*Math. Struct. in Comp. Science* (2014), vol. 24, iss. 5, e240511, 16 pages. © Cambridge University Press 2014
doi:10.1017/S0960129513000753

# The design of an IMC-PID controller based on MEOTF and its application to non-square processes with time delay†

QIBING JIN‡, LITING CAO‡§, KUN HE‡, KEWEN WANG‡ and BEIYAN JIANG‡

‡*College of Information Science and Technology, Beijing University of Chemical Technology, Beijing, China*
*Email:* jinqb@mail.buct.edu.cn;caoliting0618@163.com;hekun322@gmail.com; heaven996@hotmail.com;yanzi861111@163.com
§*Beijing Union University, Beijing, China*

*Received 20 July 2011; revised August 18 2013*

It is difficult to design a controller directly for non-square multi-variable systems with time delay. In the current paper, we propose a new design method for an Internal Model Control PID controller based on a modified effective open-loop transfer function (MEOTF) for non-square processes with time delay. The MEOTF method is used to decompose the complex non-square process into several equivalent independent single-input/single-output processes. Using the Taylor Particle Swarm Optimisation (Taylor–PSO) model reduction method, the MEOTF of the non-square process is approximated by a reduced order form. The reduced form of the MEOTF is then used to design the Internal Model Control PID controller, which is then used for the original non-square process. To improve the robust stability, a first-order filter is added in the feedback loop. Finally, we present simulation results showing the validity and reliability of this method. In particular, our method has a strong anti-interference characteristic and retains its good control performance in the presence of model perturbation and interference.

## 1. Introduction

Systems with multiple inputs and outputs are common in modern complex industrial processes. These multivariable systems are called non-square control systems when the number of inputs and outputs are different, such as in the three-order split distillation tower model and some chemical distillation towers. A common control strategy for the non-square control system has been to transform the non-square control system into a square control system, and then design the controller. The control aims can be achieved using these methods since there are numerous controller design methods for square systems. However, the disadvantages of using these control methods are higher costs or a reduced quality of control. As a result, researchers have put forward other control strategies for non-square systems.

† This work was supported by 863 Project (2008AA042131) and 973 Project (2007CB714300).

https://doi.org/10.1017/S0960129513000753 Published online by Cambridge University Press

Steven Treiber designed a kind of pre-compensator (Treiber 1984), which transforms the non-square system into a square system with decoupling, and then designed the controller using the Direct Nyquist Array method. Davison's method (Davison 1976) was applied to a non-square system by Sarma and Chidambaram (Dijkstra 1976) to design a distributed controller. The simulation results of the integral-square-error (ISE) showed that this method has better performance and a shorter adjusting time than the method of Tanttu and Lieslehto (Sarma and Chidambaram 2005). However, this method did not solve the control problems for non-square systems containing many time delays and right half plane zeroes. With this kind of system in mind, Chen *et al.* (2008) and Chen *et al.* (2009) proposed an improved internal model control method, which decomposed the transfer function matrix of a non-square system into an equivalent minimum phase part and non-minimum phase part, and then designed the controller to solve the non-square control problem containing many time delays and right half plane zeroes. However, this method often hits robustness requirement problems when the condition number of the system is large because the control performance has a big impact on the perturbation of the controlled object. Jin *et al.* (2010) and Jin *et al.* (2011) made improvements to this method to enhance the control accuracy. Jin and Liu proposed an internal model control method based on the singular value decomposition that meets the robustness of the system requirements through the design of a filter. In Jin's method, the filter in the system affects the singular values directly, thereby overcoming the loss of influence due to model mismatching.

In recent years, control methods for multi-variable systems based on the application of effective open-loop transfer function theory have become increasingly popular: in particular, Vu and Lee (2010), He *et al.* (2005), Zhu (1996), Huang *et al.* (2003) and Xiong and Cai (2006) put forward controller design methods for square control systems, which have produced better results. In the current paper, we develop a new controller design method that can be used for non-square control systems. In order to further improve the robustness and control performance of the controller, the effectively open-loop transfer function was introduced into the non-square control system. This method has many features that enable it to satisfy the control requirements of systems with many time delays, has a strong anti-interference characteristic and can retain its good control performance in the presence of model perturbation and interference.

### 1.1. *Organisation of the paper*

In Section 2, we present the IMC-PID design method based on MEOTF and analyse the decoupling of a non-square system with multiple time delays. In Section 3, we present the simulation results for IMC-PID, which show the effectiveness of the proposed method. Finally, we give a brief summary and conclusions in Section 4.

## 2. Analysis of the controller design method

It is difficult to design the controller directly because there is coupling in multi-variable systems, so the design of each single-loop controller relies on other loop controllers' prior

https://doi.org/10.1017/S0960129513000753 Published online by Cambridge University Press

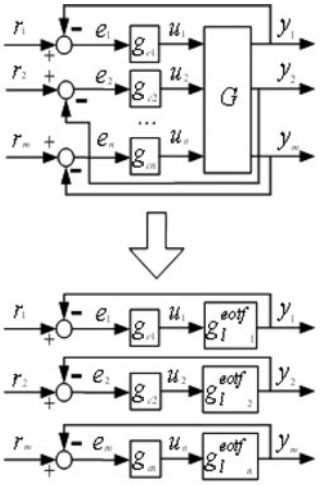



Fig. 1. Conversion of a multi-loop process to several single-loop processes.

knowledge. Figure 1 illustrates a method proposed by Truong Nguyen Luan Vu and Moonyong Lee (Jin *et al.* 2010; Vu and Lee 2010; He *et al.* 2005; Zhu 1996; Huang *et al.* 2003). Using this method, a complex multi-variable system can be decomposed into several equivalent single-variable systems using an effective open-loop transfer function. The single-variable controllers can then be designed using a single-loop EOTF, and then used in the original multivariable system. However, this method is only suitable for a square model control system. To design a controller for a non-square system requires the use of a modified EOTF, as proposed in the current paper.

