

Chapter 5

Gain Scheduling and Adaptation

Abstract This chapter introduces gain-scheduling and linear-parameter-varying techniques to address plant variability across the flight envelope. This chapter also introduces the concept of adaptive control and presents a basic model-reference adaptive control design. Matlab code and simulations using the CMAPSS nonlinear engine model are included.

This chapter is an overview of two control concepts that address plant variability across the flight envelope: scheduling and adaptation. These control methods are fundamentally different from robust approaches in that controller parameters are not fixed, but undergo significant changes during system operation. Gain scheduling as used in standard GTE control is described in Sect. 5.2. Next, in Sect. 5.3, a more systematic linear parameter-varying (LPV) approach to gain scheduling is presented. In Sect. 5.4, the basic concept of adaptive control is introduced, followed by a simple version of a well-established adaptive control methodology, namely Model Reference Adaptive Control (MRAC). Application of scheduling and adaptation concepts to fan speed control is illustrated throughout the chapter with CMAPSS-1 simulation examples.

5.1 Robustness, Scheduling, Adaptation

The techniques presented in Chap. 4 attempt to find a fixed controller that tolerates a range of plant parameter variations. Tolerance is understood first as the ability to maintain closed-loop stability upon uncertain plant variations. When plant matrices vary in a polytope, closed-loop stability of a given compensator can be checked using the definition of quadratic stability and associated LMI feasibility computations. Moreover, such fixed compensators can be synthesized using $\mathcal{H}_2/\mathcal{H}_\infty$ methods. Stability is necessary, but cannot be regarded as the sole criterion to define a successful GTE control implementation. Controllers must also maintain consistent transient response and tracking accuracy across the flight envelope. As demonstrated in Chap. 4, fixed controllers synthesized through $\mathcal{H}_2/\mathcal{H}_\infty$ methods offer a degree

of performance robustness, as measured by the minimum achievable norm. Still, significant tracking offsets and transient performance loss occur when attempting to use a single controller for all flight conditions.

Gain scheduling techniques address plant variations by introducing matching controller gain variations, attempting to obtain uniform transient responses across a wide range of operating conditions. The appropriate gain variations are determined during offline design and fixed for subsequent operation in the form of scheduling tables. In contrast, adaptive techniques induce control parameter variations online, as part of the control system implementation. Adaptive parameters can assume values outside the linear space generated by interpolation of scheduling table gains, thus improving the control system's *adaptability* to changes in the plant. Adaptive controllers introduce nonlinear dynamics in the closed-loop system. Therefore, their proper design requires an understanding of the theoretical bases governing their behavior. In this chapter, focus is limited to a simple form of adaptive control that is adequate for SISO plant models whose number of poles exceeds the number of zeroes by one. The transfer function from fuel flow to fan speed fits this case, and the adaptive technique is readily applied with good results.

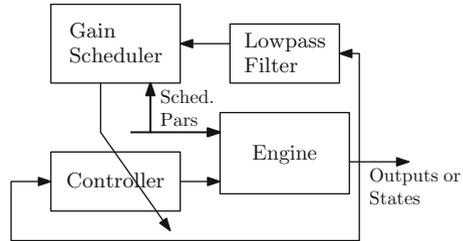
5.1.1 Input Scheduling

Standard SISO GTE control systems use fuel flow as the only actuator used in a feedback loop ultimately aimed at controlling engine thrust. Other actuators, mainly VSV and VBV, are still actively governed during flight. Although there are no “controllers” in the usual sense that provide commands to these actuators, they are still commanded through a function of real-time measurements. The term “scheduling” is used in the GTE industry to denote this form of actuator command. In CMAPSS-40k, for instance, these actuators are commanded through functions of the corrected fan speed of (1.23), corrected core speed, and Mach number. Since the corrected speeds are functions of state variables, this form of command effectively adds decoupled SISO loops to the engine. Furthermore, it corresponds to proportional control, where the proportional gain is scheduled by Mach number and Θ , the nondimensional total temperature at LPC outlet. VSV and VBV command functions are limited to the allowable ranges of these actuators. Multivariable designs such as those presented in Chap. 4 forego of these scheduling functions and allow VSV and VBV to be commanded from a multi-output feedback controller.

5.2 Standard Gain-Scheduled GTE Control

Classical GTE gain scheduling centers on the idea of a fixed-structure (say, PI) compensator for fan speed or EPR control, whose gains are adjusted according to certain scheduling variables. These variables are chosen to reflect changes in

Fig. 5.1 Gain-scheduling control system



environmental conditions such as altitude, Mach number, and sea-level temperature, which act as parameters of the linearized engine models. State variables such as fan speed may also be used as scheduling variables. Note that when a state variable – or function thereof – is used as scheduling variable, a nonlinear feedback control loop is effectively introduced. This additional loop has the potential to destabilize the main control loop if not carefully designed. As shown in Fig. 5.1, it is customary to insert a low-pass filter between the state variable used for scheduling and the input of the scheduling table. This is to ensure that the changes to the control gains brought about by state variables are slow in comparison with the bandwidth of the main loop. The scheduler used in *ad-hoc* designs is nothing more than a look-up table giving control gains as a function of scheduling variables.

Although there is no universal rule for the selection of scheduling variables, it is reasonable to identify a set of physical parameters which dictate the numerical values of system matrices obtained through linearization. Recalling (2.1) and (2.2), it is clear that *all* parameters of functions f_1 and f_2 , together with steady input values and a fixed set of health parameters, are needed to define an equilibrium point. Ideally, a set of scheduling variables would be comprised by all parameters. In practice, however, a few parameters can be identified that have the largest influence in the numerical values of the resulting linearized matrices. In the GTE, physical consideration and experience show that thermodynamic conditions at engine inlet, i.e., inlet static pressure and Mach number have the largest influence. Fan speed, an engine state, is frequently used as an additional scheduling variable to account for the intrinsic nonlinearity of functions f_1 and f_2 . Inlet pressure may be used directly, or equivalently, altitude may be used. An additional important consideration for the selection of scheduling variables is their availability as real-time measurements.

The scheduling tables are built by selecting a controller structure and repeating a controller design for various combinations of scheduling variables. The resulting gains are then included in look-up tables. Linear interpolation is used to find controller gains from real-time measurements of the scheduling variables.

