Chapter 3
Engine Control by Classical Methods

Abstract This chapter reviews and applies classical SISO design techniques (root locus and frequency domain loopshaping) to the problem of fan speed control using fuel flow rate as control input. A model-matching method is also described that is used in CMAPSS as a design tool. The shortcomings associated with the use of fixed linear compensation are illustrated with simulation examples.

As expected, classical linear compensation is adequate only to govern the engine close to a fixed operating point, as defined by the current inlet conditions and desired thrust setpoint. Engine accelerations across wide fan speed ranges, as well as thrust regulation across changing inlet conditions are handled poorly when a fixed linear controller is used. Aside from nonlinearity and parametric changes in the plant, critical variables must be maintained within safety ranges.

Linear compensation, however, is the basic building block of standard GTE control systems. Parametric changes and nonlinearity are addressed conventionally addressed with gain-scheduled linear compensators, while *limit protection logic* schemes are used to override the active linear regulator when a critical variable approaches its safety limit. In this chapter, three basic design approaches to thrust regulation by means of classical linear compensation are examined. Fuel flow rate is considered to be the only control actuator. At the end of the chapter, a CMAPSS simulation is presented that exposes the limitations of fixed-regulator schemes.

3.1 Setpoint Control via EPR or Fan Speed

As this book is being printed, no direct sensing technology yet exists that is capable of producing reliable thrust measurements suitable for feedback. Thrust estimation from other sensed quantities is very challenging due to its strong dependence on the engine's health condition, which is not precisely known [24, 25]. For this reason, alternative variables that can be reliably sensed and which are a proxy for thrust are used. Among these, EPR and N_f are commonly-used. A table-lookup routine can determine the value of EPR or N_f that results in the desired F_n setpoint, given

current inlet conditions. For all practical purposes, compensator design can proceed by assuming that a setpoint or reference profile has been given in terms of either ΔN_f or ΔEPR.

The transfer functions from ΔW_F to ΔEPR and ΔN_f are directly obtained from the linearized engine model at the appropriate flight condition, and have the form

$$\frac{\Delta \text{EPR}}{\Delta W_F} = \frac{a_0 s^2 + a_1 s + a_2}{s^2 + c_1 s + c_2}, \tag{3.1}$$

$$\frac{\Delta N_f}{\Delta W_F} = \frac{b_0 s^2 + b_1 s + b_2}{s^2 + c_1 s + c_2}. \tag{3.2}$$

Coefficients c_1 and c_2 are intrinsic to the flight condition and fixed for all engine outputs. The only assumption made about these coefficients is that they define a pair of transfer function poles having negative real parts. That is, linearized engine models are inherently stable. Nothing can be assumed about numerator coefficients, leaving open the possibility that they define transfer function zeroes with positive real parts, (*non-minimum phase zeroes*). In certain cases, however, these zeroes have real parts which are very large in comparison with the absolute value of the real parts of other zeroes and poles. Such high-frequency dynamics can usually be ignored without detriment to the accuracy of the linear model, provided the low-frequency gain of the transfer function is preserved [26]. Whenever a control technique discussed in this book is unable to handle nonminimum-phase systems, the necessary assumption will be made explicit.

3.1.1 Integral Control

It is an established fact of linear control theory [26] that a feedback compensator loop must display at least one *free integrator* (pole at the origin) for offset-free set-point attainment. In addition, such *Type 1* control loop offers enhanced disturbance rejection abilities. Indeed, step disturbances at plant input are completely rejected. Since transfer functions (3.1) and (3.2) cannot be assumed to contain a pole at the origin, a free integrator must be implemented as part of the controller.

Integral design proceeds by cascading the free integrator with the plant transfer function to form an *augmented plant* model, used as the basis for compensator selection. The zero-pole landscape associated with the augmented plant determines the simplest compensator structure to be attempted.

3.1.2 Compensator Design with the Root Locus

The classical root locus methodology is readily applied, since the plant models of (3.1) and (3.2) are of low order. Recalling classical control concepts, the objective

is to introduce a compensator and choose the loop gain so that all closed-loop poles have negative real parts, while a number of *dominant* closed-loop poles belong to a region of the complex plane which corresponds to desirable transient response. A set of poles is dominant relative to the remaining poles when their time constants are significantly larger than the remaining time constants. A time constant ratio of 8 between the fastest dominant pole and the slowest nondominant pole is typically adopted as a dominance criterion [26].

Root locus design frequently seeks to obtain *one* pair of dominant complex poles. This is because the relationship between pole locations and transient response is straightforward for the standard second-order transfer function:

$$G(s) = c\,\frac{w_n^2}{s^2 + 2\zeta w_n s + w_n^2},$$

where $c = G(0)$ is the *static*, or *DC gain*, w_n is the natural frequency, and ζ is the damping ratio. Readers are referred to Dorf [26] if a review of second-order transient properties is needed. When $0 < \zeta < 1$, the response of $G(s)$ to a step input is said to be *underdamped*, and the roots of the denominator are complex and equal to $-\zeta w_n \pm \sqrt{1 - \zeta^2} w_n i$.

The step response can be readily predicted in terms of percent overshoot and settling time according to the formulas

$$P.O. = 100\mathrm{e}^{-\left(\frac{\zeta}{\sqrt{1-\zeta^2}}\right)\pi}, \tag{3.3}$$

$$t_s = \frac{4}{\zeta w_n}. \tag{3.4}$$

According to these formulas, overshoot is a decreasing function of damping ratio, while settling time decreases with the absolute value of the real part of the complex poles. A combination of maximum allowable overshoot and settling time then results in a trapezoidal target region for the dominant poles, as illustrated in Figs. 3.1 and 3.2.

