Chapter 15

Solving the Controller Design Problem

In this chapter we describe methods for forming and solving finite-dimensional approximations to the controller design problem. A method based on the parametrization described in chapter 7 yields an inner approximation of the region of achievable specifications in performance space. For some problems, an outer approximation of this region can be found by considering a dual problem. By forming both approximations, the controller design problem can be solved to an arbitrary, and guaranteed, accuracy.

In chapter 3 we argued that many approaches to controller design could be described in terms of a family of design specifications that is parametrized by a performance vector \( a \in \mathbb{R}^L \),

\[ H \text{ satisfies } D_{\text{hard}}, \quad \phi_1(H) \leq a_1, \ldots, \phi_L(H) \leq a_L. \]  

(15.1)

Some of these specifications are unachievable; the designer must choose among the specifications that are achievable. In terms of the performance vectors, the designer must choose an \( a \in A \), where \( A \) denotes the set of performance vectors that correspond to achievable specifications of the form (15.1). We noted in chapter 3 that the actual controller design problem can take several specific forms, e.g., a constrained optimization problem with weighted-sum or weighted-max objective, or a simple feasibility problem.

In chapters 7–11 we found that in many controller design problems, the hard constraint \( D_{\text{hard}} \) is convex (or even affine) and the functionals \( \phi_1, \ldots, \phi_L \) are convex; we refer to these as convex controller design problems. We refer to a controller design problem in which one or more of the functionals is quasiconvex but not convex as a quasiconvex controller design problem. These controller design problems can be considered convex (or quasiconvex) optimization problems over \( \mathcal{H} \); since \( \mathcal{H} \)
has infinite dimension, the algorithms described in the previous chapter cannot be directly applied.

15.1 Ritz Approximations

The Ritz method for solving infinite-dimensional optimization problems consists of solving the problem over larger and larger finite-dimensional subsets. For the controller design problem, the Ritz approximation method is determined by a sequence of \( n_x \times n_w \) transfer matrices

\[
R_0, R_1, R_2, \ldots \in \mathcal{H}. \tag{15.2}
\]

We let

\[
\mathcal{H}_N \triangleq \left\{ R_0 + \sum_{1 \leq i \leq N} x_i R_i \mid x_i \in \mathbb{R}, \ 1 \leq i \leq N \right\}
\]

denote the finite-dimensional affine subset of \( \mathcal{H} \) that is determined by \( R_0 \) and the next \( N \) transfer matrices in the sequence. The \( N \)th Ritz approximation to the family of design specifications \((15.1)\) is then

\[
H \ satisfies \ D_{\text{hard}}, \ \phi_1(H) \leq a_1, \ldots, \phi_L(H) \leq a_L, \ H \in \mathcal{H}_N. \tag{15.3}
\]

The Ritz approximation yields a convex (or quasiconvex) controller design problem, if the original controller problem is convex (or quasiconvex), since it is the original problem with the affine specification \( H \in \mathcal{H}_N \) adjoined.

The \( N \)th Ritz approximation to the controller design problem can be considered a finite-dimensional optimization problem, so the algorithms described in chapter 14 can be applied. With each \( x \in \mathbb{R}^N \) we associate the transfer matrix

\[
H_N(x) \triangleq R_0 + \sum_{1 \leq i \leq N} x_i R_i, \tag{15.4}
\]

with each functional \( \phi_i \) we associate the function \( \phi_i^{(N)} : \mathbb{R}^N \to \mathbb{R} \) given by

\[
\phi_i^{(N)}(x) \triangleq \phi_i(H_N(x)), \tag{15.5}
\]

and we define

\[
D^{(N)} \triangleq \{ x \mid H_N(x) \ satisfies \ D_{\text{hard}} \}. \tag{15.6}
\]

Since the mapping from \( x \in \mathbb{R}^N \) into \( \mathcal{H} \) given by \((15.4)\) is affine, the functions \( \phi_i^{(N)} \) given by \((15.5)\) are convex (or quasiconvex) if the functionals \( \phi_i \) are; similarly the subsets \( D^{(N)} \subseteq \mathbb{R}^N \) are convex (or affine) if the hard constraint \( D_{\text{hard}} \) is. In
section 13.5 we showed how to compute subgradients of the functions \( \phi_i^{(N)} \), given subgradients of the functionals \( \phi_i \).

Let \( \mathcal{A}_N \) denote the set of performance vectors that correspond to achievable specifications for the \( N \)th Ritz approximation (15.3). Then we have

\[ \mathcal{A}_1 \subseteq \cdots \subseteq \mathcal{A}_N \subseteq \cdots \subseteq \mathcal{A}, \]

i.e., the Ritz approximations yield inner or conservative approximations of the region of achievable specifications in performance space.

If the sequence (15.2) is chosen well, and the family of specifications (15.1) is well behaved, then the approximations \( \mathcal{A}_N \) should in some sense converge to \( \mathcal{A} \) as \( N \to \infty \). There are many conditions known that guarantee this convergence; see the Notes and References at the end of this chapter.

We note that the specification \( \mathcal{H}_{\text{slice}} \) of chapter 11 corresponds to the \( N = 2 \) Ritz approximation:

\[ R_0 = H^{(c)}, \quad R_1 = H^{(a)} - H^{(c)}, \quad R_2 = H^{(b)} - H^{(c)}. \]

### 15.1.1 A Specific Ritz Approximation Method

A specific method for forming Ritz approximations is based on the parametrization of closed-loop transfer matrices achievable by stabilizing controllers (see section 7.2.6):

\[ \mathcal{H}_{\text{stable}} = \{ T_1 + T_2 Q T_3 \mid Q \text{ stable} \}. \quad (15.7) \]

We choose a sequence of stable \( n_u \times n_y \) transfer matrices \( Q_1, Q_2, \ldots \) and form

\[ R_0 = T_1, \quad R_k = T_2 Q_k T_3, \quad k = 1, 2, \ldots \quad (15.8) \]

as our Ritz sequence. Then we have \( \mathcal{H}_N \subseteq \mathcal{H}_{\text{stable}} \), i.e., we have automatically taken care of the specification \( \mathcal{H}_{\text{stable}} \).

To each \( x \in \mathbb{R}^N \) there corresponds the controller \( K_N(x) \) that achieves the closed-loop transfer matrix \( H_N(x) \in \mathcal{H}_N \): in the \( N \)th Ritz approximation, we search over a set of controllers that is parametrized by \( x \in \mathbb{R}^N \), in the same way that the family of PID controllers is parametrized by the vector of gains, which is in \( \mathbb{R}^3 \). But the parametrization \( K_N(x) \) has a very special property: it preserves the geometry of the underlying controller design problem. If a design specification or functional is closed-loop convex or affine, so is the resulting constraint on or function of \( x \in \mathbb{R}^N \). This is not true of more general parametrizations of controllers, e.g., the PID controllers. The controller architecture that corresponds to the parametrization \( K_N(x) \) is shown in figure 15.1.
Figure 15.1 The Ritz approximation (15.7–15.8) corresponds to a parametrized controller $K_N(x)$ that consists of two parts: a nominal controller $K_{nom}$, and a stable transfer matrix $Q$ that is a linear combination of the fixed transfer matrices $Q_1, \ldots, Q_N$. See also section 7.3 and figure 7.5.

15.2 An Example with an Analytic Solution

In this section we demonstrate the Ritz method on a problem that has an analytic solution. This allows us to see how closely the solutions of the approximations agree with the exact, known solution.

15.2.1 The Problem and Solution

The example we will study is the standard plant from section 2.4. We consider the RMS actuator effort and RMS regulation functionals described in sections 11.2.3 and 11.3.2:

$$\text{RMS}(y_p) \triangleq \phi_{\text{rms},y_p}(H) = \left( ||H_{12}W_{\text{sensor}}||_2^2 + ||H_{13}W_{\text{proc}}||_2^2 \right)^{1/2}$$

$$\text{RMS}(u) \triangleq \phi_{\text{rms},u}(H) = \left( ||H_{22}W_{\text{sensor}}||_2^2 + ||H_{23}W_{\text{proc}}||_2^2 \right)^{1/2}.$$
15.3 An Example with no Analytic Solution

We consider the specific problem:

\[
\min \phi_{\text{rms},yp}(H), \quad \phi_{\text{rms},yp}(H) \leq 0.1
\]  \hspace{1cm} (15.9)

The solution, \( \phi_{\text{rms},yp}^* = 0.0397 \), can be found by solving an LQG problem with weights determined by the algorithm given in section 14.5.

15.2.2 Four Ritz Approximations

We will demonstrate four different Ritz approximations, by considering two different parametrizations (i.e., \( T_1, T_2, \) and \( T_3 \) in (15.7)) and two different sequences of stable \( Q \)'s.

The parametrizations are given by the formulas in section 7.4 using the two estimated-state-feedback controllers \( K^{(a)} \) and \( K^{(d)} \) from section 2.4 (see the Notes and References for more details). The sequences of stable transfer matrices we consider are

\[
Q_i = \left( \frac{1}{s + 1} \right)^i, \quad \tilde{Q}_i = \left( \frac{4}{s + 4} \right)^i, \quad i = 1, \ldots
\]  \hspace{1cm} (15.10)

We will denote the four resulting Ritz approximations as

\[
(K^{(a)}, Q), \quad (K^{(a)}, \tilde{Q}), \quad (K^{(d)}, Q), \quad (K^{(d)}, \tilde{Q}).
\]  \hspace{1cm} (15.11)

The resulting finite-dimensional Ritz approximations of the problem (15.9) turn out to have a simple form: both the objective and the constraint function are convex quadratic (with linear and constant terms) in \( x \). These problems were solved exactly using a special algorithm for such problems; see the Notes and References at the end of this chapter. The performance of these four approximations is plotted in figure 15.2 along with a dotted line that shows the exact optimum, 0.0397. Figure 15.3 shows the same data on a more detailed scale.

15.3 An Example with no Analytic Solution

We now consider a simple modification to the problem (15.9) considered in the previous section: we add a constraint on the overshoot of the step response from the reference input \( r \) to the plant output \( y_p \), i.e.,

\[
\min \phi_{\text{rms},yp}(H), \quad \phi_{\text{rms},yp}(H) \leq 0.1
\]  \hspace{1cm} (15.12)

Unlike (15.9), no analytic solution to (15.12) is known. For comparison, the optimal design for the problem (15.9) has a step response overshoot of 39.7%.
CHAPTER 15 SOLVING THE CONTROLLER DESIGN PROBLEM

Figure 15.2 The optimum value of the finite-dimensional inner approximation of the optimization problem (15.9) versus the number of terms $N$ for the four different Ritz approximations (15.11). The dotted line shows the exact solution, $\text{RMS}(u) = 0.0397$.

