Chapter 7

Development of Empirical Models from Process Data

CHAPTER CONTENTS

7.1 Model Development Using Linear or Nonlinear Regression
- 7.1.1 Model Building Procedure
- 7.1.2 Linear Regression
- 7.1.3 Nonlinear Regression

7.2 Fitting First- and Second-Order Models Using Step Tests
- 7.2.1 Graphical Techniques for Second-Order Models
- 7.2.2 Nonlinear Regression of Step Response Data
- 7.2.3 Other Types of Input Excitation

7.3 Neural Network Models
- 7.3.1 Soft Sensors

7.4 Development of Discrete-Time Dynamic Models
- 7.4.1 Exact Discrete-Time Models

7.5 Identifying Discrete-Time Models from Experimental Data
- 7.5.1 Process Identification of More Complicated Models

Summary

Several modeling approaches are used in process control applications. Theoretical models based on the chemistry and physics of the process represent one alternative. However, the development of rigorous theoretical models may not be practical for complex processes if the model requires a large number of equations with a significant number of process variables and unknown parameters (e.g., chemical and physical properties). An alternative approach is to develop an empirical model directly from experimental data. Empirical models are sometimes referred to as *black box* models, because the process being modeled can be likened to an opaque box. Here the input and output variables (u and y, respectively) are known, but the inner workings of the box are not. (See Fig. 7.1, where vectors of time-varying variables $\boldsymbol{u}(t)$, $\boldsymbol{y}(t)$, and $\boldsymbol{d}(t)$ are shown.) The development of empirical steady-state and dynamic models is the subject of this chapter. This activity is referred to as *process* or *system identification* (Ljung and Glad, 1994; Ljung, 1999). In general, empirical dynamic models are simpler than theoretical models and offer the advantage that they can be solved in "real time." In other words, the computational time required for the model solution (e.g., transient response) is much shorter than the actual process response time. However, this may not be true for complex models with many variables and equations.

The key differences between process simulation and process identification can be summarized with the aid of Fig. 7.1. In simulation, the process model $\mathcal{M}$ is known, and we wish to generate the response $\boldsymbol{y}(t)$ for a specified input $\boldsymbol{u}(t)$ and a specified disturbance $\boldsymbol{d}(t)$. If $\mathcal{M}$ is a linear dynamic model and $\boldsymbol{u}(t)$ and $\boldsymbol{d}(t)$ are expressed

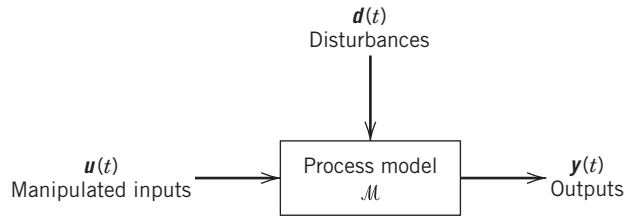

Figure 7.1 Input–output process model.

analytically, $\boldsymbol{y}(t)$ can be derived using Laplace transforms (see Chapter 4). Alternatively, $\boldsymbol{y}(t)$ can be calculated numerically using software packages such as MATLAB (Ljung, 2007). If $\mathcal{M}$ is a nonlinear dynamic model, $\boldsymbol{y}(t)$ can be obtained by numerical integration (cf. Chapter 2) after $\boldsymbol{u}(t)$ and $\boldsymbol{d}(t)$ are specified. By contrast, in process identification the model $\mathcal{M}$ is determined from data for $\boldsymbol{u}(t)$, $\boldsymbol{y}(t)$, and $\boldsymbol{d}(t)$, if $\boldsymbol{d}$ can be measured. If the model structure is postulated but contains unknown model parameters, then the model parameters can be obtained using regression techniques. This parameter estimation can be done with commercially available software regardless of whether the process model is linear or nonlinear, or whether it is theoretically based or empirical in nature.

Steady-state empirical models can be used for instrument calibration, process optimization, and specific instances of process control. Single-input, single-output (SISO) models typically consist of simple polynomials relating an output to an input. Dynamic empirical models can be employed to understand process behavior during upset conditions. They are also used to design control systems and to analyze their performance. Empirical dynamic models typically are low-order differential equations or transfer function models (e.g., first- or second-order model, perhaps with a time delay), with unspecified model parameters to be determined from experimental data. However, in some situations, more complicated models are valuable in control system design, as discussed later in this chapter.

The concept of a *discrete-time model* will now be introduced. These models are generally represented by difference equations rather than differential equations. Most process control tasks are implemented via digital computers, which are intrinsically discrete-time systems. In digital control, the continuous-time process variables are sampled at regular intervals (e.g., every 0.1 s); hence, the computer calculations are based on sampled data rather than continuous measurements. If process variables are observed only at the sampling instants, the dynamic behavior can be modeled using a discrete-time model in the form of a difference equation. The selection of discrete-time models over continuous-time models is becoming commonplace, especially for advanced control strategies.

Several methods for determining steady-state and dynamic empirical models for both continuous-time and discrete-time model types will now be presented. We first consider general model-fitting techniques based on linear and nonlinear regression that can be used to calculate model parameters for any type of model. Then simple but very useful methods are presented for obtaining first-order and second-order linear dynamic models from step response data using analytical solutions. These methods yield models suitable for the design of control systems; however, the resulting models are usually accurate only for a narrow range of operating conditions close to the nominal steady state, where the process exhibits linear behavior. We also show the relationship between continuous-time and discrete-time models. Finally, we present several methods for developing linear discrete-time models for dynamic processes.

7.1 MODEL DEVELOPMENT USING LINEAR OR NONLINEAR REGRESSION

Before developing an empirical model for two variables, a single input u and a single output y, it is instructive first to plot the available data (e.g., y vs. u for steady-state data and y and u vs. time for transient response data). From these plots, it may be possible to visualize overall trends in the data and to select a reasonable form for the model. After the model form is selected, the unknown model parameters can be calculated and the model accuracy evaluated. This parameter calculation procedure is referred to as *parameter estimation* or regression (Ljung, 1999; Montgomery and Runger, 2013). These calculations are usually based on model fitting, that is, minimizing a measure of the differences between model predictions and data. However, the problem of fitting a model to a set of input–output data becomes complicated when the model relation is not simple or involves multiple inputs and outputs.

First, we consider steady-state models. Suppose that a set of steady-state input–output data is available and shown as circles in Fig. 7.2. Variable y represents a process output (e.g., a reactor yield), whereas u represents an input variable (e.g., an operating condition such as temperature). Although a straight-line model (Model 1) provides a reasonable fit, higher-order polynomial relations (Models 2 and 3) result in smaller errors between the data and the curve representing the empirical model. Models 2 and 3 provide better agreement with the data at the expense of greater complexity because more model parameters must be determined. Sometimes the model form may be known from theoretical considerations or past experience with the process.

In Fig. 7.2, if the actual process behavior is linear, the differences (or residuals) between Model 1 and the data

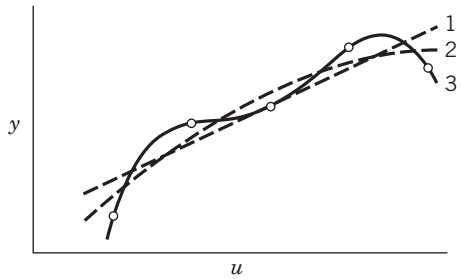

Figure 7.2 Three models for scattered data.

could be due to process disturbances or measurement errors. In empirical modeling, it is preferable to choose the simplest model structure that yields a good fit of the data, providing that the model is physically reasonable. Note that in Fig. 7.2, if Model 3 is extrapolated beyond the data range, it would apparently yield significantly different model predictions than Model 1 or 2. The selection of the best model might require collecting more data, perhaps outside the range shown in Fig. 7.2, which then could be used to *validate* each model.

7.1.1 Model Building Procedure

In this section, we present a systematic procedure for developing empirical dynamic models (Ljung, 1999). The procedure consists of the following steps:

1. Formulate the model objectives; that is, how will the model be used, and who will be the user?
2. Select the input and output variables for the model.
3. Evaluate available data and develop a plan to acquire additional data. A testing plan would specify the values of u or the form of $u(t)$, for example, a step change or some other input sequence (see Section 7.2).
4. Select the model structure and level of model complexity (e.g., steady-state vs. dynamic model, linear vs. nonlinear model).
5. Estimate the unknown model parameters using linear or nonlinear regression.
6. Using input and output data, evaluate model accuracy based on statistical considerations. It is desirable to use new data (if available) as well as the "old" data that were used to develop the model. If the model does not provide a satisfactory fit, return to Step 2 and try a different model. If possible, the model should be tested with new data (that is, *validation data*); if the model predictions agree with these data, the model is said to be validated.
7. For a dynamic model, nonstatistical criteria also can be used to evaluate a model, such as speed of response, shape of response, correct stability properties, and correct steady-state gain. The utility of a model for designing controllers is also important in process control, where an overly complex model can be a disadvantage. Thus *control-relevant models* are desirable (Rivera and Jun, 2000).

7.1.2 Linear Regression

Statistical analysis can be used to estimate unknown model parameters and to specify the uncertainty associated with the empirical model. It can also be used to compare several candidate models (Draper and Smith, 1998; Montgomery and Runger, 2013). For linear models, the *least-squares* approach is widely used to estimate model parameters. Consider the linear (or straight-line) model in Fig. 7.2 (Model 1) and let Y_i represent the data point where $\hat{y}_i$ is the model prediction for $u = u_i$. Then for the model, $y = \beta_1 + \beta_2 u + \epsilon$, the individual data points can be expressed as

$$Y_i = \beta_1 + \beta_2 u_i + \epsilon_i \tag{7-1}$$

where β_1 and β_2 are the model parameters to be estimated. ϵ_i is the random error for the particular data point.

