INTERACTIVE CONTROL ENGINEERING
Principles and Applications

Volume 7 Nonlinear Dynamics, Oscillations, and Stabilization

CHAPTER 1

Introduction: Stabilizing Unstable and Oscillating Plants

Pages 1–21

Edited by John D. Hedengren

How to cite this chapter. "Introduction: Stabilizing Unstable and Oscillating Plants," in *Nonlinear Dynamics, Oscillations, and Stabilization*, John D. Hedengren, Ed., vol. 7 of *Interactive Control Engineering: Principles and Applications*. Advanced Process Solutions, LLC, 2026, ch. 1, pp. 1–21.

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/v07-nonlinear-stabilization/

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

CHAPTER 1

Introduction: Stabilizing Unstable and Oscillating Plants



The chapters 2 exothermic reactor 3 aeropendulum 4 braked wheel 5 ball and beam 6 levitated ball 7 pole on a cart 8 two links on a cart 9 riderless bicycle 10 drifting car 11 quarter car 12 gantry crane 13 end mill 14 line of cars

Notice that the pole, the two links, the levitated ball and the drifting car would fall or spin out without their controllers, while the crane, the strut and the mill settle by themselves but ring. Illustration, not to scale.

IN PLAIN TERMS **Holding What Would Fall or Shake**

The main idea. Some machines will not stay put: a pole on a cart falls, a ball under a magnet drops or jumps to it, and a hot reactor can run away. Others, like a load on a crane, settle but keep swinging.

The challenge. A controller holds such a machine only near the point it was designed for, and only while its push is strong and quick enough. A push that comes too late or too weak cannot catch the machine again.

Why it matters. The same ideas keep braked wheels from locking, crane loads from swinging and lines of cars from passing a slowdown back larger.

THREE WAYS TO READ THIS CHAPTER

Explore	7 min	This page, Section 1.2 with its timeline and two History boxes, the Did you know? boxes in Sections 1.4.2, 1.4.7 and 1.4.11, Figure 1.11, and Key points in Section 1.8. No equations.
Study	40 min	The whole chapter in order, with each number of the pendulum example worked, then Exercises 1 to 10.
Practice	5 min	This page, Section 1.4.11 with Table 1.2, Section 1.5 with Figures 1.7 and 1.9, and Section 1.8 with Figure 1.11.

1.1 The Question This Volume Answers

Most loops in the earlier volumes act on a plant that would settle by itself. The plants of this volume do not. A reactor runs between two folds of its steady-state curve, a ball hangs under a magnet that pulls harder the closer it comes, a pole stands on a cart, a car drifts sideways at the edge of its tires, and a line of cars passes each brake tap back larger than it came. Others settle but ring: a crane load swings, a car body bounces, a cutting tool chatters. This volume answers one question: what holds such a plant, and how small is the margin by which it is held?

Thirteen chapters answer it on thirteen plants, each with an app that the reader runs in a browser. Chapters 2 to 4 are plants whose steady-state gain changes size and sign with the operating point. Chapters 5 to 10 are unstable mechanical plants held by feedback, from one unstable pole to two, and from one input to two. Chapters 11 to 13 are oscillating plants, where a resonance or a delay sets the limit. Chapter 14 closes the volume on a line of cars, where each loop is stable and the line may not be.

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

- Find the steady states of a nonlinear model, the gain at each, and the folds where steady states meet and vanish.

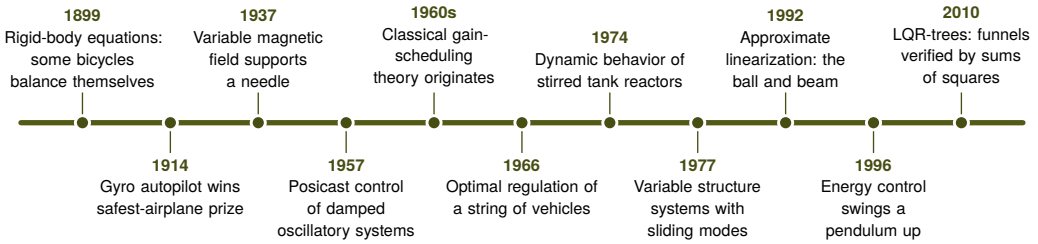


Figure 1.1 Milestones of stabilization in the systems of this volume. The dates are from [4] (1899), [6] (1914), [5] (1937), [7] (1957), [8] (1960s), [9] (1966), [10] (1974), [11] (1977), [12] (1992), [13] (1996) and [14] (2010). Notice that the bicycle, the airplane and the floating needle come before the linear theory of Section 1.4, and that the methods for large motions come last. Events are evenly spaced, not to a time scale.

- Linearize the model at an operating point, write it in state space and read stability, speed and mode shape from its eigenvalues.
- Design a state-feedback gain by pole placement or as a linear quadratic regulator (LQR), and an observer for the states not measured.
- Say what a right-half-plane pole or zero, an actuator limit, a lag or a delay costs, before any controller is tuned.
- Remove a resonance from a command by input shaping, predict a limit cycle, and test a chain of loops for string stability.

This chapter develops each of these on one small example and names the chapter that uses it. Standard texts give fuller treatments [1–3]. PID forms and tuning are taken as known from Volume 2.

1.2 A Short History

The milestones of Figure 1.1 follow the methods of this volume through very different machines, some of them held before a theory explained them. Rigid-body equations of 1899 showed that some safety bicycles balance themselves in the right range of speed [4], the window that Chapter 9 measures. A paper of 1937 described a vertical ferromagnetic needle supported by magnetic forces alone with a variable field [5], the single axis of Chapter 6.

> ⧖ HISTORY **1903 to 1914: Feedback Holds an Unstable Airplane**
>
> The Wright brothers settled the trade between stability and maneuverability with an airplane that was unstable but maneuverable, flown at Kitty Hawk in 1903, and its pilot had to keep adjusting the rudder. Sperry then used a gyro-stabilized pendulum to indicate the vertical and a feedback mechanism that pulled the stick to raise the nose when the plane pointed down, and the reverse. It was the first use of feedback in aeronautical engineering, and it won a prize in a competition for the safest airplane, in Paris in 1914. Feedback had stabilized an unstable machine [6], as Section 1.4.5 does for the upright arm.

Oscillation was first taken out of the command. Posicast control, published in 1957, split a step into two excitations spaced so that their ringing cancels [7], the root of the shapers of Chapter 12. The state-space tools of Section 1.4 followed: a new approach to filtering and prediction in 1960 [15], then the observer [16] and the question of when a linear control system is optimal [17], both in 1964.

Nonlinearity entered next, first as a family of linear models, a story the second box below tells. A paper of 1974 on the dynamic behavior of continuous stirred tank reactors [10] gave the S-shaped curve of Chapter 2, and one of 1977 set out variable structure systems with sliding modes [11], the switching law of Chapter 4. Feedback linearization was a new approach in flight control in the

mid-1980s, under the name dynamic inversion [18], as Chapter 3 tells, and a paper of 1992 took the ball and beam as its example of approximate input-output linearization [12].

The last milestones concern large motions and chains of loops. An energy method for swinging a pendulum up, presented in 1996, left the upright point to another controller to catch [13], the switch of Chapter 7. In 2010 regions of attraction became funnels around trajectories stabilized by LQR and verified by sums of squares [14], the handover test of Chapter 8. For chains, a paper of 1966 regulated a string of moving vehicles by optimal control [9], and one of 1996 set out the string stability of interconnected systems [19] that Section 1.4.10 tests.

> ⧖ HISTORY **From the 1960s: Divide and Conquer by Gain Scheduling**
>
> Wide variations of operating conditions in aerospace stimulated the development of gain scheduling and adaptive control [6]. Much of the classical theory of gain scheduling originates from the 1960s. It decomposes a nonlinear design into linear subproblems, so that well established linear methods apply, and safety certification requirements are often based on linear methods. A review of 2000 called it perhaps one of the most popular nonlinear design approaches, applied from aerospace to process control. Relaxing its condition that the plant pass between operating regions slowly was still very much an open question [8].

1.3 Notation and Conventions

A model is written $\dot{x} = f(x,u)$ with the state x and the input u. An equilibrium, or steady state, is a pair (x_0,u_0) with $f(x_0,u_0) = 0$. Deviations from it are written δx and δu, or x and u again where no confusion arises. The linearized model is $\delta\dot{x} = A\,\delta x + B\,\delta u$ with a measurement $y = Cx$. A state feedback is $u = -Kx$ with a row of gains K, an observer has the gain L, and the weights of an LQR are Q and R, as in the predictive controller of Section 1.6.3. An unstable real pole is p and a right-half-plane (RHP) zero is z, both in 1/s or rad/s. The control period is Δt, an actuator lag is τ and a pure delay is τ_d.

