4

MODEL-FREE ADAPTIVE CONTROL WITH CYBOCON

George S. Cheng

Model-Free Adaptive Control with CyboCon

In this chapter, we will introduce model-free adaptive (MFA) control. The concept of the model-free adaptive control theory and the related issues are discussed. Detailed information regarding the concept, architecture, and algorithms of model-free adaptive (MFA) control is being disclosed and discussed for the first time. The technology is protected by U.S. Patents 6,055,524, 6,360,131, and other pending patents.

Concept of MFA Control

The formal definition of model-free adaptive (MFA) control is given in this section. A model-free adaptive control system shall be defined to have at least the following properties or features:

- No precise quantitative knowledge of the process is available
- No process identification mechanism or identifier is included in the system
- No controller design for a specific process is needed
- No complicated manual tuning of controller parameters is required
- Closed-loop system stability analysis and criteria are available to guarantee the system stability

The essence of model-free adaptive control can be described with the discussions on the following five issues.

Process Knowledge Issue

Most advanced control techniques for designing control systems are based on a good understanding of the process and its environment. Laplace transfer functions or dynamic differential equations are usually used to represent the process dynamics.

In many process control applications, however, the dynamics may be too complex or the physical process is not well understood. Quantitative knowledge of the process is then not available. This is usually called a ‘‘black box’’ problem.

In many cases, we may have some knowledge of the process but are not sure whether the knowledge is accurate or not. In process control applications, we often deal with raw materials, wild inflows, unpredictable downstream demand changes, and frequent switches of product size, recipe, batch, and loads. These all lead to a common problem: that is, we are not sure if the process knowledge we have is accurate or not. This is usually called a ‘‘gray box’’ problem.

If quantitative knowledge of the process is available, we have a ‘‘white box’’ to deal with. It is a relatively simple task to design a controller for the process in this case because we can use existing, well-established control methods and tools based on the process knowledge.

Although model-free adaptive control can actually deal with black, gray, and white box problems, it is more suitable to deal with the gray box problem, since there is no need to apply a ‘‘no-model’’ control method when a process model is clear, and it is not a good idea to attack a black box problem without making the effort to understand the process.

Process Identification Issue

In most traditional adaptive control methods (model-reference or self-tuning), if the quantitative knowledge of the process is not available, an online or offline identification mechanism is usually required to obtain the process dynamics.

This contributes to a number of fundamental problems such as (i) the headache of offline training that might be required, (ii) the trade-off between the persistent excitation of signals for correct identification and the steady system response for control performance, (iii) the assumption of the process structure, the model convergence, and (iv) system stability issues in real applications.

The main reason why identification-based control methods are not well suited to process control is that control and identification are always a conflict. Good control will lead to a steady state where the key variables setpoint, controller output, and process variable will show straight lines on a trend chart. Since any stable system can reach a steady state, the three straight lines will not have information about the process characteristics. On the other hand, good identification requires persistent excitation of controller output and process variable. That is why identification-based control methods have a tough time being accepted by plant operators, if insertion of noise and disturbances is required to keep the process model updated. Operators just do not like someone else touching their processes unless it is mandatory.

MFA control avoids these fundamental problems by not using any identification mechanism in the system. Once an MFA controller is launched, it will take over control immediately. The algorithms used in MFA controllers to update the weighting factors are based on a sole objective, which is to minimize the error between the setpoint and process variable. That means, when the process is in a steady state where error is close to zero, there is no need to update the MFA controller weighting factors.

Controller Design Issue

One of the main reasons why PID is still so popular in process control is that no complicated controller design procedures are needed to use it. Designing an advanced controller usually requires special expertise and experience. That is one of the main reasons that advanced control methods are not widely used in process control applications.

MFA controllers are developed to be general-purpose controllers. In fact, a series of MFA controllers are designed to control a variety of problematic industrial loops: for instance, SISO MFA to control the processes that PID has difficulty with; nonlinear MFA to control extremely nonlinear processes such as pH or high-pressure loops, anti-delay MFA to control processes with large and varying time delays; MIMO MFA to control multivariable processes; feedforward MFA to deal with large measurable disturbances; and robust MFA to force the process variable to stay within defined bounds.

For a user, there really are no controller design procedures required. One can simply configure the controller with certain parameters and the MFA controller is ready to be launched. This is one of the major differences between a model-free adaptive controller and other model-based advanced controllers.

Controller Parameter Tuning Issue

An adaptive controller should not need to be manually tuned. This is also true of the model-free adaptive controller. However, from an engineering point of view, we have left a tuning parameter open to the user for the purpose of adjusting control performance. Since the MFA controller gain K_c is the only tuning parameter and there is no complex combination of tuning parameters, the user can simply increase the gain to make the controller work more actively and decrease it to detune the controller.

System Stability Issue

Closed-loop control system stability analysis is always an important issue because it determines whether the controller will be practically useful. When the closed-loop system stability criterion is available, one can use the criterion to decide whether the control system can be safely put in operation.

As shown in Figure 4.1, a traditional internal model based adaptive control system has three major components: controller, process, and model.

Here "model" refers to a mathematical representation that can describe the relationship between the process input and output. Self-tuning adaptive control systems typically have this architecture.

The model is usually built by an identification mechanism to minimize the model error E_m, which is the difference between the model output and the process output. The identification mechanism has a learning algorithm to minimize such model error using the process input and output data.

Stability of Model-Based Control System

The stability of the overall closed system is related to the process, the controller, and the identifier in the following manner:

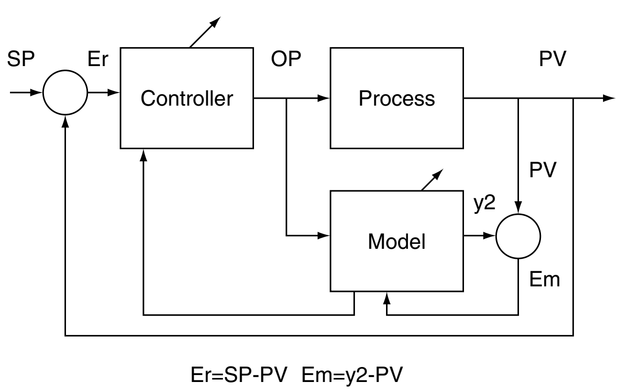

Figure 4.1 Internal model-based adaptive control system.

- The stability of the process is usually assumed or we say the process is open-loop stable
- The stability of the control loop must be guaranteed by the convergence of the identifier
- But the convergence of the identifier is dependent on the stability and persistent excitation of signals originating from the control loop

This is a circular argument that it is difficult to resolve. Therefore, many practically useful adaptive controllers are running in a semi-online and semi-offline fashion. That is, although the learning algorithm could be a recursive one that can be implemented in an online fashion, it is not turned on all the time to avoid poor identification results. In a smooth control situation, the system does not provide sufficient excitation signals for good identification.

Since the model-free adaptive controller does not have an identification mechanism, the stability of the overall closed-loop system is much relaxed compared to identification-based control methods. The stability criterion of a MFA control system was derived during the development of model-free adaptive control theory and will be discussed in the following section.

SINGLE-LOOP MFA CONTROL SYSTEM

Figure 4.2 illustrates a single-variable model-free adaptive control system. The structure of the system is as simple as a traditional single-loop control system. It includes a single-input-single-output (SISO) process, an MFA controller, and a feedback loop.

The signals shown in the figure are as follows:

$r(t)$—Setpoint (SP)
$y(t)$—Process variable (PV), $y(t) = x(t) + d(t)$
$x(t)$—Process output, $x(t)$
$u(t)$—Controller output, OP

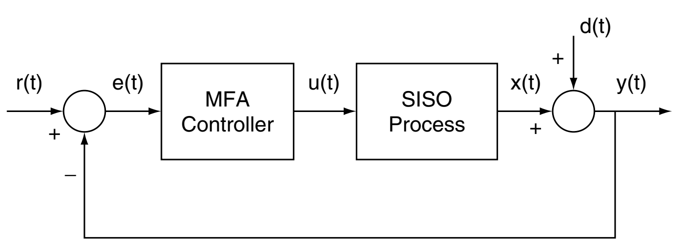

Figure 4.2 Single-loop MFA control system.

$d(t)$—Disturbance caused by noise or load changes
$e(t)$—Error between SP and PV, $e(t) = r(t) - y(t)$

SISO MFA Control Objective

The model-free adaptive controller is an online real-time regulatory controller. Its control objective is to make the process variable $y(t)$ track the given trajectory of its setpoint $r(t)$ under variations of setpoint, disturbance, and process dynamics. In other words, the task of the MFA controller is to minimize the error $e(t)$ in an online fashion.

We select the objective function for MFA control system as

$$
\begin{aligned}
E_s(t) &= \frac{1}{2}e(t)^2 \\
&= \frac{1}{2}[r(t) - y(t)]^2
\end{aligned}
\tag{4.1}
$$

The minimization of $E_S(t)$ is achieved by (i) the regulatory control capability of the MFA controller, whose output manipulates the manipulated variable forcing the process variable $y(t)$ to track its setpoint $r(t)$; and (ii) the adjustment of the MFA controller weighting factors that allow the controller to deal with the dynamic changes, large disturbances, and other uncertainties of the control system.

SISO MFA Controller Architecture

Figure 4.3 illustrates the architecture of a SISO MFA controller. A multilayer perceptron (MLP) artificial neural network (ANN) is adopted in the design of the controller. The ANN has one input layer, one hidden layer with N neurons, and one output layer with one neuron.

The input signal $e(t)$ to the input layer is converted to a normalized error signal E_1 with a range of -1 to 1 by using the normalization unit, where $N(.)$ denotes a normalization function. The E_1 signal then goes through a series of delay units iteratively, where z^{-1} denotes the unit delay operator. A set of normalized error signals E_2 to E_N is then generated. In this way, a continuous signal $e(t)$ is converted to a series of discrete signals, which are used as the inputs to the ANN. These delayed error signals E_i, $i = 1, \ldots N$, are then conveyed to the hidden layer through the neural network connections. It is equivalent to adding a feedback structure to the neural network. Then the regular static multilayer perceptron becomes a dynamic neural network, which is a key component for the model-free adaptive controller.

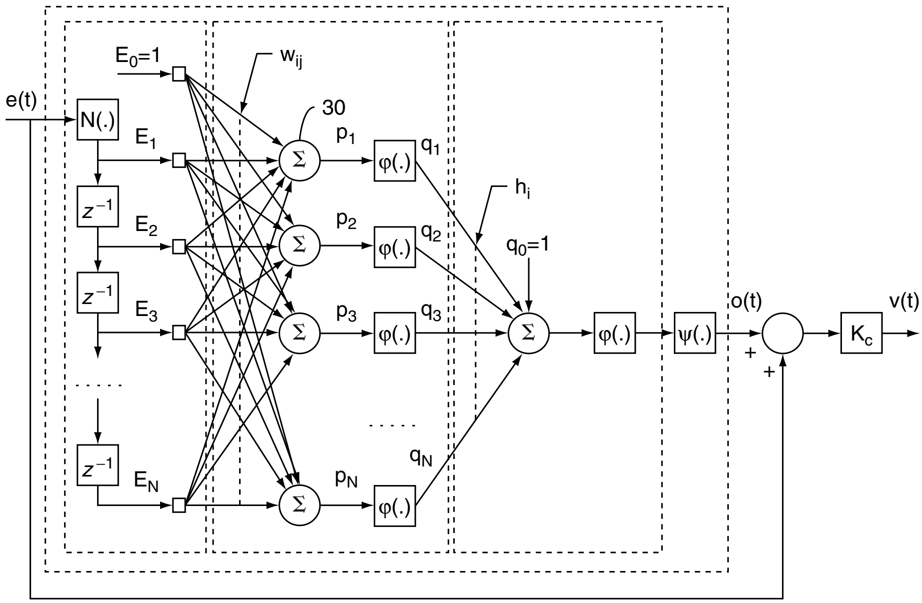

Figure 4.3 Architecture of a SISO MFA controller.

A model-free adaptive controller requires a dynamic block such as a dynamic neural network as its key component. A dynamic block is just another name for a dynamic system, whose inputs and outputs have dynamic relationships.

Each input signal is conveyed separately to each of the neurons in the hidden layer via a path weighted by an individual weighting factor w_{ij}, where $i = 1, 2, \ldots, N$, and $j = 1, 2, \ldots, N$. The inputs to each of the neurons in the hidden layer is summed by adder with $E_0 = 1$, the threshold signal for the hidden layer, through the constant weights $W_{0j} = 1$ to produce signal p_j. Then the signal p_j is filtered by an activation function to produce q_j, where j denotes the jth neuron in the hidden layer.

A sigmoidal function $\varphi(.)$ mapping real numbers to (0,1) defined by

$$\varphi(x) = \frac{1}{1 + e^{-x}} \tag{4.2}$$

is used as the activation function in the ANN.

Each output signal from the hidden layer is conveyed to the single neuron in the output layer via a path weighted by an individual weighting factor h_i, where $i = 1, 2, \ldots, N$. These signals are summed in adder with $h_0 = 1$, the threshold signal for the output layer, and then filtered by activation function. A function defined by

$$\psi(y) = \ln \frac{y}{1-y} \tag{4.3}$$

maps the range of the output layer from (0,1) back into the real space to produce the output $o(t)$ of the artificial neural network.

SISO MFA Control Algorithm

The algorithm governing the input/output of the controller consists of the following difference equations:

$$p_j(n) = \sum_{i=1}^{N} w_{ij}(n)E_i(n) + 1 \tag{4.4}$$

$$q_j(n) = \varphi(p_j(n)) \tag{4.5}$$

$$\begin{aligned} o(n) &= \psi[\varphi(\sum_{j=1}^{N} h_j(n)q_j(n) + 1)] \\ &= \sum_{j=1}^{N} h_j(n)q_j(n) + 1 \end{aligned} \tag{4.6}$$

$$v(t) = K_c[o(t) + e(t)] \tag{4.7}$$

where n denotes the nth iteration, $o(t)$ is the continuous function of $o(n)$, $v(t)$ is the output of the model-free adaptive controller, and $K_c > 0$, called controller gain, is a constant used to adjust the magnitude of the controller. This constant is useful to fine tune the controller performance or keep the system in stable range.

An online learning algorithm is developed to continuously update the values of the weighting factors of the MFA controller as follows:

$$\Delta w_{ij}(n) = \eta K_c \frac{\partial y(n)}{\partial u(n)} e(n)q_j(n)(1 - q_j(n))E_i(n) \sum_{k=1}^{N} h_k(n) \tag{4.8}$$

$$\Delta h_j(n) = \eta K_c \frac{\partial y(n)}{\partial u(n)} e(n)q_j(n) \tag{4.9}$$

where $\eta > 0$ is the learning rate, and the partial derivative $\partial y(n)/\partial u(n)$ is the gradient of $y(t)$ with respect to $u(t)$, which represents the sensitivity of the output $y(t)$ to variations of the input $u(t)$. It is convenient to define

$$S_f(n) = \frac{\partial y(n)}{\partial u(n)} \tag{4.10}$$

as the sensitivity function of the process.

Since the process is unknown, the sensitivity function is also unknown. This is the classical "black box" problem that has to be resolved in order to make the algorithm useful.

Through the stability analysis of the model-free adaptive control, it was found that if the process under control is open-loop stable, controllable, and its acting type does not change during the whole period of control, bounding $S_f(n)$ with a set of arbitrary nonzero constants can guarantee the system to be bounded-input-bounded-output (BIBO) stable.

