

Article

A Simple Multiloop Tuning Method for PID Controllers with No Proportional Kick

I-Lung ChienHsiao-Ping Huang, and Jen-Chien Yang

Ind. Eng. Chem. Res., **1999**, 38 (4), 1456-1468 • DOI: 10.1021/ie980595v

Downloaded from <http://pubs.acs.org> on November 18, 2008

More About This Article

Additional resources and features associated with this article are available within the HTML version:

- Supporting Information
- Links to the 8 articles that cite this article, as of the time of this article download
- Access to high resolution figures
- Links to articles and content related to this article
- Copyright permission to reproduce figures and/or text from this article

[View the Full Text HTML](#)



ACS Publications
High quality. High impact.

A Simple Multiloop Tuning Method for PID Controllers with No Proportional Kick

I-Lung Chien*

Department of Chemical Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan 106, Republic of China

Hsiao-Ping Huang and Jen-Chien Yang

Department of Chemical Engineering, National Taiwan University, Taipei, Taiwan 106, Republic of China

A simple tuning method for multiloop PID controllers will be presented in this article. The method is suited for PID algorithms with no proportional and derivative kick. This tuning method is derived from a controller synthesis method with a control performance specification of 5% overshoot on servo response. Depending on the interaction natures of the multiloop systems, tuning based on the diagonal elements of the model or further detuning may be necessary. For systems with a relative gain array $[RGA(\lambda_{ij})] < 1$, a detuning factor based on this information is proposed. The information needed for controller tuning purposes is the dynamic model parameters of the diagonal elements and the process gain information of the off-diagonal elements. The tuning method procedure is very simple and straightforward, utilizing only n th identification tests with n as the number of the interacting control loops. The tuning method can easily be applied to various industrial situations with almost no need for a priori process knowledge.

1. Introduction

Although many advanced control concepts have been introduced within the last 20 years, the vast majority of the controllers in industry are still of the PID type. For this reason, the proper tuning of the PID controllers so that they can perform to their expectation is an important factor for successful plant operation. There are many controller tuning methods proposed in the literature, but most of them only consider simpler SISO (single-input, single-output) systems. When loop interactions exist among the control loops, the otherwise acceptable control performance of SISO environments may quite often deteriorate to even unstable situations.

That is why, for multivariable systems, the tuning of a multiloop PID controller has to bring loop interactions into consideration. A common way to handle this problem is to introduce a detuning factor to the SISO tuning constants to stabilize the multivariable closed-loop system. If the process model of the multivariable system is available, a systematic procedure to find this detuning factor is suggested in the literature for multiloop PI tuning by Luyben.¹ Friman and Waller² suggested the use of autotuning by a relay feedback method to establish a multivariable process model and the use of a trial-and-error procedure to determine the detuning factor. Shen and Yu³ utilized sequential relay feedback identification to obtain the frequency response information of a multivariable system and to use detuned Ziegler–Nichols tuning rules to handle the loop interactions. Loh et al.⁴ used a sequential identification method similar to that of Shen and Yu³ but proposed the use of a modified PID algorithm with an additional

β parameter to decrease the interactions among control loops. Palmor et al.⁵ and Halevi et al.⁶ proposed not to use a sequential relay identification method but instead used simultaneous relays on all of the control loops during the identification stage. They obtained the desired critical point information from their experiment after several iterations and then used regular or modified Ziegler–Nichols tuning rules for control.

All of the tuning methods mentioned above had to obtain the process model or the frequency response information in “several rather long” experiments. In this paper, an alternative, very simple multiloop PID Tuning method is proposed. The method is suited for a PID algorithm with no proportional and derivative “kicks”. This tuning method is derived from a controller synthesis method with a control performance specification of 5% overshoot of the servo response. Depending on the interaction natures of the multiloop systems, controller tuning based on the diagonal elements of the model or further detuning may be necessary. For systems with a relative gain array $[RGA(\lambda_{ij})] < 1$, a detuning factor based on this information is proposed. The information needed for controller tuning purposes includes the dynamic model parameters of the diagonal elements and the process gain information of the off-diagonal elements. The tuning method procedure is very simple and straightforward, utilizing only n th identification tests with n as the number of the interacting control loops. The tuning method can easily be applied to various industrial situations with almost no need for a priori process knowledge.

Various distillation dual-point control examples with $RGA(\lambda_{ij}) > 1$ and two other examples, including an industrial reactor control example with $RGA(\lambda_{ij}) < 1$ and one 3×3 example, will be used to demonstrate the superior performance of this tuning method. Both set-point and load disturbance response will be examined.

* To whom all correspondence should be addressed. Phone: +886-2-2737-6652. Fax: +886-2-2737-6644. E-mail: Chien@ch.ntust.edu.tw.

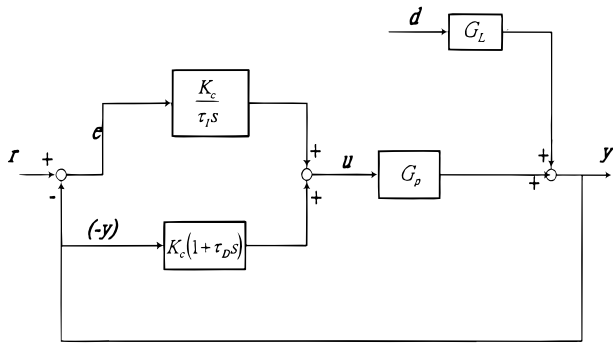


Figure 1. Block diagram of PID control loop with no proportional kick.

2. The Tuning Method

In a standard PID controller, because the proportional mode is acting on the error signal, there will be a large kick of the controller output during setpoint changes. If the multiloop control system has large interactions among the loops, the large controller action on one loop will be a large load disturbance to the other loops. To reject this large load disturbance, the corrective action by the controllers in other loops will also be large, causing disturbance back to the loop with setpoint changes. The disturbance effect will go back and forth among the control loops, causing control problems.

For this reason, an obvious solution is to not let the control action have the kick at the time of the setpoint changes. This calls for a PID form with no proportional kick, which simply means having the proportional mode acting on the negative sign of the controlled variable alone. The Laplace transform representation of this control is

$$u(s) = K_c \{-y(s) + (1/\tau_i s)E(s) + \tau_d s[-y(s)]\} \quad (1)$$

In the PID form above, it is customary to have the derivative mode also acting on the negative sign of the controlled variable to prevent a derivative kick. Also note that this PID form can be found in almost every industrial distributed control systems, thus no special control implementation is needed.

This PID form in eq 1, on one hand, will not have an initial proportional kick, but on the other hand, the closed-loop servo response will be sluggish if the same PID tuning constants derived from a standard PID form is used. For this reason, the PID tuning rules need to be modified for this particular PID form. We will use controller synthesis method to derive the PID controller tuning parameters. In the following, the derivation of the tuning rules will be divided into two parts by obtaining PI tuning rules and PID tuning rules. For controlled variables with noisy signals, PI tuning rules with no derivative action can be selected.

2.1. PI Tuning Rules. We will start our discussions by considering a single-loop control system. For a closed-loop system with a PID controller with no proportional kick (see Figure 1), the relationship between the controlled variable (y) and the setpoint (r) is as follows (dropping the Laplace transform variable s for simplicity reason):

$$\frac{y}{r} = \frac{(K_c/\tau_i s) G_p}{1 + \frac{K_c(\tau_i s + 1)}{\tau_i s} G_p} \quad (2)$$

G_p is the model of the process for controller tuning purposes. We will classify the processes into two categories.

2.1.1. Time Constant Dominant Processes. Most of the difficult control loops in the chemical industry are of this type with the dominant time constant of the system greater than 5 times the process deadtime. If this holds, the proper process model for controller tuning purposes will have the following integrating plus dead-time form:

$$G_p = \frac{R e^{-Ls}}{s} \quad (3)$$

where R is the slope of the initial unit step response of the controlled variable and L is the process apparent deadtime. If a first-order plus deadtime model is obtained from some model identification steps with K_p as the process gain, τ as the process time constant, and L as the process deadtime, for $(L/\tau) < 0.2$, the approximation of calculating R to be equal to (K_p/τ) will be used.

