

Chapter 2

Control Theory Dynamics



The mathematics of classical control theory depends on linear ordinary differential equations, which commonly arise in all scientific disciplines. Control theory emphasizes a powerful Laplace transform expression of linear differential equations. The Laplace expression may be less familiar in particular disciplines, such as theoretical biology.

2.1 Transfer Functions and State Space

Here, I show how and why control applications use the Laplace form. I recommend an introductory text on control theory for additional background and many example applications (e.g., Åström and Murray 2008; Ogata 2009; Dorf and Bishop 2016).

Suppose we have a process, P , that transforms a command input, u , into an output, y . Figure 2.1a shows the input–output flow. Typically, we write the process as a differential equation, for example

$$\ddot{x} + a_1\dot{x} + a_2x = \dot{u} + bu, \tag{2.1}$$

in which $x(t)$ is an internal state variable of the process that depends on time, $u(t)$ is the forcing command input signal, and overdots denote derivatives with respect to time. Here, for simplicity, we let the output be equivalent to the internal state, $y \equiv x$.

The dynamics of the input signal, u , may be described by another differential equation, driven by reference input, r (Fig. 2.1b). Mathematically, there is no problem cascading sequences of differential equations in this manner. However, the rapid growth of various symbols and interactions make such cascades of differential equations difficult to analyze and impossible to understand intuitively.

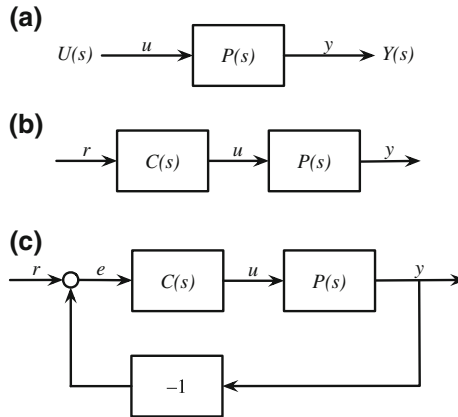


Fig. 2.1 Basic process and control flow. **a** The input–output flow in Eq. 2.2. The input, $U(s)$, is itself a transfer function. However, for convenience in diagramming, lowercase letters are typically used along pathways to denote inputs and outputs. For example, in **a**, u can be used in place of $U(s)$. In **b**, only lowercase letters are used for inputs and outputs. Panel **b** illustrates the input–output flow of Eq. 2.3. These diagrams represent open-loop pathways because no closed-loop feedback pathway sends a downstream output back as an input to an earlier step. **c** A basic closed-loop process and control flow with negative feedback. The circle between r and e denotes addition of the inputs to produce the output. In this figure, $e = r - y$

We can use a much simpler way to trace input–output pathways through a system. If the dynamics of P follow Eq. 2.1, we can transform P from an expression of temporal dynamics in the variable t to an expression in the complex Laplace variable s as

$$P(s) = \frac{Y(s)}{U(s)} = \frac{s + b}{s^2 + a_1s + a_2}. \quad (2.2)$$

The numerator simply uses the coefficients of the differential equation in u from the right side of Eq. 2.1 to make a polynomial in s . Similarly, the denominator uses the coefficients of the differential equation in x from the left side of Eq. 2.1 to make a polynomial in s . The eigenvalues for the process, P , are the roots of s for the polynomial in the denominator. Control theory refers to the eigenvalues as the *poles* of the system.

From this equation and the matching picture in Fig. 2.1, we may write $Y(s) = U(s)P(s)$. In words, the output signal, $Y(s)$, is the input signal, $U(s)$, multiplied by the transformation of the signal by the process, $P(s)$. Because $P(s)$ multiplies the signal, we may think of $P(s)$ as the signal gain, the ratio of output to input, Y/U . The signal gain is zero at the roots of the numerator’s polynomial in s . Control theory refers to those numerator roots as the *zeros* of the system.

The simple multiplication of the signal by a process means that we can easily cascade multiple input–output processes. For example, Fig. 2.1b shows a system with extended input processing. The cascade begins with an initial reference input, r , which is transformed into the command input, u , by a preprocessing controller, C , and then finally into the output, y , by the intrinsic process, P . The input–output calculation for the entire cascade follows easily by noting that $C(s) = U(s)/R(s)$, yielding

$$Y(s) = R(s)C(s)P(s) = R(s)\frac{U(s)}{R(s)}\frac{Y(s)}{U(s)}. \quad (2.3)$$

These functions of s are called *transfer functions*.