The running example uses seconds, radians and rad/s^2 for its input. Each chapter keeps the units of its plant: kelvin and minutes for the reactor, milliseconds for the levitated ball and the double pendulum, and meters per second for the vehicles. Several chapters write the unstable root differently. Chapter 7 writes it λ_0, Chapter 6 and Chapter 4 write p, and Chapter 4 also uses p for a stable slip pole. Chapter 14 writes the time gap h, which this chapter writes t_g, because h is the integration step of Section 1.6.1. The framework's $a = \pm 1$ is the controller action, so the cart's acceleration here is $\ddot{x}$.

1.4 Stability Theory on One Pendulum

1.4.1 The Running Example

The running example is one simple pendulum, a point mass on a light rod of length $\ell = g/4 = 2.45\,\text{m}$, so that $g/\ell = 4/\text{s}^2$ and $\sqrt{g/\ell} = 2\,\text{rad/s}$. The pivot is frictionless. The input u is an angular acceleration in rad/s^2, and it enters in two ways, as shown in Figure 1.2. In the arm, a motor at a fixed pivot applies a torque, and u is that torque divided by $m\ell^2$. On the cart, the pivot rides a cart whose drive sets its acceleration $\ddot{x}$, and $u = \ddot{x}/\ell$. With θ measured from hanging, the two give Equation (1.1).

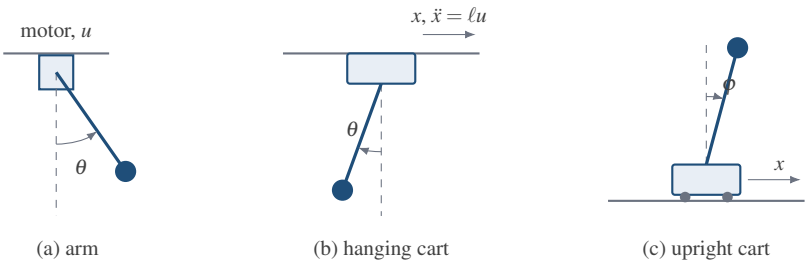
$$\text{arm:}\quad \ddot{\theta} = -4\sin\theta + u, \qquad \text{cart:}\quad \ddot{\theta} = -4\sin\theta - u\cos\theta, \quad \ddot{x} = \ell u \tag{1.1}$$

Hanging, the pendulum swings with a period of $\pi = 3.14\,\text{s}$. Upright, a small tilt grows by a factor of e every 0.5 s. The upright arm, the upright pole on a cart and the crane load of the hanging cart are the three plants of this section. The arm also stands for the aeropendulum of Chapter 3, the hanging cart for the crane of Chapter 12, and the upright cart for the pendulums of Chapter 7 and Chapter 8.

1.4.2 Steady-State Gain and Multiplicity

A constant input holds the arm at rest where gravity balances it, as shown in Equation (1.2).

$$u_0 = 4\sin\theta_0, \qquad \frac{\mathrm{d}\theta_0}{\mathrm{d}u_0} = \frac{1}{4\cos\theta_0} \tag{1.2}$$



(a) arm (b) hanging cart (c) upright cart

Figure 1.2 The Running Example. One Pendulum With $g/\ell = 4/\mathrm{s}^2$, Driven (a) by a Motor at a Fixed Pivot, (b) by the Acceleration of a Cart Below Which It Hangs and (c) by the Same Cart With the Pendulum Upright. The Tilt From Upright Is $\varphi = \theta - \pi$.

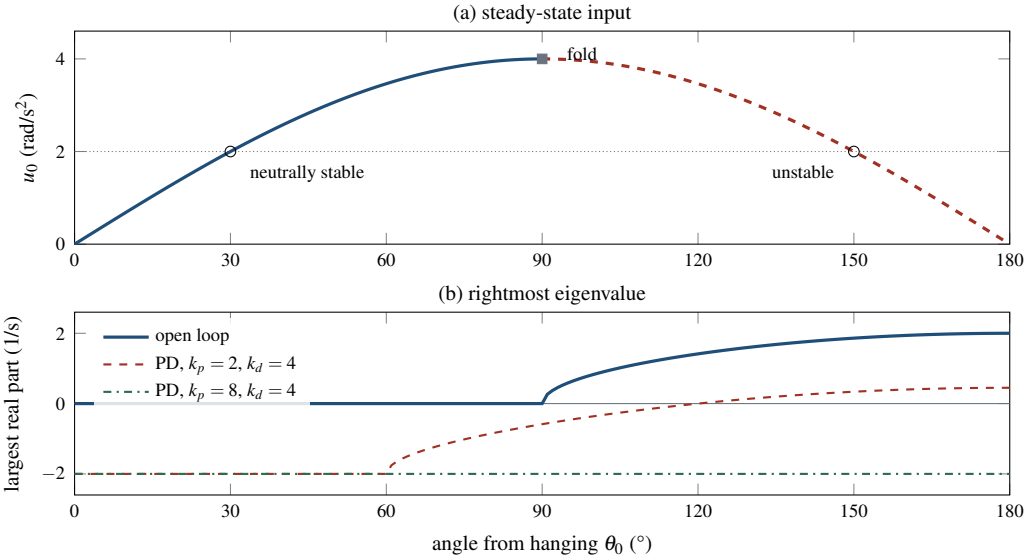


Figure 1.3 Steady States of the Arm. (a) The Input That Holds Each Angle, Neutrally Stable Below Horizontal (Solid) and Unstable Above (Dashed), With the Two Steady States of $u_0 = 2$ (Circles) and the Fold (Square). (b) The Largest Real Part of the Eigenvalues: Open Loop (Thick Solid) and With the PD and Gravity Feedforward of Equation (1.3) at $k_p = 2$ (Dashed) and $k_p = 8$ (Dash-Dotted).

The second expression is the steady-state gain, the slope of the inverse of the first. Figure 1.3(a) draws the steady-state curve. An input of $2\,\mathrm{rad/s}^2$ has two steady states, at 30° and 150°. Their gains are equal and of opposite sign, ± 0.289 rad per $\mathrm{rad/s}^2$. The largest input that holds the arm is $4\,\mathrm{rad/s}^2$, at horizontal. There the two branches meet, the gain is infinite, and above it no steady state exists. Such a turning point is a fold, or saddle-node bifurcation [1].

The two branches differ in stability. Linearizing Equation (1.1) at θ_0 gives $\delta\ddot{\theta} = -4\cos\theta_0\,\delta\theta + \delta u$, a stiffness $4\cos\theta_0$ that gravity supplies below horizontal and takes away above it. The eigenvalues are $\pm\sqrt{-4\cos\theta_0}$. At 30° they are $\pm 1.86\mathrm{j}$, an undamped swing, and at 150° they are ± 1.86, one of them unstable, as Figure 1.3(b) shows. The equilibrium loses stability exactly where the holding input is largest, because there the stiffness passes through zero.

The same picture organizes three chapters. The reactor of Chapter 2 has an S-shaped curve with two folds, three steady states inside a window of jacket temperature, and an open-loop unstable interval that contains the default setpoint [10]. The wheel of Chapter 4 has the arm's shape exactly:

the brake torque that holds a slip has a maximum, the equilibrium loses stability at that maximum, and a larger torque has no equilibrium at all. The aeropendulum of Chapter 3 is the arm with a propeller, and its stiffness changes sign at horizontal. Chapter 10 solves for its equilibrium before it designs anything, and Chapter 6 has one at every gap and none stable.

> ✦ **DID YOU KNOW? Climbing to the Peak Since 1951**
>
> An extremum-seeking controller varies its reference slowly and sinusoidally and reads from the phase of the response which side of the optimum it is on. Extremal control was first applied to aircraft engines in 1951 [6], and Chapter 4 names it as one way to stay on a tire's friction peak.

