m_l is made of the first *l*-rows of the matrix $(S_{N_2,N_u}^{\rm T}S_{N_2,N_u} + \lambda I)^{-1}S_{N_2,N_u}^{\rm T}$. Therefore u_t is

implemented as

$$u_{t} = u_{t-1} + \Delta u_{t}$$

$$= u_{t-1} + m_{l} \left\{ r_{f} - \left\{ \mathbf{y}_{t} + L_{w}^{\circ}(1:N_{2}m,:) \begin{bmatrix} \Delta y_{p} \\ \Delta u_{p} \end{bmatrix} \right\} \right\}.$$
(47)

Note that the above control law has a guaranteed integral control action and obtained directly from the subspace matrices, without any intermediate parametric model identification step.

4.2. Inclusion of feedforward control

If some of the process disturbances are measurable, then with the understanding that measured disturbances are those process input variables which cannot be manipulated for controlling the process outputs, the state space representation of the process (8)–(9) can be modified as

$$x_{k+1} = Ax_k + \begin{bmatrix} B & B_v \end{bmatrix} \begin{bmatrix} u_k \\ v_k \end{bmatrix} + K^f e_k,$$
(48)

$$y_k = Cx_k + \begin{bmatrix} D & D_v \end{bmatrix} \begin{bmatrix} u_k \\ v_k \end{bmatrix} + e_k,$$
(49)

where $v_t(h \times 1)$ is the vector of measured disturbance variables. The matrix input-output Eqs. (13)-(15) change to

$$Y_f = \Gamma_i X_f + H_i U_f + H_i^v V_f + H_i^s E_f,$$
(50)

$$\hat{Y}_f = \Gamma_i X_f + H_i U_f + H_i^v V_f \tag{51}$$

$$=L_w^{\mp}W_p^{\mp}+L_uU_f+L_vV_f,$$
(52)

where

$$W_p^{\mp} = \begin{bmatrix} Y_p & U_p & V_p \end{bmatrix}^{\mathrm{T}}$$

with V_p and V_f being the *past* and *future* data Hankel matrices of v_t defined in the same way as those corresponding to u_t in (11). The subspace matrices L_w^{\mp} , L_u and L_v are obtained by finding the prediction of future outputs, Y_f , by solving the least-squares problem

$$\min_{L_w,L_u,L_v} \left| Y_f - (L_w \quad L_u \quad L_v) \begin{pmatrix} W_p^{\mp} \\ U_f \\ V_f \end{pmatrix} \right|_F^2.$$
(53)

The solution is obtained as explained in Section 3. For predictive control we have the values of measured disturbance only upto the current sampling instant, t, (and do not have the knowledge of the future values of measured disturbance) i.e., v_k for k = t, t - 1, t - 2, ... are known but v_{t+k} for $k = 1, 2, ..., N_2$ are not available for the prediction of \hat{y}_{t+k} . Therefore we can write the

prediction expression for \hat{y}_f as

$$\hat{y}_f = \mathbf{y}_{\mathbf{t}} + L_w^{\pm}(1:N_2m,:) \begin{bmatrix} \bigtriangleup y_p \\ \bigtriangleup u_p \\ \bigtriangleup v_p \end{bmatrix} + S_{N_2,N_u} \bigtriangleup u_f$$
(54)

$$=F^{+}+S_{N_2,N_u}\Delta u_f,$$
(55)

where L_w^{\pm} is constructed from L_w^{\mp} , and F^{\mp} is the free response for the case of measured disturbances.

$$L_{w}^{\pm}(m(k-1)+1:mk,:) = \sum_{i=1}^{k} L_{w}^{\mp}(m(k-1)+1:mk,:)$$

$$1 \leq k \leq N_{2},$$
 (56)

$$F^{\mp} = \mathbf{y}_{\mathbf{t}} + L_{w}^{\pm}(1:N_{2}m,:) \begin{bmatrix} \Delta y_{p} \\ \Delta u_{p} \\ \Delta v_{p} \end{bmatrix}.$$
(57)

 S_{N_2,N_u} is the same as defined before in Eq. (41). Therefore, the feedback plus feedforward control law becomes

$$\Delta u_f = (S_{N_2,N_u}^{\mathrm{T}} S_{N_2,N_u} + \lambda I)^{-1} S_{N_2,N_u}^{\mathrm{T}} (r_f - F^{\mp}).$$
(58)