Each transfer function in a cascade can express any general system of ordinary linear differential equations for vectors of state variables, x , and inputs, u , with dynamics given by

$$\begin{aligned} x^{(n)} + a_1x^{(n-1)} + \cdots + a_{n-1}x^{(1)} + a_nx \\ = b_0u^{(m)} + b_1u^{(m-1)} + \cdots + b_{m-1}u^{(1)} + b_mu, \end{aligned} \quad (2.4)$$

in which parenthetical superscripts denote the order of differentiation. By analogy with Eq. 2.2, the associated general expression for transfer functions is

$$P(s) = \frac{b_0s^m + b_1s^{m-1} + \cdots + b_{m-1}s + b_m}{s^n + a_1s^{n-1} + \cdots + a_{n-1}s + a_n}. \quad (2.5)$$

The actual biological or physical process does not have to include higher-order derivatives. Instead, the dynamics of Eq. 2.4 and its associated transfer function can always be expressed by a system of first-order processes of the form

$$\dot{x}_i = \sum_j a_{ij}x_j + \sum_j b_{ij}u_j, \quad (2.6)$$

which allows for multiple inputs, u_j . This system describes the first-order rate of change in the state variables, \dot{x}_i , in terms of the current states and inputs. This state-space description for the dynamics is usually written in vector notation as

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{Ax} + \mathbf{Bu} \\ \mathbf{y} &= \mathbf{Cx} + \mathbf{Du}, \end{aligned}$$

which potentially has multiple inputs and outputs, \mathbf{u} and \mathbf{y} .

For example, the single input–output dynamics in Eq. 2.1 translate into the state-space model

$$\begin{aligned}\dot{x}_1 &= -a_2x_2 + bu \\ \dot{x}_2 &= x_1 - a_1x_2 + u \\ y &= x_2,\end{aligned}$$

in which the rates of change in the states depend only on the current states and the current input.

2.2 Nonlinearity and Other Problems

Classical control theory focuses on transfer functions. Those functions apply only to linear, time-invariant dynamics. By contrast, state-space models can be extended to any type of nonlinear, time-varying process.

Real systems are typically nonlinear. Nonetheless, four reasons justify the study of linear theory.

First, linear analysis clarifies fundamental principles of dynamics and control. For example, feedback often leads to complex, nonintuitive pathways of causation. Linear analysis has clarified the costs and benefits of feedback in terms of trade-offs between performance, stability, and robustness. Those principles carry over to nonlinear systems, although the quantitative details may differ.

Second, many insights into nonlinear aspects of control come from linear theory (Isidori 1995; Khalil 2002; Astolfi et al. 2008). In addition to feedback, other principles include how to filter out disturbances at particular frequencies, how time delays alter dynamics and the potential for control, how to track external setpoints, and how to evaluate the costs and benefits of adding sensors to monitor state and adjust dynamics.

Third, linear theory includes methods to analyze departures from model assumptions. Those linear methods of robustness often apply to nonlinear departures from assumed linearity. One can often analyze the bounds on a system's performance, stability, and robustness to specific types of nonlinear dynamics.

Fourth, analysis of particular nonlinear systems often comes down to studying an approximately linearized version of the system. If the system state remains near an equilibrium point, then the system will be approximately linear near that point. If the system varies more widely, one can sometimes consider a series of changing linear models that characterize the system in each region. Alternatively, a rescaling of a nonlinear system may transform the dynamics into a nearly linear system.

Given a particular nonlinear system, one can always simulate the dynamics explicitly. The methods one uses to understand and to control a simulated system arise mostly from the core linear theory and from the ways that particular nonlinearities depart from that core theory.

2.3 Exponential Decay and Oscillations

Two simple examples illustrate the match between standard models of dynamics and the transfer function expressions. First, the simplest first-order differential equation in $x(t)$ forced by the input $u(t)$, with initial condition $x(0) = 0$, is given by

$$\dot{x} + ax = u, \quad (2.7)$$

which has the solution

$$x(t) = \int_0^t e^{-a\tau} u(t - \tau) d\tau. \quad (2.8)$$

This process describes how x accumulates over time, as inputs arrive at each time point with intensity u , and x decays at rate a .