In CMAPSS-40k, fixed PI control structures are used for the fan speed, core speed, and EPR control loop options. Scheduling variables are altitude and Mach number for the P-gain, and altitude, Mach number and fan speed for the I-gain. Figures 5.2 and 5.3 are graphical representations of the P and I-gain scheduling used by default in CMAPSS-40k. As conventionally used in GTE control systems,

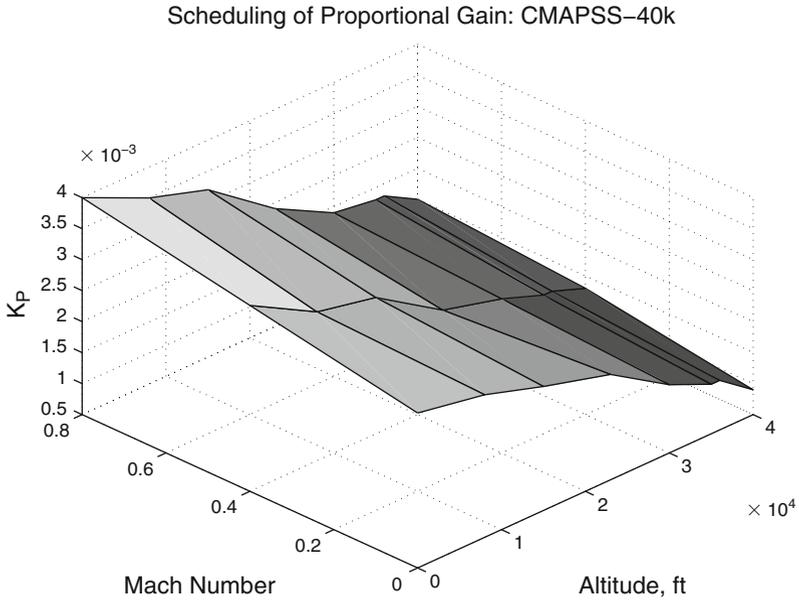


Fig. 5.2 Scheduling of proportional gain in CMAPSS-40k

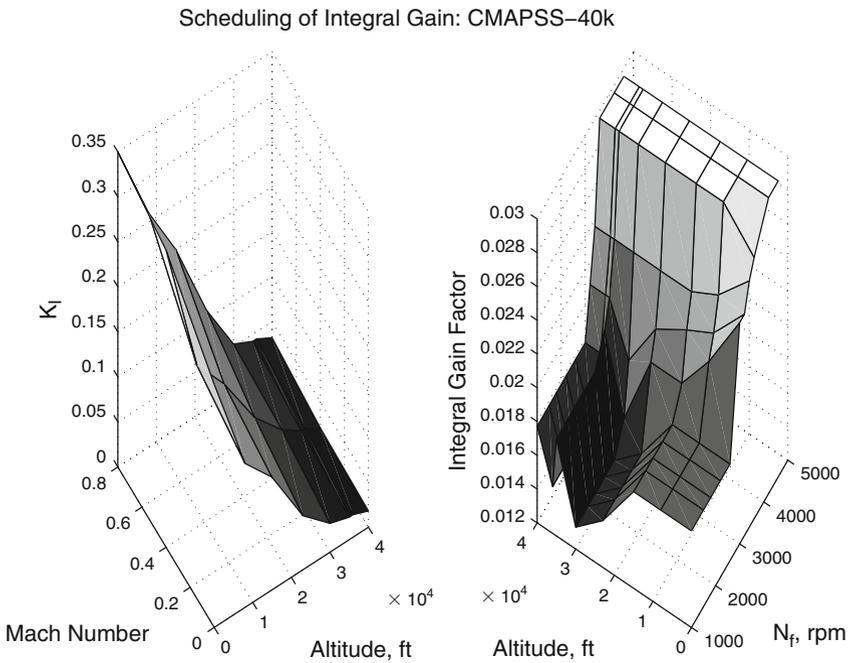


Fig. 5.3 Scheduling of integral gain in CMAPSS-40k. The value of K_i from the left plot is multiplied by the factor from the right plot

gain scheduling is a simple and effective way of achieving response specifications across the flight envelope under normal circumstances. The chief weakness of *ad-hoc* gain scheduling is its lack of robustness and its lack of adaptability to unforeseen conditions. If plant parameters change in ways that have not been accounted for in the scheduling tables, serious performance loss or instability can occur (See Shamma and Athans [38]). Engine aging and deterioration effects are in fact model parameters, whose variations are introduced as a disturbance input. These parameters are not typically used as scheduling variables. Doing so would lead to overly complex look-up tables. Besides, health parameters are not available as real-time measurements and their estimation and use in engine prognostics are challenging research problems. For more information on health parameter estimation, see references [20–22, 39].

5.3 Linear Parameter-Varying Methodologies

LPV control denotes a group of techniques based on a structured description of plant parameter variability. By imposing an LPV structure on the uncertain plant, robust designs based on \mathcal{H}_∞ theory become possible. The interested reader is referred to the work of Wolodkin et al. [40], where gain scheduling for turbofan engine control is conducted under an \mathcal{H}_∞ optimization objective. This approach offers superior performance in comparison with *ad-hoc* scheduling of several fixed designs.

An LPV description of the uncertain state-space plant has the form

$$\dot{x} = A(p)x + B(p)u, \quad (5.1)$$

$$y = C(p)x + D(p)u, \quad (5.2)$$

where $p = [p_1 \ p_2 \ p_3 \ \dots p_s]$ is a vector of s parameters, and system matrices are given by

$$A(p) = A_0 + p_1 A_1 + p_2 A_2 + \dots + p_s A_s, \quad (5.3)$$

$$B(p) = B_0 + p_1 B_1 + p_2 B_2 + \dots + p_s B_s, \quad (5.4)$$

$$C(p) = C_0 + p_1 C_1 + p_2 C_2 + \dots + p_s C_s, \quad (5.5)$$

$$D(p) = D_0 + p_1 D_1 + p_2 D_2 + \dots + p_s D_s, \quad (5.6)$$

where $A_0, A_1, \dots, B_0, B_1, \dots, C_0, C_1, \dots, D_0, D_1, \dots, D_s$ is a set of *coefficient matrices*. The parameters are chosen in the same way as scheduling parameters, that is, on the basis of knowledge or experience with the system. For fan speed control, a reasonable set of parameters is given by altitude, Mach number, and fan speed itself. For the remainder of the chapter, we assume that $p = [m \ h \ f]$, where m is the Mach number, h is the altitude normalized by a convenient scaling factor, and f is the fan speed, also normalized by a suitable scaling factor.

5.3.1 Obtaining an LPV Decomposition from Polytopic Vertices

Instances of $A(p)$, $B(p)$, $C(p)$, and $D(p)$ are available to the designer as an outcome of linearization at some steady-state condition determined by a fixed value of p and a set of matching equilibrium states and inputs. The coefficient matrices, however, need to be determined. When a set of instances of the system matrices corresponding to known parameter vectors are available, the coefficient matrices are determined through a generalized system of linear equations.

