Chapter 4

Optimization

In this chapter we discuss performance functions and how to use them as objective functions in optimization problems and . These concepts are applied to system design in examples. We begin with a review. A reader who is interested in getting quickly to a point where examples can be solved by computer should read sections 4.1 — 4.4 and then skip to Chapter 5.

4.1 Review of concepts

Consider the closed loop system $\mathcal{S}$ depicted in Fig.4.1.

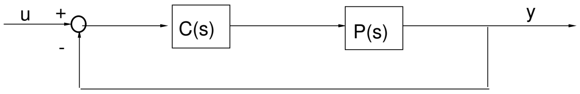


Figure 4.1: The closed loop system $\mathcal{S}$.

Here $P(s)$ and $C(s)$ are proper real rational functions of s. The function $P(s)$ is given, and the function $C(s)$ depends on the choice of designable transfer function $T(s)$. The functions $T(s)$ and $C(s)$ are related by the formula

$$T(s) = \frac{P(s)C(s)}{1+P(s)C(s)}$$

or equivalently,

$$P(s)C(s) = \frac{T(s)}{1-T(s)}.$$

The system $\mathcal{S}$ is internally stable if T has no RHP poles, and there is no RHP pole-zero cancellation in the product PC.

Performance requirements are given inequalities to be satisfied by functions associated with the system $\mathcal{S}$. The inequalities may be stated in terms of time or frequency. Frequency domain performance requirements were dealt with in Chapter 3. Time domain performance requirements are briefly discussed in Chapter 5 and treated in more detail in Chapter 6.

The basic control problem we consider is

Design

Given a plant P, a set of performance requirements $\mathcal{P}$ and a set of feasible functions

$$\mathcal{I} = \{T \in \mathcal{RH}^\infty \ : \ T \text{ is the closed loop of an internally stable system } \mathcal{S} \text{ with plant } P\}$$

determine if there exists $T \in \mathcal{I}$ satisfies all the performance requirements in $\mathcal{P}$. If the answer is yes, find one of such T.

The method used to solve *Design* is as follows:

I. Obtain a mathematical description of the performance and internal stability requirements (write down disk inequalities, take notice of the unstable poles and zeros of the plant).

II. Obtain a performance function (e.g., derive a formula for the radius and center functions).

III. Find a feasible T that yields the best performance.

Often in practice steps I – III above are followed by step IV below.

IV. If many satisfactory T exist or none exist, tighten or loosen the specifications accordingly and go to step I. Stop if this process can not be carried further.

In this chapter we will emphasize steps II and III, especially when the performance function is of the "circular" type (for a definition see Subsection 4.2). Step III requires the use of computer software, since numerical calculations are involved [1]. Step I was treated in Chapters 2 and 3, and step IV is illustrated in the problems solved in Chapter 5. See Chapter 5 for a discussion of performance functions that are not circular.

---

[1] The examples in this chapter were solved with the software package OTPDesign.

4.2 Generating a performance function

Let us suppose the we have one or more performance requirements, that can be expressed as disk inequalities. We consider in this section the case when all of these can be put together as a single disk inequality, valid over all frequencies:

$$|K(j\omega) - T(j\omega)| \leq R(j\omega) \qquad \forall\omega \tag{4.1}$$

We will build a "performance function" from (4.1) that can be used to solve design problems. Begin by rewriting (4.1) in the following way:

$$\frac{1}{R(j\omega)^2}|k(j\omega) - T(j\omega)|^2 \leq 1 \ \forall\omega \tag{4.2}$$

Now for a given T calculate the largest value of the left-hand-side of (4.2):

$$\gamma(T) = \sup_{\omega} \frac{1}{R(j\omega)^2}|k(j\omega) - T(j\omega)|^2 \tag{4.3}$$

If a particular T is available, then one can check if the original performance requirements are satisfied. To do it one just calculates $\gamma(T)$. It is not hard for the reader to convince himself that two possible outcomes of calculating $\gamma(T)$ are:

$$\begin{aligned} \gamma(T) \leq 1 &\quad \Rightarrow \quad T \text{ satisfies (4.1)} \\ \gamma(T) > 1 &\quad \Rightarrow \quad T \text{ does not satisfy (4.1)} \end{aligned} \tag{4.4}$$

