Chapter 8

Adaptive Control

Many dynamic systems to be controlled have constant or slowly-varying uncertain parameters. For instance, robot manipulators may carry large objects with unknown inertial parameters. Power systems may be subjected to large variations in loading conditions. Fire-fighting aircraft may experience considerable mass changes as they load and unload large quantities of water. Adaptive control is an approach to the control of such systems. The basic idea in adaptive control is to estimate the uncertain plant parameters (or, equivalently, the corresponding controller parameters) on-line based on the measured system signals, and use the estimated parameters in the control input computation. An adaptive control system can thus be regarded as a control system with on-line parameter estimation. Since adaptive control systems, whether developed for linear plants or for nonlinear plants, are inherently nonlinear, their analysis and design is intimately connected with the materials presented in this book, and in particular with Lyapunov theory.

Research in adaptive control started in the early 1950's in connection with the design of autopilots for high-performance aircraft, which operate at a wide range of speeds and altitudes and thus experience large parameter variations. Adaptive control was proposed as a way of automatically adjusting the controller parameters in the face of changing aircraft dynamics. But interest in the subject soon diminished due to the lack of insights and the crash of a test flight. It is only in the last decade that a coherent theory of adaptive control has been developed, using various tools from nonlinear control theory. These theoretical advances, together with the availability of cheap computation, have lead to many practical applications, in areas such as robotic
8.1 Basic Concepts in Adaptive Control

In this section, we address a few basic questions, namely, why we need adaptive control, what the basic structures of adaptive control systems are, and how to go about designing adaptive control systems.

8.1.1 Why Adaptive Control?

In some control tasks, such as those in robot manipulation, the systems to be controlled have parameter uncertainty at the beginning of the control operation. Unless such parameter uncertainty is gradually reduced on-line by an adaptation or estimation mechanism, it may cause inaccuracy or instability for the control systems. In many other tasks, such as those in power systems, the system dynamics may have well known dynamics at the beginning, but experience unpredictable parameter variations as the control operation goes on. Without continuous "redesign" of the controller, the initially appropriate controller design may not be able to control the changing plant well. Generally, the basic objective of adaptive control is to maintain consistent performance of a system in the presence of uncertainty or unknown variation in plant parameters. Since such parameter uncertainty or variation occurs in many practical problems, adaptive control is useful in many industrial contexts. These include:

- **Robot manipulation**: Robots have to manipulate loads of various sizes, weights, and mass distributions (Figure 8.1). It is very restrictive to assume that the inertial parameters of the loads are well known before a robot picks them up and moves them away. If controllers with constant gains are used and the load parameters are not accurately known, robot motion can be either inaccurate or unstable. Adaptive control, on the other hand, allows robots to move loads of unknown parameters with high speed and high accuracy.
**Ship steering:** On long courses, ships are usually put under automatic steering. However, the dynamic characteristics of a ship strongly depend on many uncertain parameters, such as water depth, ship loading, and wind and wave conditions (Figure 8.2). Adaptive control can be used to achieve good control performance under varying operating conditions, as well as to avoid energy loss due to excessive rudder motion.

**Aircraft control:** The dynamic behavior of an aircraft depends on its altitude, speed, and configuration. The ratio of variations of some parameters can lie between 10 to 50 in a given flight. As mentioned earlier, adaptive control was originally developed to achieve consistent aircraft performance over a large flight envelope.

**Process control:** Models for metallurgical and chemical processes are usually complex and also hard to obtain. The parameters characterizing the processes vary from batch to batch. Furthermore, the working conditions are usually time-varying (e.g., reactor characteristics vary during the reactor's life, the raw materials entering the process are never exactly the same, atmospheric and climatic conditions also tend to change). In fact, process control is one of the most important and active application areas of adaptive control.

![Figure 8.1: A robot carrying a load of uncertain mass properties](image-url)
Adaptive control has also been applied to other areas, such as power systems and biomedical engineering. Most adaptive control applications are aimed at handling inevitable parameter variation or parameter uncertainty. However, in some applications, particularly in process control, where hundreds of control loops may be present in a given system, adaptive control is also used to reduce the number of design parameters to be manually tuned, thus yielding an increase in engineering efficiency and practicality.

Figure 8.2: A freight ship under various loadings and sea conditions

To gain insights about the behavior of the adaptive control systems and also to avoid mathematical difficulties, we shall assume the unknown plant parameters are constant in analyzing the adaptive control designs. In practice, the adaptive control systems are often used to handle time-varying unknown parameters. In order for the analysis results to be applicable to these practical cases, the time-varying plant parameters must vary considerably slower than the parameter adaptation. Fortunately, this is often satisfied in practice. Note that fast parameter variations may also indicate that the modeling is inadequate and that the dynamics causing the parameter changes should be additionally modeled.

Finally, let us note that robust control can also be used to deal with parameter uncertainty, as seen in chapter 7. Thus, one may naturally wonder about the differences and relations between the robust approach and the adaptive approach. In principle, adaptive control is superior to robust control in dealing with uncertainties in constant or slowly-varying parameters. The basic reason lies in the learning behavior of adaptive control systems: an adaptive controller improves its performance as adaptation goes on, while a robust controller simply attempts to keep consistent performance. Another reason is that an adaptive controller requires little or no a priori information about the unknown parameters, while a robust controller usually requires reasonable a priori estimates of the parameter bounds. Conversely, robust control has some desirable features which adaptive control does not have, such as its ability to deal with disturbances, quickly varying parameters, and unmodeled
dynamics. Such features actually may be combined with adaptive control, leading to robust adaptive controllers in which uncertainties on constant or slowly-varying parameters is reduced by parameter adaptation and other sources of uncertainty are handled by robustification techniques. It is also important to point out that existing adaptive techniques for nonlinear systems generally require a linear parametrization of the plant dynamics, i.e., that parametric uncertainty be expressed linearly in terms of a set of unknown parameters. In some cases, full linear parametrization and thus adaptive control cannot be achieved, but robust control (or adaptive control with robustifying terms) may be possible.

8.1.2 What Is Adaptive Control?

An adaptive controller differs from an ordinary controller in that the controller parameters are variable, and there is a mechanism for adjusting these parameters on-line based on signals in the system. There are two main approaches for constructing adaptive controllers. One is the so-called model-reference adaptive control method, and the other is the so-called self-tuning method.

MODEL-REFERENCE ADAPTIVE CONTROL (MRAC)

Generally, a model-reference adaptive control system can be schematically represented by Figure 8.3. It is composed of four parts: a plant containing unknown parameters, a reference model for compactly specifying the desired output of the control system, a feedback control law containing adjustable parameters, and an adaptation mechanism for updating the adjustable parameters.

![Figure 8.3: A model-reference adaptive control system](image-url)
The plant is assumed to have a known structure, although the parameters are unknown. For linear plants, this means that the number of poles and the number of zeros are assumed to be known, but that the locations of these poles and zeros are not. For nonlinear plants, this implies that the structure of the dynamic equations is known, but that some parameters are not.

A reference model is used to specify the ideal response of the adaptive control system to the external command. Intuitively, it provides the ideal plant response which the adaptation mechanism should seek in adjusting the parameters. The choice of the reference model is part of the adaptive control system design. This choice has to satisfy two requirements. On the one hand, it should reflect the performance specification in the control tasks, such as rise time, settling time, overshoot or frequency domain characteristics. On the other hand, this ideal behavior should be achievable for the adaptive control system, i.e., there are some inherent constraints on the structure of the reference model (e.g., its order and relative degree) given the assumed structure of the plant model.

The controller is usually parameterized by a number of adjustable parameters (implying that one may obtain a family of controllers by assigning various values to the adjustable parameters). The controller should have perfect tracking capacity in order to allow the possibility of tracking convergence. That is, when the plant parameters are exactly known, the corresponding controller parameters should make the plant output identical to that of the reference model. When the plant parameters are not known, the adaptation mechanism will adjust the controller parameters so that perfect tracking is asymptotically achieved. If the control law is linear in terms of the adjustable parameters, it is said to be linearly parameterized. Existing adaptive control designs normally require linear parameterization of the controller in order to obtain adaptation mechanisms with guaranteed stability and tracking convergence.

The adaptation mechanism is used to adjust the parameters in the control law. In MRAC systems, the adaptation law searches for parameters such that the response of the plant under adaptive control becomes the same as that of the reference model, i.e., the objective of the adaptation is to make the tracking error converge to zero. Clearly, the main difference from conventional control lies in the existence of this mechanism. The main issue in adaptation design is to synthesize an adaptation mechanism which will guarantee that the control system remains stable and the tracking error converges to zero as the parameters are varied. Many formalisms in nonlinear control can be used to this end, such as Lyapunov theory, hyperstability theory, and passivity theory. Although the application of one formalism may be more convenient than that of another, the results are often equivalent. In this chapter, we shall mostly use Lyapunov theory.
As an illustration of MRAC control, let us describe a simple adaptive control system for an unknown mass.

Example 8.1: MRAC control of an unknown mass

Consider the control of a mass on a frictionless surface by a motor force $u$, with the plant dynamics being

$$m \ddot{x} = u \tag{8.1}$$

Assume that a human operator provides the positioning command $r(t)$ to the control system (possibly through a joystick). A reasonable way of specifying the ideal response of the controlled mass to the external command $r(t)$ is to use the following reference model

$$\ddot{x}_m + \lambda_1 \dot{x}_m + \lambda_2 x_m = \lambda_2 r(t) \tag{8.2}$$

with the positive constants $\lambda_1$ and $\lambda_2$ chosen to reflect the performance specifications, and the reference model output $x_m$ being the ideal output of the control system (i.e., ideally, the mass should go to the specified position $r(t)$ like a well-damped mass-spring-damper system).

If the mass $m$ is known exactly, we can use the following control law to achieve perfect tracking

$$u = m(\ddot{x}_m - 2 \lambda \dot{x} - \lambda^2 x) \tag{8.3}$$

with $\ddot{x} = x(t) - x_m(t)$ representing the tracking error and $\lambda$ a strictly positive number. This control law leads to the exponentially convergent tracking error dynamics

$$m \ddot{x} + 2 \lambda \dot{x} + \lambda^2 x = 0 \tag{8.4}$$

Now let us assume that the mass is not known exactly. We may use the following control law

$$u = \hat{m}(\ddot{x}_m - 2 \lambda \dot{x} - \lambda^2 x) \tag{8.3}$$

which contains the adjustable parameter $\hat{m}$. Substitution of this control law into the plant dynamics leads to the closed-loop error dynamics

$$m \ddot{\zeta} + \lambda m \zeta = \hat{m} \nu \tag{8.4}$$

where $\zeta$, a combined tracking error measure, is defined by

$$\zeta = \dot{\dot{x}} + \lambda \dot{x} \tag{8.5}$$

the signal quantity $\nu$ by

$$\nu = \dot{x}_m - 2 \lambda \dot{x} - \lambda^2 x$$

and the parameter estimation error $\hat{m}$ by
Equation (8.4) indicates that the combined tracking error $s$ is related to the parameter error through a stable filter relation. One way of adjusting parameter $\hat{m}$ (for reasons to be seen later) is to use the following update law

$$\dot{\hat{m}} = -\gamma v s$$

(8.6)

where $\gamma$ is a positive constant called the adaptation gain. One easily sees the nonlinear nature of the adaptive control system, by noting that the parameter $\hat{m}$ is adjusted based on system signals, and thus the controller (8.3) is nonlinear.

The stability and convergence of this adaptive control system can be analyzed using Lyapunov theory. For the closed-loop dynamics (8.4) and (8.6), with $s$ and $\hat{m}$ as states, we can consider the following Lyapunov function candidate,

$$V = \frac{1}{2} ms^2 + \frac{1}{\gamma} \hat{m}^2$$

(8.7)

Its derivative can be easily shown to be

$$\dot{V} = -\lambda m s^2$$

(8.8)

Using Barbalat’s lemma in chapter 4, one can easily show that $s$ converges to zero. Due to the relation (8.5), the convergence of $s$ to zero implies that of the position tracking error $\tilde{x}$ and the velocity tracking error $\dot{x}$.

For illustration, simulations of this simple adaptive control system are provided in Figures 8.4 and 8.5. The true mass is assumed to be $m = 2$. The initial value of $\hat{m}$ is chosen to be zero, indicating no a priori parameter knowledge. The adaptation gain is chosen to be $\gamma = 0.5$, and the other design parameters are taken to be $\lambda_1 = 10$, $\lambda_2 = 25$, $\lambda = 6$. Figure 8.4 shows the results when the commanded position is $r(t) = 0$, with initial conditions being $\dot{x}(0) = \dot{x}_m(0) = 0$ and $x(0) = x_m(0) = 0.5$. Figure 8.5 shows the results when the desired position is a sinusoidal signal, $r(t) = \sin(4t)$. It is clear that the position tracking errors in both cases converge to zero, while the parameter error converge to zero only for the latter case. The reason for the non-convergence of parameter error in the first case can be explained by the simplicity of the tracking task: the asymptotic tracking of $x_m(t)$ can be achieved by many possible values of the estimated parameter $\hat{m}$, besides the true parameter. Therefore, the parameter adaptation law does not bother to find out the true parameter. On the other hand, the convergence of the parameter error in Figure 8.5 is because of the complexity of the tracking task, i.e., tracking error convergence can be achieved only when the true mass is used in the control law. One may examine Equation (8.4) to see the mathematical evidence for these statements (also see Exercise 8.1). It is also helpful for the readers to sketch the specific structure of this adaptive mass control system. A more detailed discussion of parameter convergence is provided in section 8.2.
SELF-TUNING CONTROLLERS (STC)

In non-adaptive control design (e.g., pole placement), one computes the parameters of the controllers from those of the plant. If the plant parameters are not known, it is intuitively reasonable to replace them by their estimated values, as provided by a parameter estimator. A controller thus obtained by coupling a controller with an on-line (recursive) parameter estimator is called a self-tuning controller. Figure 8.6 illustrates the schematic structure of such an adaptive controller. Thus, a self-tuning controller is a controller which performs simultaneous identification of the unknown plant.

The operation of a self-tuning controller is as follows: at each time instant, the estimator sends to the controller a set of estimated plant parameters (\( \hat{a} \) in Figure 8.6),
which is computed based on the past plant input $u$ and output $y$; the computer finds the corresponding controller parameters, and then computes a control input $u$ based on the controller parameters and measured signals; this control input $u$ causes a new plant output to be generated, and the whole cycle of parameter and input updates is repeated. Note that the controller parameters are computed from the estimates of the plant parameters as if they were the true plant parameters. This idea is often called the certainty equivalence principle.

Parameter estimation can be understood simply as the process of finding a set of parameters that fits the available input-output data from a plant. This is different from parameter adaptation in MRAC systems, where the parameters are adjusted so that the tracking errors converge to zero. For linear plants, many techniques are available to estimate the unknown parameters of the plant. The most popular one is the least-squares method and its extensions. There are also many control techniques for linear plants, such as pole-placement, PID, LQR (linear quadratic control), minimum variance control, or $H^{\infty}$ designs. By coupling different control and estimation schemes, one can obtain a variety of self-tuning regulators. The self-tuning method can also be applied to some nonlinear systems without any conceptual difference.

In the basic approach to self-tuning control, one estimates the plant parameters and then computes the controller parameters. Such a scheme is often called indirect adaptive control, because of the need to translate the estimated parameters into controller parameters. It is possible to eliminate this part of the computation. To do this, one notes that the control law parameters and plant parameters are related to each other for a specific control method. This implies that we may reparameterize the plant model using controller parameters (which are also unknown, of course), and then use standard estimation techniques on such a model. Since no translation is needed in this scheme, it is called a direct adaptive control scheme. In MRAC systems, one can
simply consider direct and indirect ways of updating the controller parameters.

**Example 8.2: Self-tuning control of the unknown mass**

Consider the self-tuning control of the mass of Example 8.1. Let us still use the pole-placement (placing the poles of the tracking error dynamics) control law (8.3) for generating the control input, but let us now generate the estimated mass parameter using an estimation law.

Assume, for simplicity, that the acceleration can be measured by an accelerometer. Since the only unknown variable in Equation (8.1) is \( m \), the simplest way of estimating it is to simply divide the control input \( u(t) \) by the acceleration \( \ddot{x} \), i.e.,

\[
\hat{m}(t) = \frac{u(t)}{\ddot{x}}
\]

However, this is not a good method because there may be considerable noise in the measurement \( \ddot{x} \), and, furthermore, the acceleration may be close to zero. A better approach is to estimate the parameter using a least-squares approach, i.e., choosing the estimate in such a way that the total prediction error

\[
J = \int_0^T e^2(r) \, dr
\]

is minimal, with the prediction error \( e \) defined as

\[
e(t) = \hat{m}(t) \ddot{x}(t) - u(t)
\]

The prediction error is simply the error in fitting the known input \( u \) using the estimated parameter \( \hat{m} \). This total error minimization can potentially average out the effects of measurement noise. The resulting estimate is

\[
\hat{m} = \frac{\int_0^T w \, u \, dr}{\int_0^T w^2 \, dr}
\]

with \( w = \ddot{x} \). If, actually, the unknown parameter \( m \) is slowly time-varying, the above estimate has to be recalculated at every new time instant. To increase computational efficiency, it is desirable to adopt a recursive formulation instead of repeatedly using (8.11). To do this, we define

\[
P(t) = \frac{1}{\int_0^T w^2 \, dr}
\]

The function \( P(t) \) is called the estimation gain, and its update can be directly obtained by using
The two methods can be quite different in terms of analysis and implementation. Compared with MRAC controllers, ST controllers are more flexible because of the possibility of coupling various controllers with various estimators (i.e., the separation of control and estimation). However, the stability and convergence of self-tuning controllers are generally quite difficult to guarantee, often requiring the signals in the system to be sufficiently rich so that the estimated parameters converge to the true parameters. If the signals are not very rich (for example, if the reference signal is zero or a constant), the estimated parameters may not be close to the true parameters, and the stability and convergence of the resulting control system may not
be guaranteed. In this situation, one must either introduce perturbation signals in the input, or somehow modify the control law. In MRAC systems, however, the stability and tracking error convergence are usually guaranteed regardless of the richness of the signals.