Suppose that a set of matrices $A(p(i))$, $i = 1, 2, \dots, r$ is available, corresponding to a set of parameter vectors $p(1), p(2), \dots, p(r)$. The following linear system arises from (5.3):

$$\begin{bmatrix} I & p_1(1)I & p_2(1)I & \dots & p_s(1)I \\ I & p_1(2)I & p_2(2)I & \dots & p_s(2)I \\ \vdots & & & & \vdots \\ I & p_1(r) & p_2(r)I & \dots & p_s(r)I \end{bmatrix} \begin{bmatrix} A_0 \\ A_1 \\ \vdots \\ A_s \end{bmatrix} = \begin{bmatrix} A(p_1) \\ A(p_2) \\ \vdots \\ A(p_s) \end{bmatrix}. \quad (5.7)$$

This linear system has the form $RV = S$, where V is the unknown matrix. The dimensions of R are nr -by- ns , where n is the dimension of each square matrix $A(p)$. If $r = s$ and $p(i)$ are chosen so that R has full rank, the solution can be found as $V = R^{-1}S$. If $r > s$, the system is overdetermined and an exact solution may be found only for a rather restrictive set of problem data in R and S . An approximate solution for V can be found by minimizing the 2-norm (largest singular value) of the residual matrix $RV - S$. The solution to this case is $V = R^+S$, where R^+ denotes the Moore–Penrose pseudoinverse [41] of R , which can be calculated in Matlab using the `pinv` command. The parameter vector chosen for GTE fan speed control has $s = 3$. Thus, three instances of system matrices would be necessary to obtain the LPV coefficient matrices using R^{-1} . Three flight conditions cannot be expected cover the operating envelope, implying that the pseudoinverse must be used.

5.3.1.1 CMAPSS-1 LPV Decomposition

The 14 conditions listed in Appendix B are readily used to obtain a set of four coefficients for $A(p)$ and $B(p)$. Altitude is normalized by 10,000 and fan speed by 3,000 to define the parameter vector. Matrix $A(p)$, for instance, is decomposed as

$$\begin{aligned} A(m, h, f) = & \begin{bmatrix} 0.1974 & 0.4174 \\ 0.1285 & 1.0424 \end{bmatrix} + m \begin{bmatrix} -1.0000 & 0.0626 \\ 0.1002 & -0.2370 \end{bmatrix} \\ & + h \begin{bmatrix} 0.7476 & -0.2551 \\ -0.0653 & 0.7102 \end{bmatrix} + f \begin{bmatrix} -5.0210 & 1.3617 \\ 0.5981 & -7.2702 \end{bmatrix}. \end{aligned}$$

To illustrate the magnitude of the error associated with the use of the pseudoinverse, consider $m = 0.7$, $h = 2$, and $f = 0.775$, corresponding to FC06. Computation of A from the LPV decomposition results in

$$\begin{bmatrix} -2.8971 & 1.0059 \\ 0.5314 & -3.3350 \end{bmatrix}.$$

This represents a 6.1% error in matrix 1-norm (largest column absolute value sum) relative to the true value of A at FC06. Similar errors exist for all other matrices and flight conditions.

5.3.2 A Simple LPV Approach to Fan Speed PI Control

Since fan speed increment is the first state in (5.1), an output y defined as fan speed increment will have constant $C = [1 \ 0]$ and $D = 0$. This implies that the transfer function from fuel flow increment $u = \Delta W_F$ to $y = \Delta N_f$ always has the form

$$G(s) = \frac{k(s+z)}{(s+c_1)(s+c_2)}, \quad (5.8)$$

where c_1 and c_2 may be complex conjugates. In CMAPSS-40k, linearization at high PLA levels tends to give complex poles, while real poles are seen at low PLA settings. In CMAPSS-1, real poles are observed for all 14 flight conditions. The technique presented in this section is restricted to plants with real poles only. The technique is based on the same premise as conventional gain scheduling: scheduling variables (in this case, parameters p) are available as real-time measurements from sensors. Then system matrices $A(p)$ and $B(p)$ can be computed in real-time using the LPV decomposition of (5.3) and (5.4). This information can be used to calculate the gains of a controller whose structure has been predetermined to meet performance and stability-related objectives.

Denote the entries of $A(p)$ and $B(p)$ as $a_{ij}(p)$ and $b_i(p)$, respectively, for $i = 1, 2$ and $j = 1, 2$. Transfer function $G(s)$ may then be parameterized by these entries by using a scalar version of (4.1):

$$G(s) = \frac{b_1s - b_1a_{22} + b_2a_{12}}{s^2 - (a_{11} + a_{12})s + a_{11}a_{22} - a_{12}a_{21}}. \quad (5.9)$$

Recalling Chap. 3, the PI controller structure is a suitable choice to meet the zero steady-state error requirement and obtain adequate transient responses. The control transfer function $K(s) = K_p + K_i/s$ leads to the closed-loop characteristic equation

$$\begin{aligned} s^3 - (a_{11} + a_{22} - K_p b_1)s^2 - (a_{12}a_{21} - a_{11}a_{22} - K_i b_1 - K_p a_{12}b_2 + K_p a_{22}b_1)s \\ + K_i a_{12}b_2 - K_i a_{22}b_1 = 0. \end{aligned}$$

This equation possesses three roots, of which at least one must be real, since complex roots appear in conjugate pairs. An approach to selecting control gains is to enforce a pole-zero cancelation in the closed-loop transfer function. Recall that the zeroes of $G(s)K(s)$ are the same as those of the closed-loop transfer function $T(s)$, up to pole-zero cancelations. In this technique, the zero introduced by the PI controller at $s = -K_i/K_p$ is to cancel a real root of the closed-loop characteristic equation. This is done to remove part of the response variability arising from plant parameter changes. To place a constrain on K_i and K_p so that the sought pole-zero cancelation occurs, a symbolic computation process must be carried out. First, $s = -K_i/K_p$ is substituted in the closed-loop characteristic equation. When this is done, the characteristic equation can be factored as $K_i Q(K_i, K_p) = 0$, where Q is a quadratic polynomial in K_i . Then two nonzero solutions for K_i are found from $Q(K_i, K_p) = 0$:

$$K_i^+ = -\frac{K_p}{2} (a_{11} + a_{22} + \sqrt{\Delta}), \quad (5.10)$$

$$K_i^- = -\frac{K_p}{2} (a_{11} + a_{22} - \sqrt{\Delta}), \quad (5.11)$$

where $\Delta = a_{11}^2 - 2a_{11}a_{22} + a_{22}^2 + 4a_{12}a_{21}$. It can be readily verified that $\frac{1}{2}(a_{11} + a_{22} \pm \sqrt{\Delta})$ are the eigenvalues of $A(p)$. Hence, the restriction of the technique to real eigenvalues is justified to prevent complex controller gains. Thus, each solution produces the cancelation of one of the two real plant poles. Furthermore, although the objective was to cancel a *closed-loop* pole, (5.10) and (5.11) show that the only way to achieve this is by direct cancelation of the plant *open-loop* pole. Hence, after cancelation, the loop transfer function has the form

$$G(s)K(s) = \frac{k'(s+z)}{s(s+c)},$$

where z is the plant zero, c is the un-cancelled plant pole, and k' is a new gain.