Figure 15.3 Figure 15.2 is re-plotted to show the convergence of the finite-dimensional inner approximations to the exact solution, $\text{RMS}(u) = 0.0397$. 
The same four Ritz approximations (15.11) were formed for the problem (15.12), and the ellipsoid algorithm was used to solve them. The performance of the approximations is plotted in figure 15.4, which the reader should compare to figure 15.2. The minimum objective values for the Ritz approximations appear to be converging to 0.058, whereas without the step response overshoot specification, the minimum objective is 0.0397. We can interpret the difference between these two numbers as the cost of reducing the step response overshoot from 39.7% to 10%.

![Figure 15.4](image)

**Figure 15.4** The optimum value of the finite-dimensional inner approximation of the optimization problem (15.12) versus the number of terms $N$ for the four different Ritz approximations (15.11). No analytic solution to this problem is known. The dotted line shows the optimum value of RMS($u$) without the step response overshoot specification.

### 15.3.1 Ellipsoid Algorithm Performance

The finite-dimensional optimization problems produced by the Ritz approximations of (15.12) are much more substantial than any of the numerical example problems we encountered in chapter 14, which were limited to two variables (so we could plot the progress of the algorithms). It is therefore worthwhile to briefly describe how the ellipsoid algorithms performed on the $N = 20 (K^{(a)}, Q)$ Ritz approximation, as an example of a demanding numerical optimization problem.

The basic ellipsoid algorithm was initialized with $A_1 = 5000I$, so the initial ellipsoid was a sphere with radius 70.7. All iterates were well inside this initial ellipsoid. Moreover, increasing the radius had no effect on the final solution (and, indeed, only a small effect on the total computation time).
The algorithm took 34 iterations to find a feasible point, and 4259 iterations for the maximum relative error to fall below 0.1%. The maximum and actual relative errors versus iteration number are shown in figure 15.5. The relative constraint violation and the normalized objective function value versus iteration number are shown in figure 15.6. From these figures it can be seen that the ellipsoid algorithm produces designs that are within a few percent of optimal within about 1500 iterations.

![Figure 15.5](image)

**Figure 15.5** The ellipsoid algorithm maximum and actual relative errors versus iteration number, $k$, for the solution of the $N = 20$ ($K^{(a)}$, $Q$) Ritz approximation of (15.12). After 34 iterations a feasible point has been found, and after 4259 iterations the objective has been computed to a maximum relative error of 0.1%. Note the similarity to figure 14.15, which shows a similar plot for a much simpler, two variable, problem.

For comparison, we solved the same problem using the ellipsoid algorithm with deep-cuts for both objective and constraint cuts, with the same initial ellipsoid. A few more iterations were required to find a feasible point (53), and somewhat fewer iterations were required to find the optimum to within 0.1% (2680). Its performance is shown in figure 15.7. The objective and constraint function values versus iteration number for the deep-cut ellipsoid algorithm were similar to the basic ellipsoid algorithm.

The number of iterations required to find the optimum to within a guaranteed maximum relative error of 0.1% for each $N$ (for the ($K^{(a)}$, $Q$) Ritz approximation) is shown in figure 15.8 for both the regular and deep-cut ellipsoid algorithms. (For $N < 4$ the step response overshoot constraint was infeasible in the ($K^{(a)}$, $Q$) Ritz approximation.)
15.3 An Example with No Analytic Solution

Figure 15.6 The ellipsoid algorithm constraint and objective functionals versus iteration number, $k$, for the solution of the $N = 20$ $(K^{(e)}, Q)$ Ritz approximation of (15.12). For the constraints, the percentage violation is plotted. For the objective, $\phi_{\text{rms, uv}}$, the percentage difference between the current and final objective value, $\phi_{\text{rms, uv}}^*$, is plotted. It is hard to distinguish the plots for the constraints, but the important point here is the “steady, stochastic” nature of the convergence. Note that within 1500 iterations, designs were obtained with objective values and constraints within a few percent of optimal and feasible, respectively.
Figure 15.7 The deep-cut ellipsoid algorithm maximum and actual relative errors versus iteration number, \( k \), for the solution of the \( N = 20 \) \((K^a, Q)\) Ritz approximation of (15.12). After 53 iterations a feasible point has been found, and after 2620 iterations the objective has been computed to a maximum relative error of 0.1%. Note the similarity to figure 15.5.
15.3 AN EXAMPLE WITH NO ANALYTIC SOLUTION

The number of ellipsoid iterations required to compute upper bounds of RMS($u$) to within 0.1\%, versus the number of terms $N$, is shown for the Ritz approximation $(K^{(a)}, Q)$. The upper curve shows the iterations for the regular ellipsoid algorithm. The lower curve shows the iterations for the deep-cut ellipsoid algorithm, using deep-cuts for both objective and constraint cuts.

Figure 15.8
15.4 An Outer Approximation via Duality

Figure 15.4 suggests that the minimum value of the problem (15.12) is close to 0.058, since several different Ritz approximations appear to be converging to this value, at least over the small range of $N$ plotted. The value 0.058 could reasonably be accepted on this basis alone.

To further strengthen the plausibility of this conclusion, we could appeal to some convergence theorem (one is given in the Notes and References). But even knowing that each of the four curves shown in figure 15.4 converges to the exact value of the problem (15.12), we can only assert with certainty that the optimum value lies between 0.0397 (the true optimum of (15.9)) and 0.0585 (the lowest objective value computed with a Ritz approximation).

This is a problem of stopping criterion, which we discussed in chapter 14 in the context of optimization algorithms; accepting 0.058 as the optimum value of (15.12) corresponds to a (quite reasonable) heuristic stopping criterion. Just as in chapter 14, however, a stopping criterion that is based on a known lower bound may be worth the extra computation involved.

In this section we describe a method for computing lower bounds on the value of (15.12), by forming an appropriate dual problem. Unlike the stopping criteria for the algorithms in chapter 14, which involve little or no additional computation, the lower bound computations that we will describe require the solution of an auxiliary minimum $H_2$ norm problem.

The dual function introduced in section 3.6.2 produces lower bounds on the solution of (15.12), but is not useful here since we cannot exactly evaluate the dual function, except by using the same approximations that we use to approximately solve (15.12). To form a dual problem that we can solve requires some manipulation of the problem (15.12) and a generalization of the dual function described in section 3.6.2. The generalization is easily described informally, but a careful treatment is beyond the scope of this book; see the Notes and References at the end of this chapter.

We replace the RMS actuator effort objective and the RMS regulation constraint in (15.12) with the corresponding variance objective and constraint, i.e., we square the objective and constraint functionals $\phi_{\text{rms.u}}$ and $\phi_{\text{rms.yp}}$. Instead of considering the step response overshoot constraint in (15.12) as a single functional inequality, we will view it as a family of constraints on the step response: one constraint for each $t \geq 0$, that requires the step response at time $t$ not exceed 1.1:

$$\min \phi_{\text{rms.u}}(H)^2$$

$$\phi_{\text{rms.yp}}(H)^2 \leq 0.1^2$$

$$\phi_{\text{step},t}(H) \leq 1.1, \ t \geq 0$$

where $\phi_{\text{step},t}$ is the affine functional that evaluates the step response of the 1,3 entry at time $t$:

$$\phi_{\text{step},t}(H) = s_{13}(t).$$
In this transformed optimization problem, the objective is quadratic and the constraints consist of one quadratic constraint along with a family of affine constraints that is parametrized by $t \in \mathbb{R}_+$. This suggests the following generalized dual functional for the problem (15.13):

$$
\psi(\lambda_u, \lambda_y, \lambda_s) \triangleq \min_{H \in \mathcal{H}} \left( \lambda_u \phi_{\text{rms,\mu}}(H)^2 + \lambda_y \phi_{\text{rms,\gamma}}(H)^2 + \int_0^\infty \lambda_s(t) \phi_{\text{step,\delta}}(H) \, dt \right), \quad (15.14)
$$

where the “weights” now consist of the positive numbers $\lambda_u$ and $\lambda_y$, along with the function $\lambda_s : \mathbb{R}_+ \to \mathbb{R}_+$. So in equation (3.10), we have replaced the weighted sum of (a finite number of) constraint functionals with a weighted integral over the family of constraint functionals that appears in (15.13).

It is easily established that $-\psi$ is a convex functional of $(\lambda_u, \lambda_y, \lambda_s)$ and that whenever $\lambda_y \geq 0$ and $\lambda_s(t) \geq 0$ for all $t \geq 0$, we have

$$
\psi(1, \lambda_y, \lambda_s) - 0.1^2 \lambda_y - \int_0^\infty 1.1 \lambda_s(t) \, dt \leq \alpha_{\text{pri}}
$$

where $\alpha_{\text{pri}}$ is the optimum value of (15.12). Thus, computing $\psi(1, \lambda_y, \lambda_s)$ yields a lower bound on the optimum of (15.12). The convex duality principle (equations (6.11–6.12) of section 6.6) suggests that we actually have

$$
\alpha_{\text{pri}} = \max_{\lambda_y \geq 0, \lambda_s(t) \geq 0} \left( \psi(1, \lambda_y, \lambda_s) - 0.1^2 \lambda_y - \int_0^\infty 1.1 \lambda_s(t) \, dt \right), \quad (15.15)
$$

which in fact is true. So the optimization problem on the right-hand side of (15.15) can be considered a dual of (15.12).

We can compute $\psi(1, \lambda_y, \lambda_s)$, provided we have a state-space realization with impulse response $\lambda_s(t)$. The objective in (15.14) is an LQG objective, with the addition of the integral term, which is an affine functional of $H$. By completing the square, it can be recast as an $H_1$-optimal controller problem, and solved by (an extension of) the method described in section 12.2. Since we can find a minimizer $H^*_\lambda$, for the problem (15.14), we can evaluate a subgradient for $-\psi$:

$$
\phi_{\text{min}}(\lambda_u, \lambda_y, \lambda_s) = -\lambda_u \phi_{\text{rms,\mu}}(H^*_\lambda)^2 - \lambda_y \phi_{\text{rms,\gamma}}(H^*_\lambda)^2 - \int_0^\infty \lambda_s(t) \phi_{\text{step,\delta}}(H^*_\lambda) \, dt
$$

(c.f. section 13.4.8). By applying a Ritz approximation to the infinite-dimensional optimization problem (15.15), we obtain lower bounds on the right-hand, and hence left-hand sides of (15.15).
To demonstrate this, we use two Ritz sequences for \( \lambda \), given by the transfer functions

\[
\Lambda_0 = 0, \quad \Lambda_i(s) = \left( \frac{2}{s+2} \right)^i, \quad \tilde{\Lambda}_i(s) = \left( \frac{4}{s+4} \right)^i, \quad i = 1, \ldots
\]  

(15.16)

We shall denote these two Ritz approximations \( \Lambda, \tilde{\Lambda} \). The solution of the finite-dimensional inner approximation of the dual problem (15.15) is shown in figure 15.9 for various values of \( N \). Each curve gives lower bounds on the solution of (15.12).

