2
Process Dynamics Fundamentals

2.1
Signal and Process Dynamics

2.1.1
Measurement and Process Response

The aim of dynamic simulation is to be able to relate the dynamic output response of a system to the form of the input disturbance, in such a way that an improved knowledge and understanding of the dynamic characteristics of the system are obtained. Figure 2.1 depicts the relation of a process input disturbance to a process output response.

In testing process systems, standard input disturbances, such as the unit-step change, unit pulse, unit impulse, unit ramp, sinusoidal, and various randomised signals, can be employed.

All the above changes are easily implementable in dynamic simulations, using MADONNA and other digital simulation languages. The forms of response obtained differ in form, depending upon the system characteristics and can be demonstrated in the various MADONNA simulation examples. The response characteristics of real systems are, however, more complex. In order to be able to explain such phenomena, it is necessary to first examine the responses of simple systems, using the concept of the simple, step-change disturbance.

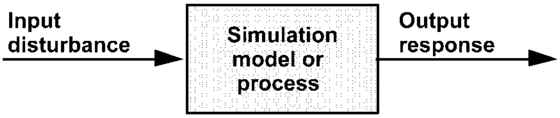

Fig. 2.1 Relation of process input to process output.

2.1.1.1 First-Order Response to an Input Step-Change Disturbance

The simplest response of a linear system is described mathematically by the following standard form of first-order differential equation

Chemical Engineering Dynamics: An Introduction to Modelling and Computer Simulation, Third Edition
J. Ingham, I. J. Dunn, E. Heinzle, J. E. Prenosil, J. B. Snape

ISBN: 978-3-527-31678-6

$$\tau\frac{dy}{dt}+y=y_0$$

$$\frac{dy}{dt}=\frac{y_0-y}{\tau}$$

where y is the measured or process response value, t is time, and τ is the equation time constant. In its second form, the equation is often described as a first-order lag equation, in that the response, y, lags behind the input value y_0, imposed on the system at time $t = 0$.

For the step-change condition, shown in Fig. 2.2, the initial conditions are given by $y = 0$, when $t = 0$. The solution to the differential equation, with the above boundary conditions, is given by

$$y=y_0(1-e^{-t/\tau})$$

and is shown in Fig. 2.2

Substituting the value $t = \tau$, gives

$$y_\tau=y_0(1-e^{-1})=0.632y_0$$

Hence the value of the equation time constant τ is simply determined as the time at which the response achieves sixty three per cent of its eventual steady-state value, when following a step change disturbance to the system.

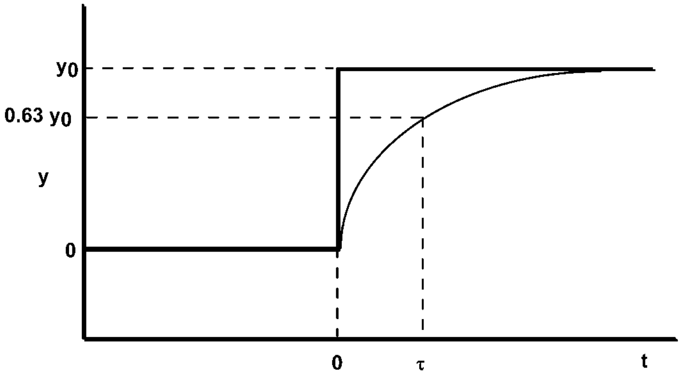

Fig. 2.2 First-order exponential response to an imposed step-change disturbance.

2.1.1.2 Case A: Concentration Response of a Continuous Flow, Stirred Tank

Liquid flows through a tank of constant volume V, with volumetric flow rate F and feed concentration C_0, as shown in Fig. 2.3.

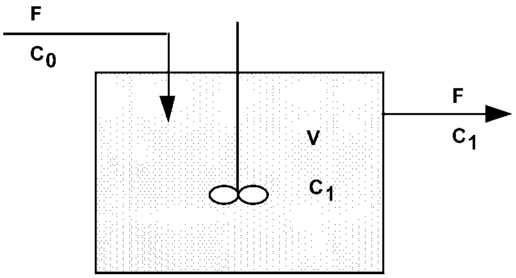

Fig. 2.3 A continuous stirred-tank reactor.

Assuming well-mixed conditions, the component balance equation is given by

$$V\frac{dC_1}{dt} = FC_0 - FC_1$$

This can be expressed as

$$\frac{dC_1}{dt} = \frac{C_0 - C_1}{\tau}$$

where τ is V/F.

Note that the equation has the general form

$$\left(\begin{matrix}\text{Rate of change}\\ \text{output variable}\end{matrix}\right) = \left(\frac{\text{Input variable} - \text{Output variable}}{\text{Process time constant}}\right)$$

Thus the time constant for the process is equal to the mean residence or hold-up time in the tank and has units of time (volume/volumetric flow rate).

Integrating with $C_1 = 0$ when $t = 0$ gives

$$C_1 = C_0(1 - e^{-t/\tau})$$

The response to a step change in feed concentration is thus given by

$$C_1 = C_0(1 - e^{-t/\tau}) = C_0(1 - e^{-Ft/V})$$

and follows the same form as that shown in Fig. 2.4.

Note that when time, t, tends to infinity, the value C_1 approaches C_0, and when time, t, is numerically equal to τ

$$C_1 = C_0(1 - e^{-1}) = 0.632C_0$$

Thus the response is 63.2% complete when the time passed is equal to the time constant τ. Further, for time t equal to four times the value of time constant τ,

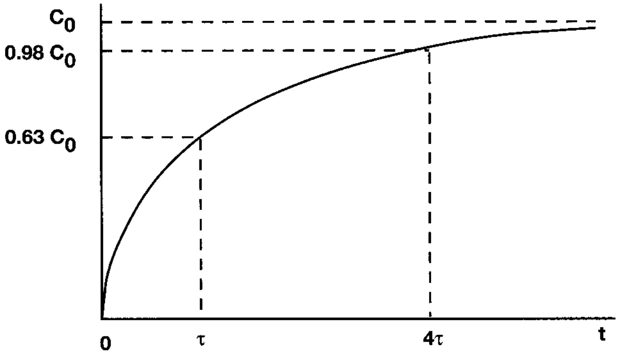

Fig. 2.4 Step response of a first-order system.

$$C_1 = C_0(1 - e^{-4}) = 0.98C_0$$

then the response is 98% complete.

2.1.1.3 Case B: Concentration Response in a Continuous Stirred Tank with Chemical Reaction

Assuming a chemical reaction in the tank, in which the rate of reaction is proportional to concentration, the component balance equation now becomes

$$V\frac{dC_1}{dt} = FC_0 - FC_1 - kC_1V$$

where k is the chemical rate coefficient (1/s). This can be rewritten as

$$\frac{V}{F + kV}\frac{dC_1}{dt} + C_1 = \frac{F}{F + kV}C_0$$

and the system time constant now has the value

$$\tau = \frac{V}{F + kV}$$

with high values of k acting to reduce the magnitude of τ.

The above equation now becomes

$$\frac{dC_1}{dt} = \frac{C_{1\infty} - C_1}{\tau}$$

where $C_{1\infty}$ is the final steady-state value at $t = \infty$, and is given by

$$C_{1\infty} = \frac{F\,C_0}{F + kV}$$

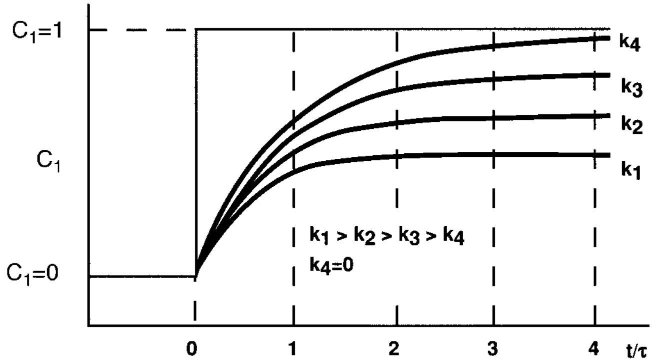

Fig. 2.5 Step response of a stirred tank with first-order chemical reaction ($V = 1$, $F = 1$, $k_4 = 0$, $k_3 = 0.2$, $k_2 = 0.5$, $k_1 = 1$).

Figure 2.5 illustrates the effect on the process response of increasing values of k. This shows that increasing the value of k will decrease the response time of the system and that the final effluent concentration leaving the tank will be reduced in magnitude. Increasing k has, however, very little influence on the initial rate of response.

Note that for $k = 0$, with zero reaction, the final steady-state response is given by C_0, and the response is identical to that of Case A.

2.1.1.4 Case C: Response of a Temperature Measuring Element

Instrument measurement response can often be important in the overall system response. The thermal response of a simple thermometer bulb, immersed in fluid, as shown in Fig. 2.6, is the result of a simple heat balance in which

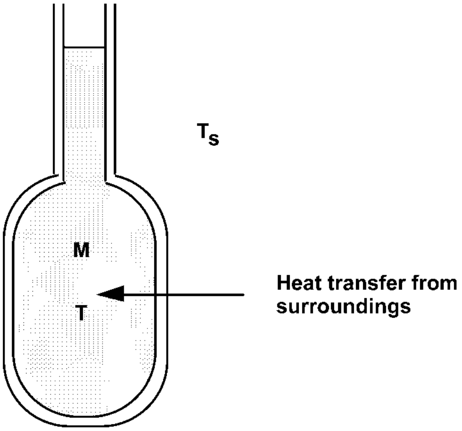

Fig. 2.6 Temperature measuring element.

$$\begin{pmatrix}\text{Rate of accumulation}\\ \text{of heat by the bulb}\end{pmatrix} = \begin{pmatrix}\text{Rate of heat transfer}\\ \text{to the bulb from the fluid}\end{pmatrix}$$

i.e.,

$$Mc_p\frac{dT}{dt} = UA(T_S - T)$$

where M is the mass of the thermometer bulb contents, c_p is the specific heat capacity of the bulb contents, U is the film heat transfer coefficient between the fluid and bulb wall, A is the heat transfer surface area and T_S is the temperature of the surrounding fluid. Note that the stirred-tank or lumped-parameter concept has again been adopted in the modelling approach and that the temperature of the fluid within the bulb is assumed to be uniform.

The simple balance equation can be reformulated as

$$\frac{M\,c_p}{U\,A}\frac{dT}{dt} + T = T_S$$

showing that the measurement time constant is

$$\tau = \frac{M\,c_p}{U\,A}$$

Often an instrument response measurement can be fitted empirically to a first-order lag model, especially if the pure instrument response to a step change disturbance has the general shape of a first-order exponential.

As shown in Section 2.1.1.1, the time constant for the instrument is then given as the time at which 63% of the final response is achieved and the instrument response may be described by the simple relationship

$$\frac{dT_{meas}}{dt} = \frac{T_{system} - T_{meas}}{\tau}$$

where T_{system}, the actual temperature, and T_{meas}, the measured temperature, are related by the measurement dynamics, as shown in Fig. 2.7, and τ is the experimentally obtained instrument time constant.

The ratio of the time constants, τ_{meas} and τ_{system}, determines whether or not the system value is significantly different from the measurement value when conditions are changing, since the measured value will tend to lag behind that of the system.