Example. Suppose that the center k and radius r are given by

$$\begin{aligned} k(\omega) &= -1/2\tfrac{j\omega}{(j\omega-1)^2} \\ r(\omega) &= \tfrac{2}{(\omega^2+1)^{1/2}} \end{aligned} \tag{4.5}$$

We now check if

$$T(j\omega) = \frac{1}{j\omega + 1}$$

satisfies the disk inequality (4.1):

$$\begin{aligned} g(\omega) &:= \tfrac{1}{r(j\omega)^2}|k(j\omega) - T(j\omega)|^2 \\ &= \sup_{\omega} \tfrac{(\omega^2+1)}{4}| - 1/2\tfrac{j\omega}{(j\omega-1)^2} - \tfrac{1}{(\omega^2+1)^{1/2}}|^2 \\ &= \tfrac{4+21\omega^2+9\omega^4}{16(1+\omega^2)^2} \end{aligned} \tag{4.6}$$

After some calculations one obtains

$$\frac{dg}{d\omega} = \frac{13\omega - 3\omega^3}{8(1+\omega^2)^3} \tag{4.7}$$

Thus $g(\omega)$ has three critical points $\omega = 0$, $\omega = \pm\sqrt{13}/3$. The value $g(0) = 1/4$ is a minimum, and $g(\pm\sqrt{13}/3) = 297/512$ is a maximum. Also, $g(\omega) \to 9/16$ as $\omega \to \infty$. A plot is most informative (see Fig. 4.2). Thus

$$\gamma(T) = \sup_{\omega} g(\omega) = \frac{297}{512} < 1$$

We conclude that T does satisfy the disk inequality (4.1).

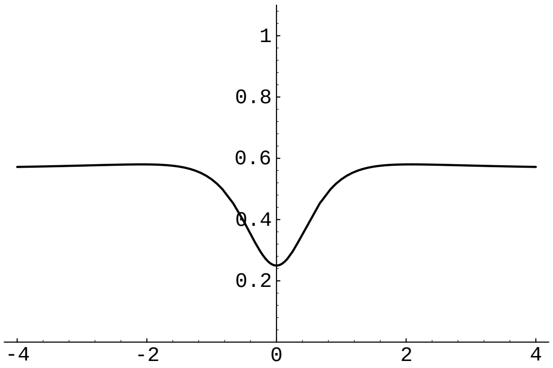


Figure 4.2: Plot of the function $g(\omega) = 1/r(\omega)^2|k(j\omega) - T(j\omega)|^2$.

For designable T and given center and radius functions k and r respectively, the expression

$$\frac{1}{R(j\omega)^2}|k(j\omega) - T(j\omega)|^2 \tag{4.8}$$

is called *circular performance function*, or simply *performance function*. The number $\gamma(T)$ is the *worst case performance of* T, but most of the time we will simply call it *performance of* T.

4.3 Finding T with best performance

A feasible function T^* with the property that

$$\gamma(T^*) \leq \gamma(T) \qquad \forall T \text{ feasible} \tag{4.9}$$

is called *optimal.* We reserve the symbol γ^* for the "optimal performance", that is, the performance of the optimal designable transfer function:

$$\gamma^* = \gamma(T^*) \tag{4.10}$$

The optimal designable function T^* is a precious object since there is no other T that has better performance. By calculating the best performance possible γ^*, the designer has conquered a peak. This vantage point offers information that is cruxial in making decisions on the next steps. If T^* satisfies the constraints one can use T^* as the choice of design, or modify T to get a low order model that still satifies the constraints. If T^* does not satisfy the constraints, then is immediately known that there is no T that satisfies the constraints! The designer then can take other actions such as to modify the "soft" constraints and then do the calculation again.

The process of finding the optimal T^*, and the corresponding optimal performance γ^* is called *optimization.* The optimal T minimizes $\gamma(T)$ over all possible T (recall that this includes internal stability). More precisely, we say that T^* is a solution to

$$\gamma^* \;=\; \inf_T \; \sup_\omega \; \frac{1}{R(j\omega)^2}|k(j\omega) - T(j\omega)|^2 \tag{4.11}$$

Because the optimization is a complicated procedure, to carry it out one needs a computer with the appropiate software. When computer algorithms and software are not available to do the optimization, one has to resort to unsatisfactory approaches, such as writing a formula for T as a rational functions with given order and unknown coefficients. Then one tries to do the get a good T by simply trying lots of possible choices of coefficients. These kind of methods are not always practical, and more importantly, the designer usually gets no indication of how far from optimal the choice of T is.