This study implies that the process sensitivity function $S_f(n)$ can be simply replaced by a constant; no special treatment for $S_f(n)$ or any detailed knowledge of the process are required in the learning algorithm of the model-free adaptive controller. By selecting $S_f(n) = 1$, the resulting learning algorithm is as follows:

$$\Delta w_{ij}(n) = \eta K_c e(n) q_j(n)(1 - q_j(n)) E_i(n) \sum_{k=1}^{N} h_k(n) \tag{4.11}$$

$$\Delta h_j(n) = \eta K_c e(n) q_j(n) \tag{4.12}$$

Equations (4.1) through (4.12) work for both process direct-acting and reverse acting types. Direct-acting means that an increase in process input will cause its output to increase, and vice versa. Reverse-acting means that an increase in process input will cause its output to decrease, and vice versa. To keep the foregoing equations working for both direct and reverse acting cases, $e(t)$ needs to be calculated differently based on the acting type of the process as follows:

$$e(t) = r(t) - y(t), \text{ if direct acting} \tag{4.13a}$$

$$e(t) = -[r(t) - y(t)], \text{ if reverse acting} \tag{4.13b}$$

This is a general treatment for the process acting types. It applies to all model-free adaptive controllers to be introduced later.

SISO MFA Control System Stability Criterion

MFA Control System Stability Criterion

A sufficient condition of MFA control stability can be described as follows: If process is passive, the closed-loop MFA control system stability is guaranteed and the process can be linear, nonlinear, time-invariant, time-varying, single-variable, and multivariable.

This means if the process is open-loop stable, we can guarantee the closed-loop system stability of a SISO MFA control system. Of course, proper setting of its key parameters such as the controller gain K_c is required. In later sections, we will discuss the idea of MFA being a robust controller as well. Since sensitivity and robustness is always a conflict, we designed the MFA controllers to provide adequate sensitivity and robustness. That means MFA is sensitive enough to the variations of its key parameters such as gain K_c. In the mean time, it is robust enough to deal with the uncertainties of process dynamics, load changes, etc., with a much larger robust range than a conventional controller such as PID.

In that sense, MFA can be quite easily placed at its nominal position, and the system stability is guaranteed as long as the uncertainties are within a reasonable range.

Since the criterion is a sufficient condition, for nonpassive processes, we will not know its stability behavior, although the control system might still work. For instance, a nonself-regulating level loop can be easily controlled by an MFA controller. This type of level loop has the integral behavior and is open-loop unstable.

In general, if the process is not open-loop stable, we cannot guarantee the closed-loop system stability. It is always a good idea to stabilize the process first before applying a feedback controller.

Stability Analysis Methods

Two of the most widely used approaches to address stability problems in nonlinear control systems are (i) the stability theory of Lyapunov, and (ii) the input/output stability theory based on functional analysis techniques. Lyapunov stability theory considers stability as an internal property of the system. It basically deals with the effect of momentary perturbations resulting in changes in initial conditions. On the other hand, input/output stability theory, as its name suggests, considers the effect of external inputs to the system.

Lyapunov stability usually deals with a dynamical system described by a nonlinear differential equation in the form of

$$\dot{x}(t) = f(x, t) \tag{4.14}$$

where $x(t_0) = x_0$.

The direct method of Lyapunov enables one to determine whether or not the equilibrium state of a dynamical system in the form in Equation (4.14) is stable without explicitly finding the solution of the nonlinear differential equation. The method has proved effective in the stability analysis of nonlinear differential equations whose solutions are difficult to obtain. It involves finding a suitable scalar function $V(x,t)$ (Lyapunov function) and examining its time derivative $\dot{V}(x,t)$ along the trajectories of the system. The reasoning behind this method is quite simple. In a purely dissipative system, the energy stored in the system is always positive, and time derivative of the energy is nonpositive.

For practical reasons, however, Lyapunov stability analysis methods are not suitable for adaptive systems in many cases, because these methods mainly study the local property of a dynamical system. The definition of Lyapunov stability says that if the state starts sufficiently close to an equilibrium point, the Lyapunov stability will guarantee that the trajectories remain arbitrarily close to the equilibrium point. In adaptive systems, we do not have any control over how close the initial conditions are to the equilibrium values. In addition, in most circumstances, there are no fixed equilibrium points because of the changes in system parameters, etc.

In contrast to the local property of the Lyapunov stability, the input/output stability theory based on functional analysis techniques mainly concerns the input/output properties of nonlinear feedback systems. It is suitable for the stability problems in adaptive systems. The functional analysis approach is also more general. Distributed and lumped systems, discrete and continuous time systems, and single-input-single-output and multi-input-multi-output systems can be treated in a unified fashion.

The concept and important results of input/output stability theory and passivity theory can be found in Desoer and Vidyasagar (1975) and Anderson et al. (1986). We were able to prove that the MFA controller described in this section is passive. Based on the input/output stability theory and passivity theory, we were able to derive a set of theorems as roughly stated in the MFA control system stability criterion. Notice that this is a global stability criterion for a nonlinear control system. Because of the length of this chapter, actual derivations and proofs of this criterion are not included.

MFA Control System Requirements

As a feedback control system, MFA requires the process to have the following behavior:

- The process is controllable
- The process is open-loop stable
- The process is either direct or reverse acting (the process does not change its sign)

If the process is not controllable, one needs to improve the process structure or its variable pairing. This effort might result in a controllable process.

If the process is not open-loop stable, it is always a good practice to stabilize it first. However, for certain simple open-loop unstable processes such as a nonself-regulating level loop, no special treatment is required for MFA to control.

If a process changes its sign within its operating range, it is still possible to design a special controller for this "ill" process. But, it will not be a general-purpose MFA controller.

SISO MFA Configuration

To show how a practical MFA controller is configured, we use a SISO MFA controller configuration screen from CyboCon MFA control software (Figure 4.4). Some of the key fields in this menu are described in the following:

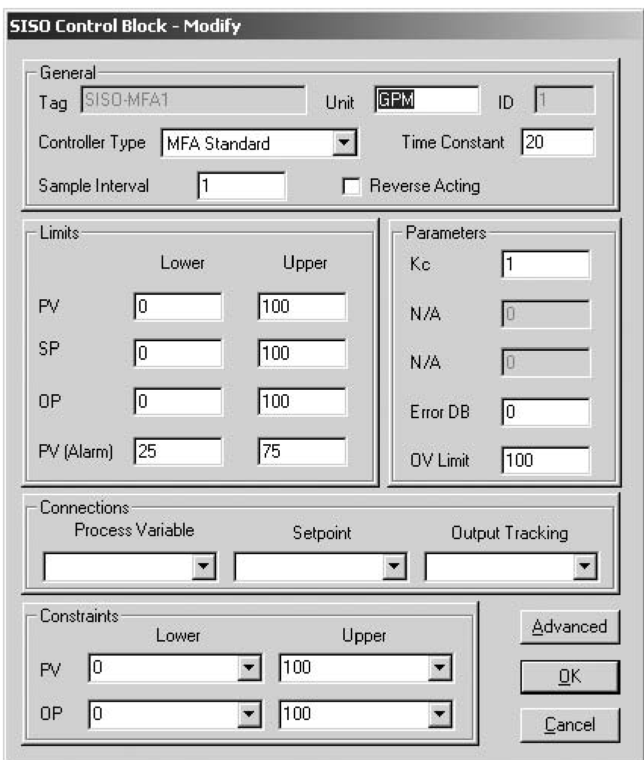

Figure 4.4 SISO MFA controller configuration screen.

- *Time Constant*. A rough estimate of the time constant of the process. Unit: seconds. Range: 0.003 to 99999 sec. Default setting: 20 sec. It is not difficult to estimate the time constant if you know the process.
- *Sample Interval*. The interval between two samples or calculations. Unit: seconds. Range: 0.001 to 999.9 sec. Default setting: 1 sec. According to the principles in the information theory, it is required that the sample interval be less than or equal to one-third of the time constant. That is,

$$T_s \leq (1/3)T_c \tag{4.15}$$

  where T_s is the sample interval, and T_c is the time constant. Notice that the sample interval can be set as small as 0.001 sec, which is 1 ms. CyboCon HS high-speed MFA control software can support this rate for calculation and update of controller output as well as MFA adaptive weighting factors.
- *Reverse Acting*. The process acting type. It is very important to check this field if the process is reverse acting. That means, if the process input increases, the process output decreases, and vice versa.
- K_c. MFA controller gain. Used to adjust the control performance. Set it higher for a more active control action, and set it lower for less overshoot.
- *Advanced Button*. Press this button to enter the menu for configuring feedforward MFA, MFA pH, robust MFA, and nonlinear MFA controllers.

SISO MFA Control Application Guide

MFA Controller Gain

The MFA controller gain K_c is designed to compensate for (i) too large or too small a static gain due to improper scaling; and (ii) severe nonlinear behavior of the process. K_c should be set based on the static gain or the nominal static gain you estimate. The rule of thumb is to make $K_cK \approx 1$.

Based on this general setting, you can adjust the controller gain to fine-tune the control performance—usually, increasing the gain to speed up the control action or decreasing the gain to slow it down. For instance, if you want to see more overshoot, you should let $K_cK > 1$. If you want to see less overshoot, you should let $K_cK < 1$.

Adjusting Time Constant

The time constant in the MFA controller configuration is to provide some qualitative process dynamic information to the controller. It can be just a rough estimate. However, if you feel that the control performance is not as good as expected, you can fine tune the system by adjusting the time constant based on the following rule of thumb.

1. If a controller is reacting to a setpoint change too quickly causing a large overshoot, increase the time constant setting.
2. If a controller is not reacting quickly enough to a setpoint change causing too slow a process response, reduce the time constant setting.

Procedure for Launching MFA Controller

It is easy to start up a new control loop with a SISO MFA controller. Once proper wiring and scaling of setpoint (SP), process variable (PV), and output (OP) are completed, and the estimated process time constant, acting type, and MFA controller gain are entered, the MFA controller can be switched from manual to automatic. The MFA will take control immediately with no bump to the process whether the current process variable is tracking its setpoint or not. In configuring the MFA control system, the controller output should track the current valve position when the controller is in the manual mode. In this way, the MFA is always ready to be switched to the automatic mode without a bump to the process.

Multivariable MFA Control System

Figure 4.5 illustrates a multivariable feedback control system with a model-free adaptive controller. The system includes a multi-input-multi-output (MIMO) process, a set of controllers, and a set of signal adders, respectively, for each control loop.

The inputs **e(t)** to the controller are presented by comparing the setpoints **r(t)** with the process variables **y(t)**, which are the process responses to controller outputs **u(t)** and the disturbance signals **d(t)**. Since it is a multivariable system, all the signals here are vectors represented in bold type as follows.

$$\mathbf{r(t)} = [r_1(t),\ r_2(t),\ \ldots,\ r_N(t)]^T \tag{4.16a}$$

$$\mathbf{e(t)} = [e_1(t),\ e_2(t),\ \ldots,\ e_N(t)]^T \tag{4.16b}$$

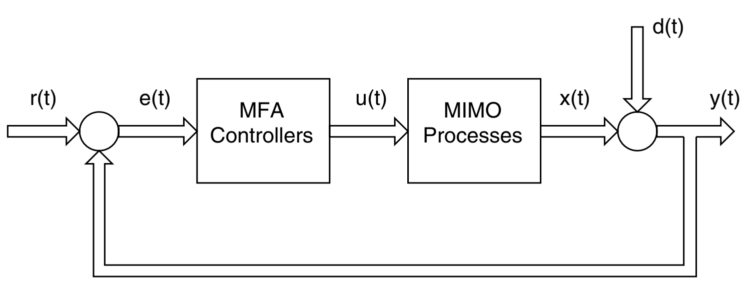

Figure 4.5 Multivariable MFA control system.

$$\mathbf{u(t)} = [u_1(t),\ u_2(t),\ \ldots,\ u_N(t)]^T \tag{4.16c}$$

$$\mathbf{y(t)} = [y_1(t),\ y_2(t),\ \ldots,\ y_N(t)]^T \tag{4.16d}$$

$$\mathbf{d(t)} = [d_1(t),\ d_2(t),\ \ldots,\ d_N(t)]^T \tag{4.16e}$$

where superscript T denotes the transpose of the vector, and subscript N denotes the total element number of the vector.

2-Input-2-Output MFA Control System

Without losing generality, we will show how a multivariable model-free adaptive control system works with a 2-input-2-output (2 × 2) system as illustrated in Figure 4.6, which is the 2 × 2 arrangement of Figure 4.5. In the 2 × 2 MFA control system, the MFA controller set consists of two controllers C_{11}, C_{22}, and two compensators C_{21} and C_{12}. The process has four subprocesses G_{11}, G_{21}, G_{12}, and G_{22}.

The measured process variables y_1 and y_2 are used as the feedback signals of the main control loops. They are compared with the setpoints r_1 and r_2 to produce errors e_1 and e_2. The output of each controller associated with one of the inputs e_1 or e_2 is combined with the output of the compensator associated with the other input to produce control signals u_1 and u_2. The output of each subprocess is cross-added to produce measured process variables y_1 and y_2. Notice that in real applications the outputs from the subprocesses are not measurable and only their combined signals y_1 and y_2 can be measured. Thus, by the nature of the 2 × 2 process, the inputs u_1 and u_2

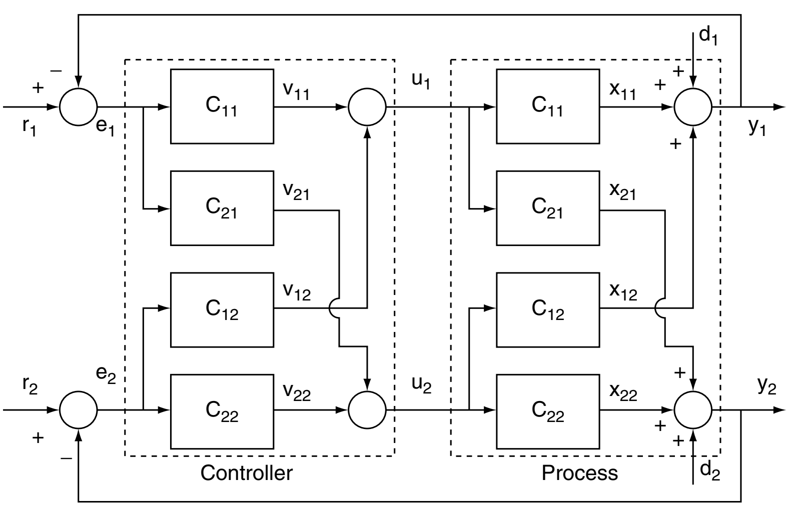

Figure 4.6 2-input-2-output MFA control system.

to the process are interconnected with its outputs y_1 and y_2. The change in one input will cause both outputs to change.

In this 2×2 system, the element number N in Equation (4.16) is equal to 2 and the signals shown in Figure 4.6 are as follows:

$r_1(t)$, $r_2(t)$—Setpoint of controllers C_{11} and C_{22}, respectively
$e_1(t)$, $e_2(t)$—Error between the setpoint and process variable
$v_{11}(t)$, $v_{22}(t)$—Output of controller C_{11} and C_{22}, respectively
$v_{21}(t)$, $v_{12}(t)$—Output of compensators C_{21} and C_{12}, respectively
$u_1(t)$, $u_2(t)$—Inputs to the process, or the outputs of the 2×2 controller set
$x_{11}(t)$, $x_{21}(t)$, $x_{12}(t)$, $x_{22}(t)$—Output of process G_{11}, G_{21}, G_{12}, and G_{22}, respectively
$d_1(t)$, $d_2(t)$—Disturbance to y_1 and y_2, respectively
$y_1(t)$, $y_2(t)$—Process variables of the 2×2 process

The relationship between these signals is as follows:

$$e_1(t) = r_1(t) - y_1(t) \tag{4.17a}$$

$$e_2(t) = r_2(t) - y_2(t) \tag{4.17b}$$

$$y_1(t) = x_{11}(t) + x_{12}(t) \tag{4.17c}$$

$$y_2(t) = x_{21}(t) + x_{22}(t) \tag{4.17d}$$

$$u_1(t) = v_{11}(t) + v_{12}(t) \tag{4.17e}$$

$$u_2(t) = v_{21}(t) + v_{22}(t) \tag{4.17f}$$

2 × 2 MFA Control Objective

The control objectives for this 2×2 MFA control system are to produce control outputs $u_1(t)$ and $u_2(t)$ to manipulate their manipulated variables so that the process variables $y_1(t)$ and $y_2(t)$ will track their setpoints $r_1(t)$ and $r_2(t)$, respectively.