For a thermometer to react rapidly to changes in the surrounding temperature, the magnitude of the time constant should be small. This involves a high

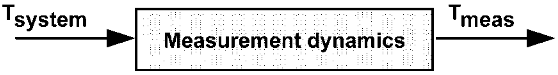

Fig. 2.7 Influence of instrument response on the measured temperature.

surface area to liquid mass ratio, a high heat transfer coefficient and a low specific heat capacity for the bulb liquid. With a large time constant, the instrument will respond slowly and may result in a dynamic measurement error.

2.1.1.5 **Case D: Measurement Lag for Concentration in a Batch Reactor**

The measurement lag for concentration in a reactor is depicted in Fig. 2.8. The actual reactant concentration in the reactor at any time t is given by C_r, but owing to the slow response of the measuring instrument, the measured concentration, shown by the instrument, C_m, lags behind C_r, as indicated in Fig. 2.9.

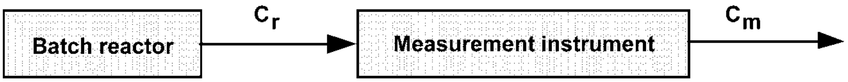

Fig. 2.8 Concentration measurement lag.

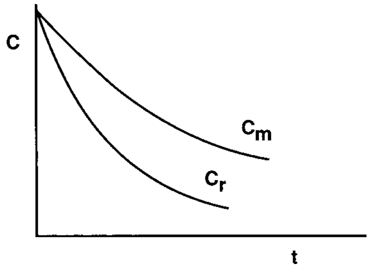

Fig. 2.9 Measured (C_m) and actual (C_r) concentration responses.

The dynamic error existing between C_m and C_r depends on the relative magnitudes of the respective time constants. For the reactor, assuming a first-order constant volume reaction

$$\frac{dC_r}{dt} = -k\,C_r$$

gives a time constant for the reaction

$$\tau_r = \frac{1}{k}$$

Assuming that the instrument response is first order, then as shown in Section 2.1.1.1, the instrument time constant τ_m is given by the value of time at the 63% point (response to a step-change disturbance), where

$$\frac{dC_m}{dt} = \frac{C_r - C_m}{\tau_m}$$

The ratio of the time constants, τ_r/τ_m, which for this case equals $(k\tau_m)$ will determine whether C_r is significantly different from C_m. When this ratio is less than 1.0 the measurement lag will be important. If $\tau_r/\tau_m > 10$, then $C_m \approx C_r$ and the measurement dynamics become unimportant.

The effects of measurement dynamics are demonstrated in the simulation examples KLADYN, TEMPCONT and CONTUN.

2.1.2
Higher Order Responses

Actual response curves often follow a sigmoidal curve as shown in Fig. 2.10. This is characteristic of systems having a series of multiple lags and hence of systems which are characterised by several time constants.

Examples of higher order response curves are shown by the following case studies.

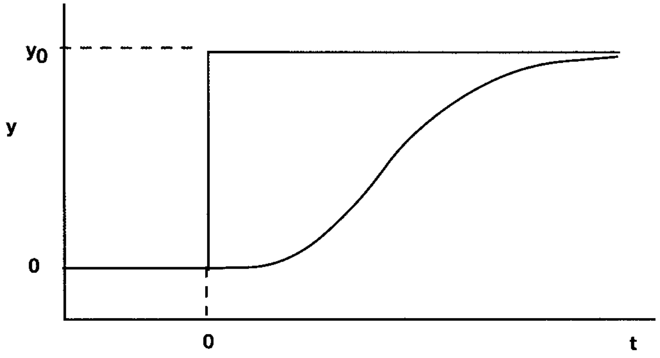

Fig. 2.10 Higher order step response.

2.1.2.1 Case A: Multiple Tanks in Series

Consider the case of three, constant-volume tanks in series, as represented in Fig. 2.11, in which the tanks have differing volumes V_1, V_2, V_3, respectively. Assuming well-mixed tanks, the component balance equations are

for tank 1

$$V_1 \frac{dC_1}{dt} = F\,C_0 - F\,C_1$$

for tank 2

$$V_2 \frac{dC_2}{dt} = F\,C_1 - F\,C_2$$

for tank 3

$$V_3 \frac{dC_3}{dt} = F\,C_2 - F\,C_3$$

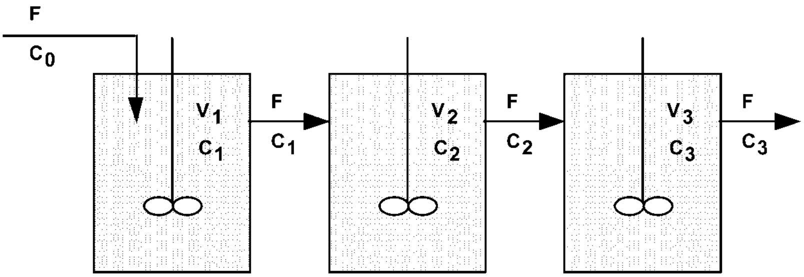

Fig. 2.11 Stirred tanks in series.

The above balance equations may be expressed as

$$\tau_1 \frac{dC_1}{dt} + C_1 = C_0 \quad \tau_1 = \frac{V_1}{F}$$

$$\tau_2 \frac{dC_2}{dt} + C_2 = C_1 \quad \tau_2 = \frac{V_2}{F}$$

and

$$\tau_3 \frac{dC_3}{dt} + C_3 = C_2 \quad \tau_3 = \frac{V_3}{F}$$

or as

$$\frac{dC_1}{dt} = \frac{C_0 - C_1}{\tau_1}$$

$$\frac{dC_2}{dt} = \frac{C_1 - C_2}{\tau_2}$$

$$\frac{dC_3}{dt} = \frac{C_2 - C_3}{\tau_3}$$

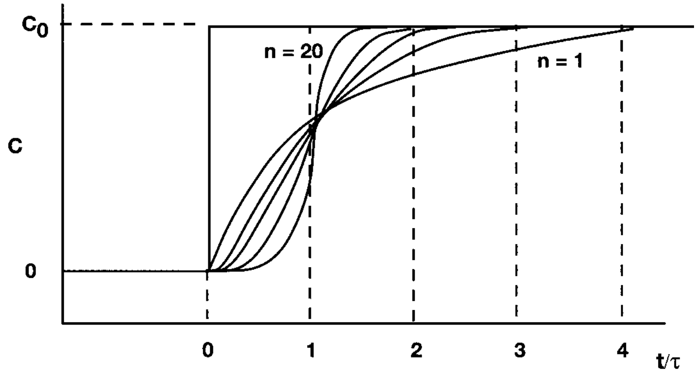

Fig. 2.12 Step response of n equal volume stirred tanks in series (n = 1, 2, 5, 10, 20).

In this case, three time constants in series, τ_1, τ_2 and τ_3, determine the form of the final outlet response C_3. As the number of tanks is increased, the response curve increasingly approximates the original step-change input signal, as shown in Fig. 2.12. The response curves for three stirred tanks in series, combined with chemical reaction are shown in the simulation example CSTRPULSE.

2.1.2.2 Case B: Response of a Second-Order Temperature Measuring Element

The temperature response of the measurement element shown in Fig. 2.13 is strictly determined by four time constants, describing (a) the response of the bulk liquid, (b) the response of the thermometer pocket, (c) the response of the heat conducting liquid between the wall of the bulb and the wall of the pocket and (d) the response of the wall material of the actual thermometer bulb. The time constants (c) and (d) are usually very small and can be neglected. A realistic model should, however, take into account the thermal capacity of the pocket, which can sometimes be significant.

Assuming the pocket to have a uniform temperature T_P, the heat balance for the bulb is now

$$\begin{pmatrix}\text{Rate of accumulation}\\ \text{of heat by the bulb}\end{pmatrix} = \begin{pmatrix}\text{Rate of heat transfer}\\ \text{to the bulb from the pocket}\end{pmatrix}$$

$$M\,c_p\frac{dT}{dt} = U_1\,A_1(T_P - T)$$

where U_1 is the heat transfer coefficient from the pocket to the bulb and A_1 is the heat transfer surface between the fluid and the bulb.

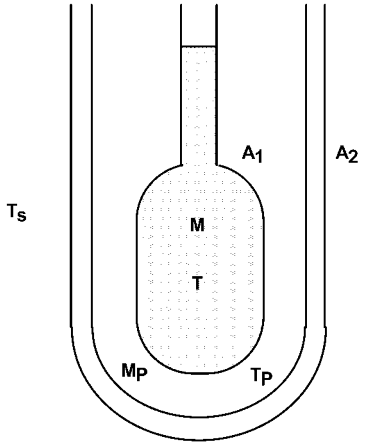

Fig. 2.13 Thermometer enclosed within a pocket.

Since the pocket temperature, T_P, is now a variable in the system, an additional heat balance equation is required for the pocket. This is of the same form as for the bulb, except that heat is now transferred both to the pocket from the surrounding and from the pocket to the bulb. Thus

$$\begin{pmatrix}\text{Rate of}\\ \text{accumulation}\\ \text{of heat}\\ \text{by the pocket}\end{pmatrix} = \begin{pmatrix}\text{Rate of heat transfer}\\ \text{to the pocket}\\ \text{from the fluid}\end{pmatrix} - \begin{pmatrix}\text{Rate of heat transfer}\\ \text{from the pocket}\\ \text{to the bulb}\end{pmatrix}$$

giving

$$M_P\, c_{pP} \frac{dT_P}{dt} = U_2\, A_2 (T_S - T_P) - U_1\, A_1 (T_P - T)$$

where U_2 is the heat transfer coefficient between the fluid and the pocket, A_2 is the heat transfer surface between the fluid and the pocket, M_P is the mass of the pocket, and c_{pP} is the specific heat of the pocket.

The overall instrument response is thus now determined by the relative magnitudes of the two major time constants, where for the liquid in the bulk

$$\tau_1 = \frac{M\, c_P}{U_1\, A_1}$$

and for the pocket,

$$\tau_2 = \frac{M_P\, c_{pP}}{U_1 A_1 + U_2 A_2}$$

For accurate dynamic measurement of process temperature, both τ_2 and τ_1 should be small compared with the time constant of the actual process.

2.1.3
Pure Time Delay

In contrast to the prior lag type response signals, time delays give no immediate response until the elapse of a given period of dead time or delay. In flow processes, this is equivalent to the time required for the system to pass through the signal in an otherwise unchanged state. An example would be the time taken to pump a sample from the process to a measuring instrument. In this case the magnitude of the time delay would simply be the time taken for the sample to travel along the pipe or the volume of the sample pipe divided by the sample flow rate, and thus equal to the mean residence time of the sample system.

