INTERACTIVE CONTROL ENGINEERING
Principles and Applications

Volume 4 Override, Split-Range, and Constraint Control

CHAPTER 1

Introduction: Selectors, Overrides, and Shared Actuators

Pages 1–23

Edited by John D. Hedengren

How to cite this chapter. "Introduction: Selectors, Overrides, and Shared Actuators," in *Override, Split-Range, and Constraint Control*, John D. Hedengren, Ed., vol. 4 of *Interactive Control Engineering: Principles and Applications*. Advanced Process Solutions, LLC, 2026, ch. 1, pp. 1–23.

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/v04-override-constraint/

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

CHAPTER 1

Introduction: Selectors, Overrides, and Shared Actuators



The chapters 2 compressor near surge 3 turbofan engine 4 combine harvester 5 cereal extruder 6 electrode dryer 7 directional drill 8 product startup 9 trigeneration plant 10 CO_2 power cycle 11 tunnel boring shield 12 fan car 13 quadruped robot 14 currency peg

Notice that the units from the compressor to the drill each guard one main actuator with several limits, while most from the trigeneration plant to the robot share one task among several actuators. Illustration, not to scale.

IN PLAIN TERMS **Which Goal Gets the Handle**

The main idea. A plant usually runs best close to a limit, such as a temperature it must not pass or a valve that is already wide open. The methods in this book decide which goal gets a shared handle, and which of several handles moves first.

The challenge. A guard that waits unused for hours must still take over smoothly at the right moment. A supply that can be spent quickly, like a cold store or cash in the bank, runs out, so it becomes a limit of its own.

Why it matters. The same few ideas protect jet engines, drills, power plants, robots and a country's money.

THREE WAYS TO READ THIS CHAPTER

Explore	8 min	This page, Section 1.2 with its timeline and two History boxes, the Did you know? boxes in Sections 1.1, 1.4.3 and 1.4.7, Figure 1.10, and Key points in Section 1.8. No equations.
Study	50 min	The whole chapter in order, with each number of the reactor example worked, then Exercises 1 to 10.
Practice	5 min	This page, Section 1.4.10 with Table 1.4, Section 1.5 with Figures 1.6 and 1.8, and Section 1.8 with Figure 1.10.

1.1 The Question This Volume Answers

A controller is usually taught with one measurement, one actuator and a setpoint well inside the actuator's range. Plants do not stay there for long. A cooling valve opens wide, a motor reaches its rating, a pressure approaches what the ground or the vessel can hold, and the most profitable operating point turns out to lie on one of these limits. This volume answers two questions that arise there. When several variables must each stay inside a limit and only one actuator can move them, which variable gets the actuator? When several actuators can move one variable, which one should move, and for how long?

The first question is answered by selectors. A low or high selector passes one of several controller demands to a single actuator, so that the normal controller runs the plant until a limit comes near and an override controller takes over to hold it. The second question is answered by split range, mid-ranging and control allocation, which share one demand among actuators that differ in range, speed and price. A third kind of actuator joins the two questions. A reserve of chilled water, compressed gas, cash or foreign currency can be spent quickly but not forever, so an actuator that depletes is also a limit that the plant approaches with time.

DID YOU KNOW? Two Selectors Keep a Ship's Boiler From Smoking

A ship's boiler is tuned for efficiency at sea but must run with excess air in harbor, where black smoke brings heavy penalties. A maximum and a minimum selector around its air and oil flow loops keep the air ahead of the fuel as the power demand rises or falls. Selectors commonly implement logic in engines and power systems [1].

Thirteen chapters test these structures, each on a plant with an app that runs in a browser. Chapter 2 and Chapter 3 protect machines that can damage themselves within seconds, a compressor near surge and a jet engine whose single fuel valve answers to six limits. Chapters 4 to 7 push throughput against limits that move: a combine harvester, a cereal extruder, a battery electrode coater and a horizontal drill. Chapter 8 runs a company against a cash floor. Chapters 9 to 11 return a fast actuator to the middle of its range in a trigeneration plant, a supercritical carbon dioxide power cycle and a tunnel boring machine. Chapters 12 and 13 allocate forces among tires, fans, thrusters and feet, and Chapter 14 defends an exchange rate with a vault that empties.

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

- Name the limits that are active at the plant's best operating point, and state what each degree of back-off from a limit costs.
- Build an override with a low or high selector and external reset, and predict when the override takes over.
- Choose the split point of a split-range controller from the actuator gains, and say when mid-ranging should replace it.
- Tune a valve-position controller from the gains of its two actuators, and state what its resting position costs.
- Write an allocation rule for actuators with different limits and prices, and compute how long an actuator that depletes will last.

This chapter develops each skill on one small example and names the chapters that use it. PID tuning is not repeated here. Volume 2 develops it, and every loop in this chapter is a PI tuned from its model by the rules of that volume. Standard texts treat selectors, overrides and split range in fuller detail [2–4].

1.2 A Short History

The milestones of Figure 1.1 follow selectors, overrides and shared actuators through the kinds of plant in this volume. Much of the development of PID control occurred in the power and process industries, where many of the bottom-up architectures around it also appeared, among them selector and midranging control [1]. Pneumatic control became the standard of the oil, gas and chemical industries during the 1940s [5], and an override built from such controllers needed a way to stop the integral of the controller it did not pass.

HISTORY 1931 Onward: A Reset Line From the Valve Stem

An early pneumatic controller left an offset, which the operator removed by resetting the setpoint by hand. A circuit that copied that correction, called automatic reset, is the integral action of today, and a controller with it was marketed in 1931 [6]. Some pneumatic controllers come with an option for external reset: the reset bellows takes its signal from where the final effect of the output is seen, such as a position transmitter on the valve stem. If the stem meets a stop or a jam, that signal freezes and integration ceases instead of winding up [16]. Section 1.4.4 feeds the override's integral from the selector's output in the same way.

Engines carried caps and floors from the start. The first jet engine built in the United States, in 1942, had a governor whose fuel metering valve was given a minimum-flow stop against flameout

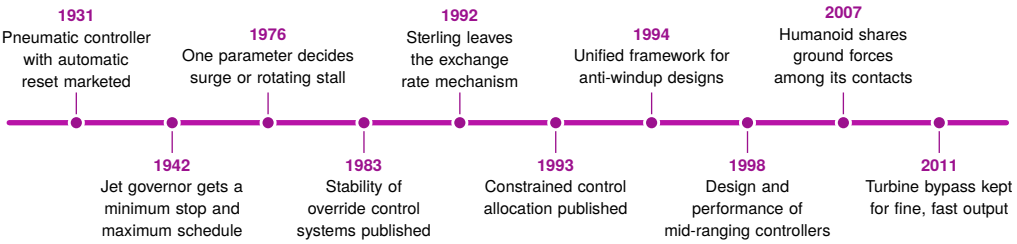


Figure 1.1 Milestones of override, selector and constraint control in the plants of this volume. The dates are from [6] (1931), [7] (1942), [8] (1976), [9] (1983), [10] (1992), [11] (1993), [12] (1994), [13] (1998), [14] (2007) and [15] (2011). Notice that a currency, a robot and a power cycle sit among the engines and the theory: the same structures served them all. Events are evenly spaced, not to a time scale.

and a maximum-flow schedule against overtemperature [7]. For compressors, a study of 1976 found one nondimensional parameter that decides whether a compression system at its stall line falls into surge or into rotating stall [8]. Chapter 2 calls it B.

Papers on the stability of override control systems appeared in 1983 [9], on constrained control allocation in 1993 [11] and on the design and performance of mid-ranging controllers in 1998 [13]. The same structures served far from process plants. On one day in September 1992 a central bank reportedly spent as much as half its foreign exchange reserves defending its band, and its currency left the exchange rate mechanism at the end of the day [10]. A humanoid robot of 2007 shared the ground force it needed among any number of contact points [14]. A 2011 report on supercritical carbon dioxide cycles kept the turbine bypass, the simplest and fastest handle, for fine and fast adjustments of the generator output [15].

> ⧖ HISTORY **1994: Every Anti-Windup Scheme in One Frame**
>
> Windup was first met in PI and PID controllers whose actuators saturate. An early remedy, an algorithm for direct digital control, has been called anti-reset windup, back calculation and tracking, or integrator resetting. Integrator windup was later seen as a special case: any controller with slow or unstable modes winds up when its actuator is limited. An override adds a second cause. When the plant input is chosen among parallel controllers, those not chosen are wrongly updated, and the switch bumps the plant input. A framework published in 1994 showed every known linear anti-windup and bumpless transfer scheme to be a special case of one design: the linear controller first, then a compensation for the limits [12].