In order to derive PI tuning parameters, the deadtime term in eq 3 needs to be approximated using a first-order Taylor series, thus

$$G_p = R e^{-Ls}/s \approx R(1 - Ls)/s \quad (4)$$

Substituting eq 4 into eq 2 and simplifying, we obtain

$$\frac{y}{r} = \frac{1 - Ls}{\left(\frac{\tau_i}{RK_c} - \tau_i L\right) s^2 + (\tau_i - L)s + 1} \quad (5)$$

The closed-loop servo response is a second-order response. Let's assume our desired closed-loop servo response to be an underdamped system with a damping coefficient (ζ) of 0.707. This corresponds to a closed-loop system with about 5% overshoot. The desired closed-loop servo response is

$$\left(\frac{y}{r}\right)^{\text{desired}} = \frac{e^{-Ls}}{\tau_{cl}^2 s^2 + 1.414\tau_{cl}s + 1} \approx \frac{1 - Ls}{\tau_{cl}^2 s^2 + 1.414\tau_{cl}s + 1} \quad (6)$$

where τ_{cl} is a user-specified closed-loop effective time constant. Equating eqs 5 and 6, we can obtain the PI tuning rules as

$$K_c = (1.414\tau_{cl} + L)/R(\tau_{cl}^2 + 1.414\tau_{cl}L + L^2) \quad (7)$$

$$\tau_i = 1.414\tau_{cl} + L \quad (8)$$

2.1.2. Deadtime Important Processes. For processes with a deadtime greater than $1/5$ of the process time constant, it is better for controller tuning purpose to model the processes as a first-order plus deadtime model. With the same Taylor approximation, the process model becomes

$$G_p = K_p e^{-Ls}/(\tau s + 1) \approx K_p(1 - Ls)/(\tau s + 1) \quad (9)$$

Substituting eq 9 into eq 2 and simplifying, we obtain

$$\frac{y}{r} = \frac{1 - Ls}{\left(\frac{\tau_i \tau}{K_c K_p} - \tau_i L\right) s^2 + \left(\frac{\tau_i}{K_c K_p} + \tau_i - L\right) s + 1} \quad (10)$$

Again, it is a second-order response. With the same desired closed-loop servo response as previously eq 6, we can obtain the PI tuning rules as

$$K_c = \frac{1}{K_p} \frac{-\tau_{cl}^2 + 1.414\tau_{cl}\tau + L\tau}{\tau_{cl}^2 + 1.414\tau_{cl}L + L^2} \quad (11)$$

$$\tau_i = \frac{-\tau_{cl}^2 + 1.414\tau_{cl}\tau + L\tau}{\tau + L} \quad (12)$$

Normally, we would specify τ_{cl} to be smaller than the open-loop time constant, τ ; thus, the negative terms in eqs 11 and 12 will not create any problems in changing the signs of K_c and τ_i .

2.2. PID Tuning Rules. The derivation of the PID tuning parameters is similar to the above PI derivation except that the deadtime approximation is a little more accurate because of the use of the first-order Padé approximation. In this case, then, the apparent deadtime terms in integrating plus deadtime or first-order plus deadtime models can be approximated to be

$$e^{-Ls} \approx \left(1 - \frac{L}{2}s\right) \left/ \left(1 + \frac{L}{2}s\right) \right. \quad (13)$$

We will use the same classification of the process model and the same closed-loop specification as in the above PI tuning rule derivation. The resulting PID tuning rules with detail derivation can be found in the Appendix.

3. Multiloop Considerations

In the above sections, we have developed the PI/PID tuning rules on the basis of a controller synthesis method. We have also assumed that by using PI/PID form with no proportional kick the interactions among the control loops will be considerably reduced. For a multiloop situation, the relationship between y_i and u_i is not simply the diagonal element of the process model matrix. Let's take a 2×2 system as an illustrating example. For a 2×2 closed-loop control system, the process transfer function between y_1 and u_1 while loop 2 is on manual is

$$\left(\frac{y_1}{u_1}\right)_{\text{loop 2 opened}} = G_{11} \quad (14)$$

On the other hand, the relationship between y_1 and u_1 while loop 2 is on automatic mode is (cf., Shen and Yu³)

$$\left(\frac{y_1}{u_1}\right)_{\text{loop 2 closed}} = G_{11}(1 - \kappa h_2) \quad (15)$$

where κ is the Rijnsdorp interaction measure⁷ and h_2 is the complementary sensitivity function for loop 2. The κ and h_2 are defined as

$$\kappa = G_{12}G_{21}/G_{11}G_{22} \quad (16)$$

and

$$h_2 = G_{22}G_{c_2}/(1 + G_{22}G_{c_2}) \quad (17)$$

The above relationship in eq 15 holds also for PID controller with the form of no proportional kick.

Assuming that the controller tuning parameters are chosen so that the complementary sensitivity function (h_2) is a stable system with unity gain, then by calculating the high- and low-frequency asymptotes of eq 15 we can have some idea of how the loop interactions affect the open-loop transfer function. For individual process transfer function to be assumed as a first-order plus deadtime model, at high frequency, this model is approaching integrating plus deadtime form as in eq 3 and at low frequency this model is approaching a pure gain model. The following results are obtained:

$$\text{At } \omega \rightarrow \infty \quad \left(\frac{y_1}{u_1}\right)_{\text{loop 2 closed}} = G_{11} \quad (18)$$

At $\omega \rightarrow 0$

$$\left(\frac{y_1}{u_1}\right)_{\text{loop 2 closed}} = G_{11} \left(1 - \frac{K_{p12}K_{p21}}{K_{p11}K_{p22}}\right) = \frac{G_{11}}{RGA(\lambda_{ij})} \quad (19)$$

because for controller design or tuning purposes, the process open-loop initial response (or high-frequency) behavior is the most important (as pointed by others; see: Skogestad and Moraris,^{8,9} Chien and Fruehauf,¹⁰ and Chien and Ogunnaike¹¹). In addition, from frequency asymptote analysis, the initial response of a particular control loop with the other loop in manual or automatic is known to be very similar in nature (from eqs 14 and 18). For these reasons, multiloop controller tuning based on the process model parameters in the main loop should provide satisfactory closed-loop results.

For systems with $RGA(\lambda_{ij}) < 1$, special consideration should be taken to avoid having overly aggressive controller action. From eq 19, the process transfer function is adjusted by multiplying a factor with values greater than one [reciprocal of $RGA(\lambda_{ij})$] at low frequencies. Using the tuning rules based on the open-loop transfer function, G_{ii} , will be too aggressive at low-frequency ranges. To obtain PID controller tuning parameters which will not be too aggressive in all frequency ranges, a reasonable choice is to introduce an extra detuning factor with the value of $RGA(\lambda_{ij})$ to detune K_c as well as τ_i and τ_d parameters in the following matter:

$$K_c = (K_c)_{\text{based on main loop}} RGA(\lambda_{ij}) \quad (20)$$

$$\tau_i = \frac{(\tau_i)_{\text{based on main loop}}}{RGA(\lambda_{ij})} \quad (21)$$

$$\tau_d = (\tau_d)_{\text{based on main loop}} RGA(\lambda_{ij}) \quad (22)$$

From the definition of $RGA(\lambda_{ij})$, we can also see that further detuning is necessary for systems with $RGA(\lambda_{ij}) < 1$. The $RGA(\lambda_{ij})$ is defined as¹²

$$\lambda_{ij} = \left(\frac{\partial y_i}{\partial u_j}\right)_{\text{other loops opened}} \left/ \left(\frac{\partial y_i}{\partial u_j}\right)_{\text{other loops closed}} \right. \quad (23)$$

Thus, if $RGA(\lambda_{ij}) < 1$, this means that the effective process gain of the main loop will be increased if the other loops are closed. Therefore, any controller tuning methods only based on the information of model parameters in the main loop will be too aggressive. This calls for an extra detuning factor. The value of the detuning factor should be related to the closeness of the

Table 1. PI Tuning Rules for PID Controller with No Proportional Kick (2 × 2 system)^a

	$(L/\tau) < 0.2$	$0.2 < (L/\tau) < 0.5$	$(L/\tau) > 0.5$
K_c	$1/2.045RL$	$\frac{1}{K_p} \frac{(1.414m + 1) - m^2(L/\tau)}{(m^2 + 1.414m + 1)(L/\tau)}$	$\frac{1}{K_p} \frac{2.414 - (L/\tau)}{3.414(L/\tau)}$
τ_i	$3.828L$	$\tau \frac{(1.414m + 1)(L/\tau) - m^2(L/\tau)^2}{1 + (L/\tau)}$	$\tau \frac{2.414(L/\tau) - (L/\tau)^2}{1 + (L/\tau)}$

^a For the cases when $0.2 < (L/\tau) < 0.5$, $m = 2 - [(L/\tau) - 0.2]/0.3$ is used.

Table 2. PID Tuning Rules for PID Controller with No Proportional Kick (2 × 2 system)^a

	$(L/\tau) < 0.2$	$0.2 < (L/\tau) < 0.5$	$(L/\tau) > 0.5$
K_c	$1/1.48RL$	$\frac{1}{K_p} \frac{(0.25 - m^2)(L/\tau) + (1.414m + 1)}{(m^2 + 0.707m + 0.25)(L/\tau)}$	$\frac{1}{K_p} \frac{2.414 - 0.75(L/\tau)}{1.957(L/\tau)}$
τ_i	$3.828L$	$\tau \frac{(0.25 - m^2)(L/\tau)^2 + (1.414m + 1)(L/\tau)}{1 + 0.5(L/\tau)}$	$\tau \frac{2.414(L/\tau) - 0.75(L/\tau)^2}{1 + 0.5(L/\tau)}$
τ_d	$0.435L$	$\tau \frac{(0.707m + 0.25)(L/\tau) - 0.5m^2(L/\tau)^2}{(0.25 - m^2)(L/\tau) + (1.414m + 1)}$	$\tau \frac{0.957(L/\tau) - 0.5(L/\tau)^2}{2.414 - 0.75(L/\tau)}$

^a For the cases when $0.2 < (L/\tau) < 0.5$, $m = 2 - [(L/\tau) - 0.2]/0.3$ is used.

