INTERACTIVE CONTROL ENGINEERING
Principles and Applications

Volume 1 Process Dynamics, Modeling, and Identification

CHAPTER 14

Summary and Outlook

Pages 338–351

Edited by John D. Hedengren

How to cite this chapter. "Summary and Outlook," 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. 14, pp. 338–351.

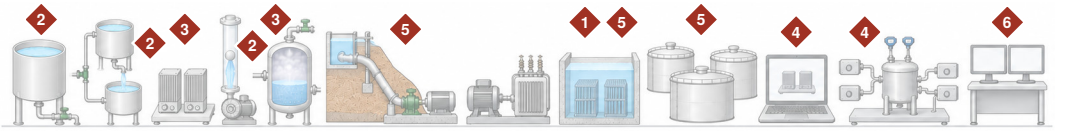
The page numbers in this file are those of the printed volume.

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 14

Summary and Outlook



Open gaps 1 slow and integrating tests 2 operating-point models 3 instrument in the model
4 physics-aware structure 5 untestable events 6 correlated monitoring

Notice that the fuel pool carries two gaps and that the laboratory plants on the left met the gaps of operating point and instrument, while the large plants met the gap of events that cannot be tested. Illustration, not to scale.

IN PLAIN TERMS **What Twelve Plants Taught About Models**

The main idea. Every chapter of this book built a model of a plant, from a water tank to the world oil market. The model came from counting what flows in and out, from pushing the plant and recording its answer, or from both.

The challenge. A curve read from one test belongs to that test and can mislead on the next. Some events, such as a fuel pool boiling, can never be tried, so their models are never checked against the real thing.

Why it matters. Checking each model on a second record, and over its whole range, catches most of the errors these chapters found.

Twelve chapters built models of twelve plants, from a tank of water to a world oil market. Each chapter had to decide which form of model its plant needed and where the constants would come from: a balance, a test, a record, or a mix of the three. This chapter sets those decisions side by side with the results they produced, draws a short guide for choosing a method, looks at where modeling and identification practice is heading, reports how the volume itself was made with AI agents and what they got wrong, and lists the problems that remain open.

14.1 What the Chapters Showed

Table 1.2 of Chapter 1 listed, for each chapter, the inventory that sets the dynamics, the model form and how the chapter obtains it. Tables 14.1 and 14.2 add the results. Each row names the difficulty the plant posed, the model that worked and how it was obtained, a model or reading that failed, and the measured result with the table or section of the chapter that reports it.

Figure 14.1 draws one like pair of those results from each row. The columns of the two tables make five points.

The balance gave the form, and a test gave or checked the numbers. The tank's gain and time constant both moved with the level, and the balance said in advance that their ratio would not. Every step test then started at the same slope. The two tanks returned one gain at every splitter setting, as the balances said they would, while the dynamics changed. The twin of Chapter 4 reproduced the gain matrix of its own energy balances. In each case the balance told the experimenter what to expect and the test confirmed or corrected the constants.

A step test reads a model that belongs to the method and the window. None of the plants of Chapter 3 or Chapter 4 has a transport delay, yet both fits returned one. The two-tank fit read 18.5 to

Table 14.1 Which Model Each Plant Needed, Chapters 2 to 7

Chapter	Difficulty	Model that worked, and its source	What failed	Measured result
2, gravity-drained tank	Gain and time constant both grow as $\sqrt{h}$. An integrator with the outlet shut.	The mass balance linearized at each level, checked by 1 % step tests and a drain test.	A step that moved the level 21 % at 2.04 m: its fitted time constant sat 7.0 % above the tangent value.	Fitted gain up 2.14 times and time constant 2.12 times from 2.04 to 9.88 m, against 2.20 predicted, and a drain in 19.65 against 19.79 s (Section 2.4).
3, two tanks and a splitter	The splitter sets the order of the plant. Its gain does not move.	Linearized balances, two lags whose pole and zero cancel with the splitter open. Two-point fits at three settings.	A tuning from the open-splitter fit run with the splitter shut. A predictive model fitted shut and run half open.	Gain 0.0323 m/% at every setting, apparent dead time 18.5, 4.0 and –0.6 s (Table 3.3). Overshoot 58.7 % (Table 3.5). Half-open IAE 8.68 fixed, 1.00 scheduled, 2.65 for the PID (Table 3.8).
4, two heaters	Four lags. A thermistor slower than the heater it measures.	Energy balances of heaters and sensors, a 2×2 gain matrix from manual steps.	The fitted time constant read as the heater's. A PID loop that meets the sensor lag only as error.	Gains 0.599, 0.050, 0.100 and 0.300 °C/%, $\tau_p = 141$ and $\theta_p = 18$ s on a 140 s sensor (Table 4.3). Loop 2 upset 0.20 under the predictive controller against 1.68 °C (Table 4.7).
5, ball in a tube	An integrator behind a valve whose gain falls from 30.4 to 1.0 mm/s per degree over its stroke.	A momentum balance about hover with a valve map, a drop test and step tests on the twin.	The valve without its orifice. A 200 ms sensor.	Usable band 1.19° without the orifice and 22.34° with it (Section 5.4). IAE 1880 with the 200 ms sensor against 72 at 66 ms (Table 5.6). No number yet from the rig.
6, flash vessel	The level reads water plus the vapor below the surface. It swells while the vessel loses water.	Mass and energy balances closed by steam tables. A controller that expects the swell.	Single-element control, which reads swell as inventory. An inventory estimate under foaming water.	Swell 3 points on a 25 % demand rise (Section 6.4.2). IAE 399, 232 and 152 % s for single-element, three-element and predictive (Table 6.4).
7, hydro unit	Water inertia: opening the gate first cuts the power.	The water column linearized to a right-half-plane zero. A computed stability map checked against the engine.	The design plan's zero at $2/T_w$. A governor tuned on the grid and run on the island.	Dip 4.38 MW, boundary gain 11.73 mapped and 11.70 in the engine (Table 7.4). Zero 0.675 rad/s at 80 MW (Section 7.4.1). Island 56.009 to 62.479 Hz (Table 7.5).