The least-squares method is the standard approach for calculating the values of β_1 and β_2 that minimize the sum of the squares of the errors S for an arbitrary number of data points, N:

$$S = \sum_{i=1}^{N} \epsilon_i^2 = \sum_{i=1}^{N} (Y_i - \beta_1 - \beta_2 u_i)^2 \tag{7-2}$$

In Eq. 7-2, note that the values of Y_i and u_i are known, while β_1 and β_2 are to be calculated so as to minimize S, the objective function. The optimal estimates of β_1 and β_2 calculated for a specific data set are designated as $\hat{\beta}_1$ and $\hat{\beta}_2$. The model predictions are given by the regression model:

$$y = \hat{\beta}_1 + \hat{\beta}_2 u \tag{7-3}$$

and the *residuals* e_i are defined as

$$e_i \triangleq Y_i - \hat{y}_i \tag{7-4}$$

These least-squares estimates will minimize the least squares index in Eq. 7-2 if two ideal conditions are satisfied (Montgomery and Runger, 2013):

1. The model structure is correct.
2. The random errors ϵ_i are independent and normally distributed.

In practical problems, both ideal conditions are seldom realized. Nevertheless, the estimated least squares models can be satisfactory if the selected model structure is appropriate.

A general nonlinear steady-state model that is linear in the parameters has the form

$$y = \sum_{j=1}^{p} \beta_j U_j + \epsilon \tag{7-5}$$

The p unknown parameters (β_j) are to be estimated, and the U_j are the p specified functions of u. Note that the unknown parameters β_j appear linearly in the model, and a constant term can be included by setting $U_1 = 1$.

The sum of the squares of the errors (SSE) analogous to Eq. 7-2 is

$$S = \sum_{i=1}^{N} \left(Y_i - \sum_{j=1}^{p} \beta_j U_{ij} \right)^2 \tag{7-6}$$

For U_{ij} the first subscript corresponds to the ith data point, and the second index refers to the jth function of u.

The least squares solution for the unknown parameters can be obtained using software such as Excel Solver, which is based on nonlinear programming (Edgar et al., 2001). The least-squares solution provides a *point estimate* for the unknown model parameters β_i but does not indicate how accurate the estimates are. The degree of accuracy is expressed by confidence intervals that have the form, $\hat{\beta}_i \pm \Delta\beta_i$. The $\Delta\beta_1$ are calculated from the (u, y) data for a specified *confidence level* (Montgomery and Runger, 2013).

Next we consider the development of a steady-state performance model, such as might be used in optimizing the operating conditions of an electrical power generator (see Chapter 19).

EXAMPLE 7.1

An experiment has been performed to determine the steady-state power delivered by a gas turbine-driven generator as a function of fuel flow rate. The following normalized data were obtained:

Fuel Flow Rate u_i	**Power Generated** Y_i
1.0	2.0
2.3	4.4
2.9	5.4
4.0	7.5
4.9	9.1
5.8	10.8
6.5	12.3
7.7	14.3
8.4	15.8
9.0	16.8

Compare the best linear and quadratic models using Excel Solver.

SOLUTION

Solving for $\hat{\beta}_1$ and $\hat{\beta}_2$ using Excel yields $\hat{\beta}_1 = 0.0785$ and $\hat{\beta}_2 = 1.859$.

Next we determine how much the model accuracy can be improved by using a quadratic model $Y = \beta_1 + \beta_2 u + \beta_3 u^2$, the estimated parameters for a quadratic model are

$$\hat{\beta}_1 = 0.1707, \quad \hat{\beta}_2 = 1.811, \quad \text{and} \quad \hat{\beta}_3 = 0.0047$$

The predicted values of $y(\hat{y})$ are compared with the measured values (actual data) in Table 7.1 for both the linear and quadratic models. It is evident from this comparison that the linear model is adequate and that little improvement results from the more complicated quadratic model.

Table 7.1 A Comparison of Model Predictions from Example 7.1 (SSE = Sum of squared errors)

		Linear Model Prediction	Quadratic Model Prediction
u_i	y_i	$\hat{y}_{1i} = \hat{\beta}_1 + \hat{\beta}_2 u_i$	$\hat{y}_{2i} = \hat{\beta}_1 + \hat{\beta}_2 u_i + \hat{\beta}_3 u_i^2$
1.0	2.0	1.94	1.99
2.3	4.4	4.36	4.36
2.9	5.4	5.47	5.46
4.0	7.5	7.52	7.49
4.9	9.1	9.19	9.16
5.8	10.8	10.86	10.83
6.5	12.3	12.16	12.14
7.7	14.3	14.40	14.40
8.4	15.8	15.70	15.72
9.0	16.8	16.81	16.85
		SSE = 0.0613	SSE = 0.0540

7.1.3 Nonlinear Regression

If the empirical model is nonlinear with respect to the model parameters, then nonlinear regression rather than linear regression must be used. For example, suppose that a reaction rate expression of the form $r_A = kc_a^n$ is to be fit to experimental data, where r_A is the reaction rate of component A, c_A is the reactant concentration, and k and n are model parameters.

This model is *linear* with respect to rate constant k but is *nonlinear* with respect to reaction order n. A general nonlinear model can be written as

$$y = f(u_1, u_2, u_3, \ldots \beta_1, \beta_2, \beta_3 \ldots) \tag{7-7}$$

where y is the model output, u_j are inputs, and β_j are the parameters to be estimated. In this case, the β_j do not appear linearly in the model. However, we can still define a sum of squares error criterion to be minimized by selecting parameters β_j:

$$\min_{\beta_j} S = \sum_{i=1}^{N} (Y_i - \hat{y}_i)^2 \tag{7-8}$$

where Y_i and $\hat{y}_i$ denote the ith output measurement and model prediction corresponding to the ith data point, respectively. The least-squares estimates are again denoted $\hat{\beta}_j$.

Consider the problem of estimating the time constants for first-order and overdamped second-order dynamic models based on the measured output response to a step input change of magnitude M. Analytical expressions for these step response were developed in Chapter 5.

Transfer Function	***Step Response***	
$\frac{Y(s)}{U(s)} = \frac{K}{\tau s + 1}$	$y(t) = KM(1 - e^{-t/\tau})$	(5-16)
$\frac{Y(s)}{U(s)} = \frac{K}{(\tau_1 s + 1)(\tau_2 s + 1)}$	$y(t) = KM\left(1 - \frac{\tau_1 e^{-t/\tau_1} - \tau_2 e^{-t/\tau_2}}{\tau_1 - \tau_2}\right)$	(5-47)

In the step response equations, t is the independent variable instead of the input u used earlier, and y is the dependent variable expressed in deviation form. Although steady-state gain K appears linearly in both response equations, the time constants are contained in a nonlinear manner, which means that linear regression cannot be used to estimate them.

Sometimes a variable transformation can be employed to transform a nonlinear model so that linear regression can be used (Montgomery and Runger, 2013). For example, if K is assumed to be known, the first-order step response can be rearranged:

$$\ln\left(1 - \frac{y(t)}{KM}\right) = -\frac{t}{\tau} \tag{7-9}$$

Because $\ln(1 - y/KM)$ can be evaluated at each time t_i, this model is linear in the parameter $1/\tau$. Thus, this model has the standard linear form as Eq. 7-1, where the left-hand side of Eq. 7-9 is Y_i, $\beta_1 = 0$, and $u_i = t_i$.

The transformation in Eq. 7-9 leads to the *fraction incomplete response* method of determining first-order models discussed in the next section. However, for step responses of higher-order models, such as Eq. 5-47, the transformation approach is not feasible. For these calculations, we must use optimization software such as Excel Solver, as discussed in Section 7.1.2 to find the least-squares estimates of the time constants (Edgar et al., 2001).

As an alternative to nonlinear regression, a number of graphical correlations can be used quickly to find approximate values of τ_1 and τ_2 in second-order models. The accuracy of models obtained in this way is often sufficient for controller design. In the next section, we present several shortcut methods for estimating transfer function parameters based on graphical analysis.

7.2 FITTING FIRST- AND SECOND-ORDER MODELS USING STEP TESTS

A plot of the output response of a process to a step change in input is sometimes referred to as the *process reaction curve*. If the process of interest can be approximated by a first-order or second-order linear model, the model parameters can be obtained by inspection of the process reaction curve. For example, recall the first-order model expressed in deviation variables,

$$\tau\frac{dy}{dt} + y = Ku \tag{7-10}$$

where the system is initially at a steady state, with $u(0) = 0$ and $y(0) = 0$. If the input u is abruptly changed from 0 to M at time $t = 0$, the step response in Eq. 5-16 results. The normalized step response is shown in Fig. 7.3. The response $y(t)$ reaches 63.2% of its final value at $t = \tau$. The steady-state change in y, Δy, is given by $\Delta y = KM$. Rearranging Eq. 5-16 and taking the limit at $t = 0$, the initial slope of the normalized step response is

$$\frac{d}{dt}\left(\frac{y}{KM}\right)_{t=0} = \frac{1}{\tau} \tag{7-11}$$

Thus, as shown in Fig. 7.3, the intercept of the tangent at $t = 0$ with the horizontal line, $y/KM = 1$, occurs at $t = \tau$. As an alternative, τ can be estimated from a step response plot using the value of t at which the response is 63.2% complete, as shown in the following example.

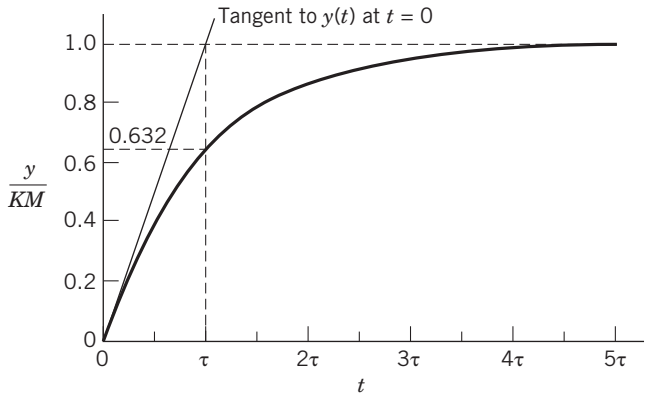

Figure 7.3 Step response of a first-order system and graphical constructions used to estimate the time constant, τ.