1.4.3 Linearization at an Operating Point

A linearization describes the plant near one point. The arm shows how much it changes from point to point. Consider the proportional-derivative (PD) law of Equation (1.3) with a gravity feedforward.

$$u = 4\sin\theta_{\mathrm{SP}} - k_p(\theta - \theta_{\mathrm{SP}}) - k_d\dot{\theta} \tag{1.3}$$

At a setpoint θ_0 the closed loop is $s^2 + k_d s + (k_p + 4\cos\theta_0) = 0$. The controller adds stiffness to what gravity supplies, and the loop is stable while the sum is positive. With $k_p = 8$ and $k_d = 4$ the damping ratio is 0.577 hanging, 0.707 at horizontal and 1.0 upright, so one tuning serves every angle. With $k_p = 2$ the loop loses the arm beyond 120°, where $2 + 4\cos\theta_0$ turns negative, as the dashed line of Figure 1.3(b) shows. Without the feedforward the same $k_p = 8$ leaves an offset, because the controller must supply the holding input from an error: a 90° setpoint settles at 64.2°.

Two remedies keep a tuning valid at every point. A gain schedule recomputes k_p from the measured angle so that the closed-loop polynomial stays the one at the design point. From $k_p = 8$ at hanging it gives $k_p = 12 - 4\cos\theta_0$, 16 upright, and $s^2 + 4s + 12$ everywhere. Feedback linearization cancels the nonlinearity instead: $u = 4\sin\theta + v$ leaves $\ddot{\theta} = v$, a double integrator at every angle, for an outer law v. Both need the model to be right. Both also trust a linearization whose validity shrinks as the plant moves faster than the schedule.

The upper branch carries one more lesson. Its steady-state gain is negative, yet the loop of Equation (1.3) uses the same signs there as below horizontal. A positive u always accelerates the arm toward larger θ at once, and only the eventual response reverses. A controller fast enough to act on the immediate response must take its sign from it, not from the steady-state gain.

Chapter 2 meets exactly this: its gain is negative on the middle branch, and its reverse-acting PI loop works because warming the jacket always warms the reactor first. Chapter 3 compares a fixed tuning, a schedule that holds the closed-loop polynomial and feedback linearization across 15 to 150°. Chapter 9 schedules on speed, Chapter 10 on speed and friction, and Chapter 6 shows one tuning giving three responses across three gaps.

1.4.4 State-Space Models and Eigenvalues

Near upright, with the tilt $\varphi = \theta - \pi$ and its rate as the state, the arm is the model of Equation (1.4).

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\varphi\\ \dot{\varphi}\end{bmatrix} = \underbrace{\begin{bmatrix}0 & 1\\ 4 & 0\end{bmatrix}}_{A}\begin{bmatrix}\varphi\\ \dot{\varphi}\end{bmatrix} + \underbrace{\begin{bmatrix}0\\ 1\end{bmatrix}}_{B}u, \qquad y = \underbrace{\begin{bmatrix}1 & 0\end{bmatrix}}_{C}\begin{bmatrix}\varphi\\ \dot{\varphi}\end{bmatrix} \tag{1.4}$$

The eigenvalues of A are ± 2, with the eigenvectors $(1, 2)$ and $(1, -2)$. In the coordinates of these eigenvectors the model splits into two first-order modes, as shown in Equation (1.5).

$$\sigma = \dot{\varphi} + 2\varphi, \quad \dot{\sigma} = 2\sigma + u, \qquad \rho = \dot{\varphi} - 2\varphi, \quad \dot{\rho} = -2\rho + u \tag{1.5}$$

The unstable mode σ is the only thing the controller must fight. A tilt that leans back toward upright at twice its size per second has $\sigma = 0$ and returns by itself.

The upright cart has four states, $(x, \dot{x}, \varphi, \dot{\varphi})$, and the linear model $\ddot{x} = \ell u$, $\ddot{\varphi} = 4\varphi + u$. Its eigenvalues are $0, 0, 2$ and -2. One input reaches all four states, so the model is controllable: the controllability matrix $[B, AB, A^2B, A^3B]$ has rank four. Observability is another matter. Because the drive sets the acceleration, the cart never feels the pendulum. The cart's encoder alone therefore says nothing about the pole, and the pole's encoder alone says nothing about where the cart is. Each observability matrix has rank two, and the two together have rank four.

Chapter 9 plots the four eigenvalues of the bicycle against speed and finds a window from 4.292 to 6.024 m/s in which all decay [4]. Chapter 8 reads the shapes of its two unstable modes from the eigenvectors, one folding the links and one leaning them together. Chapter 10 finds two unstable eigenvalues where an analysis at fixed speed finds one, and Chapter 2 tabulates the eigenvalues along its steady-state curve.

1.4.5 Pole Placement and the Linear Quadratic Regulator

A state feedback $u = -k_1\varphi - k_2\dot{\varphi}$ closes the upright arm on $s^2 + k_2 s + (k_1 - 4) = 0$, so any pair of poles can be placed. Poles at -4 and -4 need $k_1 = 20$ and $k_2 = 8$. Placement leaves open which poles to ask for. The LQR answers by cost: it minimizes $J = \int_0^\infty (x^\top Q x + R u^2)\,\mathrm{d}t$ with $K = R^{-1}B^\top P$, where P solves the algebraic Riccati equation of Equation (1.6).

$$A^\top P + PA - PBR^{-1}B^\top P + Q = 0 \tag{1.6}$$

For the arm with $Q = \mathrm{diag}(q_1, q_2)$ and $R = 1$ it has the closed form of Equation (1.7).

$$k_1 = 4 + \sqrt{16 + q_1}, \qquad k_2 = \sqrt{q_2 + 2k_1} \tag{1.7}$$

The weights $Q = \mathrm{diag}(20, 5)$ give $K = [10, 5]$ and poles at -2 and -3. As Q shrinks to zero, the gain tends to $[8, 4]$ and the poles to -2 twice. That limit is the cheapest stabilizing gain: it moves the unstable pole at $+2$ to its mirror image at -2 and leaves the stable one where it was. No controller holds the arm for less input energy, which makes the mirrored pole a floor on what stabilization costs.

Weights are usually set by Bryson's rule, one over the square of the largest acceptable excursion of each state and of the input [3]. Every single-input LQR keeps at least 60° of phase margin and tolerates a gain reduction to one half, a property of the optimal loop [17]. The apps design in discrete time, with the Riccati equation solved for the plant sampled at Δt. At $\Delta t = 0.01$ s the same weights give $K = [9.85, 4.93]$, close to the continuous gain.

Chapter 7 designs a discrete LQR on the cart and pole with Bryson's weights, and Chapter 8 shows that the regulator must push against the upper link's lean, which no pair of intuitive loops does. Chapter 5 compares an observer LQR with PD and PID, Chapter 9 schedules its LQR on speed, and Chapter 10 designs a two-input LQR at the drift equilibrium. Chapter 3 offers an LQR with integral action as a comparison.

1.4.6 Observers

The gain K needs the rate $\dot{\varphi}$, and an encoder reads only the angle. An observer runs a copy of the model and corrects it by the measurement error, as shown in Equation (1.8).

$$\dot{\hat{x}} = A\hat{x} + Bu + L(y - C\hat{x}) \tag{1.8}$$

The estimation error obeys $\dot{e} = (A - LC)e$, so L places the observer's poles as K placed the controller's [16]. For the arm the error polynomial is $s^2 + l_1 s + (l_2 - 4)$, and poles at -8 twice need $L = [16, 68]$. With $u = -K\hat{x}$ the closed loop has the controller's poles and the observer's poles together, -2, -3, -8 and -8. This is the separation principle. A Kalman filter is an observer whose gain is chosen from the noise statistics [15].

Figure 1.4 catches the arm from a tilt of 0.1 rad with $K = [10, 5]$ every 0.01 s, on an encoder of 4096 counts per turn. The three estimates of the rate, the true state, the observer and the encoder

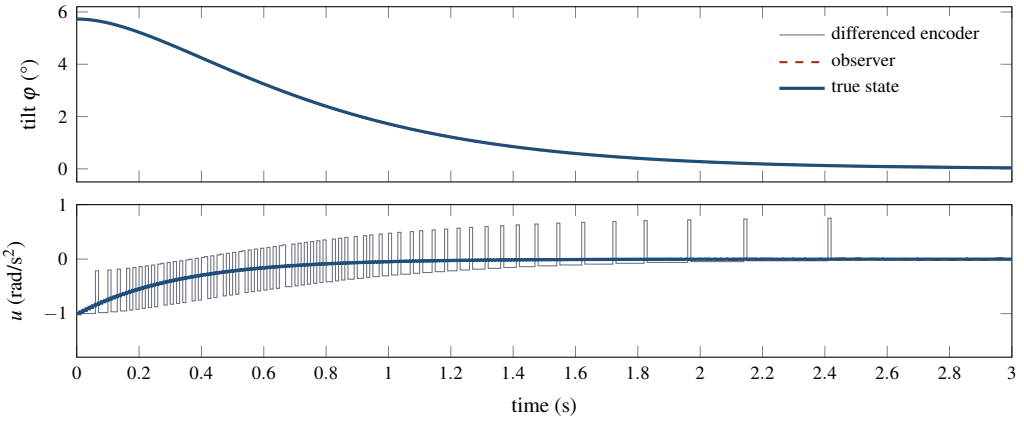


Figure 1.4 Catching the Upright Arm From 0.1 rad With $K = [10, 5]$ Every 0.01 s: the True State (Thick Solid), an Observer With Poles at -8 (Dashed) and the Encoder Differenced Over One Period (Thin Gray). The Tilt Is Nearly the Same, the Input Is Not.

differenced over one period, give nearly the same tilt. All three are within 0.01 rad of upright from 1.62 s. The input tells them apart. One count differenced over 0.01 s is a rate step of 0.15 rad/s, so the differenced law jumps by up to $1.55\,\text{rad/s}^2$ in one period. After 5 s its input has an RMS of $0.487\,\text{rad/s}^2$, 33 times the observer's $0.0146\,\text{rad/s}^2$.

