IMC CONTROLLER DESIGN FOR MULTIVARIABLE SYSTEMS

Seung Soo SHIN and Sun Won PARK

Department of Chemical Engineering, KAIST, P.O. Box 131, Cheongryang, Seoul, Korea
(Received 11 October 1989 • accepted 19 January 1990)

Abstract—A systematic internal model control (IMC) controller design methodology has been developed for various types of multivariable processes. When we try to apply IMC to various systems several implementation problems are encountered. In this paper, we resolve these problems and suggest a systematic IMC controller design methodology. IMC shows very good performance and is easy to tune for open-loop stable systems. For unstable systems we apply IMC after stabilizing the systems using the pole placement technique. A combination of quadratic programming and IMC can handle constraints on manipulated and controlled variables.

INTRODUCTION

Two powerful multivariable control techniques were developed independently in the late 1970's. One is now referred to as model algorithmic control (MAC) (Richalet et al., 1978), and the other is called dynamic matrix control (DMC) (Cutler and Ramaker, 1980). They are not the result of a new theory but have a heuristic basis. To date, these techniques have been applied successfully to such diverse systems as a crude column, fluid catalytic crackers, distillation columns, green houses, F-16 jet engine, and power plants.

Another important development is internal model control (IMC). IMC controller was proposed by Garcia and Morari (1982) for single-input-single-output (SISO) systems, and it could include many conventional schemes (Smith predictor, deadbeat controller, Dahlin's method, etc.) as its special cases. Garcia and Morari extended the IMC controller concept defined for SISO systems to multiple-input-multiple-output (MIMO) systems (1985a). Ricker combined the quadratic programming with the IMC controller (QP IMC) to handle input and output constraints (1985).

IMC consists of three parts: (1) internal model to predict the effect of the manipulated variables on the outputs; (2) filter to achieve a desired degree of robustness; (3) control algorithm to compute values of the manipulated variables based on present and past errors and setpoint trajectories. This IMC structure has several advantages: (1) The closed loop stability is guaranteed. (2) Any constraints violations can be anticipated and corrective actions can be taken. (3)

The filter allows simple on-line tuning of multivariable controllers by operating personnel. (4) An IMC controller achieves perfect set-point satisfaction despite any disturbances and model/plant mismatch, and increases robustness for model/plant mismatch and requires less violent actions in the manipulated variables through filter. (5) The structure and parameters of the "optimal" controller are known "a priori", it is the inverse of the invertible part of the system model. This target makes it simple to find suitable approximation for practical implementation. (6) IMC automatically includes an optimal multivariable time delay compensator. (7) IMC allows to obtain coupled and decoupled controllers with equal ease.

A number of papers have been presented on the theoretical aspects of IMC. Even though there are few practical applications reported, IMC is of great theoretical interest to determine how closely the ideal can be approached.

In this paper, we develop a systematic controller design methodology using the IMC concept for multivariable control problems with constraints.

THEORETICAL FUNDAMENTALS

The IMC structure (Figure 1b) is mathematically equivalent to the classical feedback structure (Figure 1a).

The relationship between IMC controller $G_c(z)$ and the conventional feedback controller $C_c(z)$ is

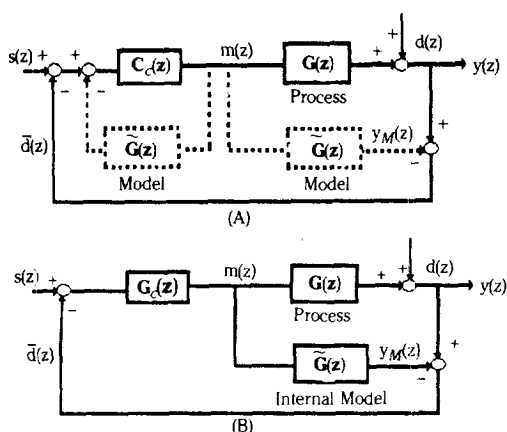$$C_c(z) = [I - G_c(z)\widetilde{G}(z)]^{-1} G_c(z) \qquad (1a)$$

Fig. 1. Classic feedback structure (A) and Basic IMC structure (B).

where $\mathbf{G}$ is the plant and $\widetilde{\mathbf{G}}$ is plant model. Conversely

$$\mathbf{G}_c(z) = [\mathbf{I} + \mathbf{C}_c(z)\,\mathbf{G}(z)]^{-1}\mathbf{C}_c(z). \qquad (1b)$$

From the IMC structure (Figure 1b) the input and output transfer functions are

$$\mathbf{m}(z) = \{\mathbf{I} + \mathbf{G}_c(z)[\mathbf{G}(z) - \widetilde{\mathbf{G}}(z)]\}^{-1}\mathbf{G}_c(z)$$
$$[\mathbf{s}(z) - \mathbf{d}(z)] \qquad (2)$$

$$\mathbf{y}(z) = \mathbf{d}(z) + \mathbf{G}(z)\{\mathbf{I} + \mathbf{G}_c(z)[\mathbf{G}(z) - \widetilde{\mathbf{G}}(z)]\}^{-1}$$
$$\mathbf{G}_c(z)[\mathbf{s}(z) - \mathbf{d}(z)] \qquad (3)$$

which are obtained by standard block diagram manipulations. From transfer functions above we find these properties:

Property 1. Dual Stability Criterion. Assume $\mathbf{G}(z) = \widetilde{\mathbf{G}}(z)$. If the controller $\mathbf{G}_c(z)$ and the process $\mathbf{G}(z)$ are stable, then closed loop stability is guaranteed.

Property 2. Perfect controller. Assume that the controller

$$\mathbf{G}_c(z) = \widetilde{\mathbf{G}}(z)^{-1}$$

yields a closed-loop stable IMC loop. Then, this controller achieves perfect set-point satisfaction despite any disturbances $\mathbf{d}(z)$ and model/plant mismatch $\mathbf{G}(z) \ne \widetilde{\mathbf{G}}(z)$.

When $\mathbf{G}_c(z) = \widetilde{\mathbf{G}}(z)^{-1}$, however, this controller is not realizable. Though "perfect control" cannot be achieved, it is of great theoretical and practical interest to determine how closely this ideal can be approached.

Property 3. Zero offset. Any controller $\mathbf{G}_c(z)$ such that

$$\mathbf{G}_c(z) = \widetilde{\mathbf{G}}(z)^{-1}$$

and which produces a stable IMC loop yields zero offset.

IMC CONTROLLER DESIGN PROCEDURE

1. Open-loop stable systems

1-1. Unconstrained systems

According to Property 2 an IMC controller is designed as $\mathbf{G}_c(z) = \widetilde{\mathbf{G}}(z)^{-1}$. However, this perfect controller cannot be implemented for the following cases:

(i) $\widetilde{\mathbf{G}}$ contains time delays

(ii) Zeros of the transfer matrix [denoted as transmission zeros of $\widetilde{\mathbf{G}}(z)$] outside the complex unit circle (UC)

(iii) Poles of $\mathbf{G}_c(z)$ close to $(-1,0)$(even when stable)'

(iv) Modeling error.