Historically, the MRAC method was developed from optimal control of deterministic servomechanisms, while the ST method evolved from the study of stochastic regulation problems. MRAC systems have usually been considered in continuous-time form, and ST regulators in discrete time-form. In recent years, discrete-time version of MRAC controllers and continuous versions of ST controllers have also been developed. In this chapter, we shall mostly focus on MRAC systems in continuous form. Methods for generating estimated parameters for self-tuning control are discussed in section 8.7.

### 8.1.3 How To Design Adaptive Controllers ?

In conventional (non-adaptive) control design, a controller structure (e.g., pole placement) is chosen first, and the parameters of the controller are then computed based on the known parameters of the plant. In adaptive control, the major difference is that the plant parameters are unknown, so that the controller parameters have to be provided by an adaptation law. As a result, the adaptive control design is more involved, with the additional needs of choosing an adaptation law and proving the stability of the system with adaptation.

The design of an adaptive controller usually involves the following three steps:

- choose a control law containing variable parameters
- choose an adaptation law for adjusting those parameters
- analyze the convergence properties of the resulting control system

These steps are clearly seen in Example 8.1.

When one uses the self-tuning approach for linear systems, the first two steps are quite straightforward, with inventories of control and adaptation (estimation) laws available. The difficulty lies in the analysis. When one uses MRAC design, the adaptive controller is usually found by trial and error. Sometimes, the three steps are coordinated by the use of an appropriate Lyapunov function, or using some symbolic construction tools such as the passivity formalism. For instance, in designing the adaptive control system of Example 8.1, we actually start from guessing the Lyapunov function $V$ (as a representation of total error) in (8.7) and choose the control and
adaptation laws so that \( V \) decreases. Generally, the choices of control and adaptation laws in MRAC can be quite complicated, while the analysis of the convergence properties are relatively simple.

Before moving on to the application of the above procedure to adaptive control design for specific systems, let us derive a basic lemma which will be very useful in guiding our choice of adaptation laws for MRAC systems.

**Lemma 8.1:** Consider two signals \( e \) and \( \phi \) related by the following dynamic equation

\[
e(t) = H(p)[k\phi^T(t)v(t)]
\]

where \( e(t) \) is a scalar output signal, \( H(p) \) is a strictly positive real transfer function, \( k \) is an unknown constant with known sign, \( \phi(t) \) is a \( m \times 1 \) vector function of time, and \( v(t) \) is a measurable \( m \times 1 \) vector. If the vector \( \phi \) varies according to

\[
\dot{\phi}(t) = -\text{sgn}(k) \gamma e(t)v(t)
\]

with \( \gamma \) being a positive constant, then \( e(t) \) and \( \phi(t) \) are globally bounded. Furthermore, if \( v \) is bounded, then

\[
e(t) \to 0 \quad \text{as} \quad t \to \infty
\]

Note that while (8.15) involves a mixture of time-domain and frequency-domain notations (with \( p \) being the Laplace variable), its meaning is clear: \( e(t) \) is the response of the linear system of SPR transfer function \( H(p) \) to the input \( [k\phi^T(t)v(t)] \) (with arbitrary initial conditions). Such hybrid notation is common in the adaptive control literature, and later on it will save us the definition of intermediate variables.

In words, the above lemma means that if the input signal depends on the output in the form (8.16), then the whole system is globally stable (i.e., all its states are bounded). Note that this is a feedback system, shown in Figure 8.7, where the plant dynamics, being SPR, have the unique properties discussed in section 4.6.1.

**Proof:** Let the state-space representation of (8.15) be

\[
\dot{x} = Ax + b [k\phi^T v]
\]

\[
e = c^Tx
\]

Since \( H(p) \) is SPR, it follows from the Kalman-Yakubovich lemma in chapter 4 that given a symmetric positive definite matrix \( Q \), there exists another symmetric positive definite matrix \( P \) such that

\[
A^TP + PA = -Q
\]
Let $V$ be a positive definite function of the form
\[ V[x, \phi] = x^T P x + \frac{|k|}{y} \phi^T \phi \] (8.18)

Its time derivative along the trajectories of the system defined by (8.17) and (8.16) is
\[ \dot{V} = x^T (PA + A^T P)x + 2x^T Pb (k \phi^T v) - 2 \phi^T (k e v) \]
\[ = -x^T Q x \leq 0 \] (8.19)

Therefore, the system defined by (8.15) and (8.16) is globally stable. The equations (8.18) and (8.19) also imply that $e$ and $\phi$ are globally bounded.

If the signal $v(t)$ is bounded, $\dot{x}$ is also bounded, as seen from (8.17a). This implies the uniform continuity of $\dot{V}$, since its derivative
\[ \ddot{V} = -2x^T Q x \]
is then bounded. Application of Barbalat's lemma of chapter 4 then indicates the asymptotic convergence of $e(t)$ to zero.

It is useful to point out that the system defined by (8.15) and (8.16) not only guarantees the boundedness of $e$ and $\phi$, but also that of the whole state $x$, as seen from (8.18). Note that the state-space realization in (8.17) can be non-minimal (implying the possibility of unobservable or uncontrollable modes) provided that the unobservable and uncontrollable modes be stable, according to the Meyer-Kalman-Yakubovich lemma. Intuitively, this is reasonable because stable hidden modes are not affected by the choice of $\phi$.

In our later MRAC designs, the tracking error between the plant output and
reference model output will often be related to the parameter estimation errors by an equation of the form (8.15). Equation (8.16) thus provides a technique for adjusting the controller parameters while guaranteeing system stability. Clearly, the tracking-error dynamics in (8.4) satisfy the conditions of Lemma 8.1 and the adaptation law is in the form of (8.16).

8.2 Adaptive Control of First-Order Systems

Let us now discuss the adaptive control of first-order plants using the MRAC method, as an illustration of how to design and analyze an adaptive control system. The development can also have practical value in itself, because a number of simple systems of engineering interest may be represented by a first-order model. For example, the braking of an automobile, the discharge of an electronic flash, or the flow of fluid from a tank may be approximately represented by a first-order differential equation

\[ \dot{y} = -a_p y + b_p u \] (8.20)

where \( y \) is the plant output, \( u \) is its input, and \( a_p \) and \( b_p \) are constant plant parameters.

**PROBLEM SPECIFICATION**

In the adaptive control problem, the plant parameters \( a_p \) and \( b_p \) are assumed to be unknown. Let the desired performance of the adaptive control system be specified by a first-order reference model

\[ \dot{y}_m = -a_m y_m + b_m r(t) \] (8.21)

where \( a_m \) and \( b_m \) are constant parameters, and \( r(t) \) is a bounded external reference signal. The parameter \( a_m \) is required to be strictly positive so that the reference model is stable, and \( b_m \) is chosen strictly positive without loss of generality. The reference model can be represented by its transfer function \( M \)

\[ y_m = M r \]

where

\[ M = \frac{b_m}{p + a_m} \]

with \( p \) being the Laplace variable. Note that \( M \) is a SPR function.

The objective of the adaptive control design is to formulate a control law, and
an adaptation law, such that the resulting model following error $y(t) - y_m$ asymptotically converges to zero. In order to accomplish this, we have to assume the sign of the parameter $b$ to be known. This is a quite mild condition, which is often satisfied in practice. For example, for the braking of a car, this assumption amounts to the simple physical knowledge that braking slows down the car.

**CHOICE OF CONTROL LAW**

As the first step in the adaptive controller design, let us choose the control law to be

$$u = \hat{a}_r(t)r + \hat{a}_y(t)y$$ \hspace{1cm} (8.22)

where $\hat{a}_r$ and $\hat{a}_y$ are variable feedback gains. With this control law, the closed-loop dynamics are

$$\dot{y} = -(a_p - \hat{a}_y b_p)y + \hat{a}_r b_p r(t)$$ \hspace{1cm} (8.23)

The reason for the choice of control law in (8.22) is clear: it allows the possibility of perfect model matching. Indeed, if the plant parameters were known, the following values of control parameters

$$a_r^* = \frac{b_m}{b_p} \quad a_y^* = \frac{a_p - a_m}{b_p}$$ \hspace{1cm} (8.24)

would lead to the closed-loop dynamics

$$\dot{y} = -a_m y + b_m r$$

which is identical to the reference model dynamics, and yields zero tracking error. In this case, the first term in (8.22) would result in the right d.c. gain, while the second term in the control law (8.22) would achieve the dual objectives of canceling the term $(-a_p y)$ in (8.20) and imposing the desired pole $-a_m$.

In our adaptive control problem, since $a_p$ and $b_p$ are unknown, the control input will achieve these objectives adaptively, i.e., the adaptation law will continuously search for the right gains, based on the tracking error $y - y_m$, so as to make $y$ tend to $y_m$ asymptotically. The structure of the adaptive controller is illustrated in Figure 8.8.

**CHOICE OF ADAPTATION LAW**

Let us now choose the adaptation law for the parameters $\hat{a}_r$ and $\hat{a}_y$. Let

$$e = y - y_m$$

be the tracking error. The parameter errors are defined as the difference between the
controller parameter provided by the adaptation law and the ideal parameters, i.e.,

\[
\vec{a}(t) = \begin{bmatrix}
\hat{a}_r \\
\hat{a}_y
\end{bmatrix} = \begin{bmatrix}
\hat{a}_r - a_r^* \\
\hat{a}_y - a_y^*
\end{bmatrix}
\]  
(8.25)

The dynamics of tracking error can be found by subtracting (8.23) from (8.21),

\[
\dot{e} = -a_m(y - y_m) + (a_m - a_p + b_p \hat{a}_y)y + (b_p \hat{a}_r - b_m)r
\]

\[
= -a_m e + b_p (\hat{a}_r r + \hat{a}_y y)
\]  
(8.26)

This can be conveniently represented as

\[
e = \frac{b_p}{p + a_m} (\hat{a}_r r + \hat{a}_y y) = \frac{1}{a_r^*} M (\hat{a}_r r + \hat{a}_y y)
\]  
(8.27)

with \(p\) denoting the Laplace operator.

Relation (8.27) between the parameter errors and tracking error is in the familiar form given by (8.15). Thus, Lemma 8.1 suggests the following adaptation law

\[
\hat{a}_r = - \text{sgn}(b_p) \gamma e r
\]  
(8.28a)

\[
\hat{a}_y = - \text{sgn}(b_p) \gamma e y
\]  
(8.28b)

with \(\gamma\) being a positive constant representing the adaptation gain. From (8.28), it is seen that \(\text{sgn}(b_p)\) determines the direction of the search for the proper controller parameters.
TRACKING CONVERGENCE ANALYSIS

With the control law and adaptation law chosen above, we can now analyze the system's stability and convergence behavior using Lyapunov theory, or equivalently Lemma 8.1. Specifically, the Lyapunov function candidate

$$V(e, \phi) = \frac{1}{2} e^2 + \frac{1}{2\gamma |b_p| (\hat{a}_r^2 + \hat{a}_y^2)}$$

(8.29)

can be easily shown to have the following derivative along system trajectories

$$\dot{V} = -a_m e^2$$

Thus, the adaptive control system is globally stable, i.e., the signals $e$, $\hat{a}_r$, and $\hat{a}_y$ are bounded. Furthermore, the global asymptotic convergence of the tracking error $\dot{e}(t)$ is guaranteed by Barbalat's lemma, because the boundedness of $e$, $\hat{a}_r$, and $\hat{a}_y$ implies the boundedness of $\dot{e}$ (according to (8.26)) and therefore the uniform continuity of $\dot{V}$.

It is interesting to wonder why the adaptation law (8.28) leads to tracking error convergence. To understand this, let us see intuitively how the control parameters should be changed. Consider, without loss of generality, the case of a positive $\text{sgn}(b_p)$. Assume that at a particular instant $t$ the tracking error $e$ is negative, indicating that the plant output is too small. From (8.20), the control input $u$ should be increased in order to increase the plant output. From (8.22), an increase of the control input $u$ can be achieved by increasing $\hat{a}_r$ (assuming that $r(t)$ is positive). Thus, the adaptation law, with the variation rate of $\hat{a}_r$ depending on the product of $\text{sgn}(b)$, $r$ and $e$, is intuitively reasonable. A similar reasoning can be made about $\hat{a}_y$.

The behavior of the adaptive controller is demonstrated in the following simulation example.

**Example 8.3: A first-order plant**

Consider the control of the unstable plant

$$\dot{y} = y + 3u$$

using the previously designed adaptive controller. The plant parameters $a_p = -1$, $b_p = 3$ are assumed to be unknown to the adaptive controller. The reference model is chosen to be

$$\dot{x}_m = -4 x_m + 4 r$$

i.e., $a_m = 4$, $b_m = 4$. The adaptation gain $\gamma$ is chosen to be equal to 2. The initial values of both parameters of the controller are chosen to be 0, indicating no a priori knowledge. The initial conditions of the plant and the model are both zero.
Two different reference signals are used in the simulation:

- $r(t) = 4$. It is seen from Figure 8.9 that the tracking error converges to zero but the parameter error does not.

- $r(t) = 4 \sin(3t)$. It is seen from Figure 8.10 that both the tracking error and parameter error converge to zero.

![Figure 8.9: Tracking performance and parameter estimation, $r(t) = 4$](image1)

![Figure 8.10: Tracking performance and parameter estimation, $r(t) = 4 \sin(3t)$](image2)

Note that, in the above adaptive control design, although the stability and convergence of the adaptive controller is guaranteed for any positive $\gamma$, $a_m$, and $b_m$, the performance of the adaptive controller will depend critically on $\gamma$. If a small gain is chosen, the adaptation will be slow and the transient tracking error will be large. Conversely, the magnitude of the gain and, accordingly, the performance of the adaptive control system, are limited by the excitation of unmodeled dynamics, because too large an adaptation gain will lead to very oscillatory parameters.
PARAMETER CONVERGENCE ANALYSIS

In order to gain insights about the behavior of adaptive control system, let us understand the convergence of estimated parameters. From the simulation results of Example 8.3, one notes that the estimated parameters converge to the exact parameter values for one reference signal but not for the other. This prompts us to speculate a relation between the features of the reference signals and parameter convergence, i.e., the estimated parameters will not converge to the ideal controller parameters unless the reference signal $r(t)$ satisfies certain conditions.

Indeed, such a relation between the features of reference signal and convergence of estimated parameters can be intuitively understood. In MRAC systems, the objective of the adaptation mechanism is to find out the parameters which drive the tracking error $y - y_m$ to zero. If the reference signal $r(t)$ is very simple, such as a zero or a constant, it is possible for many vectors of controller parameters, besides the ideal parameter vector, to lead to tracking error convergence. Then, the adaptation law will not bother to find out the ideal parameters. Let $\Omega$ denote the set composed of all the parameter vectors which can guarantee tracking error convergence for a particular reference signal history $r(t)$. Then, depending on the initial conditions, the vector of estimated parameters may converge to any point in the set or wonder around in the set instead of converging to the true parameters. However, if the reference signal $r(t)$ is so complex that only the true parameter vector $a^* = [a^*_r \ a^*_y]^T$ can lead to tracking error convergence, then we shall have parameter convergence.

Let us now find out the exact conditions for parameter convergence. We shall use simplified arguments to avoid tedious details. Note that the output of the stable filter in (8.27) converges to zero and that its input is easily shown to be uniformly continuous. Thus, $\tilde{a}_r r + \tilde{a}_y y$ must converge to zero. From the adaptation law (8.28) and the tracking error convergence, the rate of the parameter estimates converges to zero. Thus, when time $t$ is large, $\dot{\tilde{a}}$ is almost constant, and

$$r(t)\tilde{a}_r + y(t)\tilde{a}_y = 0$$

i.e.,

$$v^T(t) \dot{\tilde{a}} = 0$$

with

$$v = [r \ y]^T \quad a = [a_r \ a_y]^T$$

Here we have one equation (with time-varying coefficients) and two variables. The issue of parameter convergence is reduced to the question of what conditions the
vector \([r(t)\ y(t)]^T\) should satisfy in order for the equation to have a unique zero solution.

If \(r(t)\) is a constant \(r_o\), then for large \(t\),

\[
y(t) = y_m = \alpha r_o
\]

with \(\alpha\) being the d.c. gain of the reference model. Thus,

\[
[r\ y] = [1\ \alpha] r_o
\]

Equation (8.30) becomes

\[
\tilde{a}_r + \alpha \tilde{a}_y = 0
\]

Clearly, this implies that the estimated parameters, instead of converging to zero, converge to a straight line in parameter space. For Example 8.3, with \(\alpha = 1\), the above equation implies that the steady state errors of the two parameters should be of equal magnitudes but opposite signs. This is obviously confirmed in Figure 8.9.

However, when \(r(t)\) is such that the corresponding signal vector \(v(t)\) satisfies the so called "persistent excitation" condition, we can show that (8.28) will guarantee parameter convergence. By persistent excitation of \(v\), we mean that there exist strictly positive constants \(\alpha_1\) and \(T\) such that for any \(t > 0\),

\[
\int_t^{t+T} v v^T dr \geq \alpha_1 I
\]

(8.31)

To show parameter convergence, we note that multiplying (8.30) by \(v(t)\) and integrating the equation for a period of time \(T\), leads to

\[
\int_t^{t+T} v v^T dr \tilde{a} = 0
\]

Condition (8.31) implies that the only solution of this equation is \(\tilde{a} = 0\), i.e., parameter error being zero. Intuitively, the persistent excitation of \(v(t)\) implies that the vectors \(v(t)\) corresponding to different times \(t\) cannot always be linearly dependent.