4.3.1 Example

Physically the problem is to stabilize the plant and impose closed loop gain-phase margin and roll-off performance.

Problem: Solve *Design* for the plant $P(s) = (s+5)/(s+1)^2$, with performance requirements gain-phase margin $m = 0.5$ and closed loop roll-off $|T(j\omega)| < |P(j\omega)|$, for $1 < \omega$.

Solution. Observe that the plant is strictly proper and stable. At this point we will not spend more time discussing internal stability, since the software we intend to use to do the optimization takes care of this automatically.

We first construct the center and radius functions for our problem. Recall that the gain-phase margin constant m is the closest acceptable distance from the function $L = PC$ to the point -1, or in other words

$$|1 - T(j\omega)| < \frac{1}{0.5} = 2, \text{ for all } \omega$$

The requirement on the rolloff can be written as

$$|T(j\omega)| < 0.707|P(j\omega)|, \text{ for } 1 < \omega$$

Thus the frequency domain performance requirements are described by (see Fig. 4.3):

$$\begin{array}{rcll} |1 - T(j\omega)| & < & 2, & \text{for } 0 \leq \omega \leq 1 \\ |T(j\omega)| & < & 0.707|P(j\omega)|, & \text{for } 1 < \omega \end{array} \tag{4.12}$$

The inequalities in (4.12) can be cast as a single disk inequality

$$|K(j\omega) - T(j\omega)| < R(j\omega)$$

where

$$K(j\omega) = \begin{cases} 1 & \text{if } |\omega| < 1 \\ 0 & \text{if } |\omega| \geq 1 \end{cases} \tag{4.13}$$

and

$$R(j\omega) = \begin{cases} 2.0 & \text{if } 0 \leq |\omega| < 1 \\ 0.707|P(j\omega)| & \text{if } 1 < |\omega| \end{cases} \tag{4.14}$$

The optimal T^* is the solution to

$$\gamma^* = \inf_{T \in \mathcal{I}} \sup_{\omega} \frac{1}{R(j\omega)^2} |K(j\omega) - T(j\omega)|^2 \tag{4.15}$$

To give an idea of what the answer is, some results of calculations with the software package OPTDesign are presented below. Numerical issues have not been considered here, so the reader interested in those should read Part II of this book.

Calculations by computer give $\gamma^* = 0.09923$, so there exist solutions to *Design*. Here is a rational function that is a low order approximation to the optimal one:

$$T_0(s) = \frac{5 - s}{(s+1)^2} \frac{(0.0311 + 0.2613\, s)}{(0.8809 + s)} \tag{4.16}$$

See Figures 4.4 – 4.7 for plots of the optimal T^*.

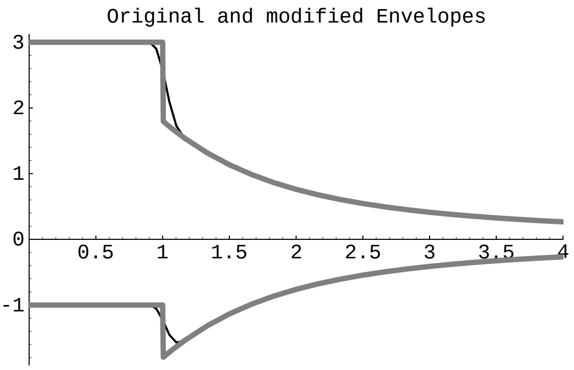


Figure 4.3: The envelope defined by center and radius functions in in Example 1 is given by the thick gray lines. The thin black line the result of automatic smoothing, a modification that is performed to improve the numerics on the computer. The horizontal axis is frequency ω, and the two functions are $K(j\omega) + R(j\omega)$ and $K(j\omega) - R(j\omega)$.

4.4 Good performance functions

There are two basic rules that have to be followed to define a useful performance function. These are:

Rule 1. The performance at any T and any frequency ω (including $\omega = \infty$) should be a well defined real number[2].

Rule 2. For any fixed frequency ω, the performance as a function of T cannot be independent of T.

The justification of rules 1 and 2 requires the results in Part III of this book, so we refer the interested reader to it.

The following are examples of performance functions that are common in the engineering literature. We assume that the underlying design problem has internal stability requirements, and that the plant P is strictly proper, and has no poles or zeros on the $j\omega$ axis. Also, W, W_1, W_2 denote given rational weight functions of frequency.

