12

Control Paradigms

12.1 Introduction

Process control systems are normally complex with many control variables and many measured signals. The bottom-up approach is one way to design such systems. In this procedure the system is built up from simple components. The systems can be implemented in many different ways. Originally, it was done by interconnection of separate boxes built of pneumatic or electronic components. Today, the systems are typically implemented in distributed control systems consisting of several hierarchically connected computers. The software for the distributed control system is typically constructed so that programming can be done by selecting and interconnecting the components. The key component, the PID controller, has already been discussed in detail. In this chapter, we present some of the components required to build complex automation systems. We also present some of the key paradigms that guide the construction of complex systems.

A collection of paradigms for control is used to build complex systems from simple components. The components are controllers of the PID type, linear filters, and static nonlinearities. Typical nonlinearities are amplitude and rate limiters and signal selectors. Feedback is an important paradigm. Simple feedback loops are used to keep process variables constant or to make them change in specified ways. Feedback has been discussed extensively in the previous chapters. Another important paradigm is feedforward. This was discussed in Chapter 5. The key problem is to determine the control variables that should be chosen to control given process variables. Another problem is that there may be interaction between different feedback loops. This was discussed in Chapter 11.

Section 12.2 gives an overview of the problem to design complex systems, and the two approaches top-down and bottom-up design are compared. This section also gives an overview and presents the outline of the chapter. The chapter ends with an example to illustrate how the different components and paradigms can be used. The process considered is a chemical reactor, and the design is given in Section 12.9. Some important observations made in the chapter are finally summarized in Section 12.10.

12.2 Bottom-Up and Top-Down Approaches

There are two general approaches for designing a complex system: bottom up and top down. In the *bottom-up* or Lego approach the system is designed by combining small subsystems. The *top-down* approach starts with a general overall design that is refined successively. In practice the approaches are often combined. In both approaches we need knowledge about the elementary building blocks or components of the system. The bottom-up approach requires principles for combining basic components, and the top-down approach requires principles for refining or decomposing a high-level objective so that it can be accomplished by the basic system components. Several components and control principles for composition and decomposition have been described earlier in the book. In this section we will give an overview of the approaches, and in later sections we will describe components and paradigms that have not been discussed previously.

The Bottom-Up Approach

Large control systems can be built from controllers, filters, and nonlinear elements. The components can either be separate pieces of hardware or function blocks implemented in software that can be combined graphically using cut and paste. Controllers and filters have been discussed in Chapters 3, 5, 9, and 11. The nonlinear elements will be discussed in Section 12.6.

Control principles like feedback, feedforward, and model following have been discussed extensively in Chapters 3 and 5. Other important control principles such as *repetitive control*, *cascade control*, *mid-range control*, *split-range control*, *ratio control*, and *selector control* will be discussed in Sections 12.3, 12.4, 12.5, and 12.6.

An advantage with the bottom-up approach is that the system can be commissioned and tuned loop by loop. There may be difficulties when the loops are interacting. The disadvantage is that it is not easy to judge if additional loops will bring benefits. The system can also be unwieldy when loops are added.

Top-Down Solutions

Top-down paradigms often start with a problem formulation in terms of an optimization problem. Paradigms that support a top-down approach are optimization, state feedback, observers, predictive control, and linearization. In the top-down approach it is natural to deal with many inputs and many outputs simultaneously. Since this is not the main topic of this book we will only give a brief discussion. The top-down approach often leads to the controller structure shown in Figure 12.1. In this system all measured process variables y together with the control variables u are sent to an observer, which uses the sensor information and a mathematical model to generate a vector $\hat{x}$ of good estimates of internal process variables and important disturbances. The estimated state $\hat{x}$ is then compared with the ideal state x_m produced by the feedforward generator, and the difference is fed back to the process. The feedforward generator also gives a feedforward signal u_{ff}, which is sent directly to the process inputs. The controller shown in Figure 12.1 is useful for process segments where there are several inputs and outputs that interact, but the

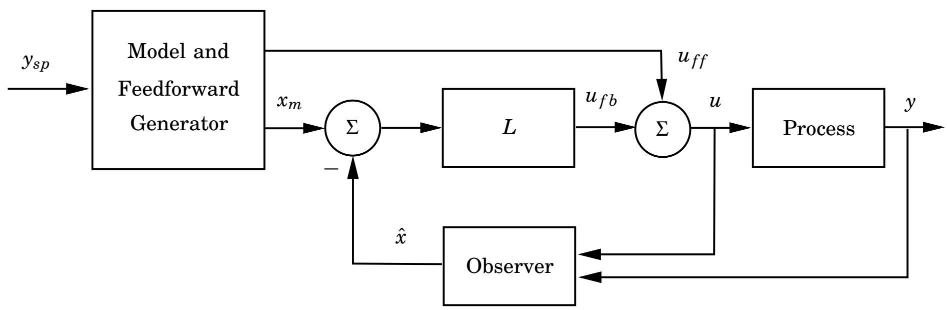

Figure 12.1 Block diagram of a controller based on model following, state feedback, and an observer.

system becomes very complicated when there is a large number of inputs and outputs. In such a case it may be better to decompose the system into several subsystems.

An advantage with the top-down approach is that the total behavior of the system is taken into account. A systematic approach based on mathematical modeling and simulation makes it easy to understand the fundamental limitations. Commissioning of the system is, however, difficult because many feedback loops have to be closed simultaneously. When using the top-down approach it is therefore good practice to first tune loops based on simulation, possibly also hardware in the loop simulation.

Soft Computing

Because of the widespread use of computers in control there has also been an influence on control from computer science. Two particular paradigms that originated from artificial intelligence are *neural networks* and *fuzzy control*, which both emerged from research in artificial intelligence. These paradigms are presented in Section 12.7 and Section 12.8. This branch of computer science is also called *soft computing*.

12.3 Repetitive Control

Attenuation of disturbances has been an essential theme in this book. For PID control we have focused on elimination of constant or slow disturbances. In this section we will show that similar ideas can be used to eliminate other types of disturbances, particularly periodic disturbances. Problems of this type are common when there are cyclic operations.

In Section 4.3 it was shown that attenuation of disturbances is captured by the transfer function from load disturbance to process output

$$G_{yd} = \frac{P}{1+PC}, \tag{12.1}$$

where P is the process transfer function and C the controller transfer function, respectively. By designing a controller that has high gain at a particular fre-

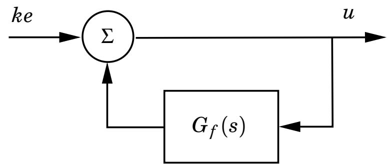

Figure 12.2 Block diagram of a controller with positive feedback of a filtered signal.

quency disturbances with that frequency are effectively reduced. The control error is zero in steady-state if the gain is infinite.

Consider the controller in Figure 12.2. Intuitively the system works as follows. The filter G_f filters out the signal component that we would like to eliminate, and the output of G_f is fed back to the input with positive feedback. The net effect is to create a high gain for the frequencies in the pass band of the filter G_f.

Constant and Sinusoidal Disturbances

To investigate the properties of the system analytically we observe that the controller has the transfer function

$$C(s) = \frac{k}{1 - G_f(s)}. \tag{12.2}$$

When $G_f(s)$ is a low-pass filter with transfer function

$$G_f(s) = \frac{1}{1 + sT},$$

we find that

$$C(s) = k\Big(1 + \frac{1}{sT}\Big),$$

which is the transfer function of a PI controller. Notice that the controller transfer function $C(s)$ has infinite gain at zero frequency, which implies that the steady-state error is zero for constant disturbances.

When $G_f(s)$ is the band-pass filter

$$G_f(s) = \frac{2\zeta\omega_0 s}{s^2 + 2\zeta\omega_0 s + \omega_0^2},$$

we find that

$$C(s) = k\frac{2\zeta\omega_0 s}{s^2 + \omega_0^2}.$$

Notice that this transfer function has infinite gain for $s = i\omega_0$, which implies that the steady-state error is zero for a sinusoidal disturbance of frequency ω_0.

Periodic Disturbances

Periodic disturbances can be reduced by choosing

$$G_f(s) = e^{-sL},$$

where L is the period of the disturbance. With this filter we find

$$C(s) = \frac{k}{1 - e^{-sL}}. \tag{12.3}$$

The relation between control error and control variable is

$$u(t) = ke(t) + u(t - L).$$

The control action at time t is thus a sum of the control error and the control signal at time $t - L$.

The controller has infinite gain for $s = 2n\pi i/L$, $n = 0, 1, \ldots$. A controller of this type is particularly useful when disturbances or set-point variations are periodic.

The transfer function from load disturbance to output (12.1) is

$$G_{yd}(s) = \frac{P(s)}{1 + P(s)C(s)} = \frac{P(s)(1 - e^{-sL})}{1 - e^{-sL} + kP(s)}.$$

The relation between load disturbance and output is then

$$(1 - e^{-sL} + kP(s))Y(s) = P(s)\big(1 - e^{-sL}\big)D(s).$$

Notice that the time function corresponding to $(1 - e^{-sL})D(s)$ is

$$d(t) - d(t - L),$$

which vanishes if D is a periodic disturbance with period L. The steady-state error caused by a periodic disturbance is thus zero.

The effective disturbance rejection does, however, come at a price that is illustrated with the following example.

Example 12.1—An Extreme Case

Consider a process with the transfer function

$$P(s) = e^{-sL},$$

with the controller

$$C(s) = \frac{1}{1 - e^{-sL}}.$$

that attenuates periodic disturbances.

The loop transfer function

$$G_l(s) = \frac{e^{-sL}}{1 - e^{-sL}}$$

is periodic with period $2\pi/L$, and its gain is infinite for $\omega = 2n\pi/L$. The frequency response is

$$G_l(i\omega) = -\frac{1}{2} - i\frac{\sin\omega L}{2(1-\cos\omega L)} = -\frac{1}{2} - i\frac{1}{\tan(\omega L/2)}.$$

The Nyquist curve is a vertical line through the point $G_l = -0.5$ and a half circle to the right. This curve is transversed once for $0 \le \omega \le 2\pi/L$ and infinitely many times when ω increases towards infinity.

The system has the gain margin 2 and the phase margin is 60°. The sensitivity functions are

$$S(s) = 1 - e^{-sL}$$
$$T(s) = e^{-sL},$$

and we find that $M_s = 2$ and $M_t = 1$.

A superficial look at traditional robustness measures like gain margin $g_m = 2$, phase margin $\varphi_m = 60°$, and maximum sensitivities $M_s = 2$ and $M_t = 1$ may indicate that the system is robust to process perturbations.

The fact that $T(i\omega) = 1$ for all frequencies is, however, an indication that the system has unusual properties. Further insight is obtained by analysing the effect of parameter variations. The system has only one parameter, the time delay L, and we will investigate the effects of variations in the time delay. To use the robustness inequality (4.32) we will convert time delay variations to an additive process perturbation. Assume that the time delay changes from L to $L + \delta L$, then

$$e^{-s(L+\delta L)} = e^{-sL}e^{-s\delta L} = e^{-sL} + e^{-sL}(e^{-s\delta L} - 1).$$

A variation in the time delay can thus be represented by the additive perturbation

$$\Delta P(s) = e^{-sL}(e^{-s\delta L} - 1).$$

Hence $|\Delta P(i\omega)| = |e^{-i\omega\delta L} - 1|$.