The minimization of $e_1(t)$ and $e_2(t)$ is achieved by (i) the regulatory control capability of the MFA controllers, whose outputs manipulate the manipulated variables, forcing the process variable $y(t)$ to track its setpoints $r_1(t)$ and $r_2(t)$; (ii) the decoupling capability of the MFA compensators, whose outputs are added to the MFA controller outputs to compensate for the interactions from the other subprocess; and (iii) the adjustment of the MFA controller weighting factors that allow the controllers to deal with the dynamic changes, large disturbances, and other uncertainties of the control system.

Since these two loops are interacting to each other, achieving the control objectives is not easy in comparison to a single-loop control system.

2×2 MFA Control Algorithm

The controllers C_{11} and C_{22} have the same structure as the SISO MFA controller shown in Figure 4.3. The input and output relationship in these controllers is represented by the following equations:

For controller C_{11}:

$$p_j^{11}(n) = \sum_{i=1}^{N} w_{ij}^{11}(n)E_i^{11}(n) + 1 \tag{4.18}$$

$$q_j^{11}(n) = \varphi(p_j^{11}(n)) \tag{4.19}$$

$$v_{11}(n) = K_c^{11}\left[\sum_{j=1}^{N} h_j^{11}(n)q_j^{11}(n) + 1 + e_1(n)\right] \tag{4.20}$$

$$\Delta w_{ij}^{11}(n) = \eta^{11}K_c^{11}e_1(n)q_j^{11}(n)(1 - q_j^{11}(n))E_i^{11}(n)\sum_{k=1}^{N} h_k^{11}(n) \tag{4.21}$$

$$\Delta h_j^{11}(n) = \eta^{11}K_c^{11}e_1(n)q_j^{11}(n) \tag{4.22}$$

For controller C_{22}:

$$p_j^{22}(n) = \sum_{i=1}^{N} w_{ij}^{22}(n)E_i^{22}(n) + 1 \tag{4.23}$$

$$q_j^{22}(n) = \varphi(p_j^{22}(n)) \tag{4.24}$$

$$V_{22}(n) = K_c^{22}\left[\sum_{j=1}^{N} h_j^{22}(n)q_j^{22}(n) + 1 + e_2(n)\right] \tag{4.25}$$

$$\Delta w_{ij}^{22}(n) = \eta^{22}K_c^{22}e_2(n)q_j^{22}(n)(1 - q_j^{22}(n))E_i^{22}(n)\sum_{k=1}^{N} h_k^{22}(n) \tag{4.26}$$

$$\Delta h_j^{22}(n) = \eta^{22}K_c^{22}e_2(n)q_j^{22}(n) \tag{4.27}$$

In these equations, $\eta^{11} > 0$ and $\eta^{22} > 0$ are the learning rate, and $K_c^{11} > 0$ and $K_c^{22} > 0$ are the controller gain for C_{11} and C_{22}, respectively. $E_i^{11}(n)$ is the delayed error signal of $e_1(n)$ and $E_i^{22}(n)$ is the delayed error signal of $e_2(n)$.

The architecture of the compensators C_{12} and C_{21} is shown in Figure 4.7. This architecture differs from the structure of the SISO MFA controller of Figure 4.3 in that no error signal is added to the neural network output $o(t)$.

The input and output relationship in these compensators is represented by the following equations:

For compensator C_{21}:

$$p_j^{21}(n) = \sum_{i=1}^{N} w_{ij}^{21}(n) E_i^{21}(n) + 1 \tag{4.28}$$

$$q_j^{21}(n) = \varphi(p_j^{21}(n)), \tag{4.29}$$

$$v_{21}(n) = K_s^{21} K_c^{21} \left[\sum_{j=1}^{N} h_j^{21}(n) q_j^{21}(n) + 1 \right] \tag{4.30}$$

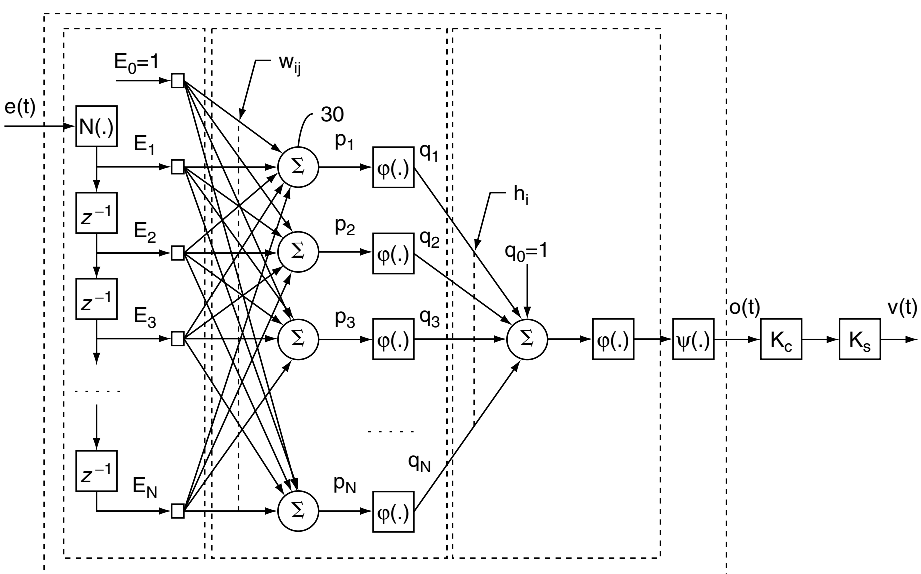

Figure 4.7 Architecture of a MIMO MFA compensator.

$$\Delta w_{ij}^{21}(n) = \eta^{21} K_c^{21} e_1(n) q_j^{21}(n)(1 - q_j^{21}(n)) E_i^{21}(n) \sum_{k=1}^{N} h_k^{21}(n) \tag{4.31}$$

$$\Delta h_j^{21}(n) = \eta^{21} K_c^{21} e_1(n) q_j^{21}(n) \tag{4.32}$$

For compensator C12:

$$p_j^{12}(n) = \sum_{i=1}^{N} w_{ij}^{12}(n) E_i^{12}(n) + 1 \tag{4.33}$$

$$q_j^{12}(n) = \varphi(p_j^{12}(n)) \tag{4.34}$$

$$v_{12}(n) = K_s^{12} K_c^{12} \left[\sum_{j=1}^{N} h_j^{12}(n) q_j^{12}(n) + 1 \right] \tag{4.35}$$

$$\Delta w_{ij}^{12}(n) = \eta^{12} K_c^{12} e_2(n) q_j^{12}(n)(1 - q_j^{12}(n)) E_i^{12}(n) \sum_{k=1}^{N} h_k^{12}(n) \tag{4.36}$$

$$\Delta h_j^{12}(n) = \eta^{12} K_c^{12} e_2(n) q_j^{12}(n) \tag{4.37}$$

In these equations, $\eta^{21} > 0$ and $\eta^{12} > 0$ are the learning rate, and $K_c^{21} > 0$ and $K_c^{12} > 0$ are the controller gain, for C_{21} and C_{12} respectively. $E_{\mathrm{i}}^{21}(n)$ is the delayed error signal of $e_1(n)$ and $E_{\mathrm{i}}^{12}(n)$ is the delayed error signal of $e_2(n)$.

The compensator sign factors K_s^{21} and K_s^{12} are a set of constants relating to the acting types of the process as follows:

$$K_s^{21} = 1, \text{ if } G_{22} \text{ and } G_{21} \text{ have different acting types} \tag{4.38a}$$

$$K_s^{21} = -1, \text{ if } G_{22} \text{ and } G_{21} \text{ have the same acting type} \tag{4.38b}$$

$$K_s^{12} = 1, \text{ if } G_{11} \text{ and } G_{12} \text{ have different acting types} \tag{4.38c}$$

$$K_s^{12} = -1, \text{ if } G_{11} \text{ and } G_{12} \text{ have the same acting type} \tag{4.38d}$$

These sign factors are needed to ensure that the MFA compensators produce signals in the correct direction so that the disturbances caused by the coupling factors of the multivariable process can be reduced.

N-Input-N-Output MFA Control System

A 3×3 multivariable model-free adaptive control system is illustrated in Figure 4.8 with a signal flow chart.

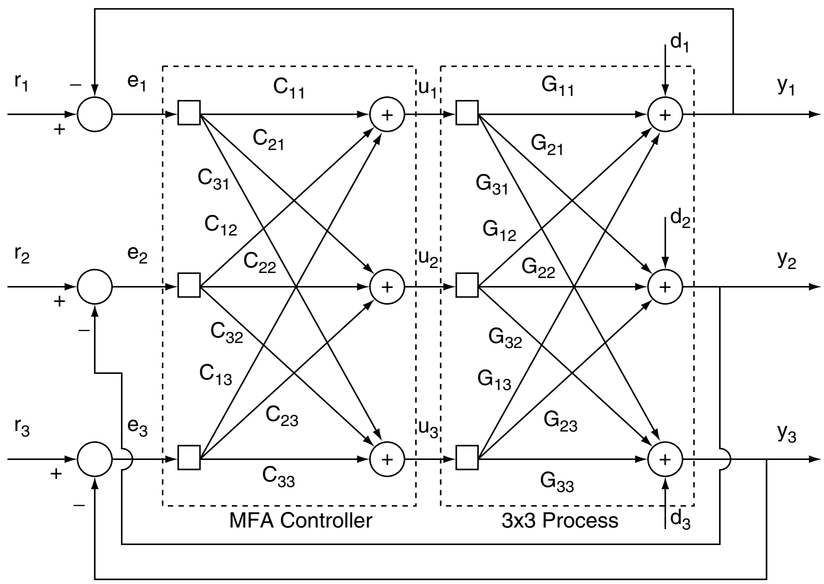

Figure 4.8 3-input-3-output MFA control system.

In the 3 × 3 MFA control system, the MFA controller set consists of three controllers C_{11}, C_{22}, C_{33}, and six compensators C_{21}, C_{31}, C_{12}, C_{32}, C_{13}, and C_{23}. The process has nine subprocesses G_{11} through G_{33}. The process variables y_1, y_2, and y_3 are used as the feedback signals of the main control loops. They are compared with the setpoints r_1, r_2, and r_3 to produce errors e_1, e_2, and e_3. The output of each controller associated with one of the inputs e_1, e_2, or e_3 is combined with the output of the compensators associated with the other two inputs to produce control signals u_1, u_2, and u_3.

Without losing generality, a set of equations that apply to an arbitrary $N \times N$ multivariable model-free adaptive control system is given in the following. If $N = 3$, it applies to the above-stated 3 × 3 MFA control system.

For controller C_{ll}:

$$p_j^{ll}(n) = \sum_{i=1}^{N} w_{ij}^{ll}(n) E_i^{ll}(n) + 1 \tag{4.39}$$

$$q_j^{ll}(n) = \varphi(p_j^{ll}(n)) \tag{4.40}$$

$$v_{ll}(n) = K_c^{ll}\left[\sum_{j=1}^{N} h_j^{ll}(n) q_j^{ll}(n) + 1 + e_l(n)\right] \tag{4.41}$$

$$\Delta w_{ij}^{ll}(n) = \eta^{ll} K_c^{ll} e_l(n) q_j^{ll}(n)(1 - q_j^{ll}(n)) E_i^{ll}(n) \sum_{k=1}^{N} h_k^{ll}(n) \tag{4.42}$$

$$\Delta h_j^{ll}(n) = \eta^{ll} K_c^{ll} e_l(n) q_j^{ll}(n) \tag{4.43}$$

where $l = 1,\ 2, \ldots, N$.

For compensator C_{lm}:

$$p_j^{lm}(n) = \sum_{i=1}^{N} w_{ij}^{lm}(n) E_i^{lm}(n) + 1 \tag{4.44}$$

$$q_j^{lm}(n) = \varphi(p_j^{lm}(n)) \tag{4.45}$$

$$v_{lm}(n) = K_s^{lm} K_c^{lm} \left[\sum_{j=1}^{N} h_j^{lm}(n) q_j^{lm}(n) + 1 \right] \tag{4.46}$$

$$\Delta w_{ij}^{lm}(n) = \eta^{lm} K_c^{lm} e_m(n) q_j^{lm}(n)(1 - q_j^{lm}(n)) E_i^{lm}(n) \sum_{k=1}^{N} h_k^{lm}(n) \tag{4.47}$$

$$\Delta h_j^{lm}(n) = \eta^{lm} K_c^{lm} e_m(n) q_j^{lm}(n) \tag{4.48}$$

where $l = 1,\ 2, \ldots N$; $m = 1,\ 2,\ \ldots, N$; and $l \neq m$.

In these equations, $\eta^{ll} > 0$ and $\eta^{lm} > 0$ are the learning rate, $K_c^{ll} > 0$ and $K_c^{lm} > 0$ are the controller gain, for C_{ll} and C_{lm}, respectively. $E_i^{ll}(n)$ is the delayed error signal of $e_l(n)$ and $E_i^{lm}(n)$ is the delayed error signal of $e_m(n)$.

K_s^{lm} is the sign factor for the MFA compensator, which is selected based on the acting types of the subprocesses as follows:

$$K_s^{lm} = 1, \text{ if } G_{ll} \text{ and } G_{lm} \text{ have different acting types} \tag{4.49a}$$

$$K_s^{lm} = -1, \text{ if } G_{ll} \text{ and } G_{lm} \text{ have the same acting type} \tag{4.49b}$$

where $l = 1,\ 2, \ldots, N$; $m = 1,\ 2, \ldots, N$; and $l \neq m$.

MIMO MFA Configuration

To show how a practical multivariable MFA controller is configured, we use a MIMO MFA controller configuration screen from CyboCon MFA control software

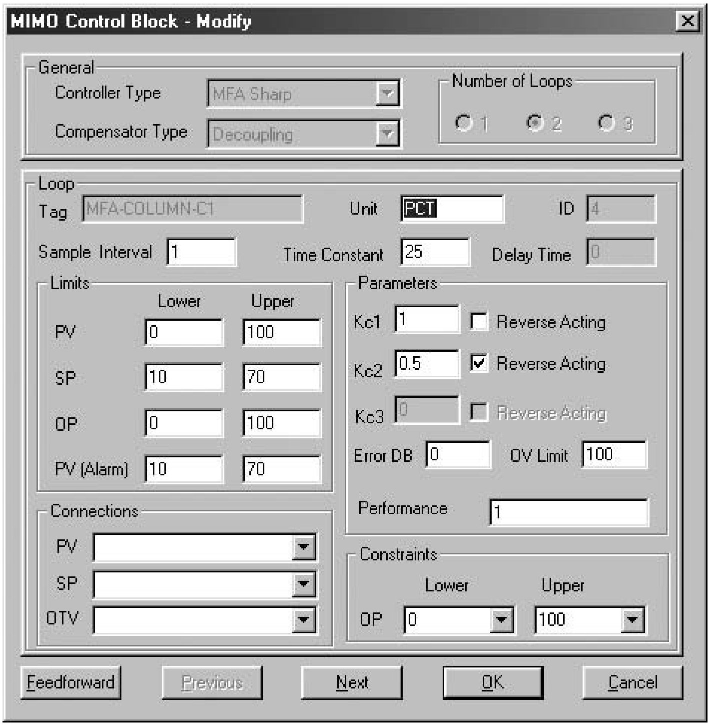

Figure 4.9 MIMO MFA controller configuration screen.

as shown in Figure 4.9. Some key variables in this menu are described in the following:

- *Compensator Type*. Specify the MIMO controller's compensator type. These are the optional controllers: decoupling, anti-delay, and combined (decoupling + anti-delay). The decoupling type is suitable for a 2×2 or 3×3 process without large time delays. If your process has large time delays, you need to specify the anti-delay option for the SISO process, or combined option for the MIMO process.
- *Sample Interval*. The interval between two samples or calculations. Unit: seconds. Range: 0.001 to 999.9 sec. Default setting: 1 sec. According to the principles in the information theory, it is required that the sample interval be smaller than or equal to one-third of the time constant. That is,

$$T_s \leq (1/3)T_c \tag{4.50}$$

where T_s is the sample interval, and T_c is the time constant.

