Chapter 5

Practical Controller Design

5.1 Introduction

Arguably, one of the biggest impacts of modern control theory is in the area of multivariable controller design. This statement does not aim to de-emphasize the importance of good single-loop controllers nor to suggest that all single-loop problems are easy. It is just that the most dramatic performance improvements and ability to operate in new regimes should be expected from the compensation of dynamic and nonlinear interactions.

Single-loop systems naturally dominate practical applications. This is a consequence of a justifiable philosophy to keep the system simple. The designers often go to great lengths to maintain a low coupling between the controlled variables and the control inputs (manipulated variables). This may impose severe constraints on the physical system itself and its mode of operation. But the benefits are quite apparent. The design of feedback to achieve the prescribed operation is limited to the compensation of relatively simple dynamics and disturbance attenuation. Nonlinear characteristics and interactions are then compensated by an outer loop controller that involves primarily static (DC) maps. In this framework, the performance improvement from a better single-loop controller is often questionable. The gains from a more precise compensation of higher order dynamics can be negated by bandwidth restrictions that arise from interactions and the overall system complexity.

On the other hand, newer processes or operating modes are driven by an economic optimization and tend to emphasize dynamic interactions (e.g., by requesting a bandwidth increase from the system). With the nonlinear control problem still containing several unresolved theoretical or computational issues, modern control theory offers systematic methods to handle interacting linear dynamics. Coupled with an ad-hoc[1] handling of the nonlinearities, this is often enough to achieve significant improvements in system performance. One of its main drawbacks preventing a more wide-spread acceptance is the requirement for a high degree of sophistication and expertise.

In the early linear quadratic methods of the 60's, there was a lot of "art" involved in the fine-tuning of a controller, especially for meeting specific and precise objectives. For example, the Q and R matrices in the Riccati equation have nice and immediate interpretation in terms of penalizing the error energy in some channels more than others. But their effect on channel maching or dynamic behavior is more obscure. The development of the H_∞ approach in the 80's and the advancements in its computational software in the 90's have introduced the ability to simplify the design procedure through the use of fairly sophisticated algorithms. Simplicity here refers to the choices made by the designer. For example, the free parameters now have become transfer function weights that are essentially defined by a corner frequency and a roll-off rate. But it is not the conceivable reduction in the number of user-selected parameters that makes this approach attractive. It is its system-interpretation, the ability to relate the weights to performance and robustness measures and, thus, quantify the interplay between modeling uncertainty and disturbances in a more precise manner. To prevent misleading impressions, it should be emphasized that "hard" designs still

[1] Or "insightful" or "educated" all of them meaning that it is a case-dependent engineering solution and not very systematic.

remain hard, perhaps even worse. And there are (always?) requirements on the controller behavior that are too complicated to formulate or incompatible with the approach. These are usually left to adjust in the controller evaluation stage. Sometimes, integrated high-level design techniques may be too inflexible to achieve that. Nevertheless, in the "normal-complexity" case, the H_∞ approach can automate much of the user interaction and produce good designs quickly.

The typical design of a control system can be decomposed into the following major steps:

1. Specifications set-up
2. System Modeling
3. Controller Design
4. Evaluation and Adjustment
5. Implementation

Depending on the application these steps may be essentially disjoint or interdependent. For example, in aerospace applications the design specifications are prescribed by the system designer or manufacturer so that the all the system components work in the desired manner. Modeling is also performed independently from first principles. The results of these two steps are then handed down to the controls engineers for the design of the controller. Of course, there is always the possibility that the design specifications are not achievable, in which case an iteration is necessary. This design mentality has driven the development of the robust control theory. The controller design evolved naturally to address model reduction, feasibility and controller computation. Iterations with the modeling and specifications steps were viewed as a last resort action since that would imply a major change in the component that could affect the operation of the entire system.

As the modern control theory matured in the 90's, it became a viable candidate for designing control systems for other industrial applications, e.g., semiconductor manufacturing, chemical process control, automotive processes. Several new issued started arising; the suitability of existing models for controller design and even their availability can no longer be taken for granted. Specifications are rarely given in a concrete definition, but they take the form of a loose optimization (the best possible controller...). A basic reason for such a direction is in the economics of industrial processes where the fundamental models are too expensive to develop or simply unavailable at the time the controller is to be designed. In the same vein, expertise in control systems may not be widely available during the process development phase and any available models do not necessarily capture control-related characteristics of the process. Consequently, controls engineers are often responsible for the complete design and implementation of the control system, including the understanding of its impact on the operation of the process. Integration of the controller design steps is now becoming more important than the controller computation itself (the latter being largely a resolved issue). The same issues arise when the control problem is often seen from a retrofitting point of view. Here the process needs to be improved through a redesign of an existing control system, already in operation. In addition to the possible lack of models or specifications, an added difficulty comes in the form of design expediency.

