Feedback for Physicists: A Tutorial Essay on Control

John Bechhoefer*

Department of Physics, Simon Fraser University, Burnaby, British Columbia, V5A 1S6, Canada

(Dated: November 30, 2004)

Feedback and control theory are important ideas that should form part of the education of a physicist but rarely do. This tutorial essay aims to give enough of the formal elements of control theory to satisfy the experimentalist designing or running a typical physics experiment and enough to satisfy the theorist wishing to understand its broader intellectual context. The level is generally simple, although more advanced methods are also introduced. Several types of applications are discussed, as the practical uses of feedback extend far beyond the simple regulation problems where it is most often employed. The author then sketches some of the broader implications and applications of control theory, especially in biology, which are topics of active research.

Contents

I. INTRODUCTION

Feedback and its big brother, control theory, are such important concepts that it is odd that they usually find no formal place in the education of physicists. On the practical side, experimentalists often need to use feedback. Almost any experiment is subject to the vagaries of environmental perturbations. Usually, one wants to vary a parameter of interest while holding all others constant. How to do this properly is the subject of control theory. More fundamentally, feedback is one of the great ideas developed (mostly) in the last century,[1] with par-

*Electronic address: johnb@sfu.ca

[1] Feedback mechanisms regulating liquid level were described over two thousand years ago, while steam-engine "governors" date back to the 18th century. However, realization of the broader implications of feedback concepts, as well as their deeper analysis and widespread application, date to the 20th century. Chapt. 1 of Franklin *et al.* (2002) gives a brief historical review of the development of control theory. In a more detailed account, Mayr (1970) describes a number of early feedback devices, from classical examples (Ktesibios and Hero, both of Alexandria) to a scattering of medieval Arab accounts. Curiously, the modern rediscovery of feedback control took place entirely in England, at

ticularly deep consequences for biological systems, and all physicists should have some understanding of such a basic concept. Indeed, further progress in areas of current interest such as systems biology is likely to rely on concepts from control theory.

This article is a tutorial essay on feedback and control theory. It is a tutorial in that I give enough detail about basic methods to meet most of the needs of experimentalists wishing to use feedback in their experiments. It is an essay in that, at the same time, I hope to convince the reader that control theory is useful not only for the engineering aspects of experiments but also for the conceptual development of physics. Indeed, we shall see that feedback and control theory have recently found applications in areas as diverse as chaos and nonlinear dynamics, statistical mechanics, optics, quantum computing, and biological physics. This essay supplies the background in control theory necessary to appreciate many of these developments.

The article is written for physics graduate students and interested professional physicists, although most of it should also be understandable to undergraduate students. Beyond the overall importance of the topic, this article is motivated by a lack of adequate alternatives. The obvious places to learn about control theory — introductory engineering textbooks (Dutton *et al.*, 1997; Franklin *et al.*, 2002, 1998) — are not very satisfactory places for a physicist to start. They are long — 800 pages is typical — with the relevant information often scattered in different sections. Their examples are understandably geared more to the engineer than to the physicist. They often cloak concepts familiar to the physicist in unfamiliar language and notation. And they do not make connections to similar concepts that physicists will have likely encountered in standard courses. The main alternative, more mathematical texts (e.g., in order of increasing sophistication, the books by Özbay (2000), Morris (2001), Doyle *et al.* (1992), and Sontag (1998)), are terse but assume the reader already has an intuitive understanding of the subject.

At the other end of the intellectual spectrum, the first real exposure of many experimentalists to control theory comes when they are faced with having to use or modify a PID ("proportional-integral-derivative") control loop that regulates some experimental quantity, such as temperature or pressure. Lacking the time to delve more deeply into the subject, they turn to the semi-qualitative discussions found in the appendices of manuals for commercial regulators and the like. While these can be enough to get by, they rarely give optimal performance and certainly do not give a full idea of the range of possible approaches to a particular experimental problem. Naturally, they also do not give an appreciation for the broader uses of control theory.

Thirty years ago, E. M. Forgan wrote an excellent introduction to control theory for experimental physicists, "On the use of temperature controllers in cryogenics," (Forgan, 1974) which, despite its seemingly narrow title, addresses many of the needs of the experimentalist discussed above. However, it predates the widespread use of digital control and time-domain methods, as well as important advances in control theory that have taken place over the past three decades. In one sense, this essay is an updated, slightly more accessible version of Forgan's article, but the strategies for control and the range of applications that are relevant to the physicist are much broader than what is implied in that earlier work. I have tried to give some feeling for this breadth, presenting simple ideas and simple cases in some detail while sketching generalizations and advanced topics more briefly.

The plan of this essay, then, is as follows: In Section II, we review some elementary features of dynamical systems at the level of an intermediate undergraduate mechanics course. In Section III, we introduce the simplest tools of control theory, feedback and feedforward. As case studies, we discuss how adding a control loop can increase the usefulness of a Michelson interferometer as a displacement sensor and how feedback plays an essential role in modern analog electronic circuits. In Section IV, we discuss the relationship between feedback and stability, focusing on how time delays generically arise and can limit the amount of feedback gain that may be applied. In Section V, we discuss various practical issues of implementation: the identification of system dynamics, the choice of control algorithm and the tuning of any parameters, the translation of continuous-time designs to discrete difference equations suitable for programming on a computer, the use of commercial software packages to simulate the effects of control on given systems, and the problems posed by noisy sensors and other types of uncertainty. We include a case study of an active vibration-isolation system. This section includes introductions to a number of advanced topics, including model reduction, optimal control, Kalman filtering, and robust methods. We deliberately intersperse these discussions in a session on practical implementation, for the need to tune many parameters, to deal with noise and uncertainties in the form of the system model itself – all require more advanced methods. Indeed, historically it has been the fail-

---

the beginning of the industrial revolution of the 18th century. Mayr speculates that the concept of feedback arose there and not elsewhere because it fit in more naturally with the prevailing empiricist philosophy of England and Scotland (e.g., Hume and Locke). On the Continent, the rationalist philosophy of Descartes and Leibniz postulated preconceived goals that were to be determined and achieved via *a priori* planning and not by comparison with experience. While Mayr's thesis might at first glance seem farfetched, the philosophical split between empiricism and rationalism was in fact reflected in various social institutions, such as government (absolute vs. limited monarchy), law (Napoleonic code vs. case law), and economics (mercantilism vs. liberalism). Engineering (feedback control vs. top-down design) is perhaps another such case. Mayr devoted much of his subsequent career to developing this thesis (Bennett, 2002). Elsewhere, Bennett (1996) gives a more detailed history of control theory and practice in the 20th century.

ure of simpler ideas that has motivated more complex methods. Theorists should not skip this section!

In Section VI, we note the limitations of standard control theory to (nearly) linear systems and discuss its extension to nonlinear systems. Here, the engineering and physics literature have trod their separate ways, with no intellectual synthesis comparable to that for linear systems. We point out some basic issues. In Section VII, we discuss biological applications of feedback, which lead to a much broader view of the subject and make clear that an understanding of "modular" biology presupposes a knowledge of the control concepts discussed here. In Section VIII, we mention very briefly a few other major applications of feedback and control, mainly as a pointer to other literature. We also briefly discuss adaptive control – a vast topic that ranges from simple parameter estimation to various kinds of artificial-intelligence approaches that try to mimic human judgment. In Section IX, we discuss some of the relations between feedback and information theory, highlighting a recent attempt to explore the connections.

Finally, we note that while the level of this tutorial is relatively elementary, the material is quite concentrated and takes careful reading to absorb fully. We encourage the reader to "browse" ahead to lighter sections. The case studies in Secs. III.D, V.C.1, and VII.A are good places for this.

II. BRIEF REVIEW OF DYNAMICAL SYSTEMS

In one way or another, feedback implies the modification of a dynamical system. The modification may be done for a number of reasons: to regulate a physical variable, such as temperature or pressure; to linearize the response of a sensor; to speed up a sluggish system or slow down a jumpy one; to stabilize an otherwise unstable dynamics. Whatever the reason, one always starts from a given dynamical system and creates a "better" one.

We begin by reviewing a few ideas concerning dynamical systems. A good reference here and for other issues discussed below (bifurcations, stability, chaos, etc.) is the book by Strogatz (1994). The general dynamical system can be written

$$\dot{\vec{x}} = \vec{f}(\vec{x},\vec{u}) \tag{2.1a}$$
$$\vec{y} = \vec{g}(\vec{x},\vec{u}) \,, \tag{2.1b}$$

where the vector $\vec{x}$ represents n independent "states" of a system, $\vec{u}$ represents m independent inputs (driving terms), and $\vec{y}$ represents p independent outputs. The vector-valued function $\vec{f}$ represents the (nonlinear) dynamics and $\vec{g}$ translates the state $\vec{x}$ and "feeds through" the input $\vec{u}$ directly to the output $\vec{y}$. The role of Eq. (2.1b) is to translate the perhaps-unobservable state variables $\vec{x}$ into output variables $\vec{y}$. The number of internal state variables (n) is usually greater than the number of accessible output variables (p). Eqs. (2.1) differ from the dynamical systems that physicists often study in that the inputs $\vec{u}$ and outputs $\vec{y}$ are explicitly identified.

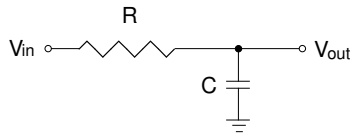

FIG. 1 Low-pass electrical filter.

Mostly, we will deal with simpler systems that do not need the heavy notation of Eq. (2.1). For example, a linear system can be written

$$\dot{\vec{x}} = \widetilde{A}\vec{x} + \widetilde{B}\vec{u} \tag{2.2a}$$
$$\vec{y} = \widetilde{C}\vec{x} + \widetilde{D}\vec{u} \,, \tag{2.2b}$$

where the dynamics $\widetilde{A}$ are represented as an $n\times n$ matrix, the input coupling $\widetilde{B}$ as an $n \times m$ matrix, the output coupling $\widetilde{C}$ as a $p \times n$ matrix, and $\widetilde{D}$ is a $p \times m$ matrix. Often, the "direct feed" matrix $\widetilde{D}$ will not be present.

As a concrete example, many sensors act as a low-pass filter[2] and are equivalent to the electrical circuit shown in Fig. 1, where one finds

$$\dot{V}_{out}(t) = -\frac{1}{RC}V_{out}(t) + \frac{1}{RC}V_{in}(t) \,. \tag{2.3}$$

Here, $n = m = p = 1$, $x = y = V_{out}$, $u = V_{in}$, $A = -1/RC$, $B = 1/RC$, $C = 1$, and $D = 0$.

A slightly more complicated example is a driven, damped harmonic oscillator, which models the typical behavior of many systems when slightly perturbed from equilibrium and is depicted in Fig. 2, where

$$m\ddot{q} + 2\gamma\dot{q} + kq = kq_o(t) \,. \tag{2.4}$$

We can simplify Eq. 2.4 by scaling time by $\omega_0^2 = k/m$, the undamped resonant frequency, and by defining $\zeta = \gamma/m\cdot\sqrt{m/k} = \gamma/\sqrt{mk}$ as a dimensionless damping parameter, with $0 < \zeta < 1$ for an underdamped oscillator and $\zeta > 1$ for an overdamped system. (In the physics literature,

[2] One finds second-order behavior in other types of sensors, too, including thermal and hydraulic systems. For thermal systems (Forgan, 1974), mass plays the role of electrical capacitance and thermal resistivity (inverse of thermal conductivity) the role of electrical resistance. For hydraulic systems, the corresponding quantities are mass and flow resistance (proportional to fluid viscosity). At a deeper level, such analogies arise if two conditions are met: (1) the system is near enough thermodynamic equilibrium that the standard "flux $\propto$ gradient" rule of irreversible thermodynamics applies; (2) any time dependence must be at frequencies low enough that each physical element behaves as a single object ("lumped-parameter" approximation). In Sec. IV.C, we consider a situation where the second condition is violated.

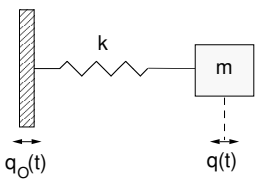

FIG. 2 Driven simple-harmonic oscillator.

one usually defines the quality factor $Q = 1/\zeta$.) Then, we have

$$\ddot{q} + 2\zeta\dot{q} + q = q_o(t) \ . \tag{2.5}$$

To put this in the form of Eq. 2.2, we let $x_1 = q$, $x_2 = \dot{q}$. Then $n = 2$ (2nd-order system) and $m = p = 1$ (one input, $u = q_0$, and one output, q), and we have

$$\frac{d}{dt}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -1 & -2\zeta \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix}\cdot u(t) \ , \tag{2.6}$$

with

$$y = (1 \quad 0)\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + 0 \cdot u(t) \ , \tag{2.7}$$

where the matrices $\widetilde{A}$, $\widetilde{B}$, $\widetilde{C}$, and $\widetilde{D}$ are all written explicitly. In such a simple example, there is little reason to distinguish between x_1 and y, except to emphasize that one observes only the position, not the velocity.[3] Often, one observes linear combinations of the state variables. (For example, in a higher-order system, a position variable may reflect the influence of several modes. Imagine a cantilever anchored at one end and free to vibrate at the other end. Assume that one measures only the displacement of the free end. If several modes are excited, this single output variable y will be a linear combination of the individual modes with displacements $\vec{x}_i$.) Note that we have written a second-order system as two coupled first-order equations. In general, an nth-order linear system can be converted to n first-order equations.

The above discussion has been carried out in the time domain. Often, it is convenient to analyze linear equations — the focus of much of practical control theory — in the frequency domain. Physicists usually do this via the Fourier transform; however, control theory books almost invariably use the Laplace transform. The latter has some minor advantages. Physical systems, for example, usually start at a given time. Laplace-transform methods can handle such initial-value problems, while Fourier transforms are better suited for steady-state situations where any information from initial conditions has decayed away. In practice, one usually sets the initial conditions to zero (and assumes, implicitly, that they have decayed away), which effectively eliminates any distinction. Inverse Laplace transforms, on the other hand, lack the symmetry that inverse Fourier transforms have with respect to the forward transform. But in practice, one almost never needs to carry out an explicit inversion. Finally, one often needs to transform functions that do not decay to 0 at infinity, such as the step function $\theta(x)$, (= 0, $x < 0$ and = 1, $x > 0$). The Laplace transform is straightforward, but the Fourier transform must be defined by multiplying by a decaying function and taking the limit of infinitely slow decay after transforming. Because of the decaying exponential in its integral, one can define the Laplace transform of a system output, even when the system is unstable. Whichever transform one chooses, one needs to consider complex values of the transform variable. This also gives a slight edge to the Laplace transform, since one does not have to remember which sign of complex frequency corresponds to a decay and which to growth. In the end, we follow the engineers (except that we use $i = \sqrt{-1}$!), defining the Laplace transform of $y(t)$ to be

$$\mathcal{L}[y(t)] \equiv y(s) = \int_0^\infty y(t)e^{-st}dt \ . \tag{2.8}$$

Then, for zero initial conditions, $\mathcal{L}[\frac{d^n y}{dt^n}] = s^n y(s)$ and $\mathcal{L}[\int y(t)dt] = \frac{1}{s}y(s)$. Note that we use the same symbol y for the time and transform domains, which are quite different functions of the arguments t and s. The abuse of notation makes it easier to keep track of variables.

An nth-order linear differential equation then transforms to an nth-order algebraic equation in s. For example, the first-order system

$$\dot{y}(t) = -\omega_0 y(t) + \omega_0 u(t) \tag{2.9}$$

becomes

$$sy(s) = \omega_0\left[-y(s) + u(s)\right] \ , \tag{2.10}$$

leading to

$$G(s) \equiv \frac{y(s)}{u(s)} = \frac{1}{1 + s/\omega_0} \ , \tag{2.11}$$

where the transfer function $G(s)$ is the ratio of output to input, in the transform space. Here, $\omega_0 = 2\pi f_0$ is the characteristic angular frequency of the low-pass filter. The frequency dependence implicit in Eq. (2.11) is made explicit by simply evaluating $G(s)$ at $s = i\omega$:

$$G(i\omega) = \frac{1}{1 + i\omega/\omega_0} \ , \tag{2.12}$$

with

$$|G(i\omega)| = \frac{1}{\sqrt{1 + \omega^2/\omega_0^2}} \tag{2.13}$$

[3] Of course, it is *possible* to measure both position and velocity, using, for example, an inductive pickup coil for the latter; however, it is rare in practice to measure all the state variables $\vec{x}$.

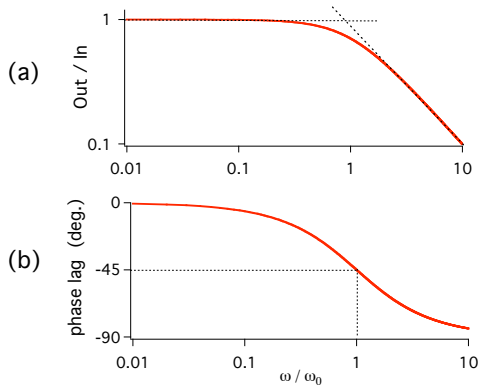

FIG. 3 (Color in online edition) Transfer function for a first-order, low-pass filter. (a) Bode magnitude plot ($|G(i\omega)|$); (b) Bode phase plot [arg$G(i\omega)$].

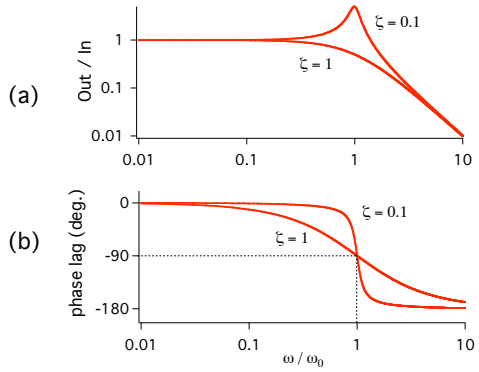

FIG. 4 (Color in online edition) Transfer function for a second-order system. $\zeta = 0.1$ gives underdamped and $\zeta = 1$ critically damped dynamics. (a) Bode magnitude plot; (b) Bode phase plot.

and

$$\arg G(i\omega) = -\tan^{-1}\frac{\omega}{\omega_0}\ . \tag{2.14}$$

Note that many physics texts use the notation $\tau = 1/\omega_0$, so that the Fourier transform of Eq. (2.12) takes the form $1/(1+i\omega\tau)$.

In control-theory books, log-log graphs of $|G(i\omega)|$ and linear-log graphs of $\arg(G(i\omega))$ are known as Bode plots, and we shall see that they are very useful for understanding qualitative system dynamics. [In the physics literature, $\chi(\omega) \equiv G(i\omega)$ is known as the dynamical linear response function (Chaikin and Lubensky, 1995).] In Fig. 3, we show the Bode plots corresponding to Eqs. (2.13) and (2.14). Note that the asymptotes in Fig 3(a) intercept at the cutoff frequency ω_0 and that in (b), the phase lag is $-90°$ asymptotically, crossing $-45°$ at ω_0. Note, too, that we break partly from engineering notation by using amplitude ratios in Fig. 3a rather than decibels ($dB = 20\log_{10}|G|$).

The transfer function $G(s)$ goes to infinity when $s = -\omega_0$. Such a point in the complex s-plane is called a pole. Here, we see that poles on the negative real s-axis correspond to exponential decay of impulses, with the decay rate fixed by the pole position. The closer the pole is to the imaginary s-axis, the slower the decay. Poles in the right-hand side correspond to exponentially increasing amplitudes.

Similarly, using the Laplace (Fourier) transfrom, the transfer function for the second-order system is

$$G(s) = \frac{1}{1+2\zeta s+s^2}\ \text{, or } G(i\omega) = \frac{1}{1+2i\zeta\omega-\omega^2}\ , \tag{2.15}$$

and Bode plots for various damping ratios ζ are shown in Fig. 4. (Recall that we have scaled $\omega_0 = 1$.) Here, there are two poles. In the underdamped case ($\zeta < 1$), they form a complex-conjugate pair $s = -\zeta \pm i\sqrt{1-\zeta^2}$. In the overdamped case ($\zeta > 1$), the two poles are both on the real s-axis.

Both the low-pass filter and the second-order system have transfer functions $G(s)$ that are rational functions; i.e., they can be written as $M(s)/N(s)$, where $M(s)$ and $N(s)$ are polynomials. Not every system can be written in this form. For example, consider a sensor that faithfully records a signal, but with a time delay Δt, i.e., $v(t) = y(t-\Delta t)$. From the shift theorem for Laplace transforms, the transfer function for such a sensor is $G(s) = e^{-s\Delta t}$, which is equivalent to an infinite-order system. Note that the magnitude of G is always 1 and that the phase increases linearly with frequency. In contrast, in the earlier two examples, the phase tended to an asymptotic value.

The convolution theorem allows important manipulations of transfer functions. If we define $G * H \equiv \int_0^\infty G(\tau)H(t-\tau)d\tau$, then $\mathcal{L}[G*H] = G(s)H(s)$. Convolution is just the tool one needs to describe compound systems where the output of one element is fed into the input of the next element. Consider, for example, a first-order sensor element that reports the position of a second-order mechanical system. We would have

$$\ddot{y} + 2\zeta\dot{y} + y = u(t)\ , \tag{2.16}$$

and

$$\dot{v} + v = y(t)\ , \tag{2.17}$$

where $u(t)$ drives the oscillator position $y(t)$, which then drives the measured output $v(t)$. Laplace transforming,

$$\begin{aligned} y(s) &= G(s)u(s) = \frac{1}{1+2\zeta s+s^2}\cdot u(s) \\ v(s) &= H(s)y(s) = \frac{1}{1+s}\cdot y(s)\ , \end{aligned} \tag{2.18}$$

and thus $v(s) = H(s)G(s)u(s)$, implying an overall loop transfer function $F(s) = H(s)G(s)$. Having two elements in series leads to a transfer function that is the product of the transfer functions of the individual elements. In the time domain, the series output would be the convolution of the two elements.

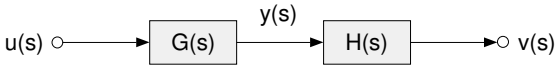

FIG. 5 Block diagram illustrating signal flow from the input u through the system dynamics G to the output y through the sensor H .

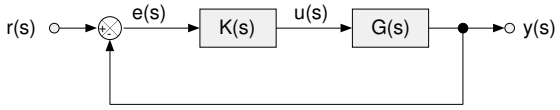

FIG. 6 Block diagram illustrating closed-loop control of a system $G(s)$. Controller dynamics are given by $K(s)$.

The above example motivates the introduction of block diagrams to represent the flow of signals for linear systems. We depict the above as shown in Fig. 5.

III. FEEDBACK: AN ELEMENTARY INTRODUCTION

Having reviewed some basic ideas from dynamical systems and having established some notation, we can understand simple feedback ideas. Here, and elsewhere below, it is understood that the reader desiring more details should consult one of the control-theory references cited above (Dutton *et al.*, 1997; Franklin *et al.*, 2002, 1998) or any of the many other similar works that exist. A good book on "control lore" is Leigh (2004), which gives a detailed, qualitative overview of the field, with a nice annotated bibliography.

A. Basic ideas: Frequency domain

Consider a system whose dynamics are described by $G(s)$. The goal is to have the system's output $y(t)$ follow a control signal $r(t)$ as faithfully as possible. The general strategy consists of two parts: First, we measure the actual output $y(t)$ and determine the difference between it and the desired control signal $r(t)$, i.e., we define $e(t) = r(t) - y(t)$, which is known as the error signal. Then we apply some "control law" K to the error signal to try to minimize its magnitude (or square magnitude).

In terms of block diagrams, we make the connections shown in Fig. 6, where a control law $K(s)$ has been added. Manipulating the block diagrams, we have

$$y(s) = K(s)G(s)e(s) \tag{3.1}$$

$$y(s) = \frac{K(s)G(s)}{1+K(s)G(s)}r(s) = \frac{L(s)}{1+L(s)}r(s)\ , \tag{3.2}$$

where the *loop gain* $L(s) \equiv K(s)G(s)$. Starting from the system dynamics $G(s)$, we have modified the "open-loop" dynamics to be $L(s) = K(s) \cdot G(s)$, and then we have transformed the dynamics a second time by "closing the loop," which leads to closed-loop dynamics given by $T \equiv KG/(1+KG) = L/(1+L)$. One hopes, of course, that one can choose $K(s)$ so that $T(s)$ has "better" dynamics than $G(s)$. (For reasons to be discussed below, T is known as the "complementary sensitivity function.")

Note the negative sign in the feedback node in Fig. 6, which implies that the signal $e(t) \equiv r(t) - y(t)$ is fed back into the controller. Such "negative feedback" gives a signal that is positive when the output y is below the set-point r and negative when above. A direct proportional control law, $u = K_p e$ with $K_p > 0$, then tends to counteract a disturbance. Absent any delays, this is a stabilizing effect that can also reduce a system's sensitivity to perturbations. "Positive feedback" reverses the sign in the feedback node, so that $r + y$ is fed into the controller.[4] It can be used to make a switch between two states, and hence a digital memory. (See Sec. VII.B.2.) It can also be used to make an oscillator.[5] More generally, though, there is no reason that a controller cannot be an arbitrary function $u(t) = f[r(t), y(t)]$. Such a controller has "two degrees of freedom" and, obviously, can have more complicated behavior than is possible with only negative or positive feedback. (As an example, one could have negative feedback in some frequency range and positive feedback at other frequencies.)

As a simple example, consider the first-order, low-pass filter described above, with

$$G(s) = \frac{G_0}{1+s/\omega_0}\ . \tag{3.3}$$

Here, we have added a DC-gain G_0. We apply the simplest of control laws, $K(s) = K_p$, a constant. This is known as proportional feedback, since the feedback signal $u(t)$ is proportional to the error signal, $u(t) = K_p e(t)$. Then

$$T(s) = \frac{K_p G_0}{K_p G_0 + 1 + s/\omega_0} = \frac{K_p G_0}{K_p G_0 + 1} \cdot \frac{1}{1 + \frac{s}{\omega_0(1+K_p G_0)}}\ . \tag{3.4}$$

This is again just a low-pass filter, with modified DC gain and cutoff frequency. The new DC gain is deduced by taking the $s \to 0$ ($\omega \to 0$) limit in $T(s)$ and gives $K_p G_0/(K_p G_0 + 1)$. The new cutoff frequency is $\omega' = \omega_0(1 + K_p G_0)$. This can also be seen in the time domain,

---

[4] Physicists often informally use "positive feedback" to denote the situation where $-e = y - r$ is fed back into the controller. If the control law is $u = K_p(-e)$, then the feedback will tend to drive the system away from an equilibrium fixed point. Using the terminology defined above, however, one would better describe this as a situation with negative feedback *and* negative gain.

[5] Negative feedback can also lead to oscillation, although typically more gain – and hence more control energy – is required than if positive feedback is used.

where

$$\dot{y} = -\omega_0 y + \omega_0 K_p G_0 (r - y) \quad (3.5)$$

$$\text{or} \quad \dot{y} = -\omega_0 (1 + K_p) y(t) + K_p G_0 r(t) \; . \quad (3.6)$$

In effect, the driving signal is $u(t) = \omega_0 K_p G_0 (r - y)$. If the control signal $r(t) = r_\infty = \text{const.}$, then the output settles to $y_\infty = \frac{K_p G_0}{(K_p G_0 + 1)} \, r_\infty$.

If we had instead the open-loop dynamics

$$\dot{y}(t) = -\omega_0 y(t) + \omega_0 K_p G_0 r(t) \; , \quad (3.7)$$

we would have the "bare" cutoff frequency ω_0 and a final value $y_\infty \equiv y(t \to \infty) = K_p G_0 r_\infty$.

It is interesting to compare the open- and closed-loop systems. The closed-loop system is faster for $K_p > 1$. Thus, if the system is a sensor, sluggish dynamics can be speeded up, which might be an advantage for a measuring instrument. Furthermore, the steady-state solution,

$$y_\infty = \frac{K_p G_0}{(K_p G_0 + 1)} \, r_\infty \; , \quad (3.8)$$

tracks r_∞ closely if $K_p \gg 1/G_0$. One might counter that in the open-loop system, one could set $K_p = 1/G_0$ and have $y_\infty = r_\infty$ exactly. However, if G_0 varies over time — for example, amplifier gains often drift greatly with temperature — this tracking will not be good. By contrast, for $K_p \gg 1/G_0$, the closed-loop system tracks r_∞ without much sensitivity to the DC-gain G_0.

We can sharpen our appreciation of this second advantage by introducing the notion of the parametric sensitivity S of a quantity Q with respect to variations of a parameter P. One defines $S \equiv \frac{P}{Q} \frac{dQ}{dP}$. We can then compare the open- and closed-loop sensitivities of the transfer function with respect to the DC gain, G_0:

$$S_{open} = \frac{G_0}{K_p G_0} \frac{d}{dG_0} (K_p G_0) = 1 \quad (3.9)$$

$$S_{closed} = \frac{1}{1 + K_p G_0} \ll 1 \; . \quad (3.10)$$

Thus, if $K_p G_0 \gg 1$, the closed-loop dynamics are much less sensitive to variations in the static gain G_0 than are the open-loop dynamics.

We next consider the effects of an output disturbance $d(t)$ and sensor noise $\xi(t)$. From the block diagram in Fig. 7 (ignoring the $F(s)$ block for the moment),

$$y = KGe + d \quad (3.11)$$

$$e = r - \xi - y \; , \quad (3.12)$$

which implies

$$y(s) = \frac{KG}{1 + KG} \left[r(s) - \xi(s) \right] + \frac{1}{1 + KG} d(s) \; . \quad (3.13)$$

In the previous example, disturbances $d(t)$ would have been rejected up to the cutoff frequency $\omega' = \omega_0 (1 + K_p)$.

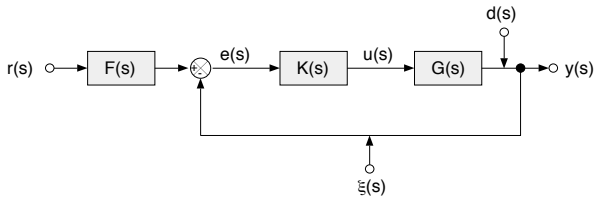

FIG. 7 Block diagram illustrating closed-loop control of a system $G(s)$ subject to disturbances $d(s)$ and measurement noise $\xi(s)$. The control signal $r(s)$ is assumed to be noise free. If present, the block $F(s)$ adds feedforward dynamics.

This is usually desirable. On the other hand, the control signal effectively becomes $r - \xi$: the system has no way to distinguish the control signal $r(t)$ from measurement noise $\xi(t)$. Thus, the higher the gain is, the noisier the output. (Putting a "prefilter" between the reference and the feedback node a is a partial solution. The controller then has two degrees of freedom, as described above.)

The frequency ω' is known as the feedback bandwidth. We can thus state that a high feedback bandwidth is good, in that it allows a system to track rapidly varying control, and bad, in that it also allows noise to feed through and contaminate the output. This tradeoff may also be expressed by rewriting Eq. (3.13) as

$$e_0(s) \equiv r(s) - y(s) = S(s) \left[r(s) - d(s) \right] + T(s) \xi(s) \; , \quad (3.14)$$

where e_0 is the tracking error, i.e., the difference between the desired tracking signal r and the actual output y. This is distinguished from $e = r - y - \xi$, the difference between the tracking signal and the measured output. In Eq. (3.14), $S(s)$ is the sensitivity function discussed above and $T(s) = KG/(1 + KG) = 1 - S$ is the complementary sensitivity function. The general performance goal is to have e_0 be small given various "inputs" to the system. Here, we regard the tracking signal r, the disturbance d and the sensor noise ξ as inputs of differing kinds. A fundamental obstacle is that $S + T = 1$ at all frequencies; thus, if S is small and disturbances are rejected, then T is large and sensor noise feeds through, and *vice versa*. We discuss possible ways around this tradeoff, below.

B. Basic ideas: Feedforward

Another basic idea of control theory is the notion of *feedforward*, which is a useful complement to feedback. Let's say that one wants a step change in the reference function. For example, in a loop controlling the temperature of a room, one can suddenly change the desired setpoint from 20 to 25 °C. The feedback loop may work satisfactorily in response to room perturbations, but if one knows ahead of time that one is making a sudden

change, one can do better than to just let the feedback loop respond. The usual way is to try to apply a prefilter $F(s)$ to the control signal $r(s)$. In the absence of any feedback, the system response is just $y(s) = G(s)F(s)r(s)$. If we can choose $F = G^{-1}$, then y will just follow r exactly. Because the actuator has finite power and bandwidth, one usually cannot simply invert G. Still, if one can apply an approximate inverse to r, the dynamical response will often be significantly improved. Similarly, if one has an independent way of measuring disturbances (for example, if one has a thermometer outside a building, it can actually anticipate disturbances to an interior room), one can apply a compensator element that feeds forward the filtered disturbance to the actuator. Again, the idea is to try to create a signal that will *a priori* undo the effect of the disturbance.

Because it is usually impossible to implement perfect feedforward and because disturbances are usually unknown, one generally combines feedforward with feedback. In Fig. 7, one includes the prefilter $F(s)$ after the time varying setpoint, $r(s)$, before the loop node. The output $y(s)$ [Eq. (3.13)] then becomes

$$y(s) = \frac{KG}{1+KG}\left[F(s)r(s) - \xi(s)\right] + \frac{1}{1+KG}d(s)\ . \quad (3.15)$$

Choosing F as close as possible to $1 + (KG)^{-1}$ reduces the "load" on the feedback while simultaneously rejecting disturbances via the feedback loop.