As shown in Fig. 2.14, the input signal from the process is transmitted through the sample pipe until it arrives at the measuring instrument at a delay

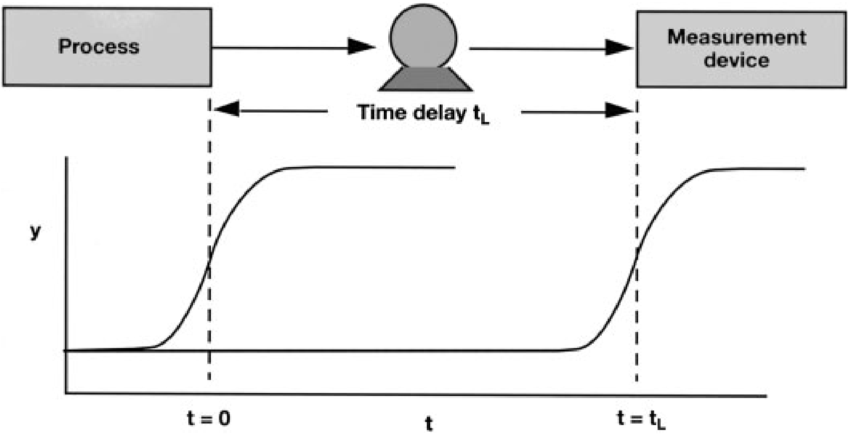

Fig. 2.14 Schematic drawing of a process with immediate and time-delayed responses to a step change of an input signal.

time t_L. In all other respects, however, the signal arriving at the measurement point is identical to the response of the actual system.

Most simulation languages include a standard time delay function, which is pre-programmed into the language structure. This facility is also available in MADONNA and is implemented in several of the simulation examples.

2.1.4
Transfer Function Representation

Complex systems can often be represented by linear time-dependent differential equations. These can conveniently be converted to algebraic form using Laplace transformation and have found use in the analysis of dynamic systems (e.g., Coughanowr and Koppel, 1965; Stephanopoulos, 1984; Luyben, 1990).

Thus, as shown in Fig. 2.15, the input-output transfer-function relationship, G(s), is algebraic, whereas the time domain is governed by the differential equation.

Model representations in Laplace transform form are mainly used in control theory. This approach is limited to linear differential equation systems or their

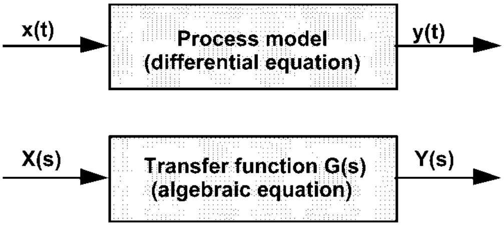

Fig. 2.15 Time and Laplace domain representations.

linearized approximations and is achieved by a combination of first order lag function and time delays. This limitation together with additional complications of modelling procedures are the main reasons for not using this method here. Specialized books in control theory as mentioned above use this approach and are available to the interested reader.

Dynamic problems expressed in transfer function form are often very easily reformulated back into sets of differential equation and associated time delay functions. An example of this is shown in the simulation example TRANSIM.

2.2
Time Constants

As shown in the preceding sections, the magnitude of various process time constants can be used to characterise the rate of response of a process resulting from an input disturbance. A fast process is characterised by a small value of the time constant and a slow process by large time constants. Time constants can therefore be used to compare rates of change and thus also to compare the relative importance of differing rate processes.

The term *time constant* is more or less equivalent to process time, characteristic time and relaxation time. Relaxation time is often used in physics, but is applied only to first-order processes and refers to the time for a process to reach a certain fraction of completion. This fraction is given by $(1 - 1/e) = 0.63$, which for a first-order process, as shown previously, is reached at a time $t = \tau$. Time constants also may be used to describe higher order processes and also non-linear processes. In these cases the time constant is defined as the time in which the process proceeds to a specified fraction of the resultant steady state. Higher order processes are often more elegantly described by a series of time constants.

A knowledge of the relative magnitude of the time constants involved in dynamic processes is often very useful in the analysis of a given problem, since this can be used to

- discover whether a change of regime occurs during scale up
- reduce the complexity of mathematical models
- determine whether the overall rate of a process is limited by a particular rate process, e.g., kinetic limitation or by diffusion, mixing, etc.
- check the controllability of a process
- check the difficulty of numerical solution due to equation stiffness

If the differing time constants for a chemical process are plotted as a function of the system variables, it can often be seen which rate process may be limiting. Many dimensionless groups can be considered as a ratio of the time constants for differing processes, and can give a clearer view on the physical meaning of the group. Such factors are discussed in much greater detail in other texts (Sweere et al., 1987), but here the intention is simply to draw attention to the importance of process time constants in the general field of dynamic simulation.

Table 2.1 Time constants defined by capacity and flow.

Capacity symbol	Dimension	Rate symbol	Dimension	Time constant
L	m	v	m/s	Travelling time τ
V	m^3	F	m^3/s	Residence time τ
V C	kmol	V k_La C	kmol/s	Mass transfer time τ_{mt}
V C	kmol	V r_C	kmol/s	Reaction time τ_r
V ρc_p dT	J	U A dT	J/s	Heat transfer time τ_{ht}
V ρc_p dT	J	V r_h	J/s	Heat production time τ_{hp}
V C	kmol	A D C/L	kmol/s	Diffusion time τ_d

As shown previously, the general form of equation serving to define time constants is as follows

$$\begin{pmatrix}\text{The rate of change}\\ \text{of the variable}\end{pmatrix} = \left(\frac{\text{Final value} - \text{Instantaneous value}}{\text{Time constant}}\right)$$

However, a more general way to define time constants is

$$\text{Time constant} = \frac{\text{Capacity}}{\text{Rate}}$$

In this definition both “capacity” and “rate” have to be used in a rather general way. Some examples are presented in Table 2.1. The symbols are defined in the Nomenclature.

The choice of capacity is sometimes a problem, and may change according to the particular circumstance. Sometimes using a definition of time constant, based on the above equations, is not very helpful and other means must be employed. For example, mixing time is a very important time constant relating to liquid mixing, and this is best obtained directly from empirical correlations of experimental data.

2.2.1
Common Time Constants

2.2.1.1 Flow Phenomena

Some common time constants, relating to particular chemical engineering flow applications, are

$$\tau = \frac{\text{Capacity}}{\text{Rate}} = \frac{\text{Length}}{\text{Velocity}} = \text{Travelling time}$$

$$\tau = \frac{\text{Capacity}}{\text{Rate}} = \frac{\text{Volume}}{\text{Volumetric flow}} = \text{Residence time}$$

$$\tau_{circ} = \text{Circulation time}$$

Various empirical equations are available for the circulation time constant, τ_{circ}, in stirred vessels, columns, etc. Usually the value of the time constant, however, will represent a mean value, owing to the stochastic nature of flow.

Mixing time constants, τ_{mix}, are also available based on an empirical correlation and are usually closely related to the value of τ_{circ} (Joshi et al., 1982). A value of $\tau_{\mathrm{mix}} = 4\tau_{\mathrm{circ}}$ is often used for stirred vessels and a value of $\tau_{\mathrm{mix}} = 2$ to $4\tau_{\mathrm{circ}}$ for columns. The exact value strongly depends on the degree of mixing obtained.

2.2.1.2 **Diffusion–Dispersion**

Diffusion and dispersion processes can be characterised by a time constant for the process, given by

$$\tau_{\mathrm{D}} = \frac{\mathrm{Capacity}}{\mathrm{Rate}} = \frac{L^2}{D}$$

where L is the characteristic diffusion or dispersion length and D is the diffusion or dispersion coefficient.

2.2.1.3 **Chemical Reaction**

Chemical reaction rate processes can be described by time constants.

In general

$$\tau_{\mathrm{r}} = \frac{\mathrm{Capacity}}{\mathrm{Rate}} = \frac{V\,C}{V\,r} = \frac{C}{r}$$

where C is concentration and r is the reaction rate. Hence

- for a zero-order process $r = k$ $\tau_{\mathrm{r}} = C/k$
- for a first-order process $r = kC$ $\tau_{\mathrm{r}} = 1/k$
- for a second-order process $r = kC^2$ $\tau_{\mathrm{r}} = 1/kC$

2.2.1.4 **Mass Transfer**

Transfer rate processes can also be characterized by time constants formulated as

$$\tau_{\mathrm{mt}} = \frac{\mathrm{Capacity}}{\mathrm{Rate}} = \frac{V\,C}{V\,Ka\,C} = \frac{1}{Ka}$$

where Ka is the mass transfer capacity coefficient, with units (1/s).

For a first-order process the time constant can be found from the defining differential equation as shown in Section 2.1.1.1. For the case of the aeration of a liquid, using a stirred tank, the following component balance equation applies

$$V_L \frac{dC_L}{dt} = k_L a (C_L^* - C_L) V_L$$

where C_L is the concentration of oxygen in the liquid phase (kg/m^3), t is time (s), $k_L a$ is the mass transfer coefficient (1/s), V_L is the liquid volume (m^3), and C_L^* is the equilibrium-dissolved oxygen concentration corresponding to the gas phase concentration, C_G.

By definition, the time constant for the process is thus $1/k_L a$ and the dissolved oxygen response to a step change in gas concentration is given by

$$C_L = C_L^* \left(1 - e^{-k_L a t}\right)$$

One has to be careful, however, in defining time constants. The first important step is to set up the correct equations appropriately. If the prime interest is not the accumulation of oxygen in the liquid as defined previously, but the depletion of oxygen from the gas bubbles, then the appropriate balance equation becomes

$$V_G \frac{dC_G}{dt} = -k_L a (C_L^* - C_L) V_L$$

Note that in this case, the gas phase concentration, C_G, relates to the total mixed gas phase volume V_G, whereas the mass transfer capacity coefficient term is more conveniently related to the liquid volume, V_L.

If we consider the case where $C_L \ll C_L^*$ and that C_L^* is related to C_G via the Henry's Law coefficient as $C_L^* = C_G/H$, we can rewrite the above equation as

$$\frac{dC_G}{dt} = \frac{-k_L a V_L}{H V_G} C_G$$

The time constant is now given by

$$\frac{H V_G}{k_L a V_L}$$

Thus for the accumulation of oxygen in the liquid phase

$$\tau_{mt} = \frac{1}{k_L a}$$

and for the depletion of oxygen from the gas phase

$$\tau_{mt} = \frac{H}{k_L a} \frac{V_L}{V_G}$$

thus representing a substantial order of difference in magnitude for the two time constants.

2.2.1.5 Heat Transfer

Heat transfer processs time constants are formulated as

$$\tau_{ht} = \frac{\text{Capacity}}{\text{Rate}} = \frac{M\, c_P\, T}{U\, A\, T} = \frac{M\, c_P}{U\, A}$$

U is the heat transfer coefficient, M the mass, c_p the heat capacity and A the heat transfer area. A knowledge and understanding of the appropriate time constants is important in interpreting many of the simulation examples.

2.2.2
Application of Time Constants

Figure 2.16 gives an example of a bubble column reactor with growing microorganisms which consume oxygen. Here the individual process time constants are plotted versus the operating variable, the superficial gas velocity. It can be seen that for high values of the superficial gas velocity (v_s) and low rates of oxygen consumption (r_{O2}), the time constants for mixing (τ_{mix}) and for the oxygen mass transfer rate (τ_{mt}) are lower in magnitude than the oxygen reaction rate time constant (τ_r). For higher reaction rates (e.g., $r_{O2} = 1 \cdot 10^{-} - 3\ \mathrm{kg/m^3 s}$) and reasonable values of v_s, it is impossible to obtain a value of τ_{mix} less than τ_r, and therefore mixing and mass transfer processes can become limiting at higher reaction rates.