The only remaining question is the relation between \(r(t)\) and the persistent excitation of \(v(t)\). One can easily show that, in the case of the first order plant, the persistent excitation of \(v\) can be guaranteed, if \(r(t)\) contains at least one sinusoidal component.
EXTENSION TO NONLINEAR PLANTS

The same method of adaptive control design can be used for the nonlinear first-order plant described by the differential equation

\[ \dot{y} = -a_p y - c_p f(y) + b_p u \]  

(8.32)

where \( f \) is any known nonlinear function. The nonlinearity in these dynamics is characterized by its linear parametrization in terms of the unknown constant \( c \). Instead of using the control law (8.22), we now use the control law

\[ u = \hat{a}_y y + \hat{a}_f f(y) + \hat{a}_r r \]  

(8.33)

where the second term is introduced with the intention of adaptively canceling the nonlinear term.

Substituting this control law into the dynamics (8.32) and subtracting the resulting equation by (8.21), we obtain the error dynamics

\[ e = \frac{1}{k_r} M (\hat{a}_y y + \hat{a}_f f(y) + \hat{a}_r r) \]

where the parameter error \( \tilde{a}_f \) is defined as

\[ \tilde{a}_f = \hat{a}_f - \frac{c_p}{b_p} \]

By choosing the adaptation law

\[ \dot{\hat{a}}_y = -\text{sgn}(b_p) \gamma e y \]  

(8.34a)

\[ \dot{\hat{a}}_f = -\text{sgn}(b_p) \gamma e f \]  

(8.34b)

\[ \dot{\hat{a}}_r = -\text{sgn}(b_p) \gamma e r \]  

(8.34c)

one can similarly show that the tracking error \( e \) converges to zero, and the parameter error remains bounded.

As for parameter convergence, similar arguments as before can reveal the convergence behavior of the estimated parameters. For constant reference input \( r = r_0 \), the estimated parameters converge to the line (with \( \alpha \) still being the d.c. gain of the reference model)

\[ r_0 \tilde{a}_r + \tilde{a}_y (\alpha r_0) + \tilde{a}_f (\alpha r_0) = 0 \]

which is a straight line in the three-dimensional parameter space. In order for the
parameters to converge to the ideal values, the signal vector \( \mathbf{v} = [r(t) \ y(t) \ f(y)]^T \) should be persistently exciting, i.e., there exists positive constants \( \alpha_j \) and \( T \) such that for any time \( t \geq 0 \),

\[
\int_{t}^{t+T} \mathbf{v}^T \mathbf{r} \, dt \geq \alpha_1 \mathbf{I}
\]

Generally speaking, for linear systems, the convergent estimation of \( m \) parameters require at least \( m/2 \) sinusoids in the reference signal \( r(t) \), as will be discussed in more detail in sections 8.3 and 8.4. However, for this nonlinear case, such simple relation may not be valid. Usually, the qualitative relation between \( r(t) \) and \( \mathbf{v}(t) \) is dependent on the particular nonlinear functions \( f(y) \). It is unclear how many sinusoids in \( r(t) \) are necessary to guarantee the persistent excitation of \( \mathbf{v}(t) \).

The following example illustrates the behavior of the adaptive system for a nonlinear plant.

**Example 8.4: simulation of a first-order nonlinear plant**

Assume that a nonlinear plant is described by the equation

\[
\dot{y} = y + y^2 + bu
\]

This differs from the unstable plant in Example 8.3 in that a quadratic term is introduced in the plant dynamics.

Let us use the same reference model, initial parameters, and design parameters as in Example 8.3. For the reference signal \( r(t) = 4 \), the results are shown in Figure 8.11. It is seen that the tracking error converges to zero, but the parameter errors are only bounded. For the reference signal \( r(t) = 4 \sin(3t) \), the results are shown in Figure 8.12. It is noted that the tracking error and the parameter errors for the three parameters all converge to zero.

In this example, it is interesting to note two points. The first point is that a single sinusoidal component in \( r(t) \) allows three parameters to be estimated. The second point is that the various signals (including \( \dot{y} \) and \( y \)) in this system are much more oscillatory than those in Example 8.3. Let us understand why. The basic reason is provided by the observation that nonlinearity usually generates more frequencies, and thus \( \mathbf{v}(t) \) may contain more sinusoids than \( r(t) \). Specifically, in the above example, with \( r(t) = 4 \sin(3t) \), the signal vector \( \mathbf{v}(t) \) converges to

\[
\mathbf{v}(t) = [r(t) \ y_{ss}(t) \ f_{ss}(t)]
\]

where \( y_{ss}(t) \) is the steady-state response and \( f_{ss}(t) \) the corresponding function value.
Figure 8.11: Adaptive control of a first-order nonlinear system, \( r(t) = 4 \)

upper left: tracking performance
upper right: parameter \( \hat{a}_s \); lower left: parameter \( \hat{a}_r \); lower right: parameter \( \hat{a}_f \)

\[
\begin{align*}
\dot{y}_{ss}(t) &= y_{m}(t) = 4A \sin(3t + \phi) \\
\dot{f}_{ss}(t) &= y_{ss}^2 = 16A^2 \sin^2(3t + \phi) = 8A^2(1 - \cos(6t + 2\phi))
\end{align*}
\]

where \( A \) and \( \phi \) are the magnitude and phase shift of the reference model at \( \omega = 3 \).

Thus, the signal vector \( v(t) \) contains two sinusoids, with \( f(y) \) containing a sinusoid at twice the original frequency. Intuitively, this component at double frequency is the reason for the convergent estimation of the three parameters and the more oscillatory behavior of the estimated parameters.

### 8.3 Adaptive Control of Linear Systems With Full State Feedback

Let us now move on to adaptive control design for more general systems. In this section, we shall study the adaptive control of linear systems when the full state is measurable. We consider the adaptive control of \( n \)-th order linear systems in
Figure 8.12: Adaptive control of a first-order nonlinear system, $r(t) = 4\sin(3t)$

- upper left: tracking performance
- upper right: parameter $\hat{a}_r$
- lower left: parameter $\hat{a}_y$
- lower right: parameter $\hat{a}_f$

Companion form:

$$a_n y^{(n)} + a_{n-1} y^{(n-1)} + \ldots + a_0 y = u$$  \hspace{1cm} (8.36)

where the state components $y, \dot{y}, \ldots, y^{(n-1)}$ are measurable. We assume that the coefficient vector $a = [a_n \ldots a_1 a_0]^T$ is unknown, but that the sign of $a_n$ is assumed to be known. An example of such systems is the dynamics of a mass-spring-damper system

$$m\ddot{y} + c\dot{y} + ky = u$$

where we measure position and velocity (possibly with an optical encoder for position measurement, and a tachometer for velocity measurement, or simply numerically differentiate the position signals).

The objective of the control system is to make $y$ closely track the response of a stable reference model.
with \( r(t) \) being a bounded reference signal.

CHOICE OF CONTROL LAW

Let us define a signal \( z(t) \) as follows

\[
z(t) = y^{(n)} - \beta_{n-1} e^{(n-1)} - \ldots - \beta_0 e
\]

with \( \beta_1, \ldots, \beta_n \) being positive constants chosen such that \( p^n + \beta_{n-1} p^{n-1} + \ldots + \beta_0 \) is a stable (Hurwitz) polynomial. Adding \((-a_n z(t))\) to both sides of (8.36) and rearranging, we can rewrite the plant dynamics as

\[
a_n [y^{(n)} - z] = u - a_n z - a_{n-1} y^{(n-1)} - \ldots - a_0 y
\]

Let us choose the control law to be

\[
u = a_n z + a_{n-1} y^{(n-1)} + \ldots + a_0 y = v^T(t) \hat{a}(t)
\]

where

\[
v(t) = [z(t), y^{(n-1)}, \ldots, y]^T
\]

\[
\hat{a}(t) = [\hat{a}_n, \hat{a}_{n-1}, \ldots, \hat{a}_1, \hat{a}_0]^T
\]

denotes the estimated parameter vector. This represents a pole-placement controller which places the poles at positions specified by the coefficients \( \beta_i \). The tracking error \( e = y - y_m \) then satisfies the closed-loop dynamics

\[
a_n [e^{(n)} + \beta_{n-1} e^{(n-1)} + \ldots + \beta_0 e] = v^T(t) \tilde{a}(t)
\]

where

\[
\tilde{a} = \hat{a} - a
\]

CHOICE OF ADAPTATION LAW

Let us now choose the parameter adaptation law. To do this, let us rewrite the closed-loop error dynamics (8.40) in state space form,

\[
\dot{x} = A x + b [(1/a_n) v^T \tilde{a}]
\]

\[
e = c x
\]

where
Consider the Lyapunov function candidate
\[ V(x, \tilde{a}) = x^T P x + \tilde{a}^T \Gamma^{-1} \tilde{a} \]
where both \( \Gamma \) and \( P \) are symmetric positive definite constant matrices, and \( P \) satisfies
\[ PA + A^T P = -Q \quad Q = Q^T > 0 \]
for a chosen \( Q \). The derivative \( \dot{V} \) can be computed easily as
\[ \dot{V} = -x^T Q x + 2 \tilde{a}^T v b^T P x + 2 \tilde{a}^T \Gamma^{-1} \tilde{a} \]
Therefore, the adaptation law
\[ \dot{\tilde{a}} = -\Gamma v b^T P x \quad (8.42) \]
leads to
\[ \dot{V} = -x^T Q x \]
One can easily show the convergence of \( x \) using Barbalat’s lemma. Therefore, with the adaptive controller defined by control law (8.39) and adaptation law (8.42), \( c \) and its \((n-1)\) derivatives converge to zero. The parameter convergence condition can again be shown to be the persistent excitation of the vector \( v \). Note that a similar design can be made for nonlinear systems in the controllability canonical form, as discussed in section 8.5.
8.4 Adaptive Control of Linear Systems With Output Feedback

In this section, we consider the adaptive control of linear systems in the presence of only output measurement, rather than full state feedback. Design in this case is considerably more complicated than when the full state is available. This partly arises from the need to introduce dynamics in the controller structure, since the output only provides partial information about the system state. To appreciate this need, one can simply recall that in conventional design (no parameter uncertainty) a controller obtained by multiplying the state with constant gains (pole placement) can stabilize systems where all states are measured, while additional observer structures must be used for systems where only outputs are measured.

A linear time-invariant system can be represented by the transfer function

\[ W(p) = k_p \frac{Z_p(p)}{R_p(p)} \]  

where

\[ R_p(p) = a_0 + a_1 p + \ldots + a_{n-1} p^{n-1} + p^n \]

\[ Z_p(p) = b_0 + b_1 p + \ldots + b_{m-1} p^{m-1} + p^m \]

where \( k_p \) is called the high-frequency gain. The reason for this term is that the plant frequency response at high frequency verifies

\[ |W(j\omega)| = \frac{k_p}{\omega^{n-m}} \]

_i.e., the high frequency response is essentially determined by \( k_p \). The relative degree \( r \) of this system is \( r = n - m \). In our adaptive control problem, the coefficients \( a_i, b_j \) \((i = 0, 1, \ldots, n-1; j = 0, 1, \ldots, m-1)\) and the high frequency gain \( k_p \) are all assumed to be unknown.

The desired performance is assumed to be described by a reference model with transfer function

\[ W_m(p) = k_m \frac{Z_m}{R_m} \]

where \( Z_m \) and \( R_m \) are monic Hurwitz polynomials of degrees \( n_m \) and \( m_m \), and \( k_m \) is positive. It is well known from linear system theory that the relative degree of the
reference model has to be larger than or equal to that of the plant in order to allow the possibility of perfect tracking. Therefore, in our treatment, we will assume that \( n_m - m_m \geq n - m \).

The objective of the design is to determine a control law, and an associated adaptation law, so that the plant output \( y \) asymptotically approaches \( y_m \). In determining the control input, the output \( y \) is assumed to be measured, but no differentiation of the output is allowed, so as to avoid the noise amplification associated with numerical differentiation. In achieving this design, we assume the following \textit{a priori} knowledge about the plant:

- the plant order \( n \) is known
- the relative degree \( n - m \) is known
- the sign of \( k_p \) is known
- the plant is minimum-phase

Among the above assumptions, the first and the second imply that the model structure of the plant is known. The third is required to provide the direction of parameter adaptation, similarly to (8.28) in section 8.2. The fourth assumption is somewhat restrictive. It is required because we want to achieve convergent tracking in the adaptive control design. Adaptive control of non-minimum phase systems is still a topic of active research and will not be treated in this chapter.

In section 8.4.1, we discuss output-feedback adaptive control design for linear plants with relative degree one, \( i.e., \) plants having one more pole than zeros. Design for these systems is relatively straightforward. In section 8.4.2, we discuss output-feedback design for plants with higher relative degree. The design and implementation of adaptive controllers in this case is more complicated because it is not possible to use SPR functions as reference models.

### 8.4.1 Linear Systems With Relative Degree One

When the relative degree is 1, \( i.e., \) \( m = n - 1 \), the reference model can be chosen to be SPR. This choice proves critical in the development of globally convergent adaptive controllers.

**CHOICE OF CONTROL LAW**

To determine the appropriate control law for the adaptive controller, we must first know what control law can achieve perfect tracking when the plant parameters are
perfectly known. Many controller structures can be used for this purpose. The following one, although somewhat peculiar, is particularly convenient for later adaptation design.

**Example 8.5: A controller for perfect tracking**

Consider the plant described by

\[
y = \frac{k_p(p + b)}{p^2 + a_p1p + a_p2} u
\]  

(8.45)

and the reference model

\[
y_m = \frac{k_m(p + b_m)}{p^2 + a_m1p + a_m2} r
\]  

(8.46)

![Figure 8.13: A model-reference control system for relative degree 1](image)

Let the controller be chosen as shown in Figure 8.13, with the control law being

\[
u = \alpha_1 z + \frac{\beta_1 p + \beta_2}{p + b_m} y + k r
\]  

(8.47)

where \( z = u(p + b_m) \), i.e., \( z \) is the output of a first-order filter with input \( u \), and \( \alpha_1, \beta_1, \beta_2, k \) are controller parameters. If we take these parameters to be

\[
\alpha_1 = b_p - b_m
\]

\[
\beta_1 = \frac{a_m1 - a_p1}{k_p}
\]
one can straightforwardly show that the transfer function from the reference input $r$ to the plant output $y$ is

$$ W_{y,y} = \frac{k_m(p + b_m)}{p^2 + a_{m1}p + a_{m2}} = W_m(p) $$

Therefore, perfect tracking is achieved with this control law, i.e., $y(t) = y_m(t)$, $\forall t \geq 0$.

It is interesting to see why the closed-loop transfer function can become exactly the same as that of the reference model. To do this, we note that the control input in (8.47) is composed of three parts. The first part in effect replaces the plant zero by the reference model zero, since the transfer function from $M_1$ to $y$ (see Figure 8.13) is

$$ W_{u1,y} = \frac{p + b_m}{p + b_p} \frac{k_p(p + b_p)}{p^2 + a_{p1}p + a_{p2}} = \frac{k_p(p + b_m)}{p^2 + a_{p1}p + a_{p2}} $$

The second part places the closed-loop poles at the locations of those of the reference model. This is seen by noting that the transfer function from $u_0$ to $y$ is (Figure 8.13)

$$ W_{u0,y} = \frac{W_{u1,y}}{1 + W_f W_{u1,y}} = \frac{k_p(p + b_m)}{p^2 + (a_{p1} + \beta_1 k_p)p + (a_{p2} + \beta_2 k_p)} $$

The third part of the control law ($k_m/k_p$) obviously replaces $k_p$, the high frequency gain of the plant, by $k_m$. As a result of the above three parts, the closed-loop system has the desired transfer function.

The controller structure shown in Figure 8.13 for second-order plants can be extended to any plant with relative degree one. The resulting structure of the control system is shown in Figure 8.14, where $k^*$, $\theta_1^*$, $\theta_2^*$ and $\theta_0^*$ represents controller parameters which lead to perfect tracking when the plant parameters are known.

The structure of this control system can be described as follows. The block for generating the filter signal $\omega_1$ represents an $(n-1)^{th}$ order dynamics, which can be described by

$$ \dot{\omega}_1 = \Lambda \omega_1 + hu $$

where $\omega_1$ is an $(n-1) \times 1$ state vector, $\Lambda$ is an $(n-1) \times (n-1)$ matrix, and $h$ is a constant vector such that $(\Lambda, h)$ is controllable. The poles of the matrix $\Lambda$ are chosen to be the
same as the roots of the polynomial $Z_m(p)$, i.e.,

$$\det[pI - A] = Z_m(p) \tag{8.48}$$

The block for generating the $(n-1)\times 1$ vector $\omega_2$ has the same dynamics but with $y$ as input, i.e.,

$$\dot{\omega}_2 = \Lambda \omega_2 + h y$$

It is straightforward to discuss the controller parameters in Figure 8.14. The scalar gain $k^*$ is defined to be

$$k^* = \frac{k_m}{k_p}$$

and is intended to modulate the high-frequency gain of the control system. The vector $\theta_1^*$ contains $(n-1)$ parameters which intend to cancel the zeros of the plant. The vector $\theta_2^*$ contains $(n-1)$ parameters which, together with the scalar gain $\theta_0^*$ can move the poles of the closed-loop control system to the locations of the reference model poles. Comparing Figure 8.13 and Figure 8.14 will help the reader become familiar with this structure and the corresponding notations.