Chapter 8 runs the same comparison on three encoders, with the true state, filtered differences and a Kalman filter side by side. Chapter 5 estimates the ball's speed with an α-β filter, a discrete observer with a double pole, and adds a four-state observer for its torque rig. Chapter 9 blends a gyro and an accelerometer in a complementary filter. Chapter 6, Chapter 12 and Chapter 2 use Kalman filters behind their predictive controllers, and Chapter 2 sets moving horizon estimation beside one, as Section 1.6.3 describes.

1.4.7 Limits Set by Right-Half-Plane Poles and Zeros

An unstable pole sets limits that no gain removes [20, 21]. The arm shows four of them through the mode σ of Equation (1.5).

The first is actuator headroom. With $|u| \le u_{\max}$, the mode $\dot\sigma = 2\sigma + u$ can be turned back only while $|\sigma| < u_{\max}/p$. With $u_{\max} = 2\,\text{rad/s}^2$ the linear model allows a tilt from rest of 0.5 rad and a push to 1 rad/s from upright. The nonlinear arm allows 30° from rest, where gravity's $4\sin\varphi$ equals the limit. This region belongs to the plant and the actuator. All three gains of Table 1.1, clipped at ± 2, catch every tilt up to 30° and none beyond, and no controller can do better.

The second is delay. Proportional feedback on σ through a delay τ_d is stable for some gain only if $p\,\tau_d < 1$. At $\tau_d = 0.1$ s the gain must lie between 2 and 14.5, at 0.25 s between 2 and 5.07, and at 0.5 s no gain works. The window has a floor at p and a ceiling that falls with the delay.

The third is lag. A PD law on the arm through an actuator lag τ gives $\tau s^3 + s^2 + (k_2 - 4\tau)s + (k_1 - 4) = 0$, and the Routh test reduces to Equation (1.9).

$$k_1 > 4, \qquad \tau < \frac{k_2}{k_1} \tag{1.9}$$

The lag must be shorter than the derivative time, whatever the gain. More gain does not help, because it raises both terms together.

The fourth is the RHP zero of the cart. Holding a lean φ needs the input $u = -4\varphi$, so the cart's position follows the lean through $x(s) = \ell\,(s^2 - 4)\,\varphi(s)/s^2$, which has a zero at $z = +2$. To move right, the cart must first move left to set up the lean. A loop around a RHP zero must be slow, and

Table 1.1 Margins of Three Stabilizing Gains on the Upright Arm

	Placement	LQR	Least energy
Gain $K = [k_1, k_2]$	$[20, 8]$	$[10, 5]$	$[8, 4]$
Closed-loop poles	$-4, -4$	$-2, -3$	$-2, -2$
Input for a 0.1 rad tilt (rad/s^2)	2.0	1.0	0.8
Lowest gain factor kept stable, $4/k_1$	0.20	0.40	0.50
Crossover (rad/s)	7.89	4.58	3.46
Phase margin (°)	72.4	66.4	60.0
Delay margin (s)	0.160	0.253	0.302
Longest actuator lag, k_2/k_1 (s)	0.40	0.50	0.50
Tilt caught from rest, $\lvert u\rvert \le 2$ (°)	30	30	30

a loop around a RHP pole must be fast, roughly a bandwidth below $z/2$ and above $2p$ [2]. On the upright cart this asks for an angle loop faster than 4 rad/s inside a cart loop slower than 1 rad/s. For a controller that reads only one output with both a pole p and a zero z, the sensitivity peak is at least $(z+p)/|z-p|$ [22]. With the drive setting the acceleration, $z = p$ and the bound is infinite, which restates that the cart's encoder alone cannot hold the pole.

Table 1.1 puts these limits beside the usual margins for three gains, with the loop broken at the input, $K(sI-A)^{-1}B = (k_2 s + k_1)/(s^2 - 4)$. The margins disagree. The fast placement has the most phase margin and the least delay margin, because its crossover is highest. The least-energy gain tolerates the longest delay, 0.302 s or 0.6 of the 0.5 s time constant of the unstable mode, but it loses the arm at the smallest drop in gain. An unstable loop also has a lower gain margin: below a factor $4/k_1$ the controller no longer outweighs gravity. No single number says how robust a stabilizing loop is.

> **+ DID YOU KNOW? The Most Phase Margin, the Least Delay Margin**
>
> From the simulator: the placement gain $[20, 8]$ keeps 72.4° of phase margin, the most of the three, yet a delay of 0.160 s loses the arm. The LQR gain $[10, 5]$, with 66.4°, tolerates 0.253 s (Table 1.1).

Chapter 6 derives the lag condition as $\tau_D > \tau_c$ and the gain floor as $K_c > p^2/B$. Chapter 7 maps the window of angle gain against delay, finds it closing at about 0.59 of the e-folding time, and bounds a loop on the cart position alone by a sensitivity peak of 46. Chapter 8 finds two RHP zeros and a sampling margin, Chapter 9 reads countersteering as a RHP zero, and Chapter 5 measures a limit on stopping the ball that no tuning overcomes.

1.4.8 Resonance and Input Shaping

Hanging, the cart's pendulum obeys $\ddot{\theta} + 4\theta = -u$. It is a resonance at 2 rad/s with no damping. Any cart move excites it. A single step of the cart's acceleration to 0.5 m/s^2 leaves the load swinging by $0.5/g = 0.051$ rad (2.92°) about its tilted rest. The move of Figure 1.5(a) accelerates at 0.5 m/s^2 for 2 s, coasts for 2 s and stops in 2 s, covering 4 m. It leaves a swing of 8.03°.

Input shaping removes the swing from the command instead of the feedback [7, 23]. An impulse leaves a sinusoidal residue, and two impulses half a period apart leave residues that cancel. The zero-vibration (ZV) shaper convolves the command with impulses of 1/2 at $t = 0$ and 1/2 at $\pi/\omega_n = 1.571$ s. The ZVD shaper also cancels the residue's derivative with respect to the frequency, using 1/4, 1/2 and 1/4 at 0, 1.571 and 3.142 s. Both amplitudes sum to one, so the cart still travels 4 m and never exceeds the unshaped acceleration. The move ends 1.57 s later with ZV and 3.14 s later with ZVD, and both leave less than 0.01° of swing.