EXAMPLE 7.2

Figure 7.4 shows the response of the temperature T in a continuous stirred-tank reactor to a step change in feed flow rate w from 120 to 125 kg/min. Find an approximate first-order model for the process and these operating conditions.

SOLUTION

First note that $\Delta w = M = 125 - 120 = 5$ kg/min. Because $\Delta T = T(\infty) - T(0) = 160 - 140 = 20\,^{\circ}\text{C}$, the steady-state gain is

$$K = \frac{\Delta T}{\Delta w} = \frac{20\,^{\circ}\text{C}}{5\,\text{kg/min}} = 4\,\frac{^{\circ}\text{C}}{\text{kg/min}}$$

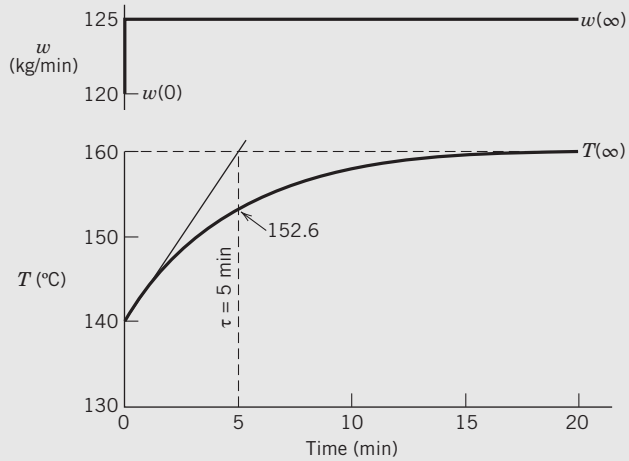

Figure 7.4 Temperature response of a stirred-tank reactor for a step change in feed flow rate.

The time constant obtained from the tangent construction shown in Fig. 7.4 is $\tau = 5$ min. Note that this result is consistent with the "63.2% method," because

$$T = 140 + 0.632(20) = 152.6\ ^\circ\text{C}$$

Consequently, the resulting process model is

$$\frac{T'(s)}{W'(s)} = \frac{4}{5s+1}$$

where the steady-state gain is 4 °C/kg/min.

Very few experimental step responses exhibit exactly first-order behavior, because

1. The true process model is usually *neither first-order nor linear*. Only the simplest processes exhibit such ideal dynamics.
2. The output data are usually corrupted with noise; that is, the measurements contain a random component. Noise can arise from normal operation of the process, for example, inadequate mixing that produces eddies of higher and lower concentration (or temperature), or from electronic instrumentation. If noise is completely random (i.e., uncorrelated), a first-order response plot may still be drawn that fits the output data well in a time-averaged sense. However, autocorrelated random noise, such as in drifting disturbances, can cause problems in the analysis.
3. Another process input (disturbance) may change in an unknown manner during the duration of the step test. In the CSTR example, undetected changes in inlet composition or temperature are examples of such disturbances.
4. It can be difficult to generate a perfect step input. Process equipment, such as the pumps and control valves discussed in Chapter 9, cannot be changed instantaneously from one setting to another but must be ramped over a period of time. However, if the ramp time is small compared to the process time constant, a reasonably good approximation to a step input may be obtained.

In summary, departures from the ideal response curve in Fig. 7.3 are common.

In order to account for higher-order dynamics that are neglected in a first-order model, a time-delay term can be included. This modification can improve the agreement between model and experimental responses. The fitting of a first-order-plus-time-delay model (FOPTD),

$$G(s) = \frac{Ke^{-\theta s}}{\tau s + 1} \tag{7-12}$$

to the actual step response requires the following steps, as shown in Fig. 7.5:

1. The process gain K is found by calculating the ratio of the steady-state change in y to the size of the input step change, M.
2. A tangent is drawn at the point of inflection of the step response; the intersection of the tangent line and the time axis (where $y = 0$) is the time delay.
3. If the tangent is extended to intersect the steady-state response line (where $y = KM$), the point of intersection corresponds to time $t = \theta + \tau$. Therefore, τ can be found by subtracting θ from the point of intersection.

The tangent method presented here for obtaining the time constant suffers from using only a single point to estimate the time constant. Use of several points from the response may provide a better estimate. Again

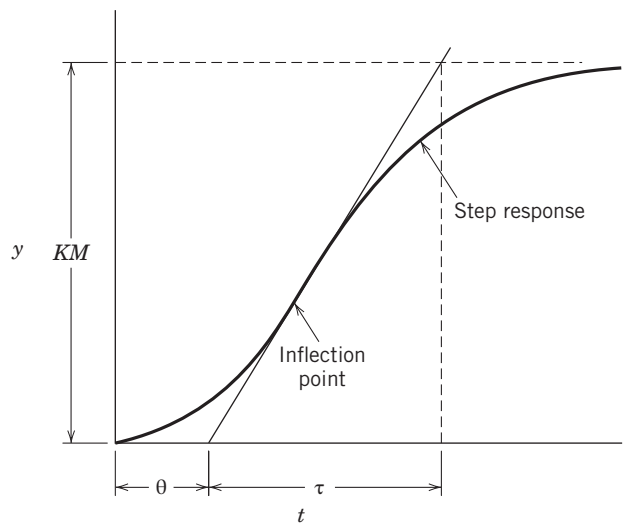

Figure 7.5 Graphical analysis of the process reaction curve to obtain parameters of a first-order-plus-time-delay model.

consider Eq. 7-9, but now introduce the time shift $t - \theta$ and rearrange to give the expression

$$\ln\left[\frac{y(\infty) - y_i}{y(\infty)}\right] = \frac{t_i - \theta}{\tau} \tag{7-13}$$

The final steady-state value, $y(\infty)$, equals KM. In Eq. 7-13, $y(\infty) - y_i$ can be interpreted as the *incomplete response* for data point i; dividing by $y(\infty)$ yields the *fraction incomplete response*: a semilog plot of $[y(\infty) - y_i]/y(\infty)$ vs. $(t_i - \theta)$ will then yield a straight line with a slope of $-1/\tau$, from which an average value of τ is obtained. An equation equivalent to Eq. 7-13 for the variables of Example 7.2 is

$$\ln\left[\frac{T(\infty) - T(t)}{T(\infty) - T(0)}\right] = -\frac{t - \theta}{\lambda\tau} \tag{7-14}$$

The major disadvantage of the time-delay estimation method in Fig. 7.5 is that it is difficult to find the point of inflection, as a result of measurement noise and small-scale recorder charts or computer displays. The method of Sundaresan and Krishnaswamy (1978) avoids use of the point of inflection construction entirely to estimate the time delay. They proposed that two times, t_1 and t_2, be estimated from a step response curve. These times correspond to the 35.3% and 85.3% response times, respectively. The time delay and time constant are then calculated from the following equations:

$$\theta = 1.3t_1 - 0.29t_2$$
$$\tau = 0.67(t_2 - t_1) \tag{7-15}$$

These values of θ and τ approximately minimize the difference between the measured response and the model response, based on a correlation for many data sets. By using actual step response data, model parameters K, θ, and τ can vary considerably, depending on the operating conditions of the process, the size of the input step change, and the direction of the change. These variations usually can be attributed to process nonlinearities and unmeasured disturbances.

7.2.1 Graphical Techniques for Second-Order Models

In general, a better approximation to an experimental step response can be obtained by fitting a second-order model to the data. Figure 7.6 shows the range of step response shapes that can occur for the second-order model,

$$G(s) = \frac{K}{(\tau_1 s + 1)(\tau_2 s + 1)} \tag{5-37}$$

Figure 7.6 includes two limiting cases: $\tau_2/\tau_1 = 0$, where the system becomes first-order, and $\tau_2/\tau_1 = 1$, the critically damped case. The larger of the two time constants, τ_1, is called the *dominant time constant*. The S-shaped response becomes more pronounced as the ratio of τ_2/τ_1 becomes closer to one.

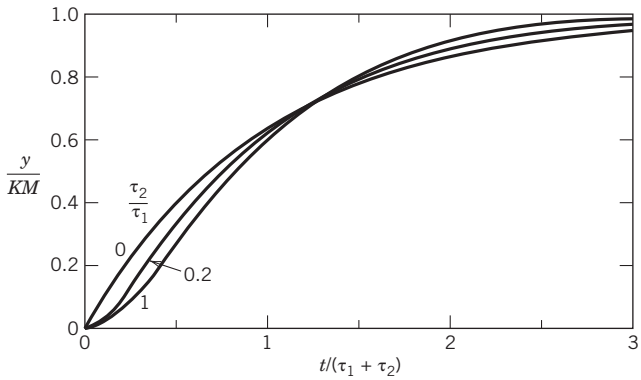

Figure 7.6 Step response for several overdamped second-order systems.

Model parameters for second-order systems which include time delays can be estimated using graphical or numerical methods. A method due to Smith (1972) utilizes a model of the form

$$G(s) = \frac{Ke^{-\theta s}}{\tau^2 s^2 + 2\zeta\tau s + 1} \tag{7-16}$$

which includes both overdamped and underdamped cases. Smith's method requires the times (with apparent time delay removed) at which the normalized response reaches 20% and 60%, respectively. Using Fig. 7.7, the ratio of t_{20}/t_{60} gives the value of ζ. An estimate of τ can be obtained from the plot of t_{60}/τ vs. t_{20}/t_{60}.