Table 14.2 Which Model Each Plant Needed, Chapters 8 to 13

Chapter	Difficulty	Model that worked, and its source	What failed	Measured result
8, grid frequency	One event, three quantities, each set by a different group of parameters.	The swing equation in closed form: rate of change from inertia alone, offset from droop and load damping alone.	Droop alone after a unit trip, which leaves the grid at 59.429 Hz (Table 8.5).	Rate of change −0.150, −0.299 and –0.746 Hz/s at the same settled 59.857 Hz. Settled 59.941, 59.857 and 59.679 Hz at one rate (Table 8.4).
9, spent-fuel pool	Self-regulating with the cooling trains running, integrating without them. Hours per time constant.	A full step test for the running pool. The two balances marched forward to the limits for the stopped one.	The first hour read as an integrator. The present rate extrapolated to boiling.	$K = -0.0551\,^{\circ}\mathrm{C}/\%$, $\tau = 3.19\,\mathrm{h}$, $\theta = 0.16\,\mathrm{h}$, and an integral time 2.6 times too long from the first hour (Section 9.5). Boiling forecast within 0.10 against 1.12 h early (Section 9.6).
10, oil market	One tank of stocks with a short-run and a long-run gain.	Balances linearized, gains of \$12.5 and \$1.34 per million barrels a day (Section 10.4), a factor of 9.3 (Section 10.9).	A PI that holds \$100 on the short-run picture.	Price \$98.3 on average while the share falls from 30 to 15.27 %. The economic predictive controller at a mean of \$62.1 keeps 32.64 % (Table 10.5).
11, heater with three balances	Nested model forms. Only the products UA and mc_p are identifiable.	The convective form fitted on one step test and judged on a second, untouched one.	A radiation term that improved its own record and reported a radiative share in data generated without radiation.	Objective 141.9 to 134.2 on the fitted step, 283.8 to 429.0 °C on the held-out step (Table 11.5). Share 70.1 % (Table 11.6).
12, identification workbench	Several inputs that move at once. Paths that may be absent.	Transfer-function matrices with exact structural zeros, ranked on free-run error over held-out data.	A zero imposed on a strong path. A FIR model on an integrating level.	Six gains within 0.9 % (Table 12.5). Validation 0.1093 to 0.1085 for a true zero, 0.3892 for a false one (Table 12.6). Tank 4.386 FIR, 0.062 state space (Table 12.4).
13, monitoring workbench	Faults hidden in records of normal operation. Correlated samples.	Balances for reconciliation, a principal component model of normal data, the valve plot.	T^2 on the cooling water step. Loop indices for a valve fault.	Station meter named at 2.35 L/s, test 313.67 to 0.135 (Section 13.5). Q at 48.05 h, T^2 never (Table 13.3). False detections 1.25 a day (Section 13.5.3).

–0.6 s as the splitter opened, and the heater fit read a time constant that belongs to the thermistor. The first hour of the pool's response looks like an integrator and gives an integral time 2.6 times too long. The oil market's short-run gain is 9.3 times its long-run gain. A fitted constant is true of the plant only together with the method and the window that produced it (Section 1.4.6).

Where a test is unsafe or impossible, the balance carries the analysis. No one steps a penstock to find its zero, boils a spent-fuel pool to time it or runs a decade of oil prices twice. The hydro chapter derived its zero and checked a stability map against the engine to about 0.26 %. The pool chapter marched its balances to boiling and beat extrapolation by an hour. The flash vessel's swell came from balances closed by steam tables and a declared bubble closure, not from a fit. The same chapters show the cost: a balance is only as good as its weakest correlation, and the ball's twin still waits for its rig.

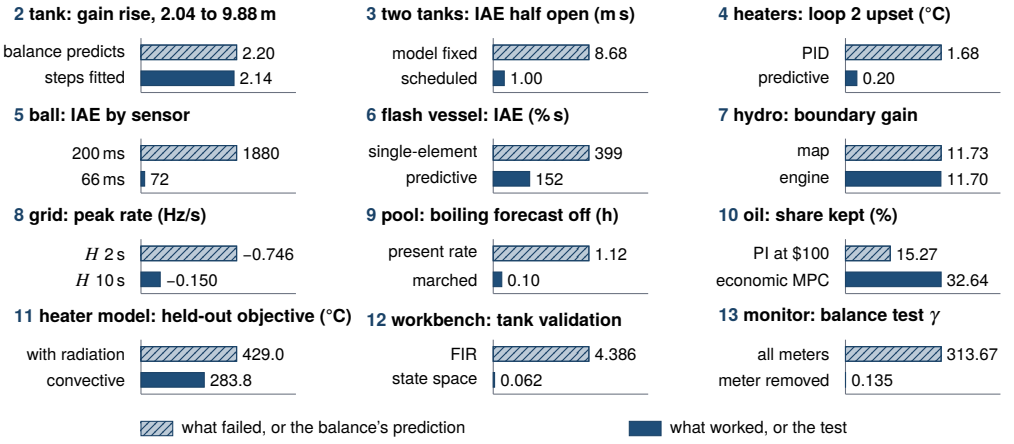


Figure 14.1 One like pair of numbers from each row, as the "Measured result" column of Tables 14.1 and 14.2 prints them. Each panel has its own scale from zero, and the grid panel's axis is reversed so that its negative rates grow to the right. In most panels the hatched bar is the model, reading or setting that failed and the solid bar the one that worked. In the draining-tank and hydro panels the hatched bar is the balance's prediction and the solid bar the test or the engine, and in the grid panel the two bars are the two inertias. Notice that the bars of every failure panel differ plainly, while a balance and its test agree so closely that their bars nearly match.