The shaper is built from a model, and its weakness is a wrong frequency. For an undamped mode the ZV residue is $|\cos(\pi r/2)|$ of the unshaped one and the ZVD residue $\cos^2(\pi r/2)$, where r is the

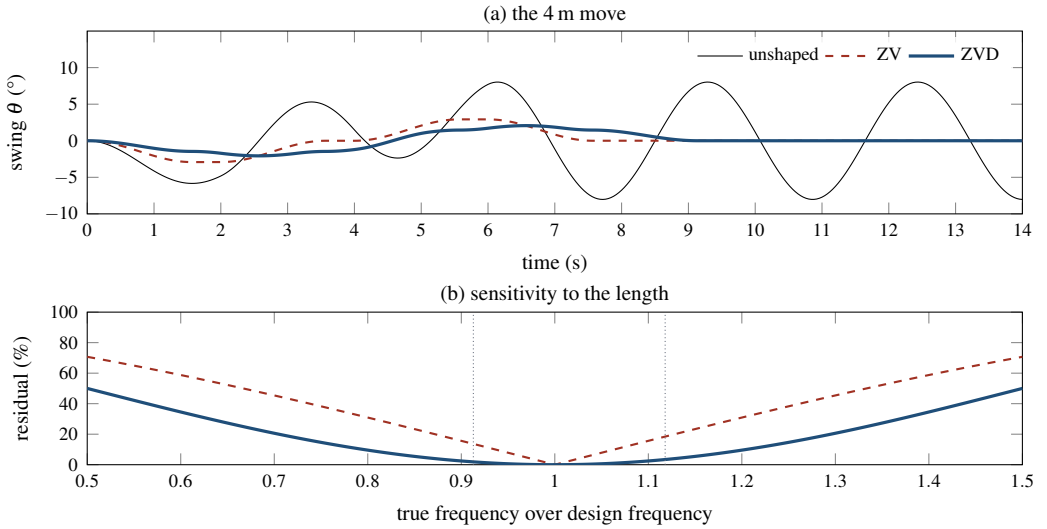


Figure 1.5 Input Shaping on the Hanging Cart. (a) The Swing During and After a 4 m Move, Unshaped (Thin Solid), With the Two-Impulse ZV Shaper (Dashed) and With the Three-Impulse ZVD Shaper (Thick Solid). (b) Residual Swing Against the Ratio of True to Design Frequency for ZV (Dashed) and ZVD (Thick Solid). The Dotted Lines Are a Cable 20 % Longer and 20 % Shorter Than the Design.

ratio of true to design frequency, as Figure 1.5(b) shows. A cable 20 % shorter than the design leaves 18.5 % of the unshaped swing with ZV and 3.4 % with ZVD. Keeping the residue below 5 % needs r within 0.968 to 1.032 for ZV and 0.856 to 1.144 for ZVD. The extra robustness costs one more half period of delay.

Chapter 12 applies both shapers on a crane whose true sway frequency sits 3.3 % above the one its shaper assumes, and measures what an assumed cable length costs. Chapter 11 treats the body bounce and wheel hop as two resonances and shows that at an invariant point near wheel hop no strut force changes the body acceleration. Chapter 13 carries one tool bending mode, and Chapter 5 shapes its reference as a trapezoid with an angle feedforward.

1.4.9 Limit Cycles

A limit cycle is an isolated periodic orbit. Nearby trajectories approach it or leave it, so its amplitude is set by the plant, not by the initial state [1]. Two mechanisms produce the limit cycles of this volume.

The first is a Hopf bifurcation, where a parameter moves a pair of eigenvalues across the imaginary axis. On the arm with $K = [10, 5]$, the actuator lag is such a parameter. At $\tau = 0.45\,\mathrm{s}$ the oscillatory pair is at $-0.111 \pm 2.58\,\mathrm{j}$, at 0.5 s it is at $\pm 2.45\,\mathrm{j}$, and at 0.55 s it is at $+0.091 \pm 2.33\,\mathrm{j}$, exactly as Equation (1.9) predicts. The linear loop then grows until a nonlinearity, such as an actuator limit, bounds it or loses the plant.

The second is a switching input with a delay. Let the arm's motor be a relay that applies $u = \pm 2$ against the sign of σ, measured τ_d late. Without the delay the relay drives σ to zero and holds it there, a sliding mode [11], and the tilt then decays with the mirrored pole at -2. With the delay, σ overshoots each switch, and Equation (1.5) gives the cycle exactly, as shown in Equation (1.10).

$$\sigma_{\max} = \frac{u_{\max}}{p}\left(\mathrm{e}^{p\tau_d} - 1\right), \qquad \frac{T_{\mathrm{cyc}}}{2} = \tau_d - \frac{1}{p}\ln\left(2 - \mathrm{e}^{p\tau_d}\right) \tag{1.10}$$

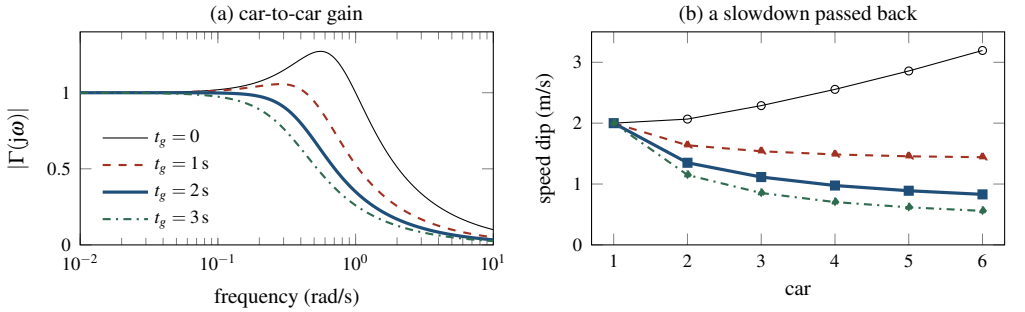


Figure 1.6 String Stability of Six Carts With $k_p = 0.5/\mathrm{s}^2$ and $k_d = 1/\mathrm{s}$. (a) The Car-to-Car Gain for Constant Spacing (Thin Solid) and Time Gaps of 1 (Dashed), 2 (Thick Solid) and 3 s (Dash-Dotted). (b) The Speed Dip of Each Car When the Lead Slows From 10 to 8 m/s and Back, With the Same Line Styles and Markers.

At $\tau_d = 0.1\,\mathrm{s}$ the cycle has $\sigma_{\max} = 0.221\,\mathrm{rad/s}$ and a period of 0.450 s, and the nonlinear arm gives 0.222 and 0.452 s with a tilt swing of 0.72°. The cycle exists only while $p\,\tau_d < \ln 2$, or $\tau_d < 0.347\,\mathrm{s}$. At 0.36 s the arm falls.

The describing function replaces the relay by its gain on the first harmonic of a sinusoid, $4u_{\max}/(\pi a)$ at the amplitude a, and finds the frequency where the linear part's phase is $-180°$ [24]. Here it predicts a period of 0.439 s, 2.6 % short, and an amplitude of σ of 0.176 rad/s, 20 % low, because σ is a train of exponentials rather than a sinusoid.

Chapter 2 locates a Hopf point on its hot branch and measures a relaxation oscillation with the jacket held just above the ignition fold. Chapter 4 finds that its sliding-mode law becomes a relay as its boundary layer shrinks, with a limit cycle the describing function puts at 15.6 Hz. Chapter 9 meets a saturated limit cycle of a fixed PD on a speed ramp. In Chapter 13 a delay turns the tool mode into chatter, and the tool leaving the cut bounds it to a limit cycle [25].

1.4.10 String Stability

Put six of the example's carts on one long rail, pendulums clamped, each following the cart ahead. Each follower sets its acceleration from the gap error and the gap's opening rate. With a time gap t_g the wanted gap grows with the follower's own speed, and the ratio of a follower's acceleration to its predecessor's is Equation (1.11), the form that Chapter 14 derives.

$$\Gamma(s) = \frac{k_d s + k_p}{(1 + k_d t_g)s^2 + (k_d + k_p t_g)s + k_p}, \qquad |\Gamma(\mathrm{j}\omega)|^2 \approx 1 + \omega^2\left(\frac{2}{k_p} - t_g^2\right) \tag{1.11}$$

The line is string stable when $|\Gamma(\mathrm{j}\omega)| \le 1$ at every frequency, so that no disturbance grows from car to car [19]. Because a lasting speed change passes whole, $\Gamma(0) = 1$, and the low-frequency expansion of Equation (1.11) needs $t_g \ge \sqrt{2/k_p}$.

With $k_p = 0.5/\mathrm{s}^2$ and $k_d = 1/\mathrm{s}$, Figure 1.6 shows the four cases. Constant spacing, $t_g = 0$, gives each follower a damping ratio of 0.707 and the line a peak gain of 1.272 at 0.556 rad/s. Every follower is well tuned and the line is not: when the lead slows by 2 m/s and recovers, the sixth car slows by 3.19 m/s. With $t_g = 1\,\mathrm{s}$ the peak falls to 1.057, a slow wave of 22 s period that grows by 5.7 % a car, too little to show in six cars and a factor of 3 over twenty. The bound $\sqrt{2/k_p} = 2\,\mathrm{s}$ is the shortest string-stable gap, and with it the sixth car slows by only 0.83 m/s. For constant spacing no linear law that sees only the car ahead is string stable [26].

