Use of Relay-Feedback Test for Automatic Tuning of Multivariable Systems

Shih-Haur Shen and Cheng-Ching Yu
Dept. of Chemical Engineering, National Taiwan Institute of Technology, Taipei, Taiwan 10672, R.O.C.

The Åström-Hägglund autotuner is extended to multivariable systems. The proposed multivariable automatic tuning performs the identification-design procedure in a sequential manner. Its advantages are discussed, as well as the process characteristics resulted from sequential design. One important feature is that the sequential design often leads to underdamped process characteristics. Given these characteristics, potential problems in the Ziegler-Nichols tuning are discussed and a modification is made. The sequential identification and the modified Ziegler-Nichols controller design method form the basic structure for the multivariable autotuner. The properties of convergence, the implications of tuning sequence, and the indications of undesirable pairing for the autotuner are explored. Consequently, a multivariable autotuning procedure is proposed. Two nonlinear distillation examples and a 3×3 linear system are used to illustrate the effectiveness of the autotuner. Simulation results show that good performance can be obtained with minimal engineering effort and the autotuner works well for very difficult processes in a very simple way.

Introduction

Intelligent control is now becoming common in the literature and control systems as some types of intelligent features begin to appear. From an operating point of view, it is helpful to have a control system which can find the tuning constants by itself, diagnose sensor or actuator failure, overcome process nonlinearity, and so on. Åström and McAvoy (1992) review recent advances. Among all these features, the ability to perform automatic tuning in a multivariable environment is of primary importance in chemical process control. The reason is obvious: most chemical processes are multivariable and the controller tuning often poses problems to plant operators.

Based on the information obtained, the methods for autotuning can be classified into frequency domain approaches and time domain approaches. A typical example of the frequency domain approach is the Åström-Hägglund autotuner (1984). First, a continuous cycling of the controlled variable is generated from a relay-feedback experiment, and the important process information, ultimate gain (K_u) and ultimate frequency (ω_u), can be extracted directly from the experiment. A controller can be designed according to K_u and ω_u (Ziegler-Nichols, 1942). The information obtained from the relay-feedback experiment is exactly the same as that from conventional continuous cycling methods (Seborg et al., 1989; Luyben, 1990). However, an important difference is that the sustained oscillation is generated under a *controlled* situation (for example, the magnitude of oscillation can be controlled) in the relay-feedback test. Applications of Åström-Hägglund autotuner are shown throughout process industries using single station controllers or a distributed control system (DCS) (Hägglund and Åström, 1991). The success of this autotuner is due to the fact that the tuning mechanism is so *simple* that operators understand how it works. Moreover, it also works well in highly nonlinear processes, for example, high purity distillation columns (Luyben, 1987b). Extensions of a relay-feedback system to monitoring and gain scheduling are also made (Chiang and Yu, 1993; Lin and Yu, 1993).

The time domain approaches, on the other hand, generate responses from step or pulse tests. The characteristics of the process response are then utilized to back-calculate the parameters of an assumed process model (Yuwana and Seborg, 1982; Chen, 1989; Hinde and Cooper, 1991). The pattern recognition controller (Bristol, 1977; Cao and McAvoy, 1990) is a typical

Correspondence concerning this article should be addressed to C.-C. Yu.

example. Since a step or pulse test is involved in this approach, it may have difficulties for highly nonlinear systems, for example, high purity distillation columns (Fuentes and Luyben, 1983).

Despite recent advances in the automatic tuning methods, application of the autotuners are mostly limited to single-input-single-output (SISO) systems. Koivo and coworkers (Penttiner and Koivo, 1980; Koivo and Pohjolainer, 1985) use step responses to find the state space model for a $n \times n$ (n input and n output) multivariable system. PI controllers are designed according to that linear model. Conceptually, this is similar to a multivariable version of the process reaction curve method (Seborg et al., 1989). Since step responses are employed in the identification phase, it can have difficulties for highly nonlinear processes. Cao and McAvoy (1990) evaluate and analyze the performance of the pattern recognition controller (EXACT) in a multivariable system. Hsu et al. (1992) attempt to extend the Åström-Hägglund autotuner to multivariable systems when I-only (integral only) controllers are used. Furthermore, the method of Hsu et al. (1992) requires that the steady-state gain matrix should be known *a priori* or can be measured. Obviously, this requirement limits the applicability of the autotuner in an operating environment.

The purpose of this work is to extend the Åström-Hägglund autotuner to unknown multivariable systems. In this work, decentralized PI controllers are used, and square and open-loop stable multivariable systems are also assumed. This article discusses the concept of the multivariable autotuner and the underlying theory for the sequential design and sequential identification employed in the autotuning procedure. Potential problems associated with Ziegler-Nichols (Z-N) type of tuning methods are raised and modifications are also made. Properties of the proposed autotuner are discussed and nonlinear distillation examples, as well as a 3×3 linear system are used to illustrate the effectiveness of the proposed method followed by the conclusion.

Concept

SISO autotuning

An automatic tuning procedure can be divided into two stages: the identification phase and the controller design phase. In the identification phase, the Åström-Hägglund autotuner is based on the observation that a feedback system in which the output (y) lags behind the input (u) by $-\pi$ rad may oscillate with a period P_u. This is a well-known observation; however, the oscillation is carried out in a new manner. To generate the sustained oscillation, a relay-feedback test is performed (Figure 1A). Initially, the input (u) is increased by h ($u = u^{ss} + h$ where u^{ss} is the steady-state value of u). As soon as the output is moving upward, the input is switched to the lower position ($u = u^{ss} - h$), as shown in Figure 1B. This procedure is repeated until the cycling reaches a steady form. From the relay-feedback test, the familiar ultimate gain (K_u) and ultimate frequency (ω_u) are readily available. They can be approximated as:

$$K_u = \frac{4h}{\pi a} \tag{1}$$

$$\omega_u = \frac{2\pi}{P_u} \tag{2}$$

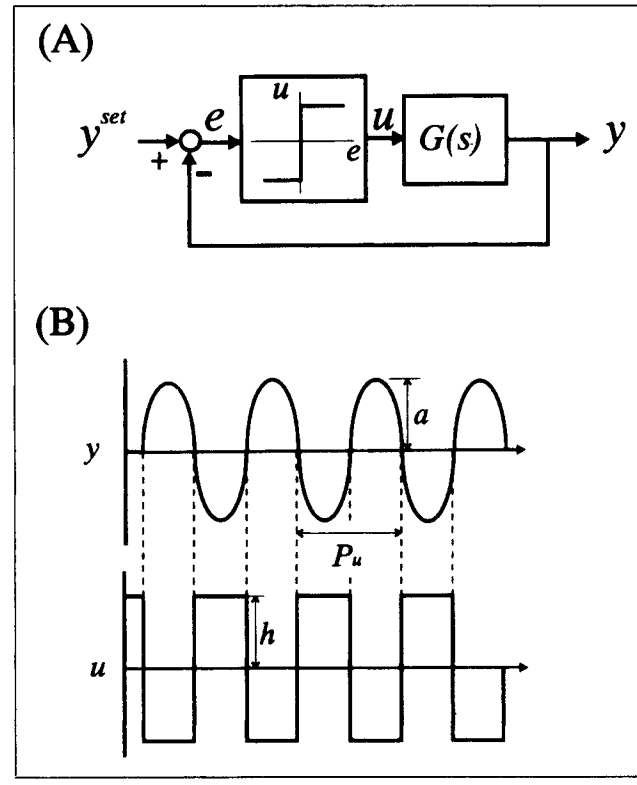

Figure 1. Relay-feedback system: (A) block diagram and (B) responses.

where a is the amplitude of the oscillation and P_u is the period (Figure 1B). Following the identification phase, the controller can be designed, from K_u and ω_u. The classical Z-N (Ziegler-Nichols) method probably is the simplest and the most popular choice. The controller gain (K_c) and reset time (τ_I) of a PI controller can be found with the simple calculation:

$$K_c = \frac{K_u}{2.2} \tag{3}$$

$$\tau_I = \frac{P_u}{1.2} \tag{4}$$

It is also possible to find the tuning constants using any other design methods, for example, gain or phase margin specification (Åström-Hägglund, 1984), dominant pole design (Åström-Hägglund, 1988) and M-circle criterion (Schei, 1992). Notice that all these methods design the controller based on a *single* point on the Nyquist curve of $G(j\omega)$.

MIMO autotuning

Luyben (1987b) proposes the use of relay-feedback tests (ATV method) for the *identification* of transfer function matrix $G(s)$ in a multivariable system. Consider a $n \times n$ multivariable system $G(s)$ with the entry $g_{ij}(s)$. A relay-feedback test is performed and each individual element in a column of the transfer function matrix can be found by fitting the corresponding point on the Nyquist curve to an assumed model. This procedure is repeated (n times) until all n^2 transfer func-

tions in $G(s)$ are found. Once $G(s)$ is identified, the familiar *independent* (as opposed to sequential) design methods (Chiu and Arkun, 1992), for example, BLT method (Luyben, 1986) and μ (structured singular value) tuning criterion (Skogestad and Landström, 1990), can be employed to find the tuning constants. By independent design, we mean all controllers are designed independently (once the specification is met).

Notice that these are *off-line* controller design methods, despite the fact they can be automated. However, the identification-design procedure along this line can be computationally extensive (n^2 transfer functions have to be fitted followed by an iterative searching for controller parameters). Furthermore, the columnwise identification procedure (identifying a column of transfer functions in $G(s)$ by perturbing a manipulated input) can lead to inconsistency in nonlinear multivariable processes (Luyben, 1987a; Chang and Yu, 1992). This is termed as *independent* identification hereafter, and it will be discussed in a greater detail in the next section.

For an autotuner, an efficient identification-design procedure has to be devised. From the efficiency point of view, an important question to ask is: do we really need to find out *all* (n^2) individual transfer functions ($g_{ij}s$) for the controlled design? The sequential design (Mayne, 1973, 1979; Bhalodia and Weber, 1979; Bernstein, 1987; O'Reilly and Leithead, 1991; Leithead and O'Reilly, 1991; Chiu and Arkun, 1992) provides an attractive alternative in MIMO autotuning. By sequential design we mean each controller (in a multivariable system) is designed in sequence. In other words, a MIMO process is treated as a sequence of SISO systems.

Based on the concept of sequential design, a simple method is proposed for MIMO autotuning. For the purpose of illustration, a 2×2 system is used (extension to a $n\times n$ system is straightforward). Consider a 2×2 system with a known pairing (y_1-u_1 and y_2-u_2) under decentralized control (Figure 2). Initially, a relay is placed between y_1 and u_1, while loop 2 is on *manual* (Figure 2A). Following the relay-feedback test, a controller can be designed from the ultimate gain and ultimate frequency. The next step is to perform relay-feedback test between y_2 and u_2 while loop 1 is on *automatic* (Figure 2B). A controller can also be designed for loop 2 following the relay-feedback test. Once the controller on the loop 2 is put on automatic, another relay-feedback experiment is performed between y_1 and u_1 (Figure 2C). Generally, a new set of tuning constants is found for the controller in loop 1. This procedure is repeated until the controller parameters converge. Typically, the controller parameters converge in $3\sim4$ relay-feedback tests for 2×2 systems. Notice that the proposed MIMO autotuning concept repeats the "identification-design" procedure on SISO transfer functions.

This approach has several advantages. First, it makes the problem *simple*. The reason is: the proposed approach treats the MIMO system as a sequence of SISO systems which the relay-feedback system is proven useful and reliable. Second, it operates in an *efficient* manner. The autotuner identifies the transfer functions needed for the controller design, instead of identifying and fitting all n^2 $g_{ij}s$ in the independent design. Third, in terms of identification, it is a more accurate approach. It can be understood qualitatively that the independent identification finds linear $g_{ij}s$ at different operating points (for example, changing one input by maintaining the rest of the inputs constant) and the controller design is based on some

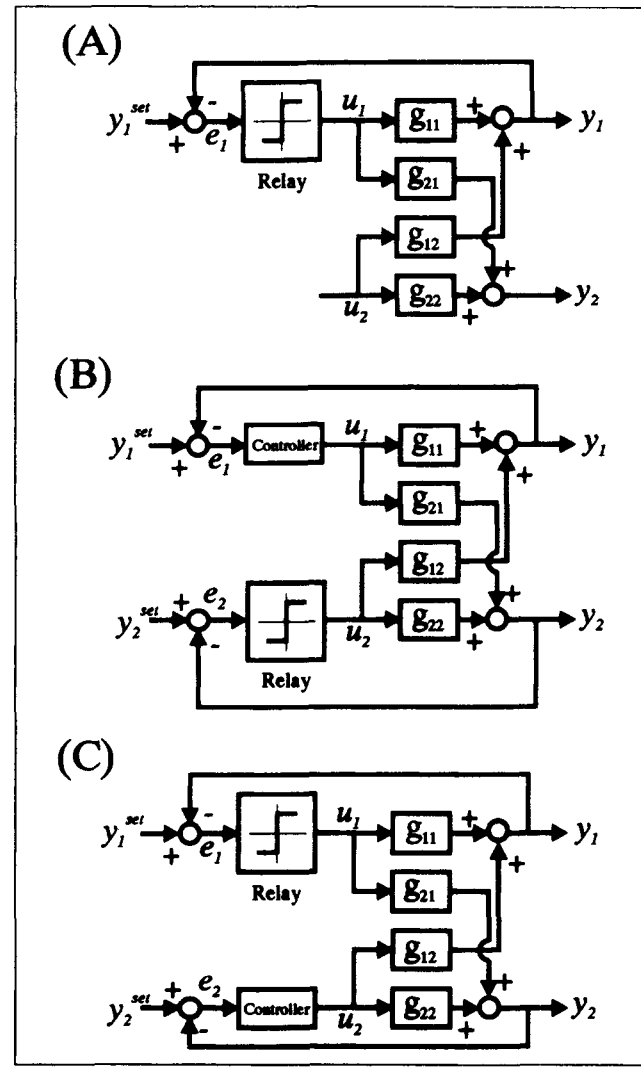

Figure 2. Sequential tuning procedure for a 2×2 system.

combinations of $g_{ij}s$ while the sequential identification finds the transfer function (can be a combination of $g_{ij}s$) it needs for controller design at a single operating point. From the above point of view, it is not necessary to identify all individual transfer $g_{ij}(s)$ for the purpose of controllers tuning.

Theory

Since the multivariable autotuning is based on the concept of sequential identification-design procedure, the fundamental theory of sequential design is addressed. Process characteristics from the sequential design are also explored. More importantly, the idea of sequential identification is proposed and the advantages for identification in a sequential manner are also discussed.

Sequential design

The concept of sequential design was proposed in the early 1970s (Mayne, 1973, 1979; Bhalodia and Weber, 1979; Bernstein, 1987; O'Reilly and Leithead, 1991; Leithead and O'Reilly, 1991; Chiu and Arkun, 1992). The idea of sequential design

is a straightforward one: to treat a $n \times n$ multivariable design problem as a sequence of n SISO design problems. Therefore, the familiar design methods can be employed. Unlike the familiar multivariable design methodology in chemical process control, the classical sequential design methods (Mayne, 1973, 1979; O'Reilly and Leithead, 1991; Leithead and O'Reilly, 1991) address the problems of variable pairing and the controller design at the same time which make the design procedure complicated. Bhalodia and Weber (1979) and Chiu and Arkun (1992), on the other hand, assume that controller structure (variable pairing) is determined *a priori* and controller design is carried out sequentially. In this work, only the problem of controller design is addressed and the variable pairing problems are discussed elsewhere (Grosdidier et al., 1985; Yu and Fan, 1990). Notice that, in sequential design, the process transfer function matrix $G(s)$ is, generally, assumed to be known.

