1 Introduction

In controls we make use of the abstract concept of a *system*: we identify a phenomenon or a process, the *system*, and two classes of *signals*, which we label as *inputs* and *outputs*. A signal is something that can be measured or quantified. In this book we use real numbers to quantify signals. The classification of a particular signal as an input means that it can be identified as the *cause* of a particular system behavior, whereas an output signal is seen as the *product* or *consequence* of the behavior. Of course the classification of a phenomenon as a system and the labeling of input and output signals is an abstract construction. A mathematical description of a system and its signals is what constitutes a *model*. The entire abstract construction, and not only the equations that we will later associate with particular signals and systems, is the model.

We often represent the relationship between a system and its input and output signals in the form of a *block-diagram*, such as the ones in Fig. 1.1 through Fig. 1.3. The diagram in Fig. 1.1 indicates that a system, G, produces an output signal, y, in the presence of the input signal, u. Block-diagrams will be used to represent the interconnection of systems and even algorithms. For example, Fig. 1.2 depicts the components and signals in a familiar controlled system, a water heater; the block-diagram in Fig. 1.3 depicts an algorithm for converting temperature in degrees Fahrenheit to degrees Celsius, in which the output of the circle in Fig. 1.3 is the algebraic sum of the incoming signals with signs as indicated near the incoming arrows.

1.1 Models and Experiments

Systems, signals, and models are often associated with concrete or abstract experiments. A model reflects a particular setup in which the outputs appear *correlated* with a prescribed set of inputs. For example, we might attempt to model a car by performing the following experiment: on an unobstructed and level road, we depress the accelerator pedal and let the car travel in a straight line.[1] We keep the pedal excursion constant and let the car reach constant velocity. We record the amount the pedal has been depressed and the car's terminal velocity. The results of this experiment, repeated multiple times with different amounts of pedal excursion, might look like the data shown in Fig. 1.4. In this experiment the signals are

[1] This may bring to memory a bad joke about physicists and spherical cows . . .

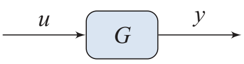

Figure 1.1 System represented as a block-diagram; u is the input signal; y is the output signal; y and u are related through $y = G(u)$ or simply $y = Gu$.

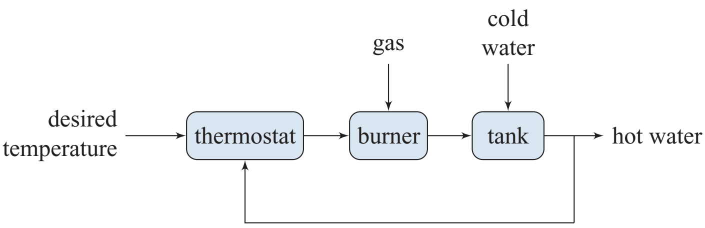

Figure 1.2 Block-diagram of a controlled system: a gas water heater; the blocks thermostat, burner, and tank, represent components or sub-systems; the arrows represent the *flow* of input and output signals.

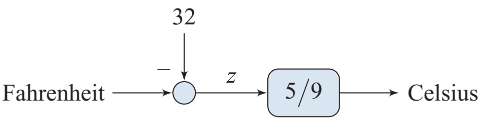

Figure 1.3 Block-diagram of an algorithm to convert temperatures in Fahrenheit to Celsius: Celsius $= 5/9$(Fahrenheit $- 32$); the output of the circle block is the algebraic sum of the incoming signals with the indicated sign, i.e. $z =$ Fahrenheit $- 32$.