1. $\Gamma(\omega, T(j\omega)) = W(j\omega)|T(j\omega)|$. If the control problem has a plant P with relative degree p, then W must have relative degree $-p$ (or larger) for the

---

[2] The performance may turn out to be infinitely large at frequencies ω_0 such that the plant has a zero at $s = j\omega_0$, or at infinite frequency. This should not be a problem, provided only T that give internally stable systems are considered.

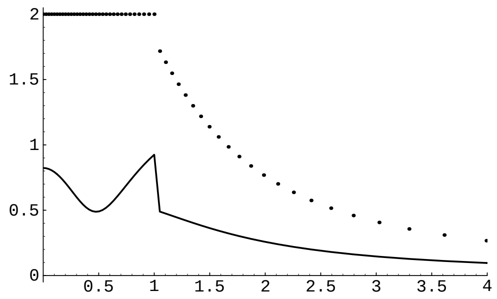


Figure 4.4: The dots represent the radius function $R(j\omega)$. The solid line represents the magnitude of the error function $|K(j\omega)-T_0(j\omega)|$. Frequency ω in on the horizontal axis. Clearly T satisfies the constraints.

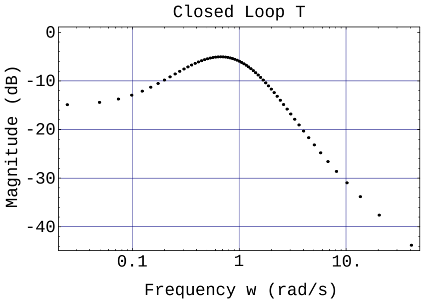


Figure 4.5: Magnitude of T_0.

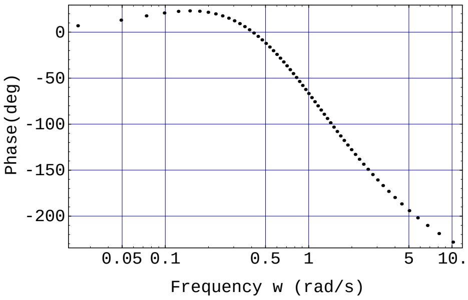


Figure 4.6: Phase of T_0.

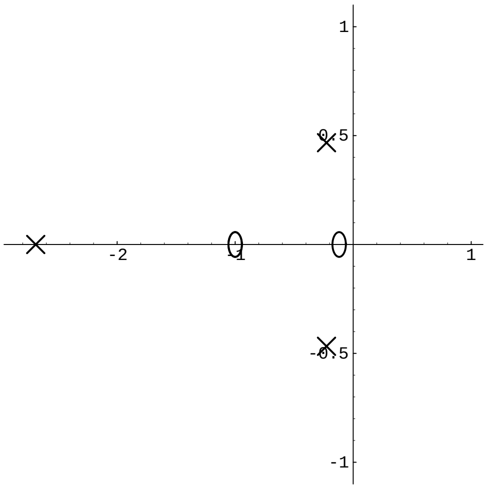


Figure 4.7: The zeros and poles of the compensator $C = (-0.0310 - 0.3235s - 0.5537s^2 - 0.2613s^3)/(0.8808 + s)$ that corresponds to T_0 are indicated by "o" and "x" respectively. Note that this compensator is stable. Stability of the compensator is not guaranteed a priori by the design method of this book.

performance to be well defined. If W does not 2 is violated at $\omega = \infty$.

2. $\Gamma(\omega, T(j\omega)) = W(j\omega)|1 - T(j\omega)|$. This is not a good performance, no matter what W is. The reason is that since T rolls off at high frequency, the performance at very high frequency is determined by W only. It has no way measuring how fast T rolls off or any relevant high frequency property of T.

3. $\Gamma(\omega, T(j\omega)) = W_1(j\omega)|T(j\omega)| + W_2(j\omega)|1 - T(j\omega)|$. The weight W_2 should be 0 at infinite frequency, so that the term with this weight does not influence performance at very high frequency. The comment in the first example also applies to the weight W_1.

4. $\Gamma(\omega, T(j\omega)) = W_1(j\omega)|T(j\omega)|^2 + W_2(j\omega)|1 - T(j\omega)|^2$. This continues example 3. The relative degree of W_2 should be $-p/2$ or larger, where p is the relative degree of the plant[3].