**Figure 15.9** The optimum value of the finite-dimensional inner approximation of the dual problem (15.15) versus the number of terms \( N \) for the two different Ritz approximations (15.16). The dotted line shows the optimum value of \( \text{RMS}(u) \) without the step response overshoot specification.

The upper bounds from figure 15.4 and lower bounds from figure 15.9 are shown together in figure 15.10. The exact solution of (15.12) is known to lie between the dashed lines; this band is shown in figure 15.11 on a larger scale for clarity.

The best lower bound on the value of (15.12) is 0.0576, corresponding to the \( N = 20 \{ \tilde{\Lambda} \} \) Ritz approximation of (15.15). The best upper bound on the value of (15.12) is 0.0585, corresponding to the \( N = 20 \{ K^{(s)}, Q \} \) Ritz approximation of (15.12). Thus we can state with certainty that

\[
0.0576 \leq \min \phi_{\text{rms}_{\text{xp}}}(H) \leq 0.1
\]

\[
\phi_{\text{os}}(H_{13}) \leq 0.1
\]

We now know that 0.058 is within 0.0005 \( (i.e., 1\%) \) of the minimum value of the problem (15.12); the plots in figure 15.4 only strongly hint that this is so.
Figure 15.10 The curves from figures 15.4 and 15.9 are shown. The solution of each finite-dimensional inner approximation of (15.12) gives an upper bound of the exact solution. The solution of each finite-dimensional inner approximation of the dual problem (15.15) gives a lower bound of the exact solution. The exact solution is therefore known to lie between the dashed lines.

Figure 15.11 Figure 15.10 is shown in greater detail. The exact solution of (15.12) is known to lie inside the shaded band.
15.5 Some Tradeoff Curves

In the previous three sections we studied two specific optimization problems, and the performance of several approximate solution methods. In this section we consider some related two-parameter families of design specifications and the same approximate solution methods, concentrating on the effect of the approximations on the computed region of achievable specifications in performance space.

15.5.1 Tradeoff for Example with Analytic Solution

We consider the family of design specifications given by

\[ \phi_{\text{rms,yp}}(H) \leq \alpha, \quad \phi_{\text{rms,u}}(H) \leq \beta, \]

for the same example that we have been studying.

Figure 15.12 shows the tradeoff curves for the \((K^{(a)}, Q)\) Ritz approximations of (15.17), for three values of \(N\). The regions above these curves are thus \(A_N\); \(A\) is the region above the solid curve, which is the exact tradeoff curve. This figure makes clear the nomenclature “inner approximation”.

15.5.2 Tradeoff for Example with no Analytic Solution

We now consider the family of design specifications given by

\[ D^{(\alpha, \beta)} : \phi_{\text{rms,yp}}(H) \leq \alpha, \quad \phi_{\text{rms,u}}(H) \leq \beta, \quad \phi_{\text{or}}(H_{1/3}) \leq 0.1. \]

Inner and outer approximations for the tradeoff curve for (15.18) can be computed using Ritz approximations to the primal and dual problems. For example, an \(N = 6\) \((K^{(d)}, Q)\) Ritz approximation to (15.18) shows that the specifications in the top right region in figure 15.13 are achievable. On the other hand, an \(N = 5\) \(A\) Ritz approximation to the dual of (15.18) shows that the specifications in the bottom left region in figure 15.13 are unachievable. Therefore the exact tradeoff curve is known to lie in the shaded region in figure 15.14.

The inner and outer approximations shown in figure 15.14 can be improved by solving larger finite-dimensional approximations to (15.18) and its dual, as shown in figure 15.15.

Figure 15.15 shows the tradeoff curve for (15.17), for comparison. The gap between this curve and the shaded region shows the cost of the additional step response specification. (The reader should compare these curves with those of figure 14.18 in section 14.5, which shows the cost of the additional specification \(\text{RMS}(\hat{y}_p) \leq 0.04\) on the family of design specifications (15.17).)
15.5 SOME TRADEOFF CURVES

The exact tradeoff between $\text{RMS}(y_p)$ and $\text{RMS}(u)$ for the problem (15.17) can be computed using LQG theory. The tradeoff curves computed using three $(K^{(s)}, Q)$ Ritz inner approximations are also shown.

With the specification $\phi_{\text{rms}}(H_{13}) \leq 0.1$, specifications on $\phi_{\text{rms}}(y_p)$ and $\phi_{\text{rms}}(u)$ in the upper shaded region are shown to be achievable by solving an $N = 6$ $(K^{(d)}, Q)$ finite-dimensional approximation of (15.18). Specifications in the lower shaded region are shown to be unachievable by solving an $N = 5$ $\Lambda$ finite-dimensional approximation of the dual of (15.18).
Figure 15.14 From figure 15.13 we can conclude that the exact tradeoff curve lies in the shaded region.

Figure 15.15 With larger finite-dimensional approximations to (15.18) and its dual, the bounds on the achievable limit of performance are substantially improved. The dashed curve is the boundary of the region of achievable performance without the step response overshoot constraint.
Notes and References

Ritz Approximations

The term comes from Rayleigh-Ritz approximations to infinite-dimensional eigenvalue problems; see, e.g., Courant and Hilbert [CH53, p175]. The topic is treated in, e.g., section 3.7 of Daniel [DAN71]. A very clear discussion of Ritz methods, and their convergence properties, appears in sections 8 and 9 of the paper by Levitin and Poljak [LP66].

Proof that the Example Ritz Approximations Converge

It is often possible to prove that a Ritz approximation "works", i.e., that \( A_N \rightarrow A \) (in some sense) as \( N \rightarrow \infty \). As an example, we give a complete proof that for the four Ritz approximations (15.11) of the controller design problem with objectives RMS actuator effort, RMS regulation, and step response overshoot, we have

\[
\bigcup_N A_N \supseteq A \quad \quad \quad \text{(15.19)}
\]

This means that every achievable specification that is not on the boundary (i.e., not Pareto optimal) can be achieved by a Ritz approximation with large enough \( N \).

We first express the problem in terms of the parameter \( Q \) in the free parameter representation of \( D_{\text{stable}} \):

\[
\psi_1(Q) \triangleq \phi_{\text{rms-act}}(T_1 + T_2 Q T_3) = \| \tilde{G}_1 + G_1 Q \|_2,
\]

\[
\psi_2(Q) \triangleq \phi_{\text{rms-reg}}(T_1 + T_2 Q T_3) = \| \tilde{G}_2 + G_2 Q \|_2,
\]

\[
\psi_3(Q) \triangleq \phi_{\text{step}}([1 \ 0](T_1 + T_2 Q T_3)[0 \ 0 \ 1]^T) = \| \tilde{G}_3 + G_3 Q \|_{\text{Hilbert}} - 1,
\]

where the \( \tilde{G}_i \) depend on submatrices of \( T_1 \), and the \( G_i \) depend on the appropriate submatrices of \( T_2 \) and \( T_3 \). (These transfer matrices incorporate the constant power spectral densities of the sensor and process noises, and combine the \( T_2 \) and \( T_3 \) parts since \( Q \) is scalar.) These transfer matrices are stable and rational. We also have \( G_3(0) = 0 \), since \( P_0 \) has a pole at \( s = 0 \).

We have

\[
A = \{ a \ | \ \psi_i(Q) \leq a_i, \ i = 1, 2, 3, \ \text{for some} \ Q \in H_2 \},
\]

\[
A_N = \left\{ a \ \bigg| \ \psi_i(Q) \leq a_i, \ i = 1, 2, 3, \ \text{for some} \ Q = \sum_{i=1}^{N} x_i Q_i \right\}
\]

(\( H_2 \) is the Hilbert space of Laplace transforms of square integrable functions from \( \mathbb{R}_+ \) into \( \mathbb{R} \)).

We now observe that \( \psi_1, \psi_2, \text{and} \psi_3 \) are continuous functionals on \( H_2 \); in fact, there is an \( M < \infty \) such that

\[
|\psi_i(Q) - \psi_i(\tilde{Q})| \leq M ||Q - \tilde{Q}||_2, \quad \text{for} \ i = 1, 2, 3.
\]

(15.20)
This follows from the inequalities
\[
|\psi_1(Q) - \psi_1(\hat{Q})| \leq ||G_1||_\infty||Q - \hat{Q}||_2,
\]
\[
|\psi_2(Q) - \psi_2(\hat{Q})| \leq ||G_2||_\infty||Q - \hat{Q}||_2,
\]
\[
|\psi_3(Q) - \psi_3(\hat{Q})| \leq ||G_3/s||_2||Q - \hat{Q}||_2.
\]
(The first two are obvious; the last uses the general fact that \(||AB||_{\text{op}} \leq ||A||_2||B||_2\), which follows from the Cauchy-Schwarz inequality.)

We now observe that the sequence \((s + \alpha)^{-i}, i = 1, 2, \ldots\), where \(\alpha > 0\), is complete, i.e., has dense span in \(H_2\). In fact, if this sequence is orthonormalized we have the Laplace transforms of the Laguerre functions on \(R_+\).

We can now prove (15.19). Suppose \(a \in \mathcal{A}\) and \(\epsilon > 0\). Since \(a \in \mathcal{A}\), there is a \(Q^* \in H_2\) such that \(\psi_i(Q^*) \leq a_i, i = 1, 2, 3\). Using completeness of the \(Q_i\)'s, find \(N\) and \(x^* \in R^N\) such that
\[
||Q^* - Q^*_N||_2 \leq \epsilon/M,
\]
where
\[
Q^*_N \triangleq \sum_{i=1}^{N} x_i^* Q_i.
\]
By (15.20), \(\psi_i(Q^*_N) \leq a_i + \epsilon, i = 1, 2, 3\). This proves (15.19).

The Example Ritz Approximations

We used the state-space parametrization in section 7.4, with
\[
P_{0}^{\text{std}}(s) = C(sI - A)^{-1}B,
\]
where
\[
A = \begin{bmatrix}
-10 & 0 & 0 \\
1 & 0 & 0 \\
0 & 1 & 0 
\end{bmatrix}, \quad B = \begin{bmatrix}
1 \\
0 \\
0 
\end{bmatrix}, \quad C = \begin{bmatrix}
0 & -1 & 10 
\end{bmatrix}.
\]

The controllers \(K^{(a)}\) and \(K^{(b)}\) are estimated-state-feedback controllers with
\[
K_{\text{est}}^{(a)} = \begin{bmatrix}
5.20000 \\
-2.08000 \\
-0.80800
\end{bmatrix}, \quad K_{\text{est}}^{(b)} = \begin{bmatrix}
-6.00000 & 5.25000 & 2.50000
\end{bmatrix},
\]
\[
K_{\text{est}}^{(d)} = \begin{bmatrix}
0.00000 \\
-3.16228 \\
-1.11150
\end{bmatrix}, \quad K_{\text{est}}^{(d)} = \begin{bmatrix}
1.42758 & 10.29483 & 2.44949
\end{bmatrix}.
\]
Problem with Analytic Solution

With the Ritz approximations, the problem has a single convex quadratic constraint, and a convex quadratic objective, which are found by solving appropriate Lyapunov equations. The resulting optimization problems are solved using a standard method that is described in, e.g., Golub and Van Loan [GL89, p564–566]. Of course an ellipsoid or cutting-plane algorithm could also be used.

