Multivariable Feedback Control

Skogestad, S., & Postlethwaite, I.

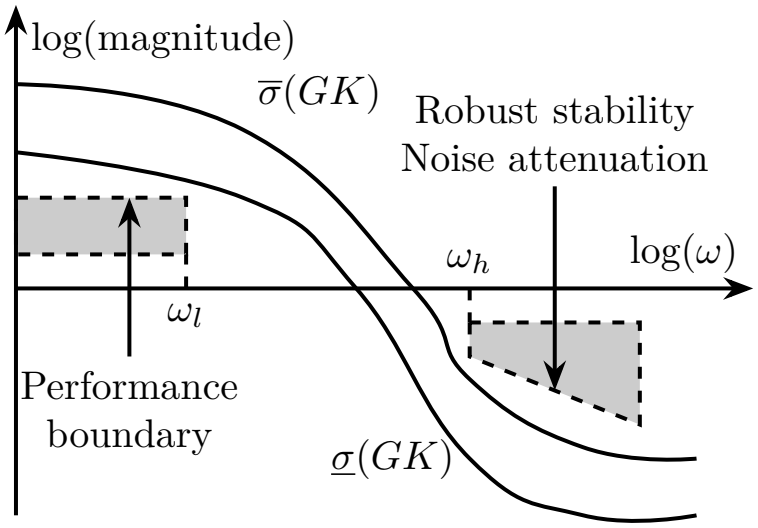


Table of Contents

1 Introduction **1**
- 1.1 The Process of Control System Design . 1
- 1.2 The Control Problem 1
- 1.3 Transfer Functions 1
- 1.4 Scaling 1
- 1.5 Deriving Linear Models 2
- 1.6 Notation 2

2 Classical Feedback Control **3**
- 2.1 Frequency Response 3
- 2.2 Feedback Control 3
- 2.3 Closed Loop Stability 3
- 2.4 Evaluating Closed-Loop Performance . . 4
- 2.5 Controller Design 5
- 2.6 Loop Shaping 5
- 2.7 Shaping Closed-Loop Transfer Functions 7

3 Introduction to Multivariable Control **9**
- 3.1 Introduction 9
- 3.2 Transfer Functions 9
- 3.3 Multivariable Frequency Response Analysis 9
- 3.4 Control of Multivariable Plants 10
- 3.5 Introduction to MIMO RHP-Zeros . . . 11
- 3.6 Condition Number and RGA 12
- 3.7 Introduction to Robustness for MIMO Plants 12
- 3.8 General Control Problem Formulation . 12
- 3.9 Conclusion 14

4 Elements of Linear System Theory **15**
- 4.1 System Descriptions 15
- 4.2 State Controllability and State Observability 16
- 4.3 Stability 16
- 4.4 Poles 16
- 4.5 Zeros 17
- 4.6 Some Remarks on Poles and Zeros . . . 17
- 4.7 Internal Stability of Feedback Systems . 18
- 4.8 Stabilizing Controllers 18
- 4.9 Stability Analysis in the Frequency Domain 18
- 4.10 System Norms 19

5 Limitations on Performance in SISO Systems **20**
- 5.1 Input-Output Controllability 20
- 5.2 Perfect Control and Plant Inversion . . 20
- 5.3 Constrain of S and T 20
- 5.4 Limitation Imposed by Time Delays . . 21
- 5.5 Limitation Imposed by RHP-Zeros . . . 21
- 5.6 Limitation Imposed by RHP-Poles . . . 22
- 5.7 Combined Unstable (RHP) Poles and Zeros 22
- 5.8 Performance Requirements Imposed by Disturbances and Commands 22
- 5.9 Limitation Imposed by Input Constraints 23
- 5.10 Limitation Imposed by Phase Lag . . . 23
- 5.11 Limitation Imposed by Uncertainty . . . 23
- 5.12 Summary: Controllability Analysis with Feedback Control 24
- 5.13 Conclusion 24

6 Limitations on Performance in MIMO Systems **25**
- 6.1 Introduction 25
- 6.2 Constraints on S and T 25
- 6.3 Functional Controllability 26
- 6.4 Limitation Imposed by Time Delays . . 26
- 6.5 Limitations Imposed by RHP-Zeros . . . 26
- 6.6 Limitation Imposed by Unstable (RHP) Poles 26
- 6.7 RHP-poles Combined with RHP-Zeros . 27
- 6.8 Limitations Imposed by Disturbances . . 27
- 6.9 Limitations Imposed by Input Constraints 27
- 6.10 Limitation Imposed by Uncertainty . . . 28
- 6.11 MIMO Input-Output Controllability . . 29
- 6.12 Conclusion 30

7 Uncertainty and Robustness for SISO Systems **32**
- 7.1 Introduction to Robustness 32
- 7.2 Representing Uncertainty 32
- 7.3 Parametric Uncertainty 33
- 7.4 Representing Uncertainty in the Frequency Domain 33
- 7.5 SISO Robust Stability 36
- 7.6 SISO Robust Performance 37
- 7.7 Examples of Parametric Uncertainty . . 38
- 7.8 Conclusion 39

8 Robust Stability and Performance Analysis **40**
- 8.1 General Control Configuration with Uncertainty 40
- 8.2 Representing Uncertainty 40
- 8.3 Obtaining P, N and M 42
- 8.4 Definitions of Robust Stability and Robust Performance 42
- 8.5 Robust Stability for the $M\Delta$-structure . 43
- 8.6 RS for Complex Unstructured Uncertainty 43
- 8.7 RS with Structured Uncertainty: Motivation 44
- 8.8 The Structured Singular Value 45
- 8.9 Robust Stability with Structured Uncertainty 46
- 8.10 Robust Performance 46
- 8.11 Application: RP with Input Uncertainty 47
- 8.12 μ-synthesis and DK-iteration 49
- 8.13 Further Remarks on μ 51
- 8.14 Conclusion 51

9 Controller Design **53**
9.1 Trade-offs in MIMO Feedback Design . 53
9.2 LQG Control 54
9.3 $\mathcal{H}_2$ and $\mathcal{H}_\infty$ Control 55
9.4 $\mathcal{H}_\infty$ Loop-Shaping Design 60
9.5 Conclusion 64

10 Controller Structure Design **65**
10.1 Introduction 65
10.2 Optimization and Control 65
10.3 Selection of Controlled Outputs 66
10.4 Selection of Manipulations and Measurements 67
10.5 RGA for Non-Square Plant 68
10.6 Control Configuration Elements 68
10.7 Hierarchical and Partial Control 70
10.8 Decentralized Feedback Control 72

11 Model Reduction **77**
11.1 Introduction 77
11.2 Truncation and Residualization 77
11.3 Balanced Realization 78
11.4 Balanced Truncation and Balanced Residualization 78
11.5 Optimal Hankel Norm Approximation . 79
11.6 Model Reduction - Practical Summary . 79
11.7 Reduction of Unstable Models 80
11.8 Conclusion 80

1 Introduction

1.1 The Process of Control System Design

The process of designing a control system is a step by step design procedure as follows:

1. Study the system (plant) to be controlled and obtain initial information about the **control objectives**
2. **model the system** and simplify the model, if necessary
3. **scale the variables** and analyze the resulting model; determine its properties
4. Decide which variables are to be controlled (controlled outputs)
5. Decide on the measurements and manipulated variables: what sensors and actuators will be used and where will they be placed?
6. Select the **control configuration**
7. Decide on the type of controller to be used
8. Decide on performance specifications, based on the overall control objectives
9. Design a controller
10. Analyze the resulting controlled system to see if the specifications are satisfied; and if they are not satisfied modify the specifications or the type of controller
11. Simulate the resulting controlled system
12. Repeat from step 2 if necessary
13. Choose hardware and software and implement the controller
14. Test and validate the control system, and tune the controller on-line, if necessary

Input-output controllability analysis is studied in section 5 for SISO systems and in section 6 for MIMO systems. The steps 4, 5, 6 and 7 are corresponding to the **control structure design**. This is treated in section 10. The design of the controller is described in section 9. The analysis of performance and robustness of a controlled system is studied in sections 7 and 8.

1.2 The Control Problem

The objective of a control system is to make the output y behave in a desired way by manipulating the plant input u. The **regulator problem** is to manipulate u to counteract the effect of a disturbance d. The **servo problem** is to manipulate u to keep the output close to a given reference input r.

In both cases, we want the control error $e = y - r$ to be small. The algorithm for adjusting u based on y is the **controller** K. To arrive at a good design for K we need information about the expected disturbances, the reference inputs, the plant model G and disturbance model G_d.

A major source of difficulty is that models may be inaccurate or may change with time. The inaccuracy in G may cause instability problems as it is part of the feedback loop. To deal with such a problem, the concept of **model uncertainty** will be used.

Definitions

Nominal Stability (NS) The system is stable with no model uncertainty

Nominal Performance (NP) The system satisfies the performance specifications with no model uncertainty

Robust Stability (RS) The system is stable for all perturbed plants about the nominal model up to the worst case uncertainty

Robust Performance (RP) The system satisfies the performance specifications for all perturbed plants about the nominal model up to the worst-case model uncertainty

1.3 Transfer Functions

Properties of transfer functions:

- A system $G(s)$ is **strictly proper** if $G(s) \to 0$ as $\omega \to \infty$
- A system $G(s)$ is **semi-proper** if $G(s) \to D \neq 0$ as $\omega \to \infty$
- A system $G(s)$ is **proper** if $G(s)$ is strictly proper or semi-proper
- The order of the system noted n and is the order of the denominator (or pole polynomial) of its matrix transfer function

1.4 Scaling

Scaling is very important in applications, both for model analysis (input-output controllability) and for controller design.

The scaling is done by **dividing each variable by its maximum expected or allowed change**. That way, the scaled variable should be less than one in magnitude.

We denote variables in their unscaled units by a hat.

- $d = \hat{d}/D_d$ with $D_d = \hat{d}_{\max}$ is the largest expected change in disturbance
- $u = \hat{u}/D_u$ with $D_u = \hat{u}_{\max}$ is the largest allowed input change

The variables $\hat{y}$, $\hat{r}$ and $\hat{e}$ are in the same unit, so we choose to scale them with respect to the maximum allowed control error:

- $e = \hat{e}/D_e$ with $D_e = \hat{e}_{\max}$ is the largest allowed control error
- $r = \hat{r}/D_e$
- $y = \hat{y}/D_e$

For MIMO systems, each variables in the vectors $\hat{d}$, $\hat{r}$, $\hat{u}$ and $\hat{e}$ may have a different maximum value, in which case D_e, D_u, D_s and D_r, become diagonal scaling matrices.

Scaled transfer functions

$$G = D_e^{-1}\hat{G}D_u$$
$$G_d = D_e^{-1}\hat{G}_d D_d$$

We then obtain the following model in terms of scaled variables:

$$y = Gu + G_d d$$

where u and d should be less than 1 in magnitude.

It is sometimes useful to introduce a **scaled reference** $\tilde{r}$ which is less than 1 in magnitude: $\tilde{r} = \hat{r}/\hat{r}_{\max} = D_r^{-1}\hat{r}$ Then we have $r = R\tilde{r}$ with $R \triangleq D_e^{-1}D_r = \hat{r}_{\max}/\hat{e}_{\max}$ is the largest expected change in reference relative to the allowed control error.

With scaling you make initial decision regarding performance. This makes **weight selection simple later** (may often select identity weights if initial scaling is reasonable!).

1.5 Deriving Linear Models

Linear models may be obtained from physical "first-principle" models or from analyzing input-output data (**identification**).

In order to obtain a linear model from the "first-principle", the following approach is used:

1. Formulate a nonlinear state-space model based on physical knowledge
2. Determine the steady-state operating point about which to linearize
3. Introduce deviation variables and linearize the model

1.6 Notation

Notations used throughout this note are summarized in tables 1, 2 and 3.

Table 1 – *Notations for the conventional control configuration*

Notation	Meaning
G	Plant model
K	Controller
G_d	Disturbance model
r	Reference inputs
n	Measurement noise
y	Plant outputs
y_m	Measurements
u	Control signals

Table 2 – *Notations for the general configuration*

Notation	Meaning
P	Generalized plant model
w	Exogenous inputs: commands, disturbances, noise
z	Exogenous outputs: signals to be minimized
v	Controller inputs: measurements
u	Control signals

Table 3 – *Notations for transfer functions*

Notation	Meaning
L	Loop gain: $L = GK$
S	Sensitivity function: $S = (I + L)^{-1}$
T	Complementary sensitivity function: $T = (I + L) * (I + L)^{-1}$

2 Classical Feedback Control

2.1 Frequency Response

By replacing s by $j\omega$ in a transfer function $G(s)$, we get the **frequency response** description. It can be used to describe:

- A system's response to sinusoids of varying frequency
- The frequency content of a **deterministic signal** via the **Fourier transform**
- The frequency distribution of a **stochastic** signal via the **power spectral density**

After sending a sinusoidal signal through a system $G(s)$, the signal's magnitude is amplified by a factor $|G(j\omega)|$ and its phase is shifted by $\angle G(j\omega)$.

Minimum Phase Systems - Definition

minimum phase systems are systems with no time delays or RHP-zeros.
The name minimum phase refers to the fact that such a system has the minimum possible phase lag for the given magnitude response $|G(j\omega)|$.
RHP-zeros and **time delays** contribute additional phase lag to a system when compare to that of a minimum phase system with the same gain (hence the term **non-minimum phase system**).

For minimum phase systems, there is a unique relationship between the gain and phase of the frequency response: the **Bode gain-phase relationship**:

$$\angle G(j\omega_0) = \frac{1}{\pi}\int_{-\infty}^{\infty} \frac{d\ln|G(j\omega)|}{d\ln\omega} \ln\left|\frac{\omega+\omega_0}{\omega-\omega_0}\right| \frac{d\omega}{\omega} \quad (1)$$

We note $N(\omega_0) = \left(\frac{d\ln|G(j\omega)|}{d\ln\omega}\right)_{\omega=\omega_0}$ that corresponds to the **slope of the magnitude** of $G(s)$ in log-variables. We then have the following approximation of the **Bode gain-phase relationship**:

$$\angle G(j\omega_0) \approx \frac{\pi}{2} N(\omega_0) \quad (2)$$

2.2 Feedback Control

a One Degree-of-Freedom Controller

The simple one degree-of-freedom controller negative feedback structure is represented in Fig. 1.
The input to the controller $K(s)$ is $r - y_m$ where $y_m = y + n$ is the measured output and n is the measurement noise. Thus, the input to the plant is $u = K(s)(r-y-n)$. The objective of control is to manipulate u (design K) such that the control error e remains small in spite of disturbances d. The control error is defined as $e = y - r$.

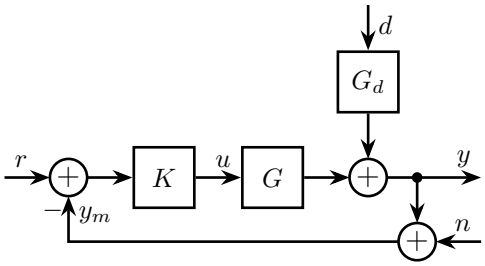


Figure 1 – *Configuration for one degree-of-freedom control*

b Closed-loop Transfer Functions

Closed-Loop Transfer Functions

$$y = Tr + SG_d d + Tn \quad (3a)$$
$$e = -Sr + SG_d d - Tn \quad (3b)$$
$$y = KSr - KSG_d d - KSn \quad (3c)$$

c Why Feedback?

We could think that we can use a "perfect" feedforward controller $K_r(s) = G^{-1}(s)$ with $r - G_d d$ as the controller input:

$$y = Gu + G_d d = GK_r(r - G_d d) + G_d d = r$$

Unfortunately, G is never an exact model and the disturbances are never known exactly.

Reasons for Feedback Control

- Signal uncertainty
- Unknown disturbance
- Model uncertainty
- An unstable plant

2.3 Closed Loop Stability

Two methods are commonly used to **determine closed-loop stability**:

1. The system is stable if and only if **all the closed-loop poles** (roots of $1 + L(s) = 0$) **are in the open LHP**. The poles are also equal to the **eigenvalues of the state-space A matrix** (this is how the poles are computed).
2. The frequency response of $L(j\omega)$ is plotted in the complex plane and the number of encirclement it makes around the critical point -1 is counted.
   - **Nyquist's stability criterion**: Closed-loop stability is inferred by equating the number of encirclement to the number of open-loop RHP-poles

- **Bode's stability condition**: The closed loop system is stable if and only if $|L(j\omega_{180})| < 1$ where ω_{180} is the phase crossover frequency defined by $\angle L(j\omega_{180}) = -180°$. This is only valid for open-loop stable systems where $\angle L(j\omega)$ falls with frequency and such that $\angle L(j\omega)$ crosses $-180°$ only once.

Method 1 is best suited for numerical calculation while method 2 has a nice graphical interpretation and may also be used for systems with time delays. Moreover, method 2 provides useful measure of relative stability and will be used for robustness test.

2.4 Evaluating Closed-Loop Performance

a Gain Margin

The **Gain Margin** is defined as:

$$\text{GM} = \frac{1}{|L(j\omega_{180})|} \tag{4}$$

with ω_{180} is the **phase crossover frequency** defined by $\angle L(j\omega_{180}) = -180°$. If there is more than one crossing ($\angle L(j\omega_{180}) = -180°$), the largest value of $|L(j\omega_{180})|$ is taken.

The GM is the factor by which the loop gain $|L(s)|$ may be increased before the closed-loop system becomes unstable.

b Phase Margin

The **Phase Margin** is defined as:

$$\text{PM} = \angle L(j\omega_c) + 180° \tag{5}$$

with ω_c the **gain crossover frequency** defined by $|L(j\omega_c)| = 1$.

The PM tells how much negative phase (phase lag) we can add to $L(s)$ at frequency ω_c before closed-loop instability appears.

Typically, we required the PM to be larger than 30°. This is a **safeguard against time delay uncertainty**, the system becomes unstable is we add a delay of $\theta_{max} = PM/\omega_c$.

Note that by decreasing the value of ω_c (lowering the closed-loop bandwidth) the system can tolerate larger time delays.

c Maximum Peak Criteria

Maximum peak criteria for S and T

$$M_S = \max_\omega |S(j\omega)| = \|S\|_\infty \tag{6a}$$

$$M_T = \max_\omega |T(j\omega)| = \|T\|_\infty \tag{6b}$$

Typically, we require $M_S < 2$ $(6dB)$ and $M_T < 1.25$ $(2dB)$.

Why do we want M_S small?

- Without feedback, with have $e = r - G_d d$ but with feedback $e = S(r - G_d d)$. Thus feedback improves performance in terms of reducing $|e|$ where $|S| < 1$. However, we cannot avoid having $|S| > 1$ at some intermediate frequency where feedback control degrades performance. The value of M_S is then a **measure of the worst-case performance degradation**
- M_S is also a **measure of the robustness** because the smallest distance between $L(\omega)$ and the critical point -1 is ${M_S}^{-1}$

There is a close **relationship between these maximum peaks and the gain and phase margins**. For a given value of M_S, we have:

$$\text{GM} \geq \frac{M_S}{M_S - 1}; \quad \text{PM} \geq \frac{1}{M_S} \tag{7}$$

Example of guaranteed stability margins:

- $M_S < 2 \Rightarrow GM > 2$ and $PM > 29°$
- $M_T < 2 \Rightarrow GM > 1.5$ and $PM > 29°$

d Bandwidth and Crossover Frequency

In general, a large bandwidth corresponds to a faster rise time, however, this also indicates an higher sensitivity to noise and to parameter variations.

Definition of bandwidth

The bandwidth, is the frequency range $[\omega_1, \omega_2]$ over which control is **effective**. In most case we simple call $\omega_2 = \omega_B$ the bandwidth.

As the word "effective" may be interpreted in different ways, there are **multiple definitions of bandwidth**:

- The **closed-loop bandwidth** ω_B is the frequency where $|S(j\omega)|$ first crosses $1/\sqrt{2} \approx -3dB$ from below.
- The **gain crossover frequency** ω_c is defined as the frequency where $|L(j\omega_c)|$ first crosses 1 from above
- The bandwidth in terms of T, ω_{BT}, is the highest frequency at which $|T(j\omega_c)|$ crosses $1/\sqrt{2} \approx -3dB$ from above.

For systems with $PM < 90°$, we have: $\omega_B < \omega_c < \omega_{BT}$
Then we have the following regions:

- $\omega < \omega_B$: $|S| < 0.7$ and control is effective
- $\omega_B < \omega < \omega_{BT}$: we may have $|S| > 1$ and control degrades performance
- $\omega_{BT} < \omega$: $|S| \approx 1$ and control has no significant effect on the response

The closed-loop time constant τ_{cl} can be related to the bandwidth:

$$\tau_{\text{cl}} \approx \frac{1}{\omega_b} \tag{8}$$

2.5 Controller Design

There is 3 mains approaches to controller design:

1. **Shaping of transfer functions**. The designer specifies the magnitude of some transfer functions as a function of frequency and then finds a controller which gives the desired shape(s)
   (a) Loop shaping of the open-loop transfer function $L(j\omega)$
   (b) Shaping of closed-loop transfer functions such as S, T and KS
2. **The signal based approach**. This involves time domain problem formulations resulting in the minimization of a norm of a transfer function. Linear Quadratic Gaussian (LQG) is an example of a signal based approach. A signal based $\mathcal{H}_\infty$ optimal control methodology can be derived.
3. **Numerical optimization**. This often involves multi-objective optimization where one attempts to optimize directly the true objectives such as rise times, stability margins, ... This problems may be difficult to solve, especially if one does not have convexity in the control parameters. This optimization may also be performed online.

2.6 Loop Shaping

a Trade-offs in Terms of L

Let's consider a feedback control system with error $e = -Sr + SG_d d - Tn$. If we want perfect control:

- For **disturbance rejection** and **command tracking**, we obtain $S \approx 0$, this implies that the loop transfer function L must be large in magnitude
- For zero **noise transmission**, we want $T \approx 0$ or equivalently $S \approx I$ which is obtained with $L \approx 0$.

This illustrate the **fundamental nature of feedback** design which always involves a **trade-off** between conflicting objectives.
The most important design objectives are:

Performance L large
Good dist. rejection L large
Limitation of meas. noise on plant output L small
Small magnitude of input signal K and L small
Strictly proper controller $K \to 0$ at high frequencies
Nominal stability L small (RHP zeros and time delays)
Robust stability L small (neglected dynamics)

Fortunately, the conflicting design objectives are generally in different frequency ranges, and we can meet most of the objectives by using large loop gain at low frequencies and a small gain at high frequencies above crossover.

b Fundamentals of Loop-Shaping Design

Definition - Loop Shaping

Design procedure that involves explicitly shaping the magnitude of the loop transfer function $|L(j\omega)|$.