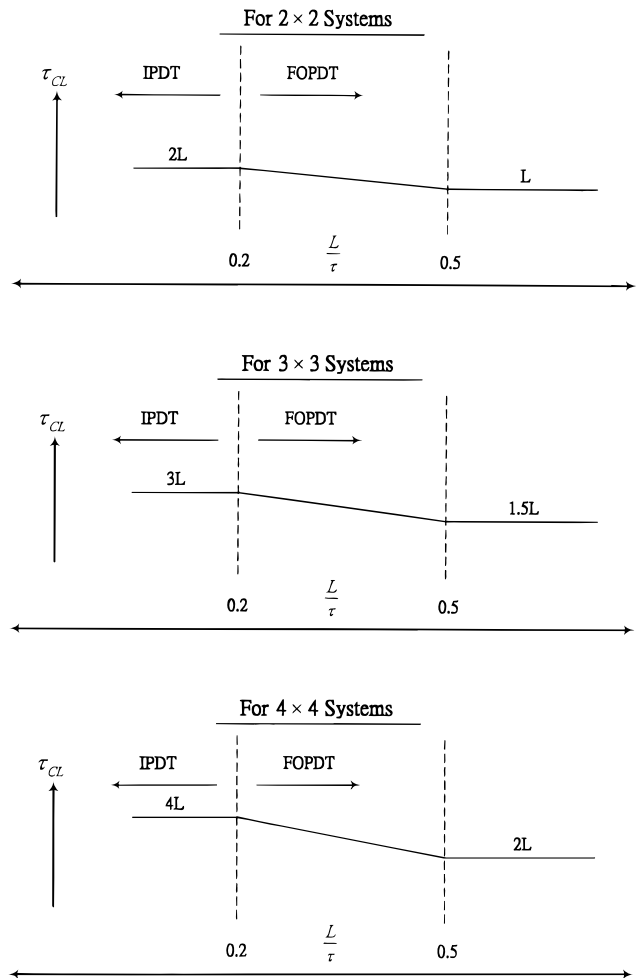
open-loop process gain to the process gain when the other loops are closed. For this reason, the detuning rules in eqs 20–22 are justified.

The choosing of the user-specified closed-loop effective time constant, τ_{cl} , is a matter of trading off between control loop performance and robustness. If τ_{cl} is chosen to be smaller, then the controller performance is faster, but the control action is more vigorous, and the model mismatch tolerance is worse. On the other hand, if τ_{cl} is chosen to be larger, the controller performance is more sluggish, but the control action will be smoother, and the model mismatch tolerance is better. With many multiloop system simulations, the following selection of τ_{cl} in Figure 2 was determined empirically, which gave satisfactory closed-loop response under both nominal and reasonable model mismatch conditions.

The reason for choosing τ_{cl} to be smaller for first-order plus deadtime model when $L/\tau > 0.5$ is because for deadtime dominant processes with large L values selecting $\tau_{cl} = nL$ with n as number of interacting control loops will give overly sluggish closed-loop responses. For $L/\tau > 1.0$, a PID controller will be sluggish anyhow. In this circumstance, a deadtime compensation scheme such as the Smith predictor will be more beneficial.

The resulting PI tuning rules for a 2 × 2 system can be found in Table 1. From the process model information obtained from model identification steps, the PI tuning parameters can easily be calculated. For a 3 × 3 or 4 × 4 system, the PI tuning parameters can be calculated in a manner similar to that of Table 1 by selecting τ_{cl} as in Figure 2 and using eqs 7, 8, 11, and 12 to calculate K_c and τ_i . Similarly, the PID tuning parameters for a 2 × 2 system can easily be calculated with the selection of τ_{cl} as in Figure 2. The results can be seen in Table 2 once the model parameters are known. For a 3 × 3 or 4 × 4 system, the PID tuning parameters can be calculated in a similar matter.

In calculating the controller tuning parameters, any suitable model identification schemes such as the biased-relay method proposed by Huang et al.¹³ can be used to obtain the necessary information by just using n th relay feedback test with n as the number of the interacting control loops. For systems with $\text{RGA}(\lambda_{ii}) > 1$, the PI/PID tuning rules can directly be calculated by using the model parameters (R , L , K_p , and τ) in the main

**Figure 2.** Selection of τ_{cl} in multiloop control system.

loops. As for systems with $\text{RGA}(\lambda_{ii}) < 1$, additional information of the process gains in the off-diagonal elements are also needed. The biased-relay method proposed by Huang et al.¹³ is suitable for calculating the process gains because these values can be obtained by calculating the ratio of two integrals in $y_i(t)$ and $u_j(t)$ for each cycle of the test result and then taking the average of several cycles.

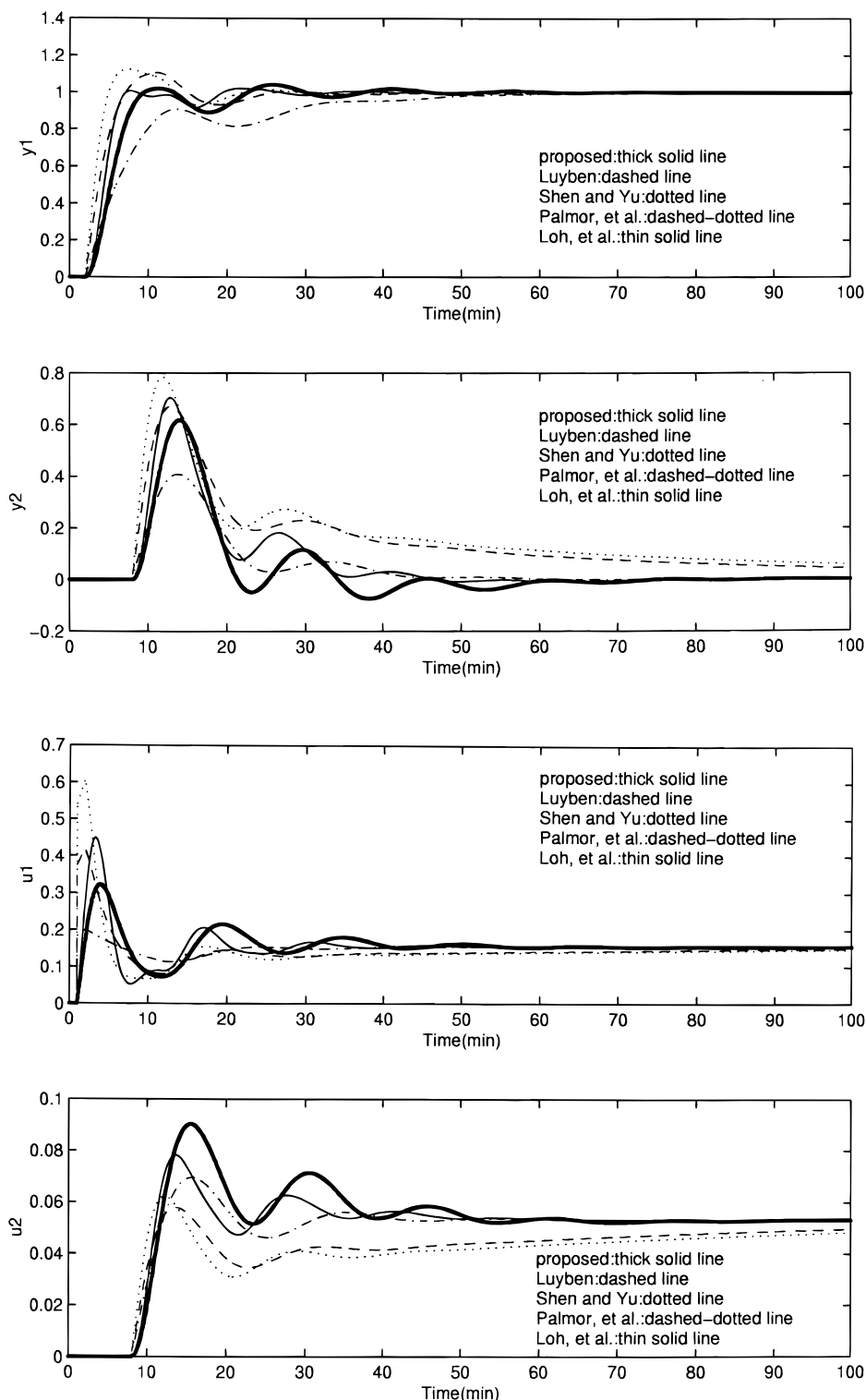


Figure 3. (a) Closed-loop response for example 1 with y_1 setpoint change. (b) Manipulated variables of a.