Dual Outer Approximation for Linear Controller Design

General duality in infinite-dimensional optimization problems is treated in the book by Rockafellar [Roc74], which also has a complete reference list of other sources covering this material in detail. See also the book by Anderson and Nash [AN87] and the paper by Reiland [Rei80].

As far as we know, the idea of forming finite-dimensional outer approximations to a convex linear controller design problem, by Ritz approximation of an appropriate dual problem, is new.

The method can be applied when the functionals are quadratic (e.g., weighted $H_2$ norms of submatrices of $H$), or involve envelope constraints on time domain responses. We do not know how to form a solvable dual problem of a general convex controller design problem.
Chapter 16

Discussion and Conclusions

We summarize the main points that we have tried to make, and we discuss some applications and extensions of the methods described in this book, as well as some history of the main ideas.

16.1 The Main Points

- *An explicit framework.* A sensible formulation of the controller design problem is possible only by considering simultaneously all of the closed-loop transfer functions of interest, i.e., the closed-loop transfer matrix $H$, which should include every closed-loop transfer function necessary to evaluate a candidate design.

- *Convexity of many specifications.* The set of transfer matrices that meet a design specification often has simple geometry—affine or convex. In many other cases it is possible to form convex inner (conservative) approximations.

- *Effectiveness of convex optimization.* Many controller design problems can be cast as convex optimization problems, and therefore can be “efficiently” solved.

- *Numerical methods for performance limits.* The methods described in this book can be used both to design controllers (via the primal problem) and to find the limits of performance (via the dual problem).

16.2 Control Engineering Revisited

In this section we return to the broader topic of control engineering. Some of the major tasks of control engineering are shown in figure 16.1:
The system to be controlled, along with its sensors and actuators, is modeled as the plant $P$.

Vague goals for the behavior of the closed-loop system are formulated as a set of design specifications (chapters 8–10).

If the plant is LTI and the specifications are closed-loop convex, the resulting feasibility problem can be solved (chapters 13–15).

If the specifications are achievable, the designer will check that the design is satisfactory, perhaps by extensive simulation with a detailed (probably non-linear) model of the system.

![Figure 16.1 A partial flowchart of the control engineer's tasks.](Image)

One design will involve many iterations of the steps shown in figure 16.1. We now discuss some possible design iterations.

**Modifying the Specifications**

The specifications are weakened if they are infeasible, and possibly tightened if they are feasible, as shown in figure 16.2. This iteration may take the form of a search over Pareto optimal designs (chapter 3).

![Figure 16.2 Based on the outcome of the feasibility problem, the designer may decide to modify (e.g., tighten or weaken) some of the specifications.](Image)
Modifying the Control Configuration

Based on the outcome of the feasibility problem, the designer may modify the choice and placement of the sensors and actuators, as shown in figure 16.3. If the specifications are feasible, the designer might remove actuators and sensors to see if the specifications are still feasible; if the specifications are infeasible, the designer may add or relocate actuators and sensors until the specifications become achievable. The value of knowing that a given set of design specifications cannot be achieved with a given configuration should be clear.

![Diagram](image)

**Figure 16.3** Based on the outcome of the feasibility problem, the designer may decide to add or remove sensors or actuators.

These iterations can take a form that is analogous to the iteration described above, in which the specifications are modified. We consider a fixed set of specifications, and a family (which is usually finite) of candidate control configurations. Figure 16.4 shows fourteen possible control configurations, each of which consists of some selection among the two potential actuators $A_1$ and $A_2$ and the three sensors $S_1$, $S_2$, and $S_3$. (These are the configurations that use at least one sensor, and one, but not both, actuators. $A_1$ and $A_2$ might represent two candidate motors for a system that can only accommodate one.) These control configurations are partially ordered by inclusion; for example, $A_1 S_1$ consists of deleting the sensor $S_3$ from the configuration $A_1 S_1 S_3$.

These different control configurations correspond to different plants, and therefore different feasibility problems, some of which may be feasible, and others infeasible. One possible outcome is shown in figure 16.5: nine of the configurations result in the specifications being feasible, and five of the configurations result in the specifications being infeasible. In the iteration described above, the designer could choose among the achievable specifications; here, the designer can choose among the control configurations that result in the design specification being feasible. Continuing the analogy, we might say that $A_1 S_1 S_2$ is a Pareto optimal control configuration, on the boundary between feasibility and infeasibility.
Modifying the Plant Model and Specifications

After choosing an achievable set of design specifications, the design is verified: does the controller, designed on the basis of the LTI model $P$ and the design specifications $\mathcal{D}$, achieve the original goals when connected in the real closed-loop system? If the answer is no, the plant $P$ and design specifications $\mathcal{D}$ have failed to accurately represent the original system and goals, and must be modified, as shown in figure 16.6.

Perhaps some unstated goals were not included in the design specifications. For example, if some critical signal is too big in the closed-loop system, it should be added to the regulated variables signal, and suitable specifications added to $\mathcal{D}$, to constrain the its size.
16.3 Some History of the Main Ideas

16.3.1 Truxal's Closed-Loop Design Method

The idea of first designing the closed-loop system and then determining the controller required to achieve this closed-loop system is at least forty years old. An explicit presentation of such a method appears in Truxal's 1950 Ph.D. thesis [TRU50], and chapter 5 of Truxal's 1955 book, *Automatic Feedback Control System Synthesis*, in which we find [TRU55, p.279]:

Guillemin in 1947 proposed that the synthesis of feedback control systems take the form ...

1. The closed-loop transfer function is determined from the specifications.
2. The corresponding open-loop transfer function is found.
3. The appropriate compensation networks are synthesized.
Truxal cites his Ph.D. thesis and a 1951 article by Aaron [Aar51].

On the difference between classical controller synthesis and the method he proposes, he states ([Tru55, p278–279]):

The word synthesis rigorously implies a logical procedure for the transition from specifications to system. In pure synthesis, the designer is able to take the specifications and in a straightforward path proceed to the final system. In this sense, neither the conventional methods of servo design nor the root locus method is pure synthesis, for in each case the designer attempts to modify and to build up the open-loop system until he has reached a point where the system, after the loop is closed, will be satisfactory.

...[The closed-loop design] approach to the synthesis of closed-loop systems represents a complete change in basic thinking. No longer is the designer working inside the loop and trying to splice things up so that the overall system will do the job required. On the contrary, he is now saying, “I have a certain job that has to be done. I will force the system to do it.”

So Truxal views his closed-loop controller design method as a “more logical synthesis pattern” (p278) than classical methods. He does not extensively justify this view, except to point out the simple relation between the classical error constants and the closed-loop transfer function (p281). (In chapter 8 we saw that the classical error constants are affine functionals of the closed-loop transfer matrix.)

The closed-loop design method is described in the books [NGK57], [RF58, CH7], [Hor63, §5.12], and [FPW90, §5.7] (see also the Notes and References from chapter 7 on the interpolation conditions).

16.3.2 Fegley’s Linear and Quadratic Programming Approach

The observation that some controller design problems can be solved by numerical optimization that involves closed-loop transfer functions is made in a series of papers starting in 1964 by Fegley and colleagues. In [Fec64] and [FH65], Fegley applies linear programming to the closed-loop controller design approach, incorporating such specifications as asymptotic tracking of a specific command signal and an overshoot limit. This method is extended to use quadratic programming in [PF66, BF68]. In [CF68] and [MF71], specifications on RMS values of signals are included. A summary of most of the results of Fegley and his colleagues appears in [FBB71], which includes examples such as a minimum variance design with a step response envelope constraint. This paper has the summary:

Linear and quadratic programming are applicable ... to the design of control systems. The use of linear and quadratic programming frequently represents the easiest approach to an optimal solution and often
makes it possible to impose constraints that could not be imposed in other methods of solution.

So, several important ideas in this book appear in this series of papers by Fegley and colleagues: designing the closed-loop system directly, noting the restrictions placed by the plant on the achievable closed-loop system; expressing performance specifications as closed-loop convex constraints; and using numerical optimization to solve problems that do not have an analytical solution (see the quote above).

Several other important ideas, however, do not appear in this series of papers. Convexity is never mentioned as the property of the problems that makes effective solution possible; linear and quadratic programming are treated as useful “tools” which they “apply” to the controller design problem. The casual reader might conclude that an extension of the method to indefinite (nonconvex) quadratic programming is straightforward, and might allow the designer to incorporate some other useful specifications. This is not the case: numerically solving nonconvex QP’s is vastly more difficult than solving convex QP’s (see section 14.6.1).

Another important idea that does not appear in the early literature on the closed-loop design method is that it can potentially search over all possible LTI controllers, whereas a classical design method (or indeed, a modern state-space method) searches over a restricted (but often adequate) set of LTI controllers. Finally, this early form of the closed-loop design method is restricted to the design of one closed-loop transfer function, for example, from command input to system output.

16.3.3 Q-Parameter Design

The closed-loop design method was first extended to MAMS control systems (i.e., by considering closed-loop transfer matrices instead of a particular transfer function), in a series of papers by Desoer and Chen [DC81A, DC81B, CD82B, CD83] and Gustafson and Desoer [GD83, DG84B, DG84A, GD85]. These papers emphasize the design of controllers, and not the determination that a set of design specifications cannot be achieved by any controller.

In his 1986 Ph. D. thesis, Salcudean [SAL86] uses the parametrization of achievable closed-loop transfer matrices described in chapter 7 to formulate the controller design problem as a constrained convex optimization problem. He describes many of the closed-loop convex specifications we encountered in chapters 8-10, and discusses the importance of convexity. See also the article by Polak and Salcudean [PS89].

The authors of this book and colleagues have developed a program called qdes, which is described in the article [BBB88]. The program accepts input written in a control specification language that allows the user to describe a discrete-time controller design problem in terms of many of the closed-loop convex specifications presented in this book. A simple method is used to approximate the controller design problem as a finite-dimensional linear or quadratic programming problem,
which is then solved. The simple organization and approximations made in qdes make it practical only for small problems. The paper by Oakley and Barratt [OB90] describes the use of qdes to design a controller for a flexible mechanical structure.

16.3.4 FIR Filter Design via Convex Optimization

A relevant parallel development took place in the area of digital signal processing. In about 1969, several researchers observed that many finite impulse response (FIR) filter design problems could be cast as linear programs; see, for example, the articles [CRR69, Rab72] or the books by Oppenheim and Schaefer [OS70, §5.6] and Rabiner and Gold [RG75, ch.3]. In [RG75, §3.39] we even find designs subject to both time and frequency domain specifications:

Quite often one would like to impose simultaneous restrictions on both the time and frequency response of the filter. For example, in the design of lowpass filters, one would often like to limit the step response overshoot or ripple, at the same time maintaining some reasonable control over the frequency response of the filter. Since the step response is a linear function of the impulse response coefficients, a linear program is capable of setting up constraints of the type discussed above.

