Linear Controller Design: Limits of Performance Via Convex Optimization

STEPHEN BOYD, CRAIG BARRATT, AND STEPHEN NORMAN

We give a tutorial presentation of an approach to the analysis and design of linear control systems based on numerical convex optimization over closed-loop maps. Convexity makes numerical solution effective: it is possible to determine whether or not there is a controller that achieves a given set of specifications. Thus, the limit of achievable performance can be computed.

Although the basic idea behind this approach can be traced back into the 1950s, two developments since then have made it more attractive and useful. This first is a simple description of the achievable closed-loop behaviors for systems with multiple sensors and actuators. The second is the development of numerical algorithms for solving convex optimization problems, and powerful computers to run them.

I. Introduction

A. Control Engineering and Controller Design

To provide a context for the material in this paper, we first give a very brief overview of control engineering. The goal of control engineering is to improve, or in some cases enable, the performance of a system by the addition of *sensors*, which measure various signals in the system and external command signals, *control processors*, which process the measured signals to drive *actuators*, which affect the behavior of the system. A schematic diagram of a general *control system* is shown in Fig. 1. The use of the sensed response of the system (and not just the command signals) in the computation of the actuator signals is called *feedback control*, an old idea which has been developed and applied with great success in this century [1], [2].

Control engineering involves

1. *Modeling or identification.* The designer develops mathematical models of the relevant aspects of system to be controlled. This can be done using knowledge of the system (for example by applying Newton's equations of motion to a mechanical system), and experimentally by observing responses of the system to various excitations, a procedure known as *system identification* [3]. In some cases, several models are developed, varying in complexity and accuracy.
2. *Control configuration: selection and placement of sensors and actuators.* The designer decides which signals in the system will be measured or sensed, with what sensor hardware, and similarly what actuators will be used. In a large industrial process, for example, the control engineer might decide which temperatures, flow rates, pressures, and concentrations to sense. Choosing the type and positioning of control surfaces on an aircraft is an example of actuator selection and placement.
3. *Control law or controller design.* This is the topic of this paper. The designer decides exactly how the actuators are to be driven by processing the incoming sensor signals. The *controller* or *control law* describes the signal processing used by the control processor to generate the actuator signals from the sensor signals. The area of control law design is extensively studied and taught.
4. *Controller implementation.* Once the control law is chosen, the control processor which implements this law must be designed. This may involve mechanical design, analog and digital circuit design, and software design.
5. *Control system testing and validation.* This may involve extensive computer simulations with a complex, detailed mathematical model, real-time simulation of the system with the actual control processor operating ("hardware-in-the-loop"), and actual field tests fo the system.

While each of these five tasks can be critical in control engineering, most research and teaching concentrates on modeling and controller design, perhaps because it is difficult to precisely formulate the problems of the other tasks. Our concentration on controller design should not be taken as an implicit assertion of its greater importance in control engineering—indeed, we note that the control configuration can have an enormous impact on the final performance of the system. In particular, the difference between a good and bad control configuration can be dramatically greater than the difference between a good and bad control law.

Manuscript received November 30, 1988; revised August 4, 1989. This work was supported in part by the National Science Foundation (NSF) under ECS-85-52465, the Air Force Office of Scientific Research (AFOSR) under 89-0228, Boeing Electronics Company under LF0937, and Bell Communications Research, and the National Science and Engineering Research Council (Canada) 1967 Science and Engineering Scholarship.

The authors are with the Dept. of Electrical Engineering, Stanford University, Stanford, CA 94305, USA.

IEEE Log Number 8933936.

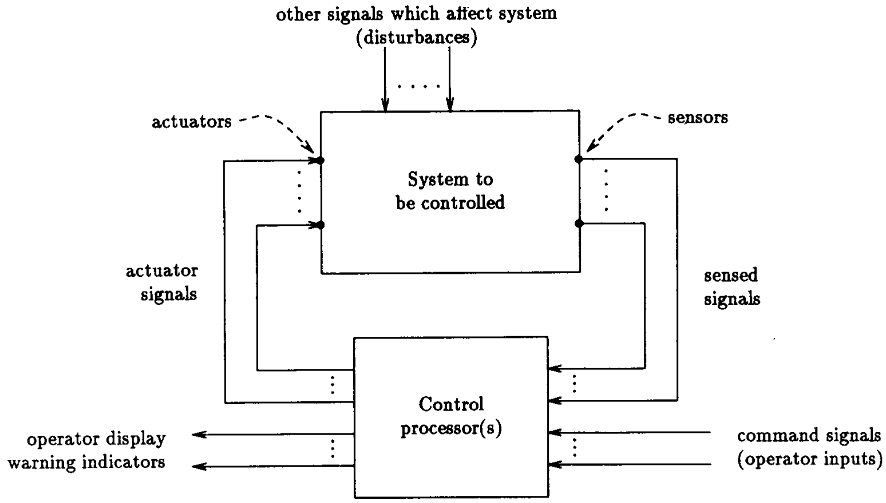


Fig. 1. A schematic diagram of a general control system.

B. Design Specifications for Control Law Design

Design specifications can be loosely divided into three categories: performance specifications, robust performance specifications, and control law specifications.

Performance specifications describe how the system performs closed-loop, meaning with the control processor operating. Examples of qualitative performance specifications are:

- the closed-loop system is stable.
- the actuator signals generated by the control law are not too large.
- the effects of disturbances or noises on the system are small.
- the system responds in particular ways to reference or command inputs.

Robust performance specifications describe how the closed-loop system would perform if some parts of the system were changed or perturbed. The perturbations could be due to any of the following:

- the system under control may have been inaccurately modeled or identified.
- the system under control, or the controller, physically changes, perhaps due to component tolerances or temperature coefficients.
- certain nonlinearities may have been ignored in the design process, but may be significant in the real closed-loop system.
- the operating point of a nonlinear system changes, so a small signal linear approximation becomes less accurate.
- gross failures, such as a sensor or actuator failure, may have occurred.

The robust performance specifications themselves can take several forms. The general flavor of these specifications is to limit the degradation (or variation) of the behavior of the closed-loop system to some subset of the perturbations. Examples of robust performance specifications are:

- *Low differential sensitivities.* The sensitivity (derivative) of some closed-loop quantity to some parameter, or some family of parameters, is small.
- *Robust stability.* The closed-loop system remains stable in the face of some specific set of perturbations, for example, parameters varying over given (not necessarily small) ranges.
- *Robust performance.* The closed-loop system continues to meet some specific set of performance specifications in the face of some specific set of perturbations.

Control law specifications describe properties of the control law itself. Examples of control law specifications are:

- the control law has a simple structure or form, for example, described by a low order linear differential equation. Another example is that it be a *decentralized* control law: each actuator signal may depend on only a few of the sensor signals.
- the controller is open loop stable.
- the controller can be implemented using a particular control processor.

In sections V–VIII we discuss design specifications in more detail.

We can describe the fundamental problem of control law design as:

Fundamental Problem: Given a specific system to be controlled, control configuration, and set of design specifications, either find a control law that meets these design specifications, or determine that none exists.

An aspect of the Fundamental Problem that we stress is the determination of *whether or not there is a control law that meets the given design specifications*, or in other words, whether the design specifications are too tight. This aspect of the Fundamental Problem is hardly trivial—we will see in section X that, even for a very simple system, it is easy to write down specifications which seem quite reasonable but which nevertheless cannot be met.

This aspect of the Fundamental Problem can be as important in control engineering as finding or synthesizing an appropriate control law when one exists. If it can be determined that no control law can achieve a given set of specifications for a given control configuration, then the search for a suitable control law can be abandoned. We may then

- change the control configuration by adding, relocating, or changing sensors or actuators,
- change (relax) the specifications.

Thus, even though the Fundamental Problem concerns control law design, it can be quite relevant to other aspects of control engineering, particularly control configuration.

No method is currently known for solving the Fundamental Problem. Various analytic methods can solve the Fundamental Problem when the specifications have very specific forms, usually far removed from specifications appropriate in a real design problem. When they can be applied, the analytic methods have the advantage that the fundamental problem can be precisely answered. In section X we present an example for which the Fundamental Problem can be answered using an analytic method.

In practice, various heuristic methods are commonly successful at finding a control law that meets the design specifications, in those cases where such a control law exists. These methods depend upon talent, experience, and a bit of luck on the part of the designer. If the designer is successful and finds a controller meeting the specifications, then of course the Fundamental Problem can be answered affirmatively. However, if the designer *fails* to design a controller that meets the given specifications, he or she in general *cannot* conclude that there is no controller meeting the specifications, although he or she may *suspect* that the specifications are not achievable. It could be that another design approach or method (or indeed, designer) would yield a controller meeting the specifications.

C. Purpose of this paper

> The purpose of this paper is to describe how the Fundamental Problem of controller design can be solved for a restricted set of systems and a restricted set of design specifications, by combining a recent theoretical result with numerical convex optimization techniques.

The restriction of the systems is that they must be linear and time-invariant (LTI).

The restriction on the design specifications is that they be *closed-loop convex*, a term we shall describe in detail in section IV. This restricted set of design specifications includes a wide (although incomplete) class of performance specifications, e.g., limits on RMS regulation or tracking errors, actuator authority, rise-time, and overshoot. Closed-loop convex design specifications include a less complete class of robust performance specifications, and essentially none of the control law specifications.

The recent theoretical result referred to above is: Given a linear system to be controlled, the set of *closed-loop* transfer matrices that can be achieved is easily characterized. Thus, focus is shifted from designing a control law that will yield good closed-loop system behavior, to directly designing good closed-loop system behavior, and only then, if at all, determining the control law that yields this system behavior.

If the specifications are achievable, it is possible to find a controller which meets the specifications, although the controller found may be complex (high order). It may be possible to find a simpler controller which meets the specifications, for example by some model reduction technique. More significantly, the Fundamental Problem can be solved for a given system to be controlled, control configuration, and set of (closed-loop convex) design specifications, and therefore the *limit of performance can be determined for a given system and control configuration*.

This ability to numerically determine limits of performance extends various rules-of-thumb used by practicing engineers to approximate the best performance achievable with a given system and control hardware. These rules-of-thumb might be based on knowledge of maximum actuator effort, of limits on loop gain imposed by delay or other excess phase in the system to be controlled, of robustness requirements, and so on.

No matter which controller design method is used by the engineer, knowledge of the achievable performance is extremely valuable practical information, since it provides an absolute yardstick against which any designed controller can be compared. To know that a certain candidate controller, which has low order and is easily implemented, achieves an RMS regulation error only 10% above the minimum achievable by *any* linear controller, is a very strong point in favor of the design. In this sense, this paper is not about a particular controller design method or synthesis procedure; rather it is about a method of determining what specifications (of a large but restricted class) can be met using any controller design method, for a given system and control configuration.

D. Paper Outline

Section II gives a broad outline of various approaches to controller design for LTI systems, as a basis for comparison with the approach presented in this paper. This section briefly describes the advantages and disadvantages of current controller design approaches. Readers already familiar with these ideas may wish to skim this section.

Section III presents a formal framework for what we describe above as the system to be controlled and the control configuration. In particular, notation which is used in the sequel is defined. This section also explicity describes the assumptions that are made about the system and controller.

Section IV shows that many performance specifications have natural and useful geometric interpretations. These geometrical properties will be used in section IX to describe *effective* numerical solutions to the Fundamental Problem. We define the notion of a *closed-loop convex* design specification.

Section V discusses the performance requirement that the closed-loop system be stable. The main result of this section is that the set of closed-loop transfer matrices achievable with stabilizing controllers has a simple representation. To develop the main idea of this paper, this main result is all that is needed. This section, however, describes at some length its historical development, explores different perspectives on it, and gives several interpretations of it.

In section VI we show that many performance specifications can be expressed as *convex* constraints on closed-loop transfer matrices. Section VII discusses robust performance specifications, and how some of these can be expressed as *convex* constraints on closed-loop transfer matrices, and section VIII shows that several important control law specifications *cannot* be expressed as convex constraints on closed-loop transfer matrices.

Section IX poses the Fundamental Problem, with a restricted set of design specifications, as a convex optimization problem. Since this optimization problem is convex, there are "good" methods for solving it numerically.

Section X shows an example of how the methods presented in this paper may be used to compute the performance limits achievable with a particular plant and control configuration.

Section XI explores the history of some of the ideas presented in this paper, and also outlines recent work related to this paper.

II. On LTI Controller Design Approaches

There are several families of approaches to LTI controller design. Historically, there are two broad categories: "classical" methods, which generally involve working with transfer function descriptions of systems, and "modern" (state space) methods, which involve working with descriptions of systems of ordinary differential equations. Of course, the boundary between the two categories is not sharp. In this section we examine some of the advantages and disadvantages of various design approaches. Our goal is not to make judgments about merit but to provide some basis for comparison between different approaches.

A. Synthetic Methods Versus Analytic Methods

An essential distinction to make is between what we shall call *synthetic* methods and what we shall call *analytic* methods. (A similar but not identical distinction is made in [4] between the "trial-and-error design method" and the "analytical design method." See section XI.)

In a *synthetic* method, the designer starts with a simple controller and continues to add to it, in the hope that after some number of additions the controller will meet the design goals. In other words, the controller is "built up" piece by piece, possibly using a variety of "tools" at different stages. Synthetic methods are described in, for example [5], [6], [7], [8].

An *analytic* method, on the other hand, is based on an analytic solution of some optimal controller design problem, for example, the Linear Quadratic Gaussian (LQG) problem. Since analytic solutions are currently available only for a few very specific types of design problems, the designer must try to formulate an analytically solvable controller design problem that captures as closely as possible the actual design goals. Analytic methods are described in, for example, [9], [10], [11].

1) Synthetic methods: In a synthetic method, each part added to the controller often has a specific purpose: a notch filter notches out a mechanical resonance, a low-pass filter reduces noise from a sensor, and a lead-lag section increases the phase of the loop gain in a certain frequency band. Thus in a synthetic approach the controller is often naturally divided into simple subsystems. This has the very great advantage that the designer "understands" how the controller works or "what part does what."

In carrying out a synthetic design, the designer will face these questions at each stage:

- How do I "tweak" the present controller to improve its performance?
- How do I decide that the current controller structure *cannot* meet the design goals (i.e., "tweaking" alone will not work), so that another part must be added to the controller?
- How do I decide what would be an appropriate part to add to the controller to improve its performance?

These questions are often dealt with using an open-ended "toolbox" or "cookbook" approach. A wide variety of tools is available to the designer. For low-order, single-input, single-output systems there are several well-known classical techniques, including the use of root-locus methods and reasoning involving Bode plots, Nyquist diagrams, or Nichols charts. For more complex systems, advances in optimization theory and numerical computing allow the application of sophisticated numerical parameter optimization methods.

Synthetic controller design methods can work well when there are "many" relatively simple controllers that meet the performance goals. The principal advantages of synthetic approaches are:

- They are sparing with controller complexity; unlike many analytic methods, they tend to use the least complex controller needed to get the job done.
- The controllers designed by a synthetic method often retain an understandable form—each subsystem of the controller performs its own identifiable task.
- They very often work, especially when the performance goals are modest.
- They are often economical in terms of the time, intellectual energy, and computation spent in the design effort.

Their main disadvantages are:

- They are not *effective*, in the following technical sense: if they fail, the designer cannot be sure that there really is no solution to the problem. Furthermore, if they fail, the designer is not left with a clear idea of what the actual limits of performance for the given system are.
- The designer is not working according to a well-defined algorithm—he or she must rely repeatedly on intuition and experience to make decisions. If the problem at hand is too far removed from problems the designer has seen before, he or she may not have any idea how to go about improving performance by adjusting the current controller or adding parts to it.
- The problem may be molded into an "understandable" form at a substantial cost in final system performance. For example, designers using synthetic methods often *pair* sensors and actuators and design the controller "loop-at-a-time," in order to be able to use all the intuition and experience built up in "single-

loop" design. This can greatly reduce the final performance of the system.

2) Analytic methods: An *analytic* method is based on an analytic solution of some optimal controller design problem, for example, the Linear Quadratic Gaussian (LQG) problem. Only very rarely is there an analytical solution for the exact problem the designer wishes to solve. To get around this difficulty, the designer essentially designs the optimal control *problem* in such a way that the resulting optimal controller, which can be analytically determined, meets the design goals. Techniques available for the design of this optimal control problem include: selection of weight matrices (as in the LQG problem); addition of fictitious noises to the system model; and addition of fictitious dynamics to the system model.

The controllers designed by analytic methods are often complex: high order, with (in the multiple-actuator multiple-sensor case) every sensor affecting every actuator signal. This complexity may obscure the fact that the controller may be very close to, say, a simple lead-lag filter with a notch filter, which could have been designed by a synthetic method. Analytic methods are often followed by a *controller reduction* phase, the goal of which is to find a less complex controller that does essentially the same job as the analytically designed controller.

An advantage of an analytic method is that, provided the optimal control problem is reasonably well-posed, the resulting optimal controller tends to yield at least *reasonable*, if not good, performance, in terms of the original design goals. In particular, controllers designed by analytic methods will stabilize the plant. This is a critical advantage in the case of, say, an unfamiliar, unstable plant with many actuators and sensors, where the designer may have difficulty even in knowing where to start in his search for an acceptable design.

A disadvantage of an analytic method is the difficulty of expressing actual design goals in the framework of an analytically solvable optimal control problem. The analytic designer has to deal with the same kinds of issues in his design of an optimal control problem that the synthetic designer deals with in his design of the actual controller. The analytic designer's problem is compounded by the fact that the physical intuition that guides the synthetic designer is much harder to apply to the design of such things as weight matrices and fictitious dynamics.

B. Classical Synthetic Open-Loop Design

Classical synthetic open-loop design methods have their origin in the work of Bode [5]. These methods are extremely widely studied and applied, and are described in many current introductory control texts, such as [12]–[15]. Examples of the numerous works developing and extending synthetic open-loop techniques are [6]–[8], [16]. In this section we will briefly describe and comment on this kind of approach to control design.

Consider the classical feedback control setup shown in Fig. 2. Given the plant, P, the designer must find a controller K that results in satisfactory closed-loop performance. Classical open-loop methods concentrate on designing the loop gain, $L = PK$. The advantage of working with the open-loop system is that L is simply the product of P, the "fixed part of the system" and K, the controller; on the other hand, the closed-loop transfer function from r to y, $PK/(1 + PK)$, depends on K in a more complicated way. The effect on the loop gain of changing K is much easier to see than its effect on the closed-loop transfer function.

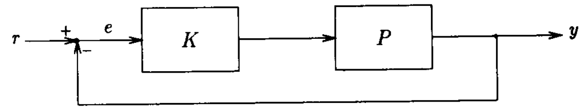


Fig. 2. Classical feedback control setup.

The various design goals for the closed-loop sytem are translated into requirements on the open-loop gain L. For many designers, this translation is so automatic that they might be said to "think open-loop"; we even find specifications for control systems described in terms of the open-loop gain. Requirements for the loop gain derived from design goals for the closed-loop system include:

- *Closed-loop stability.* This is often stated using the Nyquist stability criterion: stability of the closed-loop system is related to the number of net clockwise encirclements of the point -1 by the loop gain L in the complex plane.
- *Tracking reference signals.* The specification of good tracking performance can be expressed as the requirement that the loop gain L have large magnitude at frequencies up to the bandwidth ω_B of the reference signals.
- *Disturbance rejection.* The effect of disturbances added anywhere in the loop can be made small by requiring the loop gain to be large at those frequencies where the disturbance is significant.
- *Robustness to "loop perturbations."* The proximity of the Nyquist plot to the critical point -1 gives one measure of stability robustness of the closed-loop system. Two often used measures of this proximity are the classical gain and phase margins. Another robustness requirement is that the loop gain magnitude should be safely smaller than 1 above the plant "cutoff frequency" ω_C.
- *Stability and simple structure of the controller K.* The designer simply avoids candidate controllers that do not meet these specifications.

To achieve tracking of references and some level of robustness, the designer will require a large loop gain at frequencies lower than ω_B and a small loop gain at frequencies higher than some cutoff frequency ω_C. Typical constraints on the loop gain are shown in Fig. 3.

A designer will typically add dynamics to the controller K (and hence the loop gain $L = PK$) until the loop gain satisfies the constraints shown in Fig. 3 and the closed-loop system is stable; the designer might add such things as integrators or lowpass filters to get the desired magnitude response in the loop gain, and then, if necessary, use lead or lag networks to adjust the phase of L to ensure closed-loop stability.

Major drawbacks of this type of approach include:

- Many design goals, such as low noise sensitivity and acceptable transient behavior, can only be incorporated indirectly, by various rules-of-thumb.
- All the general disadvantages of synthetic methods which were outlined earlier in this section.

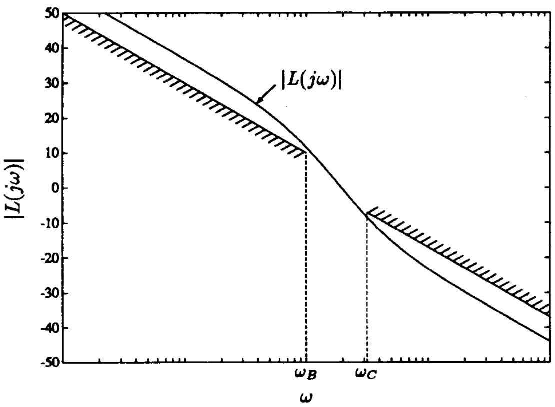


Fig. 3. Examples of constraints on the loop gain L.

- Many methods developed for single-actuator, single-sensor systems are not readily extended to methods for multivariable systems.[1]

C. Analytic State-Space Methods

"Modern" or state-space methods use systems of ordinary differential equations to describe the physical systems involved in the control problem. State-space methods for control problems are generally considered to have their origins in the 1950s, in the work of Bellman and Kalman in the United States and Pontryagin in the Soviet Union. State-space methods are applicable to a very broad class of control analysis and design problems for nonlinear and time-varying linear systems, but here we will be concerned only with applications of state-space techniques to LTI feedback controller design. Texts introducing state-space ideas include [18]–[21].

There are well-known advantages to the use of state-space models for LTI systems. A state-space model is particularly useful in the case of multivariable systems where the structure of the model emphasizes that the entire system can be described in terms of its state; multiple inputs mean multiple available degrees of freedom for affecting or redirecting the state, while multiple outputs mean multiple "perspectives" for viewing the state.

A key result, sometimes called the *separation principle*, in LTI state-space theory is that any plant without unstable hidden modes can be stabilized with a control law that consists of an observer (a system to estimate plant state from knowledge of plant input and output) and a constant state feedback gain matrix. The controller design problem can be posed as a search for satisfactory observer designs and state feedback gains. Linear Quadratic Regulator (LQR) theory provides an analytical solution to the problem of finding a state feedback that minimizes a weighted sum of quadratic measures of the cost of state excursion and the cost of actuator use in driving the state to zero. Linear Quadratic Gaussian (LQG) theory incorporates LQR theory and Kalman filter theory (dual to LQR theory) to find analytically a state feedback and an observer design which together minimize an LQR-like cost function in the presence of white Gaussian process and measurement noises [22], [9].

It is possible that the goal of a controller design effort is precisely the minimization of an LQR-like cost and that the noises affecting the plant are described well by the models that fit into the LQG formulation. But it is much more likely that the designer will have to come up with LQR cost matrices that somehow express different, actual performance specifications, and fictitious noise models that express the degree of caution the observer should maintain when using plant output to estimate plant state. In typical applications of LQG theory to controller design problems the designer keeps adjusting the weight matrices in the LQG problem until a controller appears that meets the design goals. Some discussion of how to select LQR weight matrices appears in [9]. Extensive comments on the problem of LQG design iteration appear in [23], along with many references. An attempt to deal with robustness requirements while maintaining an LQG approach has led to the development of the Loop Transfer Recovery (LTR) approach [24], which is often quite successful at producing a controller with reasonable noise sensitivity and robustness properties.

D. Parameter Optimization Methods

Parameter optimization methods for LTI feedback design start with controller structures that are motivated by ideas from classical, modern, or other techniques. What is generally meant by *controller structure* is a system model with one or more parameter values that can be adjusted. A simple example is a PI (proportional-plus-integral) controller structure—the coefficients K_p and K_i in the controller transfer function $K_p + K_i/s$ are the parameters. Another example is a fixed-order state space model for the controller—the parameters are the entries in the matrices of the state space model in some canonical form.

The next step in a parametric method is to select a cost function that will represent the quality of system performance. One way to get a cost function is to take one from an analytically solved optimization problem, for instance the LQG problem. This has the advantage that the cost yielded by the structured controller by parameter search can then be compared to the absolute minimum achieved by any controller, which is analytically computable.

Another possible way to obtain a cost function is to form a weighted sum or maximum of various performance indices, such as integrated square error in response to a step command, integrated magnitude of frequency response across some band where a disturbance is concentrated, some index representing actuator use, and so on; the idea is that the weights define the relative importance of different aspects of system performance. Finally, the designer may add explicit constraints, such as bounds on the values of the parameters, bounds on closed-loop pole locations, bounds on open-loop frequency responses, and so on.

After a controller structure, a cost function and possibly some constraints have been specified, the designer has a nonlinear optimization problem to solve, a problem that will almost certainly require numerical solution. Many techniques for numerical solution of optimization problems resulting from control design problems have been proposed in the control literature. Some of these techniques

[1]This is not to say that these extensions are impossible. To cite just three examples, MacFarlane and Kouvaratakis [8], MacFarlane and Postlethwaite [16], and Freudenberg and Looze [17] have demonstrated how many of the classical ideas involving loop gain can be extended to multivariable systems.

are simple heuristic algorithms, such as steepest descent; some are highly specialized algorithms for certain kinds of problems; others are sophisticated software packages for handling very general classes of problems.

It is beyond the scope of this paper to provide a comprehensive overview of parameter optimization methods. We make a few sample citations to give some idea of the breadth of this field: SANDY, a gradient technique for finding optimal controllers of a fixed order [25]; a survey of work on the problem of finding the optimal constant output feedback (or fixed structure controller) in terms of LQG cost function [26]; parameter optimization applied to aircraft control design problems [27]; sophisticated interactive optimization software, connected to system simulation software, for handling a wide range of constrained parameter optimization probelms [28]–[30]. A good treatment of the issues involved in the selection of a nonlinear optimization algorithm and the issues involved in computer implementation of the selected algorithm is [31].

Good reasons for using parameter optimization methods include

- If the designer is fairly certain that some controller of the selected structure will do an adequate job, parameter optimization is a good way to look for that controller.
- A much greater range of cost functions and constraints are allowed than are available in analytic methods.
- If a reasonable controller has already been designed by, say, open loop synthesis or an analytic method, parameter optimization can be used to see if better performance can be obtained by changing the parameters in the existing controller.
- The designer can allow precisely as much controller complexity as he or she sees fit.

Problems with parameter optimization techniques include:

- Designing an iterative optimization algorithm to converge quickly to a locally optimal solution is a hard task.
- "Reasonable" initial guesses at parameter values may be required to start the optimization algorithm. It may be hard to come up with such a "reasonable" guess for a complex problem.
- A general nonlinear optimization problem is likely to be *nonconvex*. When applied to nonconvex problems, optimization algorithms in general use may converge to points that are locally but not globally optimal. When the number of parameters is anything but tiny, the computational effort required to ensure that a local optimum is in fact a global optimum is prohibitive. See section IX for more discussion of this important point.
- Even if a globally optimal controller of the selected structure is found, the designer does not know whether a drastic improvement in performance could be obtained by using some other controller structure.

III. Setup and Framework

A. Some Terms and Definitions

In this section we consider in some detail a formal framework for what we described in section I as the "system to be controlled," the "control configuration," and the "control law."

We start with a model of the system to be controlled, with a fixed control configuration. The inputs to this model, in other words, those signals which affect the model, will include the actuator signals from the controller and many other signals which might represent various noises and disturbances acting on the system. We will see in section VII that it may be useful to include among these input signals some fictitious inputs. These are inputs which are not used to model any specific noise or disturbance, but rather "spare" inputs which will allow us later to ask the question, "*what if* a signal were injected here?"

Definition 1 The inputs to the model are divided into two vector signals:

- The *actuator* or *control* signal vector, denoted u, will consist of those inputs to the model *that can be manipulated by the controller.* The actuator signal u is the signal generated by the controller.
- All other input signals to the model will be lumped into a vector signal w, called the *exogenous input.*

Of course, our model of the system must provide as output every signal that we care about, i.e., every signal needed to determine whether a given controller yields an acceptable design. These signals would typically include the signals we are trying to regulate or control, all actuator signals (u), and perhaps important internal variables, for example stresses on various parts of a mechanical system. The model must also produce the sensor signals, which may or may not overlap with the output signals above.

Definition 2 The outputs of the model consist of two vector signals:

- The *sensor* or *measured* signal vector, denoted y, will consist of those output signals *that are accessible to the controller.* The sensor signal y will be the input signal to the controller.
- The output signals from the model will be lumped into a vector signal z, called the *regulated variables.*

The *plant*, denoted P, will refer to the model of the system and the two vector input signals z and w. This is shown in Fig. 4.

Figure 5 shows the plant connected to the controller. We refer to this as the *closed-loop system.*

Our notion of plant includes more detail about the system than is common in classical control. First, our notion of the plant includes information about exactly which signals are accessible to the controller. In classical control, this

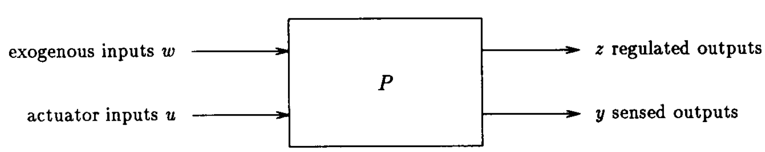


Fig. 4. Decomposition of the plant's inputs and outputs.