### 2.1. *Modified effective open-loop transfer function*

The effective open-loop transfer function for an $m \times n$ non-square process is described by

$$G(s) = \begin{bmatrix} g_{11}(s) & g_{12}(s) & \cdots & g_{1n}(s) \\ g_{21}(s) & g_{22}(s) & \cdots & g_{2n}(s) \\ \vdots & \vdots & \ddots & \vdots \\ g_{m1}(s) & g_{m2}(s) & \cdots & g_{mn}(s) \end{bmatrix}_{m \times n} \tag{1}$$

where $g_{ij}(s)$ is the transfer function from the $i$th input to the $j$th output, which can be written as

$$g_{ij}(s) = g_{ij}^{-}(s)e^{-\theta_{ij}s}$$

with $1 \leqslant i \leqslant m$ and $1 \leqslant j \leqslant n$, where $g_{ij}^{-}(s)$ is the stable regular rational part, $e^{-\theta_{ij}s}$ is the time delay part and $\theta_{ij}$ is the value of the delay time.

Figure 2 shows the block diagram of multivariable control system with stable open loop. The concept of an effective open-loop transfer function (EOTF) can be explained in terms of the variables in Figure 2:

— $R^{ij}$ describes the reference value of each loop circuit;

https://doi.org/10.1017/S0960129513000753 Published online by Cambridge University Press

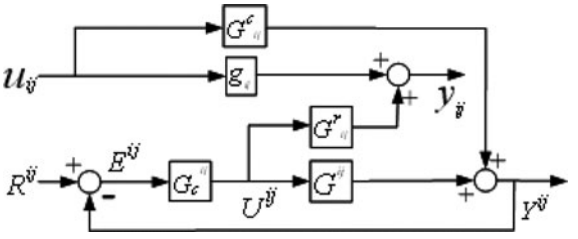


Fig. 2. Block diagram for the concept of the effective open-loop transfer function for an $m \times n$ non-square process.

— $Y^{ij}$ describes the output value of each loop circuit;
— $E^{ij}$ describes the error value of $R^{ij}$ and $Y^{ij}$;
— $U^{ij}$ describes the input value of each loop circuit;
— $u^{ij}$ describes the input value of each open loop circuit;
— $y^{ij}$ describes the output value of each open loop circuit.

The values of $i$ and $j$ must satisfy

$$\begin{aligned} i &= j, \ \ j \leqslant m \\ i &= n, \ \ j > m. \end{aligned} \tag{2}$$

This rule is suitable for both square and non-square model systems.

The transfer function matrix $G^{ij}(s)$ after removing the elements in the $i$th column and the $j$th row of $G(s)$ is given by

$$G^{ij}(s) = \begin{bmatrix} g_{11}(s) & \cdots & G_{1(j-1)} & G_{1(j+1)} & \cdots & g_{1n}(s) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ g_{(i-1)1}(s) & \cdots & g_{(i-1)(j-1)}(s) & g_{(i-1)(j+1)}(s) & \cdots & g_{i+1)n}(s) \\ g_{(i+1)1}(s) & \cdots & g_{(i+1)(j-1)}(s) & g_{(i+1)(j+1)}(s) & \cdots & g_{i+1)n}(s) \\ \vdots & \vdots & \ddots & \vdots & & \\ g_{m1}(s) & \cdots & g_{m(j-1)}(s) & g_{m(j+1)}(s) & \cdots & g_{mn}(s) \end{bmatrix}_{(m-1)\times(n-1)} \tag{3}$$

This describes the transfer functions of objects in each loop.

The transfer function matrix $G_{ij}^{T}(s)$ after removing the elements in the $j$th row in the $i$th column of $G(s)$ is given by

$$G_{ij}^{T} = [g_{ij}(s) \cdots g_{ij-1}(s) \ [g_{ij+1}(s) \cdots \ [g_{in}(s)]_{i\times(n-1)}. \tag{4}$$

This describes the effects of the $U^{ij}$ on the $Y^{ij}$ in each loop.

The transfer function matrix $G_{ij}^{C}(s)$ after removing the elements in the $j^{th}$ column in the $j^{th}$ line of $G(s)$ is given by

$$G_{ij}^{C} = [g_{1j}(s) \cdots g_{i-1j}(s) g_{i+1j}(s) \cdots g_{mj}(s)]_{(m-1)\times 1} \tag{5}$$

This describes the effects of the $u_{ij}$ on the $Y^{ij}$ in the other loop.

https://doi.org/10.1017/S0960129513000753 Published online by Cambridge University Press

When $m = n$, the system is square and the controller matrix $G_C(s)$ is a diagonal matrix:

$$G_C(s) = \begin{bmatrix} g_{C1}(s) & 0 & \cdots & 0 \\ 0 & g_{C2}(s) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & g_{Cn}(s) \end{bmatrix}_{n\times n} \tag{6}$$

When $m \neq n$, the system is non-square and the controller matrix is not a diagonal matrix:

$$G_C(s) = \begin{bmatrix} g_{C1}(s) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & g_{Cn}(s) \\ 0 & \cdots & g_{C(n+1)}(s) \\ \vdots & \ddots & \vdots \\ 0 & \cdots & g_{Cm}(s) \end{bmatrix}_{n\times m} \tag{7}$$