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. Controller settings are K_c and τ_I, and a process model has a gain K_p, a time constant τ_p and a dead time θ_p. A tag such as TIC or FIC names a temperature or flow controller, and TC-2 names the override of this chapter. Following the shared framework, v is the demand a controller computes before any limit and u is the value applied to the actuator, after the clamp, the rate limit or the selector. The tracking time of back-calculation is T_t, equal to τ_I unless a chapter says otherwise. A split-range controller divides its output c at the split point s, both in percent. A valve-position controller (VPC) holds a fast actuator at its resting position r. Heat duties are Q, in kW, and the charge of a store is E.

Each chapter keeps the time unit of its plant, from a hundredth of a second in the turbofan to weekly decisions in the currency peg. The running example of this chapter uses minutes. A few symbols need care across chapters. The shared framework uses a for the controller action and h for

the integration step, so neither letter is used here for anything else. Several chapters use v for a speed, the ground speed in Chapter 4, the line speed in Chapter 6 and the advance in Chapter 11. Chapter 2 writes s_k for the accumulator of its controller, and Chapter 8 writes z for an override level, which is a split-range input.

1.4 Limits, Selectors, and Shared Actuators

1.4.1 The Running Example

The running example is a continuous reactor that converts a liquid feed in an exothermic reaction. The feed F, in kg/min, is the product the plant sells, and each kg/min releases 25 kW. Two coolants remove the heat. Cooling water passes a jacket and removes up to 400 kW, 4 kW per percent of its valve stroke. It is cheap, but it reaches the reactor 1.5 min after its valve moves, the time the water takes to cross the jacket circuit. Chilled water from a cold store removes up to 100 kW, 1 kW per percent of stroke, and acts at once. The store holds 100 kWh, enough for one hour at full flow. The reactor temperature T follows Equation (1.1).

$$\tau_r \frac{\mathrm{d}T}{\mathrm{d}t} = -(T - 80) + \frac{\Delta H\, F - Q_w - Q_c + d}{UA} \tag{1.1}$$

where $\tau_r = 10\,\mathrm{min}$, $\Delta H = 25\,\mathrm{kW}$ per kg/min, $UA = 5\,\mathrm{kW/^\circ C}$, Q_w and Q_c are the cooling-water and chilled-water duties, and d is a load disturbance in kW. The thermometer adds 0.5 min of dead time. Each path to the measured temperature is therefore first order with $\tau_p = 10\,\mathrm{min}$. The feed path has a gain of 5 °C per kg/min and a dead time of 0.5 min, the cooling-water path −0.8 °C/% and 2 min, and the chilled-water path −0.2 °C/% and 0.5 min.

At design the feed is 10 kg/min, its 250 kW is carried by the cooling water at 62.5 % of stroke, and the temperature is 80 °C. The reactor trips at 90 °C. Two controllers act on the temperature, as shown in Figure 1.2. TIC-1 holds 80 °C with the coolants. TC-2 is an override on the feed with a setpoint of 85 °C. It does nothing while the cooling copes and cuts the feed when the cooling cannot. Both are PI controllers tuned by the SIMC rule of Volume 2 with $\tau_c = \theta_p$ and executed every 0.1 min in the velocity form of Section 1.6.2. TIC-1 has $K_c = 2.5\,\%/^\circ\mathrm{C}$ and $\tau_I = 10\,\mathrm{min}$ on the split range of Section 1.4.6. TC-2 has $K_c = 2\,\mathrm{kg/min}$ per °C and $\tau_I = 4\,\mathrm{min}$.

1.4.2 Constraints as Objectives

The reactor earns on its feed, so the operator would run it as fast as the cooling allows. At steady state Equation (1.1) gives the feed that a temperature T and the two duties can carry, which is Equation (1.2) with the cooling water at full stroke and the store unused.

$$F = \frac{Q_w + Q_c + UA\,(T - 80)}{\Delta H} = 16 + 0.2\,(T - 80)\ \mathrm{kg/min} \tag{1.2}$$

Two limits hold the best steady point. The cooling-water valve is wide open, and the temperature sits as high as the trip allows. Neither value is chosen for its own sake. Each is a constraint that the economics press the plant against, so the constrained variable becomes a controlled variable and its limit, less a margin, becomes its setpoint. Such a constraint is called active. Which constraints are active depends on prices, feed and disturbances, and a control structure must switch its controlled variables when the active set changes [17, 18].

The temperature cannot be held at the trip itself, because every upset would trip the reactor. Its setpoint is backed off from the limit by the largest excursion the control leaves, plus a margin. Each degree of back-off costs 0.2 kg/min here, 1.2 % of the 17 kg/min that the override setpoint of 85 °C allows. A structure that overshoots less permits a setpoint nearer the limit, which is how better control is paid for in throughput [19]. The runs of Section 1.4.3 show the size of the effect. With external reset the override overshoots its setpoint by 0.95 °C, and without it by 2.87 °C. With the setpoint

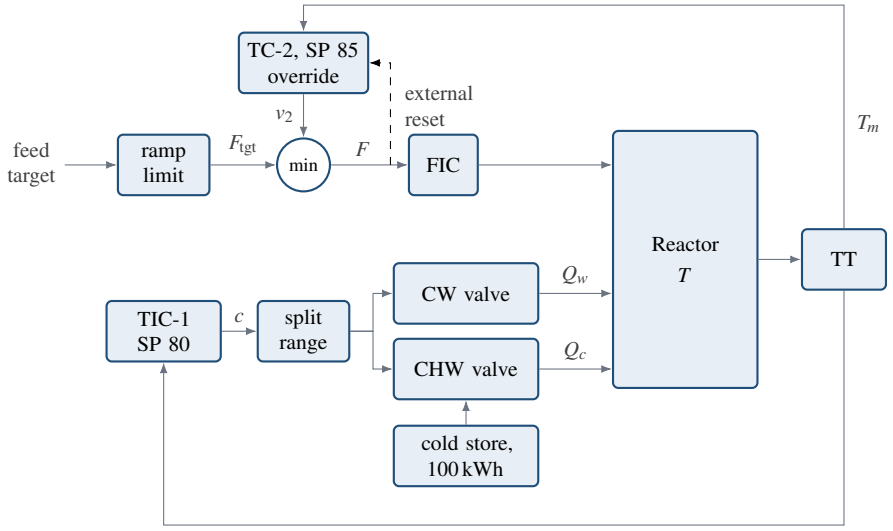


Figure 1.2 The Running Example. A Low Selector Passes the Smaller of the Ramped Feed Target and the Demand of the Temperature Override TC-2 to the Feed Flow Controller. TIC-1 Moves Cooling Water (CW) and Chilled Water (CHW) Through a Split Range. The Dashed Line Returns the Applied Feed to TC-2 (External Reset). Section 1.4.7 Replaces the Split Range With Mid-Ranging.

moved to 88 °C, the first peaks at 88.67 °C and earns 17.6 kg/min, while the second crosses the trip for 2.2 min.

The reactor shows the two shapes that every constraint problem in this volume takes. With one feed and several limits to respect, there are more variables to protect than actuators to move, and a selector decides which variable gets the actuator. Sections 1.4.3 to 1.4.5 treat that case. With two coolants for one temperature, there are more actuators than controlled variables, and split range, mid-ranging or an allocation rule decides which actuator moves. Sections 1.4.6 to 1.4.9 treat that case.

Three chapters push a throughput against its active constraints. Chapter 5 ramps its feed while every predicted constraint keeps a slack in units of its own margin, and declares a target infeasible with the binding constraint named. Chapter 6 pushes the line speed to a vapor target of 20 % of the lower explosive limit, under an override at 21 and a cap at 25 %. Chapter 7 finds that the soil chooses the binding limit, torque in clay, the mud ratio in sand and thrust in gravel. Chapter 3 measures the price of a surge margin in seconds of acceleration, and Chapter 10 argues that a plant should be run until every constraint binds, so that the operator knows which one comes first.

1.4.3 High and Low Selectors and Override

A selector passes one of several demands to a single actuator. A low selector passes the smallest, so each of its inputs acts as a cap: the actuator may move no higher than any input allows. A high selector passes the largest, so each input acts as a floor. On the reactor the feed is capped by the operator's target and by the override, as shown in Equation (1.3).

$$F = \min\left(F_{\text{tgt}},\ v_2\right), \qquad v_2 = K_c\, e_2 + I_2, \qquad e_2 = 85 - T_m \tag{1.3}$$

where T_m is the measured temperature and I_2 is the override's integral term. The override is reverse acting, since more feed raises the temperature. Its setpoint lies inside the hard limit and above the normal operating temperature. In healthy operation its demand is large and the selector ignores it, and its quality is judged by the worst excursion it allows, not by an integral of error that it never intends to remove.

