INTERACTIVE CONTROL ENGINEERING
Principles and Applications

Volume 1 Process Dynamics, Modeling, and Identification

CHAPTER 1

Introduction: Process Models From Balances and Data

Pages 1–22

Edited by John D. Hedengren

How to cite this chapter. "Introduction: Process Models From Balances and Data," in *Process Dynamics, Modeling, and Identification*, John D. Hedengren, Ed., vol. 1 of *Interactive Control Engineering: Principles and Applications*. Advanced Process Solutions, LLC, 2026, ch. 1, pp. 1–22.

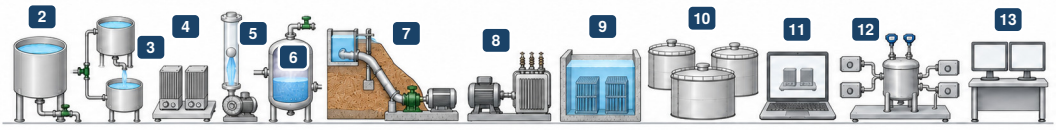
The page numbers in this file are those of the printed volume.

The volume online: contents, other chapters and citation records
https://apmonitor.com/apps/books/v01-dynamics-modeling/

Copyright © 2026 Advanced Process Solutions, LLC. All rights reserved. Produced by Advanced Process Solutions, LLC.

CHAPTER 1

Introduction: Process Models From Balances and Data



The chapters 2 draining tank 3 two tanks 4 two-heater board 5 levitated ball 6 flash steam vessel 7 hydro governor 8 grid frequency 9 spent-fuel pool 10 oil market stocks 11 heater model choice 12 identification workbench 13 monitoring workbench

Notice that the first four units are laboratory plants, the next five carry the same balances to a steam vessel, a hydroelectric unit, a grid, a fuel pool and a market, and the last three build and judge models from records. Illustration, not to scale.

IN PLAIN TERMS **How a Plant Answers a Push**

The main idea. A controller needs a model, a short account of how its plant answers a push. The model comes from counting what flows in, what flows out and what is stored, or from pushing the real plant and fitting a curve to the record.

The challenge. The answer changes as the plant moves, so a model that is right at one level can be wrong at another. Some plants answer late or start the wrong way, and a curve that fits one record can fail on the next.

Why it matters. The same few shapes describe tanks, heaters, turbines, power grids and the oil market, so reading them well pays almost everywhere.

THREE WAYS TO READ THIS CHAPTER

Explore	7 min	This page, Section 1.2 with its timeline and two History boxes, the Did you know? boxes in Sections 1.4.6, 1.4.8 and 1.4.9, Figure 1.11, and Key points in Section 1.8. No equations.
Study	40 min	The whole chapter in order, with each number of the tank example worked, then Exercises 1 to 10.
Practice	10 min	This page, Sections 1.4.8 and 1.4.9 with Table 1.1, Section 1.5 with Figures 1.6 and 1.9, and Section 1.8.

1.1 The Question This Volume Answers

Every controller in this series acts on a model of its plant. Sometimes the model is only the sign of a gain and a rough time scale, and sometimes it is a set of differential equations solved at every decision. A controller tuned on the wrong model is mistuned however good its algorithm. This volume answers the question that comes before control: how is a model of a dynamic process built, how is it reduced to the few numbers a controller needs, and how is it checked against data?

Two routes lead to a model, and most plants need both. The first writes the balances of mass, energy or momentum that the plant must obey and solves them. The second runs an experiment on the plant and fits a model to the record. A balance says which form the model should take and how its constants depend on the operating point. An experiment supplies the numbers that the balance leaves unknown and shows where the balance is wrong.

Twelve chapters follow the two routes, each on a plant with an app that runs in a browser. Chapter 2, Chapter 3, Chapter 4 and Chapter 5 are laboratory plants: a tank that drains under its own head, two tanks whose order changes with a valve, a board with two heaters and a ball floating in a tube of air. Chapter 6, Chapter 7, Chapter 8, Chapter 9 and Chapter 10 are larger plants in which the model carries the lesson: a vessel whose level rises while it loses water, a turbine whose power first falls when its gate opens, the frequency of a power grid, a fuel pool that holds its temperature only while

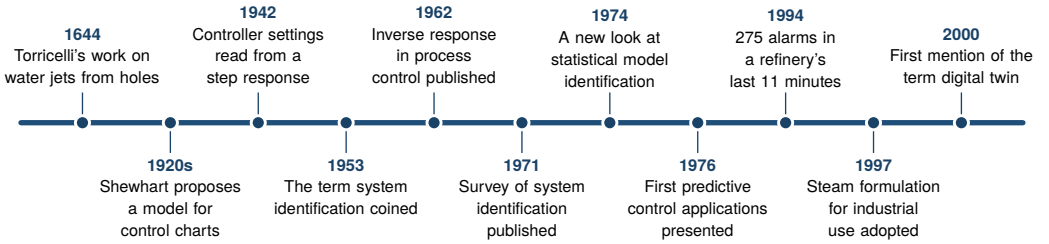


Figure 1.1 Milestones of dynamic modeling, identification and monitoring. The dates are from [4] (1644), [6] (1920s), [7] (1942), [8] (1953), [9] (1962), [10] (1971), [11] (1974), [12] (1976), [13] (1994), [5] (1997) and [14] (2000). Notice that the balance is centuries older than the fitted model, and that the checks on a model in service came last. Events are evenly spaced, not to a time scale.

its pumps run, and an oil market treated as one tank of stocks. Chapter 11, Chapter 12 and Chapter 13 build and judge models from records: choosing between energy balances, identifying a multivariable plant and diagnosing faults in plant data.

A reader who has worked through the volume can do the following on a plant that is new to the reader.

- Write the mass and energy balances of a process and find its steady state.
- Linearize the balances about an operating point in deviation variables, and say how the gain and the time constant move with that point.
- Recognize first-order, second-order, integrating, dead-time and inverse responses in a step test, and name the physical feature that produces each.
- Fit a model to a step test with a graphical rule and by least squares, in continuous and in discrete time.
- Judge a fitted model by its residuals and on data it was not fitted to, and say which of its parameters the data can determine.

This chapter develops each skill on one small example and names the chapters that use it. A reader new to process models should read Section 1.4 in order. It assumes calculus and the Laplace transform at the level of a first course. Standard texts give fuller derivations [1, 2], and a survey places the methods of identification in context [3]. Later volumes build on this one. Volume 2 tunes controllers from the models fitted here, and Volume 5 and Volume 7 return to dead time and to nonlinear models.

1.2 A Short History

The milestones of Figure 1.1 follow the two routes to a model, the balance and the test, and the monitoring that checks a model once it is in service. The balance is the older route. Torricelli's work of 1644 on water jets leaving holes in a vessel is one of the founding results of fluid mechanics [4], and the orifice of TK-1 drains by the same physics. A balance also needs the properties of what it stores. International conferences on steam tables met in 1929, 1930 and 1934, equations replaced the tables as computers spread through industry, and a formulation for industrial use was adopted in 1997 [5]. Chapter 6 closes its balances with that formulation.

The test is younger. Tuning rules published in 1942 read two numbers from a step response [2, 7], and the term system identification dates from 1953 [8]. A paper on inverse response in process control appeared in 1962 [9], the two search methods of Section 1.4.8 in 1963 and 1965 [15, 16], a survey of system identification in 1971 [10] and the information criterion of Equation (1.23) in 1974 [11]. The first industrial applications of predictive control, presented in 1976, identified an impulse response model from plant test data [12], the model family of Chapter 12.

> ⧖ HISTORY **1942: Two Numbers From a Bump Test**
>
> The first tuning rules, from Ziegler and Nichols, rested on one simple experiment. The process was brought to steady state, its input was changed by a suitable amount, and the output was recorded and scaled to a unit step. Two numbers were read from the record, the intercepts of its steepest tangent with the axes: an apparent time delay and, with it, the steepest slope. The test could stop once the response had passed its inflection point. The rules had a huge impact and makers of controllers adopted them for routine use, but they used too little process information [2, 7]. The model of Equation (1.15) adds the gain, which needs the final value, and a time constant read at 63 % of the change, as Equation (1.18) does.

Monitoring asks whether a model still holds. During the 1920s Shewhart proposed the control chart, with limits a fixed number of standard deviations from the mean [6]. Before a refinery explosion of 1994, two operators had to recognize, acknowledge and act on 275 alarms in the last 11 minutes [13], the problem behind the alarm tools of Chapter 13. The term digital twin is traced back only to 2000 [14], and Chapter 4 runs such a twin beside its board.

> ⧖ HISTORY **1940s to 1970s: Two Ways to Write Down a Plant**
>
> When control became a discipline in the 1940s, it saw dynamics through the electrical engineer's input and output view, in which a system is described by its recorded response to a step or to sinusoids. A second wave from the late 1950s took the state space of mechanics, with space flight as a typical example, and the two views merged into the state-space model of an input-output system [2]. The term system identification was coined in 1953. The field then followed two routes, a statistical one built on parameter estimation such as maximum likelihood, and a realization one that builds a linear system from input-output descriptions such as impulse responses [8]. Equation (1.7) is the state-space form, and Section 1.4.8 takes the statistical route.

1.3 Notation and Conventions

The chapters share the symbols of the notation page. The setpoint is SP, the process variable is PV and the manipulated variable is MV. A model of one input and one output has a gain K_p, a time constant τ_p and a dead time θ_p, in PV units per MV unit and in the time unit of the plant. A prime marks a deviation from the operating point, so $L' = L - \bar{L}$, and a bar marks the operating point itself. A hat marks a value fitted or predicted by a model, so $\hat{y}_k$ is the model's output at sample k. Samples are taken every T_s, and a dead time of d samples is dT_s. The difference $\varepsilon_k = y_k - \hat{y}_k$ between a measurement and a model is a residual. Time in the running example is in minutes.