Since $|P(i\omega)| = 1$, the robustness inequality (4.32) becomes

$$\frac{|\Delta P(i\omega)|}{|P(i\omega)|} = |e^{-i\omega\delta L} - 1| < \frac{1}{|T(i\omega)|} = 1.$$

This inequality is not satisfied for any $\delta L > 0$ because the left-hand side is 2 and the right hand side is 1, and we cannot guarantee stability for an arbitrary small perturbation in the time delay. □

The example shows that the effective attenuation of periodic disturbances comes at the cost of the system being extremely sensitive to parameter variations. A compromise between disturbance attenuation can be made by replacing $G_f(s)$ in Figure 12.2 by $\alpha G_f(s)$ with $\alpha < 1$. The controllers obtained for

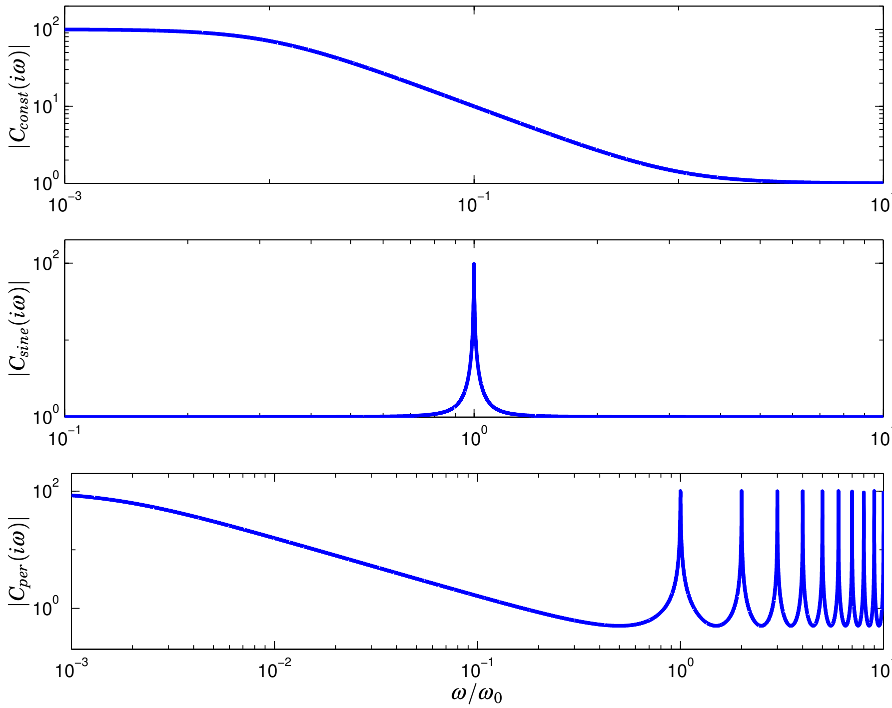

Figure 12.3 Gain curves of Bode plots for the controllers C_{const} (top), C_{sine} (middle), and C_{per} (bottom). The parameter α is 0.99 in all cases, which means that the largest gains of the controllers are 100. For the band-pass filter we have $\zeta = 0.1$, and for the repetitive controller we have $T = 2\pi/\omega_0$.

constant, sinusoidal, and periodic signals then become

$$
\begin{aligned}
C_{\text{const}}(s) &= \frac{1+sT}{1-\alpha+sT} \\
C_{\text{sine}}(s) &= \frac{s^2+2\zeta\omega_0 s+\omega_0^2}{s^2+2(1-\alpha)\zeta\omega_0 s+\omega_0^2} \\
C_{\text{per}}(s) &= \frac{1}{1-\alpha e^{-sT}}.
\end{aligned}
$$

The largest gains of the transfer functions are $1/(1-\alpha)$ in all cases. Choosing $\alpha < 1$ diminishes disturbance attenuation but improves the robustness. The properties of the controllers C_{const}, C_{sine}, and C_{per} are illustrated in Figure 12.3 which shows the gain curves of the Bode plots for the controllers. The controller C_{const} has high gain for low frequencies, the controller C_{sine} has high gain for $\omega = \omega_0$, and the controller C_{per} has high gain for the frequencies ω_0, $2\omega_0$, $3\omega_0$, etc.

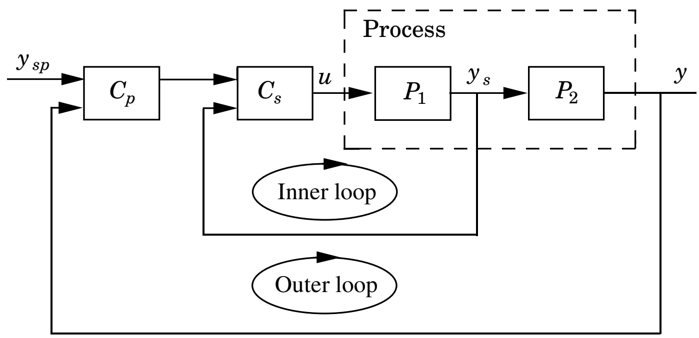

Figure 12.4 Block diagram of a system with cascade control.

12.4 Cascade Control

Cascade control can be used when there are several measurement signals and one control variable. It is particularly useful when there are significant dynamics, e.g., long dead times or long time constants, between the control variable and the process variable. Tighter control can then be achieved by using an intermediate measured signal that responds faster to the control signal. Cascade control is built up by nesting the control loops, as shown in the block diagram in Figure 12.4. The system in this figure has two loops. The inner loop is called *the secondary loop*; the outer loop is called *the primary loop*. The reason for this terminology is that the outer loop deals with the primary measured signal. It is also possible to have a cascade control with more nested loops. The performance of a system can be improved with a number of measured signals, up to a certain limit. If all state variables are measured, it is often not worthwhile to introduce other measured variables. In such a case the cascade control is the same as state feedback. We will illustrate the benefits of cascade control by an example.

EXAMPLE 12.2—IMPROVED LOAD DISTURBANCE REJECTION
Consider the system shown in Figure 12.4. Let the transfer functions be

$$P_1 = \frac{1}{s+1}$$

and

$$P_2 = \frac{1}{(s+1)^3}.$$

Assume that a load disturbance enters at the input of the process. There are significant dynamics from the control variable to the primary output. The secondary output does respond much faster than the primary output. Thus, cascade control can be expected to give improvements.

The dashed lines in Figure 12.5 show the response obtained with conventional feedback using a PI controller with the parameters $K = 0.37$ and $T_i = 2.2$. Since the response of the secondary measured variable to the control

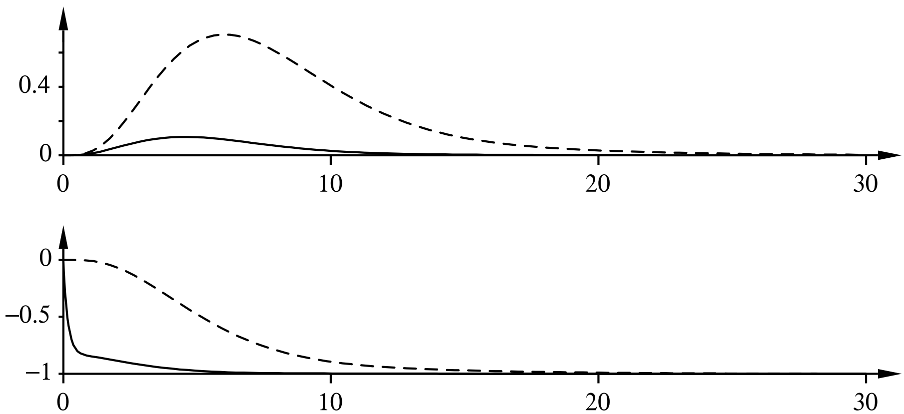

Figure 12.5 Responses to a load disturbance for a system with (solid line) and without (dashed line) cascade control. The upper diagram shows process output y, and the lower diagram shows control signal u.

signal is quite fast, it is possible to use high loop gains in the secondary loop. If the controller in the inner loop is proportional with gain K_s, the dynamics from the set point of C_s to process output becomes

$$G(s) = \frac{K_s}{(s+1+K_s)(s+1)^3}.$$

With $K_s = 5$ in the inner loop and PI control with $K = 0.55$ and $T_i = 1.9$ in the outer loop, the responses shown in solid lines Figure 12.5 are obtained. The figure shows that the disturbance response is improved substantially by using cascade control. Notice in particular that the control variable drops very much faster with cascade control. The main reason for this is the fast inner feedback loop, which detects the disturbance much faster than the outer loop.

The secondary controller is proportional, and the loop gain is 5. A large part of the disturbance is eliminated by the inner loop. The remaining error is eliminated at a slower rate through the action of the outer loop. In this case integral action in the inner loop will always give an overshoot in the disturbance response. □

Choice of Secondary Measured Variables

It is important to be able to judge whether cascade control can give improvement and to have a methodology for choosing the secondary measured variable. This is easy to do if we just remember that the key idea of cascade control is to arrange a tight feedback loop around a disturbance. In the ideal case the secondary loop can be so tight that the secondary loop is a perfect servo wherein the secondary measured variable responds very quickly to the control signal. The basic rules for selecting the secondary variable are:

- There should be a well-defined relation between the primary and secondary measured variables.

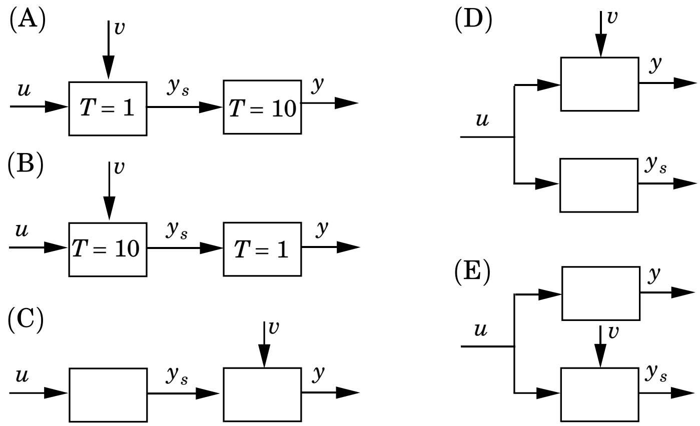

Figure 12.6 Examples of different process and measurement configurations.

- Essential disturbances should act in the inner loop.
- The inner loop should be faster than the outer loop. The typical rule of thumb is that the average residence times should have a ratio of at least five.
- It should be possible to have a high gain in the inner loop.

A common situation is that the inner loop is a feedback around an actuator. The reference variable in the inner loop can then represent a physical quantity, like flow, pressure, torque, velocity, etc., while the control variable of the inner loop could be valve pressure, control current, etc. This is also a typical example where feedback is used to make a system behave in a simple predictive way. It is also a very good way to linearize nonlinear characteristics.

A number of different control systems with one control variable and two measured signals are shown in Figure 12.6. In the figure the control variable is represented by u, the primary measured variable by y, the secondary measured variable by y_s, and the essential disturbance is v. With the rules given above it is only case A that is suitable for cascade control.

Choice of Control Modes

When the secondary measured signal is chosen it remains to choose the appropriate control modes for the primary and secondary controllers and to tune their parameters. The choice is based on the dynamics of the process and the nature of the disturbances. It is difficult to give general rules because the conditions can vary significantly. In critical cases it is necessary to analyze and simulate. It is, however, useful to have an intuitive feel for the problems.

Consider the system in Figure 12.4. To have a useful cascade control, it is necessary that the process P_2 be slower than P_1 and that the essential disturbances act on P_1. We assume that these conditions are satisfied. The secondary controller can often be chosen as a pure proportional controller or a PD controller. In some cases integral action can be useful to improve rejection of low-frequency disturbances. With controllers that lack integral action, there may be a static error in the secondary loop. This may not be a serious drawback. The secondary loop, as a rule, is used to eliminate fast disturbances. Slow disturbances can easily be eliminated by the primary loop, which will typically have integral action. There are also drawbacks to using integral control in the secondary loop. With such a system there will always be an overshoot in the response of the primary control loop. Integral action is needed if the process P_2 contains essential time delays and the process P_1 is such that the loop gain in the secondary loop must be limited.

The special case when the process P_2 is a pure integrator is quite common. In this case integral action in the inner loop corresponds to proportional control in the outer loop. If integral action is used in the inner loop, the proportional action in the outer loop must be reduced. This is a significant disadvantage for the performance of the system. A good remedy is to remove the integrator in the inner loop and to increase the gain in the outer loop.

Tuning and Commissioning

Cascade controllers must be tuned in a correct sequence. The outer loop should first be put in manual when the inner loop is tuned. The inner loop should then be put in automatic when tuning the outer loop. The inner loop is often tuned for critical or over-critical damping or equivalently for a small sensitivity (M_s). If this is not done there is little margin for using feedback in the outer loop.

Commissioning of cascade loops also requires some considerations. The following procedure can be used, starting with both controllers in manual mode.

1. Adjust the set point of the secondary controller to the value of the secondary process variable.
2. Set the secondary controller in automatic with internal set point selected.
3. Adjust the primary controller so that its set point is equal to the process variable and so that its control signal is equal to the set point of the secondary controller.
4. Switch the secondary controller to external set point.
5. Switch the primary controller to automatic mode.

The steps given above are automated to different degrees in different controllers. If the procedure is not done in the right way there will be switching transients.

Integral Windup

If integral action is used in both the secondary and primary control loops, it is necessary to have a scheme to avoid integral windup. The inner loop can be handled in the ordinary way, but it is not a trivial task to avoid windup in the outer loop. There are three situations that must be covered:

1. The control signal in the inner loop can saturate.
2. The secondary control loop may be switched to internal set point.
3. The secondary controller is switched from automatic to manual mode.

The feedback loop, as viewed from the primary controller, is broken in all these cases, and it is necessary to make sure that its integral mode is dealt with properly. This problem is solved automatically in a number of process controllers that have cascade control capabilities, but if we build up the cascade control using two independent controllers we have to solve the problem ourselves. This requires being able to inject a tracking signal into the primary controller.

