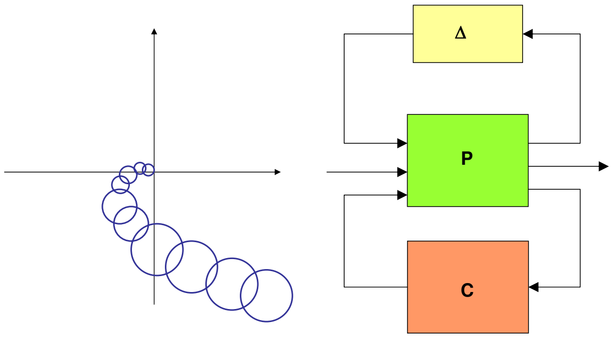
MASTER OF SCIENCE IN ENGINEERING

Advanced Control

An Overview on Robust Control



Scope: allow the student to assess the potential of different methods in robust control without entering deep into theory. Sensitize for the necessity of robust feedback control.

Keywords: uncertainty representations, H_∞, μ synthesis, LMI

Prerequisites: Nyquist criterion, gain and phase margin, LQG state space control

Contact: Raoul Herzog[1], Jürg Keller[2]

Version: 1.0

Date: May 6, 2009

[1]raoul.herzog@heig–vd.ch
[2]juerg.keller1@fhnw.ch

Contents

1 Introduction to Robust Control

1.1 Motivation of Robust Control

System modeling is generally one of the most important tasks in engineering, and in particular in control engineering. Modeling is always a delicate task, since the physical reality is often complicated, and all attempt to mathematically describe a real physical process involves simplifications and assumptions, e.g.:

- dynamics are often consciously neglected[1] to make the model tractable, e.g. unmodeled sensor and actuator dynamics, higher order modes from large scale structures modeled by finite elements (FEM),

[1] It's common to say: "My model is precise up to ... Hz."

- nonlinearities are often either hard to model or too complicated,
- parameters are often not exactly known, either because they are hard to measure precisely, or because of varying manufacturing conditions[2].

Furthermore, it is desirable to end up with *simple* plant models. Indeed, in modern control, the controller is the output of an optimization problem, and the complexity of the controller is directly linked with the complexity of the plant model: a complicated high–order plant will automatically lead to a complicated high–order controller, which is undesirable.

Robust control deals with system *analysis* and control *design* for such *imperfectly known* process models. One of the main goals of *feedback control* is to maintain overall stability and system performance despite uncertainties in the plant.

In general, robustness does not "come for free" from a controller designed via optimal control and estimation theory (observer design): a controller designed for a nominal process model generally works fine for the nominal plant model, but may fail[3] for even a "nearby" plant model.

An important point of all feedback control synthesis methods is the control engineer's awareness of *inherent* trade–offs : *increasing* the robustness will generally make the controller "less aggressive", and will thereby *decrease* system performance. Robust control allows to specify more or less directly the plant uncertainty, and allows to predict the possible trade–offs between robustness and closed–loop performance.

Sometimes bachelor–level courses in control may give the impression that everything is feasible with control, and that it's only a matter of finding a "good" controller. But this impression is completely false: the plant itself implies *inherent* limitations[4], especially if the plant is unstable. The achievable performance/robustness trade–offs strictly depend on the plant, and cannot be overcome by any sophisticated control [1].

1.2 An attempt to define robust control

A definition of robust control could be stated as:

Robust control aims at designing a fixed (non–adaptive) controller such that some defined level of performance[5] *of the controlled system is guaranteed, irrespective of changes in plant dynamics within a predefined class.*

[2]For serial production of devices incorporating feedback control, individual tuning is not desirable. The robustness of the system should be sufficient to tolerate all uncertainties and tolerances from the manufacturing processes. For mechatronical systems, the manufacturing uncertainties are coming from the mechanical and electronical parts. Generally, there are no manufacturing tolerances inside the digital controller because software is perfectly reproducable.

[3]Instability or unacceptable performance degradations may occur.

[4]e.g. by the Bode integral relation.

[5]e.g. closed–loop stability, reference tracking performance, and disturbance rejection performance.

Robust control offers a collection of powerful mathematical tools and efficient software algorithms able to partly[6] answer the following key questions:

Describing and characterizing plant uncertainty: What is a good way to describe plant variations or plant uncertainty? Some descriptions attempt to faithfully describe the real situation (e.g. probability distributions on physical parameters), but unfortunately there may exist no efficient method to solve the problem. Other uncertainty descriptions are less direct, but more convenient for the theory (e.g. analytic solutions exist).

Robustness analysis problems: As an example consider a nominal plant and a given stabilizing controller. The uncertainty in the plant model is defined by a class of perturbations. There is no "true" model, we have to deal with a given set of possible plant models. Now, we would like to know whether or not the closed–loop remains stable for the whole class of plants. This is a typical robustness *analysis* problem.

Robustness synthesis problems: Find a controller which stabilizes a given class of plants. Generally, synthesis problems are more difficult to solve than analysis problems.

1.3 Structure of this document

The document is structured as follows: In section 2, norms for signals and systems are reviewed for the single input single output case (SISO). Section 3 recapitulates the Nyquist criterion and the small gain theorem, an important working horse in robust control. Section 4 addresses the question how to describe model uncertainty. Sections 5 introduces the setup of $\mathcal{H}_\infty$ control. It is not the goal of this document to describe the underlying mathematical theory, which is quite demanding. Therefore, section 6 only sketches the $\mathcal{H}_\infty$ solution from a user point of view. Section 7 discusses some limitations and drawbacks of standard $\mathcal{H}_\infty$ methods. Finally, section 8 gives an outlook to the actual state–of–the–art in robust control. Section 9 concludes with some general remarks on robust control.

2 Review of Norms for Signals and Systems

Norms are used in many places of engineering to quantify the magnitude of an object (e.g. the amplitude of a signal), or to quantify the proximity of two objects (e.g. the proximity of two systems). In robust control, norms play a crucial role:

- the choice of metric used to quantify the amount of process uncertainty,

[6] It should be noted that some simple problems in robust control (e.g. exact stability determination of a linear system in which several parameters vary over given ranges) have shown to be NP–hard, hence as difficult as other famous problems for which no efficient solutions are known to exist, or likely to be found.

- the choice of norm used in the optimization problem associated with the controller synthesis. The linear quadratic gaussian LQG control is based on the optimization of a $||\cdot||_2$ norm, whereas $\mathcal{H}_\infty$ control is based on the optimization of a $||\cdot||_\infty$ norm.

The $\mathcal{L}_2$ (Euclidean) norm of a time–domain scalar signal $u(t)$ is defined as:

$$||u||_2 = \sqrt{\int_{-\infty}^{\infty} u^2(t)\, dt}. \tag{1}$$

If this integral is finite, then the signal u is *square integrable*, denoted as $u \in \mathcal{L}_2$. For vector–valued signals $\mathbf{u}(t)$,

$$\mathbf{u}(t) = \begin{bmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_n(t) \end{bmatrix}. \tag{2}$$

the 2–norm is defined as

$$||\mathbf{u}||_2 = \sqrt{\int_{-\infty}^{\infty} ||\mathbf{u}(t)||_2^2\, dt} = \sqrt{\int_{-\infty}^{\infty} \mathbf{u}^T(t)\, \mathbf{u}(t)\, dt}, \tag{3}$$

where the superscript ^{T} denotes transposition.
A Linear Time–Invariant (LTI) system G can be described either by a state space realization

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned}$$

or by its corresponding transfer function

$$G(s) = C(sI - A)^{-1}B + D. \tag{4}$$

Similar to signals, we would like to define different norms for systems, respectively for transfer functions. Two systems G_1 and G_2 are "close" if the norm of the difference of their transfer functions $||G_1 - G_2||$ is "small". Two systems might be "close" with respect to one defined norm, but "far" with respect to another norm[7].

The $\mathcal{H}_2$ norm of a *stable* system $G(s)$ is defined as the $\mathcal{L}_2$ norm of its impulse response $g(t)$. When dealing with stochastic signals, the $\mathcal{H}_2$ norm of a system corresponds to the variance $E[||y||_2]$ of the system response $y(t)$ when the system is excited by a unit intensity *white* noise input $u(t)$.

[7] With the metric induced by the $\mathcal{H}_\infty$ norm, the two systems $P_1(s) = \frac{1}{s+\epsilon}$ and $P_2(s) = \frac{1}{s}$ are *infinitely* "far", no matter how small ϵ is. This may seem illogical because a reasonable controller will stabilize both plants P_1 and P_2 simultaneously, and the resulting closed–loop systems will be nearly identical. A remedy consists in using the so–called Vinnicombe metric, which also allows to treat marginally stable and unstable systems.