Several letters need care across the series. The shared framework uses h for the integration step, a for the controller action and e for the control error, so this chapter writes the tank level as L, the coefficients of a discrete model as ϕ and γ and a residual as ε. The chapters keep their own notation. Chapter 2 and Chapter 3 write the level as h, and Chapter 2 writes the process gain as K_u and the disturbance gain as K_d. Chapter 5 writes the integrating gain as K_i, Chapter 9 writes it as K_I and the pool level as z, and Chapter 8 uses K_I for the gain of its integral restoring controller. Chapter 12 writes the coefficients of its discrete models as a and b and its dead times in samples as d_{ij}.

1.4 From Balances to Models That Fit Data

1.4.1 The Running Example

The running example is a heated water tank TK-1 that drains under its own head, shown in Figure 1.2. Water enters at the flow F, in $\mathrm{m^3/min}$, and at $T_{in} = 20\,^{\circ}\mathrm{C}$. It leaves through an orifice at the bottom at the rate $\beta\sqrt{L}$, where L is the level in meters and $\beta = 0.2\,\mathrm{m}^{2.5}/\mathrm{min}$ lumps the orifice area, its discharge coefficient and gravity. The tank has a cross-section of $A_1 = 1\,\mathrm{m}^2$. An electric heater supplies W percent of 280 kW. The outlet water passes through a pipe that holds $0.2\,\mathrm{m}^3$ to a thermometer TT-1 in a thermowell. The level transmitter LT-1 reads L directly. At the operating point the feed is $\bar{F} = 0.2\,\mathrm{m}^3/\mathrm{min}$, the level $\bar{L} = 1\,\mathrm{m}$, the heater $\bar{W} = 50\,\%$ and the temperature $\bar{T} = 30\,^{\circ}\mathrm{C}$. Two variants

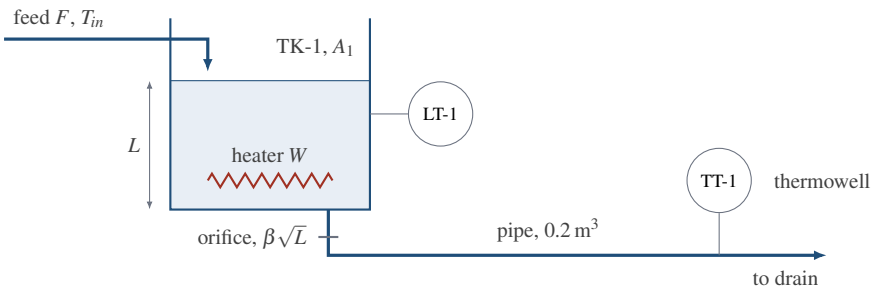


Figure 1.2 The Running Example. The Tank TK-1 Is Fed at F and Drains Through an Orifice. A Heater Warms the Water, and the Thermometer TT-1 Reads the Outlet at the End of a Pipe, Inside a Thermowell.

appear later: a pump in place of the orifice in Section 1.4.4, and a second tank under the first in Section 1.4.5.

1.4.2 Balances Become Differential Equations

Every model in this volume starts from one statement, written for a chosen quantity and a chosen boundary, as shown in Equation (1.1).

$$\text{rate of accumulation} = \text{rate in} - \text{rate out} + \text{rate generated} \tag{1.1}$$

The quantity may be mass, energy, momentum or a number of barrels, and the boundary may be a tank, a heater or a market. When the stored amount is written in terms of a measurable variable, the statement becomes a differential equation in that variable.

For TK-1 the stored mass is $\rho A_1 L$ at constant density ρ, so the mass balance is Equation (1.2).

$$A_1 \frac{\mathrm{d}L}{\mathrm{d}t} = F - \beta\sqrt{L} \tag{1.2}$$

The outflow term is the square-root law of an orifice under a head L. The stored energy is $\rho c_p A_1 L T$ relative to a reference temperature, where c_p is the heat capacity. The water brings in $\rho c_p F T_{in}$, carries out $\rho c_p \beta\sqrt{L} T$ and gains the heater power q_W in kJ/min. Expanding the derivative of the product LT and subtracting T times Equation (1.2) gives the energy balance of Equation (1.3).

$$A_1 L \frac{\mathrm{d}T}{\mathrm{d}t} = F\,(T_{in} - T) + \frac{q_W}{\rho c_p} \tag{1.3}$$

The model rests on four assumptions: the tank is well mixed, the density and heat capacity are constant, the walls lose no heat and the outlet carries the tank's temperature. With $\rho c_p = 4200\,\mathrm{kJ/(m^3\,K)}$ the flow $\bar{F}$ carries 14 kW for each kelvin of warming, so each percent of heater power, 2.8 kW, raises the outlet by 0.2 °C.

A steady state sets every derivative to zero. Equation (1.2) then gives $\bar{L} = (\bar{F}/\beta)^2 = 1\,\mathrm{m}$, and Equation (1.3) gives $\bar{T} = T_{in} + 0.2\bar{W} = 30\,^\circ\mathrm{C}$. The steady state answers where the plant settles. The derivatives answer how fast it gets there, which is the subject of the rest of this section.

Each plant of the volume is an inventory written this way. Chapter 2 and Chapter 3 write the mass balance of Equation (1.2) for one and two tanks. Chapter 4 writes an energy balance for each heater and each sensor, and Chapter 11 writes three energy balances of increasing detail for one heater. Chapter 6 and Chapter 9 write mass and energy balances together, closed by steam tables in one case and by decay heat and evaporation in the other. Chapter 8 writes the swing equation, an energy balance on the rotating machines whose stored energy is read as frequency. Chapter 7 writes a momentum balance on the water in a penstock, and Chapter 5 one on a floating ball. Chapter 10 writes a balance on commercial oil stocks, filled by supply and drained by demand.

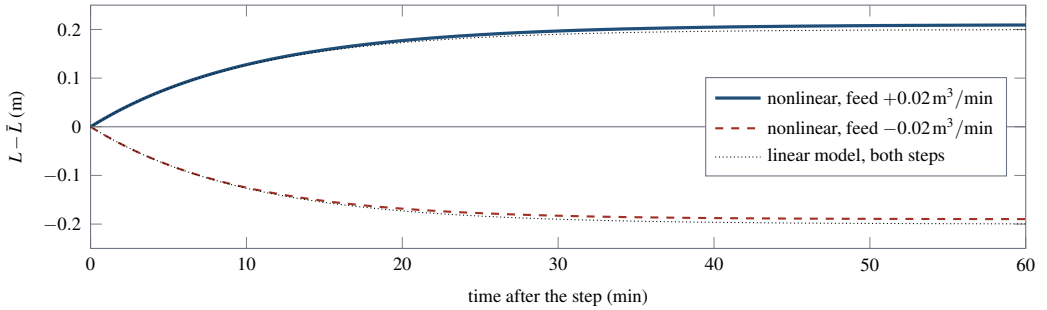


Figure 1.3 Level of TK-1 After Feed Steps of $\pm 0.02\,\mathrm{m}^3/\mathrm{min}$. Solid: Nonlinear Balance, Step Up. Dashed: Nonlinear Balance, Step Down. Dotted: the Linear Model of Equation (1.5), Which Predicts $\pm 0.20\,\mathrm{m}$.

1.4.3 Linearization and Deviation Variables

Equation (1.2) is nonlinear because of the square root, and Equation (1.3) because of the products of L with $\mathrm{d}T/\mathrm{d}t$ and of F with T. A model valid near the operating point follows from a first-order Taylor expansion. For a state equation $\dot{x} = f(x,u)$ that is at rest at $(\bar{x},\bar{u})$, the expansion is Equation (1.4).

$$\frac{\mathrm{d}x'}{\mathrm{d}t} \approx \left.\frac{\partial f}{\partial x}\right|_{\bar{x},\bar{u}} x' + \left.\frac{\partial f}{\partial u}\right|_{\bar{x},\bar{u}} u', \qquad x' = x - \bar{x}, \quad u' = u - \bar{u} \tag{1.4}$$

The constant term $f(\bar{x},\bar{u})$ is zero at a steady state, which is why deviation variables remove it. Applied to Equation (1.2), the slope of $\beta\sqrt{L}$ is $\beta/(2\sqrt{\bar{L}}) = \bar{F}/(2\bar{L})$, and the linear model is Equation (1.5).

$$\tau_L \frac{\mathrm{d}L'}{\mathrm{d}t} = -L' + K_L F', \qquad \tau_L = \frac{2A_1\bar{L}}{\bar{F}}, \qquad K_L = \frac{2\bar{L}}{\bar{F}} \tag{1.5}$$

At the operating point $\tau_L = 10\,\mathrm{min}$ and $K_L = 10\,\mathrm{m}$ per $\mathrm{m}^3/\mathrm{min}$. Both constants depend on where the tank operates. At $\bar{L} = 0.25\,\mathrm{m}$ they are 5 and 5, and at 4 m they are 20 and 20. Their ratio $K_L/\tau_L = 1/A_1$ does not change, because the first slope of a level response is the inflow change divided by the area, whatever the head. Chapter 2 derives the same model for its tank and finds that the loop gain of a fixed controller grows 3.46 times from the bottom of the tank to the top.

How far does the linear model reach? Figure 1.3 steps the feed by $\pm 0.02\,\mathrm{m}^3/\mathrm{min}$, a tenth of $\bar{F}$. The linear model predicts $\pm 0.20\,\mathrm{m}$. The nonlinear balance settles 0.21 m higher after the rise and 0.19 m lower after the fall, and it takes 10.69 min and 9.32 min to cover 63.2 % of the change against the 10 min of τ_L. The tank slows as it fills. A step test therefore returns the gain and time constant of the range it crosses, not of its starting point. Chapter 3 finds a fitted gain 14 % above the linearized one for this reason.