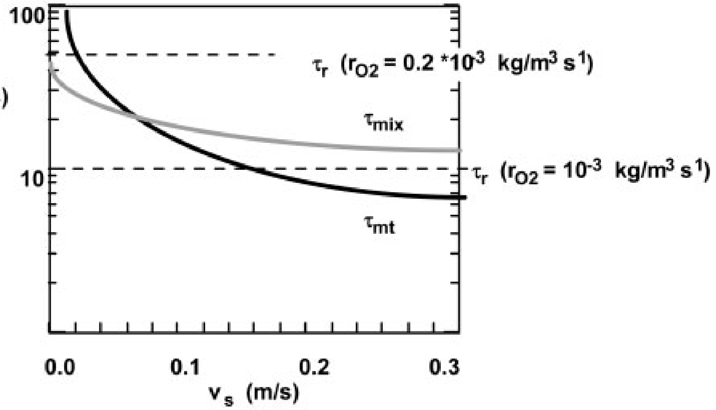

Fig. 2.16 Mixing, mass transfer and oxygen consumption in a bubble column bioreactor (Oosterhuis, 1984). τ_r reaction time constant, τ_{mt} mass transfer time constant, τ_{mix} mixing time constant. r_{O2} oxygen consumption rate, v_s superficial gas velocity.

2.3
Fundamentals of Automatic Control

Automatic process control involves the maintenance of a desired value of a measured or estimated quantity (controlled variable) within prescribed limits (deviations, errors), without the direct action of an operator. Generally, this involves three steps:

1. Measuring the present value of the controlled variable.
2. Comparing the measurement with the desired value (set point).
3. Adjusting some other variable (manipulated variable), which has influence on the controlled variable, until the set point is reached.

The most important reasons for applying process control are as follows:

- Safety for personnel and equipment.
- Uniform and high quality products.
- Increase of productivity.
- Minimisation of environmental hazards.
- Optimisation and decrease of labour costs.

Successful design of a process control system requires the following steps:

1. Selection of the control variables which are the most sensitive and easily measurable.
2. Formulation of the control objective; for example, the minimisation of some cost function.
3. Analysis of the process dynamics.
4. Selection of the optimal control strategy.

Process control is highly dynamic in nature, and its modelling leads usually to sets of differential equations which can be conveniently solved by digital simulation. A short introduction to the basic principles of process control, as employed in the simulation examples of Section 5.7, is presented.

2.3.1
Basic Feedback Control

The concept of an automatic control system is illustrated in Fig. 2.17, based on a temperature-controlled chemical reactor.

The components of the basic feedback control loop, combining the process and the controller, can be best understood using a generalised block diagram (Fig. 2.18). The information on the measured variable, temperature, taken from the system is used to manipulate the flow rate of the cooling water in order to keep the temperature at the desired constant value, or setpoint. This is illustrated by the simulation example TEMPCONT (Section 5.7.1).

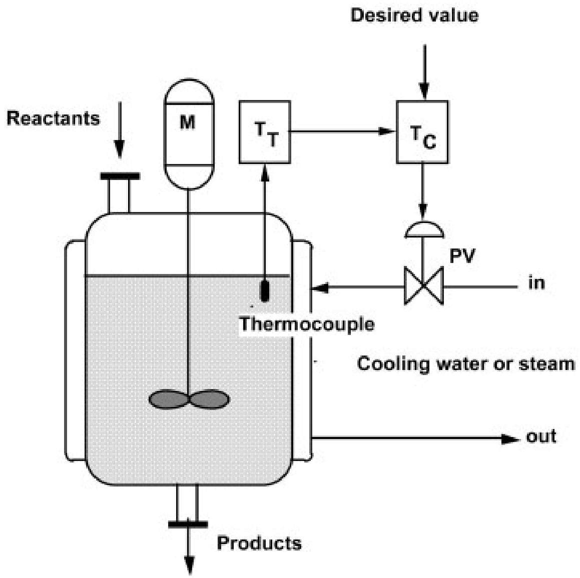

Fig. 2.17 Simple feedback temperature control system. M motor with stirrer, PV pneumatic valve, T_T temperature measurement, T_C temperature controller.

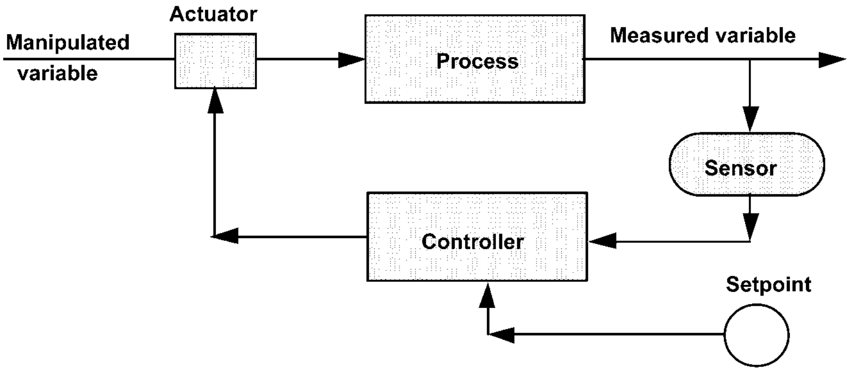

Fig. 2.18 Block diagram of a feedback control system.

2.3.2
Types of Controller Action

In the basic conventional feedback control strategy the value of the measured variable is compared with that for the desired value of that variable and if a difference exists, a controller output is generated to eliminate the error.

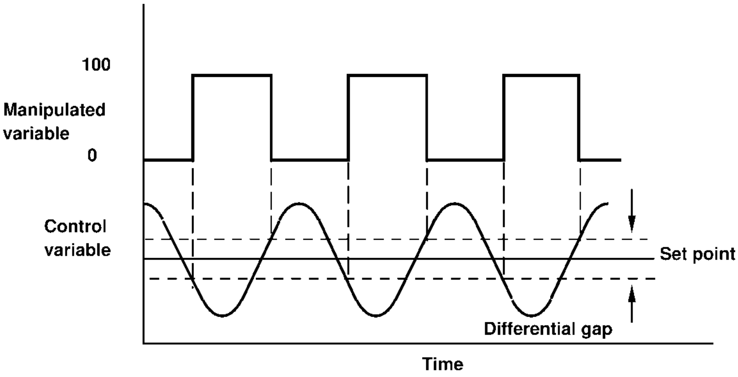

Fig. 2.19 On/off controller with differential gap or dead zone.

2.3.2.1 **On/Off Control**

The simplest and, despite its several drawbacks, the most widely used type of control is the on/off control system. An example is a contact thermometer, which closes or opens a heater circuit. The designation on/off means that the controller output, or the manipulated variable (electric current) is either fully on or completely off. To avoid oscillations around the setpoint, the real on/off controller has built into it a small interval on either side of the setpoint within which the controller does not respond, and which is called the differential gap or deadzone. When the controlled variable moves outside the deadzone, the manipulated variable is set either on or off. This is illustrated in Fig. 2.19. Such shifts from the set point are known as offset.

The oscillatory nature of the action and the offset make the resulting control rather imperfect, but the use of on/off control can be justified by its simplicity and low price, and the reasonable control obtained, especially for systems which respond slowly.

2.3.2.2 **Proportional-Integral-Derivative (PID) Control**

Three principal functional control modes are proportional (P), integral (I) and derivative (D) control. These are performed by the ideal three-mode controller (PID), described by the equation

$$P = P_0 + K_p\varepsilon(t) + \frac{K_p}{\tau_I}\int_0^t \varepsilon(t)dt + K_p\tau_D\frac{d\varepsilon(t)}{dt}$$

Controller modes: P I D

where:

P_0 is the controller output for zero error
K_p is the proportional gain

$\varepsilon(t)$ is the error or deviation of actual from desired value
τ_I is the integral time or reset time constant
τ_D is the derivative time constant

The response of a controller to an error depends on its mode. In the proportional mode (P), the output signal is proportional to the detected error, ε. Systems with proportional control often exhibit pronounced oscillations, and for sustained changes in load, the controlled variable attains a new equilibrium or steady-state position. The difference between this point and the set point is the offset. Proportional control always results in either an oscillatory behaviour or retains a constant offset error.

Integral mode controller (I) output is proportional to the sum of the error over the time. It can be seen that the corrections or adjustments are proportional to the integral of the error and not to the instantaneous value of the error. Moreover, the corrections continue until the error is brought to zero. However, the response of integral mode is slow and therefore is usually used in combination with other modes.

Derivative mode (D) output is proportional to the rate of change of the input error, as can be seen from the three-mode equation.

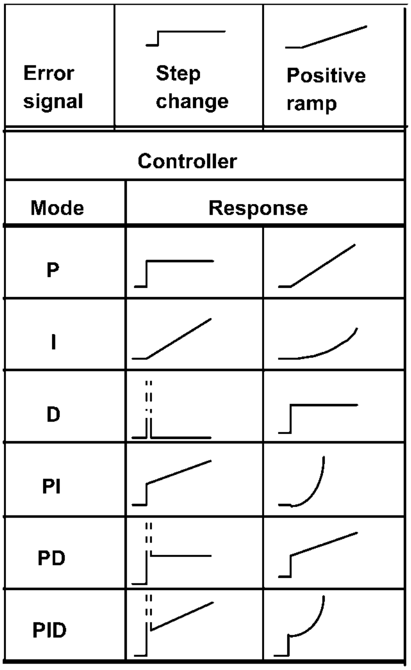

Fig. 2.20 Response of the most common controller modes for step change and ramp function of the error signal.

In industrial practice it is common to combine all three modes. The action is proportional to the error (P) and its change (D) and it continues if residual error is present (I). This combination gives the best control using conventional feedback equipment. It retains the specific advantages of all three modes: proportional correction (P), offset elimination (I) and stabilising, quick-acting character, especially suitable to overcome lag presence (D).

Simple control strategies form an integral part of many of the simulation examples, including RELUY, COOL, DEACT, REFRIG1, REFRIG2, RUN, EXTRACTCON, SULFONATION and the special control examples in Sec. 5.7, TEMPCONT, TWOTANK and CONTUN.

Figure 2.20 depicts the responses of the various control modes and their combinations to step and ramp inputs.

The performance of different feedback control modes can be seen in Fig. 2.21.

The selection of the best mode of control depends largely on the process characteristics. Further information can be found in the recommended texts listed in the reference section. Simulation methods are often used for testing control methods. The basic PI controller equations are easily programmed using a simulation language, as shown in the example programs. In the simulation examples, the general PID equation is simplified and only the P or P and I terms are used. Note that the I term can be set very low by using a high value for τ_I.

If desired, the differential term, $d\varepsilon/dt$, can be programmed as follows:

Since

$$\varepsilon = y - y_{set}$$

then

$$\frac{d\varepsilon}{dt} = \frac{dy}{dt}$$

and this derivative can be obtained directly from the model equations.

2.3.2.3 Case A: Operation of a Proportional Temperature Controller

Liquid flows continuously through a tank of volume, V, provided with an electric heater. A controller regulates the rate of heating directly in accordance with the difference between a required set point temperature, T_{set}, and the actual temperature, T_1, as shown in Fig. 2.22.