A more subtle application of these ideas to scanning probe microscopy illustrates the strengths and weaknesses of using feedforward. Scanning probe microscopes (SPM) – including, most notably the scanning tunneling microscope (STM) and atomic force microscope (AFM) – have revolutionized surface science by achieving atomic- or near-atomic-resolution images of surfaces in a variety of circumstances. Practical microscopes usually contain a feedback loop that keeps constant the distance between the sample and the SPM probe, as that probe is rastered over the sample surface (or *vice versa*). In a recent paper, Schitter *et al.* (2004) use the fact that in many SPM images, one scan line resembles its neighbor. They record the topography estimate of the previous line and calculate the actuator signal needed to reproduce that topography. This is then added to the actuator signal of the *next* scan line. The feedback loop then has to deal with only the differences between the expected topography and the actual topography. The limitations of this are (i) that the actuator may not be able to produce a strong or rapid enough signal to deal with a sharp topography change (e.g., a step) and (ii) the new scan line may in fact be rather different from the previous line, in which case the feedforward will actually worsen the performance. In practice, one puts a "gain" on the feedforward term that reflects the correlation one expects between corresponding vertical pixels. Unity gain – unity correlation – implies that one expects the new line to be exactly the same as the old line, justifying the full use of feedforward. If there is no correlation – i.e., if statistically, knowing the height of the pixel in a given column of the previous scan tells you nothing about the expected height in the next scan – then one should not use feedforward at all. The actual gain that should be applied then should reflect the expected correlation from line to line. Note the relationship here between information and feedback/feedforward, in that the appropriate amount of feedback and feedforward is determined by the amount of information one scan line gives regarding the next. See Sec. IX, below.

Finally, we can now better appreciate the distinction between "control" and "feedback." The former refers to the general problem of how to make a system behave as desired. The latter is one technique for doing so. What we have seen so far is that feedback, or "closed-loop control," is useful for dealing with uncertainty, in particular by reducing the effects of unknown disturbances. On the other hand, feedforward, or "open-loop control," is useful for making desired (i.e., known) changes. As Eq. (3.15) shows, one usually will want to combine both control techniques.

C. Basic ideas: Time domain

Until now, we have focused our discussion of feedback on the frequency domain, but it is sometimes preferable to use the time domain, working directly with Eq. (2.2). State-space approaches are particularly useful for cases where one has multiple inputs and multiple outputs (MIMO), although much of our discussion will be for the single-input-single-output (SISO) case. To further simplify issues, we will consider only the special problem of controlling about a state $\vec{x} = 0$. It is easy to generalize this (Franklin *et al.*, 2002).

In the SISO case, we have only one output $y(t)$ and one input $u(t)$. We then have

$$\begin{aligned} \dot{\vec{x}} &= \widetilde{A}\vec{x} + \vec{b}u\ ; \qquad u = -\vec{k}^T\vec{x}\ , \\ y &= \vec{c}^T\vec{x}\ . \end{aligned} \quad (3.16)$$

In Eq. (3.16), row vectors are represented, for example, as $\vec{c}^T = (c_1\ c_2)$. The problem is then one of choosing the feedback vector $\vec{k}^T = (k_1\ k_2)$ so that the eigenvalues of the new dynamical system, $\dot{\vec{x}} = \widetilde{A'}\vec{x}$ with $\widetilde{A'} = \widetilde{A} - \vec{b}\vec{k}^T$, have the desired properties.

One can easily go from the state-space representation of a dynamical system to its transfer function. Laplace transforming Eq. (3.16), one finds

$$G(s) = \vec{c}^T(s\widetilde{I} - \widetilde{A})^{-1}\vec{b}\ , \quad (3.17)$$

where $\widetilde{I}$ is the identity matrix. One can also show that if one changes coordinates for the state vector (i.e., if one defines $\vec{x}' = \widetilde{T}\vec{x}$), the transfer function deduced in Eq. (3.17) will not change. The state vector $\vec{x}$ is an internal representation of the dynamics, while $G(s)$ represents the physical input-output relation.

Because the internal state $\vec{x}$ usually has more elements than either the number of inputs or outputs (here, just one each), a few subtleties arise. To understand these, let us rewrite Eqs. (3.16) for the special case where $\widetilde{A}$ is a 2×2 diagonal matrix, with diagonal entries λ_1 and λ_2. Then

$$\begin{aligned} \dot{x}_1 &= \lambda_1 x_1 + b_1 u \\ \dot{x}_2 &= \lambda_2 x_2 + b_2 u \\ y &= c_1 x_1 + c_2 x_2 \ . \end{aligned} \tag{3.18}$$

If $b_1 = 0$, then clearly, there is no way that $u(t)$ can influence the state $x_1(t)$. More generally, if any element of $\vec{b}$ is zero, the system will not be "controllable." Likewise, if any element of $\vec{c}$ is zero, then $y(t)$ will not be influenced at all by the corresponding element of $\vec{x}(t)$, and we say that that state is not "observable." More formally, a system is controllable if it is always possible to choose a control sequence $u(t)$ that takes the system from an arbitrary initial state to an arbitrary final state in finite time. One can show that a formal test of controllability is given by examining a matrix made out of the vectors $\widetilde{A}^i\vec{b}$, for $i = 0 \ldots n-1$. If the "controllability matrix" $\widetilde{U}_b$ is invertible, the system is controllable. The $n \times n$ matrix $\widetilde{U}_b$ is explicitly

$$\widetilde{U}_b \equiv \left(\vec{b} \quad \widetilde{A}\vec{b} \quad \widetilde{A}^2\vec{b} \quad \ldots \quad \widetilde{A}^{n-1}\vec{b} \right) \ . \tag{3.19}$$

In the more general MIMO case, $\vec{b}$ will be a matrix, $\widetilde{B}$, and the controllability condition is that $\widetilde{U}_B$ have full rank. One can also show that a similar test exists for observability and, indeed, that controllability and observability are dual properties (corresponding to how one gets information into and out of the dynamical system) and that any property pertaining to one has a counterpart (Lewis, 1992). Note that we use "observable" here in its classical sense. Sec. VIII contains a brief discussion of quantum control.

One can show that if the technical conditions of controllability and observability are met, then one can choose a $\vec{k}$ that will place the eigenvalues anywhere.[6] The catch is that the farther one moves an eigenvalue, the larger the elements of $\vec{k}$ must be, which will quickly lead to unattainable values for the input u. Thus, one should move only the eigenvalues that are "bad" for system response and move them as little as possible (Franklin *et al.*, 2002).

As a quick example (Franklin *et al.*, 2002), consider an undamped pendulum of unit frequency [Eq. (2.6), with $\zeta = 0$]. We desire to move the eigenvalues from $\pm i$ to -2 and -2. In other words, we want to double the natural frequency and change the damping ζ from 0 to 1 (critical damping). Let $u = -(k_1 x_1 + k_2 x_2)$ in Eq. (2.6), which leads to a new dynamical matrix

$$\widetilde{A}' = \begin{pmatrix} 0 & 1 \\ -1 - k_1 & -k_2 \end{pmatrix} \ . \tag{3.20}$$

Computing the characteristic equation for $\widetilde{A}'$ and matching coefficients with the desired equation $[(\lambda + 2)^2 = 0]$, one easily finds $\vec{k}^T = (3\ 4)$. More systematically, there are general algorithms for choosing $\vec{k}$ so that $\widetilde{A}'$ has desired eigenvalues, such as "Ackermann's method" for pole (eigenvalue) placement [(Dutton *et al.*, 1997), section 5.4.7].

In the above discussion of a SISO system, the control law used the full state vector $\vec{x}(t)$ but there was only a single observable $y(t)$. This is a typical situation. One thus needs some way to go from the single dynamical variable y to the full state $\vec{x}$. But since y is a linear function of all the x's and since one knows the system dynamics, one can, in fact, do such an inversion. In the control-theory literature, the basic strategy is to make an "observer" that takes the partial information as an input and tends to the true state.[7] Since the observed system is being modeled by computer, one has access to its internal state, which one can use to estimate the true dynamical state. In the absence of measurement noise, "observers" are straightforward to implement (Dutton *et al.*, 1997; Özbay, 2000). The basic idea is to update an existing estimate $\hat{x}(t)$ of the state using the dynamical equations [Eqs. (2.2)] with the observed output $y(t)$. The augmented dynamics are

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + F\left[y(t) - C\hat{x}(t)\right] \ . \tag{3.21}$$

For simplicity, we have dropped tildes (A, B, etc.) and vector symbols (x, y, etc.). The new matrix F is chosen so that the error in the estimate of $x(t)$ converges to zero:

$$\dot{e}(t) = \dot{x}(t) - \dot{\hat{x}}(t) = (A - FC)\, e(t) \ . \tag{3.22}$$

To converge quickly, the real parts of the eigenvalues of $A - FC$ should be large and negative; on the other hand, large eigenvalues will lead to noisy estimates. A reasonable compromise is to choose F so that the estimator converges several times faster than the fastest pole in the system dynamics. The estimator described here is a "full-order" estimator, in that it estimates all the states x whether they are observed or not. Intuitively, it is

[6] This is true even if the system is unobservable, but then the internal state $\vec{x}$ cannot be deduced from the observed output y.

[7] There are actually two distinct control strategies. In the "state-vector-feedback" (SVF) approach, one uses a feedback law of the form $\vec{u} = -\widetilde{K}\vec{x}$ along with an observer to estimate $\vec{x}$ from the measurements $\vec{y}$. In the "output-feedback" approach, one uses directly $\vec{u} = -\widetilde{K}'\vec{y}$. The SVF approach is conceptually cleaner and more systematic, and it allows one to prove general results on feedback performance. The output-feedback approach is more direct – when it works (Lewis, 1992).

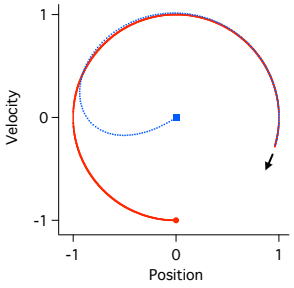

FIG. 8 (Color in online edition) Phase-space plot of harmonic oscillator and its observer. The harmonic oscillator trajectory starts from the dot and traces a circle. The observer starts from the square with incorrect initial velocity but converges to track the oscillator. For simplicity, the control signal $u = 0$.

clear that if one defines each observable $\vec{y}$ to be one of the elements of $\vec{x}$, then it should be possible to define a "partial-order" estimator, which uses the observations where they exist and estimates only the remaining unknown elements of $\vec{x}$. This is more efficient to calculate but more complicated to set up (Dutton *et al.*, 1997). Finally, if there is significant measurement noise, one generally uses a rather different strategy, the Kalman filter, to make the observer. See Sec. V.D.

To continue the example of the harmonic oscillator, above, choosing $\vec{F}^T = (4\ 3)$ gives $\widetilde{A}'$ repeated eigenvalues at -2. The phase-space plot (x_1 vs. x_0) in Fig. 8 shows how the observer tracks the physical system. The oscillator, starting from $x_0 = 0$, $x_1 = -1$ traces a circle of unit radius in phase space. The observer dynamical system starts from different initial conditions but eventually converges to track the physical system. Here, the observer time scale is only half the natural period; in a real application, it should be faster. After the observer has converged, one simply uses its values ($\hat{\vec{x}}$), fed by the observations y, in the feedback law. Obviously, in a real application, the computer simulation of the observer must run faster than the dynamics of the physical system.

The above methods are satisfactory for linear equations. In the physics literature, one often uses the "method of time delays" (Strogatz, 1994). Here, an n-element state vector is made by taking the vector $\vec{y}_\tau \equiv [(y(t),\ y(t-\tau),\ \ldots y(t-(n-1)\tau)]$, where the delay τ is chosen to be the "dominant" time constant (the choice often requires a bit of playing around to optimize). Measuring $\vec{y}_\tau$ is roughly equivalent to measuring the first $n-1$ time derivatives of $y(t)$. For example, from $y(t)$ and $y(t-\tau)$, one has information about $\dot{y}(t)$. One can show that $\vec{y}_\tau$ obeys dynamics like that of the true system (more precisely, its phase space is topologically similar). Note that one usually must choose the "embedding dimension" n. If n is too small, trajectories will cross in the n dimensional phase space, while if n is too large, one will merely be embedding the system in a space that has a larger dimension than that of the actual state vectors. Measurement noise effects are magnified when n is chosen too large. The method of time delays works even if one does not know the correct system. If the system is linear, then there is little excuse for not figuring it out (see Sect. V.A), but if the system is nonlinear, then it may not be easy to derive a good model and the method of time delays may be the best one can do.

The direct, "state-space" formulation of feedback with its "pole-placement" methods is easier for simulating on a computer and has other mathematical advantages, while the frequency domain is often more intuitive. One should know both approaches.

To summarize (and anticipate, slightly), the possible advantages of adding a feedback loop include

- Altered dynamics (e.g., faster or slower time constants);
- The ability to linearize a nonlinear sensor by holding the sensor value constant (discussed below);
- Reduced sensitivity to environmental disturbances.

Possible *disadvantages* include

- Sensor noise may contaminate the output;
- A stable system may be driven unstable.

We shall deal with the potential disadvantages below. We now give an example that highlights the use of feedback to linearize a nonlinear sensor signal.

D. Two case studies

1. Michelson interferometer

In Section III, we looked at controlling a device that was merely a low-pass filter. Such an example might seem academic in that one rarely encounters such a simple system in the real world. Yet, as the following case study shows, simplicity can often be forced on a system, purchased at the price of possible performance.

We consider the Michelson interferometer designed by Gray *et al.* for use in the next-generation LIGO gravity-wave project (Gray *et al.*, 1999). The gravity-wave detector (itself an interferometer) must have all its elements (which are several kilometers long!) isolated from earth-induced vibrations, so that any gravity-wave-induced distortions may be detected. In order to isolate the large masses, one can measure their position relative to the earth — hence the need for accurate displacement measurement. Of course, the interferometer may be used in many other applications, too. We refer the reader to the article of Gray *et al.* for details about the project and interferometer. Here, we consider a simplified version that highlights their use of feedback.

Fig. 9 shows a schematic diagram of the interferometer. Without the control element $K(s)$ and the piezo actuator shown at bottom, it would depict just an ordinary Michelson interferometer. As such, its output is a sinusoidal function of the displacement of the target mirror. In open-loop operation, the interferometer could be used as a linear sensor only over a small fraction of a wavelength. By adding the actuator, Gray *et al.* force

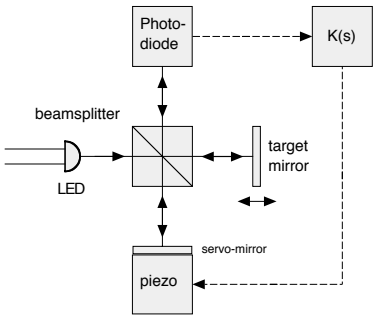

FIG. 9 Michelson interferometer with feedback element $K(s)$ added to linearize output for large displacements.

the servo-mirror to track the moving target mirror. The actuator signal to the servo-mirror effectively becomes the sensor signal.

One immediate advantage of tracking a desired "set-point" on the fringe is that if the actuator is linear, one will have effectively linearized the original, highly nonlinear sensor signal. (In fact, the actuator used piezoelectric ceramic stacks for displacement, which have their own nonlinearities. But these nonlinearities are much smaller than those of the original output.) Another widely used application of feedback to linearize a signal, mentioned briefly above in our discussion of feedforward techniques is the scanning tunneling microscope (STM), where the exponential dependence of tunneling current on the distance between conductors is linearized by feedback (Oliva *et al.*, 1995).

In their published design, Gray *et al.* chose the feedback law to be a band-limited proportional gain:

$$K(s) = \frac{K_0}{1 + s/\omega_0} \ . \tag{3.23}$$

Their controller $K(s)$ looks just like our simple system $K(s)G(s)$ in Eq. (3.2) above. They assume that their system has no dynamics ($G(s) = 1$), up to the feedback bandwidth $\omega' = \omega_0(1 + K_0)$. Of course, their system does have dynamics. For example, the photodiode signal rolls off at about 30 kHz, and the piezo actuator has a mechanical resonance at roughly the same frequency. But they chose $\omega_0 \approx 2\pi \times 25$ Hz and $K_0 \approx 120$, so that the feedback bandwidth of 3 kHz was much less than the natural frequencies of their dynamical system.

The design has much to recommend it. The large DC gain means that static displacements are measured accurately. One can also track displacements up to ω', which, if slower than the system dynamics, is much faster than their application requires. More sophisticated feedback design could achieve similar bandwidth even if the system dynamics were slower, but the added design costs would almost certainly outweigh any component-cost savings. And the performance is impressive: Gray *et al.* report a position noise of 2×10^{-14} m/$\sqrt{\text{Hz}}$, only about ten times more than the shot-noise limit imposed by the laser intensity used (≈ 10 mW at $\lambda = 880$ nm). The lesson is that it often pays to spend a bit more on high-performance components in order to simplify the feedback design. Here, one is "killing the problem with bandwidth;" i.e., one starts with far more bandwidth than is ultimately needed, in order to simplify the design. Of course, one does not always have that luxury, which motivates the study of more sophisticated feedback algorithms.

2. Operational amplifier

We briefly discuss another application of proportional feedback for first-order systems that most experimentalists will not be able to avoid, the operational amplifier ("op amp") (Mancini, 2002). The op amp is perhaps the most widely used analog device and is the basis of most modern analog electronic circuits. For example, it is used to make amplifiers, filters, differentiators, integrators, multipliers, to interface with sensors, and so forth. Almost any analog circuit will contain several of them.

An op amp is essentially a very high gain differential amplifier that uses negative feedback to trade off high gain for reduced sensitivity to drift. A typical circuit ("non-inverting amplifier") is shown in Fig. 10. The op amp is the triangular element, which is a differential amplifier of gain A:

$$V_{out} = A(V_{in} - V_-) \ . \tag{3.24}$$

The + and − inputs serve as an error signal for the feedback loop. The two resistors in the return loop form a voltage divider, with

$$V_- = V_{out}\frac{R_2}{R_1 + R_2} \equiv V_{out}\beta \ , \tag{3.25}$$

which leads to

$$G_{CL} \equiv \frac{V_{out}}{V_{in}} = \frac{A}{1 + A\beta} \approx \frac{1}{\beta} \ , \tag{3.26}$$

where G_{CL} is the closed-loop gain and the approximation holds when $A \gg 1$. Thus, the circuit is an amplifier of gain $G_{CL} \approx 1 + R_2/R_1$. The sensitivity to drift in A is

$$S = \frac{A}{G_{CL}}\frac{dG_{CL}}{dA} = \frac{1}{1 + A\beta} \ll 1 \ , \tag{3.27}$$

which again shows how a high open-loop gain $A \gg 1$ reduces the sensitivity of the closed-loop gain G_{CL} to drifts in the op amp. This, in fact, was the original technical motivation for introducing negative feedback, which occurred in the telephone industry in the 1920s and 30s.[8]

[8] See Mancini (2002). Historically, the introduction of negative feedback into electronic circuits was due to H. S. Black in 1927 (Black, 1934). While not the first use of negative feedback (steam-engine governors using negative feedback had been in use

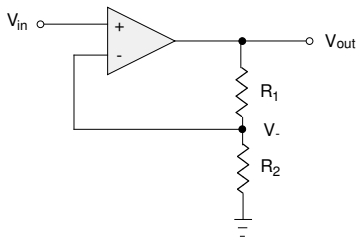

FIG. 10 Op-amp based non-inverting amplifier.

Open-loop amplifiers were prone to drift over time, leading to telephone connections that were too soft or too loud. By increasing the gain of the amplifiers and using negative feedback, one achieves much more stable performance. In effect, one is replacing the large temperature dependence of the semiconductors in transistors with the feebler temperature dependence of resistors (Mancini, 2002). Note that the $A \to \infty$ limit is one of the approximations made when introducing the "ideal" op amp that is usually first taught in electronics courses.

It is interesting to look at the frequency dependence of the op-amp circuit (Fig. 10). The amplifier is usually designed to act like a simple low-pass filter, with gain $A(\omega) = \frac{A_0}{1+i\omega/\omega_0}$. Following the same logic as in Sec. III.A, we find that the closed-loop equations correspond to a modified low-pass filter with cut-off frequency $\omega_{CL} = \beta\, A_0\, \omega_0$. One concludes that for $\omega_{CL} \gg \omega_0$,

$$G_{CL}\, \omega_{CL} \approx A_0\, \omega_0\ . \tag{3.28}$$

Thus, the closed-loop gain times the closed-loop bandwidth is a constant determined by the parameters of the amplifier. The product $A_0\ \omega_0$ is usually known as the unity-gain bandwidth, because it is the frequency at which the open-loop gain A of an op amp is 1. Modern op amps have unity-gain bandwidths that are typically between 10^6 and 10^9 Hz. Equation (3.28) states that in an op-amp-based voltage amplifier, there is a tradeoff between the gain of the circuit and its bandwidth. Mancini (2002) contains many examples of such "rough-and-ready" engineering calculations concerning feedback loops.

E. Integral control

All of the examples of feedback discussed above suffer from "proportional droop": i.e., the long-term response to a steady-state input differs from the desired setpoint. Thus, if the (static) control input to a low-pass filter is r_∞, the system settles to a solution $y_\infty = \frac{K_p}{1+K_p} r_\infty$. Only for an infinite gain will $y_\infty = r_\infty$, but in practice, the gain cannot be infinite. The difference between the desired signal r_∞ and the actual signal equals $\frac{1}{1+K_p} r_\infty$.

With integral control, one applies a control $K_i \int_{-\infty}^{t} e(t')dt'$ rather than (or in addition to) the proportional control term $K_p e(t)$. The integral will build up as long as $e(t) \neq 0$. In other words, the integral term eliminates the steady-state error. We can see this easily in the time domain, where

$$\dot{y}(t) = -\frac{1}{\tau} y(t) + \frac{K_i}{\tau^2} \int_{-\infty}^{t} [r_\infty - y(t')]dt'\ , \tag{3.29}$$

where K_i is rescaled to be dimensionless. Differentiating,

$$\ddot{y}(t) = -\frac{1}{\tau} \dot{y}(t) + \frac{K_i}{\tau^2} [r_\infty - y(t)]\ , \tag{3.30}$$

which clearly has a steady-state solution $y_\infty = r_\infty$ — no proportional droop!

It is equally interesting to examine the situation in frequency space, where the control law is $K(s) = K_i/\tau s$. One can interpret this K as a frequency-dependent gain, which is infinite at zero frequency and falls off as $1/\omega$ at finite frequencies. Because the DC gain is infinite, there is no steady-state error.

If the system transfer function is $G(s) = \frac{1}{1+\tau s}$, then the closed-loop transfer function becomes

$$T(s) = \frac{KG}{1+KG} = \frac{1}{1 + \frac{1}{K_i}\tau s + \frac{1}{K_i}\tau^2 s^2}\ . \tag{3.31}$$

Note that both Eqs. (3.30) and (3.31) imply that the integral control has turned a first-order system into effectively a second-order system, with $\omega_0^2 = \frac{K_i}{\tau^2}$ and $\zeta = \frac{1}{2\sqrt{K_i}}$. This observation implies a tradeoff: a large K_i gives good feedback bandwidth (large ω_0) but reduced ζ. Eventually (when $\zeta = 1$), even an overdamped system will become underdamped. In the latter case, perturbations relax with oscillations that overshoot the setpoint, which may be undesirable.

Integral control can be improved by combining with proportional control, $K(s) = \frac{K_i}{\tau s} + K_p$, which gives faster response while still eliminating steady-state errors. To see this, note that the closed-loop transfer function, Eq. (3.31) becomes

$$T(s) = \frac{KG}{1+KG} = \frac{1 + \left(\frac{K_p}{K_i}\right)\tau s}{1 + \frac{1+K_p}{K_i}\tau s + \frac{1}{K_i}\tau^2 s^2}\ , \tag{3.32}$$

which is 1 for $\omega \to 0$ and is asymptotically first-order, with time constant τ/K_p, for $\omega \to \infty$.

We have seen that a system with integral control always tends to the setpoint, whatever the value of K_i.

for over a century), Black's ideas spurred colleagues at the Bell Telephone Laboratories (Bell Labs) to analyze more deeply why feedback was effective. This led to influential studies by Nyquist (1932) and Bode (1945) that resulted in the "classical control," frequency-domain analysis discussed here. Around 1960, Pontryagin, Kalman, and others developed "modern" state-space methods based on time-domain analysis.

This sounds like a trivial statement, but it is our first example of how a feedback loop can lead to robust behavior that does not depend on details of the original system itself. In other words, it is only the loop itself and the fact that integration is employed that leads to the tracking properties of feedback. We return to this point in Section VII, which discusses biological applications of feedback.

IV. FEEDBACK AND STABILITY

Feedback can also be used to change the stability of a dynamical system. Usually, this is undesired. For example, as we shall see below, many systems will go into spontaneous oscillation when the proportional gain K_p is too high. Occasionally, feedback is used to stabilize an initially unstable system. A topical example is the Stealth Fighter.[9] The aircraft's body is made of flat surfaces assembled into an overall polygonal hull. The flat surfaces deflect radar beams away from their source but make the aircraft unstable. Using active control, one can stabilize the flight motion. A more prosaic example is the problem of balancing a broom upside down in the palm of your hand.

Before discussing stability in feedback systems, we briefly review some notions of stability in general in dynamical systems, in Section IV.A. In Section IV.B, we apply those ideas to feedback systems.

A. Stability in dynamical systems

Here, we review a few key ideas from stability theory (Strogatz, 1994). We return to the time domain and write a simple n'th-order equation in matrix form as

$$\dot{\vec{y}} = \widetilde{A}\vec{y}\,. \tag{4.1}$$

Assuming $\widetilde{A}$ to be diagonalizable, one can write the solution as

$$\vec{y}(t) = e^{\widetilde{A}t}\,\vec{y}(0) = \widetilde{R}\,e^{\widetilde{D}t}\,\widetilde{R}^{-1}\vec{y}(0)\;, \tag{4.2}$$

where $\widetilde{A} = \widetilde{R}\,e^{\widetilde{D}t}\,\widetilde{R}^{-1}$ and

$$\widetilde{D} = \begin{pmatrix} \lambda_1 & & & & \\ & \lambda_2 & & & \\ & & \cdot & & \\ & & & \cdot & \\ & & & & \lambda_n \end{pmatrix}\,. \tag{4.3}$$

[9] The instability of planes such as the F-117 fighter ("Nighthawk") is fairly obvious just looking at pictures of them. I have seen anecdotal mention of this but no serious discussion of how active control stabilizes its flight. See, for example, <http://www.danshistory.com/f117.shtml>.

The solution $\vec{y} = 0$ is stable against infinitesimally small perturbations if Re $\lambda_i < 0, \forall i$. Generally, the eigenvalues of the stability matrix $\widetilde{A}$ change when a control parameter — such as a feedback gain — is changed. If one of the eigenvalues has positive real part, the associated eigenmode will grow exponentially in time. (This is *linear* stability; a solution may be stable to infinitesimal perturbations but unstable to finite perturbations of large-enough amplitude. Such issues of nonlinear stability are relevant to the hysteretic systems discussed below.) The qualitative change in behavior that occurs as the eigenvalue crosses zero is known as a bifurcation. Once the system is unstable, the growth of the unstable mode means that nonlinear terms will quickly become important. The generic behavior of the bifurcation is then determined by the lowest-order nonlinear terms. (The general nonlinear term is assumed to have a Taylor expansion about the solution, $\vec{y} = 0$.) For example, the unstable mode (indexed by i) may behave either as

$$\dot{y}_i = \lambda_i y_i - a y_i^2 \quad \text{or} \quad \dot{y}_i = \lambda_i y_i - b y_i^3 \tag{4.4}$$

In Eq. (4.4), λ_i is a "control parameter," i.e., one that may be controlled by the experimenter. The parameters a and b are assumed to remain constant. The first relation in Eq. (4.4) describes a "transcritical bifurcation" and the second a "pitchfork bifurcation" (Strogatz, 1994). (If there is a symmetry $y \to -y$, one has a pitchfork bifurcation; otherwise, the transcritical bifurcation is the generic description of two solutions that exchange stabilities.) In both cases, the linearly unstable mode saturates with a finite amplitude, in a way determined by the lowest-order nonlinear term.

If the eigenvalue λ_i is complex and $\widetilde{A}$ real, then there will be a second eigenvalue $\lambda_j = \lambda_i^*$. The eigenvalues become unstable in pairs when Re $\lambda_i = 0$. The system then begins to oscillate with angular frequency Im λ_i. The instability is known as a Hopf bifurcation.

One situation that is seldom discussed in introductory feedback texts but is familiar to physicists studying nonlinear dynamics is the distinction between subcritical and supercritical bifurcations (analogous to first- and second-order phase transitions). Supercritical bifurcations refer to the case described above, where the system is stable until one of the λ's $= 0$, where the mode spontaneously begins to grow. But if the lowest nonlinear term is positive, it will reinforce the instability rather than saturating it. Then the lowest-order negative nonlinear term will saturate the instability. For example, a subcritical pitchfork bifurcation would be described by

$$\dot{y}_i = \lambda_i y_i + b y_i^3 - c y_i^5\;, \tag{4.5}$$

and a subcritical Hopf bifurcation by

$$\dot{y}_i = \lambda_i y_i + b|y_i|^2 y_i - c|y_i|^4 y_i\;. \tag{4.6}$$

In both these cases, the $y_i = 0$ solution will be metastable when Re λ_i is slightly positive. At some finite value of

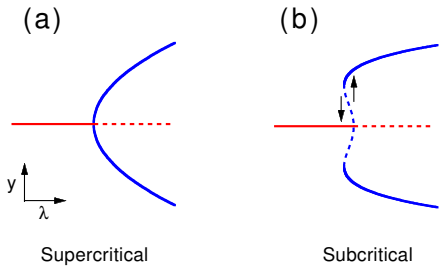

FIG. 11 (Color in online edition) Two scenarios for a pitchfork bifurcation, showing steady-state solutions y as a function of the control parameter λ. (a) supercritical; (b) subcritical, where the arrows show the maximum observable hysteresis.

Re λ_i, the system will jump to a finite-amplitude solution. On reducing Re λ_i, the eigenvalue will be slightly negative before the system spontaneously drops down to the $y_i = 0$ solution. For a Hopf bifurcation, the discussion is the same, except that y is now the amplitude of the spontaneous oscillation. Thus, subcritical instabilities are associated with hysteresis with respect to control-parameter variations. In the context of control theory, a system with a subcritical bifurcation might suddenly go unstable when the gain was increased beyond a critical value. One would then have to lower the gain a finite amount below this value to restabilize the system. Supercritical and subcritical pitchfork bifurcations are illustrated in Fig. 11.

B. Stability ideas in feedback systems

As mentioned before, in a closed-loop system, varying parameters such as K_p or K_i in the feedback law will continuously vary the system's eigenvalues, raising the possibility that the system will become unstable. As usual, one can also look at the situation in frequency space. If one has a closed-loop transfer function $T = \frac{KG}{1+KG}$, one can see that if KG ever equals -1, the response will be infinite. This situation occurs at the bifurcation points discussed above. (Exponential growth implies an infinite steady-state amplitude for the *linear* system.) In other words, in a feedback system, instability will occur when $|KG| = 1$ with a simultaneous phase lag of 180°.

The need for a phase lag of 180° implies that the open-loop dynamical system (combination of the original system and feedback compensation) must be at least 3rd-order to have an instability. Since the transfer function of an n'th-order system has terms in the denominator of order s^n, the frequency response will asymptotically be $(i\omega)^{-n}$, which implies a phase lag of $n\frac{\pi}{2}$. (This is just from the i^{-n} term.) In other words, as Fig. 3 and 4 show, a first-order system lags 90° at high frequencies, a second-order system lags 180°, etc. Thus, a second-order system will lag less than 180° at finite frequencies, implying that at least a third-order system is needed for instability.[10]

As an example, we consider the integral control of a (degenerate) second-order system, with $G(s) = \frac{1}{(1+\tau s)^2}$ and $K(s) = \frac{K_i}{s}$. Instability occurs when $K(i\omega)\times G(i\omega) = -1$. This leads to

$$\frac{i\omega}{K_i}(1-\tau^2\omega^2) - \frac{2\tau\omega^2}{K_i} + 1 = 0\ , \qquad (4.7)$$

Both real and imaginary parts of this complex equation must vanish. The imaginary part implies $\omega^* = 1/\tau$, while the real part implies $K_i^* = 2/\tau$; i.e., when K_i is increased to the critical value $2/\tau$, the system goes unstable and begins oscillating at $\omega = 1/\tau$.

Bode plots are useful tools in seeing whether a system will go unstable. Fig. 12 shows the Bode magnitude and phase plots for the second-order system described in the paragraph above, with $\tau = 1$. The magnitude response is plotted for two different values of the integral gain K_i. Note how changing a multiplicative gain simply shifts the response curve up on the log-log plot. In this simple case, the phase response is independent of K_i, which would not in general be true. For $K_i = 1$, we also show explicitly the gain margin, defined as the factor by which the gain must be increased to have instability, at the frequency where the phase lag is 180°. (We assume an open-loop-stable system.) Similarly, the phase margin is defined as the phase lag at the frequency where the magnitude response is unity. In Fig 12, the gain margin is about a factor of 2, and the phase margin is about 20°. A good rule of thumb is that one usually wants to limit the gain so that the gain margin is at least a factor of 2 and the phase margin at least 45°. For small gain margins, perturbations decay slowly. Similarly, a 90° phase margin would correspond to critically damped response to perturbations, and smaller phase margins give underdamped dynamics. The curve drawn for $K_i = 2$ shows that the system is at the threshold of instability (transfer-function response = 1, phase lag = 180°).