The energy balance has two inputs. At a steady state $\mathrm{d}T/\mathrm{d}t = 0$, so the factor L on the left of Equation (1.3) enters its linearization only as $\bar{L}$, and the result is Equation (1.6).

$$\tau_T \frac{\mathrm{d}T'}{\mathrm{d}t} = -T' + K_W W' + K_F F', \qquad \tau_T = \frac{A_1\bar{L}}{\bar{F}}, \quad K_F = -\frac{\bar{T} - T_{in}}{\bar{F}} \tag{1.6}$$

with $\tau_T = 5\,\mathrm{min}$, $K_W = 0.2\,^\circ\mathrm{C}/\%$ and $K_F = -50\,^\circ\mathrm{C}$ per $\mathrm{m}^3/\mathrm{min}$. The time constant is the residence time of the water, and more feed cools the tank. A plant with several states is linearized the same way, one partial derivative per pair, which gives the state-space model and its transfer function of Equation (1.7).

$$\frac{\mathrm{d}x'}{\mathrm{d}t} = \mathbf{A}x' + \mathbf{B}u', \qquad y' = \mathbf{C}x', \qquad G(s) = \frac{Y'(s)}{U'(s)} = \mathbf{C}\,(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} \tag{1.7}$$

where $\mathbf{A} = \partial f/\partial x$ and $\mathbf{B} = \partial f/\partial u$ at the operating point. The eigenvalues of $\mathbf{A}$ are the poles of G, and each real pole $-1/\tau$ contributes one time constant. For the level, $G(s) = K_L/(\tau_L s + 1)$.

Chapter 3 writes Equation (1.7) for two tanks and reads both poles off the diagonal. Chapter 5 linearizes its ball about the hover angle, and Chapter 7 linearizes the water column to find a zero that limits the governor. Chapter 10 computes the four eigenvalues of its market and finds that they differ on each side of the balance, so one linearization holds on one branch at a time. A gain that moves with the operating point is the reason for gain scheduling in Volume 2 and for the nonlinear methods of Volume 7.

1.4.4 First-Order and Integrating Responses

After a step Δu at $t = 0$, the model $\tau_p \dot{y}' = -y' + K_p u'$ responds as Equation (1.8).

$$y'(t) = K_p \Delta u \left(1 - \mathrm{e}^{-t/\tau_p}\right) \tag{1.8}$$

The response covers 63.2 % of its change at $t = \tau_p$ and 95 % at $3\tau_p$. It starts at its steepest, with the slope $K_p \Delta u/\tau_p$, and has no inflection. TK-1 after a feed step of $0.02\,\mathrm{m}^3$/min starts to rise at 0.02 m/min. A plant of this kind is self-regulating: its own outflow grows with the level until it matches the new feed.

Replace the orifice with a pump that draws a fixed F_p, and the outflow no longer depends on the level. Equation (1.2) becomes Equation (1.9).

$$A_1 \frac{\mathrm{d}L}{\mathrm{d}t} = F - F_p, \qquad G(s) = \frac{K'}{s}, \qquad K' = \frac{1}{A_1} \tag{1.9}$$

The tank is now an integrating process. The same feed step makes the level rise at 0.02 m/min until it overflows or an operator acts, and no steady state exists unless $F = F_p$. The gain of an integrating process is a slope, in PV units per unit time per MV unit, and its step test has no final value to read. The integrating gain $K' = 1/A_1$ is the ratio K_L/τ_L of the self-regulating tank, so the integrator is the limit of a first-order lag whose time constant grows without bound.

Over a short test the two are hard to tell apart. One minute after the step TK-1 has risen 0.0190 m against the integrator's 0.0200 m, 4.8 % less. After five minutes the gap is 21 %. Chapter 9 makes the point on a full plant. Its pool temperature has a time constant of 3.3 h while the cooling trains run and 339 h when they stop, and a test read over its first hour returns an integrating gain that the chapter calls a correct measurement of the wrong model.

Integrating inventories run through the volume. Chapter 2 closes the outlet valve of its tank and obtains Equation (1.9) exactly. The pool level of Chapter 9, the water inventory of Chapter 6, the height of the ball in Chapter 5 and the stocks of Chapter 10 all integrate a flow imbalance. In Chapter 8 the frequency integrates the power imbalance, held back only by load damping, so a finite droop gain leaves an offset that only integral control removes.

1.4.5 Second-Order Responses

Let TK-1 drain freely into a second tank TK-2 with $A_2 = 0.5\,\mathrm{m}^2$ and the same orifice, also at 1 m. The outflow of TK-1 does not depend on the level of TK-2, so the tanks are non-interacting and the transfer function from the feed to the lower level is the product of two first-order lags, Equation (1.10).

$$\frac{L_2'(s)}{F'(s)} = \frac{K_L}{(\tau_1 s + 1)(\tau_2 s + 1)}, \qquad \tau_1 = 10\,\mathrm{min}, \quad \tau_2 = 5\,\mathrm{min} \tag{1.10}$$

The response now starts with zero slope and has an inflection, the mark of more than one lag. Any second-order model can be written in the standard form of Equation (1.11).

$$G(s) = \frac{K_p}{\tau_n^2 s^2 + 2\zeta\tau_n s + 1}, \qquad \tau_n = \sqrt{\tau_1 \tau_2}, \quad \zeta = \frac{\tau_1 + \tau_2}{2\sqrt{\tau_1 \tau_2}} \tag{1.11}$$

where τ_n is the natural period and ζ the damping factor. The expressions on the right hold for two real lags. For the two tanks $\tau_n = 7.07\,\text{min}$ and $\zeta = 1.06$. Two real lags always give $\zeta \geq 1$, so tanks in series never overshoot.

A response that rings needs $\zeta < 1$, which takes two stores that trade their contents, such as a mass on a spring or rotating inertia against a restoring force. The overshoot of a step response is then Equation (1.12).

$$\text{overshoot} = \exp\left(-\frac{\pi\zeta}{\sqrt{1-\zeta^2}}\right), \qquad \text{period} = \frac{2\pi\tau_n}{\sqrt{1-\zeta^2}} \tag{1.12}$$

With $\zeta = 0.3$ and the same τ_n, the overshoot is 37.2 % and the period 46.6 min, as panel (b) of Figure 1.4 shows.

Chapter 3 is Equation (1.10) with a splitter that sends part of the feed past the upper tank. The splitter scales the upper time constant by the share of flow it still carries and adds a zero, so the same vessels range from second order to first order. Chapter 4 has four lags, two heaters and two sensors, and the sensor lag of 140 s dominates its step test. Chapter 8 has three lags in its primary loop and turns oscillatory as the droop gain rises. Chapter 10 finds a damped pair of eigenvalues near $-3.85 \pm 3.47\,j$ per year beside two slow modes, and Chapter 12 offers the second-order family with $\zeta < 1$ for a record that rings.

1.4.6 Dead Time and Lags That Look Like It

The water seen by TT-1 left the tank one pipe volume earlier. With plug flow the measurement is the tank temperature delayed by the transport time of Equation (1.13).

$$T_m(t) = T(t-\theta), \qquad \theta = \frac{V_p}{\bar{F}} = 1\,\text{min}, \qquad \mathscr{L}\{T'(t-\theta)\} = \mathrm{e}^{-\theta s}\,T'(s) \tag{1.13}$$

where V_p is the pipe volume. A dead time is an interval during which the output shows nothing of a change already made. At half the feed it doubles to 2 min, a variation that Volume 5 treats. The thermowell adds a lag of $\tau_s = 1\,\text{min}$, so the heater reaches the reading through the model of Equation (1.14).

$$\frac{T_m'(s)}{W'(s)} = \frac{K_W\,\mathrm{e}^{-\theta s}}{(\tau_T s+1)(\tau_s s+1)} \tag{1.14}$$

This is a second-order-plus-dead-time (SOPDT) model. Most tuning rules want the simpler first-order-plus-dead-time (FOPDT) model of Equation (1.15).

$$G(s) = \frac{K_p\,\mathrm{e}^{-\theta_p s}}{\tau_p s+1} \tag{1.15}$$

A small lag in series with a large one delays the response much as a dead time does, so the reduction folds the small lags into θ_p. The half rule [17] gives the largest lag half of the second largest and the dead time the other half, as shown in Equation (1.16).

$$\tau_p = \tau_1 + \frac{\tau_2}{2}, \qquad \theta_p = \theta + \frac{\tau_2}{2} + \sum_{i\geq 3}\tau_i \tag{1.16}$$

For the heater path the rule gives $\tau_p = 5.5\,\text{min}$ and $\theta_p = 1.5\,\text{min}$. For the two tanks it gives 12.5 and 2.5 min, while the two-point fit of Section 1.4.8 on their exact response gives 12.4 and 3.5 min. An effective dead time depends on the method that reads it.

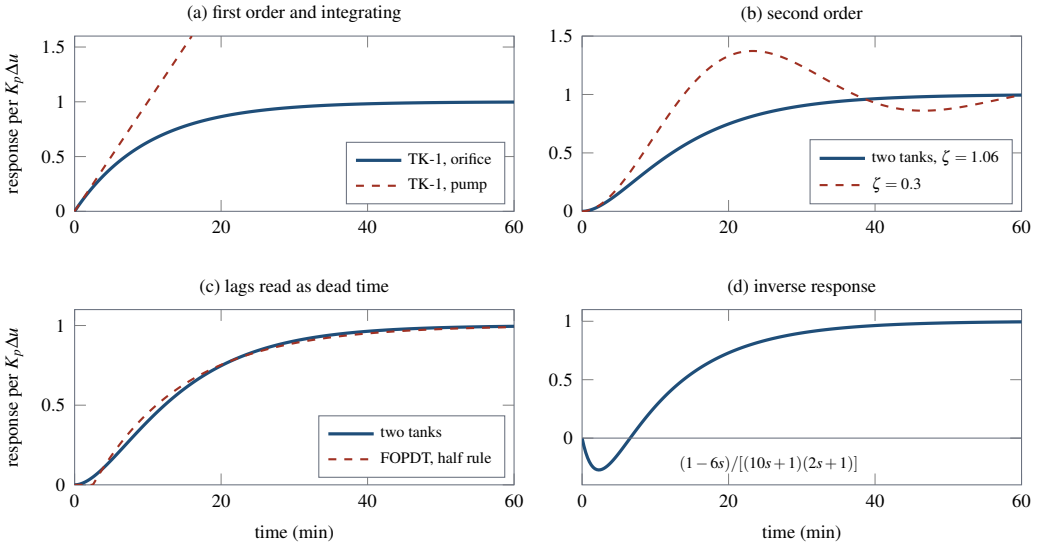


Figure 1.4 Response Shapes of the Running Example, Scaled by $K_p\Delta u$. (a) TK-1 With Its Orifice (Solid) and With a Pump on the Outlet (Dashed), Whose Ramp Reaches 1 at $t = \tau_L$. (b) Two Tanks in Series (Solid) and an Underdamped Plant With the Same τ_n (Dashed). (c) Two Tanks (Solid) and Their FOPDT by the Half Rule (Dashed). (d) Two Opposing Paths With an Inverse Response.