In the following we discuss the essense of an integrated controller design procedure that aims to handle "most" but not "all" cases. (Universal controllers and universal procedures go hand-in-hand with perpetual motion machines; in practice we would be happy if, for a given class of problems, we can adjust the procedure so that it produces a successful design, almost always.) The process modeling is performed from input-output data with a system identification approach. The controller design employs classical loop-shaping principles and its computation relies of tools from the H_∞ theory. In the final implementation, the controller is augmented by a simple observer-based anti-windup scheme to handle the ubiquitous input saturation constraints. It should be emphasized that the methods used in each step are not the only ones possible or even available. The basic criterion for their selection is compatibility that allows the integration of the identification and controller design steps. For example, the objective of the identification gives rise to a specific uncertainty structure; this, in turn, translates into constraints that define the controller design objectives. The key analytical tool that provides the theoretical support and the link between the two steps is the so-called Small Gain Theorem, presented in the next section.

5.2 The Small Gain Theorem

This fundamental result was developed independently by Zames and Sandberg in the mid-60's. In a functional analytic framework it addresses the stability problem of the feedback interconnection of two operators (systems). While deceivingly simple in its statement, it establishes a rigorous abstract formulation of the problem. In the 80's it became the cornerstone of the robust control theory, providing a concrete multivariable generalization of classical robustness measures (gain/phase margin).

The small gain theorem applies to stable operators between two Banach spaces with finite gain. Recall that for an operator $G : X \mapsto Y$, the gain of G is defined as

$$\gamma_{X,Y}(G) = \inf\{\gamma | \exists \beta : \|G[x]\|_Y \leq \gamma \|x\|_X + \beta, \ \forall x \in X\}$$

For the special case where X, Y are the spaces of finite energy vector-valued functions (L_2) and G is a stable linear time invariant system, $\gamma_2(G)$ is equal to the supremum of the maximum singular value of $G(jw)$, over $w \in \mathbf{R}$. In this case, the corresponding transfer function $G(s)$ is analytic in the right-half plane; the space of these transfer functions is referred to as H_∞; it is a normed space with norm $\|G\|_{H_\infty} = \sup_w \bar{\sigma}[G(jw)]$. The notation $\bar{\sigma}$ is used for the maximum singular value of a matrix; moreover, the H_∞ norm notation is often simplified to $\|G\|_\infty$. Observe that $\gamma_2(G)$ denotes the induced L_2-to-L_2 gain of the general operator G while $\|G\|_\infty$ is the infinity-norm of the transfer function (matrix) $G(s)$. It just happens that for linear time invariant systems the two are equal.

5.2.1 Theorem: *(Small Gain Theorem) Consider the following interconnection of linear time invariant systems, each mapping L_2 to L_2 with appropriate vector dimensions.*

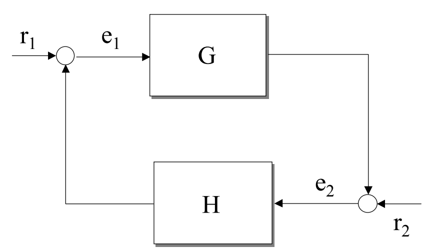

Figure 5.1: The feedback system for the Small Gain Theorem.

Suppose that $\gamma_2(H)\gamma_2(G) < 1$. Then the closed-loop system in Fig. 5.1 is L_2 stable and

$$\begin{aligned}
\|e_1\|_2 &\leq \frac{1}{1-\gamma_2(H)\gamma_2(G)}[\|r_1\|_2 + \gamma_2(H)\|r_2\|_2 + \gamma_2(H)\beta_G + \beta_H] \\
\|e_2\|_2 &\leq \frac{1}{1-\gamma_2(H)\gamma_2(G)}[\|r_2\|_2 + \gamma_2(G)\|r_1\|_2 + \gamma_2(G)\beta_H + \beta_G]
\end{aligned}$$

where β_H, β_G are constants. $\nabla\nabla$

Proof: (Outline) Successive use the triangle inequality of norms and the definition of the gain yields

$$\begin{aligned}
\|e_1\|_2 &\leq \|r_1 + H[r_2 + G[e_1]]\|_2 \\
&\leq \|r_1\|_2 + \gamma_2(H)(\|r_2\|_2 + \|G[e_1]\|_2) + \beta_H \\
&\leq \|r_1\|_2 + \gamma_2(H)\|r_2\|_2 + \gamma_2(H)\gamma_2(G)\|e_1\|_2 + \beta_H + \gamma_2(H)\beta_G
\end{aligned}$$

from which the first inequality follows (similarly for the second inequality). These inequalities show that all internal signals in the loop have finite energy for any finite energy external inputs; hence the loop is L_2 stable.

The above derivation is the heart of the theorem and provides the correct insight; unfortunately, it is not rigorous. A precise statement would involve the –tacitly assumed– causality of the operators and the so-called extended L_2 spaces. Still, the existence of solutions for the loop signals cannot be guaranteed and should be established separately. For the latter, tools from the theory of ordinary differential equations can be used.