Chapter 14 computes Γ from its cars' own law with lags, delays and a radio, adds the predecessor's broadcast acceleration, and measures the amplification along a nonlinear line of drivers and automated

Table 1.2 The Difficulty of Each Chapter, What Holds the Plant and the Margin Measured

Ch.	What makes it hard	What holds it	Margin measured
2	Two folds, an unstable middle branch, a negative steady gain	Reverse-acting PI, predictive control on the linearization	Largest setpoint step kept below the alarm
3	Stiffness that changes sign at horizontal	Feedforward with a schedule, or feedback linearization	Phase margin and least damping against the setpoint
4	Slip set at the friction peak, where the gain reverses	PI on slip, a sliding-mode law	Phase margin at the setpoint as the car slows
5	Two integrators and a lag, no damping	Filtered derivative, observer LQR	Phase margin, growth per swing
6	One real unstable pole	PID with a derivative longer than the coil lag	Gain window between the floor and the amplifier ceiling
7	Unstable root and a wrong-way zero	Cascade or LQR reading the angle	Gain window against delay, largest push
8	Two unstable modes, two RHP zeros	LQR on a Kalman filter	Longest stable sample interval, catchable tilt
9	Stability that depends on speed	Lean PD or LQR, scheduled on speed	Speed window, growth inside a band
10	Two unstable roots, a saturated tire	Two-input LQR scheduled on speed and friction	Recovery from a kick, a wet patch
11	Two resonances, invariant points	Skyhook and groundhook blend, preview	Tire load swing against liftoff
12	Lightly damped swing	Input shaping on a PID position loop	Residual swing against a wrong length
13	Delay that turns a tool mode into chatter	Adaptive feed, spindle speed selection	Depth against the stability lobe
14	Disturbances that grow along a line	Time gap, cooperative following	Peak car-to-car gain, shortest stable gap

cars. The criterion applies to any cascade of loops that passes a disturbance along, which Volume 6 meets in a supply chain.

1.4.11 What Stabilized Each Plant

The sections above give each plant two questions. What makes it hard: an unstable pole, a RHP zero, a sign change, a resonance, a delay or a chain? What holds it, and by how much: which structure stabilizes it, and which margin runs out first? The example answered both with numbers, and Table 1.1 showed that the answer depends on which margin is asked. Table 1.2 lists the difficulty of each chapter, the structure that holds its plant and the margin the chapter measures, as groundwork for the comparison in Chapter 15.

> ✦ DID YOU KNOW? **More Than a Hundred Controllers in One Car**
>
> A modern car has more than 100 electronic control units. Computer control entered cars in the late 1970s to meet stringent emission requirements, and it then took on suspension control, anti-lock braking, electronic braking and electronic stability control [6], the subjects of Chapter 4, Chapter 11 and Chapter 10.

1.5 Instruments and Final Elements in This Volume

A stabilizing loop acts only on what its sensors report and only as hard as its final element allows. Figure 1.7 lists what the controllers of each chapter read and what they move. An angle or position encoder appears in seven chapters, and an electric motor or servo drive is the final element in seven. Two chapters stand apart. The controller of Chapter 2 reads a temperature in a thermowell and moves a jacket temperature that a slave loop sets through utility valves, and those of Chapter 14 read a radar and a radio and move the throttle and brakes.

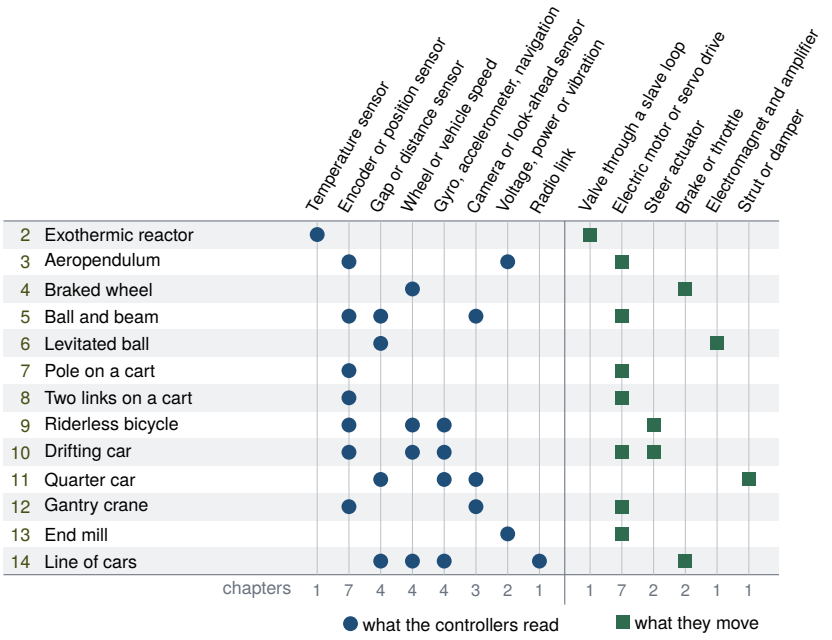


Figure 1.7 What the controllers of each chapter read (round marks) and move (square marks), from each chapter's variables table and technology section. A computed value, such as a slip, is not marked, and the crane's swing sensor serves only its predictive controller. Notice that encoders and electric drives carry the unstable mechanical plants.

An incremental encoder gives two pulse trains about a quarter of a period apart, and the one that leads tells the direction of travel [27]. It reports no speed, so the rate that a state feedback needs is differenced or estimated, the comparison of Figure 1.4.

The drives are mostly electric. A variable frequency drive sets the speed of an alternating-current motor by the frequency of the power it feeds and the torque by the voltage and current, and it can brake the machine by running the motor as a generator [27], as Figure 1.8 shows on a pump. The trolley of Chapter 12 is pulled by such a motor. Most chapters model their drive as a first-order lag with a limit, and several add a slew limit. That limit, not the controller, bounds the tilt that can be caught, as Section 1.4.7 showed on the arm. The one process plant of the volume reads a temperature, top of the figure, and Chapter 2 puts the lag of its thermowell at tens of seconds.

The laws sit in layers. Supervisory and lower-level loops execute with cycle times from milliseconds to minutes, and programmable logic controllers serve as local controllers under a supervisor in larger systems [28]. Figure 1.9 places the loops of this volume on their platforms. The laws of the mechanical plants run every 1 ms to 10 ms, the platoon's spacing law every 50 ms and the reactor's PI loop every 3 s. The drive loops beneath them run faster still, at 1 to 2 kHz in the speed controller of a multirotor's rotor (Chapter 3).

1.6 The Simulator Framework

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

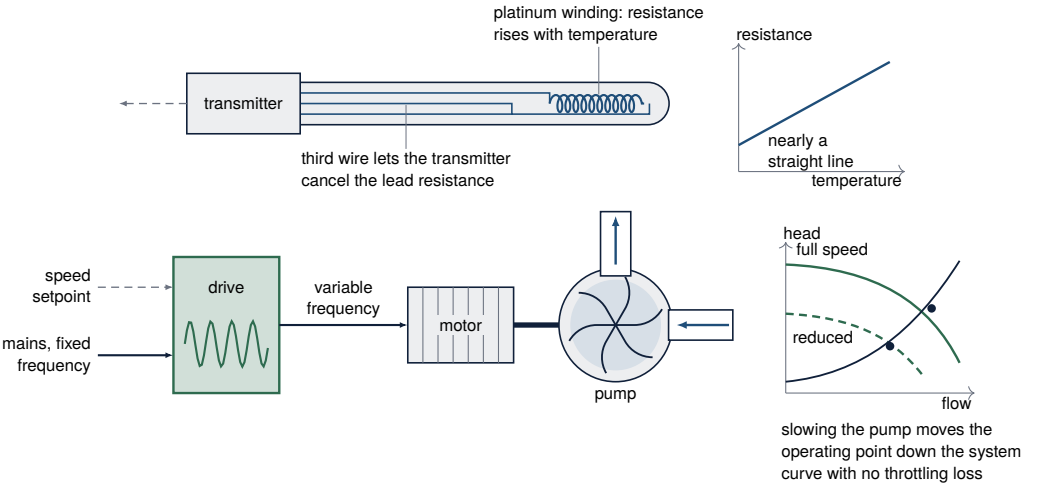


Figure 1.8 Top: principle of a resistance thermometer. Bottom: principle of a variable-speed drive on a centrifugal pump. Notice that the thermometer's resistance follows the temperature nearly in a straight line, and that the drive moves the machine onto a new curve by its speed alone, the way the trolley and spindle drives of this volume take a speed or torque command.