> **DID YOU KNOW? One Delay in Nerves, Networks and Pipes**
>
> A standard text names nerve propagation, communication systems and mass transport as typical sources of time delay, each with the transfer function $e^{-s\tau}$ [2]. The outlet pipe of TK-1 is mass transport, and the router of Volume 2, Chapter 7 carries a dead time equal to the round trip of its path.

Effective dead times appear in most chapters. In Chapter 4 the fitted FOPDT has $\tau_p = 141\,\text{s}$, the sensor lag, and $\theta_p = 18\,\text{s}$, the heater dynamics folded into a delay. Chapter 9 reads an apparent dead time of 0.16 h from a step whose transport delay is 0.06 h, and accounts for the rest as two small lags. The apparent dead time of Chapter 3 falls from 18.5 s to below zero as the splitter opens and the second lag leaves. Chapter 5 lumps a sample period, a servo lag and a flow lag into one dead time of 88 ms.

1.4.7 Inverse Response

TK-1 has no input that moves its level or its temperature the wrong way first. A plant does when one input acts through two paths of opposite sign and different speed, as in Equation (1.17).

$$G(s) = \frac{K_1}{\tau_1 s + 1} - \frac{K_2}{\tau_2 s + 1} = \frac{(K_1 - K_2) + (K_1\tau_2 - K_2\tau_1)\,s}{(\tau_1 s + 1)(\tau_2 s + 1)} \tag{1.17}$$

The final gain is $K_1 - K_2$, set by the slow path. The first slope is $K_1/\tau_1 - K_2/\tau_2$, set mostly by the fast path. When the fast path's initial slope is the larger, the response starts in the wrong direction and the numerator has a zero in the right half of the plane [9]. With the time scales of TK-1, $K_1 = 2$, $\tau_1 = 10\,\text{min}$, $K_2 = 1$ and $\tau_2 = 2\,\text{min}$ give $G = (1-6s)/[(10s+1)(2s+1)]$. Its step response falls to -0.27 at 2.29 min, crosses zero at 6.56 min and settles at $+1$, as panel (d) of Figure 1.4 shows.

Chapter 7 has the classical case. Opening the gate of a water turbine admits more flow, but the water in the penstock must first be accelerated by head taken from the turbine. The ideal model is $(1 - T_w s)/(1 + T_w s/2)$, where T_w is the water starting time. It equals $-2 + 3/(1 + T_w s/2)$, an instant

path of gain -2 in parallel with a lag of gain $+3$, so the power first moves twice as far the wrong way as it moves in the end. On the chapter's plant the first response is 1.71 to 1.94 times the final one, and the ratio of the water starting time to the mechanical starting time caps the governor's gain. Chapter 6 has the other classical case. A rise in steam demand lowers the pressure, the water boils and its bubbles lift the surface three points while the vessel loses water. A zero in the right half-plane limits how fast any controller can act, a limit that Volume 7 develops.

1.4.8 Step Tests and Regression

A balance gives the form of a model. Its numbers often come from a test. The plant is held steady in Manual, the MV is stepped and the PV is recorded until it settles. The step must stand clear of the noise, stay small enough for the linear model to hold and last long enough to settle, which for the pool of Chapter 9 means most of a day.

> **+ DID YOU KNOW? A Plant Test of 5 to 15 Days**
>
> In a survey of industrial predictive control, a formal plant test runs 24 hours a day with engineers watching. Each manipulated variable is stepped 8 to 15 times, with a signal-to-noise ratio of at least six, and the test may take up to 5 to 15 days [12].

TK-1's heater test steps W from 50 to 60 % at $t = 2\,\text{min}$. TT-1 is sampled every $T_s = 0.2\,\text{min}$ for 40 min, 201 samples, with measurement noise of standard deviation 0.03 °C. The plant is Equation (1.14) exactly, so every fitted model can be compared with the truth.

The two-point method reads a FOPDT model from two times, as shown in Equation (1.18).

$$K_p = \frac{\Delta y}{\Delta u}, \qquad \tau_p = 1.5\,(t_{63} - t_{28}), \qquad \theta_p = t_{63} - \tau_p \tag{1.18}$$

where t_{28} and t_{63} are the times after the step at which the PV has covered 28.3 and 63.2 % of its change. A first-order response reaches those levels $\tau_p/3$ and τ_p after it starts to move. On the record, with the final value taken as the mean of the last five minutes, $K_p = 0.200\,^\circ\text{C}/\%$, $t_{28} = 3.71\,\text{min}$ and $t_{63} = 7.11\,\text{min}$, so $\tau_p = 5.11\,\text{min}$ and $\theta_p = 2.00\,\text{min}$. The method uses two samples of 201, and the noise on those two moves the result.

Least squares uses every sample. The parameters ϑ of a model are chosen to minimize the sum of squared residuals of Equation (1.19) [10].

$$J(\vartheta) = \sum_{k=1}^{n} \varepsilon_k^2, \qquad \varepsilon_k = y_k - \hat{y}_k(\vartheta) \tag{1.19}$$

where n is the number of samples and $\hat{y}_k$ is the model's output computed from the input record alone. The FOPDT model fitted this way has $K_p = 0.201$, $\tau_p = 5.18\,\text{min}$ and $\theta_p = 1.86\,\text{min}$. The SOPDT model has $K_p = 0.200$, lags of 4.76 and 1.20 min and a dead time of 0.94 min, close to the true 5, 1 and 1 min. Both models are nonlinear in their time constants and dead time, so J is minimized by an iterative search. Chapter 11 uses the Nelder–Mead simplex method [16] and minimizes the sum of absolute errors in place of squares, and Chapter 12 uses a Levenberg–Marquardt search [15].

A sampled model avoids the search. When the input is held between samples, a first-order model with a dead time of d samples is exactly the difference equation of Equation (1.20).

$$y'_k = \phi\, y'_{k-1} + \gamma u'_{k-1-d}, \qquad \phi = \mathrm{e}^{-T_s/\tau_p}, \qquad \gamma = K_p\,(1 - \phi) \tag{1.20}$$

With the measured y'_{k-1} on the right, ϕ and γ enter linearly, and one linear least-squares solve gives both. This is the simplest autoregressive model with an exogenous input (ARX). The continuous constants follow from Equation (1.21).

$$\tau_p = -\frac{T_s}{\ln \phi}, \qquad K_p = \frac{\gamma}{1 - \phi} \tag{1.21}$$

Table 1.1 Models Fitted to the Heater Step Test of TK-1. Errors Are Root Mean Square Residuals of a Simulation From the Input, in °C. The Noise Has a Standard Deviation of 0.030 °C.

Model	Parameters (K_p in °C/%, times in min)	Training	Validation	κ_1	Runs
FOPDT, two-point	0.200, τ_p 5.11, θ_p 2.00	0.0372	0.0484	0.217	85
FOPDT, least squares	0.201, τ_p 5.18, θ_p 1.86	0.0340	0.0429	0.191	77
SOPDT, least squares	0.200, lags 4.76 and 1.20, θ_p 0.94	0.0311	0.0331	0.035	105
ARX, $d = 9$	0.199, τ_p 5.04, θ_p 1.8	0.0376	0.0508	0.306	72
True model	0.2, lags 5 and 1, θ_p 1	0.0319	0.0317	0.061	89

Scanning d and solving at each gives $d = 9$ samples, or 1.8 min, with $\tau_p = 5.04$ min and $K_p = 0.199$. The ARX fit minimizes the error of a one-step prediction made from the last measurement, which is a different objective from the error of a simulation run from the input alone. Table 1.1 shows the cost: run as a simulation, this model has the largest validation error of the four. Chapter 12 fits ARX models by least squares and then refines them on the simulation error for this reason. It also fits continuous transfer functions to the same sampled records [18].

1.4.9 Model Structure, Validation and Residuals

Table 1.1 compares the fitted models on two records. The training record is the step test. The validation record is a second test with fresh noise, in which W steps from 50 to 60, then to 45 and then to 55 %, at 2, 14 and 26 min. Each model is fitted once to the training record and then run unchanged on the validation record.

The training errors differ little, from 0.0311 to 0.0376 °C against a noise of 0.030. A small training error says only that the model can follow the record it was fitted to, and any model with enough parameters can. The validation column separates the models more clearly. The SOPDT model keeps its error at 0.0331 °C, near the true model's 0.0317, while the FOPDT fitted by least squares rises to 0.0429 °C on the new steps.

The residuals show why. If a model has the right structure, its residuals are the measurement noise, which is independent from sample to sample. If it does not, the residuals carry the part of the response the model cannot follow, which changes slowly. Figure 1.5 plots both sets with their one-minute moving mean. The FOPDT residuals run positive at the start of the response, with a mean of +0.030 °C from 2 to 4 min, negative at the shoulder, −0.024 from 6 to 9 min, and positive again, +0.016 from 14 to 20 min. A single lag cannot bend the way two lags do. The SOPDT residuals stay within 0.01 °C of zero over the same windows.

Two numbers test the pattern without a plot. The lag-one autocorrelation of the residuals, Equation (1.22), is near zero for independent noise.

$$\kappa_1 = \frac{\sum_{k=2}^{n} \varepsilon_k \, \varepsilon_{k-1}}{\sum_{k=1}^{n} \varepsilon_k^2} \tag{1.22}$$

For $n = 201$, noise alone keeps κ_1 within $\pm 2/\sqrt{n} = \pm 0.141$ about 19 times in 20. The FOPDT gives 0.191 and the ARX model 0.306, outside the band, and the SOPDT gives 0.035, inside it. The second number counts runs, the stretches of residuals of one sign. Independent noise gives about $(n+1)/2 = 101$ runs with a standard deviation of about 7. The FOPDT residuals form 77 and the SOPDT residuals 105. The Ljung–Box portmanteau test extends the first check to many lags at once [19].