If the output signal of the secondary controller is limited, the process variable of the secondary controller should be chosen as the tracking signal in the primary controller. This also requires a digital transfer from the secondary to the primary controller telling it when the tracking is to take place.

In the case where the secondary controller switches to work according to its local set point instead of the external one from the primary controller, the local set point should be sent back to the primary controller as a tracking signal. In this way one can avoid both integrator windup and jumps in the transition to cascade control.

When the secondary controller switches over to manual control, the process variable from the secondary controller should be sent back to the primary controller as a tracking signal.

Some Applications

Cascade control is a convenient way to use extra measurements to improve control performance. The following examples illustrate some applications.

EXAMPLE 12.3—VALVE POSITIONERS
Control loops with pneumatic valves are a very common application. In this case the inner loop is a feedback around the valve itself where the valve position is measured. The inner loop reduces the influences of pressure variations and various nonlinearities in the pneumatic system. □

EXAMPLE 12.4—MOTOR CONTROL
Figure 12.7 is a block diagram of a typical motor control system. This system has three cascaded loops. The innermost loop is a current loop where the current is measured. The next loop is the velocity loop, which is based on measurement of the velocity. The outer loop is a position loop. In this case integral action in the velocity loop is equivalent to proportional action in the position loop. Furthermore, it is clear that the derivative action in the position loop is equivalent to proportional action in the velocity loop. From this it follows directly that there is no reason to introduce integral action in the velocity controller or derivative action in the position controller. □

EXAMPLE 12.5—HEAT EXCHANGER
A schematic diagram of a heat exchanger is shown in Figure 12.8. The purpose of the control system is to control the outlet temperature on the secondary side by changing the valve on the primary side. The control system shown uses cascade control. The secondary loop is a flow control system around the valve. The control variable of the primary loop is the set point of the flow

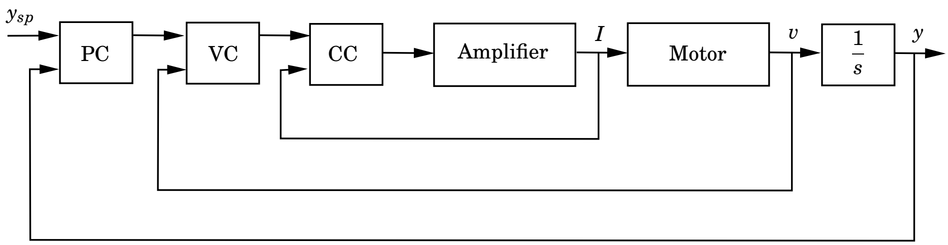

Figure 12.7 Block diagram of a system for position control. The system has three cascaded loops with a current controller (CC) with feedback from current (I), a velocity controller (VC) with feedback from velocity (v), and a position controller (PC) with feedback from position (y).

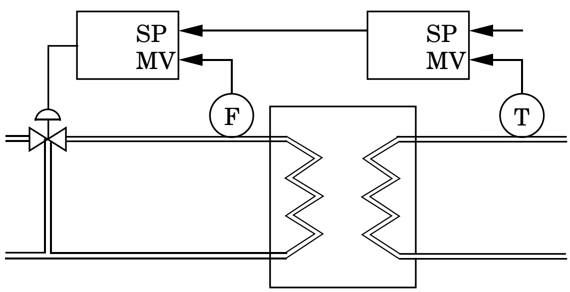

Figure 12.8 Schematic diagram of a heat exchanger with cascade control.

controller. The effect of nonlinearities in the valve, as well as flow and pressure disturbances, is thus reduced by the secondary controller. □

12.5 Mid-Range and Split-Range Control

Cascade control is a strategy where one control signal and two measurement signals are used to meet the control objective. The dual situation is when two control signals are used to control one measurement signal. The two control signals are sometimes used one at a time. This is the case in split-range control. In other situations it is necessary to use the two control signals simultaneously. A common situation is mid-range control or mid-ranging. Mid-range and split-range control are discussed in this section.

Mid-Range Control

The problem treated by mid-range control is illustrated in Figure 12.9. The figure illustrates an example where two valves are used to control a flow. One valve, v_1, is small but has a high resolution. The other valve, v_2, is large but has a low resolution.

Suppose that the small valve v_1 is in the middle of its operating range and that only small disturbances are acting on the system. In this case, one controller that manipulates valve v_1 is able to take care of the control problem.

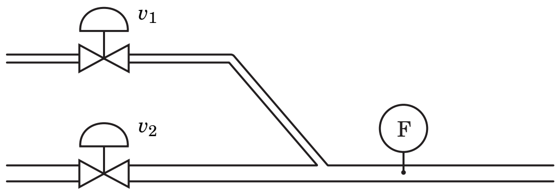

Figure 12.9 Two valves are used to control the flow.

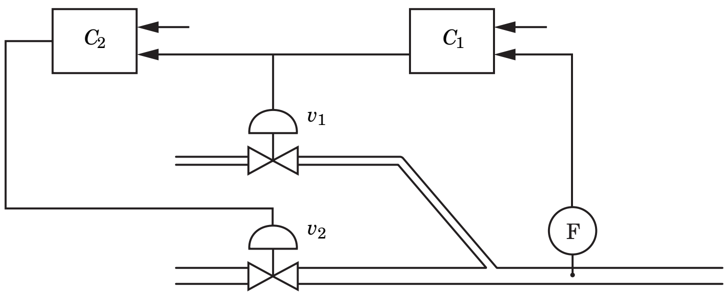

Figure 12.10 Mid-range control

However, when larger disturbances occur, valve v_1 will saturate. In this case, the larger valve v_2 must also be manipulated.

The mid-range control strategy is illustrated in Figure 12.10. Controller C_1 takes the set point y_{sp} and flow signal y as inputs and manipulates the small valve v_1. A second controller, C_2, takes the control signal from C_1 as input and tries to control it to a set point u_{sp} in the middle of its operating range by manipulating the large valve v_2. If both controllers have integral action, the flow will be at the set point y_{sp} and the valve v_1 will be at the set point u_{sp} in steady state.

A block diagram of the mid-range control strategy is given in Figure 12.11. Process P_1 and controller C_1 together form a fast feedback loop. The mid-ranging controller C_2 controls the valve position of controller C_1 via the process output y. This means that the output of controller C_1 is controlled by driving the process output y away from the set point. If this is done slowly, the deviation

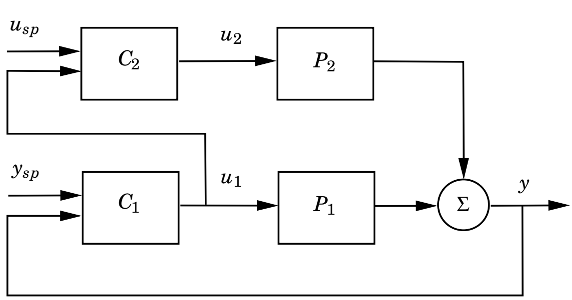

Figure 12.11 Block diagram of a system with mid-range control.

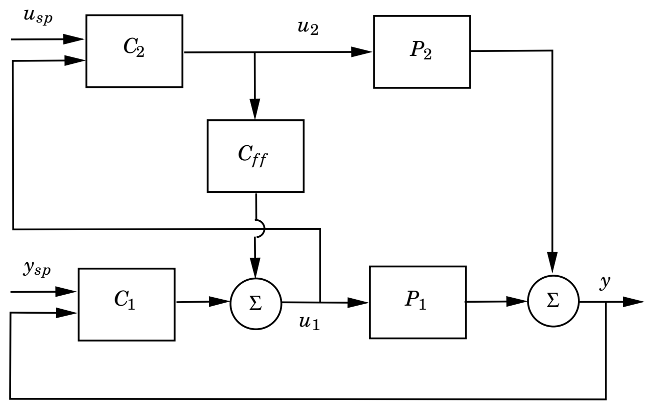

Figure 12.12 Block diagram of a system with mid-range control.

from the set point can be kept small. If not, it is recommended to use the structure given in Figure 12.12.

In Figure 12.12 a feedforward signal is added from control signal u_2 to controller C_1. If the feedforward compensator is

$$C_{ff}(s) = -\frac{P_2(s)}{P_1(s)},$$

controller C_2 will perform the mid-ranging control without any disturbance of the process output y.

It is likely that the small valve will saturate. In spite of this, it is not necessary that the controller C_1 has anti-windup. Since the control signal is controlled by the controller C_2, controller C_2 prevents controller C_1 from winding up.

Split-Range Control

In split-range control, the control is shared by two controllers that perform the control one at a time. Systems of this type are common, e.g., in connection with heating and cooling. One physical device is used for heating and another for cooling. The heating and cooling systems often have different static and dynamic characteristics. The principle of split-range control is illustrated in Figure 12.13, which shows the static relation between the measured variables and the control variables. When the temperature is too low, it is necessary to supply heat. The heater, therefore, has its maximum value when the measured variable is zero. It then decreases linearly until mid-range, where no heating is supplied. Similarly, there is no cooling when the measured variable is below mid-range. Cooling, however, is applied when the process variable is above mid-range, and it then increases.

There is a critical region when switching from heating to cooling. To avoid both heating and cooling at the same time, there is often a small dead zone where neither heating nor cooling is supplied. Switching between the different control modes may cause difficulties and oscillations.

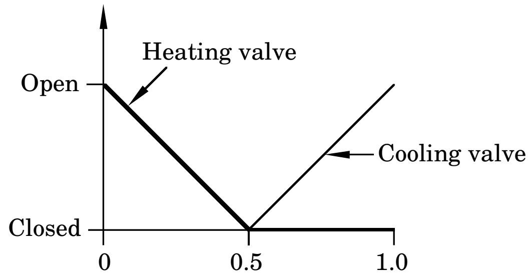

Figure 12.13 Illustration of the concept of split-range control.

Split-range control is commonly used in systems for heating and ventilation. It is also useful applications when the control variable ranges over a very large range. The flow is then separated into parallel paths, each controlled with a valve.

12.6 Nonlinear Elements

Nonlinear elements have been discussed before. In Section 3.5 we used a limiter to avoid integral windup in a controller with integral action. In Chapter 9 it was shown that controllers could be tuned by relay feedback and that performance could be improved by gain scheduling. In this section we describe more nonlinear elements and also present some control paradigms that guide the use of these elements.

Linearization

The nonlinearity in sensors and actuators can be compensated in a straightforward way. Consider, for example, an actuator that has the characteristics

$$v = f(u),$$

where v is the actual process input signal, and u is the control signal. To compensate for the nonlinearity we simply compute the control signal u_c as if the actuator was linear with unit gain. The control law

$$u = f^{-1}(u_c),$$

where f^{-1} is the inverse of the actuator nonlinearity, then gives

$$v = f(u) = f(f^{-1}(u_c)) = u_c.$$

The actuated process signal is then identical to u_c as was desired.

The same idea can be applied to sensors. Consider, for example, a sensor that has the nonlinearity $g(x)$. By designing a linear controller based on the assumption that the sensor is linear with unit gain and feeding the signal

$$y_c = g^{-1}(y)$$

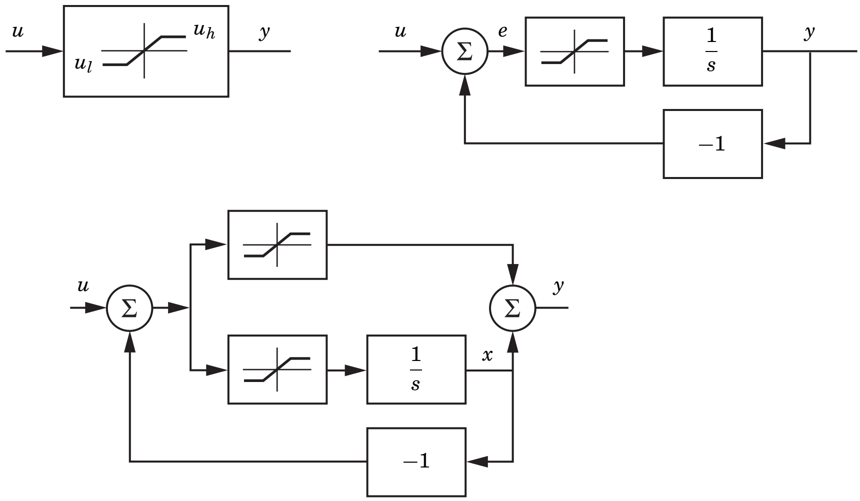

Figure 12.14 Block diagram of a simple amplitude limiter (upper left), a rate limiter (upper right), and a jump and rate limiter or a ramp unit (lower).

to the controller, the sensor nonlinearity is eliminated.