Equation (8) shows the controller transfer function matrix $G_C^{ij}(s)$ after removing the controller:

$$G_C^{ij}(s) = \begin{bmatrix} g_{C1}(s) & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & g_{C(j-1)}(s) & 0 & \cdots & 0 \\ 0 & \cdots & 0 & g_{C(j+1)}(s) & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & g_{Cn}(s) \\ 0 & \cdots & 0 & 0 & \cdots & g_{C(n+1)}(s) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & g_{Cm}(s) \end{bmatrix}_{(n-1)\times(m-1)} \tag{8}$$

Using the conditions in Equation (2), we have

$$\begin{aligned} i &= j, \quad j \leqslant m \\ i &= n, \quad j > m \end{aligned} \tag{9}$$

The effective open-loop transfer function is defined as the ratio of $y_{ij}$ to $u_{ij}$:

$$g_l^{eotf} = y_{ij}/u_{ij}. \tag{10}$$

We now have Equation (11) for open loops and Equations (12)–(14) for closed loops in Figure 2 when $R^{ij} = 0$:

$$g_{ij}u_{ij} + G_{ij}^T U^{ij} = y_{ij} \tag{11}$$

$$E^{ij} = -y^{ij} \tag{12}$$

$$U^{ij} = G_c^{ij} E^{ij} = -G_c^{ij} Y^{ij} \tag{13}$$

$$G^{ij}U^{ij} + G_{ij}^C U_{ij} = Y^{ij}. \tag{14}$$

https://doi.org/10.1017/S0960129513000753 Published online by Cambridge University Press

Substituting $Y^{ij}$ from Equation (14) into Equation (13), we get

$$U^{ij} = -G_C^{ij}(G^{ij}U^{ij} + G_C^{ij}u_{ij}), \tag{15}$$

which can be rearranged to give

$$U^{ij} = -G_C^{ij}(1 + G^{ij}G_C^{ij})^{-1}G_{ij}^C u_{ij}. \tag{16}$$

Combining Equations (16) and (11) gives

$$g_{ij}u_{ij} - G_{ij}^r\left(1 + G^{ij}G_C^{ij}\right)^{-1}G_{ij}^C u_{ij} = y_{ij}, \tag{17}$$

which can be rearranged to give

$$y_{ij} = \left|g_{ij} - G_{ij}^r G_C^{ij}\left(1 + G^{ij}G_C^{ij}\right)^{-1}G_{ij}^C\right| u_{ij}. \tag{18}$$

The relationship between $u_{ij}$ and $y_{ij}$ is very complex in Equation (18). The design of the controller is influenced not only by $g_{ij}$, but also by the processes and the controller of the other loops, and it makes each loop independent. The relationship between $u_{ij}$ and $y_{ij}$ can be simplified by adding some conditions, and the independent controller can then be realised. The required conditions are that the controller contains an integral function and the closed-loop dynamic response is quicker than the open loop dynamic response. When the frequency is less than the transition frequency, these conditions can be described by

$$G^{ij}G_C^{ij}\left(1 + G^{ij}G_C^{ij}\right)^{-1} = 1. \tag{19}$$

Inserting $(G^{ij^{-1}}G^{ij})$ into Equation (18) gives

$$y_{ij} = \left|g_{ij} - G_{ij}^r(G^{ij^{-1}}G^{ij})G_C^{ij}(1 + G^{ij}G_C^{ij})^{-1}G_{ij}^C\right| u_{ij}. \tag{20}$$

Putting Equation (19) into Equation (20) then gives

$$y_{ij} = (g_{ij} - G_{ij}^r G^{ij^{-1}}G_C^{ij})u_{ij}. \tag{21}$$

The relationship between $u_{ij}$ and $v_{ij}$ is simple in this condition, and the controller design only needs to know the transfer matrices of the control object. For non-square systems, the inverse of $G^{ij}$ cannot be calculated directly, so we need to use the generalised inverse (Xiong and Cai 2006), which is defined by

$$G^{ij^+} = G^{ij^H}\left(G^{ij}G^{ij^H}\right)^{-1}, \tag{22}$$

where the superscript $H$ represents the conjugate transpose of the matrix. Using this, we can now rewrite Equation (21) to give

$$y_{ij} = \left[g_{ij} - G_{ij}^r G^{ij^H}\left(G^{ij}G^{ij^H}\right)^{-1}G_{ij}^C\right] u_{ij}. \tag{23}$$

Hence, combining this with Equation (10), the EOTF for a non-square system is given by

$$g_l^{eotf} = g_{ij} - G_{ij}^T G^{ij^H}(G^{ij}G^{ij^H})^{-1}G_{ij}^C. \tag{24}$$

https://doi.org/10.1017/S0960129513000753 Published online by Cambridge University Press

### 2.2. *Time delay processing*

There is a time delay part $e^{g_{ijs}}$ in $g_{ij}(s)$, which we will need to approximate. Common delay approximation methods include: the first-order Pade approximation; the second-order symmetric Pade approximation; the second-order asymmetric Pade approximation; and the poles approximate (Liu *et al.* 2003). We chose the first-order Pade approximation for the current paper because it has a higher error precision in both open and closed loops, and the control effect is better when the time delay is longer. The approximation method gives

$$e^{g_{ijs}} = \frac{1 - \theta_{ij}/2s}{1 + \theta_{ij}/2s}. \tag{25}$$

### 2.3. *Order reduction for MEOTF*

It is not hard to see from the MEOTF formula in Equation (24) that the order of $g_l^{eotf}$ will be very high when the number of inputs and outputs is large. The delay approximation introduced in the previous section makes it even higher. A low-order model $g_l^{r-eotf}$ is usually adopted to approximate $g_l^{eotf}$, and the controller then designed using $g_l^{r-eotf}$ (Roy and Iqbal 2005). We used the Taylor Particle Swarm Order (Taylor–PSO) reduction method for the current paper in order to make the reduced order model more accurate. A reduced order model was produced first using Taylor reduction (Jin *et al.* 2011), but the reduced process has certain errors because the Taylor reduction leaves the high order items. The PSO global search method (Xue 1994; Kennedy and Eberhart 1995) was introduced to optimise the model parameters. The initial value is near to the optimal value, which not only prevents the PSO search finding a solution that is only locally optimal, but also improves its convergence rate. The FOPDT (first order plus dead time) model, shown in Equation (26), was adopted as an approximation model for the current paper because it is widely used in the industrial field:

$$g_l^{r-eotf} = \frac{K_l e^{-\theta s}}{T_l s + 1}. \tag{26}$$

We carry out a Taylor expansion at $s = 0$ for $g_l^{eotf}$ in Equation (24), to give

$$g_l^{eotf}(S) = a_{0l} + a_{1l}s + a_{2l}s^2 + a_{3l}s^3 +)O(s^4) \tag{27}$$

where

$$a_{il} = \frac{1}{i!}\frac{d^i g_l^{eotf}(s)}{ds^i} \mid_s = 0 \tag{28}$$

We now carry out a Taylor expansion at $s = 0$ of the approximation in Equation (26) to give

$$g_l^{eotf}(s) = K_l + K_l(\theta_l + T_l)s + K_l\left[\frac{1}{2}\theta_l^2 + (\theta_l + T_l)T_l\right]s^2 + O(s^2). \tag{29}$$

https://doi.org/10.1017/S0960129513000753 Published online by Cambridge University Press

Comparing the coefficients in Equations (27) and (29), we get

$$\begin{aligned} a_{0l} &= K_l \\ a_{1l} &= K_l(\theta_l + T_l) \\ a_{2l} &= K_l \left[\frac{1}{2}\theta_l^2 + (\theta_l + T_l)T_l\right]. \end{aligned} \tag{30}$$

So the model parameters are

$$\begin{aligned} K_l &= a_{0l} \\ T_l &= \sqrt{\frac{2a_{2l}}{a_{0l}} - \left(\frac{a_{1l}}{a_{0l}}\right)^2} \\ \theta_l &= -\frac{a_{1l}}{a_{0l}} - \sqrt{\frac{2a_{2l}}{a_{0l}} - \left(\frac{a_{1l}}{a_{0l}}\right)^2}. \end{aligned} \tag{31}$$

The solutions can be established when

$$\frac{2a_{2l}}{a_{0l}} \geqslant \left(\frac{a_{1l}}{a_{0l}}\right)^2.$$

If this condition does not hold, we need a high-order model to approximate the process, such as a second-order and pure time-delay process (SOPDT), but the procedure would be the same as for the reduced order FOPDT method.

In the current paper, we used the Particle Swarm Optimisation (PSO) algorithm to optimise the model parameters. The basic PSO algorithm uses a group of particles. In a D-dimensional search space, each particle, viewed as a point in space, is represented in the current iteration by two vectors:

$$x_i = (x_{i1}, x_{i2}, \cdots, x_{iD}),$$

representing its position, where $i = 1, 2, \cdots, m$; and

$$v_i = (v_{i1}, v_{i2}, \cdots, v_{iD})$$

representing its velocity. The initial positions and directions of the particles are randomly defined. In order to find the optimum of the fitness function, the velocity $v_i$ and position $x_i$ of each particle are adjusted in each iteration. During the search process, each particle remembers its previous best position, which is called its personal best (*pbest*)

$$P_i = (P_{i1}, P_{i2}, \cdots, P_{iD}),$$

while the best position found by any particle in the group is known as the global best (*gbest*)

$$P_g = (P_{g1}, P_{g2}, \cdots, P_{gD}).$$

The PSO algorithm updates the velocity and position of the $i$th particle in the $k$th iteration according to Equations (32) and (33), respectively:

$$v_i^{k+1} = w \cdot v_i^k + c_1 \cdot rand_1 \cdot (P_i - x_i^k) + c_2 \cdot rand_2 \cdot (P_g - x_i^k) \tag{32}$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \tag{33}$$

https://doi.org/10.1017/S0960129513000753 Published online by Cambridge University Press

where:

— $c_1$ and $c_2$ are learning factors;
— $rand_1$ and $rand_2$ are random numbers; and
— $w$ is an inertia weight (if $w = 1$, the algorithm is the basic PSO algorithm).

The main steps in the basic PSO procedure are as follows:

**Step I:**
Initialise the particle swarm values: the position $x_i$ and velocity $v_i$ of each particle; the inertia weight factor $w$; and the learning factors $c_1$ and $c_2$.

**Step II:**
Calculate the fitness function value $F_{x_{ij}}$ for each particle according to $x_{ij}$.

**Step III:**
Compare the fitness values, and if there is an improvement, update the particle's *pbest* value to the current location $\hat{x}_i = x_i$. Also make the particle position with the best fitness value the *gbest*,

$$\hat{g} = \min_{X_i} f(x_i), \qquad i = 1, \cdots, n.$$

**Step IV:**
Calculate the fitness function value $F$ according to the updated particle's position. If the condition is satisfied, stop the search, otherwise go to Step **V** to continue the iteration.

**Step V:**
Update the velocities and positions using Equations (32) and (33).

The parameters in Equation (31) can be used as the initial position of the particles in Step **I**, and there are many advantages to doing this, including:

— the particles are close to the best value at the start;
— we avoid the PSO global search method finding a solution that is only locally optimal;
— a greatly reduced number of iterations.

At this point, we have now determined a $g_l^{r-eotf}$ based on MEOTF.