A different, more restrictive version of the small gain theorem uses the notion of the incremental gain of an operator (same as the gain but for the difference $x_1 - x_2$ of any two signals in the domain). This version also guarantees the existence of solutions provided that the loop signals exist for one set of external inputs. For linear systems, the incremental gain is equal to the gain of the operator and the incremental version can be used without any increase in conservatism. □□

The small gain theorem is applicable to general input/output spaces and operators. Even though for nonlinear systems the finite gain condition can be quite restrictive, the results establish the well-posedness of the robustness problem in practical applications.

The small gain condition is only sufficient. It is also necessary if one of the two operators is arbitrary, constrained only by a gain bound. For example, consider the problem where G is a linear time invariant system and H is such that $\gamma_2(H) \leq 1$ but otherwise arbitrary. Then, a necessary and sufficient condition for the stability of their feedback loop for any H is $\gamma_2(G) < 1$. The necessity here means that if $\gamma_2(G) \geq 1$ then a stable operator H can be constructed such that $\gamma_2(H) \leq 1$ and the feedback loop is unstable.

It should be made clear that the usefulness of the small gain theorem is not in establishing nominal stability for the plant-controller loop. The reason is that, much like norms, it does not preserve any directionality or phase information. However, as a robustness analysis tool, it is indispensable. A proper framework for its use is seen in the following example. Suppose that the plant is modeled as a nominal transfer function $P_0(s)$ and an output multiplicative uncertainty $\Delta(s)$ such that $\|\Delta\|_\infty \leq \mu$ and the "true" plant is $(I + \Delta)P_0(s)$. A controller is then designed for P_0 and we ask if the perturbed closed-loop is stable for all possible Δ. To solve this problem, let $T(s)$ denote the complementary sensitivity of $C(s)$ and $P_0(s)$, that is $T(s) = [I + P_0C(s)]^{-1}P_0C(s)$. After some straightforward calculations, the perturbed closed loop can be written in a nominal-plus-uncertainty form that fits the small gain theorem framework; more specifically, in Fig. 5.1, $G(s) = T(s)$ and $H(s) = \Delta(s)$. Thus, we conclude that the closed loop is robustly stable if $\|T\|_\infty < 1/\mu$.

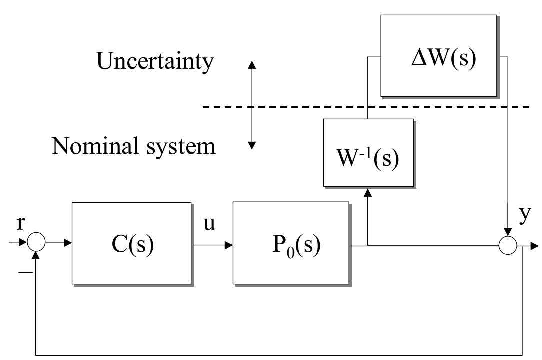

Figure 5.2: Introducing frequency-dependent weights in the Small Gain Theorem.

This example also shows the main deficiency in the small gain theorem. Typically, the multiplicative uncertainty magnitude is small at low frequencies and large at high frequencies. A straight application of the theorem would impose the unreasonable constraint that $\bar{\sigma}[T(jw)] < 1/\mu$ at all frequencies. A standard method to alleviate this problem is to use a frequency dependent weight to "whiten" the uncertainty. (White here means that the uncertainty magnitude can achieve its maximum at any frequency; ideally such a Δ would be a scaled all-pass transfer function/matrix.) Thus, we assume that Δ is such that $\|\Delta W(s)\|_\infty \leq 1$ where $W(s)$ is a stable transfer function/matrix with a stable inverse (minimum phase). Then, writing the perturbed plant as $[I + \Delta W W^{-1}]^{-1}P_0$, the small gain theorem yields the robust stability condition

$\|W^{-1}T\|_\infty < 1$. This weighting procedure is depicted in Fig. 5.2.

A similar technique is used to reduce the small gain conservatism to structured or directional perturbations. For example, suppose that Δ has a block-diagonal structure $\Delta = \text{diag}(\Delta_1, \Delta_2)$. Let $D = diag(w_1 I_1, w_2 I_2)$ be a stable minimum phase transfer matrix whose diagonal blocks are scalar-times-identity and I_1, I_2 have the same size as Δ_1, Δ_2. Introducing the identity maps DD^{-1} and $D^{-1}D$ in the paths leading to and coming from the uncertainty, the small gain theorem yields the robust stability condition $\|D^{-1}TD\|_\infty \|D^{-1}\Delta D\|_\infty < 1$. However, by its construction, D commutes with Δ and the condition becomes $\|D^{-1}TD\|_\infty \|\Delta\|_\infty < 1$. Further, since the result holds for any D (stable minimum phase) we arrive at the condition

$$\inf_D \|D^{-1}TD\|_\infty < 1/\|\Delta\|_\infty$$

This procedure, often referred to as "D-scale optimization," can be implemented by a reasonably efficient numerical algorithm with convergence guarantees. While it remains only a sufficient condition, numerical studies have shown it to be fairly tight. It forms the computational back-bone of the so-called μ-analysis and synthesis.