---

[10] Here, we are implicitly assuming the (common) scenario where instability arises dues to time delay (two poles with conjugate imaginary parts cross the imaginary s-axis together). Another case arises when a real pole crosses the imaginary s-axis from the left. Still other cases arise if the system is intrinsically unstable: then, increasing a feedback gain can actually stabilize the system. For example, the unstable inverted pendulum discussed below in Sec. IV.D can be stabilized by increasing the gains of proportional and derivative feedback terms (Sontag, 1998). These cases can all be diagnosed and classified using the Nyquist Criterion, which involves examining a polar plot of $[K(s)G(s)|_{s=i\omega}]$ for $0 < \omega < \infty$ in the complex plane (Dutton *et al.*, 1997). (See Sec. V.E for examples of such a polar plot.) In any case, it is usually clear which case is relevant in a given problem.

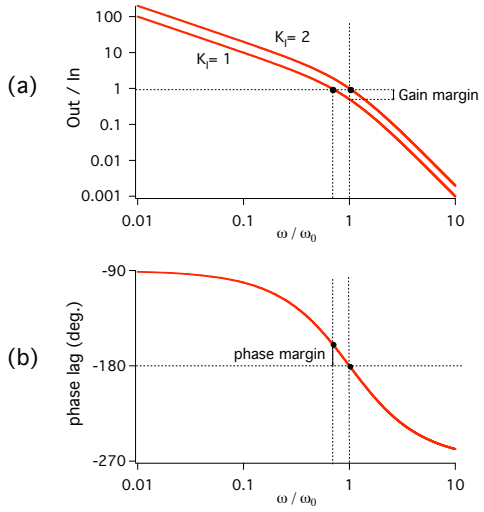

FIG. 12 (Color in online edition) Transfer function for a degenerate second-order system with integral control, showing gain and phase margins for $K_i = 1$. (a) Bode magnitude plot; (b) Bode phase plot.

C. Delays: Their generic origins and effect on stability

We see then that integral control, by raising the order of the effective dynamical system by one, will always tend to be destabilizing. In practice, the situation is even worse, in that one almost always finds that increasing the proportional gain also leads to instability. It is worth dwelling on this to understand why.

As we have seen, instability occurs when $K(s)G(s) = -1$. Because we evaluate KG at $s = i\omega$, we consider separately the conditions $|KG| = 1$ and phase lag $= 180°$. Since a proportional-gain law has $K = K_p$, increasing the gain will almost certainly eventually lead to the first condition's being satisfied.

The question is whether $G(s)$ ever develops a phase lag $> 180°$. Unfortunately, the answer in practice is that most physical systems do show lags that become important at higher frequencies.

To see one example, consider the effect of reading an experimental signal into a computer using an A/D converter. Most converters use a "zero-order hold" (ZOH). This means that they freeze the voltage at the beginning of a sampling time period, measure it over the sample

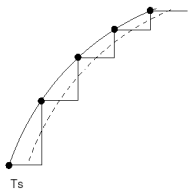

FIG. 13 Zero-order-hold digitization of an analog signal. Points are the measured values. Note that their average implies a sampling delay of half the sampling interval, T_s.

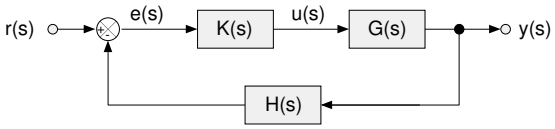

FIG. 14 Block diagram of feedback system with non-trivial sensor dynamics, given by $H(s)$.

period T_s, and then proceed to freeze the next value at the start of the next sampling interval. Fig. 13 illustrates that this leads to a delay in the digitized signal of $\Delta t = T_s/2$.

Even in a first-order system, the lag will make proportional control unstable if the gain is too large. The block diagram now has the form shown in Fig. 14, where, we have added a sensor with dynamics $H(s) = e^{-s\Delta t}$, corresponding to the digitization lag. From the block diagram, one finds

$$y(s) = \frac{KG}{1 + KGH}\, r(s)\ . \tag{4.8}$$

An instability occurs when $KGH = -1$, i.e. when

$$K_p \cdot \frac{1}{1 + s\tau} \cdot e^{-s\Delta t} = -1\ . \tag{4.9}$$

Since $|H| = 1$, $|KGH|$ has the same form as the undelayed system, $\frac{K_p}{\sqrt{1+\omega^2\tau^2}}$. To find the frequency where the phase lag hits 180°, we first assume that $\Delta t \ll \tau$, so that on increasing ω we first have the 90° phase shift from the low-pass filter. Then when $e^{-s\Delta t} = e^{-i\omega\Delta t} = -i$ ($\omega\Delta t = \pi/2$), the system is unstable. This occurs for $\omega^* = \pi/2\Delta t$ and

$$K_p^* = \sqrt{1 + \left(\frac{\pi}{2}\right)^2 \cdot \frac{\tau^2}{\Delta t^2}} \approx \frac{\pi}{2} \cdot \frac{\tau}{\Delta t} = \pi \frac{\tau}{T_s}\ . \tag{4.10}$$

i.e., the maximum allowed gain will be of order τ/T_s, implying the need to sample much more frequently than the time scale τ in order for proportional feedback to be effective. In other words, the A/D-induced lag — and any other lags present — will limit the maximum amount of gain one can apply.

Although one might argue that delays due to A/D conversion can be avoided in an analog feedback loop, there are other generic sources of delay. The most obvious one is that one often wants to control higher-order systems, where the delay is already built into the dynamics one cares about. For example, almost any mechanical system is at least second order, and if one wants integral control, the effective system is third order. A more subtle point is that the ODE models of dynamical systems that we have been discussing are almost always reductions from continuum equations. Such a system will be described by partial differential equations that have an infinity of normal modes. The larger the system, the more closely spaced in frequency are the modes. Thus, if one measures

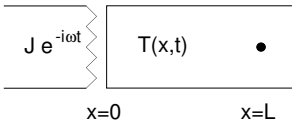

FIG. 15 Control of the temperature of a probe located a distance L inside a metal bar, with heater at the bar's end.

$G(i\omega)$ at higher frequencies, we expect to see a series of first- and second-order frequency responses, each one of which adds its 90° or 180° phase lag to G.

In the limit of large systems and/or high frequencies, the modes will be approximately continuous, and we will be guaranteed that the system will lag by 180° for high-enough frequencies. We can consider an example [simplified from a discussion in (Forgan, 1974)] of controlling the temperature of a thermometer embedded in a long (semi-infinite) bar of metal, with a heater at one end. See Fig. 15. The temperature field in the bar obeys, in a one-dimensional approximation,

$$D\frac{\partial^2 T}{\partial x^2} = \frac{\partial T}{\partial t} \ , \tag{4.11}$$

where $D = \lambda/\rho C_p$ is the thermal diffusivity of the bar, with λ the thermal conductivity, ρ the density, and C_p the heat capacity at constant pressure. The boundary conditions are

$$T(x \to \infty) \ = \ T_\infty \tag{4.12}$$

$$-\lambda \left.\frac{\partial T}{\partial x}\right|_{(0,t)} \ = \ J_0 e^{-i\omega t} \ , \tag{4.13}$$

with J_0 the magnitude of the power input.[11] In order to construct the transfer function, we assume a sinusoidal power input at frequency ω. The solution to Eqs. (4.11)-(4.13) is given by

$$\frac{T(x,t) - T_\infty}{J_0 e^{-i\omega t}} = \frac{1}{\sqrt{2}\lambda k} e^{i\left(kx + \frac{\pi}{4}\right)} e^{-kx} \ , \tag{4.14}$$

where $k = \sqrt{\frac{\omega}{2D}}$ and where we have written the solution in the form of a transfer function, evaluated at $s = i\omega$. If, as shown in Fig. 15, the thermometer is placed at a distance L from the heater, there will be a phase lag given by $kL + \frac{\pi}{4}$. This will equal 180° at a critical frequency

$$\omega^* = \frac{9}{8}\pi^2 \frac{D}{L^2} \ . \tag{4.15}$$

In other words, there will always be a mode whose frequency ω^* gives the critical phase lag. Here, D/L^2 may be thought of as the characteristic lag, or time it takes heat to propagate a distance L in the bar. In controlling the temperature of such a system, the controller gain will thus have to be low enough that at ω^*, the overall transfer function of system [Eq. (4.14) multiplied by the controller gain] is less than one. The calculation is similar to the one describing digitization lag.

The overall lesson is that response lags are generic to almost any physical system. They end up limiting the feedback gain that can be applied before spontaneous oscillation sets in. In many cases, a bit of care in the experimental design can minimize the pernicious effects of such lags. For example, in the temperature-control example discussed above, an immediate lesson is that putting the thermometer close to the heater will allow larger gains and tighter control (Forgan, 1974). Of course, the thermometer should also be near the part of the sample that matters, leading to a possible design tradeoff.

D. Non-minimum-phase systems

Delays, then, are one generic source of instability. Another arises when the system transfer function $G(s)$ has a zero in the right-hand plane. To understand what happens, consider the transfer functions $G_1(s) = 1$ and $G_2(s) = \frac{1-s}{1+s}$. Both have unit amplitude response for all frequencies (they are "all-pass" transfer functions(Doyle *et al.*, 1992)), but G_1 has no phase lag while G_2 has a phase lag that tends to 180° at high frequencies. Thus, all of our statements about leads and lags and about gain and phase margins must be revised when there are zeroes in the right-hand side of the s-plane. Such a transfer function describes a "non-minimum-phase" (NMP) system in control-theory jargon.[12] In addition to arising from delays, they can arise, for example, in controlling the position of floppy structures – e.g., the tip of a fishing rod or, to use a time-honored control example, the bob of a pendulum that is attached to a movable support. (Think of balancing a vertical stick in your hand.)

---

[11] One usually imposes an AC voltage or current across a resistive heating element. Because power is the square of either of these, the heater will inject a DC plus 2ω signal in the experiment. One can compensate for this nonlinearity by taking, in software, the square root of the control signal before sending it to the heater.

[12] The term "non-minimum-phase" (NMP) refers to Bode's gain-phase relationship, which states that for any transfer function $L(s)$ with no zeroes or poles in the right-hand plane, if $L(0) > 0$, the phase lag ϕ is given by

$$\phi(\omega_0) = \frac{1}{\pi}\int_{-\infty}^{\infty} \frac{d}{d\nu} \ln|L(i\omega)| \ln\left[\coth\left(\frac{|\nu|}{2}\right)\right] d\nu \ , \tag{4.16}$$

with $\nu = \ln(\omega/\omega_0)$ the normalized frequency. [See (Özbay, 2000), but note the misprint.] The phase at ω_0 thus depends on $|L|$, over all frequencies. However, if $|L| \sim \omega^{-n}$ over at least a decade centered on ω_0, then Eq. (4.16) is well-approximated by $\phi(\omega_0) \approx -n\frac{\pi}{2}$. Transfer functions that have more than this minimum lag are NMP systems. See (Franklin *et al.*, 1998), pp. 254–256. One can show that a general transfer function $G(s)$ can always be written as the product $G_1(s)G_2(s)$, where $G_1(s)$ is minimum phase and $G_2(s)$ is an "all-pass" filter with unit amplitude response. See Doyle *et al.* (1992), section 6.2.

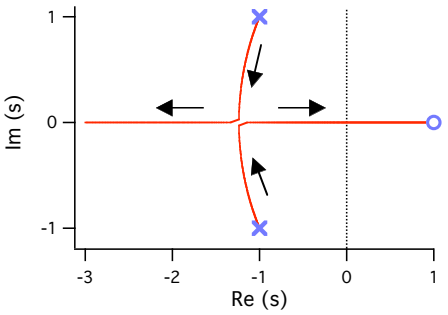

FIG. 16 (Color in online edition) Root-locus plot of the poles of Eq. (4.18) as a function of the gain, with $z = 1$ and $p = -1 + i$. The crosses denote the poles when $K_p = 0$, while the circle denotes the zero. The two poles approach each other, meeting and becoming real when $K_p = -4+2\sqrt{5} \approx 0.47$. One root then approaches 1 as $K_p \to \infty$, crossing zero at $K_p = 2$, while the other root goes to $-\infty$. The system becomes unstable at $K_p = 2$.

If an NMP system has an odd number of zeros in the right-hand plane, the time response to a step will initially be in the direction *opposite* the control signal, so that is one indication that such behavior is present (Vidyasagar, 1986). The controller must be "smart enough" to handle the opposite response of the system. Another way to see that NMP systems can be difficult to control is to note the effects of increasing the overall controller gain. We write the system transfer function $G(s) = \frac{N_G(s)}{D_G(s)}$ and the controller as $K(s) = K_p N_K(s)/D_K(s)$, where K_p is an overall gain. The loop gain $L(s) = K(s)G(s) \equiv K_p N(s)/D(s)$, and the sensitivity function $S(s)$ (Sec. III.A) is given by

$$S(s) = \frac{1}{1+L} = \frac{D(s)}{D(s) + K_p N(s)} \,. \tag{4.17}$$

Instabilities arise when the denominator of S, $\chi(s) \equiv D(s) + K_p N(s) = 0$. Clearly, as K_p increases, the zeroes of $\chi(s)$ move from the roots of $D(s)$ to the roots of $N(s)$. In other words, the poles of the closed-loop system move towards the zeroes of the open-loop system.

As an example, consider

$$G(s) = \frac{s-z}{(s-p)(s-p^*)} \tag{4.18}$$

and $K(s) = K_p$. Then $\chi(s) = s^2 + [-(p+p^*) + K_p]s + pp^* - K_p z$. Fig. 16 graphs the closed-loop poles in the complex s-plane as a function of K_p. Such "root-locus" plots can be useful in getting an intuition about a system's dynamics.[13] Note that while one often plots the pole movement as a function of overall gain, one can do the same for any parameter.

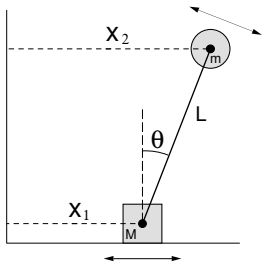

FIG. 17 Model of a hand balancing a vertical rod.

To get a feel for how and when NMP systems arise, we consider the example of balancing a rod in one's hand (Doyle *et al.*, 1992). See Fig. 17 for an illustration, where the hand is idealized as a mass M constrained to move horizontally, and the rod is taken to be a simple pendulum with massless support of length L and a bob of mass m. The deflection of the rod about the unstable vertical equilibrium is given by θ. The position of the hand (and bottom of rod) is $[x_1(t), 0]$ and that of the rod end (bob) is $[x_2(t), y_2(t)]$. The Lagrangian $\mathcal{L} = T - V$ is

$$\mathcal{L} = \frac{1}{2}M\dot{x}_1^2 + \frac{1}{2}m(\dot{x}_2^2 + \dot{y}_2^2) - mgL\cos\theta \,, \tag{4.19}$$

with $x_2 = x_1 + L\sin\theta$ and $y_2 = L\cos\theta$. If we neglect friction, this leads to nonlinear equations of motion for x_1 and θ:

$$\begin{aligned} (M+m)\ddot{x}_1 + mL(\ddot{\theta}\cos\theta - \dot{\theta}^2\sin\theta) &= u \,, \\ m(\ddot{x}_1\cos\theta + L\ddot{\theta} - g\sin\theta) &= d \,, \end{aligned} \tag{4.20}$$

where $u(t)$ is the force exerted by the hand and $d(t)$ is any disturbing torque on the rod itself. Linearizing about the unstable vertical equilibrium, we have

$$\begin{aligned} (M+m)\ddot{x}_1 + mL\ddot{\theta} &= u \,, \\ \ddot{x}_1 + L\ddot{\theta} - g\theta &= \frac{1}{m}d \,. \end{aligned} \tag{4.21}$$

The system transfer function from u to x_1 (rod bottom) is then easily found to be $G_1(s) = \frac{Ls^2 - g}{D(s)}$, with $D(s) = s^2[MLs^2 - (M+m)g]$. One can see that G_1 is unstable, with RHP poles at 0, 0, and $\sqrt{\frac{(M+m)g}{ML}}$, and NMP, with

[13] Since the point of a root-locus plot is to build intuition, one might wonder whether the traditional way of presenting such plots, as exemplified by Fig. 16, is the best one can do. Indeed, the more modern way to make a root-locus plot is to do so interactively, using a mouse to drag a slider controlling some parameter while updating the pole positions in real time. Such features are implemented, for example, in the Matlab Control toolbox (see Sec. V.C.2 on commercial tools) or are easy to create in more general graphics applications such as Igor Pro (Wavemetrics, Inc.) or Sysquake (Calerga Sarl).

a RHP zero at $\sqrt{g/L}$. On the other hand, the transfer function between u and x_2 (rod top) is $G_2(s) = \frac{-g}{D(s)}$. This has the same poles as G_1 but lacks the RHP zero. Thus, G_1 is NMP, while G_2 is not. The conclusion is that it is much easier to balance a rod looking at the top (i.e., measuring x_2) than it is looking at the bottom (i.e., measuring x_1). The reader can easily check this; a meter stick works well as a rod.

An example of an NMP system that arises in physics instrumentation is found in a recent study of the dynamics of a piezoelectric stage (Salapaka *et al.*, 2002). The article has a good discussion of how the odd impulse-response of such NMP systems make them more difficult to control and what to do about them.

E. MIMO vs. SISO systems

In section III.C, we distinguished between multiple-input-multiple-output (MIMO) and single-input-single-output systems. Here, we discuss briefly why MIMO systems are usually harder to control. Although MIMO systems are usually defined in the state-space formulation [e.g., Eq. (2.2)], they can also be defined in the frequency domain. One uses a matrix of transfer functions $\widetilde{G}(s)$, where the ij element, $g_{ij}(s)$, is the ratio $\frac{x_i(s)}{u_j(s)}$, that is, the ratio of the ith output to the jth input. Let us consider a simple example that illustrates the extra difficulties that multiple inputs and outputs can bring. Recall that for the SISO case, a first-order system ($\frac{1}{1+\tau s}$) is stable under proportional feedback for any positive gain k. [Eqs. (3.3)-(3.4)]. Consider now a 2x2 generalization

$$\widetilde{G}(s) = \begin{pmatrix} \frac{1}{1+\tau_1 s} & \frac{1}{1+\tau_2 s} \\ \frac{1}{1+\tau_2 s} & \frac{1}{1+\tau_1 s} \end{pmatrix} . \qquad (4.22)$$

If one prefers a physical picture, think about a shower where pipes carrying hot and cold water are mixed to form a single stream. The shower output is characterized by an overall flow and temperature. Assuming there are separate hot and cold shower knobs, one has two control inputs and two outputs. If the τ's in Eq. (4.22) were all different, they would reflect the individual time constants to change flow and temperature. The rather unrealistic choice of τ's in Eq. (4.22) simplifies the algebra.

Let us now try to regulate the shower's temperature and flow by adjusting the hot and cold flows with proportional gains k that are assumed identical for each. The controller matrix $\widetilde{K}(s)$ is then $k\widetilde{1}$, with $\widetilde{1}$ the 2x2 identity matrix. The loop stability is determined by the matrix analog of Eq. (3.2), $\widetilde{S} = (\widetilde{1} + \widetilde{G}\widetilde{K})^{-1}$, where the closed-loop system is unstable when one of the eigenvalues of $\widetilde{S}$ has positive real part for any value of $s = i\omega$, generalizing the ideas of Sec. IV.B. (Recall that the order of matrix multiplication matters.) After some manipulation, one can easily show that the eigenvalues of $\widetilde{S}$ are given by

$$\lambda_{\pm}(s) = \frac{1}{1 + k(\frac{1}{1+\tau_1 s} \pm \frac{1}{1+\tau_2 s})} . \qquad (4.23)$$

The negative root, λ_-, has poles at

$$\tau s \approx -\left(1 + \frac{k\Delta\tau}{2\tau}\right) \pm \sqrt{\frac{k\Delta\tau}{\tau} + \left(\frac{k\Delta\tau}{2\tau}\right)^2} , \qquad (4.24)$$

where $\tau_1 \approx \tau_2 \equiv \tau$ and $\Delta\tau = \tau_2 - \tau_1$. This implies an instability for gains larger than $k^* \approx \frac{2\tau}{\tau_1 - \tau_2}$. Thus, for $\tau_1 > \tau_2$, the system will be unstable using only proportional gain, in contrast to its SISO analog. Comfortable showers are not easy to achieve, as anyone who has been in the shower when someone else flushes the toilet can attest!

The reason that MIMO systems are so "touchy" is roughly that the sensitivity matrix $\widetilde{S}$ has different gains in different directions.[14] If the largest and smallest eigenvalues of $\widetilde{S}$ are λ_{max} and λ_{min}, respectively, then the maximum gain one can apply before the system is unstable is determined by λ_{max} while the closed-loop bandwidth is set by λ_{min}. Thus, systems with widely ranging eigenvalues will have compromised performance. A measure of the severity of this compromise is given by the ratio $\lambda_{max}/\lambda_{min}$, known as the *condition number* of the matrix $\widetilde{S}$. (More generally, this discussion should be phrased in terms of singular values, rather than eigenvalues.[15] Also, since singular values – like eigenvalues

---

[14] Another reason that the specific $\widetilde{G}(s)$ in Eq. (4.22) is difficult to control is that it has rank 1 at zero frequency. (The determinant vanishes at DC.) The system is thus nearly uncontrollable – in the technical and intuitive senses of the word – at low frequencies.

[15] The eigenvalue is not the right quantity to characterize the gain, in general, for two reasons: First, whenever the number of inputs differs from the number of outputs, the matrix $\widetilde{S}$ is not square. Second, eigenvalues give only the gain along certain directions and can miss subtle types of "cross gains." For example, consider the matrix

$$\widetilde{S} = \begin{pmatrix} 1 & 0 \\ 100 & 1 \end{pmatrix} , \qquad (4.25)$$

which implies that the unit input $(1\ 0)^T$ gives rise to an output $(1\ 100)^T$, which has a gain of roughly 100, even though the two eigenvalues of $\widetilde{S}$ are unity. The appropriate generalization of the notion of eigenvalue is that of a singular value. The singular-value decomposition, SVD, of an m x n matrix $\widetilde{S}$ is given by $\widetilde{S} = \widetilde{U}\widetilde{\Sigma}\widetilde{V}^T$, where $\widetilde{U}$ is an m x m unitary matrix, $\widetilde{V}^T$ is the transpose of an n x n unitary matrix, and where $\widetilde{\Sigma}$ is an m x n matrix that contains $k = \min(\{m, n\})$ non-negative singular values σ_i (Skogestad and Postlethwaite, 1996). For example, one can show that

$$\sigma_{max}(\widetilde{S}) = \sup_{\vec{w}} \frac{||\widetilde{S}\vec{w}||_2}{||\vec{w}||_2} , \qquad (4.26)$$

where $||\cdot||_2$ denotes the Euclidean norm. Equation (4.26) states that σ_{max} is the largest gain for all inputs $\vec{w}$. This quantity is then the largest singular value of $\widetilde{S}$. One can also show that the σ's are the square roots of the eigenvalues of $\widetilde{S}^T\widetilde{S}$. In the

– depend on frequency, the statements above should be interpreted as having to apply at each frequency.)

We do not have space here to explore the various strategies for dealing with a system $\widetilde{G}$ that is ill conditioned (i.e., has a large condition number), except to point out the obvious strategy that if one can implement a controller $\widetilde{K}(s) = \widetilde{G}(s)^{-1}$, then one will have effectively skirted the problem. Indeed, any controller leading to a diagonal "loop matrix" $\widetilde{L}(s) = \widetilde{G}\widetilde{K}$ will reduce the problem to one of independent control loops that may be dealt with separately, thus avoiding the limitations discussed above. Note that it may be sufficient to diagonalize $\widetilde{L}$ in a limited frequency range of interest. Unfortunately, this "ideal" strategy usually cannot be implemented, for reasons of finite actuator range and other issues we have discussed in connection with the SISO case. One can can try various compromises, such as transforming $\widetilde{L}$ to a block-diagonal matrix and reducing the condition number (Skogestad and Postlethwaite, 1996).

V. IMPLEMENTATION AND SOME ADVANCED TOPICS

The ideas in the above sections are enough to make a good start in applying feedback to improve the dynamics of an experimental system. There are, however, some subtle points about implementing feedback in practice, which lead to more advanced methods. First, we have previously assumed that one knows the system dynamics $G(s)$, but this often needs to be determined experimentally. We accordingly give a brief introduction to the subject of experimental measurement of the transfer function and "model building." If the resulting model is too complicated, one should find a simpler, approximate model, using "model reduction." Second, one must choose a controller, which implies choosing both a functional form and values for any free parameters. These choices all involve tradeoffs, and there is often an optimum way to choose. Third, modern control systems are usually implemented digitally, which introduces issues associated with the finite sampling time interval T_s. Fourth, measurements are almost always contaminated by a significant amount of noise, which as we have already seen in Eq. (3.13), can feed through to the actuator output of a controller. Fifth, control is always done in the face of uncertainty – in the model's form, in the parameters, in the nature of disturbances, etc., and it is important that feedback systems work robustly under such conditions. We consider each of these in turn.

example of Eq. (4.25), one finds the singular values σ are ≈ 1 and ≈ 100, showing that the singular values capture better the "size" of $\widetilde{S}$ than do the eigenvalues.

A. Experimental determination of the transfer function

Although often overlooked by hurried experimentalists, the proper place to begin the design of a control loop is with the measurement of the system transfer function, $G(s)$, the frequency-dependent ratio of system output to input. (Even if you think you know what $G(s)$ should be, it is a good idea to check out what it is.) This topic is deceptively simple, because it actually implies four separate steps: First, one must measure experimentally the transfer function; second, one must fit a model transfer function to it; third, because a full description of the experimental transfer function usually leads to very high-order systems, one needs a way to approximate a high-order system accurately by a lower-order system; and fourth, one should always ask whether the system can be "improved" to make control easier and more effective.

1. Measurement of the transfer function

The transfer function, $G(s)$ can be inferred from Bode amplitude and phase plots. The simplest way to make such measurements requires only a function generator and an oscilloscope (preferably a digital one that measures the amplitudes and phases between signals). One inputs a sine wave from the function generator into the experimental input, $u(t)$, and records the output, $y(t)$. (We assume a SISO system, but the MIMO generalization is straightforward.) By plotting the input and output directly on two channels of an oscilloscope, one can read off the relative amplitude and phase shifts as a function of the driving frequency. A better technique is to use a lock-in amplifier, which gives the amplitude and phase shifts directly, with much higher precision than an oscilloscope. They often can be programmed to sweep through frequencies automatically. ("Dynamic signal analyzers" and "network analyzers" automate this task.) Bear in mind, though, that the transfer function measured may not be fixed for all time. For example, the frequency of a mechanical resonator varies with its external load. For this reason, a good control design should not depend crucially on an extraordinarily accurate determination of the transfer function. We will pursue this point in Sec. V.E, on robust control.

The above discussion shows how to measure the transfer function directly in the frequency domain. Whenever such a measurement is possible, the results are very intuitive, and I would recommend this approach. However, for slow systems, a very long time may be required to measure the transfer function, and one may prefer time-domain methods – indeed, they are standard in the control field. The basic idea is to excite the system with a known input $u(t)$ and measure the response $y(t)$. One then computes the correlation function $R_{uy}(\tau)$ between the input and output, $R_{uy}(\tau) = \langle u(t)y(t+\tau)\rangle$ and also $R_{uu}(\tau) = \langle u(t)u(t+\tau)\rangle$, the autocorrelation function of the input. The transfer function, $G(s)$ is then found

by taking the Fourier transforms of R_{uy} and R_{uu} and computing $G = R_{yu}/R_{uu}$.[16] Time-domain methods are more efficient than frequency methods because $u(t)$ can contain all frequencies – using the system's linearity, one measures all frequencies in a Bode plot simultaneously.

In theory, one could choose $u(t)$ to be an impulse, which means that $R_{uy}(\tau)$ would be the impulse response function (the Fourier transform of $G(s = i\omega)$). In practice, it is often hard to inject enough energy to make an accurate measurement. A step-function input, which has a power spectrum $1/\omega^2$, is very easy to implement and injects enough energy at low frequencies. At higher frequencies, the injected energy is low and noise dominates. Another good choice for $u(t)$ is a "pseudorandom binary sequence" (PRBS), which is a kind of randomly switching square wave that alternates stochastically between two values. This shares with the delta-function input the property of having equal energies at all relevant frequencies. For an introduction, see Dutton *et al.* (1997); for full details, see Ljung (1999).

At this stage, it is also good to check the linearity of the system. At any frequency, one can vary the input amplitude and record the output amplitude. Note that nonlinear effects can be hysteretic – the response upon increasing the input may be different from that obtained upon decreasing the input. Almost all of the techniques described here are designed to work with linear systems. They may be adequate for small nonlinearities but will fail otherwise. (Cf. Section VI.) Remember, though, that one can use linear techniques to stabilize a system about a setpoint, even if there are strong nonlinearities as the setpoint is varied. This was the case in the interferometer example discussed in Sec. III.D.1. If the system is locally linear, the structure of the measured transfer function is usually constant as one varies the set point, but the positions of poles and zeros vary smoothly. One can then design a controller whose parameters vary smoothly with set point, as well.

2. Model building

Given measurements of the transfer function at a set of frequencies s_i, $G(s_i)$, one should work out a reasonable analytic approximation. Of course, any physics of the system should be included in the model, but often systems are too complicated to model easily. In such cases, one tries to make a model from standard elements — poles and zeroes, which individually correspond to low- and high-pass filters (first order), resonances and anti-resonances (second order), as well as delays, which technically are "infinite order," since they cannot be written as a ratio of finite polynomials. It usually is possible to do this, as most often the system can be decomposed into modes with reasonably well-separated frequencies. If there are many nearly degenerate modes, or if the system is best described as a spatially extended system, then these techniques can break down, and one should refer to the various control-theory texts for more sophisticated approaches. Usually, one can do quite well looking for 90° and 180° phase shifts and identifying them with first- and second-order terms. Remember that lags correspond to terms in the denominator and leads (forward phase shifts) correspond to terms in the numerator. Any linear phase shift with frequency on the Bode plot corresponds to a lag in the time domain. (If the signal is digitized and fed to a computer, there will be such a lag, as discussed in Sec. IV.C. Cf. Sec. V.C, below.) Once there is a reasonable functional form, one can determine various coefficients (time constants, damping rates, etc.) by a least-squares fit to the measured transfer function. Alternatively, there are a number of methods that avoid the transfer function completely: from a given input $u(t)$ and measured response $y(t)$, they directly fit to the coefficients of a time-domain model or directly give pole and zero positions (Ljung, 1999).

3. Model reduction

If the result of the model-building exercise is a high-order system, then it is often useful to seek a lower-order approximation. First, such an approximation will be simpler and faster to compute. This may be important in implementing a controller, as we have already seen that state-space controllers require observers, which require one to model the system dynamics on a computer much more rapidly than timescales of the real dynamics. (The same remark applies to the choice of a controller – lower-order ones are easier to implement.) The second reason for preferring low-order approximations is that the smaller, higher-order parts of the dynamics are not very robust, in the sense that any parameters tend to vary more than the "core" parts of the dynamical system. We will return to this in our discussion of robust systems, below (Sec. V.E).

Given Bode magnitude and phase plots of experimental data for a given transfer function, how should one simplify the functional form? The obvious strategy is *truncation*: one simply keeps enough terms to fit the transfer function accurately up to some maximum frequency ω_{max}. At a minimum, one should have $\omega_{max} > \omega_0$, the desired feedback bandwidth. Ideally, one should keep enough terms that the gain (relative to the zero-frequency, or DC, gain) at ω_{max} is $\ll 1$. Note that the usual structure of transfer functions means that higher-order terms are multiplied, not added, onto lower-order

[16] Here is a quick proof, following Doyle *et al.* (1992). In the time domain $y(t) = \int_{-\infty}^{\infty} G(t')u(t-t')dt'$. Multiplying both sides by $u(t)$ and shifting by τ gives $u(t)y(t+\tau) = \int_{-\infty}^{\infty} G(t')u(t)u(t+\tau-t')dt'$. Averaging gives $R_{uy}(\tau) = G * R_{uu}(\tau)$. Then Fourier transform and use the convolution theorem.

terms, i.e., that we truncate a system

$$G(s) = \prod_{i,j=1}^{\infty} \frac{s - z_i}{s - p_j} \tag{5.1}$$

to

$$G_t(s) = \prod_{i,j=1}^{N} \frac{s - z_i'}{s - p_j'} \tag{5.2}$$

where the last (Nth) pole or zero occurs at a frequency $\approx \omega_{max}$. Note that the zeroes and poles of the reduced system – found by refitting the data to the truncated form for $G(s)$ – will in general differ from the "exact values" of lower-order zeroes/poles of the full system.[17]

While the above method seems straightforward, it is not always the best thing to do. For example, if there happens to be a high-frequency mode (or modes) with a large amplitude, neglecting it may not wise. A more subtle situation is that there may be so many high-frequency modes that even though the amplitude of any one is small, they may have a collective effect. To deal with such cases, one strategy is to try to order the system's modes not by frequency but by some measure of their "importance." One definition of "importance" is in terms of the *Hankel singular values* of the system transfer function $G(s)$. The detailed discussion is beyond the scope of this tutorial, but the rough idea is to appeal to the notions of controllability and observability of the system modes discussed above. The singular values of the matrix describing controllability rank the effect of an input on a particular system mode, while the singular values of the matrix describing observability rank the effect of the dynamics on a particular output mode. The product of the relevant two matrices ("gramians") then gives the strength of "input-output" relationships of each mode of the system. One can show that it is possible to use the freedom in defining a state-space representation of a system to make these "gramian" matrices equal – the "balanced representation." Ordering these "balanced" modes by their singular values and retaining only the largest ones gives a systematic approach to model reduction (Skogestad and Postlethwaite, 1996).