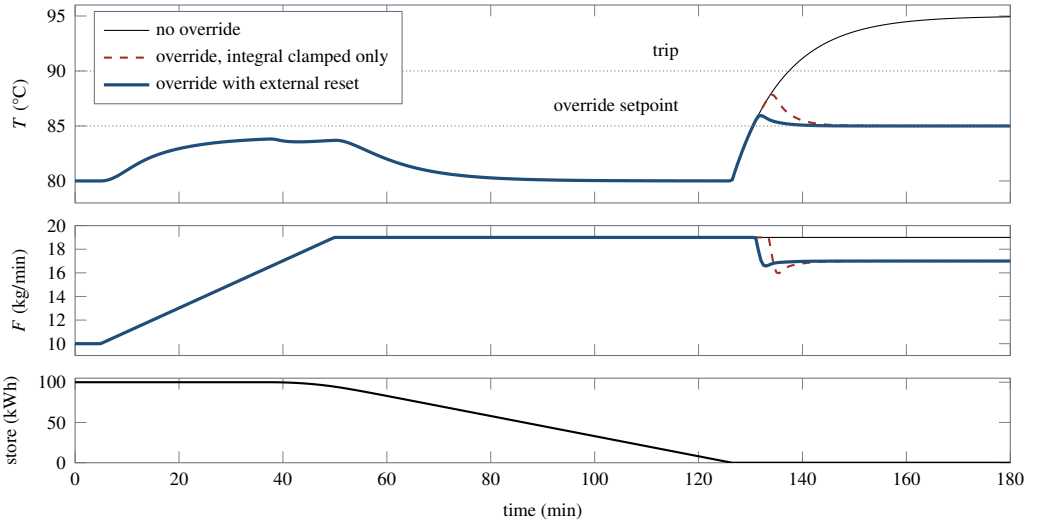


Figure 1.3 One Shift on the Reactor. The Feed Target Ramps From 10 to 19 kg/min Between 5 and 50 min, and the Cold Store Empties at 126.4 min. Thin Solid: No Override. Dashed: Override Whose Integral Is Only Clamped to Its Output Range. Thick Solid: Override With External Reset. The Store Charge Is the Same in All Three Runs.

When caps and floors act on one actuator, their order matters. The floors are applied after the caps when what a floor protects outranks what the caps protect, as in $u = \max\left[u_{\text{floor}}, \min(v_1, \ldots, v_n)\right]$. A selector can also be realized without a separate block, by setting the override's upper output limit to the other demand, so that the clamp of Equation (1.14) performs the selection. Where two demands can sit close together, a small margin that an incumbent must be undercut by keeps noise from passing the actuator back and forth.

Figure 1.3 runs the reactor through one shift. From 5 min the feed target ramps from 10 to 19 kg/min at 0.2 kg/min per minute. The cooling water reaches full stroke at 39 min, and the split range moves the chilled water in to carry the remaining 75 kW. The store empties at 126.4 min. From then on the cooling falls 75 kW short. Without an override the temperature passes the trip and heads for 95 °C, above 90 °C for 42.6 min of the run. With the override and external reset, TC-2 takes the feed at 131.0 min, when the measured temperature reaches 85.02 °C, and settles the feed at the 17 kg/min that Equation (1.2) predicts. Table 1.1 gives the measures. Without external reset, the override's integral has run to its upper limit of 25 kg/min during the hours of healthy operation. It must fall below the applied 19 kg/min before the selector passes it, so the override takes over 2.6 min later, at 87.31 °C, and the temperature peaks at 87.87 °C.

Most chapters of this volume have a selector or a clamp that acts as one. Chapter 3 takes the least of a speed governor, an acceleration schedule, a turbine temperature limiter, two overspeed limiters, a fuel rate limit and a stall cut, and then applies two floors that keep the flame lit, because a flameout outranks every limit a cap protects. Chapter 4 passes the lowest of five ground-speed demands, with an incumbent margin of 0.072 km/h. Chapter 5 realizes a high selector on the screw speed and a low selector on the feed as output limits of its two overrides. Chapter 7 and Chapter 11 take the least of several limits on the thrust and the advance, and Chapter 9 low-selects its power loop against an exhaust limiter on one fuel valve. Chapter 2 has the simplest form, a single protective loop whose output rests on its closed clamp while the machine is healthy.

Table 1.1 The Override After the Store Empties, by Anti-Windup Scheme and Tracking Time T_t. Takeover Time and the Measured Temperature Then, Peak Temperature, and Time Above 85.5 °C in the 180 min Run

Override	Takeover (min)	T_m then (°C)	Peak (°C)	Above 85.5 (min)
None	—	—	94.93	49.0
Integral clamped only	133.6	87.31	87.87	9.0
External reset, $T_t = 0.1\,\tau_I$	127.5	80.83	85.00	0
External reset, $T_t = 0.25\,\tau_I$	128.3	81.92	85.00	0
External reset, $T_t = 0.5\,\tau_I$	129.3	83.17	85.00	0
External reset, $T_t = \tau_I$	131.0	85.02	85.95	3.0
External reset, $T_t = 2\,\tau_I$	133.7	87.38	87.97	9.3
External reset, $T_t = 4\,\tau_I$	138.1	90.09	90.40	15.5

✦ DID YOU KNOW? A Hard and a Soft Override on One Well Pump

A well pump on a variable-speed motor holds the water pressure for its customers. A level switch that stops it when the well runs low saves the seals but cuts off the water, a hard override. A level controller low-selected against the pressure controller only slows the pump, a soft override [16].

1.4.4 External Reset

A controller that the selector does not pass still computes a demand, and its integral still sees an error. Left alone, the integral of the override in the reactor runs upward for as long as the temperature sits below 85 °C, which is the windup of Table 1.1. The cure is to feed the integral with the output actually applied. The shared framework already does this through back-calculation, Equation (1.15), once the applied output u is taken after the selector. For a PI with no bias, where the running sum equals $K_c e$, the integral obeys Equation (1.4).

$$\frac{\mathrm{d}I}{\mathrm{d}t} = \frac{K_c}{\tau_I}\,e + \frac{u - (K_c e + I)}{T_t} \xrightarrow{T_t = \tau_I} \frac{\mathrm{d}I}{\mathrm{d}t} = \frac{u - I}{\tau_I} \tag{1.4}$$

With $T_t = \tau_I$ the error drops out. The integral is the applied output passed through a first-order lag of time constant τ_I, and the demand is $v = K_c e + I$. This is external reset feedback [3, 20]. While the controller holds the actuator, $u = v$, and the lag of its own output reproduces integral action. While another controller holds it, the integral follows the applied output and the demand rests at its proportional distance above it, $v - u \to K_c e$. The override therefore takes over when its own error changes sign, at its setpoint, without a bump and without delay. On the reactor at 83 °C, for example, TC-2 asks for $19 + 2 \times 2 = 23\,\mathrm{kg/min}$ while 19 kg/min is applied.

A different tracking time shifts the takeover. For a steady applied output the demand rests at $v - u = K_c e\,T_t/\tau_I$, and while the error falls the integral trails its own target by about T_t. A short T_t lets the override take over on its rate of approach, before its limit is reached. A long one lets it wind partly and take over late. Table 1.1 measures both. At $T_t = 0.1\,\tau_I$ the override cuts the feed when the measured temperature is still 80.83 °C, which here costs 1.9 kg of feed and removes the overshoot. At $4\,\tau_I$ it waits until 90.09 °C and the reactor trips. Only $T_t = \tau_I$ takes over at the limit itself. Anti-windup schemes of this kind are compared in [12] and treated for selector structures in [4].

Chapter 3 measures the same trade on a jet engine. Its plan asked for every integrator to be reset to the applied fuel each scan, and with that reset the temperature limiter won the valve on its rate of approach 215 K under its limit. Tracking times of 1.5 and 2 integral times let the challenge's turbine temperature reach 973 and 1012 °C, so the chapter keeps $T_t = \tau_I$. Chapter 4 writes the lag of Equation (1.4) explicitly, and Chapter 11 and Chapter 7 build each limit as an integrator on the request actually selected. In Chapter 2 the clamp is the other input of the selector. With the valve shut, Equation (1.4) gives $I_k = (1 - \Delta t/\tau_I)\,I_{k-1}$, so the integral decays to zero and the valve opens

at the first execution after the flow crosses its setpoint. Chapter 8 documents the opposite case. Its runway override acts on the request after the growth loop, so that loop's integral keeps running while the override holds the spending down. A cascade needs the same care one level up. In Chapter 4 the loss supervisor may lower the feed setpoint while an override holds the speed but may not raise it.