Consider a $n \times n$ multivariable system with decentralized PI controllers (Figure 3A). In the autotuning procedure, initially, the relationship between y_1 and u_1 is simply (Figure 2A):

$$\left(\frac{y_1}{u_1}\right)_{OL} = g_{11} \tag{5}$$

where the subscript OL stands for open-loop. When the loop is closed sequentially, the closed-loop relationship between y_1 and u_1 becomes a bit more complicated (Figure 2C). The following notations will be used:

$$G(s) = \begin{matrix} 1 & n-1 \end{matrix} \begin{pmatrix} g_{11} & \vdots & g_{12} \\ \cdots & \vdots & \cdots \\ g_{21} & \vdots & g_{22} \end{pmatrix} \begin{matrix} 1 \\ n-1 \end{matrix}$$

$$K(s) = \begin{pmatrix} k_1 & \vdots & 0 \\ \cdots & \vdots & \cdots \\ 0 & \vdots & k_2 \end{pmatrix}$$

$$y = \begin{pmatrix} y_1 \\ \cdots \\ y_2 \end{pmatrix} \begin{matrix} 1 \\ n-1 \end{matrix}$$

$$u = \begin{pmatrix} u_1 \\ \cdots \\ u_2 \end{pmatrix}$$

With this partitioning (Figure 3B), the closed-loop relationship between y_1 and u_1 becomes:

$$g_{11,CL}(s) = \left(\frac{y_1}{u_1}\right)_{CL} = g_{11}\left[1 - \frac{g_{12}g_{22}^{-1}h_2 g_{21}}{g_{11}}\right] \tag{6}$$

where

$$h_2 = g_{22}k_2[I + g_{22}k_2]^{-1} \tag{7}$$

which is the complementary sensitivity function for the rest

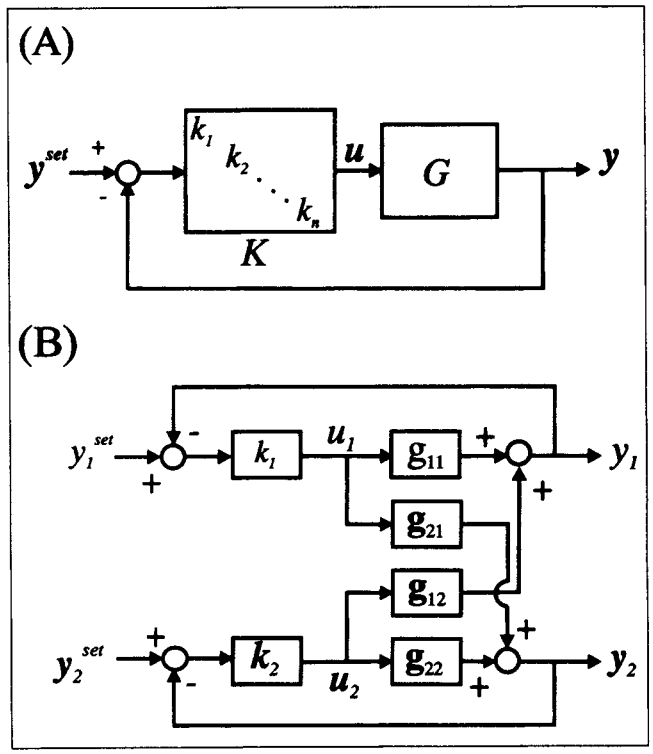

Figure 3. Block diagram of a $n \times n$ multivariable system with decentralized controllers.

loops and the subscript CL stands for closed-loop. When y_2 is under perfect control ($h_2 = I_{(n-1)\times(n-1)}$), $g_{11,CL}(s)$ is reduced to a simpler form.

$$g_{11,CL}(s) = \left(\frac{y_1}{u_1}\right)_{CL} = g_{11}\left[1 - \left(1 - \frac{1}{\lambda_{11}(s)}\right)\right] \tag{8}$$

where $\lambda_{11}(s)$ is exactly the (1,1) entry of relative gain array (RGA) (Bristol, 1966). When y_2 is without control ($h_2 = 0$), then $g_{11,CL}$ becomes:

$$g_{11,CL}(s) = g_{11}(s) \tag{9}$$

Therefore, the second factor in the RHS of Eq. 6 gives the measure of closed-loop interaction throughout the frequency range of interest. In terms of sequential design, this implies, eventually, the SISO system we are dealing with is of the form of Eq. 6

Let's take a 2×2 system as an example. The closed-loop relationship between y_1 and u_1 becomes:

$$g_{11,CL}(s) = g_{11}\left(1 - \frac{g_{12}g_{21}}{g_{11}g_{22}} \cdot h_2\right)$$

$$= g_{11}(1 - \kappa h_2) \tag{10}$$

where κ is the Rijnsdorp interaction measure (Rijnsdorp, 1965) and $h_2 = g_{22}k_2/(1 + g_{22}k_2)$. Similarly, we have:

$$g_{22,CL} = g_{22}(1 - \kappa h_1) \tag{11}$$

Here, κ takes the system interaction into account. Therefore,

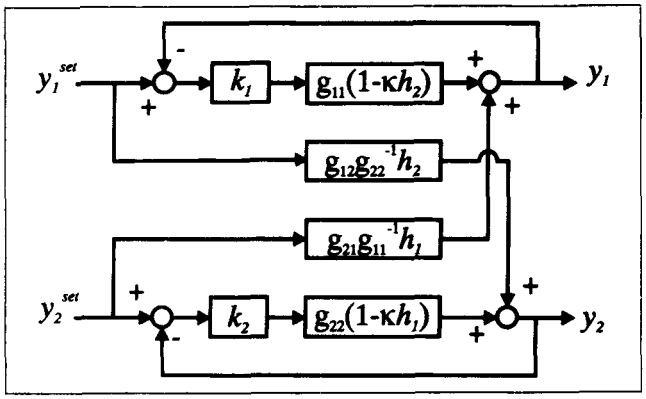

Figure 4. Block diagram of sequential design.

the concept of sequential design is illustrated in Figure 4 with the redrawn block diagram. Figure 4 shows that both controller k_1 and k_2 can be designed individually in a sequential manner (for example, the controller k_2 is embedded in $g_{11,CL}$). Despite the fact that this 2×2 system can be treated as two SISO systems, the characteristics of $g_{ii,CL}$ are quite different from the familiar process transfer functions, for example, first-order plus dead time system.

It can be shown that the roots of closed-loop characteristic equation:

$$\det(I+GK)=0 \tag{12}$$

are exactly the same as these for each loop in Figure 4.

$$1+g_{11}(1-\kappa h_2)k_1=0$$

or (13)

$$1+g_{22}(1-\kappa h_1)k_2=0$$

The poles and zeros of the transfer functions, for example, $g_{11,CL}$ and $g_{22,CL}$, play an important role in the controller design. Following the steps of multivariable autotuning (Figure 2), initially, the poles and zeros are exactly the same as that of g_{11}. However, when k_1 is designed and loop 1 is closed, the process transfer function in loop 2 becomes $g_{22}(1-\kappa h_1)$ (Figure 2B or 4). The poles of $g_{22,CL}$ are the poles of g_{22}, g_{12}, g_{21} and h_1 (O'Reilly and Leithead, 1991), as shown in Table 1. Once k_2 is designed and loop 2 is closed, we go back to loop 1, as shown in Figure 2C. At this point, the transfer function becomes $g_{11}(1-\kappa h_2)$ and the poles are the same as that of g_{11}, g_{12}, g_{21} and h_2. The zeros for $g_{11,CL}$ and $g_{22,CL}$ are zeros of $(1-\kappa h_2)$

Table 1. Poles and Zeros in Sequential Design for a 2×2 System

	Poles of	Zeros
Step 1 (Loop 1)	g_{11}	Zeros of g_{11}
Step 2 (Loop 2)	g_{22}, g_{12}, g_{21}, h_1	Zeros of $1-\kappa h_1$
Step 3 (Loop 1)	g_{11}, g_{12}, g_{21}, h_2	Zeros of $1-\kappa h_2$

and $(1-\kappa h_1)$, respectively. From the pole-zero configuration in sequential design (Table 1), the behavior of $g_{ii,CL}$ can be investigated.

Process characteristics

Consider the autotuning steps in Figure 2. Initially, controller 1, k_1, is designed based on $g_{11}(s)$ using any possible SISO tuning methods. Typically, in chemical processes, $g_{ii}(s)$ is a first-order plus dead time type of process transfer function, for example, $g_{ii}(s)=K_p e^{-\theta s}/(\tau_p s+1)$. Most SISO tuning methods result in a complementary sensitivity function (h_1) with a resonant peak ($L_{c,max}>0$). For example, if the Z-N method is used, the complementary sensitivity function h_1 shows an underdamped behavior. Figure 5 shows the damping coefficient for the poles of h_1 when the Z-N tuning is applied to first-order plus dead time systems with a range of θ/τ_p values. Notice that first-order Padé approximation is applied to find the poles of h_1. The damping coefficients fall between 0.4 to 0.5 for a range of θ/τ_p (0.001 ~ 1), a rather underdamped behavior. Figure 6 shows $L_{c,max}$ for h_1 when Z-N tuning rule is applied to the same systems. Again, underdamped behavior is observed.

The next step is to design k_2 when loop 1 is closed. As shown in Table 1, the poles of $g_{22,CL}$ are the poles of h_1, g_{22}, g_{12} and g_{21}. Therefore, $g_{22,CL}$ has a pair of *underdamped* poles (from the poles of h_1). This is a rather unusual situation, since most SISO tuning methods deal with overdamped transfer function (for example, first-order plus dead time system). Actually, chemical processes rarely show underdamped open-loop responses (for example, considering separators and reactors). Here, the underdamped characteristics are the result of sequential design for multivariable systems.

Since the multivariable system is treated as a series of SISO systems (Figure 4), a form of process transfer function for

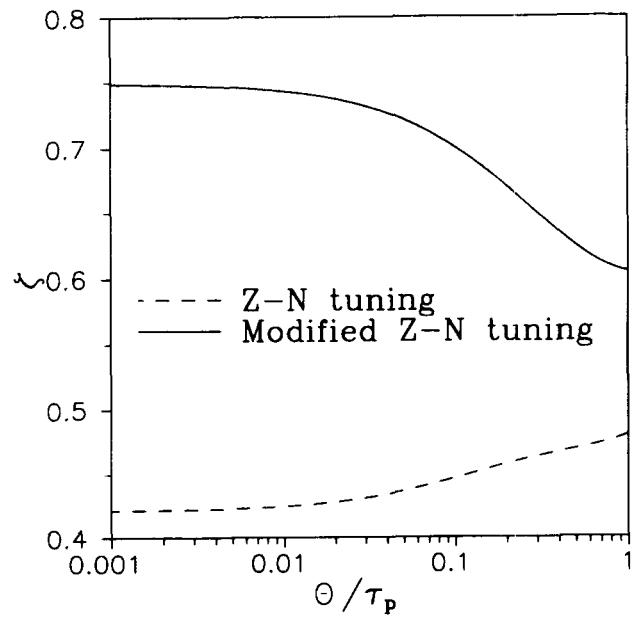

Figure 5. Damping coefficient for first-order plus dead time system with original and modified Z-N methods.

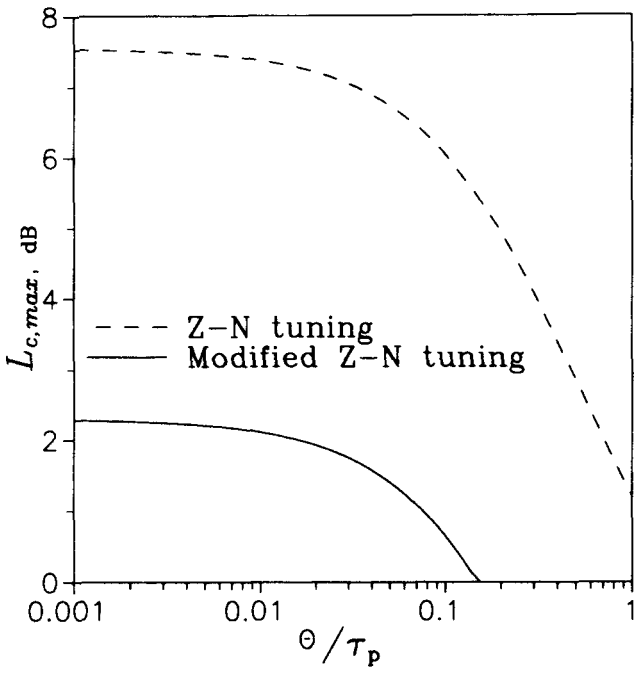

Figure 6. $L_{c,max}$ for first-order plus dead time system with original and modified Z-N methods.

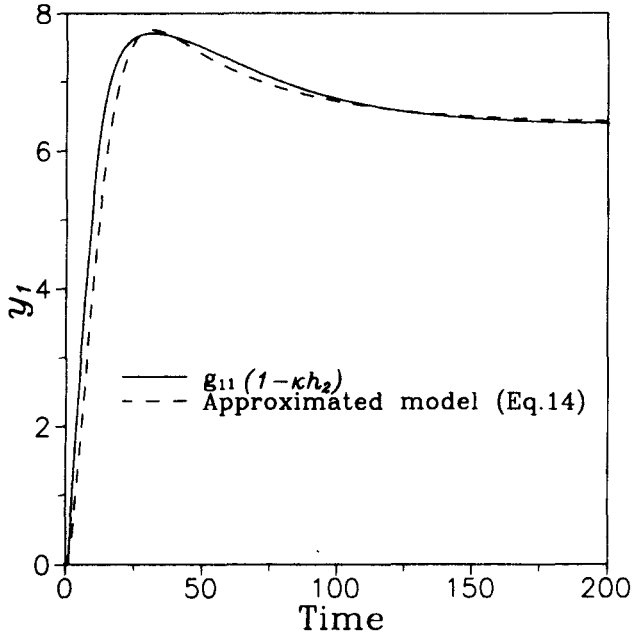

Figure 7. Step responses of the original process $g_{11}(1-\kappa h_2)$ and the approximated model (Eq. 14).

$g_{ii,CL}$ is helpful for the purpose of analyses. In addition to the underdamped poles, the pole of g_{ii} is also the pole of $g_{ii,CL}$. Therefore, an approximate transfer function is used:

$$g_{ii,CL}(s)=\frac{K_p}{(\tau_p^2 s^2+2\tau_p\zeta s+1)}\left(\frac{\tau_{p2}s+1}{\tau_{p1}s+1}\right)e^{-\theta s} \qquad (14)$$

This is a rather unusual structure for a typical process transfer function. However, it gives a description of the mix of underdamped and overdamped behavior resulting from sequential design. A 2×2 distillation column example is used to illustrate the appropriateness of Eq. 14.