5.3 Modeling via System Identification

A part of the new demands on the control systems design is the modeling from input-output data. While this is hardly a new topic, it raises certain compatibility questions between the modeling approach and the controller design methodology. Many of the older existing results have been derived with a stochastic formulation to address stochastic modeling problems. More recently, there has been a renewed interest on the subject from the point of view of control-oriented system identification. This includes modeling that is useful for the design of controllers as well as the estimation of gain bounds for the uncertainty.

In Chapter 2 we introduced a system identification approach that employs a simple least-squares estimator. The identification error induces a particular uncertainty structure that is different from the usual additive or multiplicative uncertainty. It is often referred to as "coprime factor uncertainty." Even though it is more complicated, it possesses some important conceptual advantages in the context of feedback control. In the rest of this section, we discuss its generalization to the multivariable case.

The selected system identification method relies on a least squares parameter estimation algorithm to obtain parameter estimates for a linear model that describes the process locally around an operating point. To obtain a dynamical model of the plant, a standard state-space description is considered, i.e.,

$$\dot{x} = Ax + Bu \; ; \quad y = Cx + Du$$

For multiple-input, single-output systems and under an observability assumption, the model can be written as

$$\dot{x} = Ax + Bu + Ly - Ly \; ; \quad y = Cx + Du$$

where $A - LC$ is Hurwitz with prescribed eigenvalues. After a similarity transformation we obtain

$$\dot{x} = Fx + \theta_1 u + \theta_2 y \; ; \quad y = qx + \theta_3 u$$

where F and q are selected a priori so that F is Hurwitz, (F, q) is a completely observable pair, and $\theta_1, \theta_2, \theta_3$ are adjustable parameters. The usefulness of this description is that it can be readily converted into a linear model form, which is convenient for parameter estimation, that is, $y = w^\top \Theta$. Here, Θ is a vector containing all the adjustable parameters (elements of $\theta_1, \theta_2, \theta_3$) as well as the initial conditions $x(0)$. The latter term, often ignored, has been found to have an appreciable impact, especially for short data sets that begin on a transient. The regressor vector w contains the signals $(sI - F^\top)^{-1} q^\top u$, $(sI - F^\top)^{-1} q^\top y$, u, and $(sI - F^\top)^{-1} q^\top$, where the last term corresponds to the unknown initial conditions. After generating the regressor vector, Θ can be determined in a least squares sense by minimizing the estimation error $\|y - w^\top \Theta\|_2$. The above description is repeated for each output and the resulting state-space model is concatenated to produce the overall model of the system. While this approach may result in a non-minimal model, with a proper selection of the model orders (dimensions of F), a model reduction is rarely necessary.

Emphasis of the model accuracy around the crossover frequency is, of course, crucial and should be reflected by the proper selection of the excitation sequence (typically a random binary sequence) and the identification design parameters (input signal, prefilters, etc.). On the other hand, once a model becomes available, it is important to compute uncertainty estimates that are suitable for the controller design technique used. Furthermore, these estimates –although approximate– should be able to detect a possible infeasibility of the controller design problem with the given closed-loop performance objectives.

Various model error structures have been used in control systems design for describing the uncertainty in a manner consistent with robust control theory. In a typical uncertainty estimation approach from data, the model-data mismatch is described by a multiplicative uncertainty, imposing a constraint on the closed-loop bandwidth (more precisely, on the complementary sensitivity).

While conceptually simple, the multiplicative (or additive) description of uncertainty is not entirely consistent with the above identification scheme. Instead, the minimized error corresponds to the contribution of coprime factor uncertainty. For this, the above parametrization of the system can be written as:

$$y = D_p^{-1}[N_p[u] + e]$$

where e is the residual estimation error and D_p, N_p are stable and proper systems determined by F, q and Θ. In this formulation, the coprime factor uncertainty arises naturally by attributing the error to contributions from the input and the output, i.e., $e = \Delta_N[u] + \Delta_D[y]$ (see Fig. 5.3).

The main advantages of the coprime factor uncertainty description lie in its handling of low-frequency perturbations and perturbations that can change the location and number of unstable modes. This is desirable, and often crucial in the case where the model has large low-frequency gains (near-integral action) with respect to the intended closed-loop crossover frequency. Usually, such models also exhibit large low-frequency uncertainty, due to disturbances and the length of the identification experiment. It can be argued that this uncertainty can be reduced by performing a longer experiment. This, however, would be undesirable in retrofitting applications; furthermore, it is unnecessary for the controller design since that information is well-below the intended loop bandwidth.

System identification issues in the context of coprime factor uncertainty have been considered in the literature. A difficulty with this formulation arises from the fact that the correlation between the plant input and output prohibits the estimation of separate bounds for the two uncertainty components from input-output data. To alleviate this problem, we adopt an unfalsification approach That is *"we seek to find a bound for the most favorable uncertainty that is required to describe the residual error."* Of course, the interpretation of such bounds in the controller design is also modified. Instead of sufficient condition for stability, we now have the pseudo-necessary conditions for instability. That is, loosely speaking, if the controller design violates the given bounds, then it is likely that the closed-loop system will be unstable.