The $\mathcal{H}_\infty$ norm of a *stable* transfer function $G(s)$ is defined as the maximum RMS amplification over *arbitrary* square integrable input signals $u \neq 0$.

$$||G||_\infty = \sup_{u \in \mathcal{L}_2} \frac{||y||_2}{||u||_2} \tag{5}$$

Note that the $\mathcal{H}_\infty$ norm for a *system* is *induced* by the $\mathcal{L}_2$ norm for *signals*. The physical interpretation of the $\mathcal{H}_\infty$ norm corresponds simply to the maximum *energy amplification* over all input signals. It can be shown that for single–input single–output systems $||G||_\infty$ equals the peak magnitude in the Bode diagram of the transfer function $G(j\,\omega)$:

$$||G||_\infty = \sup_\omega |G(j\,\omega)| \tag{6}$$

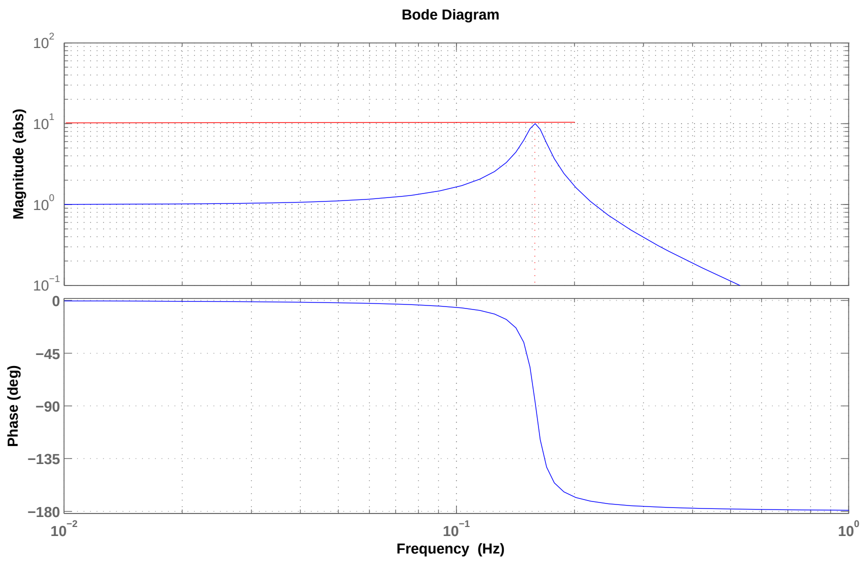


Figure 1: Example: The $||\cdot||_\infty$ norm of $G(s) = \frac{1}{s^2+0.1s+1}$ is $||G||_\infty \approx 10$.

In $\mathcal{H}_\infty$ control the performance to be optimized is defined in terms of *minimizing* the $\mathcal{H}_\infty$ norm of closed–loop transfer functions, e.g. the sensitivity function $S(s)$ and the complementary sensitivity function $T(s)$. This type of optimization problem is also called *min–max* problem: $\mathcal{H}_\infty$ control seeks to *minimize* the *worst–case* scenario, i.e. when the closed–loop function has its peak. In contrast, classical LQG control minimizes the closed–loop behaviour for *known* input signals[8], whereas $\mathcal{H}_\infty$ control works with *unknown* input signals and tries to optimize the *worst–case* scenario. Therefore, the solution of $\mathcal{H}_\infty$ control problems typically inhibits "flat[9]" Bode

[8] e.g. the energy of the resulting closed–loop impulse response
[9] allpass behaviour

magnitude plots (no more peak). When minimizing the absolute peak in the Bode magnitude diagram, automatically other local peaks pop up. This effect is called "waterbed–effect". Finally, the value of all local peaks join the value of the absolute peak, and the response becomes "flat". The theoretical background of this comes from the Bode integral theorem, see equation (12) and figure 6 page 12.

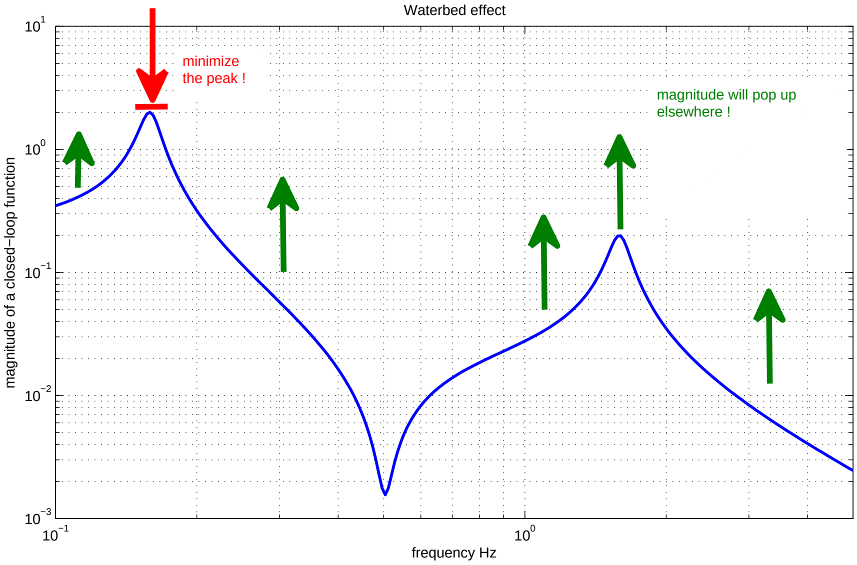


Figure 2: Illustrating the waterbed effect.

Instead of "flat" Bode magnitude plots we can also impose a given shape. This corresponds to a modern version of classical loop–shaping. In this context, important design parameters are frequency dependent *weighting* functions $W(s)$. They allow to shape given closed–loop functions, e.g. the sensitivity $S(s)$, by optimizing their *weighted* norm $||W\,S||_\infty$.

The $\mathcal{H}_\infty$ norm of a multivariable (MIMO) transfer matrix $\mathbf{G}(s)$ needs the important concept of *singular values* which is beyond the scope of this document.

3 The Nyquist Criterion and the Small Gain Theorem

The Nyquist criterion is a cornerstone of classical control, and is of fundamental importance in robust control. The following chapter recapitulates the Nyquist criterion, the small gain theorem, and shows its application to robust control.

3.1 Review of the Nyquist Criterion and Classical Stability Margins

The Nyquist criterion allows to check closed–loop stability based on the inspection of the loop gain $L(s) = C(s) \cdot P(s)$, without computing the closed–loop poles, i.e.

the roots of $1 + L(s) = 0$. The Nyquist criterion is based on Cauchy's argument, and says:

The closed–loop system with loop gain $L(s)$ and a negative feedback polarity is stable if and only if the Nyquist plot of $L(j\omega)$ encircles N_P anticlockwise times the critical point $s_{\text{crit}} = -1$ in the complex plane, where N_P is the number of unstable (right halfplane) poles of $L(s)$.

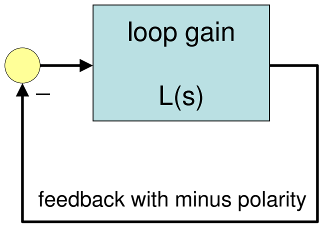


Figure 3: Elementary feedback system with loop gain $L(s)$

As a special case, if $L(s)$ already is a *stable* open loop transfer function, N_P is zero, and the Nyquist plot of $L(j\omega)$ must *not* encircle the critical point -1 in order to ensure closed–loop stability. Figure 4 recapitulates the definition of the classical gain margin A_m and phase margin ϕ_m. For example a gain margin of $A_m = 2$ means that the closed-loop stays stable even if the loop gain doubles. The phase margin ϕ_m indicates the amount of additional delay the feedback loop can tolerate before becoming unstable. Robust control does not work with the classical stability margins A_m and ϕ_m for two reasons: 1) there exist *no analytical* optimization techniques for these margins, and 2) there are cases where the classical margins indicate a good robustness against *individual* gain and phase tolerances, whereas the feedback loop is not at all robust against *simultaneous* variations of gain and phase, see exercice 2, page 31. For these reasons, robust control prefers as margin the *critical distance* d_{crit} between the Nyquist plot of $L(j\omega)$ and the critical point $s_{\text{crit}} = -1$:

$$d_{\text{crit}} = \min_{\omega} \left(\text{dist}(L(j\omega), s_{\text{crit}})\right) = \min_{\omega} ||L(j\omega) - s_{\text{crit}}||_2 = \min_{\omega} ||1 + L(j\omega)||_2. \quad (7)$$

It follows that the critical distance d_{crit} is the reciprocal of the sensitivity function peak:

$$d_{\text{crit}} = \frac{1}{||S||_\infty}, \quad \text{where} \quad S(j\omega) = \frac{1}{1 + L(j\omega)}. \quad (8)$$

Maximizing the critical distance d_{crit} corresponds to minimizing the $|| \cdot ||_\infty$ norm of the sensitivity function. Therefore, one of the natural objectives[10] of robust control consists of minimizing the $|| \cdot ||_\infty$ norm of the sensitivity function.

[10] It is important to see that sensitivity peak minimization is *not the only* objective in real world problems. In fact, minimizing *only* the $|| \cdot ||_\infty$ norm of the sensitivity function turns out to be an *ill-posed* problem, because the gain of the resulting controller would be infinite.