The overall tuning procedure is summarized in the following:

1. With all loops in manual mode, one can apply identification tests on loop 1 and collect dynamic data on u_1 versus y_1 and u_1 versus $y_{i,i \neq 1}$.

2. By using any suitable identification method, one can obtain first-order plus deadtime model parameters for G_{11} and process gain information, $K_{p1,i \neq 1}$.

3. Similar test apply to other loops and give first-order plus deadtime model parameters for all of the diagonal elements in the process model matrix and gain information for all of the off-diagonal elements.

4. If $RGA(\lambda_{ij}) > 1$, only model parameters in G_{ii} will be used in the tuning rules.

5. With $RGA(\lambda_{ij}) > 1$, one can calculate L/τ for each loop. For the cases with $(L/\tau) < 0.2$, $R (= K_p/\tau)$ can be calculated in each loop. For each L/τ range, PI/PID tuning parameters can be calculated from appropriate columns in the tables mentioned in this section.

6. If $RGA(\lambda_{ij}) < 1$, a detuning factor equal to $RGA(\lambda_{ij})$ can be calculated from the model parameters.

7. With $RGA(\lambda_{ij}) < 1$, the controller tuning parameters can be further detuned via eqs 20–22.

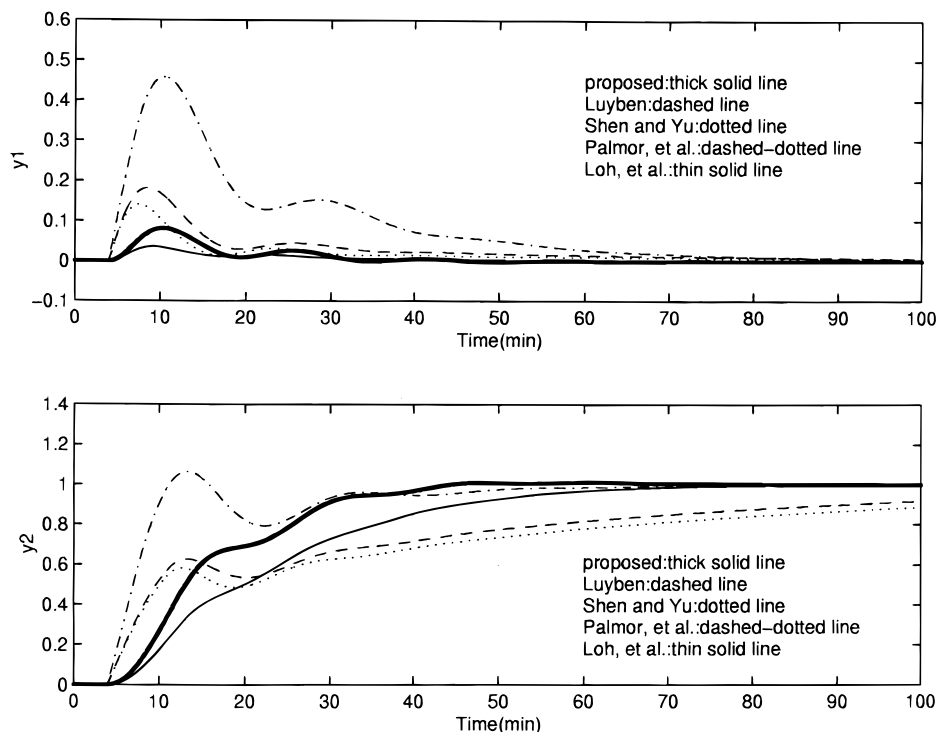


Figure 4. Closed-loop response for example 1 with y_2 setpoint change.

Table 3. PI Tuning Parameters for Wood and Berry Column

	K_{c1}	τ_{i1}	K_{c2}	τ_{i2}
Luyben	0.375	8.29	-0.075	23.6
Shen and Yu	0.540	7.92	-0.072	26.7
Palmor et al.	0.183	10.7	-0.119	10.7
Loh et al.	0.868	3.25	-0.087	10.4
proposed	0.637	3.84	-0.096	7.40

8. With the tuning parameters calculated, a PI/PID controller can be implemented with no proportional and derivative kick in your plant.

This proposed tuning method is suitable for all normal multiloop control applications. For the situation when a multiloop control system has $\text{RGA}(\lambda_{ij}) < 0$, the recommendation is to switch the pairings so that $\text{RGA}(\lambda_{ij})$ will become greater than 0. From the definition of RGA in eq 23, the sign of the controller action will be reversed when the other loops are in automatic or in manual in the situation when $\text{RGA}(\lambda_{ij}) < 0$. This means that even if the multiloop control system is tuned properly to be stable in the automatic mode, any failure in the loops in the system which causes the loops to enter a manual mode will destabilize the overall control system. The control integrity is unacceptable for a $\text{RGA}(\lambda_{ij}) < 0$ system.

4. Simulation Results

Many 2×2 systems and several 3×3 systems have been used to test the closed-loop performance of the proposed tuning method above. All give very satisfactory responses. In the following, we will present the results on a simulated column control system with $\text{RGA}(\lambda_{ii}) > 1$ and one reactor control example with $\text{RGA}(\lambda_{ii}) < 1$ as well as a 3×3 example.

Example 1. Wood and Berry Column.¹⁴ The transfer function model shown below is a pilot-scale distillation column model separating a mixture of methanol and water. This model has been discussed in a number

of other 2×2 loop tuning studies, including those by Luyben,¹ Loh et al.,⁴ Shen and Yu,³ and Palmor et al.⁵ Wood and Berry¹⁴ reported the following empirical transfer function model:

$$\begin{bmatrix} y_1(s) \\ y_2(s) \end{bmatrix} = \begin{bmatrix} 12.8e^{-s} & -18.9e^{-3s} \\ 16.7s + 1 & 21.0s + 1 \\ 6.6e^{-7s} & -19.4e^{-3s} \\ 10.9s + 1 & 14.4s + 1 \end{bmatrix} \begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix} + \begin{bmatrix} 3.8e^{-8.1s} \\ 14.9s + 1 \\ 4.9e^{-3.4s} \\ 13.2s + 1 \end{bmatrix} [d(s)] \quad (24)$$

The first step of the controller tuning procedure is to do two bias-relay feedback tests on loops 1 and 2. From the model identification results and from the fact that this system has the characteristics of $\text{RGA}(\lambda_{ii}) > 1$, no detuning is necessary. The PI or PID tuning parameters can be calculated from Tables 1 or 2 respectively. The closed-loop setpoint and load responses using this proposed tuning method will be compared to the tuning methods by Luyben,¹ Shen and Yu,³ Palmor et al.,⁵ and Loh et al.⁴ The PI tuning parameters used in the above tuning methods are tabulated in Table 3.

The closed-loop response for a setpoint change in y_1 is shown in Figure 3a with the manipulated variable changes in Figure 3b. It is clearly shown in these two figures that controller tunings using PI controllers with no proportional kick (our proposed method and the one by Loh et al.⁴) provide better closed-loop performance than the ones using standard PI controllers (Luyben,¹ Shen and Yu,³ and Palmor et al.⁵). Although the performance of Loh et al.⁴ is satisfactory, the y_2 response is more sluggish. In addition, the method of Loh et al. uses the more time-consuming "sequential" identification procedure for obtaining its tuning constants. The

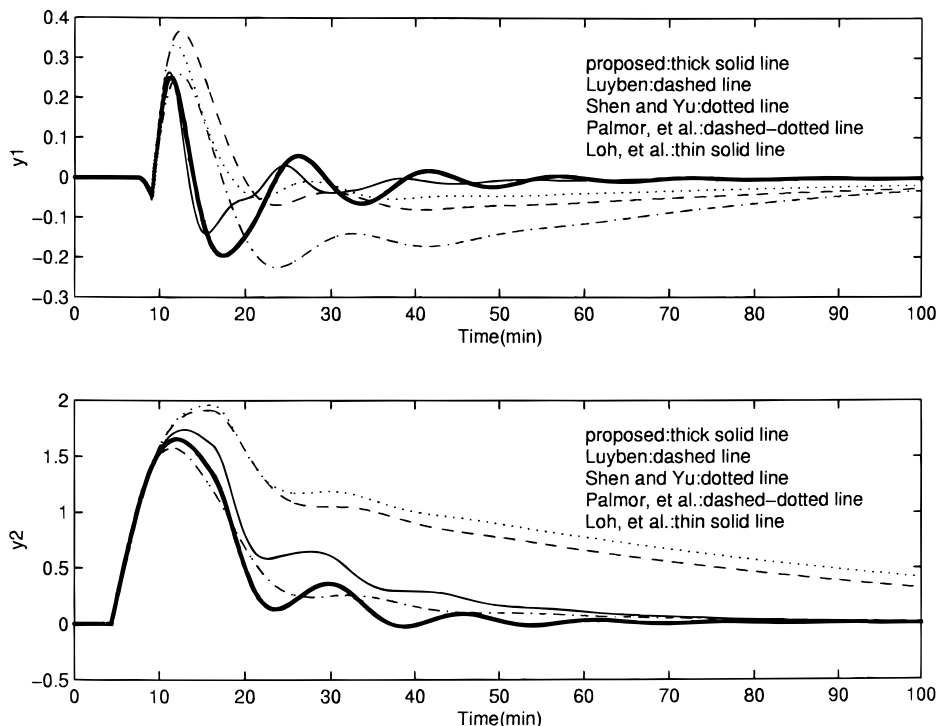


Figure 5. Closed-loop response for example 1 with load change.