More parameters fit better and predict worse. The radiation term of Chapter 11 lowered the objective on its own record and raised it from 283.8 to 429.0 °C on the held-out record, and it reported a 70.1 % radiative share in data that contained none. A structural zero on a strong path in Chapter 12 raised the validation error from 0.1093 to 0.3892, and the FIR model, with hundreds of coefficients, was the worst of five families on the integrating level. In both chapters a second record decided the structure and the objective of the first record did not (Section 1.4.9).

A statistic flags, a balance names. The workbench of Chapter 13 named the faulty meter by eliminating measurements until the balances closed. Its statistical monitor detected the cooling water step in Q and never in T^2, and on normal days it raised 1.25 detections a day where its limit assumed far fewer, because the samples were not independent. A statistic says that the record departs from normal data. A model of the plant says where and why.

14.2 Choosing a Method

The chapters suggest an order of questions for a new model. Each question names the chapter where its answer decided the result, and Figure 14.2 draws the same order.

1. *Which inventory sets the dynamics, and can its balance be written?* Write it first, even when the constants will come from data. It gives the form, the sign of each gain and the groups of parameters a test can identify, as UA and mc_p in Chapter 11.

2. *Does the operating point move the constants?* Linearize at several points (Section 1.4.3). Gain and time constant that move together, as in Chapter 2, leave the initial slope fixed. A gain that moves by a large factor, as the ball's valve gain does, needs a model at each operating point or a schedule.

3. *Is the plant self-regulating or integrating over the horizon that matters?* Run the test long enough to tell the two apart. The pool of Chapter 9 is both, depending on which pumps run.

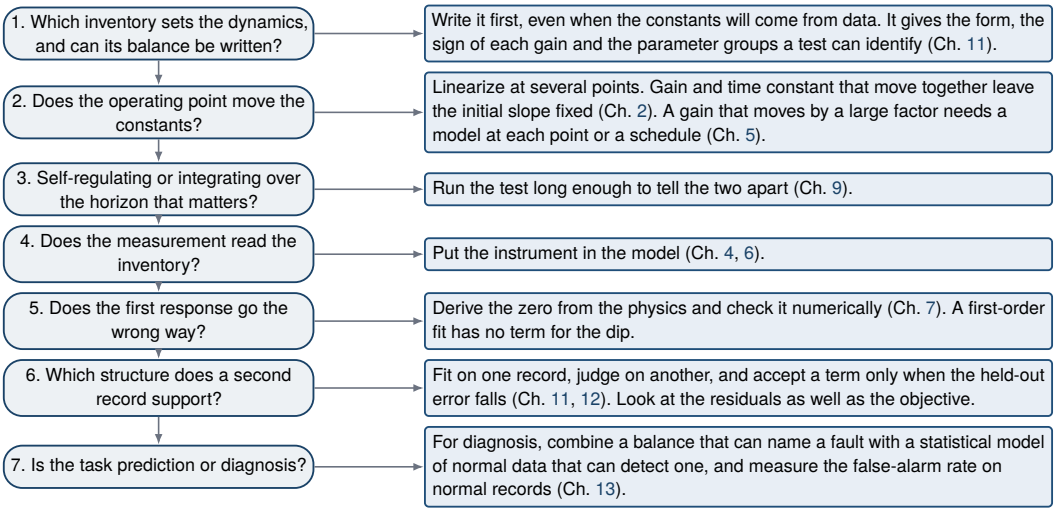


Figure 14.2 The seven questions of this section in order, with the list's answer and the chapters where it decided the result. Notice that the balance comes first and the second record comes late: the form of a model is settled before any data are fitted, and its structure is settled only after a fit has been judged on data it has not seen.

4. *Does the measurement read the inventory?* A thermistor, a swelling level and a thermowell add dynamics or reverse the first response. Put the instrument in the model (Chapter 4, Chapter 6).
5. *Does the first response go the wrong way?* Derive the zero from the physics and check it numerically, as Chapter 7 does. A first-order fit has no term for the dip (Section 1.4.7).
6. *Which structure does a second record support?* Fit on one record, judge on another, and accept a term only when the held-out error falls (Chapter 11, Chapter 12). Look at the residuals as well as the objective.
7. *Is the task prediction or diagnosis?* For diagnosis, combine a balance that can name a fault with a statistical model of normal data that can detect one, and measure the false-alarm rate on normal records (Chapter 13).

A predictive controller showed what each model was worth. It cut the two-tank IAE where its model matched the splitter and scored 3.3 times the PID's where it did not. It held the heater twin's second loop within 0.20 °C because it carried the conduction between the heaters, which is a matched-model result on a simulator. Volume 9 treats predictive control in full.

14.3 Where Practice Is Heading

Identification meets machine learning. Identification with kernels and regularization links system identification to machine learning and function estimation [1]. Common deep networks are nonlinear ARX models, and recurrent networks of the LSTM kind are nonlinear state-space models, so both fit into the established code and practice of identification [2]. For plants that are not linear, a guide for practicing engineers walks through the choices a user must make and the ways the nonlinear problem differs from the linear one [3]. In the editor's view the names of the model families will keep changing, while the questions of Chapter 11 and Chapter 12, which structure a second record supports and which parameter a record cannot determine, will not.