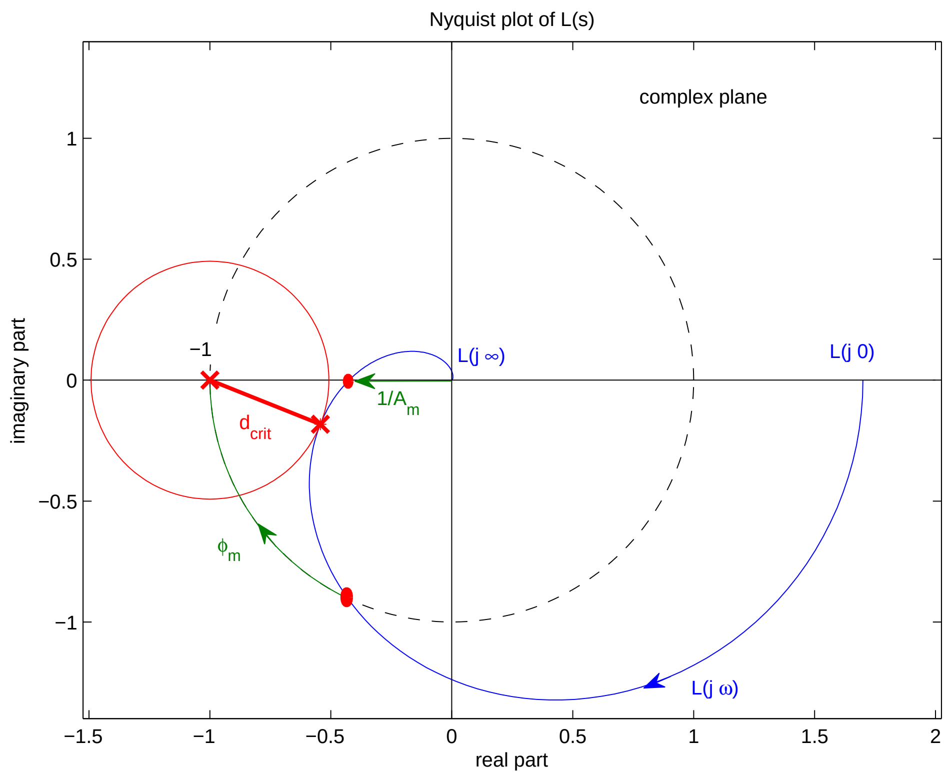


Figure 4: Definition of the critical distance d_{crit} and the classical stability margins

3.2 The Small Gain Theorem

The small gain theorem follows directly from the Nyquist criterion: if $L(s)$ is stable and $||L||_\infty < 1$, then the loop gain $|L(j\omega)|$ is smaller than 1 for all frequencies ω, and hence $L(j\omega)$ does not encircle the critical point -1. It follows that the small gain condition $||L||_\infty < 1$ is a *sufficient* but *not necessary* condition for closed–loop stability. The small gain condition corresponds to an *infinite* phase margin $\phi_m = \infty$. If $||L||_\infty < 1$, the closed loop stays stable even if the feedback polarity is wrong!

The small gain theorem is not limited to linear feedback control: it can even be generalized to *nonlinear* feedback control. In this case, L becomes a nonlinear operator in time domain, and the $||\cdot||_\infty$ norm condition on $L(s)$ must be replaced by an operator norm condition. This may sound abstract, but a simple application is the feedback of a *linear* dynamic system in cascade with a *static nonlinearity*. In this case, the small gain theorem yields a *sector bounded*[11] nonlinearity as a sufficient condition for closed–loop stability.

It makes no sense to apply the small gain theorem directly for controller synthesis: for good tracking performance a *high* loop gain is needed within the control

[11] also called Popov criterion.

bandwidth[12]. However, the small gain theorem has a great utility for analyzing feedback loops with *unstructured* uncertainty.

3.3 Applications of the Small Gain Theorem to Robust Control

Suppose that $P_0(s)$ denotes the nominal plant, and $C(s)$ a stabilizing controller. Now consider an additive perturbation which yields a "cloud" of plants $P(s) = P_0(s)+\Delta_a(s)$, where $\Delta_a(s)$ denotes an *unknown* stable transfer function representing the modeling uncertainty of P_0. The question arises "how much" uncertainty the closed loop may tolerate before becoming unstable. Figure 5 shows that the feedback

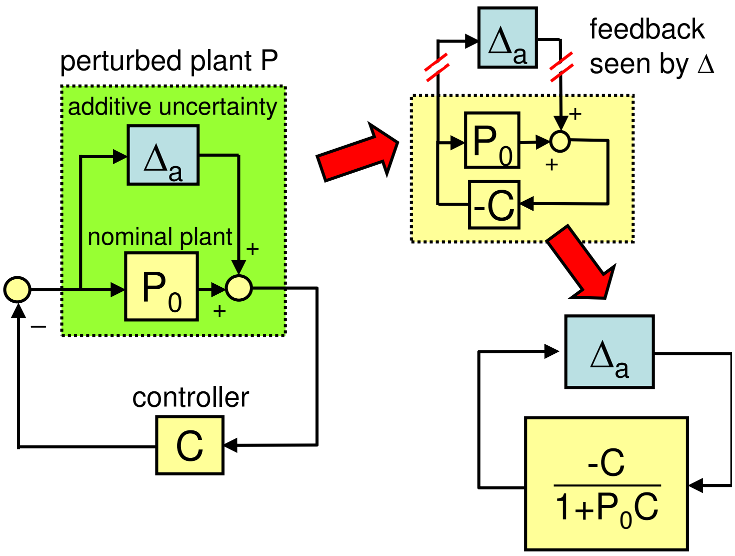


Figure 5: Application of the Small Gain Theorem to additive plant uncertainty.

seen by Δ_a is $\frac{-C}{1+P_0C}$. The corresponding loop gain is $\Delta_a \cdot \frac{-C}{1+P_0C}$. Since both Δ_a and the nominal closed–loop are stable we can apply the small gain theorem for the feedback loop seen by Δ_a. A sufficient condition for the stability of the perturbed system is therefore:

$$\left|\left|\Delta_a \cdot \frac{-C}{1+P_0C}\right|\right|_\infty < 1. \tag{9}$$

Using the submultiplicative property of norms $||AB|| \leq ||A|| \cdot ||B||$, the sufficient stability condition becomes:

$$||\Delta_a||_\infty \cdot \left|\left|\frac{C}{1+P_0C}\right|\right|_\infty < 1 \quad \Longrightarrow \quad ||\Delta_a||_\infty < \frac{1}{\left|\left|\frac{C}{1+P_0C}\right|\right|_\infty}. \tag{10}$$

[12] If the controller includes an integral action (e.g. the I part of PID), the loop gain is infinite at 0 Hz, and hence $||L||_\infty = \infty$.

Equation (10) is a *conservative bound* indicating the amount of tolerated additive uncertainty which preserves stability.

In case of multiplicative unstructured uncertainty, the set of perturbed plants is described as $P(s) = P_0(s) \cdot (1 + \Delta_m(s))$. Using the small gain theorem, the following stability bound can be found:

$$||\Delta_m||_\infty \; < \; \frac{1}{\left|\left|\frac{1}{1+P_0C}\right|\right|_\infty} \; = \; \frac{1}{||S||_\infty}. \tag{11}$$

A *high* sensitivity peak value $||S||_\infty$ leads to a *small* tolerable multiplicative perturbation Δ_m, which is intuitively clear since the distance d_{crit} of the nominal loop gain $L_0 = P_0 \cdot C$ to the critical point -1 is small.

We will now discuss an extremely fundamental relationship, called Bode integral theorem:

$$\int_0^\infty \ln\, |S(j\omega)|\, d\omega \; = \sum_{\text{unstable poles } p \text{ of } L(s)} \text{Re}(p). \tag{12}$$

This law can be seen as a *conservation law*: the integrated value of the log of the magnitude of sensitivity function $S(j\omega)$ is *conserved* under the action of feedback. If the open–loop $L(s)$ is stable, then the integral becomes zero. At low frequencies, in order to have good tracking performance, the sensitivity must be much smaller than 1 (negative dB levels), i.e. $\ln|S(j\omega)|$ must become negative. The Bode integral theorem states that the average sensitivity *improvement* at low frequencies is *compensated* by the average sensitivity *deterioration* at high frequencies. This is illustrated by figure 6. If the plant is unstable, the situation is becoming worse since the right hand side of equation (12) is now positive. This means that the average sensitivity deterioration is always larger than the improvement. The more unstable the plant is, the more positive the real part of the unstable poles, the more difficult the situation becomes. This applies to *every* controller, *no matter how it was designed*. Unstable plants are *inherently* more difficult to control than stable plants [1].