A recent article on this topic is [OKU88].

Like the early work on the closed-loop design method, convexity is not recognized as the property of the FIR filter design problem that allows efficient solution. Nor is it noted that the method actually computes the global optimum, i.e., if the method fails to design an FIR filter that meets some set of convex specifications, then the specifications cannot be achieved by any FIR filter (of that order).

16.4 Some Extensions

16.4.1 Discrete-Time Plants

Essentially all of the material in this book applies to single-rate discrete-time plants and controllers, provided the obvious changes are made (e.g., redefining stability to mean no poles on or outside the unit disk). For a discrete-time development, there is a natural choice of stable transfer matrices that can be used to form the (analog of the) Ritz sequence (15.8) described in section 15.1:

\[ Q_{ijk}(z) = E_{ij}z^{-(k-1)}, \quad 1 \leq i \leq n_u, \quad 1 \leq j \leq n_y, \quad k = 1, 2, \ldots, \]

(\(E_{ij}\) is the matrix with a unit \(i, j\) entry, and all other entries zero), which corresponds to a delay of \(k - 1\) time steps, from the \(j\)th input of \(Q\) to its \(i\)th output. Thus in the Ritz approximation, the entries of the transfer matrix \(Q\) are polynomials in \(z^{-1}\), i.e., FIR filters. This approach is taken in the program qdes [BBB88].
Many of the results can be extended to multi-rate plant and controllers, i.e., a plant in which different sensor signals are sampled at different rates, or different actuator signals are updated at different rates. A parametrization of stabilizing multi-rate controllers has recently been developed by Meyer [MEY90]; this parametrization uses a transfer matrix $Q(z)$ that ranges over all stable transfer matrices that satisfy some additional convex constraints.

16.4.2 Nonlinear Plants

There are several heuristic methods for designing a nonlinear controller for a nonlinear plant, based on the design of an LTI controller for an LTI plant (or a family of LTI controllers for a family of LTI plants); see the Notes and References for chapter 2. In the Notes and References for chapter 10, we saw a method of designing a nonlinear controller for a plant that has saturating actuators. These methods often work well in practice, but do not qualify as extensions of the methods and ideas described in this book, since they do not consider all possible closed-loop systems that can be achieved. In a few cases, however, stronger results have been obtained.

In [DL82], Desoer and Liu have shown that for stable nonlinear plants, there is a parametrization of stabilizing controllers that is similar to the one described in section 7.2.4, provided a technical condition on $P$ holds (incremental stability).

For unstable nonlinear plants, however, only partial results have been obtained. In [DL83] and [AD84], it is shown how a family of stabilizing controllers can be obtained by first finding one stabilizing controller, and then applying the results of Desoer and Liu mentioned above. But even in the case of an LTI plant and controller, this “two-step compensation” approach can fail to yield all controllers that stabilize the plant. This approach is discussed further in the articles [DL84A, DL84B, DL85].

In a series of papers, Hammer has investigated an extension of the stable factorization theory (see the Notes and References for chapter 7) to nonlinear systems; see [HAM88] and the references therein. Stable factorizations of nonlinear systems are also discussed in Verma [VER88].
Notation and Symbols

Basic Notation

<table>
<thead>
<tr>
<th>Notation</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>{ ⋯ }</td>
<td>Delimiters for sets, and for statement grouping in algorithms in chapter 14.</td>
</tr>
<tr>
<td>( ⋯ )</td>
<td>Delimiters for expressions.</td>
</tr>
<tr>
<td>$f : X \to Y$</td>
<td>A function from the set $X$ into the set $Y$.</td>
</tr>
<tr>
<td>$\emptyset$</td>
<td>The empty set.</td>
</tr>
<tr>
<td>$\land$</td>
<td>Conjunction of predicates; “and”.</td>
</tr>
<tr>
<td>$| \cdot |$</td>
<td>A norm; see page 69. A particular norm is indicated with a mnemonic subscript.</td>
</tr>
<tr>
<td>$\partial \phi(x)$</td>
<td>The subdifferential of the functional $\phi$ at the point $x$; see page 293.</td>
</tr>
<tr>
<td>$\triangleq$</td>
<td>Equals by definition.</td>
</tr>
<tr>
<td>$\approx$</td>
<td>Equals to first order.</td>
</tr>
<tr>
<td>$\approx$</td>
<td>Approximately equal to (used in vague discussions).</td>
</tr>
<tr>
<td>$\preceq$</td>
<td>The inequality holds to first order.</td>
</tr>
<tr>
<td>$\delta X$</td>
<td>A first order change in $X$.</td>
</tr>
<tr>
<td>arg min</td>
<td>A minimizer of the argument. See page 58.</td>
</tr>
<tr>
<td>$\mathbb{C}$</td>
<td>The complex numbers.</td>
</tr>
<tr>
<td>$\mathbb{C}^n$</td>
<td>The vector space of $n$-component complex vectors.</td>
</tr>
<tr>
<td>$\mathbb{C}^{m \times n}$</td>
<td>The vector space of $m \times n$ complex matrices.</td>
</tr>
<tr>
<td>$\mathbb{E} X$</td>
<td>The expected value of the random variable $X$.</td>
</tr>
<tr>
<td>$\Im(z)$</td>
<td>The imaginary part of a complex number $z$.</td>
</tr>
</tbody>
</table>
\( \inf \) The infimum of a function or set. The reader unfamiliar with the notation \( \inf \) can substitute \( \min \) without ill effect.

\( j \) A square root of \(-1\).

\( \limsup \) The asymptotic supremum of a function; see page 72.

\( \Prob(Z) \) The probability of the event \( Z \).

\( \Re(z) \) The real part of a complex number \( z \).

\( \mathbb{R} \) The real numbers.

\( \mathbb{R}_+ \) The nonnegative real numbers.

\( \mathbb{R}^n \) The vector space of \( n \)-component real vectors.

\( \mathbb{R}^{m \times n} \) The vector space of \( m \times n \) real matrices.

\( \sigma_i(M) \) The \( i \)th singular value of a matrix \( M \): the square root of the \( i \)th largest eigenvalue of \( M^*M \).

\( \sigma_{\max}(M) \) The maximum singular value of a matrix \( M \): the square root of the largest eigenvalue of \( M^*M \).

\( \sup \) The supremum of a function or set. The reader unfamiliar with the notation \( \sup \) can substitute \( \max \) without ill effect.

\( \text{Tr} M \) The trace of a matrix \( M \): the sum of its entries on the diagonal.

\( M \geq 0 \) The \( n \times n \) complex matrix \( M \) is positive semidefinite, \( i.e., z^*Mz \geq 0 \) for all \( z \in \mathbb{C}^n \).

\( M > 0 \) The \( n \times n \) complex matrix \( M \) is positive definite, \( i.e., z^*Mz > 0 \) for all nonzero \( z \in \mathbb{C}^n \).

\( \lambda \geq 0 \) The \( n \)-component real-valued vector \( \lambda \) has nonnegative entries, \( i.e., \lambda \in \mathbb{R}_+^n \).

\( M^T \) The transpose of a matrix or transfer matrix \( M \).

\( M^* \) The complex conjugate transpose of a matrix or transfer matrix \( M \).