If the input into this system is the impulse or Dirac delta function, $u(t)dt = 1$ at $t = 0$ and $u(t) = 0$ for all other times, then

$$x(t) = e^{-at}.$$

If the input is the unit step function, $u(t) = 1$ for $t \geq 0$ and $u(t) = 0$ for $t < 0$, then

$$x(t) = \frac{1}{a} (1 - e^{-at}).$$

Many processes follow the basic exponential decay in Eq. 2.8. For example, a quantity u of a molecule may arrive in a compartment at each point in time and then decay at rate a within the compartment. At any time, the total amount of the molecule in the compartment is the sum of the amounts that arrived at each time in the past, $u(t - \tau)$, weighted by the fraction that remains after decay, $e^{-a\tau}$.

The process in Eq. 2.7 corresponds exactly to the transfer function

$$P(s) = \frac{1}{s + a}, \quad (2.9)$$

in which the output is equivalent to the internal state, $y \equiv x$.

In the second example, an intrinsic process may oscillate at a particular frequency, ω_0 , described by

$$\ddot{x} + \omega_0^2 x = u.$$

This system produces output $x = \sin(\omega_0 t)$ for $u = 0$ and an initial condition along the sine curve. The corresponding transfer function is

$$P(s) = \frac{\omega_0}{s^2 + \omega_0^2}.$$

We can combine processes by simply multiplying the transfer functions. For example, suppose we have an intrinsic exponential decay process, $P(s)$, that is driven by oscillating inputs, $U(s)$. That combination produces an output

$$Y(s) = U(s)P(s) = \frac{\omega_0}{(s + a)(s^2 + \omega_0^2)}, \quad (2.10)$$

which describes a third-order differential equation, because the polynomial of s in the denominator has a highest power of three.

We could have easily obtained that third-order process by combining the two systems of differential equations given above. However, when systems include many processes in cascades, including feedback loops, it becomes difficult to combine the differential equations into very high-order systems. Multiplying the transfer functions through the system cascade remains easy. That advantage was nicely summarized by Bode (1964), one of the founders of classical control theory

The typical regulator system can frequently be described, in essentials, by differential equations of no more than perhaps the second, third or fourth order. ... In contrast, the order of the set of differential equations describing the typical negative feedback amplifier used in telephony is likely to be very much greater. As a matter of idle curiosity, I once counted to find out what the order of the set of equations in an amplifier I had just designed would have been, if I had worked with the differential equations directly. It turned out to be 55.

2.4 Frequency, Gain, and Phase

How do systems perform when parameters vary or when there are external environmental perturbations? We can analyze robustness by using the differential equations to calculate the dynamics for many combinations of parameters and perturbations. However, such calculations are tedious and difficult to evaluate for more than a couple of parameters. Using transfer functions, we can study a wide range of conditions by evaluating a function's output response to various inputs.

This chapter uses the Bode plot method. That method provides an easy and rapid way in which to analyze a system over various inputs. We can apply this method to individual transfer functions or to cascades of transfer functions that comprise entire systems.

This section illustrates the method with an example. The following section describes the general concepts and benefits.

Consider the transfer function

$$G(s) = \frac{a}{s + a}, \quad (2.11)$$

which matches the function for exponential decay in Eq. 2.9. Here, I multiplied the function by a so that the value would be one when $s = 0$.