When more than two poles are dominant or if zeroes are present in the dominance region, the above formulas and target region do not apply, and iterative design must be carried out with the aid of a simulation package. In the simplest situation, the augmented plant has no zeroes (the coefficients of s^2 and s in the numerator are zero). As seen in Fig. 3.1 (top), proportional control is the simplest stabilizing compensator. However, the free integrator contributed by the controller causes two branches of the locus to cross the imaginary axis, introducing a limit in the loop gain for stability. When plant open-loop poles are close to the imaginary axis, slow response times will be obtained, and the design specifications become unfeasible under this compensator structure. A zero can be contributed by the compensator (PD control) to distort the root locus branches so that faster responses are possible, as seen in Fig. 3.1 (bottom). Since the free integrator belongs to the controller, an overall compensator with no more zeroes than poles is obtained. Tuning is done through the location of the zero and the value of the controller gain.
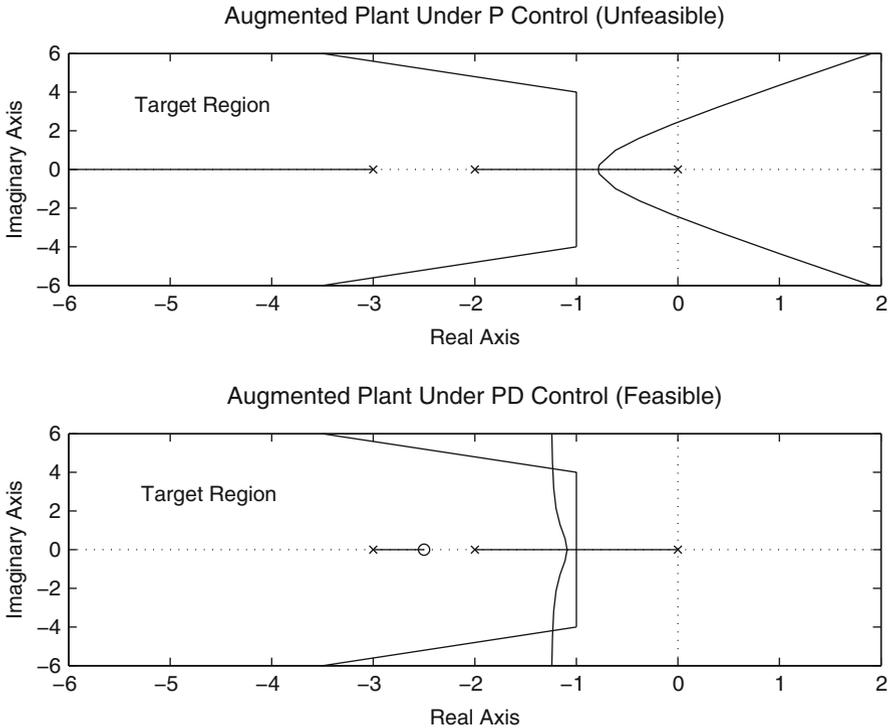
Augmented Plant Under P Control (Unfeasible)

Augmented Plant Under PD Control (Feasible)

Fig. 3.1 Root locus analysis of augmented plant with no zeroes

In contrast, one or two zeroes may exist, and they may be nonminimum phase. As shown in Fig. 3.2 (top), the loop may be inherently unstable, precluding the use of a free integrator. Alternatively, it may be possible to use negative proportional gain, as shown in Fig. 3.2, followed by a compensator zero. The reader must keep in mind that the designed compensators produce ΔW_F as their output, which is allowed to be negative. This incremental control must be added to the baseline W_F corresponding to the flight condition under which the design is being carried out. Care must be exercised in keeping W_F between realistic limits by testing the linear compensator against the full nonlinear engine model.

3.1.3 Compensation in the Frequency Domain: Manual Loopshaping

Although it may be possible to meet transient response specifications using the root locus and trial-and-error procedures, the frequency domain method bridges specifications and designer inputs in a more direct fashion. Some trial-and-error is
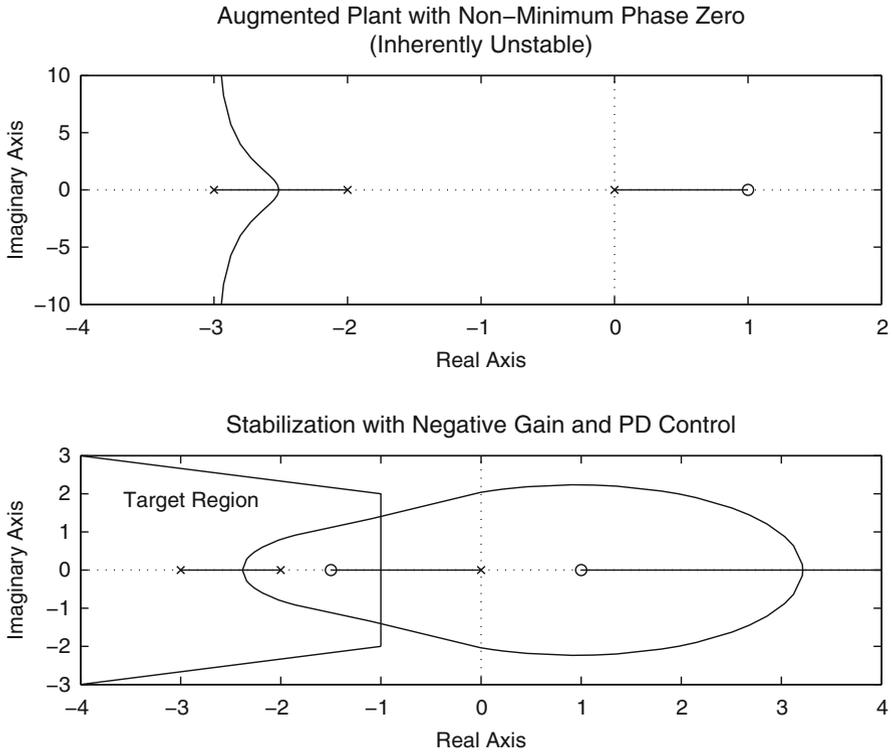
Augmented Plant with Non–Minimum Phase Zero
(Inherently Unstable)