\( M^{1/2} \) A symmetric square root of a matrix \( M = M^* \geq 0 \), \( i.e., M^{1/2}M^{1/2} = M \).
Global Symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \phi )</td>
<td>A function from the space of transfer matrices to real numbers, ( i.e. ), a functional on ( \mathcal{H} ). A particular function is indicated with a mnemonic subscript.</td>
<td>53</td>
</tr>
<tr>
<td>( \varphi )</td>
<td>The restriction of a functional ( \phi ) to a finite-dimensional domain, ( i.e. ), a function from ( \mathbb{R}^n ) to ( \mathbb{R} ). A particular function is indicated with a mnemonic subscript.</td>
<td>252</td>
</tr>
<tr>
<td>( D )</td>
<td>A design specification: a predicate or boolean function on ( \mathcal{H} ). A particular design specification is indicated with a mnemonic subscript.</td>
<td>47</td>
</tr>
<tr>
<td>( H )</td>
<td>The closed-loop transfer matrix from ( w ) to ( z ).</td>
<td>33</td>
</tr>
<tr>
<td>( H_{ab} )</td>
<td>The closed-loop transfer matrix from the signal ( b ) to the signal ( a ).</td>
<td>32</td>
</tr>
<tr>
<td>( \mathcal{H} )</td>
<td>The set of all ( n_z \times n_w ) transfer matrices. A particular subset of ( \mathcal{H} ) (( i.e. ), a design specification) is indicated with a mnemonic subscript.</td>
<td>48</td>
</tr>
<tr>
<td>( K )</td>
<td>The transfer matrix of the controller.</td>
<td>32</td>
</tr>
<tr>
<td>( L )</td>
<td>The classical loop gain, ( L = P_0 K ).</td>
<td>36</td>
</tr>
<tr>
<td>( n_w )</td>
<td>The number of exogenous inputs, ( i.e. ), the size of ( w ).</td>
<td>26</td>
</tr>
<tr>
<td>( n_u )</td>
<td>The number of actuator inputs, ( i.e. ), the size of ( u ).</td>
<td>26</td>
</tr>
<tr>
<td>( n_z )</td>
<td>The number of regulated variables, ( i.e. ), the size of ( z ).</td>
<td>26</td>
</tr>
<tr>
<td>( n_y )</td>
<td>The number of sensed outputs, ( i.e. ), the size of ( y ).</td>
<td>26</td>
</tr>
<tr>
<td>( P )</td>
<td>The transfer matrix of the plant.</td>
<td>31</td>
</tr>
<tr>
<td>( P_0 )</td>
<td>The transfer matrix of a classical plant, which is usually one part of the plant model ( P ).</td>
<td>34</td>
</tr>
<tr>
<td>( S )</td>
<td>The classical sensitivity transfer function or matrix.</td>
<td>36, 41</td>
</tr>
<tr>
<td>( T )</td>
<td>The classical I/O transfer function or matrix.</td>
<td>36, 41</td>
</tr>
<tr>
<td>( w )</td>
<td>Exogenous input signal vector.</td>
<td>25</td>
</tr>
<tr>
<td>( u )</td>
<td>Actuator input signal vector.</td>
<td>25</td>
</tr>
<tr>
<td>( z )</td>
<td>Regulated output signal vector.</td>
<td>26</td>
</tr>
<tr>
<td>( y )</td>
<td>Sensed output signal vector.</td>
<td>26</td>
</tr>
</tbody>
</table>
## Other Symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Δ</td>
<td>A feedback perturbation.</td>
<td>221</td>
</tr>
<tr>
<td>Δ</td>
<td>A set of feedback perturbations.</td>
<td>221</td>
</tr>
<tr>
<td></td>
<td>The Euclidean norm of a vector $x \in \mathbb{R}^n$ or $x \in \mathbb{C}^n$, i.e., $\sqrt{x^*x}$.</td>
<td>70</td>
</tr>
<tr>
<td>$</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>$</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>$</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>$</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>$</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>$</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>$</td>
<td></td>
<td>H</td>
</tr>
<tr>
<td>$</td>
<td></td>
<td>H</td>
</tr>
<tr>
<td>$</td>
<td></td>
<td>H</td>
</tr>
<tr>
<td>$</td>
<td></td>
<td>H</td>
</tr>
<tr>
<td>$</td>
<td></td>
<td>H</td>
</tr>
<tr>
<td>$</td>
<td></td>
<td>H</td>
</tr>
<tr>
<td>$</td>
<td></td>
<td>H</td>
</tr>
<tr>
<td>$</td>
<td></td>
<td>H</td>
</tr>
<tr>
<td>$</td>
<td></td>
<td>H</td>
</tr>
<tr>
<td>$\mathcal{A}$</td>
<td>The region of achievable specifications in performance space.</td>
<td>139</td>
</tr>
<tr>
<td>$A_p, B_w, B_u, C_x, C_y, D_{zu}, D_{zw}, D_{yu}, D_{yw}$</td>
<td>The matrices in a state-space representation of the plant.</td>
<td>43</td>
</tr>
<tr>
<td>$\text{CF}(u)$</td>
<td>The crest factor of a signal $u$.</td>
<td>91</td>
</tr>
<tr>
<td>$\mathbf{D}(\cdot)$</td>
<td>The dead-zone function.</td>
<td>230</td>
</tr>
<tr>
<td>$I_\gamma(H)$</td>
<td>The $\gamma$-entropy of the transfer function $H$.</td>
<td>113</td>
</tr>
<tr>
<td>$n_{\text{proc}}$</td>
<td>A process noise, often actuator-referred.</td>
<td>35</td>
</tr>
<tr>
<td>$n_{\text{sensor}}$</td>
<td>A sensor noise.</td>
<td>35</td>
</tr>
<tr>
<td>Symbol</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>$F_u(a)$</td>
<td>The amplitude distribution function of the signal $u$.</td>
<td></td>
</tr>
<tr>
<td>$h(t)$</td>
<td>The impulse response of the transfer function $H$ at time $t$.</td>
<td></td>
</tr>
<tr>
<td>$H^{(a)}, H^{(b)}, H^{(c)}, H^{(d)}$</td>
<td>The closed-loop transfer matrices from $w$ to $z$ achieved by the four controllers $K^{(a)}, K^{(b)}, K^{(c)}, K^{(d)}$ in our standard example system.</td>
<td></td>
</tr>
<tr>
<td>$K^{(a)}, K^{(b)}, K^{(c)}, K^{(d)}$</td>
<td>The four controllers in our standard example system.</td>
<td></td>
</tr>
<tr>
<td>$\mathcal{P}$</td>
<td>A perturbed plant set.</td>
<td></td>
</tr>
<tr>
<td>$P_0^{\text{std}}$</td>
<td>The transfer function of our standard example classical plant.</td>
<td></td>
</tr>
<tr>
<td>$p, q$</td>
<td>Auxiliary inputs and outputs used in the perturbation feedback form.</td>
<td></td>
</tr>
<tr>
<td>$s$</td>
<td>Used for both complex frequency $s = \sigma + j\omega$, and the step response of a transfer function or matrix (although not usually in the same equation).</td>
<td></td>
</tr>
<tr>
<td>$\text{Sat}(\cdot)$</td>
<td>The saturation function.</td>
<td></td>
</tr>
<tr>
<td>$\text{sgn}(\cdot)$</td>
<td>The sign function.</td>
<td></td>
</tr>
<tr>
<td>$T_1, T_2, T_3$</td>
<td>Stable transfer matrices used in the free parameter representation of achievable closed-loop transfer matrices.</td>
<td></td>
</tr>
<tr>
<td>$\star$</td>
<td>A submatrix or entry of $H$ not relevant to the current discussion.</td>
<td></td>
</tr>
<tr>
<td>$\phi^*$</td>
<td>The minimum value of the function $\phi$.</td>
<td></td>
</tr>
<tr>
<td>$x^*$</td>
<td>A minimizing argument of the function $\phi$, i.e., $\phi^* = \phi(x^*)$.</td>
<td></td>
</tr>
<tr>
<td>$\text{Tv}(f)$</td>
<td>The total variation of the function $f$.</td>
<td></td>
</tr>
</tbody>
</table>
## List of Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Meaning</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-DOF</td>
<td>One Degree-Of-Freedom</td>
<td>37</td>
</tr>
<tr>
<td>2-DOF</td>
<td>Two Degree-Of-Freedom</td>
<td>39</td>
</tr>
<tr>
<td>ARE</td>
<td>Algebraic Riccati Equation</td>
<td>122</td>
</tr>
<tr>
<td>FIR</td>
<td>Finite Impulse Response</td>
<td>380</td>
</tr>
<tr>
<td>I/O</td>
<td>Input/Output</td>
<td>38</td>
</tr>
<tr>
<td>LQG</td>
<td>Linear Quadratic Gaussian</td>
<td>278</td>
</tr>
<tr>
<td>LQR</td>
<td>Linear Quadratic Regulator</td>
<td>275</td>
</tr>
<tr>
<td>LTI</td>
<td>Linear Time-Invariant</td>
<td>29</td>
</tr>
<tr>
<td>MAMS</td>
<td>Multiple-Actuator, Multiple-Sensor</td>
<td>40</td>
</tr>
<tr>
<td>MIMO</td>
<td>Multiple-Input, Multiple-Output</td>
<td>110</td>
</tr>
<tr>
<td>PID</td>
<td>Proportional plus Integral plus Derivative</td>
<td>5</td>
</tr>
<tr>
<td>QP</td>
<td>Quadratic Program</td>
<td>345</td>
</tr>
<tr>
<td>RMS</td>
<td>Root-Mean-Square</td>
<td>72</td>
</tr>
<tr>
<td>SASS</td>
<td>Single-Actuator, Single-Sensor</td>
<td>34</td>
</tr>
<tr>
<td>SISO</td>
<td>Single-Input, Single-Output</td>
<td>95</td>
</tr>
</tbody>
</table>
Bibliography


398 BIBLIOGRAPHY


## Index

$\epsilon$-relaxed problem, 319, 329

$\|H\|_2$, 96, 111, 120

$\|H\|_\infty$, 99, 112, 118, 121

$H_{\text{disturb}}$, 103, 118, 121

$H_{\text{p}_1,\text{sgn}}^\epsilon$, 111

$H_{\text{p}_2,\text{sgn}}^\epsilon$, 99

$H_{\text{p}_2,\text{sgn}}$, 97, 111, 118, 125

$H_{\text{p}_2,\text{step}}$, 95, 118, 125

$H_{\text{rms},w}$, 95, 110

$H_{\text{rms},\text{sgn}}$, 98, 112

$H_{\text{w},c}$, 97, 125

$u_1$, 81, 88

$u_2$, 80, 88

$u_{\infty}$, 70, 86

$u_{w}$, 74, 87

$u_{\text{in}}$, 84

$u_{\text{rms}}$, 72, 86

1-DOF control system, 37, 283, 286

2-DOF control system, 39

MAMS, 202

### A

Absolute tolerance, 316

Achievable design specifications, 11, 50

Actuator, 1

- boolean, 2
- effort, 12, 37, 49, 190, 256, 275, 281, 355, 376
- effort allocation, 193
- integrated, 18
- multiple, 40
- placement, 3
- referred process noise, 35, 153
- RMS limit, 13, 73, 281
- saturation, 190, 219, 229, 239, 246
- selection, 3, 375
- signal, 25

Adaptive control, 30, 45, 215

Adjoint system, 122

### A

Affine combination, 128

definition for functional, 128

definition for set, 128

Amplitude distribution, 76, 91

Approximation

- finite-dimensional inner, 352
- finite-dimensional outer, 362
- inner, 207, 239, 262, 264
- outer, 267, 318

ARE, 11, 92, 121, 125, 277, 280, 283

arg min, 311

$\alpha$-shifted $H_\infty$ norm, 138

Assumptions, 28, 43

plant, 43

Asymptotic

- decoupling, 178
- disturbance rejection, 254
- tracking, 173, 379

Auto-correlation, 75

vector signal, 86

Average-absolute gain, 98, 111

normal, 74

vector norm, 87

Average power

- admittance load, 82
- resistive load, 72

### B

Bandwidth

- generalized, 189
- limit for robustness, 234

- regulation, 189

- signal, 118, 184

- tracking, 286

Bilinear dependence, 150, 157

Bit complexity, 346

Black box model, 4

Block diagram, 45
Bode plot, 11, 110
Bode’s log sensitivity, 196
MAMS system, 202
Book
  outline, 16
  purpose, 11
Boolean
  actuator, 2
  algebra of subsets, 127
  sensor, 2
Branch-and-bound, 21

C
CACSD, 10, 19
Cancellation
  unstable pole-zero, 152
Chebychev norm, 99
Circle theorem, 245
Classical
  controller design, 5, 11, 18, 244, 378
  optimization paradigm, 57
  regulator, 34, 172
Closed-loop
  controller design, 11, 48, 135, 143, 378
  convex specification, 135
  design, 377
  realizable transfer matrices, 147
  stability, 147, 150
  stability, stable plant, 155
  system, 26
  transfer matrices achieved by stabilizing controllers, 154, 353
  transfer matrix, 33
Command
  decoupling, 178
  following, 183
  input, 28
  interaction, 178
  signal, 28
Comparing norms, 89, 117
Comparison sensitivity, 203
Complementary sensitivity transfer function, 36
Complexity
  bit, 346
  controller, 9
  information based, 346
  quadratic programs, 346
Computing norms, 119
Concave, 139