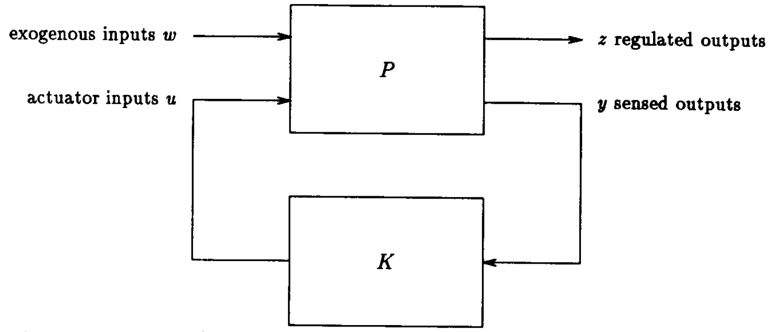


Fig. 5. The closed-loop system.

is *side information* given along with the plant in the controller design problem. For example, what would be called "state feedback" and "output feedback" for a given classical plant, we would distinguish as *two different plants,* since the sensed signals (our y) differ from the two controller design problems. We will see that other distinctions made in classical control, for example between one-degree-of-freedom, two-degree-of-freedom, and feedforward controllers, are readily expressed in our framework as controller design problems for *different* plants.

Second, the exogenous inputs and the regulated variables are *explicitly declared or described.* The intention is that *z and w should contain every signal about which we will express a constraint or specification.* In classical control, the disturbances *might* be indicated in a block diagram showing where they enter the system; some important exogenous inputs and regulated variables are commonly left out, since it is expected that the designer will simply know that an acceptable design cannot excessively amplify a signal injected at some point in the system.

The reader may have noticed that command or reference signals do not appear explicitly in Fig. 5. In a classical framework, command signals would appear as separate inputs to the controller, or subtracted from other signals to form error signals, as shown in Fig. 6.

Our treatment of command signals simply follows the definitions above. If the command signal is directly accessible to the controller, then it is included in the signal y. If the command signal is subtracted from some other signal to form an error signal which is accessible to the controller (as in Fig. 6), then the error signal (and not the command) is included in y. Thus the command signals enter the controller via y, *from the plant.* There remains the question of how the command signals enter the plant. Again, we follow the definitions above: the command signals are plant inputs, not manipulable by the controller (they are presumably manipulable by some external agent issuing the commands), and so they must be *exogenous inputs*, and therefore included in w. Often, exogenous inputs that are commands pass directly through the plant to some of the components of y. The classical closed-loop system in Fig. 6 can be redrawn in our framework as in Fig. 7.

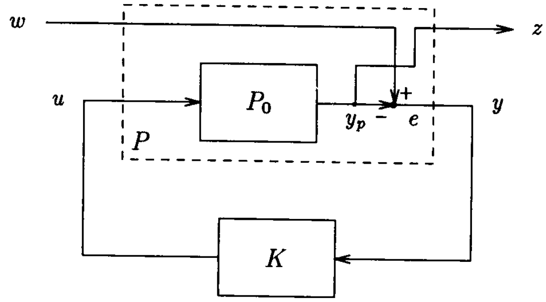


Fig. 7. Classical feedback loop in our framework.

This method of handling command signals may appear confusing at first, but the examples to follow should help clarify. For now, we make a few comments about it. First, it seems confusing to call a command an exogenous input to the plant, when the command only affects the system through the controller. On the other hand, it is not hard to think of cases where a command might directly affect the system (and thus deserve more to be called an exogenous input) as well as affect the system through the controller.

Second, it seems confusing to call a command signal a sensor signal, since in classical control a command signal is represented as ready for injection into the controller without being sensed or measured. We can, however, imagine that a command signal originates as a physical signal that must be sensed or measured in order to be accessible to the controller. Suppose, for example, that a potentiometer is used to specify a set-point for a regulator. We could view the potentiometer as a *sensor* (therefore part of the plant), the shaft angle as one of the exogenous inputs, and the voltage output as the corresponding sensed command.

B. Linear, time-invariant systems

In the sequel we assume the following:

We assume that the plant, P, and controller, K, are linear and time-invariant (LTI), and lumped.

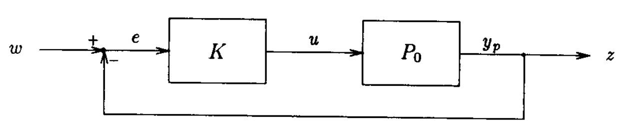


Fig. 6. A classical feedback loop with injection of a reference signal.

Before proceeding, we make a few comments about this assumption:

- Many interesting and important plants are *highly nonlinear*—most robots or mechanical systems which undergo large (e.g., slewing) motions, for example.
- The assumption always represents an approximation to some degree. In particular, the approximation will only be good for certain ranges of values of system signals, over certain time intervals or frequency ranges, and so on. The performance specifications might include requirements that these signals stay within regions where the linear system model accurately approximates the actual plant.
- Even if the plant is LTI, the restriction that the controller be LTI can also be a substantial restriction. The minimum time control problem with a limit on actuator authority is a good example of this—the optimal controller is well known to be highly nonlinear (bang-bang control [32]). Adaptive controllers ("self-tuning regulators") are another example.

We wish to emphasize that our restiction to LTI plants and controllers is hardly a minor restriction, even if it is a commonly made one. Nevertheless, we believe the material of this paper is still of great value, for several reasons:

- Many nonlinear plants *are* well modeled as LTI systems, especially in regulator applications, where the goal is to keep the system state near some operating point.
- It often occurs that a controller, designed on the basis of a linear model of a nonlinear plant, will work well with the nonlinear plant, even if the linear model of the plant is not particularly accurate. The authors are not aware of any formal study of this phenomenon.[2]
- There are some exciting new results of *feedback linearization* from the field of geometric control theory. In many cases, it is possible to construct a preliminary feedback which makes the plant, with the preliminary feedback loop closed, linear and time-invariant, and thus amenable to the methods of this paper. See for example [33]–[36] and the many references therein.
- We will see in Section VII-D that some of the effects of plant nonlinearities can be accounted for.

[2]Except Lyapunov's classical result that, roughly speaking, the system will work provided the system does not venture far from the region where the linear model is accurate.

- Linear control systems often form the core or basis of control systems designed for nonlinear systems, for example in gain-scheduled or adaptive controllers.

A consequence of our assumption is that the plant can be described by the set of transfer functions from each of its inputs (the components of the vectors w and u) to each of its outputs (the components of z and y), organized into a matrix, the transfer matrix. We will use the symbols P and K to denote the transfer matrices of the plant and controller, respectively. We partition the plant transfer matrix P as

$$P = \begin{bmatrix} P_{zw} & P_{zu} \\ P_{yw} & P_{yu} \end{bmatrix},$$

where P_{zw} is the transfer matrix from w to z, P_{zu} is the transfer matrix from u to z, P_{yw} is the transfer matrix from w to y, and P_{yu} is the transfer matrix from u to y. This decomposition is shown in Fig. 8.

Now suppose the controller is operating, as shown in Fig. 5. We can solve for the closed-loop transfer matrix from w to z, which we denote H:

$$H = P_{zw} + P_{zu}K(I - P_{yu}K)^{-1}P_{yw}. \tag{1}$$

The entries of the transfer matrix H are the closed-loop transfer functions from each exogenous input to each regulated variable. Various entries might represent, for example, closed-loop transfer functions from some disturbance to some actuator, some sensor to some internal variable, and some command signal to some actuator signal. The formula (1) above shows exactly how each of these closed-loop transfer functions depends on the controller K.

A central theme of this paper is that H *should contain every closed-loop transfer function of interest to us.* Indeed, we can arrange for any particular closed-loop transfer function in our system to appear in H, as follows. Consider the closed-loop system in Fig. 9 with two signals A and B which are internal to the plant. If our interest in the transfer function from a signal injected at point A to point B, we need only make sure that one of the exogenous signals injects at A, and that the signal at point B is one of our regulated variables, as shown in Fig. 10.

C. Some Examples

1) Example 1: A regulator: Consider the setup for a regulator design for a single-actuator, single sensor system. In a classical terminology the transfer function from the actua-

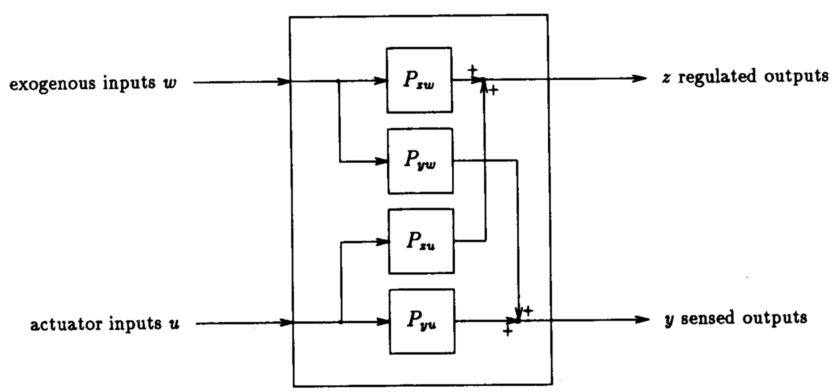


Fig. 8. The decomposed linear plant.

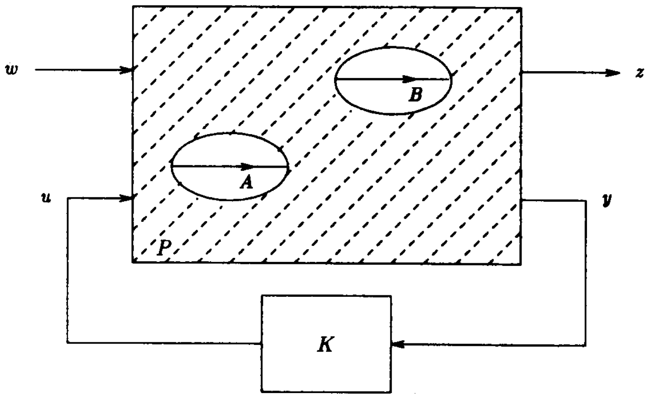


Fig. 9. Two signals A and B internal to the plant P.

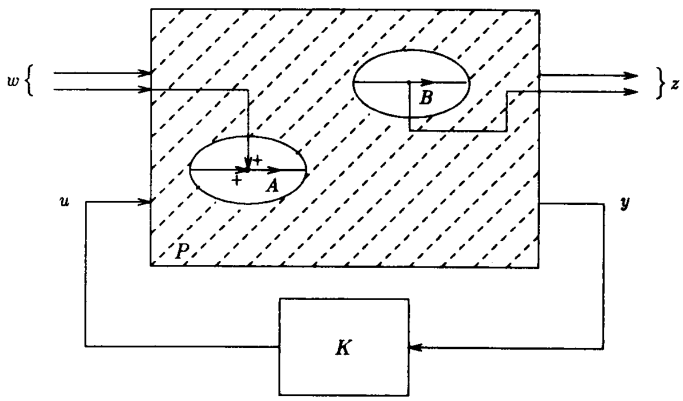


Fig. 10. Accessing internal signals A and B from w and z.

tor input to the sensed output, P_o, is the plant. As we shall see, in our framework the plant consists of P_o *and* several interconnections.

We measure the performance of the regulator by the size of the actuator signal and the output signal y_p in the face of sensor noise and actuator referred process noise. Thus, we take the exogenous input vector to consist of the process noise or disturbance d_{proc} and the sensor noise d_{sens}

$$\omega = \begin{bmatrix} d_{\text{proc}} \\ d_{\text{sens}} \end{bmatrix}$$

and we take the vector of regulated outputs to consist of the plant output y_p and the actuator signal:

$$z = \begin{bmatrix} y_p \\ u \end{bmatrix}.$$

The control input of the plant is just the actuator signal u, and the sensed output will consist of the output signal corrupted by the sensor noise:

$$y = y_p + d_{\text{sens}}.$$

The plant, which has three inputs and three outputs, is shown in Fig. 11. The plant transfer matrix is

$$P = \left[\begin{array}{c|c} P_{zw} & P_{zu} \\ \hline P_{yw} & P_{yu} \end{array}\right] = \left[\begin{array}{cc|c} P_0 & 0 & P_0 \\ 0 & 0 & 1 \\ \hline P_0 & 1 & P_0 \end{array}\right].$$

The block diagram for the closed-loop system is shown in Fig. 12. From equation (1), closed-loop transfer matrix H from w to z is

$$H = \begin{bmatrix} P_0/(1 - P_0K) & P_0K/(1 - P_0K) \\ P_0K/(1 - P_0K) & K/(1 - P_0K) \end{bmatrix}.$$

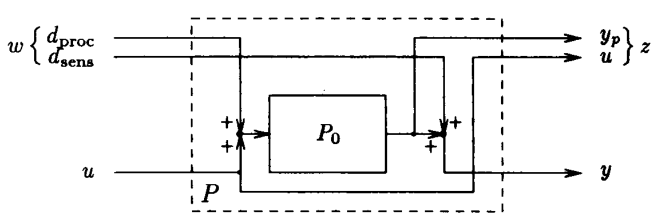


Fig. 11. The plant for a single input, single output transfer function P_0.

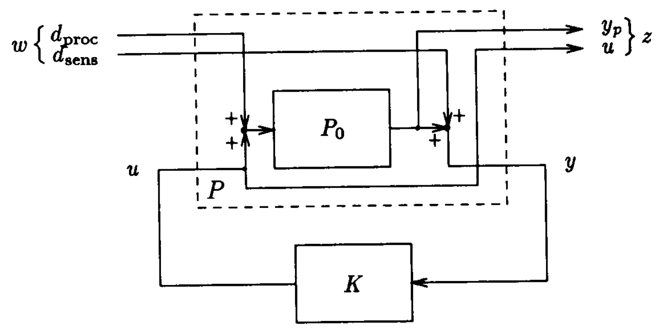


Fig. 12. The closed-loop regulator.

Each entry of H, the 2×2 closed-loop transfer matrix from w to z, has its own significance and importance. The first row consists of the closed-loop transfer functions from the process and sensor noises to the output signal y_p; our goal is to make these two transfer functions "small" in some appropriate sense. The "size" of these two transfer functions tells us something about the closed-loop *regulation* achieved by our controller.

The second row consists of the closed-loop transfer functions from the process and sensor noises to the actuator signal, and thus relate to the actuator authority our controller uses.

The idea is that H contains all the closed-loop transfer functions of interest in our regulator design. Thus, the performance of different candidate controllers could be compared by their associated H's, using (1).

2) Example 2: Tracking a command signal: Consider the previous example with the injection of a reference signal r. Now our goal is to keep the output variable y_p close to the reference input r. We will augment the exogenous input vector of the previous example to include r:

$$w = \begin{bmatrix} d_{\text{proc}} \\ d_{\text{sens}} \\ r \end{bmatrix}.$$

Assuming that the controller has access to the reference signal r as well as $y_p + d_{\text{sens}}$, we take

$$y = \begin{bmatrix} y_p + d_{\text{sens}} \\ r \end{bmatrix}.$$

The new plant, shown in Fig. 13, now has four inputs and four outputs. The transfer matrix of the plant is

$$P = \left[\begin{array}{c|c} P_{zw} & P_{zu} \\ \hline P_{yw} & P_{yu} \end{array}\right] = \left[\begin{array}{ccc|c} P_0 & 0 & 0 & P_0 \\ 0 & 0 & 0 & 1 \\ \hline P_0 & 1 & 0 & P_0 \\ 0 & 0 & 1 & 0 \end{array}\right].$$

The closed-loop system is shown in Fig. 14. This arrangement is the most general way to inject the reference signal

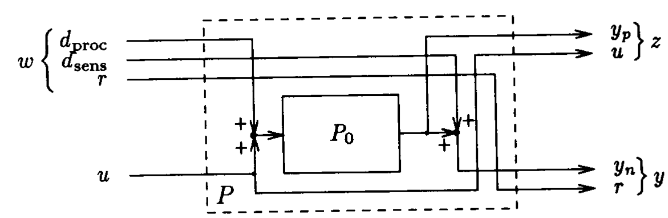


Fig. 13. The plant for a single input, single output transfer function P_0 with a reference input.

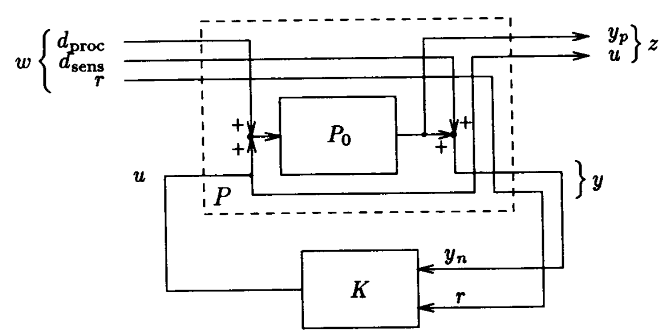


Fig. 14. The closed-loop two degree of freedom system.

r into the regulator system of example 1. The controller K has two inputs and one output. If we write its transfer matrix as[3]

$$K = [K_1 \quad K_2],$$

then the closed-loop transfer matrix H is

$$H = \begin{bmatrix} P_0/(1 - P_0K_1) & P_0K_1/(1 - P_0K_1) & P_0K_2/(1 - P_0K_1) \\ P_0K_1/(1 - P_0K_1) & K_1/(1 - P_0K_1) & K_2/(1 - P_0K_1) \end{bmatrix}.$$

The closed-loop transfer matrix H of this example consists of the closed-loop transfer matrix of example 1 with a third column appended. The third column consists of the closed-loop transfer functions from the reference signal to the output signal y_p, and the actuator u.

The controller in Fig. 14 is two-input single-output, whereas the controller of example 1 is single-input single-output. The controller in this example is known as a *two-degree-of-freedom-controller*, since the controller takes two inputs.

IV. Geometry of Design Specifications

The Fundamental Problem posed in section I-B is to find a control law that meets the design specifications or determine that none exists. As described in section I-C our focus shifts to finding a particular closed-loop transfer matrix H which meets the design specifications, or determining that none exists. Thus the restated Fundamental Problem is:

> **Fundamental Problem (restatement 1):** Given a specific plant P and a set of design specifications, either find a closed-loop transfer matrix H that meets these design specifications, or determine that none exists.

In the previous section we arranged for *every* closed-loop transfer function of interest to be available as some element of the closed-loop transfer matrix H. In this way all design specifications can be expressed as requirements on the closed-loop transfer matrix H.

In this section we will show how each design specification can be interpreted as a set of allowable transfer matrices. In many cases (as discussed in sections V, VI, and VII), the set of allowable transfer matrices H corresponding to a particular design specification has a very simple geometric form. This allows the revised Fundamental Problem (and hence the Fundamental Problem) to be solved numerically (see section IX).

A. Design Specifications as Sets

The number of exogenous inputs (n_{exog}) and regulated outputs (n_{reg}) is fixed by the choice of control configuration. Let $\mathcal{H}$ denote the set of *all* $n_{reg} \times n_{exog}$ transfer matrices.

With each design specification $\mathfrak{D}_i$ we will associate the set $\mathcal{H}_i \subseteq \mathcal{H}$ of *all* $n_{reg} \times n_{exog}$ *transfer matrices that meet the design specification* $\mathfrak{D}_i$. For example, consider the specification that the closed-loop transfer function from w_2 (a command input, perhaps) to z_3 (the variable it is supposed to command, perhaps) should have unity DC gain. The corresponding set of acceptable transfer matrices is

$$\mathcal{H}_{DC} = \{H \in \mathcal{H} \mid H_{32}(0) = 1\}. \tag{2}$$

Another example is the specification that H should be the closed-loop transfer matrix achieved by a controller K which results in a closed-loop stable system:

$$\mathcal{H}_{stab} = \{H \in \mathcal{H} \mid H = P_{zw} + P_{zu}K(I - P_{yu}K)^{-1}P_{yw} \text{ for some stabilizing } K\}. \tag{3}$$

This important specification is the topic of section V.

Simultaneous satisfaction of design specifications corresponds to the *intersection* of the associated sets of acceptable closed-loop transfer matrices. Continuing the examples in equations (2) and (3) above, $\mathcal{H}_{stab} \cap \mathcal{H}_{DC}$ consists of those transfer matrices that arise as the closed-loop transfer matrix of our plant with a stabilizing controller *and* have unity DC gain from w_2 to z_3. This set could very well be empty—which would mean that the two specifications are *inconsistent* or too tight. In practical terms, $\mathcal{H}_{stab} \cap \mathcal{H}_{DC} = \emptyset$ means that no stabilizing controller can yield a DC gain from w_2 to z_3 which equals one.

If $\mathcal{H}_1$ and $\mathcal{H}_2$ correspond to specifications $\mathfrak{D}_1$ and $\mathfrak{D}_2$, and $\mathcal{H}_1 \subseteq \mathcal{H}_2$, then specification $\mathfrak{D}_1$ is *stronger* or *tighter* than specification $\mathfrak{D}_2$ (equivalently, $\mathfrak{D}_2$ is weaker or looser than $\mathfrak{D}_1$); there are "fewer" transfer matrices which satisfy $\mathfrak{D}_1$.

If our design specifications are $\mathfrak{D}_1, \cdots, \mathfrak{D}_L$, then the set which corresponds to satisfaction of all the design specifications is

$$\mathcal{H}_{spec} = \mathcal{H}_1 \cap \cdots \cap \mathcal{H}_L.$$

Thus we can restate the Fundamental Problem as:

> **Fundamental Problem (restatement 2):** Find an $H \in \mathcal{H}_{spec}$, or determine that $\mathcal{H}_{spec} = \emptyset$.

B. Affine and Convex Sets

In many cases the sets of transfer matrices associated with design specifications have a simple *geometric form*: *affine* or *convex*.

We remind the reader that $\mathcal{H}$ is a *vector space*. This means that we have a way of adding two of its elements ($n_{reg} \times n_{exog}$ transfer matrices) and multiplying one by a scalar; the vector addition and scalar multiplication must satisfy certain standard rules [37], [38].

Definition 3 A subset of a vector space is said to be *affine* if whenever two distinct points are in the set, so is the entire line passing through them. More precisely, $\mathcal{C} \subseteq V$ is affine if for any $v_1, v_2 \in \mathcal{C}$, and any $\lambda \in \mathbf{R}$, $\lambda v_1 + (1 - \lambda)v_2 \in \mathcal{C}$.

Definition 4 A subset of a vector space is *convex* is whenever two points are in the set, so is the entire line segment between them. More precisely, $\mathcal{C} \subseteq V$ is convex if for any $v_1, v_2 \in \mathcal{C}$, and any $\lambda \in [0, 1]$, $\lambda v_1 + (1 - \lambda)v_2 \in \mathcal{C}$.

Of course, it is a stronger condition for a set to be affine than convex: all affine sets are convex. If $\lambda \in \mathbf{R}$, we will refer to $\lambda v_1 + (1 - \lambda)v_2$ as an *affine combination* of v_1 and v_2; geometrically, we may think of an affine combination of two points as being on the line passing through the points. If $0 \le \lambda \le 1$, we will refer to $\lambda v_1 + (1 - \lambda)v_2$ as a *convex combination* of v_1 and v_2; geometrically, we may think of a convex combination of two points as being on the line segment between the points.

[3]Note that for stability reasons it is not always possible to implement the controller as two separate transfer functions K_1 and K_2.

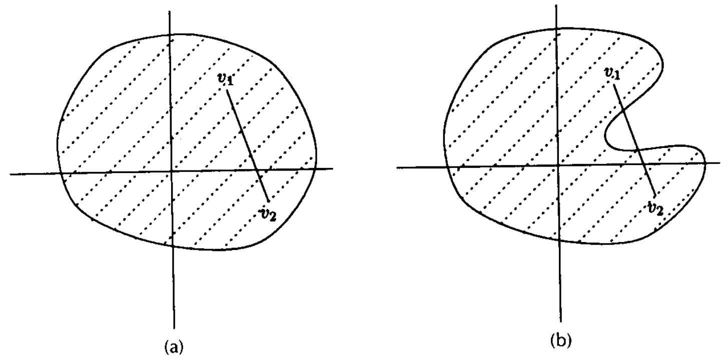


Fig. 15. Examples of (a) convex and (b) non convex subsets of $\mathbf{R}^2$.

An example of a convex subset of the vector space $\mathbf{R}^2$ is shown in Fig. 15(a). The subset of $\mathbf{R}^2$ shown in Fig. 15(b) is not convex.

An example of an affine subset of the vector space $\mathbf{R}^2$ is any straight line

$$\{x \in \mathbf{R}^2 | v^T x = c\}.$$

The set $\mathcal{H}_{DC}$ in (2) is affine, although this is harder to visualize, since $\mathcal{H}$ is infinite dimensional. This assertion is readily verified: if $H \in \mathcal{H}$ and $\tilde{H} \in \mathcal{H}$ satisfy $H_{32}(0) = \tilde{H}_{32}(0) = 1$, and λ is any real number, then the transfer matrix $H_\lambda = \lambda H + (1 - \lambda)\tilde{H}$ also satisfies $H_{\lambda 32}(0) = 1$. We can view H_λ as a transfer matrix on the "line" through the transfer matrices H and $\tilde{H}$: that $\mathcal{H}_{DC}$ is affine means that the DC gain of the 3, 2 entry of all such transfer matrices is one.

We will show in section V that the example in equation (3)—achievability by a stabilizing controller—is also affine.

Definition 5 A design specification is *closed-loop convex* if the set of transfer matrices that satisfy it is convex.

One of the themes of this paper is that *many design specifications are closed-loop convex.*

C. The Time and Frequency Domains

Design specifications about time domain quantities (such as impulse or step responses) will also correspond to sets of acceptable transfer matrices. For example, consider the specification that the overshoot of the step response from w_2 to z_3 not exceed 1.3:

$$\mathcal{H}_{OS} = \{H \in \mathcal{H} | s_{32}(t) \le 1.3 \text{ for all } t \ge 0\}, \tag{4}$$

where $s_{32}(t)$ is the inverse Laplace transform of $H_{32}(s)/s$.

The set $\mathcal{H}_{OS}$ might be called "hard to describe" in "frequency-domain terms": there is no simple description of $\mathcal{H}_{OS}$ in terms of pole and zero locations of H_{32} (although there are many approximate descriptions and rules-of-thumb). The overshoot specification seems more naturally expressed in "time-domain terms" (the step response matrix, say).

Nevertheless, we can easily verify that $\mathcal{H}_{OS}$ is convex (and thus, the overshoot specification is closed-loop convex), even if it is "hard to describe." We rewrite (4) more explicitly as

$$\mathcal{H}_{OS} = \left\{ H \in \mathcal{H} \middle| \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{H_{32}(j\omega)}{j\omega} e^{j\omega t}\, d\omega \le 1.3 \text{ for all } t \ge 0 \right\}. \tag{5}$$

Suppose that H and $\tilde{H}$ are transfer matrices in $\mathcal{H}_{OS}$, and let $H_\lambda = \lambda H + (1 - \lambda)\tilde{H}$, where $0 \le \lambda \le 1$. Then for all $t \ge 0$,

$$\begin{aligned} &\frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{H_{\lambda 32}(j\omega)}{j\omega} e^{j\omega t}\, d\omega \\ &= \lambda \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{H_{32}(j\omega)}{j\omega} e^{j\omega t}\, d\omega \\ &\quad + (1 - \lambda) \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{\tilde{H}_{32}(j\omega)}{j\omega} e^{j\omega t}\, d\omega. \end{aligned}$$

The right-hand side of this last equation is a convex combination of two real numbers, neither of which exceeds 1.3, and thus the left-hand side also does not exceed 1.3. This means that $H_\lambda \in \mathcal{H}_{OS}$.

More generally, and less explicitly, since the Laplace transform and its inverse are linear operations, convex and affine sets are preserved between the frequency and time domains. The property of convexity is independent of whether a constraint is expressed in the time or frequency domains. It is also independent of whether the constraint is "more easily" or "naturally" expressed in the time or frequency domains.

Other time domain design specifications may involve the response to a particular input signal. Consider the response $z_3(t)$ to a fixed input signal $w_2(t)$ ($w_i(t) = 0$ for $i \ne 2$),

$$z_3(t) = \int_0^t h_{32}(t - \tau)\, w_2(\tau)\, d\tau.$$

Since this convolution integral is linear, a convex or affine constraint on z_3 will correspond to a convex or affine constraint on the impulse response h_{32}, and therefore on the closed-loop transfer function H_{32}. This will be discussed more fully in section VI.

V. Realizability and Closed-Loop Stability

In this section we consider the important design requirement of *internal (closed-loop) stability*. The central result is that the set $\mathcal{H}_{stab}$ of closed-loop transfer matrices achievable with controllers that stabilize the plant is an affine set which is readily described. This description is referred to as the *parametrization of achievable closed-loop transfer matrices*. Thus, *internal stability is a closed-loop convex constraint*.

Various preliminary forms of this result, for special cases, can be found or are implicit in work dating back to 1950.

Its modern, general form appears in Desoer, Liu, Murray and Saeks, in 1980 [39]. Section XI contains some detail on the evolution of the idea, along with a more complete set of references.