To get the benefits of feedback control, we want the loop gain $|L(j\omega)|$ to be as large as possible within the bandwidth region. However, due to time delays, RHP-zeros, unmodelled high-frequency dynamics and limitations on the allowed manipulated inputs, the loop gain has to drop below one at and above the crossover frequency ω_c.

Logarithmic slope

To measure how $|L(j\omega)|$ falls with frequency, we consider the **logarithmic slope**:

$$N = \frac{d \ln |L|}{d \ln \omega} \tag{9}$$

The value of $-N$ at high frequencies is called the **roll-off rate**.

To get a high bandwidth (fast response) we want ω_c large (thus ω_{180} large), that is we want the phase lag in L to be small. Unfortunately, that is not consistent with the desire that $|L(j\omega)|$ should fall sharply (because of the approximation $\angle L \approx -N * 90°$).
The situation becomes even worse for cases with delays or RHP-zeros in $L(s)$ which add undesirable phase lag without contributing to a desirable negative slope.

We can define the **desired loop transfer function in terms of the following specifications**:

1. The gain crossover frequency ω_c, where $|L(j\omega_c)| = 1$
2. The shape of $|L(j\omega)|$:
   - Slope of $N = -1$ around crossover
   - Large roll-off at higher frequencies ($N > 2$)

- Slope at low frequencies depending on the nature of the disturbance or reference signal. We required a slope of -1 for step changes and -2 for ramp changes

3. The system type, defined as the number of pure integrators in $L(s)$

c Limitations Imposed by RHP-zeros and Time Delays

We usually want the loop shape to have a slope of -1 around crossover ω_c, then the phase lag of L at ω_c will be at least $-90°$. If we require a phase margin of $-35°$, then the additional phase contribution from delays and RHP zeros at ω_c cannot exceed $-55°$.
First consider a **time delay** θ which adds a phase of $-\theta\omega$. Thus, we want $\theta\omega_c < 55° \approx 1\,\text{rad}$. The attainable bandwidth is limited by the time delay:

$$\omega_c < 1/\theta \tag{10}$$

Next consider a **RHP-zero** at $s = z$. To avoid an increase in slope cause by the zero, we add a pole at $s = -z$, then L contains $\frac{-s+z}{s+z}$ which corresponds to an all-pass filter. The phase contribution is $\approx -55°$ at $\omega = z/2$. Thus, this limits the attainable bandwidth:

$$\omega_c < z/2 \tag{11}$$

d Inverse-Based Controller Design

The idea is to have $L(s) = \frac{\omega_c}{s}$ with ω_c the desired gain crossover frequency. The controller associated is then $K(s) = \frac{\omega_c}{s} G^{-1}(s)$ {the plant is inverted and an integrator is added}. This idea is the essential part of the **internal model control** (IMC). This loop shape yields a phase margin of 90° and an infinite gain margin.

They are many reasons why the inverse-based controller may **not** be a good choice:

- The controller will not be realizable if $G(s)$ has a pole excess of two or larger
- The loop shape is not generally desirable, unless the references and disturbances are steps

e Loop Shaping for Disturbance Rejection

We have $e = SG_d d$ with $|d(j\omega)| < 1$ at each frequency (thanks to scaling). The main control objective is to achieve $|e(j\omega)| < 1$. Then, we require: $|S(j\omega)G_d(j\omega)| < 1, \forall\omega$ or equivalently $|1 + L(j\omega)| > |G_d|\,, \forall\omega$.

Note that we don't want to have larger loop gain than necessary to not increase input signals and sensitivity to noise. A reasonable loop shape is then $|L| = |G_d|$.

The corresponding controller satisfies

$$|K| = \left|G^{-1}G_d\right| \tag{12}$$

This means that:

- For disturbances entering at the plant output ($G_d = 1$), we get $|K| = \left|G^{-1}\right|$
- For disturbances entering at the plant input ($G_d = G$), we get $|K| = 1$
- Note that reference change may be viewed as a disturbance directly affecting the output

The loop-shape $L(s)$ may be modify as follows:

- Around crossover, make the slope of $|L|$ to be about -1. This is to achieve good transient behavior with acceptable gain and phase margins
- Improve the low frequency performance by adding integral action $|K| = \left|\frac{s+\omega_I}{s}\right| \left|G^{-1}G_d\right|$
- Let $L(s)$ roll of faster at high frequencies in order to reduce the effect of noise and the input magnitude

f Two Degrees-of-freedom Design

For reference tracking, we typically want the controller to look like $\frac{1}{s}G^{-1}$, whereas for disturbance rejection we want the controller to look like $\frac{1}{s}G^{-1}G_d$.
We cannot achieve both of these simultaneously with a single feedback controller.
The solution is to use a **two degrees of freedom controller** where the reference signal r and output measurement y_m are independently treated by the controller (Fig. 2), rather than operating on their difference $r - y_m$.

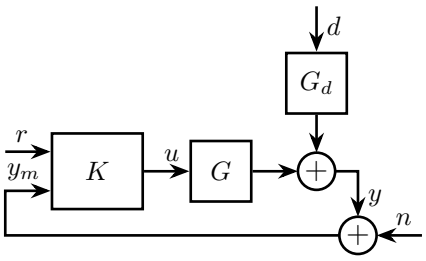


Figure 2 – *2 degrees-of-freedom control architecture*

The controller can be slit into two separate blocks (Fig. 3):

- the **feedback controller** K_y that is used to **reduce the effect of uncertainty** (disturbances and model errors)
- the **prefilter** K_r that **shapes the commands** r to improve tracking performance

It is optimal to design the combined two degrees of freedom controller K in one step, however, in practice K_y is often designed first for disturbance rejection, and then K_r is designed to improve reference tracking.

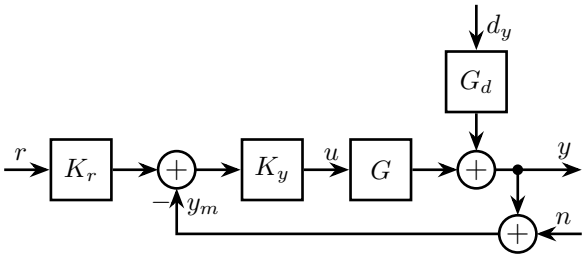


Figure 3 – *2 degrees-of-freedom control architecture with two separate blocs*

2.7 Shaping Closed-Loop Transfer Functions

Specifications on the open-loop transfer function $L = GK$ does not consider directly the closed-loop transfer functions, such as S and T which determine the final response. An alternative design strategy is to directly shape the magnitude of the closed loop transfer functions. This strategy can be formulated as an $\mathcal{H}_\infty$ optimal control problem.

a The Terms $\mathcal{H}_\infty$ and $\mathcal{H}_2$

The $\mathcal{H}_\infty$ norm of a stable scalar transfer function $f(s)$ is simply the peak value of $|f(j\omega)|$ as a function of frequency:

$$\|f(s)\|_\infty \triangleq \max_\omega |f(j\omega)| \tag{13}$$

Similarly, the symbol $\mathcal{H}_2$ stands for the Hardy space of transfer function with bounded 2-norm:

$$\|f(s)\|_2 \triangleq \left(\frac{1}{2\pi}\int_{-\infty}^{\infty} |f(j\omega)|^2\, d\omega\right)^{1/2} \tag{14}$$

b Weighted Sensitivity

The sensitivity function S is a very good indicator of closed-loop performance. The main advantage of considering S is that we want S small and **it is sufficient to consider just its magnitude** $|S|$.

Typical specifications in terms of S

- Minimum bandwidth frequency ω_B^*
- Maximum tracking error at selected freq.
- The maximum steady state tracking error A
- Shape of S over selected frequency ranges
- Maximum magnitude of S: $\|S(j\omega)\|_\infty \leq M$

The maximum peak specification prevents amplification of noise at high frequencies, and also introduces a margin of robustness. Typically, we select $M = 2$.

Mathematically, these specifications may be captured by an **upper bound** $1/|W_P(s)|$ on the magnitude of S where $W_P(s)$ is a **weight** selected by the designer. The subscript P stands for **performance** since S is mainly used as a performance indicator.
The performance requirement becomes

$$S(j\omega) < 1/|W_P(j\omega)|, \forall\omega$$

Which can be expressed as an $\mathcal{H}_\infty$:

$$\|W_P S\|_\infty < 1 \tag{15}$$

Typical performance weight

$$W_P(s) = \frac{s/M + \omega_B^*}{s + \omega_B^* A}$$

With (see Fig. 4):

- M: maximum magnitude of $|S|$
- ω_B: crossover frequency
- A: steady-state offset

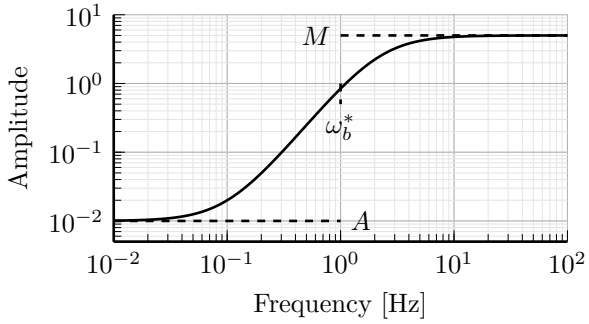


Figure 4 – *Inverse of performance weight*

If we want a steeper slope for L below the bandwidth, an higher order weight may be selected. A weight which ask for a slope of -2 for L below crossover is:

$$W_P(s) = \frac{(s/M^{1/2} + \omega_B^*)^2}{(s + \omega_B^* A^{1/2})^2}$$

c Stacked Requirements: Mixed Sensitivity

The specification $\|W_P S\|_\infty < 1$ puts a lower bound on the bandwidth, but not an upper one and nor does it allow us to specify the roll-off of $L(s)$ above the bandwidth.
To do this, we can make demands on another closed-loop transfer function T by specifying an upper bound $1/|W_T|$ on the magnitude $|T|$ to **make sure that** L **rolls off sufficiently fast at high frequencies**.
Also, to achieve robustness or to restrict the magnitude of the input signal u, one may place an upper bound $1/|W_U|$ on the magnitude KS.

To combined these **mixed sensitivity specifications**, a **stacking approach** is usually used, resulting in the following overall specification:

$$\max_\omega \overline{\sigma}(N(j\omega)) < 1; \quad N = \begin{bmatrix} W_P S \\ W_T T \\ W_U K S \end{bmatrix}$$

After selecting the form of N and the weights, the $\mathcal{H}_\infty$ optimal controller is obtained by solving the problem $\min_K \|N(K)\|_\infty$.

3 Introduction to Multivariable Control

3.1 Introduction

The main difference between a SISO system and a MIMO system is the presence of **directions** in the latter.
However, most of the ideas and techniques used for SISO systems may be extended to MIMO systems. This is done by considering the **maximum singular value** instead of the absolute value.
The **singular value decomposition** (SVD) provides a useful way of quantifying multivariable directionality. For MIMO systems the gain $\frac{|Gd|}{|d|}$ (where $|\cdot|$ is some norm) is independent of the magnitude $|d|$ (like for SISO systems), but it does **depend on its direction**. A plant is said to be **ill-conditioned** if the gain depends strongly on the input direction. It is quantified by the **condition number** Γ (which is much larger than 1 for an ill-conditioned plant).
For MIMO systems the order of the transfer functions matter, so in general:

$$GK \neq KG \tag{16}$$

even when G and K are square matrices.

3.2 Transfer Functions

MIMO Rule

The main rule for evaluating transfer functions is the **MIMO Rule**: Start from the output and write down the transfer functions as you meet them going to the input. If you exit a feedback loop then we get a term $(I-L)^{-1}$ where $L = GK$ is the transfer function around the loop (gain going backwards).

a Negative Feedback Control Systems

For negative feedback system (Fig. 5), we define L to be the loop transfer function as seen when breaking the loop at the **output** of the plant:

- $L = GK$
- $S \triangleq (I+L)^{-1}$ is the transfer function from d_1 to y
- $T \triangleq L(I+L)^{-1}$ is the transfer function from r to y

We define L_1 to be the loop transfer function as seen when breaking the loop at the **input** to the plant:

- $L_1 = KG$
- $S_1 \triangleq (I+L_1)^{-1}$
- $T_1 \triangleq L_1(I+L_1)^{-1}$ is the transfer function from d_2 to $-u$

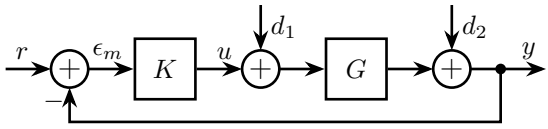


Figure 5 – *Conventional negative feedback control system*

3.3 Multivariable Frequency Response Analysis

a Obtaining the Frequency Response from $G(s)$

Consider the system $G(s)$ with input $d(s)$ and output $y(s)$. The element $g_{ij}(j\omega)$ of the matrix G represents the sinusoidal response from the input j to output i.

b Directions in Multivariable Systems

For a SISO system, the gain at ω is simply:

$$\frac{|y(\omega)|}{|d(\omega)|} = \frac{|G(j\omega)d(\omega)|}{|d(\omega)|} = |G(j\omega)| \tag{17}$$

The gain depends on the frequency ω but it is independent of the input magnitude $|d(\omega)|$.
For MIMO systems, we have to use norms to measure the amplitude of the inputs/outputs. If we select vector 2-norm, the magnitude of the vector input signal is:

$$\|d(\omega)\|_2 = \sqrt{\sum_j |d_j(\omega)|^2}$$

The gain of the system is then:

$$\frac{\|y(\omega)\|_2}{\|d(\omega)\|_2} = \frac{\|G(j\omega)d(\omega)\|_2}{\|d(\omega)\|_2} = \frac{\sqrt{\sum_j |y_j(\omega)|^2}}{\sqrt{\sum_j |d_j(\omega)|^2}} \tag{18}$$

Again the gain depends on the frequency ω and again it is independent of the input magnitude $\|d(\omega)\|_2$. However, the gain depends also on the **direction** of the input d.

c Eigenvalues as a Poor Measure of Gain

The magnitudes of the eigenvalues of a transfer function matrix $|\lambda_i(G(j\omega))|$ do not provide a useful means of generalizing the SISO gain. The main problem is that the eigenvalues measure the gain for the special case when the **inputs and the outputs are in the same direction**, namely in the direction of the eigenvectors.

d Singular Value Decomposition

We are interested by the physical interpretation of the SVD when applied to the frequency response of a MIMO system $G(s)$ with m inputs and l outputs.

Singular Value Decomposition

$$G = U\Sigma V^H \tag{19}$$

- Σ is an $l \times m$ matrix with $k = \min\{l, m\}$ non-negative **singular values** σ_i, arranged in descending order along its main diagonal, the other entries are zero.
- U is an $l \times l$ unitary matrix. The columns of U, denoted u_i, represent the **output directions** of the plant. They are orthonormal.
- V is an $m \times m$ unitary matrix. The columns of V, denoted v_i, represent the **input directions** of the plant. They are orthonormal.

The input and output directions are related through the singular values:

$$Gv_i = \sigma_i u_i \tag{20}$$

So, if we consider an input in the direction v_i, then the output is in the direction u_i. Furthermore, since $\|v_i\|_2 = 1$ and $\|u_i\|_2 = 1$, we see that **the singular value σ_i directly gives the gain of the matrix G in this direction**.

The **largest gain** for any input is equal to the **maximum singular value**:

$$\overline{\sigma}(G) \equiv \sigma_1(G) = \max_{d \neq 0} \frac{\|Gd\|_2}{\|d\|_2} = \frac{\|Gv_1\|_2}{\|v_1\|_2}$$

The **smallest gain** for any input direction is equal to the **minimum singular value**:

$$\underline{\sigma}(G) \equiv \sigma_k(G) = \min_{d \neq 0} \frac{\|Gd\|_2}{\|d\|_2} = \frac{\|Gv_k\|_2}{\|v_k\|_2}$$

We define $u_1 = \bar{u}$, $v_1 = \bar{v}$, $u_k = \underline{u}$ and $v_k = \underline{v}$. Then is follows that:

$$G\bar{v} = \overline{\sigma}\bar{u}; \quad G\underline{v} = \underline{\sigma}\underline{u}$$

e Non Square Plants

If the plant has more output than inputs, the outputs singular vectors u_i with $i > k$ correspond to the outputs directions that cannot be controlled.
Similarly, for a plant with more inputs and outputs, the additional input singular vectors tells us in which directions the input will have no effect.

f Singular Values for Performance

The gain of the MIMO system from the vector of reference inputs r and the vector of control error e is bounded by the minimum and maximum singular values of S:

$$\underline{\sigma}(S(j\omega)) < \frac{\|e(\omega)\|_2}{\|r(\omega)\|_2} < \overline{\sigma}(S(j\omega))$$

In terms of performance, we require that the gain remains small for any direction of $r(\omega)$ including the "worst-case" direction corresponding to the gain $\overline{\sigma}(S(j\omega))$. Let $1/|W_P(j\omega)|$ represent the maximum allowed magnitude of $\frac{\|e(\omega)\|_2}{\|r(\omega)\|_2}$ at each frequency:

$$\overline{\sigma}(S(j\omega)) < \frac{1}{|W_P|}, \forall\omega \Leftrightarrow \|W_P S\|_\infty < 1$$

$\mathcal{H}_\infty$ norm - MIMO Case - Definiton

The $\mathcal{H}_\infty$ norm is defined as the peak of the maximum singular value of the frequency response:

$$\|M(s)\|_\infty \triangleq \max_\omega \overline{\sigma}(M(j\omega)) \tag{21}$$

For MIMO systems **the bandwidth depends on direction**. If we want to associate a single bandwidth frequency for a multivariable system, then we consider the worst-case direction, and define the bandwidth ω_B as the frequency where $\overline{\sigma}(S)$ crosses $\frac{1}{\sqrt{2}} = 0.7$ from below.

3.4 Control of Multivariable Plants

A conceptually simple approach to multivariable control is given by a two-step procedure:

1. **Design a pre-compensator** W_1, which counteracts the interactions in the plant and results in a new **shaped plant** $G_S(s) = G(s)W_1(s)$ which is **more diagonal and easier to control** than the original plant $G(s)$.
2. **Design a diagonal controller** $K_S(s)$ for the shaped plant using methods similar to those for SISO systems.

The overall controller is then:

$$K(s) = W_1(s)K_s(s)$$

a Decoupling

There are mainly three different cases:

1. **Dynamic decoupling**: $G_S(s)$ is diagonal at all frequencies. For that we can choose $W_1(s) = G^{-1}(s)$ and this is an inverse-based controller.
2. **Steady-state decoupling**: $G_S(0)$ is diagonal. This can be obtained by selecting $W_1(s) = G^{-1}(0)$.
3. **Approximate decoupling at frequency** ω_0: $G_S(j\omega_0)$ is as diagonal as possible. Decoupling the system at ω_0 is a good choice because the effect on performance of reducing interaction is normally greatest at this frequency.

The idea of decoupling control is appealing, but there are **several difficulties**:

1. It is very sensitive to modelling errors

2. It may not be required for disturbance rejection
3. If the plant has RHP-zero, the decoupling generally introduces extra RHP-zero in the closed-loop system

b SVD-Controller

We can also introduce a **post compensator** $W_2(s)$. The shaped plant is then:

$$G_S(s) = W_2(s)G(s)W_1(s)$$

A diagonal controller K_S can then be designed for the shaped plant. The overall controller is then:

$$K(s) = W_1(s)K_S(s)W_2(s)$$

The **SVD-controller** is a special case of a pre and post compensator design: $W_1 = V_0$ and $W_2 = U_0^T$. V_0 and U_0 are obtained from a SVD of $G_0 = U_0\Sigma_0 V_0^T$ where G_0 is a real approximation of $G(j\omega_0)$.

c Decentralized Control

Another approach is to use a diagonal or block-diagonal controller $K(s)$. This works well if $G(s)$ is close to diagonal, because then the plant to be controlled is essentially a collection of independent sub-plants, and each element in $K(s)$ may be designed independently. However, if off-diagonal elements in $G(s)$ are large, the performance with decentralized diagonal control may be poor because no attempt is made to counteract the interactions.

d What is the Shape of the "best" Feedback Controller?

Consider the problem of disturbance rejection: $y = SG_d d$ where $\|d\|_2 < 1$ and our performance requirement is that $\|y\|_2 < 1$ which is equivalent to requiring $\overline{\sigma}(SG_d) < 1$.
However there is generally a trade-off between input usage and performance. The controller that minimize the input magnitude while meeting the performance requirement is the one that yields all singular values of SG_d equal to 1, i.e. $\sigma_i(SG_d) = 1, \forall\omega$. This corresponds to:

$$S_{\min}G_d = U_1$$

Where U_1 is some all-pass transfer function (which at each frequency has all its singular values equal to 1).
At frequencies where feedback is effective, we have $S \approx L^{-1}$ and then $L_{\min} = GK_{\min} \approx G_d U_1^{-1}$. In conclusion, the controller and loop shape with the minimum gain will often look like:

$$K_{\min} \approx G^{-1}G_d U_2$$

where $U_2 = U_1^{-1}$ is some all-pass transfer function matrix.
We see that for disturbances entering at the plant inputs, $G_d = G$, we get $G_{\min} = U_2$, so a simple constant unit gain controller yields a good trade-off between output performance and input usage.

e Summary of Mixed-Sensitivity $\mathcal{H}_\infty$ Synthesis

In the mixed-sensitivity S/KS problem, the objective is to minimize the $\mathcal{H}_\infty$ norm of:

$$N = \begin{bmatrix} W_P S \\ W_U KS \end{bmatrix} \tag{22}$$

Here are some guidelines for the choice of the weights W_P and W_U:

- KS is the transfer function from r to u, so for a system which has been scaled, a reasonable initial choice for the input weight is $W_U = I$
- S is the transfer function from r to $-e = r - y$. A common choice for the performance weight is $W_P = \text{diag}\{w_{p_i}\}$ with:

$$w_{p_i} = \frac{s/M_i + \omega_{B_i}^*}{s + \omega_{B_i}^* A_i}, \quad A_i \ll 1$$

  Selecting $A_i \ll 1$ ensures approximate integral action. Often we select M_i about 2 for all outputs, whereas $\omega_{B_i}^*$ may be different for each output.