Stabilization with Negative Gain and PD Control

Fig. 3.2   Root locus analysis of augmented plant with non-minimum phase zero

still required to fine tune the design. In classical loopshaping, the designer attempts to reproduce the Bode plot of a target open-loop transfer function $L(s) = G(s)K(s)$ by manipulation of the zeroes, poles, and gain of $K(s)$. The target transfer function is specified by its Bode plot features. These features are obtained from an open-loop target of the form

$$L(s) = G(s)K(s) = \frac{w_n^2}{s(s + 2\zeta w_n)}, \tag{3.5}$$

where w_n and ζ are chosen so that the closed-loop transfer function

$$T(s) = \frac{L(s)}{1 + L(s)} = \frac{w_n^2}{s^2 + 2\zeta w_n s + w_n^2}$$

has a transient response matching the design specifications. Standard formulas [26] link w_n and ζ to step response characteristics such as percent overshoot and settling time:

$$PM \approx 100\zeta, \tag{3.6}$$

$$w_b \approx (-1.196\zeta + 1.85)w_n, \tag{3.7}$$

$$w_c \approx w_b/1.6, \tag{3.8}$$

where w_b is the *closed-loop bandwidth*, the frequency at which the magnitude of $T(s)$ reaches -3 dB, and w_c is the *crossover frequency*, at which the magnitude of $L(s)$ is 0 dB. These formulas are good approximations when ζ is between 0.3 and 0.8. The design process is best accomplished with the aid of software packages such as SISOtool, a graphical user interface for interactive controller design in Matlab. Assuming that zero steady-state error to step inputs is desired along with a settling time t_s and overshoot $P.0.$, the process is as follows:

1. Use (3.3) to calculate the required damping ratio ζ using the $P.O.$ specification.
2. Use (3.4) to calculate the required natural frequency w_n using ζ from above and the t_s specification.
3. Use formulas (3.6)–(3.8) to calculate the closed-loop bandwidth w_b, the target crossover frequency w_c and the target phase margin PM.
4. According to the features of the uncompensated loop $G(s)$, zeroes and poles are added to $K(s)$ and its gain is adjusted in SISOtool until $L(s)$ attains the target phase margin and crossover frequency.

If the design process leads to an $L(s)$ that attains the target phase margin and crossover frequency while having a pole-zero structure different than that of (3.5), it should be emphasized that the original time-domain specifications should still be approximately met, provided that the designed $L(s)$ contains an integrator and a dominant real pole is achieved. That is, if the dominant factor of the designed $L(s)$ has the form

$$\frac{k}{s(s+p)},$$

then it corresponds to the target of (3.5) with $w_n = \sqrt{k}$ and $\zeta = \frac{p}{2\sqrt{k}}$.

3.1.4 Edmund's Model-Matching Method

A model-matching approach due to Edmunds [27] is implemented as a controller design tool in CMAPSS-1. It is essentially an automated frequency domain loop-shaping approach, where the target is the closed loop magnitude response, defined by the tunable bandwidth and damping ratio parameters. It also includes a tunable real pole beyond the closed-loop bandwidth. A least-squares optimization process is used to arrive at a controller that produces a closed-loop frequency response having the specified real pole and the intended bandwidth and damping ratio. Note from (3.6) that the damping ratio is an indirect phase margin specification. The advantage of the method is that it requires only three parameters as user input, eliminating iterative design and thus being suitable for automated design. Unfortunately, the least-squares process may produce spurious nonminimum phase

zeroes in the controller. These zeroes may compromise the performance and stability of the loop, invalidating the design if they cannot be removed on the basis of root dominance.

3.1.5   Comparative Example

A compensator is to be designed to produce an increment of 100 RPM in fan speed under the following requirements:

1. Zero steady-state error.
2. Overshoot less than 5%.
3. Settling time near 1 s.

The designs are conducted using the CMAPSS-1 90k-class engine model at FC01. Comparisons among root locus, loopshaping, and Edmund's model matching method will be carried out. Finally, the controllers are applied to the engine model at a drastically different flight condition (FC08) to illustrate performance deterioration and motivate the need for advanced controllers. From the tables in Appendix B, the transfer function from ΔW_F in pounds per second (pps) to ΔN_F in RPM at FC01 can be obtained from the state-space matrices as

$$\frac{\Delta N_f}{\Delta W_F} = \frac{230.7s + 2032}{s^2 + 8.564s + 17.47}. \tag{3.9}$$

The plant features two stable real poles and one minimum-phase zero. The zero-steady state error requirement is addressed by including an integrator at plant input, while formulas (3.3) and (3.4) imply that the damping ratio must be greater than 0.7 and that the dominant poles must have real part less than -4. The zero of the compensator and the gain are readily tuned with the aid of design packages such as SISOtool, part of Matlab. The overall compensator becomes

$$K(s) = 0.016 \frac{s + 3.70}{s}.$$

Figure 3.3 shows the root locus of the compensated loop. The closed-loop poles are $-4.19 \pm 3.67i$ and -3.85. Note that there is a real closed-loop pole in addition to the complex pair used to define the target trapezoidal region. This pole nearly cancels the zero of the plant, resulting in little deviation from the projected step response.