A. Closed-Loop Transfer Matrices Achieved by Controllers

A very important constraint on the transfer matrix H is that *it should be the closed-loop transfer matrix achieved by some controller* K, in other words, H should have the form $P_{zw} + P_{zu}K(I - P_{yu}K)^{-1}P_{yw}$ for some K. We will refer to this constraint as *realizability*:

$$\mathcal{H}_R = \{H \in \mathcal{H} | H = P_{zw} + P_{zu}K(I - P_{yu}K)^{-1}P_{yw} \text{ for some } K\}. \quad (6)$$

This constraint is usually quite strong, meaning that $\mathcal{H}_R$ is in general a "small" subset of $\mathcal{H}$; there are many transfer matrices $H \in \mathcal{H}$ which do not correspond to any controller. To understand what this constraint means, consider a plant with one actuator (scalar u) and one sensor (scalar y), and w and z each with 4 components, so that H is a 4×4 transfer matrix. It seems intuitively clear that since we can design only one transfer function (K), we should therefore have one one "degree of freedom" in the resulting closed-loop transfer matrix H, which contains 16 closed-loop transfer functions. This intuition is correct: examination of (6) shows that at each frequency s, choice of $K(s)$ yields only a rank one change in H.

We can think of $\mathcal{H}_R$ as expressing the *dependencies* among the various closed-loop transfer functions (entries of H). For example, suppose that the closed-loop transfer functions $S = 1/(1 - P_0K) = H_{11}$ and $T = P_0K/(1 - P_0K) = H_{21}$ are two of the entries of H in our example above. The constraint $H \in \mathcal{H}_R$ will include the well known and obvious constraint $H_{11} - H_{21} = 1$; in classical terminology, the sensitivity and complementary sensitivity must sum to one.

B. Internal Stability

1) The idea of internal stability: Recall that a rational transfer function is *stable* if it has no more zeros than poles and if each pole has negative real part; a transfer matrix is stable if all of its entries are stable transfer functions.

There are several ways to express the idea that the closed-loop system is *internally stable*:

1. *All internal transfer functions are stable.* The transfer function from a signal injected anywhere in the closed-loop system to any other point in the closed-loop system (as shown in Fig. 9) is stable.
2. *The state-space description of the closed-loop system is stable.* If the state space description of the closed-loop system is $\dot{x} = A_{cl}x + B_{cl}w$, $z = C_{cl}x + D_{cl}w$, then all the eigenvalues of A_{cl} have negative real part.
3. *In classical single-actuator, single-sensor control, there are no unstable pole-zero cancellations between plant and controller.* This form of the idea of internal stabilization is the oldest, predating the extensive use of state-space descriptions and concepts such as controllability and observability. For multi-actuator, multi-sensor systems it is much harder to say what a pole-zero cancellation is—it is possible that a multi-actuator, multi-sensor system is internally stable even if the plant has a pole at $s = 1$ and the controller a zero at $s = 1$.

2) Desoer's formal definition: In 1975 Desoer and Chan [40] proposed a formal definition of internal stability that captures the ideas listed above. Their definition has been widely used since [41, pp. 15–17], [42, pp. 99–108]. It makes use of two auxiliary signals ν_1 and ν_2, injected as shown in Fig. 16. The signal ν_1 can be interpreted as an actuator

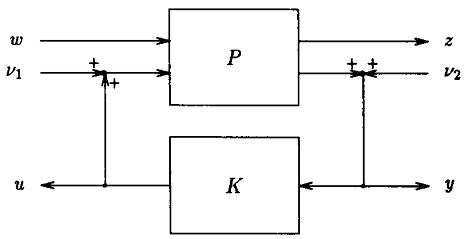


Fig. 16. Sensor and actuator noises used in formal definition of internal stability.

referred plant noise or actuator noise; ν_2 can be interpreted as a sensor noise. The definition is:

Definition 6 The closed-loop system is *internally stable* if the four closed-loop transfer matrices from ν_1 and ν_2 to u and y,

$$H_{u\nu_1} = K(I - P_{yu}K)^{-1}P_{yu},$$
$$H_{u\nu_2} = K(I - P_{yu}K)^{-1},$$
$$H_{y\nu_1} = (I - P_{yu}K)^{-1}P_{yu},$$
$$H_{y\nu_2} = (I - P_{yu}K)^{-1},$$

are stable.

The intuition behind this definition is clear: it prohibits a very small (bounded) actuator or sensor noise from having very large (unbounded) effect on u or y, and therefore z.

Under the assumption that the realizations of P and K have no unstable hidden modes, that is, modes uncontrollable from u or unobservable from y, this is equivalent to ideas (1–3) above [43]. When the closed-loop system is internally stable, we say the controller K *stabilizes* the plant P.

A simple example follows. Consider the system shown in Fig. 17 (with w and z omitted).

For $K_1 = -2$, the four closed-loop transfer functions are

$$H_{u\nu_1} = \frac{-2}{s+1},$$
$$H_{u\nu_2} = \frac{-2(s-1)}{s+1},$$

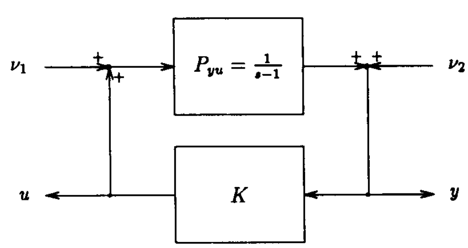


Fig. 17. An example of stabilizing a simple plant.

$$H_{y\nu_1} = \frac{1}{s+1},$$

$$H_{y\nu_2} = \frac{s-1}{s+1}.$$

Since each of these transfer functions is stable, K_1 stabilizes the plant P. The reader may verify (1-3) for this closed-loop system.

On the other hand consider the controller $K_2 = -(s-1)/(s+1)$. The closed-loop transfer functions are now

$$H_{u\nu_1} = \frac{-1}{s+2},$$

$$H_{u\nu_2} = -\frac{s-1}{s+2},$$

$$H_{y\nu_1} = \frac{s+1}{(s+2)(s-1)},$$

$$H_{y\nu_2} = \frac{s+1}{s+2}.$$

For this controller we find that three of the closed-loop transfer functions are stable, but $H_{y\nu_1}$, the transfer function from actuator noise to plant output, is unstable. Thus, K_2 does not stabilize P.

It is instructive to consider the closed-loop system with the controller K_2 in terms of the ideas (1-3) above. (1) is clearly violated—we take the injected signal to be actuator noise and the tapped signal to be the plant output y. (2) is violated; the reader can verify that a state space description of the closed-loop system will be unstable—A_{cl} will have an eigenvalue of 1. (3) is violated by the unstable plant pole at $s = 1$ being canceled by the controller zero at $s = 1$.

3) Internal stability in our framework: It should be clear that, by including ν_1 and ν_2 as parts of the exogenous input w and including u and y as part of the regulated output z, the four transfer matrices of the Desoer-Chan definition appear as blocks within the overall closed-loop transfer matrix H. Assuming that the realizations of P and K have no unstable hidden modes, with this inclusion of signals in w and z, the satisfaction of the Desoer-Chan conditions is equivalent to stability of H. This is shown in Fig. 18.

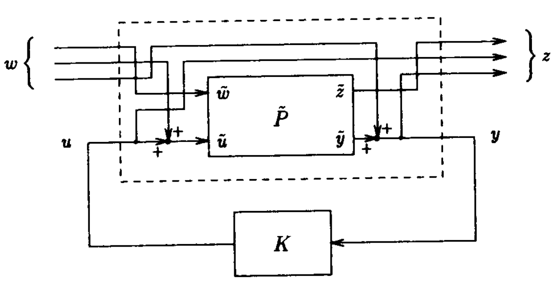


Fig. 18. When sensor and actuator noises are included in w and u and y included in z, internal stability is simply equivalent to stability of the closed loop response H.

In fact, if the definitions of w and z from section III are rigorously followed, then we *must* include u and y in z and sensor and actuator noises (ν_1 and ν_2) in w—the four transfer matrices in the definition of closed-loop stability are closed-loop transfer matrices we care about, and so they should appear in H. Thus the whole issue of internal stability can be seen as arising when one considers an H that is too "small"; H doesn't include all the closed-loop transfer functions of interest in the design. For example, in classical control, the designer might concentrate on the input/output (I/O) transfer function $P_0K/(1 - P_0K)$—our second example above shows, however, that the I/O transfer function can be acceptable, or even desirable, but the design will not work, because another important closed-loop transfer function is unstable. One of our themes is that the designer must simultaneously consider *all* closed-loop transfer functions of interest, in other words, the transfer matrix H.

C. Closed-Loop Transfer Matrices Achievable with Stabilizing Controllers

We now consider the set of closed-loop transfer matrices achieved by controllers that stabilize the plant:

$$\mathcal{H}_{stab} = \{H \in \mathcal{H} \mid H = P_{zw} + P_{zu}K(I - P_{yu}K)^{-1}P_{yw} \text{ for some } K \text{ that stabilizes } P\} \quad (7)$$

Thus $\mathcal{H}_{stab}$ is the set of possible closed-loop transfer matrices achieved with controllers that stabilize P.

An extremely important fact is that internal stability is a closed-loop affine constraint, i.e.,

> $\mathcal{H}_{stab}$ is affine: Any affine combination of closed-loop transfer matrices achievable with stabilizing controllers is also achievable with a stabilizing controller.

If K and $\tilde{K}$ each stabilize P and yield closed-loop transfer matrices H and $\tilde{H}$, respectively, then for each $\lambda \in \mathbf{R}$, there is some controller K_λ that stabilizes P and yields closed-loop transfer matrix $H_\lambda = \lambda H + (1 - \lambda)\tilde{H}$. Thus if we can find *two* controllers that stabilize P we can find an entire *one parameter family* of controllers that stabilize P, and the corresponding closed-loop transfer matrices will lie on a *line* in $\mathcal{H}$.

A very important point is that the controller K_λ that yields closed-loop transfer matrix H_λ is generally *not* $K_\lambda = \lambda K + (1 - \lambda)\tilde{K}$. Straightforward but tedious algebra yields

$$K_\lambda = (A + \lambda B)^{-1}(C + \lambda D) \quad (8)$$

where

$$A = I + \tilde{K}(I - P_{yu}\tilde{K})^{-1}P_{yu},$$

$$B = K(I - P_{yu}K)^{-1}P_{yu} - \tilde{K}(I - P_{yu}\tilde{K})^{-1}P_{yu},$$

$$C = \tilde{K}(I - P_{yu}\tilde{K})^{-1},$$

$$D = K(I - P_{yu}K)^{-1} - \tilde{K}(I - P_{yu}\tilde{K})^{-1}.$$

The special form of K_λ given in (8) is called a *bilinear* or *linear fractional* dependence on λ. We will see this form later in this section.

It is not hard to verify directly that $\mathcal{H}_{stab}$ is affine by substituting the formula (8) into the four critical closed-loop transfer matrices of definition 6. The reader will find, for example, that

$$H_{u\nu_2} = K_\lambda(I - P_{yu}K_\lambda)^{-1} = \lambda K(I - P_{yu}K)^{-1} + (1 - \lambda)\tilde{K}(I - P_{yu}\tilde{K})^{-1};$$

since the right hand side is an affine combination of stable transfer matrices, the left hand side is stable.

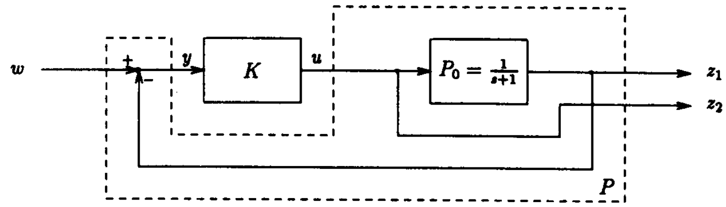


Fig. 19. A possible controller configuration for a stable plant P_0

1) An example: Consider the system shown in Fig. 19. In classical terminology, K is a one degree of freedom controller (see section III). The two PI (proportional-integral) controllers $K(s) = 1.8 + 1.5/s$ and $\tilde{K}(s) = 60 + 3/s$ each stabilize P. With K in the loop, the transfer matrix from w to $[z_1 \;\; z_2]^T$ is

$$H(s) = \begin{bmatrix} \dfrac{1.8s + 1.5}{s^2 + 2.8s + 1.5} \\[2ex] \dfrac{1.8s^2 + 3.3s + 1.5}{s^2 + 2.8s + 1.5} \end{bmatrix},$$

$$\mathcal{H}_{\text{stab}} = \left\{ P_{zw} + P_{zu}K(I - P_{yu}K)^{-1}P_{yw} \;\middle|\; \begin{array}{l} K(I - P_{yu}K)^{-1}P_{yu}, \\ K(I - P_{yu}K)^{-1}, \\ (I - P_{yu}K)^{-1}P_{yu}, \\ (I - P_{yu}K)^{-1} \end{array} \text{ are stable} \right\}. \tag{9}$$

while with $\tilde{K}$ in the loop, the closed-loop transfer matrix is

$$\tilde{H}(s) = \begin{bmatrix} \dfrac{3s + 60}{s^2 + 4s + 60} \\[2ex] \dfrac{3s^2 + 63s + 60}{s^2 + 4s + 60} \end{bmatrix}.$$

From the result above, we conclude that for every $\lambda \in \mathbf{R}$, there is a controller K_λ that stabilizes P and yields a closed-loop transfer matrix of

$$H_\lambda(s) = \lambda \begin{bmatrix} \dfrac{1.8s + 1.5}{s^2 + 2.8s + 1.5} \\[2ex] \dfrac{1.8s^2 + 3.3s + 1.5}{s^2 + 2.8s + 1.5} \end{bmatrix} + (1 - \lambda) \begin{bmatrix} \dfrac{3s + 60}{s^2 + 4s + 60} \\[2ex] \dfrac{3s^2 + 63s + 60}{s^2 + 4s + 60} \end{bmatrix}.$$

The step responses from w to z_1 for K and $\tilde{K}$ are shown in Fig. 20(a) and the corresponding step responses from w to z_2 are shown in Fig. 20(b). Figure 20(c) shows five members of the one parameter family of achievable closed-loop step responses from w to z_1 generated by K and $\tilde{K}$; Fig. 20(d) shows the five corresponding step responses from w to z_2.

For $\lambda = 0.5$, it turns out that

$$H_{0.5}(s) = \begin{bmatrix} \dfrac{2.4s^3 + 38.55s^2 + 143.25s + 90}{s^4 + 6.8s^3 + 72.7s^2 + 174s + 90} \\[2ex] \dfrac{2.4s^4 + 40.95s^3 + 181.8s^2 + 233.25s + 90}{s^4 + 6.8s^3 + 72.7s^2 + 174s + 90} \end{bmatrix},$$

and that this closed-loop transfer matrix is achieved with the controller

$$K_{0.5}(s) = \frac{2.4s^3 + 38.55s^2 + 143.25s + 90}{s^3 + 3.4s^2 + 30.75s}.$$

Note that $K_{0.5}$ is *not* the average of the PI controllers; it is not even a PI controller.

In the next two subsections we discuss two special but important cases in which we can derive a complete description of $\mathcal{H}_{\text{stab}}$.

2) Special case I: stable plant: We first make a useful transformation of our description of $\mathcal{H}_{\text{stab}}$. By definition,

Let us make the definition $R = K(I - P_{yu}K)^{-1}$, so that $K = (I + RP_{yu})^{-1}R$. These formulas describe a one-to-one correspondence between K and R. With some algebraic manipulation we can rewrite equation (9) in terms of R as

$$\mathcal{H}_{\text{stab}} = \{P_{zw} + P_{zu}RP_{yw} | RP_{yu}, R, (I + P_{yu}R), (I + P_{yu}R)P_{yu} \text{ are stable}\}. \tag{10}$$

Now we consider the case when the plant is stable. In particular P_{yu} is stable; it follows that if R is stable, then so are RP_{yu}, $I + P_{yu}R$, and $(I + P_{yu}R)P_{yu}$. Thus the set of achievable closed-loop transfer matrices can be written

$$\mathcal{H}_{\text{stab}} = \{P_{zw} + P_{zu}RP_{yw} | R \text{ stable}\}. \tag{11}$$

Given some stable R, the controller that stabilizes P and yields closed-loop transfer matrix $H = P_{zw} + P_{zu}RP_{yw}$ is [44]

$$K = (I + RP_{yu})^{-1}R. \tag{12}$$

Conversely, every controller that stabilizes P has this form for some stable R. This simple result appears in [45].

Note that equation (11) give a *free parameter representation* for $\mathcal{H}_{\text{stab}}$.

3) Special case II: interpolation conditions: To see what changes when the plant is unstable we first examine the very special case shown in Fig. 21. Here u, y, w, and z are all scalar signals; we assume that P_0 is strictly proper. This is the one-degree-of-freedom controller design problem in classical control—the closed-loop transfer function H is $P_0K/(1 + P_0K)$, the transfer function from the reference input to the output of P_0. In a practical controller design problem we would augment w and z so that we could make specifications relating to actuator authority, disturbance rejection, and robustness, but it turns out that scalar w and z are sufficient to express the specification of stability in terms of conditions on H.

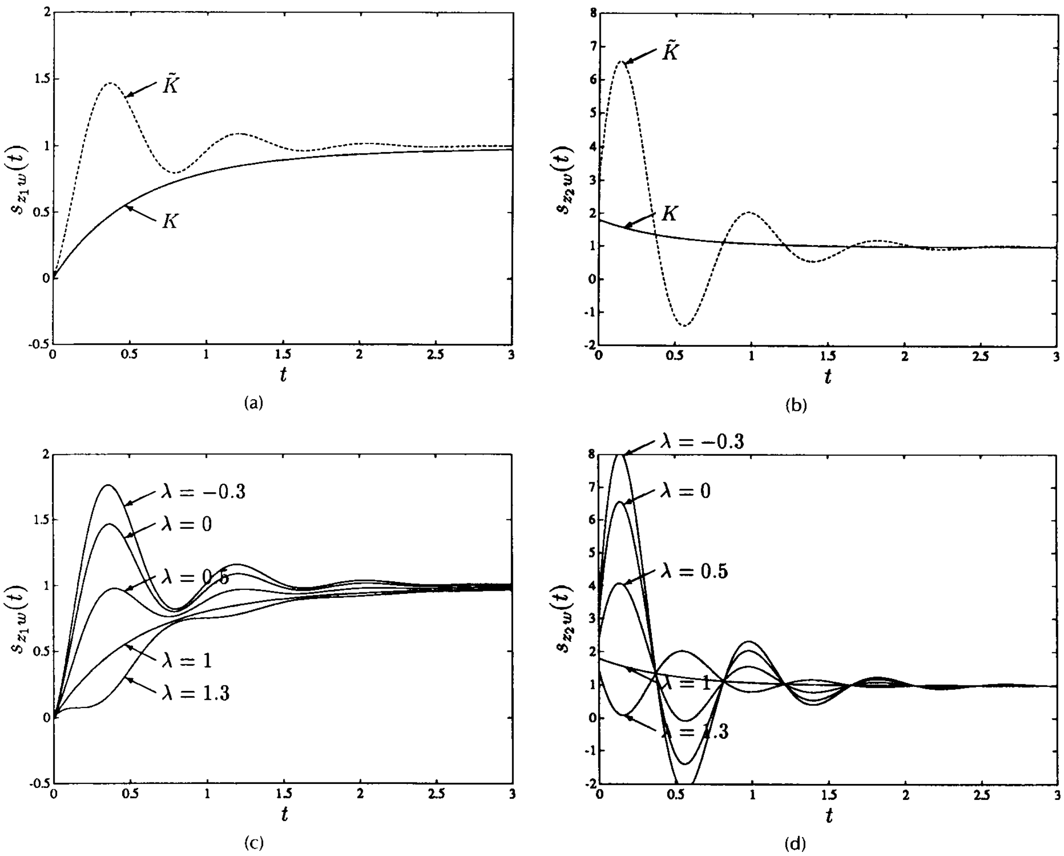


Fig. 20. Closed loop step responses achievable by a one parameter family of stabilizing controllers. (a) and (c) show the step response from the command w_1 to the plant output z_1; (b) and (d) show the step response from the command w_1 to the actuator signal $u = z_2$.

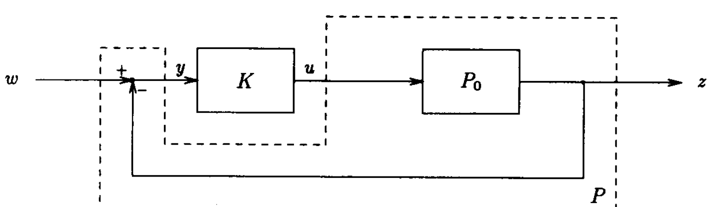


Fig. 21. Classical one degree of freedom setup.

Let $\{p_1, \cdots, p_n\}$ be the set of unstable poles of P_0 and let $\{z_1, \cdots, z_m\}$ be the set of unstable zeros (zeros with nonnegative real parts) of P_0. If P_0 has no repeated unstable poles or zeros, a closed-loop transfer function H is achievable with a stabilizing controller if and only if

1. H is stable,
2. for each unstable plant pole p_i, $H(p_i) = 1$,
3. for each unstable plant zero z_i, $H(z_i) = 0$, and
4. the excess of poles over zeros of H is at least the excess of poles over zeros of P_0.

These conditions are known as the *interpolation conditions*, and can be easily understood in classical control terms. Condition 2 reflects the fact that the loop gain P_0K is infinite at the unstable plant poles (internal stability prohibits the controller from having a zero at the unstable plant poles), and so we should have perfect tracking ($H = 1$) at these frequencies. Conditions 3 and 4 reflect the fact that there is no transmission through P_0 at a frequency where P_0 has a zero, and thus $H = 0$.

The interpolation conditions are also readily understood in terms of our description of $\mathcal{H}_{\text{stab}}$ given in (10). For the plant of this example we have $P_{zw} = 0$, $P_{zu} = P_0$, $P_{yw} = 1$, and $P_{yu} = -P_0$, so (10) is:

$$\mathcal{H}_{\text{stab}} = \{P_0R \mid RP_0,\ R,\ (1 - P_0R)P_0 \text{ are stable}\}. \quad (13)$$

using the fact that if RP_0 is stable, then so is $1 - RP_0$. Substituting $H = P_0R$ into (13) we get

$$\mathcal{H}_{\text{stab}} = \{H \mid H,\ H/P_0,\ (1 - H)P_0 \text{ are stable}\}. \quad (14)$$

H/P_0 will be stable if and only if H vanishes at $z_1, \cdots, z_m$ and in addition H has an excess of poles over zeros at least equal to the excess of poles over zeros of P_0; in other words, H/P_0 is stable if and only if conditions 1, 3, and 4 of the interpolation conditions hold. Similarly, $(1 - H)P_0$ will be stable if and only if $1 - H$ vanishes at $p_1, \cdots, p_n$, which is condition 2 of the interpolation conditions.

It can be seen directly from the interpolation conditions that $\mathcal{H}_{stab}$ is affine, since if $H(p_i) = 1$ and $\tilde{H}(p_i) = 1$, then $\lambda H(p_i) + (1 - \lambda)\tilde{H}(p_i) = 1$, and similarly if $H(z_i) = 0$ and $\tilde{H}(z_i) = 0$, then $\lambda H(z_i) + (1 - \lambda)\tilde{H}(z_i) = 0$.

The interpolation conditions give a *constrained representation* of $\mathcal{H}_{stab}$ for the special system shown in Fig. 21. The interpolation conditions are the earliest description of $\mathcal{H}_{stab}$, dating back at least to 1955 (see section XI).

4) General case: free parameter representation: In the general case there is a *free parameter* description of the set of closed-loop transfer matrices achievable with stabilizing controllers:

$$\mathcal{H}_{stab} = \{T_1 + T_2 Q T_3 | Q \text{ stable}\} \quad (15)$$

where T_1, T_2, and T_3 are certain stable transfer matrices which depend on the plant. Q is referred to as the *parameter* in (15), not in the sense of a real number which is to be designed (e.g., the integrator time constant in a PI controller), but rather in the sense that it is the free parameter in the description (15). We have already seen a special case of this form in the example of the stable plant, where P_{zw}, P_{zu}, and P_{yw} are possible choices for T_1, T_2, and T_3, respectively.

The controller that stabilizes the plant and yields closed-loop transfer matrix $H = T_1 + T_2 Q T_3$ has the linear fractional form

$$K_Q = (A + BQ)^{-1}(C + DQ) \quad (16)$$

where A, B, C, D are certain stable transfer matrices related to T_1, T_2, and T_3. Thus the dependence of K_Q on Q is bilinear (c.f. equation (8)).

A complete derivation of the free parameter description (15) and the formula (16) can be found in Vidyasagar [42] and the references cited in section XI.

D. The Modified Controller Paradigm

The description of $\mathcal{H}_{stab}$ given in the previous section can be given an interpretation in terms of *modifying* a given nominal controller which stabilizes the plant. Given one controller K_{nom} that stabilizes the plant, we can construct a large *family* of controllers that stabilize the plant, just as the formula (8) shows how to construct a one parameter family of controllers that stabilizes the plant.

The construction proceeds as follows:

- We *modify* or *augment* the nominal controller K_{nom} to produce an auxiliary output signal e (of the same size as y) and accept an auxiliary input signal v (of the same size of u) as shown in Fig. 22. This augmentation is done in such a way that the closed-loop transfer matrix from v to e is zero while the open-loop controller transfer matrix from y to u remains simply K_{nom}.
- We connect a stable transfer matrix Q from e to v as shown in Fig. 23, and collect K_{nom} and Q together to form a new controller, K.

The intuition is that K should also stabilize P, since the Q filter we added to K_{nom} "sees no feedback," and thus can-

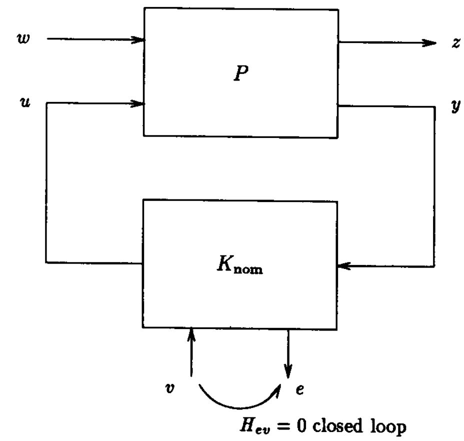


Fig. 22. The nominal controller K_{nom} is augmented to produce a signal e and accept a signal v. The closed loop transfer matrix from v to e is 0.

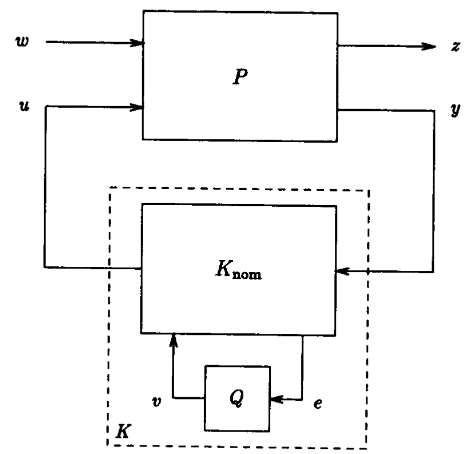


Fig. 23. Modification of nominal controller K_{nom} with a stable transfer matrix Q.

not destabilize our system. However, the Q filter *can* change the closed-loop transfer matrix H. To see how Q affects the closed-loop transfer matrix from w to z, we redraw our system as in Fig. 24.

Let us define the following transfer matrices:

- U_1 is the closed-loop transfer matrix from w to z with the controller K_{nom}.
- U_2 is the closed-loop transfer matrix from v to z.
- U_3 is the closed-loop transfer matrix from w to e.

Since the transfer matrix from v to e is zero, we can redraw Fig. 24 as Fig. 25(a). Figure 25(a) can then be redrawn as Fig. 25(b), which makes it clear that the closed-loop transfer matrix resulting from our modified controller is simply

$$H = U_1 + U_2 Q U_3, \quad (17)$$

which must be stable because Q, U_1, U_2, and U_3 are all stable.

It can be seen from equation (17) that as Q varies over all stable transfer matrices, H sweeps out the following affine set of closed-loop transfer matrices achieved by modifying

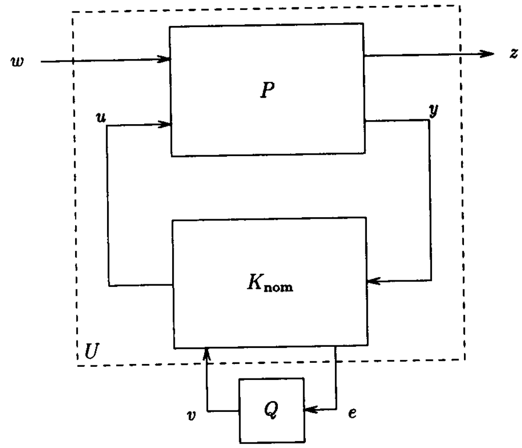


Fig. 24. Finding the closed loop transfer matrices U_1, U_2, and U_3.

the controller:

$$\mathcal{K}_{MCP} = \{U_1 + U_2 Q U_3 | Q \text{ stable}\}.$$

Of course, $\mathcal{K}_{MCP} \subseteq \mathcal{K}_{stab}$. This means that a (possibly incomplete) family of stabilizing controllers can be generated from the (augmented) nominal controller using this *modified controller paradigm*.

The remarkable fact is that if the augmentation of the nominal controller is done with skill, then the modified controller paradigm yields *every* controller that stabilizes the plant P. In other words, if K_{nom} is augmented properly, $\mathcal{K}_{MCP} = \mathcal{K}_{stab}$. In this case U_1, U_2, and U_3 qualify as possible values of T_1, T_2, and T_3 in the free parameter representation of $\mathcal{K}_{stab}$ given in equation (15). Thus we have:

> Given a controller K_{nom} which stabilizes P, suitably modified to produce the signal e and accept the signal v, *every* controller K that stabilizes P can be constructed as a connection of K_{nom} with some suitable stable transfer matrix Q, as shown in Fig. 23.