The kind of systematic model reduction discussed above has recently attracted the attention of statistical physicists, who are faced with a similar task when "coarse-graining" dynamics in a thermodynamic system. For example, if one considers a small number of objects in contact with many other elements of a "heat bath," then a common strategy is to integrate out the degrees of freedom corresponding to the bath. One derives thus a reduced dynamics for the remaining degrees of freedom. A similar task arises when one uses the renormalization group (Goldenfeld, 1992). Coarse graining is thus a kind of model reduction. The interest in control strategies is that they give a systematic way of handling a general system, while the usual coarse-graining strategy is more *ad hoc* and fails, for example, when naively applied to spatially inhomogeneous systems. A recent paper by Reynolds (2003) explores these issues.

4. Revisiting the system

We close by noting that "determination of the transfer function" has two meanings: Whatever the system dynamics are, the experimentalist should measure them; but the experimentalist also has, in most cases, the power to influence greatly, or determine, the system itself. If a system is hard to control, think about ways of changing it to make control easier. (This is a basic difference in philosophy from most engineering texts, where the system – "plant" in their jargon – is usually a given.) Often, the experimentalist simultaneously designs both the physical (mechanical, electrical) aspects and the control aspects of an experiment. We have already seen two examples: in temperature control, where minimizing the physical separation between heater and thermometer makes a system much easier to control and allows higher performance; and in balancing the rod, where one should look at the top of the rod, not the bottom. More generally, one should minimize the delay and/or separation between the actuator and its sensor. Another area in which good design plays a role is in the separation of modes. As we have discussed above, closely spaced modes are difficult to control, and it usually pays to make different modes as widely spaced in frequency as possible. For example, if one is trying to isolate vibrations, the apparatus should be as small and cube shaped, to maximize the frequency separation between the soft isolating spring and the lumped experimental mass and the internal modes of the experimental apparatus. In the next section, we shall see that systems that are up to second order over the

[17] Note that my definition of truncation differs from that used in standard control texts, such as Skogestad and Postlethwaite (1996), Ch. 11. There, the authors first define a truncation where the poles and zeroes are held to be the same. If one then looks at the difference between the exact transfer function and its truncated approximation, one sees that they differ most at low frequencies and the difference goes to zero only at infinite frequencies. But the whole point of doing a truncation is almost always to capture the low-frequency dynamics while disregarding high frequencies! Thus, this kind of truncation seems unlikely to be useful in practice. Skogestad and Postlethwaite (1996) then go on to introduce "residualization" (physicists would use the terms "slaving" or adiabatic elimination of fast variables), where, in the time domain, the fast mode derivatives are set to zero and the steady states of the fast modes are solved. These instantaneous, or adiabatic steady states are then substituted wherever fast variables appear in the equations for the remaining slow modes. The equations for the slow modes then form a smaller, closed set. If the modes are well-behaved – i.e., well-separated in frequency from each other – then the procedure described in the text, fitting to a form up to some maximum frequency, will give essentially equivalent results. Both methods share the property of agreeing well at low frequencies up to some limit, above which the approximation begins to fail.

frequency range of interest may be perfectly controlled by the elementary "PID" law commonly found in commercial controllers but higher-order systems in general cannot be adequately controlled by such laws.

Another example of how the design of an experimental system determines the quality of control comes from my own laboratory, where we routinely regulate temperature to 50 μK RMS near room temperature, i.e., to fractional variations of 2×10^{-7} (Metzger, 2002; Yethiraj *et al.*, 2002). In order to obtain such performance, the simple proportional-integral (PI) law described above sufficed. (A more sophisticated control algorithm would have further improved the control, but we did not need a better performance.) The crucial steps all involved the physical and electrical design of the system itself. One important idea is to use a bridge circuit so that the error signal is centered on zero rather than about a finite level and may thereafter be amplified. We minimized sensor noise by using a watch battery to power the bridge.[18] The overall performance turned out to be set by the temperature variations in the non-sensor resistors in the bridge. The best place to have put these would have been next to the sensor resistor itself, as that is where the temperature is most stable. In our case, that was inconvenient, and we put them inside a thick styrofoam box. Slow variations in the bridge temperature then show up as setpoint drifts that cannot be corrected by the simple feedback loop; however, the temperature was very stable on the 1-minute time scale of our particular experiment. To stabilize over longer times, we could have added a second feedback loop to regulate the temperature of the external components of the bridge. The time scales of the two controllers should be well-separated to decouple their dynamics.

B. Choosing the controller

Having identified the system (and, perhaps, having already made the system easier to control), we need to choose the control algorithm. In the frequency-domain approach, this means choosing the dynamics $K(s)$. This task is commonly broken down into two parts: choosing a general form for $K(s)$ and choosing, or "tuning," the free parameters.

[18] One would be tempted to use a lock-in amplifier to supply an AC voltage to the bridge. The lock-in technique, by centering the bandwidth about a finite carrier frequency, can lower sensor noise by moving the passband to a high enough frequency that $1/f$ noise is unimportant. In our own case, this was not the limiting factor for performance.

1. PID controllers

Probably the most common form for $K(s)$ is the "PID" controller, which is a combination of proportional, integral, and differential control:

$$K(s) = K_p + \frac{K_i}{s} + K_d s \ , \tag{5.3}$$

where K_p, K_i, and K_d are parameters that would be tuned for a particular application. We have already discussed the general motivations for proportional and integral control. The intuitive justification for derivative control is that if one sees the system moving at high "velocity," one knows that the system state will be changing rapidly. One can thus speed the feedback response greatly by anticipating this state excursion and taking counteraction immediately. For example, in controlling a temperature, if the temperature starts falling, one can increase the heat even before the objects has cooled, in order to counteract the presumably large perturbation that has occurred. The word "presumably" highlights a difficulty of derivative control. One infers a rapid temperature change by measuring the derivative of the system state. If the sensor is noisy, random fluctuations can lead to large spurious rates of change and to inappropriate controller response. Thus, many experimentalists try derivative control, find out that it makes the system noisier, and then give up. Since there are many benefits to derivative control and since spurious response to noise can be avoided, this is a shame, and hopefully this article will motivate experimentalists to work around the potential problems.

In order to better understand derivative control, we can look at it in frequency space ($K_d s$). The linear s-dependence means that the response increases with frequency, explaining why high-frequency sensor noise can have such a great effect. One obvious response is to limit the action of the derivative term by adding one or more low-pass elements the control law, which becomes

$$K(s) = \frac{K_p + \frac{K_i}{s} + K_d s}{(1 + s/\omega_0)^n} \ , \tag{5.4}$$

where we have added n low-pass filters with cutoff frequencies all at ω_0. Indeed, since no actuator can respond with large amplitude at arbitrarily large frequencies, some kind of low-pass filter will be present at high frequencies, whether added deliberately or not. As long as ω_0 is higher than the feedback bandwidth, it will have little effect on the system's dynamics, while limiting the effects of sensor noise. A more sophisticated way to minimize sensor-noise feedthrough, the Kalman filter, will be discussed in Section V.D, below.

2. Loopshaping

Many experimentalists limit their attention to PID controllers (and often just PI controllers if they have no

luck with the derivative term). While PID controllers can give good results and have the advantage that each term has an intuitive justification, they are by no means the only possibility. Indeed, the frequency-domain approach we have developed suggests that one can think of choosing $K(s)$ to "sculpt" or "shape" the closed-loop response $T(s)$. For example, given a system $G(s)$, one can invert Eq. (3.2) and write

$$K(s) = \frac{T(s)}{1-T(s)} G^{-1}(s) \ , \tag{5.5}$$

If one wanted $T(s) = 1/(1 + s/\omega_0)$, for example, one could choose $K(s) = \frac{\omega_0}{s} G^{-1}(s)$. If $G(s) = 1/(1 + 2\zeta s/\omega_1 + s^2/\omega_1^2)$, then the resulting $K(s)$ has the form $2\zeta/\omega_1 + 1/s + s/\omega_1^2$, which is the PID form. We thus have another justification for using the PID form – second-order systems are common – and an understanding of why the PID form is not always satisfactory: many systems are higher than second order over the required frequency bandwidth.

The above technique of "inverse-based controller design" sounds too good to be true, and often it is. One catch is that it often is not possible for the actuator u to give the desired response $[u(s) = K(s)e(s)$, or the equivalent time-domain requirement for $u(t)]$. All actuators have finite ranges, but nothing in the inversion design limits requirements on u. One systematic failure is that the higher the order of the characteristic polynomial in the denominator of G (n'th order for an n'th order system) the higher the derivative required in K and hence the larger u required at higher frequencies. Even one derivative cannot be implemented at arbitrarily high frequencies and must be cut off by some kind of low-pass filter, as described above. The problem is more severe with higher-order terms in the numerator of K. Thus, in general, one has to augment G^{-1} at least with low-pass elements. One then has to worry whether the additional elements degrade the controller and whether alternative structures might in fact be better.

More broadly, one can think of choosing a fairly arbitrary form for $K(s)$ in order that $G(s)$ have desirable properties. The general principle is to make the gain high ($|L| = |KG| \gg 1$) over the frequency range of interest, to track the reference and reject disturbances, while making sure that the phase lag of the loop is not too close to 180° near $|L| = 1$, to give an adequate phase margin. Recall that nearly all physical systems and physical controllers have limited response at high frequency, implying that $|L| \to 0$ as $\omega \to \infty$. In addition, recall the tradeoff between accurate tracking of a control signal r and rejection of sensor noise n: one wants $|L| \ll 1$ at frequencies where noise dominates over disturbances and control signals. [See Eq. (3.13).] Fortunately, control signals and disturbances are often mostly low frequency, while sensor noise is broad band and goes to high frequencies. Thus, one easy approach is to use large gains over the frequency range one wishes to track the control signal and small gains at higher frequencies.

One subtlety concerns frequencies near the crossover where $|L| = 1$. Because transfer functions are meromorphic (analytic everywhere but at their poles), the real and imaginary parts of transfer functions are interrelated, giving rise to a number of analytical results (along the lines of the various Cauchy theorems of complex analysis and the Kramers-Krönig relations) [Doyle *et al.* (1992), Ch. 6.] One of these is Bode's gain-phase relationship (Sec. IV.D, footnote 12), which shows that for a minimum-phase system, the phase lag is determined by a frequency integral over the transfer-function magnitude. The practical upshot, which we saw first in Sec. IV.B, is that if $L \sim (i\omega)^{-n}$ over a frequency range near a reference frequency ω_0 (here, taken to be the crossover frequency where $|L(i\omega_0)| = 1$, then the phase lag is $\approx -n\frac{\pi}{2}$. Because a phase of −180° implies instability, the transfer function $L(s)$ should show approximately single-pole, low-pass behavior near ω_0. In particular, the gain-phase theorem implies that an "ideal" low-pass filter that cuts off like a step function would *not* be a good way to limit the controller bandwidth.

Putting these various constraints – high gain at low frequencies, low gain at high frequencies, and single-pole behavior near the crossover frequency – one arrives at a "loopshape" for L that looks qualitatively like the sketch in Fig. 18a (Özbay, 2000). The left black arrow depicts schematically the desired tracking properties at low frequencies, where L should be larger than some bound, while the right black arrow depicts the desired noise suppression at high frequencies, where L should be smaller than some bound. The parallel black lines illustrate the ω^{-1} scaling desired near $|L| = 1$. Fig. 18b is the corresponding Bode phase plot, which shows that the loop has a gain margin ≈ 10 and a phase margin ≈ 70°. One also sees that the way to satisfy these constraints is to use an L with LHP zeroes (generalizations of derivative control), which can add to the phase. These various tricks constitute the engineering "art" of control design, which, though surely not optimal or systematic, often works well. Remember, finally, that the above discussion pertains to the open-loop transfer function $L(s)$. The actual controller is given by $K(s) = L(s)/G(s)$. In practice, one simply plots $|L|$ while "tweaking" the form and parameters of K.

Two common elements that can be used to shape the frequency response in a limited frequency range are lag and lead compensators. A lag compensator has the form

$$K_{lag}(i\omega) = \frac{a(1 + i\omega/\omega_0)}{1 + ai\omega/\omega_0} \tag{5.6}$$

where a is a constant (typically of order 10) and ω_0 is the scale frequency. Fig. 19 shows the Bode response of Eq. (5.6). The gain goes from a at low frequencies ($\omega \ll \omega_0$) to 1 at high frequencies ($\omega \gg \omega_0$). The phase response has a transient lag in the crossover frequency region but goes to zero at both low and high frequencies. One can think of a lag compensator as either an approxi-

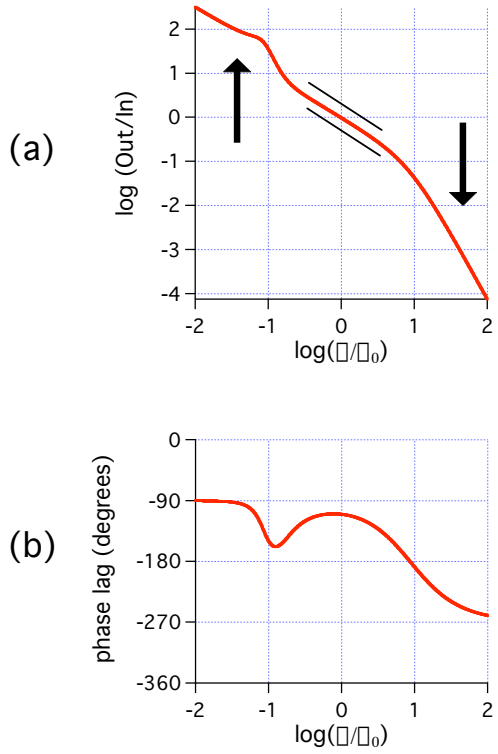

FIG. 18 (Color in online edition) (a) Schematic of a desired loopshape $L(s)$. (b) Corresponding phase plot.

mation to integral control (a boost of a at zero frequency rather than infinite gain) but without the often troublesome phase lag at higher frequencies. Lag compensators can also act as an approximation to low-pass filters at high frequencies, (one would typically remove the overall factor of a so that the gain goes from 1 to $1/a$). Again, the advantage is that the phase lag does not accumulate asymptotically.

Conversely, a lead compensator is of the form

$$K_{lead}(i\omega) = \frac{1 + ai\omega/\omega_0}{1 + i\omega/\omega_0} \tag{5.7}$$

which gives a phase lead in the crossover frequency range. The amplitude goes from 1 to a and thus does not indefinitely increase the way a pure derivative term would.

As a general conclusion, one should be open to adding enough elements – whether low-pass filters, lead or lag compensators, or other forms – to shape the response $L(s)$ as desired. For an effectively second-order system, a three-term controller can work well, but a higher-order system will require compensating dynamics $K(s)$ that is higher order and will depend on more free parameters, which must be chosen, or "tuned." How to do this is the subject of optimal control.

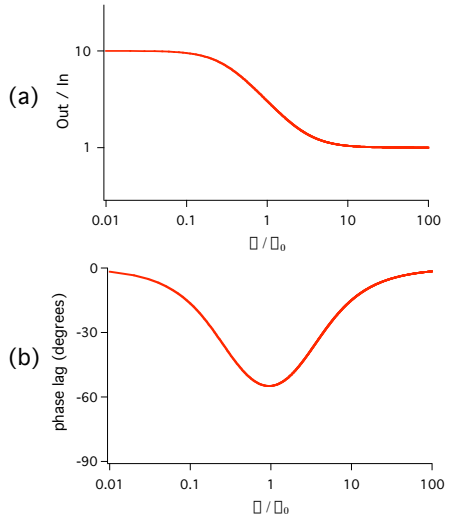

FIG. 19 (Color in online edition) Bode plots of a lag compensator [Eq. (5.6)] with $a = 10$.

3. Optimal control

The algorithm that many experimentalists use in tuning a PID control loop is (1) start by turning up the proportional gain until the system oscillates, and then turn down the gain somewhat; (2) add integral gain to eliminate setpoint droop, and turn down the proportional gain a bit more to offset the destabilizing lag of the integral term; (3) add a derivative term, become frustrated as sensor noise feeds into the system, give up and settle for the PI parameters of step (2). We have already commented on the usefulness of the derivative term and how other elements such as lead and lag compensators can shape the loop gain as a function of frequency to minimize the problem of noise feedthrough. But now with three or more terms to tune, one might wonder whether there is a systematic way of choosing parameter values.

One approach is to formalize the tuning process by defining a scalar "performance index," which is a quantity that is minimized for the "best" choice of parameters. This is "optimal control."

The standard performance indices are of the form

$$J = \int_0^\infty V[x(t), u(t)]dt = \int_0^\infty \left[e^2(t)Q + u^2(t)R\right] dt \,, \tag{5.8}$$

where the general function $V(x, u)$ commonly has a quadratic form and where Q and R are positive constants that balance the relative "costs" of errors $e(t)$ and control efforts $u(t)$. Large Q and small R penalize control errors with little regard to the control effort, while small Q and large R penalize control effort with little regard for control error. Equation (5.8) has an obvious vector-matrix generalization to higher-order systems. Implicit in Eq. (5.8) is the choice of a control signal $r(t)$ and a disturbance $d(t)$. For example, one often assumes a step

function input $r(t) = \theta(t)$, with $d = 0$. One can, alternatively, keep r constant and add a step disturbance for $d(t)$.

As a simple example, we consider the one-dimensional system $\dot{x} = -\alpha x(t) + u(t)$ with proportional control $u(t) = -K_p e(t)$ and reference signal $r(t) = 0$ (Skogestad and Postlethwaite, 1996). The proportional control gives a motion $x(t) = x_0\, e^{-(\alpha+K_p)t}$, which, when inserted into the cost function, Eq. (5.8), gives (with $Q = 1$)

$$J(K_p) = (1 + RK_p^2) \cdot \frac{x_0^2}{2(\alpha + K_p)} \tag{5.9}$$

Minimizing J with respect to K_p gives an "optimal"

$$K_p^* = -\alpha + \sqrt{\alpha^2 + \frac{1}{R}}\,. \tag{5.10}$$

The decay rate of the optimal system is $\alpha' = \sqrt{\alpha^2 + 1/R}$. From this simple example, one can see the good and bad points of optimal control. First of all, optimal control does *not* eliminate the problem of tuning a parameter. Rather, in this case, it has replaced the problem of choosing the proportional gain K_p with that of choosing the weight R. What one gains, however, is a way of making any tradeoffs in the design process more transparent. Here, for example, one is balancing the desire to have small tracking error $e(t)$ with that of wanting minimal control effort $u(t)$. The coefficient R expresses that tradeoff explicitly. For example, in the expression for the optimal K_p in Eq. (5.10), a small R ("cheap control") leads to a large K_p while a large R ("expensive control") leads to a small K_p. In such a trivial example, the result could have been easily foreseen and there would be little point in going through this exercise. When the system is more complex, one often has more intuition into how to set the matrices $\widetilde{Q}$ and $\widetilde{R}$ of the generalization of Eq. (5.8) than to tune the parameters. At any rate, do not confuse "optimal" with "good," for a poor choice of weights will lead to a poor controller.

The above discussion gives short shrift to a rich and beautiful branch of applied mathematics. One can be more ambitious and ask for more than the optimal set of parameters, given a previously chosen control law. Indeed, why not look for the best of all possible control laws? Formulated this way, there is a close connection between optimal control and variational methods. In the simplest cases, these variational methods are just those familiar from classical mechanics. [For an overview of the history of optimal control and its relations with classical problems, see Sussmann and Willems (1997).] For example, we can view the minimization of the performance index J in Eq. (5.8) as a problem belonging to the calculus of variations, where one needs to find the optimal control $u(t)$ that minimizes the functional J subject to the constraint that the equations of motion $\dot{x} = f(x, u)$ are obeyed. [In the example above, there is only one equation of motion, and $f(x, u) = -\alpha x + u$.] One can solve this problem by the method of Lagrange multipliers, minimizing the functional $\mathcal{L} = \int_0^\infty L dt$, where $L(x, \dot{x}, u, \lambda) = \lambda_0 V(x, u) + \lambda(f - \dot{x})$ by free variation of x, u, and the Lagrange multiplier λ.[19] For nth-order systems, there are n Lagrange multiplier functions $\lambda_i(t)$. Setting the variation of J with respect to x, u, and λ equal to zero leads to three sets of Euler-Lagrange equations:

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{x}} - \frac{\partial L}{\partial x} = 0 \tag{5.11a}$$

$$\frac{\partial L}{\partial u} = 0 \tag{5.11b}$$

$$\frac{\partial L}{\partial \lambda} = 0\,. \tag{5.11c}$$

Eqs. (5.11a) and (5.11b) give the equations obeyed by the performance index V; Eq. (5.11c) gives the equations of motion.

For reasons to be discussed below, the Hamiltonian formulation is usually preferred. There, one defines a Hamiltonian $H = L + \lambda\dot{x} = V + \lambda f$. The Euler-Lagrange equations can then be transformed into the control-theory version of Hamilton's equations:

$$\dot{x} = \frac{\partial H}{\partial \lambda} \tag{5.12a}$$

$$\dot{\lambda} = -\frac{\partial H}{\partial x} \tag{5.12b}$$

$$\frac{\partial H}{\partial u} = 0\,. \tag{5.12c}$$

Eqs. (5.12a) and (5.12b) describe the evolution of the "co-states" $\lambda(t)$, which are the equivalent of the conjugate momenta in the classical-mechanics formulation. Note that, in the general case, the state and co-state vectors have n components while the control vector u has r components. In the simple example discussed above, one has $H = Qx^2 + Ru^2 + \lambda(-\alpha x + u)$.

Optimal-control problems often lead to two types of generalizations that are less likely to be familiar to physicists: In classical mechanics, one typically assumes that the starting and ending states and times of the dynamical system are fixed. Only variations satisfying these boundary conditions are considered. Variational problems in control theory are less restrictive. For example, imagine that one wants to move a dynamical system from an initial state x_0 to a final state x_1 as fast as possible (given the dynamical equations $\dot{x} = f(x, u)$). One can formulate this as a problem whose goal is to minimize the unit performance index $V = 1$ over times from t_0 (which can be set to be 0 in an autonomous system) to t_1, where t_1 is to be determined. A recent book by Naidu

[19] Usually, $\lambda_0 = 1$; however, an "abnormal" case with $\lambda_0 = 0$ can arise when both the constraint and its derivative vanish simultaneously (Sontag, 1998).

(2003) catalogs the different types of variational problems commonly encountered in control theory. Note, too, that optimal control problems include both open-loop, feedforward designs, where one asks what input $u(t)$ will best move the system from state 1 to state 2, as well as closed-loop, feedback designs, where one asks what feedback law $u = -Ke$ will best respond to a given type of disturbance.

A second important type of generalization that is commonly encountered concerns constraints placed either on the control variables u or the allowable region of phase space that the state vector x is permitted to enter. These types of problems led Pontryagin and collaborators to generalize the treatment of optimizations, as expressed in the famous "Minimum Principle." (Pontryagin *et al.*, 1964).[20] The main result is that if the control variables $u(t)$ are required to lie within some closed and bounded set in the function space of all possible control laws $U(t)$, then the optimal choice is that control element u that minimizes $H(x, \lambda, u)$. In other words, the condition for a local stationary point of H, Eq. (5.12c), is replaced by the requirement for a global minimum. The Minimum Principle allows one to solve problems that would have no solution within the traditional analytical framework of calculus or the calculus of variations. For example, the derivative of the function $f(x) = x$ is never zero; however, if x is constrained to lie between 0 and 1, f has a well-defined minimum (0). The Minimum Principle allows one to consider such situations and provides a necessary condition for how the control variable u must be selected on the (in general non-trivial) boundary set of its allowed domain. Note that the Minimum Principle is a necessary, but not sufficient requirement for optimal control.

To appreciate the significance of the Minimum Principle, let us consider the simple example of a free particle, obeying Newton's laws and acted on by a controllable force, in one dimension. The equations of motion are $\dot{x_1} = x_2$, $\dot{x_2} = u$, where x_1 is the particle's position, x_2 its velocity, and $u = F/m$ is the external force divided by the particle's mass. We neglect friction. The problem is to bring the particle from an arbitrary initial state $(x_1^{(0)}, x_2^{(0)})$ to the state $(0, 0)$ as fast as possible. Assume that the applied force has limits that imply that $|u(t)| \leq 1$. The Hamiltonian is then given by $H = \lambda_1 x_2 + \lambda_2 u$. Hamilton's equations give, in addition to the equations of motion, $\dot{\lambda_1} = 0$ and $\dot{\lambda_2} = -\lambda_1$. If naively we were to try to use $\partial H/\partial u = 0$ to determine u, we would conclude that all problem variables were identically equal to zero for all time, i.e., that there was no non-trivial solution to our problem. Replacing the local condition by the global requirement that u minimize H, we find that $u = -\text{sign}(\lambda_2)$. Integrating the equations for λ_1 and λ_2, we find $\lambda_2(t) = -c_1 t + c_2$ and conclude that the applied force will always be pegged to its maximum value and that there can be at most one "reversal of thrust" during the problem. [This last conclusion follows because λ_2 makes at most one sign change. Pontryagin *et al.* showed that if an nth-order, linear system is controllable and if all eigenvalues of its system matrix A are real, then the optimum control will have at most $n - 1$ jump discontinuities (Pontryagin *et al.*, 1964).] By analyzing separately the cases where $u = \pm 1$, one finds that for $u = 1$, the system's dynamics lie on one of a family of parabolas described by $x_1 = \frac{1}{2}x_2^2 + c_+$, with c_+ determined by initial conditions. Similarly, for $u = -1$, one finds that $x_1 = -\frac{1}{2}x_2^2 + c_-$.

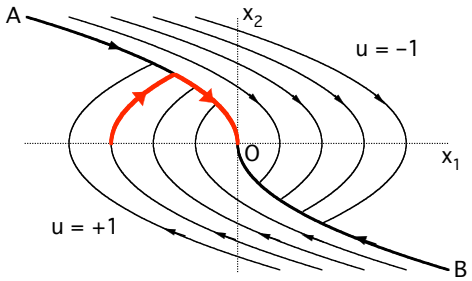

FIG. 20 (Color in online edition) Phase-space plot of dynamical trajectories for a free, frictionless particle acted on by a force $u = \pm 1$. The thrust switches as trajectories reach the curve AOB. The heavy (red) curve shows an example trajectory (Pontryagin *et al.*, 1964).

The full solution is illustrated in the phase-space plot shown in Fig. 20. The heavy curve AOB passing through the end target point (the origin) has special significance. The left-hand segment AO is defined by motion, with $u = -1$ that will end up at the target point (the origin). The right-hand segment, OB is defined by a similar segment with $u = 1$. One thus clearly sees that the optimal motion is determined by evaluating the location of the system in phase space. If the state is below the curve AOB, choose $u = 1$, wait until the system state hits the curve AO, and then impose $u = -1$. If the state is above AOB, follow the reverse recipe. The curve AOB is thus known as the "switching curve" (Pontryagin *et al.*, 1964). In Fig. 20, the red curve denotes the optimal solution for a particle initially at rest. Not surprisingly, the best strategy is to accelerate as fast as possible un-

[20] Pontryagin *et al.* actually formulated their results in terms of $-H$ and thus wrote about the "maximum principle." Modern usage has changed the sign. As an aside, Pontryagin was blinded by an accident when he was 14 and was thereafter tutored by his mother, who had no education in mathematics and read and described the mathematical symbols as they appeared to her (Naidu, 2003). Nonetheless, Pontryagin became one of the leading mathematicians of the 20th century, making major contributions to the theory of topological groups. Later in life, he turned his attention to engineering problems and was asked to solve a problem that arose in the context of the trajectory control of a military aircraft. The basic insights required to solve such problems with constraints on the control variables came after three consecutive, sleepless nights (Gamkrelidze, 1999). The use of the symbol u for the control variable(s) seems also to date from Pontryagin's work, as the word "control" is "*upravlenie*" in Russian (Gamkrelidze, 1999).

til halfway to the goal and then to decelerate as fast as possible the rest of the way. Note that formulating the problem in phase space leads to a local rule for the control – essential for implementing a practical optimal *feedback* controller – based on the geometry of motion in phase space, as determined by the switching curve. This law is nonlinear even though the original problem is linear. This induced nonlinearity is typical of constrained optimization problems and is one reason that they are much harder to solve than unconstrained ones. We return to this and other examples of nonlinearity below, in Sec. VI.

The above discussion merely gives some of the flavor of the types of analysis used. Another important method is *dynamic programming*, introduced by R. Bellman in the 1950s (Naidu, 2003). It is especially well-adapted to discrete problems. Its continuous form is analogous to the Hamilton-Jacobi method of classical mechanics.

Finally, in the above discussions of optimal control, we used quadratic functions $V(x,u)$ in the performance index [e.g., that given in Eq. (5.8)]. Such functions are sensitive to the average deviations of the state and control variables from the desired values. The choice of a quadratic form is motivated largely by the fact that one can then solve for the optimal controller for linear systems analytically. The more recent control literature tends to advocate an alternative that keeps track of the *worst-case* deviations. This more conservative measure of deviations leads to a more robust design, as discussed in Sec. V.E, below.

C. Digital control loops

The original control loops were implemented by analog mechanical, electronic, or hydraulic circuits. At present, they are almost always implemented digitally, for reasons of flexibility: very complicated control algorithms are easily programmed (and reprogrammed!), and performance is easy to assess. It is worth noting, however, that proportional or integral control can be implemented with operational amplifiers costing just a few cents each, with bandwidths easily in the 100-kHz range. Sometimes, such circuits may be the simplest solution.

One subtlety of using a digital control loops is that the input signal must be low-pass filtered to avoid aliasing. To understand aliasing, consider a continuous signal $f(t)$ sampled periodically at times nT_s. We can write the sampled signal as

$$f_{samp}(t) = \sum_{n=-\infty}^{\infty} f(t)\delta(t - nT_s) \; , \tag{5.13}$$

where δ is the Dirac δ-function. Being periodic, the δ-function has a Fourier-series representation,

$$\sum_{n=-\infty}^{\infty} \delta(t - nT_s) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} e^{ik\omega_s t} \; , \tag{5.14}$$

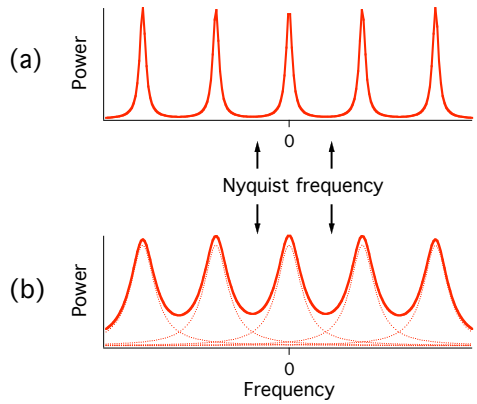

FIG. 21 (Color in online edition) Illustration of the Nyquist sampling theorem in frequency space. (a) The maximum frequency of the power spectrum of the continuous signal is less than the Nyquist frequency (positive and negative frequencies indicated by arrows). The spectrum of the sampled signal accurately represents the continuous signal. (b) The maximum frequency exceeds the Nyquist frequency ω_N, and the aliased spectra overlap, distorting the estimate of the continous spectrum from sampled data. The individual copies of the spectrum of the continuous signal are shown in the dashed lines. The overlap is apparent, particularly near multiples of ω_N.

with $\omega_s = 2\pi/T_s$. Inserting this into Eq. (5.13) and Fourier transforming, one finds

$$f_s(\omega) = \frac{1}{T_s} \sum_{n=-\infty}^{\infty} f(\omega - n\omega_s) \; . \tag{5.15}$$

In other words, the power spectrum of the continuous signal, $|f(\omega)|$ is replicated at frequencies $n\omega_s$ in the power spectrum of the sampled signal $|f_s(\omega)|$. If the highest frequency ω_{max} in f is less than $\omega_s/2 \equiv \omega_N$ (the "Nyquist frequency"), we have the situation in Fig. 21a. If not, we have the situation in Fig. 21b. In the former case, the power spectrum of the sampled signal will be a faithful replication of that of the continuous signal. In the latter case, high-frequency components of f will mix down into the spectrum of f_s (Lewis, 1992).

One can picture this alternatively in the time domain, where the situation leading to aliasing is shown in Fig. 22. The Nyquist sampling theorem derived above states that frequencies higher than $1/2T_s$ will be erroneously read as lower frequencies by the digitizer ("aliasing") and that, in general, one must sample at least twice per period in order to reconstruct the signal present at that period (Franklin *et al.*, 1998). Thus, if one samples with sampling time T_s, one must add an *analog* low-pass filter with cutoff frequency no higher than $1/2T_s$.[21] Since the noise

[21] An alternative is "one-bit delta-sigma analog-digital conversion" (Gershenfeld, 2000), where the signal is converted to a rapidly alternating sequence of 1's and 0's (representing the maximum

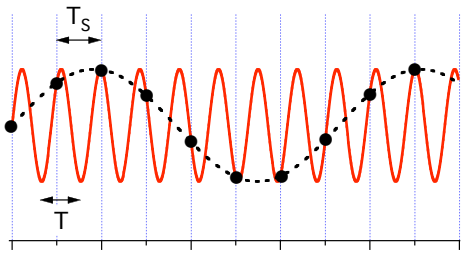

FIG. 22 (Color in online edition) Aliasing of a high frequency signal (solid curve, with period T) produced by sampling with a period T_S that is greater than the Nyquist criterion. The apparent measured points (large dots) are interpreted as coming from a much lower frequency signal (dashed curve).

requirements are stringent, either one must use a higher-order filter (having n'th-order dynamics and made from active components) or a simpler, passive "RC" filter with a somewhat lower cutoff frequency than is required by the sampling theorem. Either way, the dynamics added by the filter becomes part of the feedback loop. Because filters add a phase lag, their effect is destabilizing, as seen in the previous section. Finally, another subtlety is that sometimes it pays to add deliberately a small amount of noise to combat the quantization effects introduced by the A/D converter. See the note in Sec. VI.A for details.