Example 1: WB Column (Wood and Berry, 1973). Consider the transfer function matrix:

$$\begin{pmatrix}y_1\\y_2\end{pmatrix}=G(s)\begin{pmatrix}u_1\\u_2\end{pmatrix}=\begin{pmatrix}\dfrac{12.8e^{-s}}{16.7s+1} & \dfrac{-18.9e^{-3s}}{21s+1}\\[2ex] \dfrac{6.6e^{-7s}}{10.9s+1} & \dfrac{-19.4e^{-3s}}{14.4s+1}\end{pmatrix}\begin{pmatrix}u_1\\u_2\end{pmatrix} \qquad (15)$$

with the tuning constants for PI controllers: $K_{c1}=0.54$, $K_{c2}=-0.072$ and $\tau_{I1}=7.92$, $\tau_{I2}=26.7$. The underdamped step response of loop 1 ($g_{11,CL}=g_{11}(1-\kappa h_2)$) is shown in Figure 7. The step responses data are fitted to Eq. 14. The results of least-square regression give:

$$\hat{g}_{11,CL}(s)=\frac{6.4}{42.25s^2+11.7s+1}\left(\frac{44s+1}{60s+1}\right)e^{-s}$$

Figure 7 compares the setup responses of the original process and the approximated model ($\hat{g}_{ii,CL}$). Good approximation can be obtained with Eq. 14.

Another characteristic of $g_{ii,CL}(s)$ comes from the zeros. Table 1 shows that the zeros of $g_{22,CL}$ is that of $(1-\kappa h_1)$. Consider a case of $\kappa(0)>1$ (that is, the system is paired with negative RGA in the diagonal, $\lambda_{ii}<0$). It then becomes obvious that RHP zero *can* occur. For example, if we have:

$$h_1(s)=\frac{1}{s+1}$$

and

$$\kappa(s)=5$$

the zero of $g_{22,CL}$ is 4 (a RHP zero). However, if:

$$\kappa(s)=5(s+1)^2$$

then the zero becomes $-4/5$ (a LHP zero). This confirms the finding that pairing with negative RGA does not necessarily result in inverse responses (steady-state information is not sufficient to decide) (Grosdidier et al., 1985). Nonetheless, systems with $\kappa(0)>1$ give a different sign in the controller gain from the open-loop point of view.

Sequential identification

It is well understood that system identification plays an important role for the success for an autotuner. Traditionally, identification of MIMO systems is carried out by manipulating the inputs (u_is) *independently*. That is: the first column of the transfer function matrices (g_{ij}, $j=1, \ldots, n$) are obtained for a change in u_1 while the rest of the inputs (u_j, $j\neq 1$) are kept constant. Figure 8A illustrates the signal flow in the independent identification. However, difficulties arise for the identification of nonlinear multivariable processes (Luyben, 1987b; Chang and Yu, 1992). Despite the fact that the errors for each

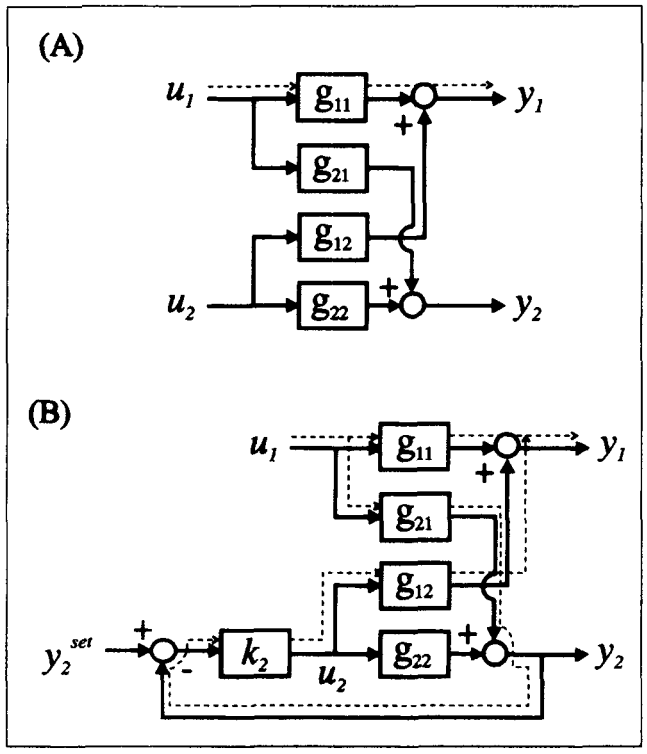

Figure 8. Signal flow in: (A) independent identification and (B) sequential identification.

individual transfer function ($g_{ij}(s)$) are in an acceptable level, the identified transfer function matrices simply fail to describe fundamental process characteristics. For example, Luyben (1987a) shows that in order to find the correct RGA (in the sign), the changes made in the manipulated input (u_i) are so small (0.05% changes) that the calculations have to be carried out with double precision on CDC Cyber 850 computer (not to mention how to implement it in an operating environment).

In their pioneering work, Häggblom and Waller (1988) point out the problem: consistency relations are not met for individual transfer functions. In a series of papers, Waller and coworkers utilize "external material balances" (consistency relations) to find the transformations between control structures, reconcile process models and design controllers for disturbance rejection, and so on. Häggblom and Waller (1992) give a recent summary. Notice that, in their work, independent identification is performed (or assumed) then the consistency relations are enforced. The goal for all the reconciliation is obvious: the elements in a process transfer function matrix should follow some sort of consistency relations (for example, satisfying material balances). A new approach is proposed to achieve this goal by *modifying* the identification procedure.

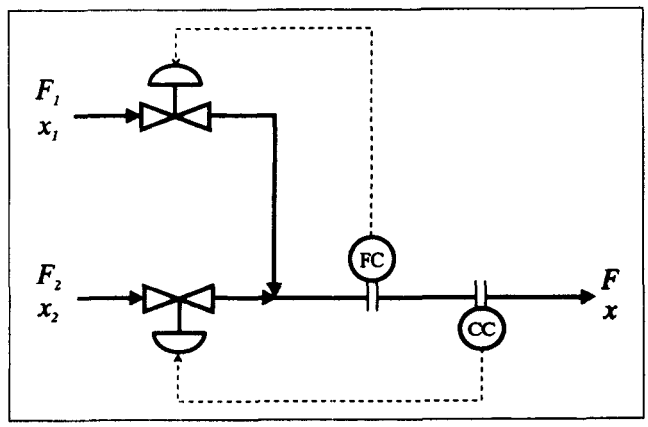

Figure 9. Blending system.

In designing controllers for a multivariable system, the actual transfer function we need is, generally, a combination of $g_{ij}s$. For example, in the sequential design for a 2×2 system the actual transfer function we need is:

$$g_{ii,\mathrm{CL}}(s) = g_{ii}\left(1-\frac{g_{12}g_{21}}{g_{11}g_{22}}\cdot h_j\right),\ i=1,2 \quad \text{and} \quad j\neq i \tag{16}$$

If these $g_{ij}s$ come directly from independent identification (without checking consistency relations), the design can be erroneous. A simple way to meet consistency relations is performing the identification in a sequential manner: sequential identification (Figure 8B). Figure 2 illustrates the procedure of sequential identification when the relay-feedback test is employed.

The advantage of sequential identification is shown in the following example and comparisons are made between independent and sequential identifications.

Example 2: Blending System. Consider a simple 2×2 blending system (Figure 9). The control objective is to maintain flow rate in the outlet stream (F) with the first stream (F_1) and the composition is controlled by changing the second stream (F_2). Material balances describing the blending system are:

$$F = F_1 + F_2 \tag{17}$$

$$xF = x_1F_1 + x_2F_2 \tag{18}$$

Linearizing Eq. 18, the process transfer function $G(s)$ describing this nonlinear system becomes:

$$\begin{pmatrix}\Delta F\\ \Delta x\end{pmatrix} = \begin{pmatrix}1 & 1\\ \dfrac{(\bar{x}_1-\bar{x})}{\bar{F}} & \dfrac{(\bar{x}_2-\bar{x})}{\bar{F}}\end{pmatrix}\begin{pmatrix}\Delta F_1\\ \Delta F_2\end{pmatrix} \tag{19}$$

The nominal steady-state conditions are: $\bar{x}=0.78$, $\bar{F}=20$, $\bar{x}_1=0.9$, $\bar{F}_1=16$, $\bar{x}_2=0.3$, $\bar{F}_2=4$. For independent identification, F_1 and F_2 are changed by a factor of 50%, respectively. Notice that the results from the step changes in F_1 and F_2 fail to satisfy the component material balance (the consistency relation):

$$\bar{x}\Delta F + \Delta x\bar{F} = \bar{x}_1\Delta F_1 + \bar{x}_2\Delta F_2 \tag{20}$$

Appendix A gives the details in checking the consistency relations. Table 2 gives the values of $g_{ij}s$ from independent identification. Results show that g_{21} and g_{22} deviate from the true value by -28.3% and -9.2%, respectively. Obviously, the errors depend on the magnitude of the step changes. Furthermore, the resultant $g_{ii,\mathrm{CL}}s$ are also quite different from the true values (Table 2). On the other hand, sequential identification

Table 2. Estimated Process Transfer Function for Different Identification Approaches

	g_{11}	g_{21}	g_{12}	g_{22}	$g_{11,CL}$	$g_{22,CL}$
True values	1	0.0060	1	−0.0250	1.250	−0.030
Independent Identification	1	0.0043	1	−0.0218	1.197	−0.026
Sequential Identification	-	-	-	-	1.250	−0.030

(Figure 8B) can find $g_{ii,CL}$ directly. Step changes of 50% are made on u_1 and u_2, sequentially, while the other loop is closed (Figure 8). Notice that sequential design finds $g_{ii,CL}$s directly (bypassing g_{ij}s), as shown in Table 2. The $g_{ii,CL}$s found are exactly the same as the true values (Table 2).

This example clearly shows the advantages of the proposed identification approach for nonlinear multivariable systems. The sequential identification finds the *essential* element, $g_{ii,CL}$, for controller design. In doing this, the consistency relations are achieved *internally*.

Controller Tuning

Potential problem in Z-N tuning

Z-N method (Ziegler and Nichols, 1942) is a very popular method in tuning PID type of controllers for the reasons of simplicity and experimental nature (an experimental procedure comes with the tuning rule). However, most studies of Z-N method deal with *overdamped systems* (Seborg et al., 1989; Luyben, 1990; Åström et al., 1992). The stability problem may arise when one tries to tune underdamped system with Z-N method. Tan and Weber (1985) explore stability problems associated with Z-N tuning for third-order systems. As pointed out earlier, the sequential design may produce an underdamped system since the poles of h_j are also the poles of $g_{ii,CL}(i \neq j)$, and typical process transfer function is given in Eq. 14. Let us take transfer function of the structure of Eq. 14 to illustrate the stability problem in the Z-N method. Consider the following underdamped system with a damping coefficient of 0.6.

$$g(s) = \frac{1}{25s^2 + 6s + 1}\left(\frac{10s+1}{5s+1}\right)e^{-s}$$

Based on the Z-N method the PI controller settings are: $K_c = 19.39$, $\tau_I = 2.84$. The Nyquist plot of GK shows that the closed-loop system is unstable (Figure 10). From Figure 10, it is clear that for a underdamped system ($\zeta = 0.6$ in this example), the Z-N method may give an unstable closed-loop system. Notice that the results of Figure 5 show that, for the most common type of transfer functions (first-order plus dead time), the Z-N method produces underdamped poles (poles of h) with the damping coefficient ranging from 0.4 to 0.5. Furthermore, in the sequential design, the underdamped transfer function, that is, $g_{ii,CL}$, has to be tuned *again* with the simple Z-N method. This will lead to an even more underdamped closed-loop system. Therefore, modifications have to be made to avoid the snowball effects of underdamped poles.

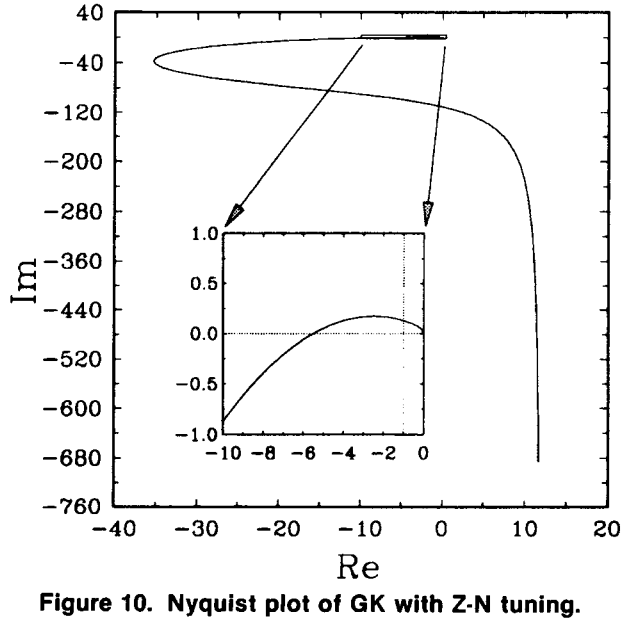

Figure 10. Nyquist plot of GK with Z-N tuning.