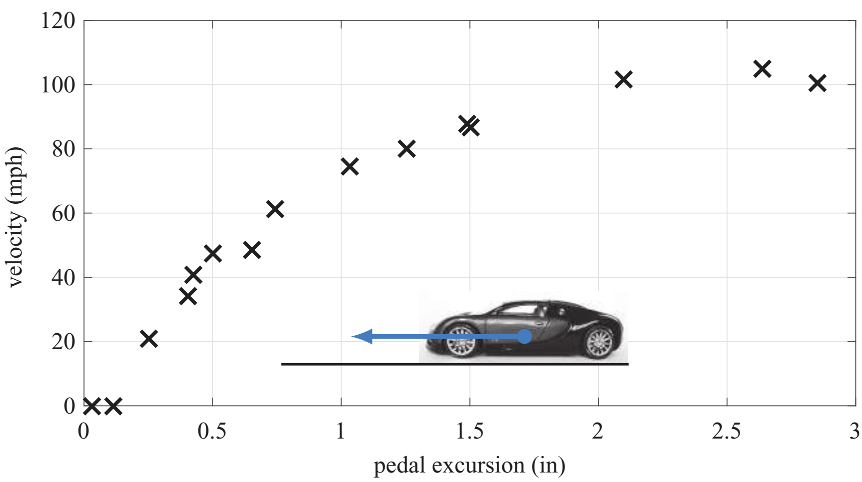

Figure 1.4 Experimental determination of the effect of pressing the gas pedal on the car's terminal velocity; the pedal excursion is the input signal, u, and the car's terminal velocity is the output signal, y.

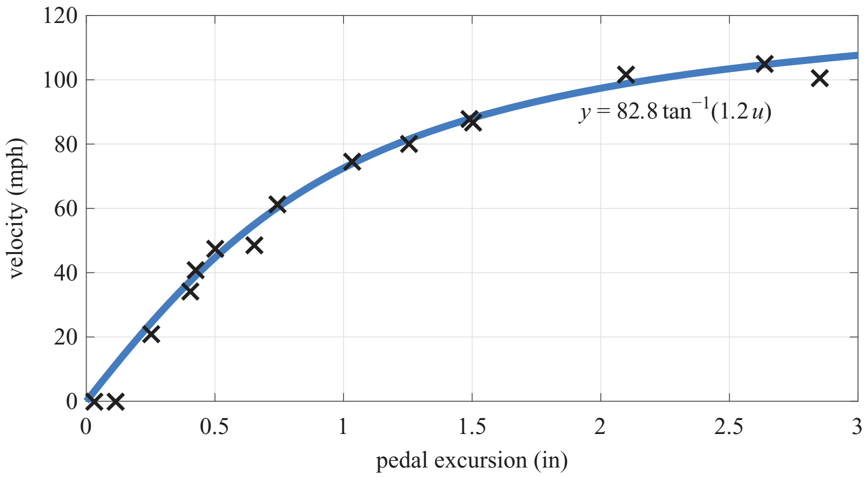

Figure 1.5 Fitting the curve $y = \alpha \tan^{-1}(\beta u)$ to the data from Fig. 1.4.

input: pedal excursion, in cm, inches, etc.;
output: terminal velocity of the car, in m/s, mph, etc.

The **system** is the car *and* the particular conditions of the experiment. The data captures the fact that the car does not move at all for small pedal excursions and that the terminal velocity *saturates* as the pedal reaches the end of its excursion range.

From Fig. 1.4, one might try to *fit* a particular mathematical function to the experimental data[2] in hope of obtaining a *mathematical model*. In doing so, one invariably loses something in the name of a simpler description. Such trade-offs are commonplace in science, and it should be no different in the analysis and design of control systems. Figure 1.5 shows the result of fitting a curve of the form

$$y = \alpha \tan^{-1}(\beta u),$$

where u is the input, pedal excursion in inches, and y is the output, terminal velocity in mph. The parameters $\alpha = 82.8$ and $\beta = 1.2$ shown in Fig. 1.5 were obtained from a standard least-squares fit. See also P1.11.

The choice of the above particular function involving the arc-tangent might seem somewhat arbitrary. When possible, one should select candidate functions from first principles derived from physics or other scientific reasoning, but this does not seem to be easy to do in the case of the experiment we described. Detailed physical modeling of the vehicle would involve knowledge and further modeling of the components of the vehicle, not to mention the many uncertainties brought in by the environment, such as wind, road conditions, temperature, etc. Instead, we make an "educated choice" based on certain physical aspects of the experiment that we believe the model should capture. In this case, from our daily experience with vehicles, we expect that the terminal velocity

[2] All data used to produce the figures in this book is available for download from the website http://www.cambridge.org/deOliveira.