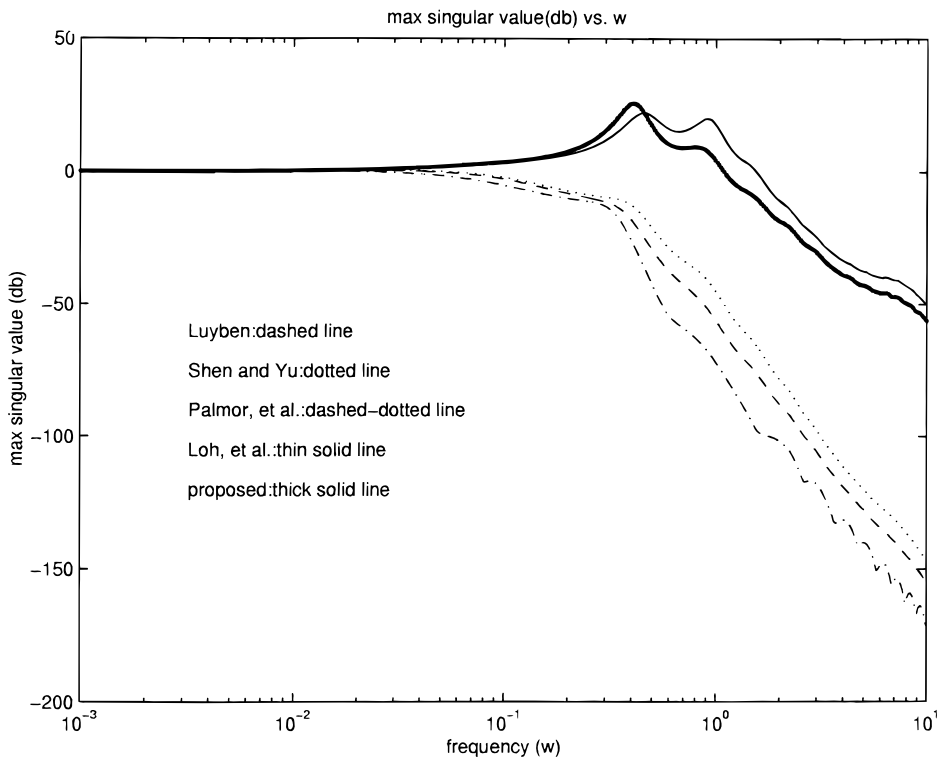


Figure 6. Plot of $\bar{\sigma}[H(i\omega)]$ vs ω for example 1.

manipulated variables of our proposed method are quite smooth, which is favorable in industrial applications.

Closed-loop responses for changes in setpoint of y_2 and the load response with the disturbance, d , from 0 to 1 at $t = 1$ can be found in Figures 4 and 5, respectively. Again, our proposed method reaches the setpoint the fastest in these two changes and is superior in comparison to other tuning methods.

The comparison of the controller robustness property can be carried out by assuming the plant with an input

uncertainty Δ to describe modeling errors. The closed-loop system is stable if and only if (cf., Morari and Zafiriou¹⁵)

$$\bar{\sigma}[H(i\omega)] < \frac{1}{\bar{\sigma}[\Delta(i\omega)]} \quad \forall \omega \quad (25)$$

where $\bar{\sigma}(\cdot)$ is the maximum singular value of (\cdot) and H is the complementary sensitivity function of the entire closed-loop system. By plotting $\bar{\sigma}[H(i\omega)]$ versus ω for the

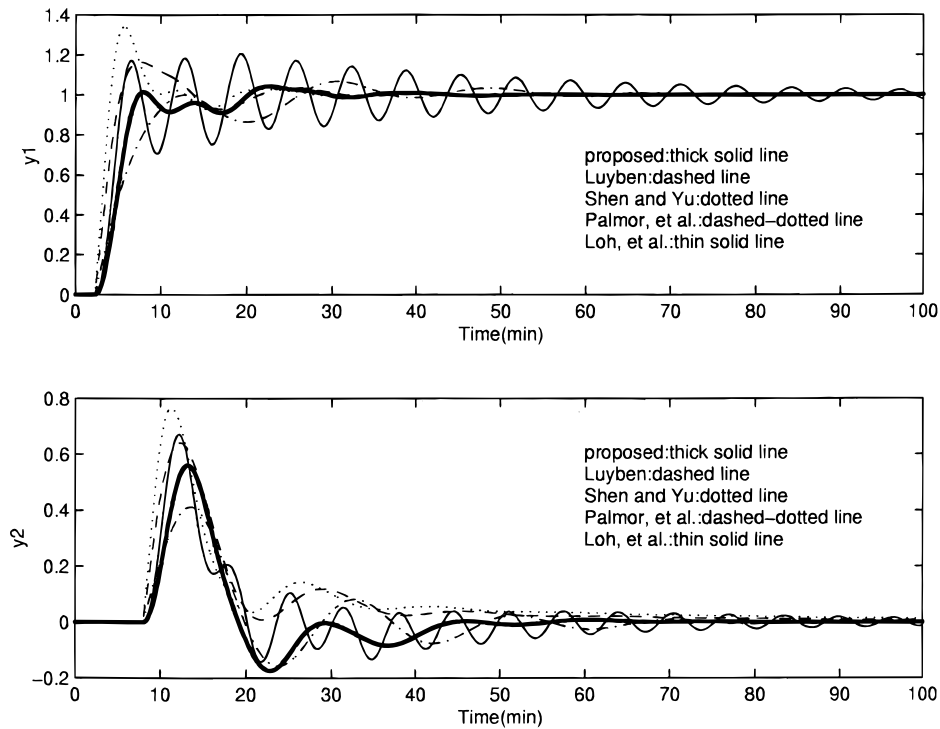


Figure 7. Model mismatch run for example 1 with y_1 setpoint change.

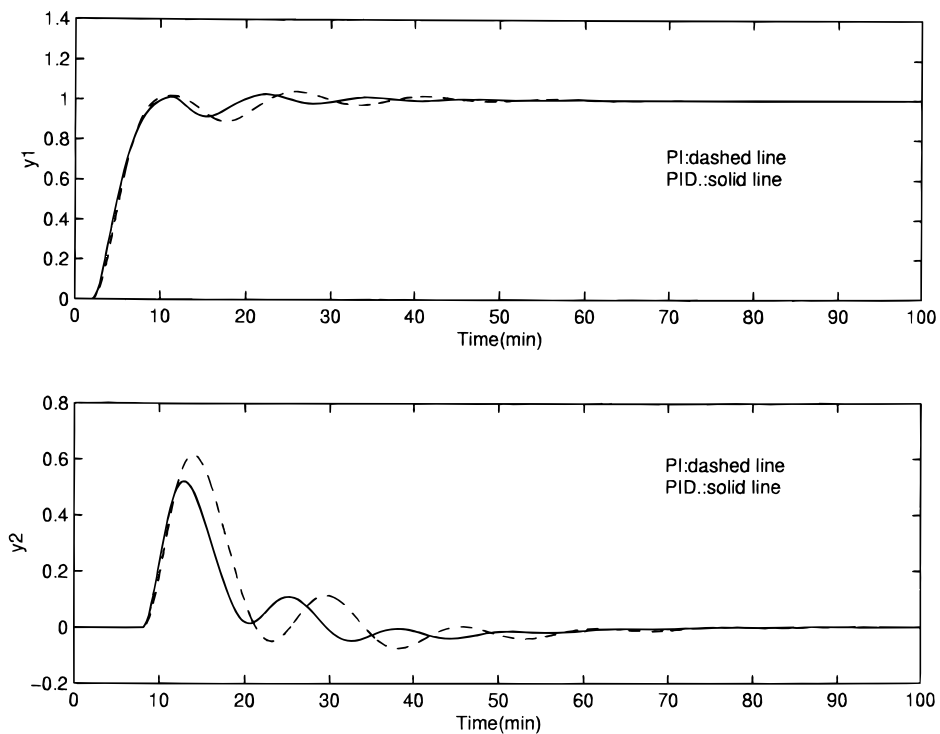


Figure 8. PID vs PI response for example 1 with y_1 setpoint change.

five tuning rules listed in Table 3, we can compare the robust stability of these five tuning rules. The tuning method will be more robust if the value of $\bar{\sigma}[H(i\omega)]$ is smaller in all frequency ranges.