Modified Z-N method

It should be emphasized that any familiar SISO tuning methods, for example, gain margin, phase margin, $L_{c,max}$ criterion, can be applied for the PI controller design. However, based on the relay-feedback type of identification, the Z-N type of method is a natural choice (since K_u and ω_u are available). It is clear that any modification should make the tuning constants more conservative. The detuning procedure is followed by the spirit of BLT (Luyben, 1986). That is: a single detuning factor is employed to find appropriate constants:

$$K_c = \frac{K_{c,ZN}}{f}$$

$$\tau_I = \tau_{I,ZN} \cdot f$$

After a number of tests on linear distillation column models (Luyben, 1986), 2×2 systems with first-order plus dead time transfer functions (Marino-Galarraga et al., 1987) and nonlinear distillation examples, a detuning factor $f \approx 2.5$ is proposed. Justifications for the proposed tuning rule will be given shortly. The modified Z-N method for PI controller becomes:

$$K_c = \frac{K_u}{3} \tag{21}$$

$$\tau_I = \frac{P_u}{0.5} \tag{22}$$

The original Z-N PI tuning rule moves the crossover point $(-1/K_u, \omega_u)$ corresponding to the ultimate frequency (ω_u) in the G-plane to $(-1/2.2, 0.087)$ with the same frequency (ω_u) in the GK-plane. In the modified method a more conservative measure is taken and the point corresponding to ω_u is moved to $(-1/3, 0.0265)$ in the GK-plane. Since the damping coefficient of $g_{22,CL}$ (Eq. 16) comes from h_1, we are interested in the damping coefficient or $L_{c,max}$ of the proposed method when applied to typical g_{ii}s. Figure 5 shows that damping coefficient of the modified Z-N method for the transfer functions of $e^{-\theta s}/$

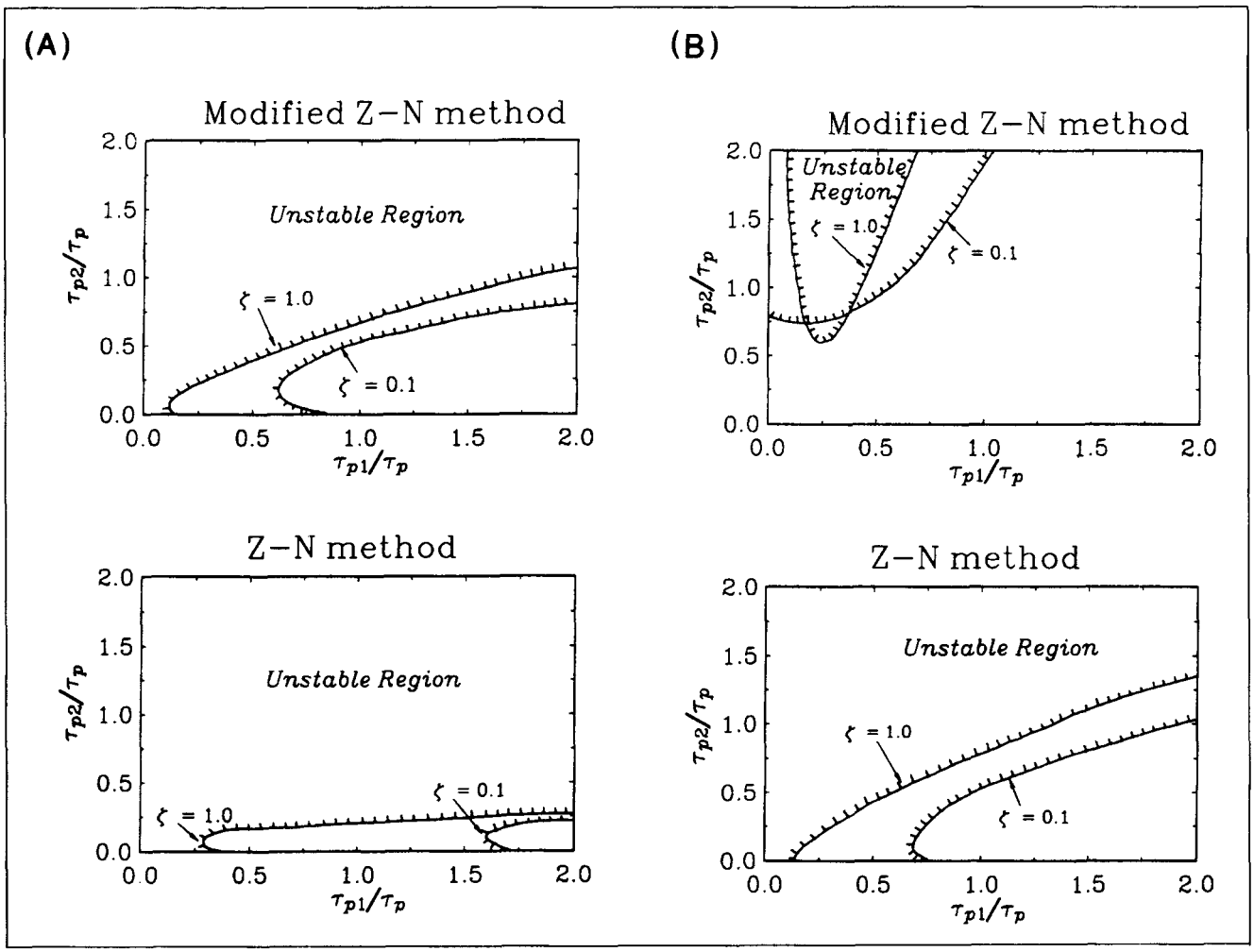

Figure 11. Contour plots of stability regions for different tuning methods with: (A) $\theta = 0.1$ and (B) $\theta = 1$.

$(\tau_p s + 1)$ type. The results (Figure 5) show that the proposed method is less underdamped (with ζ greater than 0.6 for a range of (θ/τ_p)s). The $L_{c,\max}$ plots in Figure 6 also show the same trend. This implied that for the ranges of parameters $(0.001 \le \theta/\tau \le 1)$ studied, the damping coefficient in Eq. 16 is greater than 0.6.

Stability is an important concern for any tuning method. However, tradeoffs between performance and stability region have to be made. As pointed out by Tan and Weber (1985), unstable region can always be found for different values of damping coefficient for third-order processes. Regions of instability are investigated for the transfer function of the form:

$$G(s) = \frac{K_p}{\tau_p^2 s^2 + 2\tau_p \zeta s + 1}\left(\frac{\tau_{p2} s + 1}{\tau_{p1} s + 1}\right)\exp(-\theta s) \qquad (23)$$

The following parametric spaces are studied: $K_p = 1$, $\tau_p = 5$, $\tau_{p2} = 0 \sim 10$, $\tau_{p2} = 0 \sim 10$, $\zeta = 0.1 \sim 1$ and $\theta/\tau_p = 0.02 \sim 0.2$. Figures 11A and 11B show that instability regions exist for both the original and modified Z-N method. However, the modified Z-N method reduces the instability region significantly. Figure 11 reveals that an instability region often occurs in the region when $\tau_{p2} > \tau_{p1}$. The reason is the larger lead time constant in Eq. 23 results in a resonant peak in $(g(j\omega))$ which can be viewed as an enhancement of the underdamped behavior. Obviously, the instability can be eliminated from Figure 11 by detuning the controller further (using a much larger f). However, performance will deteriorate.

Probably the most important evaluation is to test these methods in a sequential design environment. A number of 2×2 systems are studied. Consider 2×2 systems, $G(s)$, with first-order plus dead time transfer function, $g_{ij}(s)$. It is assumed that: $\lambda(s) = \lambda(0)$ (or $\kappa(s) = \kappa(0)$) and $g_{11}(s) = g_{22}(s)$. Sequential design is applied to a system with different values in of κ and θ/τ_p. The maximum closed-loop log modulus $L_{c,\max}$ of the complementary sensitivity function $(h_{ii,\mathrm{CL}} = g_{ii,\mathrm{CL}} k_i/(1 + g_{ii,\mathrm{CL}} k_i))$ are found. Notice that typically $L_{c,\max} = +2$dB is a often used heuristic in SISO tuning (Luyben, 1990). Figure 12A shows that $L_{c,\max}$ are ranging from 1.2 to 28 dB for the Z-N method with $\lambda > 1$ and $0.001 < \theta/\tau_p < 1$. Furthermore, the Z-N method produces an unstable system for $\lambda < 0.3$ (Figure 12A). On the other hand, the proposed method gives fairly constant $L_{c,\max}$ (ranging from 0 to 5.7 dB) for $\lambda > 1$ (Figure 12B) and $0.001 < \theta/\tau_p < 1$. As λ falls below unity, the $L_{c,\max}$ increases. However, the values of $L_{c,\max}$ are still acceptable for $\lambda > 0.3$. For $\lambda < 0.25$ Figure 12B shows an unstable region appearing. Certainly, a more conservative tuning method (a larger f) can be used to eliminate

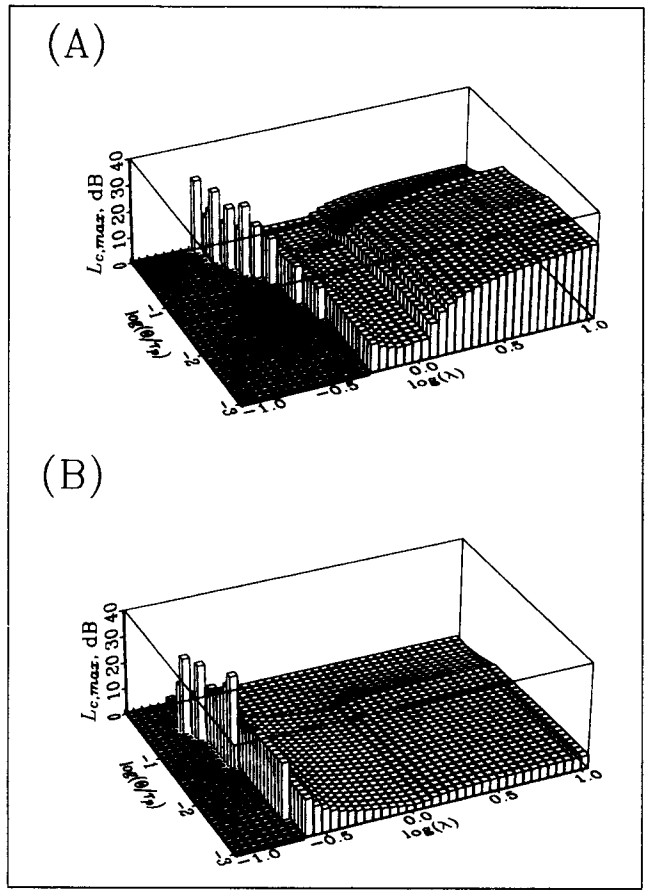

Figure 12. $L_{c,max}$ for 2 × 2 systems using different tuning methods: (A) Z-N tuning and (B) modified Z-N tuning.

the unstable region. However, this can produce sluggish responses for systems with $\lambda > 1$ whenever the constant f tuning rule is applied.

The stability and $L_{c,max}$ analyses for SISO and more realistic MIMO systems show that the modified Z-N rule works for typical process transfer function provided with reasonable variable pairing. More importantly, the tuning constants follow directly from the relay-feedback test (with very little computation).

Performance evaluation—linear model

WB column example (Example 1) is used to test the performance of the proposed tuning method. The identification-design procedure is carried out in the 2 × 2 WB column. Except for the original pairing y_1-u_1 and y_2-u_2, no prior knowledge about the system is assumed. The autotuning procedure is:

(1a) Perform relay-feedback test on y_1-u_1 while loop 2 is on manual (Figure 2A).

(1b) Design PI controller k_1 based on K_{u1} and ω_{u1} according to Eqs. 21 and 22.

(2a) Perform relay-feedback test on y_2-u_2 while loop 1 is on automatic (Figure 2B).

(2b) Design PI controller k_2.

(3a) Perform relay-feedback test on y_1-u_1 while loop 2 is on automatic (Figure 2C).

(3b) Design PI controller k_1.

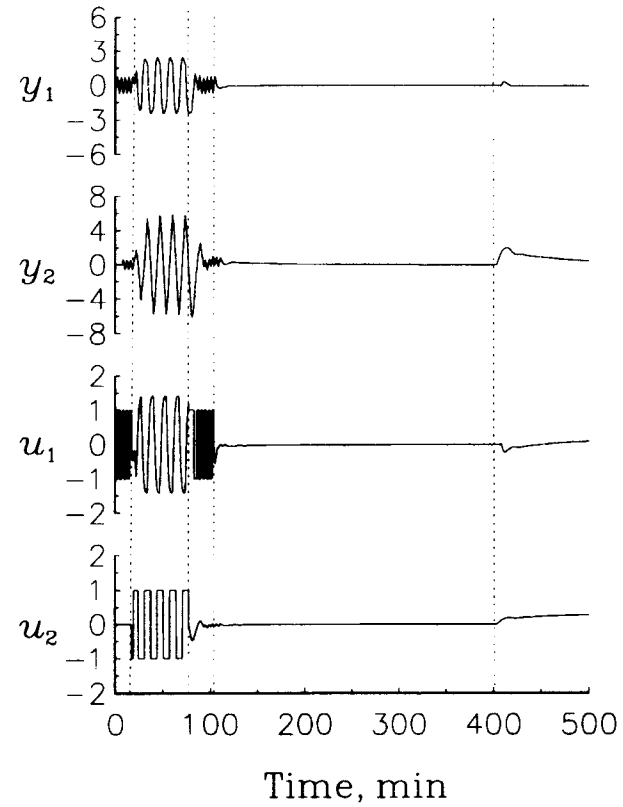

Figure 13. Automatic tuning and load responses for WB column.

This completes the identification-design procedure. Figure 13 shows the autotuning procedure which is completed in the first 100 min. Notice that, in theory, we need another step in the autotuning: redesign k_2 while the new set of PI tuning constants is available. However, simulation results show a new k_1 (or h_1) has little effect on k_2. Therefore, the autotuning is terminated in three steps. A load disturbance is introduced at $t = 400$ min, and the results show that the automatically designed controllers possess good disturbance rejection capability (Figure 13). For a closer look at the performance, the modified Z-N method is compared to the BLT method and empirical method (Luyben, 1986). Simulation results show that the load performance of the proposed tuning method is as good as the well-known tuning method (Figure 14). It should be emphasized, again, that this good performance is achieved with essentially no prior knowledge about the process (the BLT or empirical method need to know the process transfer function for the tuning) and very little engineering effort (finding K_u and ω_u and, subsequently, K_c and τ_I from Eqs. 21 and 22).

Properties

Despite the apparent success of the proposed automatic tuning method, potential problems of the proposed autotuner are raised which are helpful for the applications of the autotuner in an operating environment. Based on the discussions, a procedure for MIMO autotuning is summarized.

Convergence

In theory, the property of convergence of any iterative pro-

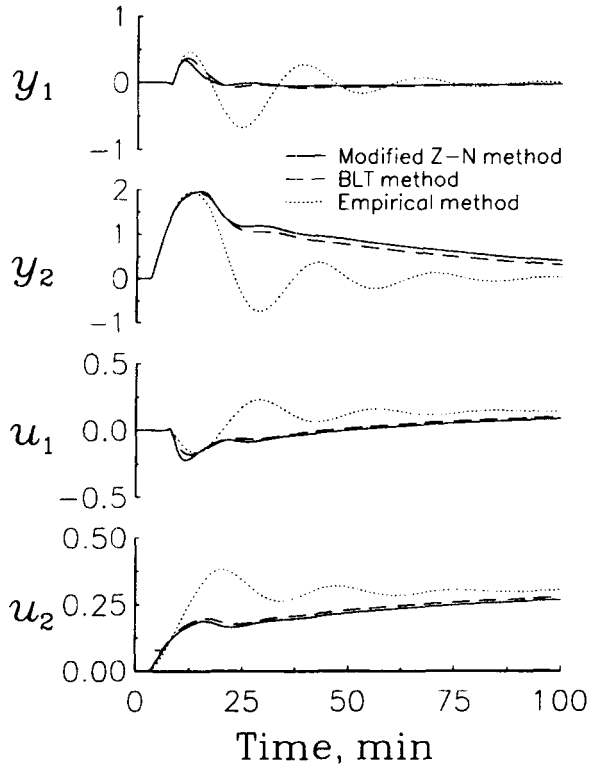

Figure 14. Load responses for WB column with different tuning methods.

cedure is very important. Since the sequential identification-design procedure in MIMO autotuning is an iterative process, the convergence of the proposed autotuning procedure is discussed. In studying 2×2 systems, Bhalodia and Weber (1979) point out that starting from different loops, the sequential design converges to the same set of tuning constants. Recall the autotuning steps (for example, steps 2 and 3 in the section on performance evaluation-linear model) that, in step 2, the identification phase finds K_{u2} and ω_{u2} while loop 1 is on automatic ($h_2(j\omega)$ is known) and the controller design phase calculates k_2, consequently h_2, from K_{u2} and ω_{u2} (Eqs. 21 and 22). When going back to loop 1 in step 3, the purpose is to find K_{u1} and ω_{u1} with h_2 (or k_2) available (found from previous step). Therefore, mathematically, the problem can be formulated as: find ω_{u1} and ω_{u2} such that the following two nonlinear equations converge:

$$f_1(\omega_{u1},\omega_{u2})=\tan^{-1}\left[\frac{Im\{g_{11,CL}(j\omega_{u1},j\omega_{u2})\}}{Re\{g_{11,CL}(j\omega_{u1},j\omega_{u2})\}}\right]=-\pi \tag{24}$$

$$f_2(\omega_{u1},\omega_{u2})=\tan^{-1}\left[\frac{Im\{g_{22,CL}(j\omega_{u1},j\omega_{u2})\}}{Re\{g_{22,CL}(j\omega_{u1},j\omega_{u2})\}}\right]=-\pi \tag{25}$$

where

$$g_{11,CL}(j\omega_{u1},j\omega_{u2})=g_{11}(j\omega_{u1})\times\left[1-\kappa(j\omega_{u1})\frac{g_{22}(j\omega_{u1})k_2}{1+g_{22}(j\omega_{u1})k_2}\right] \tag{26}$$

with

$$k_2=\frac{1}{3|g_{22,CL}(j\omega_{u2})|}\left[1+\frac{1}{\frac{4\pi}{\omega_{u2}}j\omega_{u1}}\right] \tag{27}$$

and a similar expression can be found for $g_{22,CL}(j\omega_{u1},j\omega_{u2})$.