Similar ideas can be applied to process nonlinearities, but the compensation is not ideal because of dynamics. There is a technique for compensating for nonlinearities called feedback linearization, but this is outside the scope of this book. There are also situations when the nonlinearities are beneficial.

Limiters

Since all physical values are limited, it is useful to have limiting devices in control systems too. Limiters are used in many different ways. They can be used to limit the command signals so that we are not generating set points that are demanding larger or faster changes than a system can cope with.

A block diagram of a simple amplitude limiter is shown in upper left part of Figure 12.14. The limiter can mathematically be described as the static nonlinearity

$$y = \mathrm{sat}(u, u_l, u_h) = \begin{cases} u_l & \text{if } u \le u_l \\ u & \text{if } u_l < u < u_h \\ u_h & \text{if } u \ge u_h \end{cases} .$$

where u_l and u_h are the saturation limits.

It is also useful to limit the rate of change of signals. This can be done with the *rate limiter* or the *ramp unit* shown in the upper right part of Figure 12.14. The output follows the input signal if the rate of change of the input is smaller than the rate limit. In steady state the inputs and the outputs are identical because there is integral action in the system. Since the output is generated by an integrator with limited input signal, the rate of change of the output will be limited to the bounds given by the limiter. It is possible to use different limits for increasing or decreasing rates.

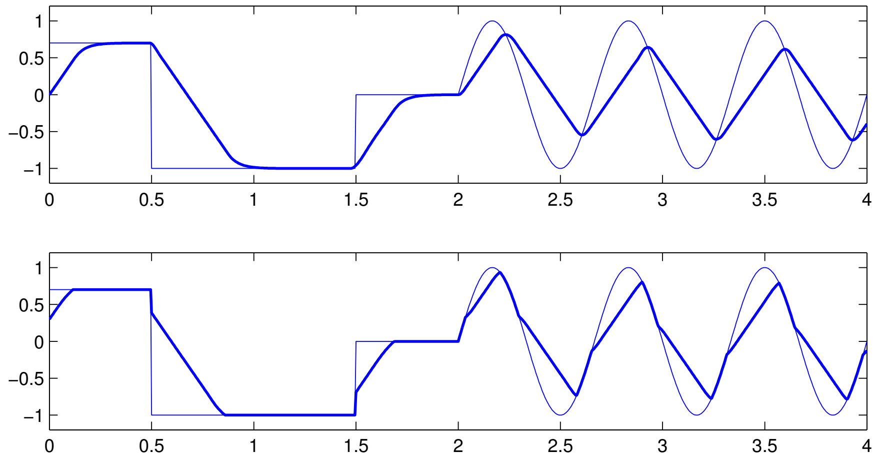

Figure 12.15 Simulation of a rate limiter (upper), and a jump and rate limiter (lower). The thin line shows the input to the limiter and the thick line shows the output of the limiter

A more sophisticated limiter called a *jump and rate limiter* is shown in the lower part of Figure 12.14. The output will follow the input for small changes in the input signal. At large changes, the output will follow the input with a limited rate. The jump and rate limiter can be described by

$$\frac{dx}{dt} = \text{sat}(u - x, -a, a)$$
$$y = x + \text{sat}(u - x, -a, a),$$

If $|u - x| \leq a$ it follows from the equations describing the system that $y = u$, and if $u \geq x + a$ it follows that $dx/dt = a$. Thus, the output signal will approach the input signal at the rate a.

The properties of the different limiters are illustrated in the simulation shown in Figure 12.15. The input signal consists of a few steps and a sinusoid. The upper curve shows a rate limiter where the rate limit is 4. The figure shows that the rate of change of the output is limited. The response to a sinusoidal input shows clearly that the rate limiter gives a phase lag. The lower curve shows the response of a jump and rate limiter. Notice that the output follows rapid changes in the input as long as the difference between x and u are less than the jump limit, which is 0.5. The rate is limited to 4.

Surge Tank Control

The control problems that were discussed in Chapter 4 were all regulation problems where the task was to keep a process variable as close to a given set point as possible. There are many other control problems that also are important. Surge tank control is one example. The purpose of a surge tank is to act as a buffer between different production processes. Flow from one

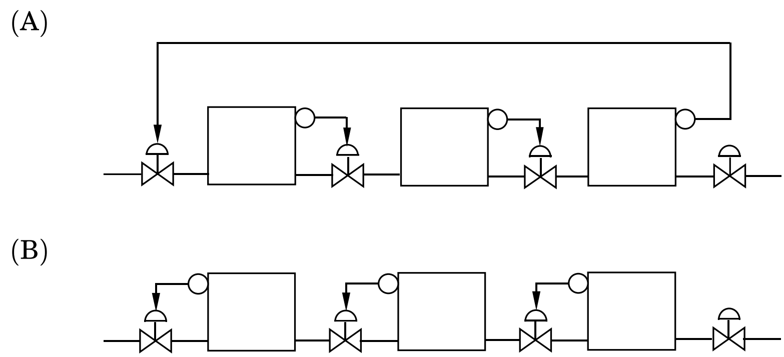

Figure 12.16 Different structures for surge tank control. The material flow is from the left to the right. The scheme in A is called control in the direction of the flow. The scheme in B is called control in the direction opposite to the flow.

process is fed to another via the surge tank. Variations in production rate can be accommodated by letting the level in the surge tank vary. Conventional level control, which attempts to keep the level constant, is clearly not appropriate in this case. To act as a buffer the level should indeed change. It is, however, important that the tank neither become empty nor overflow.

There are many approaches to surge tank control. A common, simple solution is to use a proportional controller with a low gain. Controllers with dead zones or nonlinear PI controllers are also used. Gain scheduling is a better method. The scheduling variable is chosen as the tank level. A controller with low gain is chosen when the level is between, e.g., 10 percent and 90 percent, and a controller with high gain is used outside the limits. There are also special schemes for surge tank control.

In many cases there are long sequences of surge tanks and production units, as illustrated in Figure 12.16. Two different control structures, control in the direction of the flow or opposite to the flow, are shown in the figure. Control in the direction opposite to the flow is superior because then all control loops are characterized by first-order dynamics. With control in the direction of the flow, it is easy to get oscillations or instabilities because of the feedback from the end of the chain to the beginning.

Ratio Control

Ratio control is applied when the control objective is to keep the ratio between two variables, often flows, at a certain ratio a. In combustion, for example, it is desired to control the fuel-to-air supply ratio, in order for the combustion to be as efficient as possible. Blending of chemicals is another example where it is desired to keep the ratio between different flows constant. In in-line blending systems, when there are no downstream mixing tanks, this is of special importance. If the composition is not maintained, quality problems may occur.

Ratio control is normally solved in the way shown in Figure 12.6. There are two control loops. The main loop consists of process P_1 and controller C_1. Output y_1 is the main flow, and the external set point r_1 is the desired main flow. In the second loop, consisting of process P_2 and controller C_2, it is attempted to

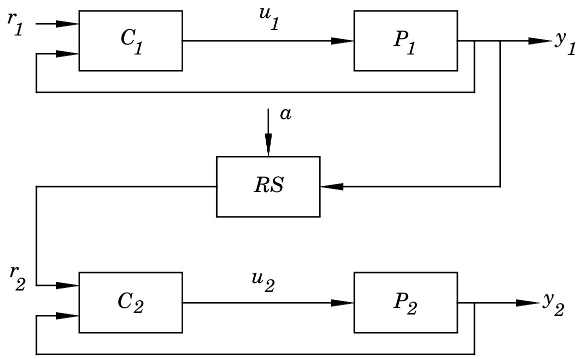

Figure 12.17 Ratio control using a Ratio station *(RS)* applied to main flow y_1.

control the flow y_2 so that the ratio y_2/y_1 is equal to ratio a. In Figure 12.6 this is obtained using a Ratio station where set point r_2 is determined by

$$r_2(t) = ay_1(t), \tag{12.4}$$

i.e., simply by multiplying the main flow y_1 with the desired ratio a.

In Equation 12.4, parameter a is assumed to be constant. This is not necessary. The desired ratio a is often time-varying. In combustion, for example, the ratio a is often adjusted based on O_2 measurements in the exhaust.

Provided the controllers have integral action, the solution given in Figure 12.6 will work in steady state, i.e., $y_1 = r_1$ and $y_2 = ay_1$. However, the simple Ratio station is not efficient during transients. The second flow y_2 will always be delayed compared to the desired flow ay_1. The length of this delay is determined by the dynamics of the second loop.

When set point r_1 is increasing, the delay causes an under-supply of the media corresponding to flow y_2, and conversely when r_1 is decreasing there is an excess of the media corresponding to flow y_2. There are cases when it is important never to get any under-supply of one of the two media. In the combustion case, one gets an under-supply of air during the transient part when the external set point increases, but an excess of air when the set point decreases. To prevent the fuel from not being fully burnt by an under-supply of air, the solution in Figure 12.6 has to be complemented with some logic using selectors. This is discussed in the next section.

The main drawback with the simple Ratio station approach shown in Figure 12.6 is that the secondary flow y_2 is delayed compared to the desired flow ay_1. This problem can be solved if not only y_1 is used to form the secondary set point, but also the main set point r_1. The structure, called the Blend station, is shown in Figure 12.18.

In the Blend station, the secondary set point is determined as

$$r_2(t) = a\left(\gamma r_1(t) + (1-\gamma)y_1(y)\right). \tag{12.5}$$

Gain γ is a weighting factor that determines the relation between set point r_1

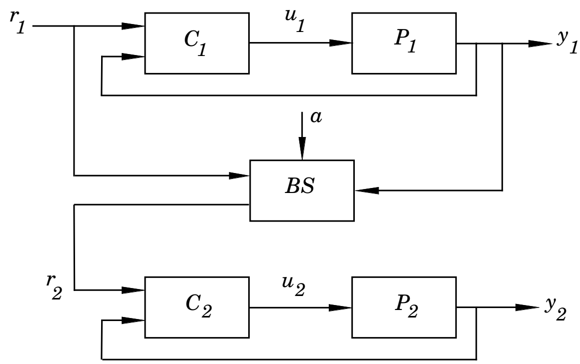

Figure 12.18 Ratio control using the Blend station *(BS)*.

and main flow y_1 when forming secondary set point r_2. When $\gamma = 0$, the Blend station is identical to the Ratio station.

EXAMPLE 12.6—PULP BLEACHING CONTROL

The Ratio station and the Blend station have been applied on a bleaching section in a paper mill. The pulp is bleached by adding Hydrosulphite to the pulp flow. The goal is to keep the ratio between the pulp flow and the Hydrosulphite flow constant.

The upper diagram in Figure 12.19 shows control using the Ratio station. The pulp flow controller, C_1, is a PI controller with setting $K_1 = 0.2$ and $T_{i1} = 4s$. The Hydrosulphite controller, C_2, is also a PI controller with setting $K_2 = 0.078$ and $T_{i2} = 1.07s$. The figure shows responses to two set-point changes in the pulp flow. The Hydrosulphite flow is scaled with the desired ratio and translated, so that the desired flow rates become identical. The figure shows that the Ratio station provides the correct ratio in steady state, but also that there is a deviation between the two flows during the transients. The Hydrosulphite flow is delayed compared to the pulp flow.

The lower diagram in Figure 12.19 shows the results obtained when using the Blend station with gain factor $\gamma = 0.75$. Here, the difference between the two flows is almost eliminated.

□

Selector Control

Selector control can be viewed as the inverse of split-range control. In split range there is one measured signal and several actuators. In selector control there are many measured signals and only one actuator. A selector is a static device with many inputs and one output. There are two types of selectors: *maximum* and *minimum*. For a maximum selector the output is the largest of the input signals.

There are situations where several controlled process variables must be taken into account. One variable is the primary controlled variable, but it is also required that other process variables remain within given ranges. Selector

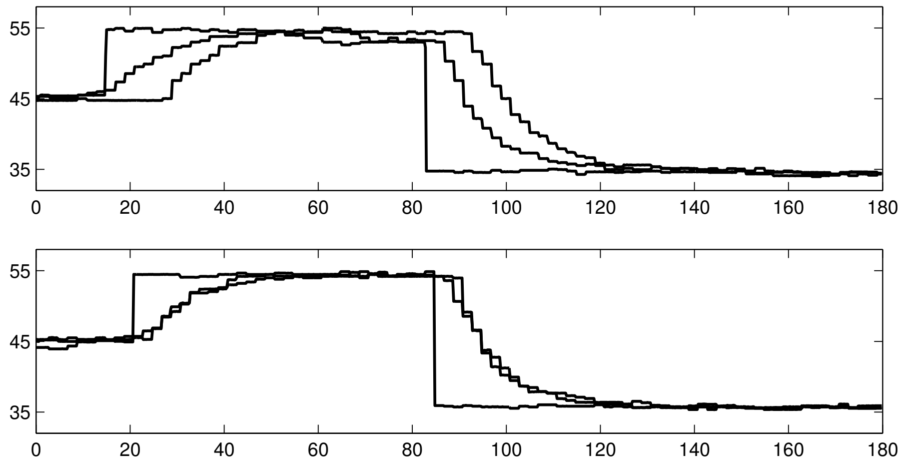

Figure 12.19 Ratio control of a pulp bleaching process using the original Ratio station (upper) and the Blend station with gain $\gamma = 0.75$ (lower). The figure shows two changes in the pulp set point, the pulp flow (fastest response) and the Hydrosulphite flow (slowest response).