Equations (5.10) and (5.11) place a constraint on the relationship between K_i and K_p , but do not completely determine their values. A root locus argument can be used to show that sufficiently high values of K_p will result in an approximate cancelation of the plant zero. Furthermore, increasing K_p leads to faster responses and insensitivity from plant parameters. Indeed, assuming without loss of generality that $c_1 > c_2$ in (5.8), it is clear that using K_i^+ will result in the cancelation of c_1 and using K_i^- results in the cancelation of c_2 . Assume that K_i^+ is chosen. The open-loop transfer function is then

$$G(s)K(s) = K_p k \frac{(s+z)}{s(s+c_2)}.$$

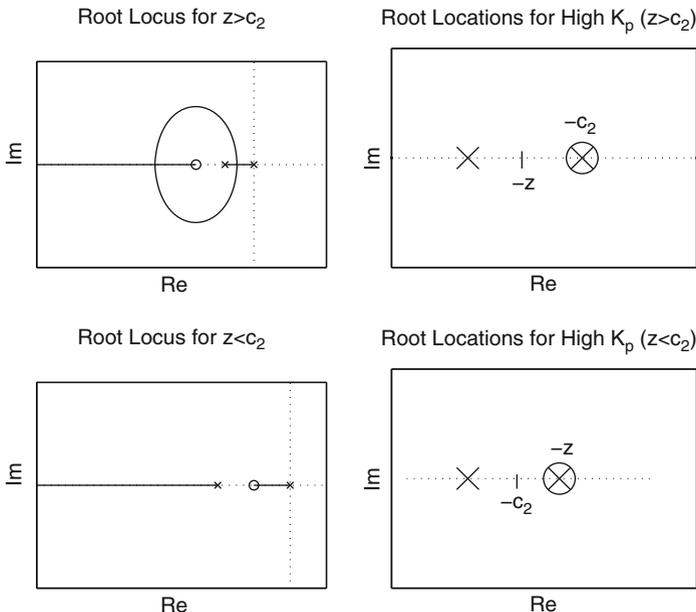


Fig. 5.4 Possible root loci and closed-loop root locations: pole-zero cancellation design with K_i^+ and high K_p

Two possible root loci are shown in Fig. 5.4, according to the relative magnitudes of z and c_2 . In both cases, a sufficiently high value of K_p will place a closed-loop pole at the location of the plant zero. The closed-loop transfer function is nearly independent from plant parameters. The remaining real pole proceeds to $-\infty$ as K_p is increased. This observation indicates that K_p does not need to be scheduled, but rather set at an appropriate value, best determined from simulation trials. The same general behavior is seen if K_i^- is chosen. In a real-time implementation, the eight coefficient matrices are stored in memory, from where the entries of $A(p)$ and $B(p)$ are computed. A fixed K_p and K_i^+ or K_i^- as calculated from (5.10) and (5.11) are used as PI gains.

5.3.2.1 Linearized Study: CMAPSS-1

A simulation study is conducted with $K_p = 0.1$, using the eight coefficient matrices of the LPV decomposition. Unit step responses corresponding to the 14 plants controlled with the LPV-based PI controlled are shown in Fig. 5.5. The results show that, in this case, K_i^+ tends to produce faster responses than K_i^- and that a constant K_p is sufficient to produce consistent transient responses across wide plant parameter changes.

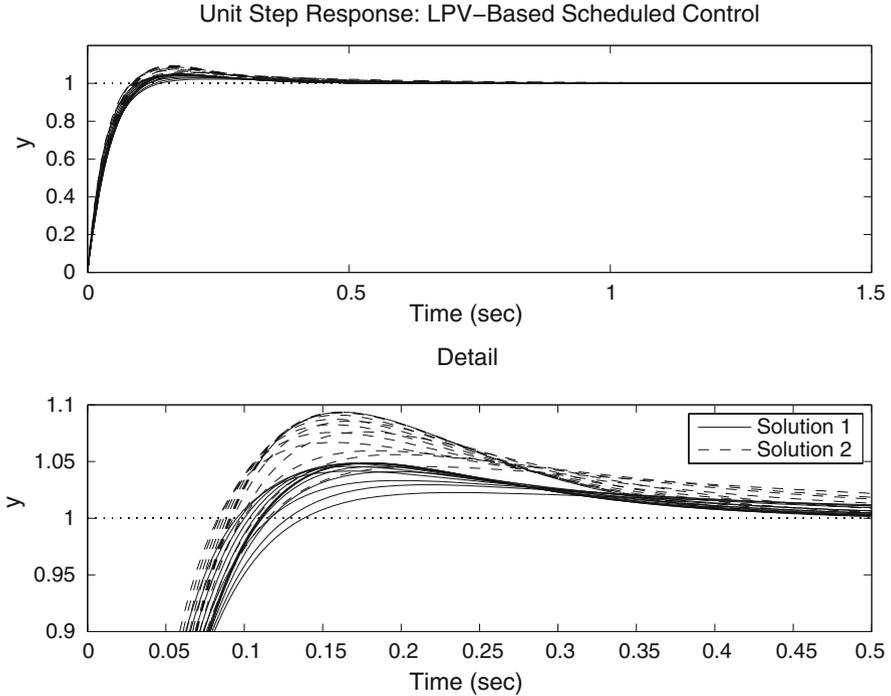


Fig. 5.5 Linearized responses of LPV-based scheduled controller across 14 flight conditions: CMAPSS-1

5.3.2.2 LPV Scheduling in Nonlinear Engine: CMAPSS-1

A simulation is now conducted using the nonlinear 90k engine. Altitude, Mach number, and fan speed are varied during the simulation, mimicking the parameter changes taking place during takeoff and climb. Initially, the engine is at sea level, the Mach number is zero, and the TRA is 20. This corresponds to FC13 in Appendix B, or near-idle conditions, where $N_f = 1,497$ rpm, as shown in Table 2.3. Altitude is changed to 20,000 ft., Mach number to 0.7, and TRA to 100, which correspond to FC06, where $N_f = 2,324$ rpm. The transition between altitude and Mach number parameters is taken as a 1-second ramp. Although no aircraft is capable of such fast altitude and airspeed changes, these parameter changes are useful to benchmark engine control systems. The fan speed reference input passed to the control system is given by a ramp having a slope of 500 rpm/s, the maximum admissible in CMAPSS-1. The proportional gain K_p is maintained at 0.1, and the formula for K_i^+ is used. Figure 5.6 shows that the LPV-based scheduling of the P-gain produces very accurate fan speed demand tracking. The lower plot shows that the control input is more aggressive in comparison with the native scheduled controller. Figure 5.7 shows the variation of K_i .