Unlike the conventional way of solving this set of nonlinear equations simultaneously, Eqs. 24 and 25 are solved sequentially. That is: in the k^{th} iteration Eq. 24 is solved for ω_{u1} ($\omega_{u1}^{(k)}$) with ω_{u2} taking constant values from previous iteration ($\omega_{u2}=\omega_{u2}^{(k-1)}$). In the *linear equations* counterpart, this is exactly the Gauss-Seidel method (Rice, 1983) for solving linear algebraic equations in a sequential manner. Consider a set of linear algebraic equations:

$$Ax=B \tag{28}$$

where A is the coefficient matrix with the entry a_{ij}, x is the solution vector, and B is a vector of constant. The necessary and sufficient condition for this sequential equation solving to converge is (Rice, 1983):

$$\rho[I-A_{\text{diag}}^{-1}A]<1 \tag{29}$$

where $\rho(\cdot)$ is the spectral radius (the largest absolute value of the eigenvalue) of $(\cdot)$ and A_{diag} is the matrix with a_{ii} in the diagonal and zero elsewhere. For a 2×2 system, Eq. 29 is equivalent to:

$$\frac{a_{12}a_{21}}{a_{11}a_{22}}<1 \tag{30}$$

As for the case of sequential identification-design, the problem can be formulated as:

$$\begin{pmatrix}\left(\frac{\partial f_1}{\partial\omega_{u1}}\right)_{\omega_{u2}} & \left(\frac{\partial f_1}{\partial\omega_{u2}}\right)_{\omega_{u1}}\\ \left(\frac{\partial f_2}{\partial\omega_{u1}}\right)_{\omega_{u2}} & \left(\frac{\partial f_2}{\partial\omega_{u2}}\right)_{\omega_{u1}}\end{pmatrix}\begin{pmatrix}\omega_{u1}\\ \omega_{u2}\end{pmatrix}=\begin{pmatrix}-\pi\\ -\pi\end{pmatrix} \tag{31}$$

where the overbar stands for the solution of the nonlinear equations. Therefore, the convergence checking is to find whether:

$$\frac{\left(\frac{\partial f_1}{\partial\omega_{u2}}\right)_{\bar\omega_{u1}}\left(\frac{\partial f_2}{\partial\omega_{u1}}\right)_{\bar\omega_{u2}}}{\left(\frac{\partial f_1}{\partial\omega_{u1}}\right)_{\bar\omega_{u2}}\left(\frac{\partial f_2}{\partial\omega_{u2}}\right)_{\bar\omega_{u1}}}<1 \tag{32}$$

An experiment is carried out to test the convergence of the multivariable autotuning procedure. 10,000 cases of 2×2 systems with first-order plus dead time type of transfer functions are generated *randomly*, and Eqs. 24 and 25 are solved simultaneously. Then, the condition for convergence (Eq. 32) is checked by linearizing Eqs. 24 and 25 numerically. Results show that all cases meet the convergence criterion. Therefore, it can be conjectured that there is no convergence problem

associated with the automatic tuning procedure around the neighborhood of the true solution. Actually, this can be understood physically. From Eq. 26, it can be seen that the only way ω_{u2} can affect $g_{11,CL}(j\omega_{u1}, j\omega_{u2})$ is through k_2 (or h_2 in a more general way). However, generally the magnitude of h_j is kept constant for a range of frequencies and the bandwidth of h_j often is larger than that of g_{ii}. Therefore, the change of ω_{u2} has little impact on the solution of $g_{11,CL}$. Since the systems one is dealing with (systems to be autotuned) are virtually unknown, the property of convergence can only be conjectured or qualitatively interpreted. As for our own experience, we have not experienced any convergence problem throughout cases studied.

Tuning sequence

Tuning sequence (which loop to be tuned first) is a problem arisen from sequential design. The work of Bhalodia and Weber (1979) imply that different tuning sequence may result in different speed of convergence (to the final controller parameters). In our experiments, this means some understanding of the tuning sequence can lead to an efficient identification-design procedure. In other words, if a tuning sequence can result in faster convergence, then the autotuning procedure can be terminated sooner. In our problem setting, the process is assumed to be virtually unknown except for some very qualitative description of the process. For example, if the information about the relative speed of the loop responses is available, one may utilize this to devise an appropriate tuning sequence. The relative loop speed of Marino-Galarraga et al. (1987) is useful in this regard. The normalized loop speed for loop i is defined as (Marino-Galarraga et al., 1987):

$$s_i = \frac{\omega_{ui}}{(\omega_{u1} + \omega_{u2})} \tag{33}$$

where ω_{ui} is the ultimate frequency of g_{ii}. From the definition, it is clear that s_i falls between 0 and 1. For a 2×2 system, if the speed of loop 1 (g_{11} to be exact) is faster than that of loop 2, then $s_1 > 0.5$. Notice that s_i is the relative loop speed when only diagonal elements are considered. Such information does not give a complete description of multivariable system in a rigorous way. (However, probably, this is the most easily available dynamic information from plant personnel.) Once s_i is available, one of the loops should be tuned first.

Following the approach of Marino-Galarraga et al. (1987), a simple experiment is performed. Consider fourteen 2×2 systems with λ_{11} values of 0.5 and 2.0 (Table 3). The proposed automatic tuning procedure is applied to these 14 systems starting from loop 1 and loop 2, respectively. Table 3 gives the model structure and the parameter values for these systems. The iterations in the sequential design are terminated when the controller parameters are within 1% of the true values. The results (Table 3) indicate the general trend: faster convergence can be achieved when the *fast* loop is tuned *first*.

Actually, this result can be understood physically. For a system with very different loop speed, dynamically, the system interaction has little impact on the fast loop. For example, the effect from system interaction (through the slow loop) does not show on the fast variable until the transient (results from the transfer function of fast loop) almost dies out. On the opposite, the effect of the fast loop always acts on the slow loop. Mathematically, the bandwidth of complementary sensitivity function for the slow loop (for example, bandwidth of h_2) is much smaller than that of the fast loop (for example, h_1) and typically h_2 is a low pass filter. Therefore, it is very likely that:

$$g_{11,CL}(j\omega_{u1}) = g_{11}(j\omega_{u1})(1 - \kappa(j\omega_{u1})h_2(j\omega_{u1})) \approx g_{11}(j\omega_{u1}) \tag{34}$$

Equation 34 indicates that if loop 1 is tuned first, we are closer to the solution (of Eqs. 24 and 25). The analyses show that if qualitative information about relative loop speed is available, we are able to utilize this in deciding the tuning sequence and, thus, lead to a more efficient autotuning procedure.

Table 3. Convergence for Systems with Different Relative Speeds

System	RGA	θ_{11}	θ_{12}	θ_{21}	θ_{22}	s_1	Iteration No. Loop 1 first/ Loop 2 first
1	0.5	0.40	2.70	3.00	5.70	0.905	2/3
2	0.5	1.00	2.70	3.00	5.70	0.811	3/4
3	0.5	2.00	2.70	3.00	5.70	0.709	3/3
4	0.5	3.00	2.70	3.00	5.70	0.633	4/5
5	0.5	4.50	2.70	3.00	5.70	0.550	3/8
6	0.5	5.00	2.70	3.00	5.70	0.527	3/8
7	0.5	6.00	2.70	3.00	5.70	0.489	4/3
8	2.0	0.40	2.70	3.00	5.70	0.905	3/4
9	2.0	1.00	2.70	3.00	5.70	0.811	3/3
10	2.0	2.00	2.70	3.00	5.70	0.709	3/3
11	2.0	3.00	2.70	3.00	5.70	0.633	2/3
12	2.0	4.50	2.70	3.00	5.70	0.550	4/5
13	2.0	5.00	2.70	3.00	5.70	0.527	4/5
14	2.0	6.00	2.70	3.00	5.70	0.489	5/3

where:

$$G(s) = \begin{pmatrix} \frac{kp_{11}}{\tau_{p11}s+1}\exp(-\theta_{11}s) & \frac{kp_{12}}{\tau_{p12}s+1}\exp(-\theta_{12}s) \\ \frac{kp_{21}}{\tau_{p21}s+1}\exp(-\theta_{21}s) & \frac{kp_{22}}{\tau_{p22}s+1}\exp(-\theta_{22}s) \end{pmatrix}$$

$\tau_{p11} = \tau_{p12} = \tau_{p21} = \tau_{p22} = 1$, for $\lambda = 0.5$, $k_{p11} = k_{p12} = k_{p22} = 1$ and $k_{p21} = -1$; $\lambda = 2.0$, $k_{p11} = k_{p12} = k_{p22} = 1$ and $k_{p21} = 0.5$.

Problem of variable pairing

In theory, variable pairing should not pose any problem at this operating stage. That is: all outputs and manipulated inputs are paired correctly and the process is controlled via DCS or single-station controllers. However, the proposed autotuning procedure can be used to eliminate undesirable variable pairing. This is helpful at the commissioning stage of a new control system since we are able to spot potential problems in the plant. The WB column example is used to illustrate the peculiar behavior and poor performance when an undesirable pairing is configured. Notice that the RGA for the correct pairing is: $\lambda_{11}(0) = 2.01$, and the well-known fact that a system paired with negative λ_{11} is undesirable (Bristol, 1966; Grosdidier et al., 1985).

Example 3: WB Column with Undesirable Pairing. If the

WB column is paired incorrectly (y_1-u_2 and y_2-u_1), then we have:

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = G' \begin{pmatrix} u_2 \\ u_1 \end{pmatrix} = \begin{pmatrix} \dfrac{-18.9e^{-3s}}{21s+1} & \dfrac{12.8e^{-s}}{16.7s+1} \\ \dfrac{-19.4e^{-3s}}{14.4s+1} & \dfrac{6.6e^{-7s}}{10.9s+1} \end{pmatrix} \begin{pmatrix} u_2 \\ u_1 \end{pmatrix} \quad (35)$$

This pairing gives a negative value in λ_{11} ($\lambda_{11} = -1.01$). Consider two cases. The first case is that: we do not know the signs of the diagonal elements (in reality, it is nearly impossible not to know the signs of the diagonal elements once the control hardware is installed) or forget to apply the knowledge of the "sign" in the relay-feedback tests. Figure 15 illustrates the process of the autotuning. A relay-feedback test ($t = 0 \sim 50$ min in Figure 15) is performed on loop 1 (the y_1-u_2 loop) first and a PI controller is designed using the modified Z-N method; then, another test is carried out on the y_2-u_1 loop ($t = 50 \sim 200$ in Figure 15) and the second PI controller is design. The procedure iterates back to loop 1 with a controller design and the autotuning process is terminated at $t = 300$ min, as shown in Figure 15. After the transient dies out, a unit step load change is applied (at $t = 400$ min). Poor load responses are observed (Figure 15) when it is compared to the load responses in Figure 13 (the load responses with the correct pairing).

With a closer look at the sustained oscillations when loop 2 (y_2-u_1) is tuned (Figure 17), one can find that the closed-loop gain ($g_{22,CL}$) is *negative* which is different from the *positive* open-loop gain (the (2,2) element in Eq. 35). Notice that if the sign of the process gain is positive, the output and the input move toward an opposite direction in a half period (Figure 1B). Therefore, the controller gain is negative in loop 2:

$$K(s) = \begin{pmatrix} -0.123\left(\dfrac{30.7s+1}{30.7s}\right) & 0 \\ 0 & -0.17\left(\dfrac{78.1s+1}{78.1s}\right) \end{pmatrix} \quad (36)$$

That implies that loop 2 cannot be stabilized by itself (Grosdidier et al., 1985; Fan and Yu, 1990). Despite the fact that the overall closed-loop system is stable, as shown in Figure 15, this control structure is lack of integrity. Furthermore, $g_{11,CL}$ is (open-loop) unstable, as shown in Appendix B. Therefore, poor responses can be expected for such a poorly conditioned closed-loop system. However, not a single indication of any wrongdoing is observed in the autotuning process (Figure 15) except for the poor responses.

Let us consider the second case that we know or are aware of the "sign" of the diagonal elements. Loop 1 (y_1-u_2 loop) is tested, and the controller is designed in the same way as that of previous case ($t = 0 \sim 50$ min in Figure 16). After loop 1 is closed, a second relay-feedback test is performed between y_2 and u_1. An increase in u_1 is made initially. Since we know the sign between y_2 and u_1 is positive, we are waiting for y_2 to cross the set point for the relay to switch. However, y_2 simply leaves off toward the negative direction (Figure 16), and the relay never switches. This means the relay-feedback test fails, (because we insist that the sign of the transfer function is

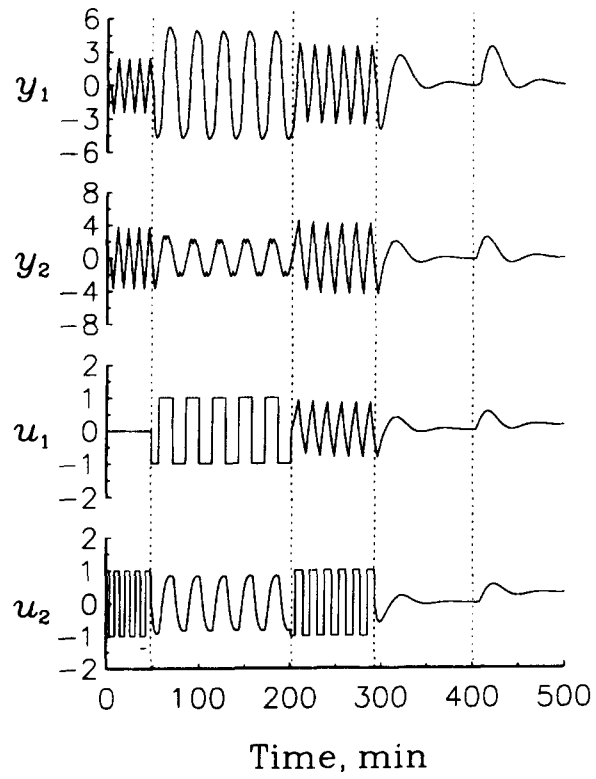

Figure 15. Automatic tuning and load responses for WB column with wrong pairing.