Layer	What it does in this volume	Where it runs	How often
Supervision or planning	a platoon's spacing supervisor, a chatter watch and speed rule, a predictive controller	**Supervisory level** of a machine controller; a **prototyping computer** on the vehicle bus	50 ms in the platoon; a predictive controller in its law's slot
Stabilizing law	state feedback, PID or PD with feedforward, a schedule, a sliding-mode law	**Microcontroller**, **electronic control unit**, **PLC**, **real-time computer**; a **DCS** for the reactor	1 ms to 10 ms on the machines and cars; 3 s on the reactor
Drive and inner loops	a rotor's speed loop, a servo or torque loop, a coil's current, a jacket's slave loop	**Electronic speed controller**, **servo drive**, **frequency inverter**, **current amplifier**	1 ms in the torque beam; 1 to 2 kHz in a rotor's speed controller
Sensors and final elements	encoders, gyros, accelerometers, a radar; motors, a steer, brakes, a strut, a coil	wires, a slip ring, a vehicle bus and a radio link	a camera frame every 33 ms; a radio broadcast every 100 ms proposed

Figure 1.9 The control layers of this volume, the platform that hosts each and how often each acts, from [6, 28] and the chapters' technology sections. Notice that only the reactor and the platoon act more slowly than every 10 ms.

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

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

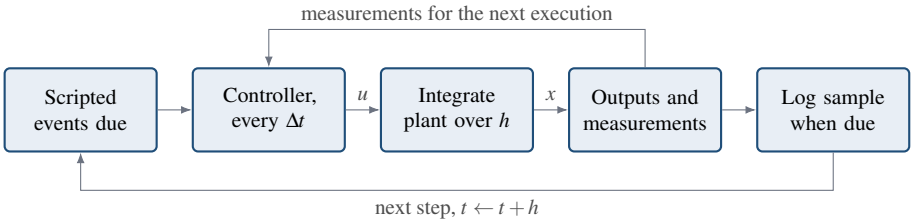


Figure 1.10 Work of the Engine in One Integration Step

1.6.1 The Simulation Engine

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

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

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

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

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

1.6.2 The Velocity-Form PID Algorithm

One PID implementation serves the PID loops of every framework app. It is the velocity (incremental) form: each execution adds the change in the proportional and derivative terms to a running sum, and the integral is a separate term, as shown in Equations (1.12) 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 [29]. 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 $I_k = u_{\mathrm{man}} - u_b - S_k$, so the output it would send equals the manual value u_{man}. The first automatic output differs from it by one period's change in P and D. In Auto the engine copies each controller output to the manual station, so a switch to Manual is also bumpless. The standalone studios with PID loops use velocity-form code of their own, with the limit on the output and anti-windup on the integral alone, in which a gain change applies only to later increments.

1.6.3 The Predictive Control and Estimator Options

More than 100 apps offer model predictive control (MPC) as an option beside the structure they open with, which stays the default. Volume 9 develops the theory. In most of them the controller plans with a small linear model in discrete time as shown in Equation (1.16).

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

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 [30]. 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 [31]. 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 [32].

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.

A gain that changes with the point	An unstable equilibrium held by feedback		A resonance or a delay	A chain of loops
Sections 1.4.2, 1.4.3	Sections 1.4.4 to 1.4.7		Sections 1.4.8, 1.4.9	Section 1.4.10
(2) folds and a negative steady gain	(5) two integrators and a lag	(8) two unstable modes no loop pair holds	(11) two resonances, one strut force	(14) disturbances that grow along a line
(3) a stiffness that changes sign	(6) one unstable pole, a capped gain	(9) stability set by speed	(12) a lightly damped swing	
(4) slip at the friction peak	(7) an unstable root and a wrong-way zero	(10) an unstable drift with a saturated tire	(13) a delay that makes a tool mode chatter	

Figure 1.11 Method map of the volume: each chapter by the kind of difficulty of Section 1.1, with its lesson from Table 1.4 and the subsections it rests on. Notice that six of the thirteen chapters are unstable mechanical plants held by feedback.

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

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

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

How the exercises train the habit. One exercise in each chapter is marked *Review the agent*. It presents a tuning, a model or a claim as an AI assistant might propose it, with a flaw, and asks the reader to find the flaw in about five minutes. The flaws are ordinary ones: a unit, a sign, a steady state that does not close, a result measured on the wrong scenario or at a different run length, or a limit that was ignored. Each is found with one of the tests above. The purpose is not distrust. An engineer who can say why an answer is right can hand an assistant much larger tasks and still stand behind the result. That larger picture, in which agents do most of the routine work and the engineer supervises and checks, is the subject of Chapter 15.

1.8 Chapter Roadmap

Table 1.4 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 1.11 places the same chapters by the kind of difficulty that Section 1.1 sorts them into.

Chapters 3, 5, 6 and 12 form a first course: a sign-changing stiffness, a double integrator, one unstable pole and one resonance, each held by a PID-type loop. They rest on Sections 1.4.2, 1.4.3, 1.4.7 and 1.4.8. Chapters 7, 8 and 10 are best read in that order, after Sections 1.4.4 to 1.4.7, since each adds a state or an input to the last. Chapters 2 and 4 rest on Section 1.4.2 and can be read at any

Table 1.4 Roadmap of the Volume

Ch.	Plant	The lesson	Level	Time
2	Exothermic stirred-tank reactor	Folds, an unstable operating point and a negative steady gain held by a PI loop.	intermediate	3 h
3	Propeller arm on a pivot	A stiffness that changes sign. Feedforward, a schedule and feedback linearization.	introductory	2.5 h
4	Braked wheel of a car	Slip at the friction peak where the gain reverses. PI against a sliding-mode law.	intermediate	3 h
5	Ball on a tilting beam	Two integrators and a lag. Derivative as the only damping, an observer LQR.	introductory	2.5 h
6	Ball under an electromagnet	One unstable pole. A derivative longer than the lag, a gain capped by the amplifier.	introductory	2.5 h
7	Pole on a cart	An unstable root and a wrong-way zero. Cascade, LQR and the recoverable push.	intermediate	3 h
8	Two links on a cart	Two unstable modes no loop pair holds. LQR on a Kalman filter, the sampling margin.	advanced	2.5 h
9	Riderless bicycle	Stability set by speed. Countersteering as a RHP zero, a scheduled LQR.	intermediate	2.5 h
10	Drifting car on a skidpad	An unstable equilibrium with a saturated tire. A two-input LQR, scheduled.	advanced	2 h
11	Quarter car	Two resonances and invariant points. Comfort against road holding through one force.	intermediate	2.5 h
12	Gantry crane	A lightly damped swing. Input shaping and its sensitivity to the cable length.	introductory	3 h
13	End mill cutting a pocket	A delay that makes a tool mode chatter. Stability lobes and the spindle speed.	intermediate	2 h
14	Line of cars	String stability. Time gap, cooperation and delay along a line.	intermediate	2.5 h

point. Chapters 9, 11, 13 and 14 each stand alone after the subsection they use. A reader who has taken a course in state-space control can take a shorter path through Chapters 2, 8, 10, 13 and 14.

KEY POINTS

- A plant that would fall, run away or ring is held only near the point its controller was designed for, and only while the actuator can push hard and fast enough.
- A linearization describes the plant near one point. A gain schedule or feedback linearization keeps one tuning valid across points, and both need the model to be right.
- An unstable pole sets limits that no gain removes: actuator headroom, a delay shorter than the pole's time constant, a lag shorter than the derivative time.
- No single margin says how robust a stabilizing loop is, and a line of well-tuned loops can amplify a disturbance.

Where else this applies. A gain schedule on a gain that moves by four orders of magnitude in Volume 2, Chapter 6, a limit cycle past the peak of a compressor characteristic in Volume 4, Chapter 2, and a lander whose position loop meets a right-half-plane zero in Volume 3, Chapter 10.

References

[1] S. H. Strogatz, *Nonlinear Dynamics and Chaos*. CRC Press, 2018.

[2] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control: Analysis and Design*, 2nd ed. John Wiley & Sons, 2005.

[3] A. E. Bryson and Y.-C. Ho, *Applied Optimal Control: Optimization, Estimation, and Control*. Routledge, 2018.

[4] J. P. Meijaard, J. M. Papadopoulos, A. Ruina, and A. L. Schwab, "Linearized dynamics equations for the balance and steer of a bicycle: A benchmark and review," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 463, no. 2084, pp. 1955–1982, 2007.