1.4.5 Saturation and Rate Limits

An actuator at its limit has no more to give, and no tuning changes that. After the store empties, TIC-1 asks for more cooling than exists, its output reaches 100 % and back-calculation holds its integral there. The plant then needs a different actuator, here the feed. A protective actuator that saturates is a sizing statement, as Chapter 2 shows with a recycle valve whose full opening cannot hold the flow, and one that strokes too slowly fails even when the controller's decision is right.

The empty store hides a subtler case. The chilled-water valve still moves, but it no longer cools, so the controller sees an actuator whose gain has fallen to zero without reaching its clamp. Back-calculation cannot react to a limit it is not told about. Chapter 13 handles the same problem by back-calculating each loop from the force that its feet actually delivered, and Chapter 12 reads the thrust its tank can still supply from the tank pressure.

A rate limit restricts how fast an actuator or a setpoint may move. A PI loop that must deliver an output rising at rate ρ lags the ramp by the error of Equation (1.5), the steady ramp error of a loop with integral action.

$$e_\rho = \frac{\rho\,\tau_I}{K_c} \tag{1.5}$$

A feed ramp of r kg/min per minute asks TIC-1 for $25r/5 = 5r$ percent of its output per minute, so $e_\rho = 20r$ °C. The ramp of 0.2 kg/min per minute lags by 4 °C in principle, and the simulated temperature reaches 83.82 °C before the ramp ends. A ramp of 0.5 kg/min per minute would lag by 10 °C, and without an override the temperature would reach 87.34 °C during the ramp, past the override's setpoint. The ramp limit is a constraint handled open loop. It should be chosen from the margin to the override, here $r \le (85-80)/20 = 0.25\,\mathrm{kg/min}$ per minute, and a faster demand hands the job to the selector.

Chapter 3 places its fuel rate limit among the caps of its selector. Chapter 4 applies acceleration limits after the selector, where no PI loop sees them, and Chapter 2 applies one inside the controller and again at the valve. Chapter 8 and Chapter 14 shape every decision with a quantum and a calendar, and the currency peg also with a move cap, so that a request can change only at a board meeting or a weekly decision and only by whole steps.

1.4.6 Split Range

A split-range controller drives several actuators in sequence from one output. TIC-1 uses the first part of its output c for the cooling water and the rest for the chilled water, as in Equation (1.6).

$$u_w = \frac{100\,c}{s}, \qquad u_c = \frac{100\,(c-s)}{100-s}, \qquad \text{each clipped to 0–100 \%} \tag{1.6}$$

The cheap actuator comes first, so the store is touched only when the cooling water is wide open. The split point sets the gain the loop sees on each segment. One percent of output moves the cooling water by $100/s$ percent and the chilled water by $100/(100-s)$ percent. The two segments give the same duty per percent of output when the split point is chosen by Equation (1.7).

$$s = 100\,\frac{Q_{w,\max}}{Q_{w,\max}+Q_{c,\max}} = 100 \times \frac{400}{500} = 80\,\% \tag{1.7}$$

Both segments then give 5 kW per percent. A split at 50 % would give 8 and 2 kW per percent, as shown in Figure 1.4, and a loop tuned on the first segment would be four times too slow on the second. A gap between the segments leaves a band of output that moves nothing, a dead zone in which the loop can cycle, and an overlap doubles the gain where both move.

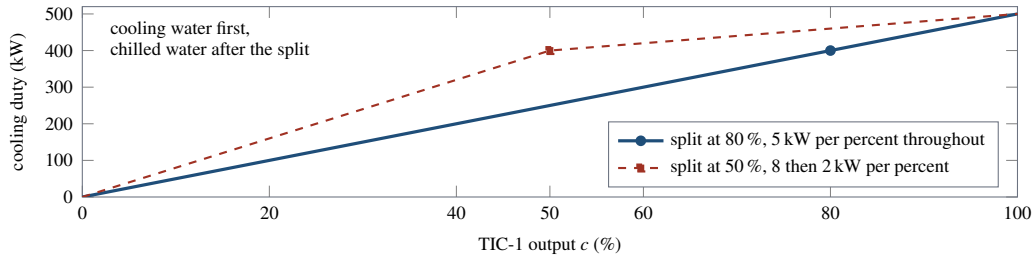


Figure 1.4 Cooling Duty Against the Output of TIC-1. Thick Solid With a Circle: Split Point at 80 % From Equation (1.7). Dashed With a Square: Split Point at 50 %. The Marker Is the Split Point.

On the shift of Figure 1.3, the split range brought the chilled water in by itself when the cooling water reached full stroke, and the output of TIC-1 settled at 95 %. That is the strength of the structure. It extends the range of one controller across several actuators in a fixed order of preference [21]. Its weakness is that it moves one actuator at a time, always the first one first, and assumes they act at similar speed. Here they do not. The cooling water reaches the reactor 1.5 min later than the chilled water, and the controller crosses from one to the other with the same tuning. Generalized forms give each actuator its own controller and pass a baton between them [22].

Three chapters use the structure. Chapter 5 heats or cools each barrel zone from one output between -1 and 1, bands above zero and cooling water below it. Chapter 8 splits the level of its runway override over three ranges, so that marketing and investment go first, then the credit line is drawn, and only then is research cut. Chapter 12 launches its car as a split range in time, with the fans spooled first, the tires pushing next and the thrusters supplying only what the tires cannot.

1.4.7 Mid-Ranging and Valve-Position Control

When the cheap actuator is slow, it should not be the first to answer. In mid-ranging the fast actuator holds the controlled variable, and a slow valve-position controller moves the second actuator until the fast one is back at a resting position r [13, 23]. On the reactor, TIC-1 moves the chilled water alone, with $K_c = 50\,\%/^{\circ}\mathrm{C}$ and $\tau_I = 4\,\mathrm{min}$ from the same SIMC rule, and the VPC moves the cooling water by integral action on the chilled-water demand v_c, as shown in Equation (1.8).

$$\frac{\mathrm{d}u_w}{\mathrm{d}t} = K_I\,(v_c - r), \qquad r = 10\,\%, \qquad K_I = 0.05\,\mathrm{min}^{-1} \tag{1.8}$$

Through the closed temperature loop, each percent that the cooling water opens is cancelled by four percent less chilled water, so the VPC sees a static gain of 4. An integrator on a static gain closes with the time constant $\tau_h = 1/(4K_I) = 5\,\mathrm{min}$, chosen well above the 2 min that the cooling water takes to act.

The handover has an exact cost. After a load step D that the cooling water can carry, the VPC must move it by $D/4$ percent, and by Equation (1.8) that move equals K_I times the integral of $v_c - r$. While the chilled water is not clamped, its energy above the resting draw is therefore fixed by Equation (1.9), whatever the tuning of TIC-1.

$$E_h = \int \left(Q_c - Q_{c,r}\right) \mathrm{d}t = \frac{K_{ch}}{K_w K_I}\, D = \tau_h D \tag{1.9}$$

where $K_w = 4$ and $K_{ch} = 1\,\mathrm{kW}$ per percent and $Q_{c,r}$ is the resting draw. A step of 50 kW costs 250 kW min, about 4.2 kWh, and the simulation gives the same 250.0 kW min.