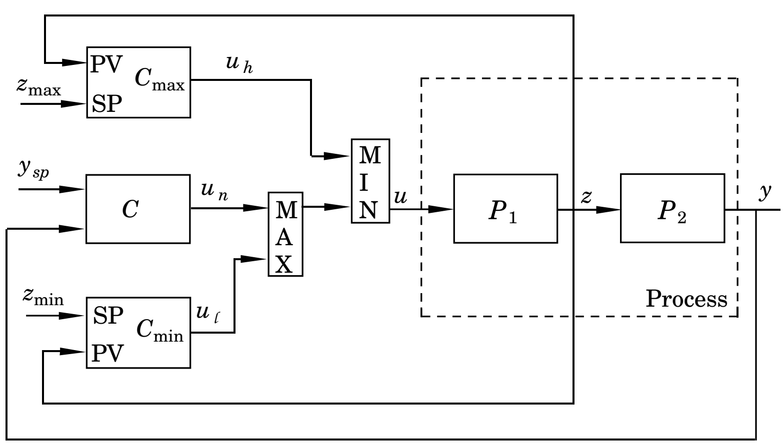

Figure 12.20 Selector control.

control can be used to achieve this. The idea is to use several controllers and to have a selector that chooses the controller that is most appropriate. One example of use is where the primary controlled variable is temperature and we must ensure that pressure does not exceed a certain range for safety reasons.

The principle of selector control is illustrated in Figure 12.20. The primary controlled variable is the process output y. There is an auxiliary measured variable z that should be kept within the limits $z_{\min}$ and $z_{\max}$. The primary controller C has process variable y, set point y_{sp}, and output u_n. There are also secondary controllers with measured process variables that are the auxiliary variable z and with set points that are bounds of the variable z. The outputs of these controllers are u_h and u_l. The controller C is an ordinary PI or PID controller that gives good control under normal circumstances. The output of the minimum selector is the smallest of the input signals; the output of the maximum selector is the largest of the inputs.

Under normal circumstances the auxiliary variable is larger than the minimum value $z_{\min}$ and smaller than the maximum value $z_{\max}$. This means that the output u_h is large and the output u_l is small. The maximum selector, therefore, selects u_n, and the minimum selector also selects u_n. The system acts as if the maximum and minimum controller were not present. If the variable z reaches its upper limit, the variable u_h becomes small and is selected by the minimum selector. This means that the control system now attempts to control the variable z and drive it towards its limit. A similar situation occurs if the variable z becomes smaller than $z_{\min}$.

In a system with selectors, only one control loop at a time is in operation. The controllers can be tuned in the same way as single-loop controllers. There may be some difficulties with conditions when the controller switches. With controllers having integral action, it is also necessary to track the integral states of those controllers that are not in operation. Selector control is very common in order to guarantee that variables remain within constraints. The technique is commonly used in the power industry for control in boilers, power systems, and nuclear reactors. The advantage is that it is built up of simple nonlinear components and PI and PID controllers. An alternative to selector control is to make a combination of ordinary controllers and logic. The following example illustrates the use of selector control.

EXAMPLE 12.7—AIR-FUEL CONTROL
In the previous section we discussed air-fuel control using ratio control. When the Ratio station is used, there may be lack of air because the set point of the air controller increases first when the fuel controller has increased the oil flow. One way to solve this problem is to use the Blend station. However, the system cannot compensate for perturbations in the air channel. This problem can be treated using selectors, such as is shown in Figure 12.21. The system uses one minimum and one maximum selector. There is one PI controller for fuel flow and one PI controller for the air flow. The set point for the air controller is the larger of the command signal and the fuel flow. This means that the air flow will increase as soon as more energy is demanded. Similarly, the set point to the fuel flow is the smaller of the demand signal and the air flow. This means that when demand is decreased, the set point to the dual flow controller will immediately be decreased, but the set point to the air controller will remain high until the oil flow has actually decreased. The system thus ensures that there will always be an excess of air. □

Median Selectors

A median selector is a device with many inputs and many outputs. Its output selects the input that represents the current median of the input signals. A special case is the two-out-of-three selector, commonly used for highly sensitive systems. To achieve high reliability it is possible to use redundant sensors and controllers. By inserting median selectors it is possible to have a system that will continue to function even if several components fail.

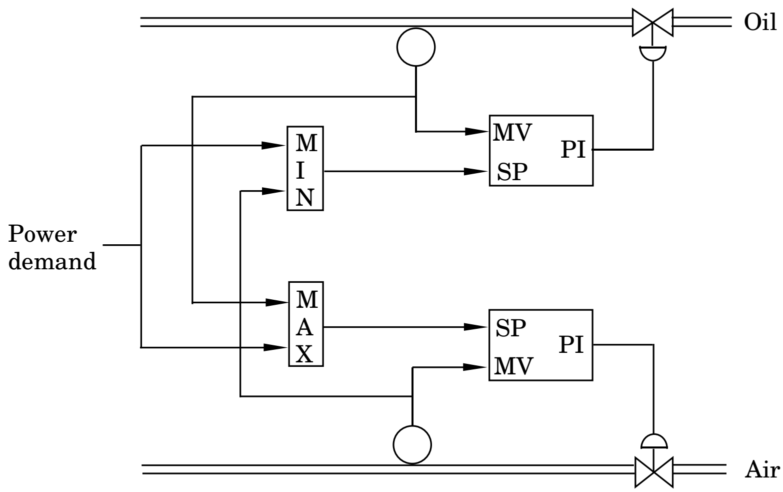

Figure 12.21 Air-fuel controller based on selectors.

12.7 Neural Network Control

In the previous section, we have seen that simple nonlinearities can be used very effectively in control systems. In this and the following section, we will discuss some techniques based on nonlinearities, where the key idea is to represent functions of several variables in a compact way. The ideas have been introduced under the names of *neural* and *fuzzy control*. At first sight, these methods may seem quite complicated, but once the colorful language is stripped off we find that the algorithms have natural representations as implementations of nonlinear functions. It is a nontrivial problem to find good representations of a nonlinear function. If we simply try to grid the variables and use an interpolation we find that the number of entries in the table for representing the function grows very rapidly with the number of variables. For example, if n variables are gridded in N points each we find that the number of entries are N^n. For a function of five variables with $N = 100$ we find that 10^{10} entries are required. Another useful property of neural networks is that there are methods to fit the parameters of the function to data.

Neural Networks

Neural networks originated in attempts to make simple models for neural activity in the brain and attempts to make devices that could recognize patterns and carry out simple learning tasks. A brief description that captures the essential idea follows.

A Simple Neuron A schematic diagram of a simple neuron is shown in Figure 12.22. The system has many inputs and one output. If the output is y

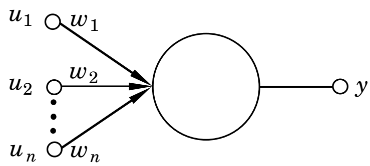

Figure 12.22 Schematic diagram of a simple neuron.

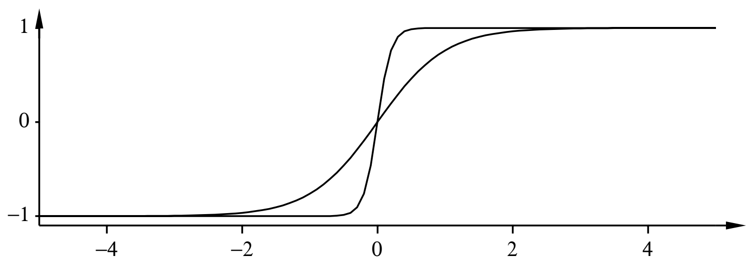

Figure 12.23 Sigmoid functions.

and the inputs are u_1, u_2, ... , u_n the input-output relation is described by

$$y = f(w_1u_1 + w_2u_2 + \cdots + w_nu_n) = f\left(\sum_{k=1}^{n} w_iu_i\right), \tag{12.6}$$

where the numbers w_i are called weights. The function f is a so-called sigmoid function, illustrated in Figure 12.23. Such a function can be represented as

$$f(x) = \sinh \alpha x = \frac{e^{\alpha x} - e^{-\alpha x}}{e^{\alpha x} + e^{-\alpha x}} \tag{12.7}$$

where α is a parameter. This model of a neuron is thus simply a nonlinear function. Some special classes of functions can be approximated by (12.6).

Neural Networks More complicated models can be obtained by connecting neurons together as shown in Figure 12.24. This system is called a neural network or a neural net. The adjective feedforward is often added to indicate that the neurons are connected in a feedforward manner. There are also other types of neural networks. In the feedforward network, the input neurons are connected to a layer of neurons, the outputs of the neurons in the first layer are connected to the neurons in the second layer, and so on, until we have the outputs. The intermediate layers in the net are called hidden layers.

Each neuron is described by Equation (12.6). The input-output relation of a neural net is thus a nonlinear static function. Conversely, we can consider a neural net as one way to construct a nonlinear function of several variables. The neural network representation implies that a nonlinear function of several variables is constructed from two components: a single nonlinear function, the sigmoid function (12.7), which is a scalar function of one variable; and linear operations. It is thus a simple way to construct a nonlinearity from simple operations. A key reason why neural networks are interesting is that practically

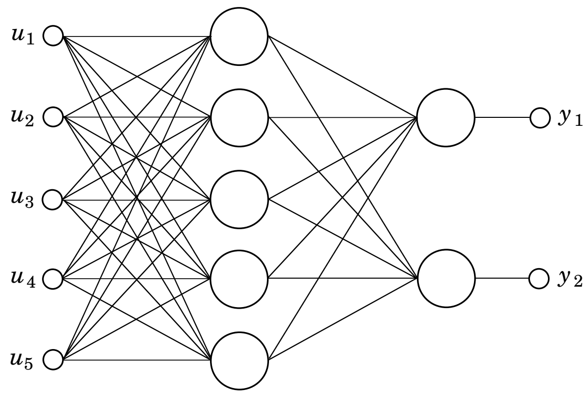

Figure 12.24 A feedforward neural network.

all continuous functions can be approximated by neural networks having one hidden layer. It has been found practical to use more hidden layers because then fewer weights can be used. Another practical feature of the sigmoidal functions is that the approximations are local.

Learning Notice that there are many parameters (weights) in a neural network. Assuming that there are n neurons in a layer, if all neurons are connected, n^2 parameters are then required to describe the connections between two layers.

Another interesting property of a neural network is that there are so-called learning procedures. This is an algorithm that makes it possible to find parameters (weights) so that the function matches given input-output values. The parameters are typically obtained recursively by giving an input value to the function and the desired output value. The weights are then adjusted so that the data is matched. A new input-output pair is then given, and the parameters are adjusted again. The procedure is repeated until a good fit has been obtained for a reasonable data set. This procedure is called training a network. A popular method for training a feedforward network is called back propagation. For this reason the feedforward net is sometimes called a back-propagation network. Fitting a neural network to experimental data is illustrated in Figure 12.25. A nice feature is that it is possible to find both the function and its inverse. The inverse function is useful when compensating for nonlinearities in sensors and actuators.

Control Applications A feedforward neural network can be viewed as a nonlinear function of several variables with a training procedure. The function has many parameters (weights) that can be adjusted by the training procedure so that the function will match given data. Even if this is an extremely simplistic model of a real neuron, it is a useful system component. In process control we can often make good use of nonlinear functions. Sensor calibration is one case. There are many situations where an instrument has many different sensors, the outputs of which must be combined nonlinearly to obtain the desired measured value. Nonlinear functions can also be used for pattern recognition.