When graphically fitting second-order models, some caution must be exercised in estimating θ. A second-order model with no time delay exhibits a point-of-inflection (see Fig. 7.6 when $\tau_1 \approx \tau_2$). If the tangent to the point-of-inflection shown in Fig. 7.5 is applied to this case, however, a nonzero time delay is indicated. To avoid this conflict, visual determination of θ is recommended for graphical estimation, but estimation of θ by trial and error may be required

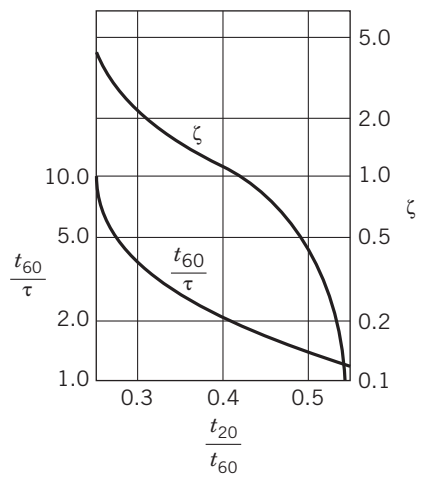

Figure 7.7 Smith's method: relationship of ζ and τ to τ_{20} and τ_{60}.

to obtain a good fit. In the folowing examples, the time delay is subtracted from the actual time value; then the adjusted time, $t' = t - \theta$, is employed for the actual graphical analysis. An alternative approach to fitting the parameters of the second-order model utilizes three points in the step response. Rangaiah and Krishnaswamy (1994, 1996).

7.2.2 Nonlinear Regression of Step Response Data

Model parameters of transfer function models can be estimated using nonlinear regression and standard software such as Excel and MATLAB. To use Excel, the measured data must be placed in one column. The model predictions to be compared with the measured data are placed in a second column. The sum of squares of the errors is calculated and put into a cell, called the target cell. The target cell value can be minimized using the built-in *Solver* in the *Tools* menu. The window of the Solver allows the user to select the cell to minimize/maximize, the range of cells to be adjusted (the model parameters), and the restrictions, if any, that apply. Clicking on ⟨solve⟩ will calculate the parameter values that minimize the sum of squares. The optimization method used by Excel is based on the generalized reduced gradient technique (Edgar et al., 2001).

In order to use MATLAB, it is necessary to write an M-file that defines the sum of squares of errors. Then the command *fminunc* is used to calculate the minimum. The default algorithm in MATLAB is the BFGS quasi-Newton method (Ljung, 2007).

EXAMPLE 7.3

Step test data have been obtained for the off-gas CO_2 concentration response obtained from changing the feed rate to a bioreactor. Use Smith's method as well as nonlinear regression based on Excel and MATLAB to estimate parameters in a second-order model from experimental step response data shown in Fig. 7.8. For all three methods, assume $\theta = 0$ because the response curve becomes nonzero immediately after $t = 0$. Compare the results with an FOPTD model that is fit using the 63.2% response method to estimate the time constant.

SOLUTION

Smith's Method

The two points of interest are the 20% response time, $t_{20} = 1.85$ min, and the 60% response time, $t_{60} = 5.0$ min. Hence, $t_{20}/t_{60} = 0.37$. From Fig. 7.7, $\zeta = 1.3$ and $t_{60}/\tau = 2.8$; thus, $\tau = 5.0/2.8 = 1.79$ min. Because the model is overdamped, the two time constants can be calculated from the following expressions modified from Eqs. 5-44 and 5-45:

$$\tau_1 = \tau\zeta + \tau\sqrt{\zeta^2 - 1}, \quad \tau_2 = \tau\zeta - \tau\sqrt{\zeta^2 - 1}$$

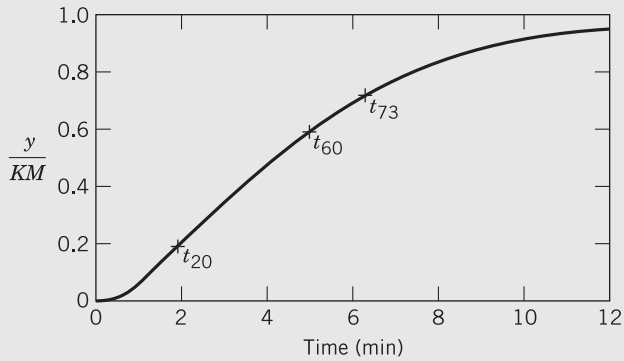

Figure 7.8 Normalized experimental step response.

Solving gives $\tau_1 = 3.81$ min and $\tau_2 = 0.84$ min.

For Fig. 7.8 the 63.2% response is estimated to occur at $t = 5.3$ min. Using the slope at the point of inflection, we can estimate the time delay to be $\theta = 0.7$ min. Note that $\tau = 4.6$ min, which is approximately equal to the sum of τ_1 and τ_2 for the second-order model.

Nonlinear Regression

Using Excel and MATLAB, we calculate the time constants in Eq. 5-47 that minimize the sum of the squares of the errors between experimental data and model predictions (see Eq. 7-8). The data consisted of 25 points between $t = 0$ and $t = 12$ min with a sampling period of 0.5 min. A comparison of the model parameters and the sum of squared errors for each method is shown below; the time delay is set to zero for the three second-order methods.

	τ_1 (min)	τ_2 (min)	Sum of Squares
Smith	3.81	0.84	0.0769
First order ($\theta = 0.7$ min)	4.60	—	0.0323
Excel and MATLAB	3.34	1.86	0.0057

Clearly, the nonlinear regression method is superior in terms of the goodness of fit, as measured by the sum of squares of the prediction error, but the required calculations are more complicated. Note that the nonlinear regression methods employed by Excel and MATLAB produce identical results.

The step responses are plotted in Fig. 7.9; all three calculated models give an acceptable fit to the original step response curve. In fact, the nonlinear regression model is indistinguishable from the experimental response. Nonlinear regression does not depend on graphical correlations and always provides a better fit to the data. It also permits the experimental step test to be terminated before the final steady state is reached; however, sufficient response data must be obtained for the regression method to be effective.

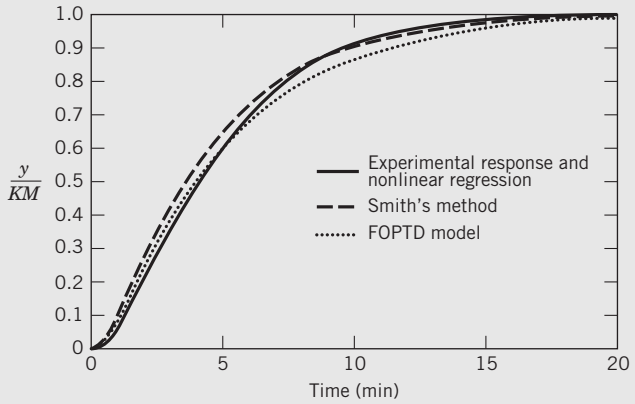

Figure 7.9 Comparison of step responses of fitted models with the original data.

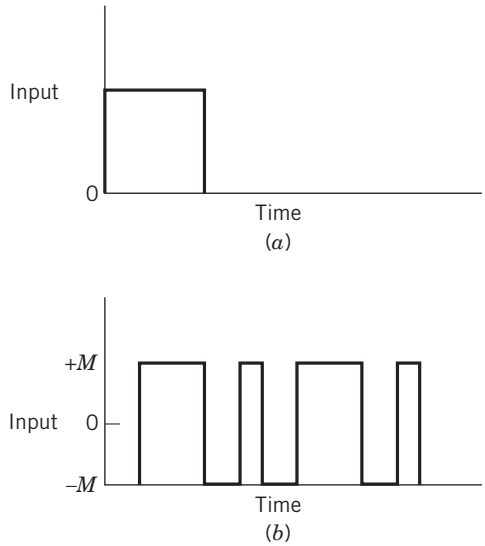

Figure 7.10 (*a*) Pulse and (*b*) PRBS inputs (one cycle).

7.2.3 Other Types of Input Excitation

Sometimes a step change in a process input is not permissible owing to safety considerations or the possibility of producing off-specification (*off-spec*) material as a result of the process output deviating significantly from the desired value. In these situations, other types of input changes that do not move the process to a new steady state can be selected. They include rectangular pulses (see Fig. 7.10), pulses of arbitrary shape, or even white (Gaussian) noise. Such "plant-friendly" inputs should be as short as possible, stay within actuator limits, and cause minimum disruption to the controlled variables (Rivera and Jun, 2000). For pulse forcing, the input is suddenly changed, left at its new value for a period of time, and then returned to its original value. Consequently, the process output also returns to its initial steady state, unless the process has an integrating mode.

Random Binary Sequence (RBS) forcing involves a series of pulses of fixed height and random duration. At each sampling instant, a random number generator determines whether the input signal is set at its maximum or minimum value. However, it is more convenient to implement a *pseudo random binary sequence (PRBS)*, which is a two-level, periodic, deterministic signal of a specified length, shown in Fig. 7.10. The actual sequence of inputs can be repeated multiple times. The term *pseudo random* indicates the input is a repeating sequence that has the spectral characteristics of a random signal (Godfrey, 1993). The advantage of a PRBS is that the input excitation can be concentrated in particular frequency ranges that correspond to the process dynamics and that are important for control system design (Rivera and Jun, 2000). See Chapter 14 for more information on frequency response analysis.

7.3 NEURAL NETWORK MODELS

Most industrial processes such as chemical reactors and separation systems exhibit nonlinear behavior. Unfortunately, many processes are so complex that significant engineering time and effort is required to develop and validate detailed theoretical dynamic models. As an alternative, an empirical nonlinear model can be obtained from experimental data. *Neural networks (NN)*, or artificial neural networks, are an important class of empirical nonlinear models. Neural networks have been used extensively in recent years to model a wide range of physical and chemical phenomena and to model other nonengineering situations such as stock market analysis, chess strategies, speech recognition, and medical diagnoses. Neural networks are attractive whenever it is necessary to model complex or little understood processes with large input–output data sets, as well as to replace models that are too complicated to solve in real time (Su and McAvoy, 1997; Himmelblau, 2008).