4.3. Constraint handling

Constraints arise due to physical limitations, quality specifications, safety concerns and limiting the wear of the equipment. One of the main features of MPC, its prediction capability, is useful in anticipating constraint violations and correcting them in an appropriate way (Camacho & Bordons, 1999). The explicit handling of constraints may allow the process to operate closer to optimal operating conditions (Camacho & Bordons, 1999). For the constrained case, the computations are more involved. The problem takes the form of a standard Quadratic Programming (QP) formulation and the optimization is done numerically. The quadratic program solved at every instant is

$$\min_{\Delta u} J = (r_f - F)^{\mathrm{T}} (r_f - F) + \Delta u^{\mathrm{T}} (S_{N_2, N_u}^{\mathrm{T}} S_{N_2, N_u}) \Delta u$$
$$- 2(r_f - F)^{\mathrm{T}} S_{N_2, N_u} \Delta u$$
$$= \Delta u^{\mathrm{T}} (S_{N_2, N_u}^{\mathrm{T}} S_{N_2, N_u} + \lambda I) \Delta u$$
$$- 2(r_f - F)^{\mathrm{T}} S_{N_2, N_u} \Delta u$$
$$= \frac{1}{2} \Delta u^{\mathrm{T}} P \Delta u + c^{\mathrm{T}} \Delta u,$$
(59)

s.t.
$$\Theta \bigtriangleup u \leqslant \Psi$$
, (60)

where

$$P = S_{N_2,N_u}^{\rm T} S_{N_2,N_u} + \lambda I, \tag{61}$$

$$c = -2S_{N_2,N_u}^{\mathrm{T}}(r_f - F),$$
 (62)

matrices Θ and Ψ are formed from the constraints (see Appendix B). The optimization of the above QP

formulation is carried out by means of the standard commercial optimization QP code at each sampling instant and then the value of u_t is sent to the process. Though computationally more involved than the other simpler algorithms, the flexible constraints handling capabilities of predictive controllers are very attractive for practical applications, since the economic operating point of a typical process unit often lies at the intersection of constraints (Prett & Morari, 1980; Camacho & Bordons, 1999). For more discussion on other types of constraints, for example soft constraints or specifications on the process response characteristics, readers are referred to Camacho and Bordons (1999), Garcia et al. (1989), Lee and Cooley (1996), Qin and Badgwell (1996) and the references therein.

5. Tuning the noise model

The disturbance dynamics of industrial processes frequently change with time. Tuning of the noise model is a key feature of predictive controller formulations such as GPC (Bitmead et al., 1990). It is necessary to incorporate such a feature in the proposed predictive controller derived from subspace matrices. For this reason we need to separate the state space model of the system in Eqs. (8)–(9) into two parts, a deterministic part and a stochastic part which are similar to the process model and noise model in an equivalent input– output transfer function framework as

$$y_t = y_t^a + y_t^s$$

= $[C(zI - A)^{-1}B + D]u_t + [C(zI - A)^{-1}K^f + 1]e_t.$
(63)

It is observed that both deterministic and stochastic parts have the same poles. Hence an equivalent representation for the above equation in the discrete transfer function domain would be an ARMAX model

$$y_t = \frac{G(z^{-1})}{H(z^{-1})}u_t + \frac{F(z^{-1})}{H(z^{-1})}e_t.$$
(64)

 $H(z^{-1})$ and $F(z^{-1})$ are monic polynomials in z^{-1} . We can write

$$[C(zI - A)^{-1}K^{f} + 1] = 1 + CK^{f}z^{-1} + CAK^{f}z^{-2} + \cdots$$
(65)

The assumption that the noise model $F(z^{-1})/A(z^{-1}) = 1$ is equivalent to assuming that the Kalman gain matrix, K^f , is equal to zero. Therefore if the user desires to change the stochastic part of the identified innovation model, without changing the deterministic model and hence without changing the system matrices C and A, the only way to do it is by changing the Kalman gain matrix, K^f .