1) Case I: stable plant: As an example of the modified controller paradigm, consider the special case of the stable plant treated in section V-C(2). Since the plant is stable, the nominal controller $K_{nom} = 0$ stabilizes the plant. How do we modify the zero controller to produce e and accept v? Perhaps the most obvious first step is to take $e = y$ and to add v into u as shown in Fig. 26. This modification results in the closed-loop transfer matrix from v to e being P_{yu}—but the modified controller paradigm requires that the modification be done in such a way that this transfer matrix is zero.

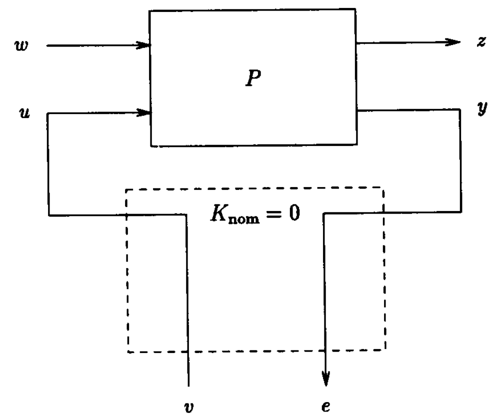


Fig. 26. An incorrect attempt at augmenting the nominal controller $K_{nom} = 0$.

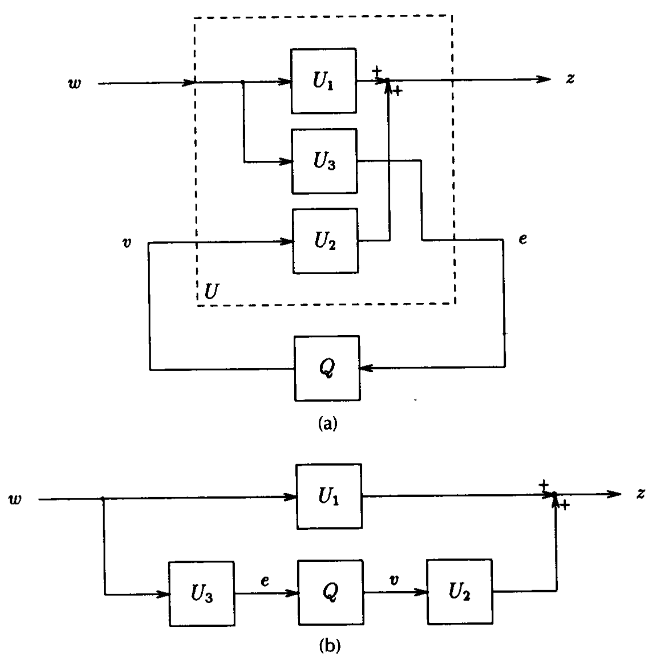


Fig. 25. Figure 24 is redrawn in (a), which is shown in a more convenient form in (b).

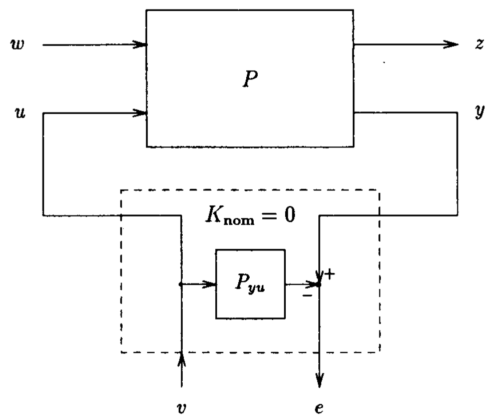


Fig. 27. Extracting e and injecting v in the case where P_{yu} is stable.

A simple remedy is shown in Fig. 27: subtracting v processed through a *copy* of the plant dynamics P_{yu} makes the transfer matrix from v to e zero, as required by the modified controller paradigm. The reader should note that the augmented controller shown in Fig. 27 is much more complex than the nominal controller!

From Fig. 27 we see that

$$U_1 = P_{zw},$$

$$U_2 = P_{zu},$$

$$U_3 = P_{yw}.$$

To apply the second step of the modified controller paradigm, we connect a stable Q as shown in Fig. 28 so that the closed-loop transfer matrix is

$$H = U_1 + U_2 Q U_3.$$

Thus the set of closed-loop transfer matrices achievable by the modified controller shown in Fig. 28 is

$$\mathcal{H}_{MCP} = \{P_{zw} + P_{zu} Q P_{yw} | Q \text{ stable}\}.$$

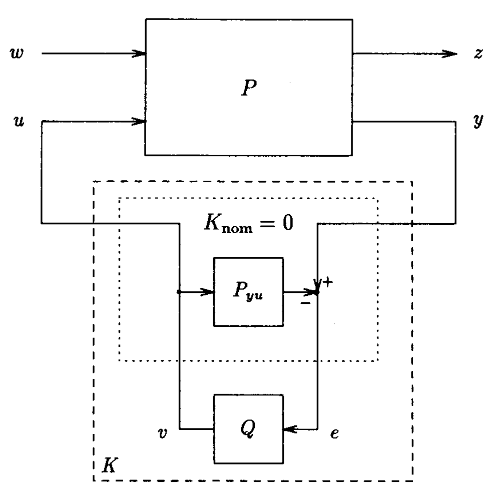


Fig. 28. The modified controller paradigm in the case where P_{yu} is stable.

The expression here for $\mathcal{H}_{MCP}$ is the same as the expression for $\mathcal{H}_{stab}$ in equation (11) in section V-C(2). So here is a case where the modified controller paradigm does indeed generate all stabilizing controllers: any stabilizing controller K for a stable plant P can be implemented with a suitable stable Q as shown in Fig. 28.

The reader can also verify that the connection of Q with the augmented nominal controller yields $K = Q(I + P_{yu}Q)^{-1}$—exactly the same formula as (12) with Q substituted for R.

2) Case II: the observer-based controller: A general method of applying the modified controller paradigm starts with a nominal controller which is an *estimated state feedback.* K_{nom} consists of a full state observer for the plant, which produces an estimate $\hat{x}$ of the plant state, and a state feedback gain K_{SF} (a constant matrix, c.f. K_{nom}!); the controller output is $u = -K_{SF}\hat{x}$. For this nominal controller we can take e to be the *output prediction error* of the observer—the difference between the sensed output y and the corresponding output of the observer, $\hat{y}$. We can add the auxiliary input v to the actuator signal, *before the observer tap,* as is shown in Fig. 29. The requirement that the closed-loop

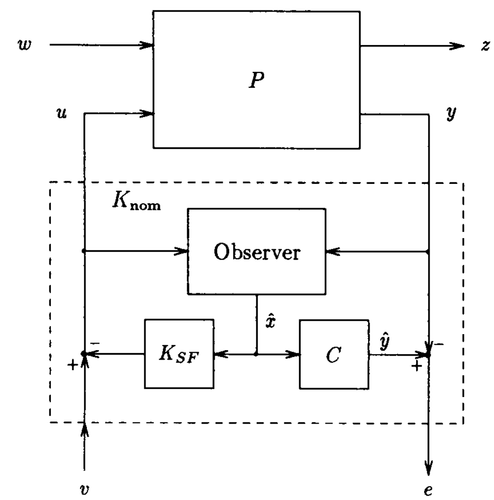


Fig. 29. Augmenting an observer based controller so that it produces e and accepts v.

transfer matrix from v to e be zero is satisfied because the observer error state, $x - \hat{x}$, is uncontrollable from v, and therefore the transfer matrix from v to $e = y - \hat{y}$ is zero.

Applying the modified controller paradigm to the estimated state feedback controller yields the *observer-based controller* shown in Fig. 30. The observer-based controller is just an estimated state feedback controller, with the output prediction error filtered by a stable transfer matrix Q and added to the actuator signal.

Remarkably, this method is one of those which produces *every* controller that stabilizes the plant. This was first pointed out by Doyle [46].

E. Comparison with Parametrized/Structured Controller

So far this section has outlined the answer to the question, "How can we describe the set of all closed-loop transfer matrices achieved by controllers that stabilize the

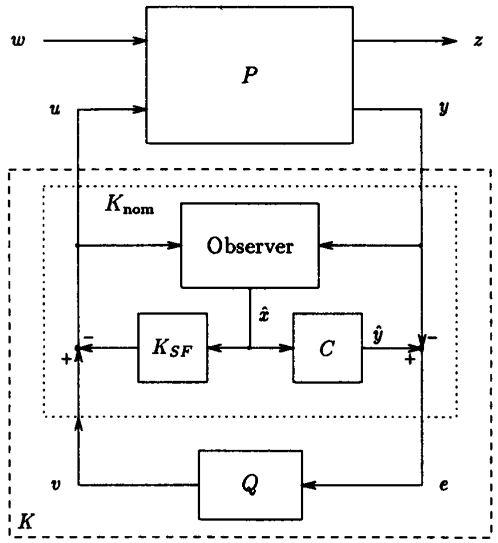


Fig. 30. Modifying an observer based controller with Q. Here K_{SF} is a stabilizing state feedback gain and $y = Cx$, where x is the plant state.

plant?" We have seen that this question has a fairly simple answer, given by equation (15). We also saw that the related question, "How can we describe the set of all controllers that stabilize the plant?" also has a simple answer, given by equation (16), which gives a description of all controllers, of any order or structure, that stabilize the plant. In this subsection we will consider an example of what happens when a particular controller structure is specified and the question becomes, "How can we describe the set of all controllers that stabilize the plant and are of the specified structure?" In some ways this question seems more natural than the one we have answered, "How can we describe the set of *all* controllers that stabilize the plant?," and intuition suggests that it might be easier to solve, since it is a less general question than the one we have answered. In fact, it is a far more difficult question to answer.

Consider the problem of finding a value of the parameters that yields a stable closed-loop system with a particular plant P_{yu}. More specifically, assume we choose a fixed form of the controller, such as

$$K(s) = \frac{\sum_{i=0}^{n} b_i s^i}{s^n + \sum_{i=0}^{n-1} a_i s^i}$$

and then try to find a value of the parameter vector $(a_0, \cdots, a_{n-1}, b_0, \cdots, b_n)$ that will give a closed-loop stable system, or determine that none exists. This is a very difficult problem to solve, and no general solutions are known.

To illustrate this difficulty, we consider an example. Consider a plant P_{yu} with transfer function

$$P_{yu}(s) = \frac{9s^3 + 8s^2 + 9s + 8}{s^4 + 9s^3 - 4s^2 + 8s - 6}$$

and a two parameter controller K with transfer function

$$K(s) = \frac{-k}{s - p}.$$

The closed-loop system is shown in Fig. 31.

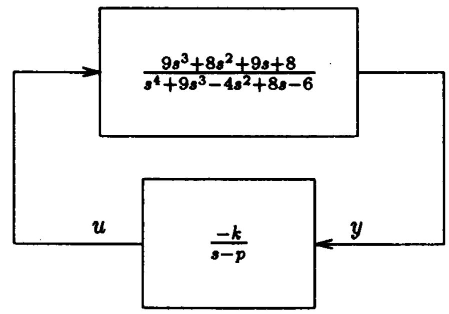


Fig. 31. Stability with a simple two parameter controller.

The condition that the closed-loop system in Fig. 31 is internally stable is equivalent to the polynomial

$$s^5 + (9 - p)s^4 + (-9p + 9k - 4)s^3 + (4p + 8k + 8)s^2 + (-8p + 9k - 6)s + 6p + 8k$$

having all its roots in the left half plane. This condition can be checked using a Routh test. The Routh table shows that internal stability is equivalent to the parameter vector (k, p) satisfying the following inequalities

$$9 - p > 0$$

$$9p^2 - (9k + 81)p + 73k - 44 > 0$$

$$44p^3 + (27k - 396)p^2 - (72k^2 + 274k + 230)p + 584k^2 - 425k + 134 > 0$$

$$134p^4 - (144k + 1206)p^3 + (9k^2 + 1032k - 536)p^2 + (80k^2 + 1204k - 1072)p - 1241k^2 + 1916k - 804 > 0$$

$$6p + 8k > 0$$

In general it is a difficult problem to determine whether a system of polynomial inequalities in several variables such as this can be satisfied.[4] For our particular case the region of parameters which give closed-loop stability is shown in Fig. 32. Notice that there are two disconnected regions in Fig. 32, and each region is itself nonconvex. Figure 32 was produced manually by solving the above equations for k in terms of p and finding the intersection of each of the inequalities; the equations for the boundaries are quite complex. This extremely specific problem, with only two free parameters, required considerable effort to solve. A more general case, where the controller K depends upon a much

[4]The difficulty is in fact related to the distinction between finding global minima for convex and nonconvex functions—see section IX. We also note that deciding whether a set of polynomial inequalities can be satisfied is *decidable*, which means that there is some Routh-like test which can be applied to the coefficients [47], [48]. Unfortunately, the size of this Routh-like array is beyond reason, even for small problems.

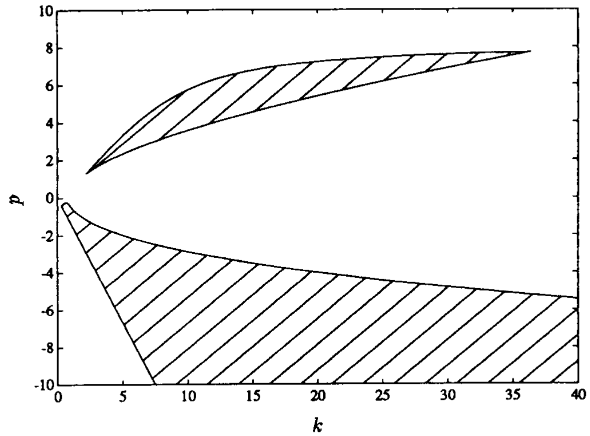


Fig. 32. The region of controller parameters k, p that give a stable closed loop system.

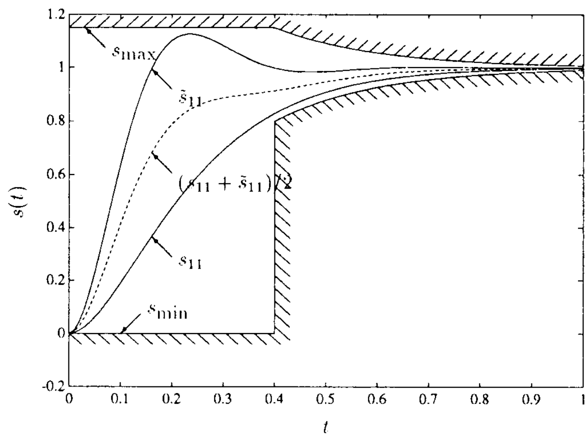


Fig. 33. Step responses with upper and lower bounds.

larger number of parameters, would be very difficult to solve.

VI. Performance Specifications as Convex Constraints

The previous section showed that the performance specification of internal stability is an affine constraint on the closed-loop transfer matrix H. This section introduces specific examples of the remaining performance specifications described in section I-A, and shows why many are closed-loop convex or affine.

For the first few examples we will briefly explain, making reference to the material in section IV, why the constraints are in fact convex. Explanations will not be provided for the remainder of the examples, but it should be apparent from similarity to some earlier examples that the constraint in question is in fact closed-loop convex.

A. Tracking a Reference Input—Time Domain

Often the primary purpose of a feedback control system is to ensure that one of the regulated outputs (usually described as "*the* output" in classical control), say, z_1, be nearly equal to a reference or command input, one of the exogenous inputs, say, w_1. This is a *tracking performance specification*. In many cases the reference input will be constant for long periods of time, and occasionally change quickly to a new value or "set-point". Since the closed-loop system is LTI, its response to such a change in set-point is readily determined from its response when the reference quickly changes from zero to one at $t = 0$—a unit step input.

Let $s_{11}(t)$ denote the unit step response from w_1 to z_1. Let us list some typical qualitative specifications on this step response. Immediately following the step, we are concerned with preventing severe undershoot or overshoot (large negative or positive excursions of $s_{11}(t)$ for small t). We would like a short settling time, that is, we would like the value of $s_{11}(t)$ to converge to within, say, five percent of one. Finally, we would like the $s_{11}(t)$ to settle exactly to one (*asymptotic tracking* of a step input). As shown in Fig. 33, all of these specifications can expressed together as follows:

$$s_{min}(t) \le s_{11}(t) \le s_{max}(t) \qquad \text{for } t \ge 0. \qquad (18)$$

Let us verify that the constraint (18) is closed-loop convex. Consider the set of closed-loop transfer matrices which meet the constraint (18),

$$\mathcal{H}_{track} = \{H \in \mathcal{H} | s_{min}(t) \le s_{11}(t) \le s_{max}(t) \quad \text{for all } t \ge 0\}.$$

Let $H, \tilde{H} \in \mathcal{H}_{track}$, so that the corresponding step responses $s_{11}(t)$ and $\tilde{s}_{11}(t)$ lie between $s_{min}(t)$ and $s_{max}(t)$ for all time as shown in Fig. 33. Now, if $0 \le \lambda \le 1$, then $s_{\lambda 11}(t) = \lambda s_{11}(t) + (1 - \lambda)\, \tilde{s}_{11}(t)$ also lies between $s_{min}(t)$ and $s_{max}(t)$ so that $H_\lambda \in \mathcal{H}_{track}$. The case $\lambda = 0.5$ is shown in Fig. 33.

Placing upper and lower bounds on the response of a regulated output to *any* specific reference input r_0 is also a closed-loop convex specification.

B. Tracking a Reference Input—Frequency Domain

It is also possible to specify how well the regulated variable z_1 tracks the command input w_1 in the frequency domain. Let us define z_2 to be the tracking error, that is, $z_2(t) = w_1(t) - z_1(t)$; H_{21} is therefore the closed-loop transfer function from the command w_1 to the tracking error z_2; roughly speaking, we require that H_{21} should be "small" at those frequencies where the command has significant energy.

Such a specification might take the following form: we require that the magnitude of $H_{21}(j\omega)$ lie below an envelope $l(\omega)$,

$$|H_{21}(j\omega)| \le l(\omega) \qquad \text{for all } \omega, \qquad (19)$$

where $l(\omega)$ is small for $0 \le \omega \le \omega_B$. An example of such a constraint is shown in Fig. 34. We can also interpret the constraint (19) as requiring a maximum asymptotic tracking error of $l(\omega)$ for sinusoidal inputs of frequency ω. The reader can verify that the constraint (19) is closed-loop convex. Figure 34 shows the magnitudes of two closed-loop transfer functions H_{21} and $\tilde{H}_{21}$ which meet this constraint, together with the magnitude of the average transfer function $(H_{21} + \tilde{H}_{21})/2$.

C. Decoupling

A system may have multiple command or reference inputs, each with a corresponding regulated output it is intended to command. It may be required that each of two regulated outputs, say z_1 and z_2, follow its own reference

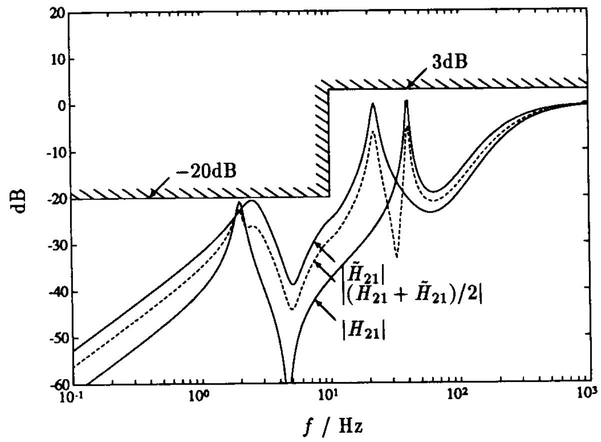


Fig. 34. Upper bounds on frequency response magnitudes are convex constraints. Of course, the magnitude of $(H_{21} + \tilde{H}_{21})/2$ is not the average of the magnitudes of H_{21} and $\tilde{H}_{21}$, although it is no larger than the average.

input, say w_1 and w_2. In this case it is important not only that each output track the appropriate command but also that each output not be much affected by the command for the other output. For example, if an autopilot system is to simultaneously track airspeed and altitude rate commands, a change in commanded airspeed should not severely affect tracking of the commanded altitude rate, and vice versa.

One possible specification is to require that one command has *no* effect on the other regulated outputs, so that the effect of the system of the reference inputs is decoupled. Such a constraint may be

$$H_{12} = 0 \quad \text{and} \quad H_{21} = 0. \tag{20}$$

This constraint is affine.

Complete decoupling may be too strong a constraint. One possible weakening of the specification (20) which limits this interaction for step command signals is shown in Fig. 35. The output signal z_1 is required to track a step applied to w_1, but reject a step applied to w_2, and vice versa for out-

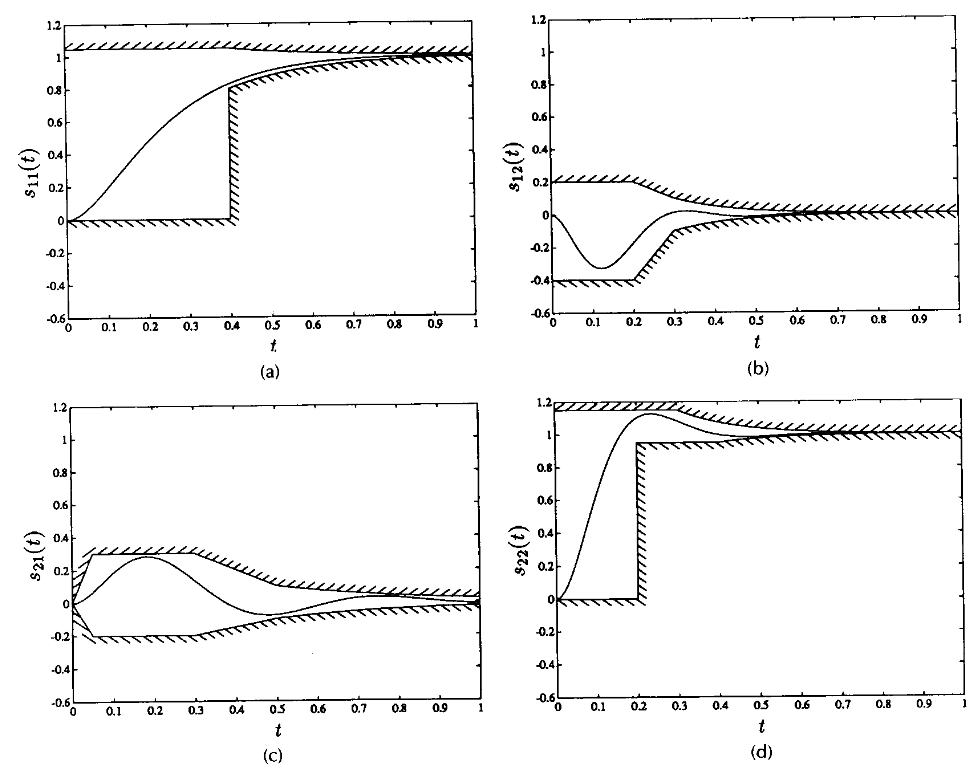


Fig. 35. Design specifications requiring the decoupling of responses to step commands.

put z_2. From section VI-A each of these step response envelope constraints is closed-loop convex.

D. Disturbance Rejection—General Ideas

It is often critical to design a control system that will ensure that the effect on the regulated outputs of noises and disturbances acting on the plant is "small."

This vague description includes a very broad range of performance specifications whose particular forms depend heavily on what the designer knows about the disturbance and what the designer means by "small". In almost all cases the specification of a certain level of performance in disturbance rejection will turn out to be a convex constraint on the closed-loop transfer matrix. The following four subsections will deal with different types of convex constraints arising from disturbance rejection considerations.

In these four subsections, d will refer to a disturbance signal or vector of signals that forms part of the exogenous input vector, while z will refer to part or all of the regulated output vector.

E. Disturbance Rejection—Simple Frequency-Domain Constraints

The specification that the response of z to a constant d tends asymptotically to zero is simply

$$H_{zd}(0) = 0.$$

Similarly, disturbance rejection over a frequency band may be specified:

$$\text{for all } \omega_0 \le \omega \le \omega_1, \qquad |H_{zd}(j\omega)| \le \alpha.$$

Both of these constraints are closed-loop convex.

F. Disturbance Rejection—Small RMS Output

Often a statistical model is assumed for the noises and disturbances entering a system. If these can be modeled collectively as a wide-sense stationary (vector) process it is sometimes reasonable to take as a performance index the root-mean-square value of the component in the output due to the noises and disturbances. If the relevant output is a vector, this RMS output value may be weighted by an appropriate positive definite matrix. It is always possible to incorporate the necessary coloring filters and weight matrices into the plant model in such a way that the input disturbance vector d is white with power spectral density matrix equal to I, while the output vector z is such that the expected (steady-state) value of z^Tz is actually the quantity that the designer would like to keep small. This result appears in most texts on stochastic control or signal processing, for example, [49], [50]; see also [51].

With z and d defined appropriately, the power spectral density of z is given by $H_{zd}(j\omega)\, H_{zd}^T(-j\omega)$. Requiring the output power in z to be less than some maximum can be written as

$$\frac{1}{2\pi} \operatorname{Tr} \int_{-\infty}^{\infty} H_{zd}(j\omega)\, H_{zd}^T(-j\omega)\, d\omega \le a^2.$$

This is a convex constraint on H, since the quadratic form in the integrand is positive-semi-definite.

It should be apparent to readers familiar with LQG design that specifying an upper bound on LQG cost can be posed in this framework. Figure 36 shows how this can be done: the LQG cost is

$$J = \lim_{t\to\infty} E[x^T(t)\, Qx(t) + u(t)^T\, Ru(t)]$$

$$= \frac{1}{2\pi} \operatorname{Tr} \int_{-\infty}^{\infty} H_{zd}(j\omega)\, H_{zd}^T(-j\omega)\, d\omega$$

where the process noise and measurement noise have power spectral density matrices W and V respectively. The *minimum variance* control problem [52] also fits naturally into this framework.

G. Disturbance Rejection—The $\mathbf{H}_\infty$ Criterion

Suppose that we know only that the disturbance input has some power less than one:

$$\lim_{T\to\infty} \frac{1}{T} \int_0^T d(t)^T\, d(t)\, dt \le 1,$$

(we assume that this limit exists). We wish to insure that the power in the output is less than some constant α^2,

$$\lim_{T\to\infty} \frac{1}{T} \int_0^T z(t)^T\, z(t)\, dt \le \alpha^2.$$

Intuition suggests that the effect of the disturbance will be worst if all of its power is concentrated at the frequency at which H_{zd} is largest, and this turns out to be the case. This leads to the convex constraint

$$|H_{zd}(j\omega)| \le \alpha \qquad \text{for all } \omega,$$

in the case that H_{zd} is a transfer function, or

$$\overline{\sigma}(H_{zd}(j\omega)) \le \alpha \qquad \text{for all } \omega,$$

in the case that H_{zd} is a transfer matrix ($\overline{\sigma}(\cdot)$ means *maximum singular value*).

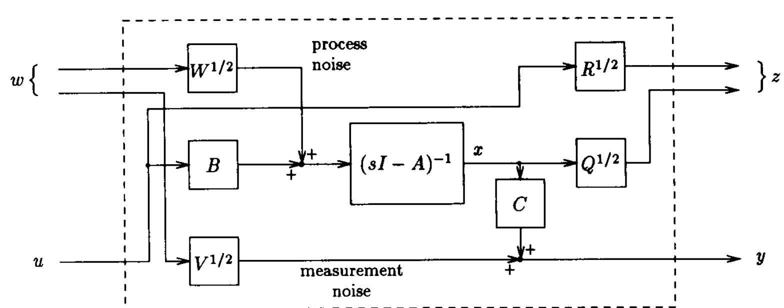


Fig. 36. The LQG cost is the trace of the integral over ω of $1/2\pi\, H(j\omega)H^T(-j\omega)$.

The quantity $\max_{0 \le \omega \le \infty} |H_{zd}(j\omega)|$ or $\max_{0 \le \omega \le \infty} \bar{\sigma}(H_{zd}(j\omega))$ is referred to as the $\mathbf{H}_\infty$-norm of a stable transfer function or matrix, and is denoted $\|H\|_\infty$. (There is a more technical definition of the $\mathbf{H}_\infty$-norm for a broad class of functions on the complex plane, but the one here suffices for rational transfer matrices.) The $\mathbf{H}_\infty$-norm arises in control theory as a measure of disturbance rejection, as shown here, and also in numerous formulations of problems in robust control, as will be shown in section VII. The field of $\mathbf{H}_\infty$ control theory came into existence in the early 1980s; since then much significant work has been done. The design of $\mathbf{H}_\infty$-optimal controllers is presented in [53]. A recent survey paper with a large set of references is [54]. In [51], simple state-space formulas are given for $\mathbf{H}_\infty$-optimal controllers.