To handle above limitations [(i)-(iv)], Garcia and Morari (1985a) suggested a two step controller design procedure:

STEP 1. Dynamic performance
The model is factored as

$$\widetilde{\mathbf{G}} = \widetilde{\mathbf{G}}_+\widetilde{\mathbf{G}}_- \qquad (4)$$

$$\widetilde{\mathbf{G}}_-(1) = \mathbf{I} \qquad (5)$$

where $\widetilde{\mathbf{G}}_+$ contains time delays and all the zeros of $\widetilde{\mathbf{G}}$ outside the UC, and a controller is

$$\mathbf{G}_c = \widetilde{\mathbf{G}}_-^{-1}. \qquad (6)$$

For MIMO systems, there is generally quite some freedom for choosing $\widetilde{\mathbf{G}}_+$. These options are discussed in detail by Holt and Morari (1985a, b).

STEP 2. Robustness to modeling errors and alleviation of strong control action

In order to detune the controller for increased robustness and less violent actions in the manipulated variables, Garcia and Morari (1985a) introduced a diagonal first order filter

$$\mathbf{G}_c = \widetilde{\mathbf{G}}_-^{-1}\mathbf{F} \qquad (7)$$

where

$$\mathbf{F} = \text{diag}\left[\frac{1 - \alpha_i}{1 - \alpha_i z^{-1}}\right]; \ 0 \le \alpha_i \le 1 \qquad (8)$$

can stabilize the closed-loop system for any model/plant mismatch satisfying

$$\text{Re}\{\lambda_j[\mathbf{G}(1)\widetilde{\mathbf{G}}(1)^{-1}]\} > 0 \quad j = 1, \cdots, n \qquad (9)$$

where $\lambda_j[\mathbf{Q}]$ denotes the jth eigenvalue of $\mathbf{Q}$.

For $\mathbf{G}(z) = \widetilde{\mathbf{G}}(z)$ IMC controller (Eq. 7) yields the closed-loop expression

$$\mathbf{y} = \widetilde{\mathbf{G}}_+(z)\mathbf{F}(z)[\mathbf{s}(z) - \mathbf{d}(z)] + \mathbf{d}(z) \qquad (10)$$

In IMC controller design procedure above, a solution for unstable zeros is to factor transmission zeros outside the unit circle into $\widetilde{\mathbf{G}}_+$ as suggested by Holt and Morari (1985a) and Garcia and Morari (1985a). An

exact determination of the zeros of $\tilde{\mathbf{G}}$ can be difficult due to numerical errors. In this situation a simpler but possibly less optimal solution is to use the input weighting matrices $\beta \neq 0$, and a large optimization horizon p in IMC controller design methodology in Section 2-2. Also, we can distinguish systems into two types which are minimum phase systems and non-minimum systems. A minimum phase system means that the system model $\mathbf{G}(z)$ has transmission zeros inside the unit circle and $\mathbf{G}(z)^{-1}$ is stable. Nonminimum phase systems have transmission zeros outside the unit cirlce, and $\mathbf{G}(z)^{-1}$ is unstable.

When an IMC controller is designed by Eq. (6), the elements of the IMC controller become polynomial rational functions. This IMC controller design procedure generates numerical difficulties. These numerical difficulties can be resolved by deriving the following formulae

$$G_{c_{ij}} = \frac{\theta_{ij,1}z^{m(ij)} + \cdots + \theta_{ij,m(ij)+1}}{\delta_1 z^n + \cdots + \delta_n z + \delta_{n-1}}$$

$$= \sum_{k=1}^{N} \tilde{A}_{ij,k} z^{m(ij)-n+1-k} \tag{11}$$

$$\tilde{A}_{ij,1} = \theta_{ij,1}/\delta_1 \quad \text{when } k = 1$$

$$\tilde{A}_{ij,k} = (\theta_{ij,k} - \sum_{h=1}^{k-1} \delta_{k-h+1}\tilde{A}_{ij,h})/\delta_1 \quad \text{when } k \neq 1$$

N: No. of terms describing the model.

In the case of 2 × 2 systems whose elements are 1st order transfer functions, we can easily derive a general inverse formula of the invertible part of the system's transfer matrix. In the other cases, i.e., $n \times n$ systems $(n \geq 3)$ and 2×2 systems whose elements are high order transfer functions, a general inverse formula of the invertible part of the system's transfer matrix is very complicated. Therefore, in the next section we will explain the IMC controller design methodology which uses the impulse response model without using the inverse matrix of the invertible part of the system's transfer matrix. However, in simple systems such as 2 × 2 systems whose elements are 1st order transfer functions, an IMC controller using the inverse matrix of the invertible part of the system's transfer matrix can deliver the same performance with less effort and time, and easier tuning.

1-2. Constrained systems

Assume $\mathbf{G}(z) = \tilde{\mathbf{G}}(z)$. It has become common practice to use an impulse-response representation for the transfer function matrices $\mathbf{G}$ and $\tilde{\mathbf{G}}$ (assumed to be open-loop stable)

$$y(z) = (\mathbf{H}_0 + \mathbf{H}_1 z^{-1} + \mathbf{H}_2 z^{-2} + \cdots)\mathbf{m}(z) + \mathbf{d}(z) \tag{12}$$

or in the difference equation form

$$\mathbf{y}_{k+p} = \mathbf{H}_0 \mathbf{m}_{k-p} + \mathbf{H}_1 \mathbf{m}_{k+p-1} + \mathbf{H}_2 \mathbf{m}_{k+p-2}$$
$$+ \cdots \qquad \text{[future]}$$
$$+ \mathbf{H}_p \mathbf{m}_k \qquad \text{[present]}$$
$$+ \mathbf{H}_{p+1} \mathbf{m}_{k-1} + \mathbf{H}_{p-2} \mathbf{m}_{k-2} + \cdots \quad \text{[past]}$$
$$+ \mathbf{d}_{k+p} \tag{13}$$

where k is the current sampling interval, k + p is a future sampling interval, and $\mathbf{y}_{k+p}$ denotes a prediction made at interval k. If a model of the impulse response type is used to represent the process, there are several advantages; the order of the process is not at all important, nonminimum phase characteristics can be easily handled, and parametric modeling is not required.