Suppose that the new Kalman gain matrix is represented as K^* . K^* is a matrix $(m \times n)$ for a

multiple-output system, and a vector $(1 \times n)$ for a singleoutput system. We can express K^* as

$$K^* = K^f + K'. ag{66}$$

We can then write

$$[C(zI - A)^{-1}K^* + 1]$$

= 1 + CK^*z^{-1} + CAK^*q^{-2} + ... (67)

$$= 1 + C(K^{f} + K')z^{-1} + CA(K^{f} + K')q^{-2} + \cdots$$
 (68)

$$= [C(zI - A)^{-1}K^{f} + 1] + [CK'z^{-1} + CAK'z^{-2} + \cdots].$$
(69)

The stochastic part of the output with the new Kalman gain matrix is

$$(y_t^s)^* = [C(zI - A)^{-1}K^* + 1]e_t$$
(70)

$$= [C(zI - A)^{-1}K^{f} + 1]e_{t} + [CK'z^{-1}]e_{t} + [CK'z^{-1}]e_{t} + CAK'z^{-1}]e_{t}$$
(71)

$$+ CAK z + \dots]e_t \tag{71}$$

$$= y_t^s + [CK'z^{-1} + CAK'z^{-2} + \dots]e_t$$
(72)

$$= y_{t}^{s} + [C \quad CA \quad \dots \quad CA^{n-1}]K' \begin{bmatrix} e_{t-1} \\ e_{t-2} \\ \dots \\ e_{t-n} \end{bmatrix}$$
(73)

$$= y_t^s + (\Gamma_n K')^{\mathrm{T}} e_p \tag{74}$$

where $e_p = [e_{t-1} \ e_{t-2} \ \dots \ e_{t-n}]^T$ and Γ_n is the observability matrix which can be estimated implicitly in the subspace identification method without having to first calculate the system matrices *C* and *A*, through the SVD approach by inspecting the number of dominant singular values in the singular value decomposition of $L_w W_p$.

$$L_{w}W_{p} = (U_{1} \quad U_{2}) \begin{pmatrix} S_{1} & 0\\ 0 & S_{2} \end{pmatrix} \begin{pmatrix} V_{1}^{\mathrm{T}}\\ V_{2}^{\mathrm{T}} \end{pmatrix}$$
$$\approx U_{1}S_{1}V_{1}^{\mathrm{T}} \quad \text{as } j \to \infty, \qquad (75)$$

$$\Gamma_i = U_1 S_1^{1/2}.$$
 (76)

If *n* is the number of dominant singular values taken in S_1 , then Γ_i will be an $(mi \times n)$ matrix, where *i* is the number of block rows taken in W_p . Since the knowledge of the state space system matrices *A* and *C* is not required, the new stochastic model can be incorporated in a model free manner. Now the prediction with the "tuned" noise model can be written as

$$(\hat{y}_t)^* = \hat{y}_t + (\Gamma_n K')^{\mathrm{T}} e_p = \hat{y}_t + \gamma_n e_p,$$
(77)

where $\gamma_n = (\Gamma_n K')^{\mathrm{T}}$, which can be considered as a vector of impulse response coefficients (Markov parameters for the multivariate case) with the new noise model. γ_n is constructed from the estimated observability matrix, Γ_n , and the user specified $(n \times m)$ matrix, K'. Noise model tuning is used as a tool to make up for the process-model mismatch resulting from changes of the process from time to time or simply as tuning parameters. e_p contains the past prediction errors and can be estimated from the data as one step ahead prediction errors. In essence adding the term $[(\Gamma_n K')^{\mathrm{T}} e_p]$ is equivalent to filtering the past prediction errors. Hence K' is used as a tuning parameter and is chosen in such a way that it minimizes the prediction errors.

Thus incorporating a new noise model simply involves the addition of a new term in the calculation of the free response of the process. Hence the free response calculation, Eq. (43), modifies as

$$F = \mathbf{y}_{\mathbf{t}} + L_{w}^{\circ}(1:N_{2}m,:) \begin{bmatrix} \Delta y_{p} \\ \Delta u_{p} \end{bmatrix} + Ye_{p},$$
(78)

where $\Upsilon(N_2m \times n)$ is a left-upper triangular matrix constructed from the elements of γ_n .

$$Y = \begin{bmatrix} \gamma_n(1) & \gamma_n(2) & \dots & \gamma_n(n-1) & \gamma_n(n) \\ \gamma_n(2) & \gamma_n(3) & \dots & \gamma_n(n) & 0 \\ \dots & \dots & \dots & \dots & \dots \\ \gamma_n(N_2) & \dots & \gamma_n(n) & \dots & 0 \end{bmatrix}.$$
 (79)

6. Simulations

The proposed control design method is tested in simulations. The system example is taken from MA-TLAB/MPC toolbox help-manual.