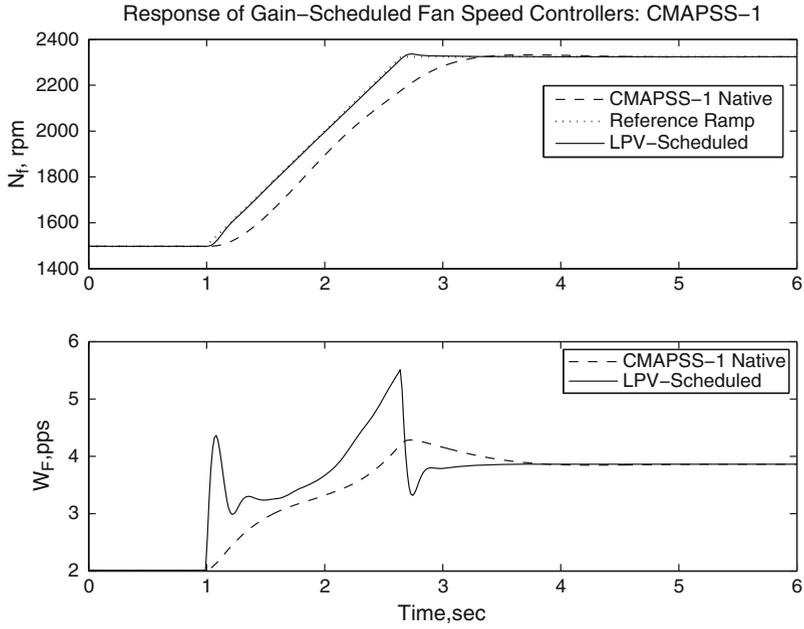


Fig. 5.6 Comparison of nonlinear engine responses. LPV-based scheduled controller and CMAPSS-1 native scheduled regulator

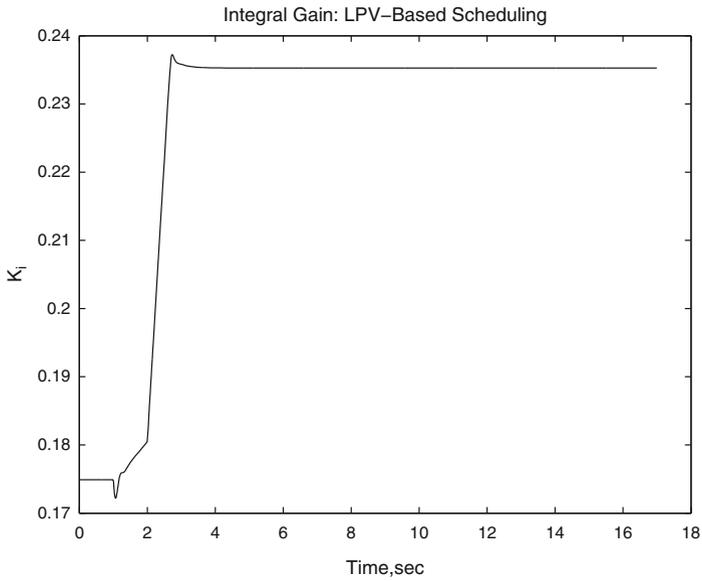


Fig. 5.7 LPV-based scheduled controller in CMAPSS-1 nonlinear engine: integral gain variation

5.3.3 Other LPV Approaches

The LPV-based pole-zero cancelation approach presented above represents an improvement over standard point design interpolations in a practical sense. Indeed, formulas replace tables, making it possible to generate gains with finer resolution in a compact format. The offline generation of coefficient matrices is very systematic and easily extensible to a larger number of vertices. The method, however, does not have any explicit robustness properties. Many multivariable robust approaches to gain scheduling using the LPV parameterization were developed in the 1990s. For a theoretical basis, see Kamen and Khargonekar [42], Apkarian and Gahinet [43] or Packard and Kantner [44]. For application of these techniques to aircraft engines, see Wolodkin et al. [40], Balas [45] or extensions of the LPV parameterization that allow polynomial dependence on coefficient matrices, but include linear dependence on controller parameters. Such *polynomial LPV synthesis* has been developed by SNECMA, a French aerospace manufacturer, see Henrion [46] and Gilbert [47].

5.4 Overview of Adaptive Control

The central idea of adaptive control schemes is to introduce a controller structure and a set of parameters Θ which, along with feedback measurements, determine the value of the control input at every instant. In ideal circumstances, when the plant is known exactly, Θ could be computed so that the closed-loop system has desirable characteristics. The premise justifying the use of adaptive control is that plant parameters are either uncertain or changing in time. Then Θ cannot be calculated beforehand. Various adaptive schemes introduce an *adaptation law*, or *parameter update law* to refine an initial guess of Θ during the course of system operation. Numerous adaptive schemes have been proposed, see, for instance, Åstrom and Wittenmark [48] or Ioannou [49]. Three architectures are frequently used: model-reference adaptive control (MRAC), indirect adaptive control, and self-tuning regulators. As Fig. 5.8 shows, the MRAC scheme introduces a reference model, which specifies the desired dynamics of the controlled plant. The MRAC is a dynamic system whose parameters Θ are adjusted so that the error between the reference model and the actual plant outputs is driven to zero. This approach can be categorized as *direct*, in that no attempt is made to estimate the unknown plant parameters as an intermediate step for the computation of the control input. The *MIT rule* was used in the early beginnings of adaptive control to update the parameters:

$$\frac{d\Theta}{dt} = -\gamma e \frac{\partial e}{\partial \Theta},$$

Fig. 5.8 Model-reference adaptive control system schematic

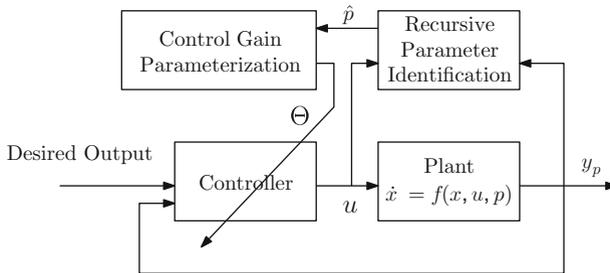
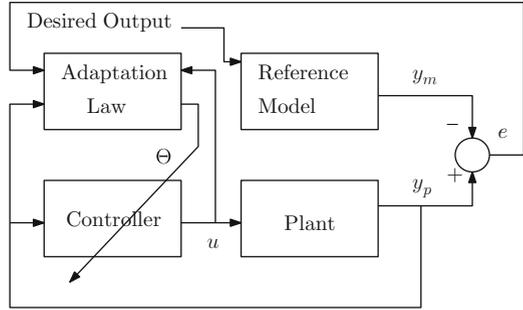


Fig. 5.9 Indirect adaptive control system schematic

where e is the error between plant and reference model outputs and γ is a positive constant governing the speed of adaptation. This rule was subsequently shown to lead to unpredictable system behavior, ranging from poor performance and slow adaptation to unstable closed-loop systems (see Anderson [50]). Present-day adaptation and control law synthesis are firmly grounded on Lyapunov stability theory.