- *Time Constant*. A rough estimate of the time constant of the main process. Unit: seconds. Range: 0.003 to 99,999 sec. Default setting: 20 sec. For example, if you are configuring C1 of a MIMO controller, this field is related to the time constant of process G_{11}.
- *Delay Time*. This field is not applicable for MIMO decoupling controllers. For the MIMO combined controllers, this field is related to the delay time of the main process. For example, if you are configuring C1 of a MIMO combined controller, this field is related to the delay time of process G11. For the MFA combined controller, this field specifies the process delay time between its input and output actions. Unit: second. If you do not know the delay time, look for the trends of the OP and PV signals. Estimate the delay time between a change in the OP signal and its corresponding response in the PV signal. You may need to adjust the delay time after running the controller to improve performance.
- K_{c1}. MFA controller gain for the main controller. Used to adjust the control performance for the main loop. Set it higher for a more active control action, and set it lower for less overshoot.
- *Reverse Acting* (for K_{c1}). The process acting type in the main loop. It is very important to check this field if the process is reverse acting. That means, if the process input increases, the process output decreases, and vice versa. For example, if you are configuring C1 of a MIMO controller, you need to check this field if G_{11} is reverse acting. If you are configuring C2 of a MIMO controller, you need to check this field if G_{22} is reverse acting.
- K_{c2}, K_{c3}. MFA compensator gain to deal with the interaction from other loops. Used to adjust the decoupling factors. Set it higher for a more active decoupling action, set it lower for less decoupling action, and set it to 0 to disable the compensator.
- *Reverse Acting* (for K_{c2}, K_{c3}). The process acting type. It is very important to check this field if the process is reverse acting. That means, if the process input increases, the process output decreases, and vice versa. For example, for a 2×2 system, if you are configuring C1, you need to check this field for K_{c2} if G_{12} is reverse acting.

MIMO MFA Application Guide

A multivariable process has multiple interactive inputs and outputs. For instance, a 2-input-2-output process (or 2×2 process) as illustrated in Figure 4.10 has two manipulated variables and two controlled variables. Changing one manipulated variable will affect both controlled variables. This phenomenon is called loop interaction.

Without loss of generality, we will discuss the principles of a multivariable process and related control system using only the case of a 2×2 process.

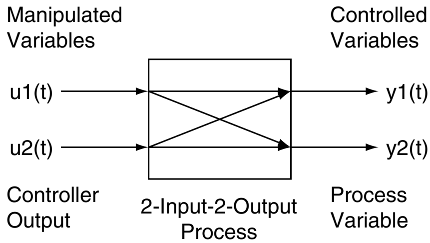

Figure 4.10 2-input-2-output process.

Relative Gain

The Bristol's relative gain is a simple yet powerful measure of the degree of loop interaction. Relative gain is defined as the ratio of open-loop gain (K_{ij}) to the closed-loop gain (K'_{ij}) (Hang et al., 1993).

The open-loop gains K_{ij} are the static gains of process G_{ij}. For instance, K_{11} is the static gain of G_{11}, and so on.

The closed-loop gains K'_{ij} are defined as the ratios of the change in one process output (for instance, y_i), to its input (u_i), when the other process output y_j is kept constant by manipulating u_j.

The relative gain is defined as

$$R_{ij} = \frac{K_{ij}}{K'_{ij}} \tag{4.51}$$

where K_{ij} is the process open-loop gain, and K'_{ij} is the process closed-loop gain. R_{ij} can be analytically computed by the following formula:

$$R_{11} = R_{22} = R = R_{ij} = \frac{K_{11}K_{22}}{K_{11}K_{22} - K_{12}K_{21}} \tag{4.52}$$

$$R_{12} = R_{21} = 1 - R \tag{4.53}$$

By ignoring the process dynamic elements such as the time constant and time delays, relative gains basically reflect the steady-state behavior of a multivariable process.

Pairing Rules of MIMO System

When designing a multivariable control system, the first step is to decide which controlled variable is paired with a manipulated variable.

Pairing Rule 1

Each process of the main loops has to be (i) controllable, (ii) open-loop stable, and (iii) either reverse or direct acting.

Pairing Rule 2

The measure of relative gains can help decide the correct pairing based on process static behavior.

1. A process with a large static gain should be included in the main loop as the main process (for instance, G_{11}, G_{22}).
2. A process with a small static gain should be treated as a subprocess (for instance, G_{21}, G_{12}).

To use the rule of relative gain:

The pair $i - j$ is correct if R_{ij} is closest to 1.

The rule can be appreciated if we examine the extreme case when either K_{12} or K_{21} equals zero.

Pairing Rule 3

It is also important to consider process dynamic behavior in determining the pairing.

1. A faster process should be paired as the main process such as G_{11} and G_{22}.
2. A slower process and processes with time delays should be treated as subprocesses (for instance, G_{21}, G_{12}).

Notes

1. If pairing rules 2 and 3 should result in a conflict, a trade-off is the only option.
2. Proper scaling may play an important role affecting the degree of difficulty of control, since it is related to both static gain and relative gain.
3. Correct pairing is one of the key issues that decide the quality of a MIMO control system. It is not unusual to see major improvements by simply correcting the pairing.

Degree of Interaction

Table 4.1 lists the control system design strategy based on the degree of interaction of the MIMO process. MFA controllers are always recommended.

Table 4.1 MIMO System Design Strategy

Interaction Measure	Control Strategy
Small to no interaction, R is close to 1 ($0.9 < R < 1.1$)	Tighten both loops with SISO MFA.
Moderate interaction ($0.7 < R < 0.9$, or $1.1 < R < 2$)	Tighten important loops with SISO MFA and detune less important loops; or use MIMO MFA for better overall control.
Severe interaction ($0.5 < R < 0.7$, or $R > 2$)	Use MIMO MFA to control the process. May need to detune less important loops.

Decoupling and Compensator Gain

The MIMO model-free adaptive controller includes decoupling compensators. They are used to eliminate or reduce the control loop interactions.

Compensator Gain K_{c2}, K_{c3}

Similar to the design of MFA controller gain K_{c1}, the compensator gains K_{c2} and K_{c3} can be used to fine-tune the decoupling effects. The rule of thumb in tuning these gains is similar to the tuning of MFA controller gains, but should be set in a more conservative fashion. For instance, a K_{c2} should be set to only 0.5 if the estimated static gain of its related subprocess is 1.

Anti-Delay MFA Control System

In process control applications, many processes have large time delays due to the delay in the transformation of heat, materials, signals, etc. A good example is a moving strip process such as a steel rolling mill or a paper machine. No matter what control action is taken, its effect is not measurable without a period of time delay.

Since the time delay makes a time-invariant system time-varying, many linear time-invariant (LTI) system analysis tools such as root loci methods and LTI state-space equations cannot be used to study the processes with time delays.

Most importantly, a time delay causes the output to not respond to the control signal promptly. It is equivalent to disabling the feedback for a period of time. Feedback information is essential to automatic control. The measure of how significantly a time delay affects the process behavior is related to the τ:T ratio.

If a PID is used to control a process with significant time delays, the controller output will keep growing during the delay time and cause a large overshoot in system responses or even make the system unstable. Typically, a PID has to be detuned significantly in order to stay in automatic but will sacrifice control performance.

Generally speaking, a PID controller usually works for the process if its τ:T ratio is less than 1, unless it is detuned. When a controller is detuned, it loses the sharpness of its control capability so that the process cannot be tightly controlled.

A regular model-free adaptive controller works for the process if its τ:T ratio is less than 2. For the process with significant time delay (τ:T ratio greater than 2), special treatment is required.

A Smith Predictor is a useful control scheme to deal with processes with large time delays. However, a precise process model is usually required to construct a Smith Predictor. Otherwise, its performance may not be satisfactory.

SISO Anti-Delay MFA Controller

Figure 4.11 shows a block diagram for a single-input-single-output anti-delay model-free adaptive control system with an anti-delay MFA controller and a process with large time delays.

A special delay predictor is designed to produce a dynamic signal $y_c(t)$ to replace the process variable $y(t)$ as the feedback signal. Then, the input to the controller is calculated as

$$e(t) = r(t) - y_c(t) \tag{4.54}$$

The idea here is to produce an $e(t)$ signal for the controller and let it "feel" its control effect without much delay so that it will keep producing proper control signals.

Since the MFA controller in the system has adaptive capability, the delay predictor can be designed in a simple form without knowing the quantitative information of the process.

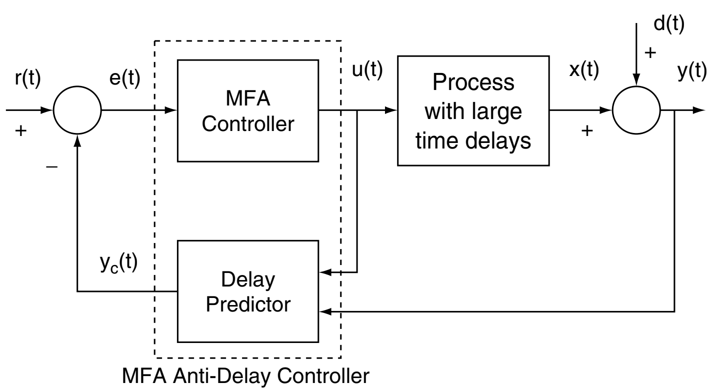

Figure 4.11 ***Anti-delay model-free adaptive control system.***

Compared to the traditional Smith Predictor, the anti-delay model-free adaptive controller does not need a precise process model. It only needs an estimated delay time as the basic information for its delay predictor. If the delay time used in the MFA delay predictor has a mismatch with the actual process delay time, the controller is robust enough to deal with the difference. Typically, it can deal with the situation where the delay time is 2 times larger or smaller than the actual delay time with satisfactory control performance. In addition, there is no real limitation on how large the τ:T ratio is allowed as long as a relatively close estimate of its delay time is provided.

Anti-delay MFA has achieved great success in real applications where processes with large and varying time delays such as electrical galvanizing lines are being effectively controlled. Anti-delay MFA's adaptive capability and special ability to deal with process time delays make it a valuable member of the MFA controller family.

SISO Anti-Delay MFA Configuration

To show how a practical anti-delay MFA controller is configured, we use an anti-delay controller configuration screen from CyboCon MFA control software as shown in Figure 4.12. Some key variables in this menu are described in the following:

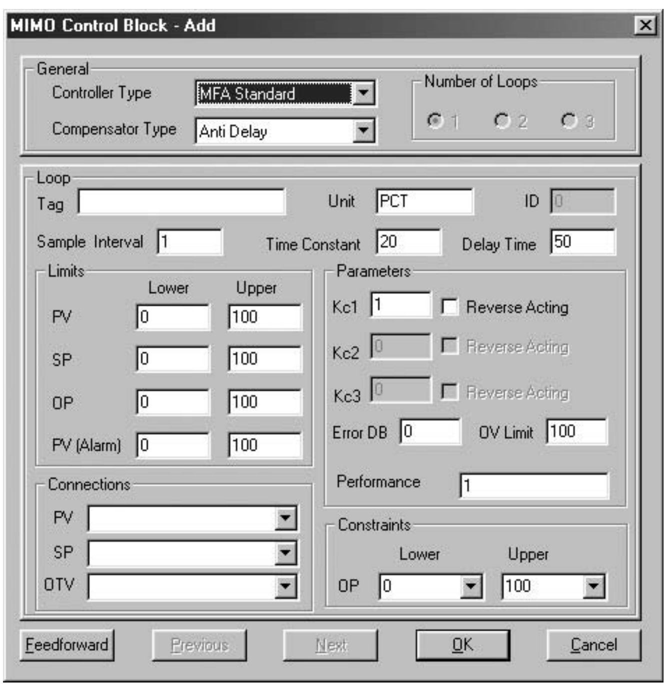

Figure 4.12 Anti-delay MFA controller configuration screen.

- *Delay Time*. The estimated delay time of the process in seconds. The time constant and controller gain can be configured like a SISO MFA controller.
- *Performance Index*. Based on the estimated process with K, T_c, and τ, let anti-delay MFA controller $K_c = 1/K$, time constant $= T_c$, and time delay $= \tau$. When the performance index (I_p) $= 1$, it is neutral. Increase I_p to tighten the control loop and decrease I_p to detune the controller.

MIMO Anti-Delay MFA Control System

Figure 4.13 illustrates a 2×2 multivariable anti-delay model-free adaptive control system. The anti-delay MFA controller set includes two MFA controllers C_{11} and C_{22}, two compensators C_{21} and C_{12}, and two predictors D_{11} and D_{22}. The process has large time delays in the main loops. Without losing generality, higher-order multivariable anti-delay MFA control systems can be designed accordingly. Because of the length of this chapter, a detailed description of this controller is not included.

MFA Cascade Control System

When a process has two or more major potential disturbances and the process can be divided into two loops (one fast and one slow), cascade control can be used to take

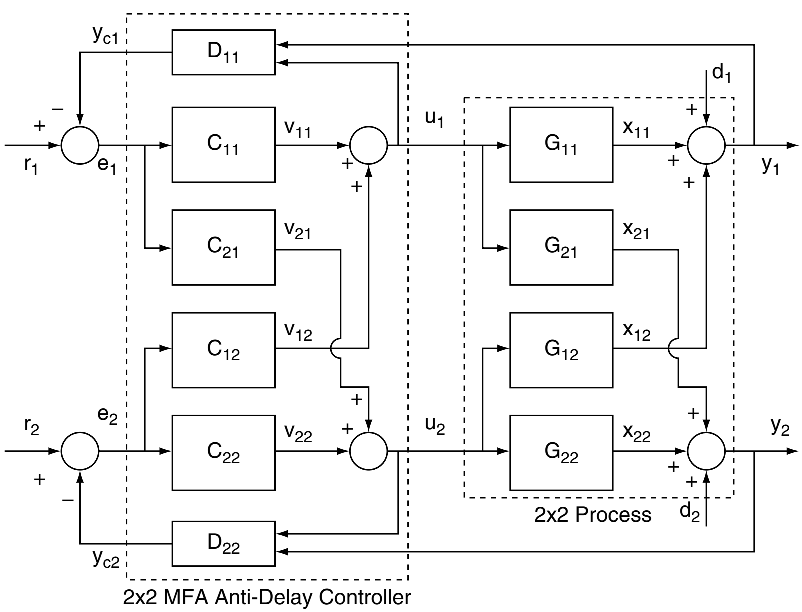

Figure 4.13 MIMO anti-delay MFA control system.