A second issue in digital control is that one must transform an analysis developed for continuous-time dynamical systems into discrete-time dynamical systems. This corresponds to a passage from ordinary differential equations to discrete maps that transform the system state vector $\vec{x}(t_n)$ into $\vec{x}(t_{n+1})$. The whole topic can get rather technical (see the various control theory texts), but the basic ideas are simple. Here, we assume that we know a reasonable analytic approximation to the transfer function $G(s)$ of the system. (See Section V.A, above.)

The next step is to write the model in the time domain. Formally, one would take the inverse Laplace transform. Usually, if the analytic model uses standard components (first- and second-order elements, lags), one can do this by inspection. For example, if we had

$$G(s) = \frac{(1 + s/\omega_0)e^{-\tau s}}{1 + \gamma s/\omega_1 + s^2/\omega_1^2} = \frac{y(s)}{u(s)} \ , \tag{5.16}$$

then we would infer

$$\frac{1}{\omega_1^2}\,\ddot{y} + \frac{\gamma}{\omega_1}\dot{y} + y = u(t-\tau) + \frac{1}{\omega_0}\dot{u}(t-\tau) \ , \tag{5.17}$$

which can be written in standard form

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\omega_1^2 & -\gamma\omega_1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} \omega_1^2/\omega_0 \\ \omega_1^2 \end{pmatrix} \cdot u(t-\tau) \ , \tag{5.18}$$

with $y = x_1$.

The final step is then to discretize the continuous-time system. The obvious algorithm is just to replace derivatives by first-order approximations, with

$$\dot{x} \approx \frac{x_{n+1} - x_n}{T_s} \ , \tag{5.19}$$

for each component of x (or use higher-order approximations for the higher derivatives of y). Usually, such a simple algorithm is sufficient, as long as the sampling time is much shorter (by a factor of at least 10, but preferably 20 to 30) than the fastest dynamics that need to be modeled in the system. For example, the PID law, when discretized using Eq. (5.19) becomes

$$u_n = A_1 u_{n-1} + B_0 e_n + B_1 e_{n-1} + B_2 e_{n-2} \ , \tag{5.20}$$

with $A_1 = 1$, $B_0 = K_p + K_i T_s/2 + K_d/T_s$, $B_1 = -K_p + K_i T_s/2 - 2K_d/T_s$, and $B_2 = K_d/T_s$. Here K_p, K_i, and K_d are the proportional, integral, and derivative terms, T_s is the sampling time, and u_n and e_n are the actuator and error signals at time nT_s. Equation (5.20) has the advantage that no integral explicitly appears. It is derived by taking the discrete differential of the straightforward expression for u_n in terms of an integral over the previous error signals. Equation (5.20) also has the advantage that it expresses the next value of the actuator signal (u_n) in terms of its previous value (u_{n-1}). This is useful if one wants to go back and forth from a "manual mode" to closed-loop control without large transients in the actuator signal. Note that e_n does appear in Eq. 5.20. We assume that we use the current error signal in calculating the current actuator response, implying that the calculation time is much less than the sampling time. If not, the delay should be taken into account.

In deriving expressions such as Eq. (5.20), it is often convenient to go directly from the transfer function to the discrete dynamics, without writing down the continuous-time equations explicitly. One can do this by looking at the Laplace transform of sampled signals. Assume that a continuous signal $f(t)$ is sampled at times nT_s, for $n = 0, 1, ...$ As above, the sampled signal is then

$$f_s(t) = \sum_{n=0}^{\infty} f(t)\delta(t - nT_s) \ . \tag{5.21}$$

and minimum voltage limits). If the voltage input is, say, 0.5, then the output will have an equal number of 1's and 0's. If the voltage input is 0.75, there will be 3 times as many 1's as 0's, etc. The cycle time for the alternation is faster (by a factor of 64, or even more) than the desired ultimate sampling time. One then digitally low-pass filters this one-bit signal to create the slower, higher resolution final output value. Because of the oversampling, a simple low-pass filter suffices to prevent aliasing. The disadvantage of delta-sigma conversion is a relatively long "latency" time – the lag between the signal and the digital output can be 10-100 times the sampling interval. Still, if the lag is small compared to the system timescales, this kind of conversion may be the simplest option. Note that delta-sigma and other A/D conversion schemes are actually helped by adding a small amount of noise. See Footnote 28, below.

The Laplace transform of $f_s(t)$ is

$$\mathcal{L}[f_s(t)] = \sum_{n=0}^{\infty} f(nT_s)e^{-s(nT_s)} \tag{5.22}$$

If we introduce $z \equiv e^{sT_s}$ and define $f_n \equiv f(nT_s)$, then Eq. (5.22) leads to the "z-transform," defined by

$$\mathcal{Z}[f] = \sum_{n=0}^{\infty} f_n z^{-n} \; . \tag{5.23}$$

For example, by direct substitution into Eq. (5.23), one can see that the z-transform of the step function $\theta(t)$ (0 for $t < 0$, 1 for $t \geq 0$) is $\frac{z}{z-1}$. Similarly, $\mathcal{Z}[f(t+T_s)] = z\mathcal{Z}[f(t)] - zf_0$. Thus, with a zero initial condition for f, the z-transform shows that shifting by T_s means multiplying the transform by z. This is analogous to the Laplace transform of a derivative (multiply by s) and implies that taking the z-transform of a difference equation allows it to be solved by algebraic manipulation, in exact analogy to the way a Laplace transform can be used to solve a differential equation.

The transformation $z = \mathrm{e}^{sT_s}$ relates the complex s-plane to the complex z-plane. Note that LHP poles (i.e., ones with $\mathrm{Re}(s) < 0$) are mapped to the interior of the unit circle. The frequency response in the z-plane is obtained by substituting $z = \mathrm{e}^{i\omega T_s}$. Frequencies higher than the Nyquist frequency of $\omega_s/2 = \pi/T_s$ are then mapped on top of lower frequencies, in accordance with the aliasing phenomenon depicted in Figs. 21 and 22.

We seek a way of discretizing the transfer function $K(s)$ of a continuous controller. In principle, this could be done using the relation $z = \mathrm{e}^{sT_s}$, but this would lead to infinite-order difference equations. One is then led to try low-order approximate relations between s and z. The first-order expansion of $z^{-1} = \mathrm{e}^{-sT_s} \approx 1 - sT_s$ leads to

$$s \to \frac{1 - z^{-1}}{T_s} \; , \tag{5.24}$$

Since z^{-1} means "delay by T_s," we see that this is a transformation of Eq. (5.19).

If the sampling time cannot be fast enough that simple discretization is accurate, then one has to begin to worry about more sophisticated algorithms. One could imagine expanding the exponential to second order, but it turns out that using a first-order Padé approximation is better. Thus, one sets $z^{-1} \approx \frac{1-sT_s/2}{1+sT_s/2}$, which is accurate to second order. This gives "Tustin's transformation" (Dutton *et al.*, 1997)

$$s \to \frac{2}{T_s}\left(\frac{1 - z^{-1}}{1 + z^{-1}}\right) \; . \tag{5.25}$$

Equation (5.25) is equivalent to using the trapezoidal rule for integrating forward the system dynamics, while Eq. (5.24) corresponds to using the rectangular rule ((Lewis, 1992), Ch. 5). One advantage of Tustin's transformation is that it, too, maps $\mathrm{Re}(s) < 0$ to the interior of the unit circle, implying that if the Laplace transform of a continuous function is stable ($\mathrm{Re}(s < 0)$), then its z-transform will also be stable. (Iterating a linear function with magnitude greater than one leads to instability.) The direct, first-order substitution, Eq. (5.24), does not have this property. Analogously, in the numerical integration of differential equations, too coarse a time step can lead to numerical instability if one uses an explicit representation of time derivatives [Eq. (5.24)]. If, on the other hand, one uses an implicit representation of the derivative [Eq. (5.25)], then stability is guaranteed, although accuracy will suffer if T_s is too big.

Finally, one can account for the effect of using a zero-order-hold (ZOH) to produce digitization (as in Fig. 13) by considering its Laplace transform. Let

$$\begin{aligned} f_{ZOH}(t) &= f(0) \qquad 0 < t < T_s \\ &= f(T_s) \qquad T_s < t < 2T_s \\ &\vdots \end{aligned} \tag{5.26}$$

Then

$$\begin{aligned} \mathcal{L}[f_{ZOH}] &= f(0)\int_0^{T_s} \mathrm{e}^{-st}dt + f(T_s)\int_{T_s}^{2T_s} \mathrm{e}^{-st}dt + \ldots \\ &= \left(\frac{1 - \mathrm{e}^{-sT_s}}{s}\right)\mathcal{Z}(f) \; . \end{aligned} \tag{5.27}$$

The effect of the ZOH is to introduce an extra factor in front of the ordinary z-transform. Thus, a common "recipe" for translating a continuous controller $K(s)$ to a discrete equivalent $D(z)$ is

$$D(z) = (1 - z^{-1})\frac{K[s(z)]}{s(z)} \; , \tag{5.28}$$

with Eq. (5.25) used to transform s to z. Because the discretization leads to a large amount of algebra, "canned" routines in programs such as Matlab are useful. (See Sec. V.C.2, below.)

Once the approximate digital controller $D(z)$ has been worked out, one can generate the appropriate difference equation by writing (Dutton *et al.*, 1997):

$$D(z) = \frac{u(z)}{e(z)} = \frac{B_0 + B_1 z^{-1} + B_2 z^{-2} + \ldots}{1 - A_1 z^{-1} - A_2 z^{-2} - \ldots} \tag{5.29}$$

In Eq. (5.29), the transfer function is between the error signal $e(z)$ input and the control variable (actuator) $u(z)$ output. (Of course, the transfer function $G(z)$ of the system itself would be between the control variable input u and the system output y.) Recalling that z^{-1} has the interpretation of "delay by T_s," we may rewrite Eq. (5.29) as a discrete difference relation (known as an "infinite impulse response," or IIR filter in the signal-processing literature), Oppenheim *et al.* (1992)),

$$u_n = A_1 u_{n-1} + A_2 u_{n-2} + \ldots B_0 e_n + B_1 e_{n-1} + B_2 e_{n-2} + \ldots \tag{5.30}$$

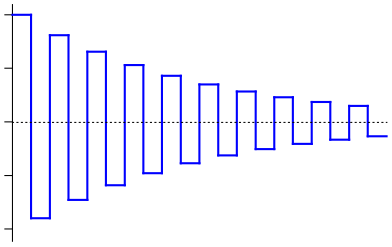

FIG. 23 (Color in online edition) Time series for a discrete dynamical system showing a "ringing pole." Generated from $y_{n+1} = (1 - \lambda)y_n$, with $\lambda = 1.9$.

which generalizes the result of Eq. (5.20).

One minor pitfall to avoid is that discrete systems can introduce "ringing poles." For example, consider the simple difference equation

$$y_{n+1} = (1 - \lambda)\, y_n \ . \tag{5.31}$$

This, obviously, is the discrete analog of a first-order system $\dot{y}(t) = -\lambda y(t)$ and gives similar stable behavior for $0 < \lambda < 1$. Now consider the range $1 < \lambda < 2$, where the system continues to be stable but oscillates violently with each time step, albeit with a decaying envelope. (Fig. 23.) Such a ringing pole occurs because of an effective aliasing — beating — of the decay rate against the sampling rate. In any case, one usually wants to avoid such a response in a discrete controller. Although nominally stable, the stability is achieved by wild cycling of the actuator between large positive and negative values.

We note that the unpleasant situation illustrated in Fig. 23 can also appear in the dynamics. Even if all the measured outputs y_n and all the control inputs u_n seem reasonable, the system may be ill-behaved between the sampling time points. One can guard against this possibility by sampling the system dynamics more rapidly in tests. If that is not possible, at the very least one should model the closed-loop system dynamics on a computer, using a shorter time interval for the internal simulation.

We can summarize the steps needed to create a digital control loop as follows:

1) Determine (empirically) the system transfer function $G(s)$.

2) Construct a feedback law $K(s)$ that gives the desired closed-loop dynamics for $T(s) = KG/(1 + KG)$, possibly taking into account sensor dynamics $H(s)$, as well.

3) Use Eq. (5.28) and either the direct method, Tustin's transformation, or any of the alternate methods discussed in control-theory books, to transform $s \to z$, thereby converting the continuous controller $K(s)$ to a discrete controller $D(z)$.

4) Deduce the difference-equation corresponding to $D(z)$ and program this on an appropriate digital device, for example a computer, microcontroller, digital signal processor (DSP), or programmable logic device (PLD).

The reader is cautioned, however, to consult a more detailed reference such as Lewis (1992) before actually implementing a digital control loop in an experiment. There are a number of practical issues that we have had to skip over. To cite just one of these, with a finite word size (integer arithmetic), a high-order difference relation such as Eq. 5.30 is prone to numerical inaccuracies. Its accuracy can be improved by rewriting the relation using a partial-fraction expansion.

At present, digital loops may be implemented on any of the platforms mentioned in (4). Probably the most popular way is to use a computer and commercial data acquisition board. Although such boards often run at 100 kHz or even faster, they cannot be used for control loops of anywhere near that bandwidth. Modern operating systems are multi-tasking, which means that they cannot be counted on to provide reliable real-time response. In practice, timing variability limits control loops based on such computers to kHz rates at best. Microcontrollers are a cheap alternative. One downloads a program (usually in C or Assembler, occasionally in a simpler, more user-friendly language) that runs in a standalone mode on the microcontroller. Because the microcontroller does nothing else, its timing is reliable. More important, it will work even if (when!) the host computer crashes. DSPs offer similar advantages, with much higher performance (and price). Microcontrollers can execute control loops at rates up to 10 kHz, while DSPs can work up to 1 MHz. (These rates are increasing steadily as more sophisticated technologies are introduced.) DSPs also offer more sophisticated possibilities for asynchronous transfer of data to and from the host computer. In general, one would consider microcontrollers for simpler, slower loops and DSP for more high-performance needs.

There are many options for microcontrollers and DSPs. Often, there is a tradeoff between inexpensive high-performance hardware that is difficult (and thus, expensive) to program (and maintain) and expensive DSP boxes that are somewhat lower performance but have an easy-to-use programming environment. In my laboratory, we have had some success with the latter approach.[22] We can implement simple control problems such as PID loops up to 500 kHz and complicated control problems such as a scanning-probe microscope controller at 20 kHz. The use of a high-level language is important in that it allows a wide range of students, from undergraduates on up, to easily modify source code as needed without making a large investment in understanding complicated software.

Finally, programmable logic devices (PLDs) are another option, albeit little known in the physics community. They grow out of specific "hard-wired" solutions for particular problems. Think, for example, of the circuitry in a dishwasher, dryer, or other common appliance.

---

[22] We have used the DSP boxes of Jäger Computergesteurt Messtechnik, GmbH., Rheinstraße 2-4 64653 Lorsch, Germany (www.adwin.de). We made no attempt to survey the market systematically, and there may well be better alternatives.

PLDs are a kind of programmable version of these. Essentially, they are arrays of logic gates, flip flops, etc., whose interconnections may be programmed. They replace custom hardware solutions to implement an essentially arbitrary digital circuit. They excel in situations where many operations can be performed in parallel, resulting in a tremendous speed advantage (of order 1000 in many cases). PLDs come in a bewildering variety of families, with varying logic-gate densities and varying ease of programmability. One popular family is the field programmable gate array (FPGA). In physics, these have been used to make trigger units in high-energy experiments that must deal with a tremendous amount of data in very short times. For example, in the D0 detector at Fermilab, a set of boards containing 100 FPGA chips evaluates $\approx 10^7$ events/sec., identifying about 1000 events/sec. as potentially "interesting." Subsequent trigger circuits then reduce the event rate to about 50 Hz (Borcherding *et al.*, 1999). Until recently, the "learning curve" in programming such devices was steep enough that they made sense only in projects large enough to justify the services of a specialist programmer. (The language and logic of FPGA devices was different enough from ordinary programs to require extensive training for such devices.) Recently, FPGAs that are programmable in ordinary, high-level languages (e.g., Labview) have become available.[23] FPGAs are so powerful that they can implement almost any conventional digital or (low-power) analog electronic circuit, and many commercial devices are now based on such circuits. Despite the D0 example above, the full impact of having "hardware you can program" has not yet been appreciated by the physics community.

In looking over the above "recipes" for making a digital controller, one might be tempted to think that it is simpler to bypass the first two steps and work in the discrete domain from the beginning. For example, any discrete, linear control law will have the form of Eq. (5.20), with different values for the A, B, and C coefficients (and with the coefficient in front of u_n not necessarily 1 and with perhaps other terms $u_{n-1}, \ldots$ and $e_{n-2} \ldots$ as well.) Indeed, our approach to digital feedback is known as "design by emulation," and it works well as long as the sampling rate is high enough relative to system dynamics. Emulation has the virtue that it uses the intuitive loop-shaping ideas discussed above. Still, direct digital design has fewer steps and potentially higher performance. We do not have space to discuss direct methods, except to say that virtually every continuous-time technique has a discrete-time analog. For example, the discrete analog of the linear system given by Eq. (2.2) is

$$\begin{aligned} x_{n+1} &= \widetilde{A}' x_n + \widetilde{B}' u_n \\ y_{n+1} &= \widetilde{C} x_{n+1} \ , \end{aligned} \tag{5.32}$$

with $\widetilde{A}' = \mathrm{e}^{\widetilde{A} T_s}$ and $\widetilde{B}' = \widetilde{A}^{-1}(\widetilde{A}' - \widetilde{I})\widetilde{B}$. In the continuous-time case, the eigenvalues of $\widetilde{A}$ needed to have negative real part for stability. Here, the analogous condition is that the eigenvalues of $\widetilde{A}'$ have magnitude less than one. Other aspects carry over as well (e.g., Laplace transform to z-transform). But when faced with a choice between reading up on fancy direct-digital-design techniques and buying a better digitizer, buy the board. The time you save will almost certainly be worth more than the price of the more expensive hardware.

1. Case study: Vibration isolation of an atom interferometer

We mention briefly a nice example of an implementation of a feedback loop that illustrates most of the topics discussed in this section. The application is the control system of a vibration isolation stage for an atom interferometer (Hensley *et al.*, 1999). The goal is to decouple the instrument from random low-frequency vibrations due to machinery, micro-seismic disturbances, etc. The conventional passive strategy is to mount the instrument on a mechanical spring system (often an air spring, but here a conventional wire spring). The idea is that if the driving vibrations have frequencies much higher than the resonant frequency of the mass-spring system, their amplitude will be damped. The lower the frequency of the external vibrations, the lower the required resonant frequency of the mass-spring system. But practical mass-spring systems have resonant frequencies of roughly 1 Hz for vertical vibrations. Lowering the resonant frequency requires softer springs, which stretch large distances under the influence of gravity. In order to lower the resonant frequency still further, one can use feedback to alter the dynamics. (Note that in the first case study, we decreased characteristic times; here we will increase them.)

The system has an equation of motion

$$\ddot{y} + 2\zeta_0\omega_0(\dot{y} - \dot{y}_g) + \omega_0^2(y - y_g) = u(t) \ , \tag{5.33}$$

where $\omega_0^2 = k/m$ is the natural resonance frequency of the undamped system, $y_g(t)$ and $\dot{y}_g(t)$ are the position and velocity of the ground, $u(t)$ is the actuator signal, and ζ_0 is the dimensionless damping, as in Eq. (2.5). Indeed Eqs. (5.33) and (2.5) differ only in that the damping is now proportional to the difference between the mass and ground velocity, and similarly for the restoring force.

The control loop design uses an accelerometer to sense unwanted vibrations and applies a PI control $[u(t) = -K_p\ddot{y}(t) - 2\omega_0 K_i \dot{y}(t)]$ to lower the effective resonant frequency of the mass-spring design. Notice that measuring the acceleration is better than measuring the position for noise rejection. (You integrate once, as opposed to differentiating twice.) The closed-loop transfer function is

$$T(s) = \frac{2s\zeta_1\omega_1 + \omega_1^2}{s^2 + 2s(\zeta_1 + K_i')\omega_1 + \omega_1^2} \ , \tag{5.34}$$

[23] Labview, FPGA module, National Instruments (www.ni.com).

where $\omega_1 = \omega_0/\sqrt{K_p+1}$, $\zeta_1 = \zeta_0/\sqrt{K_p+1}$, and $K_i' = K_i/\sqrt{K_p+1}$. The closed-loop resonance frequency, ω_1, is set nearly 50 times lower than ω_0, and the damping, $\approx K_i'$, is set near one, the critical-damping value.[24]

What makes this example particularly interesting from our point of view is that the actual feedback law used was more complicated than the simple PI law discussed above. Because the accelerometer had upper and lower limits to its bandwidth, the feedback gain needs to roll off near these limits in order to reduce noise and maintain stability. The limits are implemented by a series of lag compensators [see Eq. (5.6)] and low- and high-pass filters, which collectively shape the frequency response to counteract problems induced by the finite sensor bandwidth. The entire law is implemented digitally using Tustin's method [Eq. (5.25)]. See Hensley *et al.* (1999) for details. (If you have understood the discussion up to this point, the paper will be straightforward.) Finally, while vibration isolation is enhanced by decreasing the "Q" of an oscillator, other applications use feedback to *increase* Q. An example is "active Q control," where reduced oscillator damping increases the sensitivity for detecting small forces (Tamayo *et al.*, 2000).

2. Commercial tools

The design and analysis of feedback systems is made much easier by computer simulation. The engineering community leans heavily on commercial software products, especially Matlab.[25] All of the steps and laws discussed here, and many more, are available on the Matlab platform, although unfortunately, one must buy several modules ("toolboxes") to fully implement them. There are also some freely available packages of control routines (e.g., Scilab, Octave, and SLICOT[26]), which, however, do not seem to be as widely adopted. My own feeling is that physicists can usually get by without such specialized tools – the basic approaches we describe here are simple and easy to implement without investing in learning complicated software packages. On the other hand, for more complex problems or for obtaining the absolute highest performance from a given hardware, these software tools can be essential. Physicists should also note that most engineering departments will already have such software. One useful feature of commercial products is that they often can generate low-level code that can be downloaded to one's hardware device to run the control loop. Usually, a special add-on module is required for each type and model of platform (DSP, FPGA, etc.).

[24] Eq. 3 of Hensley *et al.* (1999) has a misprint.

[25] Matlab is a registered trademark of The Mathworks, Inc. 3 Apple Hill Dr., Natick, MA 01760-2098 (USA). See <http://www.mathworks.com>.

[26] Scilab is an open-source program that is a fairly close imitation of Matlab. (http://scilabsoft.inria.fr/). Octave is a high-level programming language that implements many of the Matlab commands, with graphics implemented in GNU Plot. (http://www.octave.org/). SLICOT is a set of FORTRAN 77 routines, callable from a wide variety of programming environments (http://www.win.tue.nl/niconet/NIC2/slicot.html).

D. Measurement noise and the Kalman filter

At the end of Section III, we briefly discussed the effect of environmental disturbances and sensor noise on control loops. In Fig. 7 and Eq. (3.13), we see the effects of noise on the variable $y(s)$ that is being controlled. To understand better the effects of such noise, we again consider the simple case of proportional feedback control of a first-order system, with no significant sensor dynamics. In other words, we set $K(s) = K_p$, $G(s) = \frac{1}{1+s/\omega_0}$, and $H(s) = 1$. We also consider the case of a regulator, where $r(s) = 0$. (We are trying to keep $y(s) = 0$.) Then, for $K_p \gg 1$,

$$y(s) \approx -\frac{1}{1+s/\omega'} \cdot \xi(s) + \frac{s/\omega'}{1+s/\omega'} \cdot d(s) \ , \qquad (5.35)$$

with $\omega' = \omega_0(1+K_p) \gg \omega_0$. Thus, low-frequency ($\omega \ll \omega'$) sensor noise and high-frequency environmental disturbances cause undesired fluctuations in y.

These effects are illustrated in Fig. 24a-c, which shows the variable $x(t)$ in a discrete-time simulation of a low-pass filter. The explicit discrete equations are

$$\begin{aligned} x_{n+1} &= \phi x_n + u_n + d_n \\ y_{n+1} &= x_{n+1} + \xi_{n+1} \ . \end{aligned} \qquad (5.36)$$

with $\phi = 1-\omega_0 T_s$ and $u_n = K_p\omega_0 T_s(r_n - y_n)$, with r_n the control signal. Here, we have reintroduced the distinction between the environmental variable being controlled (for example, a temperature), x, and the sensor reading of that variable (for example, a thermistor impedance), y. For simplicity, we have taken $y = x$, but in general the units are different (°C vs. ohms), and the sensor may have its own dynamics. The discrete dynamics are evaluated at times nT_s. The proportional feedback gain is K_p (dimensionless). The environmental noise is given by the stochastic variable d_n and the sensor noise by ξ_n. Both of these are Gaussian random variables, with zero mean and variances $\langle d^2\rangle \equiv d^2$ and $\langle \xi^2\rangle \equiv \xi^2$, respectively.[27]

In Fig. 24a, we plot a time series x_n for the open-loop system without feedback. ($K_p = r_n = 0$.) The time constant of the system low-pass filters the white noise due to the environment ($d^2 = 0.01$). In (b), we add feedback ($K_p = 10$). The plot of x_n shows how the proportional feedback increases the cutoff frequency of

[27] Caution: the variance of the noise terms d_n and ξ_n increases linearly in time. In other words, the associated standard deviation, which would be used by a random-number generator in a numerical simulation of Eq. (5.36), is proportional to $\sqrt{T_s}$.

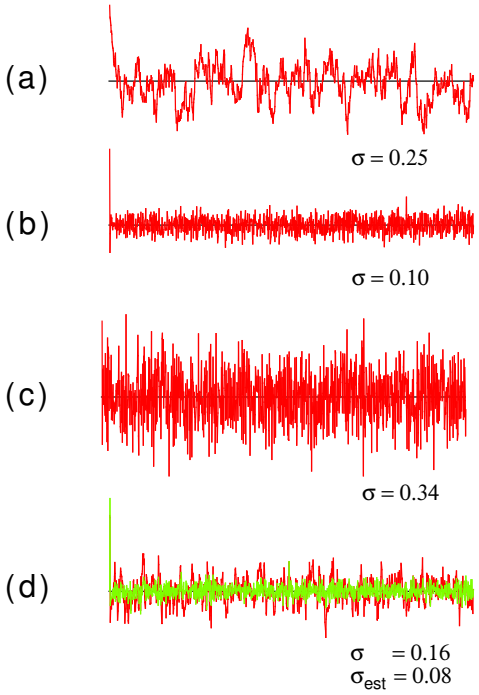

FIG. 24 (Color in online edition) Control of a first-order system in the presence of noise. (a) Open-loop fluctuations of the controlled variable in response to environmental noise; no sensor noise is present; (b) closed-loop proportional control; (c) sensor noise added; (d) Kalman filter added; actual system state and the Kalman estimate, which has smaller fluctuations, are both shown. In (a-d), σ is the standard deviation of the measured signal.

the low-pass filtering ($\omega' \gg \omega_0$). The reduced standard deviation illustrates how feedback controls the excursions of x from the setpoint.

In (c), we add sensor noise ($\xi^2 = 0.1$). Because the control loop cannot distinguish between the desired control signal r_n and the undesired sensor noise ξ_n, the regulation is noticeably worse. This degradation in performance would be aggravated were one to use derivative control, which amplifies high-frequency noise.

Sensor noise can thus strongly degrade the performance of a control loop. One might try to limit the sensor noise by reducing the sensor's bandwidth — for example, by adding a low-pass filter to the sensor circuit. This works as long as the sensor bandwidth remains much greater than that of the feedback. If not, the extra low-pass filter adds a phase lag that degrades the stability margins, forcing one to lower the feedback gain. If the sensor has significant noise within the feedback bandwidth, straightforward frequency filtering will not be sufficient.

About forty years ago, Kalman (1960) suggested a clever strategy that is a variant of the observer discussed in Sec. III.C. There, one used the dynamics to evolve an estimate of the internal state forward in time, corrected by a term proportional to the difference between the actual observed variable and the prediction. Here, the strategy is similar, but because both the dynamics and the observations are subject to noise (disturbances and sensor noise), one wants to blend the two in a way that minimizes the overall uncertainty. Like the observer, though, the key idea is to use the system dynamics to supplement what one observes about the system, in order to estimate the complete internal state of the dynamics.

We begin by noting that there are three related quantities that describe the internal state of a dynamical system at time $n+1$:

true state	$x_{n+1} = \phi x_n + u_n + d_n$
predicted st.	$\hat{x}_{n+1} = \phi \hat{x}_{n\|n} + u_n$
best estimate	$\hat{x}_{n+1\|n+1} = (1-K)\hat{x}_{n+1} + K y_{n+1}$

and two quantities that describe the measurements:

actual measurement	$y_{n+1} = x_{n+1} + \xi_{n+1}$
predicted meas.	$\hat{y}_{n+1} = \hat{x}_{n+1}$

Keeping these straight is half the battle. In the last state-vector item, the "Kalman gain" K is chosen to minimize the overall uncertainty in $\hat{x}_{n+1}$ by blending, with proper weight, the two pieces of information we have available: the best estimate based on a knowledge of previous measurements and of the system dynamics, and the actual measurement. The notation $\hat{x}_{n+1|n+1}$ means that the estimate of x_{n+1} uses observations up to the time $n+1$. By contrast, the prediction $\hat{x}_{n+1}$ uses observations only up to time n. Note that the true state x is forever unknown and that usually there would be fewer measurements y than state-vector components for x.

Our goal is to derive the optimal value for the Kalman gain K. To proceed, we write $\hat{x}_{n+1|n+1}$ as

$$\hat{x}_{n+1|n+1} = \phi \hat{x}_{n|n} + u_n + K(y_{n+1} - \hat{y}_{n+1}) \; , \qquad (5.37)$$

in the standard, "observer" form [Eq. (3.21)] using Eq. (5.36). Defining the estimation error $e_n = x_n - \hat{x}_{n|n}$, we can easily show using Eqs. (5.36) and (5.37) that

$$e_{n+1} = (1-K)\left[\phi e_n + d_n\right] - K\xi_{n+1} \; . \qquad (5.38)$$

Note how Eq. (5.38), in the absence of the noise terms d_n and ξ_{n+1}, essentially reduces to our previous equation for estimator error, Eq. (3.22).

The crucial step is then to choose the Kalman gain K in the "best" way. We do this by minimizing the expected value of the error variance at time $n+1$. This error variance is just

$$\langle e_{n+1}^2 \rangle = (1-K)^2 \left[\phi^2 \langle e_n^2 \rangle + d^2\right] + K^2 \xi^2 \; . \qquad (5.39)$$

In Eq. (5.39), the cross terms are zero because the different noise and error signals are uncorrelated with each other: $\langle d\xi \rangle = \langle ed \rangle = \langle e\xi \rangle = 0$. Differentiating Eq. (5.39) with respect to K, we minimize the error at time $n+1$ by taking

$$K_{n+1} = \frac{\phi^2 \langle e_n^2 \rangle + d^2}{\phi^2 \langle e_n^2 \rangle + d^2 + \xi^2} \; . \qquad (5.40)$$

We put the index $n+1$ on K because, in general, the minimization must be done at each time step, and the dynamics, as well as the noise statistics, may have explicit time dependence. Equation (5.40) implies that if d^2 is big and ξ^2 small, $K_{n+1} \approx 1$: one should trust the measurement. Alternatively, if sensor noise dominates, one should weight the dynamical predictions more heavily by taking $K_{n+1} \approx 0$. Equation (5.40) gives the optimal balance between the two terms. If the coefficients of the dynamical equations do not depend on time (n), there will be a time-independent optimum mix K^*. (Even with stationary dynamics, poorly known initial conditions will tend to make the Kalman filter initially put more weight on the observations. Subsequently, the optimal K's will decrease to K^*.)

In Fig. 24d, the optimal K^* is found to be ≈ 0.2 after a transient lasting 10-20 time steps, implying that afterwards, relatively little weight is placed on new measurements. This is not surprising, given that the standard deviation of the sensor noise is more than three times that of the disturbances. Still, no matter how noisy the sensor, it always pays to have $K^* > 0$ in that one needs at least some contact with the measured system state. Note in (d) the marked improvement in regulation compared to (c), showing that many of the problems created by sensor noise have been mitigated. The actual measurement signal in (d) is similar to the trace in (c), showing that we have "filtered" out the measurement noise in constructing the estimate $\hat{x}$. Note, too, how the estimate $\hat{x}$ has fewer fluctuations than the actual state x. In sensor applications, such as the Michelson interferometer discussed above, where the output is the feedback actuator signal, this is a significant advantage.