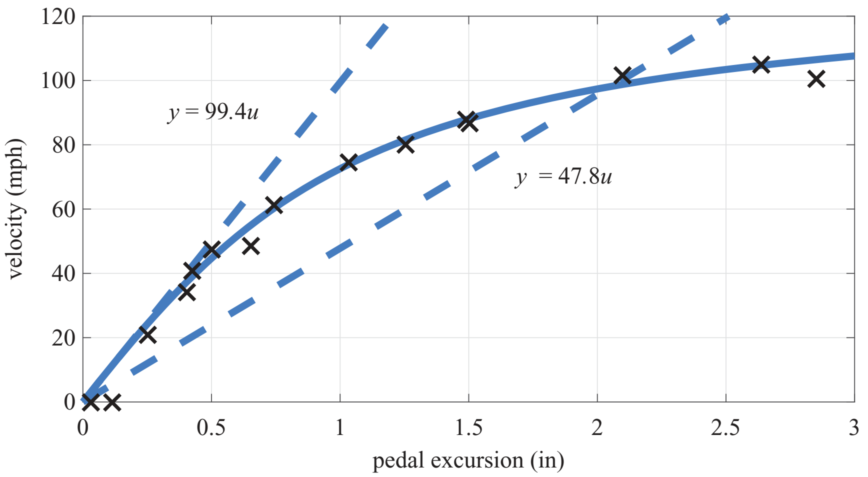

Figure 1.6 Linear mathematical models of the form $y = \gamma u$ for the data in Fig. 1.4 (dashed); the model with $\gamma = 47.8$ was obtained by a least-squares fit; the model with $\gamma = 99.4$ was obtained after linearization of the nonlinear model (solid) obtained in Fig. 1.5; see P1.12 and P1.11.

will eventually *saturate*, either as one reaches full throttle or as a result of limitations on the maximum power that can be delivered by the vehicle's powertrain. We also expect that the function be *monotone*, that is, the more you press the pedal, the larger the terminal velocity will be. Our previous exposure to the properties of the arc-tangent function *and* engineering intuition about the expected outcome of the experiment allowed us to successfully select this function as a suitable candidate for a model.

Other families of functions might suit the data in Fig. 1.5. For example, we could have used *polynomials*, perhaps constrained to pass through the origin and ensure monotonicity. One of the most useful classes of mathematical models one can consider is that of *linear models*, which are, of course, first-order polynomials. One might be tempted to equate linear with simple. Whether or not this might be true in some cases, simplicity is far from a sin. More often than not, the loss of some feature neglected by a linear model is offset by the availability of a much broader set of analytic tools. It is better to *know* when you are wrong than to *believe* you are right. As the title suggests, this book is mostly concerned with linear models. Speaking of linear models, one might propose describing the data in Fig. 1.4 by a linear mathematical model of the form

$$y = \gamma u. \tag{1.1}$$

Figure 1.6 shows two such models (dashed lines). The curve with slope coefficient $\gamma = 47.8$ was obtained by performing a least-squares fit to all data points (see P1.11). The curve with coefficient $\gamma = 99.4$ is a first-order approximation of the nonlinear model calculated in Fig. 1.5 (see P1.12). Clearly, each model has its limitations in describing the experiment. Moreover, one model might be better suited to describe certain aspects of the experiment than the other. Responsibility rests with the engineer or the scientist to select the model, or perhaps set of models, that better fits the problem in hand, a task that at times may resemble an art more than a science.

1.2 Cautionary Note

It goes without saying that the mathematical models described in Section 1.1 do not purport to capture every detail of the experiment, not to mention reality. Good models are the ones that capture *essential* aspects that we *perceive* or can experimentally validate as real, for example how the terminal velocity of a car responds to the acceleration pedal in the given experimental conditions. A model does not even need to be *correct* to be useful: for centuries humans used[3] a model in which the sun revolves around the earth to predict and control their days! What is important is that models provide a way to express *relevant* aspects of reality using mathematics. When mathematical models are used in control design, it is therefore with the understanding that the model is bound to capture only a subset of features of the actual phenomenon they represent. At no time should one be fooled into *believing* in a model. The curious reader will appreciate [Fey86] and the amusingly provocative [Tal07].

With this caveat in mind, it is useful to think of an idealized *true* or *nominal model*, just as is done in physics, against which a particular setup can be *mathematically* evaluated. This nominal model might even be different than the model used by a particular control algorithm, for instance, having more details or being more complex or more accurate. Of course *physical* evaluation of a control system with respect to the underlying natural phenomenon is possible only by means of experimentation which should also include the physical realization of the controller in the form of computer hardware and software, electric circuits, and other necessary mechanical devices. We will discuss in Chapter 5 how certain physical devices can be used to implement the dynamic controllers you will learn to design in this book.

The models discussed so far have been *static*, meaning that the relationship between inputs and outputs is *instantaneous* and is independent of the past history of the system or their signals. Yet the main objective of this book is to work with *dynamic* models, in which the relationship between present inputs and outputs may depend on the present and past history[4] of the signals.