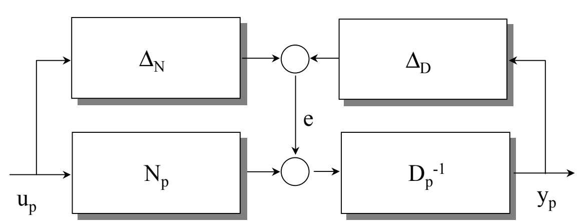

Figure 5.3: Structure of Identification Uncertainty

Guided by the formulation of a standard loop-shaping problem, it is convenient express the uncertainty estimates as weights for the loop sensitivity ($S = [I + PC]^{-1}$) and complementary sensitivity ($T = PC[I + PC]^{-1}$) functions.[2] This is compatible with the H_∞ problem: *Find a stabilizing controller C such that*

$$\min_C \gamma : \bar{\sigma} \begin{bmatrix} W_3 T \\ W_1 S \end{bmatrix} < \gamma \tag{5.3.1}$$

[2] $P = D_p^{-1} N_p$ denotes the nominal plant and C the controller.

Here, $\bar{\sigma}$ denotes the maximum singular value, and W_1, W_3 are weights defining the target sensitivity functions (or, target loop-shape as $W_1 W_3^{-1}$). This form of the H_∞ design draws considerable insight from classical loop-shaping principles, while fundamental feedback limitations are easily observed (e.g., $S + T = I$).

In this setup, the small gain theorem with diagonal scaling can be used to develop a condition for closed-loop stability in the presence of the perturbations Δ_N, Δ_D. Performing the optimization with respect to the diagonal scales and the following condition is obtained

$$\bar{\sigma}[CSD_p^{-1}]\bar{\sigma}[\Delta_N] + \bar{\sigma}[SD_p^{-1}]\bar{\sigma}[\Delta_D] < 1 \tag{5.3.2}$$

For square systems and using $CS = P^{-1}T$, we may pose the uncertainty bound estimation as the following, frequency domain, constrained optimization problem:

$$\begin{aligned} \min_{\delta_1,\delta_2} \quad & \bar{\sigma}[P^{-1}TD_p^{-1}]\delta_1 + \bar{\sigma}[SD_p^{-1}]\delta_2 \\ \text{s.t.} \quad & \Delta_N[u] + \Delta_D[y] = e \\ & \bar{\sigma}[\Delta_N] \le \delta_1 \ , \quad \bar{\sigma}[\Delta_D] \le \delta_2 \end{aligned} \tag{5.3.3}$$

A simple suboptimal solution of (5.3.3) can be derived under the restriction $\Delta_N[u] \perp \Delta_D[y]$:

$$[\delta_1, \ \delta_2] = \begin{cases} [|e|/|u|, \ 0] & \text{if } \ell < 1 \\ [0, \ |e|/|y|] & \text{if } \ell > 1 \end{cases} \qquad \ell \triangleq \frac{|y|}{|u|}\frac{\bar{\sigma}[P^{-1}]\bar{\sigma}[T]}{\bar{\sigma}[S]} \tag{5.3.4}$$

This computation is similar to the optimal distance in the gap metric except that a specific decomposition is used and that right half-plane limitations are ignored (this is acceptable under the unfalsification approach).

Although this optimization problem depends on the actual controller, a suboptimal solution can be computed by replacing T and S with the respective target values. Then, the uncertainty bounds are attractive as they depend only on the desired loop properties and not the controller itself. In addition, the various quantities have simple frequency domain definition and can be readily computed via FFT's or other spectral methods.

It should be emphasized that in this approach, the uncertainty bound estimates are a function of the target loop and they are both determined in one step. In addition to satisfying fundamental limitation constraints (e.g., right-half plane poles and zeros), the only essential requirement for selecting the target T and S is that they yield a robust stability condition less than one. At first glance, this may seem as a tedious iterative process. For the typical control objectives, however, a simple analysis of the optimization problem indicates that the high-frequency component of the estimation error is attributed to Δ_N and the low-frequency component to Δ_D. Based on this observation, it is straightforward to determine the high- and low-bandwidth constraints and loop roll-off rates. The final outcome of this procedure is target sensitivities and an "optimal" uncertainty decomposition. After designing the controller, a more precise estimate of the robust stability condition as well as an estimate of the closed-loop effective multiplicative uncertainty (as a robust performance indicator) can be computed by minimizing (5.3.2) with the actual T and S. As long as the controller results in a loop reasonably close to the target, these estimates will remain roughly unchanged. For more difficult problems, however, a controller design iteration may be required.

It is also worthwhile to mention that there is no loss of information when the uncertainty is split. Simply the energy of the error is distributed to the two uncertainty blocks. Consequently, if the actual design does not match the target loop, the resulting evaluation of the robust stability condition will simply be suboptimal and more conservative. Based on practical experience, there is rarely a need to iterate the uncertainty decomposition step as long as the designed closed-loop sensitivities are close to their targets.