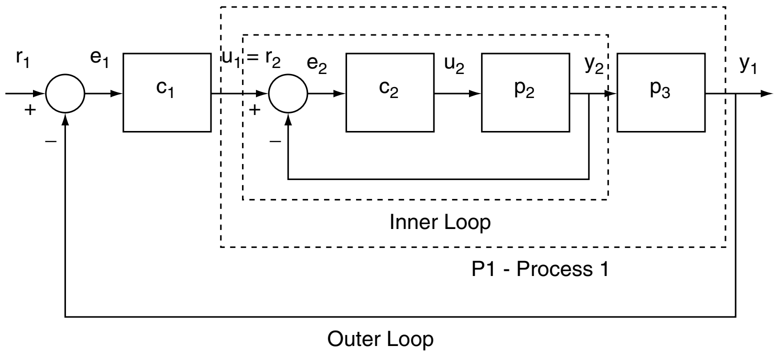

Figure 4.14 Model-free adaptive cascade control system.

corrective actions on the fast loop more promptly for better overall control performance. As illustrated in Figure 4.14, a cascade system contains two controllers, the primary controller C_1 and the secondary controller C_2. The inner loop consists of C_2 and P_2, and the outer loop consists of C_1 and P_1, where P_1 consists of C_2, P_2, and P_3. The output of C_1 drives the setpoint of C_2.

Although cascade control is one of the most useful control schemes in process control, it is often found that in real cascade control applications the outer loop is not closed.

Because of the interacting nature of the loops in the cascade control system, the requirement for proper controller tuning becomes much more important. If PI or PID controllers are used, four to six PID parameters have to be tuned. Good combinations of so many parameters are not easy to find. If the process dynamics change frequently, the controllers need to be retuned all the time. Otherwise the interacting nature of the inner and outer loop can cause serious system stability problems.

MFA in Cascade Control System

Using MFA controllers to form a cascade control system has an obvious advantage. Since the MFA controller used as C_2 can compensate for process dynamic changes, the closed-loop dynamics of the inner loop will change very little even though the process dynamics of P_2 may change a lot. This means the interconnection of the outer loop and the inner loop becomes much weaker. A more stable inner loop contributes to a more stable outer loop, and vice versa.

In addition, since each single-variable MFA controller has only one tuning parameter, the controller gain K_c, and it usually does not need to be tuned, the MFA based cascade control system becomes much easier to start up and maintain.

Multilayer Cascade Control System

Supervisory or setpoint control typically refers to a process optimization technique widely used in the process control industry. Basically, one calculates an optimal setpoint trajectory for the key process variable so that, if the process variable can track its setpoint trajectory, the process is optimized.

It is interesting to see that the cascade control system structure fits supervisory control quite well. The difference is that there is no need to derive a complex optimization algorithm to calculate the optimal setpoint trajectory; one can simply add a loop on the process variable that needs a "smart" setpoint trajectory. The question is, can this process variable be measured online?

For example, an industrial dryer typically has a cascade control system to control the dryer temperature using a temperature controller cascaded with a fuel flow controller. It is desirable to move the temperature setpoint around depending on how wet the raw material is and how large the load is. It is easy to imagine that calculating a setpoint trajectory for the temperature is not an easy task.

It is important to realize that the real control objective is almost always the quality variable. In this case, it could be the moisture of the product coming out of the dryer. Therefore, if we can measure the moisture of the product, we can simply add a control loop on top of the temperate and flow loop to make a three-layer cascade control system. The beauty of this design is that there is no need to calculate the complex optimal setpoint trajectory. The moisture controller's output is used as the setpoint trajectory of the temperature controller. The proper temperature control with an optimal setpoint trajectory will force the measured moisture to track its setpoint, which is determined based on the user's requirement or product specifications.

Automatic Control of Quality Variables

To conclude, a cascade control system has a unique importance to today's manufacturers where Six Sigma or zero defects quality objective is a necessity. Implementing a Six Sigma capable system can be very challenging. It is the author's belief that it is necessary to automatically control the quality variables such as density, moisture, product temperature, and dimension. Since there is usually a large delay time between the time when control action is taken and the time when the quality process variable is being measured, it is difficult to use a conventional controller such as a PID to control such quality variables. In this case, the anti-delay MFA controller is a good choice to be used as a quality variable controller with a multilayer cascade control system.

Cascade MFA Configuration

Configuration of a cascade control system using MFA controllers is relatively simple. According to the CyboCon software screen in Figure 4.15, it is seen that the Setpoint Connection field is filled with MFA-CAS1 (C1). Thus the setpoint of MFA-CAS2 (C2) is provided by the output of MFA-CAS1 when MFA-CAS2 is set in the remote mode.

There is no limitation on how many layers the cascade control system can include, as long as the process variables involved do not behave similarly. However, to simplify the system, it is recommended that you implement a cascade control system of only two or three layers.

The setup and tuning of the controllers in a cascade control system are carried out from the inside out. That is, the inner loop must be launched and working first; then the outer loop can be closed. It can be seen that the inner loop is part of the process for the outer loop.

The inner controller usually deals with a flow or pressure loop. If the process is relatively linear, it is not difficult to control since it is a relatively fast single-loop

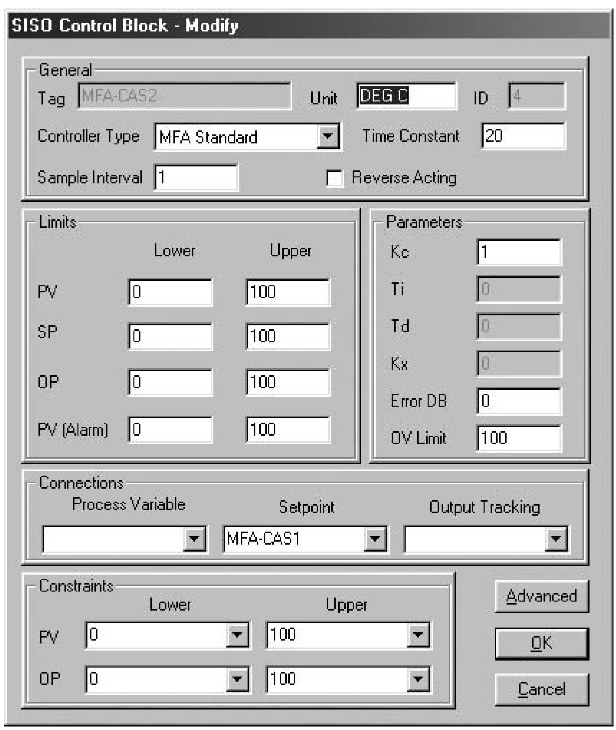

Figure 4.15 CyboCon software screen.

system. However, if the actuator has strong nonlinear characteristics or the pressure of the loop is so high that the actuator might lose its authority, the inner loop can oscillate in certain operating ranges, causing the entire cascade control system to fail. In this case, the nonlinear MFA controller to be introduced later will be very useful. This controller can be easily configured to deal with extremely nonlinear processes.

FEEDFORWARD MFA CONTROL

Basic Concept of Feedforward Control

Feedforward control, as the name suggests, is a control scheme to take advantage of forward signals. If (i) a process has a significant potential disturbance, and (ii) the disturbance can be measured; we can use a feedforward controller to reduce the effect of the disturbance on the loop before the feedback loop takes corrective action. If a feedforward controller is used properly together with a feedback controller, it can improve the control system performance quite economically.

Figure 4.16 illustrates a feedforward–feedback control system. The control signal $u(t)$ is combined with the feedback controller output $u_1(t)$ and the feedforward controllers output $u_2(t)$.

The feedforward controller is designed based on the so-called Invariant Principle. That is, with the measured disturbance signal, the feedforward controller is able to affect the loop response to the disturbance only. It does not affect the loop response to the setpoint change.

The control objective for the feedforward controller is to compensate for the measured disturbance. That is, it is desirable to have

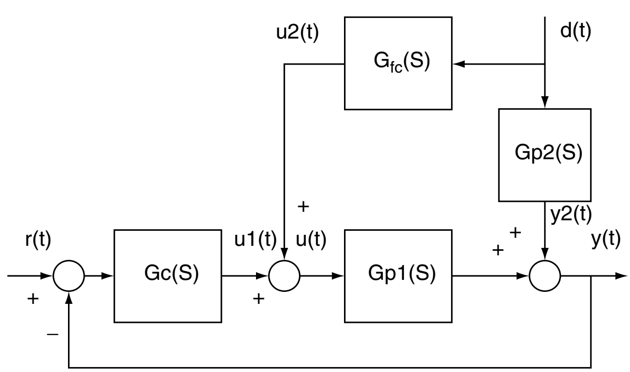

Figure 4.16 Feedforward and feedback control system.

$$G_f(S) = \frac{Y(S)}{D(S)} = 0 \tag{4.55}$$

where $G_f(S)$ is the Laplace transfer function of the feedforward loop; $Y(S)$ and $D(S)$ are the Laplace transform of process variable $y(t)$ and measured disturbance $d(t)$, respectively. Then, the FF controller can be designed as

$$G_{fc}(S) = -\frac{G_{p2}(S)}{G_{p1}(S)} \tag{4.56}$$

where $G_{fc}(S)$ is the Laplace transfer function of the feedforward controller.

Feedforward compensation can be as simple as a ratio between two signals. It could also involve complicated energy or material balance calculations.

Feedforward MFA Controller

For ease of use, it is desirable to design a more general-purpose feedforward controller so that it can be easily configured and launched.

The feedforward MFA controller is designed to be such a general-purpose feedforward controller that works with the feedback MFA controllers. The idea is not to attempt a perfect cancellation to the disturbances. This is because Invariant Principle–based perfect disturbance cancellation is very difficult to implement in industrial applications. When the feedback controller is MFA, its adaptive capability just makes many conventional control methods easier to implement and more effective.

There is no need to find the process models for $G_{p2}(S)$ and $G_{p1}(S)$ to design the feedforward controller $G_{fc}(S)$. Configuration, commissioning, and maintenance of the FF MFA controller is much simpler than for a regular FF controller.

Feedforward MFA Configuration

To show how a feedforward MFA controller is configured, we use a configuration screen from CyboCon MFA control software as shown in Figure 4.17. Some key variables in this menu are described in the following:

FF Gain K_{fc}

It is important to enter the correct sign for the gain, which can be found based on the following formula:

$$K_{fc} = -\frac{K_{p2}}{K_{p1}} \tag{4.57}$$

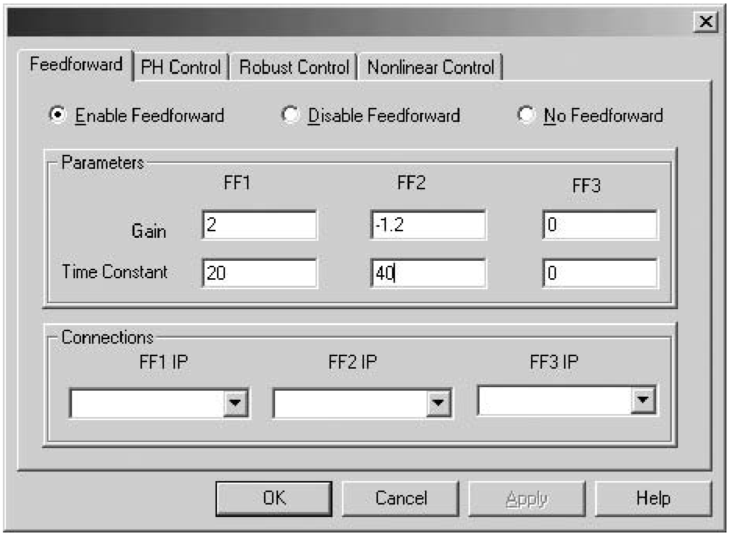

Figure 4.17 Configuration screen.

where K_{fc} is the gain for the Feedforward controller, and K_{p1} and K_{p2} are the estimated static gain for processes G_{p1} and G_{p2}, respectively.

In order to ensure that the feedforward action rejects the disturbance, the rules of selecting the sign can be summarized as follows:

- If G_{p1} and G_{p2} have the same sign, the FF's gain should be negative.
- If G_{p1} and G_{p2} have different signs, the FF's gain should be positive.

FF Time Constant

The Time Constant field can be filled with an estimate of the time constant of G_{p2}. This is related to how fast the disturbance will affect the process PV.

Feedforward MFA for Process with Large Time Delays

A process with large time delays can be complicated when applying a feedforward controller. As shown in Figure 4.18, the potential time delays can be part of G_{p1} and G_{p2} or in a separate process with pure time delay DT. Because of the length of this chapter, we will not discuss this in detail. Interested users can refer to Cheng (2000 or 2001a) for more information.

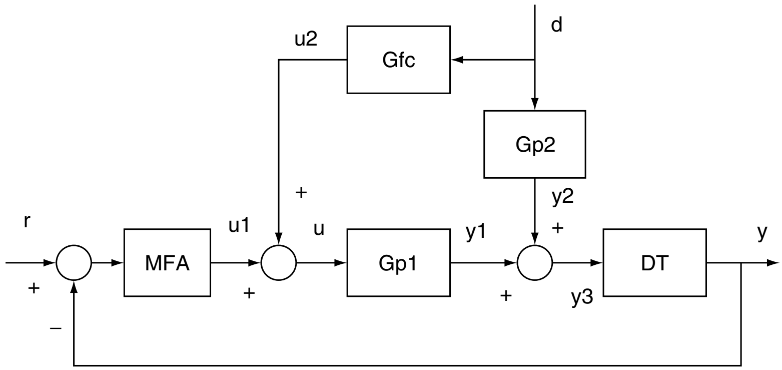

Figure 4.18 Feedforward for process with large delay time.

Nonlinear MFA Control

Nonlinear MFA Controller

Nonlinear control is one of the most challenging topics in modern control theory. Whereas linear control system theory has been well developed, nonlinear control problems present many headaches.

The main reason that a nonlinear process is difficult to control is because there could be so many variations in process nonlinear behavior. Therefore, it is difficult to develop a single controller to deal with the various nonlinear processes.

Traditionally, a nonlinear process has to be linearized first before an automatic controller can be effectively applied. This is typically achieved by adding a reverse nonlinear function to compensate for the nonlinear behavior so that the overall process input/output relationship becomes somewhat linear. It is usually a tedious job to match the nonlinear curve; and process uncertainties can easily ruin the effort.

A nonlinear MFA controller has been developed to deal with a wide range of nonlinear processes. It provides a more uniform solution to nonlinear control problems. The nonlinear MFA controller is well suited for (i) nonlinear processes and (ii) the processes with nonlinear actuators. A high-pressure loop is a typical nonlinear process that can cause the actuator to lose its authority in different operating conditions. Inevitable wear and tear on a valve typically makes a linear valve nonlinear.

Nonlinear MFA Configuration

The screen in Figure 4.19 shows how this nonlinear MFA controller is configured.

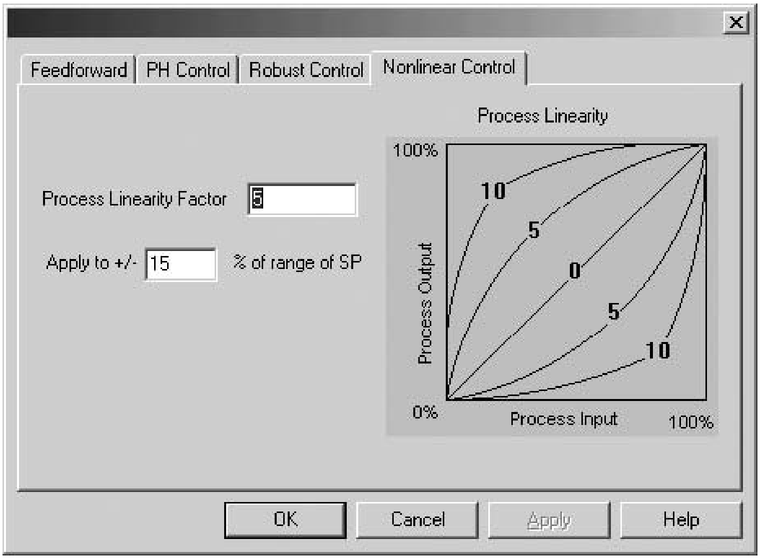

Figure 4.19 Nonlinear MFA controller configuration screen.