As before, the control input in this system is a linear combination of the reference signal $r(t)$, the vector signal $\omega_1$ obtained by filtering the control input $u$, the signals $\omega_2$ obtained by filtering the plant output $y$, and the output itself. The control input $u$ can thus be written, in terms of the adjustable parameters and the various
Adaptive Control

Chap. 8

signals, as

\[ u^*(t) = k^* r + \theta_1^* \omega_1 + \theta_2^* \omega_2 + \theta_o^* y \]  
(8.49)

Corresponding to this control law and any reference input \( r(t) \), the output of the plant is

\[ y(t) = \frac{B(p)}{A(p)} u^*(t) = W_m r(t) \]  
(8.50)

since these parameters result in perfect tracking. At this point, one easily sees the reason for assuming the plant to be minimum-phase: this allows the plant zeros to be canceled by the controller poles.

In the adaptive control problem, the plant parameters are unknown, and the ideal control parameters described above are also unknown. Instead of (8.49), the control law is chosen to be

\[ u = k(t) r + \theta_1(t) \omega_1 + \theta_2(t) \omega_2 + \theta_o(t) y \]  
(8.51)

where \( k(t), \theta_1(t), \theta_2(t), \) and \( \theta_o(t) \) are controller parameters to be provided by the adaptation law.

CHOICE OF ADAPTATION LAW

For notational simplicity, let \( \Theta \) be the \( 2n \times 1 \) vector containing all the controller parameters, and \( \Omega \) be the \( 2n \times 1 \) vector containing the corresponding signals, i.e.,

\[ \Theta(t) = [k(t) \ \theta_1(t) \ \theta_2(t) \ \theta_o(t)]^T \]

\[ \Omega(t) = [r(t) \ \omega_1(t) \ \omega_2(t) \ y(t)]^T \]

Then the control law (8.51) can be compactly written as

\[ u = \Theta^T(t) \Omega(t) \]  
(8.52)

Denoting the ideal value of \( \Theta \) by \( \Theta^* \) and the error between \( \Theta(t) \) and \( \Theta^* \) by \( \Phi(t) = \Theta(t) - \Theta^* \), the estimated parameters \( \Theta(t) \) can be represented as

\[ \Theta(t) = \Theta^* + \Phi(t) \]

Therefore, the control law (8.52) can also be written as

\[ u = \Theta^*^T \Omega + \Phi^T(t) \Omega \]

In order to choose an adaptation law so that the tracking error \( e \) converges to zero, we
have to first find out how the tracking error is related to the parameter error. Let us use a simple technique for this purpose.

With the control law given in (8.52), the control system with variable gains can be equivalently represented as shown in Figure 8.15, with $\phi^T(t)\omega/k^*$ regarded as an external signal. Since the ideal parameter vector $\theta^*$ is such that the plant output in Figure 8.15 is given by (8.50), the output here must be

$$y(t) = W_m(p)r + W_m(p)[\phi^T(t)\omega/k^*]$$  \hspace{1cm} (8.53)

Figure 8.15: An equivalent control system for time-varying gains

Since $y_m(t) = W_m(p)r$, the tracking error is seen to be related to the parameter error by the simple equation

$$e(t) = W_m(p)[\phi^T(t)\omega(t)/k^*]$$  \hspace{1cm} (8.54)

Since this is the familiar equation seen in Lemma 8.1, the following adaptation law is chosen

$$\dot{\theta} = -\text{sgn}(k_p)\gamma e(t)\omega(t)$$  \hspace{1cm} (8.55)

where $\gamma$ is a positive number representing the adaptation gain and we have used the fact that the sign of $k^*$ is the same as that of $k_p$, due to the assumed positiveness of $k_m$.

Based on Lemma 8.1 and through a straightforward procedure for establishing signal boundedness, one can show that the tracking error in the above adaptive control system converges to zero asymptotically.
8.4.2 Linear Systems With Higher Relative Degree

The design of adaptive controller for plants with relative degree larger than 1 is both similar to, and different from, that for plants with relative degree 1. Specifically, the choice of control law is quite similar but the choice of adaptation law is very different. This difference comes from the fact that the reference model now cannot be SPR.

**CHOICE OF CONTROL LAW**

We can show that the controller part of the system in Figure 8.15 is also applicable to plants with relative degree larger than 1, leading to exact tracking when the plant parameters are exactly known. Let us again start from a simple example.

Example 8.6: Consider the second order plant described by the transfer function

\[ y = \frac{k_p u}{p^2 + a_{p1}p + a_{p2}} \]

and the reference model

\[ y_m = \frac{k_m r}{p^2 + a_{m1}p + a_{m2}} \]

which are similar to those in Example 8.5, but now contain no zeros.

Let us consider the control structure shown in Figure 8.16 which is a slight modification of the controller structure in Figure 8.13. Note that \(b_m\) in the filters in Figure 8.13 has been replaced by a positive number \(X_0\). Of course, the transfer functions \(W_p\) and \(W_m\) in Figure 8.16 now have relative degree 2.

The closed-loop transfer function from the reference signal \(r\) to the plant output \(y\) is

\[ W_{ry} = k \frac{p + \lambda_o}{1 + \lambda_o} \frac{p + \alpha_1 p^2 + a_{p1}p + a_{p2}}{p + \lambda_o + \alpha_1 p + a_{p1} + a_{p2}} \]

\[ = \frac{kp(p + \lambda_o)}{(p + \alpha_1)(p^2 + a_{p1}p + a_{p2}) + k_p(\beta_1 p + \beta_2)} \]

Therefore, if the controller parameters \(\alpha_1, \beta_1, \beta_2\), and \(k\) are chosen such that

\[ (p + \lambda_o + \alpha_1)(p^2 + a_{p1}p + a_{p2}) + k_p(\beta_1 p + \beta_2) = (p + \lambda_o)(p^2 + a_{m1}p + a_{m2}) \]

and
then the closed loop transfer function $W_{ry}$ becomes identically the same as that of the reference model. Clearly, such choice of parameters exists and is unique.

For general plants of relative degree larger than 1, the same control structure as given in Figure 8.14 is chosen. Note that the order of the filters in the control law is still $(n - 1)$. However, since the model numerator polynomial $Z_{m}(p)$ is of degree smaller than $(n - 1)$, it is no longer possible to choose the poles of the filters in the controller so that $\det[p I - \Lambda] = Z_{m}(p)$ as in (8.48). Instead, we now choose

$$\lambda(p) = Z_{m}(p) \lambda_{1}(p)$$

where $\lambda(p) = \det[p I - \Lambda]$ and $\lambda_{1}(p)$ is a Hurwitz polynomial of degree $(n - 1 - m)$. With this choice, the desired zeros of the reference model can be imposed.

Let us denote the transfer function of the feedforward part $(u/u_{1})$ of the controller by $\lambda(p)/(\lambda(p) + C(p))$, and that of the feedback part by $D(p)/\lambda(p)$, where the polynomial $C(p)$ contains the parameters in the vector $\theta_{1}$, and the polynomial $D(p)$ contains $\theta_{0}$ and the parameters in the vector $\theta_{2}$. Then, the closed-loop transfer function is easily found to be

$$W_{ry} = \frac{kk_{p}Z_{p}\lambda_{1}(p)Z_{m}(p)}{R_{p}(p)[\lambda(p) + C(p)] + k_{p}Z_{p}D(p)}$$

(8.58)
The question now is whether in this general case there exist choice of values for $k, \theta_o, \theta_1$ and $\theta_2$ such that the above transfer function becomes exactly the same as $W_m(p)$, or equivalently

$$R_p(\lambda(p) + C(p)) + k_p Z_p D(p) = \lambda_1 Z_p R_m(p) \quad (8.59)$$

The answer to this question can be obtained from the following lemma:

**Lemma 8.2:** Let $A(p)$ and $B(p)$ be polynomials of degree $n_1$ and $n_2$, respectively. If $A(p)$ and $B(p)$ are relatively prime, then there exist polynomials $M(p)$ and $N(p)$ such that

$$A(p)M(p) + B(p)N(p) = A^*(p) \quad (8.60)$$

where $A^*(p)$ is an arbitrary polynomial.

This lemma can be used straightforwardly to answer our question regarding (8.59). By regarding $R_p$ as $A(p)$ in the lemma, $k_p Z_p$ as $B(p)$ and $\lambda_1 Z_p R_m$ as $A^*(p)$, we conclude that there exist polynomials $(\lambda(p) + C(p))$ and $D(p)$ such that (8.59) is satisfied. This implies that a proper choice of the controller parameters

$$k = k^* \quad \theta_o = \theta_o^* \quad \theta_1 = \theta_1^* \quad \theta_2 = \theta_2^*$$

exists so that exact model-following is achieved.

**CHOICE OF ADAPTATION LAW**

When the plant parameters are unknown, we again use a control law of the form (8.52), i.e.,

$$u = \theta^T(t) \omega(t) \quad (8.61)$$

with the $2n$ controller parameters in $\theta(t)$ provided by the adaptation law. Using a similar reasoning as before, we can again obtain the output $y$ in the form of (8.53) and the tracking error in the form of (8.54), i.e.,

$$e(t) = W_m(p)[\phi^T \omega/k^*] \quad (8.62)$$

However, the choice of adaptation law given by (8.55) cannot be used, because now the reference model transfer function $W_m(p)$ is no longer SPR. A famous technique called *error augmentation* can be used to avoid the difficulty in finding an adaptation law for (8.62). The basic idea of the technique is to consider a so-called augmented error $\varepsilon(t)$ which correlates to the parameter error $\phi$ in a more desirable way than the tracking error $e(t)$.
Specifically, let us define an auxiliary error $\eta(t)$ by

$$
\eta(t) = \theta^T(t)W_m(p)[\omega] - W_m(p)[\theta^T(t)\omega(t)]
$$

as shown in Figure 8.17. It is useful to note two features about this error. First, $\eta(t)$ can be computed on-line, since the estimated parameter vector $\theta(t)$ and the signal vector $\omega(t)$ are both available. Secondly, this error is caused by the time-varying nature of the estimated parameters $\theta(t)$, in the sense that when $\theta(t)$ is replaced by the true (constant) parameter vector $\theta^*$, we have

$$
\theta^*^T W_m(p)[\omega] - W_m(p)[\theta^*^T \omega(t)] = 0
$$

This also implies that $\eta$ can be written

$$
\eta(t) = \phi^T W_m(\omega) - W_m(\phi^T \omega)
$$

Now let us define an augmented error $\varepsilon(t)$, by combining the tracking error $e(t)$ with the auxiliary error $\eta(t)$ as

$$
\varepsilon(t) = e(t) + \alpha(t) \eta(t)
$$

where $\alpha(t)$ is a time-varying parameter to be determined by adaptation. Note that $\alpha(t)$ is not a controller parameter, but only a parameter used in forming the new error $\varepsilon(t)$. For convenience, let us write $\alpha(t)$ in the form

$$
\alpha(t) = \frac{1}{k^*} + \phi_\alpha(t)
$$

where $\phi_\alpha = \alpha(t) - \frac{1}{k^*}$. Substituting (8.62) and (8.63) into (8.64), we obtain
This implies that the augmented error can be linearly parameterized by the parameter errors $\phi(t)$ and $\phi_\alpha$. Equation (8.65) thus represents a form commonly seen in system identification. A number of standard techniques to be discussed in section 8.7, such as the gradient method or the least-squares method, can be used to update the parameters for equations of this form. Using the gradient method with normalization, the controller parameters $\theta(t)$ and the parameter $\alpha(t)$ for forming the augmented error are updated by

$$\dot{\theta} = -\frac{\text{sgn}(k_p) \gamma \epsilon \omega}{1 + \omega^T \omega}$$  \hspace{1cm} (8.67a)$$

$$\dot{\alpha} = -\frac{\gamma \epsilon \eta}{1 + \omega^T \omega}$$  \hspace{1cm} (8.67b)$$

With the control law (8.61) and adaptation law (8.67), global convergence of the tracking error can be shown. The proof is mathematically involved and will be omitted here.

Finally, note that there exist other techniques to get around the difficulty associated with equation (8.62). In particular, it can be shown that an alternative technique is to generate a different augmented error, which is related to the parameter error $\theta$ through a properly selected SPR transfer function.

### 8.5 Adaptive Control of Nonlinear Systems

There exists relatively little general theory for the adaptive control of nonlinear systems. However, adaptive control has been successfully developed for some important classes of nonlinear control problems. Such problems usually satisfy the following conditions:

1. the nonlinear plant dynamics can be linearly parameterized
2. the full state is measurable
3. nonlinearities can be canceled stably (i.e., without unstable hidden modes or dynamics) by the control input if the parameters are known
In this section, we describe one such class of SISO systems to suggest how to design adaptive controllers for nonlinear systems (as an extension of the technique in section 8.3). In chapter 9, we shall study in detail the adaptive control of special classes of MIMO nonlinear physical systems.

**PROBLEM STATEMENT**

We consider \( n \)-th-order nonlinear systems in companion form

\[
y^{(n)} + \sum_{i=1}^{a} \alpha_i f_i(x, i) = bu
\]  

(8.68)

where \( x = [y \ y' \ ... \ y^{(n-1)}]^T \) is the state vector, the \( f_i \) are known nonlinear functions of the state and time, and the parameters \( \alpha_i \) and \( b \) are unknown constants. We assume that the state is measured, and that the sign of \( b \) is known. One example of such dynamics is

\[
m x' + c f_1(x) + k f_2(x) = u
\]  

(8.69)

which represents a mass-spring-damper system with nonlinear friction and nonlinear damping.

The objective of the adaptive control design to make the output asymptotically tracks a desired output \( y_d(t) \) despite the parameter uncertainty. To facilitate the adaptive controller derivation, let us rewrite equation (8.68) as

\[
h y^{(n)} + \sum_{i=1}^{a} a_i f_i(x, i) = u
\]  

(8.70)

by dividing both sides by the unknown constant \( b \), where \( h = 1/b \) and \( a_i = \alpha_i/b \).

**CHOICE OF CONTROL LAW**

Similarly to the sliding control approach of chapter 7, let us define a combined error

\[
s = e^{(n-1)} + \lambda_{n-2} e^{(n-2)} + ... + \lambda_0 e = \Delta(p) e
\]

where \( e \) is the output tracking error and \( \Delta(p) = p^{n-1} + \lambda_{n-2} p^{(n-2)} + ... + \lambda_0 \) is a stable (Hurwitz) polynomial in the Laplace variable \( p \). Note that \( s \) can be rewritten as

\[
s = y^{(n-1)} - y^{(n-1)}_r
\]

where \( y^{(n-1)}_r \) is defined as

\[
y^{(n-1)}_r = y_d^{(n-1)} - \lambda_{n-2} e^{(n-2)} - ... - \lambda_0 e
\]
Consider the control law

\[ u = h y_r^{(n)} - k_3 + \sum_{i=1}^{n} \alpha_i f_i(x, t) \] (8.71)

where \( k \) is a constant of the same sign as \( h \), and \( y_r^{(n)} \) is the derivative of \( y_r^{(n-1)} \), i.e.,

\[ y_r^{(n)} = y_d^{(n)} - \lambda_{a-2}e^{(n-1)} - ... - \lambda_{a} \dot{e} \]

Note that \( y_r^{(n)} \), the so-called "reference" value of \( y^{(n)} \), is obtained by modifying \( y_d^{(n)} \) according to the tracking errors.

If the parameters are all known, this choice leads to the tracking error dynamics

\[ h \ddot{s} + k_3 s = 0 \]

and therefore gives exponential convergence of \( s \), which, in turn, guarantees the convergence of \( e \).

**CHOICE OF ADAPTATION LAW**

For our adaptive control, the control law (8.71) is replaced by

\[ u = \hat{h} y_r^{(n)} - k_3 + \sum_{i=1}^{n} \hat{\alpha}_i f_i(x, t) \] (8.72)

where \( \hat{h} \) and the \( \hat{\alpha}_i \) have been replaced by their estimated values. The tracking error from this control law can be easily shown to be

\[ h \ddot{s} + k_3 s = \hat{h} y_r^{(n)} + \sum_{i=1}^{n} \hat{\alpha}_i f_i(x, t) \] (8.73)

This can be rewritten as

\[ s = \frac{1}{p + (k_3/h)} [\hat{h} y_r^{(n)} + \sum_{i=1}^{n} \hat{\alpha}_i f_i(x, t)] \] (8.74)

Since this represents an equation in the form of (8.15) with the transfer function obviously being SPR, Lemma 8.1 suggests us to choose the following adaptation law

\[ \dot{\hat{h}} = -\gamma \text{sgn}(h) s y_r^{(n)} \]

\[ \dot{\hat{\alpha}}_i = -\gamma \text{sgn}(h) s f_i \]

Specifically, using the Lyapunov function candidate
\[ V = 1k\dot{s}^2 + \gamma^{-1} \left( \bar{y}^2 + \sum_{i=1}^{n} \bar{a}_i^2 \right) \]

it is straightforward to verify that

\[ \dot{V} = -2k\dot{s}^2 \]

and therefore the global tracking convergence of the adaptive control system can be easily shown.

Note that the formulation used here is very similar to that in section 8.3. However, due to the use of the compact tracking error measure \( s \), the derivation and notation here is much simpler. Also, one can easily show that global tracking convergence is preserved if a different adaptation gain \( \gamma_i \) is used for each unknown parameter.

The sliding control ideas of chapter 7 can be used further to create controllers that can adapt to constant unknown parameters while being robust to unknown but bounded fast-varying coefficients or disturbances, as in systems of the form

\[ h y^{(n)} + \sum_{i=1}^{n} \left( a_i + \sigma_{i,v}(t) \right) f_i(x, t) = u \]

where the \( f_i \) are known nonlinear functions of the state and time, \( h \) and the \( a_i \) are unknown constants, and the time-varying quantities \( \sigma_{i,v}(t) \) are unknown but of known (possibly state-dependent or time-varying) bounds (Exercise 8.8).