One important point is how to estimate d^2 and ξ^2. If the sensor can be separated from the system being measured, one can easily establish its noise properties, which are usually close to Gaussian. (A typical procedure is to short the sensor input and measure the power spectrum of the sensor output.) Disturbances can be more problematic. In the most straightforward case, they are due to noise from the output stages of the amplifier that powers the system's actuator. But more commonly, the most important disturbances are due to environmental perturbations. For example, in a scanning tunneling microscope (STM), they would be due to the roughness of the surface, and one would need some knowledge about the typical surface to fix the roughness scale. It matters less that the height profile of the surface show Gaussian fluctuations than that there is a typical scale. If so, then the Kalman algorithm, which basically assumes that the only relevant information is the second moment, will usually be reasonable. If the fluctuations deviate in a serious way – e.g., if they show a power-law distribution – then the algorithm may be inaccurate. But note that true power-law behavior is probably fairly rare. In the case of an STM scanning a surface, for example, scans are over a finite area, and for fixed scanning area, there would be a typical length scale for surface fluctuations. (This length could diverge with the overall scan size, but so weakly in practice that one could ignore the effect.)

The discussion so far has been for a scalar case with only one degree of freedom. The general case is handled similarly, with exact vector-matrix analogs for each equation discussed above. The algebra becomes more complicated, but all the basic reasoning remains the same (Dutton *et al.*, 1997). (The generalization of Eq. (5.39) leads to a "matrix Riccati equation," which requires some care in its solution.)

The Kalman filter is similar to the Wiener filter [compare Eq. (5.40) with Eq. (13.3.6) of (Press *et al.*, 1993)], which solves the problem of extracting a signal that is convoluted with an instrumental response and by sensor noise. One difference, which is one of the most attractive features of the Kalman filter, is its recursive structure: data at time $n+1$ are calculated in terms of data at time n. This means that one only has to keep track of the most recent data, rather than an entire measurement set. To understand this advantage better, consider the computation of the average of $n+1$ measurements

$$\langle x \rangle_{n+1} \equiv \frac{1}{n+1} \sum_{i}^{n+1} x_i \; . \tag{5.41}$$

In computing the average in Eq. (5.41), one normally sums the $n+1$ terms all at one. Alternatively, Eq. (5.41) can be rewritten recursively as

$$\langle x \rangle_{n+1} = \frac{n}{n+1} \langle x \rangle_n + \frac{1}{n+1} x_{n+1} \tag{5.42}$$

which is the strategy used in formulating the Kalman filter. The Kalman filter takes advantage of the fact that data are collected periodically and that the driving noise has no correlation from one step to another. Its main limitations are that the calculation assumes one knows the underlying system and that that system is linear. If the model is slightly off, the feedback nature of the filter usually does an adequate job in compensating, but too much inaccuracy in the underlying dynamical model will cause the filter to produce unreliable estimates of the state x. Perhaps the most important idea, though, is that the Kalman filter shows explicitly the advantage of tracking the system's dynamics using an internal model.

Since the Kalman filter generalizes the observer discussed earlier in Sec. III.C, one recovers a simple observer if the sensor noise is zero. Whether generated by an observer or a Kalman filter, the estimate of the state vector is used to generate an error signal in the feedback loop. This loop can then be designed using the optimal control theory described in Sec. V.B.3, which in this context is called LQG control: Linear model, integral Quadratic performance index, Gaussian noise process (Skogestad and Postlethwaite, 1996). The "separation theorem" proves that the dynamical performance of the feedback loop based on the optimal estimates provided by the Kalman filter have the same properties (poles, etc.) as they would if the internal states $x(t)$ of the model were

directly observable. Thus, one can separate the observer problem from the control problem.

Like most of the techniques discussed here, the Kalman filter is designed to deal with linear systems. Linearity actually enters in two places: in the dynamical equation for the internal state $x(t)$ and in the output relation between x and y. If one or either of these relations is nonlinear, a simple strategy (the "extended Kalman filter," or EKF) is to linearize about the current operating point. Thus, if the state $x_{n+1} = f(x_n)$ for some nonlinear function $f(x)$, one would update x using $\phi_n = \frac{df}{dx}$, with the derivative evaluated at the current state x_n. The EKF works well in slightly nonlinear situations, but it implicitly assumes that Gaussian statistics are preserved under the nonlinear dynamics. For strong nonlinearities, the analysis is much more difficult, and the search for approximate solution methods is an active area of research (Evensen, 2003; Eyink *et al.*, 2004).

Although our discussion here has been rather elementary, we hope to have motivated the reader to explore a strategy that, despite occasional attempts at popularization over the years (Cooper, 1986; Gershenfeld, 1999), remains little used by physicists.

E. Robust control

Control theory underwent something of an identity crisis during the 1970s. Although the sophisticated methods described in the above sections can give feedback schemes that work very well on paper, the results, in practice, were often disappointing. This led many practical engineers (and physicists) to conclude that it was not worth learning fancy techniques and that the tried-and-true PID controller was just about all one needed to know. Indeed, the consensus is that the academic research on control theory from 1960 to about 1980 had "negligible effect" on industrial control practice. [See the introduction to Morari and Zafirioiu (1989), as well as Leigh (2004).]

The root of the problem was that the schemes presented so far implicitly assume that the system itself and the types of inputs and disturbances it is subject to are well-known. But in practice, models of a system have uncertainties – parameters differ from setup to setup, high-frequency modes may be neglected, components age, and so on. Controllers that are optimized (e.g., in the sense of Sec. V.B.3) for one system may fail miserably on the slightly different systems encountered in practice.

The challenge of finding effective control laws in the face of such uncertainties led to two approaches, beginning in earnest the late 1970s and early 1980s. One of these, adaptive control, tries to reduce uncertainty by learning more about the system while it is under control. We discuss this briefly in Sec. VIII, below. The other approach is that of "robust control," where one tries to come up with control laws that take into account this uncertainty. Here, we give some of the basic ideas and flavor of this approach, following mainly Doyle *et al.* (1992); Özbay (2000); Skogestad and Postlethwaite (1996). In Section V.E.1, we show that in any control system, one implicitly assumes a model of the system being controlled. In Section V.E.2, we give a useful way of quantifying one's uncertainty about a system. In Section V.E.3, we show how to test whether, given a nominal system, with uncertainty, the system is stable. This leads to the notion of "robust stability." In Section V.E.4, we give an analogous method for insuring "robust performance" in the face of uncertainty about control inputs and disturbances. Finally, in Section V.E.5, we discuss how to find control laws that balance the competing objectives of robust stability and performance.

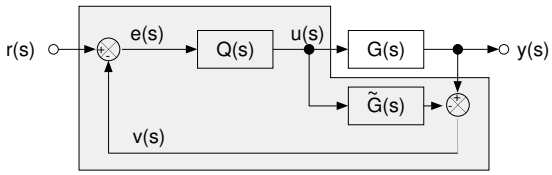

FIG. 25 Block diagram of an IMC controller. Shaded area is implemented either in a computer program or in control electronics.

1. The Internal-Model-Control parametrization

As a useful starting point, we introduce an alternative way of parametrizing control systems, known as "internal model control" (IMC) (Morari and Zafirioiu, 1989). The basic idea is to explicitly include a model $\widetilde{G}(s)$ of the system's actual dynamics, $G(s)$. This leads to the block diagram of Fig. 25. (For simplicity, we omit external disturbances, sensor noise, sensor dynamics, etc.) Solving the block dynamics in Fig. 25, one finds

$$y(s) = \frac{GQ}{1 + (G - \widetilde{G})Q}\, r \tag{5.43}$$

Notice that the feedback signal is $v = (G - \widetilde{G})u$. This shows explicitly that if we were to have a perfect model (and no disturbances), there would be no need for feedback. Feedback is required *only* because of our always imperfect knowledge of the model system and its disturbances. As we discuss in Section IX below, there is a deep connection between feedback and information.

There are a number of other advantages to the IMC formulation. It is easy to relate the IMC controller $Q(s)$ to the "classic" controller $K(s)$ discussed previously:

$$K(s) = \frac{Q}{1 - \widetilde{G}Q} \tag{5.44}$$

Because the denominator in Eq. (5.43) is $1+(G-\widetilde{G})Q$, the feedback system will become unstable only if $Q = -\frac{1}{G-\widetilde{G}}$,

which will be large when $G \approx \widetilde{G}$. In other words, as long as the model $\widetilde{G}$ is a reasonable approximation to the real system $G(s)$, a stable controller $Q(s)$ will lead to a stable closed-loop system. (Here, we assume that $G(s)$ itself is stable.) This is in contrast to the classic controller $K(s)$, where stable controllers can nonetheless destabilize the system. A more positive way of saying this is to note that for the system $G(s)$, the set of all stable $Q(s)$'s generates all stable controllers $K(s)$.

Another advantage of the IMC structure is that the above remarks about stability carry forward to cases where the system G and model $\widetilde{G}$ are nonlinear. If $G \approx \widetilde{G}$, then their difference may be well-approximated by a linear function, even when G and $\widetilde{G}$ are themselves strongly nonlinear.

Finally, in Eq. (5.43), we see that if $G = \widetilde{G}$, and we set $Q = 1/\widetilde{G}$, then we will have perfect control [$y(s) = r(s)$, exactly]. This means that even when the system is known exactly, it is necessary to invert the model transfer function $\widetilde{G}(s)$ in order to have perfect control. If $\widetilde{G}(s)$ is non-minimum phase (NMP) and has a zero in the right-hand plane (RHP), say at s_0, then the controller $Q(s)$ will have a pole at s_0 as well and will be impossible to implement at the corresponding frequency. (Intuitively, one needs infinite energy to move a system that has zero response.) This means that the bandwidth of the closed-loop system – whatever the feedback law chosen – will always be limited by the lowest-frequency RHP zero of the system $G(s)$. Thus, we see another reason to watch out for NMP dynamics and to eliminate them by redesigning the system whenever possible.

2. Quantifying model uncertainty

The IMC structure highlights the role that an internal model of the system plays. Since models are never perfect, one must learn to deal with the consequences of uncertainty. The first step is to quantify the uncertainty of the system model. The most obvious way is to allow for uncertainties in any model parameters. For example, in a second-order system such as Eq. (2.5), one could estimate uncertainties in the damping ζ and natural frequency ω_0. There are two limitations to this approach: First, it assumes that the form of the model is correct. But one may not know the physics of the model well, and, even if one does, one may choose to neglect certain parts of the dynamics (such as higher-order modes), which translate into errors in the model. Second, one would, in principle, have to come up with a special way of dealing with the effects of different kinds of parameters. For example, an uncertainty in natural frequency just amounts to a rescaling of the time axis, whereas an uncertainty in damping could mean that sometimes a system is underdamped and sometimes overdamped, which could have very different consequences for the control law.

In the control literature, the uncertainty that enters via imperfectly known parameters is called "structured uncertainty," since a certain model ("structure") is assumed. To get around the problems discussed above, one can consider instead "unstructured uncertainty." In this approach, we start with a nominal system $G_0(s)$. The actual system, which is unknown, is a member of a set of systems $\mathcal{G}(s)$, defined by an arbitrary *multiplicative uncertainty*:

$$\mathcal{G}(s) \equiv [1 + \Delta(s)W(s)]G_0(s) \tag{5.45}$$

where $\Delta(s)$ is an arbitrary transfer function that satisfies $||\Delta||_\infty \leq 1$ and where $W(s)$ is a transfer function that gives the uncertainty limits. The ∞ subscript on the norm refers to the "$\mathcal{H}_\infty$" norm, which is defined to be

$$||\Delta||_\infty \equiv \sup_\omega |\Delta(i\omega)| \, . \tag{5.46}$$

In words, the $\mathcal{H}_\infty$ norm is computed by taking that maximum of the magnitude of the function, in contrast to the more common Euclidean, or $\mathcal{H}_2$ norm, defined in frequency space by

$$||\Delta||_2 \equiv \left[\frac{1}{2\pi}\int_{-\infty}^{\infty} |\Delta(i\omega)|^2 d\omega\right]^{1/2} . \tag{5.47}$$

The reasons for using the $\mathcal{H}_\infty$ norm will become clear shortly. The best way to picture multiplicative uncertainty is in the complex s-plane (Fig. 27, where we plot $G_0(i\omega)$, the nominal system, and the band formed by superposition of the frequency-dependent multiplicative perturbations. At each frequency, the system is located within a circle of radius of radius $W(i\omega)$, as illustrated. Superposing all the circles gives the two bands shown. The multiplicative bound means that the uncertainties are expressed relative to $G_0(i\omega)$.

There is no deep reason for using multiplicative uncertainty. For example, one could use additive uncertainty, defined by $\mathcal{G}(s) \equiv G(s) + \Delta(s)W(s)$. But an additive uncertainty is easily redefined in terms of a multiplicative one; in addition, multiplicative uncertainties tend to arise naturally in control problems. For example, if the actuator (considered as part of the system) has a temperature-dependent gain and the equipment is to be used in rooms

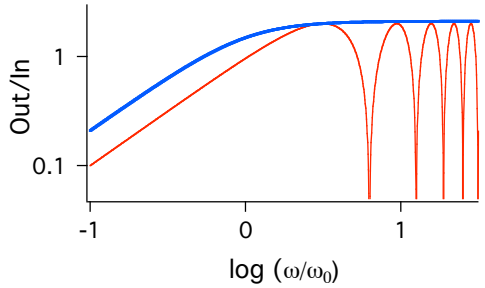

FIG. 26 (Color in online edition) Multiplicative bound for a variable delay. Red curve: magnitude of Eq. (5.48); blue curve: magnitude of Eq. (5.49).

of differing temperatures, then the set of systems ranges from $K_0 - \Delta K G_0(s)$ to $K_0 + \Delta K G_0(s)$, meaning that the fractional uncertainty $W = \frac{\Delta K}{K_0}$. In general, both K_0 and ΔK could be functions of frequency.

A more interesting example is a system that includes an unknown time delay τ, ranging from 0 to τ_{max}. If the nominal system is $G_0(s)$, then, at each frequency ω ,the uncertainty bound $|W(i\omega)|$ must be greater than

$$\left| \frac{\mathrm{e}^{-i\omega\tau} G_0}{G_0} - 1 \right| = |\mathrm{e}^{-i\omega\tau} - 1| \,, \qquad (0 \le \tau \le \tau_{max}) \,. \tag{5.48}$$

In Fig. 26, we show that a suitable bound is

$$W(s) = \frac{2.1\tau_{max} s}{1 + \tau_{max} s} \,, \tag{5.49}$$

which can be derived by noting the asymptotic behavior for $\omega \to 0$ and $\omega \to \infty$ of Eq. (5.48) and increasing the amplitude slightly to avoid "cutting the corner" in Fig. 26.

In practice, one way to construct a bounding function $W(s)$ is to measure a system transfer function several times, under a variety of conditions. Let the k^{th} transfer-function measurement be done over a set of frequencies ω_j, giving a series of magnitude-phase pairs (M_{jk}, ϕ_{jk}). For each point j, estimate a "typical value" $M_j^* \mathrm{e}^{i\phi_j^*}$. Then find an analytic function $W(s)$ that satisfies, for every j and k,

$$\left| \frac{M_{jk} \mathrm{e}^{i\phi_{jk}} - M_j^* \mathrm{e}^{i\phi_j^*}}{M_j^* \mathrm{e}^{i\phi_j^*}} \right| \le |W(i\omega_j)| \tag{5.50}$$

It is important to realize that M^* and ϕ^* need not be the averages of the measurements. The main usefulness of the robust approach is in dealing with the effects of systematic variations. If used for random variations following known statistics, it will likely be over-conservative. Typically, systematic variations arise because only part of a complex system is being modeled. As discussed above, the effects of higher-order modes may be neglected or projected away, and there may not be enough separation from the lower-order term to model the effect of those neglected modes by a white-noise term in a Langevin equation, which is the usual physics approach (Chaikin and Lubensky, 1995). Another example is nonlinear dynamics, which will be discussed below in Sec. VI. Because the methods we have been discussing are based heavily on linear techniques, one must assume that the system dynamics are linear about a given operating point. Although this is often true, the parameters and even the form of the linear system can vary with the operating point chosen. A final example is that one's system is almost always embedded within a larger system, whose dynamics are not modeled. Thus, an experiment in a room may show different behavior as a function of temperature. A biochemical network (see Sect. VII.C, below) may function in a cell whose environment changes significantly in different conditions, and so on. In all of these situations, the right thing to do is to choose typical conditions, which may or may not involve an average and then to identify the expected maximum deviation at each frequency ω_j. $|W(s)|$ is then an analytic function bounding all of these deviations.

3. Robust stability

In the previous section, we saw one way to quantify the uncertainty in a model of system dynamics. Given that one's system belongs to a family of systems $\mathcal{G}(s)$, one would like to choose a controller $K(s)$ so that, at a minimum, the feedback loop is stable for all possible realizations $G(s)$ of systems taken from the family $\mathcal{G}(s)$. This property of being stable over all possible realizations is known as robust stability.

Unlike the system, we can assume that we know the controller $K(s)$ exactly.[28]. Then, the loop gain $L = KG$ will be a member of a set $\mathcal{L} = K\mathcal{G}$ Since we have seen that a system goes unstable when the denominator of the transfer functions T and S equal zero, we must have that $1 + L(i\omega) \neq 0$ for all ω, for all values of any parameters used in K, and for all systems G. The last requirement can be restated succinctly as $1 + \mathcal{L} \neq 0$.[29]

To analyze things further, define $\mathcal{G}$ as the set of systems $G(s) = G_0(s)[1 + \Delta(s)W(s)]$, with W the uncertainty bound and Δ an arbitrary transfer function with magnitude ≤ 1. Similarly, write the set of loop transfer functions $\mathcal{L}$ as $L(s) = L_0(s)[1 + \Delta(s)W(s)]$, with $L_0(s) = K(s)G_0(s)$. Then the condition for robust stability can be written

$$|1+L(i\omega)| = |1+L_0(i\omega)+\Delta(i\omega)W(i\omega)L_0(i\omega)| > 0 \qquad \forall\omega \,. \tag{5.51}$$

This can be illustrated by a diagram analogous to Fig. 27, where instead of plotting the family of systems $\mathcal{G}$ one plots the family of loop transfer functions $\mathcal{L}$. Equation (5.51) then states that the light shaded area cannot touch the point -1. Because, at each ω, Δ is any complex number with magnitude ≤ 1, we can always choose the worst case, i.e., the Δ that minimizes the left-hand side of Eq. (5.51). This implies

$$|1 + L_0(i\omega)| - |W(i\omega)L_0(i\omega)| > 0 \qquad \forall\omega \,, \tag{5.52}$$

---

[28] If the controller is implemented digitally, this will be strictly true. Almost all practical robust controllers are implemented digitally, since the algorithms generate fairly complex controllers, which would be complicated to implement with analog circuitry.

[29] As discussed in Sec. IV.B above, we are simplifying somewhat. For a given controller $K(s)$, one would invoke the Nyquist criterion to see whether any system in the Nyquist plot of $\mathcal{L}(i\omega)$ circles -1 an appropriate number of times to be unstable. In practice, one is almost always worried about computing the limit where a system (or set of systems) crosses over from being stable to unstable, in which case the relevant criterion is $1 + \mathcal{L} = 0$.

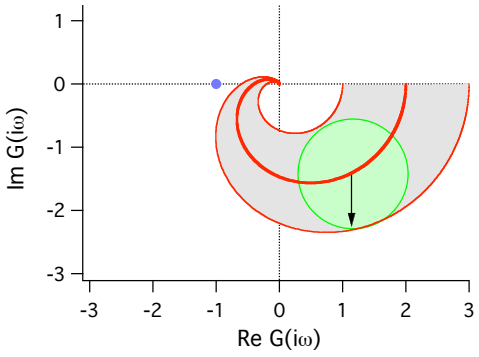

FIG. 27 (Color in online edition) Illustration of a multiplicative uncertainty bound and of robust stability. The thick, central line is the nominal system G_0, with $\mathrm{Im}[G_0(i\omega)]$ plotted vs. $\mathrm{Re}[G_0(i\omega)]$ over $0 < \omega < \infty$ ("Nyquist plot"). The thin lines that shadow it are the uncertainty limits, meaning that the actual system follows a path somewhere within the shaded area. The shaded circle gives the multiplicative uncertainty bound at one frequency. If we now interpret the thick central line as a loop transfer function and if the system is closed in a feedback loop, then robust stability implies that the set of loop transfer functions defined by the light shaded region cannot touch the point -1, which is indicated by the large dot.

or

$$\left|\frac{W(i\omega)L_0(i\omega)}{1+L_0(i\omega)}\right| < 1 \qquad \forall\omega\ . \tag{5.53}$$

Since the complementary sensitivity function of the nominal system is $T = \frac{L_0}{1+L_0}$, we can write this as

$$||WT||_\infty < 1\ , \tag{5.54}$$

where we have used the $\mathcal{H}_\infty$ norm as shorthand for $|W(i\omega)T(i\omega)| < 1\ \forall\omega$. The use of the $\mathcal{H}_\infty$ norm arises from the desire to be conservative, to be stable for the worst possible realization of one's system. Equation (5.54) also implies that the uncertainty bound W must be less than T^{-1} at all frequencies for robust stability to hold.

To see how robust stability works, let us consider again a first-order system with variable time lag. The nominal system is $G_0(s) = \frac{1}{1+\tau_0 s}$. If the maximum expected lag is τ_{max}, we saw above [Eq. (5.49)] that we can take $W(s) = \frac{2.1\tau_{max}s}{1+\tau_{max}s}$. Now, for a given controller, what is the maximum gain we can apply while still maintaining robust stability? Taking, for simplicity, $K(s) = K$, we see that Eq. (5.54) implies

$$f(K,\omega) \equiv \frac{2.1K\tau_{max}\omega}{\sqrt{1+\tau_{max}^2\omega^2}\sqrt{(1+K)^2+\tau_0^2\omega^2}} < 1 \tag{5.55}$$

We then seek $K = K_{max}$ such that $f(K_{max},\omega^*) = 1$ and $\frac{\partial f}{\partial\omega}\Big|_{K_{max},\omega^*} = 0$. Numerically, for $\tau_0 = 1$ and $\tau_{max} = 0.1$, one finds $K_{max} \approx 10$. By contrast, the stability limit found in Sec. IV.C for a first-order system with time delay was $K^*_{max} \approx 15$. [Eq. (4.10), with a factor of 2 for gain margin.] The robust-stability criterion leads to a smaller maximum gain than does a calculation based on a precise model of the system (with delay). Uncertainty thus leads to conservatism in the controller design.

4. Robust performance

The previous section was mostly concerned with stability, but the point of feedback is usually to improve the performance of an open-loop system. In Sec. V.B.3, we defined a scalar "performance index" that measures how well a system rejects errors and at what control costs [Eq. (5.8)]. One difficulty is that such optimal control assumes a particular disturbance $d(t)$ and a particular reference signal $r(t)$, whereas, in practice these signals vary. In other words, we would like to assure good performance for a set of disturbances and reference signals.

We begin by recalling that, in the absence of sensor noise, the tracking error $e_0(s) = S(s)[r(s) - d(s)]$, where the sensitivity function $S = \frac{1}{1+L}$. Thus, to have good performance, we would like to have $|S(i\omega)| \ll 1$. However, the loop gain $L = KG$ will go to zero at high frequencies, so that $S \to 1$ as $\omega \to \infty$, meaning that it is impossible to track a reference or compensate for disturbances to arbitrarily high frequencies. Actually, the situation is even more difficult, since the analytic structure of L constrains the form of S. Depending on the pole-zero structure of L (especially, the number of poles and zeroes in the RHP), one has a number of different analytical constraints on S. To give the simplest one, originally derived by Bode, assume that L has neither poles nor zeroes in the RHP (i.e., assume it is stable and NMP), and assume also that the relative degree of L is at least 2 (i.e., $L \sim \omega^{-n}$ as $\omega \to \infty$, with $n \geq 2$). Then one can show (Doyle *et al.*, 1992) that

$$\int_0^\infty \ln|S(i\omega)|d\omega = 0\ . \tag{5.56}$$

This means that on a log-linear plot of $S(i\omega)$, the area below 0 ($|S| < 1$) must be balanced by an equal area above 0 ($|S| > 1$). This is illustrated in Fig. 28. When $|S| < 1$, the control loop is decreasing the sensitivity to disturbances and the system is made more "robust." When $|S| > 1$, the control loop actually amplifies disturbances, and the system becomes more "fragile." Equation (5.56) implies a kind of "conservation of fragility," with frequency ranges where the system is robust to disturbances being "paid for" with frequency ranges where the system is fragile (Csete and Doyle, 2002). This is also known as the "waterbed effect": push $|S|$ down at some frequency, and it will pop up at another! As a consequence, increasing the gain will increase the frequency range over which disturbances will be rejected but will increase the fragility of the system at higher frequencies.

Given these constraints on S and given that r and d are usually unknown, one can proceed by defining a set

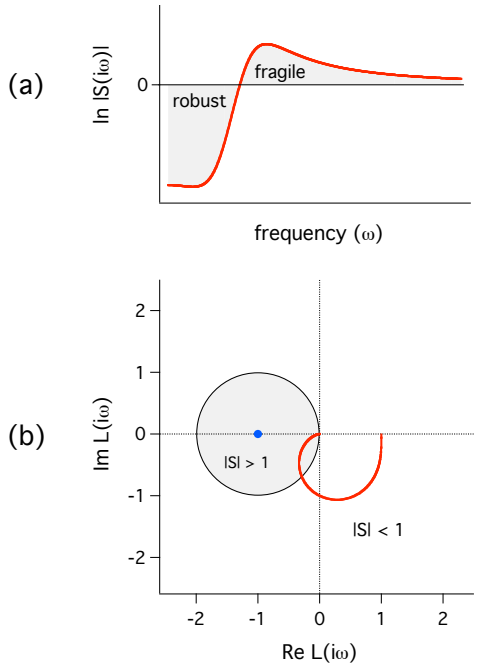

FIG. 28 (Color in online edition) Constraints on the sensitivity function $S = \frac{1}{1+L}$. (a) Log of the sensitivity function, $\ln|S(i\omega)|$ for a second-order system with proportional gain. The negative area, where control is "robust" is just balanced by the positive area, where control is "fragile." Note that the slow decay (here, $\ln|S| \sim \omega^{-2}$ at large frequencies) implies that a significant part of the positive area is off the plot. (b) Nyquist plot of the same system. Note that when the plot enters the shaded circle of radius 1, centered on -1, we have $|S| > 1$.

of expected or desired control inputs or disturbances and asking that the control error be small for any member of these sets. For simplicity, we consider only disturbances. Let the set of disturbances be characterized by $d(s) = \Delta(s)W_1(s)$, where, as before, $W_1(s)$ is a bounding function (the "performance weight") and $\Delta(s)$ is an arbitrary transfer function with $|\Delta| \leq 1$. Typically, $W_1(s)$ is large at low frequencies and cuts off at high frequencies. The functional form is often taken to be a lag compensator [Eq. (5.6)]. Again, W_1 represents the largest disturbances one expects or, at least, that one desires to suppress. The largest error that one expects is

$$|e_0(i\omega)| = |dS| \leq^{\sup}_{\omega} |W_1(i\omega)S(i\omega)| = ||W_1S||_\infty \ . \quad (5.57)$$

Then we can reasonably ask that the worst possible error resulting from the most dangerous expected disturbance be bounded, i.e. that $||W_1S||_\infty < 1$, where W_1 is implicitly rescaled to make the bound 1.

Equation (5.57) represents the desired nominal performance, given an accurate model of the system (Morari and Zafirioiu, 1989). One should also ask for robust performance, so that Eq. (5.57) holds for all systems G allowed by the uncertainty. We replace S by S_Δ, the sensitivity function of a particular realization of one of the possible systems $G_\Delta = G_0(1 + \Delta W_2)$, where we now use W_2 for the multiplicative bound on the system uncertainty. Then

$$|W_1S_\Delta| = \frac{|W_1|}{|1 + L_0 + \Delta W_2L_0|} = \frac{|W_1S|}{|1 + \Delta W_2T|} < 1 \ , \quad (5.58)$$

Multiplying through, we have

$$|W_1S| < |1 + \Delta W_2T| \leq 1 - |W_2T| \qquad \forall\omega \ , \quad (5.59)$$

or

$$|||W_1S| + |W_2T|||_\infty < 1 \ . \quad (5.60)$$

This is the robust-performance problem. Again, the $\mathcal{H}_\infty$ norm means that the relation holds for all frequencies. In Eq. (5.60), all quantities are evaluated at $s = i\omega$, and S and T refer to the nominal system G_0.

5. Robust control methods

From our point of view, the formulation of the robust-performance problem is more important than its solution. Equation (5.60) may be thought of as another type of optimal-control problem. Instead of asking that the LHS be less than 1 (which may not be possible, given that the performance criterion reflects one's desires, not what is possible), we can ask that it be less than some bound γ. Then the problem is to find a controller K that minimizes $|||W_1S| + |W_2T|||_\infty$, given performance and stability weights W_1 and W_2 and given a nominal system G_0. Finding even an approximate solution to this problem requires a sophisticated treatment (Doyle *et al.*, 1992; Özbay, 2000; Skogestad and Postlethwaite, 1996) that is beyond the scope of this tutorial. Alternatively, one can seek a numerical solution by postulating some form for $K(s)$ and running a numerical optimization code to find the best values of any free parameters. On the one hand, such optimization is in principle straightforward since one usually does not want a controller with more than a dozen or so free parameters. On the other hand, the "landscape" of the optimized function is usually rugged, and a robust optimization code is needed. Recent work has shown that genetic algorithms can be quite effective (Jamshidi *et al.*, 2003). For the problems that most readers are likely to encounter in practice, there is probably little difference between the two approaches, in that both lead to a numerical solution for which software is available (e.g., Matlab or Scilab).

Here, we limit our discussion to a reconsideration of loop shaping. If we write out Eq. (5.60) in terms of L_0 and recall that L_0 is large at low frequencies and tends to zero at high frequencies, then we easily derive that

$$\begin{aligned} |L_0(i\omega)| &> \frac{|W_1(i\omega)|}{1 - |W_2(i\omega)|} \qquad \omega \to 0 \\ |L_0(i\omega)| &< \frac{|1 - W_1(i\omega)|}{|W_2(i\omega)|} \qquad \omega \to \infty \ . \end{aligned} \quad (5.61)$$

Thus, Eq. (5.61) provides explicit criteria to use in shaping the low- and high-frequency form of the controller $K(s) = L_0(s)/G_0(s)$.

Finally, the above discussion of robust control methods neglects sensor noise. In Sec. V.D on Kalman filtering, we saw how to estimate the system state in the presence of noise. The optimal state estimated by the Kalman filter, however, assumed that one had accurate knowledge of the system's dynamics. Combining robust methods with optimal state estimation remains a topic of current research (Petersen and Savkin, 1999).

In the physics literature, robust controller design has seldom been used, but two recent examples both concern the control of a positioner for an atomic force microscope head (Salapaka *et al.*, 2002; Schitter *et al.*, 2001). In the former reference, the system used a piezoelectric stack as an actuator, with a control loop to counteract nonlinearities and hysteresis in the actuator, as well as mechanical resonances in the stage. A first attempt at using a PI controller gave a bandwidth of 3 Hz. Use of the $\mathcal{H}_\infty$ techniques led to a bandwidth of over 100 Hz. It should be noted, however, that the increase in performance came from replacing a two-term controller (P and I) with a more complicated form for K that had 12 terms to tune, thus giving much more freedom to shape $K(s)$. It did not *per se* come from the choice of the $\mathcal{H}_\infty$ (robust) metric. What robust and optimal control methods offer is a rational way of using performance objectives (e.g., high bandwidth of the closed-loop system) to choose the many free parameters in $K(s)$. Without some systematic guide, tuning a 12-parameter controller would be difficult, if not impossible. The use of robust measures helps to insure that a solution that works "on paper" will perform satisfactorily in the real world, taking account of errors in the modeling, drift in parameter values, and so forth.

VI. NOTES ON NONLINEARITY

Most of our discussion so far has focused on the control of linear systems. Many types of dynamical systems indeed are linear, or are close enough to an equilibrium that they behave approximately linearly about some equilibrium point. A straightforward approach is "gain scheduling," where one measures the transfer function locally over a range of setpoints, with the positions (and even types) of poles and zeros evolving with setpoint. One then varies the control algorithm parameters (or even structure) as a function of the local transfer function.

While gain scheduling works well for weak nonlinearities, it does not for stronger ones. The past few decades have brought an increasing awareness of the importance — and ubiquity — of strongly nonlinear systems, and there have been increasing efforts to find ways of controlling such dynamical systems. The difficulty is that most of the methods we have discussed above, including frequency response and the manipulation of pole positions, make sense in general only for linear or nearly linear systems and often cannot even be adapted to study nonlinear systems. Indeed, nonlinear systems show phenomena such as chaos, hysteresis, and harmonic generation that do not exist in linear systems, implying that the failure of the linear methods in general is a fundamental rather than a superficial one. Until recently, the physics literature and the control-theory literature on nonlinear dynamics were fairly separate. There is little point in trying to review the vast variety of different approaches that exist, a variety that no doubt reflects our more limited and less systematic understanding of nonlinear systems. Instead, we briefly revisit the notion of stability and then give a couple of illustrations.

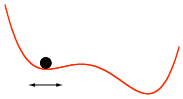

FIG. 29 (Color in online edition) Metastability: the ball is stable to small perturbations but unstable to larger ones.