Indirect adaptive approaches include a plant parameter estimator and a control design algorithm as an intermediate step, as shown in Fig. 5.9. The controller has been parameterized and tuned in terms of plant parameters, much like what was done in Sect. 5.3.2 using the LPV parameterization. A parameter estimator constantly updates plant parameters, which determine controller gains. Computationally, the only distinction between indirect adaptive control and LPV-based gain scheduling is the mechanism used to arrive at plant parameters: A recursive estimator is used in adaptive control. In contrast with the static, predetermined LPV parameterization, no offline information about plant variability is needed in the indirect adaptive case. A *self-tuning regulator* is a form of indirect adaptive control where plant parameter estimates are used to conduct controller design leading to the gains to be implemented in a fixed-structure controller, as shown in Fig. 5.10. This differs from other indirect schemes in that plant parameters are not directly used in the controller parameterization. Depending on the type of controller design to be performed, self-tuning regulators could represent a significant

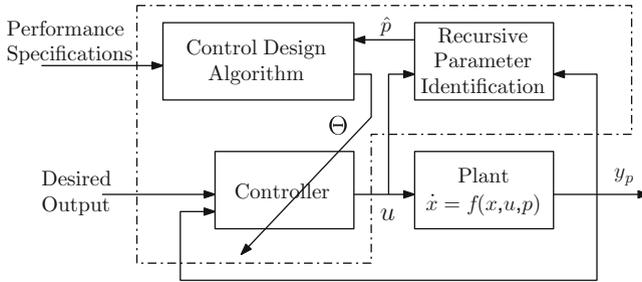


Fig. 5.10 Self-tuning regulator schematic

real-time computational burden. Some of the burden associated with the design can be transferred to offline computation, resulting in explicit implementations (see Grimble [51]).

5.4.1 Relative-Degree 1 MRAC

An MRAC scheme originally developed by Feuer and Morse [52] and further elaborated by Ioannou [49] is now summarized. This form of MRAC is applicable to linear SISO plants whose parameters are unknown, but whose *relative degree* is known to be equal to one. Recall that the relative degree of a linear transfer function is defined as the number of poles minus the number of zeroes. In addition, the plant transfer function is required to be minimum-phase and to have a high-frequency gain with known sign. Let the plant model be given in transfer function form as

$$Y_p(s) = G(s)U(s) = k \frac{N(s)}{D(s)}U(s), \quad (5.12)$$

where $N(s)$ and $D(s)$ are the numerator and denominator polynomials such that the degree of $D(s)$ is higher than the degree of $N(s)$ by one. The polynomials are assumed to be *monic*, that is, the leading coefficient (the coefficient of the highest power of s) of $N(s)$ and $D(s)$ must be one. In this case, the magnitude of the plant frequency response approaches an asymptote of the form $\frac{|k|}{\omega}$, where k is the high-frequency gain. These assumptions are satisfied by the transfer function from fuel flow to fan speed, upon which many standard GTE designs are based. The reference model is given by a transfer function of the form

$$Y_m(s) = W(s)R(s),$$

where r is the reference input to be tracked by y_p and $W(s)$ must be of relative degree one and also satisfy the following assumptions:

1. All poles of $W(s)$ must have negative real parts.
2. $W(s)$ must be minimum-phase.
3. The real part of $W(j\omega)$ must be nonnegative for all $\omega \geq 0$.
4. The high-frequency gain of $W(s)$ has the same sign as that of $G(s)$.

Conditions (2) and (3) above are satisfied by *strictly positive real* (SPR) transfer functions, which can be generated for any relative degree using two matrix conditions due to Kalman and Yakubovich, see [48]. In this section, we only consider reference models of the form

$$W_m(s) = \frac{1}{\tau s + 1}, \quad (5.13)$$

which satisfy all assumptions and are sufficient for an introductory exposition of MRAC methods. The speed of response of the reference model can be tuned using time constant τ .

This form of MRAC uses four controller gains assembled in a parameter vector $\Theta = [\Theta_1 \ \Theta_2 \ \Theta_3 \ \Theta_4]^T$. The control input is calculated as

$$u = \Theta^T \omega, \quad (5.14)$$

where $\omega = [\omega_1 \ \omega_2 \ y_p \ r]^T$. Quantities ω_1 and ω_2 are the outputs of first-order filters of the form

$$\dot{\omega}_1 = F\omega_1 + gu_p, \quad (5.15)$$

$$\dot{\omega}_2 = F\omega_2 + gy_p, \quad (5.16)$$

whose initial conditions are set as $\omega_1(0) = \omega_2(0) = 0$. An arbitrary nonzero value is chosen for g and F is chosen so that the above filters are stable, that is, $F < 0$. Finally, the parameter adaptation law is given by

$$\dot{\Theta} = -Ge\omega \text{ sign}(k_p k_m), \quad (5.17)$$

where k_m is the model's high-frequency gain, G is a tunable positive-definite matrix and the error has been defined as

$$e = y_p - y_m.$$

Matrix G controls the rate of parameter adaptation.

5.4.2 Example: CMAPSS-1

The MRAC scheme is tested first using the 14 linearized plant models of CMAPSS-1. A step command for ΔN_f of 100 rpm is used as reference input. The reference model was chosen as