4.5 Performance not of circular type

In this section we discuss performance functions more general than (4.8). To motivate the discussion suppose that we have a design problem where one of the constraints is

$$W_1(\omega)|T(j\omega)| + W_2(\omega)|1 - T(j\omega)| < 1, \quad \omega_1 \le \omega \le \omega_2 \tag{4.17}$$

where W_1 and W_2 are known weight functions[4]. Clearly this constraint is not a disk constraint. Indeed, the region

$$S_\omega(c) = \{z : W_1(\omega)|z| + W_2(\omega)|1 - z| \le c\}$$

is not a disk. However, the inequality (4.17) can be used to define a performance function, valid at least for those ω satisfying $\omega_1 \le \omega \le \omega_2$. For this we set, for z any complex number,

$$\Gamma(\omega, z) = W_1(\omega)|z| + W_2(\omega)|1 - z|$$

Thus a designable transfer function T satisfies the constraint at frequency ω if and only if

$$\Gamma(\omega, T(j\omega)) \le 1 \tag{4.18}$$

---

[3] When zeros or poles of the plant occur on the $j\omega$ axis, weight functions have to be chosen so that the weights in the terms $W_1(j\omega)|T(j\omega)|^2$ and $W_2(j\omega)|1 - T(j\omega)|^2$ have the zeros (resp. poles) at the same values of ω than those of the plant, order included. The reason for this is the internal stability requirement. See Chapter 8

[4] This may arise when considering competing constraints, see Section 6.5 in Chapter 6.

The function Γ is called a *performance function*. Let us define for $T \in \mathcal{RH}^\infty$ the number

$$\gamma(T) := \sup_\omega \Gamma(\omega, T(j\omega)) \tag{4.19}$$

The number $\gamma(T)$ represents the cost, or overall performance, associated with the designable transfer function T. It is called the *performance index*. Note that (4.18) can be written as

$$\gamma(T) \leq 1 \tag{4.20}$$

Observe that $\gamma(T) < 1$ means that T satisfies the performance requirements, with some slack. Thus we associate small values of $\gamma(T)$ with *good performance*.

The simplest functions Γ arise when a single disk inequality applies at each frequency;

4.6 Optimization

It is a fact that there is a formula that depends on the RHP poles and zeros of the given plant P which gives all $T \in \mathcal{I}$. Indeed we have that there exist functions $A,\ B \in RH^\infty$ such that

$$T \in \mathcal{I} \iff T = A + BT_1, \text{ for some } T_1 \in RH^\infty \tag{4.21}$$

For general P such formula is proved in Chapter 8, while for RHP-stable P we already saw in Chapter 2 that

$$T \in \mathcal{I} \iff T = PT_1, \text{ for some } T_1 \in RH^\infty \tag{4.22}$$

One approach to treating the design problem in practice is to use formulas (4.21) and (4.22) to account for internal stability. Depending on the specific approach and the tools available, to solve *Design* the formulas are either manipulated directly by the designer, or they are automatically handled by the computer. In this case only the list of unstable zeros and poles of the plant is necessary for the software to build and use the necessary formulas.

4.6.1 The optimization problem $OPT_{\mathcal{I}}$

Suppose that steps I and II have been completed, that is, there is available a performance function Γ and a set $\mathcal{I}$ of admissible functions. To find the "best" T possible, one must find the smallest possible value for the performance index $\gamma(T)$ for all $T \in \mathcal{I}$. That is, one must solve the following problem:

$OPT_{\mathcal{I}}$

Given $\Gamma(\omega, z)$ a positive valued function of $\omega \in R$ and of $z \in C$, and a set $\mathcal{I}$ of admissible functions, find

$$\gamma^* := \inf_{T \in \mathcal{I}} \sup_{\omega} \Gamma(\omega, T(j\omega)) \qquad (4.23)$$

and find an optimal T^* if it exists.

In general $OPT_{\mathcal{I}}$ is a constrained problem, since one may have $\mathcal{I} \neq RH^{\infty}$. By solving $OPT_{\mathcal{I}}$ we solve *Design*. This follows from Proposition 3 below.