   Manual loopshaping design is achieved by translating time domain performance specifications into a set of parameters for the target loop. Using the formulas in (3.6)–(3.8), the target crossover frequency is calculated as $w_c = 3.62$ rad/s, and the target phase margin is $PM = 70°$. The target loop shape is achieved by including a real zero in the controller in addition to the integrator and by tuning the gain.
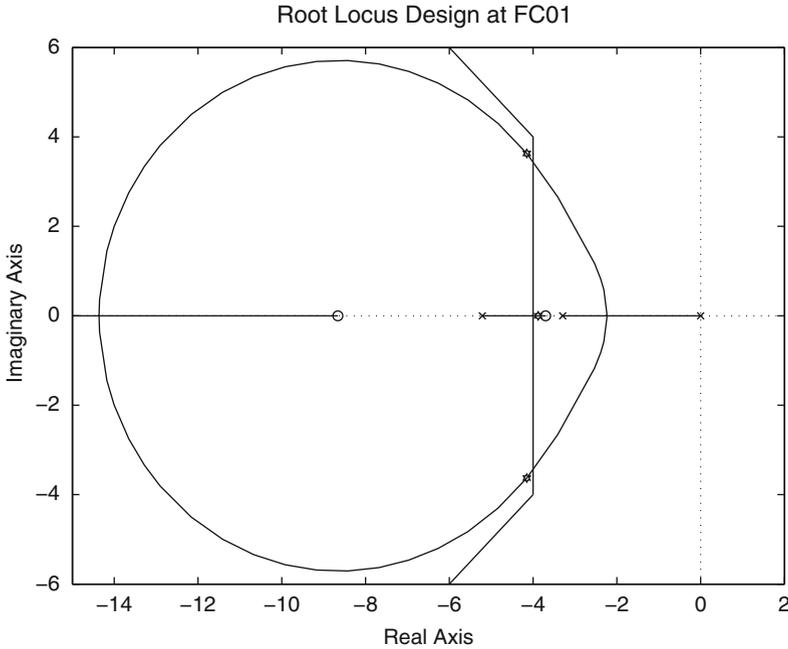
Fig. 3.3 Root locus design at FC01

The location of the zero and the value of the gain are chosen with minimal iterations in SISOtool. The resulting compensator becomes

$$K(s) = 0.012 \frac{s + 3.45}{s}.$$

This controller achieves a phase margin of 76.1° and a crossover frequency of 3.99 rad/s, as shown in Fig. 3.4.

Edmund's method is applied next. Formulas (3.6)–(3.8) indicate, again, that a closed-loop bandwidth of $w_b = 5.79$ rad/s and a damping ratio of $\zeta = 0.7$ match the specifications of this example. No guidance for the selection of the real pole is offered by the model-matching method, so it is arbitrarily set to -20. The built-in model-matching solver in CMAPSS gives the controller

$$K(s) = 0.21 \frac{s + 3.715}{s(s + 20)}. \tag{3.10}$$

Note that all three methods yield controllers attempting to cancel the zero of the plant. The transient response is about the same for the three designs, as shown in Fig. 3.5.
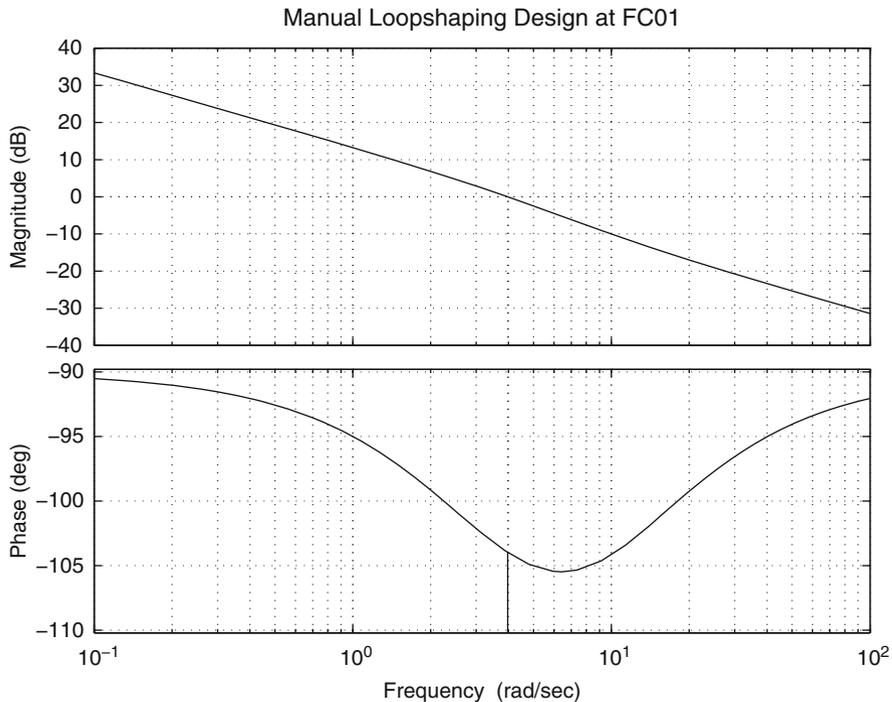
Fig. 3.4 Manual loopshaping design at FC01