H. Disturbance Rejection—Controlling Signal Peaks

In some cases all the designer can confidently say about the disturbances in the system is that their peaks are bounded by some values. If the requirement is that some set of output signals remain within specified bounds when all exogenous inputs other than the disturbances are zero, then what results is a fairly simply-described convex constraint on the closed-loop transfer matrix. Suppose for example that for all t,

$$-5 \le d_1(t) \le 5, \quad -2 \le d_2(t) \le 2, \quad -1/3 \le d_3(t) \le 1/3,$$

and it is required that one component of z due to d_1, d_2, and d_3 be bounded by 1. The resulting constraint is

$$5 \int_0^\infty |h_{zd_1}(t)|\, dt + 2 \int_0^\infty |h_{zd_2}(t)|\, dt + 1/3 \int_0^\infty |h_{zd_3}(t)|\, dt \le 1,$$

where h_{zd_i} is the impulse response from input d_i to output z. This constraint is closed-loop convex. Constraints of this type are sometimes called either $\mathbf{L}_1$ or $\mathbf{L}_\infty$ constraints, since the $\mathbf{L}_1$-norm of a scalar impulse response h is defined as $\int_0^\infty |h(t)|\, dt$ and the $\mathbf{L}_1$ norm of h is the gain from the $\mathbf{L}_\infty$-norm or peak value of the input to the $\mathbf{L}_\infty$-norm or peak value of the output (see, e.g., [55]). The $\mathbf{L}_1$-optimal control problem was posed by Vidyasagar in [56]. Solutions of various versions have been obtained by Dahleh and Pearson [57]–[60].

I. Actuator Authority

It is almost desirable to keep the actuator effort in a linear control system small. By including the actuator signals in the set of regulated outputs, many specifications of limits of actuator use can be translated into convex constraints on the closed-loop transfer matrix. Whenever the specification places bounds on the behavior of actuator signals in response to the behavior of exogenous input signals, convex constraints arise in exactly the same manner as in the preceding sections on tracking and disturbance rejection. We will briefly describe some examples to illustrate this point.

First consider a time-domain constraint on the peak actuator use in response to a step in a reference signal. As shown in Fig. 37, this is setting upper and lower limits for the step response of a regulated output to an exogenous input, and so is no different from the constraint described in section VI-A.

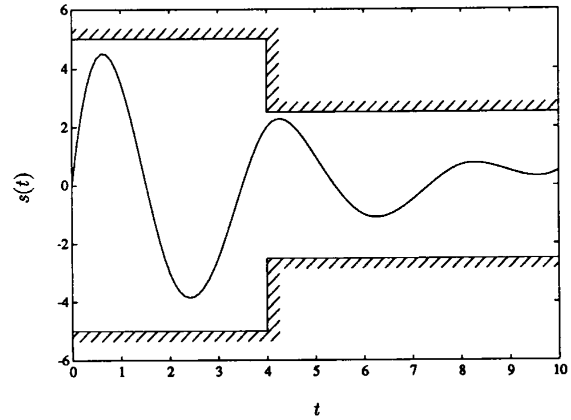


Fig. 37. Limits on actuator authority during a step command input.

Next, consider ensuring that actuator signals remain "small" in the face of noises and disturbances. In our framework this is no different from a "disturbance rejection" specification of one of the types described above.

J. Other Miscellaneous Convex Constraints

So far, we have shown that a very wide variety of performance specifications are equivalent to convex constraints on the closed-loop transfer matrix. Many performance specifications other than those described above also turn out to be convex constraints on the closed-loop transfer matrix. A few such specifications are:

- *Bounds on slew rates.* A specification limiting the slew rate of an actuator signal (or other regulated output) in response to a specified exogenous input signal.
- *ITAE tracking criterion.* An upper bound on the integral of time-multiplied absolute-value of error (ITAE) [61], defined by $\int_0^\infty t|e(t)|\, dt$, where e is the tracking error signal.
- *Higher-order asymptotic tracking.* Specifications that an output asymptotically track ramp or parabolic inputs in addition to step inputs.

VII. Robust Performance Specifications

In the previous section we considered various specifications on how the closed-loop system should perform. This included such important considerations as the response of the system to commands, noises, and disturbances which might rise in the system. In this section we focus on another extremely important consideration: how the system would perform if the plant were to change.

Some authors feel that the primary benefits of feedback are those considered in this section—robustness or insensitivity of the closed-loop system to variations or perturbations in the plant. From another point of view, it is often true that the actual performance of a control system is limited not by its ability to meet the performance specifications of the previous section, but rather by its ability to meet the specifications to be studied in this section, which limit the sensitivity or guarantee robustness of the system [23].

In this section we will show that some, but by no means all, robust performance specifications, can be expressed as closed-loop convex constraints on the *nominal system.*

A. Terminology and Notation

Throughout this section, the nominal plant will be denoted P^{nom}, and the nominal closed-loop transfer matrix will be denoted H^{nom}:

$$H^{\text{nom}} = P_{zw}^{\text{nom}} + P_{zu}^{\text{nom}} K(I - P_{yu}^{\text{nom}} K)^{-1} P_{yw}^{\text{nom}}.$$

The perturbed plant will be denoted P. If the perturbed plant depends on some parameter or transfer function, we may indicate that with a superscript, as in $P^{(\alpha)}$. If the perturbed plant P is still linear and time-invariant, then H will denote the perturbed closed-loop transfer matrix,

$$H = P_{zw} + P_{zu} K(I - P_{yu} K)^{-1} P_{yw}; \tag{21}$$

if the perturbed plant is not linear and time invariant, then of course (21) does not make sense.[5]

B. Small Differential Sensitivities

Suppose the perturbed plant is LTI, and depends on some parameter α, which might represent the gain of an actuator transducer, for example. With $\alpha = \alpha_{\text{nom}}$, the perturbed plant is the nominal plant, in other words, $P^{(\alpha_{\text{nom}})} = P$. A small differential sensitivity specification limits in some way the partial derivative of the closed loop transfer matrix with respect to the parameter α, evaluated at the nominal parameter value α_{nom}.

Such a constraint *cannot* in general be expressed as a convex constraint on the nominal closed-loop transfer matrix H^{nom}, but in some important special cases it can. In this section we will present two examples of such differential sensitivity constraints. In the first example, the differential sensitivity constraint can be expressed as a convex constraint on the nominal closed-loop transfer matrix; in the second example we will show that it cannot be.

The first example is well-known in classical control. We consider the sensitivity of the I/O transfer function of a system with a one-degree-of-freedom controller when the actuator signal is scaled by the parameter α with a nominal value of $\alpha_{\text{nom}} = 1$. The system is shown in Fig. 38. The perturbed I/O transfer function is simply H_{11}^{α}:

$$H_{11}^{\alpha} = \alpha P_0 K/(1 + \alpha P_0 K).$$

[5]If the perturbed plant is not LTI, then we can define the *perturbed operator H*, which maps exogenous input signals to the resulting regulated outputs z. Formula (21) is *not* correct in this case.

If we calculate the logarithmic (relative or percentage) differential sensitivity of H_{11}^{α} with respect to α we find, using Bode's notation [5]:

$$S_\alpha = \left. \frac{\partial H_{11}^{\alpha}/H_{11}^{\text{nom}}}{\partial \alpha/\alpha_{\text{nom}}} \right|_{\alpha = \alpha_{\text{nom}}} = \frac{1}{1 + P_0 K} = H_{12}^{\text{nom}} \tag{22}$$

which is called the *sensitivity transfer function* of the system in classical terminology. We can interpret S_α as follows: its real part gives the sensitivity of the magnitude of the I/O transfer function (in nepers; 1 neper $\approx$ 8.7 dB) with respect to (fractional changes in) α, and its imaginary part gives the sensitivity of the phase of the I/O transfer function (in radians) with respect to α. Alternatively, we can think of $|S_\alpha|$ as the overall sensitivity of the (complex) I/O transfer function with respect to α. The important fact for us is that the logarithmic sensitivity of the I/O transfer function is equal to a closed-loop transfer function of the *nominal* system—H_{12}^{nom}.

It follows that a constraint of the form

$$\text{Re } S_\alpha(s_0) = 0, \tag{23}$$

which guarantees that the magnitude of I/O transfer function at the frequency s_0 is *first order insensitive* to variations in α, i.e.,

$$\left. \frac{\partial |H_{11}^{\alpha}(s_0)|}{\partial \alpha} \right|_{\alpha = \alpha_{\text{nom}}} = 0, \tag{24}$$

can be expressed as the following *affine* constraint on the nominal closed-loop transfer matrix:

$$\text{Re } H_{12}^{\text{nom}}(s_0) = 0. \tag{25}$$

Similarly, the specification

$$|S_\alpha(j\omega)| \le 0.10 \quad \text{for } 0 \le \omega \le \omega_B, \tag{26}$$

which guarantees logarithmic sensitivity less than 0.10 over the bandwidth of the system ($\omega \le \omega_B$), is also closed-loop convex, since it is equivalent to

$$|H_{12}^{\text{nom}}(j\omega)| \le 0.10 \quad \text{for } 0 \le \omega \le \omega_B. \tag{27}$$

Note that the constraint (27) and hence the robust performance constraint (26) can also be interpreted as a performance specification, for example, limiting the response of y to a disturbance w_2. The observation (22) above is often cited as an example of a benefit of feedback, and as a reason why the loop gain should be designed to be as large as possible. Our perspective is slightly different: we cite (22) as an example of the benefits of feedback, and a reason why a certain closed-loop transfer function of the nominal system (H_{12}^{nom}) should be designed to be as small as possible.

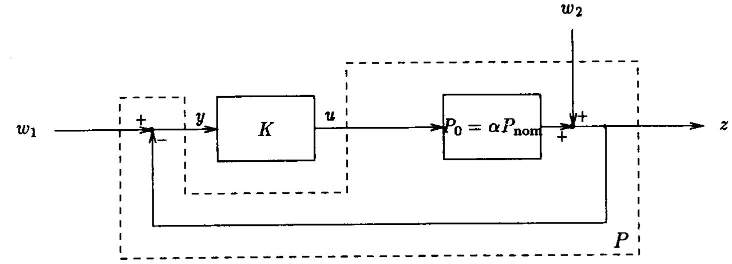


Fig. 38. A plant scaled by a factor α.

We will now give a simple example where a small differential sensitivity robust performance specification cannot be expressed as a closed-loop convex constraint on the nominal system. We consider the same system above and the constraint that the absolute (non-logarithmic) sensitivity of the closed-loop DC gain not exceed 0.10:

$$\left| \frac{\partial H_{11}^{\alpha}(0)}{\partial \alpha}\bigg|_{\alpha=\alpha_{nom}} \right| \leq 0.10. \qquad (28)$$

Now

$$\frac{\partial H_{11}^{\alpha}(s)}{\partial \alpha}\bigg|_{\alpha=\alpha_{nom}} = \frac{P_0(s)\, K(s)}{(1 + P_0(s)\, K(s))^2},$$

so it is apparent that (28) can be satisfied *either* by making the loop gain at DC small *or* by making it large. In terms of the nominal closed-loop transfer matrix, (28) can be satisfied with the entry $H_{12}^{nom}(0)$ near one or zero, *but not in between*, and thus the set of nominal closed-loop transfer matrices which satisfy the robust performance specification (28) is not convex.

Suppose, for example, that $P_0 = 1/(s + 1)$ and $\alpha_{nom} = 1$. The controller $K = 9$ yields nominal closed-loop transfer matrix

$$H^{nom} = \begin{bmatrix} \dfrac{9}{s + 10} & \dfrac{s + 1}{s + 10} \end{bmatrix}$$

and absolute sensitivity

$$\left| \frac{\partial H_{11}^{\alpha}(0)}{\partial \alpha}\bigg|_{\alpha=\alpha_{nom}} \right| = 0.09$$

and so meets the constraint (28).

The controller $\tilde{K} = 0$ yields nominal closed-loop transfer matrix

$$H^{nom} = [0 \quad 1]$$

and has zero absolute sensitivity of the I/O transfer function with respect to α, and so also satisfies the robust performance constraint (28).

But for the controller $9(s + 1)/(2s + 11)$, which yields a nominal closed-loop transfer matrix that is the average of those achieved by the previous two controllers, we find that the absolute sensitivity is

$$\left| \frac{\partial H_{11}^{\alpha}(0)}{\partial \alpha}\bigg|_{\alpha=\alpha_{nom}} \right| = 0.2475 > 0.10,$$

so this controller does not meet the robust performance constraint (28). Thus the robust performance specification (28) *cannot* be expressed as a convex constraint on the nominal closed-loop transfer matrix.

In the next two sections we examine some robustness constraints from classical control, applicable to "single-loop" systems (those with a single actuator and a single sensor). These constraints are expressed in terms of the nominal loop gain $L^{nom} = P_{yu}^{nom} K$.

C. Classical Gain and Phase Margins, "Loop Margin"

Perhaps the earliest robust performance specifications are those which limit in some way how close the Nyquist plot of the nominal loop gain can come to the critical point $+1$.[6] Well-known examples are the gain and phase margins, which express the proximity of the loop gain to $+1$ for L^{nom} real and $|L^{nom}| = 1$, respectively.

Let us show how classical specifications of minimum gain and phase margins can be interpreted in our framework as robust performance specifications. The nominal plant and controller have loop gain $L^{nom} = P_{yu}^{nom} K$; the system with the perturbed plant shown in Fig. 39 has a loop gain

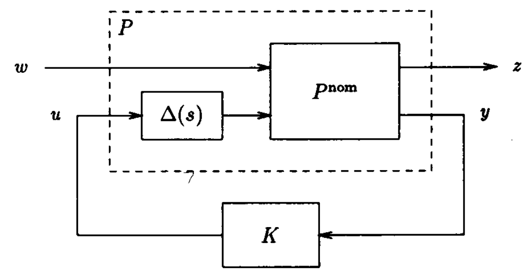


Fig. 39. The perturbed plant for gain and phase margin constraints.

$L(s) = \Delta(s)\, L^{nom}(s)$. The gain margin constraint

$$\mathrm{GM} \geq \mathrm{GM}_{min}$$

can be expressed as the robust performance constraint:

The closed-loop system is stable for all real, constant Δ such that $1/\mathrm{GM}_{min} < \Delta < \mathrm{GM}_{min}$. (29)

In words, a minimum gain margin guarantees stability of the closed-loop system despite variations in a real parameter which scales the nominal loop gain.

The phase margin constraint

$$\mathrm{PM} \geq \mathrm{PM}_{min}$$

can be expressed as the robust performance constraint

The closed-loop system is stable for all stable all-pass $\Delta(s)$ such that $|\angle \Delta(j\omega)| < \mathrm{PM}_{min}$ for all ω. (30)

In words, a minimum phase margin constraint guarantees that the closed-loop system will remain stable despite perturbations in the loop gain consisting of frequency dependent phase-shift less than PM_{min} for all frequencies.[7]

For low order systems, limits on the gain and phase margins usually limit the proximity of L^{nom} to $+1$ for all frequencies; but for higher order systems there is a greater risk that this is not the case: it is possible for a system to have large gain and phase margins, and yet have a loop gain of $L^{nom}(j\omega_0) = 0.999 + 0.001j$ at some frequency ω_0. Figure 40 shows an example of a loop gain which has comfortable gain and phase margins, but has a Nyquist plot which comes uncomfortably close to the critical point.

[6] In classical control the critical point is -1 since the actuator signal is *minus* the controller output, reflecting the tradition that feedback should be "negative."

[7] It is also possible to view the phase margin constraint as guaranteeing stability despite *constant unit magnitude complex* loop gain perturbations, which is more analogous to our interpretation of the gain margin constraint as a robust performance specification. Giving a physical interpretation of such a perturbation is quite difficult, however!

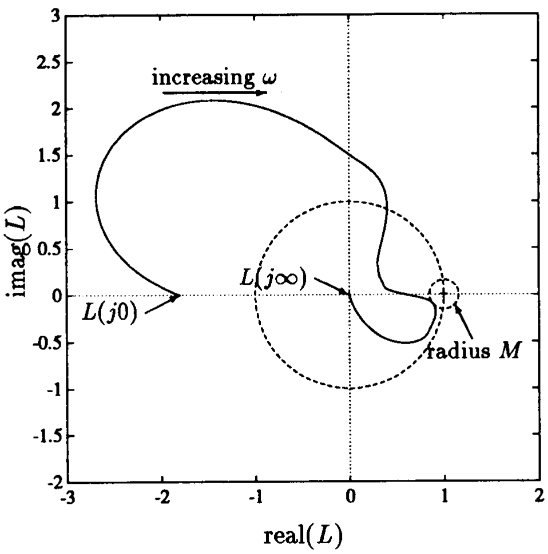


Fig. 40. A Nyquist plot of a loop gain which has good gain and phase margins, but a small loop margin.

A measure of robustness we will call *loop margin* does not share this "loophole" with the gain and phase margins. The loop margin M is simply the minimum distance in the complex plane between the Nyquist plot of the loop gain and the critical point $+1$:

$$M = \min_{\omega} \text{dist}\,(L^{\text{nom}}(j\omega), +1) = \min_{\omega} |1 - L^{\text{nom}}(j\omega)|. \quad (31)$$

This definition is illustrated graphically in Fig. 40.

A classical specification of minimum loop margin,

$$M \geq M_{\text{min}}, \quad (32)$$

can be interpreted as a robust performance constraint as shown in Fig. 41. The perturbed loop gain is

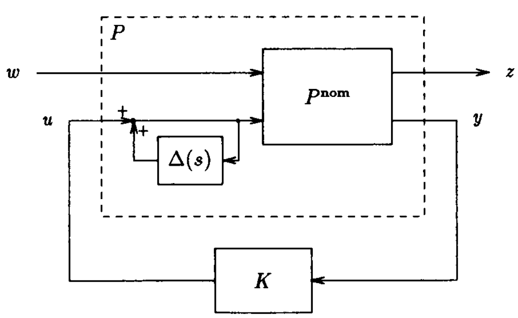


Fig. 41. The perturbed plant for loop margin constraint. Δ is a stable transfer function such that $|\Delta(jw)| < 1/M$ for all w.

$L(s) = L^{\text{nom}}(s) + \Delta(s)$, where $\Delta(s)$ is a stable transfer function. Such a perturbation is often called an *additive* loop gain perturbation, to distinguish it from the *multiplicative* perturbations used in the gain and phase margins above. (32) is equivalent to:

The closed-loop system is stable for all stable Δ

$$\text{such that } \|\Delta\|_\infty < 1/M_{\text{min}}. \quad (33)$$

In words, meeting a minimum loop margin specification is equivalent to guaranteeing stability of the closed-loop system despite additive perturbations of the nominal loop gain by any stable transfer function with magnitude less than $1/M$ for all frequencies. Unlike the phase margin case, the perturbation transfer function is not restricted to be an allpass filter.

Note that the loop margin is small in Fig. 40, despite the large gain and phase margins. From (33), this means that a small additive perturbation to the nominal loop gain could result in a closed-loop unstable system, even though it takes a relatively "large" real scalar or allpass loop gain perturbation to destabilize the system.

The minimum loop margin (32) is expressed in terms of an open-loop quantity of the nominal system, the nominal loop gain L^{nom}, but we will now show that it is in fact equivalent to a convex constraint on the nominal closed-loop transfer matrix H.

The loop margin is related to the sensitivity transfer function $H_{12}^{\text{nom}} = 1/(1 - P_{yu}^{\text{nom}} K)$ which was introduced in the preceding subsection. We can write

$$M = \min_{\omega} |1 - L^{\text{nom}}(j\omega)| = \frac{1}{\max_{\omega} \left| \dfrac{1}{1 - L^{\text{nom}}(\omega)} \right|} = \frac{1}{\|H_{12}^{\text{nom}}\|_\infty}. \quad (34)$$

Thus, a small loop margin corresponds to *peaking* of the sensitivity transfer function at some frequency. It also follows that (32) can be expressed as

$$\|H_{12}^{\text{nom}}\|_\infty \leq 1/M_{\text{min}}. \quad (35)$$

a convex constraint on the nominal closed-loop transfer matrix.

The relation (34) between the loop margin and peaking of the sensitivity transfer function is well known. Horowitz [6, p. 148] states (with critical point changed from -1 to $+1$, to match our notation),

> . . . it is not necessarily a useful practical system, if the locus [of L] passes very close to the $+1$ point. In the first place, a slight change of gain or time constant may sufficiently shift the locus so as to lead to an unstable system. In the second place, the closed-loop system response has $1 - L$ for its denominator. At those frequencies for which L is close to $+1$, $1 - L$ is close to zero, leading to large peaking in the system frequency response.

D. Circle Criterion Constraint

The reader may have recognized the minimum loop margin constraint (32) of the previous section as a special form of the *circle criterion* due to Zames [62], Sandberg [63], and Narendra and Goldwyn [64] (see also [65]–[68]). The circle criterion requires the loop gain to remain outside a *forbidden circle* in the complex plane; if the circle is centered at the critical point $+1$ and has radius M, this is the loop margin constraint (32).

The circle criterion provides a sufficient condition for closed-loop stability[8] despite certain *nonlinear* plant perturbations. The perturbations are time-varying memoryless nonlinearities in the actuator, as shown in Fig. 42; $\tilde{u}(t) = f(u(t), t)$. The time-varying memoryless nonlinearity f sat-

[8]Suitably defined for nonlinear systems.

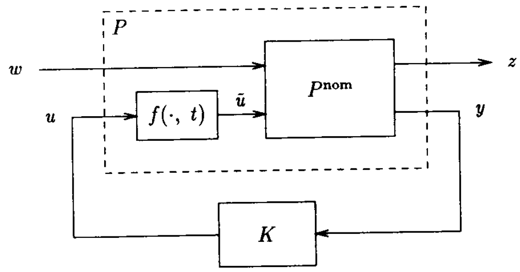


Fig. 42. System with a time varying nonlinearity f. The perturbed plant is nonlinear.

isfies a *sector condition:* roughly speaking, f multiplies u by at least α and at most β, i.e., for all t,

$$\alpha x^2 \le x f(x, t) \le \beta x^2,$$

where we assume $0 < \alpha \le 1 \le \beta$.[9] This is shown graphically (for a given time instant t_0) in Fig. 43.

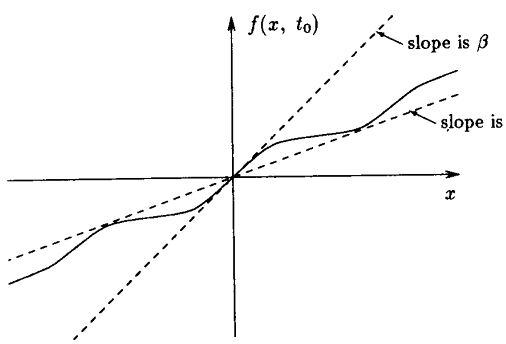


Fig. 43. An example of a sector $[\alpha, \beta]$ nonlinearity f.

The circle theorem states that if the nominal system is closed-loop stable and the Nyquist plot of the nominal loop gain remains outside the circle symmetric with respect to the real axis and passing through the points $1/\beta$ and $1/\alpha$, then the perturbed closed-loop system of Fig. 42 will be stable. It should be pointed out that the circle criterion is not generally a necessary condition for stability with such a nonlinearity in a system, so the specification of a circle criterion constraint is a conservative approximation of the robust performance specification that the system should remain stable for all nonlinearities in sector $[\alpha, \beta]$.

[9]The limiting case $\alpha = \beta = 1$ yields the nominal system.

The circle theorem yields another interpretation of (32) as a robust performance constraint: (32) and therefore (35) hold if and only if the circle criterion is satisfied for nonlinearities in sector $[1/(1 + M), 1/(1 - M)]$. Thus the closed-loop convex constraint (35) can be viewed as guaranteeing stability despite time-varying memoryless nonlinear perturbations in the loop.

In fact, any circle criterion constraint for a nonlinearity in sector $[\alpha, \beta]$, with $0 < \alpha \le 1 \le \beta$, is closed-loop convex. To see this, we note first that the constraint that the nominal loop gain remain outside the forbidden circle can be expressed as

$$\text{dist}\,(L^{\text{nom}}(j\omega), c)) > r \qquad \text{for all } \omega \tag{36}$$

where $c = (1/\alpha + 1/\beta)/2$ is the center of the circle and $r = (1/\alpha - 1/\beta)/2$ is its radius. This constraint on the nominal loop gain is equivalent to

$$\left\| \frac{\gamma L^{\text{nom}} + \delta}{1 - L^{\text{nom}}} \right\|_\infty \le 1 \tag{37}$$

if we set

$$\gamma = \frac{\alpha + \beta - 2}{\beta - \alpha} \quad \text{and} \quad \delta = \frac{\alpha + \beta - 2\alpha\beta}{\beta - \alpha}. \tag{38}$$

The constraint (37) is clearly closed-loop convex because it is simply the constraint $\|H^{\text{nom}}\|_\infty \le 1$ for the nominal system with w and z selected as in Fig. 44. The equivalence of (36) and (37) comes about because with these choices of γ and δ the bilinear transformation $H = (\gamma L + \delta)/(1 - L)$ maps the exterior of the critical circle, the permissible region for L^{nom} in (36), to the interior of the unit circle, the permissible region for H in (37). This is illustrated in Fig. 45.

A generalization of this idea can be made in the case where it is desired to insure that the Nyquist plot of the nominal loop gain L^{nom} remain outside a region that is the *union* of disks, all of which are symmetric about the real axis, contain the point $+1$, and do not cross the imaginary axis. An example with two such disks is shown in Fig. 46. From the preceding discussion it should be clear that we can express the constraint that L^{nom} remain outside the forbidden region as

$$\left\| \frac{\gamma_1 P_{yu} K + \delta_1}{1 - P_{yu} K} \right\|_\infty \le 1 \quad \text{and} \quad \left\| \frac{\gamma_2 P_{yu} K + \delta_2}{1 - P_{yu} K} \right\|_\infty \le 1, \tag{39}$$

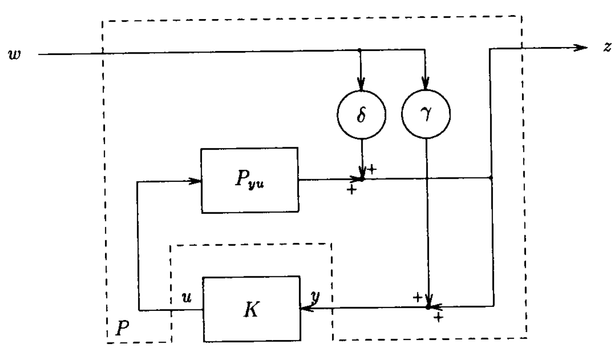


Fig. 44. $\|H\|_\infty \le 1$ for this nominal system if and only if the circle criterion is satisfied.

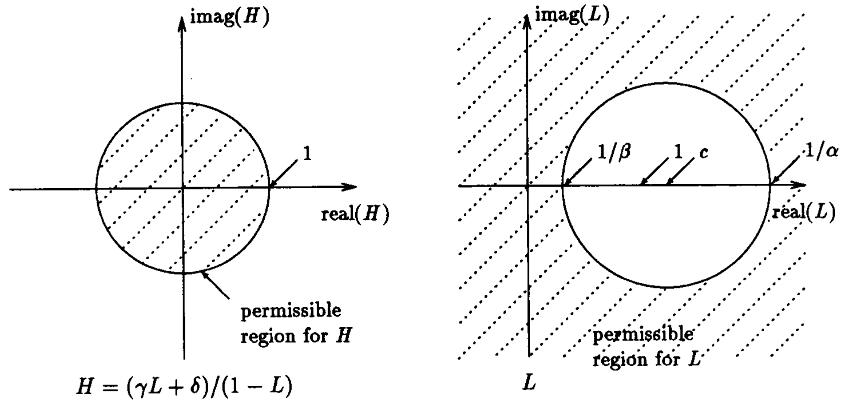


Fig. 45. Equivalence of constraints on L and H.

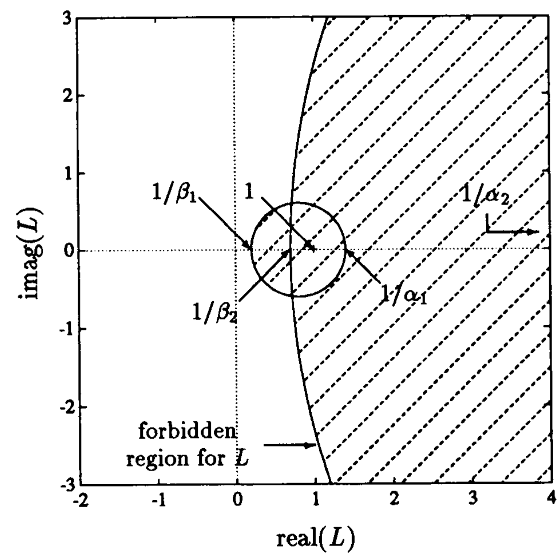


Fig. 46. Forbidden region for the loop gain L is the union of two disks.

where γ_1, δ_1, γ_2 and δ_2 are obtained from α_1, β_1, α_2 and β_2 as in (38). The constraint (39) is clearly closed-loop convex since it requires the simultaneous satisfaction of two convex constraints on the closed-loop transfer matrix.

E. Simultaneous Versus One-at-a-Time Perturbations

We showed in the preceding sections how to express certain robust performance constraints as convex constraints on the nominal closed-loop system. There is an important subtlety involving simultaneous satisfaction of two or more of these robust performance constraints, which arises particularly often in systems with multiple actuators and multiple sensors. Consider, for example, the two-sensor, two-actuator system shown in Fig. 47.

One way to impose a minimum robustness specification on this system is to constrain each loop separately—for example, we may require that the loop gains from point A_1 to point B_1 in Fig. 48(a) and from point A_2 to point B_2 in Fig. 48(b) each have a loop margin exceeding 0.5. As seen in section VII-C, this requirement can be expressed as a convex constraint on the nominal closed-loop transfer matrix—specifically, a constraint that two appropriate closed-loop transfer functions have magnitudes less than 2 for all frequencies.