4 Description of Model Uncertainty

It is worthwhile to clarify what is meant by model uncertainty. In a control system there are two categories of uncertainties: disturbance signals and perturbations in the plant dynamics. Disturbances are external stochastic inputs which are not under control. Examples of disturbance signals are sensor and actuator noise or changes of the environment. Dynamic perturbations are model uncertainties caused by not exactly known or slowly changing plant parameters, and unmodeled or approximated system dynamics. In a house temperature control, disturbances and dynamic plant perturbations would be :

- **Disturbances:**
  changing outdoor temperature, wind speed, open windows and doors, heating due to electrical equipment or human bodies, sun radiation

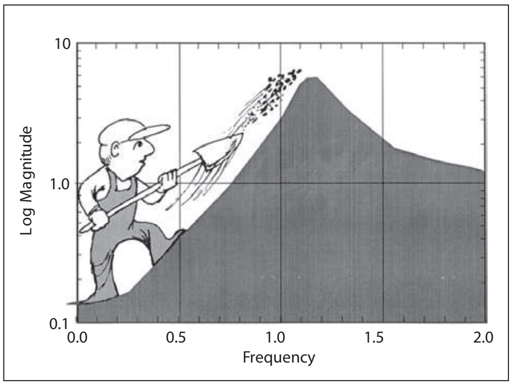


Figure 6: Loopshaping constraints: Sensitivity reduction at low frequencies *inevitably* leads to sensitivity increase at higher frequencies. Picture taken from [1].

- **Dynamic plant perturbations:**
  not exactly known isolation coefficients, unknown and changing heat capacities, uncertain efficiency of the heating device

The example shows that it is not always clear how to classify the disturbances. Therefore, it is not surprising that in μ–synthesis the two categories of disturbances are handled within the same formalism. For H_∞–controller design plant uncertainty results from approximating the real system with a mathematical model of tractable complexity. Additionally its physical parameters are not exactly known.

As explained in the motivation section, the aim of H_∞–based controller design is to include uncertainty into controller design. To achieve this it is necessary to have a mathematical description of model uncertainty. In this section, two different types of uncertainty descriptions will be introduced. The first is *unstructured* uncertainty, whereas the second is *structured* uncertainty.

4.1 Unstructured Uncertainty

A description of model uncertainty has to meet the following goals: it should be simpler than a physical model of the neglected system dynamics, and it should be tractable with a formalism which can easily be used in controller design. A simple solution is found, if all dynamics, time–invariant perturbations that may occur in different parts of the system are represented with a single perturbation transfer function $\Delta(s)$. There are many possibilities to include $\Delta(s)$ into a control system, but only the two most commonly used will be presented in the following. These are shown in figures 7 and 8. The focus is on SISO systems. Figure 7 shows $\Delta_a(s)$ as an additive perturbation of the plant transfer function $P(s)$:

$$P(s) = P_o(s) + \Delta_a(s), \tag{13}$$

where $P(s)$ is the actual, perturbed plant, and $P_0(s)$ is the nominal plant.

The transfer function $\Delta_a(s)$ is used to describe a frequency dependent *unstructured uncertainty* as follows:

$$\Delta_a(s) = W_2(s) \cdot \tilde{\Delta}_a(s) \tag{14}$$

where the normalized perturbation $\tilde{\Delta}_a(s)$ is any *stable* transfer function with:

$$||\tilde{\Delta}_a||_\infty \leq 1 \tag{15}$$

This uncertainity description is closely related to a plant transfer function representation in C, the Nyquist plot, as shown in figure 9a). $||\tilde{\Delta}_a||_\infty = 1$ defines a circle, whose radius is scaled[13] with $W_2(s)$.

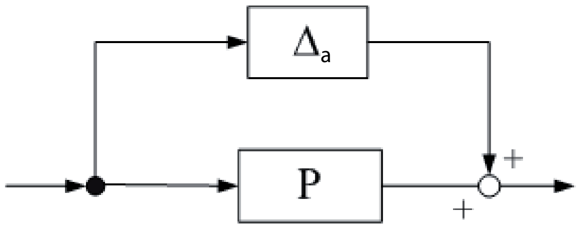


Figure 7: Additive uncertainty

Dynamic perturbations can also be described with multiplicative uncertainty. The corresponding block diagram is shown in figure 8.

$$P(s) = P_0(s) \cdot (1 + \Delta_m(s)) \tag{16}$$

In multivariable systems, transfer functions are matrices. It is known that matrix multiplication is not commutative, so for matrix–valued P_0 and Δ_m, $P_0(s) \cdot (I + \Delta_m(s))$ is generally different from $(I + \Delta_m(s)) \cdot P_0(s)$. As a consequence, input and output multiplicative perturbations have to be distinguished. Input multiplicative perturbations are able to model actuator uncertainty, whereas output multiplicative perturbations are used to describe sensor related uncertainties.

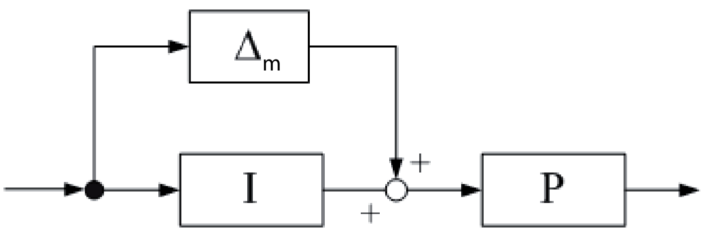


Figure 8: Multiplicative uncertainty

[13]Index 2 for W is commonly used for uncertainty weighting functions.

The capabilities of additive or multiplicative uncertainty representation are illustrated with an example. For comparison purposes the unmodeled high frequency dynamic is specified. It will be investigated how this a priori known model error can be represented with an additive or multiplicative uncertainty.

Example:

Many plants can roughly be approximated by a first order system, e.g. PT1. Such a model usually neglects the high frequency dynamic of the system. As it is well known, a control system with a PT1-plant can achieve any required closed loop performance. Consequently, we are interested in an uncertainty description which incorporates possible unmodeled phase loss into controller design and prevents a controller design with unrealistic high bandwidth.

The nominal plant model is

$$P_0(s) = \frac{5}{s+1}. \tag{17}$$

The unmodeled dynamic is represented with a set of transfer functions with the following elements:

$$P_\Delta(s) = \frac{1}{(\tau s+1)^2}, \quad \tau \in [0.02 \ldots 0.05]. \tag{18}$$

The plants to be controlled are:

$$P(s) = P_o(s) \cdot P_\Delta(s). \tag{19}$$

For comparison purposes the unmodeled dynamic described above is represented as an additive and multiplicative perturbation. Additive uncertainties are best represented in the complex plane $\mathcal{C}$, i.e. in the Nyquist diagram of figure (9a), whereas multiplicative uncertainties can easily be plotted in a Bode diagram of figure (9b). The figures show the plant's frequency response $P(j\omega)$ for different values of τ. Next, a single description of uncertainty has to be found for the specified range of values for the uncertain parameter τ. To accomplish this, the additive and multiplicative $\Delta(j\omega)$ are determined for a set of values of τ according to the following formulas:

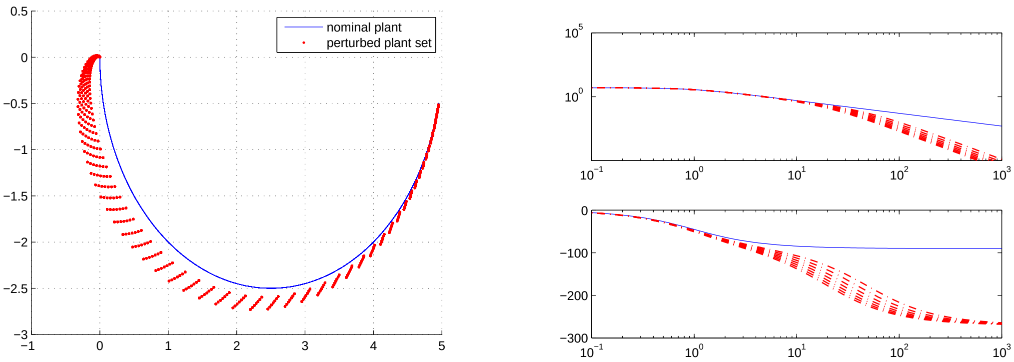
Additive unstructured perturbation:

$$\Delta_a(s) = P(s) - P_0(s). \tag{20}$$

Multiplicative unstructured perturbation:

$$\Delta_m(s) = \frac{P(s)}{P_0(s)} - 1. \tag{21}$$

The magnitude of the perturbations $\Delta_a(j\omega)$ and $\Delta_m(j\omega)$ are shown in figure 10. As defined in (14), frequency dependency is specified with the weighting transfer function $W_2(s)$. The solid line in the figures is an envelope of all error frequency responses and $\Delta(s) = W_2(s) \cdot \Delta(s)$ is therefore an upper bound for the uncertainty.