The exceptional computational abilities of the human brain have motivated the concept of an NN. The brain can perform certain types of computation, such as perception, pattern recognition, and motor control, much faster than existing digital computers (Haykin, 2009). The operation of the human brain is complex and nonlinear and involves massive parallel computation. Its computations are performed using structural constituents called *neurons* and the synaptic interconnections between them (i.e., a *neural network*). The development of artificial neural networks is an admittedly approximate attempt to mimic this biological neural network, in order to achieve some of its computational advantages.

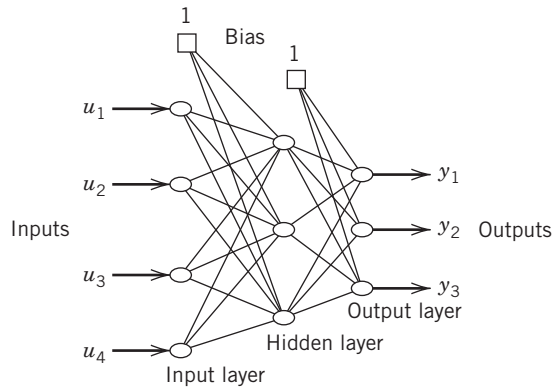

Figure 7.11 Multilayer neural network with three layers.

A multilayer feedforward network, one of the most common NN structures, is shown in Fig. 7.11. The *neurons* (or *nodes*) are organized into layers (input, output, hidden); each neuron in the hidden layer is connected to the neurons in adjacent layers via connection weights. These weights are unknown parameters that are estimated based on the input/output data from the process to be modeled. The number of unknown parameters can be quite large (e.g., 50–100), and powerful nonlinear programming algorithms are required to fit the parameters to the data using the least-squares objective function (Edgar et al., 2001). If enough neurons are utilized, an input–output process can be accurately modeled by a neural net model.

As shown in Fig. 7.12, at each neuron inputs are collected from other neurons or from bias terms, and their strength or magnitude is evaluated. These inputs are then summed and compared with a threshold level, and the appropriate output I_j is determined. The connection weight (W_{ij}) determines the relative importance of that input. The sum of the weighted inputs is then passed to a nonlinear transformation, as shown in Fig. 7.12, resulting in output y_j. One type of transformation has a sigmoidal shape as is shown in the figure, although many options are available.

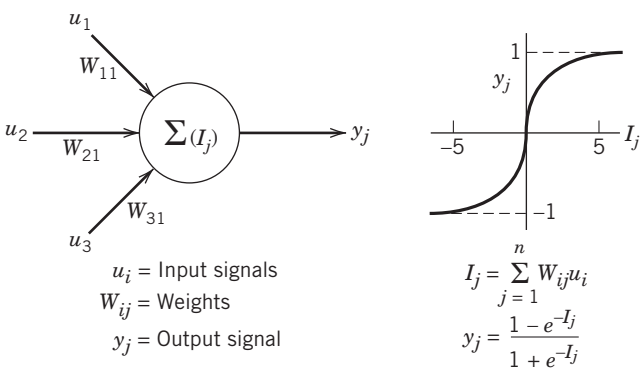

Figure 7.12 Signal diagram for a neuron.

The *training* of a neural network involves estimating the unknown parameters; this procedure generally utilizes normal operating data (often large data sets) taken in the operating region where the model is intended to be used. After the parameters are estimated (the network is *trained*), another large set of data can be used to determine the model is adequate *validation*. Sometimes the resulting NN model is not satisfactory, and changes in the model structure must be made, often by trial and error. Commercial software packages are available that make automatic empirical modeling of complex processes feasible.

Advanced applications of neural nets have been commercially implemented in the areas of fault detection and diagnosis, sensor errors, and dynamic modeling and control (Su and McAvoy, 1997). In some cases, neural nets have been used to determine controller settings in advanced control systems.

7.3.1 Soft Sensors

A common problem shared by many industrial processes is the inability to measure key process variables noninvasively and in real time, especially the compositions of process streams and product properties. The development of improved sensors, based on new techniques of analytical chemistry and modern electronic devices using fiber optics and semiconductors, has been an active area. As an alternative, the use of easily measured secondary variables to infer values of unmeasured process variables is now receiving great interest; the term *soft sensors* is often used to denote this approach. *Chemometrics* is a term related to soft sensors that describes how data from process analyzers (e.g., spectra) can be analyzed and modeled for use in process monitoring and control (Brown, 1998).

Soft sensors have become an attractive alternative to the high cost of accurate on-line measurements for applications where empirical models can accurately infer (i.e., predict) unmeasured variables (Kadlec et al., 2009). For example, the environmental regulatory agency in Texas permits NN models to be used for monitoring emissions from various process units such as power boilers. The NN models use measurements of selected input and output variables to predict pollutants at the parts per billion level (Martin, 1997). In materials manufacturing, the real-time detection of cracks, inclusions, porosity, dislocations, or defects in metallurgical or electronic materials would be highly desirable during processing, rather than after processing is completed and defective products are shipped. Use of virtual sensor models to predict quality control measures, such as the formation and location of defects, can greatly reduce the stringent requirements imposed on hardware-based sensors.

7.4 DEVELOPMENT OF DISCRETE-TIME DYNAMIC MODELS

A digital computer by its very nature deals internally with discrete-time data or numerical values of functions at equally spaced intervals determined by the sampling period. Thus, discrete-time models such as *difference equations* are widely used in computer control applications. One way a continuous-time dynamic model can be converted to discrete-time form is by employing a finite difference approximation (Chapra and Canale, 2014). Consider a nonlinear differential equation,

$$\frac{dy(t)}{dt} = f(y,\ u) \tag{7-17}$$

where y is the output variable and u is the input variable. This equation can be numerically integrated (although with some error) by introducing a finite difference approximation for the derivative. For example, the first-order, backward difference approximation to the derivative at $t = k\Delta t$ is

$$\frac{dy(t)}{dt} \cong \frac{y(k) - y(k-1)}{\Delta t} \tag{7-18}$$

where Δt is the integration interval specified by the user and $y(k)$ denotes the value of $y(t)$ at $t = k\Delta t$. Substituting Eq. 7-17 into Eq. 7-18 and evaluating $f(y, u)$ at the previous values of y and u (i.e., $y(k-1)$ and $u(k-1)$) gives

$$\frac{y(k) - y(k-1)}{\Delta t} \cong f(y(k-1),\ u(k-1)) \tag{7-19}$$

or

$$y(k) = y(k-1) + \Delta t f(y(k-1),\ u(k-1)) \tag{7-20}$$

Equation 7-20 is a first-order difference equation that can be used to predict $y(k)$ based on information at the previous time step $(k-1)$. This type of expression is called a *recurrence relation*. It can be used to numerically integrate Eq. 7-19 by successively calculating $y(k)$ for $k = 1, 2, 3, \ldots$ starting from a known initial condition $y(0)$ and a specified input sequence, $\{u(k)\}$. In general, the resulting numerical solution becomes more accurate and approaches the correct solution $y(t)$ as Δt decreases. However, for extremely small values of Δt, computer roundoff can be a significant source of error (Chapra and Canale, 2014).

EXAMPLE 7.4

For the first-order differential equation,

$$\tau\frac{dy(t)}{dt} + y(t) = Ku(t) \tag{7-21}$$

derive a recursive relation for $y(k)$ using a first-order backwards difference for $dy(t)/dt$.

SOLUTION

The corresponding difference equation after approximating the first derivative is

$$\frac{\tau(y(k) - y(k-1))}{\Delta t} + y(k-1) = Ku(k-1) \tag{7-22}$$

Rearranging gives

$$y(k) = \left(1 - \frac{\Delta t}{\tau}\right) y(k-1) + \frac{K\Delta t}{\tau} u(k-1) \tag{7-23}$$

The new value $y(k)$ is a weighted sum of the previous value $y(k-1)$ and the previous input $u(k-1)$. Equation 7-23 can also be derived directly from Eq. 7-20.

As shown in numerical analysis textbooks, the accuracy of Eq. 7-23 is influenced by the integration interval. However, discrete-time models involving no approximation errors can be derived for any linear differential equation under the assumption of a piecewise constant input signal, that is, the input variable u is held constant over Δt. Next, we develop discrete-time modeling methods that introduce no integration error for piecewise constant inputs, regardless of the size of Δt. Such models are important in analyzing computer-controlled processes where the process inputs are piecewise constant.