### 8.6 Robustness of Adaptive Control Systems

The above tracking and parameter convergence analysis has provided us with considerable insight into the behavior of the adaptive control system. The analysis has been carried out assuming that no other uncertainties exist in the control system besides parametric uncertainties. However, in practice, many types of non-parametric uncertainties can be present. These include

- high-frequency unmodeled dynamics, such as actuator dynamics or structural vibrations
- low-frequency unmodeled dynamics, such as Coulomb friction and stiction
- measurement noise
- computation roundoff error and sampling delay
Since adaptive controllers are designed to control real physical systems and such non-parametric uncertainties are unavoidable, it is important to ask the following questions concerning the non-parametric uncertainties:

- what effects can they have on adaptive control systems?
- when are adaptive control systems sensitive to them?
- how can adaptive control systems be made insensitive to them?

While precise answers to such questions are difficult to obtain, because adaptive control systems are nonlinear systems, some qualitative answers can improve our understanding of adaptive control system behavior in practical applications. Let us now briefly discuss these topics.

Non-parametric uncertainties usually lead to performance degradation, i.e., the increase of model following error. Generally, small non-parametric uncertainties cause small tracking error, while larger ones cause larger tracking error. Such relations are universal in control systems and intuitively understandable. We can naturally expect the adaptive control system to become unstable when the non-parametric uncertainties become too large.

PARAMETER DRIFT

When the signal \( v \) is persistently exciting, both simulations and analysis indicate that the adaptive control systems have some robustness with respect to non-parametric uncertainties. However, when the signals are not persistently exciting, even small uncertainties may lead to severe problems for adaptive controllers. The following example illustrates this situation.

Example 8.7: Rohrs's Example

The sometimes destructive consequence of non-parametric uncertainties is clearly shown in the well-known example by Rohrs, which consists of an adaptive first-order control system containing unmodeled dynamics and measurement noise. In the adaptive control design, the plant is assumed to have the following nominal model

\[
H_p(p) = \frac{k_p}{p + a_p}
\]

The reference model has the following SPR function

\[
M(p) = \frac{k_m}{p + a_m} = \frac{3}{p + 3}
\]
The real plant, however, is assumed to have the transfer function relation

\[ y = \frac{229}{p + 1} \frac{229}{p^2 + 30p + 229} u \]

This means that the real plant is of third order while the nominal plant is of only first order. The unmodeled dynamics are thus seen to be \( 229/(p^2 + 30p + 229) \), which are high-frequency but lightly-damped poles at \((-15 + j)\) and \((-15 - j)\).

Besides the unmodeled dynamics, it is assumed that there is some measurement noise \( n(t) \) in the adaptive system. The whole adaptive control system is shown in Figure 8.18. The measurement noise is assumed to be \( n(t) = 0.5 \sin(16.1 t) \).

![Figure 8.18: Adaptive control with unmodeled dynamics and measurement noise](image)

Corresponding to the reference input \( r(t) = 2 \), the results of the adaptive control system are shown in Figure 8.19. It is seen that the output \( y(t) \) initially converges to the vicinity of \( y = 2 \), then operates with a small oscillatory error related to the measurement noise, and finally diverges to infinity.

In view of the global tracking convergence proven in the absence of non-parametric uncertainties and the small amount of non-parametric uncertainties present in the above example, the observed instability can seem quite surprising. However, one can gain some insight into what is going on in the adaptive control system by examining the parameter estimates in Figure 8.19. It is seen that the parameters drift slowly as time goes on, and suddenly diverge sharply. The simplest explanation of the parameter drift problem is that the constant reference input contains insufficient parameter information and the parameter adaptation mechanism has difficulty distinguishing the parameter information from noise. As a result, the parameters drift in a direction along which the tracking error remains small. Note that even though the
tracking error stays at the same level when the parameters drift, the poles of the closed-loop system continuously shift (since the parameters vary very slowly, the adaptive control system may be regarded as a linear time-invariant system with three poles). When the estimated parameters drift to the point where the closed-loop poles enter the right-half complex plane, the whole system becomes unstable. The above reasoning can be confirmed mathematically.

In general, the following points can be made about parameter drift. Parameter drift occurs when the signals are not persistently exciting; it is mainly caused by measurement noise; it does not affect tracking accuracy until instability occurs; it leads to sudden failure of the adaptive control system (by exciting unmodeled dynamics).

Parameter drift is a major problem associated with non-parametric uncertainties (noise and disturbance). But there are possibly other problems. For example, when the adaptation gain or the reference signal are very large, adaptation becomes fast and the estimated parameters may be quite oscillatory. If the oscillations get into the frequency range of unmodeled dynamics, the unmodeled dynamics may be excited and the parameter adaptation may be based on meaningless signals, possibly leading to instability of the control system. For parameter oscillation problems, techniques such as normalization of signals (divide \( v \) by \( 1 + v^T v \)) or the composite adaptation in section 8.8 can be quite useful.

DEAD-ZONE

Even though the possibility of small disturbances leading to instability is quite undesirable, it does not mean that adaptive control is impractical. A number of techniques for modifying the adaptation law are available to avoid the parameter drift problem. The simplest is called the "dead-zone" technique. Because of its simplicity and effectiveness, it is most frequently used.
The dead-zone technique is based on the observation that small tracking errors contain mostly noise and disturbance, therefore, one should shut the adaptation mechanism off for small tracking errors. Specifically, we should replace an adaptation law

$$\dot{a} = -\gamma v e$$

(8.75)

by

$$\dot{a} = \begin{cases} -\gamma v e & |e| > \Delta \\ 0 & |e| < \Delta \end{cases}$$

(8.76)

where $\Delta$ is the size the dead-zone. As the following example shows, such a simple modification can greatly reduce the effects of the disturbances.

**Example 8.8: Use of Dead-Zone**

Consider again the adaptive control system of Example 8.7, but modify the adaptation law by incorporating a dead-zone of $\Delta = 0.7$. The results are shown in Figure 8.20. It is seen that tracking error stays around the ideal response of $y = 2$, with an oscillation due to the measurement noise. The parameters now do not have any indication of drifting. It is interesting to point out that the oscillation appears very fast because the time-scale in the figure is large and the noise itself is of quite high frequency.

![Figure 8.20](image_url)

**Figure 8.20**: Adaptive control with dead-zone

A number of other techniques also exist to relieve the problem of parameter drift. One involves the so-called $\sigma$-modification, which approximates the original integrator in the adaptation law by a lowpass filter. Another is the "regressor replacement" technique. By "regressor", we mean the vector $v(t)$ in (8.75). Note that $v(t)$ is usually computed based on the plant measurement $y$ and thus affected by measurement noise $u(t)$. Since the adaptation law (8.75) involves the multiplication of
by \( e(t) \), the update rate is related to the square of the measurement noise and causes parameter drift. For example, in the presence of measurement noise \( n(t) \), (8.28b) can be written

\[
\hat{a}_y = -\text{sgn}(b) \gamma (y_1 + n - y_m) (y_1 + n) \\
= -\text{sgn}(b) \gamma [(y_1 - y_m) y_1 + n (2y_1 - y_m) + n^2]
\]

where \( y_1 \) is the true plant output. It is noted that the first term truly contains parameter information, the second term tends to average out, and the third term \(-\text{sgn}(b) \gamma n^2\) is the reason for the drifting of \( \hat{a}_y \) in Figure 8.19 (\( \hat{a}_r \) drifts accordingly so that the tracking error remains small). As a result of this observation, one can relieve parameter drift by replacing \( y \) in (8.28b) by \( y_m \) which is independent of \( n \). It is desirable to start this replacement after the tracking error has converged well.

### 8.7 * On-Line Parameter Estimation*

When there is parameter uncertainty in a dynamic system (linear or nonlinear), one way to reduce it is to use parameter estimation, i.e., inferring the values of the parameters from the measurements of input and output signals of the system. Parameter estimation can be done either on-line or off-line. Off-line estimation may be preferable if the parameters are constant and there is sufficient time for estimation before control. However, for parameters which vary (even though slowly) during operation, on-line parameter estimation is necessary to keep track of the parameter values. Since problems in the adaptive control context usually involve slowly time-varying parameters, on-line estimation methods are thus more relevant.

In this section, we study a few basic methods of on-line estimation. Unlike most discussions of parameter estimation, we use a continuous-time formulation rather than a discrete-time formulation. This is motivated by the fact that nonlinear physical systems are continuous in nature and are hard to meaningfully discreteize. Furthermore, digital control systems may be treated as continuous-time systems in analysis and design if high sampling rates are used. The availability of cheap computation generally allows high sampling rates and thus continuous-time models to be used.

Note that, although the main purpose of the on-line estimators may be to provide parameter estimates for self-tuning control, they can also be used for other purposes, such as load monitoring or failure detection.
8.7.1 Linear Parametrization Model

The essence of parameter estimation is to extract parameter information from available data concerning the system. Therefore, we need an estimation model to relate the available data to the unknown parameters, similarly to the familiar experimental data fitting scenario, where we need to hypothesize the form of a curve before finding specific coefficients describing it, based on the data. This estimation model may or may not be the same as the model used for the control purpose. A quite general model for parameter estimation applications is in the linear parametrization form

$$y(t) = W(t) a$$

where the $n$-dimensional vector $y$ contains the "outputs" of the system, the $m$-dimensional vector $a$ contains unknown parameters to be estimated, and the $n \times m$ matrix $W(t)$ is a signal matrix. Note that both $y$ and $W$ are required to be known from the measurements of the system signals, and thus the only unknown quantities in (8.77) are the parameters in $a$. This means that (8.77) is simply a linear equation in terms of the unknown $a$. For every time instant $t$, there is such an equation. So if we are given the continuous measurements of $y(t)$ and $W(t)$ throughout a time interval, we have an infinite number of equations in the form of (8.77). If we are given the values of $y(t)$ and $W(t)$ at $k$ sampling instants, we have $k$ sets of such equations instead. The objective of parameter estimation is to simply solve these redundant equations for the $m$ unknown parameters. Clearly, in order to be able to estimate $m$ parameters, we need at least a total of $m$ equations. However, in order to estimate the parameters well in the presence of inevitable noise and modeling error, more data points are preferable.

In off-line estimation, one collects the data of $y$ and $W$ for a period of time, and solves the equations once and for all. In on-line estimation, one solves the equation recursively, implying that the estimated value of $\hat{a}$ is updated once a new set of data $y$ and $W$ is available.

How well and how fast the parameters $a$ are estimated depends on two aspects, namely, the estimation method used and the information content (persistent excitation) of the data $y$ and $W$. Our primary objective in this section is to examine the properties of some standard estimation methods. The generation of informative data is a complex issue discussed extensively in the system identification literature. While we shall not study this issue in detail, the relation between the signal properties and estimation results will be discussed.

Model (8.77), although simple, is actually quite general. Any linear system can
be rewritten in this form after filtering both sides of the system dynamics equation through an exponentially stable filter of proper order, as seen in the following example.

Example 8.9: Filtering Linear Dynamics

Let us first consider the first-order dynamics

\[ \dot{y} = -a_1 y + b_1 u \]  \hspace{1cm} (8.78)

Assume that \( a_1 \) and \( b_1 \) in the model are unknown, and that only the output \( y \) and the input \( u \) are available. The above model cannot be directly used for estimation, because the derivative of \( y \) appears in the above equation (note that numerically differentiating \( y \) is usually undesirable because of noise considerations). To eliminate \( y \) in the above equation, let us filter (multiply) both sides of the equation by \( 1/(p + \lambda_f) \) (where \( p \) is the Laplace operator and \( \lambda_f \) is a known positive constant). Rearranging, this leads to the form

\[ y(t) = y_f (\lambda_f - a_1) + u_f b_1 \]  \hspace{1cm} (8.79)

where

\[ y_f = \frac{y}{p + \lambda_f} \quad u_f = \frac{u}{p + \lambda_f} \]

with the subscript \( f \) denoting filtered quantities. Note that, as a result of the filtering operation, the only unknown quantities in (8.79) are the parameters \( (\lambda_f - a_1) \) and \( b_1 \).

Note that the above filtering introduces a d.c. gain of \( 1/\lambda_f \), i.e., the magnitudes of \( y_f \) and \( u_f \) are smaller than those of \( y \) and \( u \) by a factor of \( \lambda_f \) at low frequencies. Since smaller signals may lead to slower estimation, one may multiply both sides of (8.79) by a constant number, e.g., \( \lambda_f \).

Generally, for a linear single-input single-output system, its dynamics can be described by

\[ A(p)y = B(p)u \]  \hspace{1cm} (8.80)

with

\[ A(p) = a_0 + a_1 p + \ldots + a_{n-1} p^{n-1} + p^n \]

\[ B(p) = b_0 + b_1 p + \ldots + b_{n-1} p^{n-1} \]

Let us divide both sides of (8.80) by a known monic polynomial of order \( n \), leading to

\[ y = \frac{A_n(p) - A(p)}{A_0(p)} y + \frac{B(p)}{A_0(p)} u \]  \hspace{1cm} (8.81)

where
\[ A_0 = \alpha_0 + \alpha_1 p + \ldots + \alpha_{n-1} p^{n-1} + p^n \]

has known coefficients. In view of the fact that
\[ A_0(p) - A(p) = (\alpha_0 - a_0) + (\alpha_1 - a_1) p + \ldots + (\alpha_{n-1} - a_{n-1}) p^{n-1} \]
we can write (8.81) in the basic form
\[ y = \theta^T w(t) \quad (8.82) \]
with \( \theta \) containing \( 2n \) unknown parameters, and \( w \) containing the filtered versions of the input and output, defined by
\[
\theta = [(\alpha_0 - a_0) \quad (\alpha_1 - a_1) \ldots (\alpha_{n-1} - a_{n-1}) \quad b_0 \ldots b_{n-1}]^T
\]
\[
w = \left[ \frac{y}{A_0} \quad \frac{p y}{A_0} \ldots \frac{p^{n-1} y}{A_0} \quad \frac{u}{A_0} \ldots \frac{p^{n-1} u}{A_0} \right]^T
\]

Note that \( w \) can be computed on-line based on the available values of \( y \) and \( u \).

The dynamics of many nonlinear systems can also be put into the form (8.77). A simple example is the nonlinear mass-spring-damper system (8.69), for which the input force is obviously linear in terms of mass, friction coefficient, and spring coefficient. For some more complicated nonlinear dynamics, proper filtering and parameter transformation may be needed to put the dynamics into the form of (8.77), as we now show.

**Example 8.10: Linear parametrization of robot dynamics**

Consider the nonlinear dynamics (6.9) of the two-link robot of Example 6.2. Clearly, the joint torque vector \( \tau \) is nonlinear in terms of joint positions and velocities. It is also nonlinear in terms of the physical parameters \( l_1, l_2 \), and so on. However, it can be put into the form of (8.77) by a proper reparametrization and a filtering operation.

Consider the reparametrization first. Let us define
\[
a_1 = m_2
\]
\[
a_2 = m_2 l_2
\]
\[
a_3 = l_1 + m_1 l_1^2
\]
\[
a_4 = l_2 + m_2 l_2^2
\]
Then, one can show that each term on the left-hand side of (6.9) is linear in terms of the equivalent inertia parameters \( \alpha = [a_1 \quad a_2 \quad a_3 \quad a_4]^T \). Specifically
Note that $l_1$ and $l_2$ are kinematic parameters, which are assumed to be known (they can be similarly treated if not known). The above expressions indicate that the inertia torque terms are linear in terms of $a$. It is easy to show that the other terms are also linear in $a$. Thus, we can write

$$T = Y(q, \dot{q}, \ddot{q}) a$$  \hspace{1cm} (8.83)

with the matrix $Y$ expressed (nonlinearly) as a function of only $q$, $\dot{q}$ and $\ddot{q}$; and $a$ being a $m \times 1$ vector of equivalent parameters. This linear parametrization property actually applies to any mechanical system, including multiple-link robots.