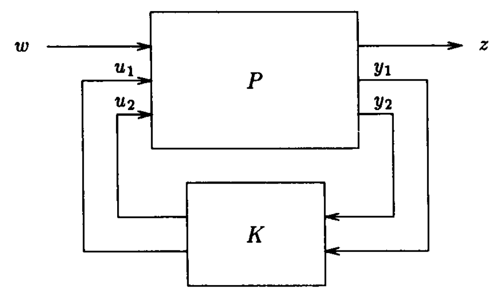
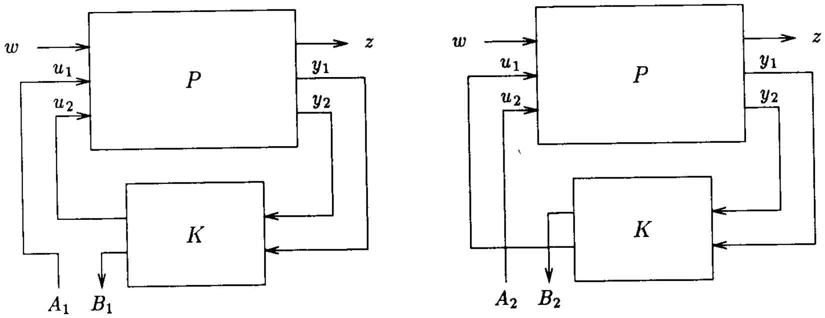


Fig. 47. Two-actuator, two-sensor system.

In a single-loop system such a loop margin would indicate robustness to additive loop gain perturbations less than 0.5 in magnitude for all frequencies; for this system it means that the system will remain stable for loop gain perturbations in either loop separately. However, it is entirely possible that our two-loop system can be destabilized by simultaneous, extremely small perturbations (magnitude $\ll 0.5$) in *both* loops. The large loop margins guarantee only that the system cannot be destabilized by *either* a small additive perturbation in one loop or the other. If the real intention is to prevent instability when *both* loops are perturbed slightly, then we cannot express our specification as two separate robust performance constraints on each loop. This subtlety, and the confusion it caused, was part of the motivation for the development of a "modern" framework for robust performance specifications.

It is interesting to note that this subtlety does not arise when we consider only differential sensitivities, since the *first order effects* of many simultaneous parameter variations simply *add*. Thus, the first order sensitivity (derivative) of some closed-loop transfer function in our two loop system above to small variations in the two actuator gains could be calculated by considering the effects of each perturbation separately. In the terms of differential sensitivity, then, the subtlety is a *higher order effect*.

One of the successes of multivariable control theory is the development of an understanding of how to properly



(a) (b)

Fig. 48. Systems with one or the other loop broken.

generalize the loop margin constraint (32) to multiple-actuator, multiple-sensor systems. One such generalization uses the *singular values* of the loop gain transfer *matrix* $L^{\text{nom}} = P_{yu}^{\text{nom}}K$.[10] A robustness constraint analogous to the loop margin constraint, introduced in [23], is:

$$\underline{\sigma}(L^{\text{nom}}(j\omega) - I) \geq M \qquad \text{for all } \omega, \tag{40}$$

where $\underline{\sigma}(\cdot)$ denotes the minimum singular value. This specification can also be expressed as a convex constraint on the nominal closed-loop transfer matrix:

$$\|(I - L^{\text{nom}})^{-1}\|_\infty \leq 1/M. \tag{41}$$

This robust performance specification is an example of a "modern" robust performance specification.

F. Modern Method—Small Gain Theorem

A modern framework for robust performance constraints has been developed by Doyle and Stein and other researchers [23], [67]. (See also the IEEE collection edited by Dorato [70].) Each perturbation is expressed as an additive perturbation inside the plant as shown in Fig. 49; the nominal plant results when each perturbation is set to zero. The perturbations are then extracted from the plant as shown in Fig. 50.

The resulting system has exactly the same form as the basic feedback system in Fig. 5, but with the *perturbations* substituted for the controller. The vector signal t plays the role of the sensed output y in Fig. 5; t is the vector of signals which *drive* or *excite* the perturbations in our perturbed system. The vector signal s plays the role of the actuator signal u in Fig. 5; s is the vector of signals *generated by* the perturbations in our system.

In our framework, it will be convenient to include the signal s in the exogenous input w, and the signal t in the regulated output signal z. The closed-loop transfer matrix of the nominal system from s to t is then a submatrix of H^{nom}; we denote it by H_{ts}^{nom}. H_{ts}^{nom} is the closed-loop feedback transfer matrix "seen" by the perturbations when they are added to the nominal system, as shown in Fig. 51.

The principle tool used to establish robust stability for such a system is the *small gain theorm* [71] which, roughly speaking, states that if the perturbations Δ and the closed-loop feedback H_{ts}^{nom} "seen" by the perturbations are not too "large," then the feedback loop in Fig. 51 is stable, and hence the closed-loop perturbed system shown in Fig. 49 is also stable. An advantage of this method is its great generality—the perturbations can be nonlinear and time-varying, and different ways of measuring what "large" and "small" are for H_{ts}^{nom} and Δ can yield different conditions for robust stability.

If the perturbations are LTI and stable, so that the perturbed plant is also linear time-invariant, then one way to express the small gain theorem is:

$$\text{If } \bar{\sigma}(H_{ts}^{\text{nom}}(j\omega))\, \bar{\sigma}(\Delta(j\omega)) < 1 \text{ for all } \omega,$$

$$\text{the perturbed closed-loop system is stable.} \tag{42}$$

In classical terms, the small gain theorem expresses the idea that if the loop gain has magnitude less than one at all frequencies, then the feedback cannot destabilize the system.

[10]This is the transfer matrix of the loop "broken at the sensor;" it is not the same as the transfer matrix KP_{yu}^{nom} of the loop "broken at the actuator." This is one of the many subtleties involved in extending concepts from classical, single actuator, single sensor control systems to multiple-actuator, multiple-sensor systems.

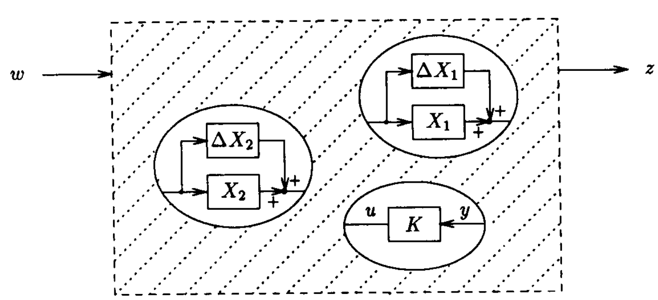


Fig. 49. Two internal additive perturbations.

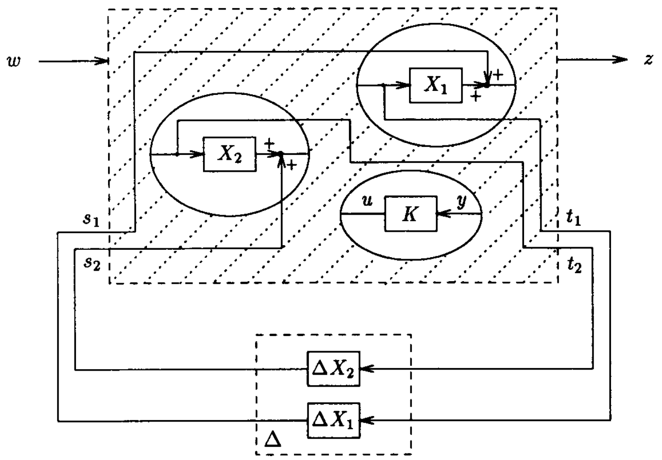


Fig. 50. Extraction of additive perturbations.

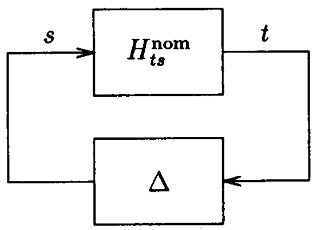


Fig. 51. The loop for the small gain theorem.

Partly because of the small gain theorem (42), it has become common to express robust performance constraints in the form:

The system remains stable for all perturbations Δ

such that $\bar{\sigma}(\Delta(j\omega)) < l(\omega)$ for all ω. (43)

The small gain theorem shows that the convex constraint on the nominal closed-loop transfer matrix,

$$\bar{\sigma}(H_{ts}^{\text{nom}}(j\omega)) \leq 1/l(\omega) \text{ for all } \omega, \tag{44}$$

is *sufficient* to guarantee the robust performance specification (43), or in other words, the constraint (44) is *stronger* than the constraint (43). Doyle observed that in fact the two constraints are equivalent, so that the closed-loop convex constraint (44) is necessary and sufficient for robust stability (43).

The careful reader will notice that in passing from Fig. 50 to the robust performance constraint (43), the set of possible perturbations quietly grew—specifically, in Fig. 50, the perturbations have a certain block structure, while in (43) they do not. Avoiding this growth in the set of possible perturbations, and the corresponding unnecessary strengthening of the robust performance specification (43) is the motivation behind Doyle's introduction of the concept of the *structured singular valve* [72]. See [69] for the case where the perturbation transfer matrix is restricted to be diagonal.

This modern framework allows many different types of perturbations to be considered, and not just perturbations on the loop gain. To give one example, consider a system with a so-called *additive plant perturbation*, meaning that $P_{yu} = P_{yu}^{\text{nom}} + \Delta$, with a bound given on the size of the perturbation, say, $\|\Delta\|_\infty < D$. This is shown in Fig. 52. The plant perturbation could represent errors in modeling the plant, component variations, or deliberately ignored plant dynamics.

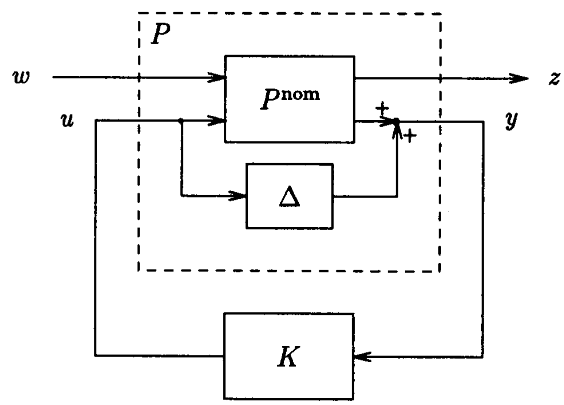


Fig. 52. An additive plant perturbation Δ.

We consider the following robust performance specification:

The closed-loop system requires stable for all

Δ with $\|\Delta\|_\infty < D$. (45)

Roughly speaking, this specification means that our controller must stabilize all plants with transfer matrices P_{yu} that differ from the nominal transfer matrix P_{yu}^{nom} by less than D. Following the method outlined above, we extract the perturbation as shown in Fig. 53.

Examination of Fig. 53 shows that for the additive plant perturbation, we have

$$H_{ts}^{\text{nom}} = K(I - P_{yu}^{\text{nom}}K)^{-1}.$$

Hence the robust performance specification (45) is equivalent to the convex constraint on the nominal closed-loop transfer matrix,

$$\|H_{ts}^{\text{nom}}\|_\infty = \|K(I - P_{yu}^{\text{nom}}K)^{-1}\|_\infty \leq 1/D. \tag{46}$$

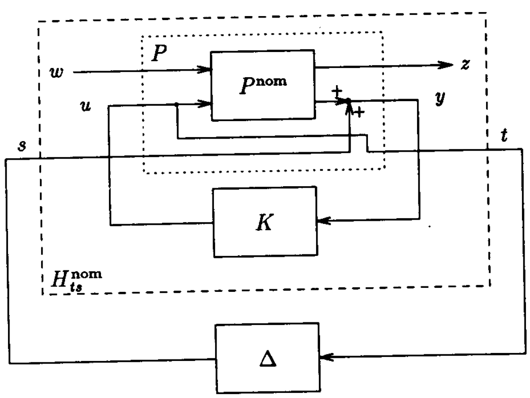


Fig. 53. The additive plant perturbation Δ extracted from the plant.

It is interesting to note that (46) can also be interpreted as a performance requirement, a bound on actuator authority, since H_{ts}^{nom} can also be interpreted as the closed-loop transfer matrix from sensor noise to actuator signal. Thus Doyle and Stein's observation of the equivalence of our robust performance specification (45) and (46) tells us that excessive actuator authority (H_{ts}^{nom} large; a performance constraint) is equivalent to the possibility of system destabilization with a small additive plant perturbation (a robust performance constraint).

G. A Special Case of Robust Performance

In the previous three sections we considered various robust performance specifications which guaranteed closed-loop stability despite certain variations in the plant. There are very few results available which guarantee satisfaction of other performance specifications, e.g., small step response overshoot, low actuator authority, or noise response, despite variations in the plant. In this section we consider one such result. We will show that it can be expressed as a convex constraint on the nominal closed-loop transfer matrix H^{nom}.

The problem and the derivation of the relevant constraint on a closed-loop transfer matrix is presented in a different but equivalent form by Francis in [73]. The system in question is shown in 54. The perturbed closed-loop transfer matrix is

$$H^{\Delta} = \begin{bmatrix} \dfrac{1}{1 + P_0K(1 + \Delta)} \\[2ex] \dfrac{P_0K(1 + \Delta)}{1 + P_0K(1 + \Delta)} \end{bmatrix}.$$

Its first component is the closed-loop transfer function from the command w to the tracking error z_1, which is also the classical sensitivity transfer function. Its second component is the classical complementary sensitivity transfer function or I/O transfer function.

We consider a frequency domain specification of tracking:

$$|H_1^{\Delta}(j\omega)| \le B(\omega) \quad \text{for all } \omega. \tag{47}$$

$B(\omega)$ is presumably small for those frequencies where the command w has appreciable power.

The perturbation Δ is stable and satisfies

$$|\Delta(j\omega)| < A(\omega) \quad \text{for all } \omega \tag{48}$$

The robust performance specification we consider is: For all Δ satisfying (48), the closed-loop system is stable and satisfies the tracking specification (47).

As seen in section VII-F, robust stability holds if and only if

$$|H_2^{nom}(j\omega)| \le \frac{1}{A(\omega)} \quad \text{for all } \omega. \tag{49}$$

Rewriting (47) in terms of H_1^{nom} and H_2^{nom} means the robust tracking specification is equivalent to

$$\left| \frac{H_1^{nom}(j\omega)}{1 + \Delta(j\omega)\, H_2^{nom}(j\omega)} \right| \le B(\omega) \quad \text{for all } \omega. \tag{50}$$

In [73] it is shown that constraints (49) and (50) together are equivalent to the single constraint

$$\frac{|H_1^{nom}(j\omega)|}{B(\omega)} + A(\omega)\,|H_2^{nom}(j\omega)| \le 1 \quad \text{for all } \omega. \tag{51}$$

The constraint (51) is clearly closed-loop convex.

VIII. Non-Convex Design Requirements

The preceding three sections have shown that a large range of design specifications for LTI feedback systems are closed-loop convex. Nevertheless there are several design specifications that are often important and that *cannot* be expressed as convex constraints on the closed-loop transfer matrix. In this section we present three such examples, all of the type we labeled *control law specifications* in section I-B.

A. Open-Loop Stability of the Controller

Consider the system shown in Fig. 55, and let u, w, y, and z be scalar with $P_{yu}(s) = 1/(s + 1)$. Two controllers that stabilize the plant are

$$K(s) = -36\frac{s + 1}{s^2 + s + 3} \quad \text{and} \quad \tilde{K}(s) = 6\frac{s + 1}{s + 8};$$

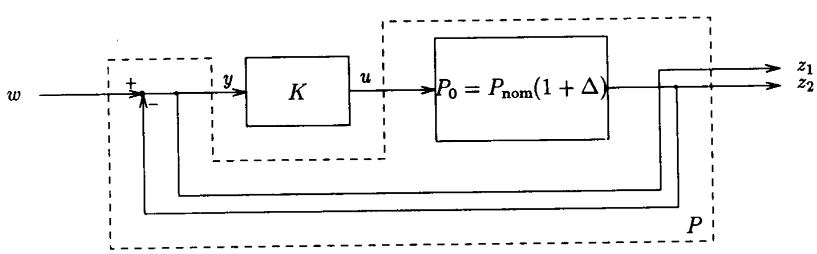


Fig. 54. Stability and tracking performance must be guaranteed as P_0 varies.

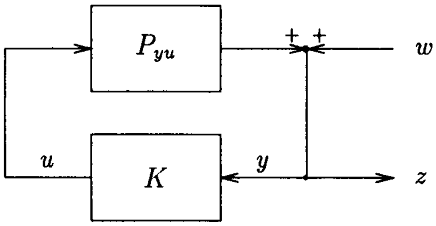


Fig. 55. H is the sensitivity transfer matrix $(I - P_{yu}K)^{-1}$.

they result in closed-loop transfer functions from w to z of

$$H(s) = \frac{s^2 + s + 3}{s^2 + s + 39} \quad \text{and} \quad \tilde{H}(s) = \frac{s + 8}{s + 2},$$

respectively. Note that both K and $\tilde{K}$ are open-loop stable.

We know that a transfer function from w to z that is the average of H and $\tilde{H}$ can be achieved with some stabilizing compensator $K_{0.5}$; computation using equation (8) yields

$$K_{0.5}(s) = \frac{3(s + 1)(s^2 - 5s - 27)}{s^3 + 6s^2 + 26s + 159},$$

the poles of which are -6.0479 and $0.0240 \pm j5.1273$. So although H and $\tilde{H}$ are achieved with open-loop stable controllers, the average of H and $\tilde{H}$ is not. Therefore open-loop stability of the controller is not a closed-loop convex specification.[11]

B. Decentralized Controller

Consider again the system of Fig. 55, but now let u, w, y, and z be two-vectors with

$$P_{yu}(s) = \begin{bmatrix} 1/s & 0 \\ 0.1/s & 1/s \end{bmatrix}.$$

Suppose it is specified that the controller be *decentralized*, i.e., that u_1 is driven only by y_1 and u_2 is driven only by y_2—meaning that the controller transfer matrix K is *diagonal*. Two controllers that meet this specification and stabilize the plant are

$$K(s) = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \quad \text{and} \quad \tilde{K}(s) = \begin{bmatrix} -5 & 0 \\ 0 & -1 \end{bmatrix}.$$

If we calculate the controller $K_{0.5}$ that stabilizes the plant and yields a closed-loop transfer matrix that is the average of the transfer matrices achieved by the controllers K and $\tilde{K}$, we find

$$K_{0.5}(s) = \begin{bmatrix} \dfrac{-3(s + 5/3)}{(s + 3)} & 0 \\ \dfrac{0.4(s + 2)}{(s + 3)} & -1 \end{bmatrix}.$$

Here u_2 is driven by both y_1 and y_2, so $K_{0.5}$ does not meet the specification of decentralization. So decentralization is not a closed-loop convex specification.

[11]However, for a single-actuator single-sensor system, it can be shown that the requirement that the controller have no *real* unstable poles, *is* closed-loop convex.

C. Controller Order

In the example in section VIII-A, K is second-order and $\tilde{K}$ is first-order; $K_{0.5}$ on the other hand is third-order. In the example in section VIII-C, K and $\tilde{K}$ are constant matrices, i.e., zeroth-order controllers, but $K_{0.5}$ has first-order dynamics. Finally, in the example of section V-C(1), K and $\tilde{K}$ are first-order PI controllers, but $K_{0.5}$ is third-order. It should be quite obvious that the specification, "The controller K must have an order no greater than n," is not closed-loop convex.

IX. Numerical Methods for the Fundamental Problem

In the previous sections we have shown that many specifications for a control system can be expressed as convex constraints on the closed-loop transfer matrix H. Each specification has the form $H \in \mathcal{H}_i$, and the Fundamental Problem can be rephrased as the question of whether the set of closed-loop transfer matrices satisfying all the specifications,

$$\mathcal{H}_{\text{spec}} = \mathcal{H}_1 \cap \cdots \cap \mathcal{H}_L,$$

is empty or not. In most cases this is an infinite dimensional convex feasibility problem.

The purpose of this section is to discuss very briefly a few of the important issues involved in numerical solution of the fundamental problem.

A. Finite-Dimensional Approximations

All numerical computations will involve some finite dimensional approximation of the infinite dimensional problem. The most obvious approximation is to restrict the search for an $H \in \mathcal{H}_{\text{spec}}$ to a finite dimensional subspace of transfer matrices $\mathcal{H}_{\text{fdapp}}$. Thus, instead of determining whether $\mathcal{H}_{\text{spec}}$ is empty or not we determine whether $\mathcal{H}_{\text{spec}} \cap \mathcal{H}_{\text{fdapp}}$ is empty or not. In effect, we have adjoined a fictitious additional constraint to our specifications.

It is often possible to make the added constraint $\mathcal{H}_{\text{fdapp}}$ meaningful in terms of some of the performance specifications. Let us consider a simple example. A discrete-time system has a single actuator and a single sensor, and one of the performance specifications is a step response bound

$$1 - 0.9^t \le s_{11}(t) \le 1 + 0.95^t, \qquad t = 0, 1, 2, \cdots \tag{53}$$

Here, w_1 might be a command input, and z_1 the variable it is intended to command. Thus for $t \ge 90$, we require that the step response has settled to within less than 1%. In many applications, we might consider the inequalities given by (53) for $t \ge 90$ as practically equivalent to the *equalities*

$$s_{11}(t) = 1, \qquad t = 90, 91, 92, \cdots \tag{54}$$

which express that the step response has actually *settled* by $t = 90$. In fact, the specification (54) yields a finite (at most 90) dimensional $\mathcal{H}$, so we may take

$$\mathcal{H}_{\text{fdapp}} = \{H \mid s_{11}(t) = 1, \quad t = 90, 91, 92, \cdots\}.$$

In this case we can see clearly what restriction the finite dimensional approximation $\mathcal{H}_{\text{fdapp}}$ has placed on us—exact settling versus 1% settling of the step response at $t = 90$, and of course less for larger t. This is surely an acceptable approximation in any real problem.

A general method for generating finite-dimensional approximation to $\mathcal{H}_{\text{spec}}$ uses the free parameter representation (15). In the free parameter representation, Q ranges over the infinite dimensional vector space of all stable $n_{act} \times n_{sens}$ transfer matrices. We may restrict Q to lie in some finite dimensional subspace, say, the space of linear combinations of some finite set of fixed, stable transfer matrices Q_i. This approach is taken in qdes [74].

What exactly does the approximation mean? Since $\mathcal{H}_{spec} \cap \mathcal{H}_{fdapp}$ is an inner approximation (subset) of $\mathcal{H}_{spec}$, if we find that it is nonempty, then we can say with certainty that the specifications are achievable. If we find that our approximation $\mathcal{H}_{spec} \cap \mathcal{H}_{fdapp}$ is empty, on the other hand, then all we really know is that if there is an $H \in \mathcal{H}_{spec}$, then it does not satisfy $\mathcal{H}_{fdapp}$.

Several views of this finite-dimensional approximation can be taken. The designer can simply ignore the approximation—this would be justified in the example above where the approximation is simply a small perturbation, of negligible practical significance, to the performance specifications.

It should also be possible to analytically determine *bounds* on the error incurred by the finite-dimensional approximation, although to our knowledge this has not been done. On the basis of $\mathcal{H}_{spec} \cap \mathcal{H}_{fdapp} = \emptyset$, we could then conclude that a slightly smaller (tighter) set of specifications was unachievable. In other words, we could *know* that if each specification were tightened slightly, then the resulting set of specifications is not achievable. This is a topic for further research.

B. Approximation of Specifications

Even after we have restricted our attention to a finite dimensional subspace of H's, it may be necessary to approximate some of the constraints. A good example is given by the convex constraint (51),

$$\frac{|H_1^{nom}(j\omega)|}{B(\omega)} + A(\omega)|H_2^{nom}(j\omega)| \le 1 \qquad \text{for all } \omega,$$

which arose in section VII-G. There is no analytical method known to determine whether a given rational 2×1 transfer matrix H satisfies the specification (51). Of course, we may approximate this specification by

$$\frac{|H_1^{nom}(j\omega_m)|}{B(\omega_m)} + A(\omega_m)|H_2^{nom}(j\omega_m)| \le 1 \quad \text{for } m = 1, \ldots, M, \tag{55}$$

where $\omega_1, \cdots, \omega_M$ is some grid of frequencies.

Some specifications can be checked (in principle) exactly by an analytical method. An $\mathbf{H}_\infty$ specification can be checked by determining whether a certain Hamiltonian matrix has imaginary eigenvalues [75], [76]; an $\mathbf{H}_2$ constraint can be exactly checked by solving a Lyapunov equation. Even though there are analytical methods for determining these specifications, it may still be desirable to approximate them as in (55) above.

One goal of research in the area of *semi-infinite programming* is to find efficient methods for picking the frequencies in (55) [28].

C. On Finite Dimensional Nondifferentiable Convex Optimization

There are many methods available to solve the resulting finite dimensional convex nondifferentiable feasibility problems.

We will use $x \in \mathbf{R}^K$ to denote the decision variables. We suppose that each specification has the form

$$\mathcal{H}_i = \{H | \phi_i(x) \le 0\} \tag{56}$$

where ϕ_i is a convex function from $\mathbf{R}^K$ into $\mathbf{R}$. For example, if

$$\mathcal{H}_1 = \{H | \, \|H\|_\infty \le 2\}$$

and our finite dimensional approximation is

$$\mathcal{H}_{fdapp} = \left\{ H \middle| H = T_1 + T_2 \sum_{i=1}^{K} x_i Q_i T_3 \right\}$$

then we could take

$$\phi_1(x) = \left\| T_1 + T_2 \sum_{i=1}^{K} x_i Q_i T_3 \right\|_\infty - 2.$$

Our optimization problem can now be expressed as

$$\underset{x \in \mathbf{R}^K}{\text{minimize}} \ \phi(x), \tag{57}$$

where $\phi(x) = \max_i \phi_i(x)$. If the minimum in (57) is zero or negative, then $\mathcal{H}_{spec} \cap \mathcal{H}_{fdapp}$ is nonempty (feasible). If the minimum in (57) is positive, $\mathcal{H}_{spec} \cap \mathcal{H}_{fdapp}$ is empty (infeasible). This technique of minimizing the maximum "infeasibility" is a standard approach in feasibility problems.

A discussion of the many numerical algorithms for the nondifferentiable convex optimization problem (57) is beyond the scope of this paper, but we shall describe one extremely simple algorithm—the *subgradient* method (Shor [77]), which is a generalization of the standard gradient method for differentiable problems.

The subgradient algorithm applied to (57) proceeds as follows: at the kth iteration, one simply picks any active ϕ_j, i.e., j such that $\phi_j(x^{(k)}) = \phi(x^{(k)})$. The next $x^{(k+1)}$ is then formed as follows:

$$x^{(k+1)} = x^{(k)} - \frac{\nabla \phi_j(x^{(k)})}{k \|\nabla \phi_j(x^{(k)})\|}$$

(we may replace $\nabla \phi_j$ with any *subgradient* at $x^{(k)}$ if all the active ϕ_j are nondifferentiable). This represents a fixed step of length $1/k$ in the direction of the negative gradient of the active ϕ_j selected, and may or may not be a descent direction for the problem (57), meaning that we can have $\phi(x^{(k+1)}) > \phi(x^{(k)})$. Nevertheless, it can be shown that this simple algorithm is convergent to the (global) minimum in (57). If the minimum value of (57) is negative, the subgradient algorithm will produce a feasible point in a finite number of iterations.

D. Effectiveness

By far the most important attribute of *convex* optimization problems is that *there are effective methods for solving them*.

This assertion can be argued on several levels. On a pragmatic level, we note that very large linear and quadratic convex programs are routinely solved numerically; the growth of computation time with numbers of variables and constraints has been observed empirically to be quite moderate (much less than combinatorically). In constrast, *global* solutions of nonconvex programs are attempted only occasionally, and then for much smaller numbers of variables. These methods generally require extremely large computation times, which in addition tend to rise combinatorically with the numbers of variables (see e.g., Pardalos and Rosen [78]). For this reason, numerical optimization of general nonconvex programs is mostly restricted to the computation of *local* optima, or heuristic methods to compute the global optimum. In many cases, these local or heuristic methods produce acceptable solutions, which may even be globally optimal, but this cannot be guaranteed.

It is also possible to put this notion that convex optimization problems are "substantially more tractable" than nonconvex problems on firm theoretical ground. We *very roughly* paraphrase some of the results from the book by Nemirovsky and Yudin, *Problem Complexity and Method Efficiency in Optimization* [79], to which we refer the reader for complete, precise statements. The minimum number of "computations" (function and gradient or subgradient evaluations) required to compute the global minimum of a general differentiable function in n variables, within an accuracy of ϵ, necessarily grows like $(1/\epsilon)^n$—roughly speaking, the complexity of a search over an ϵ-grid. If the function is convex, however, the minimum number of "computations" necessary to compute an ϵ-approximation of the global minimum has more the flavor of $n \log (1/\epsilon)$—roughly speaking, the complexity associated with a bisection method. The reader should note the slow growth of this number with both accuracy $(1/\epsilon)$ and number of variables (n).

X. An Example of Finding the Limit of Performance

The results of this section are taken from [80], to which we refer the reader for more detail.

A. The Plant

We consider the design of a regulator for a system with a single actuator (scalar u) which is disturbed by an input-referred process noise d_{proc}. The scalar output we wish to regulate is y_p, and the only signal available to the controller is $y = y_p + d_{sens}$, where d_{sens} is a sensor noise. A classical block diagram of the closed loop system is shown in Fig. 56.