(a) Additive uncertainty (b) Multiplicative uncertainty

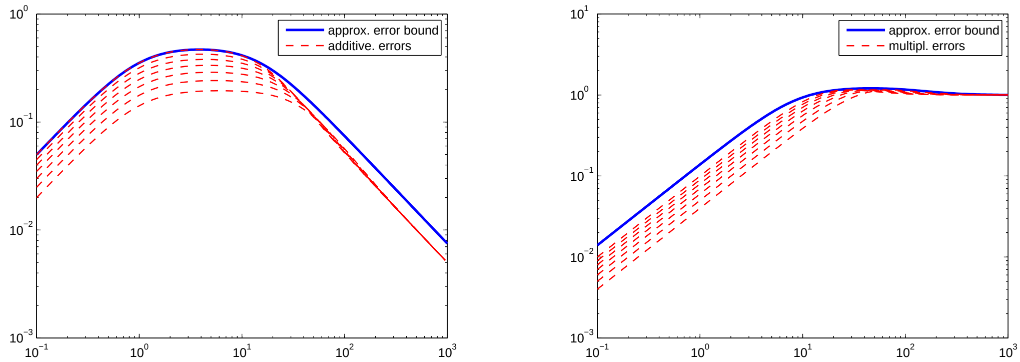
Figure 9: Plant uncertainty

Additive weighting:

$$W_{2a}(s) = \frac{7.5s}{(s+1)(s+15)}. \tag{22}$$

Multiplicative weighting:

$$W_{2m}(s) = \frac{0.997s(s+170)}{(s+9)(s+135)}. \tag{23}$$

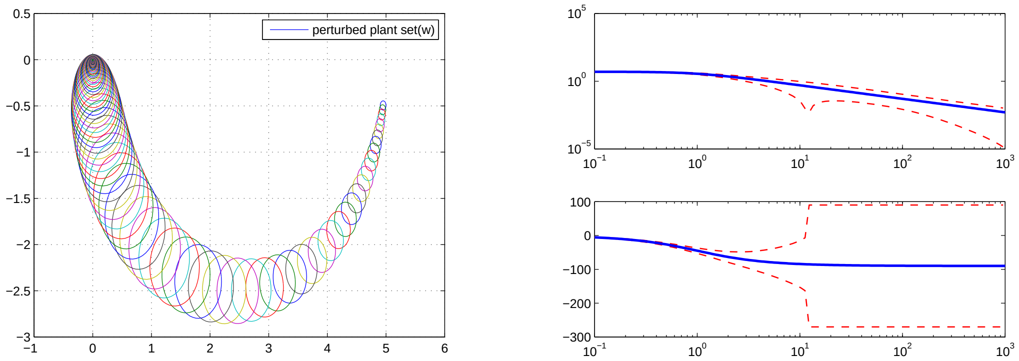


(a) Additive uncertainty (b) Multiplicative uncertainty

Figure 10: Uncertainty descriptions

In figure 11 the resulting uncertainty regions are plotted in the corresponding plots.

Compare the resulting regions with the set of plants in figure 9! Obviously, the uncertainty region is very large due to the fact, that the nominal plant is not in the center of the plant set. Since the unnecessary uncertainty regions are mainly opposite of the critical point, it might not have a large impact on controller design. After



(a) Additive bounds (b) Multiplicative bounds

Figure 11: Uncertainty bounds

$\omega = 10$ rad/s the 0–gain point lies within the uncertainty set. As a consequence, a plant phase cannot be determined anymore. In the Bode diagram of figure phase bounds are set to -270 and 90 degrees, in order to get a nice bode plot.

In a similar way, static nonlinearities (sector criterion) or uncertain dead–time can be described with unstructured uncertainties.

There arises the question, which of the representations of uncertainty should be used. Since the optimal $\mathcal{H}_\infty$–controller has the order of the plant plus the orders of all the weighting functions, it is preferable to choose a representation which leads to a minimal order weighting. In the SISO case, additive uncertainty can be recast into multiplicative uncertainty with simple algebraic operations:

$$P_0(s)\,\Delta_m(s) = \Delta_a(s) \tag{24}$$

Since the additive representation includes the plant transfer function $P_0(s)$ it is expected, that an additive representation usually is of higher degree. To illustrate this: uncertain dead–time can be fit into a multiplicative uncertainty description which is independent of the the plant:

$$P(s) = P_o(s) \cdot e^{-sT} \tag{25}$$

$$\Delta_m(s) = e^{-sT} - 1 \tag{26}$$

This function is shown in the amplitude plot of figure 12. It can be bounded with a DT1 transfer function.

The chosen uncertainty representation determines which closed-loop transfer function has to be considered in $\mathcal{H}_\infty$–controller design in order to guarantee robustness with respect to stability and performance.

Additive and multiplicative uncertainty are *special cases* of a more general framework which will be explained below. First, notice that we have to distinguish two different feedback loop: the feedback loop formed by the controller and the perturbed plant, and the feedback loop in which the uncertainty Δ resides. A general

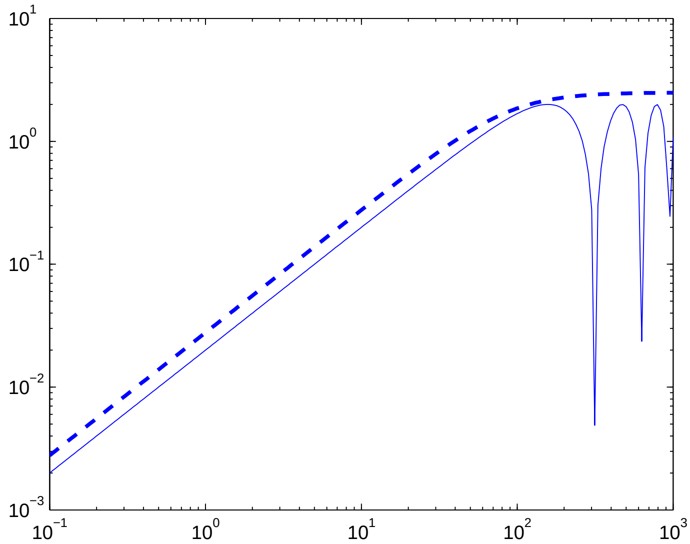


Figure 12: Dead–time uncertainty

solution for all different uncertainty descriptions can be obtained, if the control system is described in the feedback structure shown in figure 13. The plant P is

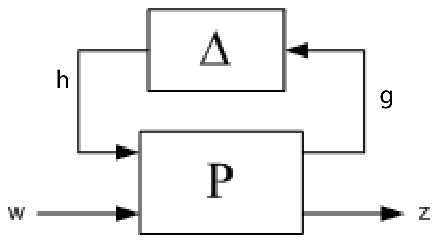


Figure 13: Standard P–Δ configuration

partitioned according to the dimensions of Δ.

$$P = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}. \tag{27}$$

With simple algebraic calculations it follows:

$$z = [P_{22} + P_{21}\Delta(I - P_{11}\Delta)^{-1}P_{12}]\, w \tag{28}$$

if $(I - P_{11}\Delta)^{-1}$ exists.

Define:

$$\mathcal{F}(P,\, \Delta) \;:=\; P_{22} + P_{21}\Delta(I - P_{11}\Delta)^{-1}P_{12} \tag{29}$$

$\mathcal{F}(P, \Delta)$ is called a *linear fractional transformation* LFT of P and Δ. In the single input single output case the LFT corresponds to a bilinear transform. This representation is also used for more sophisticated uncertainty representation (next section) and also to formulate the general $\mathcal{H}_\infty$–controller design problem.

Additive and multiplicative uncertainty representations are special cases of a linear fractional transformation. For additive perturbations, the matrix P becomes:

$$P = \begin{bmatrix} 0 & I \\ I & P_0 \end{bmatrix}. \tag{30}$$

For multiplicative perturbations, the matrix P becomes:

$$P = \begin{bmatrix} 0 & P_0 \\ I & P_0 \end{bmatrix}. \tag{31}$$

The example before shows that the conservative error bounds on the plant can be large, compared to the real plant perturbation. An unstructured uncertainty model is also not suited for perturbations which only affects a part of an interconnected system. This motivates a more general representation of uncertainty which will be introduced in the next section.