Proposition 3 *If $\gamma^* < 1$ then there exist solutions to Design, and if $\gamma^* > 1$ then there are no solutions to Design.*

Proof. By definition, the number γ^* is the smallest value possible for γ so that the inequality

$$\Gamma(\omega, T(j\omega)) \leq \gamma \qquad (4.24)$$

has solutions $T \in \mathcal{I}$. Therefore $\gamma^* > 1$ says that there are no functions Γ that satisfy (4.20) i.e., that there are no solutions to *Design*. Similarly, $\gamma^* < 1$ says that there is at least one function $T \in \mathcal{I}$ that satisfies (4.20), which implies that there exist solutions to *Design*. ■

4.7 Internal stability and optimization

It is a very simple matter to transform $OPT_{\mathcal{I}}$ into an unconstrained optimization problem OPT. This is desirable because powerful mathematical algorithms for solving unconstrained problems are available. It is so easy that even the computer can do this in a manner that is transparent to the user, for example, as implemented in the package OPTDesign. Ocasionally the designer may want to do it directly by hand (or be forced to do it). Thus our discussion of this below.

4.7.1 The optimization problem OPT

To do the advertised transformation of $OPT_{\mathcal{I}}$ into OPT we use the formula (4.21) in combination with the performance $\Gamma(\omega, z)$. Suppose T is as in (4.21). Define a new performance function as follows:

$$\Gamma_1(\omega, z_1) = \Gamma(\omega, A(j\omega) + B(j\omega)\, z_1\,) = \Gamma(\omega, z) \qquad (4.25)$$

Therefore $OPT_{\mathcal{I}}$ is equivalent to the following unconstrained optimization problem:

Given $\Gamma_1(\omega, z)$ a positive valued function of $\omega \in R$ and of $z \in C$, find

OPT

$$\gamma_1^* := \inf_{T_1 \in \mathcal{I}} \sup_{\omega} \Gamma_1(\omega, T_1(j\omega)) \tag{4.26}$$

and find an optimal T_1^* if it exists.

4.7.2 *OPT* with circular Γ

Note that if the function Γ is circular, then the function Γ_1 in (4.25) is also circular. To see this, combine the disk inequality (??) with (??) to obtain the inequality

$$\frac{1}{R(j\omega)^2}|K(j\omega) - A(j\omega) - B(j\omega)T_1(j\omega)|^2 \leq 1 \quad (\forall \omega) \tag{4.27}$$

The resulting inequality is itself a disk inequality in the variable T_1:

$$\frac{|B(j\omega)|^2}{R(j\omega)^2}|\frac{K(j\omega) - A(j\omega)}{B(j\omega)} - T_1(j\omega)|^2 \leq 1 \quad (\forall \omega) \tag{4.28}$$

Thus we can choose $\Gamma_1(\omega, T_1(j\omega))$ to be the left-hand-side of inequality (4.27) to arrive at the unconstrained optimization problem *OPT*.

4.8 Exercises

4.9 References and further reading

Chapter 5

A Design Example with OPTDesign

5.1 Introduction

This chapter presents a computer session and introduces the reader to practical issues. This lets the student see how the ideas he has learned feed into a computer implementation. We use our own package OPTDesign, so this chapter can be utilized as an introduction to it.

5.1.1 The package OPTDesign

OPTDesign and supporting software are written in Mathematica, and they were produced at the Laboratory for Mathematics and Statistics at the University of California San Diego, La Jolla, California, from 1990 - 1994. By J. W. Helton, O. Merino, J. Myers and T. Walker. The development of OPTDesign and its companion the optimization program Anopt has been supported by the Air Force Office for Scientific Research and by the National Science Foundation.

5.1.2 How to use this notebook

This chapter actually is something called a Mathematica Notebook, which has been automatically converted to the standard scientific typesetting language called TeX in order to include it in this book. Notebooks have the advantage that they integrate text, computer input and computer output. The latter includes excellent graphics. The user may modify it and re-execute commands easily, and the new output replaces the old one automatically.

You can use this notebook in two ways:

1. As an illustrated guide to the use of OPTDesign package. A design example is treated here in depth to illustrate the basic commands and the interpretation of results.

2. As a template to treat many problems. Once you are using a Notebook it is easy to modify equations and parameters.

The notebook has been organized so that there are very few places where the user specifies the input. This has the tremendous advantage of making the set up of computer runs a very simple task.

The name OPTDesign denotes a collection of programs (or "package") for doing control design. OPTDesign is also the name of the main optimization program in the package. To avoid confusion we will denote the program as OPTDesign[].