### 2.4. *IMC-PID controller design for a non-square system*

Having determined the reduced EOTF, the multivariable controller design is decomposed into several single variable system controller designs. We used the IMC-PID controller design method in the current paper because it combines the characteristics of IMC and PID, has an ideal control effect and a background in a wide range of industrial applications (Shi and Eberhart 1998). Figure 3 shows the structure of an internal model control, in which:

— $q$ represents the internal model controller;
— $g_m$ is the internal model of the controlled process;
— $g$ is the model of the actual controlled process;
— $f_v$ is a first-order filter;
— $r$ is a reference value; and
— $d$ is noise.

https://doi.org/10.1017/S0960129513000753 Published online by Cambridge University Press

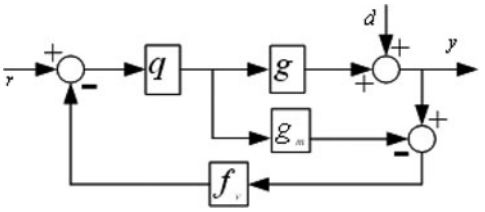



Fig. 3. Block diagram of the Internal Model Control.

We can decompose $g_l^{r-eotf}$ into a minimum phase part and a non-minimum phase part:

$$g_l^{r-eotf} = g_l^- g_l^+ \tag{34}$$

where $g_l^-$ is the regular rational minimum phase part and $g_l^+$ is the non-minimum phase part, which includes the time delay and the right half plane pole part.

The filter was set according to

$$f_l(s) = \frac{1}{(\lambda_l s + 1)^{m_l}} \tag{35}$$

where $\lambda_l$ is an appropriate positive integer acting as an adjustable parameter for the controller to change the control performance of the system and its robust stability.

The IMC controller is given by

$$q_l(s) = g_l^-(s)^{-1} f_l(s) \tag{36}$$

and the equivalent feedback controller by

$$\begin{aligned} g_{Cl}(s) &= \frac{q_l(s)}{1 - q_l(s) g_{ml}(s)} \\ &= \frac{f_l(s)}{g_l^-(s) - f_l(s) g_l^{T-eotf}}. \end{aligned} \tag{37}$$

Using the decomposition

$$\begin{aligned} g_l^- &= \frac{K_l}{T_l s + 1} \\ g_l^+ &= e^{-\theta_i s} \end{aligned}$$

for $g_l^{T-eotf}$ and substituting into Equation (37) together with Equation (25), the feedback controller for $m_l = 1$ is given by

$$g_{Cl}(s) = \frac{\frac{T_l \theta_l}{2} s^2 + (T_L + \frac{\theta_l}{2}) s + 1}{\frac{K_l \theta_l \lambda_l}{2} s^2 + K_l(\theta_l + \lambda_l) s}. \tag{38}$$

The incomplete differential PID form of $g_{Cl}(s)$ is

$$g_{C-PID}(s) = K_{Cl} + \frac{1}{T_{il} s} + \frac{T_{dl} s}{T_{fl} s}. \tag{39}$$

https://doi.org/10.1017/S0960129513000753 Published online by Cambridge University Press

The PID controller parameters for the non-square system are then found as the solution of equations (38) and (39) to be

$$\begin{aligned} K_{Cl} &= \frac{\theta_l^2 + 2T_l\theta_l + 2T_l\lambda_l}{2K_l(\theta_l + \lambda_l)^2}, \\ T_{il} &= K_l(\theta_l + \lambda_l), \\ T_{dl} &= \frac{\theta_l^2 + 2T_l\theta_l + 2T_l\lambda_l - \theta_l\lambda_l}{4K_l(\theta_l + \lambda_l)^2} \\ T_{fl} &= \frac{\theta_l\lambda_l}{2(\theta_l + \lambda_l)}. \end{aligned} \tag{40}$$

### 2.5. *The feedback filter*

As shown in Figure 2, all kinds of uncertain quantitative effects, such as external disturbance, often emerge in control systems. A feedback filter $f_v$ was used to make the controller more able to maintain the original control performance and robustness. In this instance, we used

$$f_v = \frac{1}{T_{fv}s + 1},$$

where $T_{fv}$ is the filter time (Rivera *et al.* 1986; Normey *et al.* 1997). In IMC, the robustness and control performance of systems are usually opposed, and there is always a compromise achieved by adjusting the filter parameter.

## 3. Simulation

The Integral Square Error (ISE) was adopted as an evaluation index for the control performance. This is defined by

$$ISE = \int_0^\infty e(t)^2 dt. \tag{41}$$

### 3.1. *Simulation examples*

The $2 \times 3$ non-square shell standard control problem (Yao *et al.* 2008) was adopted as an example for the current paper. The model for this system is the transfer matrix

$$G(s) = \begin{bmatrix} \frac{4.05e^{-81s}}{50s+1} & \frac{1.77e^{-84s}}{60s+1} & \frac{5.88e^{-81s}}{50s+1} \\ \frac{5.39e^{-54s}}{50s+1} & \frac{5.72e^{-42s}}{60s+1} & \frac{6.9e^{-45s}}{40s+1} \end{bmatrix} \tag{42}$$

https://doi.org/10.1017/S0960129513000753 Published online by Cambridge University Press

Using the method descibed in Section 2, we get the following results:

$$g_l^{eotf}(s) = \frac{0.6843(345.875s+1)}{(321.53s+1)(52.249s+1)}e^{-7.7077s} \tag{43}$$

$$g_2^{eotf}(s) = \frac{3.5533}{(62.449s+1)}e^{-37.938s} \tag{44}$$

$$g_3^{eotf}(s) = \frac{-2.7179}{(65.969s+1)}e^{-69.447s}. \tag{45}$$

The performances are for $\lambda_1 = 952.2$, $\lambda_2 = 30.9$ and $\lambda_3 = 262.7$ when ISE is minimum and the controller is

$$G(s) = \begin{bmatrix} \frac{(321.53s+1)(52.259s+1)(3.854s+1)}{656.8468s(345.875s+1)(3.8229s+1)} & 0 \\ 0 & \frac{(62.449s+1)(18.969s+1)}{244.6021s(8.5148s+1)} \\ 0 & \frac{(65.969s+1)(39.724s+1)}{-902.7423(27.4633s+1)} \end{bmatrix} \tag{46}$$