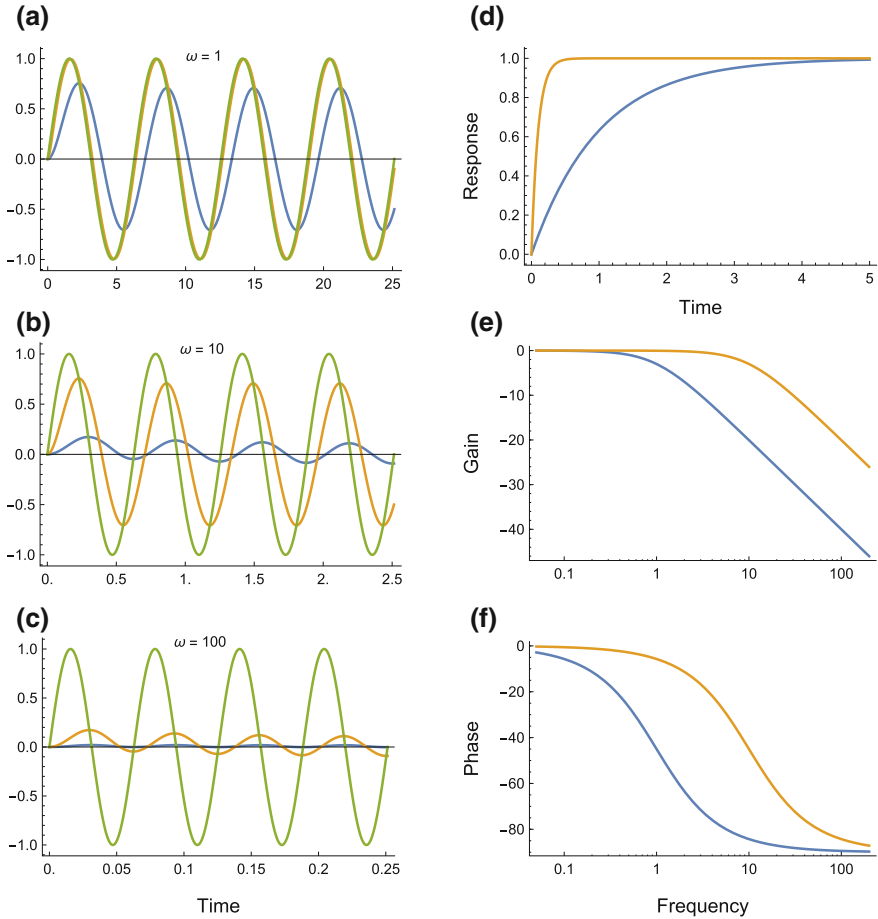


Fig. 2.2 Dynamics, gain, and phase of the low-pass filter in Eq. 2.11 in response to sine wave inputs at varying frequencies, ω . Details provided in the text. **a–c** Dynamics given by a multiplied by the transfer function on the right-hand side of Eq. 2.10. **d** Response of Eq. 2.11 to unit step input. **e** The scaling of the Bode gain plot is $20 \log_{10}(\text{gain})$. That scaling arises from the relation between the magnitude, $M = |G(j\omega)|$, and power, $P = M^2$, of a signal at a particular frequency, ω , or equivalently $M = \sqrt{P}$. If we consider gain as the magnitude of the output signal, then the scale for the gain is given as $20 \log_{10}(\sqrt{P}) = 10 \log_{10}(P)$, the standard decibel scaling for the relative power of a signal. **f** Bode phase plot

We can learn about a system by studying how it responds to different kinds of fluctuating environmental inputs. In particular, how does a system respond to different frequencies of sine wave inputs?

Figure 2.2 shows the response of the transfer function in Eq. 2.11 to sine wave inputs of frequency, ω . The left column of panels illustrates the fluctuating output in response to the green sine wave input. The blue (slow) and gold (fast) responses

correspond to parameter values in Eq. 2.11 of $a = 1$ and $a = 10$. All calculations and plots in this book are available in the accompanying Mathematica code (Wolfram Research 2017) at the site listed in the Preface.

In the top-left panel, at input frequency $\omega = 1$, the fast (gold) response output closely tracks the input. The slow (blue) response reduces the input by $\sqrt{2} \approx 0.7$. This output–input ratio is called the transfer function’s *gain*. The slow response output also lags the input by approximately 0.11 of one complete sine wave cycle of $2\pi = 6.28$ radians, thus the shift to the right of $0.11 \times 6.28 \approx 0.7$ radians along the x -axis.

We may also consider the lagging shift in angular units, in which 2π radians is equivalent to 360° . The lag in angular units is called the *phase*. In this case, the phase is written as $-0.11 \times 360^\circ \approx -40^\circ$, in which the negative sign refers to a lagging response.

A transfer function always transforms a sine wave input into a sine wave output modulated by the gain and phase. Thus, the values of gain and phase completely describe the transfer function response.