The heat balance equation for the tank is similar to that of Case A of Section 1.2.5.1, i.e.,

$$V\rho c_p \frac{dT}{dt} = F_0\rho c_p(T_0 - T_1) + Q$$

but where Q is the rate of electric heating and is expressed by a proportional control equation (omitting the I and D terms)

$$Q = Q_0 + K_p(T_{set} - T_1)$$

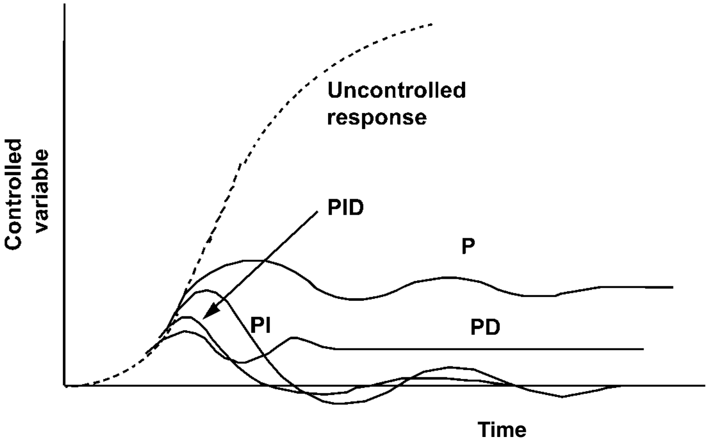

Fig. 2.21 Response of controlled variable to a step change in error using different control modes.

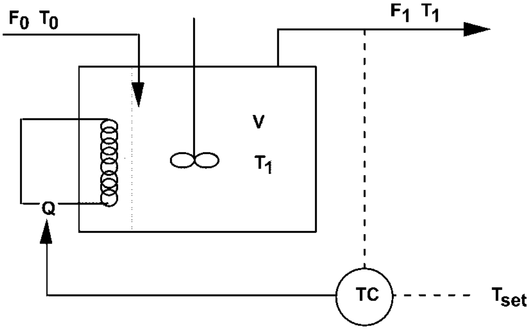

Fig. 2.22 Temperature control, TC, of a continuously operated stirred tank with an electric heater (Q).

2.3.3
Controller Tuning

The purpose of controller tuning is to choose the correct controller constants to obtain the desired performance characteristics. This usually means that the control variables should be restored in an optimal way to acceptable values, following either a change in the set point or the appearance of an input disturbance. Numerous books discussing the subject are available (e.g. Shinskey, 1996). The Internet is a good source of information, where some e-Books appeared, e.g. "The PID Controller Tuning Methods" by John Shaw. Simulation examples TEMPCONT and CONTUN provide exercises for controller tuning using the methods explained below.

2.3.3.1 Trial and Error Method

Controllers can be adjusted by changing the values of gain K_p, reset time τ_I and derivative time τ_D. The controller can be set by trial and error by experimenting, either on the real system or by simulation. Each time a disturbance is made the response is noted. The following procedure may be used to test the control with small set point or load changes:

1. Starting with a small value, K_p can be increased until the response is unstable and oscillatory. This value is called the ultimate gain K_{p0}.
2. K_p is then reduced by about one-half.
3. Integral action is brought in with high τ_I values. These are reduced by factors of 2 until the response is oscillatory, and τ_I is set at 2 times this value.
4. Include derivative action and increase τ_D until noise develops. Set τ_D at 1/2 this value.
5. Increase K_p in small steps to achieve the best results.

2.3.3.2 Ziegler–Nichols Open-Loop Method

This empirical open-loop tuning mode, known also as the “Reaction Curve” method, is implemented by uncoupling the controller. It is an empirical open-loop tuning technique, obtained by uncoupling the controller. It is based on the characteristic curve of the process response to a step change in manipulated variable of magnitude A. The response, of magnitude B, is called the process reaction curve. The two parameters important for this method are given by the slope through the inflection point normalised by A, so that S = Slope/A, and by its intersection with the time axis (lag time T_L), as determined graphically in

Table 2.2 Controller settings based on process responses.

Controller	K_p	τ_I	τ_D
Ziegler–Nichols			
P	$1/(T_L S)$		
PI	$0.9/(T_L S)$	$3.33\ T_L$	
PID	$1.2/(T_L S)$	$2\ T_L$	$T_L/2$
Cohen–Coon			
P	$\frac{\tau}{K\,T_L}\left(1+\frac{T_L}{3\tau}\right)$		
PI	$\frac{\tau}{K\,T_L}\left(0.9+\frac{T_L}{12\tau}\right)$	$T_L\frac{30+3T_L/\tau}{9+20T_L/\tau}$	
PID	$\frac{\tau}{K\,T_L}\left(\frac{4}{3}+\frac{T_L}{12\tau}\right)$	$T_L\frac{32+6T_L/\tau}{13+8T_L/\tau}$	$T_L\frac{4}{12+2T_L/\tau}$
Ultimate Gain			
P	$0.5\ K_{p0}$		
PI	$0.45\ K_{p0}$	$1/1.2\ f_0$	
PID	$0.6\ K_{p0}$	$1/2\ f_0$	$1/8\ f_0$

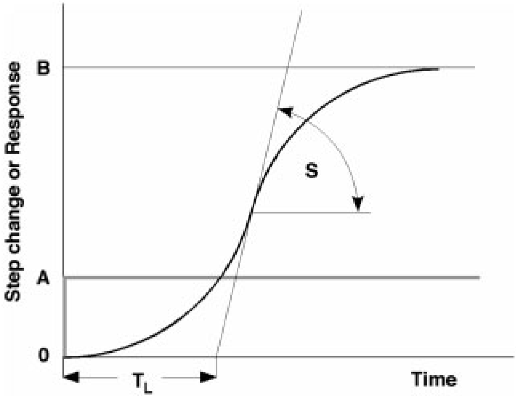

Fig. 2.23 Process reaction curve for the Ziegler–Nichols method.

Fig. 2.23. The actual tuning relations, based on empirical criteria for the "best" closed-loop response are given in Table 2.2.

2.3.3.3 **Cohen–Coon Controller Settings**

Cohen and Coon observed that the response of most uncontrolled (controller disconnected) processes to a step change in the manipulated variable is a sigmoidally shaped curve. This can be modelled approximately by a first-order system with time lag T_L, as given by the intersection of the tangent through the inflection point with the time axis (Fig. 2.23). The theoretical values of the controller settings obtained by the analysis of this system (e.g. Luyben and Luyben, 1997) are summarised in Table 2.2. The model parameters for a step change A to be used with this table are calculated as follows:

$$K = B/A \qquad \tau = B/S$$

where B is the extent of response, S is the slope at the inflection point, and T_L is the lag time as determined in Fig. 2.23.

2.3.3.4 **Ultimate Gain Method**

The previous transient-response tuning methods are sensitive to disturbances because they rely on open-loop experiments. Several closed loop methods have been developed to eliminate this drawback. One of these is the empirical tuning method, ultimate gain or continuous cycling method. The ultimate gain, K_{p0}, is the gain which brings the system with sole proportional control only to sustained oscillation (stability limits) with frequency f_{p0}, where $1/f_{p0}$ is called the ultimate period. This is determined experimentally by increasing K_p from low values in small increments until continuous cycling begins. The controller settings are then calculated from K_{p0} and f_{p0} according the tuning rules given in Table 2.2.

While this method is very simple it can be quite time consuming in terms of number of trials required and especially when the process dynamics are slow. In addition, it may be hazardous to experimentally force the system into unstable operation.

2.3.3.5 **Time Integral Criteria**

Several criteria may be used to estimate the quality of control (Stephanopoulos, 1984). One of these is the integral of the time-weighted absolute error (ITAE), where

$$\mathrm{ITAE} = \int_0^t |t\,\varepsilon(t)|(dt)$$

Integral error criteria are ideally suited to simulation applications since only one additional program statement is required for the simulation. The optimal control parameters K_p, τ_I and τ_D can be then found at minimal ITAE. For this, it is useful to be able to apply the available optimisation tools implemented in such programs as MATLAB, ACSL-OPTIMIZE or MADONNA.

2.3.4

Advanced Control Strategies

2.3.4.1 **Cascade Control**

In control situations with more than one measured variable but only one manipulated variable, it is advantageous to use control loops for each measured

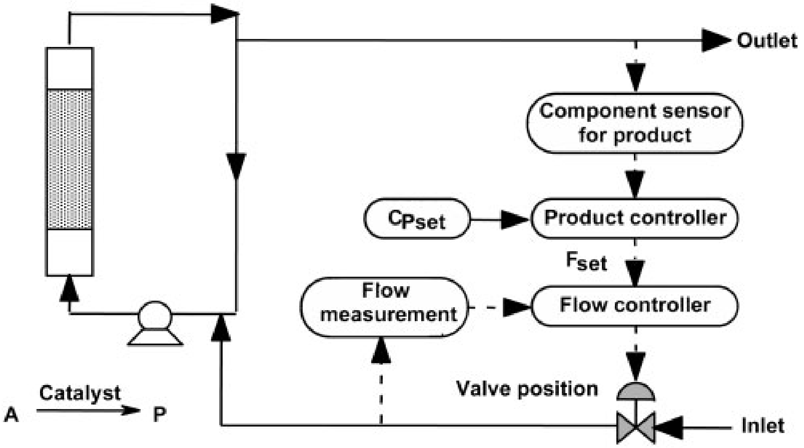

Fig. 2.24 Cascade control to maintain product concentration by manipulating the reactant concentration in the feed.

variable in a master-slave relationship. In this, the output of the primary controller is usually used as a set point for the slave or secondary loop.

An example of cascade control could be based on the simulation example DEACT and this is shown in Fig. 2.24. The problem involves a loop reactor with a deactivating catalyst, and a control strategy is needed to keep the product concentration C_P constant. This could be done by manipulating the feed rate into the system to control the product concentration at a desired level, C_{set}. In this cascade control, the first controller establishes the setpoint for flow rate. The second controller uses a measurement of flow rate to establish the valve position. This control procedure would then counteract the influence of decreasing catalyst activity.

2.3.4.2 **Feedforward Control**

Feedback control can never be perfect as it reacts only to disturbances in the process outlet. Feedforward control can theoretically be perfect, because the inlet disturbances are measured, and their effects on the process are anticipated via the use of a model. If the model is perfect then the calculated action to be taken will be exact.

The example simulation THERMFF illustrates this method of using a dynamic process model to develop a feedforward control strategy. At the desired setpoint the process will be at steady-state. Therefore the steady-state form of the model is used to make the feedforward calculations. This example involves a continuous tank reactor with exothermic reaction and jacket cooling. It is assumed here that variations of inlet concentration and inlet temperature will disturb the reactor operation. As shown in the example description, the steady state material balance is used to calculate the required response of flowrate and the steady state energy balance is used to calculate the required variation in jacket temperature. This feedforward strategy results in perfect control of the simulated process, but limitations required on the jacket temperature lead to imperfections in the control.