For disturbance rejection, we may in some cases want a steeper slope for $w_{P_i}(s)$ at low frequencies. However it may be better to **consider the disturbances explicitly** by considering the $\mathcal{H}_\infty$ norm of:

$$N = \begin{bmatrix} W_P S & W_P S G_d \\ W_U KS & W_U KSG_d \end{bmatrix} \tag{23}$$

We can also considerate T which is the transfer function from $-n$ to y. To reduce the sensitivity to noise and uncertainty, we want T small at high frequencies, and so we may want **additional roll-off** in L. This can be achieved in several ways:

- One approach is to add $W_T T$ to the stack for N where $W_T = \text{diag}\{w_{T_i}\}$ and $|w_{T_i}|$ is smaller than 1 at low frequencies and large at high frequencies
- A more direct approach is to **add high-frequency dynamics $W_1(s)$ to the plant model** to ensure that the resulting shaped plant, $G_S = GW_1$ rolls off with the desired slope. We then obtain an $\mathcal{H}_\infty$ optimal controller K_S for this shaped plant, and finally include $W_1(s)$ in the controller $K = W_1 K_S$

3.5 Introduction to MIMO RHP-Zeros

Whereas the poles p of MIMO system G are essentially poles of elements of G, the zeros are generally not the zeros of elements of G. However, for square MIMO plants, the poles and zeros are in most cases the poles and zeros of $\det G(s)$.

Definition of zeros for MIMO sytems

The zeros z of a MIMO system G are defined as the values $s = z$ where $G(s)$ loses rank.

As for SISO systems, we find that **RHP-zeros impose fundamental limitations on control**. Poles and zeros of MIMO systems have **directions**:

- We can find the **direction of a zero** by looking at the direction in which the matrix $G(z)$ has zero gain
- Pole direction is direction where $G(p)$ is infinite

It is generally possible to move the effect of RHP-zero to particular outputs. If it is not, the zero is called a "**pinned zero**".

3.6 Condition Number and RGA

a Condition Number

Condition Number - Definition

We define the condition number of a matrix as the ratio between its maximum and minimum singular values:

$$\gamma(G) \triangleq \overline{\sigma}(G)/\underline{\sigma}(G) \tag{24}$$

A matrix with large condition number is said to be **ill-conditioned**.
For a non-singular square matrix $\underline{\sigma}(G) = 1/\overline{\sigma}(G^{-1})$, so $\gamma(G) = \overline{\sigma}(G)\overline{\sigma}(G^{-1})$. It then follows that the condition number is large if the product of the largest element in G and G^{-1} is large.
Note that the condition number depends strongly on scaling. One might consider minimizing the condition number over all possible scalings. This results in the **minimized or optimal condition number** which is defined by:

$$\gamma^*(G) = \min_{D_1, D_2} \gamma(D_1 G D_2) \tag{25}$$

If the condition number is small, then the multivariable effects of uncertainty are not likely to be serious. However if the condition number is large (say, larger than 10), then this may indicate control problems.

b Relative Gain Array (RGA)

Relative Gain Array - Definition

The relative gain array (RGA) for a non-singular square matrix G is a square matrix defined as:

$$\text{RGA}(G) = \Lambda(G) \triangleq G \times G^{-T} \tag{26}$$

where $\times$ is element-by-element multiplication

In most case, it is the value of the RGA at frequencies close to crossover which is most important.
The RGA has interesting algebraic properties:

- It is independent of input and output scaling
- Its rows and columns sum to one
- The sum-norm of the RGA $\|\Lambda\|_{\text{sum}}$ is close to the minimized condition number γ^*. Plants with large RGA-elements are thus always ill-conditioned
- The RGA is the identity matrix if G is upper of lower triangular. This follows that $\Gamma - I$ provides a **measure of two-way interactions**

It has also a number of useful **control properties**:

- Plants with large RGA-elements around the crossover frequency are fundamentally difficult to control because of sensitivity to input uncertainty
- If the sign of a RGA-element changes from $s = 0$ to $s = \infty$, then there is a RHP-zero in G
- The definition of the RGA may be generalized to non-square matrices by using the pseudo inverse
- The **RGA-number** can be used as a measure of diagonal dominance: $\|\Lambda(G) - I\|_{\text{sum}}$
- For decentralized control, we prefer pairing input and outputs for which the RGA-number at crossover frequencies is close to 0

3.7 Introduction to Robustness for MIMO Plants

Multivariable plants can show a sensitivity to uncertainty which is fundamentally different from what is possible in SISO systems. It is possible to have excellent stability margins (GM and PM) when considering one loop at a time, but small simultaneous input gain errors can give instability.
For SISO systems, we generally have that nominal performance and robust stability imply robust performance, but this is not the case for MIMO systems.
Although we have **useful indicators of robustness problems** (RGA-number, Sensitivity Peaks, etc), they provide no exact answer to whether a given source of uncertainty will yield instability or poor performance. The **structured singular value** μ is a tool for analyzing the effects of model uncertainty.

3.8 General Control Problem Formulation

The general control problem formulation is represented in Fig. 6.

Control Design Problem

Find a controller K which based on the information in v, generates a control signal u which counteracts the influence of w on z, thereby minimizing the closed-loop norm from w to z.

a Obtaining the Generalized Plant P

We must first find a block diagram representation of the system and identify the signals w, z, u and v. Then

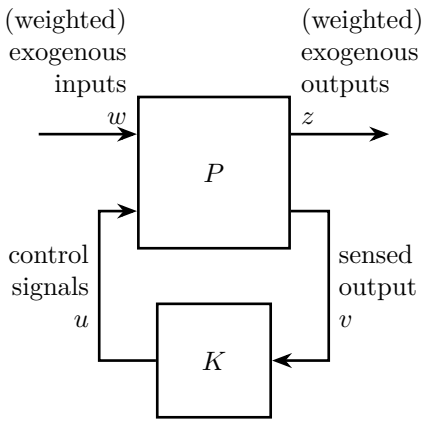


Figure 6 – *General control configuration*

we have to break all the "loops" entering and exiting the controller K to obtain P such that:

$$\begin{bmatrix} z \\ v \end{bmatrix} = P \begin{bmatrix} w \\ u \end{bmatrix} \tag{27}$$

b Controller Design: Including Weights in P

In order to get a meaningful controller synthesis problem, for example in terms of the $\mathcal{H}_\infty$ norms, we generally have to include the weights W_z and W_w in the generalized plant P (Fig. 7). We consider:

- The weighted or normalized exogenous inputs w (where $\tilde{w} = W_w w$ consists of the "physical" signals entering the system)
- The weighted or normalized controlled outputs $z = W_z \tilde{z}$ (where $\tilde{z}$ often consists of the control error $y - r$ and the manipulated input u)

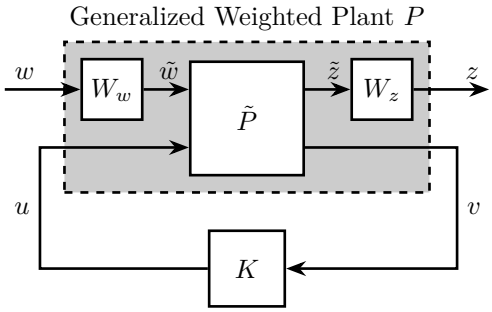


Figure 7 – *General Weighted Plant*

The weighted matrices are usually frequency dependent and typically selected such that weighted signals w and z are of magnitude 1.

c Partitioning the Generalized Plant P

We often partition P as:

$$\begin{bmatrix} z \\ v \end{bmatrix} = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix} \tag{28}$$

P_{22} has dimensions compatible with the controller.

d Analysis: Closing the Loop the get N

In the previous representations, the controller K has a separate block. This is useful when **synthesizing** the controller. However, for **analysis** of closed-loop performance the controller is given, and we may absorb K into the interconnection structure and obtain the system N.

Closed-loop transfer function N

$$z = Nw \tag{29}$$

N is given by:

$$N = P_{11} + P_{12}K(I - P_{22}K)^{-1}P_{12} \triangleq F_l(P, K)$$

where $F_l(P, K)$ denotes a **lower linear fractional transformation** (LFT).

e A General Control Configuration Including Model Uncertainty

The general control configuration may be extended to include model uncertainty as shown in Fig. 8.

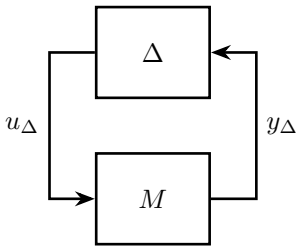


Figure 8 – *General control configuration for the case with model uncertainty*

The matrix Δ is a block-diagonal matrix that includes all possible perturbations (representing uncertainty). It is usually normalized in such a way that $\|\Delta\|_\infty \leq 1$.

3.9 Conclusion

Conclusion on MIMO Control

The **Singular Value Decomposition** (SVD) of the plant transfer function matrix provides insight into **multivariable directionality**.

Other useful tools for analyzing directionality and interactions are the **condition number** and the **Relative Gain Array** (RGA).

Closed loop performance may be analyzed in the frequency domain by evaluating the **maximum singular value of the sensitivity function** as the function of frequency.

Multivariable RHP-zeros impose fundamental limitations on performance, but for MIMO systems we can often direct the undesired effect of a RHP-zero to a subset of the outputs.

MIMO systems are often **more sensitive to uncertainty** than SISO systems.

4 Elements of Linear System Theory

4.1 System Descriptions

For linear systems there are several alternative system representations:

- **state-space representation** often follows directly from a physical model, and is used in most **numerical calculations**.
- **transfer function representation** is a nice compact representation which yields invaluable insights; it allows for series connections to be represented by multiplication of transfer functions. It also leads directly to the frequency response by setting $s = j\omega$.
- **coprime factorization** is a factorization into two stable systems, and that it is useful for representing the class of all stabilizing controllers. It forms the basis for the very useful coprime uncertainty description.

a State-Space Representation

A natural way to represent many physical systems is by nonlinear state-space models of the form

$$\dot{x} \triangleq \frac{dx}{dt} = f(x,u); \quad y = g(x,u)$$

Linear state-space models may then be derived from the linearization of such models.

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t) + Du(t)$$

where A, B, C and D are real matrices.
These equations may be rewritten as

$$\begin{bmatrix} \dot{x} \\ y \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix}$$

which gives rise to the short-hand notation

$$G = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right]$$

The state-space representation of a system is not unique, there exist realizations with the same input-output behavior, but with additional unobservable and/or uncontrollable state.

Minimum Realization

A minimal realization is a realization with the **fewest number of states** and consequently **no unobservable or uncontrollable modes**.

The state-space representation yields an internal description of the system which may be useful if the model is derived from physical principles. It is also more suitable for numerical calculations.

b Impulse Response Representation

The impulse response matrix is

$$g(t) = \begin{cases} 0 & t < 0 \\ Ce^{At}B + D\delta(t) & t \geq 0 \end{cases}$$

The ij'th element of the impulse response matrix, $g_{ij}(t)$, represents the response $y_i(t)$ to an impulse $u_j(t) = \delta(t)$ for a systems with a zero initial state.
With initial state $x(0) = 0$, the dynamic response to an arbitrary input $u(t)$ is

$$y(t) = g(t) * u(t) = \int_0^t g(t-\tau)u(\tau)d\tau$$

c Transfer Function Representation - Laplace Transforms

The transfer function representation is unique and is defined as the Laplace transform of the impulse response.

Laplace transform

$$G(s) = \int_0^\infty g(t)e^{-st}dt$$

We can also obtain the transfer function representation from the state-space representation by taking the Laplace transform of the state-space equations

$$sx(s) = Ax(s) + Bu(s) \;\Rightarrow\; x(s) = (sI - A)^{-1}Bu(s)$$

$$y(s) = Cx(s) + Du(s) \;\Rightarrow\; y(s) = \underbrace{\left(C(sI - A)^{-1}B + D\right)}_{G(s)} u(s)$$

Time delays and improper systems can be represented by Laplace transforms, but do not have a state-space representation.

d Coprime Factorization

Right coprime factorization of G

$$G(s) = N_r(s)M_r^{-1}(s)$$

where $N_r(s)$ and $M_r(s)$ are stable coprime transfer functions.

The stability implies that $N_r(s)$ should contains all the RHP-zeros of $G(s)$, and $M_r(s)$ should contain as RHP-zeros all the RHP-poles of $G(s)$. Mathematically, coprimeness means that there exist stable $U_r(s)$ and $V_r(s)$ such that the Bezout identity is satisfied: $U_rN_r + V_rM_r = I$

4.2 State Controllability and State Observability

There are **many ways to check for state controllability and observability**, e.g. with Gramians, input/output pole vectors, controllability/observability matrix, etc.

Input and output pole vectors The method which yields the most insight is probably to compute the input and output directions associated with each pole (mode). For the case when A has distinct eigenvalues, we have the following dyadic expansion of the transfer function matrix from inputs to outputs

$$G(s) = \sum_{i=1}^{n} \frac{Ct_i q_i^H B}{s - \lambda_i} + D = \sum_{i=1}^{n} \frac{y_{p_i} u_{p_i}}{s - \lambda_i} + D$$

- The i'th **input pole vector** $u_{p_i} \triangleq q_i^H B$ is an indication of how much the i'th mode is excited (and thus may be "controlled") by the inputs.
- The i'th **output pole vector** $y_{p_i} \triangleq Ct_i$ indicates how much the i'th mode is observed in the outputs.

State Controllability Let λ_i be the i^{th} eigenvalue of A, q_i the corresponding left eigenvector ($q_i^H A = \lambda_i q_i^H$), and $u_{p_i} = B^H q_i$ the i^{th} input pole vector. Then the system (A, B) is state controllable if and only if

$$u_{p_i} \neq 0, \forall i$$

That is if and only if all its input pole vectors are nonzero.

State Observability Let λ_i be the i^{th} eigenvalue of A, t_i the corresponding right eigenvector ($At_i = \lambda_i t_i$), and $y_{p_i} = Ct_i$ the i^{th} output pole vector. Then the system (A, C) is state observable if and only if

$$y_{p_i} \neq 0, \forall i$$

That is if and only if all its output pole vectors are nonzero.

Minimal realization A state space realization (A, B, C, D) of $G(s)$ is said to be a minimal realization of $G(s)$ if A has the smallest possible dimension. The smallest dimension is called the **McMillan degree** of $G(s)$. A mode is hidden if it is not state controllable or observable and thus does not appear in the minimal realization. It follows that a state-space realization is minimal if and only if (A, B) is state controllable and (A, C) is state observable.

4.3 Stability

Internal Stability

A system is (internally) stable is none of its components contain hidden unstable modes and the injection of bounded external signals at any place in the system result in bounded output signals measured anywhere in the system.

Stability - Detectability

A system is (state) **stabilizable** if all unstable modes are state controllable. A system is (state) **detectable** if all unstable modes are state observable.
A system with unstabilizable or undetectable modes is said to contain hidden unstable modes.

4.4 Poles

Definition: Multivariable Pole

The poles p_i of a system with state-space description are the **eigenvalues** $\lambda_i(A), i = 1, \ldots, n$ of the matrix A. The **pole or characteristic polynomial** $\phi(s)$ is defined as $\phi(s) \triangleq \det(sI - A) = \Pi_{i=1}^{n}(s - p_i)$. Thus the poles are the roots or the characteristic equation

$$\phi(s) \triangleq \det(sI - A) = 0$$

a Poles and Stability

A linear dynamic system is **stable if and only if all the poles are in the LHP**, that is, $\text{Re}\{\lambda_i(A)\} < 0, \forall i$

b Poles from Transfer Functions

The pole polynomial $\phi(s)$ corresponding to a minimal realization of a system with transfer function $G(s)$ is the **least common denominator** of all non-identically-zero minors of all orders of $G(s)$.
The poles are essentially the sum of the poles in the elements of the transfer function, but to get the correct multiplicity a more careful analysis is needed.

c Pole Vectors and Directions

In multivariable system poles have **directions** associated with them. To quantify this, we use the **input and output pole vectors**.

Input pole vector

$$u_{p_i} = B^H q_i$$

With q_i the left eigenvector of A (${q_i}^T A = \lambda_i {q_i}^T$).
The input pole direction is $\frac{1}{\|u_{p_i}\|_2} u_{p_i}$

Output pole vector

$$y_{p_i} = C t_i$$

With t_i the right eigenvector of A ($A t_i = \lambda_i t_i$).
The output pole direction is $\frac{1}{\|y_{p_i}\|_2} y_{p_i}$

The pole directions may be defined in terms of the transfer function matrix by evaluating $G(s)$ at the pole p_i and considering the directions of the resulting complex matrix $G(p_i)$. The matrix is infinite in the direction of the pole, and we may write

$$G(p_i) u_{p_i} = \infty \cdot y_{p_i}$$

where u_{p_i} is the input pole direction and y_{p_i} is the output pole direction.
The pole directions may in principle be obtained from an SVD of $G(p_i) = U \Sigma V^H$. Then u_{p_i} is the first column in V (corresponding to the maximum singular value) and y_{p_i} the first column in U.
The pole direction is usually very interesting because it gives information about which output (or combination of outputs) may be difficult to control.

4.5 Zeros

Zeros of a system arise when competing effects, internal to the system, are such that the output is zero even when the inputs (and the states) are not themselves identically zero.

Definition: Multivariable Zero

z_i is a zero of $G(s)$ if the rank of $G(z_i)$ is less than the normal rank of $G(s)$. The zero polynomial is defined as $z(s) = \Pi_{i=1}^{n_z}(s - z_i)$ where n_z is the number of finite zeros of $G(s)$

a Zeros from State-Space Realizations

The state-space equations of a system may be written as

$$P(s) \begin{bmatrix} x \\ u \end{bmatrix} = \begin{bmatrix} 0 \\ y \end{bmatrix}, \quad P(s) = \begin{bmatrix} sI - A & -B \\ C & D \end{bmatrix}$$

The zeros are then the values $s = z$ for which the polynomial system matrix, $P(s)$, loses rank, resulting in zero output for some non-zero input.

b Zeros from Transfer Functions

The zero polynomial $z(s)$, corresponding to a minimal realization of the system, is the greatest divisor of all the numerator of all order-r minors of $G(s)$, where r is the normal rank of $G(s)$, provided that these minors have been adjusted in such a way as to have the pole polynomial $\phi(s)$ as their denominator.
The zeros are values of s for which $G(s)$ looses rank. In general, there is no relationship between the elements of the transfer function and its (multivariable) zeros.

c Zero Directions

Let $G(s)$ have a zero at $s = z$. Then $G(s)$ loses rank at $s = z$, and there will exist non-zero vectors u_z and y_z such that

$$G(z) u_z = 0 \cdot y_z$$

Here u_z is defined as the **input zero direction** and y_z is defined as the **output zero direction**.
From a practical point of view, y_z is usually of more interest than u_z because it give information about **which combination of outputs may be difficult to control**.
Again, we may obtain input and output zero directions from an SVD of $G(s)$: u_z is the last column of U and y_z is the last column of V (corresponding to the zero singular value of $G(z)$).

4.6 Some Remarks on Poles and Zeros

- We should always find a **minimal realization** of the system **before computing the zeros**.
- For a square system $G(s)$, the poles and zeros are *essentially* the poles and zeros of $\det G(s)$.
- Poles and zeros can occurs at the same location, but their directions may be different so they do not cancel or otherwise interact with each other.
- If $G^{-1}(s)$ exists, then the poles of $G(s)$ are the zeros of $G^{-1}(s)$ and vice versa (as for SISO systems).
- Zeros usually appear when there are fewer inputs or outputs than states or when $D \neq 0$
- **Moving poles and zeros**:
  - **Feedback**: $G(I + GK)^{-1}$. Poles (of G) are moved and zeros (of G) are unchanged (in addition we get as zeros the poles of K)
  - **Series**: GK. Poles and zeros are unchanged (with the exception of possible cancellations between poles and zeros in G and K)
  - **Parallel**: $G + K$. Poles are unchanged, zeros are moved (but note that physically a parallel interconnection requires an additional manipulated input)
- **Pinned zeros**. A zero is pinned to a subset of the outputs if y_z has one or more elements equal to zero. Their effect cannot be moved freely to any output. Similarly, a zero is pinned to certain input if u_z has one or more elements equal to zero.

Effect of feedback on poles and zeros

Consider a SISO negative feedback system with plant $G(s) = \frac{z(s)}{\phi(s)}$ and a constant gain controller, $K(s) = k$. The closed-loop response from reference r to output y is

$$T(s) = \frac{kG(s)}{1 + kG(s)} = \frac{kz(s)}{\phi(s) + kz(s)} = k\frac{z_{\text{cl}}(s)}{\phi_{\text{cl}}(s)}$$

We note that:

- The zero locations are unchanged by feedback
- The pole locations are changed by feedback

$$\phi_{\text{cl}(s)} \underset{k\to 0}{\longrightarrow} \phi(s)$$
$$\phi_{\text{cl}(s)} \underset{k\to \infty}{\longrightarrow} kz(s)$$

That is, as we increase the feedback gain, **the closed loop poles moves from open-loop poles to the open-loop zeros**. RHP-zeros therefore imply high gain instability.

4.7 Internal Stability of Feedback Systems

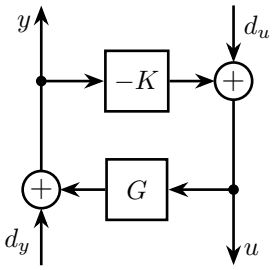


Figure 9 – *Block diagram used to check internal stability*

Assume that the components G and K contain no unstable hidden modes. Then the feedback system in Fig. 9 is **internally stable** if and only if all four closed-loop transfer matrices are stable.

$$\begin{matrix}(I + KG)^{-1} & -K(I + GK)^{-1} \\ G(I + KG)^{-1} & (I + GK)^{-1}\end{matrix}$$

Assume there are no RHP pole-zero cancellations between $G(s)$ and $K(s)$, the feedback system in Fig. 9 is internally stable if and only if **one** of the four closed-loop transfer function matrices is stable.

4.8 Stabilizing Controllers

The **Q-parameterization** is a parameterization that generates all controllers that yield internal stability of the closed loop transfer function.

Q-parameterization for stable plant

For stable plants, a parameterization of all stabilizing negative feedback controllers for the stable plant $G(s)$ is given by

$$K = (I - QG)^{-1}Q = Q(I - GQ)^{-1}$$

where the parameter Q is any stable transfer function matrix.

This may have significant advantages in controller synthesis where the objective is to a find a K which minimizes some norm of $N(K)$. The search over stabilizing K (which involves checking the stability of closed-loop transfer functions) is replaced by a search over stable Q. The closed-loop transfer functions turn out to be affine in Q, e.g. S or T can be written $H1 + H2QH3$, which may significantly simplify the optimization (e.g. compared to $GK(I + GK)^{-1}$ which is fractional in K).