When the process dead time is τ, the effect of m_k will not appear until $\tau + 1$ sampling intervals have elapsed. After $\tau + 1$ sampling intervals have elapsed, in order that IMC includes a multivariable time delay compensator Garcia (1985b) introduced a diagonal factorization of original transfer function matrix.

$$\tilde{\mathbf{G}}(z) = \tilde{\mathbf{G}}_+(z)\tilde{\mathbf{G}}_-(z) \tag{14}$$

where

$$\tilde{\mathbf{G}}_+(z) = \text{diag}(z^{-(\tau_1+1)} \cdots z^{-(\tau_n+1)}) \tag{15}$$

$$\tau_i = \min_j \tau_{ij} \geq 0 \quad i = 1, n \quad j = 1, m \tag{16}$$

and τ_{ij} is the number of sampling intervals of process dead time between m_{ij} and y_i. Then

$$\mathbf{y}_{+,k+p} = \tilde{\mathbf{G}}_+(z)^{-1}\mathbf{y}(z)_{k+p}$$
$$= \mathbf{H}_{*1}\mathbf{m}_{k+p-1} + \mathbf{H}_{*2}\mathbf{m}_{k+p-2} + \mathbf{H}_{*3}\mathbf{m}_{k+p-3}$$
$$+ \cdots \qquad \text{[future]}$$
$$+ \mathbf{H}_{*p}\mathbf{m}_k \qquad \text{[present]}$$
$$+ \mathbf{H}_{*p+1}\mathbf{m}_{k-1} + \mathbf{H}_{*p+2}\mathbf{m}_{k-2} + \cdots \text{[past]}$$
$$+ \mathbf{d}_{+,k+p}. \tag{17}$$

Let $\mathbf{b}_{+,k+p} = \mathbf{H}_{*p+1}\mathbf{m}_{k-1} + \mathbf{H}_{*p+2}\mathbf{m}_{k-2} + \cdots + \mathbf{d}_{+,k+p}$, and collecting the p future values of $\mathbf{y}_+$ into a vector gives

$$\begin{pmatrix} \mathbf{y}_{+,k+1} \\ \vdots \\ \mathbf{y}_{+,k+p} \end{pmatrix} = \begin{pmatrix} \mathbf{H}_{*1} & & \mathbf{0} \\ \mathbf{H}_{*2} & \mathbf{H}_{*1} & \\ \vdots & & \ddots \\ \mathbf{H}_{*p} & \mathbf{H}_{*p-1} & \cdots & \mathbf{H}_{*1} \end{pmatrix} \begin{pmatrix} \mathbf{u}_k \\ \vdots \\ \mathbf{u}_{k+p} \end{pmatrix} + \begin{pmatrix} \mathbf{b}_{+,k+1} \\ \vdots \\ \mathbf{b}_{+,k+p} \end{pmatrix} \tag{18}$$

or

$$\mathbf{y}_{*k} = \mathbf{G}_*\mathbf{m}_{*k} + \mathbf{b}_{*k} \tag{19}$$

where $\mathbf{y}_{*k}$ and $\mathbf{b}_{*k}$ are vectors containing np elements, $\mathbf{m}_{*k}$ is a vector containing mp elements, and $\mathbf{G}_*$ is a $np \times mp$ lower-block-triangular matrix. The best pre-
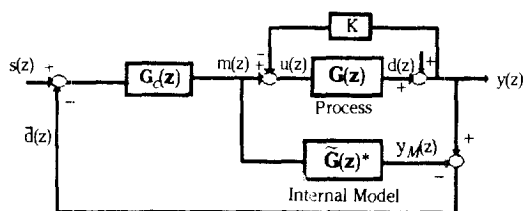
Fig. 2. IMC structure with pole placement.

diction of the disturbance is to set it equal to the disturbance at the present time

$$d_{*,k+p} = d_k \text{ for all } k+p > k. \tag{20}$$

At each time k, the following general problem is solved

$$\min 1/2 [(s_{*k} - y_{*k})^T \Gamma (s_{*k} - y_{*k}) + m_{*k}^T \beta m_{*k}] \tag{21}$$

where Γ and β are weighting matrices.

Solution of Eq. (21) is subject to the following constraints

$$m_{L*} \leq m_{*k} \leq m_{U}* \tag{22}$$

$$y_{L*} \leq y_{*k} \leq y_{U}* \tag{23}$$

where m_L*, m_U*, y_L*, and y_U* are manipulated variables' lower and upper bounds and controlled variables' lower and upper bounds constant vectors respectively.

Let us define a new nonnegative vector

$$u = m_{*k} - m_L*. \tag{24}$$

Equations (22) and (23) can be combined in the general form by a new vector (24)

$$Du \geq r. \tag{25}$$

Finally, substituting Eqs. (19) and (24) into Eq. (21) converts the optimization problem to the form

$$\min_{u} (1/2 u^T R u - a^T u) \tag{26}$$

where

$$a^T = (s_{*k} - b_{*k})^T \Gamma G_* - m_L*^T R \tag{27}$$

$$R = G_*^T \Gamma G_* + \beta \tag{28}$$

and **u** is to be determined subject to Eq. (25).

The solution of Eq. (26) provides a sequence of inputs to be implemented in the future. However, in order to compensate for disturbances, only the present input **m**(k) are implemented and Eq. (26) is "solved" again at subsequent intervals.

2. Open-loop unstable systems

If a process is unstable, Property 1 does not hold and the closed loop system will generally be unstable.

In this case, the unstable process is stabilized by pole placement (Figure 2). The stabilized process transfer functions are

$$G^* = [I + GK]^{-1} G \tag{29}$$

$$d^* = [I + GK]^{-1} d (z) \tag{30}$$

$$u (z) = m (z) - Ky (z) \tag{31}$$

Consider G^* is a controlled process in the IMC structure, then we can design IMC controller with the model $\tilde{G}^*$ of the stabilized process G^*. In Section 2-1 we explain a numerical method for calculating the output feedback gain **K**.

If it were not for constraints, we could use the IMC controller design methodology which is explained in Section 1-1. If we design IMC controller with quadratic programming for the stabilized process model $\tilde{G}^*$, then we can handle the unstable process with constraints. Actually **u**(k) is implemented for the control valve, but IMC controller with quadratic programming can only handle the constraints of **m**(k). Therefore, we take into account the value of feedback signal at steady state when we determine the constraints of **m**(k). However, there is no direct way of consistently recognizing that hard limits on **u**(k) have been reached in a multivariable control scheme of this sort. A constraint action scheme has been imposed on the control algorithm to compute future values of the manipulated variables, which accomplishes the necessary clamping. The control algorithm has the capability that is related to alarming of measured values. Alarms may be invoked for any measured point in the computer database. For example a high limit and a low limit may be declared for a measurement. The points designated for alarm are constantly monitored and if a limit is exceeded a unique alarm condition is declared for that point. An alarm action program was written which accomplishes the appropriate clamping required in the control strategy. Alarm points were declared for variables in the control strategy. The function of the alarm action program is to set a digital bit that is one of the clamp limits of the control algorithm blocks. The program is completely generic.

2-1. Design of an output feedback matrix

Consider a controllable, observable and cyclic multivariable system described by the state and output equations

$$x (k+1) = Ax (k) + Bu (k) \tag{32}$$

$$y (k) = Cx (k) \tag{33}$$

where **x**(k) is the $n \times 1$ state vector, **u**(k) is the $j \times 1$ control input vector with $1 \leq j \leq n$, **y**(k) is the $l \times 1$ output vector with $1 \leq l \leq n$ and the matrices **B** and **C** are of full ranks j and l respectively. The system can be

described by the transfer function relationship

$$\mathbf{Y}(z) = \mathbf{C}(z\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}\mathbf{U}(z) = -\frac{\mathbf{W}(z)}{V(z)}\mathbf{U}(z) \qquad (34)$$

where $\mathbf{W}(z) = \mathbf{C}\,adj(z\mathbf{I}\text{-}\mathbf{A})\mathbf{B}$ is the $l \times j$ numerator polynomial matrix and $V(z) = |z\mathbf{I}\text{-}\mathbf{A}|$ is the nth order open loop characteristic polynomial. The numerator polynomial matrix $\mathbf{W}(z)$ and the denominator polynomial $V(z)$ can be expressed in powers of z as

$$\mathbf{W}(z) = \mathbf{W}_n z^{n-1} + \cdots + \mathbf{W}_2 z + \mathbf{W}_1 \qquad (35)$$

$$V(z) = z^n + e_n z^{n-1} + e_2 z + e_1 \qquad (36)$$

where $\mathbf{W}_i$'s are constant $l \times j$ coefficient matrices and e_i's are constant scalar coefficients.