Physics and data are combined. Hybrid models join parametric submodels from first principles with nonparametric submodels fitted to data. A review of their use in process systems engineering found that they do not lead to better results automatically, although a reasoned combination of knowledge can improve model-based operation and design [4]. Physics-informed machine learning trains networks on data together with the physical laws they must obey [5]. Sparse regression can select the fewest terms of a governing equation that represent the data, trading accuracy against complexity to avoid overfitting [6]. Language models have been used to propose the skeleton of an equation from domain knowledge, with the parameters then fitted to data. On four benchmark problems this found more accurate equations than symbolic regression baselines, most of all outside the range of the training data [7]. The radiation term of Chapter 11 is the warning for all of these methods: a physically plausible term can fit noise and report a physical quantity that is not there.

> ⧖ HISTORY **1976 to 2003: The Questions Industry Asked About Models**
>
> The timeline of Chapter 1 ends its testing thread in 1976, with the first predictive control applications and their impulse-response models identified from plant test data. A survey of industrial predictive control published in 2003 found that the number of applications had about doubled from 1995 to 1999, and that most of these controllers still used empirical dynamic models identified from test data, so shorter plant tests would have a large positive impact for their users. It found the research needs that industry perceived to be mostly engineering issues rather than algorithms, among them when a model is good enough to stop the identification test and when the added expense of a nonlinear controller can be justified [8]. In the editor's reading, the twelve chapters of this volume answered both questions for one plant at a time, and neither has a routine answer yet.

Twins run beside plants, and monitors watch them. A digital twin is a virtual representation of an asset, kept current by data and simulators, for prediction, monitoring, control and decisions [9]. Data-driven monitoring and fault diagnosis has a large literature of its own [10]. The app of Chapter 4 runs its twin beside the board it models under one controller, and Chapter 5 shows the other side: a twin whose numbers have not yet met their rig. In the editor's view a twin is only as current as its last comparison with the plant, and that comparison, with its date and its residuals, belongs in the twin's record.

Models are corrected from routine data. The performance of a predictive controller depends closely on its model, and a process drifts from the model identified at commissioning as feedstocks, instruments, equipment and operating conditions change. A step test to re-identify it disturbs closed-loop operation, while routine setpoints may stay constant for several months. One study re-identifies the model from routine operating data with a regularized regression and, in numerical experiments on a pilot-scale distillation column, corrected mismatch in static gain and in transport delay without injecting an excitation signal [11]. The predictive controller of Chapter 3 lost to the PID when its model no longer matched the splitter. In the editor's view such a method can be trusted only as far as it states which constants the routine record could not determine.

Learned models are given a physical shape. Relying on data alone does not guarantee that an identified model keeps the physical properties of the system. One study builds monotonic behavior into a neural network model, trains it on 250 points from a two-heater laboratory kit with heater inputs between 10 and 50 percent, and runs it in a predictive controller whose input reaches 65 percent, outside the training range [12]. The kit is the board of Chapter 4. In the editor's view the constraint does the work that a balance did in this volume: it fixes the sign of each gain before the data arrive.

Property models run fast enough for the loop. Chapter 6 closes its balances with the industrial formulation of steam properties, which the steam power industry uses as its standard for contracting

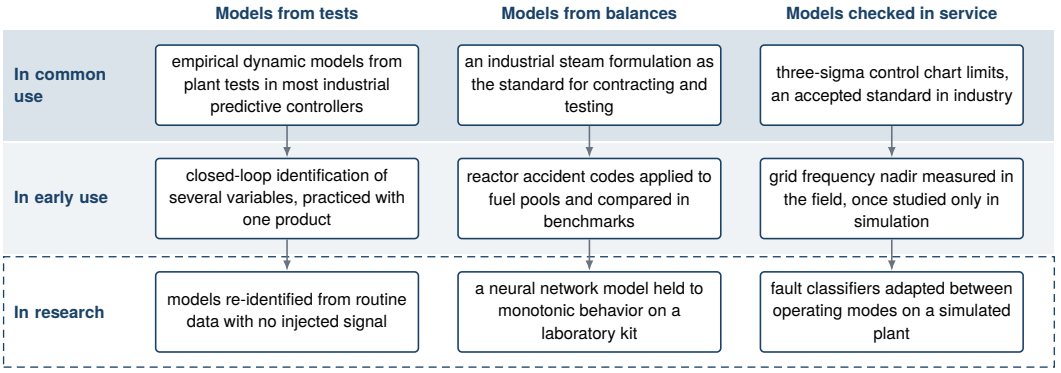


Figure 14.3 Developments in three threads. The top lane holds what the cited source calls standard or common practice, the middle lane what it reports in service at some sites or on field data, and the bottom lane what it reports from simulation, from a laboratory kit or a benchmark, in the editor's reading. Sources by thread: models from tests [8, 11], models from balances [12, 13, 15] and models checked in service [16–18]. Notice that each thread moves from a model fixed once toward one that is corrected or checked while the plant runs.

and testing [13]. A guideline from the same international association describes a spline-based table look-up intended for fast property calculations in simulations of non-stationary processes and in real-time process optimization, where the usual equations of state cost too much computing time. In flow simulations it cut computing times by factors between 6 and 10 against the industrial formulation used directly [14]. In the editor's view this brings the full balance of a steam vessel closer to the controller that needs it, so that the model of record and the model in the loop can be the same. Figure 14.3 sets developments of modeling practice in three threads.

14.4 Engineering With AI Agents

An AI agent is a language model that works toward a goal in a loop. It plans, calls tools such as a compiler, a simulator or a browser, reads the results and continues [19]. In the editor's view, agents can now carry out much of the work that used to fill the hours of a control project. They derive a model from balances or fit one to plant data. They write the simulator and its tests, code the controller, tune it against stated criteria, script commissioning checks, write the documentation and watch running loops for degradation. The apps and chapters of this series were produced that way under the editor's direction, and the account that follows this section reports what the agents got wrong in this volume and how it was caught.