Relation (8.83) cannot be directly used for parameter estimation, because of the presence of the unmeasurable joint acceleration $\ddot{q}$. To avoid the joint acceleration in this relation, we can use the above filtering technique. Specifically, let $w(t)$ be the impulse response of a stable, proper filter (for example, for the first-order filter $\lambda/(\rho + \lambda)$, the impulse response is $e^{-\lambda t}$). Then, convolving both sides of (6.9) by $w$ yields

$$\int_0^t w(t-r) \tau(r) dr = \int_0^t w(t-r) [H(\ddot{q}) + C\dot{q} + G] dr$$  \hspace{1cm} (8.84)

Using partial integration, the first term on the right-hand side of (8.84) can be rewritten as

$$\int_0^t w(t-r) [H(\ddot{q})] dr = w(t) H(\ddot{q})_0 - \int_0^t \frac{d}{dr} [w(t-r) H(\ddot{q})] dr$$

$$= w(0) H(q) \ddot{q} - w(0) H(q(0)) \ddot{q}(0) - \int_0^t [w(t-r) H \dddot{q} - w(t-r) H \dot{q}] dr$$

This means that equation (8.84) can be rewritten as

$$y(t) = W(q, \dot{q}) a$$  \hspace{1cm} (8.85)

where $y$ is the filtered torque and $W$ is the filtered version of $Y$. Thus, the matrix $W$ can be computed from available measurements of $q$ and $\dot{q}$. The filtered torque $y$ can also be computed (assuming no actuator dynamics) because the torque signals issued by the computer are known. \[ \square \]

It is obvious at this point that the "output" $y$ in model (8.77) does not have to be the same as the output in a control problem. In the above robotic example, the "output" $y$ is actually the filtered version of the physical input to the robot. From a parameter estimation point of view, what we want is just a linear relation between known data and the unknown parameters. The following example further demonstrates...
Example 8.11: An alternative estimation model for robots

Consider again the above 2-link robot. Using the principle of energy conversation, one sees that the rate of change of mechanical energy is equal to the power input from the joint motors, i.e.,

\[ \tau^T \dot{q} = \frac{dE}{dt} \tag{8.86} \]

where \( E(q, \dot{q}) \) is the total mechanical energy of the robot. Since it is easy to show that the mechanical energy can be linearly parameterized, i.e.,

\[ E = v(q, \dot{q}) a \]

one can write the energy relation (8.86) as

\[ \tau^T \dot{q} = \frac{dv(q, \dot{q})}{dt} a \]

with \( v \) computable from the measurement of \( q \) and \( \dot{q} \). To eliminate the joint acceleration \( \ddot{q} \) in this relation, we can again use filtering by a first-order filter \( 1/(p + \lambda_f) \). This leads to

\[ \frac{1}{p + \lambda_f} [\tau^T q] = [v - \frac{\lambda_f v}{p + \lambda_f}] a \]

This is in the form of (8.77), with

\[ y = \frac{1}{p + \lambda_f} [\tau^T q] \]

\[ w = v - \frac{\lambda_f v}{p + \lambda_f} \]

In the above example, note that the control model and the estimation model are drastically different. Actually, while the control model (6.9) has two equations and two outputs, the estimation model (8.86) has only one equation and one output (actually, the scalar model (8.86) is applicable to robots with any number of links because the energy relation is always scalar). With the energy relation (8.86), the computation of \( w \) is greatly simplified compared with that of \( W \) in (8.85), because there is no need to compute the complex centripetal and Coriolis forces (which may contain hundreds of terms for multiple-DOF robots).
8.7.2 Prediction-Error-Based Estimation Methods

Before studying the various methods for parameter estimation, let us first discuss the concept of prediction error. Assume that the parameter vector in (8.77) is unknown, and is estimated to be $\hat{a}(t)$ at time $t$. One can predict the value of the output $y(t)$ based on the parameter estimate and the model (8.77),

$$\hat{y}(t) = W(t)\hat{a}(t)$$  \hspace{1cm} (8.87)

where $\hat{y}$ is called the predicted output at time $t$. The difference between the predicted output and the measured output $y$ is called the prediction error, denoted by $e_1$, i.e.,

$$e_1(t) = \hat{y}(t) - y(t)$$  \hspace{1cm} (8.88)

The on-line estimation methods to be discussed in this section are all based on this error, i.e., the parameter estimation law is driven by $e_1$. The resulting estimators belong to the so-called prediction-error based estimators, a major class of on-line parameter estimators. The prediction error is related to the parameter estimation error, as can be seen from

$$e_1 = W\hat{a} - Wa = W\hat{a}$$  \hspace{1cm} (8.89)

where $\hat{a} = \hat{a} - a$ is the parameter estimation error.

In the following, we shall discuss the motivation, formulation, and properties of the following methods:

- Gradient estimation
- Standard least-squares estimation
- Least-squares with exponential forgetting
- A particular method of variable exponential forgetting

Note that in the convergence analysis of these estimators we shall assume that the true parameters are constant, so that insights concerning the estimator's behavior can be obtained. However, in the back of our mind, we will always be aware of the task of handling time-varying parameters.

8.7.3 The Gradient Estimator

The simplest on-line estimator is the gradient estimator. Let us discuss its formulation, convergence properties, and robustness properties.
FORMULATION AND CONVERGENCE

The basic idea in gradient estimation is that the parameters should be updated so that the prediction error is reduced. This idea is implemented by updating the parameters in the converse direction of the gradient of the squared prediction error with respect to the parameters, i.e.,

\[ \dot{\hat{a}} = -p_o \frac{\partial[e^T e]}{\partial \hat{a}} \]

where \( p_o \) is a positive number called the estimator gain. In view of (8.88) and (8.87), this can be written as

\[ \dot{\hat{a}} = -p_o W^T e_1 \]  \hspace{1cm} (8.90)

To see the properties of this estimator, we use (8.90) and (8.89) to obtain

\[ \dot{\hat{\hat{a}}} = -p_o W^T W \hat{\hat{a}} \]

Using the Lyapunov function candidate

\[ V = \hat{a}^T \hat{a} \]

its derivative is easily found to be

\[ \dot{V} = -2p_o \hat{a}^T W^T W \hat{\hat{a}} \leq 0 \]

This implies that the gradient estimator is always stable. By noting that \( V \) is actually the squared parameter error, we see that the magnitude of the parameter error is always decreasing.

However, the convergence of the estimated parameters to the true parameters depends on the excitation of the signals. To gain some insights on that point, let us consider the estimation of a single parameter.

Example 8.12: Gradient estimation of a single parameter

Consider the estimation of one parameter from the model

\[ y = wa \]

(with the mass estimation in Example 8.2 being such a case). The gradient estimation law is

\[ \dot{\hat{a}} = -p_o w e_1 \]

This implies that
\[ \dot{\tilde{a}} = -p_0w^2\tilde{a} \]

which can be solved as

\[ \tilde{a}(t) = \tilde{a}(0) \exp \left[ -\int_0^t p_0 w^2(r) \, dr \right] \]

This implies that the parameter error will converge to zero if the signal \( w \) is such that

\[ \lim_{t \to \infty} \int_0^t w^2(r) \, dr = \infty \]

Note that \( \tilde{a} \) will exponentially converge to zero if \( w \) is persistently exciting, i.e., if there exist positive constants \( T \) and \( \alpha_1 \) such that for all \( t \geq 0 \),

\[ \int_t^{t+T} w^2 \, dr \geq \alpha_1 \]

In fact, the convergence rate is easily found to be \( p_0\alpha_1/T \).

Clearly, in this case, a constant non-zero \( w \) can guarantee the exponential convergence of \( \tilde{a} \). However, if the signal \( w \) decays too fast (e.g., \( w = e^{-t} \)), one easily shows that the parameter error does not converge to zero.

The above convergence result concerning the one-parameter case can be extended to the estimation of multiple parameters. Specifically, if the matrix \( W \) is persistently exciting, i.e., there exist positive constants \( \alpha_1 \) and \( T \) such that \( \forall T \geq 0 \)

\[ \int_t^{t+T} W^T W \, dr \geq \alpha_1 I \]  

then the parameter error \( \tilde{a} \) will converge exponentially, as shown in [Anderson, 1977; Morgan and Narendra, 1977] through a fairly involved procedure. Such a condition for parameter convergence is easily understandable, analogously to the case of model-reference adaptive control. In the case of linear systems, as described by (8.80), it is easy to verify that \( m \) sinusoids in the input signal \( u \) can guarantee the estimation of up to \( 2m \) parameters. In the case of nonlinear systems, the relation between the number of sinusoids and the number of parameters which can be estimated is not so clear. It is possible to estimate more than \( 2m \) parameters with an input \( u \) containing \( m \) sinusoids, as explained in section 8.2.

A slightly more general version of the gradient estimator is obtained by replacing the scalar gain by a positive definite matrix gain \( P_0 \).
\[
\dot{\hat{a}} = -P_0 \frac{\partial (e_1^T e_1)}{\partial \hat{a}}
\]

The global stability of the algorithm can be shown using the Lyapunov function

\[
V = \hat{a}^T P_0^{-1} \hat{a}
\]

The convergence properties are very similar.

**EFFECTS OF ESTIMATION GAIN**

The choice of estimation gain \(p_o\) has a fundamental influence on the convergence behavior of the estimator. For the single-parameter estimation case, one easily sees that a larger \(p_o\) implies faster parameter convergence. In fact, the convergence rate is linearly related to the estimation gain \(p_o\). For the multiple-parameter case, however, the relation between the magnitude of \(p_o\) and the convergence rate of the estimated parameters is not as simple. Generally speaking, in a small range, increasing estimation gain leads to faster parameter convergence. But beyond some point, further increasing the estimation gain leads to more oscillatory and slower convergence, as can be demonstrated with the estimation of the four load parameters in the robot example 8.10. This phenomenon is caused by the gradient nature of the estimation, similarly to what happens in the gradient search method in optimization: within a small range, increase in step size in the gradient direction leads to faster convergence; but beyond some point, larger size leads to more oscillatory and possibly slower convergence.

Besides the effect on convergence speed, the choice of \(p_o\) also has implications on the ability of the estimator to track time-varying parameters and withstand disturbances, as will be discussed soon.

**ROBUSTNESS PROPERTIES**

The above analysis and simulations have been based on the assumed absence of parameter variation and non-parametric uncertainties. In order for an estimator to have practical value, however, it must have some robustness, i.e., maintain reasonably good parameter estimation in the presence of parameter variation, measurement noise, disturbances, etc.

The quality of the parameter estimates in a gradient estimator depends on a number of factors, mainly,

- the level of persistent excitation of the signal \(W\)
• the rate of parameter variation and the level of non-parametric uncertainties

• the magnitude of the estimator gain $p_o$

The level of persistent excitation in $W$ is decided by the control task or experiment design. Persistent excitation is essential for the robustness of the estimator. If the signals in the original design are not persistently exciting, parameters will not converge even in the absence of non-parametric uncertainties. In the presence of non-parametric uncertainties, the estimator may possibly become unstable, i.e., the parameters may diverge. One may have to add some perturbation signals to the control input to obtain good parameter estimation. The specific details in the data generation may be complicated, but the bottom line is that one should produce as much persist excitation as allowed by the involved constraints.

How fast the true parameters vary and how large the non-parametric uncertainties also affect the quality of the parameter estimates. Obviously, if the true parameters vary faster, it is harder for the parameter estimator to estimate accurately. If a lot of noise and unmodeled disturbances and dynamics are present, the estimates also become poor. To see this, let us consider the mass estimation problem, but now in the presence of disturbance (unmodeled Coulomb friction or measurement noise) and time-varying mass. The mass dynamics becomes

$$\tau(t) = m(t) w(t) + d(t)$$

where $d(t)$ is the disturbance and $w$ is the acceleration. The prediction error in this case is

$$e_1 = \hat{\tau} - \tau = \hat{m}w - d$$

By substituting this into (8.90), the parameter error is easily shown to satisfy

$$\frac{d\hat{a}}{dt} = -p_o w^2\hat{a} - \dot{\hat{a}} + p_o w d$$

(8.92)

This can be interpreted as a time-varying filter with "output" $\hat{a}$ and "input" $(-\dot{\hat{a}} + p_o w d)$. Clearly, larger parameter variation rate $\dot{\hat{a}}$ and larger disturbance $d(t)$ leads to larger parameter error $\hat{a}$. If parameter $a$ varies too fast or the non-parametric uncertainties have too large values, one should consider using more accurate models, i.e., modeling the dynamics of the parameter variation and also the disturbance.

The magnitude of the estimator gain also has a considerable influence on the robustness of the estimator. If $p_o$ is chosen to be large, the "bandwidth" of the filter $p_o w^2$ becomes large (so that higher-frequency noise can pass) and the input component $p_o w d$ to the filter becomes larger. This means that parameter estimation
error due to disturbance will become larger. The increase of estimation gain has the opposite effect on the estimator ability to estimate time-varying parameters. Consider the case of \( w \) being a constant for example. The steady-state error of the estimated parameter in (8.92) is \( \frac{m}{p_o w^2} \), which shows that the parameter error is decreased by the increase of the estimation gain.

The following simple simulation illustrates the behavior of the gradient estimator in the presence of measurement noise.

**Example 8.13: Gradient estimation of a constant mass**

Consider the problem of mass estimation for the dynamics

\[
u = m w(t) + d(t)
\]

with \( w(t) = \sin(t) \), and \( d(t) \) is interpreted as either disturbance or measurement noise. The true mass is assumed to be \( m = 2 \). When the disturbance \( d(t) = 0 \), the estimation results are shown in the left plot in Figure 8.21. It is seen that larger gain corresponds to faster convergence, as expected. When the disturbance is \( d(t) = 0.5 \sin(20t) \), the estimation results are shown on the right plot in Figure 8.21. It is seen that larger estimation gain leads to larger estimation error.

![Figure 8.21: gradient method, left: without noise, right: with noise](image)

The following simple simulation illustrates the behavior of the gradient estimator in the presence of both parameter variation and measurement noise.

**Example 8.14: Gradient estimation of a time-varying mass**

Now suppose that the true parameter is slowly time-varying, with \( m(t) = 1 + 0.5 \sin(0.5t) \). Let us take \( p_o = 1 \). The estimation results in the absence of disturbance is shown in the left plot in Figure 8.22. In the presence of the disturbance of the previous example, the parameter estimation result is shown in the right plot. It is seen that the gradient method works quite well in the presence of parameter variation and disturbance.
8.7.4 The Standard Least-Squares Estimator

We all have some experience with least squares estimation (data fitting). In this subsection, we shall formalize the technique and carefully study its properties.

FORMULATION

In the standard least-squares method, the estimate of the parameters is generated by minimizing the total prediction error

\[ J = \int_{0}^{T} \| y(r) - W(r) \hat{a}(r) \|^2 \, dr \]  

with respect to \( \hat{a}(t) \). Since this implies the fitting of all past data, this estimate potentially has the advantage of averaging out the effects of measurement noise. The estimated parameter \( \hat{a} \) satisfies

\[ [\int_{0}^{T} W^T W \, dr] \hat{a}(t) = \int_{0}^{T} W^T y \, dr \]  

Define

\[ P(t) = [\int_{0}^{T} W^T(r) W(r) \, dr]^{-1} \]  

To achieve computational efficiency, it is desirable to compute \( P \) recursively, instead of evaluating the integral at every time instant. This amounts to replacing the above equation by the differential equation

\[ \frac{d}{dt} [P^{-1}(t)] = W^T(t) W(t) \]
Differentiating (8.94) and using (8.95) and (8.96), we find that the parameter update satisfies

$$\dot{\hat{a}} = -P(t)W^T e_1$$  \hspace{1cm} (8.97)

with $P(t)$ being called the estimator gain matrix, similarly to the case of gradient estimation. In the implementation of the estimator, it is desirable to update the gain $P$ directly, rather than using (8.96) and then inverting the matrix $P^{-1}$. By using the identity:

$$\frac{d}{dt}[PP^{-1}] = \dot{P}P^{-1} + P\frac{d}{dt}[P^{-1}] = 0$$

we obtain

$$\dot{P} = -PW^TW$$  \hspace{1cm} (8.98)

In using (8.97) and (8.98) for on-line estimation, we have to provide an initial parameter value and an initial gain value. But there is a difficulty with this initialization, because (8.97) and (8.98) imply that $P$ should be infinity, while $\hat{a}$ is initially undefined. To avoid this problem, we shall provide finite values to initialize $P$ and $\hat{a}$. Clearly, one should use the best guess to initialize the $\hat{a}$. The choice of the initial gain $P(0)$ should be chosen as high as allowed by the noise sensitivity. $P(0)$ can be chosen to be diagonal, for simplicity.

It is useful to remark that the above least-squares estimator can be interpreted in a Kalman filter framework, with $\hat{a}$ being the state to be estimated and $P$ being the estimation covariance matrix. Based on this perspective, the initial gain $P(0)$ should be chosen to represent the covariance of the initial parameter estimates $\hat{a}(0)$.

**PARAMETER CONVERGENCE**

The convergence property of the estimator can be best understood by solving the differential equations (8.96) and (8.97), assuming the absence of noise and parameter variation. From (8.96), (8.97) and (8.98), one easily shows that

$$P^{-1}(t) = P^{-1}(0) + \int_0^t W^T(r)W(r)dr$$  \hspace{1cm} (8.99)

$$\frac{d}{dt}[P^{-1}(t)\tilde{a}(t)] = 0$$

Thus,

$$\tilde{a}(t) = P(t)P^{-1}(0) \tilde{a}(0)$$  \hspace{1cm} (8.100)
If $W$ is such that
\[ \lambda_{\text{min}} \{ \int_{0}^{t} W^T W \, dr \} \to \infty \quad \text{as} \quad t \to \infty \] (8.101)
where $\lambda_{\text{min}}[\cdot]$ denotes the smallest eigenvalue of its argument, then the gain matrix converges to zero, and the estimated parameters asymptotically (but usually not exponentially) converge to the true parameters. Note that the "infinite-integral" condition (8.101) is a weaker condition than the persistent excitation (8.91). Indeed, for any positive integer $k$,
\[ \int_{\delta}^{k\delta+\delta} W^T W \, dr = \sum_{k=0}^{k} \int_{\delta}^{k\delta+\delta} W^T W \, dr \geq k \alpha_1 \mathbf{I} \] (8.102)

Thus, if $W$ is persistently exciting, (8.102) is satisfied, $P \to 0$ and $\bar{a} \to 0$.

The effects of the initial gain and initial parameter estimates can be easily seen from (8.100) and (8.99). Obviously, a small initial parameter error $\bar{a}(0)$ results in a small parameter error all the time. A large initial gain $P(0)$ also leads to a small parameter error. This is particularly clear if we choose $P(0) = p_0 \mathbf{I}$, which leads to
\[ \bar{a}(t) = [\mathbf{I} + p_0 \int_{0}^{t} W^T(r) W(r) \, dr]^{-1} \bar{a}(0) \]

**ROBUSTNESS**

Roughly speaking, the least-squares method has good robustness with respect to noise and disturbance, but poor ability in tracking time-varying parameters. The reason for the good noise-rejection property is easy to understand: noise, particularly high-frequency noise, is averaged out. The estimator's inability in tracking time-varying parameters can also be understood intuitively, from two points of views. From a mathematical point of view, $P(t)$ converges to zero when $W$ is persistently exciting, i.e., the parameter update is essentially shut off after some time, and the changing parameters cannot be estimated any more. From an information point of view, the least-square estimate attempts to fit all the data up to the current time, while, in reality, the old data is generated by old parameters.