4.2 Structured Uncertainty

Structured uncertainty means uncertainty (tolerances) of concrete *physical parameters.* Some examples are listed below:

- electrical components like resistors or capacitors are always affected with tolerances, e.g. 2% for a standard resistor.
- In a magnetic bearing system, the nominal air gap is an important parameter which is affected by manufacturing tolerances and by thermal growth when the machine is running.
- The relative permeability μ_r of a magnetic material is usually not very precisely known

The internal structure of a linear plant can be represented as a block diagram with integrators, summators, and gains. The physical parameters are gains in the block diagram. It can be shown that the plant transfer function is always an LFT (linear fractional transform) with respect to every physical parameter $p_1, \ldots, p_n$.
As an example, consider the uncertain plant equation (17), (18) and (19) page 15. We will treat the time constant τ in $P_\Delta(s)$ as uncertain parameter lying in the range of $0.02 \ldots 0.05$. Suppose that the nominal value τ_0 corresponds to the mid–range value $\tau_0 = 0.035$, and $\tau = \tau_0 + \Delta_\tau$, where Δ_τ is a real parameter varying between $-0.015 \ldots + 0.015$. The uncertain system $P(s) = P_0(s) \cdot P_\Delta(s) = \frac{5}{s+1} \cdot \frac{1}{(\tau s+1)^2}$ can be recasted with the following linear fractional transform in figure 14. Note that the uncertainty block Δ in figure 14 is diagonal with the same diagonal element Δ_τ

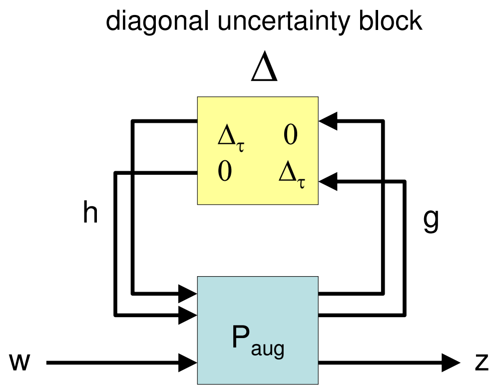


Figure 14: Structured uncertainty with an unknown time constant τ appearing twice.

repeated twice. The reason is that the uncertain time constant τ appears twice in the second order term $P_\Delta(s)$. The augmented plant $P_{aug}(s)$ will be of third order with 3 inputs and 3 outputs.
Exercise : Calculate $P_{aug}(s)$ and show that the associated linear fractional transform yields

$$\mathcal{F}(P_{aug}, \Delta) = \frac{5}{s+1} \cdot \frac{1}{((\tau_0 + \Delta_\tau)s + 1)^2} \tag{32}$$

There are available software tools capable to directly define and handle systems with structured or unstructured uncertainty. Using the Robust Control toolbox of Matlab the example above can be entered as follows :

```
>> P0 = tf(5, [1, 1])
Transfer function:

  5
-----
s + 1

>> tau = ureal('tau', 0.035, 'range', [0.02, 0.05])
Uncertain Real Parameter: Name tau, NominalValue 0.035, Range [0.02  0.05]
>> Pdelta = tf(1, [tau, 1])*tf(1, [tau, 1])
USS: 2 States, 1 Output, 1 Input, Continuous System
  tau: real, nominal = 0.035, range = [0.02  0.05], 2 occurrences
>> P = P0 * Pdelta
USS: 3 States, 1 Output, 1 Input, Continuous System
  tau: real, nominal = 0.035, range = [0.02  0.05], 2 occurrences
>> bode(P)
```

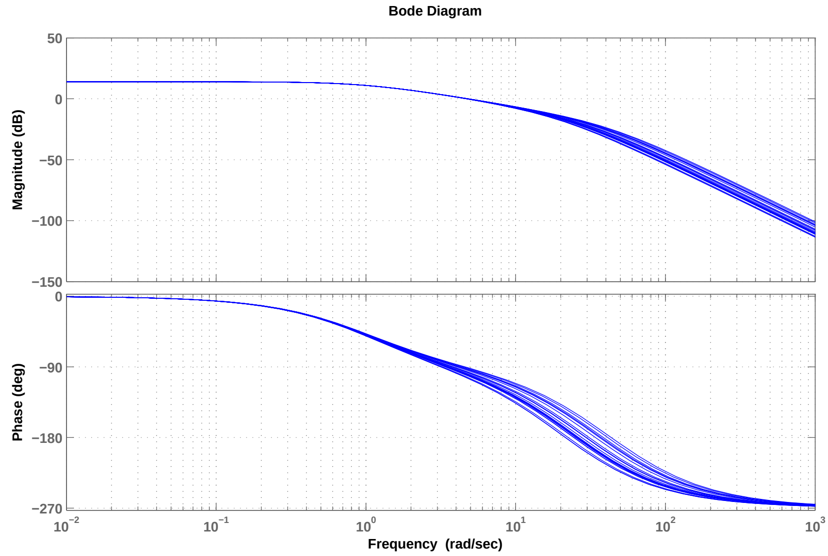


Figure 15: Example : structured uncertainty with an unknown time constant τ.

which gives the family of Bode plot in figure 15.

In contrast to unstructured uncertainty, see figure 13, *structured* uncertainty leads to a feedback configuration figure 15 with a *static diagonal* block Δ. In addition, the diagonal terms are *real–valued*, whereas in the unstructured case they are mostly *complex–valued* norm–bounded *transfer functions* $\Delta(s)$.

It is obvious that *structured* uncertainty representations are often closer to physical specifications. However, the algorithms needed to tackle robust control analysis or synthesis problems with *structured* uncertainty are much more complicated, see outlook section page 26.

We disadvise from recasting highly *structured* parameter uncertainty as additive or multiplicative *unstructured* uncertainty, because this implies a high degree of conservativeness. Indeed, the class of perturbations may become excessively large, and the resulting controller, if there is any complying with the robustness specifications is likely to show poor performance.

5 Formulation of the Standard $\mathcal{H}_\infty$ Problem

In this section the controller design problem is formulated in the $\mathcal{H}_\infty$ formalism. The objective is to design a controller, which does not only have nominal stability and performance, but has this properties with all the plants represented by some uncertainty description. $\mathcal{H}_\infty$–controller design was specially developed, to solve this problem in a systematic manner. As system which is stable with all the plant of the uncertainty set is called robustly stable. It has a robust performance, if the performance specifications are met for all the admissible plant behaviors.

The control system with controller and uncertainty model is shown in figure 16. In $\mathcal{H}_\infty$–controller design, the $\mathcal{H}_\infty$–norm of the mapping from w to z can be minimized. As a consequence the control problem has to be formulated in this formalism. The inputs w are typically reference or disturbance signals, whereas the outputs z can be the control error or the controller output. The $\mathcal{H}_\infty$–controller

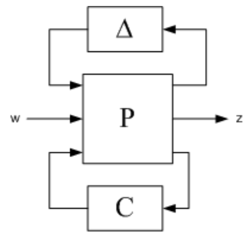
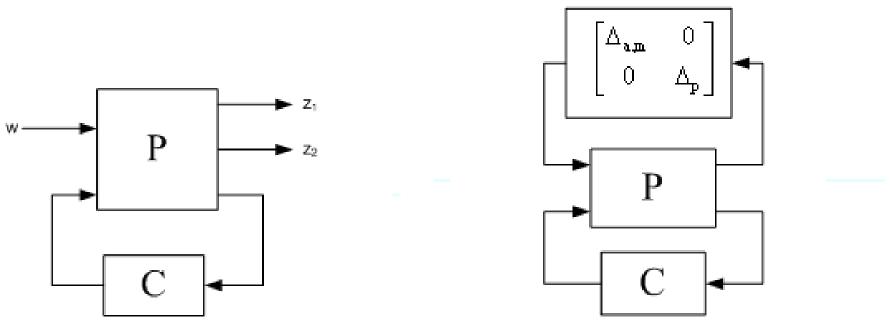


Figure 16: Control system with uncertainty

design problem cannot directly by solved in structure of figure 16. Since performance specifications and stability conditions can both be expressed as $\mathcal{H}_\infty$–norm conditions on some transfer functions, there are two ways to simplify the control system of figure 16. Either the stability conditions are formulated in the same formalism as performance specifications, resulting in a structure as in figure 17(a), or the performance specification is formulated similarly to the uncertainty description Δ_p (see figure 17(b)). For unstructured uncertainties the structure of figure 17(a) is easier to go, but for structured uncertainties, only the structure in figure 17(b) leads to an elegant representation. The structured uncertainty problem is solved with μ-synthesis, which is beyond the scope of this introduction. Consequently, the first approach will be followed in the sequel.



(a) Robust Stability as norm condition (b) Performance as artificial plant uncertainty

Figure 17: Possible system representations

As a result from the small gain section, it is clear that as system is robustly stable, if

for additive uncertainty:

$$||W_2\, C(I + PC)^{-1}||_\infty \leq 1 \tag{33}$$

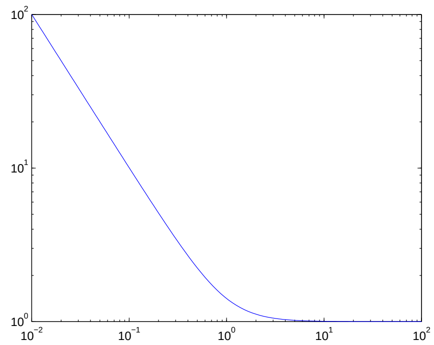


Figure 18: Performance Weighting Function

for multiplicative uncertainty:

$$||W_2\, PC(I + PC)^{-1}||_\infty \leq 1 \tag{34}$$

In a control system as shown in figure 19 the robust stability condition corresponds to the $\mathcal{H}_\infty$–norm of the mapping from r to u for the additive case and form r to y for the multiplicative case.