Figure 2.2b shows the same process but driven at a higher input frequency of $\omega = 10$. The fast response is equivalent to the slow response of the upper panel. The slow response has been reduced to a gain of approximately 0.1, with a phase of approximately -80° . At the higher frequency of $\omega = 100$ in the bottom panel, the fast response again matches the slow response of the panel above, and the slow response’s gain is reduced to approximately 0.01.

Both the slow and fast transfer functions pass low-frequency inputs into nearly unchanged outputs. At higher frequencies, they filter the inputs to produce greatly reduced, phase-shifted outputs. The transfer function form of Eq. 2.11 is therefore called a low-pass filter, passing low frequencies and blocking high frequencies. The two filters in this example differ in the frequencies at which they switch from passing low-frequency inputs to blocking high-frequency inputs.

2.5 Bode Plots of Gain and Phase

A Bode plot shows a transfer function’s gain and phase at various input frequencies. The Bode gain plot in Fig. 2.2e presents the gain on a log scale, so that a value of zero corresponds to a gain of one, $\log(1) = 0$.

For the system with the slower response, $a = 1$ in blue, the gain is nearly one for frequencies less than a and then drops off quickly for frequencies greater than a . Similarly, the system with faster response, $a = 10$, transitions from a system that passes low frequencies to one that blocks high frequencies at a point near its a value. Figure 2.2f shows the phase changes for these two low-pass filters. The slower blue system begins to lag at lower input frequencies.

Low-pass filters are very important because low-frequency inputs are often external signals that the system benefits by tracking, whereas high-frequency inputs are often noisy disturbances that the system benefits by ignoring.

In engineering, a designer can attach a low-pass filter with a particular transition parameter a to obtain the benefits of filtering an input signal. In biology, natural selection must often favor appending biochemical processes or physical responses that act as low-pass filters. In this example, the low-pass filter is simply a basic exponential decay process.

Figure 2.2d shows a key tradeoff between the fast and slow responses. In that panel, the system input is increased in a step from zero to one at time zero. The fast system responds quickly by increasing its state to a matching value of one, whereas the slow system takes much longer to increase to a matching value. Thus, the fast system may benefit from its quick response to environmental changes, but it may lose by its greater sensitivity to high-frequency noise. That tradeoff between responsiveness and noise rejection forms a common theme in the overall performance of systems.

To make the Bode plot, we must calculate the gain and phase of a transfer function's response to a sinusoidal input of frequency ω . Most control theory textbooks show the details (e.g., Ogata 2009). Here, I briefly describe the calculations, which will be helpful later.

Transfer functions express linear dynamical systems in terms of the complex Laplace variable $s = \sigma + j\omega$. I use j for the imaginary number to match the control theory literature.

The gain of a transfer function describes how much the function multiplies its input to produce its output. The gain of a transfer function $G(s)$ varies with the input value, s . For complex-valued numbers, we use magnitudes to analyze gain, in which the magnitude of a complex value is $|s| = \sqrt{\sigma^2 + \omega^2}$.

It turns out that the gain of a transfer function in response to a sinusoidal input at frequency ω is simply $|G(j\omega)|$, the magnitude of the transfer function at $s = j\omega$. The phase angle is the arctangent of the ratio of the imaginary to the real parts of $G(j\omega)$.

For the exponential decay dynamics that form the low-pass filter of Eq. 2.11, the gain magnitude, M , and phase angle, ϕ , are

$$M = |G(j\omega)| = \frac{a}{\sqrt{\omega^2 + a^2}}$$

$$\phi = \angle G(j\omega) = -\tan^{-1} \frac{\omega}{a}.$$

Any stable transfer function's long-term steady-state response to a sine wave input at frequency ω is a sine wave output at the same frequency, multiplied by the gain magnitude, M , and shifted by the phase angle, ϕ , as

$$\sin(\omega t) \xrightarrow{G} M \sin(\omega t + \phi), \quad (2.12)$$

in which the angle is given in radians. For example, if the phase lags by one-half of a cycle, $\phi = -\pi \equiv -180^\circ$, then $M \sin(\omega t + \phi) = -M \sin(\omega t)$.

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