4.9 Stability Analysis in the Frequency Domain

Generalized (MIMO) Nyquist theorem

Let P_{ol} denote the number of unstable poles in $L(s) = G(s)K(s)$. The closed-loop system with loop transfer $L(s)$ and negative feedback is stable if and only if the Nyquist plot of $\det(I + L(s))$:

1. makes P_{ol} anti-clockwise encirclements of the origin
2. does not pass through the origin

Spectral radius

The spectral radius $\rho(L(j\omega))$ is defined as the maximum eigenvalue magnitude:

$$\rho(L(j\omega)) \triangleq \max_i |\lambda_i(L(j\omega))|$$

Spectral radius stability condition

Consider a system with a stable loop transfer function $L(s)$. Then the closed-loop system is stable if

$$\rho(L(j\omega)) < 1 \quad \forall\omega$$

Small Gain Theorem

Consider a system with a stable loop transfer function $L(s)$. Then the closed-loop system is stable if

$$\|L(j\omega)\| < 1 \quad \forall\omega$$

Where $\|L\|$ denotes any matrix norm that satisfies the multiplicative property $\|AB\| \leq \|A\| \cdot \|B\|$

The Small gain theorem for SISO system says that the system is stable if $|L(j\omega)| < 1$ at all frequencies ω. This is clearly a **very conservative condition** as no phase information is taken into account.
This may be understood as follows: the signals which "return" in the same direction after "one turn around the loop" are magnified by the eigenvalues λ_i (and the directions are the eigenvectors x_i):

$$Lx_i = \lambda_i x_i$$

So if all the eigenvalues λ_i are less than 1 in magnitude, all signals become smaller after each round, and the closed-loop system is stable.

4.10 System Norms

a $\mathcal{H}_2$ norm

$\mathcal{H}_2$ norm

Consider a strictly proper system $G(s)$. The $\mathcal{H}_2$ norm is:

$$\|G(s)\|_2 \triangleq \sqrt{\frac{1}{2\pi}\int_{-\infty}^{\infty} \mathrm{tr}\left(G(j\omega)^H G(j\omega)\right) d\omega}$$
$$= \sqrt{\frac{1}{2\pi}\int_{-\infty}^{\infty} \sum_i {\sigma_i}^2(G(j\omega)) d\omega}$$

The $\mathcal{H}_2$ norm can have a stochastic interpretation where we measure the **expected root mean square value of the output in response to white noise excitation**.

b $\mathcal{H}_\infty$ norm

$\mathcal{H}_\infty$ norm

Consider a proper linear stable system $G(s)$. The $\mathcal{H}_\infty$ norm is the peak value of its maximum singular value:

$$\|G(s)\|_\infty \triangleq \max_\omega \overline{\sigma}(G(j\omega))$$

The $\mathcal{H}_\infty$ norm has several interpretations in the time and frequency domains:

- it is the peak of the transfer function magnitude
- by introducing weights, it can be interpreted as the **magnitude of the some closed-loop transfer function relative to an upper bound**
- it is the worst case steady-state gain for sinusoidal inputs at any frequency
- it is equal to the 2-norm in the time domain:

$$\|G(s)\|_\infty = \max_{w(t)\neq 0} \frac{\|z(t)\|_2}{\|w(t)\|_2} = \max_{\|w(t)\|_2=1} \|z(t)\|_2$$

- is has an interpretation as an induced norm in terms of the expected values of stochastic signals

c Difference Between the $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norms

Minimizing the $\mathcal{H}_\infty$ norm corresponds to minimizing the peak of the largest singular value, whereas minimizing the $\mathcal{H}_2$ norm corresponds to minimizing the sum of the square of all the singular values over all frequencies.

Why is the $\mathcal{H}_\infty$ norm is so popular?

The $\mathcal{H}_\infty$ norm is **convenient for representing unstructured model uncertainty** and because if satisfies the multiplicative property $\|A(s)B(s)\|_\infty \leq \|A(s)\|_\infty \cdot \|B(s)\|_\infty$ It follows that the $\mathcal{H}_\infty$ norm is an **induced norm**.

The $\mathcal{H}_2$ norm on the other hand is not and induced norm and does not satisfies the multiplicative property. This implies that we cannot, by evaluating the $\mathcal{H}_2$ norm of the individual components say anything about how their series interconnection will behave.

d Hankel norm

The Hankel norm of a stable system $G(s)$ is obtained when one applies an input $w(t)$ up to $t = 0$ and measures the output $z(t)$ for $t > 0$, and selects $w(t)$ to maximize the ratio of the 2-norms:

$$\|G(s)\|_H \triangleq \max_{w(t)} \frac{\sqrt{\int_0^\infty \|z(\tau)\|_2^2 \, d\tau}}{\sqrt{\int_{-\infty}^0 \|w(\tau)\|_2^2 \, d\tau}}$$

The Hankel norm is a kind of induced norm from past inputs to future outputs.
It may be shown that the Hankel norm is equal to $\|G(s)\|_H = \sqrt{\rho(PQ)}$ where ρ is the spectral radius, P is the controllability Gramian and Q the observability Gramian.

5 Limitations on Performance in SISO Systems

5.1 Input-Output Controllability

Input-output controllability

The input-output controllability is the **ability to achieve acceptable control performance**; that is, to keep the outputs (y) within specified bounds from their references (r), in spite of unknown but bounded variations, such as disturbances (d) and plant changes, using available inputs (u) and available measurements (y_m).

A plant is controllable if there **exists** a controller that yields acceptable performance for all expected plant variation. Thus, **controllability is independent of the controller and is a property of the plant alone**. It may be affected by changing the plant itself:

- changing the mechanical design
- relocating sensors and actuators
- adding new equipment to dampen disturbances
- adding extra sensor or actuators
- changing the configuration of the lower layers of control already in place

Input-output controllability analysis

Input-output controllability analysis is applied to a plant to find out **what control performance can be expected**.
It is also called **performance targeting**.

If the system has been **scaled**, the requirement for acceptable performance is: For any disturbance $|d| \leq 1$ and any reference $|r| \leq R$, the **performance requirement** is to keep the control error $|e| \leq 1$ using an input $|u| \leq 1$.

5.2 Perfect Control and Plant Inversion

To obtain insight into the inherent limitations on performance, let's consider the **input needed to achieve perfect control**. Let the plant model be: $y = Gu + G_d d$ Since we want perfect control, $y = r$ and we have $u = G^{-1}r - G^{-1}G_d d$ that represents a perfect feedforward controller.
For a feedback control, $u = K(r - y)$, and we have $u = KSr - KSG_d d$ that we can rewrite $u = G^{-1}Tr - G^{-1}TG_d d$.
We see that at frequency where feedback is effective ($T \approx I$), the input generated by feedback is the same as the perfect control input. That is, **high gain feedback generates an inverse of** G.

Perfect control requires the controller to somehow generate an inverse of G. Perfect control cannot be achieved if:

- G contains RHP-zeros (since then G^{-1} is unstable)
- G contains time delay (since then G^{-1} contains non-causal prediction)
- G has more pole than zero (since then G^{-1} is unrealizable)

The required input must not exceed maximum physically allowed value ($|u| \leq 1$), therefore perfect control cannot be achieve if:

- $\left|G^{-1}G_d\right|$ is large (≥ 1)
- $\left|G^{-1}R\right|$ is large (≥ 1)

5.3 Constrain of S and T

a S Plus T is One

S plus T is one

From the definitions $S = (I + L)^{-1}$ and $T = L(I + L)^{-1}$ we derive

$$S + T = I \tag{30}$$

Ideally, we want S small to obtain small control error for commands and disturbances, and T small to avoid sensitivity to noise. There requirements are not simultaneously possible at any frequency.

b The Waterbed Effects

In general, a trade-off between sensitivity reduction and sensitivity increase must be performed whenever:

1. $L(s)$ has at least two more poles than zeros (first waterbed formula)
2. $L(s)$ has a RHP-zero (second waterbed formula)

First Waterbed Formula

Suppose that the open-loop transfer function $L(s)$ is rational and has at least two more poles than zeros. Suppose also that $L(s)$ has N_p RHP-poles at locations p_i. Then for closed-loop stability, the sensitivity function must satisfy the following **Bode Sensitivity Integral**:

$$\int_0^\infty \ln|S(j\omega)|\, d\omega = \pi \sum_{i=1}^{N_p} \mathrm{Re}(p_i) \tag{31}$$

For a **stable plant**, we must have:

$$\int_0^\infty \ln|S(j\omega)|\, d\omega = 0 \tag{32}$$

The area of sensitivity reduction ($\ln|S|$ negative) must equal the area of sensitivity increase ($\ln|S|$ positive): **the benefits and costs of feedback are balanced**. For **unstable plant**, the presence of unstable poles usually increase the peak of $|S|$ as seen from the contribution $\pi\sum_{i=1}^{N_p}\mathrm{Re}(p_i)$. This is the price to pay for stabilizing the system.
From the first waterbed formula, we expect that an increase in the bandwidth must come at the expense of a large peak in $|S|$. Although this is true in most practical cases, however this is not strictly implied by the formula. This is because the increase in area may happen over an infinite frequency range.

Second Waterbed Formula

Suppose that $L(s)$ has a single real **RHP-zero** z or a complex conjugate pair of zero $z = x \pm jy$, and has N_p RHP-poles p_i. For closed-loop stability, the sensitivity function must satisfy

$$\int_0^\infty \ln|S(j\omega)|\, w(z,\omega)d\omega = \pi \ln \prod_{i=1}^{N_p}\left|\frac{p_i+z}{\bar{p}_i-z}\right|$$

where if the zero is real

$$w(z,\omega) = \frac{2z}{z^2+\omega^2}$$

and if the zero pair is complex

$$w(z,\omega) = \frac{x}{x^2+(y-\omega)^2} + \frac{x}{x^2+(y+\omega)^2}$$

The second waterbed formula implies that the peak of $|S|$ is even higher for plants with RHP-zeros.
The weight $w(z,\omega)$ effectively "cuts off" the contribution from $\ln|S|$ to the integral at frequencies $\omega > z$. So we have approximately:

$$\int_0^z \ln|S(j\omega)|\, d\omega \approx 0$$

This is similar to the Bode sensitivity integral, except that the trade-off is done over a limited frequency range. Thus, a large peak for $|S|$ is unavoidable if we try to push down $|S|$ at low frequencies.

c Interpolation Constraints

Interpolation contraints

If p is a **RHP-pole** of the loop transfer function $L(s)$ then

$$T(p) = 1, \quad S(p) = 0 \tag{33}$$

If z is a **RHP-zero** of the loop transfer function $L(s)$ then

$$T(z) = 0, \quad S(z) = 1 \tag{34}$$

d Sensitivity Peaks

Maximum modulus principle

Suppose $f(s)$ is stable, then the maximum value of $|f(s)|$ for s in the RHP is attained on the region's boundary (somewhere along the $j\omega$-axis):

$$\|f(j\omega)\|_\infty = \max_\omega |f(j\omega)| \geq |f(s_0)| \quad \forall s_0 \in \text{RHP}$$

We can derive the following bounds on the peaks of S and T from the maximum modulus principle:

$$\|S\|_\infty \geq \max_j \prod_{i=1}^{N_p}\frac{|z_j+\bar{p}_i|}{|z_j-p_i|} \qquad \|T\|_\infty \geq \max_i \prod_{j=1}^{N_z}\frac{|\bar{z}_j+p_i|}{|z_j-p_i|}$$

This shows that **large peaks** for $|S|$ and $|T|$ are unavoidable if we have a **RHP-zero and RHP-pole located close to each other**.

5.4 Limitation Imposed by Time Delays

Consider a plant $G(s)$ that contains a time delay $e^{-\theta s}$. Even the "ideal" controller cannot remove this delay and the "ideal" sensitivity function is $S = 1 - T = 1 - e^{-\theta s}$.

Upper bound on ω_c for a time delay θ

S crosses 1 at a frequency of about $1/\theta$, so we expect to have an upper bound on ω_c:

$$\omega_c < 1/\theta$$

5.5 Limitation Imposed by RHP-Zeros

RHP-zeros typically appear when we have **competing effects of slow and fast dynamics**. Their presence induces many limitations.

a Inverse Response

We can show that the output of a step change in the input of a stable plant with n_z real RHP-zeros will cross zero n_z times, that is, we have **inverse response**.

b High Gain Instability

It is well known that the closed-loop poles migrate from the open-loop poles to the open-loop zeros as the feedback gain increases. Thus **the presence of RHP-zeros implies high-gain instability**.

c Bandwidth Limitation

To derive bounds for the bandwidth, we select performance weight $w_P(s)$ and we then use the interpolation constraint $S(z) = 1$.
We require $|S(j\omega)| < 1/|w_P(j\omega)| \quad \forall\omega$, so we must at least require that the weight satisfies $|w_P(z)| < 1$.

Performance at low frequencies If we specify performance at low frequencies, we may use the following weight:

$$w_P = \frac{s/M + \omega_B^*}{s + \omega_B^* A}$$

Where ω_B^* is the minimum wanted bandwidth, M the maximum peak of $|S|$ and A the steady-state offset.
If we consider a **real RHP-zero**:

$$\omega_B^* < z\frac{1 - 1/M}{1 - A}$$

For example, with $A = 0$ and $M = 2$, we must at least require $\omega_B^* < 0.5z$.
If we consider an **imaginary RHP-zero**:

$$\omega_B^* < |z| \sqrt{1 - \frac{1}{M^2}}$$

For example, with $M = 2$, we must at least require $\omega_B^* < 0.86\,|z|$.

Performance limitation at low frequency

The presence of RHP-zero imposes an **upper bound on the achievable bandwidth** when we want tight control at low frequencies

Performance at high frequencies We consider the case where we want **tight control at high frequencies**, by use of the performance weight:

$$w_P = \frac{1}{M} + \frac{s}{\omega_B^*}$$

If we consider a **real RHP-zero**:

$$\omega_B^* > z\frac{1}{1 - 1/M}$$

For example, with $M = 2$ the requirement is $\omega_B^* > 2z$, so we can only achieve tight control at frequencies beyond the frequency of the RHP-zero.

Performance limitation at high frequencies

The presence of RHP-zero imposes and **lower bound on the achievable bandwidth** when we want tight control at high frequencies

5.6 Limitation Imposed by RHP-Poles

For unstable plants with a RHP-pole at $s = p$, we **need** feedback for stabilization.

RHP-pole Limitation - Input Usage

In presence of a RHP-pole at $s = p$:

$$\|KS\|_\infty \geq \left|G_s(p)^{-1}\right|$$

where G_s is the "stable version" of G with its RHP-poles mirrored into the LHP.
Since $u = -KS(G_d d + n)$ and because of the previous inequality, the presence of disturbances d and measurement noise n may require the input u to saturate. When the inputs saturate, the system is practically open-loop and the **stabilization is not possible**.

RHP-pole Limitation - Bandwidth

We need to react sufficiently fast. For a real RHP-pole p we must require that the closed-loop bandwidth is larger than $2p$. The presence of **RHP-poles generally imposes a lower bound on the bandwidth**.

5.7 Combined Unstable (RHP) Poles and Zeros

A strictly proper plant with a single real RHP-zero z and a single real RHP-pole p can be stabilized by a stable proper controller if and only if $z > p$. In words "the system may go unstable before we have time to react".
In order to achieve acceptable performance and robustness, we must approximately require $z > 4p$. That is, we want to RHP-pole to be much lower than the RHP-zero.
The presence of RHP-zeros (or time delays) make stabilization more difficult.

5.8 Performance Requirements Imposed by Disturbances and Commands

Disturbance rejection Consider a single disturbance d and a constant reference $r = 0$. Without control, we have $e = G_d d$. We conclude that no control is needed if $|G_d(j\omega)| < 1$ at all frequencies. In that case, the plant is said to be "**self-regulated**".
If $|G_d(j\omega)| > 1$ at some frequency, then **we need control**. In case of feedback control, we have

$$e(s) = S(s)G_d(s)d(s)$$

The performance requirement $|e(\omega)| < 1$ for any $|d(\omega)|$ at any frequency is satisfied if and only if

$$|SG_d(j\omega)| < 1 \quad \forall\omega \quad \Leftrightarrow \quad |S(j\omega)| < 1/\,|G_d(j\omega)| \quad \forall\omega$$

RHP-zero requirement

If the plant has a RHP-zero at $s = z$, then $S(z) = 1$ and we have the following condition:

$$|G_d(z)| < 1$$

Bandwidth requirement

We also have that

$$\omega_B > \omega_d$$

where ω_d is defined by $|G_d(j\omega_d)| = 1$.

The actual bandwidth requirement imposed by disturbances may be higher than ω_d if $|G_d(j\omega)|$ drops with a slope steeper than -1 just before the frequency ω_d. This is because we cannot let the slope of $|L(j\omega)|$ around the crossover be much larger than -1 due to stability margins. It is however possible to overcome this issue using local feedback loops in series.

Command tracking Assume than $d = 0$ and $r(t) = R\sin(\omega t)$. For acceptable control ($|e| < 1$) we must have

$$|S(j\omega)R| < 1 \quad \forall \omega \leq \omega_r$$

where ω_r is the frequency up to which performance tracking is required.

5.9 Limitation Imposed by Input Constraints

Inputs for acceptable control

To achieve acceptable control ($|e| < 1$) and avoid input saturation ($|u| < 1$), we must require:
For **disturbance rejection**:

$$|G| > |G_d| - 1 \text{ at frequencies where } |G_d| > 1$$

For **command tracking**:

$$|G| > |R| - 1 \quad \forall \omega \leq \omega_r$$

5.10 Limitation Imposed by Phase Lag

Phase lag in the plant present no fundamental limitations, however is usually does on practical designs.

Definition - ω_u

Let define ω_u as the frequency where the phase lag of the plant G is $-180°$

$$\angle G(j\omega_u) \triangleq -180° \tag{35}$$

With simple controllers such as a proportional controller or a PI-controller, the phase lag does pose a fundamental limitation on the achievable bandwidth because of stability bounds:

$$\omega_c < \omega_u$$

However, if the model is exactly known and there are no RHP-zeros or time delays, one may extend ω_c to infinite frequency by placing zeros in the controller at the plant poles.

5.11 Limitation Imposed by Uncertainty

Uncertainty with feedforward control Perfect control is obtained using a controller which generates the control input

$$u = G^{-1}r - G^{-1}G_d d$$

When we apply this perfect controller to the actual plant $y' = G'u + G'_d d$, we find

$$e' = y' - r = \underbrace{\left(\frac{G'}{G} - 1\right)}_{\text{rel. error in } G} r - \underbrace{\left(\frac{G'/G'_d}{G/G_d} - 1\right)}_{\text{rel. error in } G/G_d} G'_d d$$

For feedforward control, **the model error propagates directly to the control error**.
If we want acceptable control ($|e'| < 1$), we must require that the model error in G/G_d is less than $1/|G'_d|$. This is very difficult to satisfy at frequencies where $|G'_d|$ is much larger than 1.
The presence of uncertainty then requires us to use feedback control rather than just feedforward control.

Uncertainty with feedback control With feedback control, the closed-loop response is $e = y - r = SG_d d - Sr$. With model error, we get $y' - r = S'(G'_d d - r)$ where $S' = (I + G'K)^{-1}$. S' can be rewritten as $S' = S\frac{1}{1+ET}$ with $E = \frac{G'-G}{G}$ the relative error for G.
We see that the **control error in only weakly affected by model error at frequencies where feedback is effective** ($T \approx 1$).

Uncertainty in the crossover region

Uncertainty in the crossover frequency region can result in poor performance and even instability:

- Uncertainty which keeps $|G(j\omega_u)|$ approximately constant will not change the gain margin.
- Uncertainty which increases $|G(j\omega_u)|$ may yield instability.

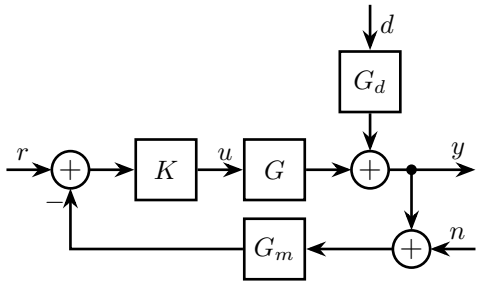


Figure 10 – *Feedback control system*

5.12 Summary: Controllability Analysis with Feedback Control

Consider the control system in Fig. 10. Here $G_m(s)$ denotes the measurement transfer function and we assume $G_m(0) = 1$ (perfect steady-state measurement).

Controllability analysis rules

1. **Speed of response to reject disturbances**. We approximately require $\omega_c > \omega_d$. With feedback control we require $|S(j\omega)| \leq |1/G_d(j\omega)| \quad \forall\omega$.
2. **Speed of response to track reference changes**. We require $|S(j\omega)| \leq 1/R$ up to the frequency ω_r where tracking is required.
3. **Input constraints arising from disturbances**. For acceptable control we require $|G(j\omega)| > |G_d(j\omega)| - 1$ at frequencies where $|G_d(j\omega)| > 1$.
4. **Input constraints arising from setpoints**. We require $|G(j\omega)| > R - 1$ up to the frequency ω_r where tracking is required.
5. **Time delay θ in $G(s)G_m(s)$**. We approximately require $\omega_c < 1/\theta$.
6. **Tight control at low frequencies with a RHP-zero z in $G(s)G_m(s)$**. For a real RHP-zero we require $\omega_c < z/2$ and for an imaginary RHP-zero we approximately require $\omega_c < |z|$.
7. **Phase lag constraint**. We require in most practical cases $\omega_c < \omega_u$. Here the ultimate frequency ω_u is where $\angle GG_m(j\omega_u) = -180°$. Since time delays and RHP-zeros also contribute to the phase lag, it is possible to combine the corresponding rules in the single rule $\omega_c < \omega_u$.
8. **Real open-loop unstable pole in $G(s)$ at $s = p$**. We need high feedback gains to stabilize the system and we approximately require $\omega_c > 2p$.

In summary:

- rules 1, 2 and 8 tell us that we need high feedback gain in order to reject disturbances, to track setpoints and to stabilize the plant.
- rules 5, 6 and 7 tell us we must use low feedback gains in the frequency range where there are RHP-zeros or delays or where the plant has a lot of phase lag.