More parameters always lower the training error, so the comparison must charge for them. The Akaike information criterion [11], for residuals that are independent and normal, is Equation (1.23).

$$\mathrm{AIC} = n \ln \frac{J}{n} + 2m \tag{1.23}$$

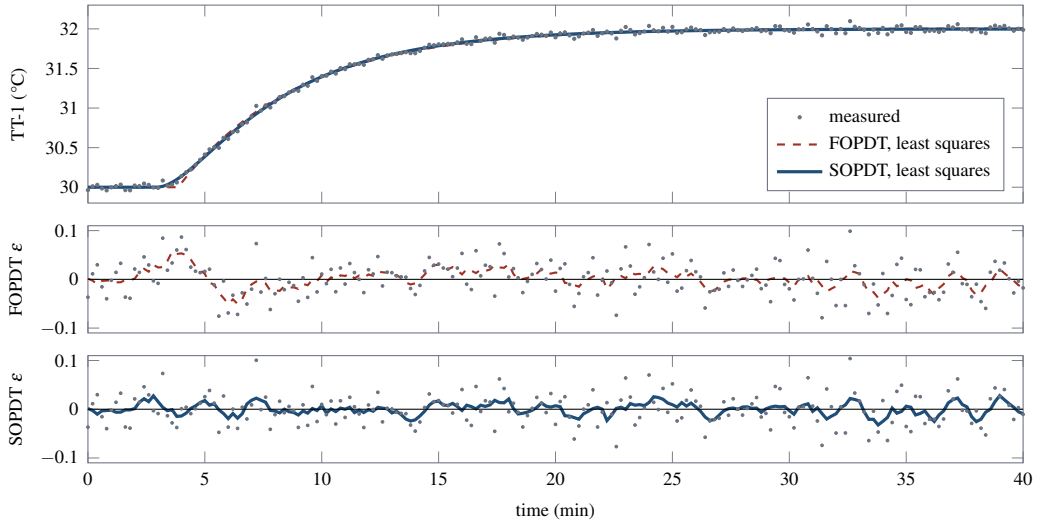


Figure 1.5 Heater Step Test on TK-1 (W From 50 to 60 % at 2 min) With Two Least-Squares Fits. Top: Measurements (Dots), FOPDT (Dashed) and SOPDT (Solid). Middle and Bottom: Residuals in °C (Dots) and Their One-Minute Moving Mean (Dashed for the FOPDT, Solid for the SOPDT).

where m is the number of fitted parameters. The lower value wins. The FOPDT, with $m = 3$, scores -1352.9, and the SOPDT, with $m = 4$, scores -1387.0. The fourth parameter more than pays for itself. A criterion is a guide and not a verdict. Validation on data the fit has not seen is the stronger test, and Chapter 12 builds its workflow around it, with chronological training, validation and final-test intervals and the test scores hidden until a model is chosen.

> ✦ DID YOU KNOW? **A Fit That Beats the Truth on Its Own Record**
>
> From the simulator: on the training record of Table 1.1, the SOPDT model fitted by least squares scores 0.0311, below the 0.0319 of the true model that made the data. On the validation record the order turns round, 0.0331 against 0.0317. Part of what the fit learned was noise.

None of this makes the FOPDT useless. Its validation error is 0.043 °C on steps of 2 °C, and it is the model that the tuning rules of Volume 2 consume. The residual tests say that its structure is incomplete, and how much a purpose can tolerate is a separate question. Chapter 11 shows the opposite error. Adding radiation to an energy balance lowers the fitting objective on both of its step tests, by 5.4 and 11.8 %, yet the same model loses by 10 to 26 % on three other runs and claims that 70.1 % of the heat left by radiation in data generated with none.

A test can only determine what it excites. The heater step fixes τ_T and K_W, but $\tau_T = A_1\bar{L}/\bar{F}$ and K_W depends only on $\rho c_p \bar{F}$. Doubling A_1 while halving $\bar{L}$, or doubling ρ while halving c_p, leaves the record unchanged. Only the groups are identifiable [20]. Chapter 11 demonstrates exactly this: doubling the area and halving the heat transfer coefficient of its convective model leaves the objective unchanged in every displayed digit. Chapter 12 faces the multivariable form of the problem, where two inputs that moved together cannot be credited separately.

The chapters keep three models apart. The simulation model is the full balance, nonlinear and exact within its assumptions. The identified model is a fitted FOPDT or transfer function, good near one operating point. The controller's model is whatever the controller holds, which for a PID is nothing but its settings and for a predictive controller is one identified model, often fitted at a single point. Chapter 2 and Chapter 3 name the three explicitly. Chapter 13 turns residuals into diagnosis: a balance that does not close points to a meter in error, and a statistic that leaves its limits flags a fault.

Table 1.2 The Inventory, Model Form and Source of the Model in Each Chapter

Ch.	Inventory	Model form	How the chapter obtains it
2	water, one tank	first order with K_p and τ_p both $\propto \sqrt{L}$, integrating with the outlet shut	balance linearized, step tests and a least-squares time constant
3	water, two tanks	second order with a zero, first order with the splitter open	balances linearized, two-point fits at three splitter positions
4	heat, two heaters	four lags, 2×2 gain matrix, FOPDT for tuning	energy balances, manual step tests
5	momentum of a ball	integrator with a lag and dead time, gain set by the valve map	balance about hover, drop and step tests
6	water and energy	integrator with swell and shrink	balances with steam tables, open-loop cases
7	water momentum, rotor energy	right-half-plane zero, oscillatory modes	balances linearized, computed stability map
8	rotating energy	third order in the primary loop, offset set by stiffness	swing equation and closed-form analysis
9	water and heat	level integrating, temperature FOPDT or integrating	balances and one step test
10	oil stocks	damped pair and slow modes, two gains at two speeds	balances linearized, eigenvalues, steps
11	heat, one heater	three nested energy balances	fitted to records, checked by cross-validation
12	any recorded plant	FOPDT, SOPDT, ARX, FIR and state space for several inputs	fitted to records with held-out intervals
13	plant records	balances and statistical models of normal data	reconciliation, principal components, charts

1.4.10 Which Model Form a Plant Needs

Three kinds of model appear in the chapters. A model from balances alone uses physical constants and needs no test. A model from a test alone fits a standard form, FOPDT or ARX, without asking why the plant has that form. A model between the two takes its form from the balances and its constants from data, as in Chapter 11. Table 1.2 lists, for each chapter, the inventory that sets the dynamics, the form of model the chapter works with and how the chapter obtains it. The final chapter compares the plants along these columns, with the measured results of each.

1.5 Instruments and Final Elements in This Volume

A model is fitted to what an instrument reports, not to the plant itself. Figure 1.6 lists what the model of each chapter reads and what it moves. Level and temperature are read most often, by four chapters each. Speed or frequency serves the two power chapters, and three chapters work on a stored record in place of a live signal. The final elements run from a control valve and a pump to a heater, a turbine gate and a production quota, and Chapter 13 moves nothing at all.

A thermowell protects a temperature element and adds a first-order lag through its own mass and heat capacity. An air gap between the element and the bottom of the well adds a second lag, and such lags can compromise the stability of a feedback loop [21]. The thermowell of TT-1 is the lag τ_s of Equation (1.14), drawn in Figure 1.7, and the sensor lag of Chapter 4 sets the time constant fitted to its step test.

A pressure transmitter at the floor of a vessel infers the level from the weight of the column above it. The reading gives the height only while the density of the liquid is known and constant, although in a vessel of constant cross-section it measures the mass of liquid whatever its density [21]. In Chapter 6 such a reading sits closer to the chapter's mass-equivalent level than to the boiling surface. Figure 1.8 shows the two taps.

Three chapters start from records. A data historian polls process data from the control system at regular intervals and keeps it over long periods, typically years [21], and a standards glossary

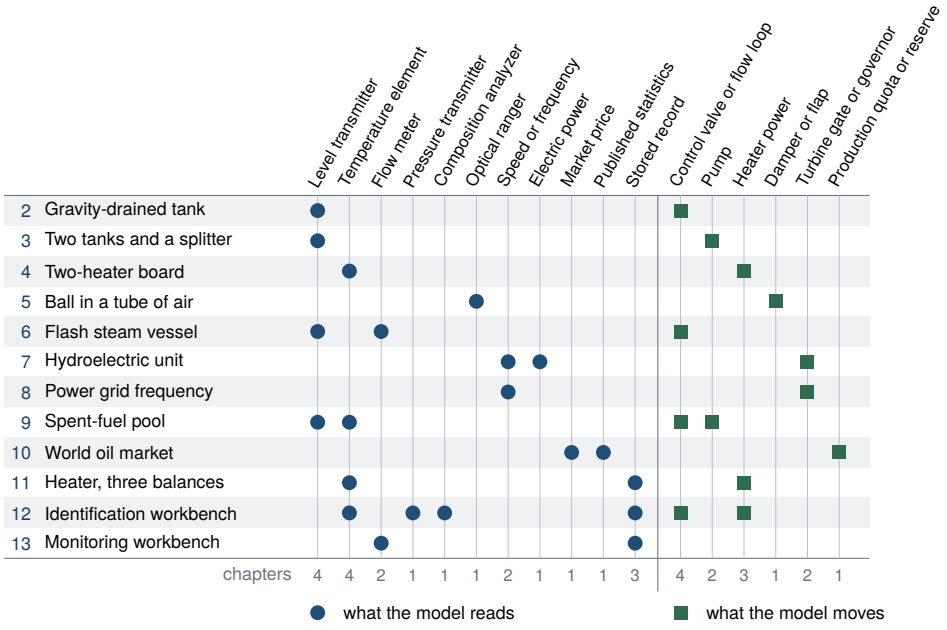


Figure 1.6 What the model of each chapter reads and moves: a round mark where the chapter's model reads the source and a square mark where it moves the element, as its variables table and its technology section name them. A chapter that works on stored records is also marked under the instruments its records hold. Notice that level, temperature and the control valve lead the counts, with four chapters each.