Figure 1.5 and Table 1.2 compare the two structures. Upward, mid-ranging cuts the peak from 2.96 to 0.95 °C and the IAE nearly eightfold, because the fast actuator answers first and the slow one takes over behind it. Downward the gain is smaller. Resting at 10 kW, the chilled water can give back

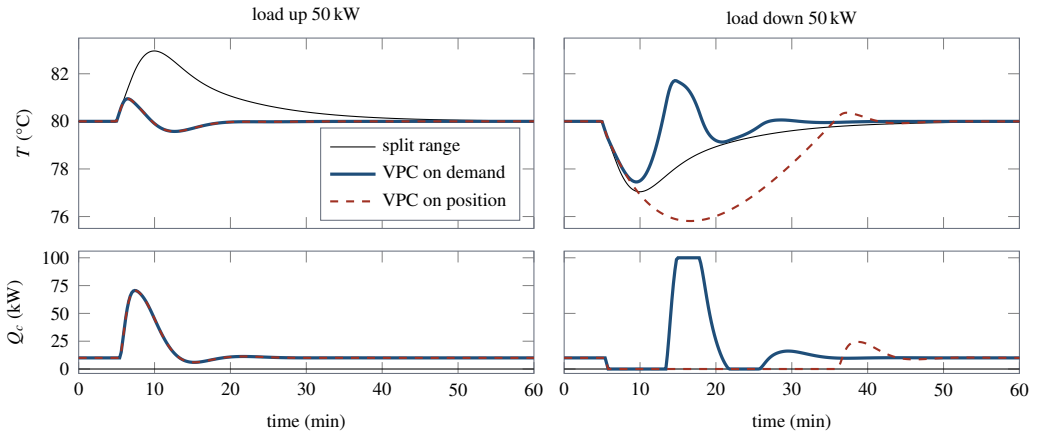


Figure 1.5 A Load Step of 50 kW at 5 min, Up (Left) and Down (Right), at the Design Feed. Top: Reactor Temperature. Bottom: Chilled-Water Duty. Thin Solid: Split Range. Thick Solid: Mid-Ranging With the VPC Watching the Unclamped Chilled-Water Demand. Dashed: VPC Watching the Clamped Valve Position, Identical to Thick Solid on the Upward Step.

Table 1.2 Split Range Against Mid-Ranging on the Steps of Figure 1.5, Over 60 min. Peak Deviation in °C, IAE in °C min, Chilled Water Above the Resting Draw in kW min

	Load up 50 kW			Load down 50 kW	
Structure	Peak	IAE	Chilled	Peak	IAE
Split range	2.96	39.8	0	−2.96	39.8
VPC on the unclamped demand	0.95	5.2	250	−2.55	24.1
VPC on the clamped position	0.95	5.2	250	−4.19	85.3

only 10 kW at once, then sits at its lower clamp, and the cooling water must do the rest. What the VPC watches decides how fast. Watching the unclamped demand, it sees how far below zero the temperature loop would like to go and moves the cooling water accordingly, fast enough to overshoot. The temperature rises to 81.71 °C and the chilled water runs at full stroke for 3 min, yet the IAE is 24.1 against 85.3. Watching the clamped position, it sees $0 - 10$ for any deficit and walks the cooling water down at 0.5 % per minute. A clamped output carries no information about how far past the clamp a controller wanted to go.

Table 1.3 A Cooling Demand Shared Between Cooling Water (CW, 400 kW at Most) and Chilled Water (CHW, 100 kW at Most) by Priority and by Equation (1.10) With Prices 1 and 4. Duties in kW

	Priority, CW first			Quadratic prices 1 and 4		
Demand	CW	CHW	Short	CW	CHW	Short
250	250	0	0	200	50	0
375	375	0	0	300	75	0
450	400	50	0	360	90	0
480	400	80	0	384	96	0
520	400	100	20	400	100	20

✦ DID YOU KNOW? **Mid-Ranging Can Lose to Split Range**

From the simulator: on the downward load step of Table 1.2, split range leaves an IAE of 39.8 and a dip of 2.96. Mid-ranging whose VPC watches the clamped valve position leaves 85.3 and 4.19, worse than the simpler structure it was meant to improve.

The resting position buys authority in both directions with the stock it draws. At $r = 10\,\%$ the store loses 10 kW at all times, 600 kW min an hour, more than two handovers of a 50 kW step, and that draw alone would empty it in ten hours. At 5, 10 and 20 %, the dip after the downward step is 2.86, 2.55 and 1.96 °C, and the chilled water drawn over the 60 min upward run is 550, 850 and 1450 kW min. When the slow actuator itself saturates, as the cooling water does on the shift of Figure 1.3, the VPC must stop integrating, or it winds up exactly as an override does [24]. A feedforward from the fast actuator's demand to the slow one can speed the handover [25].

Chapter 9 returns cooling from fast electric chillers to a slow absorption chiller with a VPC whose setpoint is an electric reserve of 0.5 MW, and lets the VPC's tracking stop it from winding up at a wide-open valve. Chapter 10 stacks three mid-ranging layers over a turbine bypass, the compressor speed and the firing rate, and finds that the order by cost need not match the order by speed. It also built the handover from the clamped bypass first, and the plant shed load in seconds and picked it up in hours. Chapter 11 uses mid-ranging as a limit inside a selector. The fast screw holds the face pressure, and the slow advance yields when the screw nears the top of its range.

1.4.8 Control Allocation

Split range and mid-ranging share a demand by a fixed rule. Control allocation states the rule as an optimization. A vector of virtual demands $\mathbf{w}$, such as a total force and moment, must be produced by actuators $\mathbf{u}$ through an effectiveness matrix B, within limits and at least cost [11, 26]. For one demand and quadratic prices w_i the answer has a closed form, Equation (1.10).

$$\min_{\mathbf{u}} \sum_i w_i u_i^2 \ \text{ s.t. } \sum_i b_i u_i = v \quad \Rightarrow \quad u_i = \frac{b_i / w_i}{\sum_j b_j^2 / w_j}\, v \tag{1.10}$$

When an actuator reaches a limit, it is fixed there and the remainder is shared among the others by the same formula. When every actuator is at a limit, a shortfall remains that no allocation removes.

Table 1.3 shares a cooling demand on the reactor by two rules. The priority rule is the split range of Section 1.4.6, a linear price that ranks cooling water before chilled water. The quadratic rule with prices of 1 and 4 shares the demand four to one until the cooling water reaches its limit at a demand of 500 kW. Both meet every demand up to 500 kW and leave the same 20 kW short at 520 kW. They differ in what they spend. At the design load of 250 kW the quadratic rule draws 50 kW of chilled water and would empty the store in two hours, while the priority rule draws none. A quadratic price suits actuators that wear or saturate, which should share a load. It does not suit an actuator that depletes, which should be kept for what the others cannot do.

A shortfall must reach the loops that asked for the demand, or their integrals wind up against a force that was never delivered. The same holds when demands compete for actuators that serve several of them. Weighting the residual of each virtual demand ranks them, so that what matters most is met first.

Chapter 12 allocates by fixed priority, fans before tires before thrusters, and its thruster mixer gives the attitude torques priority over the total lift. Chapter 13 solves a quadratic program every period for the forces of the planted feet, inside friction cones cut off by the leg ratings, with the pitch moment weighted thirty times the forces and the tangential pushes kept toward the middle of each cone, and feeds the delivered wrench back to its loops. Chapter 9 offers a predictive allocator that prices absorption and electric cooling, and Chapter 7 shares one electric bus among the spindle, the carriage and the mud pump.

1.4.9 Actuators That Deplete

The cold store is an actuator with a stock behind it. Every kilowatt it delivers is drawn from a charge E that the run cannot replenish. At a steady shortfall Q_{short}, the time it can still act, its runway, is given by Equation (1.11).

$$t_R = \frac{E - E_{\text{res}}}{Q_{\text{short}}} \tag{1.11}$$

where E_{res} is a reserve held back. The 100 kWh store carries the reactor's 75 kW shortfall for 80 min. On the shift of Figure 1.3 it began to drain as the cooling water saturated and was empty at 126.4 min. A store that is spent to the last kilowatt leaves nothing for the next fast disturbance. By Equation (1.9) one handover of a 100 kW step needs $\tau_h \times 100\,\text{kW} = 500\,\text{kW min}$, so a reserve of that size shortens the runway at 75 kW to 73.3 min.

A depleting actuator turns the stock into a constraint of its own. Rather than wait for the temperature override, a third input to the selector of Equation (1.3) could cut the feed as the runway falls below the time a slower remedy needs, and keep the reserve for the transients only the store can meet. The runway of Equation (1.11) is also optimistic whenever the drain grows as the stock falls, and in two chapters it does.

Chapter 8 runs a PI on the stressed runway of its cash, with a target of twelve months, whose output cuts spending, draws credit and cuts research in that order. Chapter 12 finds that its nitrogen tank runs out by pressure before mass. The gas cools as it expands, so a twelve-second hover ends in a landing at 2.83 m/s with 22.6 % of the nitrogen still in the tank. The car therefore lands when the gauge reads 8 % of the fill or, as an option, when the pressure reaches the regulator's setpoint. Chapter 14 defends a band with a vault of reserves above a floor. Below a threshold the market expects a devaluation and the drain grows by itself, and with a monetized deficit the date of the attack is set by credit growth alone, so no intervention policy moves it. Chapter 9 discharges a cold store in proportion to the chilled-water error, and Chapter 7 compares a battery with shore power.