Let us now apply the constant output feedback control law $\mathbf{u} = \mathbf{m} - \mathbf{K}\mathbf{y}$ where $\mathbf{m}$ is the $j \times 1$ command input vector, and restrict the $j \times l$ constant output feedback matrix $\mathbf{K}$ to have the unity rank structure $\mathbf{K} = \mathbf{q}\mathbf{h}$ where $\mathbf{q}$ and $\mathbf{h}$ are constant $j \times 1$ and $1 \times l$ vectors respectively. The closed-loop system then becomes

$$\mathbf{x}(k+1) = (\mathbf{A} - \mathbf{B}\mathbf{K}\mathbf{C})\mathbf{x}(k) + \mathbf{B}\mathbf{m}(k) \qquad (37)$$

and the characteristic polynomial of the closed-loop system is (see Ref. 2)

$$H(z) = |z\mathbf{I} - \mathbf{A} + \mathbf{B}\mathbf{K}\mathbf{C}| = V(z) + \mathbf{h}\mathbf{W}(z)\mathbf{q}$$
$$= z^n + (\mathbf{h}\mathbf{W}_n\mathbf{q} + e_n)z^{n-1} + \cdots + \mathbf{h}\mathbf{W}_1\mathbf{q} + e_1. \qquad (38)$$

Let us denote the desired closed-loop characteristic polynomial by the following equation

$$H(z) = z^n + g_n z^{n-1} + \cdots + g_2 z + g_1. \qquad (39)$$

Equation expressions (38) and (39) and matching coefficients of like powers of z on both sides give

$$\mathbf{h}\mathbf{W}_1\mathbf{q} + e_1 = g_1$$
$$\mathbf{h}\mathbf{W}_2\mathbf{q} + e_2 = g_2$$
$$\vdots$$
$$\mathbf{h}\mathbf{W}_n\mathbf{q} + e_n = g_n. \qquad (40)$$

The pole placement problem thus reduces to finding the $j + l$ unknown elements of the two vectors $\mathbf{q}$ and $\mathbf{h}$ to satisfy the set of n nonlinear algebraic equations, Equations (40), as closely as possible.

A numerical method is now described for the calculation of $\mathbf{q}$ and $\mathbf{h}$. The two vectors $\mathbf{q}$ and $\mathbf{h}$ are recursively modified in turn to solve Eq. (40) by minimizing the error function $\mathbf{E}(\mathbf{q},\mathbf{h}) = \sum_{i=1}^{n} [\mathbf{h}\mathbf{W}_i\mathbf{q} + e_i - g_i]^2$.

Each equation of (40) is a special kind of nonlinear equation called "bilinear" in that for a given $\mathbf{q}$ the equation is linear in $\mathbf{h}$ and for a given $\mathbf{h}$ it is linear in $\mathbf{q}$. We make use of this bilinearity property and solve Eq. (40) as follows: Treating $\mathbf{q}$ as constant, Eq. (40) can be written as a set of linear equations in $\mathbf{h}$ as

$$\mathbf{L}\mathbf{h}^T = \mathbf{f} \qquad (41)$$

where $\mathbf{L}$ is a constant $n \times l$ matrix whose ith row is $\mathbf{q}^T\mathbf{W}_i^T$ and $\mathbf{f} = [g_1 - e_1, \cdots, g_n - e_n]^T$ is a constant $n \times 1$ vector. Alternatively, treating $\mathbf{h}$ as constant, Eq. (40) can be written as a set of linear equations in $\mathbf{q}$ as

$$\mathbf{M}\mathbf{q} = \mathbf{f} \qquad (42)$$

where $\mathbf{M}$ is a constant $n \times m$ matrix whose ith row is $\mathbf{h}\mathbf{W}_i$. Equations (41) and (42) are now solved in the least-square sense by the following recursive algorithm to minimize the error function

$$\mathbf{E}(\mathbf{q}, \mathbf{h}) = \sum_{i=1}^{n} [\mathbf{h}\mathbf{W}_i\mathbf{q} + e_i - g_i]^2 = \| \mathbf{L}\mathbf{h}^T - \mathbf{f} \|^2$$
$$= \| \mathbf{M}\mathbf{q} - \mathbf{f} \|^2.$$

(i) Set $\mathbf{q} = \mathbf{q}^{(1)}$, the initial value of $\mathbf{q}$, and find the least-squares solution of Eq. (41) as

$$\mathbf{h}^{T(1)} = [\mathbf{L}(\mathbf{q}^{(1)})]^+\mathbf{f}$$

where $+$ denotes pseudoinverse. The least-squares error of Eq. (41) is then given by

$$\mathbf{E}_1 = \mathbf{E}(\mathbf{q}^{(1)}, \mathbf{h}^{(1)}) = \| \mathbf{L}(\mathbf{q}^{(1)})\mathbf{h}^{T(1)} - \mathbf{f} \|^2.$$

(ii) Set $\mathbf{h} = \mathbf{h}^{(1)}$ and obtain the least-square solution of Eq. (42) for $\mathbf{q}$ as

$$\mathbf{q}^{(2)} = [\mathbf{M}(\mathbf{h}^{(1)})]^+\mathbf{f}.$$

The least-squares error of Eq. (42) is then given by

$$\mathbf{E}_2 = \mathbf{E}(\mathbf{q}^{(2)}, \mathbf{h}^{(1)}) = \| \mathbf{M}(\mathbf{h}^{(1)})\mathbf{q}^{(2)} - \mathbf{f} \|^2.$$

(iii) Update $\mathbf{q}$ to its refined value $\mathbf{q}^{(2)}$ and repeat (i) to obtain $\mathbf{h}^{(2)}$, the refined value of $\mathbf{h}$, and evaluate the least-squares error $\mathbf{E}_3$.

(iv) Set $\mathbf{h} = \mathbf{h}^{(2)}$ and repeat (ii) to obtain $\mathbf{q}^{(3)}$ and evaluate the least-squares error $\mathbf{E}_4$.

(v) Set $\mathbf{q} = \mathbf{q}^{(3)}$ and repeat from (i) until convergence.