3.2 Shortcomings of Fixed Linear Compensator Designs

Although only three design methods have been discussed, it should be clear that other classical compensation techniques still produce a fixed control transfer function. Advanced linear compensation methods such as $\mathcal{H}_\infty$ control and μ synthesis [28–30] ultimately deliver a fixed compensator. If properly conducted, these compensators should match the specified nominal transient response. These optimized compensators, however, offer enhanced robustness properties. In the GTE control problem, a properly designed robust compensator will maintain prescribed degrees of closed-loop stability and performance as the parameters of the plant change. As seen earlier, linearized models change due to varying inlet conditions, inherent engine nonlinearity and engine aging.

As it will become evident in the following sections and chapters, linear compensators designed on the basis of linearized models are sufficient to maintain stability in the face of plant parameter variations and inherent engine nonlinearity. *The major challenge for control design arises from the need to maintain critical operating variables within allowable limits, without undue penalties to transient response qualities.* The tendency of variables such as stall margin and turbine temperature
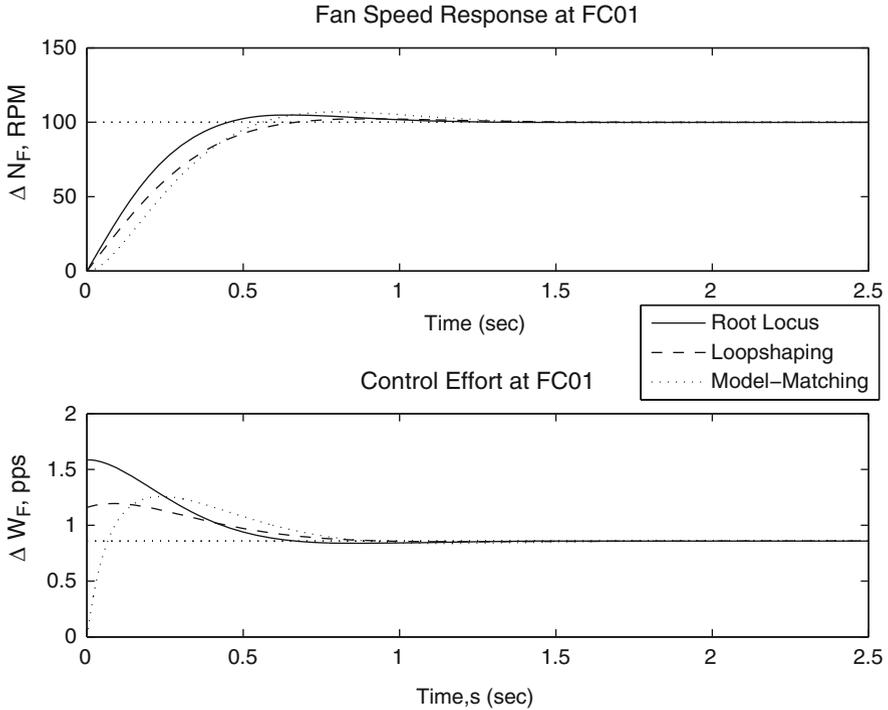
Fan Speed Response at FC01

Control Effort at FC01

Fig. 3.5 Comparison of step responses obtained with root locus, manual loopshaping, and model-matching designs at FC01

to cross their allowable limits is exacerbated by engine aging, which increases the limit protection challenge by introducing robustness requirements.

3.2.1 Parameter Variations Across the Flight Envelope

To illustrate the extent of performance loss due to parametric changes in the plant, the loopshaped controller designed at FC01 is applied to the linearized model at FC08 (see Table 2.3), given by

$$\frac{\Delta N_f}{\Delta W_F} = \frac{252.2s + 1011}{s^2 + 3.919s + 3.528}.$$

A quick calculation reveals that the phase margin is reduced to $42.2°$, which predicts a higher overshoot (30%), as shown in Fig. 3.6. Although both settling time and overshoot have fallen outside specifications, performance loss can be qualified as mild, and stability is not compromised even at this drastically different flight
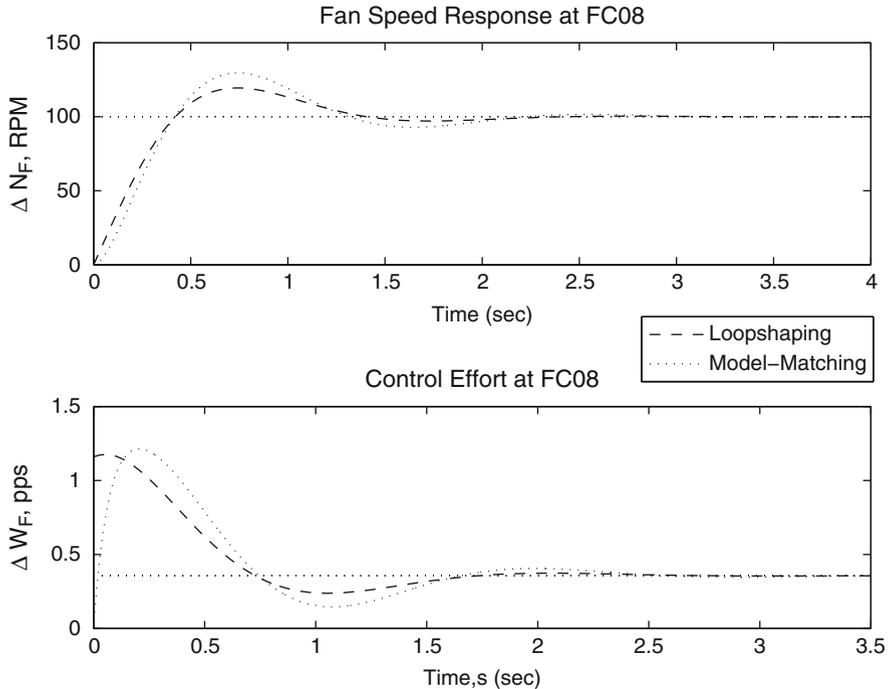
Fig. 3.6 Response of controllers designed at FC01 when applied to FC08