In the editor's view, this changes where an engineer's value lies. When a model, a controller and a report can each be produced in minutes, producing them is no longer the scarce skill. The work moves to five things.

Quality control. An agent's output arrives as a claim: this model fits, this tuning settles in four minutes, this code implements the equation. The engineer checks claims against the plant or the simulator. A reported number is measured again on a fresh run at the stated seed. The code is read for what it clamps, what it initializes and what it reads, because a display that shows the controller's model in place of the plant looks correct until the two differ. A tuning is tried on a disturbance the agent did not test. The most useful checks are independent of the author, made by a second agent or a person who has not seen the first answer, and they are checks that can fail.

Supervision. The engineer sets the scope of each task and decides what an agent may touch. Reading a historian and proposing a tuning is one level of access. Writing a setpoint, downloading a configuration or changing an interlock is another, and it passes through the plant's management of change with a named approver. Supervision also fixes the order of work: a baseline before a change, one change at a time and a comparison that shows what moved.

Creativity. An agent answers the question it is given. Choosing the question remains the engineer's task: which loop is worth improving, whether a disturbance calls for feedforward, a cascade or a change to the process, which step test the operation can tolerate, and which comparison would show that a predictive controller earns its place over a retuned PID loop. The experiment that separates two explanations is designed by someone who can imagine both.

Process insight. A model contains what its author put in. The equipment does more. A valve sticks, the dead time of an analyzer grows when its sample flow drops, a thermowell lags, a heat exchanger fouls and an operator intervenes at the same hour each night. The engineer who knows the unit recognizes a result that is too good, asks what the model left out and knows which measurement to distrust first.

Accountability. Responsibility does not transfer to a tool. Functional safety standards assign the activities of the safety lifecycle to persons and organizations of stated competence, and they require an assessment that includes someone independent of the design team [20]. An agent may draft the hazard analysis or the test procedure. A person owns the decision to accept it, to start up and to keep running.

These five are the professional form of the habit introduced in Section 1.7: check before trusting, and know enough to check.

14.4.1 How This Volume Was Made

Each of the twelve apps was built and tested by agents. An agent then wrote the paper on it, keeping verification notes that trace every number to a run, and each paper received a referee report. At least three of the twelve reports are self-reviews by the agent that wrote the paper. Defects found along the way went into a log for each app with a severity, a cause, a suggested fix and a status. For this book one agent per chapter reran the numbers its exercises use. The editor set the scope of each round and what each agent could change, and statements that runs contradicted were left for the editor to decide.

The defect logs. The twelve logs hold 117 entries with a severity. Of these, 30 record wrong behavior of an app (one of them records the fix to another), 60 are documentation, 17 are cosmetic or display and 10 are other kinds, such as content and requested changes. The cases below are the ones that concern models.

A correlation used with the wrong definition. The pool's evaporation correlation takes the density of dry air per unit volume of moist air. The code added the vapor's own density. At the design surface temperature of 35 °C the rate came out 13.9 % low, and at the 95 °C cap that drives the loss-of-cooling case it came out 3.3 times high. The paper agent found it by measuring the rate in the running model at five temperatures and comparing it with the published correlation the code cites. The fix was first deferred, with the departure stated in the paper. Once applied it moved boiling after a loss of cooling from 16.88 to 14.74 h, while the design point still closed. A check at the design point alone would have passed the error.

Records that were not what they claimed. In the two-heater twin the exported temperature carried a fresh draw of sensor noise, independent of the one the controller acted on, so the two differed by up to about 0.6 °C at the default noise. A model fitted to that export fits the wrong pair of numbers. In the model-selection app the first-order form exported a heat loss in degrees per second under a column headed in watts, visible only in the export. Both were found while the papers were written, and the papers' runs avoided both.

Held-out data inside the training. The identification workbench estimated parameter spread with a bootstrap that drew residual blocks from the validation and test parts of each record. Held-out data thus shaped the training refits. The blocks now come only from training stretches. In the same app every exported Python model stopped with an error at line 57. The test that would have caught it was skipped whenever a Python environment was not named. A check that needs no Python now runs in every build.

A derivation against the plan. The design plan for the hydro app placed the turbine's zero at $2/T_w$, which is the magnitude of the pole. The derivation, checked with the framework's eigenvalue routine, puts it near $1/T_w$, at 0.675 rad/s at 80 MW. The app follows the derivation, and the referee report records the reason.

The controller's model was the plant. Section 1.4.9 separates the simulation model, the identified model and the controller's model. In the oil market the economic predictive controller linearized the simulator with the run's own parameters, every hidden shock in force and the true state. At one meeting, 23 of 31 hidden changes moved its request while the price it could observe stayed the same. Its model is now one frozen design and it starts from its own estimate, and the paper's results for it were measured again. In the tank and two-tank apps the controllers started from the run's own balances. There no measured run moved when the read was closed. Three logs record such reads.

A rig claim with no rig. The referee of the levitation paper found a draft that described the hardware of the surveyed rigs beyond what their publications state, and numbers that could not be regenerated from the shipped engine. The claims were softened and the numbers removed, and the chapter says plainly that none of its results has met the hardware.

The book round. The twelve chapter records list 206 rows of numeric verification, each a stated value set beside its reproduction, and every value an exercise uses reproduced. Three records found drift elsewhere: a count of checks that had grown since the paper, a recorded alarm file that no longer regenerates byte for byte, and a gain in a limitations paragraph that did not recompute, 3.68 where the fit gives 3.545 °C/%, since corrected in the paper and the chapter. The levitation Conclusion said a dead-band cycle held "whatever the tuning" while its own results showed retuning moving the cycle between ± 1.52 and ± 6.35 mm, and it was aligned with the results. The agent writing Chapter 1 had an ARX delay index off by one sample until a noise-free test showed which index fitted to machine precision.