With the goal of introducing the main ideas behind feedback control in a simpler setup, we will continue to work with static models for the remainder of this chapter. In the case of static models, a mathematical *function* or a set of *algebraic equations* will be used to represent such relationships, as done in the models discussed just above in Section 1.1.

Dynamic models will be considered starting in Chapter 2. In this book, signals will be continuous functions of time, and dynamic models will be formulated with the help of *ordinary differential equations*. As one might expect, experimental procedures that can estimate the parameters of dynamic systems need to be much more sophisticated than the ones discussed so far. A simple experimental procedure will be briefly discussed in Section 2.4, but the interested reader is encouraged to consult one of the many excellent works on this subject, e.g. [Lju99].

[3] Apparently 1 in 4 Americans and 1 in 3 Europeans still go by that model [Gro14].
[4] What about the future?

1.3 A Control Problem

Consider the following problem:

Under the experimental conditions described in Section 1.1 and given a target terminal velocity, $\bar{y}$, is it possible to design a system, the controller, that is able to command the accelerator pedal of a car, the input, u, to produce a terminal velocity, the output, y, equal to the target velocity?

An *automatic* system that can solve this problem is found in many modern cars, with the name *cruise controller*. Of course, another *system* that is capable of solving the same problem is a human driver.[5] In this book we are mostly interested in solutions that can be implemented as an *automatic control*, that is, which can be performed by some combination of mechanical, electric, hydraulic, or pneumatic systems running without human intervention, often being programmed in a digital computer or some other logical circuit or calculator.

Problems such as this are referred to in the control literature as *tracking* problems: the controller should make the system, a car, *follow* or *track* a given target output, the desired terminal velocity. In the next sections we will discuss two possible approaches to the cruise control problem.

1.4 Solution without Feedback

The role of the controller in tracking is to compute the input signal u which produces the desired output signal y. One might therefore attempt to solve a tracking problem using a system (controller) of the form

$$u = K(\bar{y}).$$

This controller can use only the reference signal, the target output $\bar{y}$, and is said to be in *open-loop*,[6] as the controller output signal, u, is not a function of the system output signal, y.

With the intent of analyzing the proposed solution using mathematical models, assume that the car can be represented by a *nominal model*, say G, that relates the input u (pedal excursion) to the output y (terminal velocity) through the mathematical function

$$y = G(u).$$

The connection of the controller with this idealized model is depicted in the block-diagram in Fig. 1.7. Here the function G can be obtained after fitting experimental data as done in Figs. 1.5 and 1.6, or borrowed from physics or engineering science principles.

[5] After some 16 years of *learning*.
[6] As opposed to *closed-loop*, which will be discussed in Section 1.5.

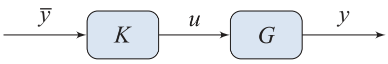

Figure 1.7 Open-loop control: the controller, K, is a function of the reference input, $\bar{y}$, but not a function of the system output, y.

The block-diagram in Fig. 1.7 represents the following relationships:

$$y = G(u), \qquad u = K(\bar{y}),$$

that can be combined to obtain

$$y = G\left(K(\bar{y})\right).$$

If G is *invertible* and K is chosen to be the inverse of G, that is $K = G^{-1}$, then

$$y = G\left(G^{-1}(\bar{y})\right) = \bar{y}.$$

Matching the controller, K, with the nominal model, G, is paramount: if $K \neq G^{-1}$ then $y \neq \bar{y}$.

When both the nominal model G and the controller K are linear,

$$y = G\,u, \qquad u = K\,\bar{y}, \qquad y = G\,K\,\bar{y},$$

from which $\bar{y} = y$ only if the product of the *constants* K and G is equal to one:

$$K\,G = 1 \qquad \Longrightarrow \qquad K = G^{-1}, \qquad u = G^{-1}\,\bar{y}.$$

Because the control law relies on knowledge of the nominal model G to achieve its goal, any imperfection in the model or in the implementation of the controller will lead to less than perfect tracking.

1.5 Solution with Feedback

The controller in the open-loop solution considered in Section 1.4 is allowed to make use only of the target output, $\bar{y}$. When a measurement, even if imprecise, of the system output is available, one may benefit from allowing the controller to make use of the measurement signal, y. In the case of the car cruise control, the terminal velocity, y, can be measured by an on-board speedometer. Of course the target velocity, $\bar{y}$, is set by the driver.