This analysis provides only estimates of the uncertainty bound and closed-loop stability cannot be guaranteed in a strict sense. However, practical experience indicates that there is a very strong correlation between these bounds and successful controller designs. Furthermore, it is implicitly assumed that the source of the estimation error is unmodeled dynamics. If part of the error is due to bounded noise and/or disturbances, this would make the conditions conservative. On the other hand, such an "all-inclusive" computation has the following desirable by-product: When the identification experiment is performed under normal operation, the computed uncertainty bounds reflect correctly the amount of sensitivity reduction required to attenuate the typical low-frequency disturbances entering the loop.

5.4 Controller Design

Using a (sensitivity) loop-shaping approach the uncertainty bounds obtained in the system identification step can be used to define simple sensitivity and complementary sensitivity weights. These weights are chosen to maximize the disturbance attenuation properties without violating the constraints imposed by the uncertainty estimates. An H_∞ approach was selected for the controller computations because it minimizes the weight selection iterations for achieving a target loop shape. It also offers excellent model-matching properties with few -if any- controller design iterations. Of course, other controller design tools may be used as well, as long as the loop-shaping objectives are met. The computation of the controller solving (5.3.1) can be performed with widely available software (MATLAB's Robust Control Toolbox).

5.4.1 The standard H_∞ problem

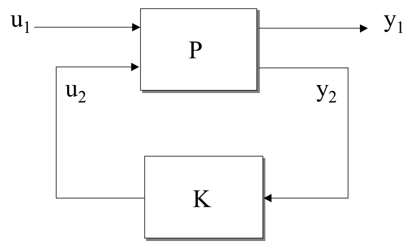

Figure 5.4: Augmented-plant/controller interconnections in the standard H-infinity problem.

The standard H_∞ problem is stated as follows: Consider the system in Fig. 5.4 where

$$P(s) \leftrightarrow [A,B,C,D] \leftrightarrow \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{array}\right]$$

with the obvious partitioning. (Here we use the symbol K for the controller to avoid possible confusion with the system output matrix C.) We seek to find a stabilizing control law $u_2 = K[y_2]$ such that the transfer function matrix $T_{u_1,y_1} : u_1 \mapsto y_1$ is small, e.g.:

$$\text{Optimal } H_\infty \text{ control: } \min_K \|T_{u_1,y_1}\|_\infty \;\; ; \;\; \text{Standard } H_\infty \text{ control: } K : \|T_{u_1,y_1}\|_\infty < \gamma$$

This transfer function is also referred to as a linear fractional transformation and has the form

$$T_{u_1,y_1}(s) = P_{11}(s) + P_{12}(s)[I - K(s)P_{22}(s)]^{-1}K(s)P_{21}(s)$$

It represents the transfer function between external signals (u_1) and outputs of interest (y_1) that should be minimized. The transformation of a controller design problem into the standard H_∞ problem is discussed in the next subsection.

The following theorem describes the solution in a special case that contains the essential features of the theory without being overly complicated. The general solution is omitted as it is much more involved algebraically.

5.4.1 Theorem: *Consider the special case where the following assumptions hold:*

1. $[A, B_1]$ *is stabilizable and* $[A, C_1]$ *is detectable;*
2. $[A, B_2]$ *is stabilizable and* $[A, C_2]$ *is detectable;*
3. $D_{11} = 0$, $D_{21} = 0$;

4. $D_{12}^\top [C_1,\ D_{12}] = [0,\ I]$;

5. $\begin{bmatrix} B_1 \\ D_{21} \end{bmatrix} D_{21}^\top = \begin{bmatrix} 0 \\ I \end{bmatrix}$.

Then, there exists an admissible controller such that $\|T_{u_1,y_1}\|_\infty < \gamma$ *iff the following conditions hold*

1. *There exists* $X \geq 0$ *such that* $A^\top X + XA + X(\gamma^{-2} B_1 B_1^\top - B_2 B_2^\top)X + C_1^\top C_1 = 0$ *(Control Riccati);*

2. *There exists* $Y \geq 0$ *such that* $YA^\top + AY + Y(\gamma^{-2} C_1^\top C_1 - C_2^\top C_2)Y + B_1 B_1^\top = 0$ *(Observer Riccati);*

3. $\lambda_{max}(XY) < \gamma^2$ *(Spectral radius; can also be written as* $Y^{-1} - X/\gamma^2 > 0$*)*

Moreover, when these conditions hold, one such controller is

$$K_\infty(s) \leftrightarrow \left[\begin{array}{c|c} A_\infty & -Z_\infty L_\infty \\ \hline F_\infty & 0 \end{array}\right]$$

where

$$\begin{aligned} A_\infty &= A + \gamma^{-2} B_1 B_1^\top X + B_2 F_\infty + Z_\infty L_\infty C_2 \\ F_\infty &= -B_2^\top X \\ L_\infty &= -Y C_2^\top \\ Z_\infty &= (I - \gamma^{-2} Y X)^{-1} \end{aligned}$$

This controller is often called the central controller or minimum entropy controller. The set of all possible solutions is then described as a linear fractional transformation of the central controller with a stable filter.

$\nabla\nabla$

Some interesting properties of the (general) solution are listed next.