In Ref. 12, it is shown that the successive errors are monotonically decreasing for all initial value $\mathbf{q}^{(1)}$ until:

(a) The absolute minimum value of $\mathbf{E}$ is reached at i^{th} iteration and $\mathbf{E}_{min} > 0$. Then the error cannot decrease any further and $\mathbf{E}_j = \mathbf{E}_i$ for all $j > i$. In this case the pole placement problem does not have an exact solution and the best approximate solution is given by the last values of $\mathbf{q}$ and $\mathbf{h}$ in the ith iteration.

or (b) A local minimum of error function is reached at the ith iteration and $\mathbf{E}_{i-1}$ is not outside of this local valley. Then again we have $\mathbf{E}_j = \mathbf{E}_i$ for all $j > i$. In this case different initial values $\mathbf{q}^{(1)}$ must be tried in order to avoid the local minimum.
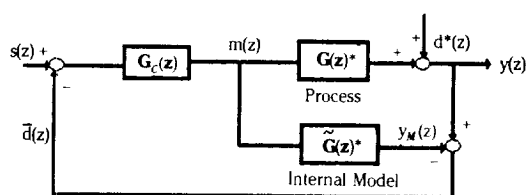
If neither (a) nor (b) occurs, the pole placement

Fig. 3. Equivalent representation of Figure 2 for internal stability analysis.

problem has an exact solution and the error decreases monotonically towards zero and the recursive procedure is continued until the error is acceptable.

APPLICATION EXAMPLES

1. Minimum phase systems

1-1. Open-loop stable systems

1-1-1. Unconstrained systems

Pulse testing of a distillation column yields the following dynamic model between product concentrations y_1, y_2 and product drawoff rates m_1, m_2:

$$\begin{pmatrix} y_1(s) \\ y_2(s) \end{pmatrix} = \begin{pmatrix} \dfrac{1.0e^{-s}}{10s+1} & \dfrac{0.3e^{-3s}}{8s+1} \\ \dfrac{0.2e^{-7s}}{7s+1} & \dfrac{0.5e^{-3s}}{5s+1} \end{pmatrix} \begin{pmatrix} m_1(s) \\ m_2(s) \end{pmatrix}$$

Time is measured in minutes. Assume the process model is exact. When a sampling interval of 1 min is used, the discrete time transfer matrix has the form

$$G(z) = \begin{pmatrix} z^{-2}g_{11}(z)\, z^{-4}g_{12}(z) \\ z^{-8}g_{21}(z)\, z^{-4}g_{22}(z) \end{pmatrix}$$

where the elements g_{ij} are semi-proper functions of z.

We obtain the factorization matrix through the IMC controller design procedure

$$\widetilde{G}_+(z) = \begin{pmatrix} z^{-2} & 0 \\ 0 & z^{-4} \end{pmatrix} \quad \widetilde{G}_-(z) = \begin{pmatrix} g_{11}(z) & z^{-2}\, g_{12}(z) \\ z^{-4}g_{21}(z) & g_{22}(z) \end{pmatrix}$$

An IMC controller is

$$G_c = \widetilde{G}_-^{-1} F$$

where $F = \dfrac{1-\alpha}{1-\alpha z^{-1}} I$.

The response of the perfect controller $G_c = \widetilde{G}_-^{-1}$ to a set-point change in y_1 to 1.0 is shown in Figure 6. This deadbeat response shown in Figure 6 is obtained through relatively strong input action. A tuning parameter α which has a direct effect on the closed-loop response (Eq. 10) can alleviate the strong input action (Figure 4,5).
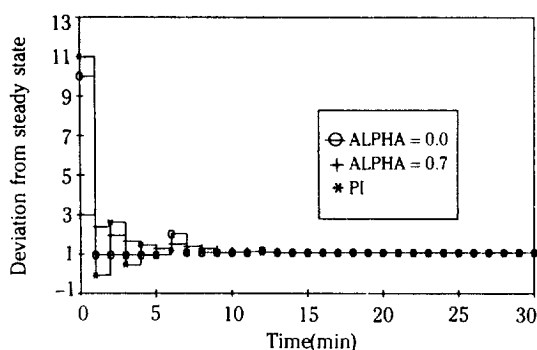

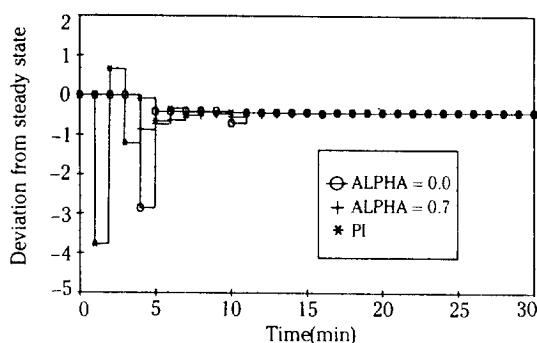Fig. 4. Manipulated variable 1 (m_1) responses to y_1 setpoint change.

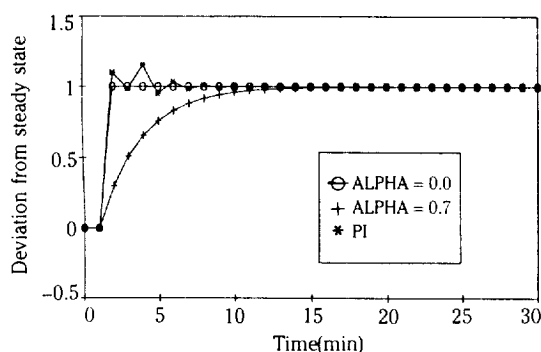Fig. 5. Manipulated variable 2 (m_2) responses to y_1 setpoint change.

Fig. 6. Controlled variables 1 (y_1) responses to y_1 setpoint change.

The performance of IMC is compared with that of PI controller using deadtime compensator. SISO PI loops with controllers

$$g_{c1} = 10 + 1/s$$

$$g_{c2} = 10 - 2/s$$

are used. Figures 6 and 7 show that the performance of IMC is superior to PI controllers with dead time
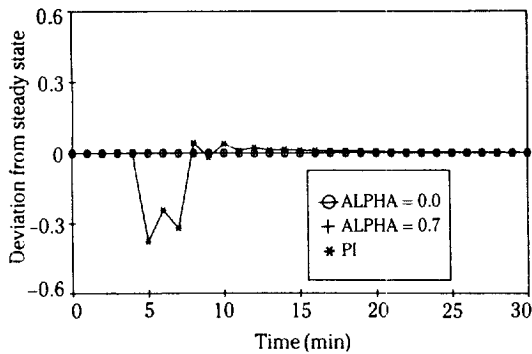
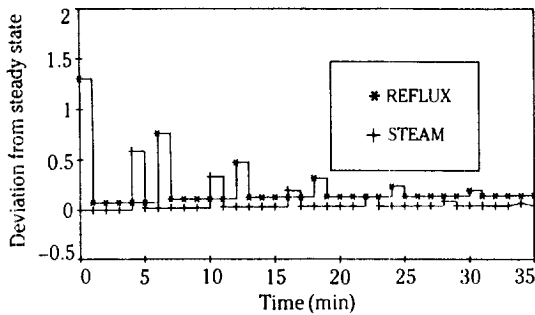Fig. 7. Controlled variables 2 (y_2) responses to y_1 setpoint change.