Sometimes, the disturbances are so large that we hit input saturation or the required bandwidth is not achievable. To avoid the latter problem, we must at least require that the effect of the disturbance is less than 1 at frequencies beyond the bandwidth:

$$|G_d(j\omega)| < 1 \quad \forall\omega \geq \omega_c$$

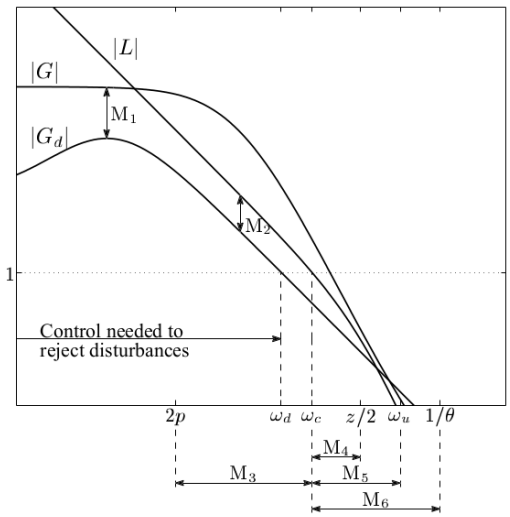


Figure 11 – *Illustration of controllability requirements*

Controllability analysis with feedforward control We find that essentially the same conclusions apply to feedforward control when relevant.
A major difference is that a delay in $G_d(s)$ is an advantage for feedforward control ("it gives the feedforward controller more time to make the right action").

5.13 Conclusion

The controllability analysis is summarized in terms of **eight controllability rules**. These rules are **necessary conditions to achieve acceptable control performance**. They are not sufficient since among other things they only consider one effect at a time. The rules may be used to **determine whether or not a given plant is controllable**.

6 Limitations on Performance in MIMO Systems

6.1 Introduction

In a MIMO system, disturbances, the plant, RHP zeros, RHP poles, delays and disturbances have each **directions** associated with them.
We quantify the directionality of the various effects in G and G_d by their output directions:

- y_z: output dir. of RHP-zero, $G(z)u_z = 0 \cdot y_z$
- y_p: output dir. of RHP-pole, $G(p_i)u_p = \infty \cdot y_p$
- y_d: output dir. of disturbance, $y_d(s) = \frac{1}{\|g_d(s)\|_2} g_d(s)$
- u_i: i'th output dir. (singular vector) of the plant, $G(s)v_i(s) = \sigma_i(s)u_i(s)$

We may also consider input directions, however we are primarily concerned with the performance at the output of the plant.
The **angle between various output directions** is quantified using their inner products.
For example, the output angle between a pole and a zero is $\phi = \cos^{-1}\left|y_z^H y_p\right|$, and:

- if $\phi = 90°$, then the pole and zero are in completely different directions and there is no interaction (they may be considered separately)
- if $\phi = 0°$, then they interact as in a SISO system

6.2 Constraints on S and T

a S plus T is the Identity Matrix

From the identity $S + T = I$, we get:

$$|1 - \overline{\sigma}(S)| \leq \overline{\sigma}(T) \leq 1 + \overline{\sigma}(S) \tag{36a}$$
$$|1 - \overline{\sigma}(T)| \leq \overline{\sigma}(S) \leq 1 + \overline{\sigma}(T) \tag{36b}$$

This shows that we cannot have S and T small simultaneously and that $\overline{\sigma}(S)$ is large if and only if $\overline{\sigma}(T)$ is large.

b Sensitivity Intregrals

The waterbed effect can be generalized for MIMO systems:

$$\int_0^\infty \ln|\det S(j\omega)|d\omega = \sum_j \int_0^\infty \ln \sigma_j(S(j\omega))d\omega$$
$$= \pi \cdot \sum_{i=1}^{N_p} \mathrm{Re}(p_i)$$

c Interpolation Constraints

The basis of many of the results in this chapter are the "**interpolation constraints**".

Interpolation Constraints - RHP-zero z

If $G(s)$ has a RHP-zero at z with output direction y_z, $T(s)$ must have a RHP-zero at z, i.e., $T(z)$ has a zero gain in the direction of output direction y_z of the zero, and we get

$$y_z^H T(z) = 0; \quad y_z^H S(z) = y_z^H$$

Interpolation Constraints - RHP-pole p

If $G(s)$ has a RHP-pole at p with output direction y_p, $S(s)$ must have a RHP-zero at p, i.e. $S(p)$ has a zero gain in the input direction of the output direction y_p of the RHP-pole, and we get

$$S(p)y_p = 0; \quad T(p)y_p = y_p$$

d Sensitivity Peaks

Consider a plant $G(s)$ with RHP-poles p_i and RHP-zeros z_j. The factorization of $G(s)$ in terms of **Blaschke products** is:

$$G(s) = B_p^{-1}G_s(s), \quad G(s) = B_z(s)G_m(s)$$

where G_s is the stable and G_m the minimum-phase version of G. B_p and B_z are stable all-pass transfer matrices (all singular values are 1 for $s = j\omega$) containing the RHP-poles and RHP-zeros respectively.

MIMO sensitivity peaks Suppose that $G(s)$ has N_z RHP-zeros z_j with output directions y_{zj}, and N_p RHP-poles p_i with output direction y_{pi}. We define the all-pass transfer matrices from the Blaschke factorization and compute the real constants:

$$c_{1j} = \left\|y_{zj}^H B_p(z_j)\right\|_2 \geq 1; \quad c_{2i} = \left\|B_z^{-1}(p_i)y_{pi}\right\|_2 \geq 1$$

Let $w_P(s)$ be a stable weight. Then, for closed-loop stability the weighted sensitivity function must satisfy for each RPH-zero z_j

$$\|w_p S\|_\infty \geq c_{1j}\,|w_p(z_j)|$$

Let $w_T(s)$ be a stable weight. Then, for closed-loop stability the weighted complementary sensitivity function must satisfy for each RPH-pole p_i

$$\|w_T T\|_\infty \geq c_{2j}\,|w_T(p_i)|$$

Lower bound on $\|S\|_\infty$ and $\|T\|_\infty$

By selecting $w_P(s) = 1$ and $w_T(s) = 1$, we get

$$\|S\|_\infty \geq \max_{\text{zeros } z_j} c_{1j}; \quad \|T\|_\infty \geq \max_{\text{poles } p_i} c_{2j}$$

6.3 Functional Controllability

Functional controllability

An m-input l-output system $G(s)$ is **functionally controllable** is the normal rank of $G(s)$, denoted r, is equal to the number of outputs ($r = l$), that is, if $G(s)$ has full row rank. A system is functionally uncontrollable if $r < l$.

A square MIMO system is uncontrollable if and only if $\det G(s) = 0, \ \forall s$.
A plant is functionally uncontrollable if and only if $\sigma_l(G(j\omega)) = 0, \ \forall \omega$. $\sigma_l(G(j\omega))$ is then a **measure of how close a plant is to being functionally uncontrollable**.

Uncontrollable output direction

If the plant is not functionally controllable ($r < l$), then there are $l-r$ output directions, denoted y_0 which cannot be affected. These directions will vary with frequency, and we have

$$y_0^H(j\omega)G(j\omega) = 0$$

From an SVD of $G(j\omega) = U\Sigma V^H$, the uncontrollable output directions $y_0(j\omega)$ are the last $l - r$ columns of $U(j\omega)$.

By analyzing the uncontrollable output directions, an engineer can decide on whether it is acceptable to keep certain output combinations uncontrolled, or if additional actuators are needed.

6.4 Limitation Imposed by Time Delays

Time delays pose limitation also in MIMO systems. Let θ_{ij} denote the time delay in the ij'th element of $G(s)$. Then a **lower bound on the time delay for output** i is given by the smallest delay in row i of $G(s)$, that is

$$\theta_i^{\min} = \min_j \theta_{ij}$$

For MIMO systems, we have the surprising result that an increase time delay may sometimes improve the achievable performance. The time delay may indeed increase the decoupling between the outputs.

6.5 Limitations Imposed by RHP-Zeros

The limitations imposed by RHP-zeros on MIMO systems are similar to those for SISO system, although they only apply in particular directions.
The limitations of a RHP-zero located at z may be derived from the bound:

$$\|w_P S(s)\|_\infty = \max_\omega |w_P(j\omega)|\,\overline{\sigma}(S(j\omega)) \geq |w_P(z)|$$

All the results derived for SISO systems generalize if we consider the "worst" direction corresponding to the maximum singular value $\overline{\sigma}(S)$. For instance, if we choose $w_P(s)$ to require tight control at low frequencies, the bandwidth must satisfy $w_B^* < z/2$.

In MIMO systems, one can often **move the deteriorating effect of a RHP-zero to a given output** which may be less important to control well. This is possible because, although the interpolation constraint $y_z^H T(z) = 0$ imposes a certain relationship between the elements within each column of $T(s)$, the columns of $T(s)$ may still be selected independently.

Requiring a decoupled response from r to y generally leads to the introduction of additional RHP-zero in $T(s)$ which are not present in $G(s)$. Moving the effect of the RHP-zero to a particular output generally add some interaction. Also, moving to RHP-zero in a direction where y_z is small usually introduces more interaction than in a direction where y_z is large.
For example, if we have a RHP-zero with $y_z = [0.03, \ -0.04, \ 0.9, \ 0.43]^T$, then one may in theory move the bad effect of the RHP-zero to any of the outputs. However, in practice, it will be difficult to avoid the effect of the RHP-zero on output 3, because the zero direction is mainly in that output. Trying to move it somewhere else will give large interactions and poor performance.

6.6 Limitation Imposed by Unstable (RHP) Poles

For unstable plants, feedback is needed for stabilization. More precisely, the presence of an unstable pole p requires for internal stability $T(p)y_p = y_p$ where y_p is the output pole direction.

Input Usage Limitation

The transfer function KS from plant output to plant inputs must satisfy for any RHP-pole p

$$\|KS\|_\infty \geq \left\|u_p^H G_s(p)^{-1}\right\|_2$$

where u_p is the input pole direction, and G_s is the "stable version" of G with its RHP-poles mirrored in the LHP.

Bandwidth Limitation

From the bound $\|w_T(s)T(s)\|_\infty \geq |w_T(p)|$, we find that a RHP-pole p imposes restrictions on $\overline{\sigma}(T)$ which are identical to those derived on $|T|$ for SISO systems. Thus, we need to react sufficiently fast and we must require that $\overline{\sigma}(T(j\omega))$ is about 1 or larger up to the frequency $2\,|p|$.

6.7 RHP-poles Combined with RHP-Zeros

For a MIMO plant with single RHP-zero z and single RHP-pole p, we derive

$$\|S\|_\infty \geq c \quad \|T\|_\infty \geq c$$
$$\text{with } c = \sqrt{\sin^2\phi + \frac{|z+p|^2}{|z-p|^2}\cos^2\phi}$$

where $\phi = cos^{-1}\left|y_z^H y_p\right|$ is the angle between the RHP-zero and the RHP-pole.
Thus the angle between the RHP-zero and the RHP-pole is of great importance, we usually want $\left|y_z^H y_p\right|$ close to zero so that they don't interact with each other.

6.8 Limitations Imposed by Disturbances

For SISO systems, we found that large and "fast" disturbances require tight control and a large bandwidth. The same results apply for MIMO systems, but again the issue of **directions** is important.

Definition - Disturbance Direction

Consider a scalar disturbance d and let the vector g_d represents its effect on the outputs ($y = g_d d$). The disturbance direction is defined as

$$y_d = \frac{1}{\|g_d\|_2} g_d \tag{37}$$

For a plant with multiple disturbances, g_d is a column of the matrix G_d.

Disturbance Condition Number

$$\gamma_d(G) = \overline{\sigma}(G)\overline{\sigma}(G^\dagger y_d) \tag{38}$$

where $G^\dagger$ is the pseudo inverse of G

The disturbance condition number provides a **measure of how a disturbance is aligned with the plant**. It may vary between 1 (for $y_d = \bar{u}$) if the disturbance is in the "good" direction, and the condition number $\gamma(G) = \overline{\sigma}(G)\overline{\sigma}(G^\dagger)$ (for $y_d = \underline{u}$) if it is in the "bad" direction.

Let assume $r = 0$ and that the system has been scaled. With feedback control $e = Sg_d d$ and the performance objective is

$$\|Sg_d\|_2 = \overline{\sigma}(Sg_d) < 1\ \forall\omega \quad \Leftrightarrow \quad \|Sg_d\|_\infty < 1$$

We derive bounds in terms of the singular values of S:

$$\underline{\sigma}(S)\,\|g_d\|_2 \leq \|Sg_d\|_2 \leq \overline{\sigma}(S)\,\|g_d\|_2$$

For acceptable performance **we must at least require that**

$$\overline{\sigma}(I+L) > \|g_d\|_2$$

And **we may require that**

$$\underline{\sigma}(I+L) > \|g_d\|_2$$

If $G(s)$ has a **RHP-zero** at $s = z$, then the **performance may be poor if the disturbance is aligned with the output direction of this zero**. To satisfy $\|Sg_d\|_\infty < 1$, we must require

$$\left|y_z^H g_d(z)\right| < 1$$

where y_z is the direction of the RHP-zero.

6.9 Limitations Imposed by Input Constraints

a Inputs for Perfect Control

We here consider the question: can the disturbances be rejected perfectly while maintaining $\|u\| < 1$?

For a square plant, the input needed for perfect disturbance rejection is $u = -G^{-1}G_d d$.
For a single disturbance, as the worst-cast disturbance is $|d(\omega)| = 1$, we get that input saturation is avoided ($\|u\|_{\max} \leq 1$) if all elements in the vector $G^{-1}g_d$ are less than 1 in magnitude:

$$\|G^{-1}g_d\|_{\max} < 1,\ \forall\omega$$

It is first recommended to **consider one disturbance at a time** by plotting as a function of frequency the individual elements of $G^{-1}G_d$. This will yields more information about which particular input is most likely to saturate and which disturbance is the most problematic.

b Inputs for Acceptable Control

We here consider the question: is it possible to achieve $\|e\| < 1$ while using inputs with $\|u\| \leq 1$?
For SISO systems, we have to required $|G| > |g_d| - 1$ at frequencies where $|g_d| > 1$. We would like to generalize this result to MIMO systems.

Approximate conditions - SVD

Each singular value σ_i of G must approximately satisfy:

$$\sigma_i(G) \geq \left|u_i^H g_d\right| - 1 \text{ where } \left|u_i^H g_d\right| > 1 \quad (39)$$

with u_i the i'th output singular vector of G. $u_i^H g_d$ may be interpreted as the projection of g_d onto the i'th output singular vector of the plant.

Using the previous approximation, we can find out:

- For which disturbances and at which frequencies input constraints may cause problems. This may give ideas on **which disturbances should be reduced**.
- In which direction i the plant gain is too small. By looking at the corresponding input singular vector v_i, one can determine **which actuators should be redesigned**. By looking at the corresponding output singular vector u_i, one can determine on which outputs we may have to reduce our performance requirements.

For combined disturbances, one requires the i'th row sum of $U^H G_d$ to be less than $\sigma_i(G)$. However, we usually derive more insight by considering one disturbance at a time.

c Unstable Plant and Input Constraints

Active use of inputs are needed to stabilize an unstable plant. We must require $\|KS\|_\infty \geq \left\|u_p^H G_s(p)^{-1}\right\|_2$. If the required inputs exceed the constraints, then stabilization is most likely not possible.

6.10 Limitation Imposed by Uncertainty

The presence of **uncertainty requires the use of feedback** rather than simply feedforward control to get acceptable performance. Sensitivity reduction with respect to uncertainty is achieved with high-gain feedback, but for any real system, we have a crossover frequency range where the loop gain has to drop below 1. The presence of uncertainty in this frequency range may result in poor performance or even instability.
The issues are the same for SISO and MIMO systems, however, with MIMO systems there is an additional problem in that there is also **uncertainty associated with the plant directionality**.

a Input and Output Uncertainty

In practice, the difference between the true perturbed plant G' and the plant model G is caused by a number of different sources. We here focus on input and output uncertainty. In multiplicative form, the input and output uncertainties are given by (see Fig. 12):

$$G' = (I + E_O)G(I + E_I)$$

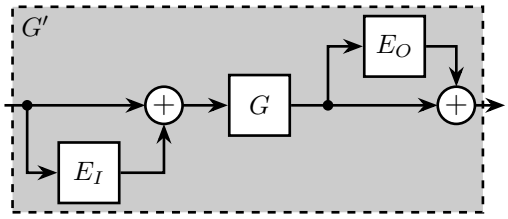


Figure 12 – *Plant with multiplicative input and output uncertainty*

Input and output uncertainty may seem similar, but their **implications for control may be very different**.

If all the elements of E_O and E_I are non-zero, then we have **full block (unstructured) uncertainty**.
In many cases, the source of uncertainty is in the individual input or output channels, and we have that E_I and E_O are **diagonal matrices**. For example $E_I = \text{diag}\{\epsilon_1, \epsilon_2, \dots\}$ where ϵ_i is the **relative uncertainty in input channel** i.
Diagonal input uncertainty is **always** present in real systems and the magnitude of ϵ_i is typically 0.1 or larger.

b Effect of Uncertainty on Feedforward Control

Consider a feedforward controller $u = K_r r$ for the case with no disturbance ($d = 0$). We assume that G is inversible and we select $K_r = G^{-1}$ to achieve perfect control ($e = 0$). However, for the actual plant G' (with uncertainty), the actual control error $e' = y' - r = G'G^{-1}r - r$ is not null and we get:

- For output uncertainty: $e' = E_O r$
- For input uncertainty: $e' = GE_I G^{-1} r$

For output uncertainty, we have an identical result as for SISO systems: the worst case relative control error $\|e'\|_2 / \|r\|_2$ is equal to the magnitude of the relative output uncertainty $\overline{\sigma}(E_O)$. However, for input uncertainty, the sensitivity may be much larger because the elements in the matrix $GE_I G^{-1}$ can be much larger than the elements in E_I.

Diagonal Input Uncertainty

For diagonal input uncertainty, the elements of $GE_I G^{-1}$ are directly related to the RGA:

$$\left[GE_I G^{-1}\right]_{ii} = \sum_{j=1}^{n} \lambda_{ij}(G)\epsilon_j$$

Since diagonal input uncertainty is always present, we can conclude that **if the plant has large RGA elements within in the frequency range where effect control is desired, then it is not possible to achieve good reference tracking with feedforward control** because of strong sensitivity to diagonal input uncertainty. The reverse statement is not true.

c Uncertainty and the Benefits of Feedback

To illustrate the benefits of feedback control in reducing the sensitivity to uncertainty, we consider the effect of output uncertainty on reference tracking both for feedforward and feedback.

Feedforward Let the nominal transfer function with feedforward control be $y = T_r r$ where $T_r = GK_r$ and $K_r = G^{-1}$. With model error $T_r' = G'K_r$ and the change in response is $y' - y = (T_r' - T_r)r = (G' - G)G^{-1}T_r r = E_O T_r r$. Thus, the control error caused by the uncertainty is equal to the relative output uncertainty.

Feedback control The output is $y = Tr$. The change in response is $y' - y = (T' - T)r = S'E_O Tr = S'E_O y$. With feedback control, **the effect of the uncertainty is reduced** by a factor S' compared to that with feedforward control.

d Uncertainty and the Sensitivity Peak

Consider a controller $K(s) = l(s)G^{-1}(s)$ which results in a nominally decoupled response with sensitivity $S = s \cdot I$ and complementary sensitivity $T = t \cdot I$ where $t(s) = 1 - s(s)$. Suppose the plant has diagonal input uncertainty of relative magnitude $|w_I(j\omega)|$ in each input channel. Then there exists a combination of input uncertainties such that at each frequency:

$$\overline{\sigma}(S') \geq \overline{\sigma}(S) \left(1 + \frac{|w_I t|}{1 + |w_I t|} \|\Lambda(G)\|_{i\infty}\right)$$

where $\|\Lambda(G)\|_{i\infty}$ is the maximum row sum of the RGA and $\overline{\sigma}(S) = |s|$.

We can see that with an inverse based controller, the worst case sensitivity will be much larger than the nominal sensitivity at frequencies where the plant has large RGA elements.

Input uncertainty and feedback control

These statements apply to the frequency range around crossover. By "small", we mean smaller than 2 and by "large" we mean larger than 10.

- Condition number $\gamma(G)$ or $\gamma(K)$ small: robust performance to both diagonal and full-block input uncertainty
- Minimized condition number $\gamma_I^*(G)$ or $\gamma_O^*(K)$ small: robust performance to diagonal input uncertainty
- RGA(G) has large elements: inverse based controller is not robust to diagonal input uncertainty. Since diagonal input uncertainty is unavoidable in practice, the rule is never to use a decoupling controller for a plant with large RGA-elements. **Plant with large RGA elements are fundamentally difficult to control**.

e Element-by-element Uncertainty

Consider any complex matrix G and let λ_{ij} denote the ij'th element in the RGA-matrix of G.

The matrix G becomes singular if we make a relative change $-1/\lambda_{ij}$ in its ij'th elements, that is, if a single element in G is perturbed from g_{ij} to $g_{pij} = g_{ij}(1 - \frac{1}{\lambda_{ij}})$

Thus, the RGA-matrix is a **direct measure of sensitivity to element-by-element uncertainty** and matrices with large RGA-values become singular for small relative errors in the elements.
The above result has important implications:

- **Identification**. Models of multivariable plants $G(s)$ are often obtained by identifying one element at a time, for example using step responses. This simple analysis will most likely give meaningless results if there are large RGA-elements within the bandwidth where the model is intended to be used.
- **RHP-zeros**. Consider a plant with transfer function matrix $G(s)$. If the relative uncertainty in an element at a given frequency is larger than $|1/\lambda_{ij}(j\omega)|$ then the plant may be singular at this frequency, implying that the uncertainty allows for a RHP-zero on the $j\omega$-axis.

6.11 MIMO Input-Output Controllability

The following procedure assumes that we have made a decision on the plant inputs and plant outputs, and we want to analyze the model G to find out **what control performance can be expected**. It can also be used to assist in control structure design.