Conservative design, 210
Constrained optimization, 311, 348, 351, 379
Continuous-time signal, 3, 29
Control configuration, 3, 375
Control engineering, 1
Controllability
  Gramian, 121, 125
Control law, 5
  irrelevant specification, 10
  specification, 7
Controller, 5
  1-DOF, 37, 148, 155, 283, 286
  2-DOF, 39
  adaptive, 30, 45, 215
  assumptions, 29
  complexity, 8, 10
  decentralized, 8
  diagnostic output, 28
  digital, 19
  discrete-time, 2, 380
  estimated-state feedback, 162, 282
  gain-scheduled, 30, 215
  $H_2$-optimal, 279
  $H_{\infty}$ bound, 282
  implementation, 5, 8, 19
  LQG-optimal, 5, 279
  LQR-optimal, 277
  minimum entropy, 282
  nonlinear, 30, 45, 246, 286, 381
  observer-based, 164
  open-loop stable, 268
  order, 10, 150
  parameters, 150
  parametrization, 353
  PID, 5, 14, 21
  PID, 5, 18, 353
  rapid prototyping, 19
  state-feedback, 277
  state-space, 43
  structure, 353
Controller design, 1
  analytic solution, 275
  challenges, 10
  classical, 5, 11, 18, 378
  convex problem, 351
  empirical, 5
  feasibility problem, 51, 335
  goals, 6
  “modern”, 5
  open vs. closed-loop, 135
INDEX

problem, 8, 11, 351
software, 19
state-space, 5
trial and error, 20
Control processor, 1, 9, 19
DSP chips, 19
power, 10
Control system, 1
1-DOF, 37, 283, 286
2-DOF, 39
2-DOF MAMS, 40, 202
architecture, 25
classical regulator, 34, 172
continuous-time, 3
specifications, 12, 47
testing, 6
Control theory
classical, 5
geometric, 45
optimal, 125, 275
state-space, 5
Convex
analysis, 293
combination, 128
controller design problem, 351
definition for functional, 129
definition for set, 128
duality, 139
functional, 128
inner approximation, 207, 239, 262, 264
optimization, 311
outer approximation, 267
set, 128
supporting hyperplane for set, 298
Convexity
midpoint rule, 129
multicriterion optimization, 139
Pareto optimal, 139
performance space, 139
Convex optimization, 11, 311
e-relaxed problem, 319, 329
complexity, 345
cutting-plane algorithm, 313
descent methods, 312
duality, 292
ellipsoid algorithm, 324
linear program, 126, 271, 292, 313, 319, 348, 379
quadratic program, 345, 378, 380
stopping criterion, 315
subgradient algorithm, 348
upper and lower bounds, 315
Convolution, 31
Crest factor, 91
Cutting-plane, 294
algorithm, 313
algorithm initialization, 349
deep-cut, 294

D
Damping ratio, 166
dB, 197
Dead-zone nonlinearity, 229, 246
Decentralized controller, 8
Decibel, 197
Decoupling
asymptotic, 178
exact, 180
Deep-cut, 294
ellipsoid algorithm, 358
Descent
direction, 297, 312
method, 312
Describing function, 220, 246
Design
closed-loop, 377
conservative, 210
FIR filters, 380
Pareto optimal, 55, 281, 337, 369
procedure, 63
trial and error, 20
Design specification, 11, 47
achievable, 14, 50
as sets, 127
boolean, 48
comparing, 49
consistent, 50
families of, 51
feasible, 50
hardness, 51
inconsistent, 50
infeasible, 50
nonconvex, 205
ordering, 49
qualitative, 49
quantitative, 54
robust stability, 212
time domain, 134
total ordering, 54
Diagnostic output, 28, 45
Differential sensitivity, 7, 195
versus robust stability, 210, 233, 244
Digital
controller, 19
signal processing, 19, 380
Directional derivative, 297
Discrete-time
plant, 380
signal, 2, 29
Disturbance, 6, 35
modeling, 187
stochastic, 188, 194
unknown-but-bounded, 190
Dual
functional, 61, 139, 332, 363
functional via LQG, 281
objective functional, 61
optimization problems, 142
problem, 125, 362
Duality, 61, 139, 332, 362
convex optimization, 292
for LQG weight selection, 332
gap, 142
infinite-dimensional, 371

E
Ellipsoid, 121, 324
Ellipsoid algorithm, 324
deep-cut, 358
initialization, 349
Empirical model, 4
Entropy, 136, 282, 291
definition, 112
interpretation, 114
state-space computation, 123
Envelope specification, 177, 193
Error
model reference, 186
nulling paradigm, 46
signal, 34, 46, 69
tracking, 252
Estimated-state feedback, 162, 279, 282
controller, 282
Estimator gain, 162
Exact decoupling, 180
Exogenous input, 25

F
Factorization
polynomial transfer matrix, 169
stable transfer matrix, 169
Feedback
estimated-state, 162, 279, 282
linearization, 30, 45
output, 27
paradigm, 46
perturbation, 221
plant perturbations, 221
random, 114
state, 27
well-posed, 32, 116
Fictitious inputs, 25
Finite-dimensional approximation
inner, 352
outer, 362
Finite element model, 19
FIR, 380
filter design, 380
Fractional perturbation, 202
Frequency domain weights, 81, 188
Functional
affine, 128
bandwidth, 189
convex, 128
definition, 53
dual, 61, 139, 363
equality specification, 131, 137
general response-time, 177
inequality specification, 52, 131
level curve, 250
linear, 296
maximum envelope violation, 177
objective, 54
of a submatrix, 134
overshoot, 173
quasiconcave, 189
quasiconvex, 132, 176, 189, 296
relative overshoot, 173
relative undershoot, 173
response-time, 177
rise time, 175
settling time, 175
subgradient of, 293
sub-level set, 131
underneath, 173
weighted-max, 62, 131, 139, 268
weighted-sum, 59, 131, 139, 261

G
Gain, 94, 115, 231
  average-absolute, 98, 111
  cascade connection, 115
  feedback connection, 116
  generalized margin, 238, 263
  margin, 215, 225, 236, 262
  peak, 111
  scaling, 235
  scheduling, 45
  small gain theorem, 231
  variation, 198
Generalized
  bandwidth, 189
  gain margin, 238
  response time, 189
  stability, 165, 232
Geometric control theory, 45
Global
  optimization, 21, 345, 347
  positioning system, 9, 18
Goal programming, 63
Gramian
  controllability, 121, 125
  observability, 120, 123, 125
G-stability, 165

H
H₂ norm, 276
  subgradient, 301
H₂-optimal
  controller, 279
H∞ norm, 99, 282
  bound and Lyapunov function, 245
  shifted, 101
  subgradient, 303
Half-space, 294
Hamiltonian matrix, 121, 277, 282
Hankel norm, 103, 118, 121
Hardness, 51
History
  closed-loop design, 377
  feedback control, 18, 208
  stability, 169
Hunting, 169

I
I/O specification, 172, 250
Identification, 4, 10, 18, 215
Impulse response, 31, 96, 181, 305, 380
Inequality specification, 100
Infeasible design specification, 50
Information based complexity, 346
Inner approximation, 207, 239, 262, 264
  definition, 207
  finite-dimensional, 352
Input, 25
  actuator, 25
  exogenous, 25
  fictitious, 25
  signal, 25
Input-output
  specification, 172, 250
  transfer function, 38
Integrated
  actuators, 18
  sensors, 9, 18
Interaction of commands, 178
Interactive design, 63
Internal model principle, 168
Internal stability, 150
  free parameter representation, 156
  with stable plant, 155
Interpolation conditions, 155, 168, 284, 287
Intuition
  correct, 118, 140, 287, 362
  false, 89, 108, 117, 379
ITAE norm, 84

J
Jerk norm, 83, 190

L
Laguerre functions, 370
Laplace transform, 30
Level curve, 250
Limit of performance, 12, 20, 139, 208, 347
Linear
  fractional form, 150, 157
  functional, 296
  program, 126, 271, 292, 313, 319, 348, 379
  time-invariant system, 30
  transformation, 81
Linearity consequence, 191, 193
Linearly ordered specifications, 132, 300
Logarithmic sensitivity, 197
MAMS system, 202
Loop gain, 36
LQG, 13, 154, 278, 291
multicriterion optimization, 280
optimal controller, 5, 279
LQG weights
selection, 332, 349
trial and error, 350
tweaking, 281, 335
via dual function, 350
LQR, 5, 275
LTI, 8, 28
Lumped system, 29
Lyapunov equation, 120, 123, 125, 271, 371
Lyapunov function, 125, 245

MAMS system, 40
logarithmic sensitivity, 202
Margin, 196
Matrix
Hamiltonian, 121, 277, 282
Schur form, 123, 277
weight, 89, 278
Maximum absolute step, 118
Maximum value method for weights, 63
M-circle constraint, 244
Memoryless nonlinearity, 231
Midpoint rule, 129
Minimax, 62
Minimizer, 311
half-space where it is not, 294
Model, 4, 10, 18, 215
black box, 4
detailed, 213
empirical, 4
LTI approximation, 45
physical, 4, 10, 215
reference, 186
Modified controller paradigm, 157
for a stable plant, 160
observer-based controller, 162
Multicriterion optimization, 54
convexity, 139
LQG, 280, 332
Multiple
actuator, 40
sensor, 40
Multi-rate plants, 30, 381

N
Neglected
nonlinearities, 219
plant dynamics, 213, 225, 234
Nepers, 197
Nested family of specifications, 132, 178, 300
Noise, 6
white, 96, 106, 110, 193, 275, 278
Nominal plant, 211
Nominal value method for weights, 63
Nonconvex
convex approximation for specification, 207
design specification, 205
specification, 205, 245
Noninferior specification, 55
Nonlinear
controller, 30, 45, 246, 286, 381
plant, 29, 45, 212, 219, 233, 381
Nonlinearity
dead-zone, 229
describing function, 220
memoryless, 231
neglected, 219
Non-minimum phase, 283
Nonparametric plant perturbations, 216
Norm, 93
$\|H\|_1$, 96, 111, 120, 276
$\|H\|_\infty$, 99, 112, 118, 121, 282
$\|H\|_{\text{LTI, gen}}$, 103, 118, 121
$\|H\|_{\text{LTI, gen}}$, 111
$\|H\|_{\text{x, gen}}$, 97
$\|H\|_{\text{LTI, gen}}$, 97, 111, 118, 125
$\|H\|_{\text{LTI, gen}}$, 95, 118, 125
$\|H\|_{\text{LTI, gen}}$, 95, 110
$\|H\|_{\text{LTI, gen}}$, 95, 110
$\|H\|_{\text{LTI, gen}}$, 98, 112
$\|H\|_{\text{LTI, gen}}$, 97, 125
$\|u\|_1$, 81, 88
$\|u\|_2$, 80, 88
amusing, 83
average-absolute value, 74
Chebychev, 99
comparing, 89
comparison of, 117
computation of, 119, 125
definition, 69, 92
false intuition, 89
frequency domain weight, 81
  gains, 94, 115
Hankel norm, 103, 118, 121
inequality specification, 100, 135
ITAE, 84
jerk, 83, 190
maximum absolute step, 118
MIMO, 110
multivariable weight, 88, 115
peak, 70
peak acceleration, 83
peak gain, 97, 111, 118, 125, 291,
  305
peak-step, 95, 125
RMS, 72
RMS gain, 98, 112
RMS noise response, 95, 110
scaling, 88, 115
seminorm, 92
shifted $H_{\infty}$, 101
signal, 69
SISO, 95
snap, crackle and pop, 83
stochastic signals, 75
time domain weights, 83
total absolute area, 81
total energy, 80
using an average response, 94
using a particular response, 93
using worst case response, 94
vector average-absolute value, 87
vector peak, 86
vector RMS, 86
vector signal, 86
vector total area, 88
vector total energy, 88
weight, 88, 115
worst case response, 97, 306
Notation, 30, 383
Nyquist plot, 244