1.4.10 What Each Chapter Protects and Shares

The section reduces to two questions that can be asked of any plant. Which limits must be protected, and which selector or clamp protects each? Which actuators can move the same variable, and by what rule, priority, mid-range or price, do they share it? Table 1.4 answers both for every chapter, together with any stock that depletes. The final chapter returns to this table with the measured result of each structure.

Three patterns run through the table. The machines of Chapters 2 to 7 have one main actuator and several limits, so their structure is a selector and their lesson is how the override takes over. The plants of Chapters 9 to 13 have more actuators than controlled variables, so their structure is a sharing rule and their lesson is what each actuator costs. The stocks of Chapter 8, Chapter 12 and Chapter 14 join the two, because the actuator that is spent becomes a limit of its own. A systematic treatment of how these simple elements combine is given in [27], and Volume 9 develops the predictive alternative that handles limits and prices in one optimization.

Table 1.4 What the Selectors of Each Chapter Protect, How Its Actuators Are Shared, and Which Stock Depletes

Ch.	Limits protected, and how	Actuators shared or limited	Stock
2	Surge margin, by one PI resting on its closed clamp	One recycle valve, by stroke time and capacity	—
3	Surge margin, turbine temperature, spool speeds, fuel rate, by six caps, then flame-out by two floors	One fuel valve for every limit	—
4	Engine power, rotor torque, speed and a preview cap, by a low selector on the ground speed	One engine for every drive	—
5	Torque, die pressure and dryer air, by overrides realized as clamps	Barrel zones split between heater bands and cooling water	—
6	Solvent vapor in each zone, by an override with three handles in order	Exhaust, then line speed or zone heat	—
7	Mud pressure against the ground's limit and torque, by a low selector on thrust	One electric bus for spindle, carriage and pump	Battery
8	Cash runway, by an override whose level is split over three ranges	Six decisions on a quarterly calendar	Cash above a zero floor
9	Exhaust temperature, by low selection on the fuel, and grid import	Electric chillers handed back to absorption by a VPC	Cold store
10	Turbine inlet metal and pressure rating	Bypass, compressor speed and firing in three mid-ranging layers	—
11	Torque, thrust, screw room and ramp, by the least of five limits on the advance	Screw holds the face, advance yields near the screw's top	—
12	Tire grip and attitude, by priority in the launch and in the thruster mixer	Fans, tires, then thrusters	Nitrogen, by pressure
13	Friction cones and leg ratings, feet that push but cannot pull	Weighted least squares among planted feet	—
14	Band edges and the reserve floor, with a capital-controls override	Desk sales and a policy rate capped by politics	Reserves

1.5 Instruments and Final Elements in This Volume

An override acts only on what its transmitter reports, and only as fast as its final element moves. Figure 1.6 lists what the controllers of each chapter read and what they move. Temperature and pressure lead the readings, with six chapters each, because most of the limits that the selectors protect are a temperature or a pressure. A motor or drive leads the final elements with eight chapters, ahead of the valve or flow loop with six.

In four chapters a pressure is the very variable the structure exists to hold: the die pressure of Chapter 5, the mud pressure against the ground in Chapter 7, the face pressure of Chapter 11 and the discharge pressure of Chapter 10. The top of Figure 1.7 shows the principle of the transmitter.

A variable-speed motor is often a better final element than a control valve, because slowing a pump in low-flow conditions saves the energy that a constant-speed pump and a throttling valve would waste [16]. Drives move the ground speed of Chapter 4, the screws of Chapter 5 and Chapter 11, the compressor of Chapter 10 and the legs of Chapter 13. The bottom of Figure 1.7 shows a drive on a pump.

The selector itself is a standard function block. In one fieldbus standard the control selector block sends a back-calculation signal to each upstream controller, so that a controller it de-selects knows at once to halt its integral action [16]. In a process plant, programmable logic controllers serve as local controllers under a distributed control system, and nested loops run with cycle times from milliseconds to minutes [28]. A jet engine's computer has two channels, each an independent computer that can control the engine completely [29], and a car executes its control in electronic control units [1]. Figure 1.8 places the loops of this volume on their platforms. The apps run from the



Figure 1.6 What the controllers of each chapter read and move: a round mark where a controller reads the instrument and a square mark where it moves the element, as the chapter's variables table and its technology section name them. A flow set through a flow station counts as a valve or flow loop, and a computed value is marked under the instruments it is computed from. Notice that temperature and pressure lead the readings and a motor or drive leads the final elements.

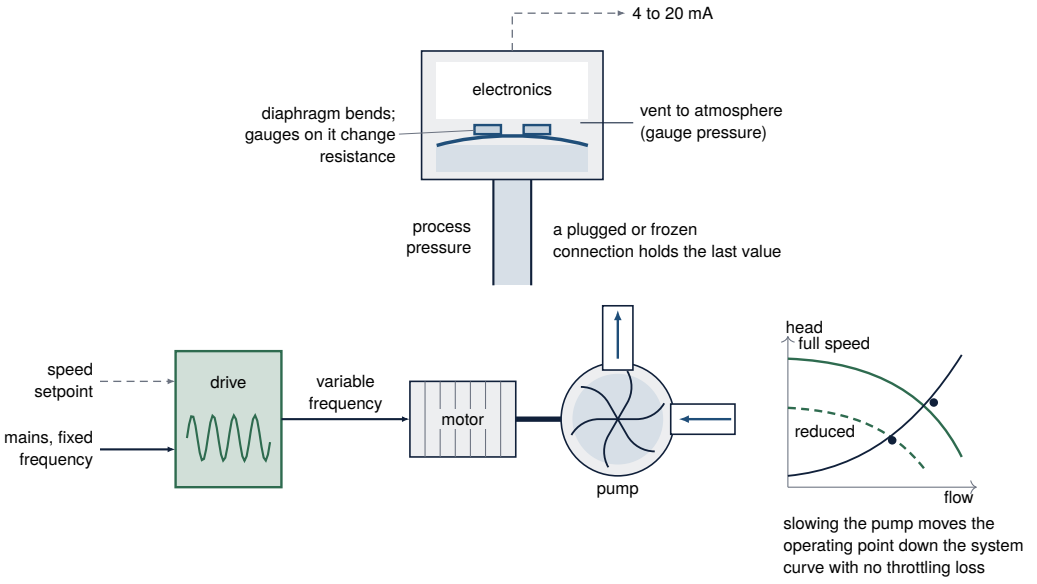


Figure 1.7 Top: principle of a gauge pressure transmitter. Bottom: a variable-speed drive on a centrifugal pump, with the pump curves at two speeds on one system curve. Notice that a plugged connection freezes the pressure signal, so an override that watches it never takes over, and that the drive moves the operating point by speed instead of by throttling.

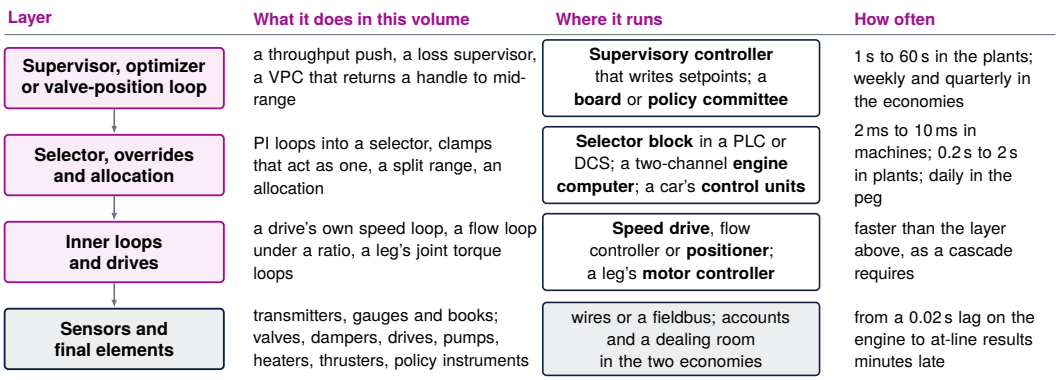


Figure 1.8 The control layers of this volume, the platform that hosts each and how often each acts. Platforms are from [16], [28], [29] and [1], and the periods from the chapters' technology sections. Notice that the selector layer runs in every chapter, from milliseconds in the machines to a day in the currency peg.

Table 1.5 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.

2 ms period of Chapter 13 to the quarterly calendar of Chapter 8.

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 1.5 lists what a reader meets.