$$\begin{bmatrix} y_{1}(s) \\ y_{2}(s) \end{bmatrix} = \begin{bmatrix} \frac{12.8 \text{ e}^{-s}}{16.7s+1} & \frac{-18.9 \text{ e}^{-3s}}{21.0s+1} \\ \frac{6.6 \text{ e}^{-7s}}{10.9s+1} & \frac{-19.4 \text{ e}^{-3s}}{14.4s+1} \end{bmatrix} \begin{bmatrix} u_{1}(s) \\ u_{2}(s) \end{bmatrix} \\ + \begin{bmatrix} \frac{3.8 \text{ e}^{-8s}}{14.9s+1} \\ \frac{4.9 \text{ e}^{-3s}}{13.2s+1} \end{bmatrix} w(s) + \begin{bmatrix} e_{1}(s) \\ e_{2}(s) \end{bmatrix}.$$
(80)

Open loop input/output data is obtained by exciting the open loop system using a designed 'RBS' signal of magnitude 1 for the inputs, $\mathbf{u}_{\mathbf{k}}$ and random numbers of standard deviation 0.1 for the white noise sequences, $\mathbf{e}_{\mathbf{k}}$, in MATLAB-Simulink. A random walk signal is designed for the measured disturbance w_k by passing a white noise signal of standard deviation 0.1 through an integrator. Two-time units is used as sampling period. Using subspace identification, with i = 50 (row blocks) and j = 2000 (column blocks) in the data Hankel matrices, the subspace matrices $L_w(100 \times 250)$, $L_u(100 \times 100)$ and $L_v(100 \times 50)$ are identified. The simulation data and the models from subspace matrices are plotted in Figs. 2 and 3.



Fig. 2. Inputs, measured disturbance and outputs data from simulations.



Fig. 3. Comparison of process and noise models from subspace matrices with the true models.



Fig. 4. Predictive controller without and with the integrator. w(t) = 0; e(s) = 0.



Fig. 5. Predictive controller without and with the feedforward control. $\mathbf{e}(\mathbf{s}) = 0$.

As can be seen from (3), the impulse response models from the identified subspace matrices match very well with the true impulse response models. Note that even though the signal used for the measured disturbance is not a white noise signal, we can still identify the model corresponding to the measured disturbance very accurately.

In Fig. 4 the simulation results with subspace based predictive controller without an integrator (SPC in Favoreel & De Moor, 1998) is compared with the predictive controller with integral action. As illustrated, the controller with integrator gave no offset for non-zero setpoints. In Fig. 5 the predictive controller performance is compared for the cases without and with feedforward control. As illustrated, better controller performance is achieved with a feedforward control.

For a range of values for λ , N_2 , N_u and constraints on the input moves, $\Delta \mathbf{u}$, a subspace matrices based predictive controller is implemented on the above process in MATLAB-*Simulink*. The closed loop system response for different sets of tuning parameters is illustrated in Figs. 6–9.

In Fig. 6 it can be seen that as the weighting, λ , on the input increases the controller response becomes less aggressive. For a given prediction horizon, as the control horizon, N_u , increases, the controller gives more aggressive tracking performance as shown in Fig. 7. For a given control horizon, as the prediction horizon, N_2 , increases the controller gives better setpoint tracking performance (8). Fig. 9 shows the setpoint tracking under different constraints on the incremental control moves, $\Delta \mathbf{u}$. It can be seen that smaller the magnitude of the maximum allowed control moves, more sluggish is the controller response to setpoint changes.

Noise model tuning: To illustrate the tuning of noise model with the subspace approach consider the process model changes with time, in other words there is a mismatch between the true process model and the identified process model used in the controller design. Consider the case when the process model from Eq. (80) changes to

$$\begin{bmatrix} y_{1}(s) \\ y_{2}(s) \end{bmatrix} = \begin{bmatrix} \frac{12.8 \text{ e}^{-s}}{16.7s+1} & \frac{-18.0 \text{ e}^{-3s}}{21.0s+1} \\ \frac{8 \text{ e}^{-7s}}{10.9s+1} & \frac{-18 \text{ e}^{-3s}}{14.4s+1} \end{bmatrix} \begin{bmatrix} u_{1}(s) \\ u_{2}(s) \end{bmatrix} \\ + \begin{bmatrix} \frac{3.8 \text{ e}^{-8s}}{14.9s+1} \\ \frac{4.9 \text{ e}^{-3s}}{13.2s+1} \end{bmatrix} w(s) + \begin{bmatrix} e_{1}(s) \\ e_{2}(s) \end{bmatrix}.$$
(81)

Fig. 10 illustrates the controller response without and with the on-line noise model tuning feature.