System performance is usually specified by conditions on disturbance attenuation, i.e. on the $\mathcal{H}_\infty$–norm of the mapping from d to y, or for reference tracking on the $\mathcal{H}_\infty$–norm of $r \rightarrow e$. For both mappings an acceptable performance is achieved, if the following condition on the sensitivity function $S = (I + PC)^{-1}$ is met:

$$||\gamma W_1\, (I + PC)^{-1}||_\infty \leq 1 \tag{35}$$

This condition is fulfilled, if $S(j\omega) \leq \frac{1}{|W_1(j\omega)|}$. Consequently, a typical weighting function $W_1(s)$ has a shape as shown in figure 18. An integrator in $W_1(s)$ forces the sensitivity function to be zero at $\omega = 0$. The parameter $\gamma < 1$ is used as an optimization parameter. The larger γ the better the disturbance attenuation.

Performance and robust stability can therefore be achieved, if the two conditions are combined to the following condition (mixed sensitivity approach):

Additive uncertainty:

$$\left\|\begin{bmatrix} \gamma W_1\, (I + PC)^{-1} \\ W_{2,a}\, C(I + PC)^{-1} \end{bmatrix}\right\|_\infty . \tag{36}$$

Multiplicative uncertainty:

$$\left\|\begin{bmatrix} \gamma W_1\, (I + PC)^{-1} \\ W_{2,m}\, PC(I + PC)^{-1} \end{bmatrix}\right\|_\infty . \tag{37}$$

The block diagram representation of the mixed sensitivity problem is shown in 19. Here it becomes obvious, that the robust stability condition can be viewed as a

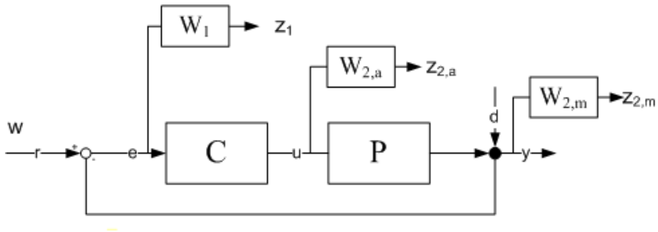


Figure 19: Control System for mixed Sensitivity Optimization

mapping of signals.
A controller can be found by solving the following optimization problem:
Additive uncertainty:

$$\max_{\gamma} \left\| \begin{bmatrix} \gamma W_1(s)(I+PC)^{-1} \\ W_{2,a}(s)C(I+PC)^{-1} \end{bmatrix} \right\|_{\infty} = 1, \tag{38}$$

(Similarly for the multiplicative uncertainty)

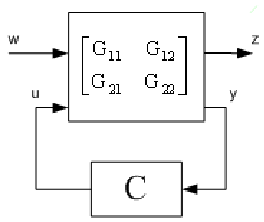


Figure 20: LFT for $\mathcal{H}_\infty$–optimization

With some matrix algebra, the equation (38) can be transformed into a lower linear fractional transformation $\mathcal{F}(P,C)$ as shown in figure 20. Therefore, the following problem has to be solved:

Find a stabilizing controller C, for which:

$$||\mathcal{F}(P,C)||_\infty := ||P_{11} + P_{12}C(I-P_{22}C)^{-1}P_{21}||_\infty \ = \ 1 \tag{39}$$

Nowadays, various solutions to this problem are known. The two most important are state–space methods based on the solution of two Riccati equation and the solution based on linear matrix inequalities (LMI). Special attention has to be payed on additional assumptions, that are imposed for the algorithms. Some of them will be treated in the next chapter.

6 A Glimpse on the $\mathcal{H}_\infty$ State Space Solution

$\mathcal{H}_\infty$–controller design problems can be formulated in different ways, but finally, the problem has to be brought to a representation as shown in figure 20. Its state–space representation is:

$$
\begin{aligned}
\dot{x} &= Ax + B_1 w + B_2 u && (40)\\
z &= C_1 x + D_{11} w + D_{12} u && (41)\\
y &= C_2 x + D_{21} w + D_{22} u && (42)\\
& && (43)
\end{aligned}
$$

The system matrices are usually represented as follows:

$$
G_s = \begin{bmatrix} A & \vdots & B_1 & B_2 \\ \cdots & & \cdots & \cdots \\ C_1 & \vdots & D_{11} & D_{12} \\ C_2 & \vdots & D_{21} & D_{22} \end{bmatrix}. \tag{44}
$$

A closed optimal solution for the above general system is not yet published. For the $\mathcal{H}_\infty$ problem a closed solution was first developped in [2] for the following special case:

$$
G_s = \begin{bmatrix} A & \vdots & B_1 & B_2 \\ \cdots & & \cdots & \cdots \\ C_1 & \vdots & 0 & \begin{bmatrix} 0 \\ I \end{bmatrix} \\ C_2 & \vdots & \begin{bmatrix} 0 & I \end{bmatrix} & 0 \end{bmatrix}. \tag{45}
$$

In a design tool the general state space description of equation (44) is transformed into the representation of equation (45) by means of loop shifting [3]. The transformation of D_{21} and D_{12} into the form with the identity matrix, requires that the two matrices have full rank. This condition is usually violated, when weighting functions are strictly proper, i.e. have a zero D–matrix. To circumvent this problem, the weighting D-matrix can usually be set to a small value, without influencing the controller design.

For a solution to exist, (A, B_2) must be stabilisable and (A, C_2) must be detectable.

Another important aspect is that the $\mathcal{H}_\infty$ problem must have a solution at $\omega = \infty$. An unappropriate weighting at $\omega = \infty$ can lead to unsatisfactory controller designs. For a closed loop system with plant as in equation (44) the direct feedthrough term (gain at $\omega = \infty$) is:

$$
D_{cl} = D_{11} + D_{12}(I - D_c D_{22})^{-1} D_c D_{21} \tag{46}
$$

where D_c is the controller D-matrix.
The term D_{11} is modified with the controller D_c-matrix. This can only be done to a limited extent, which is depending on the sizes of D_{12} and D_{21}. Bounds are documented in [3].

7 Limitation of $\mathcal{H}_\infty$ Methods

In this section some of the limitations of $\mathcal{H}_\infty$–controller design will be summarized.

- The $\mathcal{H}_\infty$–controller design methods offers powerful tools to solve various design problems. Nevertheless $\mathcal{H}_\infty$ performance measures are not always adequate for the investigated design problem and although robustness and performance measures are combined in the design procedure, it does not really guarantee, that performance is robust with respect to plant uncertainty. Consequently it was proposed to combine $\mathcal{H}_\infty$ robustness measures with other performance measures, for example measures similar to the LQR-design, the mixed $\mathcal{H}_2/\mathcal{H}_\infty$ design. Also LMI (see chapter 'outlook') offers many possibilities for 'mixed' design problems.

- $\mathcal{H}_\infty$–controller design leads to controllers of high order. A high order controller needs more ressources for its implementation and is suspect to numerical problems. Therefore it is favorable to have low order controllers. The optimal solution for a system as described by equation ((44)) has the same order as the system itself (size of the A-matrix). The order of the system is the order of the plant to be controlled and the sum of the orders of all the weighting functions, see figure 19. As already mentioned in the section on uncertainty description, the weighting functions have to be of minimal order. The problem of high order controllers can be diminished with order reduction methods. There are several order reduction methods available. These can be applied either on plant before $\mathcal{H}_\infty$–controller design or after design on the controller itself. The designer should be cautious not to spoil the controller design with a radical order reduction. There are also suboptimal $\mathcal{H}_\infty$–controller design algorithm which allow a direct design of a reduced order controller, but these are not currently available in the commercially available controller design tools, although a direct low order controller design is doubtless the way to go.

- As it can be seen from figures 11 the uncertainty description is not tight in the sense, that it incorporates a much larger set of plants than necessary. This leads to a conservative controller design, which might result in very poor closed loop performance. This problem is reduced, when a structured uncertainty description is used in conjuction with μ–synthesis (see Chapter 'Outlook').

8 Outlook: μ Synthesis and LMI Methods

8.1 Structured Singular Values (SSV) and μ Synthesis

The standard $\mathcal{H}_\infty$ minimizes an $\mathcal{H}_\infty$ norm between input signal w and output signal z. Often these signals are vector–valued (MIMO), as it is already the case for z in the mixed–sensitivity approach. In practice we are interested to *individual* transfer functions between a scalar component w_k and a scalar component z_l. A large amount of conservativeness is introduced in the design when optimizing an overall $\mathcal{H}_\infty$–norm.

The μ framework allows a more *selective* optimization related to *structured* uncertainty. A detailed description is out of the scope of this document. Basically the μ framework introduces a new norm based on *structured singular values* which is the new quantity to be minimized. The solution procedure is iterative (D–K iteration) and involves a sequence of minimizations, first over the controller K (holding the scaling variable D associated with the μ), and then optimizing over the scaling D (holding the controller K fixed). The D–K procedure is not guaranteed to converge neither to the global minimum nor to a local minimum of the μ value, but often works well in practice. It has been applied to a number of real–world applications with success. These applications include vibration suppression for flexible structures, flight control, chemical process control, and acustic reverberation suppression in enclosures.