[5] M. H. Tuttle, R. A. Kilgore, and R. P. Boyden, "Magnetic suspension and balance systems: A selected, annotated bibliography," NASA Langley Research Center, Hampton, Virginia, NASA Technical Memorandum TM-84661, Jul. 1983. [Online]. Available: https://ntrs.nasa.gov/citations/20080047362

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

[7] O. J. M. Smith, "Posicast control of damped oscillatory systems," *Proceedings of the IRE*, vol. 45, no. 9, pp. 1249–1255, 1957.

[8] D. J. Leith and W. E. Leithead, "Survey of gain-scheduling analysis and design," *International Journal of Control*, vol. 73, no. 11, pp. 1001–1025, 2000.

[9] W. Levine and M. Athans, "On the optimal error regulation of a string of moving vehicles," *IEEE Transactions on Automatic Control*, vol. 11, no. 3, pp. 355–361, 1966.

[10] A. Uppal, W. H. Ray, and A. B. Poore, "On the dynamic behavior of continuous stirred tank reactors," *Chemical Engineering Science*, vol. 29, no. 4, pp. 967–985, 1974.

[11] V. Utkin, "Variable structure systems with sliding modes," *IEEE Transactions on Automatic Control*, vol. 22, no. 2, pp. 212–222, 1977.

[12] J. Hauser, S. Sastry, and P. Kokotović, "Nonlinear control via approximate input-output linearization: the ball and beam example," *IEEE Transactions on Automatic Control*, vol. 37, no. 3, pp. 392–398, 1992.

[13] K. J. Åström and K. Furuta, "Swinging up a pendulum by energy control," *Automatica*, vol. 36, no. 2, pp. 287–295, 2000.

[14] R. Tedrake, I. R. Manchester, M. Tobenkin, and J. W. Roberts, "LQR-trees: Feedback motion planning via sums-of-squares verification," *The International Journal of Robotics Research*, vol. 29, no. 8, pp. 1038–1052, 2010.

[15] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[16] D. G. Luenberger, "Observing the state of a linear system," *IEEE Transactions on Military Electronics*, vol. 8, no. 2, pp. 74–80, 1964.

[17] R. E. Kalman, "When is a linear control system optimal?" *Journal of Basic Engineering*, vol. 86, no. 1, pp. 51–60, 1964.

[18] J. T. Bosworth, "Success stories in control: Nonlinear dynamic inversion control," NASA Armstrong Flight Research Center, report DFRC-E-DAA-TN1713, preprint, NASA Technical Reports Server 20110008197, https://ntrs.nasa.gov/citations/20110008197, 2010.

[19] D. Swaroop and J. K. Hedrick, "String stability of interconnected systems," *IEEE Transactions on Automatic Control*, vol. 41, no. 3, pp. 349–357, 1996.

[20] G. Stein, "Respect the unstable," *IEEE Control Systems Magazine*, vol. 23, no. 4, pp. 12–25, 2003.

[21] K. J. Åström, "Limitations on control system performance," *European Journal of Control*, vol. 6, no. 1, pp. 2–20, 2000.

[22] J. Freudenberg and D. Looze, "Right half plane poles and zeros and design tradeoffs in feedback systems," *IEEE Transactions on Automatic Control*, vol. 30, no. 6, pp. 555–565, 1985.

[23] N. C. Singer and W. P. Seering, "Preshaping command inputs to reduce system vibration," *Journal of Dynamic Systems, Measurement, and Control*, vol. 112, no. 1, pp. 76–82, 1990.

[24] K. J. Åström and T. Hägglund, "Automatic tuning of simple regulators with specifications on phase and amplitude margins," *Automatica*, vol. 20, no. 5, pp. 645–651, 1984.

[25] J. Tlusty and F. Ismail, "Basic non-linearity in machining chatter," *CIRP Annals*, vol. 30, no. 1, pp. 299–304, 1981.

[26] P. Seiler, A. Pant, and K. Hedrick, "Disturbance propagation in vehicle strings," *IEEE Transactions on Automatic Control*, vol. 49, no. 10, pp. 1835–1841, 2004.

[27] 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.

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

[30] 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.

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

[32] 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 · 4 min

The arm of Equation (1.1), $\ddot{\theta} = -4\sin\theta + u$ with θ from hanging, is held by the constant input $u_0 = 2\sqrt{3} = 3.46\,\mathrm{rad/s^2}$. Find both steady states between 0 and 180°, the steady-state gain $\mathrm{d}\theta_0/\mathrm{d}u_0$ at each, and the eigenvalues of the linearization at each.

1.2 Concept check · 2 min

The arm is to be held at 150°, where its steady-state gain is -0.289 rad per rad/s^2. The PD law with gravity feedforward of Equation (1.3) is used, with $k_d = 4$. Which statement is right?

(a) k_p and k_d must both change sign, because the steady-state gain is negative

(b) only k_p changes sign, and k_d keeps its sign because the rate response is unchanged

(c) no gains hold the arm there, because the plant is unstable in open loop

(d) both keep their signs, and k_p must exceed 3.46 to keep the total stiffness positive

1.3 Theory and calculation · 6 min

The PD law with gravity feedforward of Equation (1.3) holds the arm with $k_p = 8$ and $k_d = 4$.

(a) Write the closed-loop polynomial at a setpoint of 120° and find its natural frequency and damping ratio. *Checkpoint:* $s^2 + 4s + 6$, 2.45 rad/s, 0.816.

(b) With $k_p = 2$ instead, find the largest setpoint angle the loop holds. *Checkpoint:* 120°.

(c) A schedule keeps the polynomial $s^2 + 4s + 12$ that $k_p = 8$ gives hanging. Find the scheduled k_p at 120°. *Checkpoint:* 14.

1.4 Theory and calculation · 5 min

For the upright arm, $\ddot{\varphi} = 4\varphi + u$, an LQR uses the weights $Q = \mathrm{diag}(65, 10)$ and $R = 1$. Use the closed form of Equation (1.7) to find K and the closed-loop poles. Then find the lowest factor by which K can be scaled down before the arm falls, and the longest actuator lag the unscaled loop tolerates by Equation (1.9).

1.5 Theory and calculation · 4 min

An observer of Equation (1.8) for the upright arm reads the angle only, $C = [1, 0]$. Find the gain $L = [l_1, l_2]$ that places both observer poles at -10, and list the closed-loop eigenvalues when the estimate feeds $K = [10, 5]$.

1.6 Concept check · 3 min

The upright cart of the running example has a drive that sets its acceleration, $\ddot{x} = \ell u$, and an encoder on the cart's position but none on the pole. Which statement is right?

(a) no controller holds the pole, because the pole's motion never reaches the cart's encoder

(b) an observer with poles fast enough recovers the tilt from the cart's position

(c) an LQR with a heavy weight on the cart's position holds it

(d) it can be held, with a sensitivity peak of at least 46

1.7 Theory and calculation · 6 min

The upright arm has the unstable pole $p = 2/\mathrm{s}$.

(a) The placement gain $K = [20, 8]$ acts through an actuator lag τ. Find the longest lag it tolerates and the frequency of the oscillation that appears there. *Checkpoint:* 0.4 s, 4 rad/s.

(b) A proportional law acts on the unstable mode σ through a pure delay τ_d. Find the longest delay for which some gain stabilizes it, and the longest delay for which a relay of ± 2 on σ keeps a limit cycle instead of losing the arm. *Checkpoint:* 0.5 s and 0.347 s.

1.8 Theory and calculation · 5 min

A crane load hangs on a cable of 9.81 m, so that $\omega_n = \sqrt{g/\ell} = 1\,\mathrm{rad/s}$, with negligible damping. Give the impulse times and amplitudes of the ZV and ZVD shapers, the delay each adds to a move, and the residual swing each leaves, as a percentage of the unshaped one, if the cable is in fact 10 % longer.

1.9 Theory and calculation · 4 min

Followers in a line use the time-gap law of Equation (1.11) with $k_p = 2/\mathrm{s}^2$ and $k_d = 1/\mathrm{s}$. Find the shortest string-stable time gap from the low-frequency condition, and the sign of the coefficient of ω^2 in $|\Gamma|^2$ at $t_g = 0.6\,\mathrm{s}$. Say what that sign means for a slow wave passing down the line.

1.10 Concept check · 3 min

The motor of the upright arm is replaced by one with $u_{\max} = 3\,\mathrm{rad/s}^2$. Which tilts from rest can a stabilizing state feedback, clipped at the limit, recover?

(a) tilts up to 28.6°, the same as before, because the unstable pole has not changed

(b) any tilt, once the gain K is large enough

(c) up to 0.75 rad (43.0°) on the linear model and 48.6° on the nonlinear arm, at best

(d) up to 0.75 rad only if the LQR weight on the input is made small