Fig. 8. Wood/Berry column: Manipulated variables responses to y_1 setpoint change (without constraints).
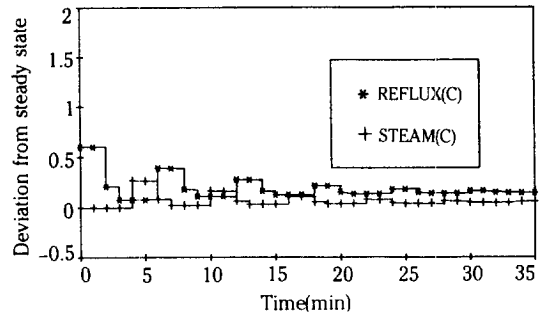

Fig. 9. Wood/Berry column: Manipulated variables responses to y_1 setpoint change (with constraints).

Fig. 10. Wood/Berry column: Controlled variables responses to y_1 setpoint change.

compensator. When IMC are used, smoother approaches to the setpoints are obtained.

1-1-2. Constrained systems

We have studied two systems which are Wood/Berry column model and Gagnepain and Sebog model.

1-1-2-1. Wood/Berry column model

IMC with quadratic programming is applied to Methanol/Water Distillation column. Wood/Berry (1973) have reported transfer function models of an experimental methanol/water column. The column model is

$$\begin{pmatrix} y_1(s) \\ y_2(s) \end{pmatrix} = \begin{pmatrix} \dfrac{12.8e^{-s}}{16.7s+1} & \dfrac{-18.9e^{-3s}}{21.0s+1} \\ \dfrac{6.6e^{-7s}}{10.9s+1} & \dfrac{-19.4e^{-3s}}{14.4s+1} \end{pmatrix} \begin{pmatrix} m_1(s) \\ m_2(s) \end{pmatrix}$$

$$+ \begin{pmatrix} \dfrac{3.8e^{-8s}}{14.9s+1} \\ \dfrac{4.9e^{-3s}}{13.2s+1} \end{pmatrix} d(s).$$

Time is measured in minutes. Assume the process model is exact. We used a sampling interval of 1 min.

The diagonal time delay factorization matrix for this system is

$$\tilde{G}_+(z) = \begin{pmatrix} z^{-2} & 0 \\ 0 & z^{-4} \end{pmatrix}.$$

Weighting matrices of the optimization problem are $\Gamma = I$, $\beta = 0$ and the optimization horizon is 10. Lower and upper bounds of the manipulated variable are assumed to be $m_L = -0.4$ and $m_U = 0.6$ respectively. When the filter time constant α is 0 and manipulated variables have no constraints, the IMC controller behaves like a perfect controller (Figure 8,10). If constraints of the type given in Eq. (25) are imposed, the IMC controller can handle constraints with somewhat loose performance (Figure 9,10).

To investigate the case of model/plant mismatch we obtained a model with a structure

$$\tilde{G} = \begin{pmatrix} \dfrac{11.8e^{-s}}{15.7s+1} & \dfrac{-17.9e^{-3s}}{20.0s+1} \\ \dfrac{6.0e^{-7s}}{10.9s+1} & \dfrac{-18.4e^{-2s}}{13.4s+1} \end{pmatrix}$$

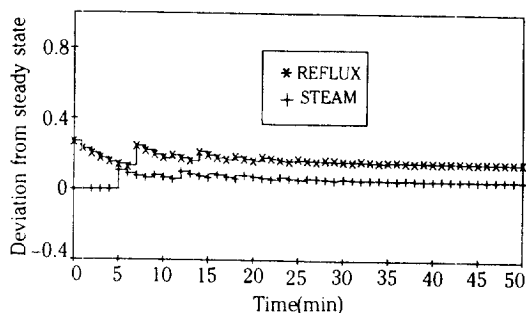For this model, the diagonal time delay factorization matrix is

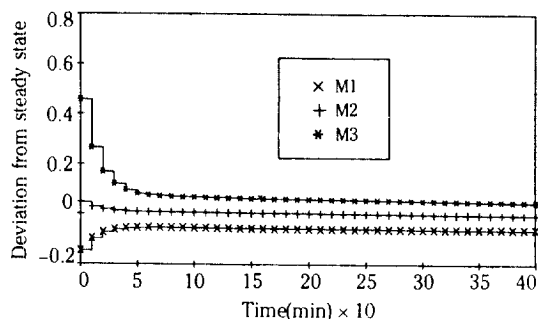**Fig. 11. Wood/Berry column: Manipulated variables responses to y_1 setpoint change (with modeling error).**

Fig. 13. Gagnepain and Seborg model: Manipulated variables responses to y_1 setpoint change.
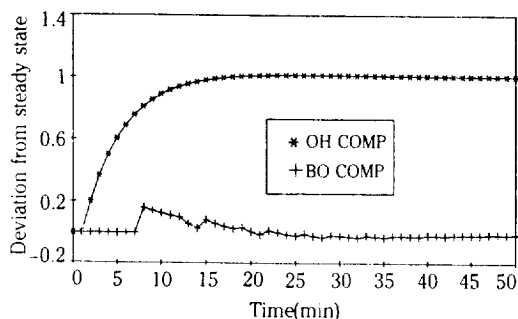

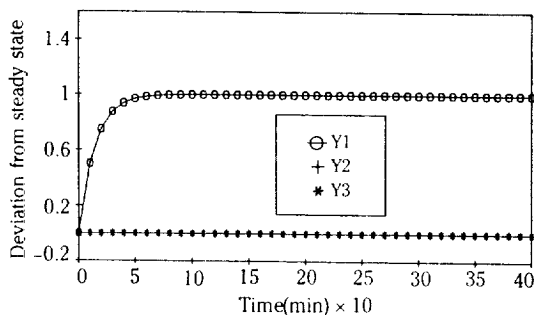**Fig. 12. Wood/Berry column: Controlled variables responses to y_1 setpoint change (with modeling error).**

Fig. 14. Gagnepain and Seborg model: Controlled variables responses to y_1 setpoint change.

$$\tilde{G}_-(z) = \begin{pmatrix} z^{-2} & 0 \\ 0 & z^{-3} \end{pmatrix}.$$

The simulation result of plant/model mismatch case is shown in Figure 11, 12. When $\alpha = 0.8$ the IMC controller achieves good setpoint tracking even for the case with the model/plant mismatch.

1-1-2-2. Gagnepain and Seborg model

The 3×3 system is

$$G = \begin{pmatrix} \dfrac{-2}{10s-1} & \dfrac{1.5}{5s+1} & \dfrac{1}{s+1} \\[2ex] \dfrac{1.5}{5s-1} & \dfrac{-1}{s+1} & \dfrac{2}{10s+1} \\[2ex] \dfrac{1}{s+1} & \dfrac{2}{10s+1} & \dfrac{1.5}{5s+1} \end{pmatrix}.$$

Time is measured in minutes. Assume the model is exact. We used a sampling interval of 0.1 min. Filter time constant α is 0.5. Weighting matrices of the optimization problem are $\Gamma = \mathbf{I}$, $\beta = \mathbf{0}$ and the optimization horizon is 10. When the constraints were not imposed on the manipulated variables, the

responses of the IMC controller with quadratic programming are shown in Figure 13, 14. The results show that IMC controller achives good setpoint tracking. We have easily designed IMC controller using the impulse response model for the 3×3 system.