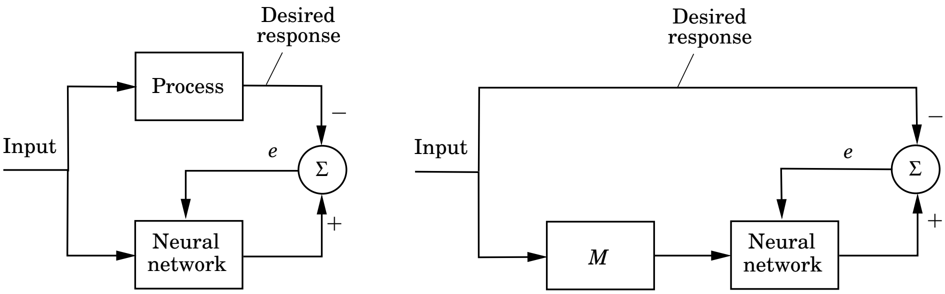

Figure 12.25 Illustration of training of a simple feedforward network. The block diagram on the left shows training of a function, and the figure on the right shows training of an inverse function.

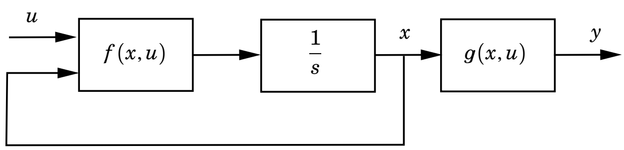

Figure 12.26 Implementation of a nonlinear dynamical system using integrators and neural networks.

It is also possible to model dynamic systems by combining the neural network with integrators as is illustrated in Figure 12.26. The system in the figure implements the nonlinear system

$$\frac{dx}{dt} = f(x, u)$$
$$y = g(x, u),$$

where the nonlinear functions are represented by neural networks.

12.8 Fuzzy Control

Fuzzy control is an old control paradigm that has received a lot of attention recently. In this section we will give a brief description of the key ideas. We will start with fuzzy logic, which has inspired the development.

Fuzzy Logic

Ordinary Boolean logic deals with quantities that are either true or false. Fuzzy logic is an attempt to develop a method for logic reasoning that is less sharp. This is achieved by introducing linguistic variables and associating them with *membership functions*, which take values between 0 and 1. In fuzzy control the logical operations *and*, *or*, and *not* are operations on linguistic variables. These operations can be expressed in terms of operations on the membership functions of the linguistic variables. Consider two linguistic variables with the

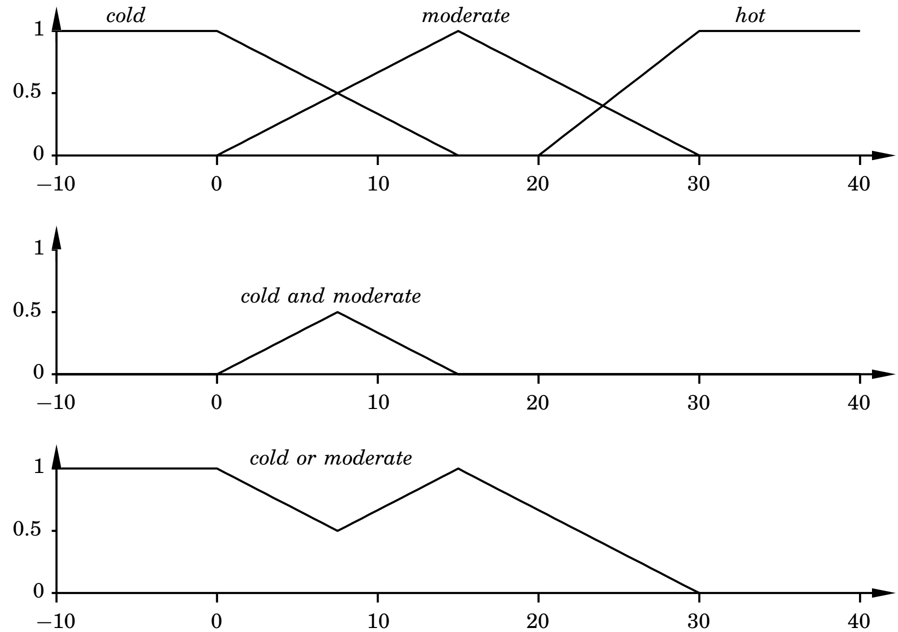

Figure 12.27 Illustration of fuzzy logic. The upper diagram shows the membership functions of *cold*, *moderate*, and *hot*. The middle diagram shows the membership functions for *cold and moderate* the lower diagram shows the membership functions for *cold or moderate*.

membership functions $f_A(x)$ and $f_B(x)$. The logical operations are defined by the following operations on the membership functions.

$$\begin{aligned} f_{A \text{ and } B} &= \min(f_A(x), f_B(x)) \\ f_{A \text{ or } B} &= \max(f_A(x), f_B(x)) \\ f_{\text{not } A} &= 1 - f_A(x). \end{aligned}$$

A linguistic variable, where the membership function is zero everywhere except for one particular value, is called a crisp variable.

Assume, for example, that we want to reason about temperature. For this purpose we introduce the linguistic variables *cold*, *moderate*, and *hot*, and we associate them with the membership functions shown in Figure 12.27. The membership function for the linguistic variables *cold and moderate* and *cold or moderate* are also shown in the figure.

A Fuzzy Controller

A block diagram of a fuzzy PD controller is shown in Figure 12.28. The control error, which is a continuous signal, is fed to a linear system that generates the derivative of the error. The error and its derivative are converted to so-called linguistic variables in a process called "fuzzification." This procedure converts continuous variables to a collection of linguistic variables. The number of linguistic variables is typically quite small, for example: negative large

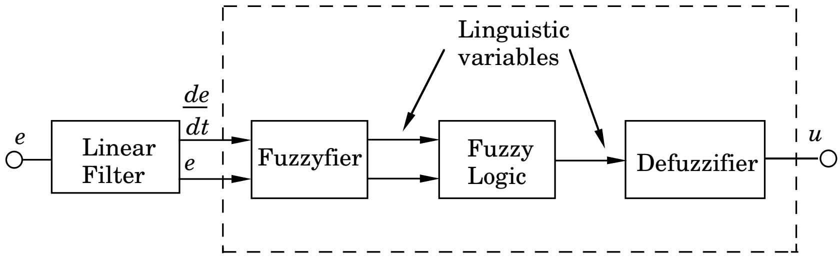

Figure 12.28 A fuzzy PD controller.

(*NL*), negative medium (*NM*), negative small (*NS*), zero (*Z*), positive small (*PS*), positive medium (*PM*), and positive large (*PL*). The control strategy is expressed in terms of a function that maps linguistic variables to linguistic variables. This function is defined in terms of a set of rules expressed in fuzzy logic. As an illustration we give the rules for a PD controller where the error and its derivative are each characterized by three linguistic variables (*N*, *Z*, *P*) and the control variable is characterized by five linguistic variables (*NL*, *NM*, *Z*, *PM*, and *PL*).

```
Rule 1: If e is N and de/dt is P then u is Z
Rule 2: If e is N and de/dt is Z then u is NM
Rule 3: If e is N and de/dt is N then u is NL
Rule 4: If e is Z and de/dt is P then u is PM
Rule 5: If e is Z and de/dt is Z then u is Z
Rule 6: If e is Z and de/dt is N then u is NM
Rule 7: If e is P and de/dt is P then u is PL
Rule 8: If e is P and de/dt is Z then u is PM
Rule 9: If e is P and de/dt is N then u is Z}
```

These rules can also be expressed in table form; see Table 12.1. The membership functions representing the linguistic variables normally overlap (see Figure 12.27). Due to this, several rules contribute to the control signal. The linguistic variable representing the control signal is calculated as a weighted sum of the linguistic variables of the control signal. The linguistic variable representing the control signal is then mapped into a real number by an operation called "defuzzification." More details are given in the following.

Fuzzy Inference Many different shapes of membership functions can be used. In fuzzy control it is common practice to use overlapping triangular shapes like the ones shown in Figure 12.27 for both inputs and control variables. Typically only a few membership functions are used for the measured variables.

Fuzzy logic is only used to a moderate extent in fuzzy control. A key issue is to interpret logic expressions of the type that appears in the description of the fuzzy controller. Some special methods are used in fuzzy control. To describe these we assume that f_A, f_B, and f_C are the membership functions associated with the linguistic variables A, B, and C. Furthermore let x and y represent

Table 12.1 Representation of the fuzzy PD controller as a table.

		$\frac{de}{dt}$ P	$\frac{de}{dt}$ Z	$\frac{de}{dt}$ N
	N	Z	NM	NL
e	Z	PM	Z	NM
	P	PL	PM	Z

measurements. If the values x_0 and y_0 are measured, they are considered as crisp values. The fuzzy statement

```
If x is A and y is B
```

is then interpreted as the crisp variable

$$z^0 = \min(f_A(x_0), f_B(y_0))$$

where *and* is equivalent to minimization of the membership functions. The linguistic variable u defined by

```
If x is A or y is B then u is C
```

is interpreted as a linguistic variable with the membership function

$$f_u(x) = z^0 f_C(x).$$

If there are several rules, as in the description of the PD controller, each rule is evaluated individually. The results obtained for each rule are combined using the `or` operator. This corresponds to taking the maximum of the membership functions obtained for each individual rule.

Figure 12.29 is a graphical illustration for the case of the first two rules of the PD controller. The figure shows how the linguistic variable corresponding to each rule is constructed and how the control signal is obtained by taking the maximum of the membership functions obtained from all rules.

The inference procedure described is called "product-max." This refers to the operations on the membership functions. Other inference procedures are also used in fuzzy control. The `and` operation is sometimes represented by taking the product of two membership functions and the `or` operator by taking a saturated sum. Combinations of the schemes are also used. In this way it is possible to obtain "product-max" and "min-sum" inferences.

Defuzzification Fuzzy inference results in a control variable expressed as a linguistic variable and defined by its membership function. To apply a control signal we must have a real variable. Thus, the linguistic variable defining the control signal must be converted to a real number through the operation of "defuzzification." This can be done in several different ways. Consider a

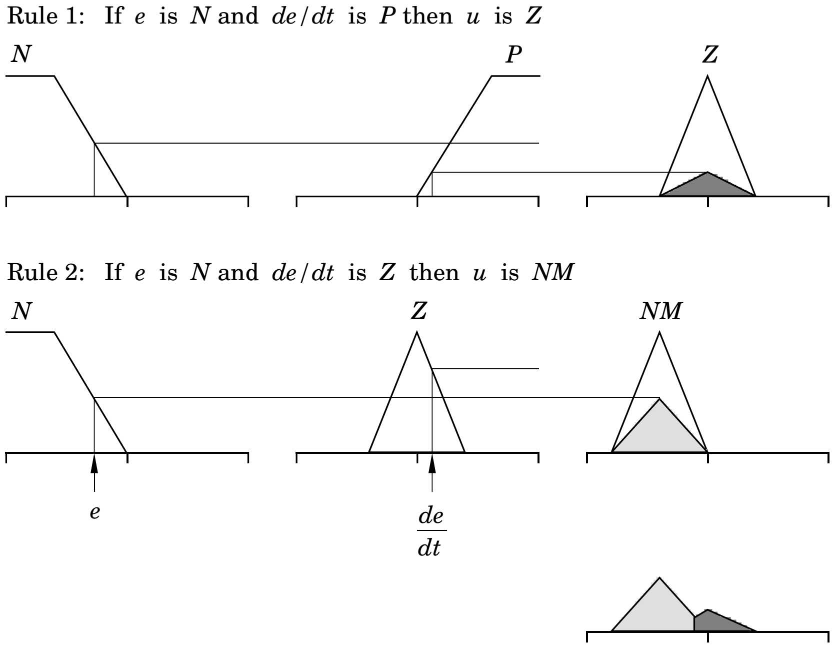

Figure 12.29 Illustration of fuzzy inference with two rules using the min-max rule.

linguistic variable A with the membership function $f_A(x)$. Defuzzification by mean values gives the value

$$x_0 = \frac{\int x f_A(x)dx}{\int f_A(x)dx}.$$

Defuzzification by the centroid gives a real variable x_0 that satisfies

$$\int_{-\infty}^{x_0} f_A(x)dx = \int_{x_0}^{\infty} f_A(x)dx.$$

Nonlinear Control

Having gone through the details, we return to the fuzzy PD controller in Figure 12.28. We first notice that the operations fuzzification, fuzzy logic, and defuzzification can be described in a very simple way. Stripping away the vocabulary and considering the final result, a fuzzy controller is nothing but a nonlinear controller. The system in Figure 12.28 can in fact be expressed as

$$u = F\left(e, \frac{de}{dt}\right),$$

where F is a nonlinear function of two variables. Thus, the fuzzy PD controller is a controller where the output is a nonlinear function of the error e and its derivative de/dt. In Figure 12.30 we give a graphic illustration of the

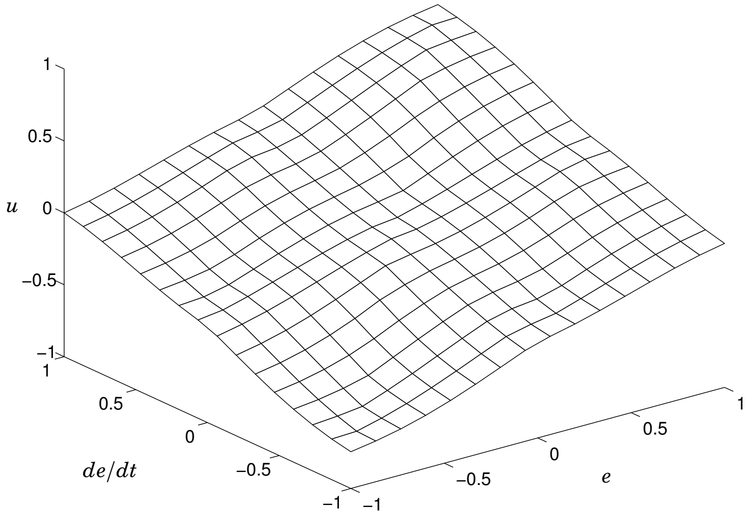

Figure 12.30 Graphic illustration of the nonlinearity of the fuzzy controller showing control signal u as function of control error e and its derivative.