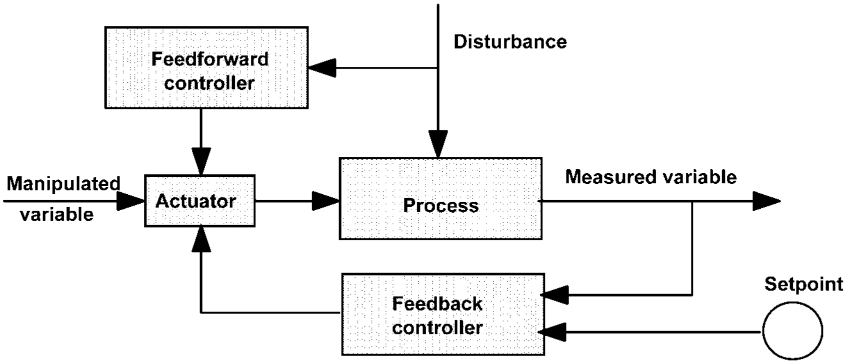

Fig. 2.25 Feed-forward control with additional feedback loop.

The success of this control strategy depends largely on the accuracy of the model prediction, which is often imperfect as models can rarely exactly predict the effects of process disturbances. For this reason, an additional feedback loop is often used as a backup or to trim the main feedforward action, as shown in Fig. 2.25. Many of the continuous process simulation examples in this book may be altered to simulate feedforward control situations.

2.3.4.3 Adaptive Control

An adaptive control system can automatically modify its behaviour according to the changes in the system dynamics and disturbances. They are applied especially to systems with non-linear and unsteady characteristics. There are a number of actual adaptive control systems. Programmed or scheduled adaptive control uses an auxiliary measured variable to identify different process phases for which the control parameters can be either programmed or scheduled. The "best" values of these parameters for each process state must be known *a priori*. Sometimes adaptive controllers are used to optimise two or more process outputs, by measuring the outputs and fitting the data with empirical functions.

2.3.4.4 Sampled Data or Discrete Control Systems

When discontinuous measurements are involved, the control system is referred to as a sampled data or discrete controller. Concentration measurements by chromatography would represent such a case.

Here a special consideration must be given to the sampling interval ΔT (Fig. 2.26). In general, the sampling time will be short enough if the sampling frequency is 2 times the highest frequency of interest or if ΔT is 0.5 times the minimum period of oscillation. If the sampling time satisfies this criterion, the system will behave as if it were continuous. Details of this and other advanced topics are given in specialised process control textbooks, some of which are listed in the references.

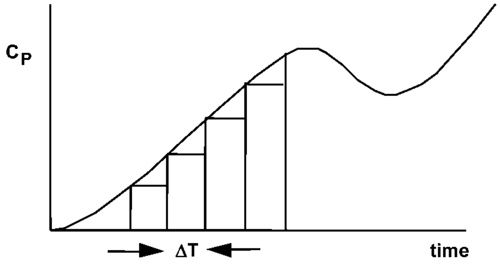

Fig. 2.26 Sampled control strategy.

2.4
Numerical Aspects of Dynamic Behaviour

2.4.1
Optimisation

Optimisation may be used, for example, to minimise the cost of reactor operation or to maximise conversion. Having set up a mathematical model of a reactor system, it is only necessary to define a cost or profit function and then to minimise or maximise this by variation of the operational parameters, such as temperature, feed flow rate or coolant flow rate. The extremum can then be found either manually by trial and error or by the use of numerical optimisation algorithms. The first method is easily applied with MADONNA, or with any other simulation software, if only one operational parameter is allowed to vary at any one time. If two or more parameters are to be optimised this method becomes extremely cumbersome. To handle such problems, MADONNA has a built-in optimisation algorithm for the minimisation of a user-defined objective function. This can be activated by the OPTIMIZE command from the Parameter menu. In MADONNA the use of parametric plots for a single variable optimisation is easy and straight-forward. It often suffices to identify optimal conditions, as shown in Case A below.

Basically two search procedures for non-linear parameter estimation applications apply (Nash and Walker-Smith, 1987). The first of these is derived from Newton's gradient method and numerous improvements on this method have been developed. The second method uses direct search techniques, one of which, the Nelder-Mead search algorithm, is derived from a simplex-like approach. Many of these methods are part of important mathematical packages, e.g., ASCL and MATLAB.

2.4.1.1 Case A: Optimal Cooling for a Reactor with an Exothermic Reversible Reaction

A reversible exothermic reaction $A \Leftrightarrow B$ is carried out in a stirred-tank reactor with cooling. The details of the model are given in the simulation example REVTEMP. Specific heats are functions of temperature. The temperature dependency of the reaction rate constants are given by the Arrhenius equation, that of the equilibrium constant by the van't Hoff equation. Adiabatic operation restricts conversion because of an unfavourable equilibrium at high temperature. Early cooling favours the equilibrium conversion but reduces reaction rates, according to the Arrhenius equation. It is assumed that the cooling water temperature is constant, and that the cooling water flow, FC, may be either on or off. At time TIMEON the cooling water flow is set to FCON. A profit function is defined as

$$\mathrm{SPTYB} = \frac{\mathrm{C}_\mathrm{B}^2}{\mathrm{t}}$$

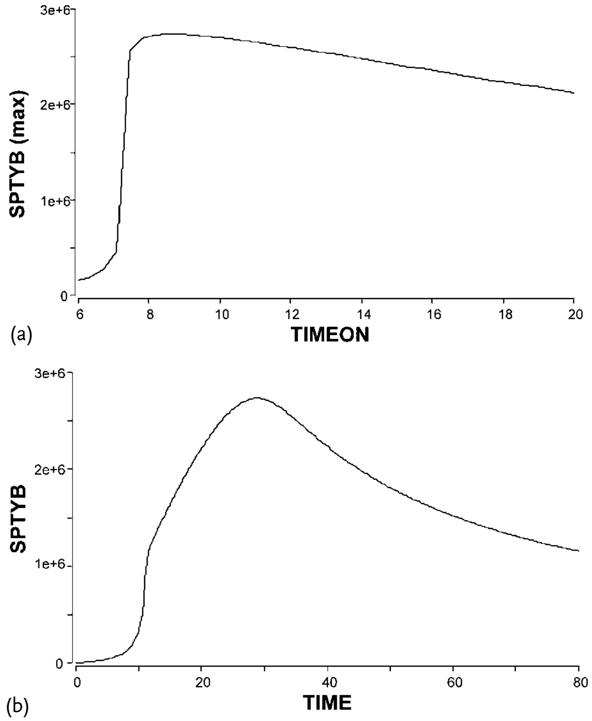

Fig. 2.27 (a) Parametric run of REVTEMP varying variable TIMEON. (b) Run of REVTEMP with optimal value of TIMEON from (a).

This reflects the desire to have high conversion in a short time period. SPTYB always passes through a maximum during a batch run. The problem is defined as finding the optimal times to switch on the cooling water flow (TIMEON) and to harvest the tank contents (TFIN). The program listing is given on the CD.

In MADONNA this problem is easily solved in the following way. Plotting SPTYB versus TIME always gives a maximum. This would be the optimal harvesting time for a preset value of TIMEON. The optimal value for TIMEON is found by **Batch Runs**, **Initial value** = 6, **Final Value** = 20 and e.g. 40 Runs. Selecting the Mode **Parametric Plot**, choosing SPTYB as variable and selecting **Maximum value**, gives a plot as depicted in Fig. 2.27a. Increasing TIMEON from 6 to 8 increases SPTMB (max.) dramatically showing a flat maximum between 8 and 10. Inspection of the tabular output gives an optimal value of about 8.5. From making one **Run** with the optimal value of TIMEON by setting TIMEON = 8.5 in the **Parameters Window** and plotting SPTYB versus TIME the maximum value of SPTYB is directly obtained (Fig. 2.27b). From the shape of the curves it is clearly seen that the system is very robust for values of TIMEON > 8.

In non-linear systems one can usually not predict a priori whether the optimum found is global or whether the optimum obtained represents only a local condition. A good judgement on the behaviour of the model can be seen in contour and three-dimensional plots, which are easily obtained using other alternative software packages, such as ACSL-OPTIMIZE or MATLAB.

2.4.2
Parameter Estimation

Having set up a model to describe the dynamics of the system, a very important first step is to compare the numerical solution of the model with any experimental results or observations. In the first stages, this comparison might be simply a check on the qualitative behaviour of a reactor model as compared to experiment. Such questions might be answered as: Does the model confirm the experimentally found observations that product selectivity increases with temperature and that increasing flow rate decreases the reaction conversion?

Following the first preliminary comparison, a next step could be to find a set of parameters that give the best or optimal fit to the experimental data. This can be done by a manual trial-and-error procedure or by using a more sophisticated mathematical technique which is aimed at finding those values for the system parameters that minimise the difference between values given by the model and those obtained by experiment. Such techniques are general, but are illustrated here with special reference to the dynamic behaviour of chemical reactors.

Table 2.3 is used to classify the differing systems of equations, encountered in chemical reactor applications and the normal method of parameter identification. As shown, the optimal values of the system parameters can be estimated using a suitable error criterion, such as the methods of least squares, maximum likelihood or probability density function.

Table 2.3 Classification of systems of reactor equations with a set of parameters and time-dependent variables.

	Examples of linear systems	Examples of non-linear systems
Algebraic equations	Steady state of CSTR with first-order kinetics. Algebraic solution and optimisation (least squares, Draper and Smith, 1981).	Steady state of CSTR with complex kinetics. Numerical solution and optimisation (least squares or likelihood function).
Differential equations	Batch reactor with first-order kinetics. Analytical or numerical solution with analytical or numerical parameter optimisation (least squares or likelihood).	Batch reactor with complex kinetics. Numerical integration and parameter optimisation (least squares or likelihood).

2.4.2.1 **Non-Linear Systems Parameter Estimation**

The methods concerned with differential equation parameter estimation are, of course, the ones of most concern in this book. Generally reactor models are non-linear in their parameters, and therefore we are concerned mostly with non-linear systems.

Given a model in the form of a set of differential equations,

$$\frac{dy}{dt} = f(k_1 \ldots k_n, y)$$

A model described by this differential equation is linear in the parameters $k_1 \ldots k_n$, if

$$\frac{\partial f}{\partial k_i} \neq g(k_1 \ldots k_n)$$

but is non-linear, if for at least one of the parameters k_i

$$\frac{\partial f}{dk_i} = h(k_1 \ldots, y)$$

The application of optimisation techniques for parameter estimation requires a useful statistical criterion (e.g., least-squares). A very important criterion in non-linear parameter estimation is the likelihood or probability density function. This can be combined with an error model which allows the errors to be a function of the measured value.

If basic assumptions concerning the error structure are incorrect (e.g., non-Gaussian distribution) or cannot be specified, more robust estimation techniques may be necessary, e.g., Maria and Heinzle (1998). In addition to the above considerations, it is often important to introduce constraints on the estimated parameters (e.g., the parameters can only be positive). Such constraints are included in the simulation and parameter estimation package ACSL-OPTIMIZE and in the MATLAB Optimisation Toolbox. Because of numerical inaccuracy, scaling of parameters and data may be necessary if the numerical values are of greatly differing order. Plots of the residuals, difference between model and measurement value, are very useful in identifying systematic or model errors.

Non-linear parameter estimation is far from a trivial task, even though it is greatly simplified by the availability of user-friendly program packages such as (a) ACSL-OPTIMIZE, (b) MADONNA, (c) a set of BASIC programs (supplied with the book of Nash and Walker-Smith, 1987) or (d) by mathematical software (MATLAB). MADONNA has only limited possibilities for parameter estimation, but MADONNA programs can easily be translated into other more powerful languages.