Figure 6 is a plot of $\bar{\sigma}[H(i\omega)]$ versus ω for this system. From this figure, we see that the proposed tuning method is more robust than the method by Loh et al.⁴ for most frequencies but is less robust than the other three tuning methods for high frequencies. Even though the robust stability property of the methods by Luyben,¹

Shen and Yu,³ and Palmor et al.⁵ are favorable, the nominal performances of these three tuning methods are considerably more sluggish, as shown in above Figures 3–5.

One of the criticisms of using $\bar{\sigma}[H(i\omega)]$ in robust stability analysis is that the stability criterion may be too conservative. As an independent check, closed-loop results using the tuning constants as in Table 3 but with process parameters mismatch of +30% in diagonal gains and deadtime elements are simulated. Figure 7 shows

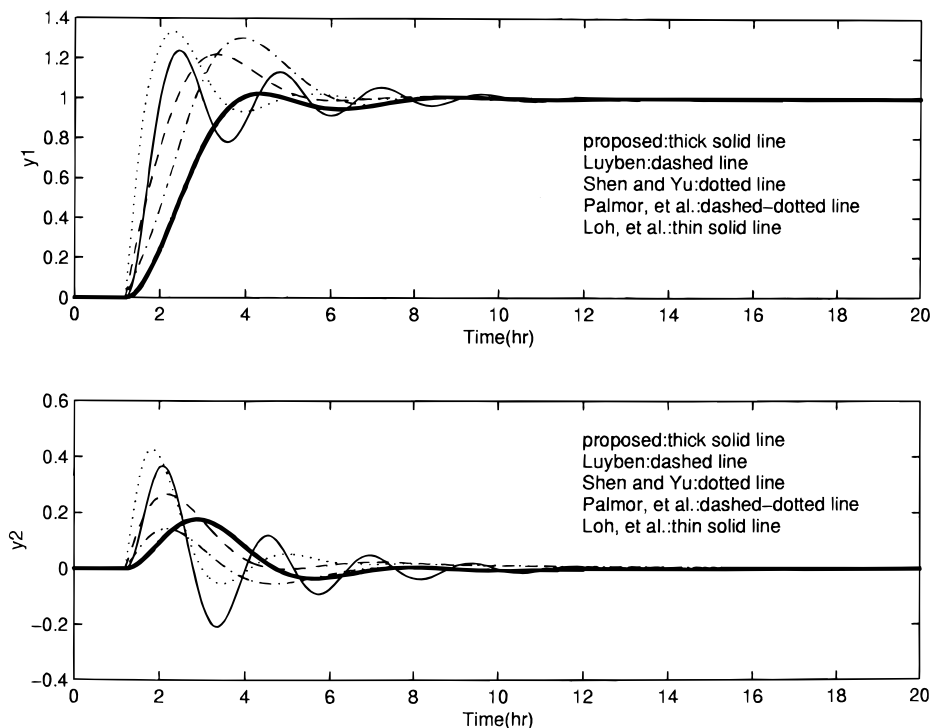


Figure 9. Closed-loop response for example 2 with y_1 setpoint change.

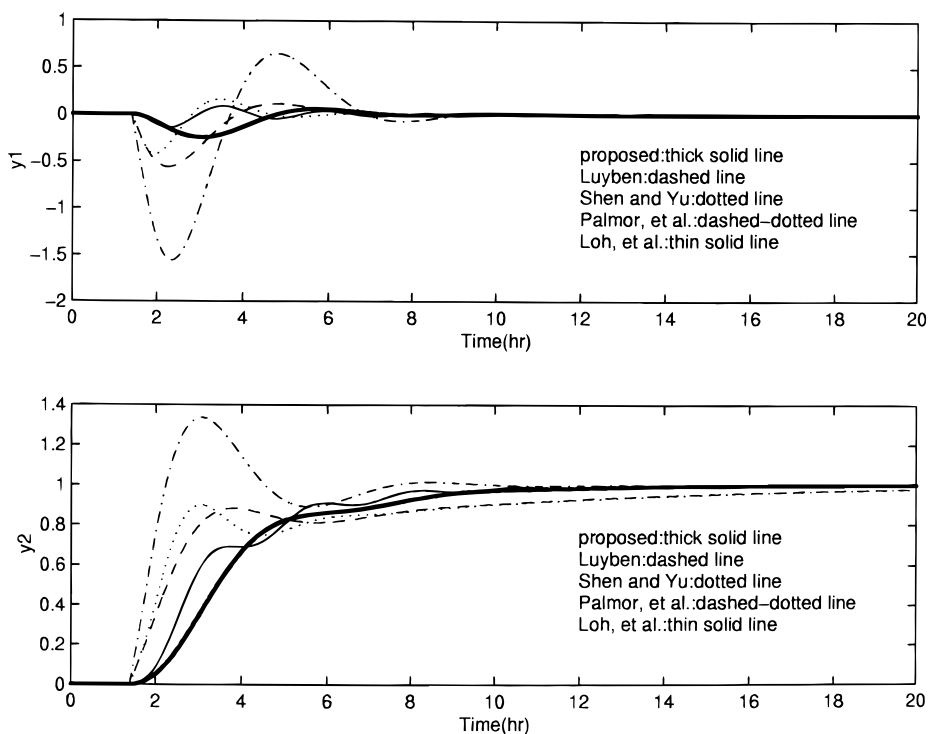


Figure 10. Closed-loop response for example 2 with y_2 setpoint change.

the setpoint change in y_1 under model mismatch condition. The proposed method still provides excellent closed-loop performance with this severe model mismatch condition. The methods of Luyben,¹ Shen and Yu,³ and Palmor et al.⁵ are still stable because their controller actions are too sluggish during nominal condition. The method by Loh et al.,⁴ although the nominal response is satisfactory, is much too oscillatory under this model mismatch condition. Considering the

trade-off between loop performance and robustness, the proposed tuning method is the most favorable.

In order to further improve the closed-loop response, a controller with an additional derivative mode can be considered if the process variable measurements are not very noisy. The PID tuning parameters for this column can be calculated from Table 2 to be $K_{c1} = 0.881$, $\tau_{i1} = 3.84$, $\tau_{d1} = 0.436$, $K_{c2} = -0.136$, $\tau_{i2} = 8.24$, and $\tau_{d2} = 1.23$. The closed-loop response of the PID controller with no

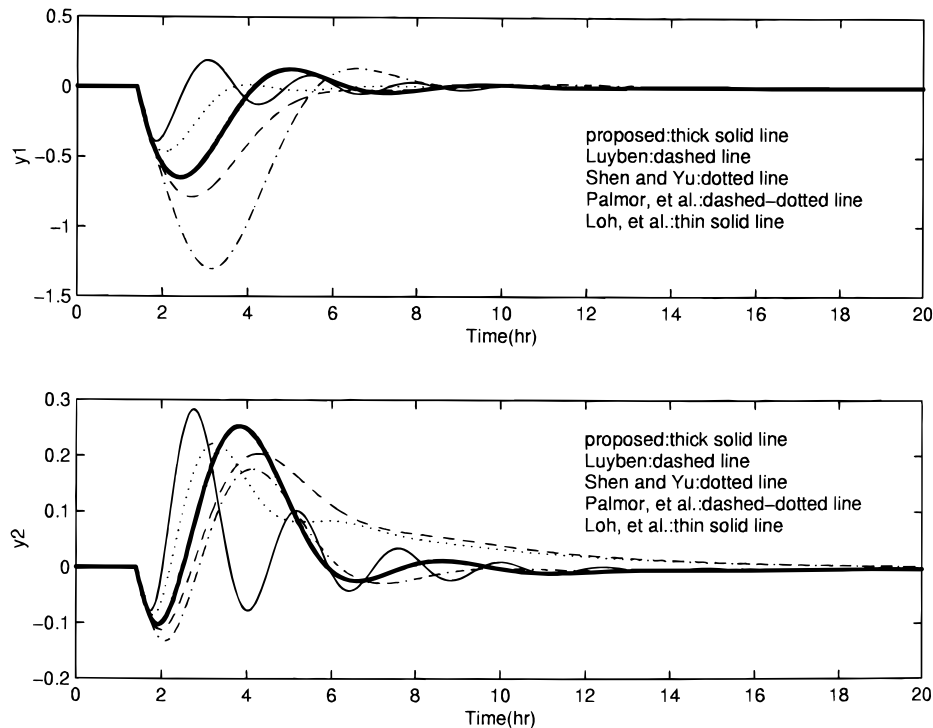


Figure 11. Closed-loop response for example 2 with load change.

proportional kick using the proposed tuning method is comparing to the response of a PI controller in Figure 8 for a setpoint change in y_1 . The closed-loop performance of PID controller is further improved.