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

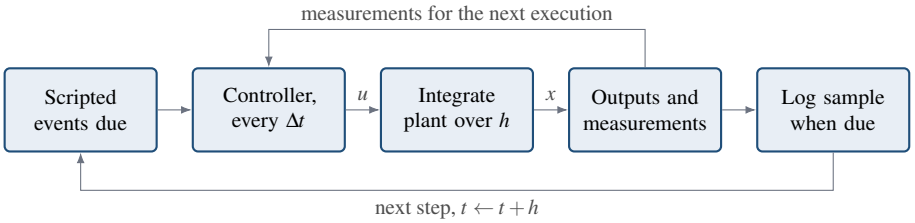


Figure 1.9 Work of the Engine in One Integration Step

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 1.9. 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 (1.12) to (1.15) for execution k and control period Δt.

$$e_k = a\,(\mathrm{SP}_k - \mathrm{PV}_k), \qquad P_k = K_c\,e_k, \qquad D_k = -a\,K_c\,\tau_D\,\frac{\mathrm{PV}^f_k - \mathrm{PV}^f_{k-1}}{\Delta t} \tag{1.12}$$

$$S_k = S_{k-1} + (P_k - P_{k-1}) + (D_k - D_{k-1}) \tag{1.13}$$

$$v_k = u_b + S_k + I_{k-1}, \qquad u_k = \mathrm{sat}(v_k) \tag{1.14}$$

$$I_k = I_{k-1} + \left[\frac{K_c}{\tau_I}\,e_k + \frac{u_k - v_k}{T_t}\right]\Delta t \tag{1.15}$$

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 (1.15) drives I toward the

value that reproduces the output applied, which is back-calculation [30]. With no integral action, Equation (1.15) 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 $I_k = u_{\text{man}} - u_b - S_k$, 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 (1.16).

$$x_{k+1} = A\,x_k + B\,u_k + E\,w_k + c, \qquad y_k = C\,x_k \tag{1.16}$$

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 (1.17) over a horizon of N steps.

$$J = \sum_{k=1}^{N} (y_k - r_k)^{\top} Q\,(y_k - r_k) + \sum_{j} \Delta u_j^{\top} R\,\Delta u_j + \sum_{i} \rho_i\, s_i^2 \tag{1.17}$$

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 [31]. 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 [32]. 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 [33].

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.

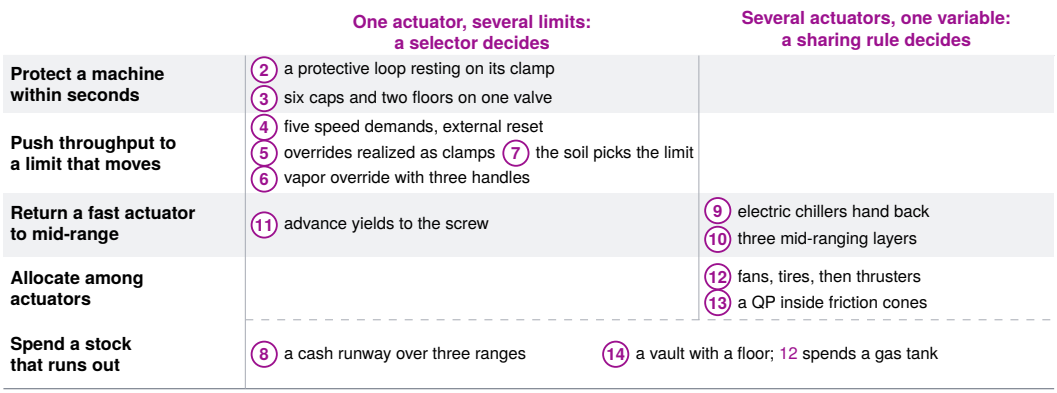


Figure 1.10 Method map of the volume. Each chapter is placed by the shape of its constraint problem, from Section 1.4.2 and Table 1.4, and by the lesson it centers on, shortened from Table 1.6. The placement is the editor's reading of the two tables. Notice that the stocks of the bottom row join the two columns, because an actuator that is spent becomes a limit of its own.

- *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 15.

1.8 Chapter Roadmap

Table 1.6 lists the chapters with the 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 1.10 places the same chapters by the shape of their constraint problem and by the lesson each centers on.

Chapter 2 and Chapter 4 are the first to read. The first isolates one protective loop, and the second shows a low selector with external reset among five demands. Chapter 3 then adds floors, a rate limit and the choice of tracking time. Chapters 5 to 7 push throughput against limits that move and can be read in any order once Sections 1.4.2 to 1.4.5 are familiar. For sharing, Chapter 9 is the gentler introduction to mid-ranging and Chapter 10 the fuller one, with Chapter 11 as a variant inside a selector. Chapter 12 and then Chapter 13 carry allocation from a fixed priority to an optimization. Chapter 8 and Chapter 14 form a path for readers from economics and finance, where the depleting stock is money.

Table 1.6 Roadmap of the Volume

Ch.	Plant	The lesson	Level	Time
2	Centrifugal compressor near surge	A protective loop resting on its clamp, and a valve too slow for a correct decision.	introductory	2.5 h
3	Two-spool turbofan engine	Six caps and two floors on one fuel valve, and what the tracking time decides.	intermediate	2.5 h
4	Combine harvester in a patchy field	Five speed demands in a low selector with external reset, under a loss supervisor.	introductory	2.5 h
5	Cereal extrusion line	Overrides realized as clamps, split-range zones and a throughput supervisor.	advanced	2.5 h
6	Battery electrode coater and dryer	An explosive-limit override with three handles, and a push to the first binding limit.	advanced	2.5 h
7	Horizontal directional drill	A rate optimizer whose binding limit changes with the soil, and pressure overrides.	advanced	2 h
8	Product startup company	A runway override over three ranges, on a quarterly calendar.	intermediate	2.5 h
9	Gas turbine trigeneration plant	An exhaust limiter on the fuel, and a VPC that returns cooling to the cheap chiller.	intermediate	2.5 h
10	Supercritical CO_2 Brayton cycle	Three mid-ranging layers, and why the handover reads the unclamped demand.	advanced	3 h
11	Earth pressure balance shield	An inventory held through forced stops, with a mid-ranging limit on the advance.	intermediate	2.5 h
12	Car with fans and cold-gas thrusters	Allocation by priority on a tank that runs out by pressure.	intermediate	2.5 h
13	Planar quadruped robot	Allocation by a quadratic program inside friction cones that switch with each step.	advanced	2.5 h
14	Central bank defending a peg	A vault with a floor, a drain that grows as it empties, and too few instruments.	intermediate	2.5 h

KEY POINTS

- When several limits share one actuator, a selector passes the most cautious demand, and the guard of a limit takes over only as that limit comes near.
- External reset feeds each integral from the output actually applied, so with a tracking time equal to the integral time the override takes over at its setpoint, without a bump.
- Split range moves one actuator at a time in a fixed order. Mid-ranging lets a fast actuator answer first and hands the load to a slow one.
- An actuator that draws on a stock is also a limit. Its runway is the stock above a reserve divided by the shortfall.

Where else this applies. A low selector with external reset on a treadmill belt in Volume 19, Chapter 6, a four-way low selector on a fast charger in Volume 17, Chapter 9, and predictive control of limits in Volume 9.

References

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

[2] 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.

[3] F. G. Shinskey and J. P. Shunta, “Selective, override, and limit controls,” in *Process Control*. Elsevier, 1995, pp. 112–118.

[4] A. H. Glattfelder and W. Schaufelberger, *Control Systems with Input and Output Constraints*, ser. Advanced Textbooks in Control and Signal Processing. London: Springer, 2003.

[5] M. Pitt, “Taking a look back at control: Part 2,” *The Chemical Engineer*, no. 989, Nov. 2023, thechemicalengineer.com/features/taking-a-look-back-at-control-part-2.

[6] ——, "Taking a look back at control: Part 1," *The Chemical Engineer*, no. 988, pp. 51–54, Oct. 2023, thechemicalengineer.com/features/taking-a-look-back-at-control-part-1.

[7] L. C. Jaw and S. Garg, "Propulsion control technology development in the United States: A historical perspective," NASA Glenn Research Center, Cleveland, OH, Technical Memorandum NASA/TM-2005-213978, 2005, nTRS 20050238468.

[8] E. M. Greitzer, "Surge and rotating stall in axial flow compressors—part I: Theoretical compression system model," *Journal of Engineering for Power*, vol. 98, no. 2, pp. 190–198, 1976.

[9] A. H. Glattfelder, W. Schaufelberger, and H. P. Fässler, "Stability of override control systems," *International Journal of Control*, vol. 37, no. 5, pp. 1023–1037, 1983.

[10] B. Eichengreen and C. Wyplosz, "The unstable EMS," *Brookings Papers on Economic Activity*, vol. 1993, no. 1, pp. 51–144, 1993.