O
Objective functional, 54
dual, 61
weighted-max, 62
weighted-sum, 59, 281
Observability
  Gramian, 120, 123, 125
Observer-based controller, 162, 164
Observer gain, 162
Open-loop
  controller design, 135
equivalent system, 203
Operating condition, 215
Optimal
  control, 125, 275
  controller paradigm, 47
  Pareto, 55, 139, 281, 312, 337, 347,
  369, 374
Optimization
  branch-and-bound, 21
classical paradigm, 57
complexity, 345
constrained, 311
convex, 311
cutting-plane algorithm, 313
descent method, 312, 347
ellipsoid algorithm, 324
feasibility problem, 312
global, 21, 345
linear program, 126, 271, 292, 313,
  319, 348, 379
multicriterion, 54, 65, 280
quadratic program, 345, 378, 380
unconstrained, 311
Order of controller, 11, 150
Outer approximation, 267, 318
definition, 267
finite-dimensional, 362
Output, 26
diagnostic, 28
referred perturbation, 202
referred sensitivity matrix, 41
regulated, 26
sensor, 26
Overshoot, 13, 48, 173
P
$\mathcal{P}$, 211
Parametrization
  stabilizing controllers in state-space,
  162
Pareto optimal, 55, 143, 281, 285, 312,
  337, 347, 369, 374
convexity, 139
interactive search, 63
Parseval’s theorem, 81, 110
PD controller, 5, 14, 21
Peak
  acceleration norm, 83
gain from step response, 97
gain norm, 97, 111, 118, 125, 291
signal norm, 70
step, 95, 125
subgradient of norm, 305
vector signal norm, 86
Performance
limit, 12, 20, 139, 347
loop, 241
specification, 6, 171
Performance space, 52
achievable region, 139
convexity, 139
Perturbation, 7
fractional, 202
nonlinear, 212, 233
output-referred, 202
plant gain, 198
plant phase, 199
singular, 217
step response, 205
time-varying, 212, 233
Perturbed plant set, 211
Phase
angle, 197
margin, 216
unwrapping, 197
variation, 199
Physical
model, 4
units, 88
PID controller, 18, 353
Plant, 26
assumptions, 29, 43
discrete-time, 380
failure modes, 214
gain variation, 198
large perturbation, 210
multi-rate, 381
neglected dynamics, 213, 225, 234
nominal, 211
nonlinear, 29, 45, 212, 219, 233, 381
parameters, 214
perturbation feedback form, 221
perturbation set, 211
phase variation, 199
pole variation, 216, 228, 238
relative uncertainty, 217, 229, 238
small perturbation, 195
standard example, 41, 103, 150, 198,
205, 213, 216, 249, 307, 320,
331, 337, 354
state-space, 43, 164
strictly proper, 43
time-varying, 212, 233
transfer function uncertainty, 216
typical perturbations, 210
Pole
plant pole variation, 223, 238
zero cancellation, 152
Polynomial matrix factorization, 169
Power spectral density, 75
Predicate, 48
Primal-dual problems, 142
Process noise
actuator-referred, 35
Programmable logic controllers, 2, 19
Q
Q-parameters, 162, 169, 353, 379,
381
Quadratic program, 345, 378, 380
Quantization error, 70
Quasiconcave, 189
Quasiconvex functional, 132, 176, 189,
296
relative overshot, undershoot, 175
Quasigradient, 296
settling time, 302
R
Realizability, 147
Real-valued signal, 2
Reference signal, 28, 34
Regulated output, 26
Regulation, 6, 12, 187, 254
bandwidth, 189
RMS limit, 188
Regulator
classical, 34, 172
Relative
degree, 156
overshoot, 173
tolerance, 317
Resource consumption, 81
Response time, 177
functional, 177
generalized, 189
Riccati equation, 92, 122, 277, 280, 283
Rise time, 175
unstable zero, 283
INDEX

Ritz
  approximation, 291, 352, 369
RMS
  gain norm, 98, 112
  noise response, 95, 110
  signal norm, 72
  specification, 281, 332
  vector signal norm, 86
Robustness, 6
  bandwidth limit, 234
  definition for specification, 211
  specification, 7, 210, 262
Robust performance, 239, 246, 266
Robust stability, 212, 264
  classical Nyquist interpretation, 244
  norm-bound specification for, 231
  versus differential sensitivity, 210, 233, 244
  via Lyapunov function, 245
Root locus, 11, 18, 65, 378
Root-mean-square value, 72

S
Saturation, 88, 108, 219, 229, 239, 246
  actuator, 190, 229, 239
Saturator, 220
Scaling, 88, 235
Schur form, 123, 277
Seminorm, 92
Sensible specification, 154, 169, 277, 285, 287
Sensitivity
  comparison, 203
  complementary transfer function, 36
  logarithmic, 197, 260
  matrix, 41
  output-referred matrix, 202
  step response, 205, 261
  transfer function, 36, 196
  versus robustness, 210, 233, 244
Sensor, 1
  boolean, 2
  dynamics, 213
  integrated, 9, 18
  many high quality, 10
  multiple, 40
  noise, 35, 153
  output, 26
  placement, 3
  selection, 3, 375
Set
  affine, 128
  boolean algebra of subsets, 127
  convex, 128
  representing design specification, 127
  supporting hyperplane, 298
Settling time, 175, 250
  quasigradient, 302
Shifted $H_\infty$ norm, 136, 138
Side information, 26, 35, 152, 285, 376
Signal
  amplitude distribution of, 76
  average-absolute value norm, 74
  average-absolute value vector norm, 87
  bandwidth, 118, 184
  comparing norms of, 89
  continuous, 3
  continuous-time, 29
  crest factor, 91
  discrete-time, 3, 29
  ITAE norm, 84
  norm of, 69
  peak norm, 70
  peak vector norm, 86
  real-valued, 2
  RMS norm, 72
  size of, 69
  stochastic norm of, 75
  total area, 81
  total energy, 80
  transient, 72, 80, 92
  unknown-but-bounded, 70, 184
  vector norm of, 86
  vector RMS norm, 86
  vector total area, 88
  vector total energy, 88
Simulation, 6, 10
Singing, 170
Singular
  perturbation, 217, 244
  structured value, 245
  value, 110, 112, 124, 304
  value plot, 110
Size
  of a linear system, 93
  of a signal, 69
Slew rate, 82, 181, 185
  limit, 97
Small gain
  method, 210, 231, 244
method state-space connection, 245
degree, 138, 166, 232, 238, 263

theorem, 117, 231
for a stable plant, 100
free parameter representation, 156
generalized, 165, 232
historical, 169
internal, 150
modified controller paradigm, 157
robust, 212
specification, 136
transfer function, 96, 150, 165
transfer matrix, 110, 150
transfer matrix factorization, 169
using interpolation conditions, 155, 284
via Lyapunov function, 245
via small gain theorem, 117

Stable
transfer matrix, 117

Standard example plant, 41, 103, 150, 188, 205, 213, 216, 240, 307, 320, 331, 337, 354
tradeoff curve, 12, 338, 366

State-estimator gain, 279

State-feedback, 277
gain, 162, 279

State-space
computing norms, 119, 125
controller, 43
parametrization of stabilizing controllers, 162
plant, 43
small gain method connection, 245

Step response, 95
and peak gain, 98
envelope constraint, 177
interaction, 178
overshoot, 48, 173, 355
overshoot subgradient, 301
perturbation, 205
relative overshoot, 173
sensitivity, 205
settling time, 250
slew-rate limit, 181
specification, 134, 172
total variation, 98
undershoot, 173

Stochastic signal norm, 75

Stopping criterion, 313, 362
absolute tolerance, 316
convex optimization, 315
relative tolerance, 317
INDEX

Strictly proper, 43
Structured singular value, 245
Subdifferential
definition, 293
Subgradient, 293
algorithm, 348
computing, 301
definition, 293
$H_2$ norm, 301
$H_{\infty}$ norm, 303
infinite-dimensional, 294
peak gain, 305
quasigradient, 296
step response overshoot, 301
tools for computing, 299
worst case norm, 306
Sub-level set, 131
Supporting hyperplane, 298
System
failure, 45
identification, 4, 10, 19, 215
linear time-invariant, 30
lumped, 29

T
Technology, 9
Theorem of alternatives, 141
Time domain
design specification, 134
weight, 83
Tolerance, 316
feasibility, 319
Total
design specification, 134
weight, 83
Track
asymptotic, 173
bandwidth, 286
error, 37, 182, 252
peak error, 184, 286
RMS error, 183
Tradeoff
curve, 12, 57, 290, 338, 366
interactive search, 63
norm selection, 119
rise time versus undershoot, 283
RMS regulated variables, 13, 280, 338
standard example plant, 12, 338, 366
surface, 57, 281
tangent to surface, 60
tracking bandwidth versus error, 286
Transducer, 18
Transfer function, 30
Chebyshev norm, 99
$G$-stable, 165
maximum magnitude, 99
peak gain, 97
RMS response to white noise, 96
sensitivity, 196
shifted, 101
stability, 96, 150, 165
uncertainty, 216
unstable, 96
Transfer matrix, 30
closed-loop, 33
factorization, 169
singular value plot, 110
singular values, 110
size of, 93
stability, 110, 117, 150
weight, 89
Transient, 72, 80, 92
Trial and error
controller design, 20
LQG weight selection, 350
Tuning, 6
rules, 5, 18
Tw($f$), 98
Tweaking, 63

U
Unconstrained optimization, 311
Undershoot, 173
unstable zero, 283
Unknown-but-bounded
model, 70
signal, 184, 190
Unstable
pole-zero cancellation rule, 152, 168
transfer function, 96
zero and rise time, 283
zero and undershoot, 283

V
Vector signal
autocorrelation, 86
W

W_{contr}, 121
W_{obs}, 120
Weight
- a smart way to tweak, 332
- constant matrix, 89
- diagonal matrix, 89
- for norm, 88
- frequency domain, 81, 188
- H_{\infty}, 100
- informal adjustment, 63
- L_2 norm, 100
- matrix, 89, 276, 278
- maximum value method, 63
- multivariable, 88
- nominal value method, 59, 63
- selection for LQG controller, 332, 349
- time domain, 83
- transfer matrix, 89
- tweaking, 63
- tweaking for LQG, 281, 335
- vector, 59
Weighted-max functional, 62, 131, 139, 268
Weighted-sum functional, 59, 131, 139, 281
Well-posed
- feedback, 32, 116
- problem, 57
White noise, 96, 106, 110, 193, 275, 278
Worst case
- norm, 185
- response norm, 97, 125
- subgradient for norm, 306

Z

Zero
- unstable, 283