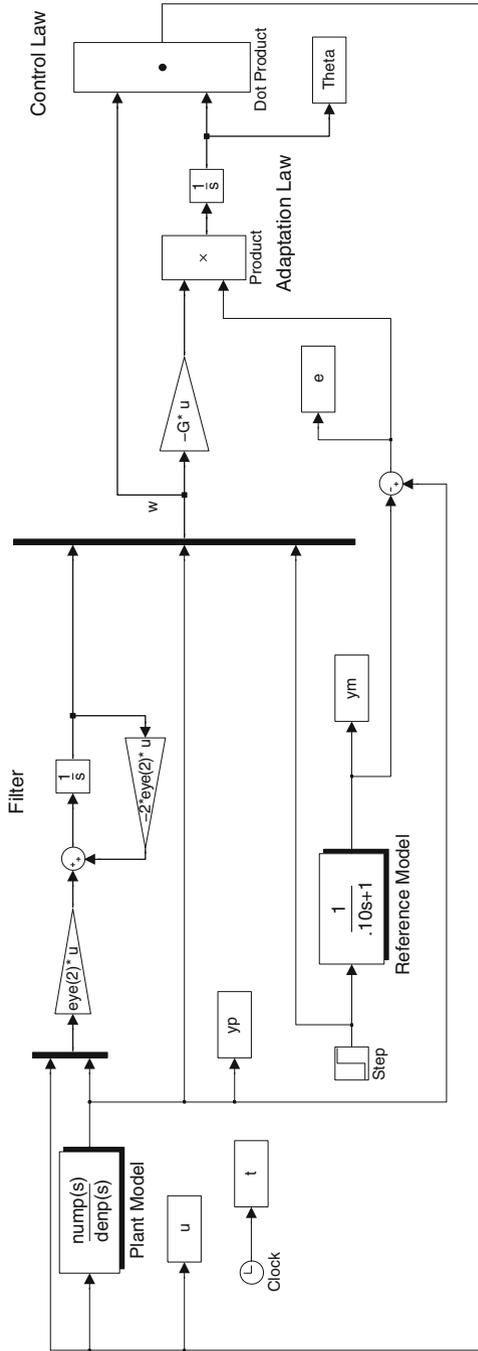


Fig. 5.11 Simulation diagram for model-reference adaptive control with linearized plant model

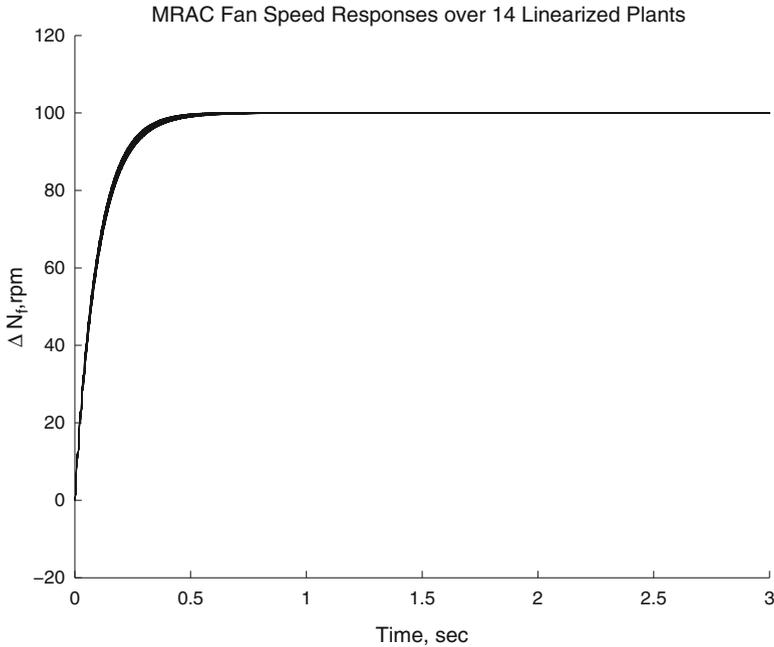


Fig. 5.12 Fan speed responses of model-reference adaptive control over 14 linearized plants

$$W_m(s) = \frac{1}{0.1s + 1}.$$

The adaptive gains were chosen with almost no trial-and-error as

$$G = \begin{bmatrix} 0.1 & 0 & 0 & 0 \\ 0 & 0.1 & 0 & 0 \\ 0 & 0 & 0.1 & 0 \\ 0 & 0 & 0 & 0.1 \end{bmatrix}, \quad F = -2 \quad g = 1.$$

The reader interested in reproducing these results may refer to Fig. 5.11, which shows the Simulink implementation. The initial parameter vector was chosen as $\Theta_0 = [0 \ 0 \ 0 \ 0]^T$. The step response simulation was repeated using the 14 transfer functions from incremental fuel flow to incremental fan speed using the data from Appendix B, and without modifications to G , F , g , or Θ_0 . Figure 5.12 shows that the fan speed responses closely match each other and the reference model: there is no overshoot and the settling time is about four time constants. Figure 5.13 shows how parameters adapt and converge to different values to achieve the specified model-matching objective.

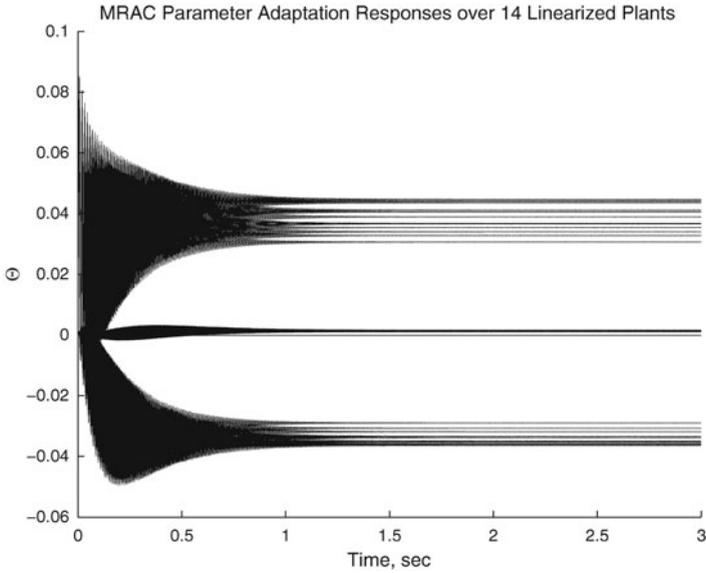


Fig. 5.13 Parameter adaptation responses of model-reference adaptive control over 14 linearized plants

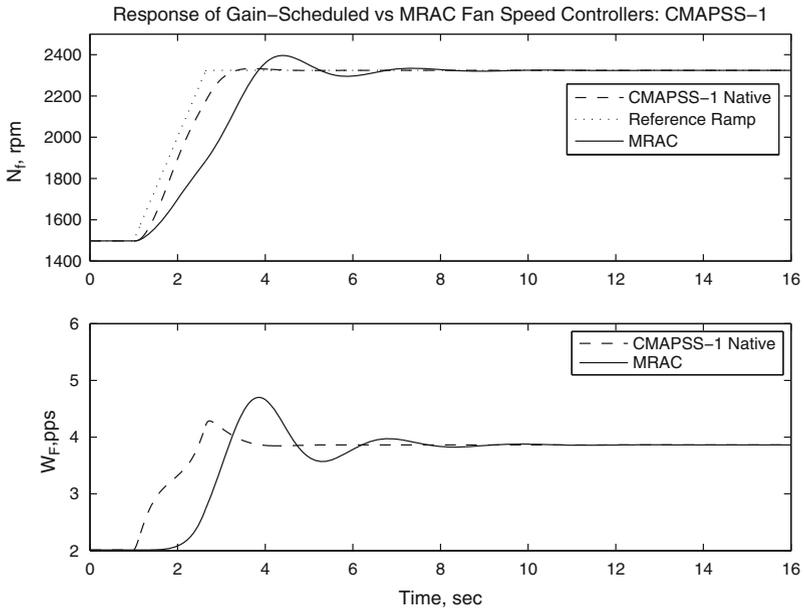


Fig. 5.14 Comparison of nonlinear engine responses: model-reference adaptive control and CMAPSS-1 native scheduled regulator