[11] W. C. Durham, "Constrained control allocation," *Journal of Guidance, Control, and Dynamics*, vol. 16, no. 4, pp. 717–725, 1993.

[12] M. V. Kothare, P. J. Campo, M. Morari, and C. N. Nett, "A unified framework for the study of anti-windup designs," *Automatica*, vol. 30, no. 12, pp. 1869–1883, 1994.

[13] B. J. Allison and A. J. Isaksson, "Design and performance of mid-ranging controllers," *Journal of Process Control*, vol. 8, no. 5–6, pp. 469–474, 1998.

[14] S. Hyon, J. G. Hale, and G. Cheng, "Full-body compliant human–humanoid interaction: Balancing in the presence of unknown external forces," *IEEE Transactions on Robotics*, vol. 23, no. 5, pp. 884–898, 2007.

[15] A. Moisseytsev and J. J. Sienicki, "Development of the ANL plant dynamics code and control strategies for the supercritical carbon dioxide Brayton cycle and code validation with data from the Sandia small-scale supercritical carbon dioxide Brayton cycle test loop," Argonne National Laboratory, Argonne, IL, USA, Tech. Rep. ANL-ARC-218, 2011.

[16] 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.

[17] D. Krishnamoorthy and S. Skogestad, "Systematic design of active constraint switching using selectors," *Computers & Chemical Engineering*, vol. 143, p. 107106, 2020.

[18] A. Reyes-Lúa and S. Skogestad, "Systematic design of active constraint switching using classical advanced control structures," *Industrial & Engineering Chemistry Research*, vol. 59, no. 6, pp. 2229–2241, 2020.

[19] C. Loeblein and J. D. Perkins, "Structural design for on-line process optimization: I. dynamic economics of MPC," *AIChE Journal*, vol. 45, no. 5, pp. 1018–1029, 1999.

[20] K. J. Åström and T. Hägglund, *Advanced PID Control*. Research Triangle Park, NC: ISA – The Instrumentation, Systems, and Automation Society, 2006.

[21] A. Reyes-Lúa, C. Zotică, K. Forsman, and S. Skogestad, "Systematic design of split range controllers," *IFAC-PapersOnLine*, vol. 52, no. 1, pp. 898–903, 2019.

[22] A. Reyes-Lúa and S. Skogestad, "Multi-input single-output control for extending the operating range: Generalized split range control using the baton strategy," *Journal of Process Control*, vol. 91, pp. 1–11, 2020.

[23] B. J. Allison and S. Ogawa, "Design and tuning of valve position controllers with industrial applications," *Transactions of the Institute of Measurement and Control*, vol. 25, no. 1, pp. 3–16, 2003.

[24] S. Haugwitz, M. Karlsson, S. Velut, and P. Hagander, "Anti-windup in mid-ranging control," in *Proceedings of the 44th IEEE Conference on Decision and Control*. IEEE, 2005, pp. 7570–7575.

[25] T. Hägglund, "A feedforward approach to mid-ranging control," *Control Engineering Practice*, vol. 108, p. 104713, 2021.

[26] T. A. Johansen and T. I. Fossen, "Control allocation—a survey," *Automatica*, vol. 49, no. 5, pp. 1087–1103, 2013.

[27] S. Skogestad, "Advanced control using decomposition and simple elements," *Annual Reviews in Control*, vol. 56, p. 100903, 2023.

[28] 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.

[29] Federal Aviation Administration, *Aviation Maintenance Technician Handbook–Powerplant*. U.S. Department of Transportation, Federal Aviation Administration, 2023, no. FAA-H-8083-32B.

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

[31] 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.

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

[33] 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

The reactor of Section 1.4.1 runs with the cooling water at full stroke and the cold store unused. TIC-1 has $K_c = 2.5\,\%/°\mathrm{C}$ and $\tau_I = 10\,\mathrm{min}$ on a split range that gives 5 kW per percent of its output, and each kg/min of feed releases 25 kW.

(a) Find the steady feed the reactor can carry if the override setpoint is raised from 85 to 87 °C, and the gain over the feed at 85 °C. *Checkpoint:* 17.4 kg/min, 0.4 kg/min more.

(b) TIC-1 holds 80 °C. Find the fastest feed ramp, in kg/min per minute, whose steady ramp lag keeps the temperature at least 2 °C under an override setpoint of 85 °C. *Checkpoint:* 0.15 kg/min per minute.

1.2 Concept check · 2 min

A gas turbine's fuel demand passes several caps, among them a temperature limiter and a rate limit, and a minimum fuel flow keeps the flame lit. Which arrangement keeps the flame lit even when a cap asks for less than the minimum fuel?

(a) One low selector over the caps and the minimum fuel together

(b) A high selector with the minimum fuel, applied after a low selector over the caps

(c) A low selector over the caps, applied after a high selector with the minimum fuel

(d) The average of all caps and the minimum fuel, sent to the valve

1.3 Theory and calculation · 3 min

The override TC-2 of the reactor has $K_c = 2\,\mathrm{kg/min}$ per °C, setpoint 85 °C, external reset with $T_t = \tau_I$, and the selector applies a feed of 19 kg/min. Find the override's demand once its integral has settled, at measured temperatures of 83 and 84.5 °C, and the temperature at which it takes over.

1.4 Theory and calculation · 4 min

A split-range controller moves a cooling-water valve worth 6 kW per percent of stroke (600 kW at full stroke) and then a chilled-water valve worth 2 kW per percent (200 kW). Find the split point that gives the same duty per percent of controller output on both segments, and that duty. Then find the duty per percent on each segment for a split at 50 %, and the ratio of the two loop gains.

1.5 Concept check · 2 min

A valve-position controller returns a fast bypass valve to a resting opening of 5 %. The bypass is clamped at zero. When the plant falls short of its target, the fast controller asks for a negative opening. What happens if the VPC watches the clamped bypass position instead of the fast controller's unclamped demand?

(a) The VPC moves faster, because the clamped position is less noisy

(b) Nothing changes, because the two signals are equal in steady state

(c) The VPC sees the same error for a small and a large shortfall and moves at one slow rate

(d) The VPC winds up and drives the slow actuator to its opposite limit

1.6 Theory and calculation · 4 min

A reactor uses mid-ranging with chilled water as the fast actuator, 1 kW per percent, and cooling water as the slow one, 5 kW per percent. The VPC moves the cooling water at $K_I = 0.05\,\mathrm{min}^{-1}$ times the chilled-water demand less its resting position.

(a) Find the static gain the VPC sees and its closed-loop time constant. *Checkpoint:* 5, and 4 min.

(b) A load step of 30 kW arrives that the cooling water can carry. Find how far the cooling water moves and the chilled-water energy drawn above the resting draw during the handover. *Checkpoint:* 6 %, and 120 kW min (2 kWh).

1.7 Concept check · 2 min

An override's integral is corrected by back-calculation from the selector's output, with a tracking time T_t of one tenth of its integral time instead of the integral time itself. Compared with $T_t = \tau_I$, what does the override do as its constraint approaches the limit?

(a) It takes over earlier, on the rate of approach, before the limit is reached

(b) It takes over later, after the constraint has passed its limit

(c) It takes over at the same moment, because the tracking time only affects noise

(d) It never takes over, because its integral cannot leave the applied output

1.8 Theory and calculation · 4 min

A cooling demand is shared between cooling water (at most 400 kW) and chilled water (at most 100 kW) by Equation (1.10) with prices of 1 and 9 and unit effectiveness. Find the two duties for demands of 300 and 450 kW, and compare each with the priority rule that uses cooling water first.

1.9 Concept check · 2 min

A protective PI controller rests on its lower output clamp while the plant is healthy, with back-calculation at $T_t = \tau_I$ and an error that stays negative. What does its integral term do while it rests there?

(a) It integrates the negative error without bound

(b) It stays frozen at the value it had when the clamp was reached

(c) It is reset to zero at every execution

(d) It decays toward the clamp value with time constant τ_I, whatever the error

1.10 Theory and calculation · 4 min

A cold store holds 100 kWh (6000 kW min).

(a) The store must cover a steady shortfall of 60 kW. Find how long it lasts. *Checkpoint:* 100 min.

(b) The store keeps a reserve of 500 kW min for one handover. Find the runway at the same shortfall. *Checkpoint:* 91.7 min.

(c) With no shortfall, a VPC rests the chilled-water valve at 5 %, worth 5 kW. Find how long the resting draw alone takes to empty the full store. *Checkpoint:* 20 h.