condition. Robust linear controllers designed with suitable methods are able to maintain specifications across a larger range of operating conditions, as shown in Chap. 4.

Gain scheduling, or the tailoring of controller gains to operating points or regions, has been used by the GTE industry for decades as the standard way to account for large parameter variations. To illustrate the extent of these variations, consider the linearized plant transfer functions at FC01 and FC08. A quick examination reveals that their four parameters (two pole locations, one zero location and a gain) have undergone significant changes. In a typical gain scheduling design, linear functions linking plant parameters to a set of *scheduling variables* are sought. Typical choices for scheduling variables in GTE control are inlet static pressure and fan speed itself. The first scheduling variable accounts for parametric changes arising from varying altitude, while the second captures intrinsic plant nonlinearity. Fixed linear compensators are then designed for various combinations of scheduling variables. Gain interpolation is used during real-time operation. Alternatively, it may be possible to parameterize controller gains using the scheduling variables as parameters. Gains are computed in real-time using formulas rather than table look-ups. Gain scheduling and its parent technique, linear-parameter-varying (LPV) control, will be examined in Chap. 5.
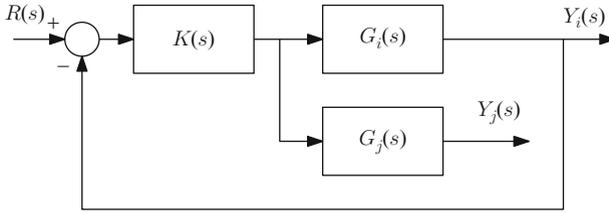
Fig. 3.7 Block diagram for output y_j under closed-loop control on y_i

3.2.2 Engine Limits

As discussed in Sect. 1.3, several critical engine variables must be kept under allowable limits at all times. Engine models such as those included in the CMAPSS family include critical variables among the outputs. Moreover, the linearization functions in all versions of CMAPSS produce output matrices for these outputs. Given a flight condition and its associated A and B matrices, a pair of C and D matrices defines a transfer function from fuel flow increment to an incremental output of choice. These output transfer functions are essential for the design of limit protection strategies. For instance, the linearized transfer function for the high-pressure turbine outlet temperature at FC01 is

$$\frac{\Delta T_{48}}{\Delta W_{\mathrm{F}}} = 146.24 \frac{(s + 4.73)(s + 2.30)}{s^2 + 8.564s + 17.47},$$

where ΔT_{48} is in $^\circ R$. As expected, the denominator of the above transfer function is the same as that of (3.9). Indeed, the set of poles of the transfer function from ΔW_{F} to any system output is the constant for a fixed flight condition. The transfer function poles match the eigenvalues of state-space matrix A. Thus, the differences in the transient behavior of critical outputs are characterized by the zeroes of their respective transfer functions. Suppose $G_i(s)$, $G_j(s)$ are two output transfer functions, and $K(s)$ is a compensator placed in a feedback loop involving $G_i(s)$, as shown in Fig. 3.7. The closed-loop transfer function relative to $G_j(s)$ is given by

$$\frac{Y_j(s)}{R(s)} = \frac{G_j(s)K(s)}{1 + G_i(s)K(s)}. \tag{3.11}$$

Using this formula, the closed-loop transfer function between fan speed increment demand, N_{f}, dmd and ΔT_{48} under the loopshaped fan speed controller becomes

$$\frac{\Delta T_{48}}{\Delta N_{\mathrm{f}}, \mathrm{dmd}} = 1.756 \frac{(s + 4.733)(s + 3.45)(s + 2.301)}{(s + 3.515)(s^2 + 7.816s + 23.92)}.$$
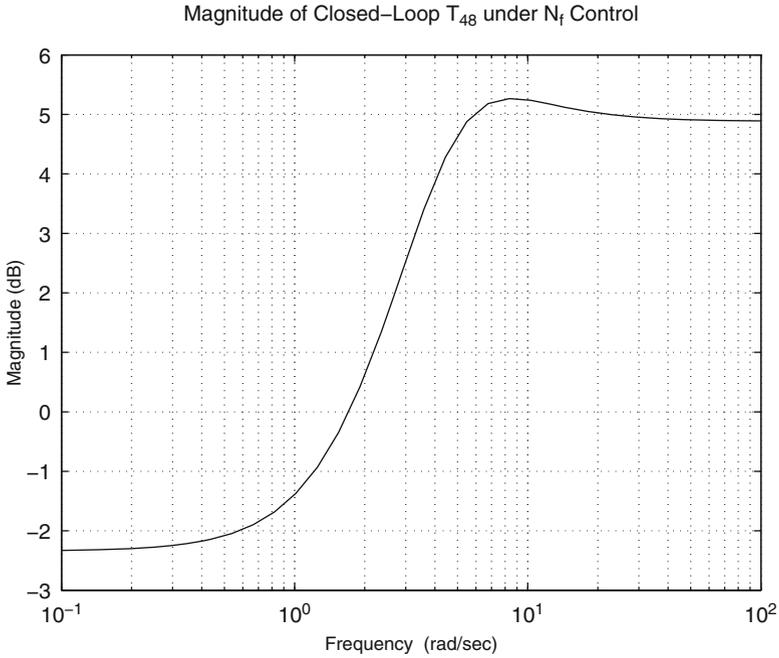
Fig. 3.8 Closed-loop magnitude response of turbine temperature under fan speed control