In our framework, we take

$$w = \begin{bmatrix} d \\ d_{sens} \\ d_{proc} \end{bmatrix}, \quad z = \begin{bmatrix} u \\ y_p \end{bmatrix},$$

where d is a fictitious exogenous input injected into y_p which we will use to express a robustness constraint. This is shown in Fig. 57.

Thus the first row of H consists of the closed-loop transfer functions from d, d_{sens}, and d_{proc} to the actuator signal u; the second row consists of the closed-loop transfer functions from d, d_{sens}, and d_{proc} to the plant output y_p.

We take

$$P_0(s) = \frac{1}{s^2}\frac{4 - s}{4 + s}.$$

which consists of a double integrator with some excess phase from the allpass term $(4 - s)/(4 + s)$. This allpass term approximates a 1/2 second delay $(e^{-s/2})$ at low frequencies. We may think of the allpass term $(4 - s)/(4 + s)$ as accounting for any and all of a variety of sources of excess phase in a real control loop, for example, small delays and anti-alias filters. The idea of using a double integrator system with some excess phase as a simple but realistic typical system with which to explore control design tradeoffs is taken from a study presented by Gunter Stein in [81].

The plant was discretized using a zero order hold at 10 Hz. The discrete time transfer function is

$$P_0(z) = \frac{-0.00379(z^2 - 0.7241z - 1.1457)}{z^3 - 2.6703z^2 + 2.3406z - 0.6703} = \frac{-0.00379(z - 1.492)(z + 0.7679)}{(z - 1)^2(z - 0.6703)}.$$

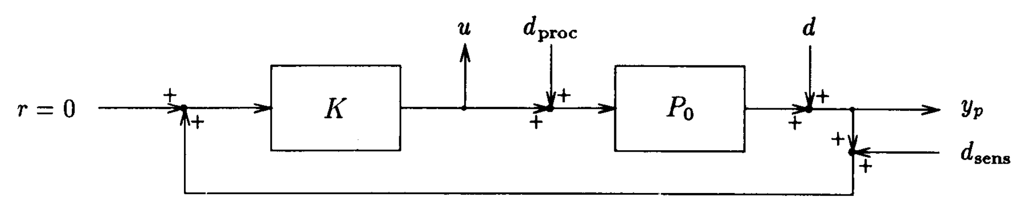


Fig. 56. The regulator system.

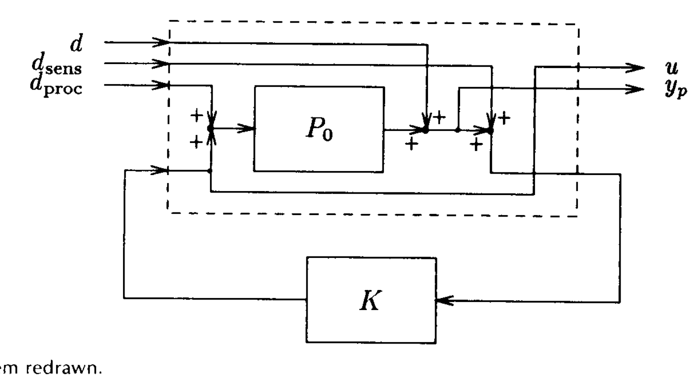


Fig. 57. The regulator system redrawn.

We will assume that the discrete time sensor and input referred process noises are zero mean, white, independent, with RMS values

$$(Ed_{\text{sens}}(t)^2)^{1/2} = 1$$

$$(Ed_{\text{proc}}(t)^2)^{1/2} = 10.$$

B. The Specifications

The specifications we will explore are:

- *Plant output regulation*—a maximum RMS response at the plant output y_p due to the process and sensor noises.
- *Actuator effort*—a maximum RMS response at the actuator u due to the process and sensor noises.
- *Robustness to loop gain perturbations*—a minimum loop margin constraint, or equivalently, a minimum robustness of the closed-loop system to additive loop perturbations.
- *Robustness to additive plant perturbations*—a minimum robustness of the closed-loop system to additive plant perturbations.

These specifications can be expressed as follows:

Plant output regulation. Assuming K stabilizes P, we can calculate the RMS value of y_p in terms of H_{22} and H_{23}, the closed-loop transfer functions from the sensor and process noises to y_p:

$$[\text{RMS}(y_p)]^2 = \lim_{t\to\infty} Ey_p(t)^2 = \frac{1}{2\pi}\int_{-\pi}^{\pi} |H_{22}(e^{j\Omega})|^2\, d\Omega + \frac{10}{2\pi}\int_{-\pi}^{\pi} |H_{23}(e^{j\Omega})|^2\, d\Omega.$$

Thus the regulation specification $\text{RMS}(y_p) \le \alpha$ can be expressed as the convex specification on H:

$$\frac{1}{2\pi}\int_{-\pi}^{\pi} |H_{22}(e^{j\Omega})|^2\, d\Omega + \frac{10}{2\pi}\int_{-\pi}^{\pi} |H_{23}(e^{j\Omega})|^2\, d\Omega \le \alpha^2.$$

Actuator effort. The actuator authority specification $\text{RMS}(u) \le \beta$ can be expressed as the convex specification on H:

$$\frac{1}{2\pi}\int_{-\pi}^{\pi} |H_{12}(e^{j\Omega})|^2\, d\Omega + \frac{10}{2\pi}\int_{-\pi}^{\pi} |H_{13}(e^{j\Omega})|^2\, d\Omega \le \beta^2.$$

Robustness to loop gain perturbations. From section VII-C, the robustness specification that the loop margin exceed $M_{\min}$ is equivalent to the closed loop constraint

$$\|H_{21}\|_\infty < 1/M_{\min},$$

since H_{21} is the classical sensitivity transfer function.

Robustness to additive plant perturbations. We require that our closed-loop system remain stable if P_0 is perturbed to $P_0 + \Delta P$, where ΔP is any stable transfer function with $\|\Delta P\|_\infty < D_{\min}$. From section VII-F, this constraint can be expressed in terms of the closed-loop transfer matrix H as:

$$\|H_{12}\|_\infty \le 1/D_{\min},$$

where H_{12} is the closed-loop transfer function from the sensor noise to the actuator signal. We may interpret $D = 1/\|H_{12}\|_\infty$ as the maximum size (in terms of $\|\cdot\|$) additive plant perturbation that the closed-loop system can be guaranteed to withstand while remaining stable.

A general specification composed from these can be expressed as:

$$\text{RMS}(y_p) \le \alpha,\quad \text{RMS}(u) \le \beta,\quad M \ge M_{\min},\quad D \ge D_{\min}.$$

In general, the robustness requirements that M and D be large are independent. A system may have good margins (i.e. large M), but be quite sensitive to additive plant perturbations, (i.e. small D), and vice-versa.

C. LQG: An Analytically Computable Tradeoff of RMS Regulation Versus RMS Control

We first concentrate on the first two specifications, on regulation and actuator authority. In this section we will show how LQG theory can be used to determine *analytically* exactly which specifications of the form

$$\text{RMS}(u) \le \alpha \qquad \text{RMS}(y_p) \le \beta$$

are achievable by a stabilizing regulator. The boundary between achievable and unachievable specifications (in the (α, β) plane) we refer to as the *tradeoff curve* of regulation versus actuator authority.

Given a fixed positive ρ, the LQG optimal regulator minimizes (over all stabilizing regulators) the weighted cost function

$$J = \lim_{t\to\infty} E\{y_p(t)^2 + \rho u(t)^2\} = \text{RMS}(y_p)^2 + \rho\text{RMS}(u)^2.$$

We now show that this implies that the RMS values of u and y_p achieved by the LQG regulator give a point on the tradeoff curve between these quantities. If some other LTI regulator stabilized P and achieved better (smaller) RMS values of *both* u and y_p, then it would achieve a cost J smaller than the LQG regulator (recall that $\rho > 0$). This is impossible, since the LQG optimal regulator gives the smallest J achievable by any LTI regulator that stabilizes P.

In multi-objective optimization, a point which has the property that no other point yields lower values for *every* objective, is referred to as a *Pareto optimal* point. We have just observed that the LQG regulators correspond precisely to the Pareto optimal regulators, where the two objectives are RMS regulation and RMS actuator authority [26], [82].

As ρ varies, the RMS values of u and y_p achieved by the corresponding LQG optimal regulator sweeps out the tradeoff curve, as shown in Fig. 58.

Several interpretations of the curve and shaded region in Fig. 58 can be given:

- The design specifications $\text{RMS}(y_p) \le \alpha$, $\text{RMS}(u) \le \beta$ are achievable with a regulator that stabilizes P if and only if the point (α, β) lies in the shaded region of Fig. 58.
- Every regulator K that stabilizes P achieves RMS values of y_p and u that lie in the shaded region, on or above the tradeoff curve.
- No regulator K that stabilizes P can achieve RMS values of u and y that lie below the curve. This is true for regulators of any order, designed by any method.

For example, the simple lead-lag regulator

$$K(z) = -10\,\frac{z - 0.98}{z - 0.56}$$

stabilizes P, and achieves RMS values of u and y_p of 27.55 and 5.98, respectively. This is shown in Fig. 59.

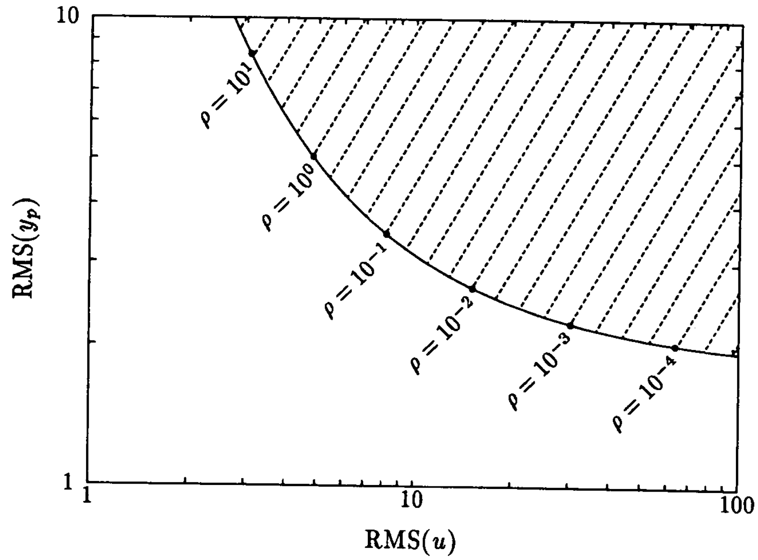


Fig. 58. Tradeoff between achievable RMS noise sensitivities.

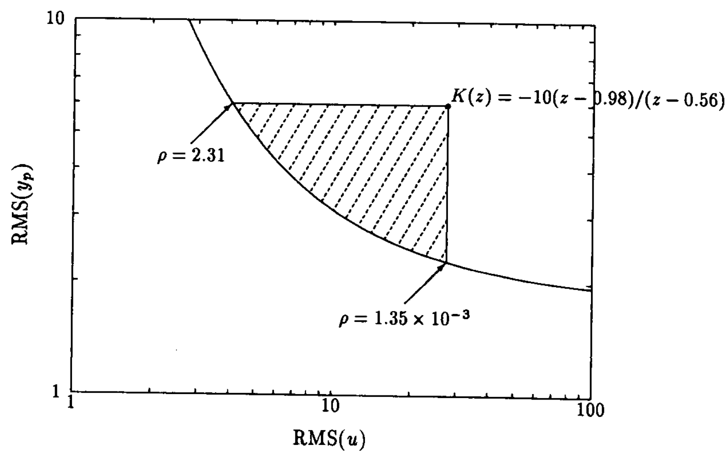


Fig. 59. A simple regulator which achieves performance above the tradeoff curve.

Any regulator $\tilde{K}$ that gives closed-loop performance in the shaded region in Fig. 59 will achieve the same or better output regulation than the simple lead-lag regulator K (RMS y_p no worse than 5.98) *and* use the same or less actuator effort than K (RMS u no worse than 27.55). One family of such regulators are the LQG optimal regulators with $1.35 \times 10^{-3} \le \rho \le 2.31$.

From Fig. 58 we may conclude that the specification

$$\mathrm{RMS}(u) \le 10, \quad \mathrm{RMS}(y_p) \le 4$$

can be achieved by a LTI regulator that stabilizes P, while the specification

$$\mathrm{RMS}(u) \le 3, \quad \mathrm{RMS}(y_p) \le 5$$

cannot be achieved by any LTI regulator that stabilizes P.

D. Tradeoff Curves Involving Noise Sensitivity and Robustness

We now examine how the four different specifications interact. To simplify matters we will lump the regulation and actuator authority into one specification, of the form

$$J = \mathrm{RMS}(y_p)^2 + \rho\,\mathrm{RMS}(u)^2 \le \alpha$$

where $\rho = 10^{-4}$. We will refer to J as the *noise sensitivity.*

If there are no constraints on robustness to plant perturbations or robustness to loop gain perturbations, then the $\rho = 10^{-4}$ LQG regulator K_{LQG} minimizes J over all stabilizing controllers. The LQG optimal (current estimator) regulator is

$$K_{LQG}(z) = -\frac{45.974z^3 - 72.79z^2 + 28.138z}{z^3 - 0.8061z^2 + 0.7107z - 0.1071}$$

$$= -\frac{45.974z(z - 0.91305)(z - 0.6703)}{(z - 0.17897)(z - 0.3136 + 0.7072j)(z - 0.3136 - 0.7072j)}.$$

For this regulator, the RMS values of u and y_p are 63.73 and 2.0184, respectively. The value of the cost function is $J = 4.48$.

For K_{LQG} we find that $\|H_{12}\|_\infty = 3.34$, so that $M = 0.30$ is the closest the Nyquist plot comes to the critical point $+1$. This can be seen from the Nyquist plot in Fig. 60, where we

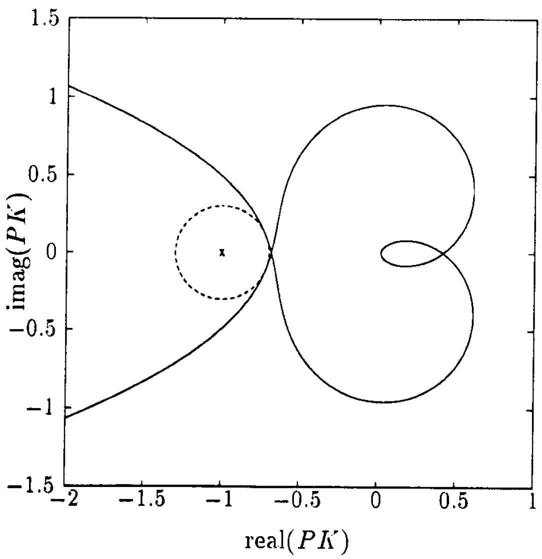


Fig. 60. Nyquist plot of $\rho = 10^{-4}$ LQG regulator, with $M = 0.30$ circle.

have plotted the loop gain in the classical style, with -1 as the critical point. The circle of radius 0.30 centered at -1 is also shown in Fig. 60. The magnitude of the sensitivity transfer function H_{12} for K_{LGQ} is shown in Fig. 61. Note that its maximum is 10.5 dB ($= -20 \log M$).

For K_{LQG}, we find that $\|H_{21}\|_\infty = 83.02$, so $D = 1/83.02 = 0.0121$. Hence, there are stable plant perturbations ΔP that destabilize the system with $\|\Delta P\|_\infty$ as small as 0.0121. One destabilizing perturbation is

$$\Delta P(z) = \frac{0.2152 \times 10^{-3}(1 - z)}{z^2 - 0.985z + 0.9801},$$

for which $\|\Delta P\|_\infty = 0.0125$, which is not much above D. The frequency response magnitude of this ΔP is plotted in Fig. 62. ΔP might represent a mechanical resonance that was ignored when the plant was modeled for the controller design.

Now we turn to the question: exactly which specifications of the form

$$J \le J_{max}, \quad D \ge D_{min}, \quad M \ge M_{min}$$

are achievable by regulators that stabilize P? Unlike the tradeoff of RMS actuator effort, no exact analytical method is known that will answer this question, although some work has been done [83]. *A major point of this paper is that it can be determined numerically.*

We computed various "slices" of the achievable region in the space of design goals (J, D, M), by fixing D_{min} and M_{min} and computing the boundary of achievable noise sensitivities, which we denote J_{min}. For three fixed values of D_{min}, the tradeoff between J_{min} and $1/M_{min}$ is shown in Fig. 63. The $\rho = 10^{-4}$ LQG regulator is also shown. Since this regulator achieves $1/D = 83.02$, it lies below the $1/D_{min} = 10$ curve, and on the $1/D_{min} = 83.02$ curve. All tradeoff curves with $1/D_{min} \ge 83.02$ will pass through the LQG performance point, and be horizontal to the right of it.

Note the interesting fact that by allowing the loop margin to be less than 0.5, only modest improvement in the noise response is gained, with the same tolerance D to additive plant perturbations.

For four fixed values of M_{min}, the tradeoff between J_{min} and $1/D_{min}$ is shown in Fig. 64. The $\rho = 10^{-4}$ LQG regulator is also shown. Since this regulator achieves $1/M = 3.34$, it lies below the $1/M_{min} = 2.0$ curve, and on the $1/M_{min} = 3.34$ curve. All tradeoff curves with $1/M_{min} \ge 3.34$ will pass through the LQG performance point, and be horizontal to the right of it.

From Fig. 64 we can draw some interesting conclusions. Consider the $1/M \le 3.34$ curve, that corresponds to regulators which yield the same or larger loop margin as the $\rho = 10^{-4}$ LQG regulator. The curve is relatively flat for $1/D \ge 20$, meaning that D can be increased to about 0.05 with relatively small increase in RMS noise response, and the same or larger loop margin (0.3). For the $\rho = 10^{-4}$ LQG regulator, this represents an increase in additive plant perturbation tolerance D of a factor of four.

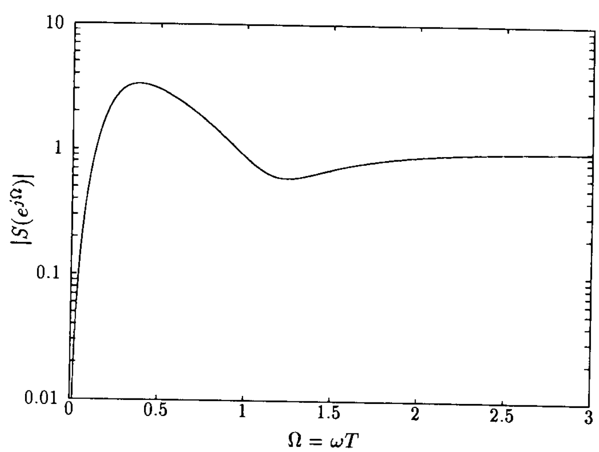


Fig. 61. Magnitude of sensitivity transfer function of $\rho = 10^{-4}$ LQG regulator.

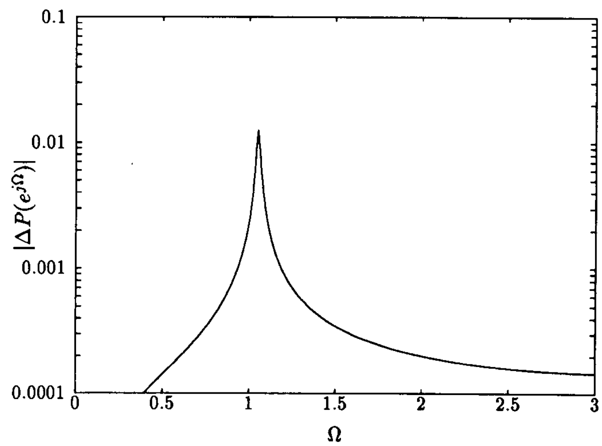


Fig. 62. Frequency response of a destabilizing ΔP.

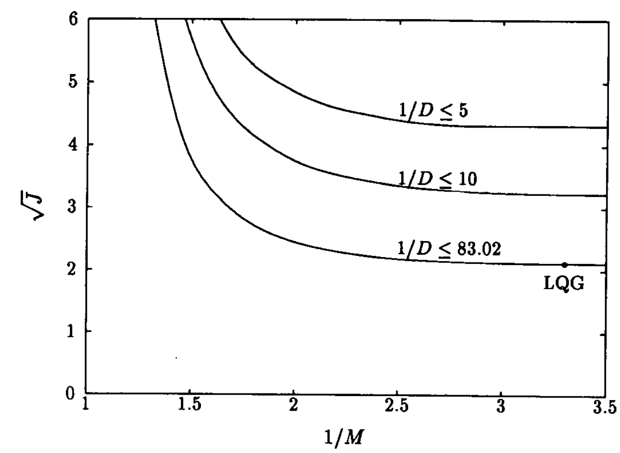


Fig. 63. Tradeoff between RMS noise cost and loop margin.

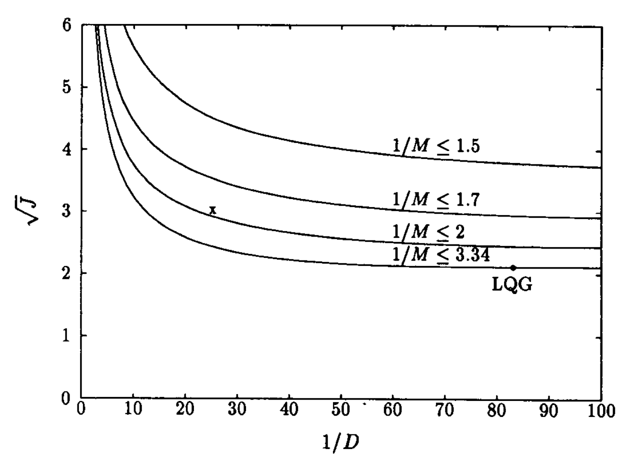


Fig. 64. Tradeoff between RMS noise and additive plant sensitivities.

Of course, all regulators that stabilize P yield a noise sensitivity $J \ge J_{LQG}$, so that all curves lie on or above the horizontal asymptote $\sqrt{J_{LQG}} = 2.117$. Imposing the robustness constraints will naturally increase the minimum noise sensitivity J achievable with LTI stabilizing regulators. What is neither intuitively obvious nor analytically computable is *how much* the minimum noise sensitivity J must increase when we impose the two robustness constraints. Figures 63 and 64 show precisely this tradeoff.

Let us give two examples of specific conclusions we may draw from Figs. 63 and 64. First, the specifications

$$\sqrt{J} \leq 3, \quad M \geq 0.5, \quad D \geq 0.04$$

can be achieved with a LTI regulator that stabilizes P (this point is marked "x" in Fig. 64). This specification represents an increase in noise response over the LQG regulator (as any achievable specification must), a moderate improvement in loop margin over the LQG controller, and a substantial improvement in D, tolerance to additive plant perturbation, over the LQG controller:

$$\sqrt{J_{\mathrm{LQG}}} = 2.117, \quad M_{\mathrm{LQG}} = 0.3, \quad D_{\mathrm{LQG}} = 0.0121.$$

Another improvement conclusion we may draw from Figs. 63 and 64 is that the design goals

$$\sqrt{J} \leq 3, \quad M \geq 0.5, \quad D \geq 0.08$$

are not achievable by any LTI regulator that stabilizes P.

E. *Comments About Finding the Limits of Performance*

We have given an example demonstrating how the methods described in this paper can be used to determine numerically the various tradeoff curves for competing design goals or objectives. It should be emphasized that the tradeoff curves shown in Figs. 63 and 64 represent *fundamental limits of performance:* they do not merely show the best we can do; they show the best anybody can do with a LTI regulator of any form or complexity, designed by any scheme or method. We believe that this information can be very useful to the designer.

No control engineer would be surprised by the general shape of the tradeoff curves shown in Figs. 63 and 64: it is intuitively obvious that some improvement in robustness can be realized at the cost of some degradation in RMS noise sensitivity. But even for this very simple, typical plant, the numerical values of the tradeoffs—how much improvement in robustness can be 'bought' for a given increase in noise response—is not at all obvious. Naturally for a complicated multi-actuator multi-sensor plant and a much larger set of design specifications, the tradeoffs would be considerably less obvious, and such computations correspondingly more valuable.

XI. Historical Notes

In this section we provide a brief history of some of the central ideas in this paper. The purpose of this section is twofold: first, to point out that some concepts relating to the central ideas of this paper substantially predate the development of "modern" linear control theory; and second, to provide additional discussion of and references for recent work related to the work in this paper.

A. *Designing the Closed-Loop Transfer Matrix to Meet Specifications*

The idea of first designing the closed-loop transfer matrix and then determining the compensation required to achieve this closed-loop transfer matrix is at least forty years old. An explicit presentation of a such a method appears in J. G. Truxal's Ph.D. thesis [84] in 1950. Chapter 5 of Truxal's 1955 *Automatic Feedback Control System Synthesis* [85] contains an exposition of this method, which Truxal attributes to Guillemin in 1947.[12] Truxal also cites a 1951 article by Aaron [86], which presents a similar method.

On pages 278–279 of Truxal's book we find:

> 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.
>
> ...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.
>
> Such an 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 over-all 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."

The method presented by Truxal consists essentially of selecting the poles and zeros of the closed-loop transfer function in such a way that the closed-loop system meets the performance specifications. He recognizes that the transfer function of what he calls the "fixed components", i.e., the plant, places restrictions on the set of achievable closed-loop transfer functions, but is not clear about what these restrictions are.

Still, a rudimentary, partial version of the interpolation conditions form of the parameterization of closed-loop transfer matrices appears in Truxal's book on pages 308–309. There he states that if the plant has a nonminimum-phase zero, so should the closed-loop transfer function ($H = PK/(1 - PK)$). His reasoning is interesting: he argues that if the closed-loop transfer function does not also vanish at the unstable plant zero, then an (open loop) unstable controller results, and "practical difficulties make this solution completely undesirable."

So Truxal has essentially the right parametrization, at least for stable plants, but not quite the right justification. He has confused several issues—in particular open-loop stability of the controller and stability of the closed-loop system.

B. *The Fundamental Problem*

An excellent discussion of what we call the Fundamental Problem appears on pages 28–34 of *Analytical Design of Linear Feedback Controls* (1957) by Newton, Gould, and Kaiser [4]. Here are some crucial passages:

> Unfortunately, the trial-and-error design method is beset with certain fundamental difficulties, which must be clearly understood and appreciated in order to employ it properly. From both a practical and theoretical viewpoint its principal disadvantages is that it cannot recognize an inconsistent set of specifications.

[12]The authors thank Professor D. D. Siljak for bringing this citation to their attention.

The analytical design procedure has several advantages over the trial and-error method, the most important of which is the facility to detect immediately and surely an inconsistent set of specifications. The designer obtains a "yes" or "no" answer to the question of whether it is possible to fulfill any given set of specifications; he is not left with the haunting thought that if he had tried this or that form of compensation he might have been able to meet the specifications.

Even if the reader never employs the analytical procedure directly, the insight that it gives him into linear system design materially assists him in employing the trial-and-error design procedure.

What Newton, Gould and Kaiser mean by "analytical design" is the following:

> In place of a relatively simple statement of the allowable error, the analytical design procedure employs a more or less elaborate *performance index*. The objective of the performance index is to encompass in a single number a quality measure for the performance of the system.

They note:

> Offsetting the above advantages of the analytical design procedure are two important disadvantages. First of all, it is unfortunate that many performance indices which have engineering usefulness lead to analytically insolvable problems. Thus one if forced to compromise his choice of performance index in order to obtain a solution to the design problem. The other major disadvantage is that... many practical problems lead to involved solutions requiring considerable numerical calculation.

This paper is about how convex optimization theory and recent developments in control theory allow us to develop a new method of dealing with the Fundamental Problem. This method, together with enormous increases in available computing power, helps us to overcome many of the stated disadvantages of analytical control design.

C. Parametrization of Closed-Loop Transfer Functions: Interpolation Conditions

The first essentially correct and explicit statement of the interpolation conditions appear in a 1956 paper by Bertram on discrete-time feedback control [87]. Another early exposition, citing Bertram, can be found in chapter 7 of Ragazzini and Franklin's 1958 book, *Sampled-Data Control Systems* [88]. In particular, on pp. 157–8 they give a complete description of the interpolation conditions (on $H = PK/(1 - PK)$) for SISO plants:

> In words, it is necessary that the specified over-all pulse transfer function [$H(z)$] contains as its zeros all those zeros of the plant pulse transfer function which lie outside or on the unit circle in the z-plane and that [$1 - H(z)$] contain as its zeros all those poles of the plant which lie outside or on the unit circle of the z-plane.

The equivalent interpolation conditions for continuous-time systems first appear in a 1958 paper by Bigelow [89].

The justification for the interpolation conditions given by Bertram is

> Unfortunately, perfect cancellation is impossible in any practical system. It has previously been shown [here Bertram cites work by Barker in 1950 [90]] that any attempt to cancel a zero of [the plant] in the z-plane outside the unit circle leads to a system which is unstable in practice.

The explanations provided by Ragazzini and Franklin and by Bigelow are essentially the same. What is interesting is that these authors have an understanding of the essential idea of internal stability for single-input single-output systems, which is that the unstable poles and zeros of the plant impose restrictions on the closed-loop transfer function. This understanding comes long before the appearance of any formal definition, and well before the widespread use of state-space methods and the notion of controllability and observability.

Youla, Bongiorno, and Jabr present the interpolation conditions as necessary for the avoidance of unstable hidden modes in 1976 [91].

For multivariable systems, the necessary appearance of non-minimum-phase plant zeros in the closed-loop transfer matrix is shown in [92]. Multivariable interpolation conditions appear in [93] and are more completely discussed in [94].