Fig. 6. Variation of input weighting, λ . w(t) = 0; $\mathbf{e}(\mathbf{s}) = 0$.





Fig. 8. Variation of prediction horizon, N_2 . w(t) = 0; e(s) = 0.

Fig. 9. Constrained case predictive controller. w(t) = 0; e(s) = 0.

Fig. 10. Noise model tuning for model mismatch. w(t) = 0; e(s) = 0.

Fig. 11. Experimental setup.

Fig. 12. Step response coefficients from the subspace matrices identified using the open loop data.

7. Experiment on a pilot scale process

The proposed predictive controller is tested on a multivariate pilot scale system. The system considered, shown in Fig. 11, is a three tank system with two inlet water flows. The levels of tank-1 and tank-2 are the two controlled variables (CVs). The setpoints (SPs) for the flow rates through the valves-A and B are the two manipulated variables (MVs). The flow rates through the valves-A and B are controlled through the *local*-PID controllers on each valve. The setpoints for the flow rates come from a higher level advanced controller application. The *local*-PID controllers, which are single variate, are at faster sampling (1 s). The higher level controller, which is multivariate and does computations to minimize an optimization function, sends controller outputs every 6 s. The system is configured so as to emulate a typical multivariate system in the industries.

Fig. 13. Subspace based predictive controller on the pilot scale process.

Tank-3 and valve-C are used primarily to introduce interactions between the variables in the system. As can be seen in Fig. 11 a change in the level in tank-1 effects the level in tank-2 via tank-3 level. The degree of interaction can be manipulated by changing the valve-C position. If the valve-C is completely closed then the level in tank-2 is independent of the level tank-1 (zero interactions). By opening the in valve-C interactions are introduced in the tank-2 level. Valve-C is maintained at a fixed open position through out the exercise. Note that the level in tank-1 is independent of the levels in tank-2 and 3. The step response models for the system, which are formed from the impulse response coefficients in the subspace matrices, are plotted in Fig. 12. The correlations between the variables are clear from the step response plots.

Open loop step-test data for the system is collected by sending two (uncorrelated) designed 'PRBS' signals for the SPs of the flow rates through valves-A and B. Subspace matrices are identified using the open loop data. A multivariate subspace matrices based predictive controller is then designed for the system. The controller parameters (weighting matrices, prediction horizon, control horizon and noise model) are tuned for a smooth controller performance. The closed loop response for the unconstrained and constrained ($|\Delta u| \leq 0.5$) cases are plotted in Fig. 13 and Fig. 14, respectively.

8. Conclusions

In this paper design of the predictive controller, in the GPC framework, using the subspace matrices calculated through the subspace identification method is addressed. Important issues in practical implementation of the predictive controllers such as integral action, constraint handling and feedforward control are discussed. It has been shown that the noise model can be independently specified by the user through the addition of a new term to the predictor equation in the model-free manner, which is shown to be equivalent to changing the Kalman filter gain matrix. The equivalence of the predictive controller designed from subspace matrices to the traditional GPC is shown in Appendix C. The proposed predictive controller is tested on multivariate systems in simulations and on a pilot scale process.

Appendix A. A closer look at the subspace matrices

The matrix input-output equations used in subspace identification, shown below, are derived by the recursive substitution of Eqs. (8)-(9):

$$Y_p = \Gamma_i X_p + H_i U_p + H_i^s E_p, \tag{A.1}$$

$$Y_f = \Gamma_i X_f + H_i U_f + H_i^s E_f, \tag{A.2}$$

Fig. 14. Subspace based predictive controller for the constrained case on the pilot scale process.

$$X_f = A^i X_p + \Delta_i U_p + \Delta_i^s E_p, \tag{A.3}$$

where

$$\Delta_i = [A^{i-1}B \quad A^{i-2}B \quad \dots \quad B]; \Delta_i^s = [A^{i-1}K \quad A^{i-2}K \quad \dots \quad K].$$

Substituting Eq. (A.1) in Eq. (A.3) we obtain

$$X_{f} = \begin{bmatrix} A^{i} \Gamma_{i}^{\dagger} & (\Delta_{i} - A^{i} \Gamma_{i}^{\dagger} H_{i}) & (\Delta_{i}^{s} - A^{i} \Gamma_{i}^{\dagger} H_{i}^{s}) \end{bmatrix} \begin{bmatrix} Y_{p} \\ U_{p} \\ E_{p} \end{bmatrix}.$$
(A.4)

Substituting Eq. (A.4) in Eq. (A.2), as $\{i, j\} \rightarrow \infty$, we can write

$$Y_f = L_w W_p + L_u U_f + L_e E_f, \tag{A.5}$$

where L_w is the subspace matrix corresponding to the states, L_u the subspace matrix corresponding to the deterministic inputs, and L_e the subspace matrix corresponding to the stochastic inputs.