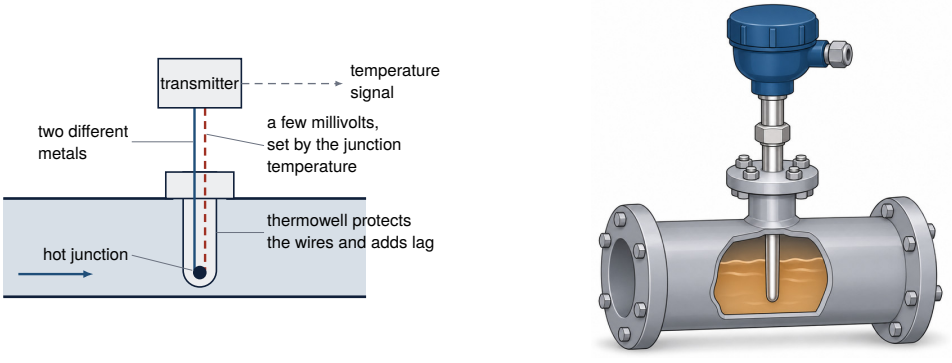


Figure 1.7 Principle of a thermocouple in a thermowell. Notice that the heat must cross the wall of the well before it reaches the junction, which is why TT-1 reads the outlet through a lag of its own (Section 1.4.6).

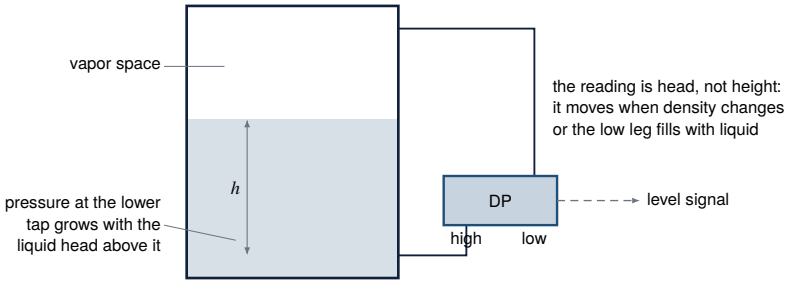


Figure 1.8 Level by differential pressure. Notice that the transmitter weighs the liquid above its lower tap, so it reads a height only while the density is known (Chapter 2 and Chapter 6).

Layer	What it does in this volume	Where it runs	How often
People who decide	engineers step the plant in Manual for a test, and a conference of governments sets a quota	**By hand**, in a plant test or at a meeting	a test of up to 5 to 15 days; meetings at least twice a year
Supervisory and predictive	predictive control holds an identified model, and generation control restores the grid	**Supervisory level** of a distributed control system, or an energy management system	0.1 s to 1 min in the apps
PID loop	the velocity-form PID of each chapter, with limits and anti-windup	**Distributed control system**, PLC, or a browser page beside a laboratory board	20 ms to 0.01 h; weekly in the oil market
Inner loop	a feed flow loop under level control, a gate servomotor, a flap servo	**Flow controller**, positioner or servo drive	0.25 s in the flash vessel
Sensors, final elements, records	transmitters, rangers and statistics; valves, pumps, heaters and gates; the archive of readings	wires, a serial link or a network, and a **data historian**	records from every 0.1 s to every 15 min, kept for years

Figure 1.9 The control layers of this volume, the platform that hosts each and how often each acts. Platforms are from [12], [22] and [21], and the app periods from the chapters' technology sections. Notice that the historian at the bottom keeps the records from which the models of Section 1.4.8 are fitted.

describes it as a central database that supports analysis by statistical process control [22]. That archive is the raw material of Chapter 11, Chapter 12 and Chapter 13.

Predictive control sits one level above the PID control of valves [12], and it holds an identified model in place of three settings. A test for such a model must excite both the steady state and the dynamics of the process [12]. Loops at every level run with cycle times from milliseconds to minutes [22]. Figure 1.9 places the loops of this volume on their platforms. The apps go from the 20 ms governor of Chapter 7 to the oil market of Chapter 10, whose quota changes only at meetings held at least twice a year.

1.6 The Simulator Framework

Every chapter of this book has an app that runs in a web browser, collected with the others at https://apmonitor.com/apps/. About 240 apps exist across the series. All but a few are built on one shared framework, so the engine, the PID algorithm, the trends and the data export behave the same way in every chapter. The exceptions are standalone studios, such as the cascade studio and the two Temperature Control Lab apps, which follow the same ideas in their own code. This section describes the framework once and states where the standalone apps differ. Table 1.3 lists what a reader meets.

Table 1.3 What Every App Provides. Standalone Apps Are Described in Their Own Chapters.

Element	What it does
Run, Pause, Reset	Start, hold and restart the run. Reset keeps the reader's settings.
Speed	Simulated time per second of real time. It changes the pace and never the result.
Manual and Auto	Hand each actuator to the reader or to the controller, without a bump.
Scenario list	Numbered cases, each with a briefing, a completion rule and a reload button.
Control panel	Setpoints, manual outputs, tuning, options and disturbances, at a Beginner or an Advanced level.
Trends	SP and PV above the MV for each loop, with limits, event marks and a cursor.
Scorecard	IAE, overshoot, settling time, MV travel and time at a limit for each loop.
Learn card	Predict, Try It, two hints and Explain, opened by the reader in that order.
Show Model	Equations, live variable values, editable parameters and assumptions.
Export CSV	Every logged sample of the run as a text file with a header row.
Instructions	A separate page with background, an operating guide and the model's limitations.

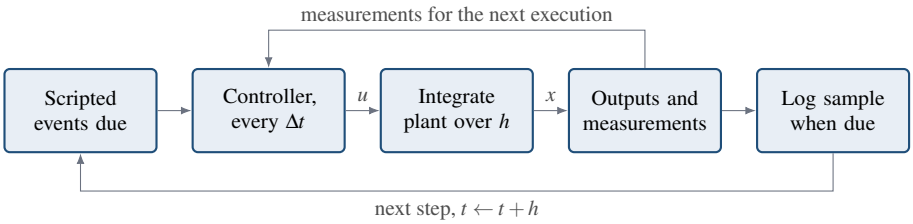


Figure 1.10 Work of the Engine in One Integration Step

1.6.1 The Simulation Engine

Each app carries a plant model written as ordinary differential equations $\frac{dx}{dt} = f(x,u,d,p)$ in the states x, the inputs u, the disturbances d and the parameters p. The engine integrates the model with the classical fourth-order Runge–Kutta method on a fixed step h set for the fastest dynamics of the app. Some apps supply their own step, such as explicit Euler for a discontinuous right-hand side or an exact update for a model driven by events.

Three clocks run at separate rates. The plant advances every step h. The controller executes every control period Δt, equal to h unless the app declares a slower controller, and holds its outputs between executions. The log takes one sample at a fixed interval of simulated time for the trends, the scorecard and the exported file. The order of work in one step is shown in Figure 1.10. The controller also executes once when a scenario is loaded, so every run starts from an initialized controller.

Drawing the screen is kept apart from these clocks. A speed of 5 means five units of the app's time base per second of real time. Speed, frame rate and pauses change how fast a run is shown and never what it computes, so a phone and a workstation produce the same trajectory and the same file.

Noise comes from a deterministic pseudorandom generator. Its seed is declared by the app, or by a scenario that needs its own noise record, and it is restored whenever a scenario is loaded or Reset is pressed. The framework gives the reader no seed setting, so *default seed* in this book means the seed the app ships with. A run repeats bit for bit when the same scenario is loaded and the same settings are made before Run. A change made by hand during a run lands on a different step each time, which is one reason numeric answers carry a tolerance. The standalone simulators draw noise from the browser's own generator, so a noisy run there repeats in its statistics and not sample for sample.

Run and Pause start and hold the clock. Reset restarts the run at $t = 0$, paused, and keeps the reader's settings: every setpoint, tuning value, option, model parameter, disturbance slider and mode the reader changed is put back before the plant and the controller are initialized. Scripted events replay, and a one-shot disturbance button pressed during the run is cleared. The reload button beside

the scenario list restores the scenario's own settings instead.

1.6.2 The Velocity-Form PID Algorithm

One PID implementation serves the PID loops of every framework app. It is the velocity (incremental) form: each execution adds the change in the proportional and derivative terms to a running sum, and the integral is a separate term, as shown in Equations (1.24) to (1.27) for execution k and control period Δt.

$$e_k = a\,(\mathrm{SP}_k - \mathrm{PV}_k), \qquad P_k = K_c\,e_k, \qquad D_k = -a\,K_c\,\tau_D\,\frac{\mathrm{PV}^f_k - \mathrm{PV}^f_{k-1}}{\Delta t} \tag{1.24}$$

$$S_k = S_{k-1} + (P_k - P_{k-1}) + (D_k - D_{k-1}) \tag{1.25}$$

$$v_k = u_b + S_k + I_{k-1}, \qquad u_k = \mathrm{sat}(v_k) \tag{1.26}$$

$$I_k = I_{k-1} + \left[\frac{K_c}{\tau_I}\,e_k + \frac{u_k - v_k}{T_t}\right]\Delta t \tag{1.27}$$

where $a = \pm 1$ is the controller action, PV^f is the measurement after an optional first-order filter, u_b is the output bias, S is the running sum of proportional and derivative increments, v is the output before limiting, u is the output sent to the actuator, I is the integral term and T_t is the tracking time constant, equal to τ_I unless a loop sets another value. The function sat clamps v_k to the actuator range and then applies a rate limit if the loop declares one.

The limit acts on the emitted output u_k only. The sum S_k is never clamped, so it equals $P_k + D_k$ and the output before limiting is the position form $u_b + K_c e_k + D_k + I_{k-1}$. Anti-windup touches the integral alone. While the output is limited, the last term of Equation (1.27) drives I toward the value that reproduces the output applied, which is back-calculation [23]. With no integral action, Equation (1.27) is skipped. Clamping the running sum would instead discard each increment that crosses a limit and keep each one that returns, so symmetric noise would push the output off the limit even in a proportional-only loop.

The derivative acts on the measurement, so a setpoint step gives a proportional kick and no derivative kick. A change of K_c acts on the whole error at the next execution, as in the position form. The action $a = +1$ suits a process whose PV rises with the MV, a reverse-acting controller in the ISA convention, and $a = -1$ gives the direct-acting one.