Every error above was caught by a check that could fail: a correlation evaluated across its range, a record compared with what the controller saw, a test that actually ran, a derivation checked numerically, a hidden value moved to see what else moved. Four working rules follow. Check a model over its whole range, not at its design point. Fit on one record and judge on another. Know what each recorded column is. Read what a controller's model contains.

14.5 Gaps in Modeling and Identification

Six gaps belong to this volume's topic, and the gaps common to agent-era practice follow them in Section 14.5.1. In the editor's view none is closed. Each item gives the chapters' evidence, then the state of the literature with its sources. Why each gap is hard and what progress would look like are the editor's view. Figure 14.4 places the gaps by the chapters that met them and by the maturity of the method available, and the same numbers mark the plants in the picture that opens this chapter.

1. *Tests for slow and integrating plants.* The pool's time constant is hours, and its first hour reads as an integrator. A survey of industrial predictive control found no tool to determine whether the collected data are adequate to represent the process dynamics, so that in practice the plant is generally over-tested [8]. The gap is hard because a long lag and an integrator trace nearly the same curve early in a test, and here a long enough record means hours of a plant held off its normal duty. A routine method is needed that says, while a test runs, whether the record can yet separate a long lag from an integrator and what the tuning would lose by stopping now.

2. *Models that follow the operating point.* The tank's constants move with the level, the ball's valve gain falls from 30.4 to 1.0 mm/s per degree and the two-tank predictive model lost its advantage when the splitter moved. The Open challenge box after this list takes the gap further.

3. *The instrument as part of the model.* A thermistor set the heater's apparent time constant, and foaming water misled the flash vessel's inventory estimate with its residual small. A report on steam-generator level control found unplanned reactor trips due to that loop to be significant contributors to plant unavailability, with the measured level reacting at first in reverse to a change of inventory, and named the mismatch between steam and feedwater flow as the only true indication of an inventory change [21]. The gap is hard because the instrument and the process reach the record as one response, and a fit has no reason to split it. Methods are needed that identify an instrument's dynamics apart from the process and notice when the physics of the measurement changes.

4. *Structure selection that knows the physics.* The radiation term fitted noise and reported a physical share, and a structural zero imposed on a strong path in Chapter 12 raised the validation error from 0.1093 to 0.3892. A review of digital twins notes that sparse regression recovers a model's terms from many candidates in little computing time but performs poorly on data corrupted with noise, and that the errors of a purely data-driven model cannot be bounded [9]. The gap is hard because a wrong term and a right one both lower the objective on the record they were fitted to. Before a term is fitted, a check of identifiability and of the information a record holds about it should be as routine as the fit itself.

5. *Models for events that cannot be tested.* Load rejection on a penstock, boiling in a pool and a decade of oil prices are predicted from balances and correlations with no record of the event. The evaporation defect shows how far such a prediction can move. A status report on spent fuel pools found that the experiment database consists largely of tests made for reactors, with only one integral test targeted at a pool accident, that the available codes are intended primarily for reactor accidents, and that benchmarks between codes helped identify their limitations [15]. A standard for models and simulations requires a record of each model's domain of validation and allows extrapolation outside it only within the model's limits [22]. The gap is hard because the event that would validate the model is the event the plant exists to avoid. A practice for validating these models, against bench rigs, partial events and independent codes, is still thin.

6. *Monitoring with correlated data and rare faults.* The monitor's limit assumed independent samples and raised 1.25 detections a day, and a small feed temperature step stayed below what that rate allows. A handbook of statistical methods gives the standard three-sigma chart a false alarm every 371 points on average and notes that such properties are generally stated as if the

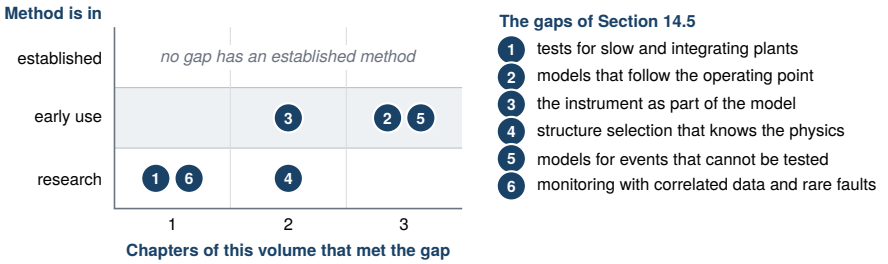


Figure 14.4 The six gaps of this section. Across is the number of chapters that each item names as evidence: one for gap 1 (Chapter 9), three for gap 2 (2, 3 and 5), two for gap 3 (4 and 6), two for gap 4 (11 and 12), three for gap 5 (7, 9 and 10) and one for gap 6 (13). Up is the maturity of the method available, the editor's reading of the sources cited in each item and in the Open challenge box. Notice that the gaps met by the most chapters, models that follow the operating point and models of events that cannot be tested, have methods in early use, while judging a test while it runs, choosing a structure by its physics and monitoring correlated data are still research.

mean and spread were known [16]. A benchmark of fault diagnosis on the same simulated process as Chapter 13 notes that learned models assume training and test data from one distribution, which is seldom verified in practice, and that a model trained in one operating mode may not generalize to another [18]. The gap is hard because correlated samples, estimated limits and a change of mode all move the false-alarm rate away from its design value at once. Detection that accounts for autocorrelation and links each statistic to a balance that can name the fault remains open.