A frequency response plot for $\frac{\Delta T_{48}}{\Delta N_{\text{f}}.\text{dmd}}$ is shown in Fig. 3.8, revealing large magnitudes near the design crossover region. This predicts significant transient peaking of T_{48}, a highly undesirable feature. A simulation of the loopshaped controller applied to the nonlinear engine model further illustrates the limitations of a fixed-compensator approach. Gain scheduling and built-in limit protection features in CMAPSS-1 were bypassed, replacing them by the single control transfer function. The designed control transfer function $K(s)$ has ΔW_{F} as its output. To deploy the controller to the nonlinear engine, the absolute fuel flow command W_{F} must be calculated. To do this, the integrator in $K(s)$ is factored out as follows:

$$K(s) = K'(s)\frac{1}{s}.$$

If the loopshaped design is used for $K(s)$, the new compensator $K'(s)$ is of the PD type. The output of $K'(s)$ is the derivative of fuel flow rate, $\dot{W}_{\text{F}}$. Thus, the absolute fuel flow command is obtained by integration of $\dot{W}_{\text{F}}$, using the linearization value of W_{F} as initial condition, as shown in Fig. 3.9. Since the control transfer function is driven from the fan speed tracking error, no adjustments are required at the input (the linearization value of N_{f} at the starting flight condition cancels out at the summing node). The simulation corresponds to a TRA demand changing from 0 to 100 degrees as a step, with inlet conditions fixed at FC01 values. Figures 3.10 and 3.11 show that, although the fan speed demand is met with no offset, severe
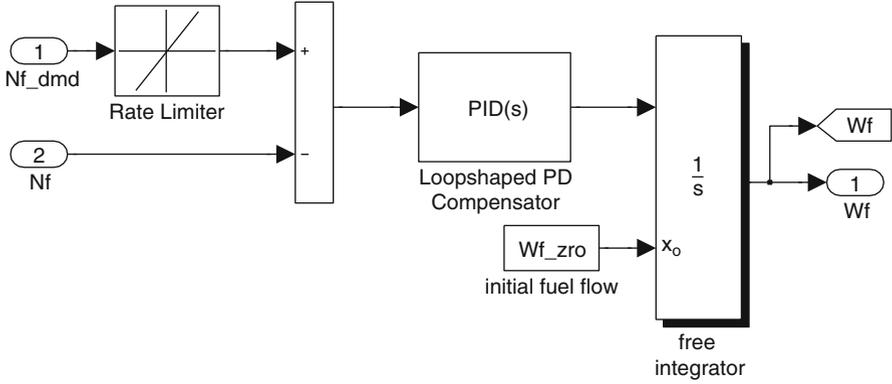
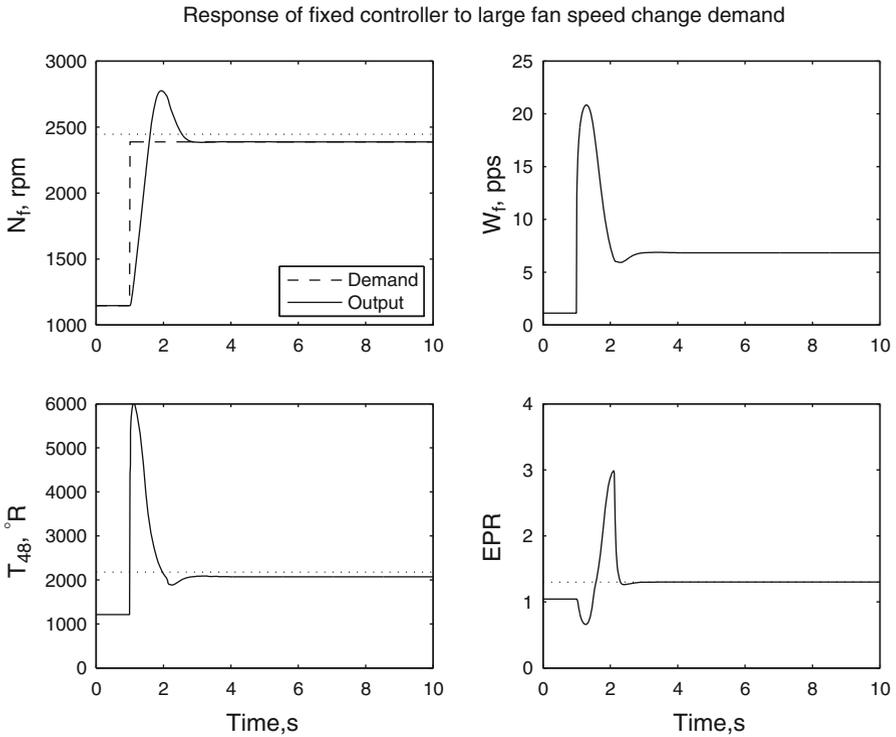**Fig. 3.9** CMAPSS implementation of an incremental linear compensator