In Manual the controller keeps executing and sets $I_k = u_{\mathrm{man}} - u_b - S_k$, so the output it would send equals the manual value u_{man}. The first automatic output differs from it by one period's change in P and D. In Auto the engine copies each controller output to the manual station, so a switch to Manual is also bumpless. The standalone studios with PID loops use velocity-form code of their own, with the limit on the output and anti-windup on the integral alone, in which a gain change applies only to later increments.

1.6.3 The Predictive Control and Estimator Options

More than 100 apps offer model predictive control (MPC) as an option beside the structure they open with, which stays the default. Volume 9 develops the theory. In most of them the controller plans with a small linear model in discrete time as shown in Equation (1.28).

$$x_{k+1} = A\,x_k + B\,u_k + E\,w_k + c, \qquad y_k = C\,x_k \tag{1.28}$$

where w is a measured or scheduled disturbance and c is a constant left by linearization. The model comes from a step test at the design point, from the app's balances linearized there or again at every decision, or from the model the reader fits. It never reads the true states or parameters of the run, so a parameter the reader changes in Show Model is model mismatch, as on a real plant.

At each decision the controller minimizes the objective of Equation (1.29) over a horizon of N steps.

$$J = \sum_{k=1}^{N} (y_k - r_k)^{\top} Q (y_k - r_k) + \sum_{j} \Delta u_j^{\top} R \, \Delta u_j + \sum_{i} \rho_i s_i^2 \tag{1.29}$$

where r is the setpoint, Q weights the tracking error, R is the move suppression on each input change Δu_j and ρ_i is the price of a slack $s_i \geq 0$ that lets an output band be broken when it cannot be held. Actuator limits are hard constraints. The quadratic program is solved in the browser by a dual active-set method [24]. The first move is applied, and the plan is made again at the next decision.

The plan starts from an estimate of the model state. About half of these apps use a steady-state Kalman filter on the model with an added disturbance state, which absorbs what the model did not predict and removes steady offset without an integral term [25]. Others correct the model output with a filtered bias to the same end. About a dozen apps also offer moving horizon estimation (MHE), which solves the estimation problem over a window of recent samples and can hold bounds that a Kalman filter cannot, such as a concentration that is never negative [26].

With the MPC selected, the tuning fields of the loops it replaces leave the Control panel. In most of these apps the Advanced level shows up to three settings, each a multiple of the shipped value: the prediction horizon, the move suppression and the band price. The trend draws the accepted plan ahead of the present time, and a list of constraint prices shows which limits bind. A plan that is not optimal, not finite or outside its limits is rejected, and a fallback, in most apps the shipped controller, holds the actuator until a plan succeeds. The Model Fitting and Tuning Studio builds the same kind of controller from the reader's own data in its tab, MPC from the fit.

1.6.4 Scenarios, Progress and Data

An app teaches through scenarios, the numbered cases at the top of the Control panel, nine to thirteen in most apps. Loading one sets the model parameters, the controller settings, the disturbances, the mode, the initial state and any scripted events. The information button beside the list opens a briefing in four parts: what the case is, why it matters, what to do and one detail for the practicing engineer. The Learn card holds the lesson back, with Predict and Try It before the run, hints on request and Explain afterward.

Each scenario has a completion rule that the app judges from the run on the simulated clock: a run length, the scripted events passed, a setting made by the reader, a state held for a stated time or a target met. Choosing a case earns nothing. A completed case is marked in the list, and a badge row shows the count and a level, Explorer after one case, Practitioner at half and Completed at all. Progress is kept in the reader's browser with no account. A certificate of progress or of completion can be issued from it, as a record of self-paced work and not an assessed credential.

One case, usually the last, is the graded challenge. It states a target, such as an error band to hold through a sequence of upsets. Changing the model parameters or the disturbances voids the attempt in most apps, and a record made with Reveal suggested tuning says so.

Show Model opens the equations, live values of the variables, the editable parameters and the assumptions. The trends plot SP and PV above the MV for each loop, with actuator limits and marks at setpoint moves, disturbances, mode changes and scripted events. The scorecard reports for each loop the integral of the absolute error (IAE), overshoot, settling time, MV travel and share of time at a limit, with the app's own measures, and it does not combine them into one score. IAE and travel accumulate from Reset, so two runs are compared at the same clock reading.

Export CSV saves the log as a text file with a header row of column names that carry their units and one row per logged sample. In most apps the time column is in seconds, minutes or hours according to the run length. The log holds the most recent 60 000 samples unless the app sets another size. Apps with the MPC option begin the file with comment lines that start with #, which a script must skip.

An app exercise in this book names the scenario by its number and name as the app shows them, every setting that differs from the scenario's defaults, the run length in simulated time, the seed

(the default seed unless stated), what to read and where (a trend, a scorecard tile or a column of the exported file) and a tolerance on a numeric answer. Settings are made before Run unless the exercise gives a time.

1.7 Working With AI Agents While Studying

A reader of this book is likely to have an AI assistant within reach, and the apps make its work easy to test. This section gives the editor's view of how to use one while working through a chapter. In short, the assistant may do the algebra and the scripting, and the reader keeps the checking.

What to delegate. Tasks with a clear input and a result that can be tested are good candidates. An assistant can derive the linearization of a chapter's balance at an operating point and the transfer function that follows from it. It can write a short script that reads the exported CSV file and fits a first-order-plus-dead-time model to a step test. It can propose a first tuning from that fit with a named rule, convert units, explain a term in a model or outline a report. The assistant knows nothing about the app until it is told. Give it the equations and parameter values from Show Model, the column names of the exported file and the scenario that produced the data, as described in Section 1.6.4. Ask it to state its assumptions and the units of every result.

What to check, and how. An assistant returns a fluent answer whether or not the answer is right, so each result needs a test that it could fail. Four tests cover most of the errors a chapter will produce.

- *Units.* Every term of an equation carries the same units, and a process gain is in PV units per MV unit. A gain computed with the valve as a fraction is 100 times the gain of a loop whose valve is in percent.
- *Signs.* Decide from the physics which way the PV moves when the MV rises. Compare that with the sign of the fitted gain and with the controller action of Section 1.6.2.
- *Steady state.* Set the derivatives to zero. The model must return the operating point that Show Model reports, and a linearization must give zero deviation there.
- *A run on the app.* Load the named scenario at the default seed, enter the proposed numbers and run. The trend and the scorecard decide. A tuning that reads well on paper and holds the valve at a limit for half the run has failed.

A fitted model gets one more test: compare it with data it was not fitted to, such as a second step in the opposite direction. When a check fails, report the failure to the assistant with the evidence and ask for the cause, then check the correction the same way.

How the exercises train the habit. One exercise in each chapter is marked *Review the agent*. It presents a tuning, a model or a claim as an AI assistant might propose it, with a flaw, and asks the reader to find the flaw in about five minutes. The flaws are ordinary ones: a unit, a sign, a steady state that does not close, a result measured on the wrong scenario or at a different run length, or a limit that was ignored. Each is found with one of the tests above. The purpose is not distrust. An engineer who can say why an answer is right can hand an assistant much larger tasks and still stand behind the result. That larger picture, in which agents do most of the routine work and the engineer supervises and checks, is the subject of Chapter 14.

1.8 Chapter Roadmap

Table 1.4 lists the chapters with the modeling lesson each adds to this one. The level says how much of Section 1.4 a chapter assumes. The time is an estimate for reading the chapter once and running each case of its app once, at about eight minutes a case. Figure 1.11 places the same chapters by where their models come from and by the response their lessons center on.

	From balances alone	Form from balances, constants from tests	From records alone
First order and integrating		(2) gain moves with level (5) an integrator behind a valve (9) one pool, two plants	
Second and higher order	(8) inertia and droop offset (10) two gains at two speeds	(3) order set by piping	
Lags read as dead time		(4) sensor lag dominates	
Inverse response	(6) swell and shrink (7) a zero from water inertia		
Structure and validation		(11) three nested balances	(12) scored on held-out data (13) residuals flag faults

Figure 1.11 Method map of the volume. Each chapter is placed by how it obtains its model, from the last column of Table 1.2, and by the response its lesson centers on, with that lesson shortened from Table 1.4. The placement is the editor's reading of the two tables. Notice that most chapters take the form of the model from a balance and its numbers from a test.

Table 1.4 Roadmap of the Volume

Ch.	Plant	The modeling lesson	Level	Time
2	Gravity-drained tank	Linearization: gain and time constant both follow the square root of the level. The integrating limit.	introductory	2.5 h
3	Two tanks and a splitter valve	Tanks in series, plant order set by piping, the apparent dead time of a second lag.	introductory	3 h
4	Two-heater laboratory board	Energy balances, a gain matrix and a step test dominated by the sensor.	introductory	3 h
5	Ball in a tube of air	An integrator behind a valve whose gain changes thirtyfold.	intermediate	2.5 h
6	Flash steam vessel	Swell and shrink: a level that moves against its inventory.	advanced	3 h
7	Hydroelectric unit	A right-half-plane zero from water inertia and the gain it allows.	advanced	2 h
8	Power grid frequency	An energy inventory: inertia, droop offset and integral restoration.	intermediate	2.5 h
9	Spent-fuel pool	One measurement, two plants: self-regulating or integrating with the pumps.	intermediate	3 h
10	World oil market	A tank of stocks with a short-run and a long-run gain.	intermediate	2.5 h
11	Heater with three balances	Model structure, identifiability and cross-validation.	intermediate	3 h
12	Identification workbench	Multivariable models from records, scored on held-out data.	advanced	3 h
13	Monitoring workbench	Residuals of balances and statistics as evidence of faults.	advanced	2 h

Chapters 2 to 4 form a first course and are best read in order: one tank, two tanks, then heat in place of water. Chapter 5 and Chapters 8 to 10 each study one inventory in depth and can be read in any order after Chapter 2. Chapters 6 and 7 need Section 1.4.7. Chapters 11 and 12 need Sections 1.4.8 and 1.4.9, and Chapter 11 is best read after Chapter 4, whose board it models. Chapter 13 uses records from plants across the series and is best read last. A reader who has taken a first course in process dynamics can take a shorter path through Chapters 3, 6, 9, 11 and 12.