The PID form of the controller is:

$$G(s) = \begin{bmatrix} 0.04836 + \frac{1}{656.8648s} + \frac{11.0433s}{336.9832s+1} & 0 \\ 0 & 0.2980 + \frac{1}{244.6021s} + \frac{2.3051s}{8.5148s+1} \\ 0 & -0.0811 + \frac{1}{-902.7423s} + \frac{11.0433s}{27.436s+1} \end{bmatrix}_{3\times 2} \tag{47}$$

The filters are

$$f_{v1} = \frac{1}{3.95s+1}$$

$$f_{v2} = \frac{1}{0.93s+1}.$$

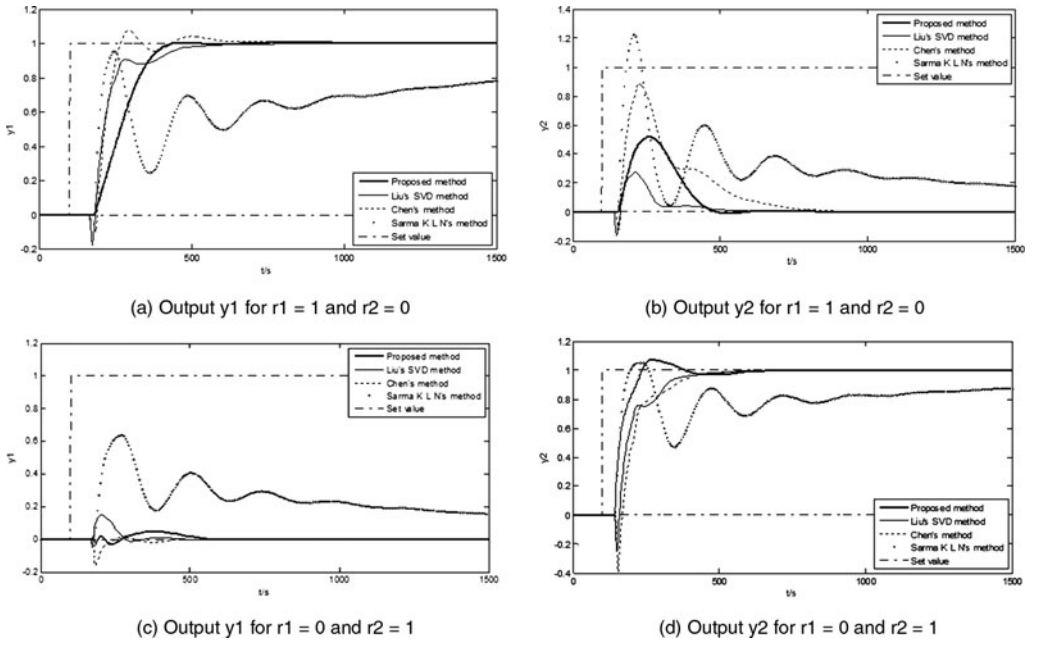
Simulations were made in Simulink. The responses for various given values are shown in Figure 4. The style of the lines in the figure show the responses for various methods: a thick solid line for the method used in the current paper; a broken line for Chen's method (Tanttu and Lieslehto 1991; Chen *et al.* 2009); a solid line for Liu's method (Jin *et al.* 2010); and a diamond point line for Sarma's method (Davison 1976).

The responses when there are interferences are shown in Figure 5, with a 0.2 step interference at $t = 500$ and a $-0.2$ step interference at $t = 600$.

As shown in Figures 4 and 5, the method proposed in the current paper provides decoupling, has several outputs and gives fast and stable tracking of the given value. The response curves obtained using this method are smoother, adapt more quickly and overshoot less compared with the methods of Liu, Chen and Sarma. The comprehensive control performance of the system is best for the method in the current paper. The local control also has good performance and there is a strong response to disturbance.

Table 1 shows the ISE performance index for each method. Comparing these results, the ISE value for the method in the current paper is much less than the values for the

https://doi.org/10.1017/S0960129513000753 Published online by Cambridge University Press




(a) Output y1 for r1 = 1 and r2 = 0

(b) Output y2 for r1 = 1 and r2 = 0

(c) Output y1 for r1 = 0 and r2 = 1

(d) Output y2 for r1 = 0 and r2 = 1

Fig. 4. Responses for various given values.

Table 1. *ISE performance index*

| | Disturbed | EOTF | Liu | Chen | Sarma |
|---|---|---|---|---|---|
| **ISE** | No | 210.2907 | 201.9491 | 301.5051 | 696.5417 |
| | Yes | 234.0799 | 224.0754 | 337.9575 | 749.4713 |

Chen and Sarma methods, so the control effect was better than for those two methods and is close to the value for Liu's method. Analysing the response curves, we see a little overshoot only in control input 1's influence on output 2, but in the other three cases, the method has faster response speed and a smoother excessive process than for Liu's method. Summarising, the method adopted in the current paper is superior to other methods in control performance.