OPEN CHALLENGE **A Model That Knows When It No Longer Applies**

Three chapters met gap 2: a tank whose constants follow its level, a valve whose gain follows its stroke and a predictive model undone by a splitter. A roadmap of nonlinear identification notes that the real world is nonlinear and time-varying, and that nonlinear identification needs a more tedious experiment, a more involved model choice and a harder estimation [3]. A survey of industrial practice found no validation method to tell whether a model had deteriorated after commissioning [8]. In the editor's view the gap is hard because routine operation seldom moves a plant far enough to show which constant has changed. Progress would be a model that flags when it leaves its validated range and refits from routine records.

14.5.1 Gaps Common to Agent-Era Practice

The final chapter of each volume lists gaps specific to its topic. Seven more are common to every topic in the series. In the editor's view none of them is closed.

1. *Verifying agent-written control code and models before they reach a plant.* Tests show that a controller behaves on the cases tested. For controllers learned from data, guarantees of stability and constraint satisfaction rest on assumptions about the model and its uncertainty, and open problems remain [23]. In the editor's view, code written by an agent has no comparable body of method yet. Practice needs a routine path from an agent's draft to code a plant can trust: independent derivation, tests of what the code reads and clamps, and trials with hardware in the loop.

2. *Trustworthy models from sparse or poor plant data.* Plant records are short, taken in closed loop, noisy and marked by unmeasured disturbances. A fitting routine returns a model from any record. What is missing is a dependable statement, delivered with the model, of where it can be trusted, and a method that says so when the data cannot identify a parameter.

3. *Safety cases and standards for agents in commissioning and operation.* Functional safety standards define a lifecycle of specified requirements, verification and assigned human roles [20]. In the editor's view, how an agent that writes configuration, runs commissioning checks or advises an operator is qualified, limited and audited under them is not yet settled.

4. *Security of agent access to control systems.* An agent with credentials to a control system is a new path into it. Industrial security standards state the requirements a control system must meet at each security level [24]. An agent that reads documents, messages or web pages can also be steered by text placed there. In the editor's view, least privilege, read-only access by default and human approval of every write are a starting point and not a solution.

5. *Benchmarks and reproducibility.* An agent asked twice may deliver two different controllers, and a result reported without its plant, seed and scoring cannot be checked. Control needs shared benchmark plants with fixed disturbances and published measures, on which agent-built and conventional designs are compared like for like. The apps of this series, with a default seed and an exported record of every run, are a start.

6. *Transfer from simulation to hardware.* A controller developed on a simulator inherits what the simulator omits. The robotics literature documents the gap between simulated and real performance and the methods used to narrow it, such as randomizing the simulated plant and adapting on the real one [25]. Process plants add slow dynamics, rare upsets and little freedom to experiment, so each transfer still needs a staged plan from the simulator to a bench rig to the unit.

7. *Keeping engineers' judgment sharp when agents do the routine work.* Automation removes the routine work through which skill is built and leaves the person the hard cases, an irony described for process operators in 1983 [26]. The same now applies to engineers. An engineer who has never tuned a loop by hand or closed a balance on paper is poorly placed to review an agent that has. Deliberate practice, of the kind the exercises in this book provide, has to be designed into education and into work.

KEY POINTS

- A balance gave each model its form, and a test gave or checked its numbers.
- A fitted constant is true of the plant only together with the method and the window that produced it.
- Where a test is unsafe or impossible the balance carries the analysis, and it is only as good as its weakest correlation.
- More parameters fit better and predict worse, so a second record decides the structure.
- A statistic says that a record departs from normal data, and a model of the plant says where and why.

References

[1] G. Pillonetto, F. Dinuzzo, T. Chen, G. De Nicolao, and L. Ljung, "Kernel methods in system identification, machine learning and function estimation: A survey," *Automatica*, vol. 50, no. 3, pp. 657–682, 2014.

[2] L. Ljung, C. Andersson, K. Tiels, and T. B. Schön, "Deep learning and system identification," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 1175–1181, 2020.

[3] J. Schoukens and L. Ljung, "Nonlinear system identification: A user-oriented road map," *IEEE Control Systems Magazine*, vol. 39, no. 6, pp. 28–99, 2019.

[4] M. von Stosch, R. Oliveira, J. Peres, and S. Feyo de Azevedo, "Hybrid semi-parametric modeling in process systems engineering: Past, present and future," *Computers & Chemical Engineering*, vol. 60, pp. 86–101, 2014.

[5] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, "Physics-informed machine learning," *Nature Reviews Physics*, vol. 3, no. 6, pp. 422–440, 2021.

[6] S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Discovering governing equations from data by sparse identification of nonlinear dynamical systems," *Proceedings of the National Academy of Sciences*, vol. 113, no. 15, pp. 3932–3937, 2016.

[7] P. Shojaee, K. Meidani, S. Gupta, A. B. Farimani, and C. K. Reddy, "LLM-SR: Scientific equation discovery via programming with large language models," arXiv:2404.18400, International Conference on Learning Representations (ICLR), 2025.

[8] 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.

[9] A. Rasheed, O. San, and T. Kvamsdal, "Digital twin: Values, challenges and enablers from a modeling perspective," *IEEE Access*, vol. 8, pp. 21 980–22 012, 2020.

[10] S. J. Qin, "Survey on data-driven industrial process monitoring and diagnosis," *Annual Reviews in Control*, vol. 36, no. 2, pp. 220–234, 2012.

[11] Y. Kono and M. Koizumi, "Model life extension for continuous process: non-invasive correction of model-plant mismatch with regularization," in *2023 European Control Conference (ECC)*, 2023, pp. 1–8.

[12] N. T. Nguyen and J. C. Tique, "Physics-constrained Taylor neural networks for learning and control of dynamical systems," arXiv preprint 2410.02258, 2024.