2.4.2.2 Case B: Estimation of Rate and Equilibrium Constants in a Reversible Esterification Reaction Using MADONNA

The objective is to demonstrate the use of MADONNA in the estimation of model parameters. Here the parameters are estimated using the CURVE FIT feature of MADONNA. This allows data to be imported by clicking the IMPORT DATA in this menu and selecting an external text file. The file must have the time in the first column and the data values in the second column. Two data columns can also be used, but they must correspond to equally spaced times. The number of parameters to be estimated can be one or more.

In this example, ethanol and acetic acid react reversibly to ethyl acetate, using a catalyst, ethyl hydrogensulfate, which is prepared by reaction between sulfuric acid and ethanol.

$$\text{Acetic acid} + \text{Ethanol} \rightleftharpoons \text{Ethyl acetate} + \text{water}$$

$$A + B \underset{k_2}{\overset{k_1}{\rightleftharpoons}} C + D$$

The rate of batch reaction for reactant A (acetic acid) is modelled as

$$\frac{dC_A}{dt} = r_A = -k_1 C_A C_B + k_2 C_C C_D$$

The progress of the reaction is followed by taking samples at regular time intervals and titrating the remaining free acid with alkali (mL).

The table of measured data, time (min) versus titrated volume (mL) is imported into the program from the external file ESTERdat.txt and will be plotted after a graph window is defined. Clicking CURVE-FIT will allow the selection of the variable and the parameters. For each parameter, two preliminary guessed values and the maximum and minimum allowable values can be entered. On running under CURVE-FIT, the values of the required parameters are repeatedly updated, until the final converged values are obtained. The updated values can be found in the Parameter Window. On clicking Run, a final run is made, enabling the final simulation results to be compared with the original data values. The MADONNA program ESTERFIT and the graph of the data and the final results of the parameter estimation are shown below.

```
{ESTERFIT}
{Fitting of experimental titration data to determine the rate
constants for a reversible reaction:
ethanol + acetic Acid ↔ ethyl acetate +water}

METHOD Auto
STARTTIME = 0
STOPTIME = 1200
DT = 0.3
```

```
{Initial guess on rate constants, m3/kmol min}
k1 = 0.003
k2 = 0.001

INIT CA = 5.636   {initial acetic acid, kmol/m3}
INIT CB = 5.636   {initial ethanol, kmol/m3}
INIT CC = 0       {initial ethyl acetate, kmol/m3}
INIT CD = 0.634   {initial water, kmol/m3}
CA0 = 5.636       {initial acetic acid, kmol/m3}

{Batch material balances}
d/dt(CA)=-r1+r2
d/dt(CB)=-r1+r2
d/dt(CC)=+r1-r2
d/dt(CD)=+r1-r2

r1=k1*CA*CB
r2=k2*CC*CD

MLtitrated=CA/Titfact
Titfact=CA0/ML0 {Ratio (kmoles/m3)/mL 1 N NaOH}
ML0=16.25       {mL titrated for calibration}

File of data (ESTERdat) giving time (min.) versus mL titrated
0.0     16.25
2.0     15.1
4.0     14.25
6.0     13.8
8.0     12.9
10.0    11.7
15.0    12.1
20.0    11.05
25.0    10.6
30.0    10.45
35.0    10.2
40.0    9.5
45.0    9.5
55.0    8.9
70.0    8.5
103.0   7.5
143.0   7.1
1000.0  6.5
1043.0  6.5
1100.0  6.5
```

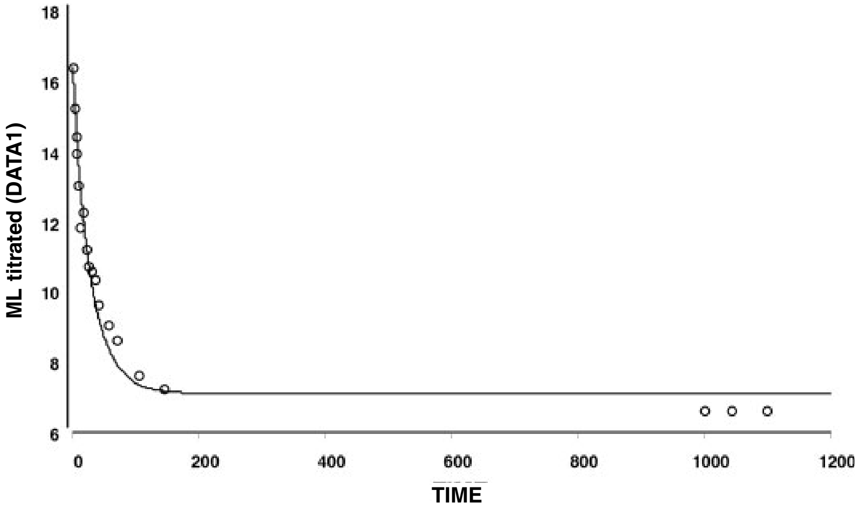

Fig. 2.28 Experimental data with fitted profile of mL titrated. Values found: $k_1 = 0.00397$ and $k_2 = 0.00185$

From Fig. 2.28 it is obvious that a reasonable fit is easily obtained. A detailed analysis of the results, however, discloses that there seems to be a systematic deviation in the residuals with high predicted values at the equilibrium conditions (TIME > 1000) and a low prediction between TIME around 50 and 150. These differences can be caused by an inadequacy in the model or by systematic experimental errors. A more appropriate objective function may also be desirable.

2.4.3
Sensitivity Analysis

The sensitivity of a model or a real system can be determined by making changes in the parameters of interest and noting their influences on each variable. The simplest measure of sensitivity is the derivative of the variable with respect to the change in the parameter, $\partial V/\partial P$. MADONNA has an automatic means for making "Sensitivity Runs". This is done by making two runs, one at the normal value of the parameter and another run using a value which is increased by 0.1%. The sensitivity is calculated using the difference in the variable, ΔV, divided by the difference in the parameter, ΔP, to give $\partial V/\partial P = \Delta V/\Delta P$. These derivative values are plotted for all the selected variables and parameters versus time. Obviously during any simulation, or during a real experiment, the sensitivity of the process to a particular parameter will vary as the conditions vary with time. Such a sensitivity analysis provides guidance as to how accurately a parameter needs to be determined. If the process is not sensitive to a model parameter then sometimes this enables the model to be simpli-

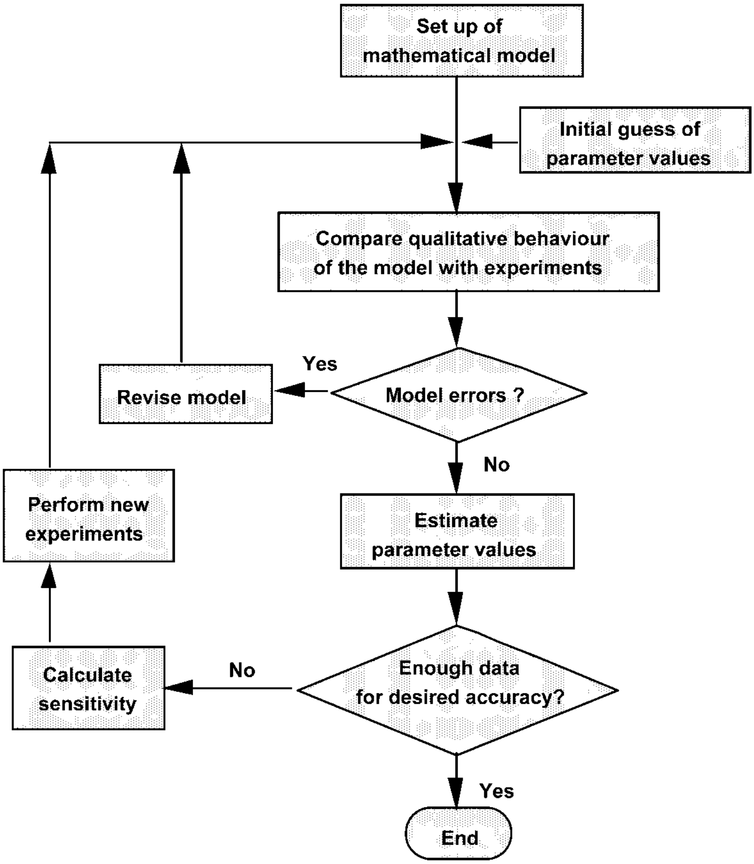

Fig. 2.29 Iterative procedure for parameter estimation, sensitivity analysis and experimentation.

fied. Figure 2.29 illustrates an iterative experimental procedure for parameter estimation, involving sensitivity analysis.

Below the results of Sensitivity Runs with MADONNA are given from the BIOREACT example that is run as a batch fermenter system. This example involves Monod growth kinetics, as explained in Section 1.4. In this example, the sensitivity of biomass concentration X, substrate concentration S and product concentration to changes in the Monod kinetic parameter, K_S, was investigated. Qualitatively, it can be deduced that the sensitivity of the concentrations to K_S should increase as the concentration of S becomes low at the end of the batch. This is verified by the results in Fig. 2.30. The results in Fig. 2.31 give the sensitivity of biomass concentration X and substrate concentration S to another biological kinetic parameter, the yield coefficient Y, as defined in Section 1.4.

Sensitivity analysis is a very important tool in analysing the relative importance of the model parameters and in the design of experiments for their optimal determination. In many cases, it is found that a model may be rather in-

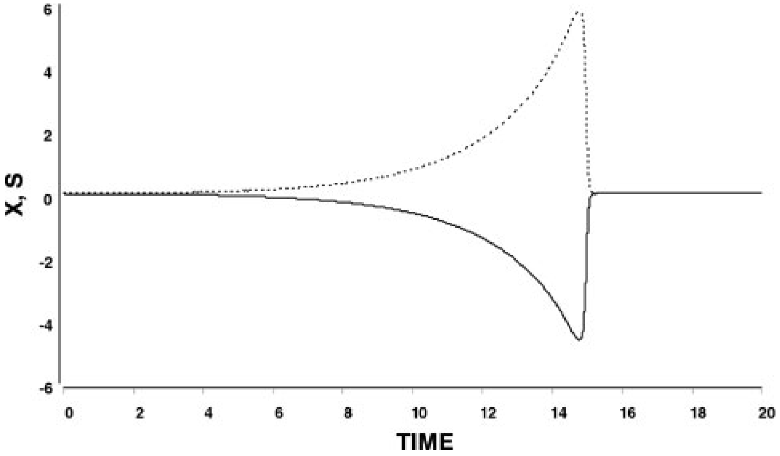

Fig. 2.30 Sensitivity of biomass concentration X and substrate concentration S ($\Delta X/\Delta K_S$ solid line and $\Delta S/\Delta K_S$ dotted line) to changes in K_S from a batch run of example BIOREACT.