The following example illustrates the behavior of the least-squares estimator in the presence of measurement noise.

**Example 8.15: Least-squares estimation of one parameter**

Consider again the estimation of the mass system, now with a least-square method. The parameter estimation results in the absence of disturbance, are shown in the left plot of Figure 8.23, with the two curves corresponding to the initial gain $p_o = 2$ and $p_o = 10$, respectively. It is seen that the parameter converges faster with larger initial gain. Another feature is that least
Sect. 8.7

*On-Line Parameter Estimation* 373

square converge fast initially, but slowly afterward.

The parameter estimation results in the presence of disturbance are presented in the right plot of Figure 8.23. The same initial gain values are used. It is seen that the estimated parameters are much smoother than those from the gradient method.

![Figure 8.23: Least-squares method; left: without noise; right: with noise](image)

The inability of the least-squares method to estimate time-varying parameters is demonstrated in the following example.

**Example 8.16: Least-squares estimation of time-varying parameter**

Now the true parameter \( m = 1 \) is replaced by \( m(t) = 1 + 0.5 \sin(0.5 t) \). The initial gain value is chosen to be \( P(0) = 1 \). The true and estimated parameters in the absence of disturbance are shown in the left plot of Figure 8.24. Those in the presence of disturbance are shown in the right plot. It is seen that the estimated parameters cannot follow the true parameter variation, regardless of the disturbance.

![Figure 8.24: Least-squares time-varying gradient](image)
8.7.5 Least-Squares With Exponential Forgetting

Exponential forgetting of data is a very useful technique in dealing with time-varying parameters. Its intuitive motivation is that past data are generated by past parameters and thus should be discounted when being used for the estimation of the current parameters. In this subsection, we describe the general formulation of least square method with time-varying forgetting factor. In the next subsection, a desirable form of forgetting factor variation will be discussed.

If exponential forgetting of data is incorporated into least-square estimation, one minimizes

$$J = \int_0^t \exp[-\int_s^t \lambda(r) dr] \| y(s) - W(s) \hat{a}(t) \|^2 ds$$

instead of (8.93), where $\lambda(t) \geq 0$ is the time-varying forgetting factor. Note that the exponential term in the integral represents the weighting for the data. One easily shows that the parameter update law is still of the same form,

$$\dot{\hat{a}} = -P(t) W^T e_1$$  \hspace{1cm} (8.103)

but that the gain update law is now

$$\frac{d}{dt} [P^{-1}] = -\dot{\lambda}(t) P^{-1} + W^T(t) W(t)$$  \hspace{1cm} (8.104)

In implementation, it is more efficient to use the following form of the gain update

$$\frac{d}{dt} [P] = \lambda(t) P - P W^T(t) W(t) P$$  \hspace{1cm} (8.105)

To understand the convergence properties of the estimator, let us again solve for the gain and parameter errors explicitly. The estimator gain can be explicitly solved from (8.104) to be

$$P^{-1}(t) = P^{-1}(0) \exp[-\int_0^t \lambda(r) dr] + \int_0^t \exp[-\int_v^t \lambda(y) dv] W^T(r) W(r) dr$$  \hspace{1cm} (8.106)

To solve the explicit form of the parameter error, let us note that one can easily obtain

$$\frac{d}{dt} [P^{-1} \tilde{a}] = -\dot{\lambda} P^{-1} \tilde{a}$$  \hspace{1cm} (8.107)

Therefore,
Sect. 8.7  
* On-Line Parameter Estimation  375

\[
\tilde{a}(t) = \exp[-\int_0^t \alpha(r) dr] P(t) P^{-1}(0) \tilde{a}(0) \tag{8.108}
\]

That is,

\[
\tilde{a}(t) = [P^{-1}(0) + \int_0^t \exp[\int_0^v \alpha(v) dv] W^T(r) W(r) dr]^{-1} P^{-1}(0) \tilde{a}(0)
\]

Comparing this and (8.100), and noting that

\[
\exp[\int_0^t \alpha(v) dv] \geq 1
\]

one sees that exponential forgetting always improves parameter convergence over standard least-squares. It also shows that the "infinite integral" condition (8.101) for standard least-squares still guarantees the asymptotic convergence of the estimated parameters. This also implies that persistent excitation of the signal W can guarantee the convergence of the estimated parameters.

It is easy to see that exponential forgetting leads to exponential convergence of the estimated parameters, provided that \( \lambda(t) \) is chosen to be larger than or equal to a positive constant and that signals are p.e. Specifically, assume that \( \lambda(t) \geq \lambda_o \), with \( \lambda_o \) being a positive constant. Then, we have from (8.104)

\[
\frac{d}{dt} [P^{-1}] = -\lambda_o P^{-1} + W^T W + (\lambda(t) - \lambda_o) P^{-1}
\]

\[
P^{-1}(t) = P^{-1}(0) e^{-\lambda_o t} + \int_0^t e^{-\lambda_o (t-r)} [W^T W + (\lambda(t) - \lambda_o) P^{-1}] dr
\]

This guarantees from (8.92) that \( P^{-1}(t) \geq e^{-\lambda_o \delta \alpha_1} I \) and, accordingly, that

\[
P(t) \leq \frac{e^{\lambda_o \delta}}{\alpha_1} I
\]

for \( t \geq \delta \). This and (8.108) show the exponential convergence of \( \tilde{a}(t) \) to zero with a rate of at least \( \lambda_o \).

It is interesting to remark that, if the forgetting factor is constant, then the exponential convergence rate of the estimated parameters is the same as the forgetting factor. However, a constant forgetting factor may lead to diminishing magnitude in certain directions of \( P^{-1} \) (and accordingly, unbounded magnitude in certain directions of \( P \)) in the absence of p.e., due to the exponential decaying components in (8.106). Unboundedness (or even large magnitude) of the gain matrix is undesirable, since it implies that the disturbance and noise in the prediction error may, through the update
Adaptive Control

Chap. 8

law (8.103), lead to violent oscillations of the estimated parameters.

The following one-parameter example demonstrates the desirable and undesirable properties for exponential forgetting with a constant forgetting factor.

Example 8.17: Estimating one parameter with a constant forgetting factor

Consider the estimation of parameter $a$ from the equation $y = wa$ using the least-squares method with a constant forgetting factor. The cost to be minimized is

$$ J = \int_{0}^{t} e^{-\lambda_0 (t-r)} [y(r) - w(r) \hat{a}(t)]^2 dr $$

The parameter update law is

$$ \dot{\hat{a}}(t) = -P(t)w(t)e(t) $$

and the gain update law is

$$ \frac{d}{dt}[P^{-1}] = -\lambda_0 P^{-1} + w w $$

The gain update is easily solved to be

$$ P^{-1}(t) = P^{-1}(0)e^{-\lambda_0 t} + \int_{0}^{t} e^{-\lambda_0 (t-r)} w^2 dr $$

and the parameter error is

$$ \tilde{a}(t) = e^{-\lambda_0 t} P(t) P^{-1}(0) \tilde{a}(0) $$

If the signal $w$ is persistently exciting, $P(t)$ will be upper bounded and $\tilde{a}$ will converge to zero exponentially with a rate $\lambda_0$. If $w$ is not persistently exciting (for example, for $w = e^{-t}$, $P^{-1}(t)$ may converge to zero and thus $P(t)$ to infinity).

Thus, we see that caution has to be exercised in choosing the forgetting factor. A zero forgetting factor leads to vanishing gain (standard least-squares and thus inability of tracking time-varying parameters) in the presence of persistent excitation, while a constant positive factor leads to exploding gain in the absence of persistent excitation. Since an estimator may encounter signals with different levels of persistent excitation, an automatic tuning method for the forgetting factor is necessary.

8.7.6 Bounded-Gain Forgetting

To keep the benefits of data forgetting (parameter tracking ability) while avoiding the possibility of gain unboundedness, it is desirable to tune the forgetting factor variation.
so that data forgetting is activated when \( w \) is persistently exciting, and suspended when \( w \) is not. We now discuss such a tuning technique and the resulting estimator convergence and robustness properties.

**A FORGETTING FACTOR TUNING TECHNIQUE**

Since the magnitude of the gain matrix \( P \) is an indicator of the excitation level of \( W \), it is reasonable to correlate the forgetting factor variation with \( \|P(t)\| \). A specific technique for achieving this purpose is to choose

\[
\lambda(t) = \lambda_0 \left( 1 - \frac{||P||}{k_0} \right)
\]

with \( \lambda_0 \) and \( k_0 \) being positive constants representing the maximum forgetting rate and prespecified bound for gain matrix magnitude, respectively. The forgetting factor in (8.109) implies forgetting the data with a factor \( \lambda_0 \) if the norm of \( P \) is small (indicating strong p.e.), reducing the forgetting speed if the norm of \( P \) becomes larger and suspends forgetting if the norm reaches the specified upper bound. Since a larger value of \( \lambda_0 \) means faster forgetting (which implies both stronger ability in following parameter variations, but also more oscillations in the estimated parameters due to shorter-time "averaging" of noisy data points), the choice of \( \lambda_0 \) represents a tradeoff between the speed of parameter tracking and the level of estimated parameter oscillation. The gain bound \( k_0 \) affects the speed of parameter update and also the effects of disturbance in the prediction error, thus involving a similar tradeoff. To be consistent with our gain-bounding intention, we choose \( ||P(0)|| \leq k_0 \) (hence \( P(0) \leq k_0 I \)). We shall refer to the least-squares estimator with the forgetting factor (8.109) as the bounded-gain-forgetting (BGF) estimator, because the norm of the gain matrix can be shown to be upper bounded by the pre-specified constant \( k_0 \) regardless of persistent excitation.

**CONVERGENCE PROPERTIES**

We now show that the form (8.109) of forgetting factor variation guarantees that the resulting gain matrix \( P(t) \) is upper bounded regardless of the persistent excitation of \( W \), unlike the case of constant forgetting factor. With the forgetting factor form (8.109), the gain update equation (8.104) can be expressed as

\[
\frac{d}{dt} [P^{-1}] = -\lambda_0 P^{-1} + (\lambda_0/k_0) \|P\| P^{-1} + W^T W
\]

This leads to
\[ P^{-1}(t) = P^{-1}(0) e^{-\lambda_o t} + \int_0^t e^{-\lambda_o (t-r)} \left[ \frac{\lambda_o}{k_o} \|P\|P^{-1} + W^T W \right] dr \]

\[ \geq (P^{-1}(0) - k_o^{-1} I) e^{-\lambda_o t} + \left( \frac{1}{k_o} \right) I + \int_0^t e^{-\lambda_o (t-r)} W^T W dr \]

(8.111)

where we used the inequality \( \|P(t)\|P^{-1}(t) \geq 1 \), obtained from the fact that

\[ \|P\|P^{-1} - I = P^{-1/2} [\|P\|I - P] P^{-1/2} \geq 0 \]

Note that \( \|P(0)\| \leq k_o \) guarantees the positive-definiteness of \( (P^{-1}(0) - k_o^{-1} I) \), and therefore for all \( t \geq 0 \),

\[ P^{-1}(t) \geq \frac{1}{k_o} I \]

so that \( P(t) \leq k_o I \). Note that this implies, from (8.109), that

\[ \lambda(t) \geq 0 \]

Thus, we have shown the boundedness of \( P \) and the non-negative nature of \( \lambda(t) \).

If \( W(t) \) is p.e. as defined by (8.91), we can further show that \( \lambda(t) \geq \lambda_1 > 0 \), and, thus, the estimated parameters are exponentially convergent. To show this, note that, from (8.111) and (8.91),

\[ P^{-1}(t) \geq \left[ \frac{1}{k_o} + e^{-\lambda_o \delta} \alpha_1 \right] I \]

\[ P(t) \leq \frac{k_o}{1 + k_o \alpha_1 e^{-\lambda_o \delta}} I \]

(8.112)

This, in turn, leads to the uniform lower boundedness of the forgetting factor by a positive constant,

\[ \lambda(t) = \frac{\lambda_o}{k_o} \left( k_o - \|P\| \right) \geq \frac{\lambda_o k_o \alpha_1 e^{-\lambda_o \delta}}{1 + k_o \alpha_1 e^{-\lambda_o \delta}} = \lambda_1 \]

which in turn implies the exponential convergence of the estimated parameters. Note that, if \( W \) is strongly p.e., i.e., \( \alpha_1 \) is very large, \( \lambda(t) \approx \lambda_o \).

Under p.e., one can also show that \( P(t) \) is uniformly lower bounded by a constant p.d. matrix, a property which is desirable for estimating time-varying parameters. Indeed, from (8.106) and (8.113),
\[ P^{-1}(t) \leq P^{-1}(0) + \int_{0}^{t} \exp \left[ -\lambda(t - \tau) \right] W^{T}W \, d\tau \]

The second term on the right-hand side can be regarded as the output of the stable filter

\[ \dot{M} + \lambda(t)M = W^{T}W \quad (8.114) \]

\( M \) is bounded if \( W \) is bounded. Thus, from (8.114) and (8.112), if \( W \) is p.e. and upper bounded, \( P \) will be upper bounded and lower bounded uniformly, i.e.,

\[ k_2 I \leq P(t) \leq k_1 I \]

where \( 0 < k_2 < k_1 < k_0 \).

The properties of the BGF estimator are summarized below:

**Theorem 8.1** In the bounded-gain-forgetting estimator, the parameter errors and the gain matrix are always upper bounded; if \( W \) is persistently exciting, then the estimated parameters converge exponentially and \( P(t) \) is upper and lower bounded uniformly by positive definite matrices.

The advantage of the BGF method over the gradient method is that the estimated parameters are smooth. This implies, given the allowable level of estimated parameter oscillation, that much larger gain bound can be used in BGF method. As a result, faster parameter convergence can be achieved.

**AN ALTERNATIVE TECHNIQUE OF DATA FORGETTING**

It is interesting to note that the bounded-gain forgetting is not the only way to keep data forgetting while avoiding gain explosion. In view of the fact that the problem with the non-zero forgetting factor is the possibility of \( P^{-1} \) diminishing, we may, as an alternative to variable-forgetting approach, use the following simple gain update modification

\[ \frac{d}{dt} P^{-1} = -\lambda(t)(P^{-1} - K_o^{-1}) + W^{T}W \quad (8.115) \]

where \( K_o \) is a symmetric p.d. matrix specifying the upper bound of the gain matrix \( P \), and \( \lambda(t) \) is a constant or time-varying positive forgetting factor independent of \( P \). \( \lambda(t) \) is chosen to satisfy \( \lambda_0 \leq \lambda(t) \leq \lambda_1 \) with \( \lambda_0 \) and \( \lambda_1 \) denoting two positive constants. The parameter update law is still (8.103). One can easily show that this gain update leads to similar properties as the bounded-gain forgetting.
ROBUSTNESS PROPERTIES

Let us now look at the behavior of the estimator in the presence of parameter variations and disturbances. Consider again the single-parameter estimation problem of Example 8.12. The gain update is

$$\frac{d}{dt} P^{-1}(t) = -\lambda(t) P^{-1} + w^2$$

One notes that the presence of noise and parameter variation does not affect the gain value.

To see how the parameter error behaves, let us note that

$$\dot{P}^{-1} = (\lambda(t) P^{-1} a + w d)$$

This describes a first-order filter of "bandwidth" $\lambda(t)$ with $(-P^{-1} a + w d)$ as input and $P^{-1} a$ as output. We can easily draw some conclusions about the estimator behavior based on this relation, assuming that the signal $w$ is persistently exciting, the disturbance $d$ is bounded, and the rate of parameter variation is bounded. Since $P^{-1}$ has been shown to be upper bounded, the input to the filter is bounded. Furthermore, since the forgetting factor $\lambda(t)$ has been shown to be lower bounded, equation (8.116) represents an exponentially stable filter and its output $P^{-1} a$ is guaranteed to be bounded. The upper boundedness of $P$ further indicates the boundedness of $\lambda_0$. Note that the bound of $a$ will depend on the magnitude of the disturbance and the parameter variation rate $\dot{a}$. It also depends on the persistent excitation level of the signal $w(t)$ (which affects $\lambda(t)$).

If the signal $w$ is not persistently exciting, then the estimator can no longer guarantee the boundedness of parameter errors in the presence of parameter variation and disturbances. We can only say that $P(t)$ (being independent to the additive disturbances and parameter variations) is bounded and the parameter errors cannot diverge too fast.

Example 8.18: BGF estimation of the single parameter

The BGF estimator is used to estimate the time-varying parameter. In the absence of measurement noise, the estimated and true parameters are shown in the left plot of Figure 8.25. In the presence of measurement noise, they are shown in the right-hand side plot of Figure 8.25. □
USE OF DEAD-ZONE

Similarly to the MRAC case, one should use a small dead-zone in estimating parameters by the previous methods. It is particularly important when the signal $W(t)$ is not persistently exciting, leading to the avoidance of parameter drift. The dead-zone should now be placed on the prediction error. The size of the dead-zone should be chosen to be larger than the range of noise and disturbance in the prediction error. The motivation is still that small error signals are dominated by noise and cannot be used for reliable estimation.

8.7.7 Concluding Remarks and Implementation Issues

We have discussed a number of estimators in this section. Persistent excitation is essential for good estimation. Most estimators do not have both fast convergence and good robustness. The gradient estimator is simple but has slow convergence. The standard least-squares estimator is robust to noise but cannot estimate time-varying parameters. Least-squares estimation with exponential forgetting factor has the ability to track time-varying parameters but there is a possibility of gain windup in the absence of persistent excitation.