[13] International Association for the Properties of Water and Steam, "FAQs about IAPWS," Web page, https://iapws.org/faqs/faq2, 2026, accessed 21 November 2026.

[14] ——, "Guideline on the fast calculation of steam and water properties with the spline-based table look-up method (SBTL)," IAPWS, Stockholm, Sweden, Guideline G13-15, 2015. [Online]. Available: https://iapws.org/technical-guidance/release/SBTL

[15] OECD Nuclear Energy Agency, Committee on the Safety of Nuclear Installations, "Status report on spent fuel pools under loss-of-cooling and loss-of-coolant accident conditions: Final report," OECD Nuclear Energy Agency, Paris, Report NEA/CSNI/R(2015)2, 2015. [Online]. Available: https://www.oecd-nea.org/upload/docs/application/pdf/2021-02/csni-r2015-2.pdf

[16] 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/

[17] J. H. Eto, J. Undrill, P. Mackin, R. Daschmans, B. Williams, B. Haney, R. Hunt, J. Ellis, H. Illian, C. Martinez, M. O'Malley, K. Coughlin, and K. H. LaCommare, "Use of frequency response metrics to assess the planning and operating requirements for reliable integration of variable renewable generation," Lawrence Berkeley National Laboratory, Berkeley, CA, Report LBNL-4142E, Dec. 2010.

[18] E. Fernandes Montesuma, M. Mulas, F. Ngolè Mboula, F. Corona, and A. Souloumiac, "Benchmarking domain adaptation for chemical processes on the Tennessee Eastman process," arXiv preprint 2308.11247v2, 2024, published in Machine Learning and Principles and Practice of Knowledge Discovery in Databases, Communications in Computer and Information Science, Springer, 2026, pp. 307–322, doi:10.1007/978-3-032-25305-7_21. [Online]. Available: https://arxiv.org/abs/2308.11247

[19] L. Wang, C. Ma, X. Feng, Z. Zhang, H. Yang, J. Zhang, Z. Chen, J. Tang, X. Chen, Y. Lin, W. X. Zhao, Z. Wei, and J. Wen, "A survey on large language model based autonomous agents," *Frontiers of Computer Science*, vol. 18, no. 6, p. 186345, 2024.

[20] International Electrotechnical Commission, "Functional safety – Safety instrumented systems for the process industry sector – Part 1: Framework, definitions, system, hardware and application programming requirements," IEC 61511-1:2016, Edition 2.0, 2016.

[21] A. M. H. Basher and J. March-Leuba, "Development of a robust model-based water level controller for U-tube steam generator," Oak Ridge National Laboratory, Oak Ridge, TN, Tech. Rep. ORNL/TM-2001/166, Sep. 2001.

[22] National Aeronautics and Space Administration, "Standard for models and simulations," Office of the NASA Chief Engineer, Washington, DC, NASA Technical Standard NASA-STD-7009B, 2024, approved 2024-03-05, standards.nasa.gov.

[23] L. Brunke, M. Greeff, A. W. Hall, Z. Yuan, S. Zhou, J. Panerati, and A. P. Schoellig, "Safe learning in robotics: From learning-based control to safe reinforcement learning," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 5, pp. 411–444, 2022.

[24] International Electrotechnical Commission, "Industrial communication networks – Network and system security – Part 3-3: System security requirements and security levels," IEC 62443-3-3:2013, Edition 1.0, 2013.

[25] W. Zhao, J. P. Queralta, and T. Westerlund, "Sim-to-real transfer in deep reinforcement learning for robotics: A survey," in *Proc. 2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2020, pp. 737–744.

[26] L. Bainbridge, "Ironies of automation," *Automatica*, vol. 19, no. 6, pp. 775–779, 1983.

14.6 Exercises

14.1 Discussion · 8 min

An agent hands you a first-order-plus-dead-time model of a new heated tank, fitted by the two-point method to one heater step: gain, time constant and a dead time of about one tenth of the time constant. The tank has no transport delay and its thermometer sits in a thermowell. Using Chapter 3 and Chapter 4, list what you would ask before the model is used to tune a loop, and say what each question could reveal.

14.2 Discussion · 8 min

An agent fits the radiative form of Chapter 11 to one step test, reports a 5.4 % lower objective than the convective form and recommends the radiative model "because it contains more physics". Write the reply. Say what evidence the recommendation lacks, what the chapter measured, and which experiment would settle the question on a physical board.

14.3 Discussion · 8 min

The evaporation defect of Section 14.4.1 made the rate 13.9 % low at the pool's design temperature and 3.3 times high at the 95 °C cap. Explain why tests at the design point would not have caught it, why the error mattered most where it was largest, and propose a routine check for any empirical correlation an agent codes into a model.

14.4 Discussion · 7 min

A utility asks for a model of a hydroelectric unit to tune its governor, and no field step test on the gate is allowed. Using Chapter 7 and Section 14.2, describe how the model should be built, which results must be checked numerically, and what the derivation in the chapter caught that a plan written from memory got wrong.

14.5 Discussion · 7 min

The app of Chapter 4 runs a twin beside the board it models, while the twin of Chapter 5 has not yet met its rig. An agent proposes to commission a new rig's controller on its twin alone. Discuss what evidence the twin needs first, what record should be kept with it, and which defects of Section 14.4.1 show why the record matters.

14.6 Discussion · 7 min

An agent deploys a principal component monitor on a unit with the limit set for a 1 % false-alarm rate per sample. In the first week it raises more than one alarm episode a day on normal operation. Using Chapter 13, explain the likely cause, what the agent should measure before changing anything, and how a balance could make the monitor more useful to the operators.