In order to verify the robustness of the method, the static gain and time delay of each element in the transfer function matrix was increased by 20% and the time constant reduced by 20%. The transfer function matrix of the mismatch object is then

$$G(s) = \begin{bmatrix} \dfrac{4.86e^{-97.2s}}{40s+1} & \dfrac{2.124e^{-100.8s}}{48s+1} & \dfrac{7.056e^{-97.2s}}{40s+1} \\ \dfrac{6.468e^{-64.8s}}{40s+1} & \dfrac{6.864e^{-50.4s}}{48s+1} & \dfrac{8.28e^{-54s}}{32s+1} \end{bmatrix}_{2\times 3} \tag{48}$$

With the controller parameters and filter parameters fixed, a step signal with amplitude 2 is applied to input 1 at $t = 100s$ and a unit step signal applied to input 2 at $t = 1600s$.

https://doi.org/10.1017/S0960129513000753 Published online by Cambridge University Press

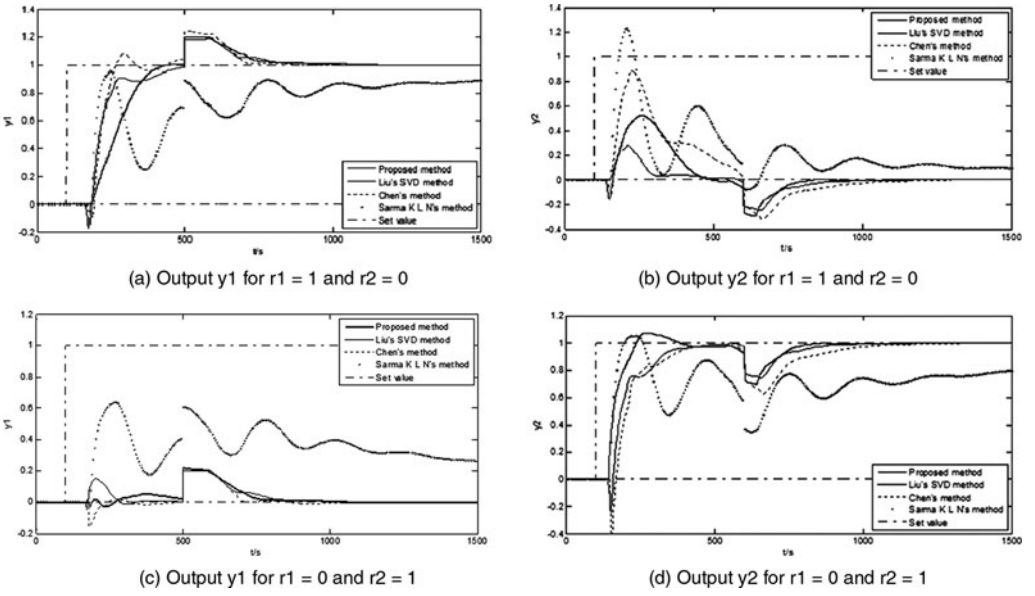



Fig. 5. Responses with disturbance.

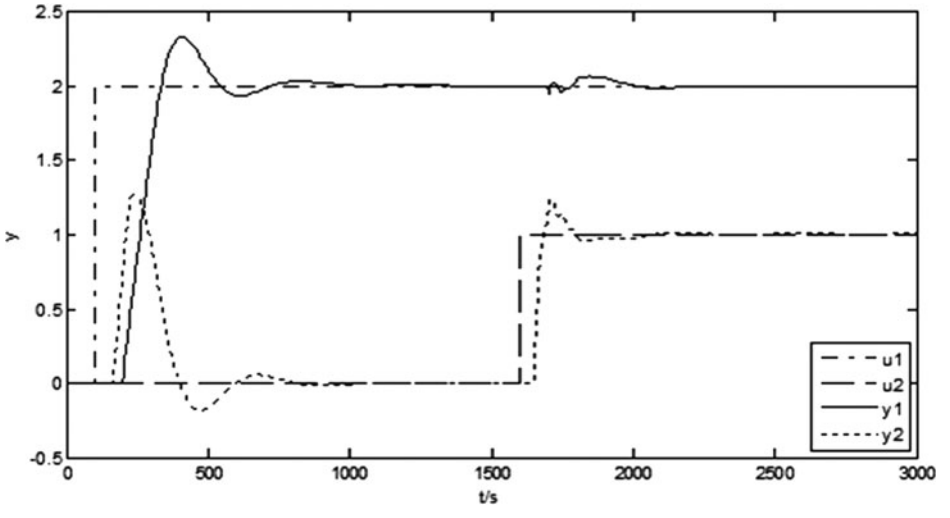



Fig. 6. Responses to given values for the mismatch model.

Figure 6 plots the input response and shows that the output of the controller can still reach the setting value rapidly and smoothly under model perturbation.

To prove that the control system can still maintain its control performance when there is output interference, we added a +20% step disturbance at $t = 1000$ to output 1 and a −20% step disturbance at $t = 2000$ to output 2.

https://doi.org/10.1017/S0960129513000753 Published online by Cambridge University Press

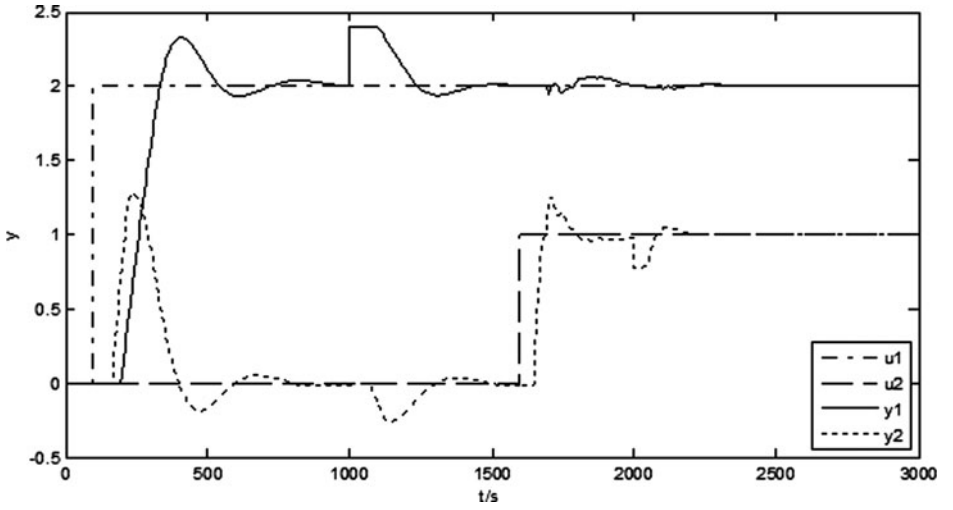



Fig. 7. Responses to disturbance for the mismatch model.