Controllers that make use of output signals to compute the control inputs are called *feedback controllers*. In its most general form, a feedback controller has the functional form

$$u = K(\bar{y}, y).$$

In practice, most feedback controllers work by first creating an *error signal*, $\bar{y} - y$, which is then used by the controller:

$$u = K(e), \qquad e = \bar{y} - y. \tag{1.2}$$

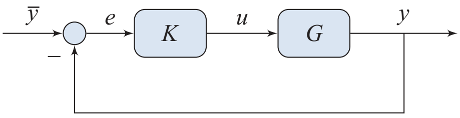

Figure 1.8 Closed-loop feedback control: the controller, K, is a function of the reference input, $\bar{y}$, and the system output, y, by way of the error signal, $e = \bar{y} - y$.

This scheme is depicted in the block-diagram in Fig. 1.8. One should question whether it is possible to implement a physical system that replicates the block-diagram in Fig. 1.8. In this diagram, the measurement, y, that takes part in the computation of the control, u, in the controller block, K, is the same as that which comes out of the system, G. In other words, the signals flow in this diagram is *instantaneous*. Even though we are not yet properly equipped to address this question, we anticipate that it will be possible to construct and analyze *implementable* or *realizable* versions of the feedback diagram in Fig. 1.8 by taking into account dynamic phenomena, which we will start discussing in the next chapter.

At this point, we are content to say that if the computation implied by feedback is performed *fast enough*, then the scheme *should* work. We analyze the proposed feedback solution only in the case of static linear models, that is, when both the controller, K, and the system to be controlled, G, are linear. Feedback controllers of the form (1.2), which are *linear* and *static*, are known by the name *proportional controllers*, or P controllers for short. In the *closed-loop* diagram of Fig. 1.8, we can think of the signal $\bar{y}$, the target velocity, as an input, and of the signal y, the terminal velocity, as an output. A mathematical description of the relationship between the input signal, $\bar{y}$, and output signal, y, assuming linear models, can be computed from the diagram:

$$y = Gu, \qquad u = Ke, \qquad e = \bar{y} - y.$$

After eliminating the signals e and u we obtain

$$y = GKe = GK(\bar{y} - y) \qquad \Longrightarrow \qquad (1 + GK)y = GK\bar{y}.$$

When $GK \neq -1$,

$$y = H\bar{y}, \qquad H = \frac{GK}{1 + GK}.$$

A mathematical relationship governing a particular pair of inputs and outputs is called a *transfer-function*. The function H calculated above is known as a *closed-loop transfer-function*.

Ironically, a first conclusion from the closed-loop analysis is that it is not possible to achieve exact tracking of the target velocity since H cannot be equal to one for any finite value of the constants G and K, not even when $K = G^{-1}$, which was the open-loop solution. However, it is not so hard to make H get close to one: just make K large! More precisely, make the product GK large. How large it needs to be depends on the particular system G. However, a welcome side-effect of the closed-loop solution is that the controller gain, K, does not depend directly on the value of the system model, G.

Table 1.1 Closed-loop transfer-function, H, for various values of K and G

	K				
G	0.02	0.05	0.5	1	3
47.8	0.4888	0.7050	0.9598	0.9795	0.9931
73.3	0.5945	0.7856	0.9734	0.9865	0.9955
99.4	0.6653	0.8325	0.9803	0.9900	0.9967

As the calculations in Table 1.1 reveal, the closed-loop transfer-function, H, remains within 1% of 1 for values K greater than or equal to 3 for *any* value of G lying between the two crude linear models estimated earlier in Fig. 1.6.

In other words, feedback control does not seem to rely on exact knowledge of the system model in order to achieve good tracking performance. This is a major feature of feedback control, and one of the reasons why we may get away with using incomplete and not extremely accurate mathematical models for feedback design. One might find this strange, especially to scientists and engineers trained to look for accuracy and fidelity in their models of the world, a line of thought that might lead one to believe that better accuracy *requires* the use of complex models. For example, the complexity required for accurately modeling the interaction of an aircraft with its surrounding air may be phenomenal. Yet, as the Wright brothers and other flight pioneers demonstrated, it is possible to design and implement effective feedback control of aircraft without relying explicitly on such complex models.