Fig. 3.10 Engine response to large fan speed demand with fixed linear compensator

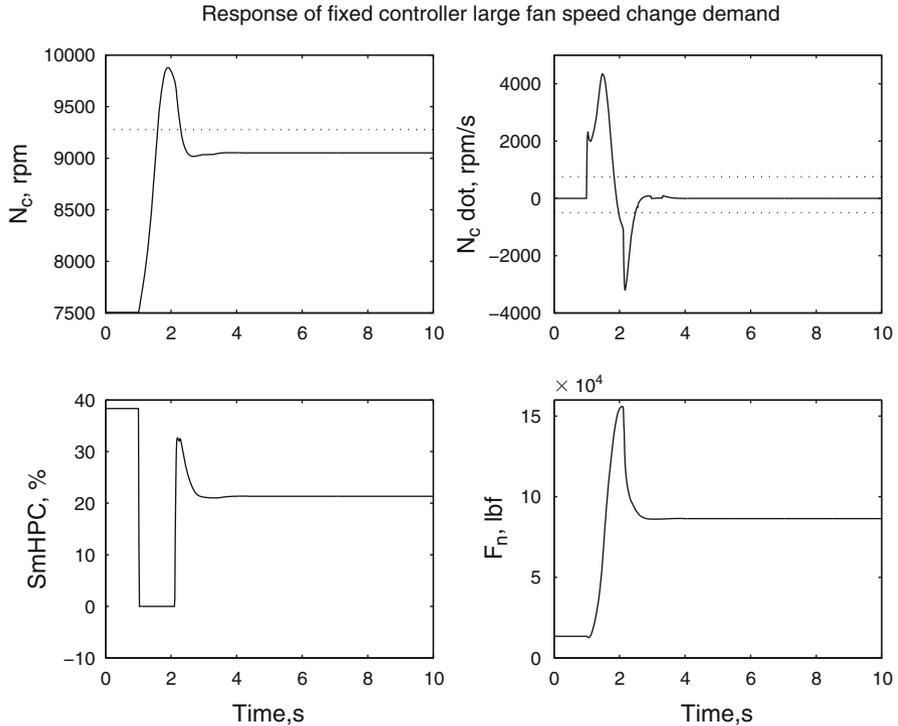Response of fixed controller large fan speed change demand

Fig. 3.11 Engine response to large fan speed demand with fixed linear compensator

transient peaking occurs in every output. Referential limits for N_f, EPR, T_{48}, $\dot{N}_c$, and SmHPC have been represented with dashed lines. All limits are exceeded by large amounts, strongly suggesting that, regardless of synthesis method, no fixed linear regulator can be found that is able to achieve limit protection and adequate fan speed response.

Finally, the CMAPSS simulation data has been represented in a compressor map in Fig. 3.12. The horizontal coordinate is the corrected flow through the HPC, calculated through (1.20), (1.21), and (1.22), with P_t and T_t taken at HPC inlet. The value from (1.22) is then divided by a scaling factor. The vertical coordinate is the HPC pressure ratio from the simulation, also scaled. The scaling factors are applied so that both coordinates are compatible with the scaling used in CMAPSS to store map data (efficiency and speed contours and R-lines). The initial and final steady conditions have been represented on the engine operating line. The trajectory starting from FC14 proceeds toward the stall limit, reaching a zero stall margin condition. The CMAPSS solver will not produce negative stall margin values hence,
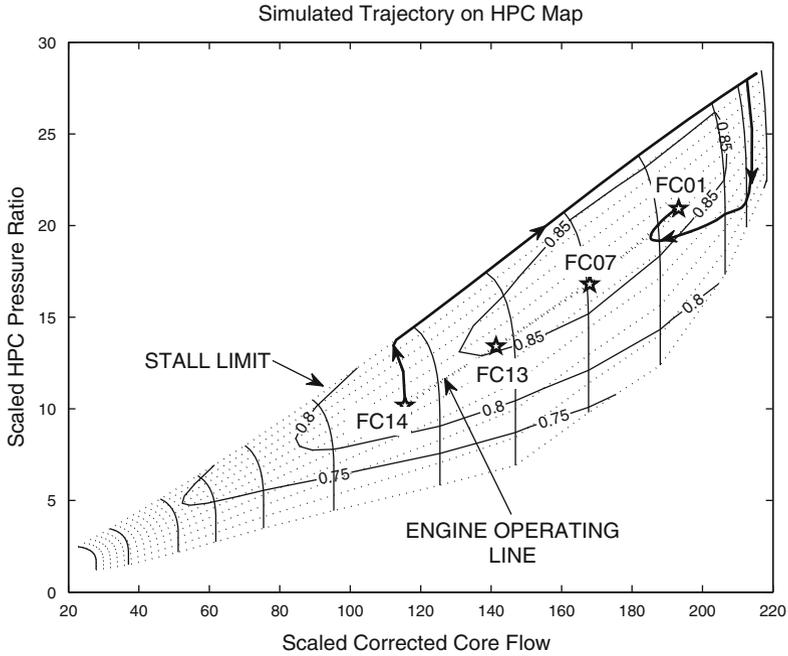
Simulated Trajectory on HPC Map

Fig. 3.12 Simulated trajectory in compressor map coordinates

the trajectory remains on the stall line for some time. It later abandons the stall line and proceeds to the regulation point at FC01. Although the simulated trajectory is inadmissible from a practical standpoint, it exemplifies the challenges associated with control design in the presence of operating limits.