5.1.3 Starting a session

Begin by loading OPTDesign into a Mathematica session. Computer input and output below is in "typewriter type" (darker) style. The input has regular size (this is what you can modify), while the output appears in smaller type. The rest of the text are comments.

```
<<OPTDesign`
```

```
 Anopt 3.0 (c) Copyright 1991-94
 J.W.Helton and O.Merino.  All rights reserved.
OPTDesign 1.0 (c) Copyright 1991-94
    J.W. Helton and O.Merino.  All rights reserved.
```

5.2 The problem

The design problem is, given a plant P and a set of performance requirement, find a system (S) that is internally stable and that satisfies the performance requirements.

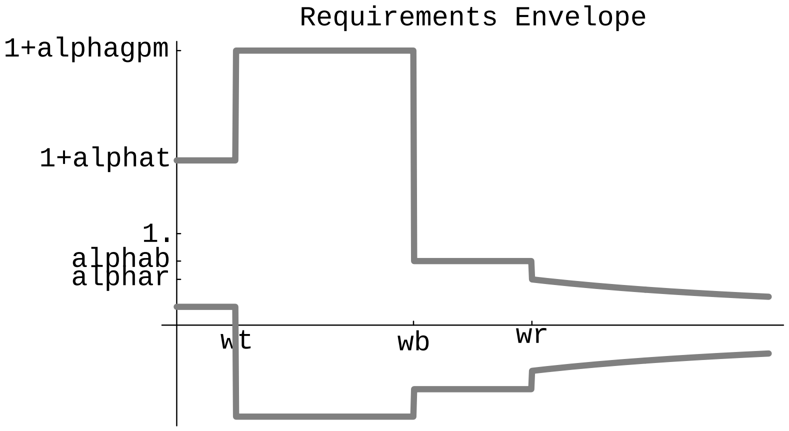
We now give the data for a specific problem. The plant is

$$P(s) = \frac{(1 - s/5)}{(1 + s/2)^2}$$

and the requirements are, tracking on the band $0 \leq \omega \leq 0.3$ with bound 1.5, gain-phase margin constant of 0.5, bandwidth $0 < \omega \leq 2.$, and closed loop roll off with bound 2.5 on the band $4.0 \leq \omega$.

A list of the requirements in this problem in the form of inequalities is the following:

Tracking	$\lvert T(j\omega)-1\rvert$	$<$	α_t	$0<\omega<\omega_t$
Gain-Phase	$\lvert T(j\omega)-1\rvert$	$<$	α_{gpm}	$\omega_t<\omega<\omega_b$
Bandwidth	$\lvert T(j\omega)\rvert$	$<$	α_b	$\omega_b<\omega<\omega_r$
Roll-Off	$\lvert T(j\omega)\rvert$	$<$	$\alpha_r\lvert P(j\omega)\rvert$	$\omega_r<\omega$



Thus we see that in our problem, $\omega_t = 0.3$, $\alpha_t = 1.5$, $\omega_b = 2.0$, $\alpha_b = 0.7$, $\omega_r = 4.0$, $\alpha_r = 2.5$.

5.2.1 Specifying the problem.

The plant and a set of design requirements are specified now. *Frequently this will be the only place where you enter data.* The plant should be entered as a rational function:

```
p[s_]= ( 1 - s/5 )/( 1 + s/2 )^2;
```

We set the values of important constants below:

```
      wt  = 0.3;
  alphat  = 1.5;

 alphagpm = 2.0;

       wb = 2.0;
```

```
alphab = 0.7;

    wr = 4.0;
alphar = 2.5;
```

5.2.2 The requirements envelope

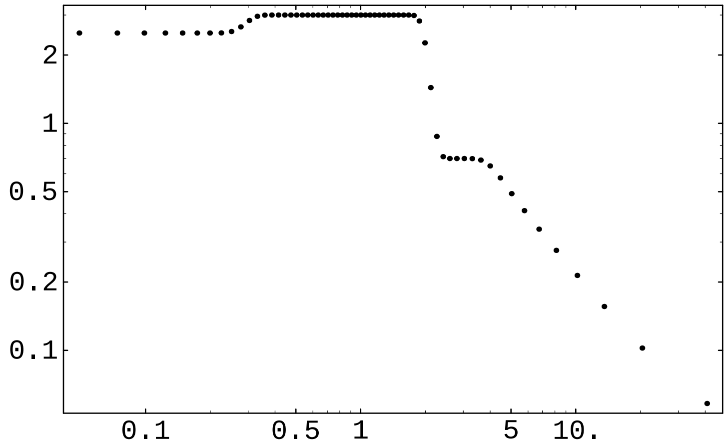
Now we define a center and radius functions, which determine the envelope.