Finally, a slightly different but equivalent form of the interpolation conditions, derived from Desoer's formal definition of internal stability, is presented in a paper by Zames and Francis [95].

D. Parametrization of Closed-Loop Transfer Matrices: Factorization and MIMO Systems

The interpolation conditions for single-sensor, single-actuator systems result naturally from the idea that unstable pole-zero cancellations result in unstable closed-loop behavior. In the case of multiple sensors or multiple actuators, this kind of simple intuitive reasoning is impossible. The results on parametrization of achievable closed-loop transfer matrices in the multiple-actuator, multiple-sensor case depend on *factorizations* of transfer matrices. Early treatments of the factorization of transfer matrices in terms of matrices of polynomials are [96] and [97]; extensive discussion appears in [20]. The first parametrization of closed-loop transfer matrices achievable with MIMO controllers appears in Youla, Jabr, and Bongiorno's 1976 article on Wiener-Hopf design for MIMO controllers [98]; the factorization is in terms of matrices of polynomials. A complete discussion of achievable closed-loop transfer matrices, in terms of polynomial matrix factorizations, can be found in [94].

A more recent version of the parametrization uses factorization in terms of stable transfer matrices, and appears first in Desoer, Liu, Murray, and Saeks [39], although the idea used in Vidyasagar [99]. Vidyasagar [42, ch. 3, 5] contains a complete and thorough treatment of the parametrization of achievable closed-loop transfer matrices in terms of stable factorizations. A clear and compact exposition can be found in Francis [41, ch. 4].

E. Convex Optimization for Linear Feedback Control Design

The proposed use of convex optimization in linear feedback control design dates back at least as far as 1964, when the first of an interesting series of papers by K. A. Fegley and colleagues appeared.

In [100], Fegley presents a linear programming solution to the problem of designing a feedback controller for a discrete-time system. He restricts the closed-loop system to be deadbeat, i.e., have a transfer function of the form $\Sigma_{i=p}^{q} A_i z^{-i}$; the decision variables of the linear program are the coefficients A_i. Constraints on the difference between the output and a specific reference input appear as affine constraints on the closed-loop transfer function.

A longer paper by Porcelli and Fegley [101] expands and generalizes the approach taken in [100]. Once again, the problem is discrete-time control with the closed-loop transfer function restricted to be deadbeat, but this time a wider range of constraints is presented, including a constraint corresponding precisely to the interpolation conditions as they appear in Ragazzini and Franklin's book. So, embryonic versions of four central ideas of the present paper appear in [101]: 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 convex optimization to obtain a solution.

Another two papers in the series also consider discrete time feedback control design: in [102], quadratic programming is used to handle constrained optimization of quadratic performance indices for tracking reference signals; and in [103], quadratic programming is used for constrained optimal design for stochastic inputs with known spectra. Other papers consider related techniques for open-loop discrete-time and continuous-time control and filter-design problems [104]–[106]. A summary of most of the results of Fegley and his colleagues appear in [107].

A series of papers by Desoer and Gustafson [108]–[111] make a number of proposals for the design of multivariable feedback systems, based on modern versions of the parametrization of closed-loop transfer matrices and numerical optimization techniques. They pose the problem as one of optimizing some performance index subject to constraints representing performance or robust performance specifications. They make some highly practical suggestions for avoiding numerical difficulties by doing design based directly on measurements of the plant.

Recently, Dahleh and Pearson have presented solutions to several versions of a problem posed by Vidyasagar in [56]. The essential problem is that of finding the feedback compensator that minimizes the peak value of a regulated output in response to a bounded exogenous input. The versions solved by Dahleh and Pearson are: MIMO discrete-time plant with unknown but bounded inputs [57], [59]; SISO continuos-time plant with unknown but bounded input [58]; and SISO discrete-time plant with known, bounded input [60]. All of these are special cases of the general infinite-dimensional convex optimization problems formulated in this paper. What is remarkable about the work of Dahleh and Pearson is that in each case they show that either an exact solution or an arbitrarily accurate approximate solution can be obtained by solving a finite-dimensional linear program.

In his 1986 Ph.D. thesis, Salcudean [112] uses the parametrization of achievable closed-loop maps to formulate the multivariable feedback control design problem as a constrained convex optimization problem. His formulation allows constraints corresponding to time-domain and frequency-domain performance specifications; these constraints are very similar to some of those developed in sections VI and VII of this paper. The advantage of convex optimization over nonconvex optimization is emphasized. Salcudean also presents an interesting numerical alogirthm for solution of his optimization problems and some careful analysis of the properties of the algorithm. See also [113].

A program called qdes has been developed by the authors of this paper and colleagues [74]. It accepts input written in a *control specification language* that allows the user to describe a discrete-time control design problem in terms of many of the closed-loop convex specifications presented in this paper. The program uses a simple method to determine a finite-dimensional linear or quadratic programming approximation to the control design problem and then solves this approximation.

A recent paper by Helton and Sideris [114] considers numerical methods for minimizing the $\mathbf{H}_\infty$-norm of one closed-loop transfer function of a system subject to convex time-domain constraints on other closed-loop transfer functions. The paper explores various alternatives to the naive approximations by linear or quadratic programs used in [74].

The material of this paper will appear in more detail in [115].

XII. Conclusion

The basic points we have tried to make in this paper are:

- A sensible formulation of the controller design problem is possible only by considering simultaneously *all* of the closed-loop transfer functions of interest—what we call the closed-loop transfer matrix H. The closed-loop transfer matrix H should include every closed-loop transfer function necessary to evaluate a candidate controller or compare competing designs, without "side information" (unstated requirements on transfer functions not appearing in H). H will generally have many more entries than we have degrees of freedom to design (i.e., H will be much bigger than K).
- It is useful to associate with each design specification the set of transfer matrices that meet the design specification. These sets often have simple geometry—affine or convex. An important example is the specification that the closed-loop system should be stable—the recent *parametrization of achievable closed-loop transfer matrices* shows that this constraint is affine. One of the themes of this paper is that *many design specifications are closed-loop convex.* More explicitly, many performance specifications are closed-loop convex; some important robustness specifications are closed-loop convex, and virtually no control law specifications (important examples being open-loop stable controller, decentralized controller, and fixed order controller) are closed-loop convex.
- When the design specifications are all closed-loop convex, the Fundamental Problem can be numerically solved by solving a nondifferentiable convex program. This is an *effective* procedure, unlike local methods of parameter optimization. Parameter optimization schemes generally cannot determine that a set of specifications is infeasible—instead they simply fail to find

a feasible point. On the other hand, for a convex problem, there are effective methods to determine whether a set of specifications is infeasible or too tight.

- There are good reasons *not* to use these methods to design controllers, e.g., the inability to specify the order or structure of the controller, which are constraints readily handled by a parameter optimization method. Still, the methods described in this paper can be used to check the limits of performance achievable when there is no constraint on controller complexity (or any other non closed-loop convex constraints). This provides an absolute yardstick against which less-complex or structured controllers can be compared.
- Many of the ideas in this paper are not new. The basic idea of closed-loop design goes back to Truxall and Guilleman around 1950; the idea of using numerical convex optimization for feedback controller design goes back at least as far as the work of Fegley and colleagues in the 1960s. On the other hand, the development of theory about the set of achievable closed-loop transfer matrices is more recent. Also recent is the arrival of cheap and powerful computers and workstations for numerical computation. Finally, the idea of using numerical techniques and a general (matrix) H is, as far as we know, new.

XIII. Acknowledgment

We thank the reviewers for extensive constructive comments. The abstract is paraphrased from one reviewer's suggestions.

References

[1] H. S. Black, "Stabilized feedback amplifiers," *Bell Syst. Tech. J.*, vol. 13, pp. 1–18, 1934.
[2] O. Mayr, *The Origins of Feedback Control.* Cambridge, MA: MIT Press, 1970.
[3] L. Ljung, *System Identification: Theory for the User.* Englewood Cliffs, NJ: Prentice-Hall, Inc., 1987.
[4] G. Newton, L. Gould, and J. Kaiser, *Analytical Design of Linear Feedback Controls.* New York, NY: Wiley, 1957.
[5] H. W. Bode, *Network Analysis and Feedback Amplifier Design.* New York, NY: Van Nostrand, 1945.
[6] I. M. Horowitz, *Synthesis of Feedback Systems.* New York, NY: Academic Press, 1963.
[7] I. Horowitz, "Quantitative feedback theory," *IEE Proc.*, vol. 129-D, pp. 215–226, Nov. 1982.
[8] A. G. J. MacFarlane and B. Kouvaritakis, "A design technique for linear multivariable feedback systems," *Int. J. Control*, vol. 25, no. 6, pp. 875–883, 1977.
[9] A. E. Bryson and Y. C. Ho, *Applied Optimal Control.* New York, NY: Hemisphere Publishing Co., 1975.
[10] H. Kwakernaak and R. Sivan, *Linear Optimal Control Systems.* New York, NY: Wiley, 1972.
[11] M. Athans and P. Falb, *Optimal Control.* New York: McGraw-Hill, 1966.
[12] R. C. Dorf, *Modern Control Systems.* Reading, MA: Addison-Wesley, 2nd ed., 1980.
[13] G. F. Franklin, J. D. Powell, and A. Emani-Naeni, *Feedback Control of Dynamic Systems.* Reading, MA: Addison-Wesley, 1986.
[14] B. C. Kuo, *Automatic Control Systems.* Englewood Cliffs, NJ: Prentice-Hall, 4th ed., 1982.
[15] K. Ogata, *Modern Control Engineering.* Englewood Cliffs, NJ: Prentice-Hall, 1970.
[16] I. Postlethwaite and A. MacFarlane, *A Complex Variable Approach to the Analysis of Linear Multivariable Feedback Systems.* Berlin, West Germany: Springer-Verlag, 1979.
[17] J. S. Freudenberg and D. P. Looze, *Frequency Domain Properties of Scalar and Multivariable Feedback Systems.* Lecture Notes in Control and Information Sciences. Berlin, West Germany: Springer-Verlag, 1988.
[18] L. A. Zadeh and C. A. Desoer, *Linear System Theory: The State Space Approach.* New York, NY: McGraw-Hill, 1963.
[19] R. W. Brockett, *Finite Dimensional Linear Systems.* New York, NY: Wiley, 1970.
[20] T. Kalaith, *Linear Systems.* Englewood Cliffs, NJ: Prentice-Hall, 1980.
[21] C. T. Chen, *Linear System Theory and Design.* New York, NY: Holt, Rinehart and Winston, 1984.
[22] R. E. Kalman and R. S. Bucy, "New results in linear filtering and prediction theory," *Trans. ASME, Series D*, vol. 83, pp. 95–107, Dec. 1961.
[23] J. C. Doyle and G. Stein, "Multivariable feedback design: Concepts for a classical/modern synthesis," *IEEE Trans. Aut. Control*, vol. AC-26, pp. 4–16, Feb. 1981.
[24] G. Stein and M. Athans, "The LQG/LTR procedure for multivariable feedback control design," *IEEE Trans. Aut. Control*, vol. 32, pp. 105–114, Feb. 1987.
[25] U. Ly, A. E. Bryson, and R. H. Cannon, "Design of low-order compensators using parameter optimization," in *Applications of Nonlinear Programming to Optimization and Control.* Laxenburg, Austria: IFAC, 1983.
[26] P. M. Mäkilä and H. T. Toivonen, "Computational methods for parametric LQ minimization: A survey," *IEEE Trans. Aut. Control*, vol. AC-32, pp. 658–671, Aug. 1987.
[27] D. Gangsaas, K. Bruce, J. Blight, and U. Ly, "Application of modern synthesis to aircraft control: three case studies," *IEEE Trans. Aut. Control*, vol. AC-31, pp. 995–1014, Nov. 1986.
[28] E. Polak, D. Q. Mayne, and D. M. Stimler, "Control system design via semi-infinite optimization: A review," *Proc. IEEE*, vol. 72, no. 12, pp. 1777–1794, Dec. 1984.
[29] E. Polak, P. Siegel, T. Wuu, W. T. Nye, and D. Q. Mayne, "DELIGHT.MIMO: An interactive, optimization-based multivariable control system design package," in *Computer-Aided Control Systems Engineering*, M. Jamshidi and C. J. Herget, Eds. Amsterdam, The Netherlands: North-Holland, 1985. Reprinted from *IEEE Cont. Sys. Mag.*, no. 4, (Dec. 1982).
[30] M. K. H. Fan, L. Wang, J. Joninckx, and A. Tits, "Software package for optimization-based design with user-supplied simulators," *IEEE Cont. Syst. Mag.*, vol. 9, no. 1, pp. 66–71, 1989.
[31] P. E. Gill, W. Murray, and M. Wright, *Practical Optimization.* London, UK: Academic Press, 1981.
[32] L. S. Pontryagin, *Optimal Regulation Processes.* vol. 18 of 20, American Mathematical Society Translations, 1981.
[33] L. Hunt, R. Su, and G. Meyer, "Global transformations of nonlinear systems," *IEEE Trans. Aut. Control*, vol. AC-28, pp. 24–31, 1983.
[34] A. Isadori, *Nonlinear Control Systems: An Introduction.* Berlin, West Germany: Springer-Verlag, 1985.
[35] M. Fliess and M. Hazewinkel, Eds., *Algebraic and Geometric Methods in Non-linear Control Theory.* Dordrecht Riedel, 1985.
[36] S. Sastry and M. Bodson, *Adaptive Control: Stability, Convergence, and Robustness.* Englewood Cliffs, NJ: Prentice-Hall, 1988.
[37] G. Strang, *Linear Algebra and its Applications.* New York, NY: Academic Press, 1980.
[38] D. G. Luenberger, *Optimization by Vector Space Methods.* New York, NY: Wiley, 1969.
[39] C. A. Desoer, R. Liu, J. Murray, and R. Saeks, "Feedback system design: The fractional representation approach to analysis and synthesis," *IEEE Trans. Aut. Control*, vol. AC-25, pp. 399–412, June 1980.
[40] C. A. Desoer and W. S. Chan, "The feedback interconnection of lumped linear time-invariant systems," *J. Franklin Inst.*, vol. 300, pp. 335–351, Nov.-Dec. 1975.
[41] B. A. Francis, *A course in* $\mathbf{H}_\infty$ *Control Theory.* vol. 88 of Lecture Notes in Control and Information Sciences. Berlin, West Germany: Springer-Verlag, 1987.
[42] M. Vidyasagar, *Control System Synthesis: A Factorization Approach.* Cambridge, MA: MIT Press, 1985.
[43] F. M. Callier and C. A. Desoer, *Multivariable Feedback Systems.* Berlin, West Germany: Springer-Verlag, 1982.
[44] A. Bhaya and C. A. Desoer, "Necessary and sufficient con-

ditions on $Q(=C(I+PC)^{-1})$ for stabilization of the linear feedback system $S(P, C)$," *Syst. Control Lett.*, vol. 7, pp. 35–38, Feb. 1986.

[45] G. Zames, "Feedback and optimal sensitivity: Model reference transformations, multiplicative seminorms, and approximate inverses," *IEEE Trans. Aut. Control*, vol. AC-26, pp. 301–320, April 1981.

[46] J. C. Doyle, Matrix interpolation theory and optimal control. Ph.D. thesis, University of California, Berkeley, 1984.

[47] B. D. O. Anderson, N. K. Bose, and E. I. Jury, "Output feedback stabilization and related problems—Solution via decision methods," *IEEE Trans. Aut. Control*, vol. AC-20, pp. 53–66, Feb. 1975.

[48] N. K. Bose, *Applied Multidimensional System Theory.* New York, NY: Van Nostrand Rheinhold, 1982.

[49] B. D. O. Anderson and J. B. Moore, *Optimal Filtering.* Englewood Cliffs, NJ: Prentice-Hall, 1979.

[50] P. E. Caines, *Linear Stochastic Systems.* New York: Wiley, 1988.

[51] J. Doyle, K. Glover, P. P. Khargonekar, and B. A. Francis, "State-space solutions to standard $\mathbf{H}_2$ and $\mathbf{H}_\infty$ control problems," *IEEE Trans. Aut. Control*, vol. 34, pp. 831–847, Aug. 1989.

[52] K. J. Åström and B. Wittenmark, *Computer Controlled Systems.* Englewood Cliffs, NJ: Prentice-Hall, 1984.

[53] B. A. Francis, J. W. Helton, and G. Zames, "$\mathbf{H}_\infty$-Optimal feedback controllers for linear multivariable systems," *IEEE Trans. Aut. Control*, vol. AC-29, pp. 888–900, Oct. 1984.

[54] B. A. Francis and J. C. Doyle, "Linear control theory with an $\mathbf{H}_\infty$ optimality criterion," *SIAM J. Control and Optimization*, vol. 25, pp. 815–844, July 1987.

[55] S. Boyd and J. Doyle, "Comparison of peak and RMS gains for discrete-time systems," *Syst. Control Lett.*, vol. 9, pp. 1–6, June 1987.

[56] M. Vidyasagar, "Optimal rejection of persistent bounded disturbances," *IEEE Trans. Aut. Control*, vol. AC-31, pp. 527–535, June 1986.

[57] M. A. Dahleh and J. B. Pearson, Jr., "$\mathbf{l}_1$-Optimal feedback controllers for MIMO discrete-time systems," *IEEE Trans. Aut. Control*, vol. AC-32, pp. 314–322, Apr. 1987.

[58] M. A. Dahleh and J. B. Pearson, Jr., "$\mathbf{L}_1$-Optimal compensators for continuous-time systems," *IEEE Trans. Aut. Control*, vol. AC-32, pp. 889–895, Oct. 1987.

[59] M. A. Dahleh and J. B. Pearson, Jr., "Optimal rejection of persistent disturbances, robust stability, and mixed sensitivity minimization," *IEEE Trans. Aut. Control*, vol. AC-33, pp. 722–731, Aug. 1988.

[60] M. A. Dahleh and J. B. Pearson, Jr., "Minimization of a regulated response to a fixed input," *IEEE Trans. Aut. Control*, vol. AC-33, pp. 924–930, Oct. 1988.

[61] D. Graham and R. C. Lathrop, "The synthesis of "optimum" transient response: Criteria and standard forms," *AIEE Transactions*, vol. 69, pp. 273–288, Nov. 1953.

[62] G. Zames, "On the input-output stability on nonlinear time-varying feedback systems—Part II: conditions involving circles in the frequency plane and sector nonlinearities," *IEEE Trans. Aut. Control*, vol. AC-11, pp. 465–476, July 1966.

[63] I. W. Sandberg, "A frequency-domain condition for the stability of feedback systems containing a single time-varying nonlinear element," *Bell Syst. Tech. J.*, vol. 43, pp. 1601–1609, July 1964.

[64] K. S. Narendra and R. M. Goldwyn, "A geometrical criterion for the stability of certain nonlinear nonautonomous systems," *IEEE Trans. Circuit Theory*, vol. CT-11, pp. 406–408, Sept. 1964.

[65] C. A. Desoer and M. Vidyasagar, *Feedback Systems: Input-Output Properties.* New York, NY: Academic Press, 1975.

[66] M. Vidyasagar, *Nonlinear Systems Analysis.* Englewood Cliffs, NJ: Prentice-Hall, 1978.

[67] M. G. Safonov, *Stability and Robustness of Multivariable Feedback Systems.* Cambridge, MA: MIT Press, 1980.

[68] M. G. Safonov and M. Athans, "A multiloop generalization of the circle criterion for stability margin analysis," *IEEE Trans. Aut. Control*, vol. AC-26, pp. 415–422, 1981.

[69] M. G. Safonov, "Stability margins of diagonally perturbed multivariable feed-back systems," *IEE Proc.*, vol. 129-D, pp. 251–256, 1982.

[70] P. Dorato, Ed., *Robust Control.* New York, NY: IEEE Press, 1987.

[71] G. Zames, "On the input-output stability on nonlinear time-varying feedback systems—Part I: conditions derived using concepts of loop gain, conicity, and positivity," *IEEE Trans. Aut. Control*, vol. AC-11, pp. 228–238, Apr. 1966.

[72] J. C. Doyle, "Analysis of feedback systems with structured uncertainties," *IEE Proc.*, vol. 129-D, pp. 242–250, Nov. 1982.

[73] B. A. Francis, Snippets of $\mathbf{H}_\infty$ control theory," Tech. Rep. 8802, Systems Control Group, Department of Electrical Engineering, University of Toronto, Toronto, Canada, July 1988.

[74] S. Boyd, V. Balakrishnan, C. Barratt, N. Khraishi, X. Li, D. Meyer, and S. Norman, "A new CAD method and associated architectures for linear controllers," *IEEE Trans. Aut. Control*, vol. AC-33, pp. 268–283, Mar. 1988.

[75] S. Boyd, V. Balakrishnan, and P. Kabamba, "On computing the $\mathbf{H}_\infty$-norm of a transfer matrix," *Mathematics of Control, Signals, and Systems*, vol. 2, pp. 207–219, 1989.

[76] S. Boyd and V. Balakrishnan, "A regularity result for the singular values of a transfer matrix and a quadratically convergent algorithm for computing its $\mathbf{L}_\infty$-norm," *Systems Control Lett.*, 1990.

[77] N. Z. Shor, *Minimization Methods for Non-differentiable Functions.* Springer Series in Computational Mathematics. Berlin: Springer-Verlag, 1985.

[78] P. M. Pardalos and J. B. Rosen, *Constrained Global Optimization: Algorithms and Applications.* New York, NY: Springer-Verlag, 1987.

[79] A. S. Nemirovsky and D. B. Yudin, *Problem Complexity and Method Efficiency in Optimization.* New York, NY: Wiley, 1983.

[80] C. Barratt and S. Boyd, "Example of exact tradeoffs in linear controller design," *IEEE Cont. Syst. Mag.*, vol. 9, pp. 46–52, Jan. 1989.

[81] D. Flamm, S. Boyd, G. Stein, and S. Mitter, "Tutorial workshop on $\mathbf{H}_\infty$-optimal control," Dec. 1987. Preconference workshop, CDC.

[82] P. Khargonekar and M. Rotea, "Optimal control with multiple objectives: The $\mathbf{H}_2$ case," in *Proc. American Control Conf.* (Pittsburgh, PA), pp. 171–176, 1989.

[83] D. Bernstein and W. Haddad, 'LQG control with an $\mathbf{H}_\infty$ performance bound: a Riccati equation approach," *IEEE Trans. Aut. Control*, vol. AC-34, no. 3, pp. 293–305, 1989.

[84] J. G. Truxal, "Servomechanism synthesis through pole-zero configurations," Tech. Rep. 162, M.I.T. Research Laboratory of Electronics, Cambridge, MA, Aug. 1950.

[85] J. G. Truxal, *Automatic Feedback Control System Synthesis.* New York, NY: McGraw-Hill, 1955.

[86] M. R. Aaron, "Synthesis of feedback control systems by means of the closed loop function," *Trans. AIEE*, vol. 70, pp. 1439–1445, 1951.

[87] J. E. Bertram, "Factors in the design of digital controllers for sampled-data feedback systems," *Trans. AIEE*, vol. 75, pp. 151–59, July 1956.

[88] J. R. Ragazzini and G. F. Franklin, *Sampled-Data Control Systems.* New York, NY: McGraw-Hill, 1958.

[89] S. C. Bigelow, "The design of analog computer compensated control systems," *Trans. AIEE*, vol. 77, pp. 409–15, Nov. 1958.

[90] R. H. Barker, "The theory of pulse-monitored servos and their use for prediction," Tech. Rep. 1046, Signals Research and Development Establishment, Christchurch, England, Nov. 1950.

[91] D. C. Youla, J. J. Bongiorno, Jr., and H. A. Jabr, "Modern Wiener-Hopf design of optimal controllers—Part I: The single-input-output case," *IEEE Trans. Aut. Control*, vol. AC-21, pp. 3–13, Feb. 1976.

[92] M. K. Sain, B. F. Wyman, R. R. Gejji, P. J. Antsaklis, and J. L. Peczkowski, "The total synthesis problem of linear multivariable control—Part I: Nominal design," in *Proc. Joint American Control Conf.*, 1981.

[93] P. Antsaklis and M. Sain, "Unity feedback compensation of unstable plants," in *Proc. IEEE Conf. on Decision and Control*, 1981.

[94] P. J. Antsaklis and M. K. Sain, ch. Feedback Controller Parameterizations: Finite Hidden Modes and Causality,

Multivariable Control. Berlin, West Germany: D. Reidel Publishing Company, 1984, pp. 85–104.

[95] G. Zames and B. A. Francis,"Feedback, minimax sensitivity and optimal robustness," *IEEE Trans. Aut. Control*, vol. AC-28, pp. 585–601, May 1983.

[96] H. H. Rosenbrock, *State-space and Multivariable Systems.* New York, NY: Wiley, 1970.

[97] W. A. Wolovich, *Linear Multivariable Systems.* Berlin, West Germany: Springer-Verlag, 1974.

[98] D. C. Youla, H. A. Jabr, and J. J. Bongiorno, Jr., "Modern Wiener–Hopf design of optimal controllers—Part II: The multivariable case," *IEEE Trans. Aut. Control*, vol. AC-21, pp. 319–338, June 1976.

[99] M. Vidyasagar, "Input-output stability of a broad class of linear time-invariant multivariable sytems," *SIAM J. Control*, vol. 10, pp. 203–209, Feb. 1972.

[100] K. A. Fegley, "Designing sampled-data control systems by linear programming," *IEEE Trans. on Appl. and Ind.*, vol. 83, pp. 198–200, May 1964.

[101] G. Porcelli and K. A. Fegley, "Linear programming design of digitally compensated systems," in *Proc. Joint American Control Conf.*, 1964.

[102] G. Porcelli and K. A. Fegley, "Optimal design of digitally compensated systems by quadratic programming," *J. Franklin Inst.*, vol. 282, pp. 303–317, 1966.

[103] A. J. Calise and K. A. Fegley, "Quadratic programming in the statistical design of sampled-data control systems," *IEEE Trans. Aut. Control*, pp. 77–80, Feb. 1968.

[104] K. A. Fegley and M. I. Hsu, "Optimum discrete control by linear programming," *IEEE Trans. Aut. Control*, pp. 114–115, Jan. 1965.

[105] J. O. Bergholm and K. A. Fegley, "Time domain synthesis using linear and quadratic programming," *IEEE International Convention Record*, vol. 15, pp. 121–126, 1967.

[106] J. E. Marowitz and K. A. Fegley, "System design using quadratic programming," *IEEE Trans. Aut. Control*, vol. AC-16, pp. 241–247, June 1971.

[107] K. A. Fegley, S. Blum, J. Bergholm, A. J. Calise, J. E. Marowitz, G. Porcelli, and L. P. Sinha, "Stochastic and deterministic design and control via linear and quadratic programming," *IEEE Trans. Aut. Control*, vol. AC-16, pp. 759–766, Dec. 1971.

[108] C. L. Gustafson and C. A. Desoer, "A CAD methodology for linear multivariable systems based on algebraic theory," in *Proc. IEEE Conf. on Decision and Control*, 1983.

[109] C. L. Gustafson and C. A. Desoer, "Controller design for linear multivariable feedback systems with stable plants, using optimization with inequality constraints," *Int. J. Control*, vol. 37, no. 5, pp. 881–907, 1983.

[110] C. A. Desoer and C. L. Gustafson, "Design of multivariable feedback systems with simple unstable plant," *IEEE Trans. Aut. Control*, vol. AC-29, pp. 901–908, Oct. 1984.

[111] C. L. Gustafson and C. A. Desoer, "A CAD methodology for linear multivariable feedback systems based on algebraic theory," *Int. J. Control*, vol. 41, no. 3, pp. 653–675, 1985.

[112] S. E. Salcudean, Algorithms for Optimal Design of Feedback Compensators. Ph.D. thesis, University of California, Berkeley, 1986.

[113] E. Polak and S. Salcudean, "On the design of linear multivariable feedback systems via constrained nondifferentiable optimization in $\mathbf{H}_\infty$ spaces," *IEEE Trans. Aut. Control*, vol. AC-34, no. 3, pp. 268–276, 1989.

[114] J. W. Helton and A. Sideris, "Frequency response algorithms for $\mathbf{H}_\infty$ optimization with time domain constraints," *IEEE Trans. Aut. Control*, vol. 34, pp. 427–434, Apr. 1989.

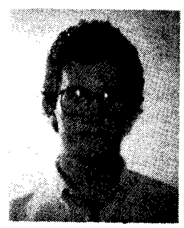
[115] S. Boyd and C. Barratt, *Linear Controller Design: Limits of Performance.* Prentice-Hall, 1990.

Stephen Boyd received the A.B. degree in mathematics from Harvard University, Cambridge, MA, in 1980, and the Ph.D. degree in electrical engineering and computer science from the University of California, Berkeley, in 1985.

Since 1985, he has been an Assistant Professor in the Information Systems Laboratory of the Department of Electrical Engineering at Stanford University, Stanford, CA.

Craig Barratt received the B.Sc. degree in mathematics and physics in 1983 and the B.E. degree (Hons.) in electrical engineering in 1985 from the University of Sydney, Sydney, Australia, and the M.S. degree in electrical engineering from Stanford University, Stanford, CA, in 1987.

He is currently pursuing a Ph.D. degree in electrical engineering at Stanford.



Stephen Norman received the B.Sc. (Eng.) degree in mathematics and engineering from Queen's University, Kingston, Ontario, Canada, in 1985, and the M.S. degree in electrical engineering from Stanford University, Stanford, CA, in 1986.

He is currently pursuing a Ph.D. degree in electrical engineering at Stanford.