Over a century ago, Lyapunov generalized the notions of stability discussed in Sec. IV.A. Such a generalization is necessary because the linear stability analysis discussed there is not always reliable. For example, a system that is stable to infinitesimal perturbations may be unstable to finite perturbations ("metastability"), as illustrated in Fig. 29. See Strogatz (1994) for a physicist's discussion and Dutton *et al.* (1997) for a control engineer's discussion of stability in nonlinear systems and how Lyapunov functions can be used to prove the stability of a solution to perturbations lying within a given region of phase space. Most of the methods from the control literature – describing functions, Popov's method, Zames' circle criterion, etc. (Dutton *et al.*, 1997) – deal only with rather specific types of nonlinearity and have correspondingly limited ranges of applicability. Instead of cataloguing all these cases, we give one example of a (basically unavoidable) nonlinearity that is a common topic in control-theory texts and a second, more recent example from the physics literature, chosen to illustrate how different the approaches can be.

A. Saturation effects

In our discussion of control laws, such as proportional control, we always assumed that however large the error might be, it is always be possible to generate the appropriate control signal. But every actuator has its limits. For example, if temperature is being controlled, then the actuator will often be a Peltier element, which pumps heat into a sample (i.e., heats it) when positive current is used and pumps heat out of a sample (i.e., cools it) when negative current is used. The Peltier element uses a bipolar current source with a maximum possible current. The actual control signal will then resemble the

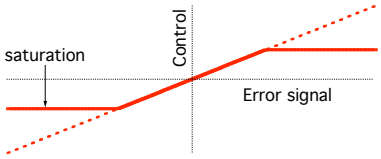

FIG. 30 (Color in online edition) Saturation of proportional gain. Thin dashed lline is the proportional gain with no saturation. Thick lines show saturation. There is, in general, no reason for saturation to be symmetric about zero, as depicted here.

thick trace in Fig. 30 rather than the thin dashed one. The control law is therefore nonlinear for large-enough error signals. Such a saturation always occurs, although one may or may not encounter in a given situation the large errors needed to enter the nonlinear regime.

Saturation need not have a dramatic effect on proportional control. When the error signal is too large, the system applies its largest correction signal, which is smaller than it should be. Intuitively, the quality of control will smoothly degrade as the errors become larger and larger, relative to the saturation value (often called the "proportional band" in the control literature).

The same kind of saturation effect, however, also occurs with integral control, and there its effects are more subtle. Consider a pure integral control law. If the system receives a large, long-lasting perturbation, the error integral will begin to build up, and hence the control signal. At some point, the control signal saturates, leaving the error to continue to build up (assuming that the error signal still has the same sign). The control signal will have reached its maximum value, where it will stay, until the *integral* term has been reduced. For this to happen, it is not sufficient that the system return to a state of zero error, for a large constant part of the integral will have built up. The effect will be to create a large error of the opposite sign (and perhaps to be unstable). The easy fix — universally used in practical integral control laws, including PID ones — is to freeze the value of the integral error whenever the control signal saturates. When the signal reenters the proportional band, one updates the integral term as usual, and the performance will then be as calculated for the linear system. This runaway of the integral term due to the saturation of the activator is known as "integral windup." It may be analyzed more formally than done here (Dutton *et al.*, 1997), but the intuitive explanation should suffice for our purposes.

Saturation effects are unavoidable in control situations because they are part of the control mechanism itself. Similar nonlinearities may also be dealt with in various *ad hoc* ways. For example, if a digital controller is used, there will be quantization nonlinearity due to the finite word size of the A/D and D/A converters.[30]

(These effects can be usually be cured most easily by making good use of the full dynamic range and, when more desperate, by investing in more expensive A/D and D/A chips that have more bits.) But the more interesting and important nonlinearities are to be found in the system dynamics themselves.

As the above examples show, nonlinearities associated with discontinuities in a function or one of its derivatives are typical in engineering problems, while the systems that physicists encounter more often have analytic nonlinearities. The following example shows how an analytic nonlinearity leads to complex, chaotic behavior that is nonetheless controllable.

B. Chaos: The ally of control?

A chaotic dynamical system is one that shows sensitive dependence to initial conditions (Strogatz, 1994). Two nearby initial conditions will diverge exponentially. Most often, one deals with dissipative chaotic systems, where the dissipation implies that the system will be confined to a finite volume of phase space (state space, in our language here.) Although the system is always locally unstable, the state vector stays in some general region. One can ask whether it is possible, through *small* variations of one or more of the system's control parameters, to stabilize the dynamics. (Stabilization through large variations is less interesting, because one can usually find a stable region for some range of control parameters.)

The first algorithm to stabilize a system with chaotic dynamics was given by Ott, Grebogi, and York (OGY) in 1990 (Ott *et al.*, 1990). Their algorithm drew on several key features of chaotic motion: First, once initial transients have passed, the system's motion is on an attractor, a set of points in phase space. Second, the motion is ergodic on that attractor. This means that the system revisits arbitrarily close to a given point arbitrarily many times. Third, embedded in each attractor is a dense set of unstable periodic orbits. Without getting into the types of orbits, we will think about the simplest kind, a fixed point. In that case, each attractor will contain an unstable fixed point.

The idea of OGY is to stabilize motion about an unsta-

[30] In A/D conversion, the effects of quantization error can often be reduced by deliberately adding noise to the signal. If the noise amplitude is roughly one least-significant bit (LSB), then successive digitizations will reflect the true amplitude of the signal. For example, imagine measuring a signal level of 0.4 with a digitizer that reports 0 or 1. With no noise, one measures always 0. If the noise level (i.e., its standard deviation) is on the order of 0.5, one will measure a "0" 60% of the time and a "1" 40% of the time. Measuring the signal several times will suffice to get a reasonable estimate of the average. The method, similar in spirit to the technique of delta-sigma conversion discussed above in Sec. V.C, is known as *dithering* (Etchenique and Aliaga, 2004). It gives an example of how, in a nonlinear system, noise can sometimes improve a measurement.

ble orbit (or fixed point) x^* by waiting until the system brings the state vector x near to x^* (Grebogi and Lai, 1999). By the ergodicity properties of chaotic motion, this will always happen if one waits long enough. (The closer one wants x to approach x^*, the longer one has to wait.) Once this happens, the control algorithm is activated. If the dynamics are given by $x_{n+1} = f(\lambda, x_n)$ (assume discrete dynamics for simplicity), then one may use the proximity of x_n to x^* to linearize about the fixed point. The linearized dynamics are

$$x_{n+1} - x^* \approx \frac{\partial f}{\partial x}(x_n - x^*) + \frac{\partial f}{\partial \lambda}\Delta\lambda_n \ . \tag{6.1}$$

One then changes λ in Eq. (6.1) so that $x_{n+1} = x^*$, i.e., one sets

$$\Delta\lambda_n = -\frac{\frac{\partial f}{\partial x}(x_n - x^*)}{\frac{\partial f}{\partial \lambda}} \ . \tag{6.2}$$

Of course, choosing $\Delta\lambda_n$ in accordance with Eq. (6.2) will not make x_{n+1} precisely equal to x^*, since that choice is based upon the approximation in Eq. (6.1). But if the original distance $x_n - x^*$ is small enough, the system will quickly converge to x^*.

We can illustrate the OGY algorithm on a simple chaotic dynamical system, the logistic map

$$x_{n+1} = \lambda x_n(1 - x_n) \ , \tag{6.3}$$

which is a standard example of a simple dynamical map that shows complex behavior as the control parameter λ is adjusted (Strogatz, 1994). Here, at $\lambda = 3.8$, the system is normally chaotic. The goal will be to stabilize the system's motion about the normally unstable fixed point, given by $x^* = 1 - 1/\lambda \approx 0.74$. The *OGY* control algorithm is turned on at $t = 0$. See Fig. 31a, which shows a logistic map with control initiated at $t = 0$. One waits until the natural motion of the system brings it to within a predefined tolerance $x^* \pm \varepsilon$ (here, $\varepsilon = 0.02$). In the particular run shown in Fig. 31a, this happens at time step 24. The OGY algorithm is then activated, as shown in Fig. 31b. Note that the system state x_{24} is just slightly above the setpoint. The idea, as discussed above, is to change λ to $\lambda_n = \lambda + \Delta\lambda_n$ so as to position the fixed point of the modified dynamical system (logistic map with control parameter λ') *above* the point x_{24} so that the repulsive dynamics of the modified system pushes the point towards x^*. Eq (6.2) shows that we set

$$\Delta\lambda_n = -\left[\frac{\lambda(1 - 2x^*)}{x^*(1 - x^*)}\right](x_n - x^*) \approx 9.3\,(x_n - x^*) \ . \tag{6.4}$$

The values of λ_n are shown in Fig. 31b. Note how λ_{24} shoots up in response to x_{24}, which represents the first time the system has entered the tolerance zone $x^* \pm \varepsilon$. The system quickly settles down and stabilizes about the desired setpoint x^*. In most applications, there would be noise present in addition to the deterministic dynamics,

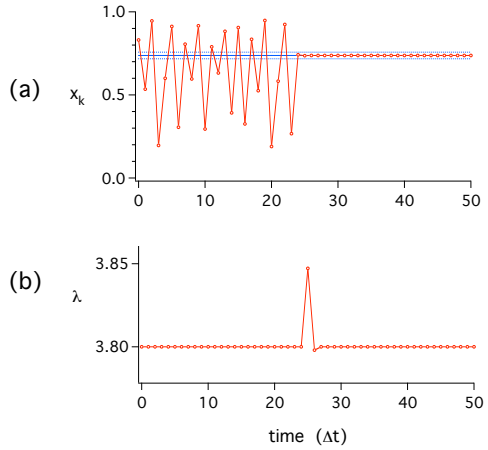

FIG. 31 (Color in online edition) Illustration of the OGY chaos-control algorithm. (a) Time series of x_n. Control is initiated at $t = 0$ about unstable fixed point x^* (horizontal solid line). Dashed lines show the tolerance range $\pm\,\varepsilon$ within which OGY algorithm is active. Here, the algorithm actives at $t = 24$. (b) Control parameter λ_n, as adjusted by OGY algorithm to stabilize fixed point.

in which case the algorithm would lead to small fluctuations about the fixed point for small noise. The occasional large kick may cause the system to lose control. One would then have to wait for the system to re-enter the tolerance zone to reactivate control.

Can one do better than merely waiting for the system to wander near the setpoint? In fact, small "targeted" corrections to the motion can drastically reduce the waiting time. The idea is to extrapolate backwards from the setpoint to find a point near the current state of the system. Then one works out the perturbation needed to approach the target. Small control-parameter perturbations at the right time can have a large effect. In a real-life application, NASA scientists have used such ideas to send spacecraft on remote missions throughout the solar system using small thrusts to perturb the motion in the planets' and sun's gravitational fields. These and other aspects of targeting are discussed in Shinbrot (1999).

OGY uses the system's dynamics, notably the ergodic property of chaos, where the dynamical system generically passes arbitrarily close to every point in phase space ("state space"). In contrast, most control algorithms "fight the system," in that they change the dynamical system by overpowering it. In OGY, only small perturbations to a control parameter are required. One nudges the system in just the right way at just the right point at just the right moment, to persuade it to behave as desired. One limitation of OGY is that one cannot choose the setpoint arbitrarily. One can stabilize the system about an unstable fixed point, but that unstable fixed point must first exist in the ordinary dynamical system. This limitation is often not too important practically, because there are lots of unstable fixed points and periodic orbits to use. Also, with targeting, the time to find

the required region of phase space is often short. But conceptually, these limitations do not exist in classical algorithms.

The seeming dichotomy between chaotic control and classical control algorithms is not as sharp as I have made it seem in the last paragraph. OGY was the first in a long series of algorithms for chaotic control, and many of them blend elements of classical algorithms (Schuster, 1999). On the other side, some of the classical control algorithms have at least some of the flavor of the OGY algorithm. For example, in the example of the unstable Stealth fighter plane referred to above, instability is actually desirable feature as it allows the plane to respond much faster to control signals than a stable system would. Again, relatively small control signals (wing flap movements, etc.) can produce large effects in the planes motion, by working about unstable equilibria.

VII. APPLICATIONS TO BIOLOGICAL SYSTEMS

From the beginnings of control theory, there have been attempts to make connections to biological systems (Wiener, 1961). Indeed, one does not have to look far to find numerous examples of regulation, or "homeostasis." Body temperature is regulated to the point where variations of one degree imply sickness and of ten degrees, death. When we are hot, we sweat and cool by evaporation. When we are cold, we shiver and increase the circulation of warm blood to cool areas. Similarly, osmotic pressure, pH, the size of the eye's pupils, the vibration of hair cells in the inner ear – all are tightly controlled. Over the years, there have been numerous attempts to model such processes at the "system" or physiological level (Keener and Sneyd, 1998). In many cases, it has been possible to come up with models that mimic, at least partly, the observed behavior. Because the models describe high-level phenomena, they are largely phenomenological. That is, they involve somewhat arbitrarily chosen elements that have the right behavior but may be only loosely tied to the underlying physiological elements. In addition, one commonly finds that the more closely one studies a phenomenon, the more baroquely complex become the models needed.

With these limitations in mind, we can trace two additional approaches to understanding the role of feedback in biology. The first course is to find simple enough settings where standard feedback ideas may safely be employed. These are largely found in the biochemistry of enzyme-catalyzed reactions and, more recently, in gene-expression models. The second course is to be begin to tackle more complex situations. Here, the main innovation is that instead of a simple feedback loop, one has a complex network of interactions, and one has to consider both the geometry and topology of that network. In the following sections, we will first discuss a phenomenological example, then some simple biochemical feedback loops, and finally give a brief overview of current efforts to understand networks of interacting genes and proteins.

The reader will notice that the biological examples to be discussed lack any man-made element. Our previous discussions all started from a natural physical system and added a deliberate coupling between variables, creating a feedback loop. In the natural system, a variable Y depends on X; then one creates an additional dependence of X on Y. (For example, in the vibration-isolation device described in Sec. V.C.1, the accelerometer signal depends on the forces exerted on it by the earth. The feedback apparatus then creates a way for the accelerator signal to influence those forces, via the piezo-element actuator.) In the biological systems below, the couplings between Y and X and then X and Y are usually both "natural." But insofar as the two couplings may be separately identified, feedback will be a useful way of looking at the dynamics, whether the couplings are created by man or by nature.

A. Physiological example: The pupil light reflex

The eye is an amazing organism. It can respond with reasonable sensitivity over a wide range of light levels, from single photons to bright sunlight. The eye uses a mix of feedback methods, the most important of which is adaptation, as summarized by the empirical response law of Weber: the eye's sensitivity is inversely proportional to the background light level (Keener and Sneyd, 1998).

Another mechanism that is particularly important at the highest light levels is the pupil light reflex. When light levels are high, the pupil contracts, reducing the light flux onto the retina. The size of the pupil is controlled by circularly arranged constricting muscles, which are activated and inhibited (left to relax) by control signals from the brain. More light causes activation of the constricting muscles, which shrinks the pupil area and limits the light flux at the retina.

The pupil light reflex is a particularly attractive example of physiological feedback. First, the state variable, the retinal light flux, is an intuitive quantity, and the actuator mechanism, pupil size, is easy to understand. Second, using a trick, one can "break open" the feedback loop. Ordinarily, the incident light flux covers a larger area than the pupil, so that adjustments to the pupil size adjust the retinal light flux. (Fig. 32a.) If, however, one uses a beam of light that is narrower than the minimum pupil size, adjusting the area will not change the retinal light flux. (Fig. 32b.) The feedback loop is broken, and one can then study how variations in the light intensity change the pupil size. For example, one could impose sinusoidal intensity variations in order to measure the transfer function of the pupil response. While we do not have space to describe that transfer function fully, its most important feature is that the pupil response to a changing light stimulus is delayed by about 300 ms. This leads to an amusing consequence. If a narrow beam of light shines on the edge of the pupil (Fig. 32c), the pupil

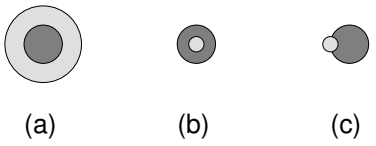

FIG. 32 Light beams and pupil sizes. Light beam is shaded light; pupil is shaded dark. (a) The ordinary case: the beam is larger than the pupil. (b) Breaking the feedback loop: the beam is smaller than the pupil. (c) Inducing pupil oscillations: a small beam falls on the edge of the pupil.

will begin to contract. Because of the delay, it will continue to contract after it is small enough that the beam no longer enters the pupil. After the delay, the eye realizes there is no light and starts to enlarge the pupil. This continues, because of the delay, somewhat past the moment when the pupil is large enough to admit light to the retina. The system then begins to contract, and thus continues on, in steady oscillation. This really happens!

The ability to open the feedback loop by using a narrow stimulus light beam also allows one to substitute an electronic feedback for the natural one. One measures the pupil-size variations and adjusts the light intensity electronically, according to an algorithm chosen by the experimentalist. In effect, one creates a primitive "cyborg," melding man and machine. If one implements ordinary proportional feedback, one regulate artificially the light intensity to a constant value. The system goes unstable and the pupil oscillates when the gain is too high.

Here, we shall account qualitatively for these observations using a model describing the pupil light reflex that is due originally to Longtin and Milton (1989a,b). Our discussion is a simplified version of that of Keener and Sneyd (1998).

We start by relating the retinal light flux φ to the light intensity I and pupil area A:

$$\varphi = IA \ . \tag{7.1}$$

We next relate the muscular activity x to the rate of arriving action potentials (the signals that stimulate the muscles):

$$\tau_x \frac{dx}{dt} + x = E(t) \ . \tag{7.2}$$

Here, we have taken simple first-order dynamics with a time constant τ_x. The driving $E(t)$ is the rate of arriving action potentials:

$$E(t) = \gamma F \left(\ln \left[\frac{\varphi(t-\Delta t)}{\bar{\varphi}} \right] \right) \ , \tag{7.3}$$

where γ is a phenomenological rate constant and the response delay is modeled by evaluating the retinal light flux a time Δt in the past (i.e., by $\varphi(t-\Delta t)$). The function $F(x) = x$ for $x \geq 0$ and 0 for $x < 0$, so that $\bar{\varphi}$ acts as a threshold retinal light level. In other words, muscles are activated only when there is sufficient light. The logarithm incorporates Weber's law, mentioned above. Equation (7.2) illustrates the "phenomenological" character of such large-scale models, where the form of the equation, in particular of $E(t)$ is chosen to be in qualitative agreement with empirical observations.

In order to close the model, we need to relate the pupil area A to the muscular activity x. High activity should give a small pupil area. Again, one uses an empirical, phenomenological form:

$$A(x) = A_{min} + (A_{max} - A_{min}) \frac{\theta^n}{x^n + \theta^n} \ , \tag{7.4}$$

which smoothly interpolates between A_{max} at zero activity ($x = 0$) to A_{min} at infinite activity. The parameters θ and n must be fit to experiment. Putting Eqs. (7.1)-(7.4) together, we have

$$\tau_x \frac{dx}{dt} + x \ = \ \gamma F \left(\ln \left[\frac{I(t-\Delta t) A(x(t-\Delta t))}{\bar{\varphi}} \right] \right) \tag{7.5}$$

$$\equiv \ g(x(t-\Delta t), I(t-\Delta t)) \ . \tag{7.6}$$

Because $A(x)$ is a decreasing function of x, one can always find a steady-state solution to Eq. (7.6) when the light intensity I is constant. Let us call this solution x^*, which satisfies $x^* = g(x^*, I)$. We linearize the equations about x^*, defining $x(t) = x^* + X(t)$. This gives

$$\tau_x \frac{dX}{dt} + X = -K_p X(t-\Delta t) \ , \tag{7.7}$$

where $K_p = -g_x(x^*, I)$ can be viewed as a proportional feedback gain. Equation (7.7) is nothing more than the first-order system with sensor lag and proportional feedback that we considered previously in Sec. IV.C. [See Eqs. (4.8) and (4.9).] The stability may be analyzed analogously; one again finds that for high-enough gain K_p, the system begins to oscillate spontaneously. This is in accordance with the "cyborg" experiments using artificial gain and with the use of light that falls on the pupil edge to excite oscillations, too. In the latter case, the feedback becomes nonlinear, since the coupling changes discontinuously when the area reaches a critical value A^*, which divides the region where changes in A do or do not affect the retinal light flux. Crudely, this functions as a locally infinite gain, which is unstable.

Notice that our modeling was in two steps. In the first, we formulated nonlinear equations. We related the muscle activity to the light flux, the light flux to the pupil area, and the pupil area to the muscle activity. This "loop" of relationships is the nonlinear generalization to our previous discussions of feedback loops, which applied to linear equations. Because the system is entirely a "natural" one, it may not be immediately obvious how to identify the "system," "sensor," and "controller." Here, the "system" sets the pupil size, the "sensor" is the retina, and the "controller" is presumably circuitry in the brain (or an explicit computer algorithm in the cyborg setup described above). In general, one is confronted with a system of coupled variables. Whether they neatly

decouple into traditional feedback elements or not will depend on the particular situation.

In the second step, we linearized about a fixed point, coming up with equations that could be analyzed using block-flow diagrams. This strategy is not infallible. For example, there may not be a steady state to linearize about. Still, it is one of the handier approaches for dealing with nonlinear systems. And in biology, strong nonlinearities are the rule.

In the example of pupil-light-flux control, the time delay in reacting to light changes plays a key role. Time delays are present in most physiological responses, and people compensate for such time delays by using feedforward algorithms. In a very interesting recent experiment on reaction responses, Ishida and Sawada (2004) show that people in fact slightly over compensate; they suggest that being "proactive" minimizes transient errors while tracking erratic motion .

B. Fundamental mechanisms

While feedback is present in many macroscopic, physiological processes, it also plays a role in more fundamental, microscopic settings. The traditional viewpoint focuses on enzyme-catalyzed biochemical reactions, where, the rate of production of some desired molecule is greatly accelerated by a catalyst (often a protein, or enzyme). By inhibiting or enhancing the production of the enzyme, one gains great control over the quantity of product. As we shall discuss below, in a cell, there are thousands of such reactions, all coupled together in a "genetic network." The theoretical analysis of simple networks has a long history (Wolf and Eeckman, 1998). What is exciting is that recently, it has become possible to create, by genetic engineering techniques, simple artificial genetic networks that illustrate basic feedback mechanisms. We shall give two examples. The first, due to Becksei and Serrano (2000) (cf. Gardner and Collins (2000)), illustrates how negative feedback can limit variability in gene expression, providing robustness against changing external conditions. The second shows how positive feedback can be used to switch a gene on and off (Gardner *et al.*, 2000; Kaern *et al.*, 2003).

1. Negative feedback example

We begin with an example of negative feedback. Becksei and Serrano constructed a simple genetic circuit in which a gene expresses a protein that actively represses its own production. In their experiment, the protein "tetracycline repressor" (TetR) was fused to a fluorescent protein, so that its expression could be monitored optically. (A cell's fluorescence is proportional to the number of TetR molecules present.) In order for the gene to be expressed, RNA polymerase (RNAP) must bind just "upstream" of the DNA coding for the gene. However, the TetR protein also binds to the DNA in competition with the RNAP. If the TetR is bound, then RNAP cannot bind, and TetR is not produced. The concentration of TetR, R, is governed by an equation of the form

$$\frac{dR}{dt} = \frac{\alpha}{1+KR} - \lambda R \,, \tag{7.8}$$

where α is the rate of production of TetR in the absence of the feedback mechanism. (The rate α depends on the concentration of RNAP, which produces mRNA, which leads to the production of the actual protein TetR. All of these dynamics are assumed to be fast compared to the production rates of TetR.) In Eq. (7.8), λ is the rate of degradation of TetR in the cell, and K is related to the binding affinity of TetR to the upstream DNA site. If $K > 0$, the feedback is negative, since increasing concentrations of TetR suppress the production rate of the enzyme. ($K = 0$ implies no feedback.) See (Becksei and Serrano, 2000) for the full equations and (Wolf and Eeckman, 1998) for the thermodynamic background. [Essentially, the KR term comes from considering the relative Gibbs free energies of the DNA when TetR is bound or not. Such kinetic laws are variants of the Michaelis-Menten law for enzyme kinetics (Keener and Sneyd, 1998).]

The main result of Becksei and Serrano is experimental evidence that negative feedback reduces the variability of gene expression. (They disable the feedback both by modifying the TetR protein so that it does not bind to the DNA site and by introducing additional molecules that bind to the site but do not interfere with the RNAP.) From our point of view, it is easy to see where such a behavior comes from. Intuitively, adding negative feedback speeds the system's response to perturbations. The system will spend more time in the unperturbed state than without feedback, reducing variation.

To see this in more detail, we solve first for the steady-state production level of TetR: Setting the LHS of Eq. (7.8) = 0, we find

$$R^* = \frac{1}{2}\left(-\frac{1}{K} + \sqrt{\frac{1}{K^2} + \frac{4\alpha/\lambda}{K}}\right) \,, \tag{7.9}$$

which decreases from $\alpha/\lambda \to 0$ when K goes from $0 \to \infty$. Small perturbations to the cell then lead to small variations in $R(t) = R^* + r(t)$, which obeys the linear equation,

$$\dot{r} = -\lambda^* r \,, \qquad \lambda^* = \lambda + \frac{\alpha K}{(1+KR^*)^2} \,. \tag{7.10}$$

The decay rate λ^* goes from $\lambda \to 2\lambda$ for K going from $0 \to \infty$. To model fluctuation effects, one adds a stochastic term $\xi(t)$ to Eq. 7.10, with $\langle\xi\rangle = 0$ and $\langle\xi(t)\xi(t')\rangle = \Lambda^2\delta(t-t')$ (white noise). Then the fluctuations of r obey what is essentially the equipartition theorem:

$$\sigma_r \equiv \langle r^2\rangle = \frac{\Lambda}{\sqrt{\lambda^*}} \,. \tag{7.11}$$

In Eq. (7.11), we see explicitly that the negative feedback loop, which increases the value of λ^*, reduces the fluctuation of the system in response to a fixed level of noise. This is what Becksei and Serrano observed in their experiments. In the same system, Rosenfeld *et al.* (2002) later showed directly that the negative feedback indeed leads to faster response times.

2. Positive feedback example

In biology, and elsewhere, one use of positive feedback is in constructing a switch that can go between two separate states, either one of which is locally stable to small perturbations. A large-enough perturbation, however, will make the system switch from one state to the other. In electronics, such ideas are the basis of the *flipflop* circuit, which toggles between two states (conventionally known as 0 and 1) and forms the basis of digital memories. For essentially topological reasons, in phase space, between any two stable states must lie an unstable intermediate state. (See Fig. 33b.) Near the intermediate state, the system shows a positive feedback that drives the system into either of the adjoining stable states. The statement that "positive feedback leads to a switch or to oscillations" comes from the common situation where, in the absence of feedback, there is only one stable state. Adding positive feedback then converts the stable to an unstable state via either a pitchfork (or transcritical) bifurcation (with two stable states as the outcome) or via a Hopf bifurcation (limit-cycle oscillator).

In biology, positive feedback can lead to a situation where the expression of one gene inhibits the expression of a second gene, and *vice versa*, an idea that goes back at least to Monod and Jacob (1961). A simple model for the concentration dynamics of two proteins u and v gives

$$\begin{aligned} \dot{u} &= \frac{\alpha_1}{1+v^n} - u \\ \dot{v} &= \frac{\alpha_2}{1+u^n} - v \,. \end{aligned} \tag{7.12}$$

where α_1 and α_2 are the rates of production of u and v in the absence of a repressor and where the "cooperativity exponent" $n = 2$. Equation (7.12) resembles Eq. (7.8), with some important differences. First, as mentioned above, increased production of u inhibits the production of v, and *vice versa*. Second, the inhibitor binding, in both cases, is assumed to show cooperativity. The exponents $n = 2$ mean that two molecules of u or v bind in rapid succession to the DNA. [The specific value $n = 2$ is not important, but at least one of the exponents needs to be larger than one to have a switch of the kind discussed below (Cherry and Adler, 2000). Greater cooperativity (larger n) enlarges the region of α_1-α_2 parameter space where bistability exists.]

By setting $\dot{u} = \dot{v} = 0$, one can easily see that, depending on the values of α_1 and α_2, there can be either one or three stationary solutions. For example, if

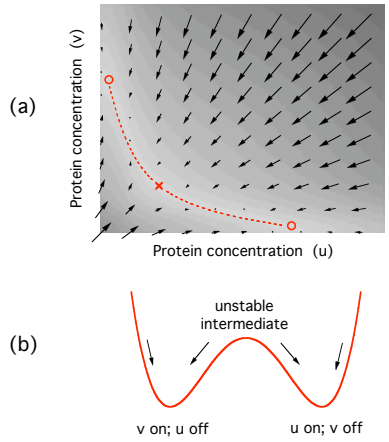

FIG. 33 (Color in online edition) Dynamics of a genetic toggle switch. (a) dynamics in the u-v concentration plane (arrows), with gray scale representing the magnitude of the vector field; (b) Equivalent dynamics on center manifold. The three equilibrium positions are represented by the two minima and the unstable intermediate state.

$\alpha_1 = \alpha_2 = \alpha \gg 1$, then the three solutions are (1) $u \approx \alpha$, $v \approx \frac{1}{\alpha}$; (2) $v \approx \alpha$, $u \approx \frac{1}{\alpha}$; and (3) $u \approx v \approx \alpha^{1/3}$. Solutions (1) and (2) are stable, while solution (3) is a saddle, with one stable and one unstable eigenvector. These three solutions are illustrated in Fig. 33a, where the two stable solutions are denoted by circles and the unstable solution by a cross. The vectors illustrate the local u-v dynamics. An arbitrary initial condition will relax quickly onto a "center manifold" – here, the one-dimensional dashed line connecting the three solutions. The dynamics will then occur along this curved line in parameter space. One can derive equations of motion for the dynamics along this center manifold: the system behaves like a particle in a double-well potential; c.f. Fig. 33b.

In the work by Gardner *et al.* (2000), the authors constructed a simple artificial genetic circuit (within the bacterium *E. coli*) that had the mutual-repression mechanism described above and were able to observe bistability. Examples from real biological systems are more complicated. For example, the *lac* operon is the prototype of a genetic switch, having been studied for some 50 years. (The details are not relevant here: briefly, *E. coli* can use glucose or lactose as a food source. Normally, it does not produce the enzyme to digest lactose, but the presence of a small amount of lactose switches on a gene that produces the necessary enzyme.) Even this prototypical example can be complicated. For example, Keener and Sneyd (1998) cite studies that begin with simplified dynamics of six coupled equations and then argue that these equations can be approximated by three others. Vilar *et al.* (2003) have argued that such simplified models need to incorporate stochastic fluctuations (because of the small number of molecules of the relevant species in each cell) to agree with observations. Very recently, Ozbudak *et al.* (2004) have explored the phase diagram of the lac operon in *E. coli* (modified by fus-

ing fluorescent "reporter" proteins to the genome). They make quantitative contact to the kinds of genetic-switch models discussed here.

Rather than entering into the details of how best to model a particular feedback mechanism, we want to emphasize merely that many, if not all, basic cell functions depend on interconnected positive and negative feedback loops. Indeed, it seems likely that such feedbacks are necessary in living organisms. For catalogs of such mechanisms, see, e.g., Freeman (2000); Keener and Sneyd (1998); for an analysis of elementary biochemical mechanisms (amplifications, etc.) that adopts an explicit engineering perspective, see Detwiler *et al.* (2000).

C. Network example

In the previous section, we saw that both positive and negative feedback loops are present in basic biochemical systems. In the cell, however, vast numbers of chemical species are present, constantly being synthesized, degraded, and otherwise interacting with each other. Thus, instead of a simple, isolated system modified by a single feedback loop, one has many interacting systems, connected to each other in a network. In such a network, the notion of a loop can be generalized to be the set of interconnections between nodes (the individual proteins), with positive and negative values assigned to each interconnection, depending on whether the presence of protein 1 increases or decreases the concentration of protein 2. (Here, "1" and "2" represent arbitrary proteins in the network). The structure of such networks is a topic of intense current interest (Albert and Barabási, 2002; Newman, 2003). Much attention has been focused on the statistical properties of large networks, for example on the distribution of the number of connections k a node has with its neighbors. Random networks have a distribution $P(k)$ peaked about an average number of interconnections $\langle k \rangle$ (Poisson distribution), while for *scale-free* networks, there is a power-law distribution $P(k) \sim k^{-\gamma}$, with γ typically have a value of 2 to 3. Scale-free networks have interesting properties and are found in many places, including communications settings (e.g., webpages and their links to each other), social interactions (e.g., collaborations among individuals, whether as actors in films, sexual partners, or scientific co-authors of papers), and in biological networks, as we will discuss more below (Albert and Barabási, 2002). Because of the power-law distribution of connections, a few important nodes ("hubs") will have many more than the average number of interconnections and play a central role in the network. They serve, first of all, to create the "small-world" phenomenon, where the number of steps needed to go between any two nodes increases only logarithmically with the number of nodes. They also give robustness to the structure of the networks: removing nodes other than one of the rare hubs will not affect substantially the connectivity of the network. But if scale-free networks are robust to the destruction of random nodes, they are fragile and sensitive to the destruction of hubs.

Much work in biology has been devoted to the identification of different types of networks, most notably metabolic and protein interaction networks (Jeong *et al.*, 2000). (Both have proteins as nodes. In the former, the interactions are chemical; in the latter, physical.) An equally important type of network is that of gene regulatory networks, which govern the production of the proteins involved in the metabolic and interaction networks (Maslov and Sneppen, 2002). This is in addition to more physical networks such as the networks of neurons in the brain and nervous system and the network of blood vessels. [For a review of all of these, see Newman (2003).]