7.4.1 Exact Discrete-Time Models

For a process described by a linear differential equation, the corresponding discrete-time model can be derived from the analytical solution for a piecewise constant input. This analytical approach eliminates the discretization error inherent in finite-difference approximations. Consider a first-order model in Eq. 7-21 with previous output $y[(k-1)\Delta t]$ and a piecewise constant input $u(t) = u[(k-1)\Delta t]$ over the time interval, $(k-1)\Delta t \le t < k\Delta t$. The analytical solution to Eq. 7-21 at $t = k\Delta t$ is

$$y(k\Delta t) = (1 - e^{-\Delta t/\tau})Ku[(k-1)\Delta t] + e^{-\Delta t/\tau} y[(k-1)\Delta t] \tag{7-24}$$

Equation 7-24 can be written more compactly as

$$y(k) = e^{-\Delta t/\tau} y(k-1) + K(1 - e^{-\Delta t/\tau})u(k-1) \tag{7-25}$$

Equation 7-25 is the exact solution to Eq. 7-21 at the sampling instants provided that $u(t)$ is piecewise constant for each sampling interval of length Δt. Note that the continuous output $y(t)$ is not necessarily constant between sampling instants, but Eq. 7-25 provides an exact solution for $y(t)$ at the sampling instants, $k = 1, 2, 3, \ldots$

In general, when a linear differential equation of order p is converted to discrete time, a linear difference

equation of order p results. For example, consider the second-order model:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{K(\tau_a s + 1)}{(\tau_1 s + 1)(\tau_2 s + 1)} \tag{7-26}$$

The analytical solution for a piecewise constant input provides the corresponding difference equation, which is also referred to as an autogressive model with external (or exogenous) input, or *ARX model* (Ljung, 1999):

$$y(k) = a_1 y(k-1) + a_2 y(k-2) + b_1 u(k-1) + b_2 u(k-2) \tag{7-27}$$

where

$$a_1 = e^{-\Delta t/\tau_1} + e^{-\Delta t/\tau_2} \tag{7-28}$$

$$a_2 = -e^{-\Delta t/\tau_1} e^{-\Delta t/\tau_2} \tag{7-29}$$

$$b_1 = K\left(1 + \frac{\tau_a - \tau_1}{\tau_1 - \tau_2} e^{-\Delta t/\tau_1} + \frac{\tau_2 - \tau_a}{\tau_1 - \tau_2} e^{-\Delta t/\tau_2}\right) \tag{7-30}$$

$$b_2 = K\left(e^{-\Delta t(1/\tau_1 + 1/\tau_2)} + \frac{\tau_a - \tau_1}{\tau_1 - \tau_2} e^{-\Delta t/\tau_2} + \frac{\tau_2 - \tau_a}{\tau_1 - \tau_2} e^{-\Delta t/\tau_1}\right) \tag{7-31}$$

In Eq. 7-27 the new value of y depends on the values of y and u at the two previous sampling instants; hence, it is a second-order difference equation. If $\tau_2 = \tau_a = 0$ in Eqs. 7-27 through 7-31, the first-order difference equation in Eq. 7-24 results.

The steady-state gain of the second-order difference equation model can be found by considering steady-state conditions. Let $\bar{u}$ and $\bar{y}$ denote the new steady-state values after a step change in u. Substituting these values into Eq. 7-27 gives

$$\bar{y} = a_1 \bar{y} + a_2 \bar{y} + b_1 \bar{u} + b_2 \bar{u} \tag{7-32}$$

Because y and u are deviation variables, the steady-state gain is simply $\bar{y}/\bar{u}$, the steady-state change in y divided by the steady-state change in u. Rearranging Eq. 7-32 gives

$$\text{Gain} = \frac{\bar{y}}{\bar{u}} = \frac{b_1 + b_2}{1 - a_1 - a_2} \tag{7-33}$$

Substitution of Eqs. 7-28 through 7-31 into Eq. 7-33 gives K, the steady-state gain for the transfer function model in Eq. 7-26.

Higher-order linear differential equations can be converted to a discrete-time, difference equation model using a state-space analysis (Åström and Wittenmark, 1997).

7.5 IDENTIFYING DISCRETE-TIME MODELS FROM EXPERIMENTAL DATA

If a linear discrete-time model is desired, one approach is to fit a continuous-time model to experimental data (cf. Section 7.2) and then to convert it to discrete-time form using the above approach. A more attractive approach is to estimate parameters in a discrete-time model directly from input–output data based on linear regression. This approach is an example of *system identification* (Ljung, 1999). As a specific example, consider the second-order difference equation in Eq. 7-27. It can be used to predict $y(k)$ from data available at times, $(k-1)\Delta t$ and $(k-2)\Delta t$. In developing a discrete-time model, model parameters a_1, a_2, b_1, and b_2 are considered to be unknown. They are estimated by applying linear regression to minimize the error criterion in Eq. 7-6 after defining

$$\beta^T = [a_1 \;\; a_2 \;\; b_1 \;\; b_2], \quad U_1 = y(k-1), \quad U_2 = y(k-2),$$
$$U_3 = u(k-1), \text{ and } U_4 = u(k-2)$$

EXAMPLE 7.5

Consider the step response data $y(k)$ in Table 7.2, which were obtained from Example 7.3 and Fig. 7.8 for $\Delta t = 1$. At $t = 0$ a unit step change in u occurs, but the first output change is not observed until the next sampling instant.

Estimate the model parameters in Eq. 7-27 from the input–output data. Compare this model with the models obtained in Example 7.3 using nonlinear regression. For $k < 0$, assume $y(k) = 0$ and $u(k) = 0$, that is, the system is at rest.

Table 7.2 Step Response Data

k	$y(k)$
1	0.058
2	0.217
3	0.360
4	0.488
5	0.600
6	0.692
7	0.772
8	0.833
9	0.888
10	0.925

SOLUTION

For linear regression, there are four independent variables, $y(k-1)$, $y(k-2)$, $u(k-1)$, $u(k-2)$, one dependent variable $y(k)$, and four unknown parameters (a_1, a_2, b_1, b_2). We structure the data for regression as shown in Table 7.3 and solve using Excel.

Table 7.4 compares the estimated parameters obtained by the two approaches. The results labeled nonlinear regression were obtained by fitting a continuous-time model (overdamped second order with time constants τ_1 and τ_2 and gain K) to the data using nonlinear regression. The continuous-time model was then converted to the corresponding discrete-time model using Eqs. 7-27 to 7-31.

Table 7.3 Data Regression for Example 7.5

$$\boldsymbol{Y} = \begin{bmatrix} 0.058 \\ 0.217 \\ 0.360 \\ 0.488 \\ 0.600 \\ 0.692 \\ 0.722 \\ 0.833 \\ 0.888 \\ 0.925 \end{bmatrix} \qquad \boldsymbol{U} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0.058 & 0 & 1 & 1 \\ 0.217 & 0.058 & 1 & 1 \\ 0.360 & 0.217 & 1 & 1 \\ 0.488 & 0.360 & 1 & 1 \\ 0.600 & 0.488 & 1 & 1 \\ 0.692 & 0.600 & 1 & 1 \\ 0.772 & 0.692 & 1 & 1 \\ 0.833 & 0.772 & 1 & 1 \\ 0.888 & 0.833 & 1 & 1 \end{bmatrix}$$

Table 7.4 Comparison of Estimated Model Parameters for Example 7.5

	Linear Regression	Nonlinear Regression
a_1	0.975	0.984
a_2	−0.112	−0.122
b_1	0.058	0.058
b_2	0.102	0.101
K	1.168	1.159

The parameters obtained from linear regression in Table 7.4 are slightly different from those for nonlinear regression. This result occurs because for linear regression, four parameters were estimated; with nonlinear regression, three parameters were estimated. The estimated gain for linear regression, $K = 1.168$, is about 1% higher than the value obtained from nonlinear regression.

Table 7.5 compares the simulated responses for the two empirical models. Linear regression gives slightly better predictions, because it fits more parameters. However, in this particular example, it is difficult to distinguish graphically among the three model step responses.

Table 7.5 Comparison of Simulated Responses for Various Difference Equation Models*

n	y	$\hat{y}_L$	$\hat{y}_N$
1	0.058	0.058	0.058
2	0.217	0.217	0.216
3	0.360	0.365	0.366
4	0.488	0.487	0.487
5	0.600	0.595	0.596
6	0.692	0.690	0.690
7	0.772	0.768	0.767
8	0.833	0.835	0.835
9	0.888	0.886	0.885
10	0.925	0.933	0.932

*y, experimental data; $\hat{y}_L$, linear regression; $\hat{y}_N$, nonlinear regression

Example 7.5 has shown how a second-order difference equation model can be fit to data directly. The linear regression approach can also be used for higher-order models, provided that the parameters still appear linearly in the model. It is important to note that the estimated parameter values depend on the sampling period Δt for the data collection.

An advantage of the regression approach is that it is not necessary to make a step change in u in order to estimate model parameters. Other types of discrete-time models are the finite impulse response (FIR) model and the step response model, which are considered in Chapter 20 because of their close association with model predictive control.

7.5.1 Process Identification of More Complicated Models

In this section, we briefly consider three classes of more complicated process models: MIMO (multiple input, multiple output) models, stochastic models, and nonlinear discrete-time models.

MIMO process modeling is inherently more complicated than SISO modeling. For linear systems, the Principle of Superposition holds, which allows MIMO models to be developed through a series of single step tests for each input, while holding the other inputs constant. For a process with three inputs ($\boldsymbol{u}$) and three outputs ($\boldsymbol{y}$), we can introduce a step change in u_1, and record the responses for y_1, y_2, and y_3. The three transfer functions involving u_1, namely

$$\frac{Y_1}{U_1} = G_{11}, \quad \frac{Y_2}{U_1} = G_{21}, \quad \frac{Y_3}{U_1} = G_{31}$$

can be obtained using the techniques described in Section 7.2. In a similar fashion, step changes in U_2 and U_3 can be introduced in order to determine the other six G_{ij}. Alternatively, discrete-time models can be developed for each output, as discussed earlier in this section, using linear regression techniques. See Chapter 20 for a discussion of how such models are developed and used in model predictive controller calculations.

It is also possible to use PRBS forcing (Fig. 7.10(b)) to obtain MIMO models. To generate a multi-input PRBS signal, it is desirable that the input changes be independent. One way to accomplish this is to implement "shifted" or delayed versions of a single PRBS signal in each input. This means that if a single-input PRBS test requires 20 hours, a three-input test is designed to take three times as long. Hokanson and Gerstle (1992) suggest that about 20 total moves in each independent variable should be made.

Linear discrete-time models can also be developed that include the effects of unmeasured stochastic disturbances. For example, separate process models and disturbance models, also called *noise models*, can be

obtained from data (Ljung, 1999). In this case the error term ϵ in Eq. 7-1 is not white noise but *colored noise* that is autocorrelated. In other words, there are underlying disturbance dynamics that causes ϵ to depend on previous values.