If we take a closer look by expanding subspace matrices, see Fig. 1, we observe that the first row of the subspace matrix L_w is the ARX model of the system. As $i \rightarrow \infty$, the last row of subspace matrices L_u and L_e transform into impulse response models for the process and disturbance respectively.

In the open loop subspace state space identification methods, the sequence of the future states, X_f , and the extended observability matrix, Γ_i , are estimated using Eqs. (14)–(15) and it is required that the pair $\{A, C\}$ be

observable since only the modes that are observable can be identified. Furthermore, the pair $\{A, [B, Q^{1/2}]\}$ requires to be controllable. This implies that all modes are excited by both the deterministic input u_k and the stochastic input e_k . Note that the deterministic and stochastic subsystems may have common or completely decoupled input–output dynamics. If the pair $\{A, C\}$ are observable, then the rank of Γ_i is equal to the state order *n*. Hence *i* should be chosen higher than the number of states in the state space model to be identified. *j* is typically chosen as $100 \times i$. Using the SVD approach the reduced order observability matrix, Γ_n , and the nonsteady state Kalman filter estimate of state sequence, \hat{X}_{f} , are obtained. The state space matrices A, B, C, D, K^{f} and S can then be estimated by using either the observability matrix or the state sequence estimate (Chou & Verhaegen, 1997; Larimore, 1996, 1990; Van Overschee & De Moor, 1994, 1995, 1996; Ruscio, 1997a, d).

Appendix B. QP formulation for constraints handling

QP-formulation of the constraints is well known and available in the literature. Typical process constraints are as follows:

 $u_{\min} \leq u_t \leq u_{\max} \quad \forall t \text{ Amplitude limits,}$

 $\triangle u_{\min} \leq \triangle u_t = u_t - u_{t-1} \leq \triangle u_{\max} \quad \forall t \text{ Slewrate limits,}$

 $y_{\min} \leq y_t \leq y_{\max} \quad \forall t \text{ Quality limits.}$

These constraints for the predictive controller can be expressed as

$$u_{\min} \leq u_{t+k} \leq u_{\max} \quad k = 0, 1, 2, ..., N_u - 1,$$

$$\triangle u_{\min} \leq \triangle u_{t+k} \leq \triangle u_{\max} \quad k = 0, 1, 2, ..., N_u - 1,$$

$$y_{\min} \leq \hat{y}_{t+k|t} \leq y_{\max} \quad k = 1, 2, ..., N_2.$$

Define

$$L_1 = [\bigtriangleup u_{\min} \quad \dots \quad \bigtriangleup \quad \bigtriangleup \quad u_{\min}]^{\mathrm{T}},$$
$$U_1 = [\bigtriangleup u_{\max} \quad \dots \quad \ldots \quad \bigtriangleup \quad u_{\max}]^{\mathrm{T}},$$

$$L_{2} = \begin{bmatrix} u_{\min} - u_{t-1} \\ \dots \\ \dots \\ u_{\min} - u_{t-1} \end{bmatrix}; \quad U_{2} = \begin{bmatrix} u_{\max} - u_{t-1} \\ \dots \\ \dots \\ u_{\max} - u_{t-1} \end{bmatrix},$$

$$L_3 = \begin{bmatrix} y_{\min} & \dots & y_{\min} \end{bmatrix}^{\mathrm{T}} - F,$$

$$U_3 = \begin{bmatrix} y_{\max} & \dots & y_{\max} \end{bmatrix}^{\mathrm{T}} - F, \text{ and}$$

$$\begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix}$$

$$R = \begin{bmatrix} 1 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 1 & 1 & 1 & 1 \end{bmatrix}.$$

The constraints can be rewritten as

 $L_1 \leq \triangle u_f \leq U_1,$

 $L_2 \leq R \bigtriangleup u_f \leq U_2,$

 $L_3 \leq S_{N_2,N_u} \bigtriangleup u_f \leq U_3.$

These constraints can be combined to the form of a single matrix inequality:

$$\Theta \bigtriangleup u \leqslant \Psi \tag{B.1}$$

with

$$egin{array}{lll} artheta &= [-I & -R & -S_{N_2,N_u} & I & R & S_{N_2,N_u}]^{\mathrm{T}}, \ &\Psi &= [-L_1 & -L_2 & -L_3 & U_1 & U_2 & U_3]^{\mathrm{T}}. \end{array}$$