1-2. Open-loop unstable system

A two-input-two-output system is

$$\mathbf{x}(k+1) = \mathbf{Ax}(k) + \mathbf{Bu}(k)$$

$$\mathbf{y}(k) = \mathbf{Cx}(k)$$

where $\mathbf{A} = \begin{pmatrix} -1.1 & 0.1 \\ 0.2 & -1.3 \end{pmatrix}$ $\mathbf{B} = \begin{pmatrix} 0.1 & 0.2 \\ 0 & 0.1 \end{pmatrix}$

$\mathbf{C} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$

As the poles of this system are -1.0268 and -1.3732, the system is open-loop unstable. This system can be stabilized by pole placement (Figure 2).

$$\mathbf{x}(k+1) = \mathbf{Ax}(k) + \mathbf{Bu}(k) \tag{43}$$

$$\mathbf{y}(k) = \mathbf{Cx}(k) \tag{44}$$

$$\mathbf{u}(k) = -\mathbf{Ky}(k) + \mathbf{m}(k). \tag{45}$$

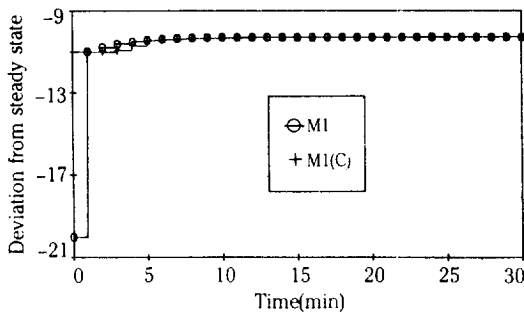Substituting Eq. (45) into Eq. (48) gives

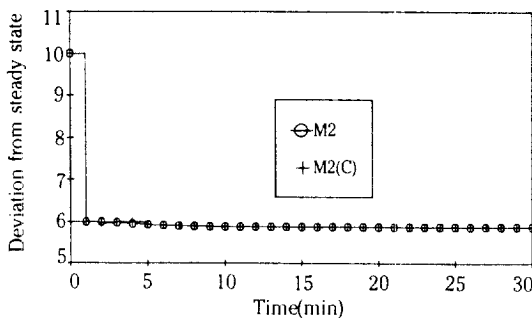**Fig. 15. Unstable process: Manipulated variable 1 (m_1) responses to y_2 setpoint change.**

Fig. 17. Unstable process: Controlled variables responses to y_2 setpoint change.

Fig. 16. Unstable process: Manipulated variable 2 (m_2) responses to y_2 setpoint change.
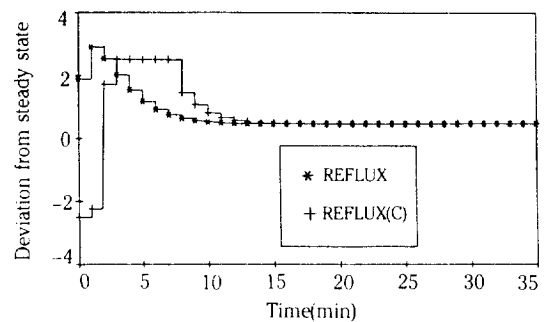
Fig. 18. Karim model: Reflux responses to y_1 setpoint change.

$$\mathbf{x}\,(k+1) = (\mathbf{A} - \mathbf{BKC})\mathbf{x}\,(k) + \mathbf{Bm}\,(k).$$

When $\mathbf{K} = \begin{pmatrix} -17 & 40 \\ 1 & -19 \end{pmatrix}$, the poles of the closed-loop

process are 0.5 and 0.5. It is stable. IMC control can be applied to the stabilized process to improve control performance. Weighting matrices are $\Gamma = \mathbf{I}$ and $\beta = \mathbf{0}$ and the optimization horizon is 5.

We have simulated two cases to a set-point change in y_2 to 1.0. In the first case, the filter time constant α was set to 0 and the constraints were not imposed on the manipulated variables. In the second case, we used $\alpha = 0$ and imposed constraints on manipulated variables. The first manipulated variable's lower bound was $m_{1L} = -11.0$. The second manipulated variable's upper bound was $m_{2U} = 6.0$. No other constraints were considered. The results from the first case show the perfect control for set-point tracking at the expense of big changes in the manipulated variables as shown in Figure 15, 17. The result from the second case shows poorer and slower response to set-point change with smaller changes in the manipulated variables as shown in Figure 16,17. But both cases show stable control.
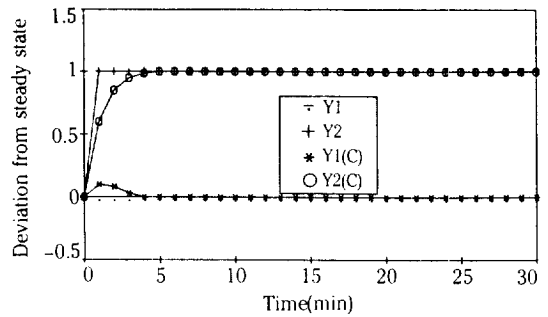
2. Nonminimum phase systems

We have studied Karim model. The open-loop model of a 30-plate, 22.8 cm diameter sieve tray column has been reported previously (1979). The controlled variables are the pressure-corrected temperatures at plates 4 and 24. The manipulated variables are reflux and steam flow rates. The open-loop responses are obtained by giving step changes of -10 percent change is

$$\begin{pmatrix} y_1\,(s) \\ y_2\,(s) \end{pmatrix} = \begin{pmatrix} \dfrac{-0.427e^{-s}}{15s+1} & \dfrac{0.543e^{-s}}{11.5s+1} \\ \dfrac{-0.306e^{-3s}}{21.5s+1} & \dfrac{0.069e^{-s}}{11s+1} \end{pmatrix} \begin{pmatrix} m_1\,(s) \\ m_2\,(s) \end{pmatrix}.$$

The output vector and the input vector are

$$\mathbf{y}\,(s) = \begin{pmatrix} T_{24} \\ T_4 \end{pmatrix} \qquad \mathbf{m}\,(s) = \begin{pmatrix} Re \\ St \end{pmatrix}$$

where T_{24} and T_4 are the pressure-corrected temperatures of plates 24 and 4, respectively, and Re and St are reflux and steam flow rates.