A typical **MIMO controllability analysis** may proceed as follows:

1. **Scale all variables** (inputs u, outputs y, disturbances d, references r) to obtain a scaled model $y = G(s)u + G_d(s)d$, $r = R\tilde{r}$
2. **Obtain a minimal realization**
3. **Check functional controllability**. To be able to control the outputs independently, we first need at least as many inputs u as outputs y. Second, we need the rank of $G(s)$ to be equal to the number of outputs l, i.e. the minimum singular value $G(j\omega)$, $\underline{\sigma}(G) = \sigma_l(G)$, should be non-zero (except at possible $j\omega$-axis zeros). If the plant is not functionally controllable, then compute the output direction where the plant has no gain to have insight into the source of the problem
4. **Compute the poles**. For RHP poles, obtain their locations and associated directions. "Fast" RHP-poles far from the origin are bad
5. **Compute the zeros**. For RHP zeros, obtain their locations and associated directions. Look for zeros pinned into certain outputs. "Small" RHP-zeros (close to the origin) are bad if tight performance is needed at low frequencies
6. **Obtain the frequency response** $G(j\omega)$ and **compute the RGA matrix** $\Gamma = G \times (G^\dagger)^{-1}$. Plants with large RGA-elements at crossover frequencies are difficult to control and should be avoided
7. **Compute the singular values** of $G(j\omega)$ and **plot them as a function of frequency**. Also consider the associated input and output singular vectors
8. The **minimum singular value** $\underline{\sigma}(G(j\omega))$ is a particularly useful **controllability measure**. It should generally be as large as possible at frequencies where control is needed. If $\underline{\sigma}(G(j\omega)) < 1$ then we cannot at frequency ω make independent output changes of unit magnitude by using inputs of unit magnitude
9. For **disturbances**, consider the elements of the matrix G_d. At frequencies where one or more elements is larger than 1, we need control. We get more information by considering one disturbance at a time (the columns g_d of G_d). We must require for each disturbance that S is less than $1/\left\|g_d\right\|_2$ in the disturbance direction y_d, i.e. $\left\|Sy_d\right\|_2 \leq 1/\left\|g_d\right\|_2$. Thus, we must at least require $\underline{\sigma}(S) \leq 1/\left\|g_d\right\|_2$ and we may have to require $\overline{\sigma}(S) \leq 1/\left\|g_d\right\|_2$
10. **Disturbances and input saturation**:
    - **First step**. Consider the input magnitudes needed for perfect control by computing the elements in the matrix $G^\dagger G_d$. If all elements are less than 1 at all frequencies, then input saturation is not expected to be a problem. If some elements of $G^\dagger G_d$ are larger than 1, then perfect control cannot be achieve at this frequency, but "acceptable" control may be possible
    - **Second step**. Consider the elements of $U^H G_d$ and make sure that the elements in the i'th row are smaller than $\sigma_i(G) + 1$ at all frequencies
11. **Are the requirements compatible?** Look at disturbances, RHP-poles, RHP-zeros and their associated locations and directions. For example, we must required for each disturbance and each RHP-zero that $\left|y_z^H g_d(z)\right| \leq 1$. Similar relations exist for combined RHP-zero and RHP-pole.
12. **Uncertainty**. If the condition number $\gamma(G)$ is small then we expect no particular problems with uncertainty. If the RGA-elements are large, we expect strong sensitivity to uncertainty.

Plant design changes If the plant is not input-output controllable, then it must be modified. Some possible modifications are:

- **Controlled outputs**. Identify the outputs which cannot be controlled satisfactory. Can the specifications for these be relaxed?
- **Manipulated inputs**. If input constraints are encountered, then consider replacing or moving actuators. If there are RHP-zeros which cause control problems, then the zeros may often be eliminated by adding another input. This may not be possible if the zero is pinned to a particular output
- **Extra measurements**. If the effect of disturbances or uncertainty is large, and the dynamics of the plant are such that acceptable control cannot be achieved, then consider adding "fast local loops" based on extra measurements which are located close to the inputs and disturbances
- **Disturbances**. If the effect of disturbances is too large, then see whether the disturbance itself may be reduced. This may involve adding extra equipment to dampen the disturbances. In other cases, this may involve improving or changing the control of another part of the system: we may have a disturbance which is actually the manipulated input for another part of the system
- **Plant dynamics and time delays**. In most cases, controllability is improved by making the plant dynamics faster and by reducing time delays. An exception to this is a strongly interactive plant, where an increased dynamic lag or time delay may be helpful if it somehow "delays" the effect of the interactions

6.12 Conclusion

We have found that most of the insights into the performance limitation of SISO systems carry over to MIMO systems. For RHP-zeros, RHP-poles and disturbances, the issue of directions usually makes the limitation **less severe** for MIMO than for SISO systems. However, the

situation is usually the opposite with model uncertainty because for MIMO systems, there is also uncertainty associated with plant directionality.

7 Uncertainty and Robustness for SISO Systems

7.1 Introduction to Robustness

A control system is robust if it is insensitive to differences between the actual system and the model of the system which was used to design the controller. The key idea in the $\mathcal{H}_\infty$ robust control paradigm is to check whether the design specifications are satisfied even for the **"worst-case" uncertainty**.
Our approach is then as follows:

1. **Determine the uncertainty set**. Find a mathematical representation of the model uncertainty
2. **Check Robust Stability (RS)**. Determine whether the system remains stable for all plants in the uncertainty set
3. **Check Robust Performance (RP)**. If RS is satisfied, determine whether the performance specifications are met for all plants in the uncertainty set

This approach may not always achieve optimal performance. In particular, if the worst case plant rarely occurs, other approaches, such as optimizing some average performance or using adaptive control may yield better performance.

To account for model uncertainty, we will assume that the dynamic behavior of a plant is described not by a single linear time invariant model but by a **set Π of possible linear time invariant models**, sometimes denoted the "**uncertainty set**".

We adopt the following notation:

- Π - a set of possible perturbed plant models
- $G(s) \in \Pi$ - nominal plant model
- $G_p(s) \in \Pi$ - particular perturbed plant models

We will use a "**norm-bounded uncertainty description**" where the set Π is generated by allowing $\mathcal{H}_\infty$ norm-bounded stable perturbations to the nominal plant $G(s)$. We let E denote a perturbation which is not normalized, and let Δ denote a normalized perturbation with its $\mathcal{H}_\infty$ norm less than 1.

7.2 Representing Uncertainty

Uncertainty in the plant model may have **several origins**:

1. There are always parameters in the linear model which are only known approximatively
2. Parameters in the model may vary due to **nonlinearities** or changes in the operating conditions
3. Measurement devices have imperfections
4. At high frequencies, even the structure and the model order is unknown, and the uncertainty will always exceed 100 % at some frequency
5. Even when a very detailed model is available, we may choose to work with a simpler nominal model and **represent the neglected dynamics as "uncertainty"**
6. The controller implemented may differ from the one obtained by solving the synthesis problem. One may include uncertainty to allow for controller order reduction and implementation inaccuracies

The various sources of model uncertainty may be grouped into two main classes:

1. **Parametric uncertainty**. The structure of the model is known, but some parameters are uncertain
2. **Neglected and unmodelled dynamics uncertainty**. The model is in error because of missing dynamics, usually at high frequencies

Parametric uncertainty set

Parametric uncertainty will be quantified by assuming that **each uncertain parameters is bounded within some region** $[\alpha_{\min}, \alpha_{\max}]$. That is, we have parameter sets of the form

$$\alpha_p = \bar{\alpha}(1 + r_\alpha \Delta); \quad r_\alpha = \frac{\alpha_{\max} - \alpha_{\min}}{\alpha_{\max} + \alpha_{\min}} \tag{40}$$

where $\bar{\alpha}$ is the mean parameter value, r_α is the relative uncertainty in the parameter, and Δ is any real scalar satisfying $|\Delta| \leq 1$.

Neglected and unmodelled dynamics uncertainty is somewhat less precise and thus more difficult to quantify, but it appears that frequency domain is particularly well suited for this class. This leads to **complex perturbations** which we normalize such that $\|\Delta\|_\infty \leq 1$.

There is also a third class of uncertainty (which is a combination of the other two) called **Lumped uncertainty**. Here the uncertainty description represents one or several sources of parametric and/or unmodelled dynamics uncertainty combined into a single lumped perturbation of a chosen structure. The frequency domain is also well suited for describing lumped uncertainty.

Multiplicative uncertainty

In most cases, we prefer to lump the uncertainty into a multiplicative uncertainty of the form

$$G_p(s) = G(s)(1 + w_I(s)\Delta_I(s)); \quad |\Delta_I(j\omega)| \leq 1 \, \forall \omega$$

which may be represented by the diagram in Fig. 13.

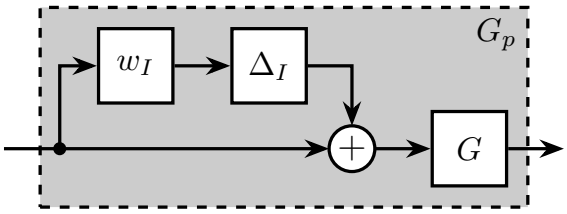


Figure 13 – *Plant with multiplicative uncertainty*

7.3 Parametric Uncertainty

Parametric uncertainty may also be represented in the $\mathcal{H}_\infty$ framework if we restrict Δ to be real.

Example - Gain uncertainty

$$G_p(s) = k_p G_0(s); \quad k_{\min} \le k_p \le k_{\max}$$

where k_p is an uncertain gain and $G_0(s)$ is a transfer function with no uncertainty. By writing $k_p = \bar{k}(1 + r_k\Delta)$ where r_k is the relative magnitude of the gain uncertainty and $\bar{k}$ is the average gain, be may write

$$G_p = \underbrace{\bar{k}G_0(s)}_{G(s)}(1 + r_k\Delta), \quad |\Delta| \le 1$$

where Δ is a real scalar and $G(s)$ is the nominal plant.

Example - Time constant uncertainty

$$G_p(s) = \frac{1}{\tau_p s + 1} G_0(s); \quad \tau_{\min} \le \tau_p \le \tau_{\max}$$

By writing $\tau_p = \bar{\tau}(1 + r_\tau\Delta)$, with $|\Delta| \le 1$, the model set can be rewritten as

$$G_p(s) = \frac{G_0}{1 + \bar{\tau}s + r_\tau\bar{\tau}s\Delta} = \underbrace{\frac{G_0}{1 + \bar{\tau}s}}_{G(s)} \frac{1}{1 + w_{iI}(s)\Delta}$$

with $w_{iI}(s) = \dfrac{r_\tau\bar{\tau}s}{1 + \bar{\tau}s}$.

As shown in the two examples, one can represent parametric uncertainty in the $\mathcal{H}_\infty$ framework. However, **parametric uncertainty is often avoided** for the following reasons:

1. It usually requires a **large effort** to model parametric uncertainty
2. A parametric uncertainty model is somewhat deceiving in the sense that it provides a very detailed and accurate description, even though the underlying assumptions about the model and the parameters may be much less exact
3. The **exact model structure is required** and so unmodelled dynamics cannot be dealt with
4. Real perturbations are required, which are more difficult to deal with mathematically and numerically, especially when it comes to controller synthesis

Therefore, parametric uncertainty is often represented by **complex perturbations**. For example, we may simply replace the real perturbation, $-1 \le \Delta \le 1$ by a complex perturbation with $|\Delta(j\omega)| \le 1$. This is of course conservative as it introduces possible plants that are not present in the original set. However, if there are several real perturbations, then the conservatism if often reduced by lumping these perturbations into a single complex perturbation.

7.4 Representing Uncertainty in the Frequency Domain

a Uncertain Regions

To illustrate how parametric uncertainty translate into frequency domain uncertainty, consider in Fig. 14 the Nyquist plots generated by the following set of plants

$$G_p(s) = \frac{k}{\tau s + 1} e^{-\theta s}, \quad 2 \le k, \theta, \tau \le 3$$

- **Step 1**. At each frequency, a region of complex numbers $G_p(j\omega)$ is generated by varying the parameters. In general, these uncertain regions have complicated shapes and complex mathematical descriptions
- **Step 2**. We therefore approximate such complex regions as discs, resulting in a **complex additive uncertainty description**

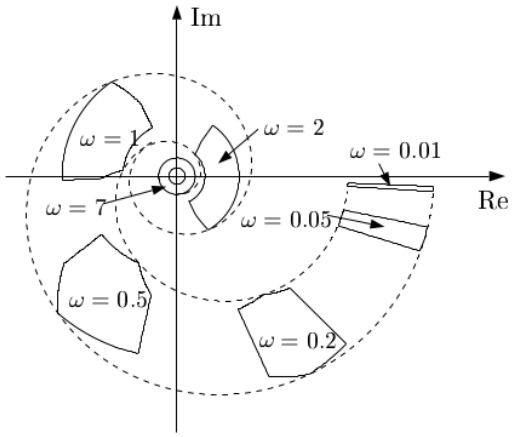


Figure 14 – *Uncertainty regions of the Nyquist plot at given frequencies*

b Representing Uncertainty Regions by Complex Perturbations

Additive uncertainty

The disc-shaped regions may be generated by additive complex norm-bounded perturbations around a nominal plant G

$$\Pi_A: \; G_p(s) = G(s) + w_A(s)\Delta_A(s) \quad \text{with } |\Delta_A(j\omega)| \le 1 \, \forall\omega \tag{41}$$

At each frequency, all possible $\Delta(j\omega)$ "generates" a disc-shaped region with radius 1 centered at 0, so $G(j\omega)+w_A(j\omega)\Delta_A(j\omega)$ generates at each frequency a disc-shapes region of radius $|w_A(j\omega)|$ centered at $G(j\omega)$ as shown in Fig. 15.

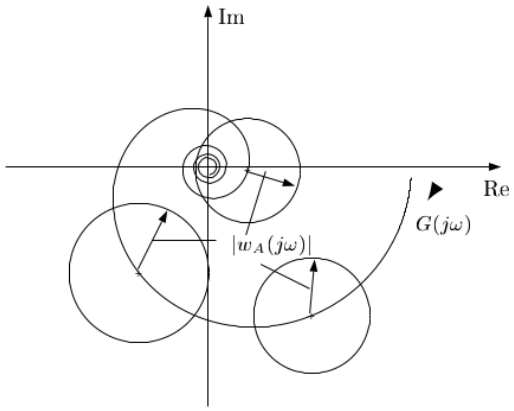


Figure 15 – *Disc-shaped uncertainty regions generated by complex additive uncertainty*

Multiplicative uncertainty

The disc-shaped region may alternatively be represented by a **multiplicative uncertainty**

$$\Pi_I: \; G_p(s) = G(s)(1 + w_I(s)\Delta_I(s)); \quad \text{with } |\Delta_I(j\omega)| \le 1 \, \forall\omega \tag{42}$$

And we see that for SISO systems, additive and multiplicative uncertainty are equivalent if at each frequency:

$$|w_I(j\omega)| = |w_A(j\omega)| \,/\, |G(j\omega)|$$

However, **multiplicative weights are often preferred because their numerical value is more informative**. At frequencies where $|w_I(j\omega)| > 1$ the uncertainty exceeds 100 % and the Nyquist curve may pass through the origin. Then, at these frequencies, we do not know the phase of the plant, and we allow for zeros crossing from the left to the right-half plane. **Tight control is then not possible** at frequencies where $|w_I(j\omega)| \ge 1$.

c Obtaining the Weight for Complex Uncertainty

Consider a set Π of possible plants resulting, for example, from parametric uncertainty. We now want to describe this set of plants by a single complex perturbation Δ_A or Δ_I.
This complex disc-shaped uncertainty description may be generated as follows:

1. Select a nominal $G(s)$
2. **Additive uncertainty**. At each frequency, find the smallest radius $l_A(\omega)$ which includes all the possible plants Π

   $$l_A(\omega) = \max_{G_p \in \Pi} |G_p(j\omega) - G(j\omega)|$$

   If we want a rational transfer function weight, $w_A(s)$, then it must be chosen to cover the set, so

   $$|w_A(j\omega)| \ge l_A(\omega) \quad \forall\omega$$

   Usually $w_A(s)$ is of low order to simplify the controller design.
3. **Multiplicative uncertainty**. This is often the preferred uncertainty form, and we have

   $$l_I(\omega) = \max_{G_p \in \Pi} \left| \frac{G_p(j\omega) - G(j\omega)}{G(j\omega)} \right|$$

   and with a rational weight $|w_I(j\omega)| \ge l_I(\omega)$, $\forall\omega$

Example - Parametric uncertainty

We want to represent the following set using multiplicative uncertainty with a rational weight $w_I(s)$

$$\Pi: \quad G_p(s) = \frac{k}{\tau s + 1} e^{-\theta s}, \quad 2 \le k, \theta, \tau \le 3$$

To simplify subsequent controller design, we select a delay-free nominal model

$$G(s) = \frac{\bar{k}}{\bar{\tau} s + 1} = \frac{2.5}{2.5 s + 1}$$

To obtain $l_I(\omega)$, we consider three values (2, 2.5 and 3) for each of the three parameters (k, θ, τ). The corresponding relative errors $\left|\frac{G_p - G}{G}\right|$ are shown as functions of frequency for the $3^3 = 27$ resulting G_p (Fig. 16). To derive $w_I(s)$, we then try to find a simple weight so that $|w_I(j\omega)|$ lies above all the dotted lines.

d Choice of Nominal Model

With parametric uncertainty represented as complex perturbations, there are three main options for the choice of nominal model:

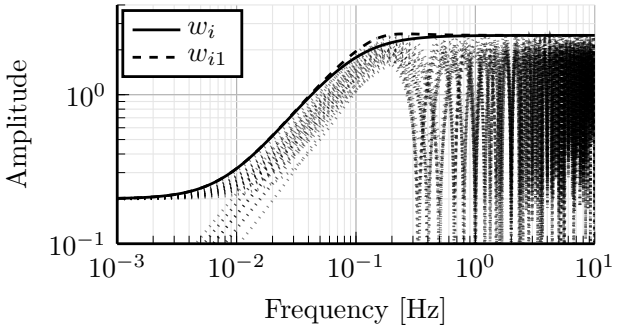


Figure 16 – *Relative error for 27 combinations of k, τ and θ. Solid and dashed lines: two weights $|w_I|$*

1. **A simplified model**, for instance a low order, delay free model. It usually yields the largest uncertainty region, but the model is simple and this facilitates controller design in later stages.
2. **A model of mean parameter values**, $G(s) = \bar{G}(s)$. It is probably the most straightforward choice.
3. **The central plant obtained from a Nyquist plot**. It yields the smallest region, but in this case a significant effort may be required to obtain the nominal model which is usually not a rational transfer function.

For SISO systems, we find that for plants with an uncertain time delay, it is simplest and sometimes best to use a delay-free nominal model, and to represent the nominal delay as additional uncertainty.
If we use a parametric uncertainty description, based on multiple real perturbations, then we should always use the mean parameter values in the nominal model.

e Neglected Dynamics Represented as Uncertainty

We saw that one advantage of frequency domain uncertainty description is that one can choose to work with a simple nominal model, and **represent neglected dynamics as uncertainty**.
Consider a set of plants

$$G_p(s) = G_0(s)f(s)$$

where $G_0(s)$ is fixed. We want to neglect the term $f(s) \in \Pi_f$, and represent G_p by multiplicative uncertainty with a nominal model $G = G_0$.
The magnitude of the relative uncertainty caused by neglecting the dynamics in $f(s)$ is

$$l_I(\omega) = \max_{G_p} \left| \frac{G_p - G}{G} \right| = \max_{f(s)\in\Pi_f} |f(j\omega) - 1|$$

Neglected delay Let $f(s) = e^{-\theta_p s}$, where $0 \le \theta_p \le \theta_{\max}$. We want to represent $G_p(s) = G_0(s)e^{-\theta_p s}$ by a delay-free plant $G_0(s)$ and multiplicative uncertainty. Let first consider the maximum delay, for which the relative error $\left|1 - e^{-j\omega\theta_{\max}}\right|$ is shown as a function of frequency (Fig. 17). If we consider all $\theta \in [0, \theta_{\max}]$ then:

$$l_I(\omega) = \begin{cases} \left|1 - e^{-j\omega\theta_{\max}}\right| & \omega < \pi/\theta_{\max} \\ 2 & \omega \ge \pi/\theta_{\max} \end{cases}$$

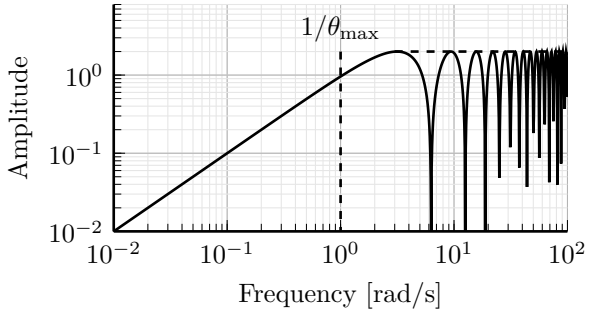


Figure 17 – *Neglected time delay*

Neglected lag Let $f(s) = 1/(\tau_p s + 1)$, where $0 \le \tau_p \le \tau_{\max}$. In this case the resulting $l_I(\omega)$ (Fig. 18) can be represented by a rational transfer function with $|w_I(j\omega)| = l_I(\omega)$ where

$$w_I(s) = \frac{\tau_{\max} s}{\tau_{\max} s + 1}$$

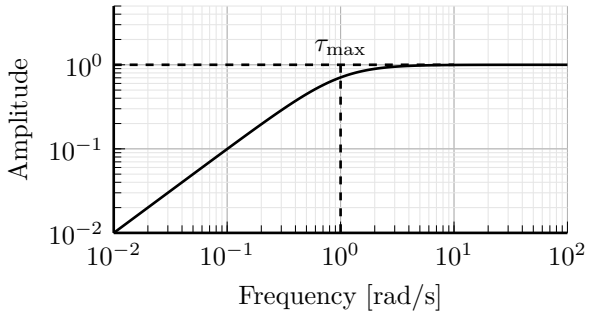


Figure 18 – *Neglected first-order lag uncertainty*

Multiplicative weight for gain and delay uncertainty Consider the following set of plants

$$G_p = k_p e^{-\theta_p s} G_0(s); \quad k_p \in [k_{\min}, k_{\max}],\ \theta_p \in [\theta_{\min}, \theta_{\max}]$$

which we want to represent by multiplicative uncertainty and a delay-free nominal model $G(s) = \bar{k}G_0(s)$. There is an exact expression, its first order approximation is

$$w_I(s) = \frac{(1 + \frac{r_k}{2})\theta_{\max} s + r_k}{\frac{\theta_{\max}}{2} s + 1}$$

However, as shown in Fig. 19, the weight w_I is optimistic, especially around frequencies $1/\theta_{\max}$. To make

sure that $|w_I(j\omega)| \leq l_I(\omega)$, we can apply a correction factor:

$$w_I'(s) = w_I \cdot \frac{(\frac{\theta_{\max}}{2.363})^2 s^2 + 2 \cdot 0.838 \cdot \frac{\theta_{\max}}{2.363} s + 1}{(\frac{\theta_{\max}}{2.363})^2 s^2 + 2 \cdot 0.685 \cdot \frac{\theta_{\max}}{2.363} s + 1}$$

It is suggested to start with the simple weight and then if needed, to try the higher order weight.