Appendix C. Equivalence of subspace and GPC predictor matrices

The vector of predictor equations used in subspace based predictive controller is

$$\begin{bmatrix} \hat{y}_{t+1} \\ \dots \\ \hat{y}_{t+N_2} \end{bmatrix} = L_w(1:N_2m,:) \begin{bmatrix} y_{t-i+1} \\ \dots \\ y_t \\ u_{t-i} \\ \dots \\ u_{t-1} \end{bmatrix} + L_u(1:N_2m,1:N_ul) \begin{bmatrix} u_t \\ \dots \\ u_{t+N_u-1} \end{bmatrix}.$$

_

Therefore for a k-step ahead prediction

$$\hat{y}_{t+k} = L_w((k-1)m + 1 : km, :) \begin{bmatrix} y_{t-i+1} \\ \cdots \\ y_t \\ u_{t-i} \\ \cdots \\ u_{t-1} \end{bmatrix} + L_u((k-1)m + 1 : km, :) \begin{bmatrix} u_t \\ \cdots \\ u_{t+N_u-1} \end{bmatrix}$$

$$= L_{w}^{\circ}((k-1)m+1:km,:)\begin{bmatrix} y_{t-i+1} \\ \cdots \\ y_{t} \\ u_{t-i} \\ \cdots \\ u_{t-1} \end{bmatrix}$$

+ $S_{N_{2},m}((k-1)m+1:km,:)\begin{bmatrix} \Delta u_{t} \\ \cdots \\ \Delta u_{t+N_{u}-1} \end{bmatrix}$
= $[p_{i-1} \ \cdots \ p_{1} \ p_{0} \ q_{i-1} \ \cdots \ q_{1} \ q_{0}]$
× $\begin{bmatrix} y_{t-i+1} \\ \cdots \\ y_{t} \\ u_{t-i} \\ \cdots \\ u_{t-1} \end{bmatrix}$ + $[s_{k-1} \ \cdots \ s_{1} \ s_{0}]\begin{bmatrix} \Delta u_{t} \\ \cdots \\ \Delta u_{t+m-1} \end{bmatrix}$

$$= [p_0 + p_1 z^{-1} + \dots + p_{i-1} z^{-i+1}] y_t$$

+ $[q_0 + q_1 z^{-1} + \dots + q_{i-1} z^{-i+1}] u_{t-1}$
+ $[s_0 + s_1 z^{-1} + \dots + s_{k-1} z^{-z+1}] \triangle u_{t+k-1}$
= $P(z^{-1}) y_t + Q(z^{-1}) u_{t-1} + S_k \triangle u_{t+k-1}.$ (C.1)

Comparing the above equation with Eq. (5), we observe that

$$P = \frac{F_k}{C}; \quad Q = \frac{\Gamma_k \Delta}{C}; \quad S_k = G_k. \tag{C.2}$$

Therefore, the parametric matrices obtained in GPC design by first identifying an ARIMAX model for the process and then through recursive solution of the Diophantine equations (Bitmead et al., 1990), are directly identified by the subspace identification. This removes the requirement of pre specifying order and structure (ARIMAX) for the process model.