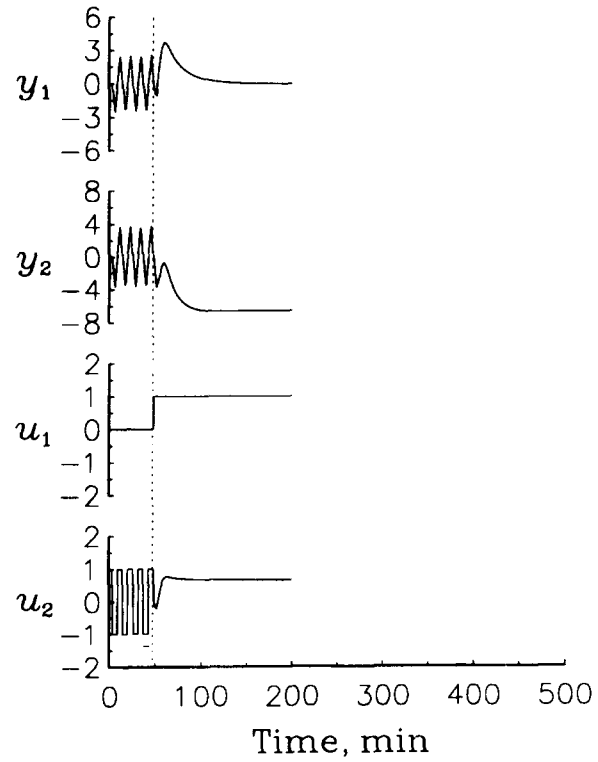

Figure 16. Automatic tuning procedure for WB column with wrong pairing provided with knowing the signs of diagonal elements.

positive). Once such a situation occurs, an undesirable pairing is realized.

From the ongoing analyses, it is clear that if the knowledge about the signs of the diagonal elements are available or, most likely, one is aware of the signs in the relay-feedback tests, the undesirable pairing can be eliminated in the process. Unfortunately, it can only be used to eliminate *undesirable* pairings but not for finding the *best* pairing.

Summary of procedure

From the discussed properties, an effective procedure for multivariable autotuning is proposed depending on the extent of process knowledge available. Process information is classified into "required" and "helpful." The "signs" of the diagonal elements are the required input data. As mentioned earlier, as the control hardware is installed, it is nearly impossible not to know the sign of the steady-state gain (for example, not to know the controller is direct or reverse acting for the diagonal elements). The relative speed of diagonal transfer function, for example, s_i in Eq. 33, is helpful since knowing this can lead to faster convergence for systems with very different loop speeds. With the process information available, the automatic tuning procedure for a $n \times n$ multivariable system becomes (Figure 17):

(0) Rank the loop speed from fast to slow into 1, 2, . . . , n.

(1a) Perform relay-feedback test on loop 1 (the relay is switched according to the sign of g_{ii}).

(1b) Design k_1 using modified Z-N method (Eqs. 21 and 22) and put k_1 into automatic.

(2a) Perform relay feedback test on loop 2.

(2b) Design k_2 using modified Z-N method and put k_2 into automatic.

⋮

(na) Perform relay-feedback test on loop n.

(nb) Design k_n using modified Z-N method and put k_n into automatic.

This procedure is repeated (back to step 1) until the controller parameters converge. Typically, total of $n + (n-1)$ identification-design steps will suffice, as shown in the next section. Figure 17 shows the flow chart of the MIMO autotuning procedure.

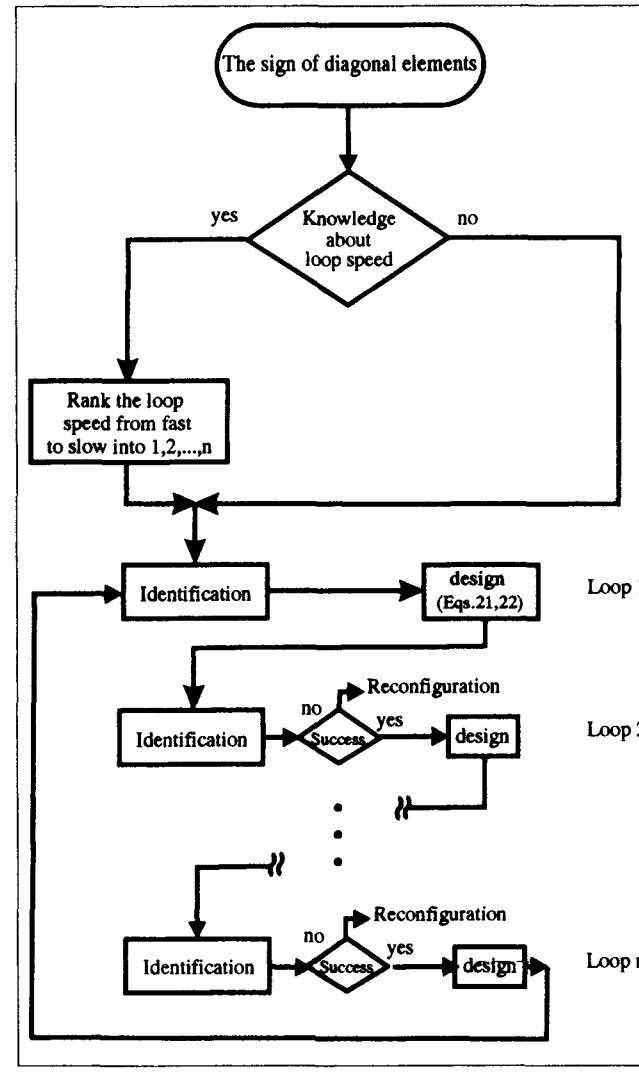

Figure 17. MIMO autotuning procedure.

Applications

Two nonlinear distillation columns and a 3×3 linear example are used to illustrate the MIMO autotuning procedure. For the nonlinear examples, one is a moderate-purity column (Shen and Yu, 1992), and the other one is a high-purity column (Papastathopoulou and Luyben, 1990). The 3×3 linear system is a transfer function matrix for distillation column (T4) (Luyben, 1986).

Moderate-purity column

The column studied by Shen and Yu (1992) is a 20-tray distillation column. The product specifications are 98% and 2% of the light mole fractions component on the top and bottoms of the column. The relative volatility is 2.26 with a reflux ratio 1.76. Table 4 summarizes the steady-state operating conditions. The control objective is to maintain to top and bottom compositions ($x_D = 0.98$ and $x_B = 0.02$) by changing the reflux flow rate (R) and vapor boilup (V). This is the conventional R-V control structure. In the nonlinear simulation, the following assumptions are made:

(1) binary system;
(2) equal molar overflow;
(3) 100% tray efficiency;
(4) saturated liquid feed;
(5) total condenser and partial reboiler; and
(6) perfect level control

According to the autotuning procedure in Figure 18, relay-feedback tests are performed on the R-V controlled column. The relay heights (h) of 3% are used in R and V. The x_D-R loop is tuned first and the controller (k_1) is designed (Eqs. 21 and 22) followed by the second relay-feedback test on the x_B-V loop (Figure 18). After the bottoms controller (k_2) is designed and puts on automatic, the third relay feedback experiment is performed and the controller parameters for k_1 are finalized. The resultant controller parameters are: $K_{c1} = 245.34$, $\tau_{I1} = 97.1$ and $K_{c2} = -157.4$, $\tau_{I2} = 47.8$. The results of the first ($t = 0 \sim 200$ min) and the third ($t = 400 \sim 600$ min) relay-feedback test clearly show that the controller (k_1) designed from

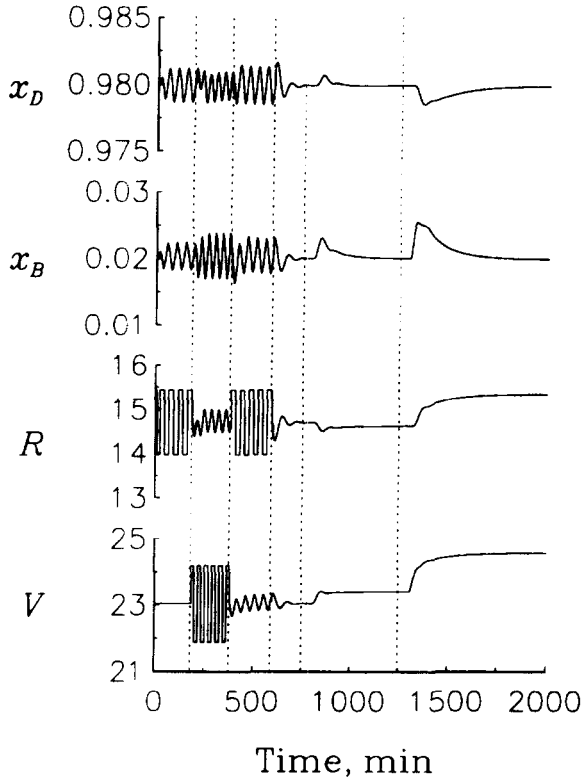

Figure 18. Automatic tuning and load responses for a 5% step change in feed composition for the Shen and Yu column.

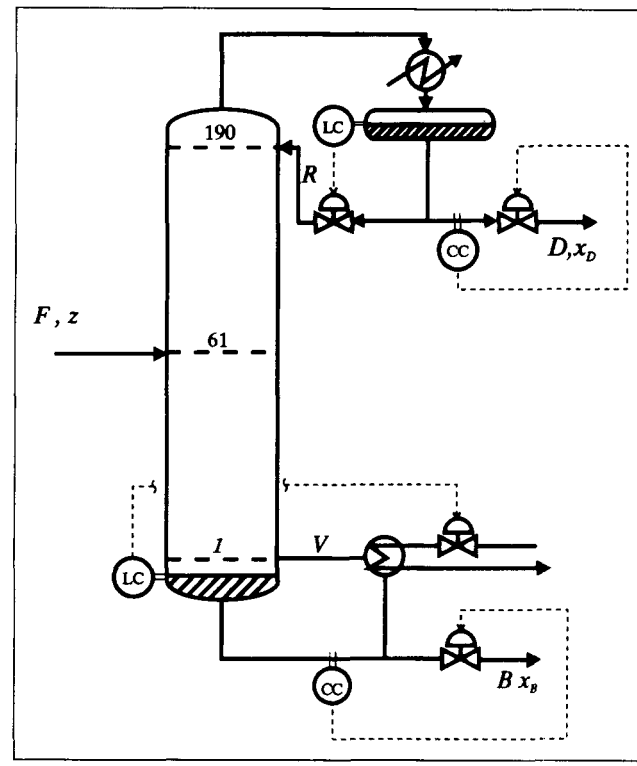

Figure 19. D-B control structure for C_3 splitter.

these two tests are almost the same ($K_{c1} = 223.25$ and 245.36 and $\tau_{I1} = 89.0$ and 97.1). That means, actually, two relay-feedback tests are sufficient for this column. A +5% step flow rate change is introduced at t = 800 min and a +5% feed composition change is also made at $t = 1{,}300$ min. Figure 18 shows good load rejections are achieved with the tuning constants from the autotuning procedure. It should be emphasized that the good load responses are achieved with little engineering effort; no transfer function are fitted and no frequency domain plots are generated. The only knowledge assumed is that: one has to know the "sign" between x_D and R is positive and the "sign" between x_B and V is negative. Actually, this information is not applied since the outputs and inputs are paired correctly.

High-purity column

The high-purity column is a C_3 splitter (Papastathopoulou and Luyben, 1990) which separates propane and propylene in 190-tray column. The relative volatility ranging from 1.12 to 1.24 and the product specifications are 99.66% and 0.02% light component (propylene) at the top and bottoms of the column, respectively. This is a very high purity column with difficult separation. In terms of control, this is a highly nonlinear system (Fuentes and Luyben, 1983). The *D-B* control structure is considered for dual-composition control (Figure 19). The steady-state operating conditions are given in Table 4. The autotuning procedure is applied to the C_3 splitter. The x_D-D loop is tuned first and the autotuning is terminated in three relay-feedback experiments. Again, the controller parameters for the x_D-D loop are expected to be almost the same from the first and the third relay-feedback tests (Figure 20). Table 5 shows the corresponding tuning constants from the autotuning step. A 5% feed composition change is made at $t = 1{,}000$ min (Figure 20) and simulation results show that good load responses are obtained. Chang and Yu (1992) also study the modeling, tuning and robustness aspects of the C_3 splitter. Since this column is highly nonlinear and nonconventional, control structure is employed (result in transfer function with 1/s term), a great deal of effort is spent to find the transfer function matrix (Papastathopoulou and Luyben, 1990; Chang and Yu, 1992). Chang and Yu (1992) use stepping technique (Luyben, 1990) to find frequency responses for the R-V struc-

Table 4. Steady-State Values for the Binary Distillation Columns

Column	Shen and Yu	C_3 Splitter
Number of trays	20	190
Feed Tray	10	61
Relative Volatility	2.26	1.12 ~ 1.24*
Operating Pressure (atm)	1.0	7.5
Feed Flow Rate (kg·mol/min)	36.3	16.67
Distillation Flow Rate (kg·mol/min)	18.15	11.71
Bottoms Flow Rate (kg·mol/min)	18.15	4.96
Reflux Ratio	1.76	10.3
Feed Composition (mol fraction)	0.50	0.7
Distillation Composition (mol fraction)	0.98	0.9966
Bottoms Composition (mol fraction)	0.02	0.0002

* $\alpha = a_0 - a_1 x - a_2 x^2$ with $\begin{cases} a_0 = 1.23644 \\ a_1 = 0.076695 \\ a_2 = 0.036147 \end{cases}$

where x = mole fraction of light component on each tray.

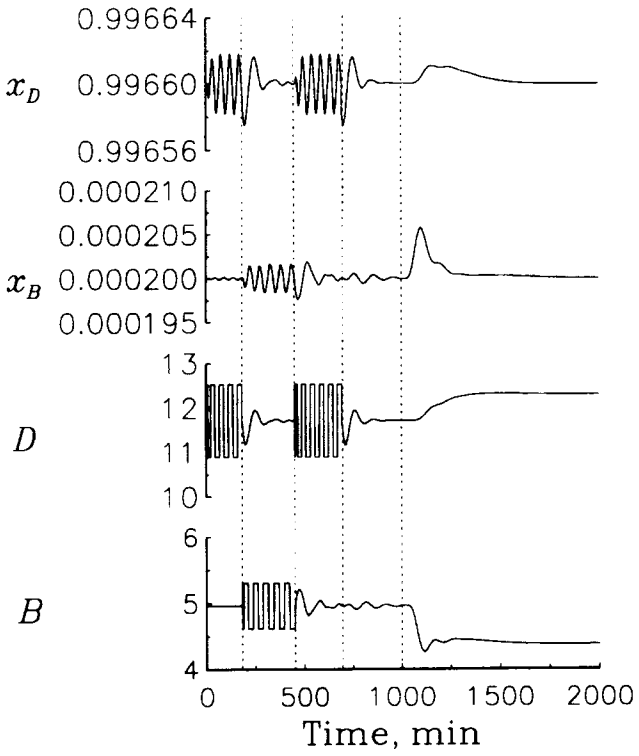

Figure 20. Automatic tuning and load responses for a 5% step change in feed composition for the C_3 splitter.