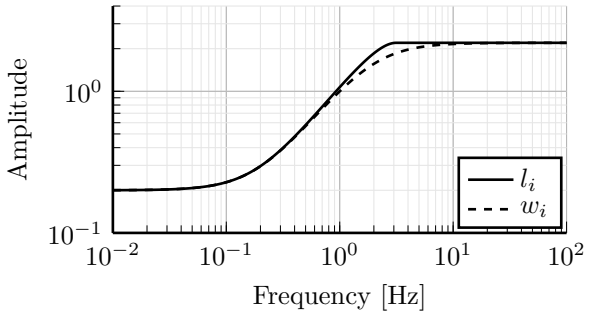


Figure 19 – *Multiplicative weight for gain and delay uncertainty*

f Unmodelled Dynamics Uncertainty

The most important reason for using frequency domain ($\mathcal{H}_\infty$) uncertainty description and complex perturbations, is the **incorporation of unmodelled dynamics**. Unmodelled dynamics, while being close to neglected dynamics, also include unknown dynamics of unknown or even infinite order.

Unmodelled dynamics - Weight

To represent unmodelled dynamics, we usually use a simple **multiplicative weight** of the form

$$w_I(s) = \frac{\tau s + r_0}{(\tau / r_\infty) s + 1} \tag{43}$$

where r_0 is the relative uncertainty at steady-state, $1/\tau$ is the frequency at which the relative uncertainty reaches 100 %, and r_∞ is the magnitude of the weight at high frequency.

7.5 SISO Robust Stability

a RS with Multiplicative Uncertainty

We want to determine the stability of the uncertain feedback system in Fig. 20 where there is multiplicative uncertainty of magnitude $|w_I(j\omega)|$. The loop transfer function becomes

$$L_P = G_p K = GK(1 + w_I \Delta_I) = L + w_I L \Delta_I$$

We assume (by design) the stability of the nominal closed-loop system (with $\Delta_I = 0$). We use the Nyquist stability condition to test for robust stability of the closed loop system:

$$\begin{aligned} \text{RS} &\overset{\text{def}}{\iff} \text{System stable } \forall L_p \\ &\iff L_p \text{ should not encircle -1, } \forall L_p \end{aligned}$$

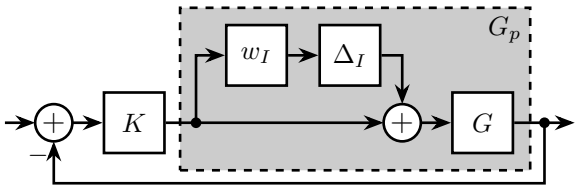


Figure 20 – *Feedback system with multiplicative uncertainty*

Graphical derivation of RS-condition Consider the Nyquist plot of L_p as shown in Fig. 21. $|1 + L|$ is the distance from the point -1 to the center of the disc representing L_p and $|w_I L|$ is the radius of the disc. Encirclements are avoided if none of the discs cover -1, and we get:

$$\begin{aligned} \text{RS} &\Leftrightarrow |w_I L| < |1 + L|, \ \forall \omega \\ &\Leftrightarrow \left| \frac{w_I L}{1 + L} \right| < 1, \ \forall \omega \\ &\Leftrightarrow |w_I T| < 1, \ \forall \omega \end{aligned}$$

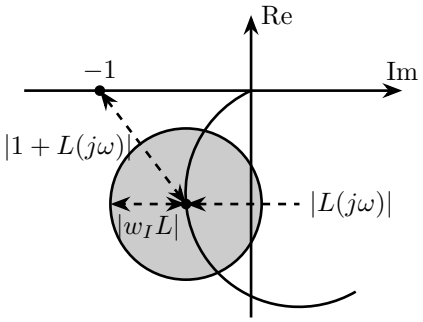


Figure 21 – *Nyquist plot of L_p for robust stability*

Requirement for Robust Stability

The requirement of robust stability for the case with multiplicative uncertainty gives an **upper bound on the complementary sensitivity**

$$\text{RS} \Leftrightarrow |T| < 1/|w_I|, \ \forall \omega \tag{44}$$

We see that we have to make T small at frequencies where the relative uncertainty $|w_I|$ exceeds 1 in magnitude.

Algebraic derivation of RS-condition Since L_p is assumed stable, and the nominal closed-loop is stable, the nominal loop transfer function $L(j\omega)$ does not

encircle -1. Therefore, since the set of plants is norm-bounded, it then follows that if some L_{p1} in the uncertainty set encircles -1, then there must be another L_{p2} in the uncertainty set which goes exactly through -1 at some frequency. Thus

$$\begin{aligned} \text{RS} \quad &\Leftrightarrow |1+L_p| \neq 0, \ \forall L_p, \forall\omega \\ &\Leftrightarrow |1+L_p| > 0, \ \forall L_p, \forall\omega \\ &\Leftrightarrow |1+L+w_I L\Delta_I| > 0, \ \forall |\Delta_I| \leq 1, \forall\omega \end{aligned}$$

At each frequency, the last condition is most easily violated when the complex number $\Delta_I(j\omega)$ is selected with $|\Delta(j\omega)| = 1$ and with phase such that $1+L$ and $w_I L\Delta_I$ point in the opposite direction. Thus

$$\text{RS} \ \Leftrightarrow \ |1+L| - |w_I L| > 0, \ \forall\omega \ \Leftrightarrow \ |w_I T| < 1, \ \forall\omega$$

And we obtain the same condition as before.

b RS with Inverse Multiplicative Uncertainty

We will derive a corresponding RS-condition for feedback system with inverse multiplicative uncertainty (Fig. 22) in which

$$G_p = G(1 + w_{iI}(s)\Delta_{iI})^{-1}$$

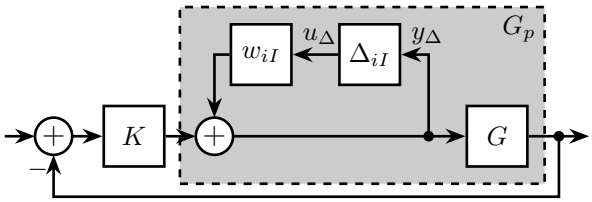


Figure 22 – *Feedback system with inverse multiplicative uncertainty*

We assume that L_p and the nominal closed-loop systems are stable. Robust stability is guaranteed if $L_p(j\omega)$ does not encircles the point -1:

$$\begin{aligned} \text{RS} \quad &\Leftrightarrow \quad |1+L_p| > 0, \ \forall L_p, \forall\omega \\ &\Leftrightarrow \quad \left|1 + L(1+w_{iI}\Delta_{iI})^{-1}\right| > 0, \ \forall |\Delta_{iI}| < 1, \forall\omega \\ &\Leftrightarrow \quad |1 + w_{iI}\Delta_{iI} + L| > 0, \ \forall |\Delta_{iI}| < 1, \forall\omega \\ &\Leftrightarrow \quad |1+L| - |w_{iI}\Delta_{iI}| > 0, \ \forall\omega \\ &\Leftrightarrow \quad |w_{iI}S| < 1, \ \forall\omega \end{aligned}$$

RS - inverse multiplicative uncertainty

The requirement for robust stability for the case with inverse multiplicative uncertainty gives an **upper bound on the sensitivity**

$$\text{RS} \quad \Leftrightarrow \quad |S| < 1/|w_{iI}|, \ \forall\omega \qquad (45)$$

We see that we need tight control and have to make S small at frequencies where the uncertainty is large and w_{iI} exceeds 1 in magnitude. The reason is that the uncertainty represents pole uncertainty, and at frequencies where $|w_{iI}|$ exceeds 1, we allow for poles crossing from the left to the right-half plant, and we then need feedback ($|S| < 1$) in order to stabilize the system.

7.6 SISO Robust Performance

a SISO Nominal Performance

Nominal performance

The condition for nominal performance when considering performance in terms of the **weighted sensitivity** function is

$$\begin{aligned} \text{NP} &\Leftrightarrow |w_P S| < 1 \ \forall\omega \\ &\Leftrightarrow |w_P| < |1+L| \ \forall\omega \end{aligned} \qquad (46)$$

Now $|1+L|$ represents at each frequency the distance of $L(j\omega)$ from the point -1 in the Nyquist plot, so $L(j\omega)$ must be at least a distance of $|w_P(j\omega)|$ from -1. This is illustrated graphically in Fig. 23.

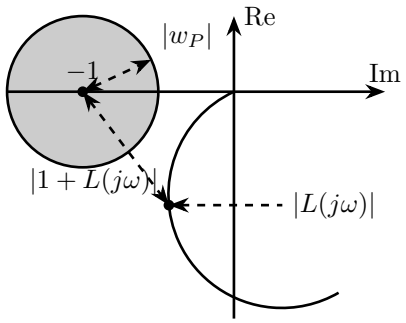


Figure 23 – *Nyquist plot illustration of the nominal performance condition $|w_P| < |1+L|$*

b Robust Performance

For robust performance, we require the performance condition to be satisfied for **all** possible plants:

$$\begin{aligned} \text{RP} &\overset{\text{def}}{\Leftrightarrow} |w_P S| < 1 \quad \forall S_p, \forall\omega \\ &\Leftrightarrow |w_P| < |1+L_p| \quad \forall L_p, \forall\omega \end{aligned} \qquad (47)$$

Let's consider the case of multiplicative uncertainty as shown on Fig. 24. The robust performance corresponds to requiring $|\hat{y}/d| < 1 \ \forall\Delta_I$ and the set of possible loop transfer functions is

$$L_p = G_p K = L(1 + w_I\Delta_I) = L + w_I L\Delta_I$$

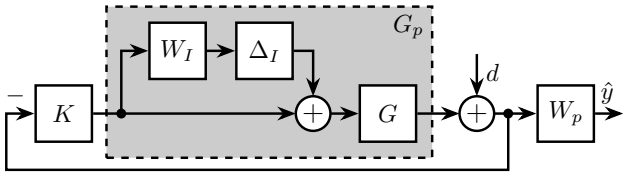


Figure 24 – *Diagram for robust performance with multiplicative uncertainty*

Graphical derivation of RP-condition As illustrated on Fig. 23, we must required that all possible $L_p(j\omega)$ stay outside a disk of radius $|w_P(j\omega)|$ centered on -1. Since L_p at each frequency stays within a disk of radius $|w_I(j\omega)L(j\omega)|$ centered on $L(j\omega)$, the condition for RP becomes:

$$\begin{aligned} \text{RP} &\Leftrightarrow |w_P| + |w_I L| < |1 + L| \quad \forall\omega \\ &\Leftrightarrow \left|w_P(1+L)^{-1}\right| + \left|w_I L(1+L)^{-1}\right| < 1 \quad \forall\omega \end{aligned}$$

Robust Performance Condition

Finally, we obtain the following condition for Robust Performance:

$$\text{RP} \Leftrightarrow \max_{\omega} \left(|w_P S| + |w_I T|\right) < 1 \qquad (48)$$

Algebraic derivation of RP-condition RP is satisfied if the worst-case weighted sensitivity at each frequency is less than 1:

$$\text{RP} \Leftrightarrow \max_{S_p} |w_P S_p| < 1, \quad \forall\omega$$

The perturbed sensitivity S_p is

$$S_p = \frac{1}{1+L_p} = \frac{1}{1+L+w_I L \Delta_I}$$

Thus:

$$\max_{S_p} |w_P S_p| = \frac{|w_P|}{|1+L| - |w_I L|} = \frac{|w_P S|}{1 - |w_I T|}$$

And we obtain the same RP-condition as the graphically derived one.

Remarks on RP-condition

1. The RP-condition for this problem is closely approximated by the mixed sensitivity $\mathcal{H}_\infty$ condition:

$$\left\| \begin{matrix} w_P S \\ w_I T \end{matrix} \right\|_\infty = \max_{\omega} \sqrt{|w_P S|^2 + |w_I T|^2} < 1$$

   This condition is within a factor at most $\sqrt{2}$ of the true RP-condition. This means that **for SISO systems, we can closely approximate the RP-condition in terms of an $\mathcal{H}_\infty$ problem**, so there is no need to make use of the structured singular value. However, we will see that the situation can be very different for MIMO systems.
2. The RP-condition can be used to derive bounds on the loop shape $|L|$:

$$|L| > \frac{1+|w_P|}{1-|w_I|}, \text{ at frequencies where } |w_I| < 1$$

$$|L| < \frac{1-|w_P|}{1+|w_I|}, \text{ at frequencies where } |w_P| < 1$$

c The Relationship Between NP, RS and RP

Consider a SISO system with multiplicative input uncertainty, and assume that the closed-loop is nominally stable (NS). The conditions for nominal performance (NP), robust stability (RS) and robust performance (RP) as summarized as follows:

$$\text{NP} \Leftrightarrow |w_P S| < 1, \ \forall\omega \qquad (49a)$$
$$\text{RS} \Leftrightarrow |w_I T| < 1, \ \forall\omega \qquad (49b)$$
$$\text{RP} \Leftrightarrow |w_P S| + |w_I T| < 1, \ \forall\omega \qquad (49c)$$

From this we see that **a prerequisite for RP is that we satisfy both NP and RS**. This applies in general, both for SISO and MIMO systems and for any uncertainty.

In addition, for SISO systems, if we satisfy both RS and NP, then we have at each frequency:

$$|w_P S| + |w_I T| < 2 \cdot \max\{|w_P S|, |w_I T|\} < 2$$

It then follows that, within a factor at most 2, we will automatically get RP when NP and RS are satisfied. This, RP is not a "big issue" for SISO systems.

To satisfy RS we generally want T small, whereas to satisfy NP we generally want S small. However, we cannot make both S and T small at the same frequency because of the identity $S+T=1$. This has implications for RP:

$$\begin{aligned} |w_P S| + |w_I T| &\geq \min\{|w_P|, |w_I|\}(|S| + |T|) \\ &\geq \min\{|w_P|, |w_I|\}(|S+T|) \\ &\geq \min\{|w_P|, |w_I|\} \end{aligned}$$

This means that we cannot have both $|w_P| > 1$ (i.e. good performance) and $|w_I| > 1$ (i.e. more than 100% uncertainty) at the same frequency.

7.7 Examples of Parametric Uncertainty

a Parametric Pole Uncertainty

Consider the following set of plants:

$$G_p(s) = \frac{1}{s - a_p} G_0(s); \quad a_{\min} \leq a_p \leq a_{\max}$$

If $a_{\min}$ and $a_{\max}$ have different signs, then this means that the plant can change from stable to unstable with the pole crossing through the origin.
This set of plants can be written as

$$G_p(s) = \frac{G_0(s)}{s - \bar{a}(1 + r_a\Delta)}; \quad -1 \le \Delta \le 1$$

which can be exactly described by inverse multiplicative uncertainty:

$$G(s) = \frac{G_0(s)}{(s - \bar{a})}; \quad w_{iI}(s) = \frac{r_a\bar{a}}{s - \bar{a}}$$

The magnitude of $w_{iI}(s)$ is equal to r_a at low frequency and goes to 0 at high frequencies.

Time constant form It is also interesting to consider another form of pole uncertainty, namely that associated with the time constant:

$$G_p(s) = \frac{1}{\tau_p s + 1} G_0(s); \quad \tau_{\min} \le \tau_p \le \tau_{\max}$$

The corresponding uncertainty weight is

$$w_{iI}(s) = \frac{r_\tau \bar{\tau} s}{1 + \bar{\tau} s}$$

This results in uncertainty in the pole location, but here the uncertainty affects the model at high frequency.

b Parametric Zero Uncertainty

Consider zero uncertainty in the "time constant" form as in:

$$G_p(s) = (1 + \tau_p s) G_0(s); \quad \tau_{\min} \le \tau_p \le \tau_{\max}$$

This set of plants may be written as multiplicative uncertainty with:

$$w_I(s) = \frac{r_\tau \bar{\tau} s}{1 + \bar{\tau} s}$$

The magnitude $|w_I(j\omega)|$ is small at low frequencies and approaches r_τ at high frequencies. For cases with $r_\tau > 1$ we allow the zero to cross from the LHP to the RHP.

c Parametric State-Space Uncertainty

A general procedure for handling parametric uncertainty which is more suited for numerical calculations, is parametric state-space uncertainty. Consider an uncertain state-space model:

$$\dot{x} = A_p x + B_p u$$
$$y = C_p x + D_p u$$

Assume that the underlying cause for the uncertainty is uncertainty in some real parameters $\delta_1, \delta_2, \ldots$ and assume that the state space matrices depends linearly on these parameters:

$$A_p = A + \sum \delta_i A_i; \quad B_p = B + \sum \delta_i B_i$$
$$C_p = C + \sum \delta_i C_i; \quad D_p = D + \sum \delta_i D_i$$

where A, B, C and D model the nominal system.
We can collect the perturbations δ_i in a large diagonal matrix Δ with the real δ_i's along its diagonal:

$$A_p = A + \sum \delta_i A_i = A + W_2 \Delta W_1$$

In the transfer function form:

$$\begin{aligned}(sI - A_p)^{-1} &= (sI - A - W_2\Delta W_1)^{-1} \\ &= (I - \Phi(s) W_2 \Delta W_1)^{-1} \Phi(s)\end{aligned}$$

with $\Phi(s) \triangleq (sI - A)^{-1}$.
This is illustrated in the block diagram of Fig. 25, which is in the form of an inverse additive perturbation.

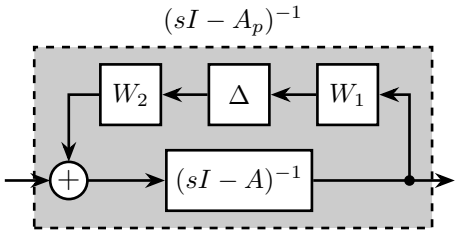


Figure 25 – *Uncertainty in state space A-matrix*

7.8 Conclusion

Model uncertainty for SISO systems can be represented in the frequency domain using complex norm-bounded perturbations $\|\Delta\|_\infty \le 1$.

Requirements of robust stability for the case of multiplicative complex uncertainty imposes an upper bound on the allowed complementary sensitivity, $|w_I T| < 1, \ \forall\omega$.

Similarly, the inverse multiplicative uncertainty imposes an upper bound on the sensitivity, $|w_{iI} S| < 1, \ \forall\omega$.

We also derived a condition for robust performance with multiplicative uncertainty, $|w_P S| + |w_I T| < 1, \ \forall\omega$.

8 Robust Stability and Performance Analysis

8.1 General Control Configuration with Uncertainty

The starting point for our robustness analysis is a system representation in which the uncertain perturbations are "pulled out" into a **block diagonal matrix**

$$\Delta = \text{diag}\{\Delta_i\} = \begin{bmatrix} \Delta_1 & & \\ & \ddots & \\ & & \Delta_i \\ & & & \ddots \end{bmatrix}$$

where each Δ_i represents a **specific source of uncertainty**, e.g. input uncertainty Δ_I or parametric uncertainty δ_i.
If we also pull out the controller K, we get the generalized plant P as shown in Fig. 26. This form is useful for controller synthesis.

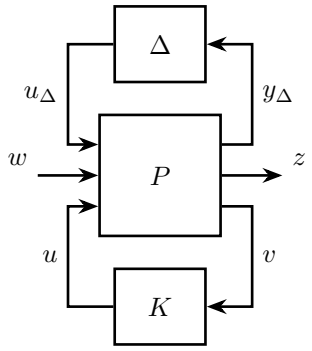


Figure 26 – *General control configuration used for controller synthesis*

If the controller is given and we want to analyze the uncertain system, we use the $N\Delta$-structure in Fig. 27.

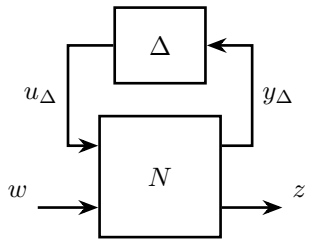


Figure 27 – *$N\Delta$-structure for robust performance analysis*

N is related to P and K by a **lower LFT**

$$\begin{aligned} N &= F_l(P, K) \\ &\triangleq P_{11} + P_{12}K(I - P_{22}K)^{-1}P_{21} \end{aligned}$$

Similarly, the uncertain closed-loop transfer function from w to z, is related to N and Δ by an **upper LFT**

$$\begin{aligned} F &= F_u(N, \Delta) \\ &\triangleq N_{22} + N_{21}\Delta(I - N_{11}\Delta)^{-1}N_{12} \end{aligned}$$

To analyze robust stability of F, we can rearrange the system into the $M\Delta$-structure shown in Fig. 28 where $M = N_{11}$ is the transfer function from the output to the input of the perturbations.

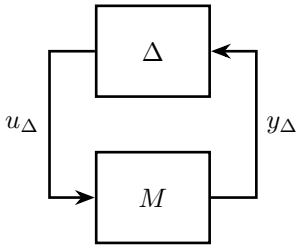


Figure 28 – *$M\Delta$-structure for robust stability analysis*

8.2 Representing Uncertainty

Each individual perturbation is assumed to be **stable and normalized**:

$$\overline{\sigma}(\Delta_i(j\omega)) \leq 1 \quad \forall\omega$$

As the maximum singular value of a block diagonal matrix is equal to the largest of the maximum singular values of the individual blocks, it then follows for $\Delta = \text{diag}\{\Delta_i\}$ that

$$\overline{\sigma}(\Delta_i(j\omega)) \leq 1 \quad \forall\omega, \forall i \quad \Leftrightarrow \quad \boxed{\|\Delta\|_\infty \leq 1}$$

a Differences Between SISO and MIMO Systems

The main difference between SISO and MIMO systems is the concept of directions which is only relevant in the latter. As a consequence, MIMO systems may experience **much larger sensitivity to uncertainty** than SISO systems.

b Parametric Uncertainty

The representation of parametric uncertainty for MIMO systems is the same as for SISO systems. However, the inclusion of parametric uncertainty may be more significant for MIMO plants because it offers a simple method of representing uncertain transfer function elements.

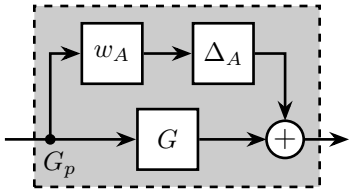
c Unstructured Uncertainty

Unstructured perturbations are often used to get a simple uncertainty model. We here define unstructured uncertainty as the use of a "full" complex perturbation matrix Δ, usually with dimensions compatible with those of the plant, where at each frequency any $\Delta(j\omega)$ satisfying $\overline{\sigma}(\Delta(j\omega)) < 1$ is allowed.

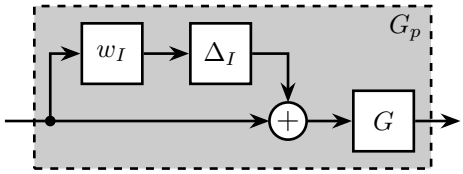
Three common forms of **feedforward unstructured uncertainty** are shown Fig. 29: additive uncertainty, multiplicative input uncertainty and multiplicative output uncertainty.