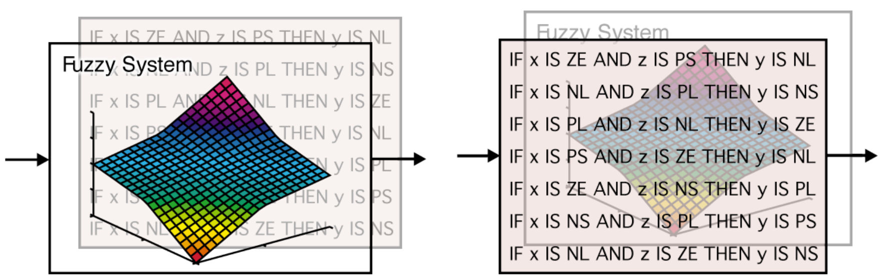

Figure 12.31 Two views of a fuzzy controller. The figure on the left shows that the fuzzy controller can be viewed as a nonlinear controller. The figure on the right instead emphasizes the rules.

nonlinearity defined by given rules for the PD controller with standard triangular membership functions and product fuzzification. The figure shows that the function is close to linear. In this particular case the fuzzy controller will behave similarly to an ordinary linear PD controller.

Fuzzy control may be considered as a way to represent a nonlinear function. This is illustrated in Figure 12.31. Notice that it is still necessary to deal with the generation of derivatives or integrals, integral windup, and all the other matters in the same way as for ordinary PID controllers. We may also inquire as to when it is useful to introduce the nonlinearities and what shape they should have.

Representation of a nonlinearity by fuzzification, fuzzy logic, and defuzzification is not very different from representation of a nonlinear function as a table with an interpolation procedure. Roughly speaking, the function values correspond to the rules; the membership functions and the fuzzification and defuzzification procedures correspond to the interpolation mechanism. To illustrate this we consider a function of two variables. Such a function can be visualized as a surface in two dimensions. A linear function is simply a tilted plane. This function can be described completely by three points on a plane, i.e., three rules. More complex surfaces or functions are obtained by using more function values. The smoothness of the surface is expressed by the interpolation procedures.

From the point of view of control, the key question is understanding when nonlinearities are useful and what shape they should have. These are matters where much research remains to be done. There are cases where the nonlinearities can be very beneficial but also cases where the nonlinearities cause problems. It is also a nontrivial task to explore what happens. A few simulations of the behavior is not enough because the response of a nonlinear system is strongly amplitude dependent.

Let us also point out that the properties of the controller in Figure 12.28 are strongly influenced by the linear filter used. It is thus necessary to limit the high-frequency gain of the approximation of the derivative. It is also useful to take derivatives of the process output instead of the error, as was discussed in Section 3.3. Other filters can also be used; by adding an integrator to the output of the system in Figure 12.28, we obtain a fuzzy PI controller.

Applications

The representation of the control law as a collection of rules for linguistic variables has a strong intuitive appeal. It is easy to explain heuristically how the control system works. This is useful in communicating control strategies to persons with little formal training. It is one reason why fuzzy control is a good tool for automation of tasks that are normally done by humans. In this approach it is attempted to model the behavior of an operator in terms of linguistic rules. Fuzzy control has been used in a number of simple control tasks for appliances. It has also been used in controllers for processes that are complicated and poorly known. Control of a cement kiln is one example of this type of application. Fuzzy control has also been used for controller tuning.

12.9 System Structuring

In this section we illustrate how complex control systems can be built from simple components by using the paradigms we have discussed. The problem is quite complex. It involves selection of measured variables and control variables, and it requires significant physical understanding of the process.

The Process

The process to consider is a chemical reactor. A schematic diagram is shown in Figure 12.32. Two substances A and B are mixed in the reactor. They react to form a product. The reaction is exothermic, which means that it will generate heat. The heat is dissipated through water that is circulating in cooling pipes

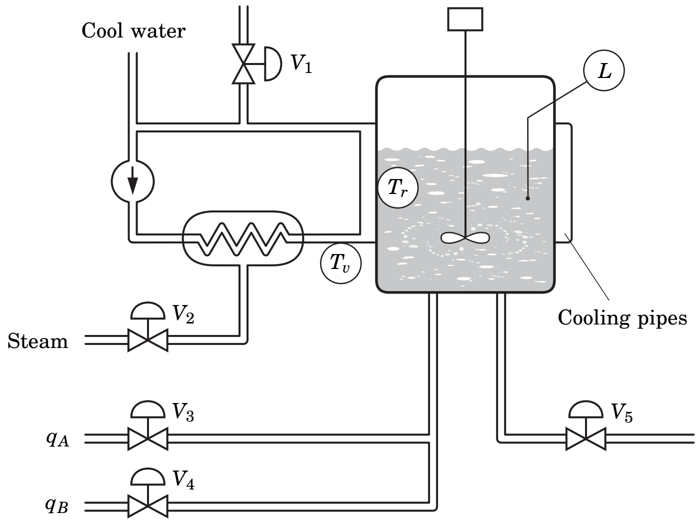

Figure 12.32 Schematic diagram of a chemical reactor.

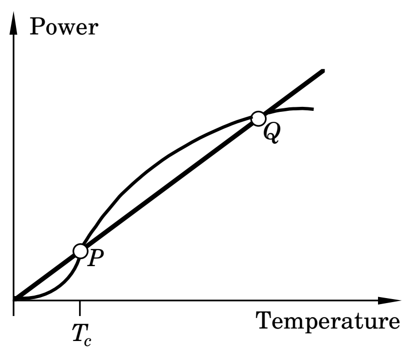

Figure 12.33 Static process model for the exothermic reactor.

in the reactor. The reaction is very fast; equilibrium is achieved after a time that is much shorter than the residence time of the reactor. The flow q_A of substance A is considerably larger than q_B. Efficiency of the reaction and the heat generation is essentially proportional to the flow q_B.

A static process model is useful in order to understand the control problem. Figure 12.33 shows the efficiency and the heat generation as a function of temperature. A model of this type was derived in Section 2.5. In the figure we have drawn a straight line that corresponds to the cooling power. There are equilibria where the power generated by the reaction is equal to the cooling power represented at points P and Q in the figure. The point P corresponds to an unstable equilibrium. It follows from Figure 12.33 that if the temperature is increased above P the power generated by the reaction is larger than the cooling power. Temperature will thus increase. The catalyst in the reactor may be damaged if the temperature becomes too high. Similarly, if the temperature decreases below point P it will continue to decrease and the reaction stops. This phenomenon is called "freezing." Freezing starts at the surface of the cooling tube and will spread rapidly through the reactor. If this happens the reactor must be switched off and restarted again.

Design Requirements

There are considerable risks in running an exothermic reactor. The reactor can explode if the temperature is too high. To reduce the risk of explosion, the reactors are placed in special buildings far away from the operator. Because of the risk of explosion, it is not feasible to experiment with controller tuning. Consequently, it is necessary to compute controller setting beforehand and verify that the settings are correct before starting the reactor. Safety is the overriding requirement of the control system. It is important to guarantee that the reaction temperature will not be too high. It is also important to make sure that process upsets do not lead to loss of coolant flow and that stirring does not lead to an explosion. It is also desirable to operate the reactor efficiently. This means that freezing must be avoided. Besides, it is desirable to keep the efficiency as high as possible. Because of the risks, it is also necessary to automate start and stop as well as normal operation. It is desirable to avoid having to run the reactor under manual control. In this particular case the operator can set two variables: the reactor temperature and the ratio between the flows q_A and q_B. The reaction efficiency and the product quality can be influenced by these two variables.

Controller Structure

The reactor has five valves. Two of them, V_1 and V_2, influence the coolant temperature. The flow of the reactor is controlled by V_3 and V_4, and the product flow is controlled by the valve V_5. In this particular application the valve V_5 is controlled by process steps downstream. (Compare this with the discussion of surge tanks in Section 12.6).

There are five measured signals: the reactor temperature T_r, the level in the reactor tank L, the cooling temperature T_v, and the flows q_A and q_B. The physical properties of the process give a natural structuring of the control system. A mass balance for the material in the reactor tank shows that the level is essentially influenced by the flow q_A and the demanded production. It follows from the stoichiometry of the reaction that the ratio of the flows q_A and q_B should be kept constant for an efficient reaction. The reactor temperature is strongly influenced by the water temperature, by the temperature of the coolant flow, and the flows q_A and q_B. Coolant temperature is influenced by the valve V_1 that controls the amount of flow and by the steam valve V_2.

This simple physical discussion leads to the diagram shown in Figure 12.34, which shows the causality of the variables in the process. The valve V_5 can be regarded as a disturbance because it is set by downstream process units. Figure 12.34 suggests that there are three natural control loops:

- Level control: Controlling the tank level with valve V_3.
- Temperature control: Control of the reactor temperature with valves V_1 and V_2.

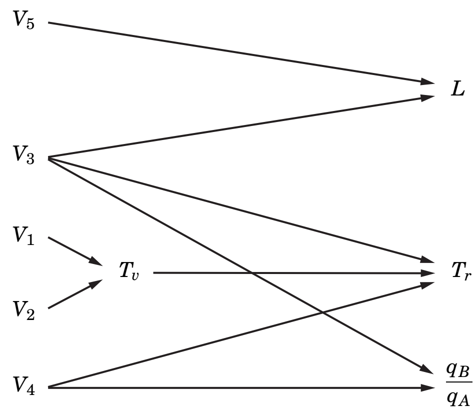

Figure 12.34 Causality diagram for the process variable.

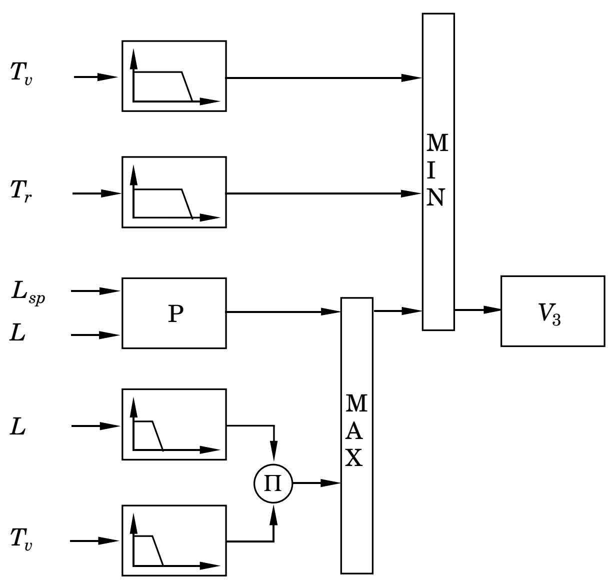

Figure 12.35 Block diagram for the level control through valve V_3.

- Flow ratio control: Control of ratio q_B/q_A with valve V_4.

These control loops are discussed in detail.