Time is measured in minutes. A sampling interval of 1 min is selected. When the process model is exact, the discrete time transfer matrix has the form
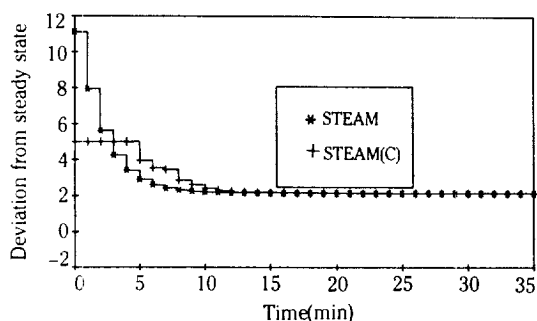
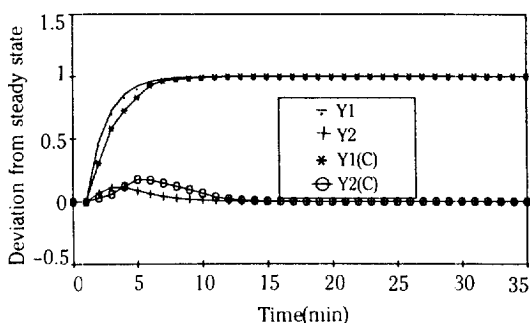**Fig. 19. Karim model: Steam responses to y_1 setpoint change.**

Fig. 20. Karim model: Controlled variables responses to y_1 setpoint change.

$$G(z) = \widetilde{G}(z) = \begin{pmatrix} \dfrac{-2.8647 \times 10^{-2}z^{-1}}{z - 0.93551} & \dfrac{4.72 \times 10^{-2}z^{-1}}{z - 0.91672} \\[3mm] \dfrac{-1.4233 \times 10^{-2}z^{-3}}{z - 0.95455} & \dfrac{6.273 \times 10^{-2}z^{-1}}{z - 0.9131} \end{pmatrix}$$

which is open-loop stable. Then the roots of the numerator of

$$\det \widetilde{G}(z) = (-1.786 + 3.3414z^{-1} + 5.15771 2z^{-2}$$
$$- 12.423\ z^{-3} + 5.740513z^{-4}) \times 10^{-4} /$$
$$(z - 0.93551)(z - 0.91672)(z - 0.95455)$$
$$(z - 0.9131)$$

gives the system zeros

$$z = 0.933453,\ 1.96089,\ 0.908620,\ -1.93225$$

and consequently **G** has transmission zeros outside the unit circle.

If **G**(z) has transmission zeros outside the unit circle, simple exponential filtering alone cannot stabilize the closed-loop system. If we select output weighting matrices as **I** and the optimization horizon as 10, this system can be stabilized with any input weight: in fact $\beta = 10^{-4}$**I** produces a stable controller. The filter time constant α is 0.5. Lower and upper

bounds of the manipulated variables are assumed to be $\mathbf{m}_L = \begin{pmatrix} -2.5 \\ -1.0 \end{pmatrix}$, $\mathbf{m}_U = \begin{pmatrix} 2.5 \\ 5 \end{pmatrix}$ respectively.

When manipulated variables have no constraints, IMC controller with quadratic programming achieves good setpoint tracking (Figure 18, 20). If constraints are imposed on manipulated variables, the IMC controller with quadratic programming can explicitly handle constraints with poorer performance in the controlled variables as shown in Figure 19, 20.

CONCLUSIONS

IMC is implemented for different types of multivariable control problems. The following conclusions have emerged; (1) When an IMC controller is designed by inverting the invertible part of the system's transfer matrix, the elements of the IMC controller become polynomial rational functions. This IMC controller design procedure generates numerical difficulties. We derived general formulae for 2×2 systems to resolve these numerical difficulties. (2) Using quadratic programming we could easily design the IMC controllers for 3×3 systems and nonminimum phase systems without using an inverse matrix of the invertible part of the system's transfer matrix. (3) Quadratic programming (QP) can handle MIMO systems with constraints. (4) IMC control can be used for open-loop unstable as well as stable systems. For open-loop unstable processes, we can apply IMC control after stabilizing the open loop unstable system using the pole placement technique. Simulation results show stable and good control performance.

NOMENCLATURE

$\widetilde{A}_{ij}$: coefficient of controller transfer function
$C_c(z)$: conventional feedback controller
$\mathbf{d}(z)$: disturbance
$\mathbf{F}(z)$: filter
$\mathbf{G}_c(z)$: IMC controller
$\mathbf{G}(z)$: process transfer function
$\widetilde{\mathbf{G}}(z)$: process model transfer function
$\mathbf{H}_i$: impulse response coefficient
$\mathbf{m}(z)$: manipulated variable vector
$\mathbf{m}_{*k}$: collecting vector of p future values of future inputs

$\mathbf{m}_L$. : manipulated variables' lower bounds
$\mathbf{m}_U$. : manipulated variables' upper bounds
$\mathbf{s}(z)$: set-point vector
$\mathbf{y}(z)$: controlled variable vector
$\mathbf{y}._k$: collecting vector of p future values of future outputs

Greek Letters

α : filter time constant
β : input weighting matrix
Γ : output weighting matrix
τ : time delay

REFERENCES

1. Arulalan, G.R. and Deshpande, P.B.: "Simplified Model Predictive Control", *Ind. Eng. Chem. Res.*, **26**, 347 (1987).
2. Fallside, F. and Seraji, H.: "Pole-Shifting Procedure for Multivariable Systems using Output Feedback", *Proc. I.E.E.*, **118**(11), 1648 (1971).
3. Garcia, C.E. and Morari, M.: "Internal Model Control. 1. A Unifying Review and Some New Results", *Ind. Eng. Chem. Process Dev.*, **21**, 308 (1982).
4. Garcia, C.E. and Morari, M.: "IMC. 2. Design Procedure for Multivariable Systems", *Ind. Eng. Chem. Process Dev.*, **24**, 472 (1985a).
5. Garcia, C.E. and Morari, M.: "IMC. 3. Multivariable Control Law Computation and Tuning Guidelines", *Ind. Eng. Chem. Process Dev.*, **24**, 472 (1985b).
6. Holt, B.R. and Morari, M.: "Design of Resilient Processing Plants-V. The Effect of Deadtime on Dynamic Resilience", *Chem. Eng. Sci.*, **40**(1),
1229 (1985a).
7. Holt, B.R. and Morari, M.: "Design of Resilient Processing Plants-VI. The Effect of Right-Half-Plane Zeros on Dynamic Resilience", *Chem. Eng. Sci.*, **40**(1), 59 (1985b).
8. Karim, M.N. and Gordon, K.F. Lee: "On the Design of Robust Control Systems for Distillation Columns", *Chem. Eng. Comm.*, **68**, 81 (1988).
9. Karim, M.N. and Stainthorpe, F.P.: Proceedings of the 1979 JACC, Denver, Colorado, August 17-20 (1979).
10. Morari, M. and Zafiriou, E.: "Robust Process Control", Prentice hall press (1989).
11. Ricker, N.L.: "Use of Quadratic Programming for Constrained Internal Model Control", *Ind. Eng. Chem. Process Dev.*, **24**, 925 (1985).
12. Seraji, H.: "Pole Assignment Techniques for Multivariable Systems using Unity Rank Output Feedback", *Int. J. Control*, **21**(6), 945 (1975).
13. Treiber, S. and Hoffman, D.W.: "Multivariable Constraint Control using a Frequency Domain Approach", Chem. Proc. Control-CPC III Proc., 185 (1986).
14. Wood, R.K. and Berry, W.W.: "Terminal Composition Control of a Binary Distillation Column", *Chem. Eng. Sci.*, **28**, 1707 (1973).
    Study of Internal Model Control", *Ind. Eng. Chem. Process Dev.*, **25**, 102 (1986).
15. W. Harmon Ray: "Advanced Process Control", Mcgraw-Hill press (1981).
16. Yaman Arkun and John Hollet: "Experimental Study of Internal Model Control", *Ind. Eng. Chem. Process Dev.*, **25**, 102 (1986).
17. "Control System Design by Pole-Zero Assignment", Academic Press, 196-210 (1977).