References

- Bitmead, R. R., Gevers, M., & Wertz, V. (1990). Adaptive optimal control: A thinking man's GPC. Englewood Cliffs, NJ: Prentice-Hall.
- Camacho, C. F., & Bordons, C. (1999). *Model predictive control.* London: Springer.
- Chou, C. T., & Verhaegen, M. (1997). Subspace algorithms for the identification of multivariable dynamic errors-in-variables models. *Automatica*, 33(10), 1857–1869.
- Clarke, D. W., Mohtadi, C., & Tuffs, P. S. (1987a). Generalized predictive control—part i. the basic algorithm. *Automatica*, 23(2), 137–148.
- Clarke, D. W., Mohtadi, C., & Tuffs, P. S. (1987b). Generalized predictive control—part ii. extensions and interpretations. *Automatica*, 23(2), 149–160.
- Cutler, C. R., Morshedi, A., & Haydel, J. (1983). An industrial perspective on advanced control. *AIChE annual meeting*, Washington, DC.
- Cutler, C. R., & Ramaker, B. L. (1979). Dynamic matrix control—a computer control algorithm. *AIChE national meeting*, Houstan, TX.
- Cutler, C. R., & Ramaker, B. L. (1980). Dynamic matrix control—a computer control algorithm. *Proceedings of the joint automatic control conference*.
- Favoreel, W., & De Moor, B. (1998). SPC: Subspace predictive control. Technical report 98-49, Katholieke Universiteit, Leuven.
- Favoreel, W., De Moor, B., Gevers, M., & van Overschee, P. (1998). Model-free subspace based LQG-design. Technical report 98-34, Katholieke Universiteit, Leuven.
- Favoreel, W., De Moor, B., Van Overschee, P., & Gevers, M. (1999). Model-free subspace-based LQG-design. *Proceedings of the American control conference*, San Diego, CA (pp. 3372–3376).
- Garcia, C. E., & Morshedi, A. M. (1986). Quadratic programming solution of dynamic matrix control (QDMC). *Chemical Engineer*ing Communications, 46, 73–86.
- Garcia, C., Prett, D., & Morari, M. (1989). Model predictive control: Theory and practice—a survey. *Automatica*, *25*(3), 335–348.

- Larimore, W. E. (1990). Canonical variate analysis in identification, filtering, and adaptive control. *Proceedings of the 29th conference* on decision and control. Honolulu, Hawaii (pp. 596–604).
- Larimore, W. E. (1996). Statistical optimality and canonical variate analysis system identification. *Signal Processing*, 52, 131–144.
- Lee, J. H., & Cooley, B. (1996). Recent advances in model predictive control and other related areas. *Chemical process control-CPC V*, *CACHE*. Tahoe City, CA (pp. 201–216).
- Morari, M. (1994). Chapter model predictive control: Multivariable control technique of choice in the 1990s? Advances in Model-Based Predictive Control, 22–37.
- Prett, D. M., & Morari, R. D. (1980). Optimization and constrained multivariable control of a catalytic cracking unit. *Proceedings of the joint ACC*.
- Qin, S. J., & Badgwell, T. A. (1996). An overview of industrial model predictive control technology. *Proceedings of CPC-V*, Tahoe City, California.
- Richalet, J., Rault, A., Testud, J. L., & Papon, J. (1978). Model predictive heauristic control: Applications to industrial processes. *Automatica*, 14, 413–428.
- Rossiter, J. A., & Kouvaritakis, B. (2001). Modelling and implicit modelling for predictive control. *International Journal of Control*, 74(11), 1085–1095.
- Rossiter, J. A., & Yao, L. (2000). Application of MPC to a non-linear power generation model. *Proceedings of the IEE control 2000*, Cambridge, UK.
- Ruscio, D. D. (1997a). A method for identification of combined deterministic stochastic systems. In: M. Aoki, & A. Hevenner (Eds.), *Applications of computer aided time series modeling* (pp. 181–235). Berlin: Springer.
- Ruscio, D. D. (1997b). Model based predictive control: An extended state space approach. *Proceedings of the 36th conference on decision* and control. San Diego, CA (pp. 3210–3217).
- Ruscio, D. D. (1997c). Model predictive control and identification: A linear state space model approach. *Proceedings of the 36th conference on decision and control*. San Diego, CA (pp. 3202–3209).
- Ruscio, D. D. (1997d). On subspace identification of the extended observability matrix. *Proceedings of the 36th conference on decision* and control. San Diego, CA (pp. 1841–1847).
- Ruscio, D. D., & Foss, B. (1998). On state space model based predictive control. *IFAC dynamics and control of process* systems.
- Stenman, A. (1999). Model on demand: Algorithm, analysis and applications. PhD thesis, Department of EE, Link ping University, SE-581 83 Link ping, Sweden.
- Van Overschee, P., & De Moor, B. (1994). N4SID: Subspace algorithm for the identification of combined deterministic-stochastic systems. *Automatica*, 30(1), 75–93.
- Van Overschee, P., & De Moor, B. (1995). A unifying theorem for three subspace system identification algorithms. *Automatica*, 31(12), 1877–1883.
- Van Overschee, P., & De Moor, B. (1996). Subspace identification for linear systems: Theory implementation applications. Dordrecht: Kluwer Academic Publishers.
- Verhaegen, M. (1994). Identification of the deterministic part of mimo state space models given in innovations form from input–output data. *Automatica*, 30(1), 61–74.