Level Control

The block diagram for the level control is shown in Figure 12.35. The primary function is a proportional feedback from the level to the flow q_A, which is controlled by the valve V_3. The reactor is also used as a surge tank to smooth out the difference between actual production and commanded production. The level in the tank will vary during normal operations. Reasonable limits are that the level should be between 50 percent and 100 percent. If the proportional band of the controller is chosen as 50 percent, the control variable will be

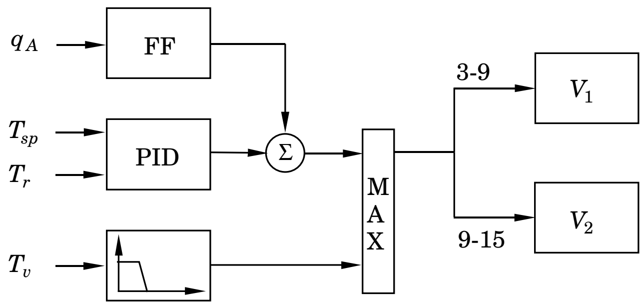

Figure 12.36 Block diagram showing temperature control through valves V_1 and V_2.

fully closed when the tank is full and half-open when the tank is half-full. It is important that the reactor temperature remains within given bounds. The flow q_A is constrained, therefore, by two selectors based on measurements of the temperature in the reactor tank (T_r) and the coolant temperature (T_v). When starting the reactor the level is kept at the lower limit until the coolant temperature becomes sufficiently high. This is achieved by a combination of limiters, multipliers, and selectors, as shown in Figure 12.35.

Temperature Control

Figure 12.36 gives a block diagram for controlling the reactor temperature. Since the chemical reaction is fast compared to temperature and flow dynamics, the reactor can be viewed as a heat exchanger from the control point of view. During normal conditions the temperature is controlled by adjusting the coolant flow through the valve V_1. The primary control function is a feedback from temperature to the valves V_1 and V_2. The set point in this control loop can be adjusted manually. The parameters of this control loop can be determined as follows. The transfer function from coolant flow to the reactor temperature is approximately given by

$$G(s) = \frac{K_p}{(1+sT_1)(1+sT_2)}, \tag{12.8}$$

where the time constant typically has values $T_1 = 300$ s and $T_2 = 50$ s. The following rough calculation gives approximate values of the controller parameter. A proportional controller with gain K gives the loop transfer function

$$G_0(s) = \frac{KK_p}{(1+sT_1)(1+sT_2)}. \tag{12.9}$$

The characteristic equation of the closed loop becomes

$$s^2 + s\left(\frac{1}{T_1} + \frac{1}{T_2}\right) + \frac{1+KK_p}{T_1T_2} = 0.$$

The closed system is thus of second order. The relative damping ζ and the undamped natural frequency ω are given by

$$2\zeta\omega = \frac{1}{T_1} + \frac{1}{T_2} \approx \frac{1}{T_2} \tag{12.10}$$

and

$$2\zeta\omega^2 = \frac{1 + KK_p}{T_1 T_2}. \tag{12.11}$$

The approximation in the first expression is motivated by $T_1 \gg T_2$. With a relative damping $\zeta = 0.5$ the Equation (12.10) then gives $\omega \approx 1/T_2$. Furthermore, it follows from Equation (12.11) that

$$1 + KK_p = \frac{T_1}{T_2} = \frac{300}{50} = 6.$$

The loop gain is thus essentially determined by the ratio of the time constants. The controller gain becomes

$$K = \frac{5}{K_p},$$

and the closed-loop system has the undamped natural frequency.

$$\omega = 1/T_2 = 0.02 \text{ rad/s}.$$

If PI control is chosen instead, it is reasonable to choose a value of the integration time

$$T_1 \approx 5T_2.$$

Control can be improved by using derivative action. The achievable improvement depends on the time constant of the temperature sensor. In typical cases this time constant is between 10 s and 40 s. If it is as low as 10 s it is indeed possible to obtain improved control by introducing a derivative action in the controller. The derivative time can be chosen to eliminate the time constant T_2. We then obtain a system with the time constants 300 s and 10 s. The gain can then be increased so that

$$1 + KK_p = \frac{300}{10} = 30$$

and the undamped natural frequency of the system then becomes $\omega \approx 0.1$ rad/s. If the time constant of the temperature sensor is around 40 s, the derivative action gives only marginal improvements.

The heat generated by the chemical reaction is proportional to the flow q_A. To make sure that variations in q_A are compensated rapidly we have also introduced a feedforward from the flow q_A. This feedforward will only operate when the tank level is larger than 50 percent in order to avoid freezing when the reactor is started.

To start the reaction the reactor must be heated so that the temperature in the reaction vessel is larger than T_c (compare with Figure 12.33). This is done by using the steam valve V_2. Split-range control is used for the steam and water valves (compare Section 12.6). The water valve is open for low signals (3–9 PSI), and the steam valve is open for large pressures (9–15 PSI).

To avoid having the reactor freeze, it is necessary to make sure that the reaction temperature is always larger than T_c. This is the reason for the extra feedback from water temperature to T_v through a maximum selector. This feedback makes sure that the steam valve opens if the temperature in the coolant flow becomes too low. Cascade control would be an alternative to this arrangement.

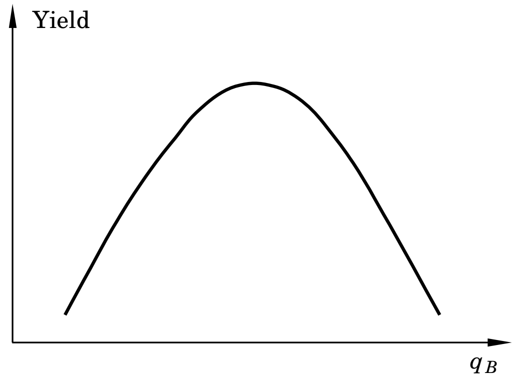

Figure 12.37 Reaction yield as a function of q_B at constant q_A.

Flow Ratio Control

The ratio of the flows q_A and q_B must be kept constant. Figure 12.37 shows how the efficiency of the reaction depends on q_B when q_A is kept constant. The flow q_B is controlled with a ratio control system (as shown in Figure 12.38), which is the primary control function. The reaction rate depends strongly on q_B. To diminish the risk of explosion, there is a nonlinearity in the feedback that increases the gain when q_B/q_A is large. The flow loop has several selectors. At startup it is desirable that substance B not be added until the water temperature has reached the critical value T_c and the reactor tank is half-full. To achieve this the feedback from water temperature and tank level has been introduced through limiters and a minimum selector. There are also limiters and a selector that closes valve V_4 if flow q_A is lost. There is also a direct feedback from q_A through limiters and selectors and a feedback from the reactor temperature that closes valve V_4, if the reactor temperature becomes too high.

Override Control of the Outlet Valve

The flow out of the reactor is determined by valve V_5. This valve is normally controlled by process steps downstream. The control of the reactor can be improved by introducing an override, which depends on the state of the reactor. When starting the reactor, it is desirable to have the outlet valve closed until the reactor tank is half-full and the reaction has started. This is achieved by introducing the tank level and the tank temperature to the set point of the valve controller via limiters and minimum selectors as is shown in Figure 12.39. The valve V_5 is normally controlled by q_{sp}. The minimum selector overrides the command q_{sp} when the level L or the temperature T_r are too low.

12.10 Summary

In this chapter we have illustrated how complex control systems can be built from simple components such as PID controllers, linear filters, gain schedules,

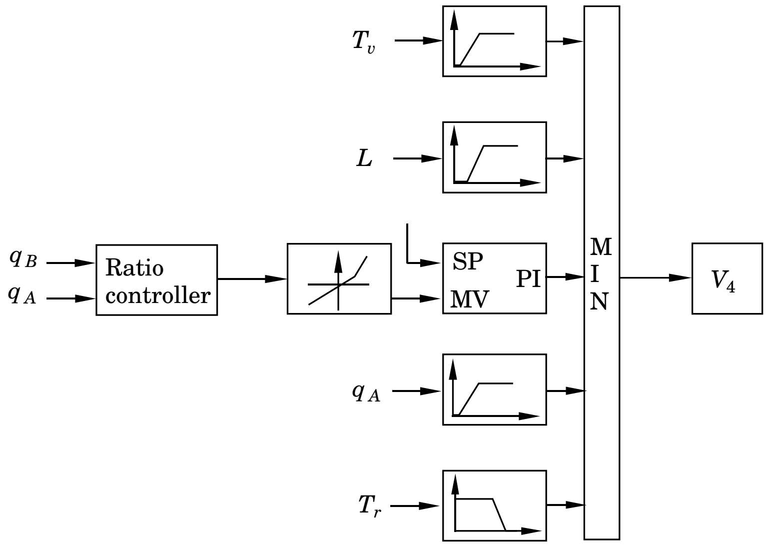

Figure 12.38 Block diagram for controlling the mixing ratio q_B/q_A through valve V_4.

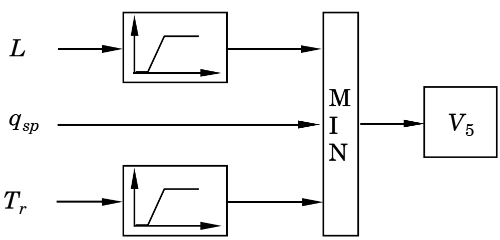

Figure 12.39 Block diagram for controlling the outflow of the reactor through valve V_5.

and simple nonlinear functions. A number of control paradigms have been introduced to guide system design.

The primary linear control paradigms are feedback by PID control and feedforward. Cascade control can be used to enhance control performance through the use of extra measurements. State feedback may be viewed as an extreme case of cascade control where all states of a system are measured. Observers can be used to infer values of variables that are not measured by combining mathematical models with available measurements. Mid-range and split-range control are paradigms for control when there are several control signals but only one measured signal. These paradigms are the dual of cascade control. Repetitive control is a technique that is efficient for cases where the disturbances are periodic. The idea is to create a high loop gain at the frequency of the disturbance.

We also discussed several nonlinear components and related paradigms including nonlinear functions, gain schedules, limiters, and selectors. Recall that it was shown in Section 3.5 how PID controllers could be enhanced by simple nonlinear functions to avoid windup. Ratio control is a nonlinear strategy that admits control of two process variables so that their ratio is constant. In Section 9.3 we showed how gain schedules could be used to cope with changes in process dynamics. Gain schedules and nonlinear functions are also useful for control of buffers, where the goal is not to keep constant levels in the buffers but to allow them to vary within given ranges. Selector control is another important paradigm that is used for constraint control where certain process variables have to be kept within given constraints. Neural and fuzzy techniques were also discussed briefly. It was shown that they could be interpreted both as rule-based control and as nonlinear control.

We also gave an example how the components and the paradigms could be used to develop a control system for a chemical process.

12.11 Notes and References

Many aspects of the material in this chapter are found in classical textbooks on process control such as [Buckley, 1964; Shinskey, 1988; Bequette, 2003; Seborg *et al.*, 2004] and in the books [Shinskey, 1981; Klefenz, 1986] which focus on energy systems. A more specialized presentation is given in [Hägglund, 1991].

The methods discussed in this chapter can all be characterized as bottom-up procedures in the sense that a complex system is built up by combining simple components. An interesting view of this is given in [Bristol, 1980]. A top-down approach is another possibility. A discussion of this, which is outside the scope of this book, is found in [Seborg *et al.*, 1986] and [Morari and Zafiriou, 1989].

Cascade and feedforward control are treated in the standard texts on control. A presentation with many practical aspects is found in [Tucker and Wills, 1960]. Selector control is widely used in practice. A general presentation is given in [Åström, 1987b]. It is difficult to analyse nonlinear systems. A stability analysis of a system with selectors is given in [Foss, 1981]. The Blend station is presented in [Hägglund, 2001].

Fuzzy control has been around for a long time; see [Mamdani, 1974; Mamdani and Assilian, 1974; King and Mamdani, 1977; Tong, 1977]. It has received a lot of attention particularly in Japan: see [Zadeh, 1988; Tong, 1984; Sugeno, 1985; Driankov *et al.*, 1993; Wang, 1994]. The technique has been used for automation of complicated processes that have previously been controlled manually. Control of cement kilns is a typical example; see [Holmblad and Østergaard, 1981]. There has been a similar development in neural networks; see, for example, [Hecht-Nielsen, 1990; Pao, 1990; Åström and McAvoy, 1992]. There was a lot of activity in neural networks during the late 1960s, which vanished rapidly. There was a rapid resurgence of interest in the 1980s. There are a lot of exaggerations both in fuzzy and neural techniques, and no balanced view of the relevance of the fields for control has yet emerged. The paper [Willis *et al.*, 1991] gives an overview of possible uses of neural networks for process control, and the paper [Pottman and Seborg, 1993] describes an application to control of pH. The papers [Lee, 1990; Huang, 1991; Swiniarski, 1991] describe applications to PID controllers and their tuning. There have also been attempts to merge fuzzy and neural control; see [Passino and Antsaklis, 1992] and [Brown and Harris, 1994]. Section 12.9 is based on [Buckley, 1970].