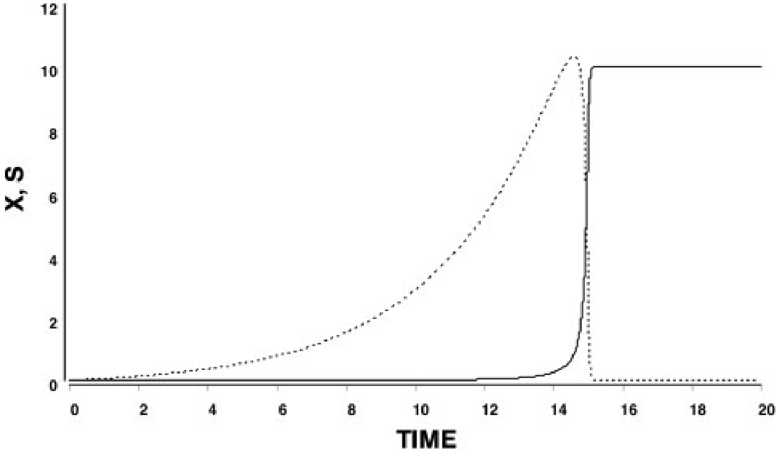

Fig. 2.31 Sensitivity of biomass X and substrate concentrations S ($\Delta X/\Delta Y$ solid line and $\Delta S/\Delta Y$ dashed line) to changes in the yield coefficient Y from the run of Fig. 2.31.

sensitive to a particular parameter value in the region of main interest, and then the parameter obviously does not need to be determined very accurately.

Model parameters are usually determined from experimental data. In doing this, sensitivity analysis is valuable in identifying the best experimental conditions for the estimation of a particular model parameter. Sensitivity analysis is easy effected with MADONNA, and sensitivity analysis is also provided in other more advanced software packages, such as ACSL-OPTIMIZE.

2.4.4
Numerical Integration

Only a very short introduction to numerical integration is given here, simply to demonstrate the basic principles and possible sources of error. In the great majority of simulation studies, the numerical integration will not be found to create problems and a detailed knowledge of the differing numerical integration methods is generally unnecessary. For more complex problems, where numerical difficulties may occur, the reader is referred to more specialist texts, e.g., Press et al. (1992), Walas (1991), Noye (1984).

In the solution of mathematical models by digital simulation, the numerical integration routine is usually required to achieve the solution of sets of simultaneous, first-order differential equations in the form

$$\frac{dy_i}{dt} = f_i(y_1, y_2, y_3, y_4, \ldots y_n) \quad \text{for } i = 1, 2, \ldots, m$$

The differential equations are often highly non-linear and the equation variables are often highly interrelated. In the above formulation, y_i represents any one of the dependent system variables and f_i is the general function relationship, relating the derivative, dy_i/dt, with the other related dependent variables. The system-independent variable, t, will usually correspond to time, but may also represent distance, for example, in the simulation of steady-state models of tubular and column devices.

In order to solve the differential equations, it is first necessary to initialise the integration routine. In the case of initial value problems, this is done by specifying the conditions of all the dependent variables, y_i, at initial time $t = 0$. If, however, only some of the initial values can be specified and other constant values apply at further values of the independent variable, the problem then becomes one of a split-boundary type. Split-boundary problems are inherently more difficult than the initial value problems, and although most of the examples in the book are of the initial value type, some split-boundary problems are presented.

In both types of problem, solution is usually achieved by means of a step-by-step integration method. The basic idea of this is illustrated in the information flow sheet, which was considered previously for the introductory MADONNA complex reaction model example (Fig. 1.4).

Referring to Fig. 1.4, the solution begins with the initial concentration conditions A_0, B_0, C_0 and D_0, defined at time $t = 0$. Knowing the magnitudes of the kinetic rate constants k_1, k_2, k_3 and k_4, thus enables the initial rates of change dC_A/dt, dC_B/dt, dC_C/dt and dC_D/dt, to be determined. Extrapolating these rates over a short period of time Δt, from the initial conditions A_0, B_0, C_0 and D_0, enables new values for A, B, C and D to be estimated at the new time, $t = t + \Delta t$. If the incremental time step Δt is sufficiently small, it is assumed that the error in the new estimated values of the concentration, A, B, C and D, will also be small. This procedure is then repeated for further small increments of time until the entire concentration versus time curves have been determined.

In this approach, the true solution is approximated as a series of discrete points along the axis of the independent variable t. The solution then proceeds step-by-step from one discrete time step to the next. In the simplest case, the time steps will be spaced at uniform intervals, but the spacing can also be varied during the course of solution. Where there are several dependent variables involved, all the variables must be updated to their new value, by projecting all the respective rates of change or concentration-time gradients over the identical time increment or integration step length, h. In order to do this, the integration method has to carry out a number of separate evaluations of the gradient terms. These evaluations are known as the "stages" of the computation.

Thus taking the single rate equation $dy/dt = f(y,t)$ and knowing the solution at any point $P_n(y_n, t_n)$, the value of the function at the next point can be predicted, knowing the local rate of change dy_n/dt_n. In the simplest case, this can be approximated by a simple difference approximation

$$\frac{d_{yn}}{dt} = f(y_n, t_n) = \frac{y_{n+1} - y_n}{t_{n+1} - t_n} = \frac{y_{n+1} - y_n}{h_n}$$

from which

$$y_{n+1} = y_n + h_n f(y_n, t_n)$$

This procedure is illustrated in Fig. 2.32.

Much effort has been devoted to producing fast and efficient numerical integration techniques, and there is a very wide variety of methods now available. The efficiency of an integration routine depends on the number of function evaluations, required to achieve a given degree of accuracy. The number of evaluations depends both on the complexity of the computation and on the number of integration step lengths. The number of steps depends on both the na-

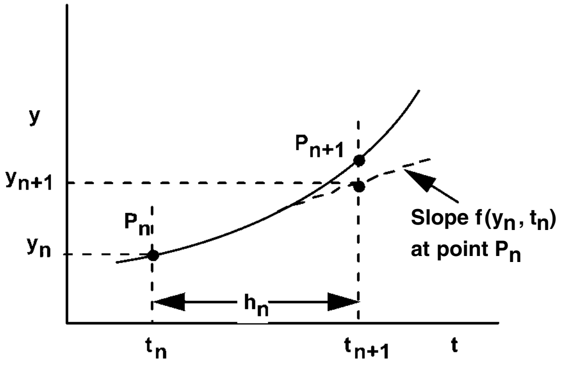

Fig. 2.32 The difference approximation for rate of change or slope.

ture and complexity of the problem and the degree of accuracy required in the solution. In practice, an over complex integration routine will require excessive computing time, owing to the many additional function evaluations that are required, and the use of an inappropriate integration algorithm can lead to an inaccurate solution, excessive computing time and, sometimes, a complete inability to solve the problem.

It is thus very important that the output of any simulation is checked, using other integration methods. Most simulation languages allow a choice of integration routine which can be made best on the basis of experience. It is important to remember that all methods generate only approximate solutions, but these must be consistent with a given error criterion. As the models themselves, however, are also approximate, errors in the numerical solution must be seen in the general context of the problem as a whole. Numerical errors occur in the approximation of the original function and also are due to limits in the numerical precision of the computation. From experience, it can be shown that most cases of "strange" behaviour, in the results of a simulation, can be attributed largely to errors in the model and inadequate model parameter selection, rather than to numerical inaccuracies. Very powerful integration routines for stiff systems are supplied by MATLAB.

Integration methods used by MADONNA

1. Fixed step Euler method (Euler).
2. Fixed step, 2nd-order, Runge-Kutta method (RK2).
3. Fixed step, 4th-order, Runge-Kutta method (RK4).
4. Variable step, 5th-order, Runge-Kutta method (AUTO).
5. Rosenbrock Method (Stiff).

Important integration parameters and default settings

METHOD	Choice of integration method as above (Euler).
DT	Calculation interval or integration step length in fixed step methods (0.25).
DTOUT	Output time interval.
STOPTIME	Value of the independent variable with which the run is terminated (12).
STARTTIME	Value of the independent variable at the start of the run (0).
TOLERANCE	Relative accuracy for the Auto and Stiff methods (0.01).

The most common numerical problem, as shown by some of the simulation examples, is that of equation stiffness. This is manifested by the need to use shorter and shorter integration step lengths, with the result that the solution proceeds more and more slowly and may come to a complete halt. Such behaviour is exhibited by systems having combinations of very fast and very slow processes. Stiff systems can also be thought of as consisting of differential equations, having large differences in the process time constants. Sometimes, the stiffness is the result of bad modelling practice and can be removed by assuming the very fast processes to be virtually instantaneous, as compared to the slower overall rate determining processes. In this way, the differential equations involving the troublesome very fast processes are replaced by steady-state algebraic equations, in which the rate of accumulation is, in effect, taken to be zero. Solving the implicit equations that result from such procedures often requires a root finder algorithm, as is supplied by MADONNA. Unfortunately, this technique is not always possible, and many systems are stiff in their own right and therefore need special integration methods.

2.4.5
System Stability

System instability can also be a problem in dynamic simulation, and this can origin either from the integration routine or from the model itself. Instability in the integration routine can arise owing to the approximation of the real functions by finite-difference approximations, which can have their own parasitical exponential solutions. When the unwanted exponentials decrease with respect to time, the numerical solution will be stable, but if the exponential is positive then this can increase very rapidly and either swamp or corrupt the solution, sometimes in a manner that may be difficult to detect. Many integration algorithms show a dependence of the stability of the solution on the integration step length. There can thus be a critical integration step, which if exceeded can lead to instability. This type of instability can be seen in many of the simulation examples, where an injudicious choice of DT can cause numerical overflow. Reducing the integration step size makes the solution run more slowly, and rounding off errors caused by the limited accuracy of the digital representation may then become important. Practically, one can try to solve such problems by changing the integration routine or by adjusting the error criteria in MADONNA.

Model instability is demonstrated by many of the simulation examples and leads to very interesting phenomena, such as multiple steady states, naturally occurring oscillations, and chaotic behaviour. In the case of a model which is inherently unstable, nothing can be done except to completely reformulate the model into a more stable form.

In general, the form of the solution to the dynamic model equations will be in the form

$$y_i(t) = \text{steady-state solution} + \text{transient solution}$$

where the transient part of the solution can be represented by a series of exponential functions

$$Y_{\text{trans}} = A_1 e^{\lambda t} + A_2 e^{\lambda_2 \cdot t} + \ldots + A_n e^{\lambda_n t}$$

In the above relationship, the coefficients A_1 to A_n depend on the initial conditions of the problem and the exponential values, λ_i, are determined by the parameters of the system and in fact represent the eigenvalues or roots of the characteristic solution of the system.

In a stable system, the above transient terms must decay to zero to give the steady-state solution. This applies when all the roots are simple negative exponentials. The system is also stable with all the roots occuring as negative real parts of complex roots, causing a decaying oscillatory approach to the eventual steady-state condition. If any of the roots are positive real numbers or complex numbers with real positive parts, the corresponding transient terms in the solution will grow in magnitude, thus directing the solution away from the unstable steady-state condition. Where the roots of the transient solution are pure imaginary numbers, the result is an oscillation of constant amplitude and frequency. The dynamic stability of such systems is often shown most conveniently on a phase-plane diagram, as shown in several of the simulation examples. Model instability is discussed further in Section 3.2.7, with regard to the stability of continuous stirred-tank reactors.

For a fuller treatment of dynamic stability problems, the reader is referred to Walas (1991), Seborg et al. (1989), Perlmutter (1972) and to the simulation examples THERM, THERMPLOT, COOL, REFRIG1, REFRIG2 and OSCIL.