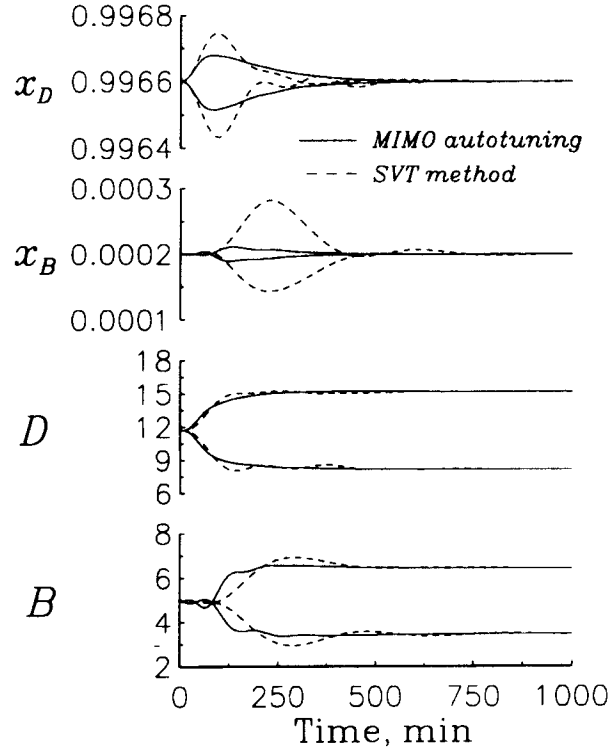

Figure 21. Load responses for ±30% feed composition changes with different design methods.

ture followed by a transformation from *R-V* to *D-B* structure (Papastathopoulou and Luyben, 1990). Once the transfer function matrix is obtained, a modified version of singular value tuning (SVT) (Chiang and Luyben, 1988) is used to find the tuning constants for the PI controllers (Chang and Yu, 1992). Table 5 gives the autotuning constants from the modified SVT.

Comparisons are made between the autotuning approach and the SVT tuned control system. Figure 21 shows the load responses for ±30% feed composition changes, and Figure 22 shows the load responses for ±30% feed flow rate disturbances. The results show that better load responses are achieved using the proposed autotuning procedure. More importantly, good performance is achieved with very little engineering effort. The multivariable autotuner is activated at $t=0$ and the tuning is completed at $t=700$ min following three relay-feedback tests.

3×3 linear example (T4) (Luyben, 1986)

This autotuning procedure can be extended to a $n \times n$ multivariable system in a straightforward manner. The following 3×3 literature example (Luyben, 1986) illustrates this extension.

The transfer function matrix of T4 column is:

Table 5. Controller Parameters for C_3 Splitter Using the Proposed Autotuning Procedure and SVT

Tuning Method	Step 1 x_D-D Loop (K_c/τ_I)	Step 2 x_B-B Loop (K_c/τ_I)	Step 3 x_D-D Loop (K_c/τ_I)
MIMO autotuner	150.42/93.42	42.28/108.32	150.08/91.68
SVT method	45.64/79.35	7.91/150.0	-

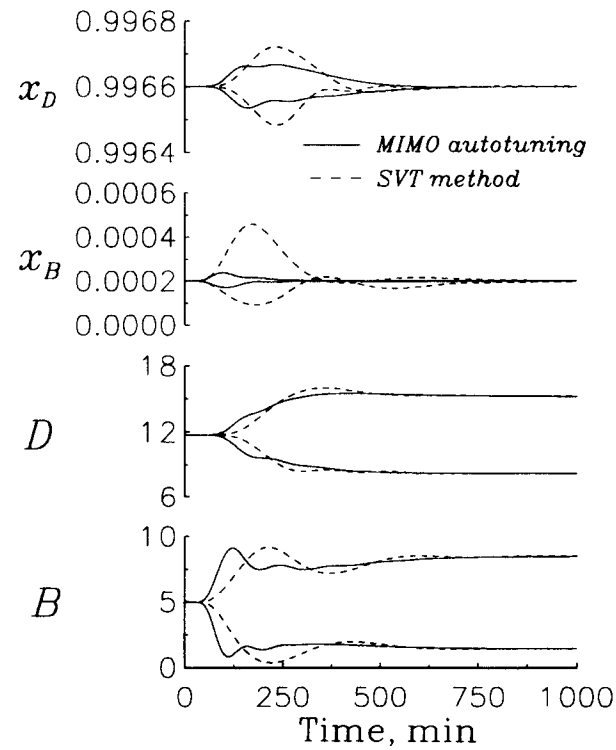

Figure 22. Load responses for ±30% feed flow rate changes with different design methods.

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} \dfrac{-2.986e^{-0.71s}}{66.7s+1} & \dfrac{-5.24e^{-60s}}{400s+1} & \dfrac{-5.984e^{2.24s}}{14.29s+1} \\ \dfrac{0.0204e^{-0.59s}}{(7.14s+1)^2} & \dfrac{-.33e^{-0.68s}}{(2.38s+1)^2} & \dfrac{2.38e^{-0.42s}}{(1.43s+1)^2} \\ \dfrac{0.374e^{-7.75s}}{22.22s+1} & \dfrac{-11.3e^{-3.79s}}{(21.74s+1)^2} & \dfrac{-9.811e^{-1.59s}}{11.36s+1} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix}$$

Following the autotuning procedure, the loop speed is ranked from fast to slow as: loop 1, loop 2, and loop 3. The autotuning is carried out according to Figure 17 and the resultant controller parameters are: $K_{c1} = -2.51$, $K_{c2} = -6.34$, $K_{c3} = -0.23$ and $\tau_{I1} = 16.2$, $\tau_{I2} = 12.4$, $\tau_{I3} = 12.6$. Figure 23 shows that the tuning procedure is completed in the first 180 min and a set point change is made at $t = 300$ min. Notice that except for K_{c1}, the controller parameters from 5 relay-feedback tests are essentially the same as the resultant parameters ($K_{c1} = -3.24$, $K_{c2} = -6.05$, $K_{c3} = -0.28$ and $\tau_{I1} = 12.05$, $\tau_{I2} = 12.17$ and $\tau_{I3} = 12.37$). The results (Figure 23) show that the autotuning procedure gives reasonable servo responses. Figure 24 compares the set of responses of the modified Z-N method and BLT method. The results show that the better performance is achieved using the modified Z-N method. Again, the results show that the proposed autotuning procedure achieves good performance with very little engineering effort for a higher-order (3×3) system.

Conclusion

The Åström-Hägglund autotuner is extended to MIMO systems. Based on the concept of sequential identification-design, an approach for the automatic tuning of multivariable systems is proposed. The consistent nature (satisfying consistency relations) of sequential identification is discussed and the advantages are shown. The poles-zeros configurations of the sequential design are discussed with the implications on underdamped process characteristics. From the observed process

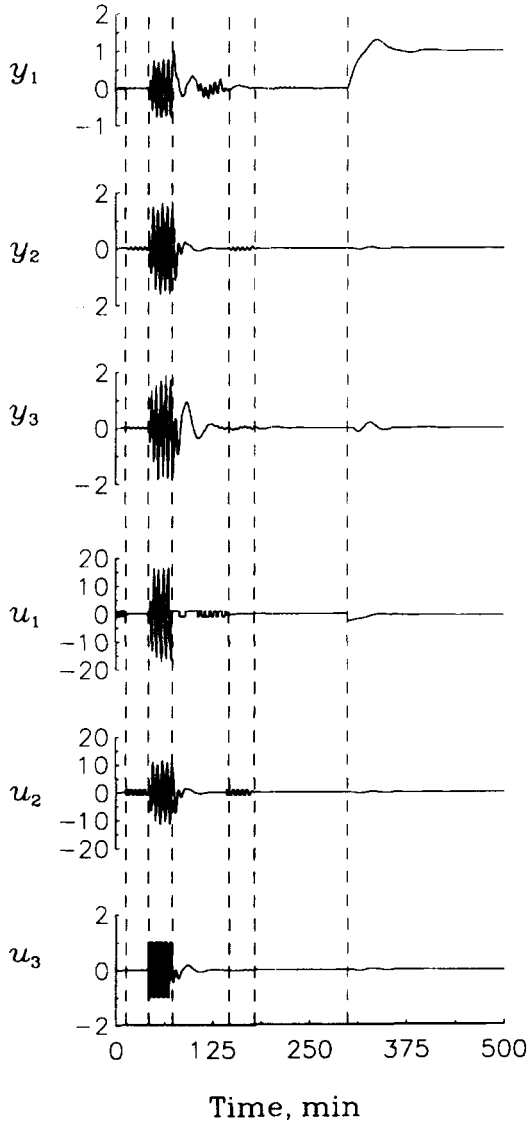

Figure 23. Automatic tuning and set point change in loop 1 for T4 column.

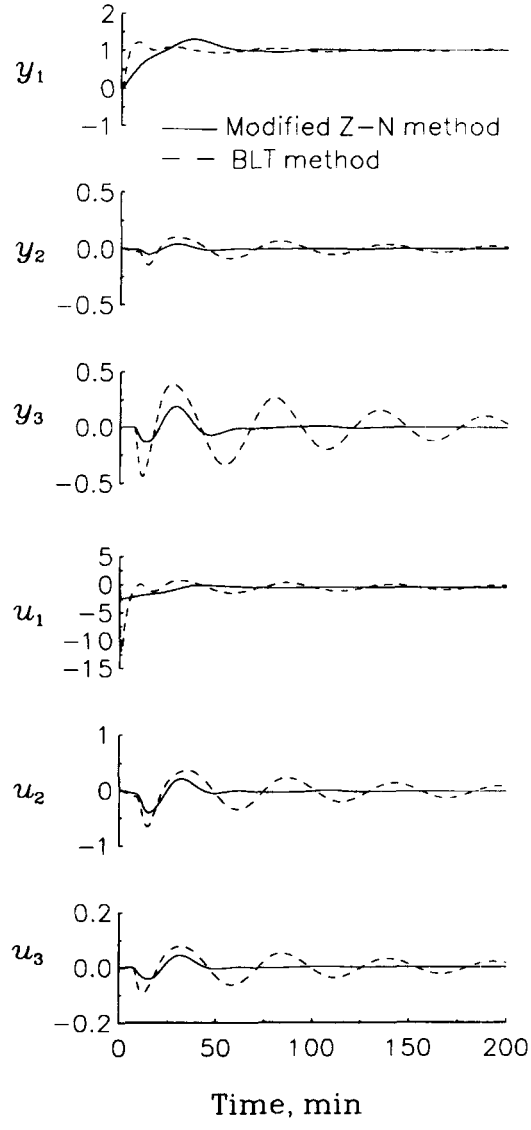

Figure 24. Set point change in loop 1 for T4 column with different tuning methods.

characteristic, the well-known Ziegler-Nichols method is modified. The concept of sequential identification using relay-feedback tests and the modified Z-N controller design method forms the basis for the MIMO autotuner. The convergent nature of the autotuner is conjectured. The properties of convergence, tuning sequence and undesirable variable pairing are explored. Based on these results, a multivariable autotuning procedure is proposed. Two nonlinear distillation examples and a 3×3 linear system are used to illustrate the effectiveness of the proposed autotuner. The results show that good performance can be achieved with minimal engineering effort. More importantly, the proposed MIMO autotuner works and works well for very difficult processes in a transparent manner.

Acknowledgment

This work is supported by the National Science Council of the R.O.C. under grant No. NSC 82-0402-E-011-038.

Notation

a = amplitude of limit cycle
B = bottoms flow rate
D = distillate flow rate
f = detuning factor
F = outlet flow rate of blending system
F_1 = flow rate of stream 1 in blending system
F_2 = flow rate of stream 2 in blending system
$g_{ij}(s)$ = (i, j) entry of $G(s)$
$g_{ii,\mathrm{CL}}$ = closed-loop transfer function of loop i in sequential design
$G(s)$ = process transfer matrix
h = magnitude of relay output
h_j = complementary sensitivity function of loop j
$K(s)$ = controller transfer function matrix
K_c = controller gain
$k_i(s)$ = controller transfer function for loop i
k_p = steady-state gain
K_u = ultimate gain
$L_{c,\max}$ = maximum closed-loop log modules
P_u = period of limit cycle
R = reflux flow rate
s = Laplace transform variable
s_i = relative speed of loop i
$\boldsymbol{u}$ = vector of process inputs
u_i = ith process input
V = vapor boilup flow rate
x = composition of outlet stream in blending system
x_D = top component
x_B = bottoms component
x_1 = composition of stream 1 in blending system
x_2 = composition of stream 2 in blending system
$\boldsymbol{y}$ = vector of process output
y_i = ith process output

Greek letters

α = relative volatility
ζ = damping coefficient
θ = time delay
κ = Rijndorp interaction measurement $(= g_{12}g_{21}/g_{11}g_{22})$
λ_{ij} = (i, j) entry of RGA
ρ = spectral radius
τ_I = reset time of PI controller
ω_u = ultimate frequency