This remarkable feature remains for the most part true even if nonlinear[7] models are considered, although the computation of the transfer-function, H, becomes more complicated.[8] Figure 1.9 shows a plot of the ratio $y/\bar{y}$ for various choices of gain, K, when a linear controller is in feedback with the static nonlinear model, G, fitted in Fig. 1.5. The trends are virtually the same as those obtained using linear models. Note also that the values of the ratio of the terminal velocity by the target velocity are close to the values of H calculated for the linear model with gain $G = 99.4$ which was obtained through "linearization" of the nonlinear model, especially at low velocities.

Insight on the reasons why feedback control can achieve tracking without relying on precise models is obtained if we look at the control, the signal u, that is effectively computed by the closed-loop solution. Following steps similar to the ones used in the derivation of the closed-loop transfer-function, we calculate

$$u = Ke = K(\bar{y} - y) = K(1 - H)\,\bar{y} = \frac{K}{1 + GK}\,\bar{y} = \frac{1}{K^{-1} + G}\,\bar{y}.$$

Note that $\lim_{K\to\infty} u = G^{-1}\,\bar{y}$, which is exactly the same control as that computed in open-loop (see Section 1.4). This time, however, it is the feedback loop that *computes* the function G^{-1} based on the error signal, $e = \bar{y} - y$. Indeed, u is simply equal to

[7] Many but not all nonlinear models.
[8] It requires solving the nonlinear algebraic equation $y = G(K(\bar{y} - y))$ for y. The dynamic version of this problem is significantly more complex.

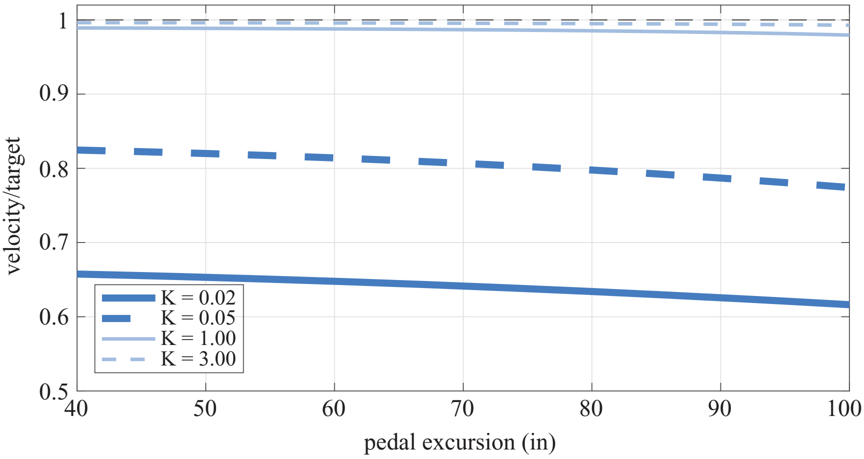

Figure 1.9 Effect of the gain K on the ability of the terminal velocity, y, to track a given target velocity, $\bar{y}$, when the linear feedback control, $u = K(y - \bar{y})$, is in closed-loop (Fig. 1.8) with the nonlinear model, $y = G(u) = 82.8 \tan^{-1}(1.2u)$ from Fig. 1.5.

$K(\bar{y} - y)$, which, when K is made large, converges to $G^{-1}\bar{y}$ by virtue of feedback, no matter what the value of G is. A natural question is what are the side-effects of raising the control gain in order to improve the tracking performance? We will come back to this question at many points in this book as we learn more about dynamic systems and feedback.

1.6 Sensitivity

In previous sections, we made statements regarding how insensitive the closed-loop feedback solution was with respect to changes in the system model when compared with the open-loop solution. We can quantify this statement in the case of static linear models.

As seen before, in both open- and closed-loop solutions to the tracking control problem, the output y is related to the target output $\bar{y}$ through

$$y = H(G)\,\bar{y}.$$

The notation $H(G)$ indicates that the transfer-function, H, depends on the system model, G. In the open-loop solution $H(G) = GK$ and in the closed-loop solution $H(G) = GK(1 + GK)^{-1}$.

Now consider that G assumes values in the neighborhood of a certain nominal model $\bar{G}$ and that $H(\bar{G}) \neq 0$. Assume that those changes in G affect H in a continuous and differentiable way so that[9]

$$H(G) = H(\bar{G}) + H'(\bar{G})(\Delta G) + O\left(\Delta G^2\right), \qquad \Delta G = G - \bar{G},$$

[9] The notation $O(x^n)$ indicates a polynomial in x that has only terms with degree greater than or equal to n.