8.2 Linear Matrix Inequalities (LMI)

Linear Matrix Inequalities (LMI) have emerged as powerful numerical design tools in areas ranging from robust control (e.g. $\mathcal{H}_\infty$) to system identification.

The canonical form of linear matrix inequality is any constraint of the form

$$A(x) = A_0 + x_1 A_1 + \ldots + x_n A_n < 0 \tag{47}$$

where $x = [x_1, \ldots, x_n]$ is a vector of unknowns called *decision variables*[14], and $A_0, \ldots, A_n$ are given symmetric matrices. $A(x) < 0$ stands for *negative definite*, i.e. all eigenvalues of $A(x)$ are negative. There are three generic LMI problems :

Feasibility problem: Find a solution x to the LMI system $A(x) < 0$. The set of all solutions is also called *feasibility set*.

Linear objective minimization problem: Minimize $c^T x$ subject to $A(x) < 0$. It is easy to show that the feasibility set of equation (47) is *convex*. Convexity has an important consequence : even though the optimization problem has no *analytical* solution in general, it can be solved *numerically* with a guaranteed convergence to the solution when one exists. This is in sharp contrast to general nonlinear optimization algorithms which may not converge towards the global optimum.

Generalized eigenvalue minimization problem: Minimize λ subject to $A(x) < \lambda\, B(x)$, $B(x) > 0$, and $C(x) < 0$.

Efficient[15] interior–point algorithms [4] are available to solve these three generic LMI problems. Many control problems and general design specifications can be formulated in terms of LMI. The main strength of LMI formulations is the ability to

[14] It is possible to convert from *scalar* decision variables to *matrix* valued variables.

[15] However the complexity of LMI computations can grow quickly with the problem order. For example, the number of operations required to solve a standard Riccati equation is $o(n^3)$ where n is the state dimension. Solving an equivalent LMI Riccati inequality needs $o(n^6)$ operations!

combine various *convex* design constraints or objectives in a numerically tractable manner [5] [6].

A non–exhaustive list of problems addressed by LMI techniques include the following:

1. robust pole placement
2. optimal LQG control
3. robust $\mathcal{H}_\infty$ control
4. multi–objective synthesis
5. robust stability of systems with structered parameter uncertainty (μ–analysis)

It is fair to say the advent of LMI optimization has significantly influenced the direction of research in robust control. A widely–accepted technique for numerically solving robust control problems is to reduce them to LMI problems.

9 Conclusion

Robust control emerged around 1980, and the progress in theory and numerical algorithms during the last 25 years has been enormous. Efficient commercial and public domain software tools are available nowadays. It is possible to successfully use these tools without understanding the deepest details of the underlying theory. However, a minimum of theoretical knowledge is necessary, and as it is the case with any complex scientific software tools[16], a lot of *practical experience* is needed before being able to solve real–world[17] problems.

In robust control, specific reasons for this are the following: Modelling is a challenging *engineering task*, and finding an estimation the modelling uncertainty needs some experience. Additionally, either control specifications are not entirely known in the beginning, i.e. incomplete, or real–world specifications may be very complex[18] to such an extent that *no direct* synthesis procedure exist. No matter which controller synthesis method is used it remains an *iterative* process which also needs some experience. In robust control, the user often needs to play around with frequency weighting functions in order to understand the performance/robustness trade–offs of his specific problem.

One of the main achievements in robust control might be the consciousness about the *industrial importance* of robustness. Generally speaking, system failure is not accepted in our society, and robustness considerations will remain very important.

[16] e.g. tools for finite element modelling, or any other complex scientific tool
[17] = non–academic
[18] e.g. mixed time domain and frequency domain specifications, low controller order, etc.

Bibliography

[1] G. Stein. Respect the unstable. *IEEE Control Systems Magazine*, August 2003.

[2] J. Doyle, K. Glover, P.P. Khargonekar, and B.A. Francis. State–space solutions to standard $\mathcal{H}_2$ and $\mathcal{H}_\infty$ control problems. *IEEE Trans. Aut. Control*, 34(8), August 1989.

[3] M. Safonov and D. Limebeer. Simplifying $\mathcal{H}_\infty$ theory via loop shifting. *Proc. 27th Conference on Decision and Control*, December 1988.

[4] Y. Nesterov and A. Nemirowski. *Interior Point Polynomial Methods in Convex Programming: Theory and Applications.* Studies in Applied Mathematics, SIAM books Philadelphia, 1994.

[5] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in Systems and Control Theory.* Studies in Applied Mathematics, SIAM books Philadelphia, vol. 15, 1994.

[6] P. Gahinet and P. Apkarian. A linear matrix inequality approach to $\mathcal{H}_\infty$. *Int. J. Robust and Nonlinear Control*, 4, 1994.

[7] M. Green and D. Limebeer. *Linear Robust Control.* Pearson Education, Inc., 2002.

[8] J. Doyle, B. Francis, and A. Tannenbaum. *Feedback Control Theory.* McMillan Publishing Co., 1990.

[9] *Robust Control Toolbox 3, User's Guide, The Mathworks*, 2008.

[10] *LMITOOL: A Package for LMI Optimization in Scilab User's Guide*, 1995.

[11] D. Gu, P. Petkov, and M. Konstantinov. *Robust Control Design with MATLAB.* Springer, 2005.

Exercises

1 First exercise: two cart problem

Consider the following mechanical plant: two carts are moving on a horizontal plane with no friction. The control force w is acting on the left cart with mass m_1, and the position of the right cart $z = x_2$ is measured. The carts are connected by a linear spring k. The nominal parameters are $m_1 = 1$ kg, $m_2 = 1$ kg, and $k = 1$ N/m. All parameters are affected by *independent* tolerances of $\pm 10\%$ for the masses, and $\pm 20\%$ for the spring.

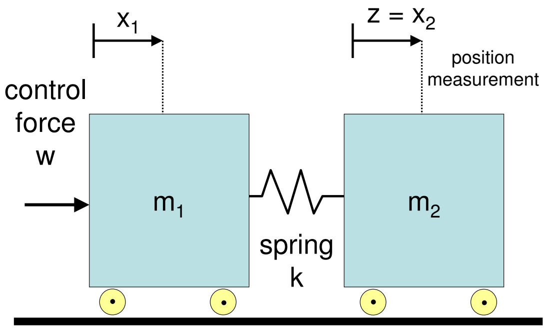


Figure 21: Two cart ACC benchmark problem

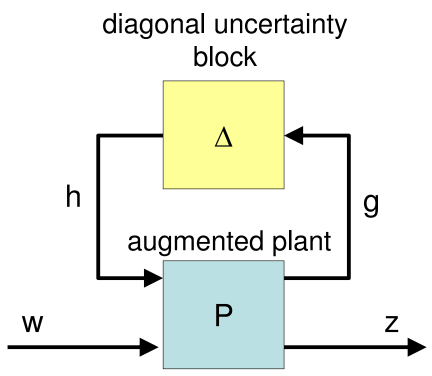


Figure 22: Structured uncertainty setup

1. Find a state space realization of the augmented plant $\mathbf{P}$, where the structured parameter uncertainty is represented as a diagonal feedback block

$$\boldsymbol{\Delta} = \begin{bmatrix} \delta_1 & 0 & 0 \\ 0 & \delta_2 & 0 \\ 0 & 0 & \delta_3 \end{bmatrix}.$$

   The uncertainty δ_1 should correspond to the uncertainty of m_1, and a variation of $-1 < \delta_1 < +1$ should correspond to a variation of $\pm 10\%$ of m_1 around its nominal value. Similarly, δ_2, δ_3 correspond to the uncertainties of m_2, k, and both should also be normalized to a range of ± 1.

2. Why is it inappropriate to formulate the uncertainties of this system as additive or multiplicative uncertainties?

2 Second exercise: drawback of classical stability margins

Consider a feedback system with the following loop gain

$$L(s) = \frac{0.38(s^2 + 0.1s + 0.55)}{s(s+1)(s^2 + 0.06s + 0.5)}.$$

The feedback system exhibits very good gain and phase margins, but is not at all robust with respect to *simultaneous* uncertainties of gain and phase.

1. Calculate the gain margin A_m and the phase margin ϕ_m using Matlab.
2. Determine the peak of the sensitivity $||S||_\infty$.
3. Determine the distance between the Nyquist plot of loop gain $L(j\omega)$ and the critical point -1.

This example shows that it is much better to assess robustness using $||S||_\infty$ instead of using traditional gain and phase margins. A low value of $||S||_\infty$, e.g. implies good gain and phase margins while the converse is not true. It follows from elementary trigonometric relations that

$$A_m > \frac{||S||_\infty}{||S||_\infty - 1}, \quad \phi_m > 2 \ \arcsin \frac{1}{||S||_\infty}$$

For example $||S||_\infty = 2$ gives a garanteed gain margin of $A_m > 2$, corresponding to a worst case gain margin of 6 dB, and a phase margin $\phi_m > 60°$.