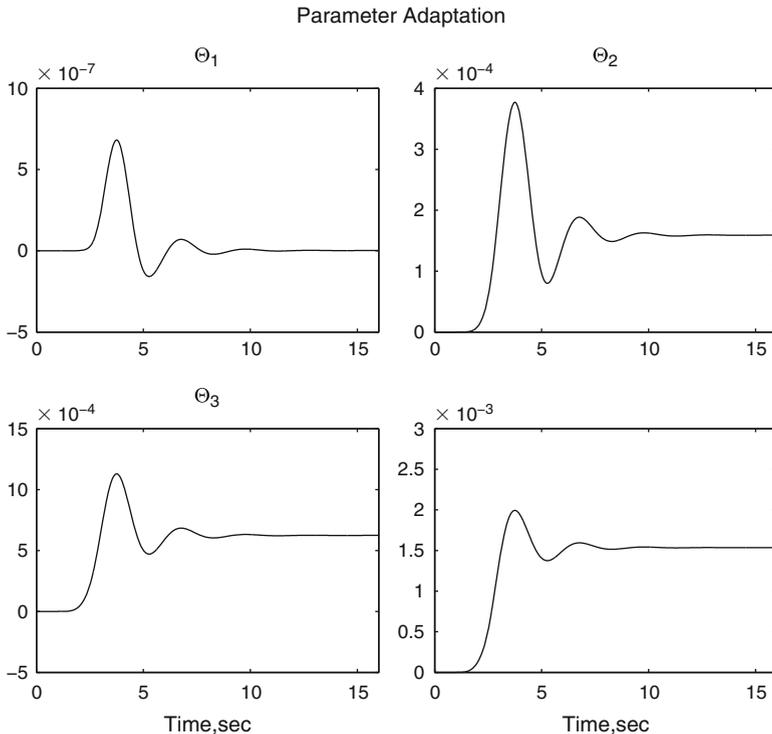


Fig. 5.15 Model-reference adaptive control in nonlinear engine: parameter adaptation histories

The relative-degree 1 MRAC design is now applied to the 90k nonlinear engine of CMAPSS-1. Altitude, Mach number, and fan speed are varied during simulation, the same way as in the example of Sect. 5.3.2.2. The MRAC system was tuned with $G = 1 \times 10^{-8} I_4$, $\tau = 0.5$, and initial parameter vector $\Theta_0 = [0 \ 0 \ 0 \ 0]^T$. Figure 5.14 shows the fan speed and fuel flow input responses corresponding to the MRAC in comparison with the CMAPSS-1 native gain-scheduled fan speed controller. Even with zero as initial parameter guesses, the adaptive control system is able to attain zero offset and reasonable transient response characteristics. Figure 5.15 shows how parameters are adapted and converge to steady values. A “bootstrapping” tuning procedure may be used to improve transient response. After a first simulation with $\Theta_0 = 0$, the resulting steady values of Θ may be used as Θ_0 . A new simulation is run, with an improvement in transient response. The first simulation converged to $\Theta = [0 \ 0.0002 \ 0.0006 \ 0.0015]^T$. Figure 5.16 shows how performance is significantly improved by setting Θ_0 to these new values. The process may be repeated to produce further improvements, but the refinement process will tend to tailor Θ_0 to a particular simulation, and the same transient qualities will not be observed for a new set of plant parameter changes. A good choice for Θ_0 is best obtained by a simulation study comprising several plant parameter variation

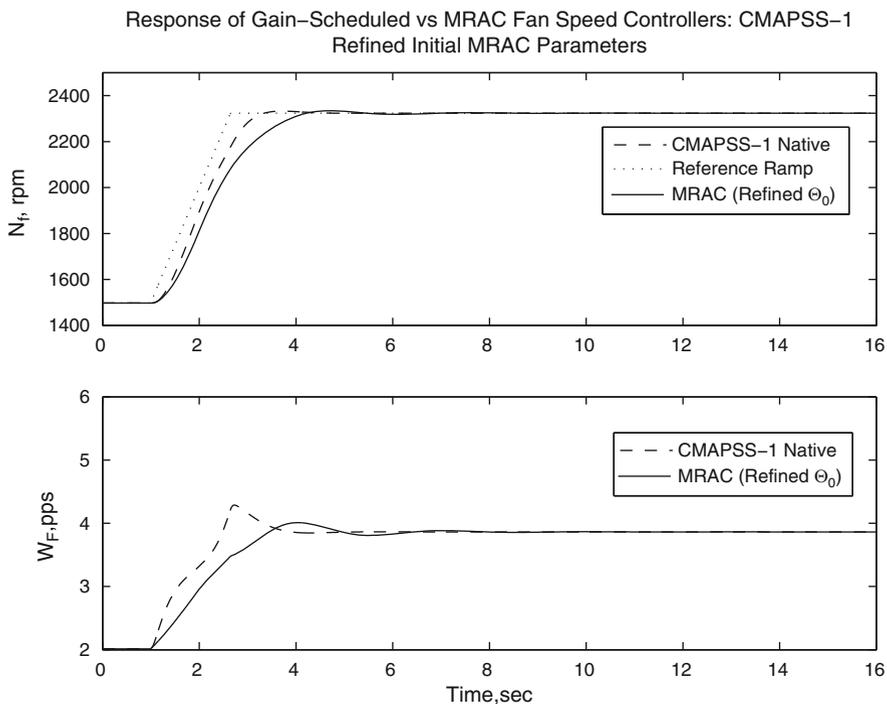


Fig. 5.16 Response of model-reference adaptive control with initial parameter refinement

scenarios. Theoretical results guarantee that the error will converge to zero and that the parameters will converge to steady values. Instability can be observed in simulation, however, due to interaction of adaptive dynamics and numerical solution algorithms. The CMAPSS simulation presented here used a 2nd-order Runge-Kutta method (Heun’s algorithm) with a fixed step size of 0.015. Under these conditions, choosing the entries of G as 1×10^{-7} leads to large numerically-induced oscillations. Values near 5×10^{-7} cause unboundedness of Θ and ω .