The statistical point of view is not the only way to understand network structure. A number of authors have focused on the "modular" aspects of complex biological structures, with the goal of identifying the structures and interactions between relatively independent elements (Hartwell *et al.*, 1999). This has led to the search for "network motifs," (Mangan and Alon, 2003; Shen-Orr *et al.*, 2002), which are relatively simple clusters of nodes that behave as individual elements in a larger network.

If complex networks are generically present in biological systems, one might suspect that they confer some overall benefit to the host organisms, and one such hypothesized benefit that has lately received much attention is the notion of robustness. In influential work, Leibler and collaborators have looked at the relatively simple network involved in chemotaxis in the bacterium *E. coli* and shown, both in a model (Barkai and Leibler, 1997) and in experiment (Alon *et al.*, 1999) that certain properties show a remarkable robustness in the face of large concentration variations of elements within the cell. Chemotaxis in *E. coli* arises by controlling the time the bacterium spends in two states, "smooth runs" and "tumbling." During smooth runs, the cell swims in a relatively straight line. It then stops and begins tumbling, a motion that leads to a random reorientation. Then the cell swims for another period of time. The cell carries receptors for various chemical attractants. If the level of an attractant is rising, the cell will tend to swim longer before tumbling. In other words, by reducing the tumbling frequency (rate of tumbling events), the cell will tend to swim up the spatial gradient of attractor. Thus, chemotaxis occurs via a modulation of tumbling frequency. The kind of robustness explored by Leibler and collaborators looks at the adaptation of tumbling frequency to various changes. For example, the cell responds to gradients of chemical attractants in a way that is nearly independent of their absolute concentration. More precisely, the tumbling frequency should, after a transient, return to its original value after a sudden increase in the overall concentration of an attractant.

As Barkai and Leibler (1997) have emphasized, one can imagine two ways in which perfect adaptation can be obtained. One way involves a model that has fine-tuned parameter values that happen to lead to a canceling out of

the effects of a concentration increase. The problem with such a model is that it implies that adaptation would be a fragile phenomenon, easily disrupted by changes in any of the parameters of the system, which does not seem to be the case experimentally (Alon *et al.*, 1999). Alternatively, the robustness could be a property of the network itself. We have already seen examples where the level of an output in the face of a step-change disturbance. These all trace back to the use of integral feedback, and, indeed, Yi *et al.* (2000) have shown not only that integral feedback is present implicitly in the model of Barkai and Leibler but also that such feedback *must* be present in the dynamics. Rather than enter into the details of the chemotaxis model (even the "simplified" version of Yi *et al.* has 14 coupled equations), we sketch the proof that integral feedback must be present in order to have robust adaptation. We follow the appendix of (Yi *et al.*, 2000).

Consider first a linear SISO model

$$\begin{aligned} \dot{\vec{x}} &= \widetilde{A}\vec{x} + \vec{b}u \\ y &= \vec{c}^T\vec{x} + du \ , \end{aligned} \tag{7.13}$$

where $\vec{x}$ is an n-element internal state vector, y is the single output, and u is the single input. At steady state, $\dot{\vec{x}} = 0$, so that $y = (d - cA^{-1}b)u$. (We drop tildes and vector symbols for simplicity.) Then for constant input u, the output $y = 0$ if and only if either $c = d = 0$ (trivial case) or

$$\det \begin{bmatrix} A & b \\ c & d \end{bmatrix} = 0 \tag{7.14}$$

where the matrix in Eq. (7.14) has $n+1$ by $n+1$ elements. If the determinant is zero, then there is vector $\vec{k}$ such that $k[A \quad b] = [c \quad d]$. Defining $z = \vec{k}^T\vec{x}$, we have $\dot{z} = k\dot{x} = k(Ax + bu) = cx + du = y$. Thus, if $y = 0$ for all parameter variations (here, these would be variations in the elements of A, b, c, or d), then $\dot{z} = y$, a condition that is equivalent to having integral feedback be part of the structure of the system itself.

As Yi *et al.* (2000) comment and we have implicitly seen in this paper, the requirement of having integral feedback in order to reject step disturbances of parameters is a special case of the "Internal Model Principle," (IMP) which states that the controller must contain a model of the external signal in order to track robustly.[31] This is another motivation for the internal model control discussed in Sec. V.E. Finally, we note that the restriction to linear systems is not necessary (the biochemical models are all nonlinear). For example, consider the nonlinear dynamical system $\dot{x} = f(x, u)$, with x again an n-dimensional state vector and f a nonlinear function of x and the input u. If we want the output $y = x$ (for simplicity) to track a constant state r, we can set $u = \int^t (r - x)dt'$ (integral feedback). Then differentiating $\dot{x} = f$ shows that $x = r$ is a steady-state solution to the modified dynamics. One caveat about integral feedback is that the modified dynamical system must be stable. This must be verified case by case.

The chemotaxis example discussed above is just one of many instances where a complicated network of feedback interactions plays an important biological role. Another case considers neural networks, where the firings of one neuron stimulate or inhibit other connected neurons. Doiron *et al.* (2003) have recently shown that negative feedback plays an essential role in the neural response of electric fish to communication and prey signals. While the organism communicates with others, neurons in the sensory system studied switch to a globally synchronized oscillatory state that is maintained by negative feedback. At other times, the firings are not synchronized.

VIII. OTHER APPLICATIONS, OTHER APPROACHES

At this point, it is perhaps proper to note some of the many areas that, for reasons of space but not for lack of interest, we have decided not to explore. As our discussion of biological networks suggests, complex systems can often be modeled as an interacting network, and complex networks often have a modular structure whose function can be understood by appeal to elementary feedback motifs. Thus, one finds feedback to be a relevant concept in understanding the weather, climate change, and in economics, etc. For example, many of the arguments in the field of global warming amount to debates about the magnitudes and sizes of feedback effects. While the link between industrial and other human activity and the increased amount of CO_2 and other greenhouse gasses is clear, the climatic effects are much harder to predict, because of the complexity of feedback effects. There is a tremendous flux of carbon between the atmosphere and various "sinks," such as the oceans and forests, and the dependence of these fluxes on greenhouse gasses must be evaluated accurately to know the cumulative effect on climate (Sarmiento and Gruber, 2002). Even more disturbing, it has become clear that small changes have in the past led to rapid climate shifts – on the timescale of decades, or even less. Models of thermohaline circulation in the oceans show bistable behavior analogous to the genetic switch discussed above, where positive feedback effects can toggle the climate between warm and icy states (Weart, 2003).

One rather distinct area left out is the application of control principles to quantum phenomena (Rice and

[31] A more precise statement is that if all disturbances $d(t)$ satisfy a *known* differential equation of the form $\sum_{i=0}^{n} p_i d^{(i)} = 0$, then one can design a control system that perfectly tracks the disturbances. Here, $d^{(i)}$ is the i^{th} time derivative of $d(t)$, and the p_i are known, constant coefficients. For example, all step disturbances satisfy $d^{(1)} = 0$, while oscillatory disturbances satisfy $d^{(2)} + \omega_d^2 d^{(2)} = 0$. The IMP states that disturbances or reference signals are canceled in steady state if their transfer function is contained in the denominator of the controller $K(s)$. See Lewis (1992), Ch. 4.

Zhao, 2000; Shapiro and Brumer, 2003; Walmsley and Rabitz, 2003). The basic insight is that adding a coherent perturbation to a system can enhance or suppress the amplitude of desired "product" states. The famous two-slit experiment of Young, where two beams of light interfere to produce light and dark fringes, is an elementary example. Much of the practical work has used coherent light beams interacting with matter to enhance or suppress phenomena such as dissociation and chemical reaction. Feedback often enters here in only a weak way, where one conducts repeated trials, using the results to adaptively tune experimental control parameters such as the amplitude and phase of different frequency components in a shaped pulse; however, some recent work has emphasized real-time corrections, where the ideas discussed here, including state estimation (Kalman filtering) and robustness, are beginning to be explored (Wiseman *et al.*, 2002). Many of these approaches are based on adaptive optics, a technique that has many other applications – for example, compensating for turbulence in astronomy, where one changes rapidly the shape of a mirror to remove phase disturbances added by atmospheric fluctuations (Roggemann *et al.*, 1999). Finally, most of the work to date on quantum systems has been "semiclassical," in that sensors perform measurements on a system, classical computers process the results, and semiclassical fields are used to influence the future evolution of the system. S. Lloyd has emphasized that one can imagine a fully quantum feedback scheme wherein one adds an element to the quantum system that interacts (without making a measurement) with it in such a way that the dynamics are altered in a desired way (Lloyd, 2000). Because information is not destroyed (as it is in the measurement stage of the semiclassical scheme), higher performance is in principle possible. But this field is still young.

There are also many valuable topics in control theory that we do not have space to discuss. Foremost among these is adaptive control, which can be considered as a complement to the approach of robust control, discussed in Sec. V.E. In both cases, the system to be controlled is at least partially unknown. In robust control, one first tries to describe the set of possible transfer functions of close linear systems and then tries to find a controller that is stable in all situations and still performs adequately, even under a worst-case scenario. In essence, this is a conservative approach, which works best for smaller sets of transfer functions. (If the variations are too large, trying to satisfy all the constraints all the time will lead to very weak control.) Alternatively, one can try to "learn" which of the possible set of transfer functions best describes the system at the present time and to design a control law for the current best estimate of the dynamical system. This is the overall approach of adaptive control, which in its simplest forms treats topics such as autotuning of parameters for simple control loops (e.g., PID loops) (Dutton *et al.*, 1997; Franklin *et al.*, 1998). The basic algorithms are akin to the Kalman filter, in that model parameters are estimated using a recursive version of least-squares fitting that updates the parameter estimates at each time step. In more sophisticated analyses, the controller's structure can be varied, as well(Isermann *et al.*, 1992). Adaptation and "learning" are used also in approaches from computer science that include the genetic algorithms mentioned above (Jamshidi *et al.*, 2003), neural networks (Norgaard *et al.*, 2000), fuzzy logic (Verbruggen *et al.*, 1999), and "intelligent" control systems (Hangos *et al.*, 2001). Broadly speaking, all of these are attempts to mimic the judgment of an experienced human operator manually controlling a system. Of course, they have applications far beyond problems of control. For most physicists, the more straightforward control techniques described in this review will be more than adequate.

IX. FEEDBACK AND INFORMATION THEORY

In various places in the discussion above, we have noted the informal connection between information and control. For example, in Sec. III.A, we saw how the strategy of feedforward relied on prior information about the nature of a disturbance. In Sec. V.E.1, we saw that feedback is required only when our knowledge of the system and its disturbances is incomplete. Also, in Sec. V.D, we saw that the Kalman filter gives the optimal way to blend the two pieces of information one has about the current state of a linear dynamical system subject to white noise perturbations and to white-noise in the measurements. That way best blends the actual measurement and the prediction based on the system dynamics.

In all these discussions, the use of the term "information" has been informal, but it is natural to wonder whether there are links to the technical subject of "information theory." Indeed, the influential book, *Cybernetics*, by Wiener (first published in 1948, the year of Shannon's fundamental papers on information theory) explored some connections (Wiener, 1961). Still, there has been very little development of these ideas. Recent papers by Touchette and Lloyd (2000, 2004) begin to explore more formally these links and derive a fundamental relationship between the amount of control achievable ("decrease of entropy" in their formulation) and the "mutual information" (Cover and Thomas, 1991) between the dynamical system and the controller created by an initial interaction. They show that if one measures the accuracy of control by the statistical reduction of uncertainty about the state of the controlled object, then that reduction is limited by the information that measurements can extract about the system, in conjunction with any improvements that open-loop control can offer. In their formulation, information becomes the common language for both measurement and control. The full implications of their results and how they connect to the "optimal estimation theory" of Kalman discussed here have yet to

be worked out.[32]

It is interesting that while the link between feedback and information theory has been developing almost since the beginning of the subject itself, the connection has been slow to sink in and difficult to grasp. As Bennett (1993) has noted, pre-20th-century controllers were almost invariably "direct" acting, with the force required to effect a change on the system developed by the measuring device itself. For example, the buoyant force on a ball directly controls the water valve in a toilet. It took a long time for engineers to recognize that the sensor and the actuator were logically distinct devices and that the function of the controller was just to process the information gained through the sensor, converting it to a response by the actuator.

X. CONCLUSIONS

One of the main goals of this article has been to give a pragmatic introduction to control theory and the basic techniques of feedback and feedforward. We have discussed a broad range of applications, including details of practical implementation of feedback loops. We have emphasized basic understanding and outlined the starting points for more advanced methods. We have tried in many places to show how a bit of thought about the design of the physical system can reduce the demands on the controller. (Remember to limit time lags by putting sensors close to actuators!) We have also emphasized that it often pays to spend a bit more on equipment and less on fancy control-loop design. We have argued that a mix of feedforward and feedback can work much better than either technique alone. (To anticipate is better than to react, but cautious anticipation is better still.) Taken together, the techniques outlined here are probably more sophisticated, and more systematic, than what is commonly practiced among physicists. I certainly hope that this article "raises the consciousness" and perhaps even the level of practice of physicists as regards feedback loops. And if higher performance is needed in an application, one can, and should, consult the professional control-theory books, which I hope will now be more accessible.

At the same time, we have explored some of the deeper implications of control theory. We have seen how feedback loops and other complicated networks play a fundamental role in complex systems encountered in nature, particular in biology. We have seen how information is the "common coin" of measurements and that feedback can be thought of in terms of information flows, from system to controller and back again in the simplest case, or from node to node for more complicated networks. What is interesting is how the reality of the physical system itself begins to fade from the picture once control loops are implemented. In simple cases, this occurs when we use feedback to speed up (Sec. III.A) or slow down dynamics (Sec. V.C.1), implying that one can use feedback to compensate for the physical limitations of the particular elements one may have at hand. Certainly, some of the simpler applications of feedback depend on such abilities.

Other, deeper applications of control depend on the robustness that feedback can lead to. We saw this in integral control, where tracking is achieved over a wide range of control parameters, and it is the loop structure that ensures the desired result. Such feedback is, of course, at the heart of the PID loops beloved in industrial applications. But we saw that it also exists in mathematical and physical models of biological networks where feedback leads to a robustness that begins to approximate the behavior of real biological systems.

What is perhaps most interesting to a physicist is the way new kinds of behavior arise from the structure of control loops. The tracking property of integral feedback comes from the structure of the feedback loop, not the nature of the individual elements. In this sense, it is a kind of "emergent phenomenon," but one that differs from the examples familiar in physics, such as the soft modes and phase rigidity that accompany symmetry-breaking phase transitions. Thus, engineering provides an alternate set of archetypes for emergent phenomena, which – as Carlson and Doyle (2002); Csete and Doyle (2002); Kitano (2002) have argued – will perhaps be more fruitful for understanding biological mechanisms than physics-inspired archetypes. My own view is that one should not be too dogmatic about which discipline provides the better model, for in the past, physicists have often been successful precisely because they were willing to try to solve the problem at hand "by any means necessary."

Still, the easy integration of certain engineering concepts into common knowledge is striking. We physicists are justly proud of having developed concepts such as "work," "energy," "momentum," "resonance," and even "entropy" that have entered the popular language as metaphors and are used by most people with confidence even when the technical meaning is obscure. More recently, engineering terms as "bits," "information," and "feedback," which also treat topics that physicists deal with, have become equally familiar. If we take up the last of these terms, the word "feedback" is less than a century old and was coined to describe an advance in radio technology (Simpson and Weiner, 1989). In popular language, it quickly went from a novelty to a fad to a word so often used that one can scarcely imagine its recent origin. Feedback is a useful technical tool that underlies much of modern technology, but it is also an essential ingredient of biological systems and even of life

[32] One link between the Kalman filter and information theory is via the use of Bayes' theorem in constructing the optimal estimate (Maybeck, 1979); however the work of Touchette and Lloyd (2000) implies deeper connections. Links to areas such as the robust methods of Sec. V.E are even less clear. See Kåhre (2002), Ch. 12, for other ideas on how information theory can be related to control issues.

itself. Is it then merely by chance that "feedback," despite its narrow technical origins, has found such ready acceptance and wide use in popular culture?

Acknowledgments

I thank Vincent Fourmond, Russ Greenall, Christoph Hebeisen, Laurent Talon, Bloen Metzger, and Anand Yethiraj for collaboration on some of the experimental work cited here. I am grateful for comments by Adrienne Drobnies, Ted Forgan, Bruce Francis, Jean-Christophe Géminard, Albert Libchaber, Andrew Lewis, and Juan Restrepo, and for support by NSERC (Canada).

List of abbreviations

A/D	analog to digital	V.C
AFM	atomic force microscope	III.B
D/A	digital to analog	VI.A
DNA	deoxyribonucleic acid	VII.B.1
DSP	digital signal processor	V.C
EKF	extended Kalman filter	V.D
FPGA	field-programmable gate array	V.C
IIR	infinite impulse response	V.C
IMC	internal model control	V.E.1
IMP	internal model principle	VII.C
LHP	left-hand plane	V.B.2
LIGO	Laser Interferometer Gravitational Wave Observatory	III.D.1
LQG	linear quadratic Gaussian	V.D
LSB	least-significant bit	VI.A
MIMO	multiple input, multiple output	III.C
mRNA	messenger RNA	VII.B.1
NMP	non-minimum phase	IV.D
OGY	Ott-Grebogi-Yorke	VI.B
PI	proportional-integral	V.A.4
PID	proportional-integral-derivative control law	I
PLD	programmable logic device	V.C
PRBS	pseudorandom binary sequence	V.A.1
RHP	right-hand plane	IV.D
RMS	root-mean square	V.A.4
RNA	ribonucleic acid	VII.B.1
RNAP	RNA polymerase	VII.B.1
SISO	single input, single output	III.C
SPM	scanning probe microscope	III.B
STM	scanning tunneling microscope	III.B
SVD	singular-value decomposition	IV.E
SVF	state-vector feedback	III.C
TetR	tetracycline repressor	VII.B.1
ZOH	zero-order hold	IV.C

References

Albert, R., and A.-L. Barabási, 2002, "Statistical mechanics of complex networks," Rev. Mod. Phys. **74**, 47–97.

Alon, U., M. G. Surette, N. Barkai, and S. Leibler, 1999, "Robustness in bacterial chemotaxis," Nature (London) **397**, 168–171.

Barkai, N., and S. Leibler, 1997, "Robustness in simple biochemical networks," Nature (London) **387**, 913–917.

Becksei, A., and L. Serrano, 2000, "Engineering stability in gene networks by autoregulation," Nature (London) **405**, 590–593.

Bennett, S., 1993, "Development of the PID controller," IEEE Contr. Syst. Mag. **13**(6), 58–65.

Bennett, S., 1996, "A brief history of automatic control," IEEE Contr. Syst. Mag. **16**(3), 17–25.

Bennett, S., 2002, "Otto Mayr: Contributions to the history of feedback control," IEEE Contr. Syst. Mag. **22**(2), 29–33.

Black, H. S., 1934, "Stabilized feedback amplifiers," Bell. Syst. Tech. J. **13**, 1–18.

Bode, H. W., 1945, *Network Analysis and Feedback Amplifier Design* (D. van Nostrand and Co., New York).

Borcherding, F., S. Grünendahl, M. Johnson, M. Martin, J. Olsen, and K. Yip, 1999, "A first-level tracking trigger for the upgraded D0 detector," IEEE Trans. Nucl. Sci. **46**, 359–364.

Carlson, J. M., and J. Doyle, 2002, "Complexity and robustness," Proc. Natl. Acad. Sci. USA **99**, 2538–2545.

Chaikin, P. M., and T. C. Lubensky, 1995, *Principles of Condensed Matter Physics* (Cambridge Univ. Press, Cambridge, UK).

Cherry, J. L., and F. R. Adler, 2000, "How to make a biological switch," J. Theor. Biol. **203**, 117–133.

Cooper, W. S., 1986, "Use of optimal estimation theory, in particular the Kalman filter, in data processing and signal analysis," Rev. Sci. Instr. **57**, 2862–2869.

Cover, T., and J. Thomas, 1991, *Elements of Information Theory* (John Wiley & Sons, Inc., New York).

Csete, M. E., and J. C. Doyle, 2002, "Reverse Engineering of Biological Complexity," Science **295**, 1664–1669.

Detwiler, P. B., S. Ramanathan, A. Sengupta, and B. I. Shraiman, 2000, "Engineering aspects of enzymatic signal transduction: photoreceptors in the retina," Biophys. J. **79**, 2801–2817.

Doiron, B., M. J. Chacron, L. Maler, A. Longtin, and J. Bastian, 2003, "Inhibitory feedback required for network oscillatory responses to communication but not prey stimuli," Nature (London) **421**, 539–543.

Doyle, J. C., B. A. Francis, and A. R. Tannenbaum, 1992, *Feedback Control Theory* (Macmillan Publishing Co., New York), URL `http://www.control.utoronto.ca/people/profs/francis/dft.html`.

Dutton, K., S. Thompson, and B. Barraclough, 1997, *The Art of Control Engineering* (Addison-Wesley, Harlow, England).

Etchenique, R., and J. Aliaga, 2004, "Resolution enhancement by dithering," Am. J. Phys. **72**, 159–163.

Evensen, G., 2003, "The ensemble kalman filter: theoretical formulation and practical implementation," Ocean Dyn. **53**, 343–367.

Eyink, G. L., J. M. Restrepo, and F. J. Alexander, 2004, "A mean field approximation in data assimilation for nonlinear dynamics," Physica D **195**, 347–368.

Forgan, E. M., 1974, "On the use of temperature controllers in cryogenics," Cryogenics **14**, 207–214.

Franklin, G. F., J. D. Powell, and A. Emami-Naeini, 2002, *Feedback Control of Dynamical Systems* (Prentice Hall, Upper Saddle River, NJ), 4th edition.

Franklin, G. F., J. D. Powell, and M. Workman, 1998, *Digital Control of Dynamical Systems* (Addison-Wesley, Reading, Mass.).

Freeman, M., 2000, "Feedback control of intercellular signalling in development," Nature (London) **408**, 313–319.

Gamkrelidze, R. V., 1999, "Discovery of the maximum principle," J. of Dyn. and Contr. Syst. **5**, 437–451.

Gardner, T. S., C. R. Cantor, and J. J. Collins, 2000, "Construction of a genetic toggle switch in *Escherichia coli*," Nature (London) **405**, 520–521.

Gardner, T. S., and J. J. Collins, 2000, "Neutralizing noise in gene networks," Nature (London) **403**, 339–342.

Gershenfeld, N., 1999, *The Nature of Mathematical Modeling* (Cambridge Univ. Press, Cambridge, UK).

Gershenfeld, N., 2000, *The Physics of Information Technology* (Cambridge Univ. Press, Cambridge, UK).

Goldenfeld, N., 1992, *Lectures on Phase Transitions and the Renormalization Group* (Perseus, Reading, MA).

Gray, M., D. McClelland, M. Barton, and S. Kawamura, 1999, "A simple high-sensitivity interferometric position sensor for test mass control on an advanced LIGO interferometer," Optical and Quantum Electronics **31**, 571–582.

Grebogi, C., and Y.-C. Lai, 1999, in *Handbook of Chaos Control*, edited by H. G. Schuster (Wiley-VCH, New York), 1–20.

Hangos, K. M., R. Lakner, and M. Gerzson, 2001, *Intelligent Control Systems: An Introduction with Examples* (Kluwer Academic Publishers, Dordrecht; Boston).

Hartwell, L. H., J. J. Hopfield, S. Leibler, and A. W. Murray, 1999, "From molecular to modular biology," Nature (London) **402**, C47–C52.

Hensley, J. M., A. Peters, and S. Chu, 1999, "Active low frequency vertical vibration isolation," Rev. Sci. Instr. **70**, 2735–2741.

Isermann, R., K.-H. Lachmann, and D. Matko, 1992, *Adaptive Control Systems* (Prentice Hall, New York).

Ishida, F., and Y. E. Sawada, 2004, "Human hand moves proactively to the external stimulus: An evolutional strategy for minimizing transient error," Phys. Rev. Lett. **93**(168105), 1–4.

Jamshidi, M., L. dos Santos Coelho, R. A. Krohling, and P. J. Fleming, 2003, *Robust Control Systems with Genetic Algorithms* (CRC Press, Boca Raton, FL).

Jeong, H., B. Tombor, R. Albert, Z. N. Oltvai, and A.-L. Barabási, 2000, "The large-scale organization of metabolic networks," Nature (London) **407**, 651–654.

Kaern, M., W. J. Blake, and J. J. Collins, 2003, "The engineering of gene regulatory networks," Annu. Rev. Biomed. Eng. **5**, 179–206.

Kåhre, J., 2002, *The Mathematical Theory of Information* (Kluwer Academic Publishers, Boston).

Kalman, R. E., 1960, "A new approach to linear filtering and prediction problems," ASME J. Basic Engineering **82**, 35–45.

Keener, J., and J. Sneyd, 1998, *Mathematical Physiology* (Springer-Verlag, New York).

Kitano, H., 2002, "Looking beyond the details: a rise in systems-oriented approaches in genetics and molecular biology," Curr. Genet. **41**, 1–10.

Leigh, J. R., 2004, *Control Theory* (The Institution of Electrical Engineers, London), 2nd edition.

Lewis, F. L., 1992, *Applied Optimal Control and Estimation* (Prentice Hall, Englewood Cliffs, NJ).

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

Lloyd, S., 2000, "Coherent quantum feedback," Phys. Rev. A **62**, 022108.

Longtin, A., and J. Milton, 1989a, "Insight into the transfer function, gain, and oscillation onset for the pupil light reflex using nonlinear delay-differential equations," Biol. Cybern. **61**, 51–58.

Longtin, A., and J. Milton, 1989b, "Modelling autonomous oscillations in the human pupil light reflex using non-linear delay-differential equations," Bull. Math. Biol. **51**, 605–624.

Mancini, R., 2002, *Op Amps for Everyone: Texas Instruments guide*, Technical Report, Texas Instruments, URL `http://focus.ti.com/lit/an/slod006b/slod006b.pdf`.

Mangan, S., and U. Alon, 2003, "Structure and function of the feed-forward loop network motif," Proc. Natl. Acad. Sci. USA **100**, 11980–11985.

Maslov, S., and K. Sneppen, 2002, "Specificity and stability in topology of protein networks," Science **296**, 910–913.

Maybeck, P. S., 1979, *Stochastic Models, Estimation, and Control*, volume 1 (Academic Press, New York).

Mayr, O., 1970, "The origins of feedback control," Sci. Am. **223**(4), 110–118.

Metzger, B., 2002, *Intensity fluctuation microscopy applied to the nematic–smectic-A phase transition*, Thèse de magistère, Ecole Normale Supérieure de Lyon.

Monod, J., and F. Jacob, 1961, "General conclusions: Teleonomic mechanisms in cellular metabolism, growth, and differentiation," Cold Spring Harbor Symp. Quant. Biol. **26**, 389–401.

Morari, M., and E. Zafirioiu, 1989, *Robust Process Control* (Prentice Hall, Upper Saddle River, NJ).

Morris, K., 2001, *Introduction to Feedback Control Theory* (Academic Press, San Diego).

Naidu, D. S., 2003, *Optimal Control Systems* (CRC Press, Boca Raton, FL).

Newman, M. E. J., 2003, "The structure and function of complex networks," SIAM Review **45**, 167–256.

Norgaard, M., O. Ravn, N. K. Poulsen, and L. K. Hansen, 2000, *Neural Networks for Modelling and Control of Dynamic Systems: A Practitioner's Handbook* (Springer Verlag).

Nyquist, H., 1932, "Regeneration theory," Bell. Syst. Tech. J. **11**, 126–147.

Oliva, A. I., E. Angulano, N. Denisenko, and M. Aguilar, 1995, "Analysis of scanning tunneling microscopy feedback system," Rev. Sci. Instr. **66**, 3195–3203.

Oppenheim, A. V., R. W. Schafer, and J. R. Buck, 1992, *Discrete-time signal processing* (Prentice Hall, Upper Saddle River, NJ), 2nd edition.

Ott, E., C. Grebogi, and J. A. Yorke, 1990, "Controlling chaos," Phys. Rev. Lett. **64**, 1196–1199.

Özbay, H., 2000, *Introduction to Feedback Control Theory* (CRC Press, Boca Raton).

Ozbudak, E. M., M. Thattai, H. N. Lim, B. I. Shraiman, and A. van Oudenaarden, 2004, "Multistability in the lactose utilization network of *Escherichia coli*," Nature (London) **427**, 737–740.

Petersen, I. R., and A. V. Savkin, 1999, *Robust Kalman Fil-*

tering for Signals and Systems with Large Uncertainties (Birkhäuser, Boston).

Pontryagin, L. S., V. G. Boltyanskii, R. V. Gamkrelidze, and E. F. Mishchenko, 1964, *The Mathematical Theory of Optimal Processes* (The Macmillan Co., New York).

Press, W. H., B. P. Plannery, S. A. Teukolsky, and W. T. Vetterling, 1993, *Numerical Recipes in C: The Art of Scientific Computing* (Cambridge Univ. Press, Cambridge, UK), 2nd edition.

Reynolds, D. E., 2003, "Coarse graining and control theory model reduction," cond-mat/0309116 .

Rice, S. A., and M. Zhao, 2000, *Optical Control of Molecular Dynamics* (Wiley-Interscience, New York).

Roggemann, M. C., B. M. Welsh, and R. Q. Fugate, 1999, "Improving the resolution of ground-based telescopes," Rev. Mod. Phys. **69**, 437–505.

Rosenfeld, N., M. Elowitz, and U. Alon, 2002, "Negative autoregulation speeds the response times of transcription networks," J. Mol. Biol. **323**, 785–793.

Salapaka, S., A. Sebastian, J. P. Cleveland, and M. V. Salapaka, 2002, "High bandwidth nano-positioner: A robust control approach," Rev. Sci. Instr. **73**, 3232–3241.

Sarmiento, J. L., and N. Gruber, 2002, "Sinks for anthropogenic carbon," Phys. Today **55**(8), 30–36.

Schitter, G., F. Allgöwer, and A. Stemmer, 2004, "A new control strategy for high-speed atomic force microscopy," Nanotechnology **15**, 108–114.

Schitter, G., P. Menold, H. F. Knapp, F. Allgöwer, and A. Stemmer, 2001, "High performance feedback for fast scanning atomic force microscopes," Rev. Sci. Instr. **72**, 3320–3327.

Schuster, H. G. (ed.), 1999, *Handbook of Chaos Control* (Wiley-VCH, New York).

Shapiro, M., and P. Brumer, 2003, *Principles of the Quantum Control of Molecular Processes* (Wiley-Interscience, New York).

Shen-Orr, S. S., R. Milo, S. Mangan, and U. Alon, 2002, "Network motifs in the transcriptional regulation network of *Escherichia Coli*," Nat. Gen. **31**, 64–68.

Shinbrot, T., 1999, in *Handbook of Chaos Control*, edited by H. G. Schuster (Wiley-VCH, New York), 157–180.

Simpson, J., and E. Weiner (eds.), 1989, *Oxford English Dictionary* (Clarendon Press, Oxford), 2nd edition.

Skogestad, S., and I. Postlethwaite, 1996, *Multivariable Feedback Control* (John Wiley and Sons, Chichester, UK).

Sontag, E. D., 1998, *Mathematical Control Theory* (Springer Verlag, New York), 2nd edition.

Strogatz, S. H., 1994, *Nonlinear Dynamics and Chaos* (Addison-Wesley, Reading, Mass.).

Sussmann, H. J., and J. C. Willems, 1997, "300 years of optimal control," Control. Syst. Mag. **17**, 32–44.

Tamayo, J., A. D. L. Humphris, and M. J. Miles, 2000, "Piconewton regime dynamic force microscopy in liquid," Appl. Phys. Lett. **77**, 582–584.

Touchette, H., and S. Lloyd, 2000, "Information-theoretic limits of control," Phys. Rev. Lett. **84**, 1156–1159.

Touchette, H., and S. Lloyd, 2004, "Information-theoretic approach to the study of control systems," Physica A **331**, 140–172.

Verbruggen, H. B., H.-J. Zimmermann, and R. Babuska (eds.), 1999, *Fuzzy Algorithms for Control* (Kluwer Academic Publishers, Boston).

Vidyasagar, M., 1986, "On undershoot and nonminimum phase zeros," IEEE Trans. Auto. Cont. **31**, 440.

Vilar, J. M. G., C. C. Guet, and S. Leibler, 2003, "Modeling network dynamics: the *lac* operon, a case study," J. Cell. Biol. **161**, 471–476.

Walmsley, I., and H. Rabitz, 2003, "Quantum physics under control," Phys. Today **56**(8), 43–49.

Weart, S., 2003, "The discovery of rapid climate change," Phys. Today **56**(8), 30–36.

Wiener, N., 1961, *Cybernetics: Control and Communication in the Animal and the Machine* (The MIT Press, Cambridge, MA), 2nd edition.

Wiseman, H. M., S. Mancini, and J. Wang, 2002, "Bayesian feedback versus Markovian feedback in a two-level atom," Phys. Rev. A **66**, 013807.

Wolf, D. M., and F. H. Eeckman, 1998, "On the relationship between genomic regulatory element organization and gene regulatory dynamics," J. Theor. Biol. **195**, 167–186.

Yethiraj, A., R. Mukhopadhyay, and J. Bechhoefer, 2002, "Two experimental tests of a fluctuation-induced first-order phase transition: Intensity fluctuation microscopy at the nematic–smectic-A transition," Phys. Rev. E **65**, 021702.

Yi, T.-M., Y. Huang, M. I. Simon, and J. Doyle, 2000, "Robust perfect adaptation in bacterial chemotaxis through integral feedback control," Proc. Natl. Acad. Sci. USA **97**, 4649–4653.