Example 2. A Reactor Control Problem. The second example is an industrial-scale polymerization reactor control problem. In order to improve the product quality consistency, a control improvement project is initiated for which the first phase of the project is to do automatic reactor condition control. The manipulated variables of this control tier are the setpoints of various reactor feed flow control loops. The control hierarchy proposed is similar to Figure 30.25 in Ogunnaike and Ray.¹⁶ The future higher tier product quality control loops will be setting the setpoints of the reactor condition control loops.

For proprietary reasons, we will not describe in detail this reactor control system. The process model for the reactor condition control loops is obtained from model identification to be

$$\begin{bmatrix} y_1(s) \\ y_2(s) \end{bmatrix} = \begin{bmatrix} \frac{22.89e^{-0.2s}}{4.572s + 1} & \frac{-11.64e^{-0.4s}}{1.807s + 1} \\ \frac{4.689e^{-0.2s}}{2.174s + 1} & \frac{5.80e^{-0.4s}}{1.801s + 1} \end{bmatrix} \begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix} + \begin{bmatrix} -4.243e^{-0.4s} \\ \frac{3.445s + 1}{1.982s + 1} \\ -0.601e^{-0.4s} \end{bmatrix} [d(s)] \quad (26)$$

The time scales are in hours, so the process dynamic response is quite slow. The two controlled variables are two measurements representing the reactor condition, and the two manipulated variables are the setpoints of two reactor feed flow loops with load disturbance as the purge flow of the reactor.

Notice that in this case the interactions is such that $RGA(\lambda_{ij}) < 1$ thus further detuning is necessary. When

Table 4. PI Tuning Parameters for Reactor Control Problem

	K_{c1}	τ_{i1}	K_{c2}	τ_{i2}
Luyben	0.210	2.26	0.175	4.25
Shen and Yu	0.459	1.50	0.183	4.45
Palmor et al.	0.0978	1.60	0.375	1.60
Loh et al.	0.620	0.60	0.247	1.78
proposed	0.263	1.42	0.163	1.77

Table 5. PI Tuning Parameters for ex 3

	K_{c1}	τ_{i1}	K_{c2}	τ_{i2}	K_{c3}	τ_{i3}
Luyben	1.51	16.4	-0.295	18.0	2.63	6.61
Shen and Yu	2.55	16.5	-0.235	26.1	3.39	7.39
Halevi et al.	1.25	10.5	-0.339	10.5	0.923	10.5
proposed	1.08	4.25	-0.233	3.32	2.78	5.24

the tuning procedure in the above section is followed, the detuning factor for this system is 0.709. The resulting PI tuning parameters together with the tuning parameters calculated using other tuning methods are tabulated in Table 4.

Figures 9–11 show the closed-loop responses for this system under setpoint change of y_1 , setpoint change of y_2 , and load disturbance changes, respectively. Again, the proposed method provides satisfactory closed-loop performance with the minimum identification tests needed.

Various other 2×2 systems have been used to test the adequacy of this proposed method. They are

For $RGA(\lambda_{ij}) > 1$ systems, TS, VL, and WW models in Luyben,¹ and a column model in Taiwo.¹⁷

For $RGA(\lambda_{ij}) < 1$ systems, a process model in Niederlinski¹⁸ with higher order process model.

Our proposed tuning method performs all quite satisfactory.

Example 3. A 3×3 System. A 3×3 example originated in Ogunnaike and Ray¹⁹ and later studied by Luyben¹ and Halevi et al.⁶ will be used here to test

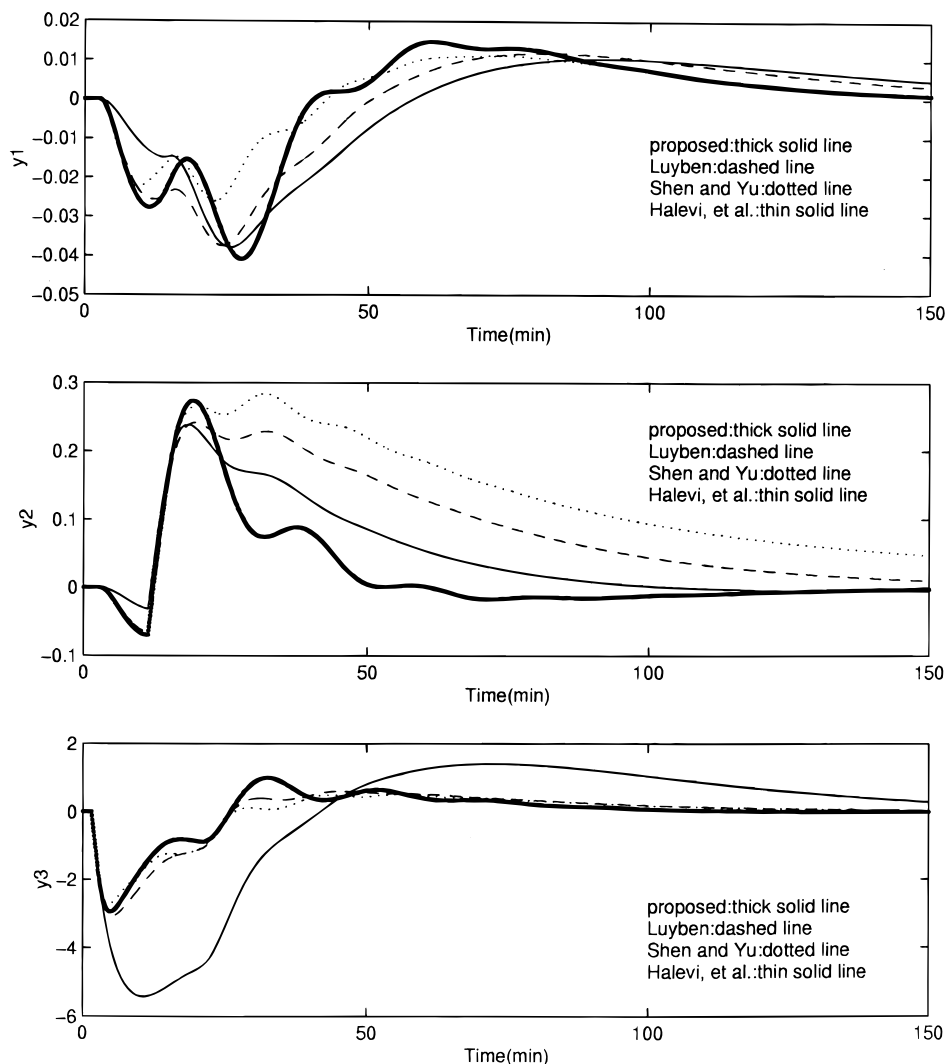


Figure 12. Closed-loop response for example 3 with load change.

the performance of the proposed method for systems with control loops greater than two. The process model is

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} \frac{0.66e^{-2.6s}}{6.7s + 1} & \frac{-0.61e^{-3.5s}}{8.64s + 1} & \frac{-0.0049e^{-s}}{9.06s + 1} \\ \frac{1.11e^{-6.5s}}{3.25s + 1} & \frac{-2.36e^{-3s}}{5s + 1} & \frac{-0.01e^{-1.2s}}{7.09s + 1} \\ \frac{-34.68e^{-9.2s}}{8.15s + 1} & \frac{46.2e^{-9.4s}}{10.9s + 1} & \frac{0.87(11.61s + 1)e^{-s}}{(3.89s + 1)(18.8s + 1)} \end{bmatrix} \times \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} + \begin{bmatrix} \frac{0.14e^{-12s}}{(19.2s + 1)^2} \\ \frac{0.53e^{-10.5s}}{6.9s + 1} \\ \frac{-11.54e^{-0.6s}}{7.01s + 1} \end{bmatrix} [d] \quad (27)$$

This 3 × 3 system has RGA(λ_{ij}) elements all greater than one; thus, no further detuning is necessary. By

application of the bias-relay tests to obtain the model parameters of the three diagonal elements of the process, the PI tuning parameters can be calculated using the procedure outlined in Section 3. The resulting PI tuning parameters together with the tuning parameters reported in the literatures for other tuning methods are tabulated in Table 5.

Figure 12 illustrates the closed-loop response for this system under load disturbance. The closed-loop performance of the proposed method is excellent. Similar good performance can be found for setpoint changes in all three controlled variables.