A variety of nonlinear discrete-time models have also been used in process control (Pearson, 1999). They include the neural net models discussed in Section 7.3 as well as nonlinear models obtained by adding nonlinear terms to the linear models of the previous section.

SUMMARY

When theoretical models are not available or are very complicated, empirical process models provide a viable alternative. In these situations, a model that is sufficiently accurate for control system design can often be obtained from experimental input/output data. Step response data can be analyzed graphically or by computer (nonlinear regression) to obtain a first- or second-order transfer function model. Discrete-time models in the form of linear difference equations are frequently used in process control. These models can be readily obtained by least-squares fitting of experimental response data.

REFERENCES

Åström, K. J., and B. Wittenmark, *Computer-Controlled Systems: Theory and Design*, 3rd ed., Prentice-Hall, Englewood Cliffs, NJ, 1997.

Brown, S. D., Information and Data Handling in Chemistry and Chemical Engineering: The State of the Field from the Perspective of Chemometrics, *Computers and Chem. Engr.*, **23**, 203 (1998).

Chapra, S. C., and R. P. Canale, *Numerical Methods for Engineers*, 7th ed, McGraw-Hill, New York, 2014.

Draper, N. R., and H. Smith, *Applied Regression Analysis*, 3rd ed., John Wiley and Sons, Hoboken, NJ, 1998.

Edgar, T. F., D. M. Himmeblau, and L. S. Lasdon, *Optimization of Chemical Processes*, 2nd ed., McGraw-Hill, New York, 2001.

Godfrey, K. (Ed.), *Perturbation Signals for System Identification*, Prentice-Hall, Englewood Cliffs, NJ (1993).

Haykin, S. S., *Neural Networks and Learning Machines*, 3rd ed., Prentice-Hall, Upper Saddle River, NJ, 2009.

Himmelblau, D. M., Accounts of Experience in the Application of Artificial Neural Networks in Chemical Engineering, *Ind. Eng. Chem. Res.,* **47**, 5782 (2008).

Hokanson, D. A., and J. G. Gerstle, Dynamic Matrix Control Multivariable Controllers, in *Practical Distillation Control*, W. L. Luyben (Ed.), Van Nostrand Reinhold, New York (1992), p. 248.

Kadlec, P., B. Gabrys, and S. Strandt, Data-driven Soft Sensors in the Process Industries, *Computers and Chem. Engr.*, **33**, 795 (2009).

Ljung, L., *System Identification: Theory for the User,* 2nd ed., Prentice-Hall, Upper Saddle River, NJ, 1999.

Ljung, L., *System Identification Toolbox for Use with Matlab. Version 7.* MathWorks, Inc, Natick, MA, 2007.

Ljung, L., and T. Glad, *Modeling of Dynamic Systems*, Prentice-Hall, Upper Saddle River, NJ, 1994.

Martin, G., Consider Soft Sensors, *Chem. Engr. Prog.*, **93**(7), 66 (1997).

Montgomery, D. C., and G. C. Runger, *Applied Statistics and Probability for Engineers*, 6th ed., John Wiley and Sons, Hoboken, NJ (2013).

Pearson, R. K., *Discrete-Time Dynamic Models*, Oxford University Press, New York, 1999.

Rangaiah, G. P., and P. R. Krishnaswamy, Estimating Second-Order plus Dead Time Model Parameters, *Ind. Eng. Chem. Res.*, **33**, 1867 (1994).

Rangaiah, G. P., and P. R. Krishnaswamy, Estimating Second-Order Dead Time Parameters from Underdamped Process Transients, *Chem. Engr. Sci.*, **51**, 1149 (1996).

Rivera, D. E., and K. S. Jun, An Integrated Identification and Control Design Methodology for Multivariable Process System Applications, *IEEE Control Systems Magazine*, **20**(3), 25 (2000).

Smith, C. L., *Digital Computer Process Control*, Intext, Scranton, PA, 1972.

Su, H. T., and T. J. McAvoy, Artificial Neural Networks for Nonlinear Process Identification and Control, Chapter 7 in *Nonlinear Process Control*, M. A. Henson and D. E. Seborg (Eds.), Prentice Hall, Upper Saddle River, NJ, 1997.

Sundaresan, K. R., and R. R. Krishnaswamy, Estimation of Time Delay, Time Constant Parameters in Time, Frequency, and Laplace Domains, *Can. J. Chem. Eng.*, **56**, 257 (1978).

EXERCISES

7.1  An operator introduces a step change in the flow rate q_i to a particular process at 3:05 A.M., changing the flow from 500 to 520 gal/min. The first change in the process temperature T (initially at 120 °F) occurs at 3:08 A.M. After that, the response in T is quite rapid, slowing down gradually until it appears to reach a steady-state value of 124.7 °F. The operator notes in the logbook that there is no change after 3:34 A.M. What approximate transfer function might be used to relate temperature to flow rate for this process in the absence of more accurate information? What should the operator do next time to obtain a better estimate?

7.2  A single-tank process has been operating for a long period of time with the inlet flow rate q_i equal to 30.1 ft^3/min. After the operator increases the flow rate suddenly at $t = 0$ by 10%, the liquid level in the tank changes as shown in Table E7.2.

Assuming that the process dynamics can be described by a first-order model, calculate the steady-state gain and the time constant using three methods:

(a) From the time required for the output to reach 63.2% of the total change.

(b) From the initial slope of the response curve.

(c) From the slope of the fraction incomplete response curve.

(d) Compare the data and the three models by simulating their step responses.

Table E7.2

t (min)	h (ft)	t (min)	h (ft)
0	5.50	1.4	6.37
0.2	5.75	1.6	6.40
0.4	5.93	1.8	6.43
0.6	6.07	2.0	6.45
0.8	6.18	3.0	6.50
1.0	6.26	4.0	6.51
1.2	6.32	5.0	6.52

7.3 A process consists of two stirred tanks with input q and outputs T_1 and T_2 (see Fig. E7.3). To test the hypothesis that the dynamics in each tank are basically first-order, a step change in q is made from 82 to 85 L/min, with output responses given in Table E7.3.

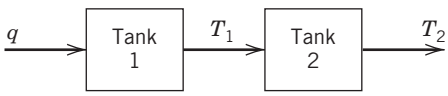

Figure E7.3

Table E7.3

t (min)	T_1 (°C)	T_2 (°C)	t (min)	T_1 (°C)	T_2 (°C)
0	10.00	20.00	11	17.80	25.77
1	12.27	20.65	12	17.85	25.84
2	13.89	21.79	13	17.89	25.88
3	15.06	22.83	14	17.92	25.92
4	15.89	23.68	15	17.95	25.94
5	16.49	24.32	16	17.96	25.96
6	16.91	24.79	17	17.97	25.97
7	17.22	25.13	18	17.98	25.98
8	17.44	25.38	19	17.99	25.98
9	17.60	25.55	20	17.99	25.99
10	17.71	25.68	50	18.00	26.00

(a) Find the transfer functions $T_1'(s)/Q'(s)$ and $T_2'(s)/T_1'(s)$ Assume that they are of the form $K_i/(\tau_i s + 1)$.

(b) Calculate the model responses to the same step change in q and plot with the experimental data.

7.4 For a multistage bioseparation process described by the transfer function,

$$G(s) = \frac{2}{(5s+1)(3s+1)(s+1)}$$

(a) Calculate the response to a step input change of magnitude, 1.5.

(b) Obtain an approximate first-order-plus-delay model using the fraction incomplete response method.

(c) Find an approximate second-order model using a method of Section 7.2.

(d) Calculate the responses of both approximate models using the same step input as for the third-order model. Plot all three responses on the same graph. What can you conclude concerning the approximations?

7.5 Assume that step response data obtained from an FOPTD system with $K = \tau = 1$ are available. Determine the accuracy of the FOPTD approximate model derived from these data using the Sundaresan and Krishnaswamy method, and consider three cases: $\theta/\tau = 0.1, 1.0, 10.0$. Plot the results and calculate the sum of squared errors.

7.6 For the unit step response shown in Fig. E7.6, estimate the following models using graphical methods:

(a) First-order plus time delay.

(b) Second-order using Smith's method and nonlinear regression.

Plot all three predicted model responses on the same graph.

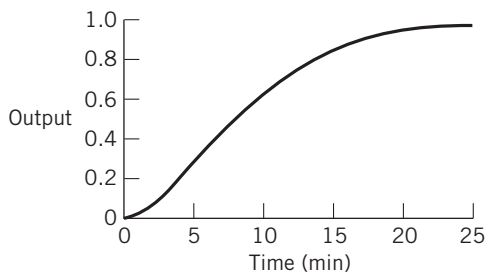

Figure E7.6

7.7 A heat exchanger used to heat a glycol solution with a hot oil is known to exhibit FOPTD behavior, $G_1(s) = T'(s)/Q'(s)$, where T' is the outlet temperature deviation and Q' is the hot oil flow rate deviation. A thermocouple is placed 3 m downstream from the outlet of the heat exchanger. The average velocity of the glycol in the outlet pipe is 0.5 m/s. The thermocouple also is known to exhibit first-order behavior; however, its time constant is expected to be considerably smaller than the heat exchanger time constant.

(a) Data from a unit step test in Q' on the complete system are shown in Fig. E7.7. Using a method of your choice, calculate the time constants of this process from the step response.

(b) From your empirical model, find transfer functions for the heat exchanger, pipe, and thermocouple. Think of the model as the product of three transfer functions: process, pipe flow, and sensor. What assumptions do you have to make to obtain these individual transfer functions from the overall transfer function?