Figure 7 plots the response to the disturbance and shows that the controller has good anti-interference capability even when there is both model mismatch and interference. This demonstrates that the method has very good robustness and control performance.

## 4. Conclusions

In the current paper we have proposed an improved internal model PID control method for non-square systems in the field of industrial control. In this method, a complex non-square system is first decomposed into several single loop systems using a modified effective open-loop transfer function, and then a single loop IMC-PID controller is designed. The Taylor–PSO order reduction method was used to make the reduction more accurate. The simulation results show that the method has better system control performance and robustness than other control methods. This method has many good characteristics, including its ability to satisfy the control requirements of systems with many time delays, strong anti-interference capability and the fact that it can retain its good control performance when there is model perturbation and interference.

## References

Chen, P. Y., Ou, L. L., Gu, D. Y. and Zhang, W. D. (2009) Robust analytical scheme for linear non-square systems. *48th IEEE Conference on Decision and Control and 28th Chinese Control Conference*, Edinburgh University Press 1890–1895.

Chen, P. Y., Ou, L. L., Sun, J. and Zhang, W. D. (2008) Modified internal model control and its application in non-square processes. *Control and Decision* **23** (5) 581–584.

Davison, E. J. (1976) Multivariable tuning regulators: The feed forward and robust control of a general servo mechanism problem. *IEEE Transactions on Automatic Control* **20** (9) 17–41.

Dijkstra, E. W. (1976) *A Discipline of Programming*, Prentice-Hall.

https://doi.org/10.1017/S0960129513000753 Published online by Cambridge University Press

He, M. J., Cai, W. J., Wu, B. F. and He, M. (2005) Simple decentralized PID controller design method based on dynamic relative interaction analysis. *Industrial and Engineering Chemistry Research* **44** (8) 8334–8344.

Huang, H. P., Jeng, J. C., Chiang, C. H. and Pan, W. (2003) A direct method for multi-loop PI/PID controller design. *Journal of Process Control* **13** (7) 736–745.

Jin, Q. B., Guo, Y. and Liu, Z. Y. (2010) Modified internal model control and its application in non-square processes. *International Conference on Measuring Technology and Mechatronics Automation* 898–901.

Jin, Q. B., Liu, S. W., Quan, L. and Cao, L. T. (2011) Internal model control based on singular value decomposition and its application in non-square processes. *Acta Automatica Sinica* **37** (3) 354–359.

Kennedy, J. and Eberhart, R. C. (1995) Particle swarm optimization. *Proceedings IEEE International Conference on Neural Networks* 1942–1948.

Liu, H., Yuan, W. Y. and Jiang, D. Q. (2003) *Matrix Theory and Application*, Beijing: Chemical Industry Press 200–205.

Normey-Rico, J. E., Bordons, C. and Camacho, E. F. (1997) Improving the Robustness of Dead Time Compensating PI Controllers. *Control Engineering Practice* **5-6** 801–810.

Rao, A. S. and Chidambaram, M. (2006) Smith delay compensator for multivariable non-square systems with multiple time delays. *Computer and Chemical Engineering* **30** (8) 1243–1255.

Rivera, D. E., Morari, M. and Skogestad, S. (1986) Internal model control: 4. PID controller Design. *Industrial and Engineering Chemistry Process Design and Development* **25** (1) 252–265.

Roy, A. and Iqbal, K. (2005) PID controller tuning for the first-order-plus-dead-time process model via Hermite–Biehler theorem. *ISA Transactions* 363–378.

Sarma, K. L. N. and Chidambaram, M. (2005) Centralized PI/PID controllers for non-square systems with RHP zeros. *Journal of Indian Institute Science* **85** (4) 201–214.

Shi, Y. and Eberhart, R. C. (1998) A modified particle swarm optimizer. *IEEE International Conference of Evolutionary Computation* 69–73.

Tanttu, J. T. and Lieslehto, J. (1991) A comparative study of some multivariable PI controller tuning methods. *Intelligent tuning and adaptive control* **8** (39) 357–362.

Treiber, H. (1984) Multivariable control of non-square systems. *Industrial and Engineering Chemistry Process Design and Development* **23** (4) 854–857.

Vu, T. N. L. and Lee, M. (2010) Independent design of multi-loop PI/PID controllers for interacting multivariable processes. *Acta Automatica Sinica* **20** (9) 922–933.

Xiong, Q. and Cai, W. J. (2006) Effective transfer function method for decentralized control system design of multi-input multi-output processes. *Journal of Process Control* **16** (7) 773–784.

Xue, D. Y. (1994) Control System Optimal Reduction Technique and its Applications. *Control and Decision* **9** (6) 420–425.

Yao, Y. J., Wang, J. and Pan, L. D. (2008) Decoupling internal model control for multi-variable non-square system with time delays. *Journal of Chemical Industry and Engineering* **59** (7) 1737–1742.

Zhu, Z. X. (1996) Structural analysis and stability conditions of decentralized control systems. *Process Design and Control* **35** (7) 736–745.

https://doi.org/10.1017/S0960129513000753 Published online by Cambridge University Press