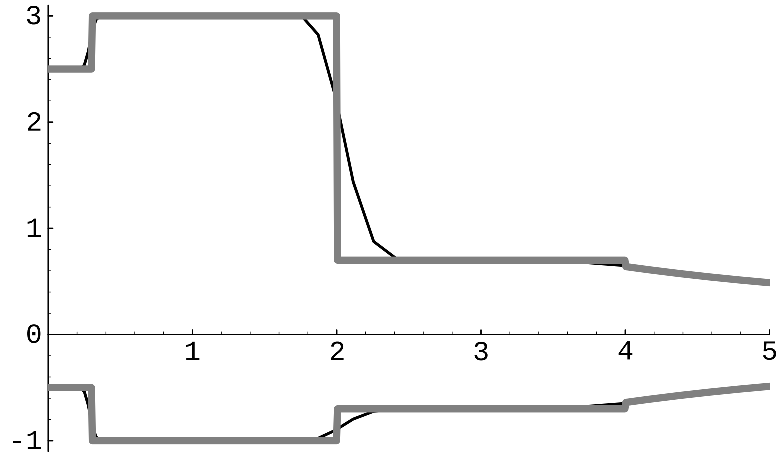
```
k0[w_]= Which[ Abs[w] < wb, 1.0,
                     True, 0.0];

r0[w_] = Which[Abs[w] < wt,  alphat,
               Abs[w] < wb,  alphagpm,
               Abs[w] < wr,  alphab,
                     True ,  alphar*Abs[p[I w]]
                ];
```

The requirements envelope has jump discontinuities at two locations on the $\omega > 0$ semiaxis. For better results, smoothing of functions is convenient in this case. OPTDesign does a very small amount of smoothing automatically; we will rely on it for now.

The function EnvelopePlot[ ] produces two plots. The first is a plot of the discrete upper limit of the requirements envelope. The second figure has a plot of the original envelope and of the envelope resulting after smoothing.

```
EnvelopePlot[Radius->r0,Center->k0,FrequencyBand->{0.,wr+1.}]
```


Discrete Profile of Smooth Requirements Envelope


Original and modified Envelopes

```
-Graphics-
```

The plot of the upper limit of the profile of the requirements envelope can be used to judge if there are enough frequency gridpoints in the band of interest. This is of course relative to the precision you want to attain (to get a feel for this several examples must be run) . In our example we have that most of the gridpoints are in the band $0.1 < \omega < 10.0$, which is where the center and radius functions have their distinctive features. This is good.

The plot of the requirements envelopes before and after smoothing gives an idea of how much distortion we are introducing into the problem by doing smoothing of functions. We must keep in mind that the solution obtained with OPTDesign will be optimal with respect to the "modified" envelope.

If we do not like either one of the two plots, then several parameters can be modified to obtain more satisfactory plots. We leave this for a later subsection, and proceed now with a run of the program OPTDesign[] with the given data.

5.3 OPTDesign iteration

The program OPTDesign[] is now run with a minimal set of explicit input. Other input that can be specified to modify default values will be discussed later.

Running OPTDesign[] produces information on the progress of the run that is printed on the screen.

```
OPTDesign[p,Radius->r0,Center->k0];
```

```
Parameterization for internal stability:
  T = A + B * T1, where T1 is RHP stable and
      -5 + s
A[s]=---------
             2
     (2. + s)
      -5 + s
B[s]=---------
             2
     (2. + s)


Processing performance function...
Sampling Radius and Center ...
Smoothing Radius and Center ...
Optimization routine Anopt now number-crunching ...

It :    Current Value     :  Step   : Optimality Tests  :  Error  : Sm. : Grid
   :      SupGamma        :         :   OT1      OT2    :   NED   :     :
------------------------------------------------------------------------------
 0 : 2.2485924730083E+00 :   N/A   : 8.9E-01 : 1. E+00 :   N/A   : NON : 128
 1 : 1.5969785364195E-01 : 1.8E+00 : 3.6E-03 : 0  E+00 : 1.5E-02 : NON : 128


Summary
-------
Supremum of gamma: SupGamma = 1.596978536419496E-01
Optimality Test  : OT1      = 3.5999393585E-03
Error diagnostic : NED      = 1.54E-02
```