A particular technique of gain tuning, called bounded-gain-forgetting, is developed to maintain the benefits of data forgetting while avoiding gain-windup. The resulting estimator has both fast convergence and good noise sensitivity. The computation and analysis of this estimator is reasonably simple.

In order to have good estimation performance, many implementation issues have to be considered carefully, including

- choice of the bandwidth of the filter for generating (8.77)
• choice of initial parameter and initial gain matrix
• choice of forgetting rate and gain bound
• choice of excitation signals

Tradeoffs and judgement have to be used in making the above choices. The filter bandwidth should be chosen to be larger than the plant bandwidth, so that the system signals are able to pass through. But it should be smaller than frequency range (usually high frequency) of the noise. Of course, the initial parameter estimates should be chosen to be as accurate as possible. The initial gain matrix should be chosen to obtain the proper convergence speed and noise robustness. The forgetting factor should be chosen to be large enough so that parameter variation can be tracked sufficiently accurately. However, it cannot be chosen too large lest the gain matrix is too large or too oscillatory. The gain bound is chosen based on the knowledge of the noise magnitude and the allowable level of estimated parameter oscillation, with larger noise leading to smaller gain bound. The excitation signals should contain sufficient spectrum lines to allow parameter convergence, and their frequencies should be within the bandwidth of the plant so as to be able to excite the plant. Note that, although unknown parameters can be time-varying, the speed of the variation should be much smaller than the plant bandwidth, otherwise the parameter dynamics should be modeled.

8.8 Composite Adaptation

In the MRAC controllers developed in sections 8.2-8.4, the adaptation laws extract information about the parameters from the tracking errors $e$. However, the tracking error is not the only source of parameter information. The prediction error $e_1$ also contains parameter information, as reflected in the parameter estimation schemes of section 8.7. This section examines whether the different sources of parameter information can be combined for parameter adaptation, and whether such a combined use of information sources can indeed improve the performance of the adaptive controller.

These questions are answered positively by a new adaptation method, called composite adaptation, which drives the parameter adaptation using both tracking error and prediction error. As we shall see, such an adaptation scheme not only maintains the global stability of the adaptive control system, but also leads to fast parameter convergence and smaller tracking errors. Indeed, the fundamental feature of composite adaptation is its ability to achieve quick adaptation.
In the following, we first describe the concept of composite adaptation using the simple mass control example, and then extend the results to more general linear and nonlinear systems.

**COMPOSITE ADAPTATION FOR A MASS**

Although the benefits of composite adaptation are most obvious in adaptive control problems involving multiple parameters, its mechanism is most easily explained first on the simple example of controlling a mass. Consider the system \( m\ddot{x} = u \). A MRAC controller has been designed in Example 8.1 for this system, with the control law being

\[
u = \hat{m}v
\]

(8.117)

and the adaptation law being

\[
\dot{\hat{m}} = -\gamma v s
\]

(8.118)

If, instead, a prediction-error based estimator is used to estimate the parameter \( m \), then the parameter update law is

\[
\dot{\hat{m}} = -\gamma w e_1
\]

(8.119)

Note that while the adaptation law (8.118) is driven by the tracking error \( s \), the estimation law (8.119) is driven by \( e_1 \).

In the so-called composite adaptive control, the control is the same as (8.117), but the adaptation law is now a "combination" of the earlier adaptation law and the prediction-error based estimation law

\[
\dot{\hat{m}} = -P[v s + w e_1]
\]

Note that this adaptation law now is driven by both \( s \) and \( e_1 \). Also note that the gain \( P(t) \) is provided by any of the gain update laws in section 8.7.

To show the stability of the resulting adaptive control system, let us still use the Lyapunov function candidate.
This is essentially the same as the $V$ in (8.7) except for the replacement of $\gamma$ by $P(t)$. 

If $P(t)$ is chosen to be a constant, the derivative of $V$ is easily found to be

$$\dot{V} = -\lambda ms^2 - \omega^2 \bar{m}^2$$

Note that the expression of $\dot{V}$ now contains a desirable additional quadratic (only semi-negative) term in $\bar{m}$. Using a similar reasoning as that in Example 8.1, one can now show that both $s \to 0$ and $e \to 0$ as $t \to \infty$.

If $P$ is time-varying and generated as in BGF estimator, i.e., using equation (8.104), we obtain

$$\dot{V} = -\lambda ms^2 - \frac{1}{2} \omega^2 \bar{m}^2 - \frac{1}{2} \lambda(t) P^{-1} \bar{m}^2$$

This can also be shown to lead to $s \to 0$ and $e \to 0$ as $t \to \infty$. Furthermore, if $w$ is persistently exciting, one can easily show that both $s$ and $\bar{m}$ are exponentially convergent to zero, i.e., the adaptive controller has exponential convergence.

**COMPOSITE ADAPTIVE CONTROL OF FIRST-ORDER SYSTEMS**

The advantages of composite adaptation are most clearly seen in systems with more than one unknown parameters. In such cases, the composite adaptation law allows high adaptation gain to be used without getting the oscillatory behavior and slow convergence observed for the tracking-error-based adaptation laws (see section 8.2). Let us use the first-order system in section 8.2 to illustrate the performance features of composite adaptive controllers.

**Tracking-Error Based Adaptive Control**

Using the MRAC controller in section 8.2, the control law is of the form

$$u = v^T a$$

(8.120)

The parameter adaptation law is

$$\dot{a} = -\text{sgn}(k_p) \gamma v e$$

(8.121)

with

$$a = [a_x \ a_y]^T \quad v = [r \ y]^T$$
Sect. 8.8

Composite Adaptation

Prediction-Error Based Estimation

The estimation methods of section 8.7 can also be used to estimate \( a_r \) and \( a_y \). But before doing this we have to parameterize the plant in terms of these parameters. By adding \( a_m y \) to both sides of (8.20), we obtain

\[
\dot{y} + a_m y = -(a_p - a_m) y + b_p u
\]

This leads to

\[
u = \frac{1}{b_p} (p + a_m) y + \frac{a_p - a_m}{b_p} y
\]

Since the ideal controller parameters and the plant parameters are related by (8.24), we obtain

\[
u = a_r \frac{(p + a_m) y}{b_m} + a_y y
\]

Multiplying both sides by \( 1/(p + a_m) \), we then obtain

\[
\frac{u}{p + a_m} = a_r \frac{y}{b_m} + a_y \frac{y}{p + a_m}
\]

This can be compactly expressed in the linear parametrization form

\[
u_1 = wa
\]

with

\[
u_1 = \frac{u}{p + a_m} \quad a = \begin{bmatrix} a_r & a_y \end{bmatrix}^T \quad w = \begin{bmatrix} y & y \end{bmatrix} \quad b_m
\]

Therefore, we can straightforwardly use the estimation algorithms of section 8.7 to obtain the following estimation laws

\[
\dot{a} = -P \dot{w} e_1 \tag{8.122}
\]

where \( e_1 \) is the prediction error on \( u_1 \).

Composite Adaptive Control

The composite adaptation laws can be easily constructed, from (8.121) and (8.122), as

\[
\dot{a} = -P \{ \text{sgn}(b_p) \dot{v} e + \alpha(t) \dot{w} e_1 \} \quad \tag{8.123}
\]
The controller is the same as in (8.120).

To analyze the stability and convergence of the composite adaptive controller, one can still use the Lyapunov function in (8.29). If \( P \) is chosen to be constant, one has

\[
\dot{V} = -a_m e^2 - e_1^2
\]

One can easily show that \( e \rightarrow 0 \) and \( e_1 \rightarrow 0 \) as \( t \rightarrow \infty \).

If \( P \) is updated by the least-squares type algorithms, one can obtain similar results. For example, if \( P \) is updated by (8.104), we can use the Lyapunov function candidate

\[
V = \frac{1}{2} [ e^2 + \tilde{a}^T P^{-1} \tilde{a} ]
\]

whose derivative is

\[
\dot{V} = -a_m e^2 - \frac{1}{2} e_1^2 - \frac{1}{2} \lambda(t) \tilde{a}^T P^{-1} \tilde{a}
\]

From this, we can infer asymptotic and exponential convergence results, similarly to the previous mass-control case.

Simulation Results

Simulations are used to illustrate the behavior of the composite adaptive controller.

Example 8.19: Composite adaptive control of first-order systems

The same first-order system as in section 8.2 is used for composite adaptive control. Everything is as in section 8.2, except that now the prediction error term is incorporated into the adaptation law. For the case of \( r(t) = 4 \), the tracking and estimation results are shown in Figure 8.27. The results for \( r(t) = 4 \sin 3t \) are shown in Figure 8.28. It is seen the parameter and tracking results are both smooth. One can further increase the adaptation gain to obtain smaller tracking errors without incurring much oscillation in the estimated parameters.

The advantage of the composite adaptive controller essentially comes from the smoothness of the results. This has significant implications on the adaptive control performance. Due to the possibility of using high adaptation gain, we can obtain smaller tracking error, faster parameter convergence without exciting high-frequency unmodeled dynamics. In fact, simulations show that composite adaptive controllers perform much better than standard adaptive controllers when unmodeled dynamics are present.
Interpretation of Composite Adaptation

To gain an intuitive interpretation of the composite adaptation law, let us for simplicity choose $P(t) = \gamma I$. Using (8.123) and (8.89), the composite adaptation law can be expressed as

$$\dot{\hat{a}} + \gamma w^T w \hat{a} = -\gamma v^T e$$  \hspace{1cm} (8.125)

Without the prediction term, we obtain the original MRAC adaptation law (8.28)

$$\dot{\hat{a}} = -\gamma v^T e$$  \hspace{1cm} (8.126)

Comparing (8.125) and (8.126), one notes that (8.126) represents an integrator while (8.125) a time-varying low-pass filter. Both tend to attenuate the high-frequency components in $e$, and thus parameters in both cases tend to be smoother than the tracking errors. However, an integrator also attenuates low frequency components, as
can be seen from its Bode plot. The stable filter in (8.125), on the other hand, has much less attenuation on the low frequency components in $e$ (for example, if $w$ is a constant, (8.125) is a stable filter with a bandwidth of $\gamma w^{-2}$). Therefore, the parameter search in the composite adaptation goes along an "average" or "filtered" direction as specified by the low frequency components in the tracking error $e$. This explains the smoothness of the parameter estimates in composite adaptive control.

As shown in this section, the principle of composite adaptation is to combine a tracking-error based adaptation law and a prediction-error based parameter estimation law. This idea can be straightforwardly extended to adaptive control in general. Note that one must parameterize both the plant model and the controller using a common set of parameters in order to be able to combine the two types of errors for adaptation. In chapter 9, we shall discuss the application of this idea to the adaptive control of robot manipulators, an important class of multi-input multi-output nonlinear systems.

8.9 Summary

Adaptive control is an appealing approach for controlling uncertain dynamic systems. In principle, the systems can be uncertain in terms of its dynamic structure or its parameters. So far, however, adaptive control can only deal with parameter-uncertain systems. Furthermore, existing adaptation methods generally require linear parametrization of the control law or the system dynamics.

Systematic theories exist on how to design adaptive controllers for general linear systems. If the full state is available, adaptive control design and implementation is quite simple. If only output feedback is available, however, adaptive control design is much more involved because of the need to introduce dynamics into the controller. Some classes of nonlinear systems can also be adaptively controlled.

In MRAC systems, the adaptation law extracts parameter information from the tracking errors. In self-tuning controllers, the parameter estimator extracts information from prediction errors. In a new technique called composite adaptive control, the adaptation law extracts parameter information from both sources. This new adaptation method leads to faster adaptation without incurring significant oscillation in the estimated parameters, thus yielding improved performance.
8.10 Notes and References

Analyses and discussions on the material of this chapter can be found, e.g., in the recent books [Narendra and Annaswamy, 1989], which provides a systematic treatment of model-reference adaptive control theory for linear systems, and [Aström and Wittenmark, 1989], which includes extensive studies of self-tuning algorithms as well as discussions of adaptive control implementation issues. The basic formulation of a Lyapunov formalism for model-reference adaptive control is due to [Parks, 1966]. Detailed discussions of discrete versions and self-tuning control can be found in [Goodwin and Sin, 1984]. Passivity interpretations of model reference adaptive controllers are discussed in [Landau, 1979]. Detailed studies of robustness can be found, e.g., in [Ioannou and Kokotovic, 1983; Anderson, et al., 1986; Sastry and Bodson, 1989] and references therein.

The theoretical framework in section 8.4 is based on [Narendra and Annaswamy, 1989], from which Lemmas 8.1 and 8.2 and Figures 8.14 and 8.17 are adapted. Example 8.7 and Figure 8.18 are based on [Rohrs, et al., 1985].

The development of section 8.5 is adapted from [Slotine and Coetsee, 1986], which also details “robust” combinations with sliding control terms. Extensions and applications to specific classes of nonlinear MIMO systems (robot manipulators) are developed in [Slotine and Li, 1986, 1987], and will be further detailed in chapter 9. [Taylor, et al., 1988, 1989] study extensions to general MIMO systems under certain matching conditions, and most interestingly a singular perturbation analysis of the effects of unmodeled dynamics. [Kanellakopoulos, et al., 1989] develop adaptive nonlinear controllers under some less stringent, "extended" matching conditions. [Sastry and Isidori, 1989] consider output tracking under the two assumptions of uniformly exponentially stable zero-dynamics and of global Lipschitzness of the regressor. [Bastin and Campion, 1989; Pomet and Praly, 1989] study indirect approaches to adaptive control of nonlinear systems.


Our discussion of section 8.8 is based on [Slotine and Li, 1987d, 1989] (see also chapter 9). A closely related approach is presented in [Narendra and Duarte, 1988, 1989] using separate controller and regressor parametrizations.

8.11 Exercises

8.1 Draw the block diagram for the adaptive mass control system in Example 8.1. Discuss the intuitive reasonableness of the adaptation law.
8.2 Simulate the adaptive control system for the nonlinear first order plant in section 8.2, but with the nonlinearity being $f(y) = \sin y$.

8.3 Simulate the adaptive control system for the second order plant

$$y = \frac{p + b_p}{p^2 + a_p1p + a_p2}$$

with $a_p1 = 0.1$, $a_p2 = -4,b_p = 2$.

8.4 Consider a second-order linear system with a constant unknown disturbance $d$. Compare the structure of a P.I.D. controller with that of a P.D. controller with adaptive compensation of $d$.

8.5 Carry out the detailed proof of tracking error convergence for general linear systems with relative degree 1 (section 8.4.1).

8.6 Simulate the adaptive control system for the second order plant

$$y = \frac{b_p}{p^2 + a_p1p + a_p2}$$

with $a_p1 = 0.1$, $a_p2 = -4,b_p = 2$ for two cases. In the first case, the output $y$ and its derivative $\dot{y}$ are assumed to be available. In the second case, only $y$ is assumed to be available.

8.7 For the system

$$\begin{align*}
\dot{x}_1 &= \sin x_2 + \sqrt{t+1} \cdot x_2 \\
\dot{x}_2 &= \alpha_1 x_1^4 \cos x_2 + \alpha_2 \dot{x}_1 x_2 \sin x_2 + n
\end{align*}$$

design an adaptive controller to track an arbitrary desired trajectory $x_d(t)$. Assume that the state $[x_1 \ x_2]^T$ is measured, that $x_d(t), \dot{x}_d(t), \ddot{x}_d(t)$ are all known, and that $\alpha_1$ and $\alpha_2$ are unknown constants.

Write the full expression of the controller, as a function of the measured state $[x_1 \ x_2]^T$. Check your design in simple simulations. (Hint: First feedback linearize the system by differentiating the first equation.)

8.8 Consider systems of the form

$$h \ y^{(n)} + \sum_{i=1}^{n} \{ a_i + a_i(t) \} f_i(x, t) = u$$

where the state vector $x = \{ y \ \dot{y} \ \ldots \ y^{(n-1)} \}^T$ is measured, the $f_i$ are known nonlinear functions of the state and time, $h$ and the $a_i$ are unknown constants (or very slowly varying "average" coefficients), with $h > 0$, and the time-varying quantities $a_i(t)$ are unknown but of known bounds.
(possibly state-dependent or time-varying)

\[ |a_p(t)| \leq A_t(t) \]

Show that the ideas of chapter 7 and section 8.5 can be combined to yield a robust adaptive controller. To do so, essentially replace \( s \) by \( s_A = s - \Phi \text{sat}(s/\Phi) \) in the Lyapunov function of section 8.5, i.e., use

\[
V = hs_A^2 + \left[ \gamma_s^{-1} \tilde{n}_1^2 + \sum_{i=1}^{n} \gamma_i^{-1} \tilde{a}_i^2 \right]
\]

For simplicity, you may first choose \( \Phi \) to be constant.

Show that, assuming that the noise level in the system is small, the boundary layer concept leads to a natural choice of adaptation dead-zone (Hint: Choose the sliding control terms and the boundary layer thickness \( \Phi \) as if the constant parameters \( a_i \) were known, and note that \( s_A = 0 \) in the boundary layer).

Illustrate your design in simple simulations. (Adapted from [Slotine and Coetsee, 1986]).

8.9 For the nonlinear adaptive controller of section 8.5, show that if a priori bounds are known on certain parameters, tracking convergence is preserved by temporarily stopping adaptation on a given parameter estimate if the adaptation law would otherwise drive the estimate outside the known bounds.

8.10 Discuss intuitively why parameter estimation can be easier on unstable systems than on stable systems.

8.11 Derive a composite adaptive controller for the nonlinear first-order plant in section 8.2. Simulate its behavior using different gains.