Process Linearity Factor

The graph in Figure 4.19 shows how severe the nonlinear behavior is between the process input and process output. The process linearity factor is a number between 0 and 10. A 10 represents an extremely nonlinear process while a 0 represents a linear process. Notice that the graph shows a nonlinear curve marked with 10 on both upper and lower positions. This means a nonlinear MFA controller does not care what the nonlinear characteristics are for this process. For instance, the valve can be either “fast open” or “fast close” as represented by these two convex and concave curves.

When using nonlinear MFA, you do not need to worry about how the nonlinear curve is laid out. It is your option to tell the controller whether the process is extremely nonlinear (give a 9 or 10), quite nonlinear (give a 5 or 6), or somewhat nonlinear (give 1 or 2). The nonlinear MFA controller will be smart enough to handle the rest.

Simulations and real applications show that the nonlinear MFA controller can easily deal with a nonlinear process even if its gain changes hundreds of times. For a nonlinear MFA, there is no linearization calculation or process model. The MFA controller gain K_c is simply set at its nominal point and not retuned.

To conclude, the nonlinear MFA controller provides effective and tight control of flow, pressure, and other key process variables with consistent performance.

Less loop oscillation and smoother operation result in higher product quality, improved production efficiency, and less energy and material consumption. Since it can effectively control the flow and pressure in its entire operating range without the need to retune its parameters, this controller is especially useful for the multilayer cascade control system as the inner controller. It improves the overall control system stability and enables automatic control of the quality variable that is controlled by the outer loop controller.

MFA pH Controller

Most process plants generate a wastewater effluent that must be neutralized prior to discharge or reuse. Consequently, pH control is needed in just about every process plant, yet a large percentage of pH loops perform poorly. Results are inferior product quality, environmental pollution, and material waste.

With ever-increasing pressure to improve plant efficiency and tighter regulations in environmental protection, effective pH control is very desirable. However, implementing a pH system is like putting a puzzle together. It will only work when all the components are in place.

The pH puzzle includes effective pH probes, actuators, and controllers. Whereas various pH probes and actuators for pH control are available, commercial adaptive pH controllers are still scarce.

The MFA pH value controller is able to control a wide range of pH loops because its powerful adaptive capability allows it to compensate for the large nonlinear gain changes. It controls full pH range with high precision and enables automatic control of acid or alkaline concentration, which are critical quality variables for chemicals.

MFA pH Configuration

A SISO MFA controller can be configured with special capability for controlling strong-acid-strong-base pH loops. As shown in Figure 4.20, you can enter the break points A and B to define the estimated shape of the titration curve of the pH process. Then you can enter the MFA controller gain Kc for the flat portion and steep slope, respectively. Because of the adaptive capability of the MFA controller, the titration curve does not have to be accurate and, in fact, its shape can vary in real applications.

MFA pH control has helped many users to effectively control their tough pH loops. Quick return-on-investment was reported with savings on chemical reagents, no violation of discharge code, and smoother production operation.

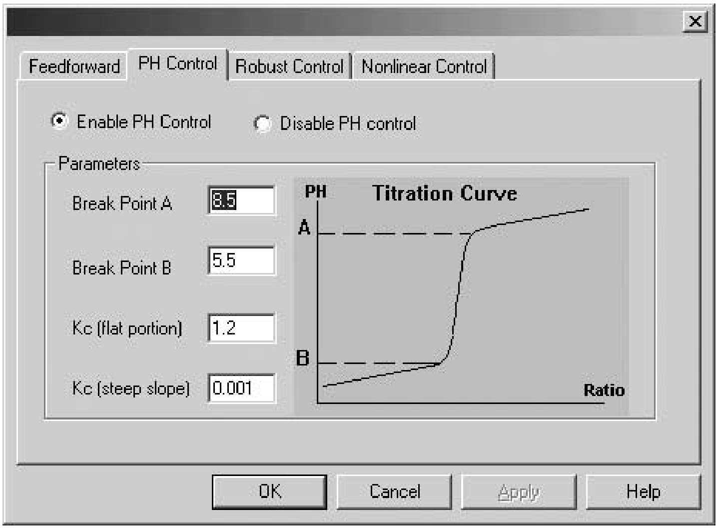

Figure 4.20 MFA pH controller configuration screen.

Robust MFA Control

In complex control applications, we may face the following challenges that even a regular MFA controller cannot effectively handle:

1. There is a big change in the system dynamics so that a regular MFA controller is unable to provide prompt control action to meet the control performance criteria;
2. The dominant disturbance to the system cannot be economically measured so that feedforward compensation cannot be easily implemented;
3. A controller purposely detuned to minimize the variations in its manipulated variable may lose control when there is a large disturbance or significant dynamic behavior change; or
4. The system dynamic behavior or load change does not provide triggering information to allow the control system to switch operating modes.

A Chemical Reactor Example

To describe the application in more detail, a chemical reactor control problem is studied in the following. Chemical batch reactors are critical operating units in the chemical processing industry. Controlling the batch reaction temperature is always a challenge because of the complex nature of the process, large potential disturbances, interactions between key variables, and multiple operating conditions. A large percentage of batch reactors running today cannot keep the reactor temperature in automatic control throughout the entire operating period, thus resulting in lower efficiency, wasted manpower and materials, and inconsistent product quality.

An exothermal batch reactor process typically has four operating stages:

1. *Startup stage*. Ramps up the reactor temperature by use of steam to a predefined reaction temperature.
2. *Reaction and holding stage*. Holds the temperature by use of cooling water while chemical reaction is taking place and heat is being generated.
3. *No-reaction and holding stage*. Holds the temperature by use of steam after main chemical reaction is complete and heat is not being generated.
4. *Ending stage*. Ramps down the reactor temperature for discharging the products.

During the transition period from stage 2 to stage 3, the reactor can change its nature rapidly from a heat-generation process to a heat-consumption process. This change happens without any triggering signal because the chemical reaction can end at any time depending on the types of chemicals, their concentration, catalyst, and reaction temperature. Within a very short period of time, the reactor temperature can drop significantly. The control system must react quickly to cut off the cooling water and send in a proper amount of steam to drive the reactor temperature back to normal. A regular feedback controller is not able to automatically control a batch reactor during this transition if it is tuned to control the process in stages 1 and 2. In practice, batch reactors are usually switched to manual control and rely on well-trained operators during critical transitions. It is a tedious and nerve-wracking job that can result in low product quality and yield.

Robust MFA Controller

A robust MFA controller is developed to control the problematic processes described. Robust control is usually referred to as a controller design method that focuses on the reliability (robustness) of the control algorithm. Robustness is defined as the minimum requirement a control system has to satisfy to be useful in a practical environment. Once the controller is designed, its parameters do not change and control performance is guaranteed.

Robust MFA is not a control system design method. We use the term "robust" here because this novel controller is able to dramatically improve the control system robustness. Without the need to redesign a controller, use feedforward compensation, or retune the controller parameters, the robust MFA controller is able to keep the system in automatic control through normal and extreme operating conditions when there are significant disturbances or system dynamic changes.

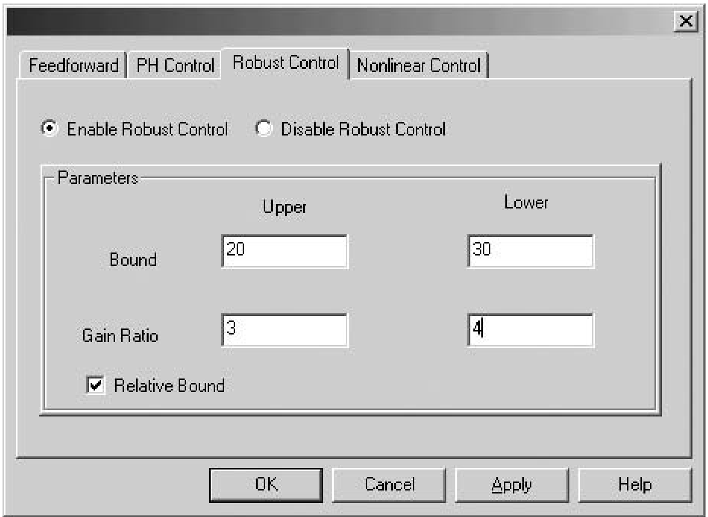

Figure 4.21 Robust MFA controller configuration screen.

Robust MFA Configuration

As shown in Figure 4.21, the robust MFA controller can be easily configured with these parameters.

Upper and Lower Bound

These are the bounds for the process variable (PV) being controlled. They provide intelligent upper and lower boundaries for the process variable. These bounds are typically the marginal values that the process variable should not go beyond.

PV is unlike OP where a hard limit or constraint can be set. PV is a process variable that can only be controlled by manipulating the OP. Therefore, the upper and lower bounds for PV are very different from the OP constraints.

For instance, if you have a level loop with a setpoint of 50 and you want to have a $+20$ and -30 bound around the setpoint, you should enter upper bound $= 20$, lower bound $= 30$.

Relative Bound

If this box is checked, the upper and lower bounds you enter are related to the setpoint as described above. If this box is not checked, the upper and lower bounds you enter are the actual bounds.

Gain Ratio

The gain ratio is the coefficient to increase or decrease the MFA control action. Since MFA controller gain K_c is the only tuning parameter we use, we use the so-called gain ratio to relate this parameter to the actual MFA gain K_c. Typically, you want to enter Gain Ratio = 3, which implies that the MFA gain working in the abnormal situation is 3 times higher than the regular MFA gain setting.

It is important to understand that this is not a gain scheduling approach, although it appears to be this way. Gain scheduling will not be able to resolve the complex problems described.

MFA Control Methodology and Applications

The concept, architecture, algorithm, system structure, and configuration of various model-free adaptive control systems were described in the previous section. To have a deeper understanding of this unique control method, it is necessary to (i) study the philosophy and methodology used in developing these MFA control systems, and (ii) review the MFA control applications that have proven the concept.

MFA Control Methodology

"All roads lead to Rome."

A problem usually has multiple possible solutions. A process can usually be controlled using different controllers based on different control methods. Almost every control method has its merits and weakness. What's important is to develop the right type of controller to fit the right type of application at a minimum cost.

In natural science, the combination of physics, mathematics, and philosophy plays an integral part in developing a theory that is practically useful. Physics is the foundation for the study of the physical process or environment; mathematics provides the tools to precisely describe the physical process or phenomenon; and, equally important, philosophy provides directions.

The development of model-free adaptive control technology started from a simple desire to develop a new controller that could easily and effectively solve various industrial control problems. The actual development process has evolved from a prolonged interest in the study of combined intelligence methodology. Since model-free adaptive control does not follow the traditional path of model-based adaptive control, the philosophy behind combined intelligence has led the way up this long and rocky road.

Combined Intelligence Methodology

The combined intelligence methodology developed by the author of this chapter consists of the following problem-solving philosophy:

1. Always seek a simple solution for a complex problem
2. Use all information available
3. Do not depend on the information's accuracy
4. Apply a technique that fits the application

These four key points are described in the following.

Seek a Simple Solution

A simple solution is almost always the best solution. A complex solution might achieve a little better result, but the cost can be very high. Most users want to have a tool or system that is easy to use, launch, and maintain with the best price–performance ratio. A simple solution usually fits this need well.

Use All Information Available

It is a cliché, but we are living in the information era. Information has value. A small piece of information can make all the difference. When solving a problem, do not waste the valuable information available. For instance, a process delay time can easily be seen from the process trend chart. A regular PID controller ignores this important piece of information.

Do Not Depend on the Information's Accuracy

Practically, information received may not be accurate. What's worse is that we often do not even know whether the information is accurate or not. If we knew, we would simply have the option to use or not use the information. For this reason, the solution we provide has to be agile or adaptive enough that it can deal with inaccuracy of the information and with uncertainties.

Apply a Technique That Fits the Application

Arguments often arise between people who believe in very different problem-solving methods. For instance, model-based and rule-based methods are two very distinctive approaches in control theory. Since almost all methods have their merits and shortcomings, why argue? Let's use the technique that fits the application.

The MFA Control Approach

To see how the MFA control method is developed based on this problem-solving philosophy, we will relate MFA to each one of the points in the following discussion.

Seek a Simple Solution

PID control is simple since it is a general-purpose controller and its algorithm is easy to understand. However, PID is almost too simple to control complex systems. In this regard, PID cannot be considered an effective solution to the more difficult control problems. On the other hand, model-based advanced control methods have proven themselves too complex to launch and maintain since they depend on either a first principle or an identification-based process model. A dream controller has to be powerful enough to control various complex processes yet simple enough to use, launch, and maintain. MFA is a solution that fits these requirements.

Use All Information Available

Model-free adaptive control, as its name suggests, is a control method that does not depend on either first principle or identification-based process models. However, we do try to use all the process information available. For this reason, it can be considered an information-based controller.

For instance, the process time constant defines how fast a dynamical system responds to its input. A slow process might have a 10-hour time constant and a fast process might have a 10-millisecond time constant. It would be unwise not to use this information for the controller. In addition, it is relatively easy to estimate the time constant by reading a trend chart. Other important yet easily obtained information about a process includes its acting type (either direct or reverse), static gain, and delay time, if any. As described earlier, an MFA controller is designed to use the process parameters that can be easily estimated.

Do Not Depend on the Information's Accuracy

A process can be classified as a white, gray, or black box. If its input/output relationship is clear, the process is a white box. We can easily use existing well-established control methods and tools to design a controller for this process.

When we are not sure if the process input/output relationship is accurate, or if the process has potential disturbances, dynamic changes, and uncertainties, the process is a gray box. In this case, MFA's adaptive capability is able to handle such changes and uncertainties. PID or model-based control methods will have a much tougher time or higher cost addressing these uncertainties.

Apply a Technique That Fits the Application

MFA is neither model-based nor rule-based. We might say that it is an information-based control method. If the argument is made that the process information used is equivalent to a process model, that's perfectly acceptable. The key to our approach is that we focus on delivering a simple, adaptive, and effective solution.

To extend this idea, a series of MFA controllers have been developed to address different difficult control problems. Users can simply pick the appropriate MFA, configure its parameters, and launch the controller.

The Inside of MFA

Since its introduction, the concept of model-free adaptive (MFA) control has puzzled many people. The very definition of MFA control makes strong claims and it is difficult to comprehend how it works. Although detailed architecture and algorithm of MFA controllers were presented earlier, the following discussion summarizes what MFA control is and how it works.

A comprehensive article about model-free adaptive control was published in *Control Engineering Europe* magazine's February/March 2001 issue (VanDoren, 2001).

What MFA Is Not

MFA is not

- A self-tuning PID controller
- A model-based adaptive controller
- A model-based neural network controller
- A fuzzy controller

What MFA Is

MFA is a novel, patented, and "model-free" adaptive controller. MFA consists of a nonlinear dynamic block that performs the tasks of a feedback controller. A dynamic block is just a dynamic system with inputs and outputs. The control objective is to produce an output to minimize the error between the setpoint and the process variable (PV) being controlled.

Within the dynamic block there is a group of weighting factors that can be changed as needed to vary the nonlinear functions of the block. Then the MFA controller can adapt as the process dynamics change. In addition, the MFA controller "remembers" a portion of the process data enabling it to make quick and precise changes to its output.