Superscripts

$-$ = nominal steady-state value
(k) = kth iteration

Literature Cited

Åström, K. J., and T. Hägglund, "Automatic Tuning of Simple Regulators with Specifications on Phase and Amplitude Margins," *Automatica*, **20**, 645 (1984).
Åström, K. J., C. C. Hang, P. Persson, and W. K. Ho, "Towards Intelligent PID Control," *Automatica*, **28**, 1 (1992).
Åström, K. J., and T. J. McAvoy, "Intelligent Control," *J. Process Control*, **2**, 115 (1992).
Bhalodia, M., and T. W. Weber, "Feedback Control of a Two-Input, Two-Output Interacting Process," *Ind. Eng. Chem. Process Des. Dev.*, **18**, 599 (1979).
Bernstein, D. S., "Sequential Design of Decentralized Dynamic Compensators Using the Optimal Projection Equations," *Int. J. Control*, **46**, 1569 (1987).
Bristol, E. H., "Pattern Recognition: An Alternative to Parameter Adaptive PID Controller," *Automatica*, **13**, 197 (1977).
Bristol, E. H., "New Measure of Interaction for Multivariable Process Control," *IEEE Trans. Automat. Control*, **AC-11**, 133 (1966).
Cao, R., and T. J. McAvoy, "Evaluation of Pattern Recognition Adaptive PID Controller," *Automatica*, **26**, 797 (1990).
Chang, D. M., and C. C. Yu, "The Distillate-Bottoms Control of Distillation Columns: Modeling, Tuning and Robustness Issues," *J. Chin. I. Ch. E.*, **23**, 344 (1992).
Chen, C. L., "A Simple Method for On-Line Identification and Controller Tuning," *AIChE J.*, **35**, 2037 (1989).
Chiang, T. P., and W. L. Luyben, "Comparison of the Dynamic Performances of Three Heat-Integrated Distillation Configurations," *Ind. Eng. Chem. Res.*, **27**, 99 (1988).
Chiang, R. C., and C. C. Yu, "Monitoring Procedure for Intelligent Control: On-Line Identification of Maximum Closed-Loop Log Modulus," *Ind. Eng. Chem. Res.*, **32**, 90 (1993).
Chiu, M. S., and Y. Arkun, "A Methodology for Sequential Design of Robust Decentralized Control Systems," *Automatica*, **28**, 997 (1992).
Fuentes, C., and W. L. Luyben, "Control of High-Purity Distillation Columns," *Ind. Eng. Chem. Process Des. Dev.*, **22**, 361 (1983).
Häggblom, K. E., and K. V. Waller, "Transformations and Consistency Relations of Distillation Control Structures," *AIChE J.*, **34**, 1634 (1988).
Häggblom, K. E., and K. V. Waller, "Control Structure, Consistency, and Transformations," *Practical Distillation Control*, W. L. Luyben, ed., Van Nostrand Reinhold, New York (1992).
Hägglund, T., and K. J. Åström, "Industrial Adaptive Controllers Based on Frequency Response Techniques," *Automatica*, **27**, 599 (1991).
Hsu, L., M. Chan, and A. Bhaya, "Automated Synthesis of Decentralized Tuning Regulators for Systems with Measurable DC Gain," *Automatica*, **28**, 185 (1992).
Hinde, R. F., and D. J. Cooper, "Adaptive Process Control Using Pattern-based Performance Feedback," *J. Process Control*, **1**, 228 (1991).
Grosdidier, P., M. Morari, and R. B. Holt, "Closed-Loop Properties from Steady-State Gain Information," *Ind. Eng. Chem. Process Des. Dev.*, **24**, 221 (1985).
Koivo, H. N., and S. Pohjolainen, "Tuning of Multivariable PI-Controller for Unknown Systems with Input Delay," *Automatica*, **21**, 81 (1985).
Leithead, W. E., and J. O'Reilly, "Performance Issues in The Individual Channel Design of 2-input 2-output Systems. Part 1. Structural Issues," *Int. J. Control*, **54**, 47 (1991).
Lin, J. Y., and C. C. Yu, "Automatic Tuning and Gain Scheduling for pH Control," *Chem. Eng. Sci.*, **48**, 3150 (1993).
Luyben, W. L., "Simple Method for Tuning SISO Controllers in Multivariable Systems," *Ind. Eng. Chem. Process Des. Dev.*, **25**, 654 (1986).
Luyben, W. L., "Sensitivity of Distillation Relative Gain Arrays to Steady-State Gains," *Ind. Eng. Chem. Res.*, **26**, 2076 (1987a).
Luyben, W. L.,"Derivation of Transfer Functions for Highly Nonlinear Distillation Columns," *Ind. Eng. Chem. Res.*, **26**, 2490 (1987b).
Luyben, W. L., *Process Modeling, Simulation and Control for Chemical Engineers*, McGraw-Hill, Singapore (1990).
Marino-Galarraga, M., T. J. McAvoy, and T. E. Marlin, "Short-Cut

Operability Analysis. 2. Estimation of f_i Detuning Parameter for Classical Control Systems," *Ind. Eng. Chem. Res.*, **26**, 511 (1987).
Mayne, D. Q., "The Design of Linear Multivariable Systems," *Automatica*, **9**, 201 (1973).
Mayne, D. Q., "Sequential Design of Linear Multivariable Systems," *Proc. IEE Part D*, **126**, 568 (1979).
Ogunnaike, B. A., and W. H. Ray, "Multivariable Controller Design for Linear Systems Having Multiple Time Delays," *AIChE J.*, **25**, 1043 (1979).
O'Reilly, J., and W. E. Leithead, "Multivariable Control by Individual Channel Design," *Int. J. Control*, **54**, 1 (1991).
Papastathopoulou, H. S., and W. L. Luyben, "Tuning Controllers on Distillation Columns with the Distillate-Bottoms Structure," *Ind. Eng. Chem. Res.*, **29**, 1859 (1990).
Penttinen, J., and H. N. Koivo, "Multivariable Tuning Regulators for Unknown Systems," *Automatica*, **16**, 393 (1980).
Rice, J. R., *Numerical Methods, Software, and Analysis*, McGraw-Hill, New York (1983).
Rijnsdorp, J. E., "Interaction in Two-Variable Control Systems for Distillation," *Automatica*, **1**, 15 (1965).
Schei, T. S., "A Method for Closed Loop Automatic Tuning of PID Controllers," *Automatica*, **28**, 587 (1992).
Seborg, D. E., T. F. Edgar, and D. A. Mellichamp, *Process Dynamics and Control*, John Wiley, New York (1989).
Shen, S. H., and C. C. Yu, "Indirect Feedforward Control: Multivariable Systems," *Chem. Eng. Sci.*, **47**, 3085 (1992).
Tan, L. Y., and T. W. Weber, "Controller Tuning of a Third-Order Process under Proportional-Integral Control," *Ind. Eng. Chem. Process Des. Dev.*, **24**, 1155 (1985).
Skogestad, S., and P. Landström, "μ-Optimal LV-Control of Distillation Columns," *Comput. Chem. Eng.*, **14**, 401 (1990).
Wood, R. K., and M. W. Berry, "Terminal Composition Control of a Binary Distillation Column," *Chem. Eng. Sci.*, **28**, 1707 (1973).
Yu, C. C., and M. K. H. Fan, "Decentralized Integral Controllability and *D*-stability," *Chem. Eng. Sci.*, **45**, 3299 (1990).
Yuwana, M., and D. E. Seborg, "A New Method for On-Line Controller Tuning," *AIChE J.*, **28**, 434 (1982).
Ziegler, J. G., and N. B. Nichols, "Optimum Settings for Automatic Controllers," *Trans. ASME*, **12**, 759 (1942).

Appendix A

Consider the blending system in Figure 9. Equations 17 and 18 give the overall and component material balances. From the material balances the process transfer function linearized at the steady state ($\bar{F}$, $\bar{F}_1$, $\bar{F}_2$, $\bar{x}$, $\bar{x}_1$, $\bar{x}_2$) is shown in Eq. 19. From the Eq. 19, the closed-loop gains for loops 1 and 2 are:

$$g_{11,CL} = \left(\frac{\Delta F}{\Delta F_1}\right)_{\Delta x=0} = g_{11}\left(1-\frac{g_{12}g_{21}}{g_{11}g_{22}}\right) = \frac{\bar{F}}{\bar{F}_1} \tag{A1}$$

$$g_{22,CL} = \left(\frac{\Delta x}{\Delta F_2}\right)_{\Delta F=0} = g_{22}\left(1-\frac{g_{12}g_{21}}{g_{11}g_{22}}\right) = \frac{\bar{x}_2-\bar{x}_1}{\bar{F}} \tag{A2}$$

Sequential identification

In the sequential identification, the closed-loop gain for loop 1 can be found when loop 2 is on automatic. From Eqs. 17 and 18, the closed-loop gain can be derived for a change in F_1 ($F_1=\bar{F}_1+\Delta F_1$) by solving the following equations:

$$(\bar{F}+\Delta F) = (\bar{F}_1+\Delta F_1) + (\bar{F}_2+\Delta F_2) \tag{A3}$$

$$(\bar{x}+\Delta x)(\bar{F}+\Delta F) = \bar{x}_1(\bar{F}+\Delta F_1) + \bar{x}_2(\bar{F}_2+\Delta F_2) \tag{A4}$$

The closed-loop gain becomes:

$$g_{11,CL} = \left(\frac{\Delta F}{\Delta F_1}\right)_{\Delta x=0} = \frac{\bar{F}}{\bar{F}_1} \tag{A5}$$

Similarly, $g_{22,CL}$ can be found for a change in F_2 ($F_2=\bar{F}_2+\Delta F_2$):

$$g_{22,CL} = \left(\frac{\Delta x}{\Delta F_2}\right)_{\Delta F=0} = \frac{\bar{x}_2\bar{x}_1}{\bar{F}} \tag{A6}$$

Notice that $g_{ii,CL}$s are constants regardless of the magnitude of changes in F_1 and F_2.

Independent identification

The independent identification, on the other hand, finds g_{ij}s first and the combinations of g_{ij}s are used to calculate $g_{ii,CL}$. Following the same procedure, the g_{ij}s can be found from the material balances (Eqs. 17 and 18) for changes in ΔF_1 and ΔF_2, respectively. First, g_{11} and g_{21} can be found for a change in ΔF_1:

$$g_{11} = \left(\frac{\Delta F}{\Delta F_1}\right)_{\Delta F_2=0} = 1 \tag{A7}$$

$$g_{21} = \left(\frac{\Delta x}{\Delta F_1}\right)_{\Delta F_2=0} = \frac{\bar{x}_1-\bar{x}}{\bar{F}+\Delta F_1} \tag{A8}$$

Similarly, a change in ΔF_2 gives:

$$g_{12} = \left(\frac{\Delta F}{\Delta F_2}\right)_{\Delta F_1=0} = 1 \tag{A9}$$

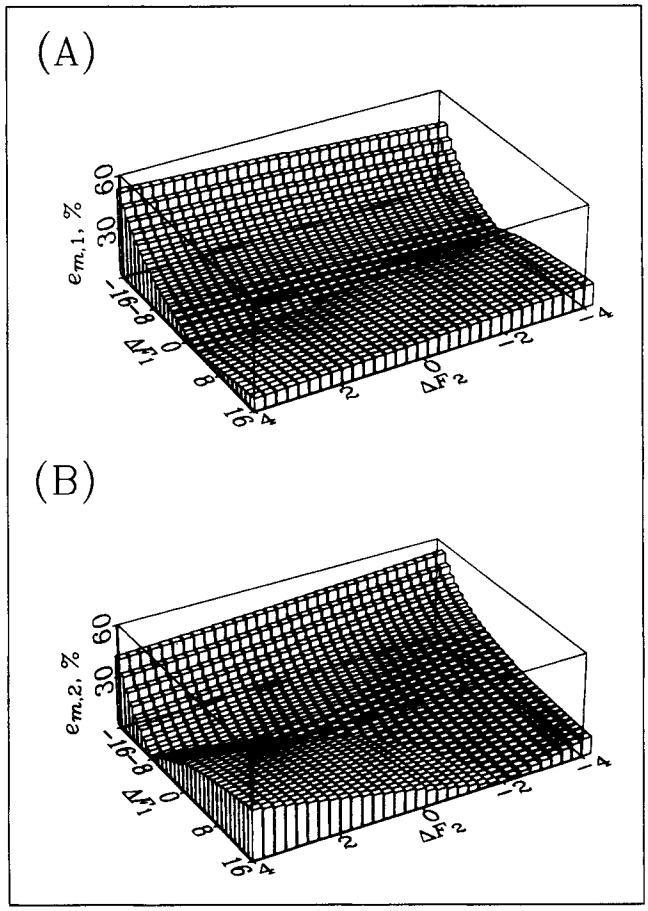

Figure A1. Consistency checking using independent identification in terms of multiplicative errors for: (A) $g_{11,CL}$ and (B) $g_{22,CL}$.

$$g_{22} = \left(\frac{\Delta x}{\Delta F_2}\right)_{\Delta F_1 = 0} = \frac{\bar{x}_2 - \bar{x}}{\bar{F} + \Delta F_2} \tag{A10}$$

From Eqs. A7 ~ A10, the closed-loop gains are:

$$g_{11,CL} = \frac{(\bar{x}_2 - \bar{x}_1)\bar{F} + (\bar{x}_2 - \bar{x})\Delta F_1 + (\bar{x} - \bar{x}_1)\Delta F_2}{(\bar{F} + \Delta F_1)(\bar{x}_2 - \bar{x})} \tag{A11}$$

$$g_{22,CL} = \frac{(\bar{x}_2 - \bar{x}_1)\bar{F} + (\bar{x}_2 - \bar{x})\Delta F_1 + (\bar{x} - \bar{x}_1)\Delta F_2}{(\bar{F} + \Delta F_1)(\bar{F} + \Delta F_2)} \tag{A12}$$

Equations A11 and A12 show that the closed-loop gains depend on the magnitude of changes in ΔF_i. Figure A1 shows the multiplicative error ($e_{m,i} = |(\hat{g}_{ii,CL} - g_{ii,CL})/g_{ii,CL}|$, where $\hat{g}_{ii,CL}$ is the estimated closed-loop gains and $g_{ii,CL}$ is the true closed-loop gain. The results show that the magnitude of error is increased when the changes in the input variables are increased.

Appendix B

Consider a 2×2 system under decentralized PI control. Assume that:

(1) $g_{ij}(s)$s are rational, strictly proper transfer functions with no RHP pole.

(2) The closed-loop system is stable along the tuning sequence (loop 1 tuned first, followed by loop 2 then back to loop 1).

(3) No pole-zero cancellation occurs in $g_{ii,CL}$.

If $\kappa(0) > 1$ (or $\lambda_{11} < 0$) and loop 1 is tuned first, then $g_{11,CL}(s)$ has at least one RHP pole.

Proof. Since loop 1 is tuned first (with loop 2 on manual) and the system is stable, this means $1 + g_{11}(s)k_1(s) = 0$ does not have RHP zero or, equivalently, h_1 does not have RHP pole.

Let g_{22} take the form:

$$g_{22}(s) = k_{p22}\frac{N(s)}{D(s)} \tag{B1}$$

where $N(s)$ and $D(s)$ are numerator and denominator polynomials with deg $(D(s)) >$ deg $(N(s))$ and (from the strictly proper assumption) $D(0) = N(0) = 1$. The second controller k_2 is designed according to:

$$g_{22,CL}(s) = g_{22}(s)[1 - \kappa(s)h_1(s)] \tag{B2}$$

Since $\kappa(0) > 1$ and $h_1(0) = 1$, it then becomes obvious that:

$$g_{22,CL}(0) = g_{22}(0)[1 - \kappa(0)] \tag{B3}$$

This indicates the sign of $g_{22,CL}$ is different from that of g_{22}. Since $g_{22,CL}$ does not have RHP pole (the poles of $g_{22,CL}$ are the poles of g_{22}, g_{12}, g_{21}, and h_1) and the closed-loop system $1 + g_{22,CL}k_2 = 0$ is stable, the controller gain K_{c2} has the same sign as that of $g_{22,CL}$:

$$g_{22,CL}(0)K_{c2} > 0 \tag{B4}$$

or

$$g_{22}(0)K_{c2} < 0 \text{ (or } k_{p22}K_{c2} < 0) \tag{B5}$$

Now back to loop 1 to design k_1 for $g_{11,CL}$. Since the poles of $g_{11,CL}$ are the poles of g_{11}, g_{12}, g_{21} and h_2. Consider the poles of h_2. The zeros of the closed-loop characteristic equation becomes:

$$1 + g_{22}K_{c2}\frac{\tau_I s + 1}{\tau_I s} = 0 \tag{B6}$$

$$1 + \frac{k_{p22}K_{c2}(\tau_I s + 1)}{\tau_I s}\frac{N(s)}{D(s)} = 0 \tag{B7}$$

$$\tau_I s D(s) + k_{p22}K_{c2}\tau_I s N(s) + k_{p22}K_{c2}N(s) = 0 \tag{B8}$$

Since $D(0) = 1$ and g_{22} is stable and strictly proper, the coefficient of the highest degree is positive. From $N(0) = 1$, the constant terms $k_{p22}K_{c2}$ is negative. It is obvious that the closed-loop characteristic equation has at least one RHP zero. This implies h_2 or $g_{11,CL}$ has at least one RHP pole.

Manuscript received Apr. 28, 1993, and revision received July 26, 1993.