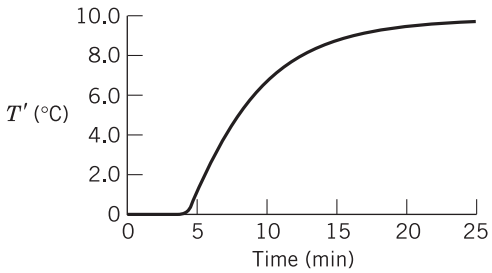

Figure E7.7

7.8 The level in a tank responds as a first-order system to changes in its inlet flow. The data shown below were gathered after the inlet flow was increased quickly from 1.5 to 4.8 gal/min.

(a) Determine the transfer function by estimating the time constant using one of the methods of Section 7.2. Be sure to use deviation variables and include units for the model parameters.

(b) Repeat part (a) using nonlinear regression (e.g., Excel) and the liquid level data.

(c) Graphically compare the two model responses with the data. Which model is superior? (Justify your answer)

Table E7.8

Time (min)	Level (ft)	Time (min)	Level (ft)
0.00	10.4	1.75	20.3
0.25	12.0	2.00	21.5
0.50	13.5	2.25	22.1
0.75	15.1	2.50	22.9
1.00	16.8	2.75	23.7
1.25	18.1	…	…
1.50	19.2	15.0	30.7 (steady state)

7.9 The output response data y shown in Table E7.9 were generated from a step change in input u from 1 to 5 at time $t = 0$. Develop a transfer function model of the form

$$\frac{Y(s)}{U(s)} = \frac{Ke^{-\theta s}}{(\tau_1 s + 1)(\tau_2 s + 1)}$$

Table E7.9

t	y	t	y
0	0	7	1.8
1	0	8	2.4
2	0	9	2.7
3	0.3	10	2.8
4	0.6	11	2.9
5	0.9	12	3.0
6	1.3	13	3.0

7.10 Noisy data for the step response of a boiler temperature T to a decrease in air flow rate q from 1000 to 950 cfm are shown below. Develop a FOPTD model using a method from Chapter 7. Be sure to use deviation variables and report units for the model parameters.

Table E7.10

t (min)	q (cfm)	T (°C)
0	1000	849
1	1000	851
2	1000	850
3	950	851
4	950	849
5	950	860
6	950	867
7	950	873
8	950	878
9	950	882
10	950	886
11	950	888
12	950	890
13	950	890

7.11 The response of a system to a unit step change in the input (occurring at time 0) is shown in Fig. E7.11.

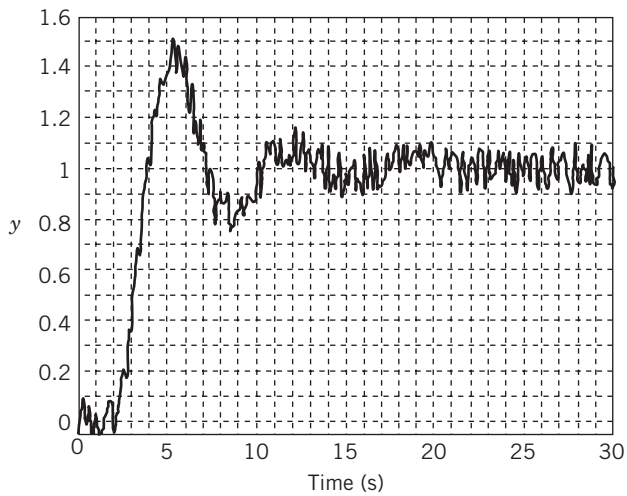

Figure E7.11

(a) Derive a second-order plus time delay model approximation for the system. Provide values for the gain, time constraints, and time delay of the SOPTD model.

(b) Why would a FOPTD model not be as accurate for fitting the data?

7.12 Fig. E7.12 presents the response of a system to a unit step in the input.

(a) Use these data to derive an FOPTD model of this system.

(b) Plot the response of the model and compare with the data.

(c) The FOPTD model does not capture all the features of the original response. What is the feature that the FOPTD model fails to capture? What causes this dynamic behavior?

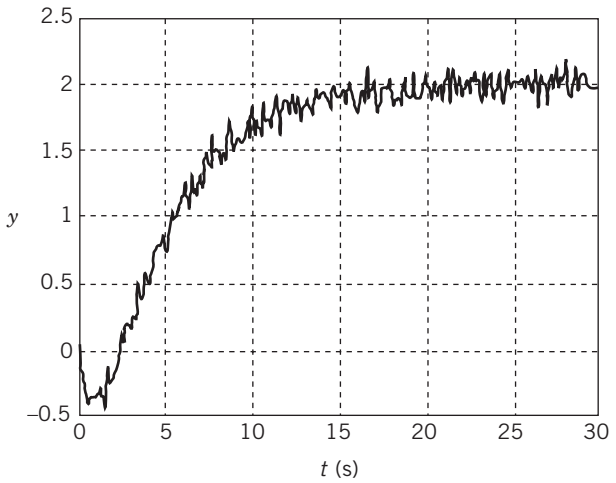

Figure E7.12

7.13 Consider the first-order differential equation

$$5\frac{dy}{dt} + y(t) = 6u(t) \quad y(0) = 3$$

where $u(t)$ is piecewise constant and has the following values:

$$\begin{array}{llll} u(0) = 0 & u(3) = 3 & & \\ u(1) = 1 & u(4) = 0 & & \\ u(2) = 2 & u(t) = 0 & \text{for} & t > 4 \end{array}$$

Derive a difference equation for this ordinary equation using $\Delta t = 1$ and

(a) Exact discretization

(b) Finite difference approximation

Compare the integrated results for $0 \le t \le 10$. Examine whether $\Delta t = 0.1$ improves the accuracy of finite difference model.

7.14 The following data were collected from a cell concentration sensor measuring absorbance in a biochemical stream. The input u is the flow rate deviation (in dimensionless units) and the sensor output y is given in volts. The flow rate (input) is piecewise constant between sampling instants. The process is not at steady state initially, so y can change even though $u = 0$.

Table E7.14

Time (s)	u	y
0	0	3.000
1	3	2.456
2	2	5.274
3	1	6.493
4	0	6.404
5	0	5.243
6	0	4.293
7	0	3.514
8	0	2.877
9	0	2.356
10	0	1.929

Fit a first-order model, $y(k) = a_1 y(k-1) + b_1 u(k-1)$, to the data using the least-squares approach. Plot the model response and the actual data. Can you also find a first-order continuous transfer function $G(s)$ to fit the data?

7.15 Unit step response data are given in Table E7.15 for a process with gain $K = 2$. Fit the data to a first-order model with no time delay. Next use linear regression to fit a first-order discrete-time equation to the data with $\Delta t = 1$. Then plot the predicted value of $y(t)$ for both models and compare with the data in Table E7.15. Calculate the gain of the discrete-time model and compare with the known value of $K = 1$. Why are the two values different?

Table E7.15

Time	$y(t)$
0	0.0
1	0.05
2	0.22
3	0.37
4	0.47
5	0.58
6	0.68
7	0.77
8	0.82
9	0.87
10	0.90

7.16 Data for a person with type 1 diabetes are available as both MATLAB and Excel data files on the book web site.[1] Glucose measurements (y) were recorded every five minutes using a wearable sensor that measures subcutaneous glucose concentration. The insulin infusion rate (u) from a wearable subcutaneous insulin pump was also recorded every five minutes. The data files consist of experimental data for two step changes in the insulin infusion rate. The data are reported as deviations from the initial values that are considered to be the nominal steady-state values.

It is proposed that the relationship between the glucose concentration y and the insulin infusion rate u can be described by a discrete-time, dynamic model of the form:

$$y(k) = a_1 y(k-1) + a_2 y(k-2) + b_1 u(k-1) + b_2 u(k-2)$$

(a) Use the least squares approach to estimate the model parameters from the *basal1* dataset. This data will be referred to as the *calibration data*. Graphically compare the model response and this data.

(b) In order to assess the accuracy of the model from part (a), calculate the model response $\hat{y}$ to the u step changes in the *validation data* (*basal2*). Then graphically compare the model response $\hat{y}$ *with* the validation data y.

[1] Book web site: www.wiley.com/college/seborg

(c) Repeat Steps (a) and (b) using an alternative transfer function model:

$$\frac{Y(s)}{U(s)} = \frac{K}{\tau s + 1}$$

Estimate the model parameters using graphical techniques and the *basal1* dataset. Then compare the model and experimental response data for both datasets.

(d) Which model is superior? Justify your answer by considering the least squares index for the one-step-ahead prediction errors,

$$S = \sum_{k=1}^{N} [y(k) - \hat{y}(k)]^2$$

where N is the number of data points.

7.17 Consider the PCM furnace module of Appendix E. Assume that hydrocarbon temperature T_{HC} is the output variable and that air flow rate F_A is the input variable.

(a) Develop an FOPTD model from response data for a step change in F_A at $t = 10$ min from 17.9 to 20.0 m^3/min. Summarize your calculated model parameters in a table and briefly describe the method used to calculate them.

(b) Repeat (a) for an SOPTD model.

(c) Plot the actual T_{HC} response and the two model responses for the F_A step change of part (a).

(d) Are the two models reasonably accurate? Which model is superior? Justify your answer by considering the least squares index for the prediction errors,

$$S = \sum_{k=1}^{N} [y(k) - \hat{y}(k)]^2$$

where N is the number of data points.

7.18 Consider the PCM distillation column module of Appendix E. Assume that distillate MeOH composition x_D is the output variable and that reflux ratio R is the input variable.

(a) Develop an FOPTD transfer function model from response data for a step change in R at $t = 10$ min from 1.75 to 2.0. Summarize your calculated model parameters in a table and briefly describe the method used to calculate them.

(b) Repeat (a) for an SOPTD model.

(c) Plot the actual x_D response and the two model responses for the R step change of part (a).

(d) Are the two models reasonably accurate? Which model is better? Justify your answer by considering the least squares index for the prediction errors,

$$S = \sum_{k=1}^{N} [y(k) - \hat{y}(k)]^2$$

where N is the number of data points.