- The H_∞ controller maintains a (less obvious) full-state/observer separation structure. The coupling between the controller and observer design is introduced by the parameter γ in the two Riccati equations. These are non-standard Riccati in the sense that the matrix of the quadratic term may be indefinite.[3]

- The optimal H_∞ problem is solved as a sequence of standard problems $\|T_{u_1,y_1}\|_\infty < \gamma$ with one adjustable parameter γ; this is usually referred to as the γ iteration.

- Solutions to this problem exist iff four conditions hold: D_{11} small enough; control Riccati $X \geq 0$; observer Riccati $Y \geq 0$; spectral radius $\lambda_{max}(XY) < 1$.

- The H_∞-optimal cost function T_{u_1,y_1} is all-pass ($\sigma(T_{u_1,y_1}) = 1$).

- An H_∞ suboptimal controller has order equal to the augmented plant.

- In weighted mixed-sensitivity problems, the H_∞ controller always cancels the stable plant poles with its transmission zeros. Unstable plant poles inside the control bandwidth are shifted approximately to their jw-axis mirror image.

[3]Non-standard Riccati are related to the problem of determining the H_∞ norm of a transfer function matrix: Let $G(s) \leftrightarrow [A, B, C, 0]$. Then $\|G\|_\infty < \gamma$ iff there exists $X \geq 0$ such that $XA + A^\top X + \gamma^{-2} X B B^\top X + C^\top C = 0$.

5.4.2 Transformation to the standard H_∞ problem

In the system identification section we discussed how the nominal system can be identifed from input-output data. The identification uncertainty was also expressed as (approximate) constraints on the sensitivity and complementary sensitivity transfer function matrices. Furthermore, performance objectives (disturbance attenuation) are usually defined in terms of a target sensitivity. Thus, the typical controller design problem takes the form of a loop-shaping weighted mixed sensitivity optimization

$$\min_C \gamma : \bar{\sigma} \begin{bmatrix} W_3 T \\ W_1 S \end{bmatrix} < \gamma \tag{5.4.1}$$

The selection of the two weights is such that W_1^{-1} is minimized subject to $|W_1^{-1}| + |W_3^{-1}| \geq 1$ (imposed by $S + T = 1$) and W_3^{-1} obeying the corresponding uncertainty constraint. Additional considerations follow the classical loop-shaping principles, e.g., location of right-half plane poles (minimum bandwidth) and zeros (maximum bandwidth), lightly damped modes etc.

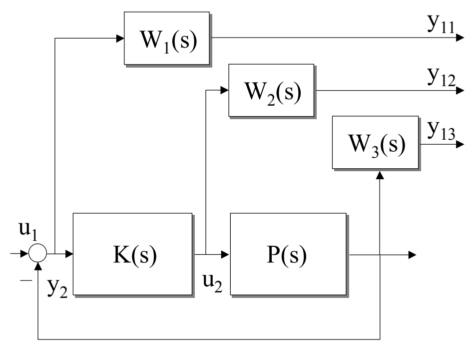

Figure 5.5: Transformation of the loop-shaping to the standard H_∞ problem.

To transform this problem to the standard H_∞ problem, the weights are appended to the closed loop system as filters on the quantities of interest (see Fig. 5.5). Notice that an additional weight W_2 is used to penalize the control input. This serves to ensure that the problem is well-posed and can be chosen as ϵI where ϵ is a small number. Of course, different selections of W_2 are possible, as long as there is sufficient and concrete justification. Now, it is a tedious but straightforward exercise to write a state-space representation for the augmented plant (or, super-plant) from the two inputs u_1, u_2 to the two outputs $[y_{11}, y_{12}, y_{13}]^\top$ and y_2. The controller, in a state-space realization, can then be obtained using standard software (e.g., MATLAB).

In the solution of the H_∞ optimization, γ controls the amount of relaxation of the constraints defined by the weights, until a solution is feasible. Recall that at the optimum, the cost objective is all-pass, so roughly, $T = \gamma W_3^{-1}$ and $S = \gamma W_1^{-1}$. Thus, a value of $\gamma = 1$ means that the constraints are feasible but tighter constraints are not. A value of $\gamma = .1$ means that the weights can be increased by a factor of 10 (tighter constraints) and a solution is still feasible. And a value of $\gamma = 10$ means that the weights must be reduced by a factor of 10 (more relaxed constraints) before a solution becomes feasible. In general, the optimum γ should be around unity so that the prescribed loop shape is achieved. Large deviations from this value mean that the weights need to be considerably adjusted before computing the solution. Their automatic adjustment (through γ), however, is not necessarily in a desirable direction (keeping the crossover region small so as to achieve the best sensitivity reduction). Practical experience indicates that good designs are obtained with $\gamma \sim 1/0.8 - 1/0.5$. Notice that in the Robust Control Toolbox the role of γ is inverted so that the right-hand side is always normalized to one.