5. Conclusions

In this paper, a very simple PID tuning method is proposed for multiloop control systems. The tuning method is based on the PID controller implementation with the form of no proportional kick. With this controller form, the interactions between the control loop can be minimized, and the desired closed-loop performance can still be specified by controller synthesis method. Depending on the interaction natures of the multiloop systems, controller tuning based on the diagonal elements of the model or further detuning may be necessary. For systems with RGA(λ_{ij}) < 1, a detuning

factor based on the value of $RGA(\lambda_{ij})$ is proposed. One of the biggest advantage of the proposed tuning method is that because the tuning method only requires dynamic model parameters of the diagonal elements and the process gain information of the off-diagonal elements only n th identification tests (with n as the number of the interacting control loops) are needed to determine the PID tuning parameters as opposed to other tuning methods which require more iterative tests. Because less interference of process operation is always preferable for industrial situations, this tuning method has the potential to be widely applicable in industry. Many simulation tests have confirmed the performance of this proposed tuning method. Its closed-loop servo and load responses are all very satisfactory.

Acknowledgment

This work is supported by National Science Council of the Republic of China under Grant NSC 87-2214-E-011-025.

Appendix: Derivation of PID Tuning Rules

The closed-loop transfer function between controlled variable (y) and setpoint (r) is

$$\frac{y}{r} = \frac{(K_c/\tau_i s) G_p}{1 + \frac{K_c(\tau_i \tau_d s^2 + \tau_i s + 1)}{\tau_i s} G_p} \quad (\text{A.1})$$

For time constant dominant processes, the process model, G_p , can be approximated using Padé approximation as

$$G_p = \frac{R e^{-Ls}}{s} \approx \frac{R(1 - (L/2)s)}{s(1 + (L/2)s)} \quad (\text{A.2})$$

Substituting eq A.2 into A.1 and simplifying, we get

$$\frac{y}{r} = (1 - (L/2)s) \left[\left(\frac{L\tau_i}{2K_c R} - \frac{L\tau_i \tau_d}{2} \right) s^3 + \left(\frac{\tau_i}{K_c R} + \tau_i \tau_d - \frac{L\tau_i}{2} \right) s^2 + \left(\tau_i - \frac{L}{2} \right) s + 1 \right] \quad (\text{A.3})$$

Let us assume our desired closed-loop servo response to be a underdamped system with damping coefficient of 0.707. This corresponds to a closed-loop system with about 5% overshoot. The desired closed-loop servo response is

$$\left(\frac{y}{r} \right)_{\text{desired}} = \frac{e^{-Ls}}{\tau_{cl}^2 s^2 + 1.414\tau_{cl}s + 1} \approx \frac{1}{(\tau_{cl}^2 s^2 + 1.414\tau_{cl}s + 1)} \frac{1 - (L/2)s}{1 + (L/2)s} \quad (\text{A.4})$$

where τ_{cl} is an user-specified closed-loop effective time constant. Equating eqs A.3 and A.4 and doing some algebraic manipulation, we can solve for the PID tuning parameters as

$$K_c = \frac{1.414\tau_{cl} + L}{R \left(\tau_{cl}^2 + 0.707\tau_{cl}L + \frac{L^2}{4} \right)} \quad (\text{A.5})$$

$$\tau_i = 1.414\tau_{cl} + L \quad (\text{A.6})$$

$$\tau_d = \frac{(L^2/4) + 0.707\tau_{cl}L}{1.414\tau_{cl} + L} \quad (\text{A.7})$$

For processes with deadtime greater than $1/5$ of the process time constant, it is better for controller tuning purposes to model the processes as a first-order-plus deadtime model. With the same Padé approximation as

$$G_p = \frac{K_p e^{-Ls}}{\tau s + 1} \approx \frac{K_p}{\tau s + 1} \frac{1 - (L/2)s}{1 + (L/2)s} \quad (\text{A.8})$$

Substituting eq A.8 into A.1 and simplifying, we obtain

$$\frac{y}{r} = (1 - (L/2)s) \left[\left(\frac{L\tau_i \tau}{2K_c K_p} - \frac{L\tau_i \tau_d}{2} \right) s^3 + \left(\frac{\tau_i L}{2K_c K_p} + \frac{\tau_i \tau}{K_c K_p} + \tau_i \tau_d - \frac{L\tau_i}{2} \right) s^2 + \left(\frac{\tau_i}{K_c K_p} + \tau_i - L/2 \right) s + 1 \right] \quad (\text{A.9})$$

Again, equating eqs A.9 and A.4 and doing some algebraic manipulation, we can solve for the PID tuning parameters as

$$K_c = \frac{\tau L + (L^2/4) + 1.414\tau_{cl}\tau - \tau_{cl}^2}{K_p(\tau_{cl}^2 + 0.707\tau_{cl}L + L^2/4)} \quad (\text{A.10})$$

$$\tau_i = \frac{\tau L + (L^2/4) + 1.414\tau_{cl}\tau - \tau_{cl}^2}{\tau + (L/2)} \quad (\text{A.11})$$

$$\tau_d = \frac{0.707\tau_{cl}L + (L^2/4)\tau - \tau_{cl}^2(L/2)}{\tau L + (L^2/4) + 1.414\tau_{cl}\tau - \tau_{cl}^2} \quad (\text{A.12})$$

By selecting τ_{cl} as in Figure 2, the negative terms in eqs A.10–A.12 will not cause any problem in changing the signs of the PID tuning parameters. With the τ_{cl} selection as in Figure 2, combining with eqs A.5–A.7 and A.10–A.12, the final PID tuning rules in Table 2 can be obtained.

Literature Cited

- (1) Luyben, W. L. Simple Method for Tuning SISO Controllers in Multivariable Systems. *Ind. Eng. Chem. Res.* **1986**, *25*, 654.
- (2) Friman, M.; Waller, K. V. Autotuning of Multiloop Control Systems. *Ind. Eng. Chem. Res.* **1994**, *33*, 1708.
- (3) Shen, S.-H.; Yu, C.-C. Use of Relay-Feedback Test for Automatic Tuning of Multivariable Systems. *AIChE J.* **1994**, *40*, 627.
- (4) Loh, A. P.; Hang, C. C.; Quek, C. K.; Vasnani, V. U. Autotuning of Multiloop Proportional-Integral Controllers Using Relay Feedback. *Ind. Eng. Chem. Res.* **1993**, *32*, 1102.
- (5) Palmor, Z. J.; Halevi, Y.; Krasney, N. Automatic Tuning of Decentralized PID Controllers for TITO Processes. *Automatica* **1995**, *31* (7), 1001.
- (6) Halevi, Y.; Palmor, Z. J.; Efrati, T. Automatic Tuning of Decentralized PID Controllers for MIMO Processes. *J. Proc. Cont.* **1997**, *7* (2), 119.
- (7) Rijnsdorp, J. E. Interaction in Two-Variable Control Systems for Distillation. *Automatica* **1965**, *1*, 15.

(8) Skogestad, S.; Morari, M. LV-Control of a High-Purity Distillation Column. *Chem. Eng. Sci.* **1988**, *43*, 33.

(9) Skogestad, S.; Morari, M. Understanding the Dynamic Behavior of Distillation Columns. *Ind. Eng. Chem. Res.* **1988**, *27*, 1848.

(10) Chien, I-L.; Fruehauf, P. S. Consider IMC Tuning to Improve Controller Performance. *Chem. Eng. Prog.* **1990**, *86* (October), 33.

(11) Chien, I-L.; Ogunnaike, B. A. Modeling and Control of a Temperature-Based High-Purity Distillation Column. *Chem. Eng. Commun.* **1997**, *158*, 71.

(12) Bristol, E. On a New Measure of Interaction for Multivariable Process Control. *IEEE Trans. Autom. Control* **1966**, AC-11, 133.

(13) Huang, H.-P.; Wang, G.-B.; Lee, M.-W.; Chien, I-L. Identification of Transfer Function Models from the Relay Feedback Test. *Chem. Eng. Commun.* **1998**, submitted.

(14) Wood, R. K.; Berry, M. W. Terminal Composition Control of Binary Distillation Columns. *Chem. Eng. Sci.* **1973**, *28*, 1707.

(15) Morari, M.; Zafiriou, E. *Robust Process Control*; Prentice-Hall: Englewood Cliffs, New Jersey, 1989.

(16) Ogunnaike, B. A.; Ray, W. H. *Process Dynamics, Modeling, and Control*; Oxford University Press: New York, 1994.

(17) Taiwo, O. Application of the Method of Inequalities to the Multivariable Control of Binary Distillation Columns. *Chem. Eng. Sci.* **1980**, *35*, 847.

(18) Niederlinski, A. A Heuristic Approach to the Design of Linear Multivariable Interacting Control Systems. *Automatica* **1971**, *7*, 691.

(19) Ogunnaike, B. A.; Ray, W. H. Multivariable Controller Design for Linear Systems Having Multiple Time Delays. *AIChE J.* **1979**, *25*, 1043.

Received for review September 21, 1998
Revised manuscript received January 11, 1999
Accepted January 12, 1999

IE980595V