Why "No-Model" Is Possible

Automatic control methods are all based on the concept of feedback. The essence of feedback theory consists of three components: measurement, comparison, and correction. Measuring the quantity of the variable to be controlled, comparing it with the desired value, and using the error to provide the control action is the basic procedure of automatic control.

The key to the theory and application of automatic control is the ability to make the correction based on proper measurement and comparison. An automatic controller does not require a process model to produce an output. Therefore, an adaptive controller does not have to rely on a process model to produce an output.

Why Use a Model?

A feedback automatic controller relies on the feedback signal to produce its output and does not require a process model to produce an output. That means there should be a way to compute control actions directly from the process input and output signal without first creating a model to mimic the process behavior. The basic process information such as time constant is available and the dynamical information that involves the changes and uncertainties is included in the process input and output data. That means we should be able to "crunch" the data correctly to let the controller generate a proper output.

In other words, computing control action does not require a process model. The real trick is to compute the control action and vary the weighting factors based on the input/output data. The key to MFA is a set of algorithms that can do this job.

MFA System Stability

In model-free adaptive control theory, a sufficient condition for MFA control system stability is derived and can be described as follows: If the process is passive (such as open-loop stable), the closed-loop MFA system stability is guaranteed and the process can be linear, nonlinear, time-invariant, time-varying, single-variable, or multivariable.

MFA Requirements

1. *The process is controllable*. If it is not controllable, you need to change the process or improve the variable pairing.
2. *The process is open-loop stable*. If it is not open-loop stable, it must first be stabilized. However, for certain simple open-loop unstable processes such as a nonself-regulating level loop, no special treatment is required.
3. *The process is either direct or reverse acting (the process does not change its sign)*. If a process changes its sign within its operating range, it is still possible to design a special controller for this "ill" process.

Key Issues of MFA

- *MFA does not include an identification mechanism.* No dynamic modeling mechanism is used in MFA. In other words, there is no identification engine inside MFA. This approach eliminates the headaches that a model-based control method causes, such as offline model training or the possibility of obtaining a bad model through identification. For instance, when a sensor fails, an identification mechanism may learn a bad process model, which can jeopardize the control system.
- *MFA adapts as is needed.* Thus, MFA does not need continuous excitation of the process signals. In other words, if the process dynamics do not change, the controller will not attempt to vary the weighting factors.
- *MFA is a robust controller.* When MFA is installed, the adaptation will not cause it to develop a strange behavior because it does not rely on a model. In addition, since it is an adaptive controller, its robust range is wider than that of a regular PID, which does not adapt at all.
- *MFA is not a single solution.* Based on the core of MFA, different types of MFA controllers have been developed to deal with special problems. For instance, anti-delay MFA is very effective in controlling processes with large time delays; and nonlinear MFA is well suited to control nonlinear processes such as pressure loops where valves may lose authority. More special MFA controllers are being developed as needed.

CASE STUDIES

So far, we have discussed the ideas of MFA control in theory. Let's see how it works in practical applications. Five application stories were selected that discuss different types of MFA controllers, including SISO, MIMO, anti-delay, nonlinear pH, and robust MFA.

SISO MFA on Tomato Hot Breaks

The tomato hot-breaks temperature control application is at Del Monte Foods' Woodland, CA, plant, and an article entitled "Adaptive and Predictive Controls Boost Product Quality" describing this application was published in *Food Engineering Magazine*'s December 1999 issue (Morris, 1999).

From July through early October, the plant operates 24 hours per day as a continuous caravan of gondola trucks unloads tomatoes into flumes feeding the hot break lines. Tomatoes are chopped and fed into heating vessels to become tomato slurry. Production throughput is critical to cost efficiency during the short processing season. The major challenge is to maintain good temperature control of these rotary-coiled heating vessels. Tomato inflow can be wild between truckloads, causing large temperature variations. Too high a temperature causes burning and clogging, and too low a temperature affects the production efficiency of the downstream

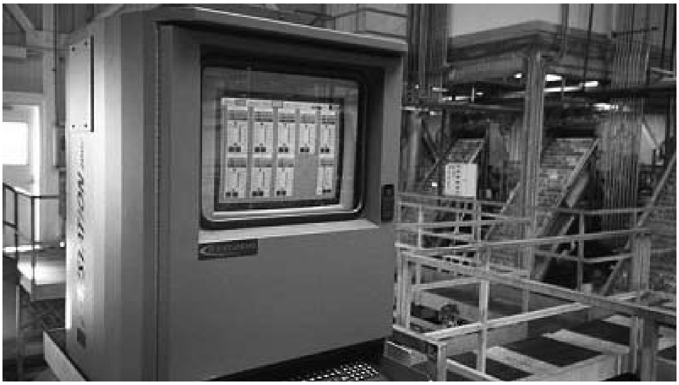

Figure 4.22 MFA control station and tomato lines.

evaporators. In addition, overheating the tomato slurry wastes energy and causes quality problems. Figure 4.22 illustrates the MFA control station and the tomato lines used at Del Monte's Woodland plant.

If the tomato flow could be measured, feedforward control would be applied to solve the control problem. Measuring the solidity of tomatoes is difficult and costly. Most tomato processing plants are not equipped to measure solids. PID controllers were previously used to control these nine heating vessels where temperature swings typically varied as much as 15 to 20°F.

The plant installed CyboCon model-free adaptive (MFA) control software with nine SISO MFA controllers to control the temperature of the heating vessels. CyboCon was installed in just a few hours. The PID loops were retained offering the operator a choice of control, "but since installation the operators have used CyboCon 100% of the time," according to Del Monte's plant operations manager. Product temperature now typically varies within less than ±2°F.

Figure 4.23 shows that the MFA controllers in CyboCon software quickly and tightly control the temperature by manipulating steam to compensate for wild tomato inflow without using feedforward control.

Since its installation, no maintenance has been required to retune these controllers. The plant installed more MFA controllers the following season to control boilers and steam injection systems for sterilization, which all achieved the desired control objectives and economic benefits.

MIMO MFA on Multizone Temperature Control

A multizone temperature MFA control for industrial coking furnaces is installed at the Guangzhou Petrochemical Complex of SINOPEC in China, and an article

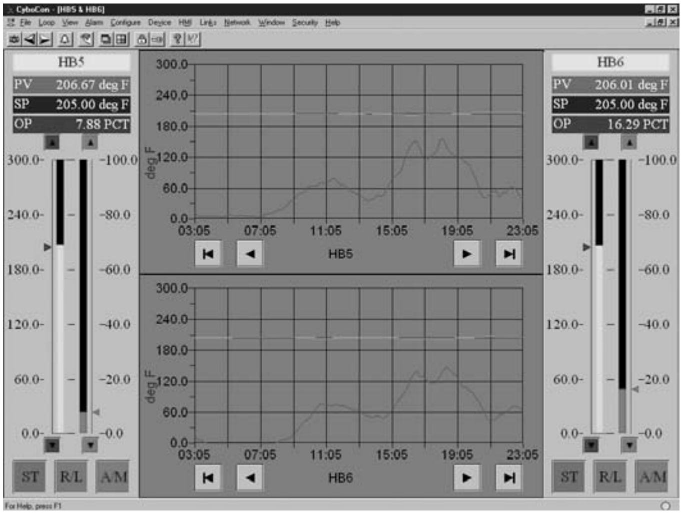

Figure 4.23 MFA control screen.

entitled, "Model-Free Coking Furnace Adaptive Control" presenting the application was published in *Hydrocarbon Processing* magazine's December 1999 issue (Cheng et al., 1999).

Control of temperature loops in multiple zones can be problematic, especially when a narrow specification is required. Successful installation of a model-free adaptive (MFA) control system in the delayed coking process at the Guang-Zhou Petrochemical Complex (GPC) shows how this problem can be resolved.

As shown in Figure 4.24, a coker consists of two coking furnaces, each with two combustion chambers. High temperatures create carbon that clogs pipes, and a below-spec temperature causes an insufficient reaction so that the yield drops.

Control difficulties result from (i) large time delays, (ii) serious coupling between these two temperature loops because the separation wall between the two combustion chambers is quite low, and (iii) multiple disturbances in gas pressure, oil flow rate, oil inflow temperature, and oil composition. The oil outlet temperature is sensitive to the gas flow rate change, and the temperature specification is tight ($\pm 1^{\circ}$C).

A 2×2 anti-delay MFA controller was installed to manipulate the fuel flow valves on each furnace to control the outlet temperature. The MIMO anti-delay MFA solved large time delay and coupling problems. MFA controllers compensated for disturbances and uncertainties. Constraints on controller outputs prevented temperatures from running too high or too low.

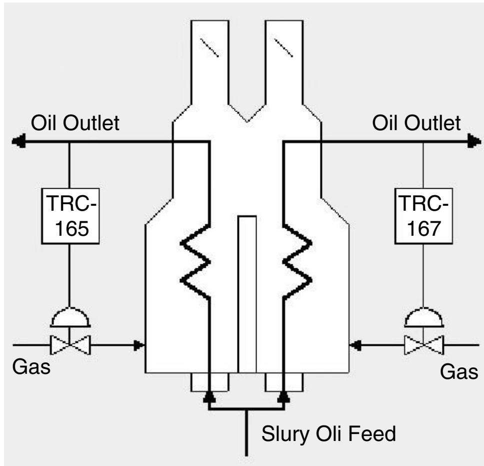

Figure 4.24 Control diagram of coking furnace.

According to the chief engineer at GPC, MFA controllers started automatic control with no bumps to the system. Commissioning took 3 days, with the following results: (i) both furnaces are automatically controlled under all conditions; (ii) outlet oil temperature is controlled to within $\pm 1°C$ with energy savings and consistent product quality; (iii) operators have been relieved of tedious, ineffective manual control responsibilities; and (iv) higher efficiencies and yields have been achieved.

Anti-Delay MFA on Metal Galvanizing Process

This anti-delay MFA control application is at the rolling mill of Wuhan Iron & Steel Corporation in China.

In a rolling mill, a galvanizing process produces tinned plating. Plating thickness is key to product quality. Plating that is too thick or thin causes an economic loss. If the layer is too thick, precious metal is wasted; and if it is too thin, defective parts are produced.

The tinning processing line of Wuhan Iron & Steel (Group) Co. (WISCO) performs large-scale continuous, high-speed tinned plating. Usually, on most plating lines, thickness can only be controlled manually. Differences in operator skills led inevitably to inconsistent product quality.

As shown in Figure 4.25, to keep plating continuous, buffers that store strip up to 150 meters long are arranged on both sides of the galvanizing tank. The outlet buffer stores strip when cutting while the inlet buffer supplies strip when welding.

Plating thickness could only be controlled manually, and it was manipulated by the electrical current. The amount of current applied was calculated according to the

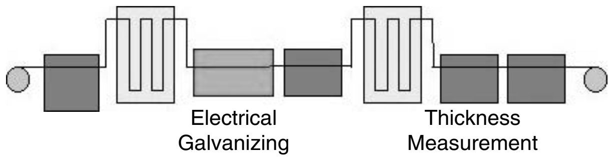

Figure 4.25 Galvanizing line.

thickness, width, and speed of strip. Concentration and temperature of the galvanizing liquid are other variables that can affect the thickness. Large disturbances in strip speed affected thickness even though the current was adjusted based on a process model.

Behind the outlet buffer, a sensor with a range of 0–15 g/m^2 measures plating thickness online. This provides an opportunity to control this quality variable automatically. However, the buffer between the galvanizing tank and the sensor creates a large and random time delay that changes from 30 to 150 sec and cannot be easily predicted. A variation in sizes of products (more than 20 types per day) causes frequent changes in process dynamic behavior and time delay. Other disturbances affect product quality and also make automatic control extremely difficult.

Two anti-delay MFA controllers were installed to control the plating thickness of the top and bottom layers. A feedforward MFA controller for each feedback MFA was also used to quickly overcome the speed disturbance. MFA control succeeded in reaching the objectives defined by WISCO:

1. *Control of plating thickness to within* $\pm 0.5\,g/m^2$. MFA controllers maintain thickness within $\pm 0.3\,g/m^2$ even if strip speed changes severely; and
2. *Ease of installation and operation*. Soon after connecting to the PLC, MFA controllers for both top and bottom layers were launched and controlled plating thickness immediately. Commissioning was completed in 2 days. Return-on-investment was achieved in a short period of time because of automatic control of the quality variables.

MFA pH Control on Water Neutralization Process

This MFA pH control application is at the Rohm & Haas Company in Houston, Texas. Rohm and Haas, a leading chemical company, is successfully using an MFA pH controller to control a problematic pH loop to neutralize an organic process stream with an estimated savings at $170,000 per year. A reliability improvement is also achieved due to reduced formation of solids, according to the control engineer at Rohm & Haas.

The stream to be neutralized is a two-phase stream with varying concentrations of acidic species. Caustic water is added as reagent to neutralize the stream. In this case, pH control is difficult because of the strong nonlinearity of the pH process and the large load changes in the stream inflow.

The original system was designed with one caustic valve manipulated by a single-loop PID controller. The control performance was not satisfactory. One tedious solution might involve reengineering the process and/or using the process data to create a mathematical model to linearize the process input and output relationship.

The recommended pH setpoint was 10.6. However, operators typically ran the process at 12 because the pH loop became unstable when it was close to the recommended setpoint. Excess caustic from the higher pH resulted in solids formation in the downstream separation equipment. The result was wasted chemical material and excessive clogging. On the other hand, setting the PID gain low to improve control stability would result in an extremely sluggish control response when a large upset pushed the pH too far away from the neutrality region.

As illustrated in Figure 4.26, a CyboCon CE model-free adaptive control instrument was installed to control this pH process. The CyboCon CE was mounted on the existing panel and wired directly to the process. Launching the MFA pH controller was simple with no complicated tuning, step testing, or data collection involved.

Improved pH control enabled Rohm & Haas to lower the pH setpoint from 12 to 11. Not only were cost benefits achieved, operators also liked the improved process

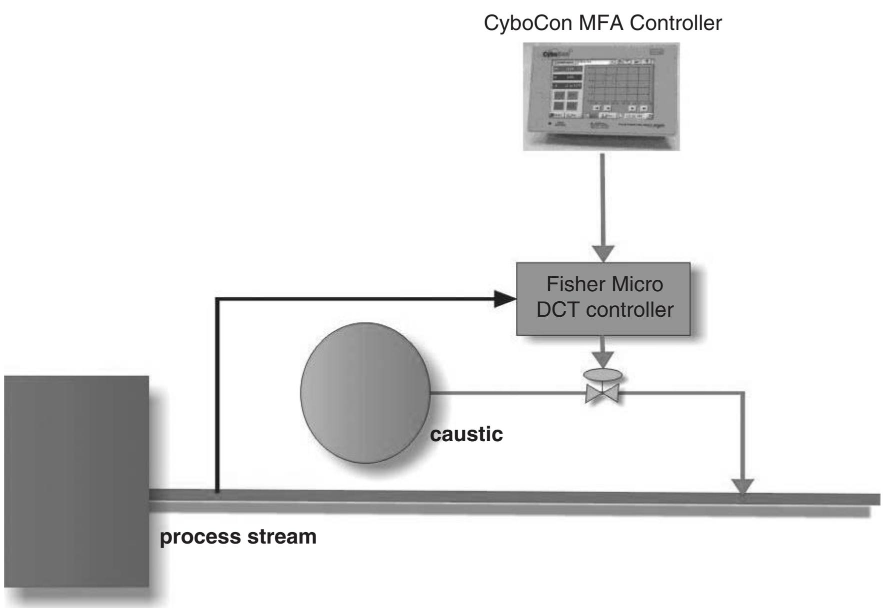

Figure 4.26 MFA pH control diagram.

upset handling capabilities. In addition, reduction of excess caustic and reduced solids formation meant an unquantified improvement in overall system reliability.