KEY POINTS

- A model comes from balances, from a test or from both. The balance gives its form and how its constants move, and the test supplies the numbers.
- Gain and time constant follow the operating point, so a step test returns the values of the range it crosses.
- Small lags in series look like dead time, and an effective dead time depends on the method that reads it.
- A small training error says only that a model can follow its own record. Validation on new data and residuals that look like noise are the stronger tests.

Where else this applies. Tuning from these models in Volume 2, dead time in Volume 5, nonlinear models in Volume 7.

References

[1] D. E. Seborg, T. F. Edgar, D. A. Mellichamp, and F. J. Doyle, III, *Process Dynamics and Control*, 4th ed. Hoboken, NJ: John Wiley & Sons, 2016.

[2] K. J. Åström and R. M. Murray, *Feedback Systems: An Introduction for Scientists and Engineers*, 2nd ed. Princeton, NJ: Princeton University Press, 2021.

[3] L. Ljung, "Perspectives on system identification," *Annual Reviews in Control*, vol. 34, no. 1, pp. 1–12, 2010.

[4] O. Tramis, E. Merlin-Anglade, G. Paternoster, M. Rabaud, and N. M. Ribe, "Torricelli's curtain: morphology of horizontal laminar jets under gravity," *Physics of Fluids*, vol. 33, no. 6, p. 062116, 2021.

[5] International Association for the Properties of Water and Steam, "Obsolete IAPWS recommendations," IAPWS, Information document, Mar. 2005. [Online]. Available: https://iapws.org/documents/release/Obsolete.download

[6] W. F. Guthrie, "NIST/SEMATECH e-Handbook of Statistical Methods (NIST Handbook 151)," National Institute of Standards and Technology, 2020, section 6.3.2, What are Variables Control Charts? [Online]. Available: https://www.itl.nist.gov/div898/handbook/

[7] J. G. Ziegler and N. B. Nichols, "Optimum settings for automatic controllers," *Transactions of the ASME*, vol. 64, no. 8, pp. 759–765, 1942.

[8] G. Pillonetto, T. Chen, A. Chiuso, G. De Nicolao, and L. Ljung, *Regularized System Identification: Learning Dynamic Models from Data*, ser. Communications and Control Engineering. Cham: Springer, 2022.

[9] K. Iinoya and R. J. Altpeter, "Inverse response in process control," *Industrial & Engineering Chemistry*, vol. 54, no. 7, pp. 39–43, 1962.

[10] K. J. Åström and P. Eykhoff, "System identification—a survey," *Automatica*, vol. 7, no. 2, pp. 123–162, 1971.

[11] H. Akaike, "A new look at the statistical model identification," *IEEE Transactions on Automatic Control*, vol. 19, no. 6, pp. 716–723, 1974.

[12] S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control Engineering Practice*, vol. 11, no. 7, pp. 733–764, 2003.

[13] Health and Safety Executive, "Better alarm handling," Health and Safety Executive, HSE Information Sheet Chemicals Sheet No 6 (CHIS6), 2000, information current at 3/00. [Online]. Available: https://www.hse.gov.uk/pubns/chis6.pdf

[14] A. Rasheed, O. San, and T. Kvamsdal, "Digital twin: Values, challenges and enablers," arXiv preprint 1910.01719 (version 1); published in revised form as IEEE Access 8, 21980–22012, 2020, 2019.

[15] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *Journal of the Society for Industrial and Applied Mathematics*, vol. 11, no. 2, pp. 431–441, 1963.

[16] J. A. Nelder and R. Mead, "A simplex method for function minimization," *The Computer Journal*, vol. 7, no. 4, pp. 308–313, 1965.

[17] S. Skogestad, “Simple analytic rules for model reduction and PID controller tuning,” *Journal of Process Control*, vol. 13, no. 4, pp. 291–309, 2003.

[18] H. Garnier, M. Mensler, and A. Richard, “Continuous-time model identification from sampled data: Implementation issues and performance evaluation,” *International Journal of Control*, vol. 76, no. 13, pp. 1337–1357, 2003.

[19] G. M. Ljung and G. E. P. Box, “On a measure of lack of fit in time series models,” *Biometrika*, vol. 65, no. 2, pp. 297–303, 1978.

[20] R. Bellman and K. J. Åström, “On structural identifiability,” *Mathematical Biosciences*, vol. 7, no. 3–4, pp. 329–339, 1970.

[21] T. R. Kuphaldt, *Lessons in Industrial Instrumentation*, version 2.33 ed. Open textbook, Creative Commons Attribution 4.0 license, ibiblio.org/kuphaldt/socratic/sinst, 2024.

[22] K. Stouffer, M. Pease, C. Tang, T. Zimmerman, V. Pillitteri, S. Lightman, A. Hahn, S. Saravia, A. Sherule, and M. Thompson, “Guide to operational technology (OT) security,” National Institute of Standards and Technology, Gaithersburg, MD, Special Publication 800-82r3, 2023.

[23] K. J. Åström and L. Rundqwist, “Integrator windup and how to avoid it,” in *Proc. 1989 American Control Conference*, 1989, pp. 1693–1698.

[24] D. Goldfarb and A. Idnani, “A numerically stable dual method for solving strictly convex quadratic programs,” *Mathematical Programming*, vol. 27, no. 1, pp. 1–33, 1983.

[25] G. Pannocchia and J. B. Rawlings, “Disturbance models for offset-free model-predictive control,” *AIChE Journal*, vol. 49, no. 2, pp. 426–437, 2003.

[26] C. V. Rao, J. B. Rawlings, and D. Q. Mayne, “Constrained state estimation for nonlinear discrete-time systems: Stability and moving horizon approximations,” *IEEE Transactions on Automatic Control*, vol. 48, no. 2, pp. 246–258, 2003.

1.9 Exercises

1.1 Theory and calculation · 5 min

A gravity-drained tank has a cross-section of $2\,\mathrm{m}^2$ and an outflow $\beta\sqrt{L}$ with $\beta = 0.5\,\mathrm{m}^{2.5}/\mathrm{min}$. It operates at a level of 4 m. Find the steady feed, then the gain and the time constant of the linear model from the feed to the level.

1.2 Theory and calculation · 4 min

TK-1 of the running example rests at 1 m with a feed of $0.2\,\mathrm{m}^3/\mathrm{min}$ and $\beta = 0.2\,\mathrm{m}^{2.5}/\mathrm{min}$. The feed is raised to $0.24\,\mathrm{m}^3/\mathrm{min}$. Find the new steady level from the nonlinear balance and the change the linear model predicts. By how many percent does the linear model miss the change?

1.3 Concept check · 2 min

The orifice of TK-1 is replaced by a pump that draws a fixed $0.2\,\mathrm{m}^3/\mathrm{min}$. With the tank at 1 m, the feed is raised from 0.2 to $0.22\,\mathrm{m}^3/\mathrm{min}$ and left there. What does the level do?

(a) It settles 0.21 m higher after about 30 min

(b) It rises at 0.02 m/min until the tank overflows or the feed changes

(c) It rises 0.20 m and then returns to 1 m

(d) It falls first and then rises, because the pump lags the feed

1.4 Theory and calculation · 5 min

A well-mixed tank holds $2\,\mathrm{m}^3$ of water with a throughflow of $0.4\,\mathrm{m}^3/\mathrm{min}$ at a fixed level, and $\rho c_p = 4200\,\mathrm{kJ}/(\mathrm{m}^3\,\mathrm{K})$. Its heater power is raised by 14 kW. Find the time constant of the outlet temperature, its steady change and the time at which it has covered 63.2 % of that change.

1.5 Theory and calculation · 6 min

A plant has the model $G(s) = 3/[(8s+1)(2s+1)]$ with time in minutes.

(a) Write it in the standard second-order form and give τ_n and ζ. Can its step response overshoot? *Checkpoint:* $\tau_n = 4\,\mathrm{min}$, $\zeta = 1.25$, no overshoot.

(b) Reduce it to a first-order-plus-dead-time model with the half rule. *Checkpoint:* $K_p = 3$, $\tau_p = 9\,\text{min}$, $\theta_p = 1\,\text{min}$.

1.6 Theory and calculation · 3 min

The thermometer of a heated tank sits at the end of a pipe that holds $0.3\,\text{m}^3$. Find the dead time at flows of 0.2, 0.1 and $0.3\,\text{m}^3$/min, assuming plug flow.

1.7 Concept check · 3 min

An ideal water turbine has the model $(1 - T_w s)/(1 + T_w s/2)$ from gate to power, with $T_w = 2\,\text{s}$. The gate is opened by a small step. What does the power do, per unit of its final change?

(a) It rises at once to its final value and stays there

(b) It rises with no overshoot and a time constant of 1 s

(c) It overshoots to twice its final value and then settles

(d) It jumps to -2, rises through zero near 1.1 s and settles at $+1$

1.8 Theory and calculation · 4 min

A least-squares fit of the ARX model $y'_k = \phi\, y'_{k-1} + \gamma u'_{k-1-d}$ to a record sampled every $T_s = 1\,\text{min}$ returns $\phi = 0.9$ and $\gamma = 0.03$. Find the time constant and the gain of the equivalent first-order model.

1.9 Theory and calculation · 5 min

Three steady tests on a plant give PV changes of 1.1, 1.9 and 3.1 units for MV changes of 5, 10 and 15 units. Fit the gain of the model $\Delta y = K_p \Delta u$ by least squares and give the three residuals.

1.10 Concept check · 3 min

Two models are fitted to the same step test, with measurement noise of standard deviation $0.030\,^\circ\text{C}$. Model A has a training error of $0.034\,^\circ\text{C}$, a validation error of $0.043\,^\circ\text{C}$ and a lag-one residual autocorrelation of 0.19 over 201 samples. Model B has 0.031, 0.033 and 0.035. Which reading is right?

(a) Model A, because fewer parameters are always preferred

(b) Neither, because the training errors differ by less than the noise

(c) Model B, because its residuals look like noise and it holds on new data

(d) Model A, because a positive autocorrelation shows it follows the dynamics