5.5 Controller Adjustments

5.5.1 Model Reduction

Before its final implementation, the controller obtained from an H_∞ design needs to undergo several adjustments and modifications. The first adjustment is a model-order reduction, necessitated by the usually very high order of the controller (same as the augmented plant). The general and unified H_∞ problem formulation and the selection of convenient bi-proper weights introduce artificial modes in the controller that are essentially irrelevant for its performance and can potentially degrade its reliability and discretization properties. For this reason, a three-step order reduction is performed. The first two steps remove fast and slow modes, by using a "slow-fast" decomposition algorithm and singular perturbation principles. Modes much faster than the closed-loop bandwidth are undesirable for discretization and for their numerical sensitivity. Modes much slower than the bandwidth are artificial and will cause the appearance of residual terms in the response that are small but decay very slowly with time. In the last step a simple frequency-weighted balanced model reduction is used to eliminate other controller states with small contributions. A considerable part of the literature has focused on the subject of controller reduction with the objective of offering closed-loop stability and performance guarantees. However, without attempting to minimize the controller order and by allowing for user input, the previous reduction sequence can be automated to become almost transparent to the user.

5.5.2 Response to reference inputs

The primary responsibility of the controller is to attenuate disturbances and maintain robust stability. Although important, achieving a good response to commands is addressed separately. There are three different (but related) approaches to command response. One is the design of a prefilter ($u = Ke + Fr$) that can be handled as an approximation problem. A second is through the design of an outer loop controller. The third involves the separation of the controller into a cascade and feedback part (two-degrees-of-freedom compensators). Each method has its own individual characteristics in terms of computations, handling saturations and sensitivity. All methods, on the other hand, share the same principle, that the command response can be shaped roughly up to the bandwidth of the sensitivity; the latter is nothing more than the frequency range where the closed-loop transfer function can be confidently inverted.

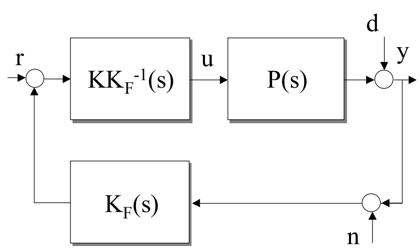

Figure 5.6: Simplified prefilter design as a 2-DOF compensator

Here, we consider only a special case of the third method where the compensator is split into a cascade and a forward path, i.e., $K = K_c K_f$. With a suitable factorization, slow zeros of the compensator are included in the feedback path and, thus, they do not appear in the transfer function from the reference to the output. Observe that, regardless of the split, the loop transfer function remains the same and so do the sensitivity and complementary sensitivity of the feedback systems. The general factorization can be somewhat involved and double the order of the controller. Instead, a simpler approach may be used to handle certain special cases. For example, when the plant contains an integrator or slow dynamics, the complementary sensitivity will necessarily exhibit an overshoot. Often, it is sufficient to use a low-pass filter, say $F = \frac{s+z}{s+p}$, $z > p > 0$, at the reference to avoid such overshoots in the command response. Bringing this filter inside the loop (see Fig. 5.6), we have that $K_f = F^{-1}$ and $K_c = KF$. The advantage of this form of prefiltering is that it remains active, even when the control input saturates.

5.5.3 Anti-windup modifications

Integrator windup is an old and serious problem, recognized from the days of the first PID controllers. It occurs when the input saturates and the controller keeps integrating the error, demanding even higher levels of control input. This results in unacceptable overshoots and potential controller instability. With simple PID controllers, a simple effective anti-windup strategy is to stop the error integration when the input saturates. On the other hand, general multivariable controllers require more elaborate anti-windup strategies.

Several techniques to approach the anti-windup problem can be found in the literature. The simplest, and often sufficient, is the augmentation of the controller with an observer-based feedback. That is,

$$\dot{x}_c = A_c x_c + B_c e + L(u - u_{sat}) \; ; \quad u = C_c x_c + D_c e$$

where $[A_c, B_c, C_c, D_c]$ is the controller state-space representation, e is the controller input (usually an error signal) and u_{sat} is the saturated control input. The main idea behind this approach is to prevent the controller states from diverging too far while the input is saturated. An essential condition for the success of the design is that the the system will recover from the saturation. This is not easy to guarantee with multivariable systems that exhibit strong interactions. A partial remedy for that is to adjust the observer to maintain an a priori known, desirable directionality for the saturation (e.g., “cold-start” and “free-fall” behavior). A reasonable design of the observer gain L is through a Riccati equation:

$$L = -PC_c^\top R^{-1} \quad ; \quad PA_c^\top + A_c P - PC^\top R^{-1} CP + Q = 0$$

where $Q = \delta I + B_c B_c^\top$ (δ
1) and $R = \mathrm{diag}(\rho_1, \ldots, \rho_m)$. In this form, the directionality properties of the anti-windup modification can be adjusted by manipulating the entries of R (e.g., via simulation trial-and-error).

As a last comment, notice that most controllers are now implemented in discrete time. Typically, as long as the sampling frequency is well above the closed-loop bandwidth, the discretization should present no special problems. However, it is a good practice to perform the discretization using backwards differences or a Tustin transformation to ensure that fast controller poles map to stable poles in discrete-time. Furthermore, to avoid similar discretization problems, it is preferable that the design of the anti-windup modification is performed entirely in discrete time.