Other MFA pH control applications, including the ones at Chiron in California and Ultrafertil in Brazil, achieved similar results within a short return-on-investment period.

Robust MFA on Distillation Column

Robust MFA controllers were installed in two Air Liquide America locations, and an article entitled "Air Separation Advances with MFA Control" describing the details of these applications was published in *Control Magazine*'s May 2001 issue (Seiver and Marin, 2001).

Air Liquide America, a global provider of industrial, electronic, and health care gases, has standardized on model-free adaptive control for advanced regulatory control applications after successful MFA installation on two air separation units (ASUs).

Figure 4.27 illustrates the process diagram of an air separation unit. The main goal of operating an ASU is to maximize yields of gases including oxygen, nitrogen, and argon; and maintain the operation in as steady a state as possible. The specific goal of

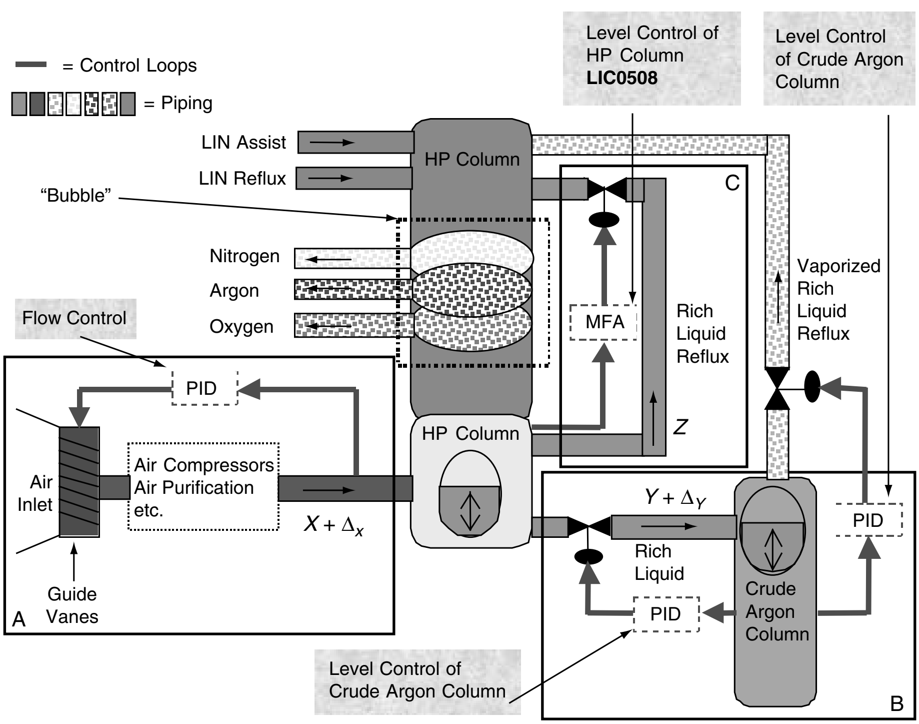

Figure 4.27 Diagram of an air separation unit.

the initial application was to control the rich liquid (RL) reflux level in the high pressure (HP) cryogenic column so that it would remain as constant as possible, even during plant ramping and upsets. The RL reflux flow to the low-pressure (LP) cryogenic column is used to manipulate the HP column RL reflux level.

It is difficult to properly tune a PID controller for good control performance under all conditions on an ASU because of the variable rates of the HP column inflow and outflow. Overly tight level control will result in large oscillations in the reflux flow, which causes a lower product yield. A PID level controller is usually detuned to allow the level to fluctuate to minimize variations of the reflux flow. This may result in safety problems during a plant upset since the detuned PID cannot deal with large disturbances. In addition, oscillations in level can cause the process to swing, which also results in a lower yield.

The robust MFA controller has been used to tightly control the level. A surprising result is that both level and reflux variations have been reduced at the same time. MFA immediately started to set production records as soon as it came on line. MFA control proved quite easy to install on ASUs. Air Liquide staff engineers performed the entire installation and commissioning at their McMinnville, Oregon, plant within a single day. Since its installation, virtually no maintenance or retuning has been required.

Figures 4.28 to 4.31 show the trend charts obtained from the actual system at Air Liquide that compare the performance of PID and MFA control.

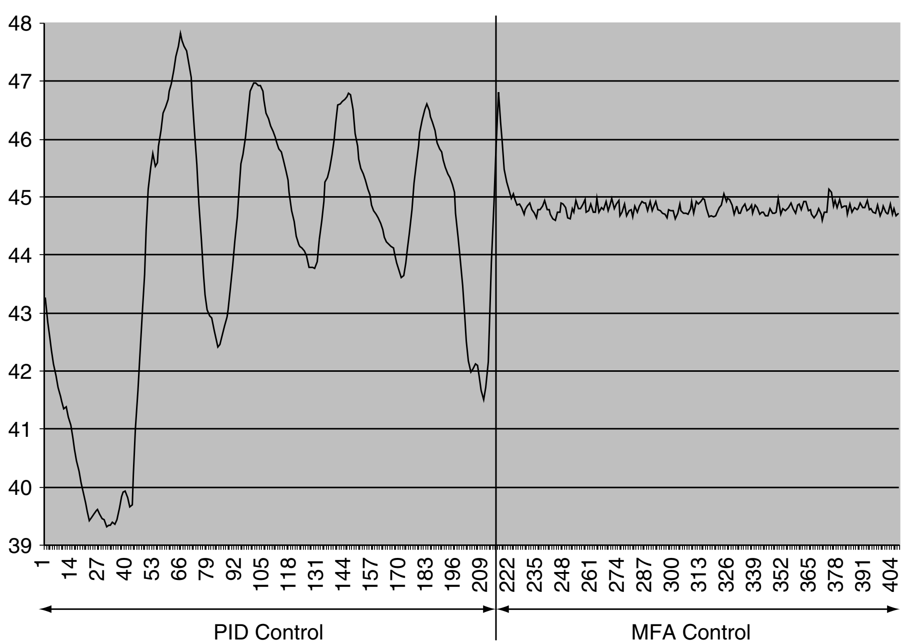

Figure 4.28 Comparison of key process variable controlled by PID and MFA.

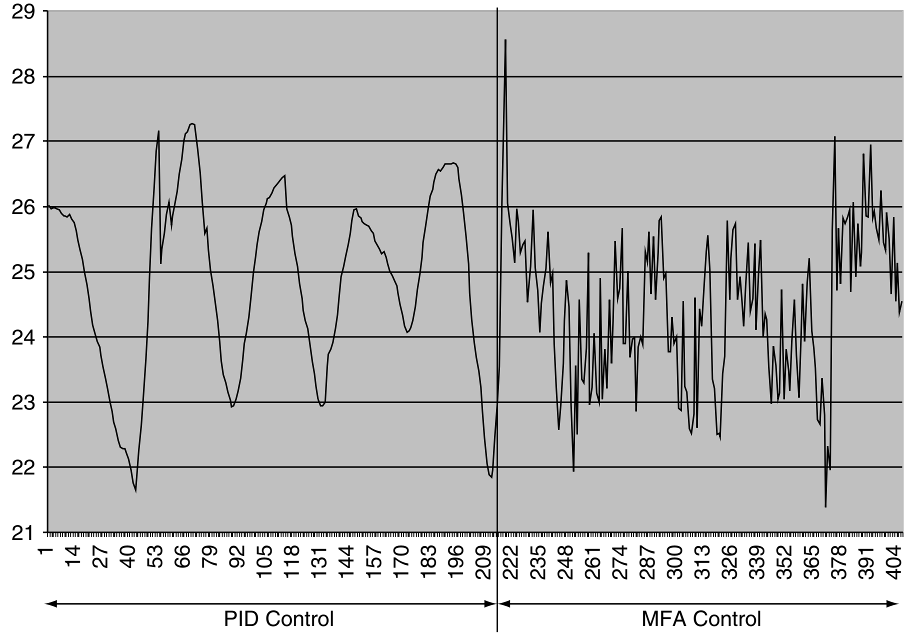

Figure 4.29 Reflux flow change from PID control to MFA control.

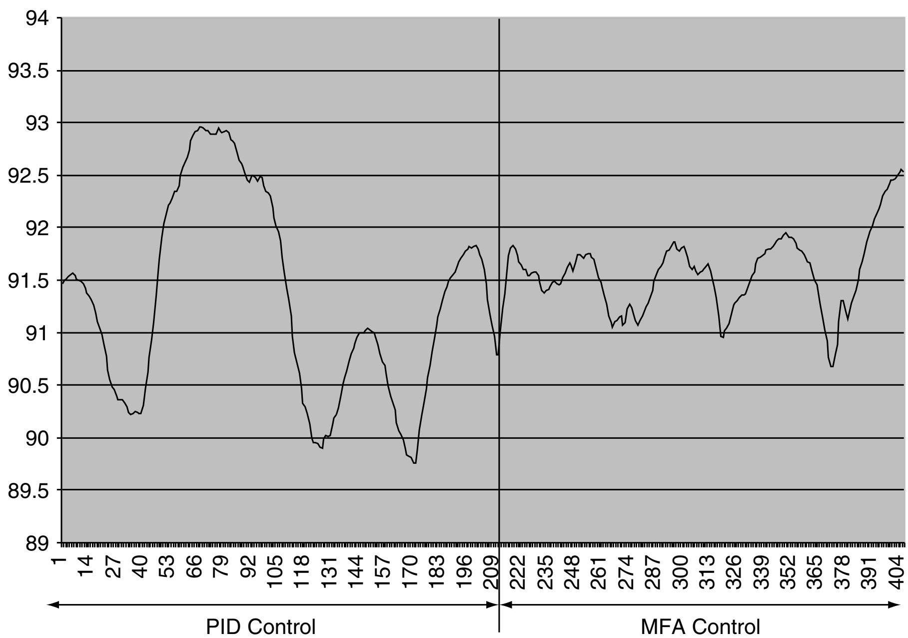

Figure 4.30 Principal column purity when switching from PID to MFA control.

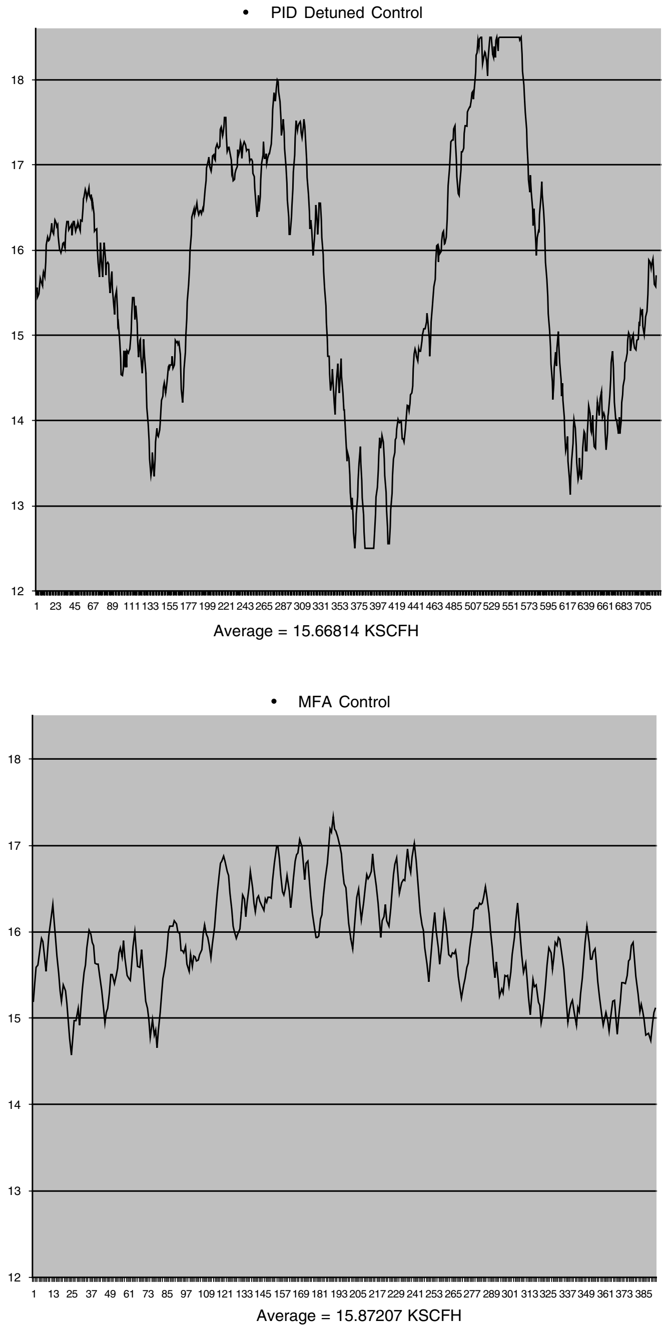

Figure 4.31 Improvement in gas flow yield.

According to Air Liquide's advanced control manager, by using model-free adaptive control, Air Liquide achieved benefits in the areas of product yield, quality control, and, most importantly, operational stability.

References

Anderson, B. D. O., et al. (1986). *Stability of Adaptive Systems: Passivity and Averaging Analysis*. MIT Press, Cambridge, MA.

Cheng, George S. (1996). *Model-Free Adaptive Control Theory and Applications*. A dissertation for the doctor of engineering degree, Shanghai Jiao-Tong University, Shanghai, China.

Cheng, George S. (2000). *MFA in Control with CyboCon*. CyboSoft, General Cybernation Group Inc., Rancho Cordova, CA.

Cheng, George S. (2001a). *CyboCon and Model-Free Adaptive Control Training Manual*. CyboSoft, General Cybernation Group Inc., Rancho Cordova, CA.

Cheng, George S., Min He, and De-Lin Li (1999). "Model-Free Coking Furnace Adaptive Control," in *Hydrocarbon Processing*. Gulf Publishing Company, Houston TX, December 1999.

Desoer, C.A., and M. Vidyasagar (1975). *Feedback Systems: Input–Output Properties*. Academic Press, New York.

Hang, Chang C., Tong H. Lee, and Weng K. Ho (1993). *Adaptive Control*. Instrument Society of America, Research Triangle Park, NC.

Morris, Charles (1999). "Adaptive and Predictive Controls Boost Product Quality," *Food Engineering*, December.

Murrill, P. (1981). *Fundamentals of Process Control Theory*. Instrument Society of America, Research Triangle Park, NC.

Seiver, Dave, and Ovidiu Marin (2001). "Air Separation Advances with MFA Control," *Control*, May.

Shinskey, F. G. (1979). *Process-Control Systems*, McGraw-Hill Book Co., New York.

VanDoren, Vance (2001). "Model Free Adaptive Control—This New Technique for Adaptive Control Addresses a Variety of Technical Challenges," *Control Engineering Europe*, March.

Suggested Reading

Åström, K. J. (1984). *Computer Controlled Systems, Theory and Design*. Prentice-Hall, Inc., New York.

Åström, K. J., and Tore Hagglund (1988). *Automatic Tuning of PID Controllers*. Instrument Society of America, Research Triangle Park, NC.

Åström, K. J., and B. Wittenmark (1989). *Adaptive Control*. Addison-Wesley, New York.

Athans, Michael, and Peter Falb (1966). *Optimal Control*, McGraw-Hill Book Co., New York.

Bennett, S., and D. A. Linkens, editors (1984). *Real-Time Computer Control*. Peter Peregrinus Ltd., London.

Buzzard, W. S. (1994). *Flow Control*. Instrument Society of America, Research Triangle Park, NC.

Cheng, George S. (2001b). *CyboCon CE MFA Control Instrument User Manual*. CyboSoft, General Cybernation Group Inc., Rancho Cordova, CA.