Feedforward unstructured uncertainty

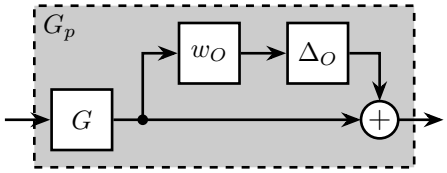
$$\begin{aligned} \Pi_A: &\quad G_p = G + E_A; && E_a = w_A \Delta_a \\ \Pi_I: &\quad G_p = G(I + E_I); && E_I = w_I \Delta_I \\ \Pi_O: &\quad G_p = (I + E_O)G; && E_O = w_O \Delta_O \end{aligned}$$



(a) – *Additive uncertainty*



(b) – *Multiplicative input uncertainty*



(c) – *Multiplicative output uncertainty*

Figure 29 – *Common feedforward unstructured uncertainty*

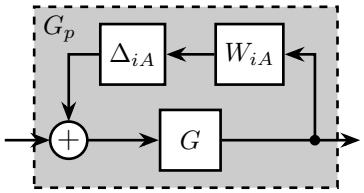
In Fig. 30, three **feedback or inverse unstructured uncertainty** forms are shown: inverse additive uncertainty, inverse multiplicative input uncertainty and inverse multiplicative output uncertainty.

Feedback unstructured uncertainty

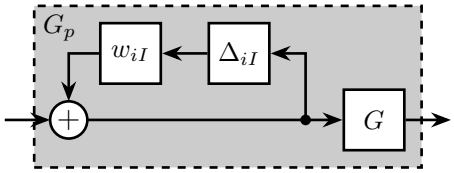
$$\begin{aligned} \Pi_{iA}: &\quad G_p = G(I - E_{iA}G)^{-1}; && E_{ia} = w_{iA}\Delta_{ia} \\ \Pi_{iI}: &\quad G_p = G(I - E_{iI})^{-1}; && E_{iI} = w_{iI}\Delta_{iI} \\ \Pi_{iO}: &\quad G_p = (I - E_{iO})^{-1}G; && E_{iO} = w_{iO}\Delta_{iO} \end{aligned}$$

Lumping uncertainty into a single perturbation For SISO systems, we usually lump multiple sources of uncertainty into a single complex perturbation; often in the multiplicative form. This may be also done for MIMO systems, but then it makes a difference whether the perturbation is at the input or the output.

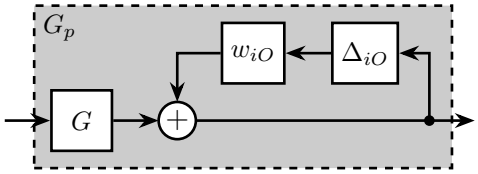
Since **output uncertainty is frequently less restrictive than input uncertainty in terms of control performance**, we first attempt to lump the uncertainty at the output. For example, a set of plant Π may



(a) – *Inverse additive uncertainty*



(b) – *Inverse multiplicative input uncertainty*



(c) – *Inverse multiplicative output uncertainty*

Figure 30 – *Common feedback unstructured uncertainty*

be represented by multiplicative output uncertainty with a scalar weight $w_O(s)$ using

$$G_p = (I + w_O \Delta_O)G, \quad \|\Delta_O\|_\infty \leq 1$$

where

$$l_O(\omega) = \max_{G_p \in \Pi} \overline{\sigma}\left((G_p - G)G^{-1}\right);\ |w_O(j\omega)| \geq l_O(\omega),\ \forall \omega$$

If the resulting uncertainty weight is reasonable and the analysis shows that robust stability and performance may be achieve, then this lumping of uncertainty at the output is fine. If this is not the case, then one may try to lump the uncertainty at the input instead, using multiplicative input uncertainty with a scalar weight,

$$G_p = G(I + w_I \Delta_I), \quad \|\Delta_I\|_\infty \leq 1$$

where

$$l_I(\omega) = \max_{G_p \in \Pi} \overline{\sigma}\left(G^{-1}(G_p - G)\right);\ |w_I(j\omega)| \geq l_I(\omega),\ \forall \omega$$

However, in many cases, this approach of lumping uncertainty either at the output or the input does **not** work well because **it usually introduces additional plants** that were not present in the original set.

Conclusion Ideally, we would like to lump several sources of uncertainty into a single perturbation to get a simple uncertainty description. Often an unstructured multiplicative output perturbation is used. However, we should be careful about doing this, at least for plants with a large condition number. In such cases we may have to represent the uncertainty as it occurs physically (at the input, in the elements, etc.) thereby generating several perturbations.

d Diagonal Uncertainty

By "diagonal uncertainty" we mean that the perturbation is a complex diagonal matrix

$$\Delta(s) = \text{diag}\{\delta_i(s)\}; \quad |\delta_i(j\omega)| \leq 1, \ \forall\omega, \forall i$$

Diagonal uncertainty usually arises from a consideration of uncertainty or neglected dynamics in the **individual input or output channels**. This type of diagonal uncertainty is **always present**.

Example - Input channel uncertainty

Let us consider uncertainty in the input channels. With each input u_i, there is a physical system (amplifier, actuator, etc.) which based on the controller output signal u_i, generates a physical plant input m_i

$$m_i = h_i(s)u_i$$

The scalar transfer function $h_i(s)$ is often absorbed into the plant model $G(s)$. We can represent its uncertainty as multiplicative uncertainty

$$h_{pi}(s) = h_i(s)(1+w_{Ii}(s)\delta_i(s)); \quad |\delta_i(j\omega)| \leq 1, \forall\omega$$

which after combining all input channels results in diagonal input uncertainty for the plant

$$G_p(s) = G(I + W_I\Delta_I) \text{ with } \Delta_I = \text{diag}\{\delta_i\}$$
$$W_I = \text{diag}\{w_{Ii}\}$$

Normally, we would represent the uncertainty in each input or output channel using a simple weight in the form

$$w(s) = \frac{\tau s + r_0}{(\tau/r_\infty)s + 1}$$

where r_0 is the relative uncertainty at steady-state, $1/\tau$ is the frequency where the relative uncertainty reaches 100 %, and r_∞ is the magnitude of the weight at high frequencies.

Diagonal input uncertainty should always be considered because:

- it is **always** present and a system which is sensitive to this uncertainty will not work in practice
- it often **restrict achievable performance** with multivariable control

8.3 Obtaining P, N and M

Let's consider the feedback system with multiplicative input uncertainty Δ_I shown Fig. 31. W_I is a normalization weight for the uncertainty and W_P is a performance weight.
We want to derive the generalized plant P which has inputs $[u_\Delta,\ w,\ u]^T$ and outputs $[y_\Delta,\ z,\ v]^T$.

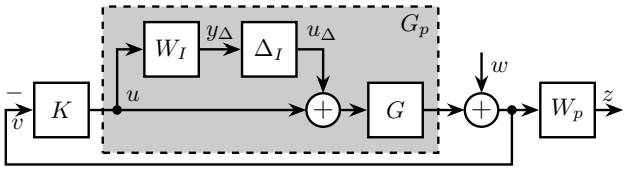


Figure 31 – *System with multiplicative input uncertainty and performance measured at the output*

By breaking the loop before and after K and Δ_I, we get

$$P = \begin{bmatrix} 0 & 0 & W_I \\ W_PG & W_P & W_PG \\ -G & -I & -G \end{bmatrix}$$

Next, we want to derive the matrix N. We fist partition P to be compatible with K:

$$P_{11} = \begin{bmatrix} 0 & 0 \\ GW_P & W_P \end{bmatrix}, \quad P_{12} = \begin{bmatrix} W_I \\ GW_P \end{bmatrix}$$
$$P_{21} = \begin{bmatrix} G & -1 \end{bmatrix}, \quad P_{22} = -G$$

and then we find N using $N = F_l(P, K)$.

8.4 Definitions of Robust Stability and Robust Performance

The next step is to check whether we have stability and acceptable performance for all plant in the set:

1. **Robust stability analysis**: with a given controller K we determine whether the system remains stable for all plants in the uncertainty set
2. **Robust performance analysis**: is RS is satisfied, we determine how "large" the transfer function from exogenous inputs w to outputs z may be for all plants in the uncertainty set

We have $z = F(\Delta) \cdot w$ with

$$F = F_u(N, \Delta)$$
$$\triangleq N_{22} + N_{21}\Delta(I - N_{11}\Delta)^{-1}N_{12}$$

We here use $\mathcal{H}_\infty$ norm to define performance and require for RP that $\|F(\Delta)\|_\infty \leq 1$ for all allowed Δ. A typical choice is $F = w_P S_P$ where w_P is the performance weight and S_P represents the set of perturbed sensitivity functions.

Stability and Performance Requirements

In terms of the $N\Delta$-structure, our requirements for stability and performance can be summarized as follows:

$$\begin{aligned}
\text{NS} &\overset{\text{def}}{\iff} N \text{ is internally stable} \\
\text{NP} &\overset{\text{def}}{\iff} \text{NS and } \|N_{22}\|_\infty < 1 \\
\text{RS} &\overset{\text{def}}{\iff} \text{NS and } F = F_u(N,\Delta) \text{ is stable } \forall\Delta \\
\text{RP} &\overset{\text{def}}{\iff} \text{NS and } \|F\|_\infty < 1, \quad \forall\Delta, \|\Delta\|_\infty \le 1
\end{aligned}$$

8.5 Robust Stability for the $M\Delta$-structure

Consider the uncertain $N\Delta$-system for which the transfer function from w to z is

$$F_u(N,\Delta) = N_{22} + N_{21}\Delta(I - N_{11}\Delta)^{-1}N_{12}$$

Suppose that the system is nominally stable (with $\Delta = 0$) that is N is stable. We also assume that Δ is stable. We then see from the above equation that the **only possible source of instability** is the feedback term $(I - N_{11}\Delta)^{-1}$. Thus, when we have nominal stability, the stability of the $N\Delta$-structure is equivalent to the stability of the $M\Delta$-structure where $M = N_{11}$.
We thus need to derive conditions for checking the stability of the $M\Delta$-structure.

Determinant Stability Condition

Assume that the nominal system $M(s)$ and the perturbations $\Delta(s)$ are stable. Consider the convex set of perturbations Δ, such that if Δ' is an allowed perturbation then so is $c\Delta'$ where c is any **real** scalar such that $|c| \le 1$. Then the $M\Delta$-structure is stable for all allowed perturbations **if and only if** the Nyquist plot of $\det(I - M\Delta(s))$ does not encircle the origin, $\forall\Delta$:

$$\det(I - M\Delta(j\omega)) \neq 0, \quad \forall\omega, \forall\Delta \qquad (50)$$

Spectral Radius Condition

Assume that the nominal system $M(s)$ and the perturbations $\Delta(s)$ are stable. Consider the class of perturbations, Δ, such that if Δ' is an allowed perturbation, then so is $c\Delta'$ where c is any **complex** scalar such that $|c| \le 1$. Then the $M\Delta$-structure is stable for all allowed perturbations **if and only if**:

$$\begin{aligned}
&\rho(M\Delta(j\omega)) < 1, \quad \forall\omega, \forall\Delta \\
\Leftrightarrow \quad &\max_\Delta \rho(M\Delta(j\omega)) < 1, \quad \forall\omega
\end{aligned} \qquad (51)$$

8.6 RS for Complex Unstructured Uncertainty

Let Δ be the set of all complex matrices such that $\overline{\sigma}(\Delta) \le 1$ ($\|\Delta\|_\infty \le 1$). This is often referred to as **unstructured uncertainty** or as full-block complex perturbation uncertainty. Then we have

$$\begin{aligned}
\max_\Delta \rho(M\Delta) &= \max_\Delta \overline{\sigma}(M\Delta) \\
&= \max_\Delta \overline{\sigma}(\Delta)\overline{\sigma}(M) \\
&= \overline{\sigma}(M)
\end{aligned}$$

RS for unstructured perturbations

Assume that the nominal system $M(s)$ is stable and that the perturbations $\Delta(s)$ are stable. Then the $M\Delta$-system is stable for all perturbations Δ satisfying $\|\Delta\|_\infty \le 1$ if and only if

$$\overline{\sigma}(M(j\omega)) < 1 \ \forall\omega \quad \Leftrightarrow \quad \|M\|_\infty < 1 \qquad (52)$$

a Application of the Unstructured RS-condition

We will now present necessary and sufficient conditions for robust stability for each of the six single unstructured perturbations in Figs 29 and 30 with

$$E = W_2\Delta W_1, \quad \|\Delta\|_\infty \le 1$$

To derive the matrix M we simply "isolate" the perturbation, and determine the transfer function matrix

$$M = W_1 M_0 W_2$$

from the output to the input of the perturbation, where M_0 for each of the six cases is given by

$$\begin{aligned}
G_p = G + E_A: &\qquad M_0 = K(I + GK)^{-1} = KS \\
G_p = G(I + E_I): &\qquad M_0 = K(I + GK)^{-1}G = T_I \\
G_p = (I + E_O)G: &\qquad M_0 = GK(I + GK)^{-1} = T \\
G_p = G(I - E_{iA}G)^{-1}: &\qquad M_0 = (I + GK)^{-1}G = SG \\
G_p = G(I - E_{iI})^{-1}: &\qquad M_0 = (I + KG)^{-1} = S_I \\
G_p = (I - E_{iO})^{-1}G: &\qquad M_0 = (I + GK)^{-1} = S
\end{aligned}$$

Using the theorem to check RS for unstructured perturbations

$$\text{RS} \quad \Leftrightarrow \quad \|W_1 M_0 W_2(j\omega)\|_\infty < 1, \ \forall\omega$$

For instance, for feedforward input uncertainty, we get

$$\text{RS } \forall G_p = G(I + w_I\Delta_I), \|\Delta_I\|_\infty \le 1 \Leftrightarrow \|w_I T_I\|_\infty < 1$$

In general, **the unstructured uncertainty descriptions in terms of a single perturbation are not "tight"** (in the sense that at each frequency all complex

perturbations satisfying $\overline{\sigma}(\Delta(j\omega)) \leq 1$ may not be possible in practice). Thus, the above RS-conditions are often **conservative**. In order to get tighter condition we must use a tighter uncertainty description in terms of a block-diagonal Δ.

b RS for Coprime Factor Uncertainty

Robust stability bound in terms of the $\mathcal{H}_\infty$ norm (RS $\Leftrightarrow$ $\|M\|_\infty < 1$) are in general only tight when there is a single full perturbation block. An "exception" to this is when the uncertainty blocks enter or exit from the same location in the block diagram, because they can then be stacked on top of each other or side-by-side, in an overall Δ which is then full matrix.
One important uncertainty description that falls into this category is the **coprime uncertainty description** shown in Fig. 32, for which the set of plants is

$$G_p = (M_l + \Delta_M)^{-1}(Nl + \Delta_N), \quad \|[\Delta_N,\ \Delta_N]\|_\infty \leq \epsilon$$

Where $G = M_l^{-1}N_l$ is a left coprime factorization of the nominal plant.
This uncertainty description is surprisingly **general**, it allows both zeros and poles to cross into the right-half plane, and has proven to be very useful in applications.

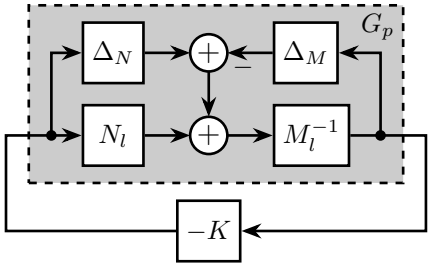


Figure 32 – *Coprime Uncertainty*

Since we have no weights on the perturbations, it is reasonable to use a normalized coprime factorization of the nominal plant. In any case, to test for RS we can rearrange the block diagram to match the $M\Delta$-structure with

$$\Delta = [\Delta_N,\ \Delta_M]; \quad M = -\begin{bmatrix} K \\ I \end{bmatrix}(I + GK)^{-1}M_l^{-1}$$

And we get

$$\text{RS } \forall\ \|\Delta_N,\ \Delta_M\|_\infty \leq \epsilon \quad \Leftrightarrow \quad \|M\|_\infty < 1/\epsilon$$

The coprime uncertainty description provides a good **generic uncertainty description** for cases where we do not use any specific a priori uncertainty information. Note that the uncertainty magnitude is ϵ, so it is not normalized to be less than 1 in this case. This is because this uncertainty description is most often used in a controller design procedure where the objective is to maximize the magnitude of the uncertainty ϵ such that RS is maintained.

8.7 RS with Structured Uncertainty: Motivation

Consider now the presence of structured uncertainty, where $\Delta = \text{diag}\{\Delta_i\}$ is block-diagonal. To test for robust stability, we rearrange the system into the $M\Delta$-structure and we have

$$\text{RS if } \overline{\sigma}(M(j\omega)) < 1,\ \forall\omega$$

We have here written "if" rather than "if and only if" since this condition is only sufficient for RS when Δ has "no structure". The question is whether we can take advantage of the fact that $\Delta = \text{diag}\{\Delta_i\}$ is structured to obtain an RS-condition which is tighter. On idea is to make use of the fact that stability must be independent of scaling.
To this effect, introduce the block-diagonal scaling matrix

$$D = \text{diag}\{d_i I_i\}$$

where d_i is a scalar and I_i is an identity matrix of the same dimension as the i'th perturbation block Δ_i.
Now rescale the inputs and outputs of M and Δ by inserting the matrices D and D^{-1} on both sides as shown in Fig. 33. This clearly has no effect on stability.

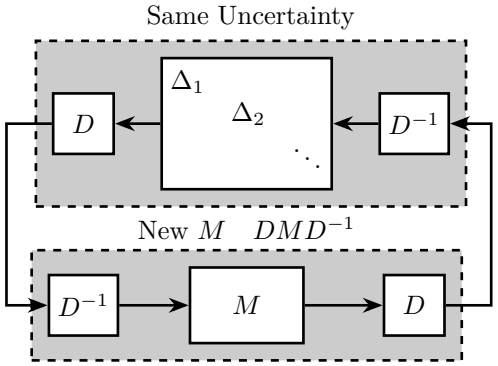


Figure 33 – *Use of block-diagonal scalings,* $\Delta D = D\Delta$

Note that with the chosen form for the scalings we have for each perturbation block $\Delta_i = d_i\Delta_i d_i^{-1}$, that is we have $\Delta = D\Delta D^{-1}$.
This means that we have

$$\text{RS if } \overline{\sigma}(DM(j\omega)D^{-1}) < 1,\ \forall\omega$$

Improved RS-condition

This applies for any D, and therefore the "most improved" (least conservative) RS-condition is obtained by minimizing at each frequency the scaled singular value and we have

$$\text{RS if } \min_{D(\omega)\in\mathcal{D}} \overline{\sigma}(D(\omega)M(j\omega)D(\omega)^{-1}) < 1,\ \forall\omega$$

where $\mathcal{D}$ is the set of block-diagonal matrices whose structure is compatible to that of Δ, i.e, $\Delta D = D\Delta$.

When Δ is a full matrix, we must select $D = dI$ and we have $\overline{\sigma}(DMD^{-1}) = \overline{\sigma}(M)$, and we cannot improve the RS-condition. However, when Δ has structure, we get more degrees of freedom in D and $\overline{\sigma}(DMD^{-1})$ may be significantly smaller than $\overline{\sigma}(M)$.

8.8 The Structured Singular Value

a Definition

The structured singular value μ is a function which provides a **generalization of the singular value** $\overline{\sigma}$ and the **spectral radius** ρ. We will use μ to get necessary and sufficient conditions for robust stability and also for robust performance.
μ can be explained as follow:

> Find the smallest structured Δ (measured in terms of $\overline{\sigma}(\Delta)$) which makes the matrix $I - M\Delta$ singular; then $\mu(M) = 1/\overline{\sigma}(\Delta)$.

Mathematically

$$\mu(M)^{-1} \triangleq \min_{\Delta}\{\overline{\sigma}(\Delta) | \det(I - M\Delta) = 0 \text{ for struct. } \Delta\}$$

Clearly, $\mu(M)$ depends not only on M but also on the **allowed structure** for Δ. This is sometimes shown explicitly by using the notation $\mu_{\Delta}(M)$.

The above definition of μ involves varying $\overline{\sigma}(\Delta)$. However, we prefer to normalize Δ such that $\overline{\sigma}(\Delta) \leq 1$. We can do that by scaling Δ by a factor k_m, and looking for the smallest k_m which makes the matrix $I - k_m M\Delta$ singular. μ is then the reciprocal of this small k_m: $\mu = 1/k_m$. This results in the following alternative definition of μ.

Definition - Structured Singular Value

Let M be a given complex matrix and let $\Delta = \text{diag}\{\Delta_i\}$ denote a set of complex matrices with $\overline{\sigma}(\Delta) \leq 1$ and with a given block-diagonal structure. The real non-negative function $\mu(M)$, called the structured singular value, is defined by

$$\mu(M) \triangleq (\min\{k_m | \det(I - k_m M\Delta) = 0 \text{ for structured } \Delta, \overline{\sigma}(\Delta) \leq 1\})^{-1}$$

If no such structured Δ exists then $\mu(M) = 0$

A value of $\mu = 1$ means that there exists a perturbation with $\overline{\sigma}(\Delta) = 1$ which is just large enough to make $I - M\Delta$ singular.
A larger value of μ is "bad" as it means that a smaller perturbation makes $I - M\Delta$ singular, whereas a smaller value of μ is "good".

b Remarks on the Definition of μ

1. The structured singular value was introduced by Doyle while at the same time, Safonov introduced the **Multivariable Stability Margin** k_m for a diagonally perturbed system as the inverse of μ, that is $k_m(M) = \mu(M)^{-1}$.
2. Note that with $k_m = 0$ we obtain $I - k_m M\Delta = I$ which is clearly non-singular. Thus, one possible way to obtain μ numerically, is to start with $k_m = 0$, and gradually increase k_m until we first find an allowed Δ with $\overline{\sigma}(\Delta) = 1$ such that $I - k_m M\Delta$ is singular.

c Properties of μ for Real and Complex Δ

1. $\mu(\alpha M) = |\alpha|\,\mu(M)$ for any real scalar α
2. Let $\Delta = \text{diag}\{\Delta_1, \Delta_2\}$ be a block-diagonal perturbation and let M be partitioned accordingly. Then

$$\mu_{\Delta} \geq \max\{\mu_{\Delta_1}(M_{11}), \mu_{\Delta_2}(M_{22})\}$$

d Properties of μ for Complex Perturbations Δ

1. For complex perturbations Δ with $\overline{\sigma}(\Delta) \leq 1$

$$\mu(M) = \max_{\Delta, \overline{\sigma}(\Delta) \leq 1} \rho(M\Delta) \tag{53}$$

2. $\mu(\alpha M) = |\alpha|\,\mu(M)$ for any (complex) scalar α
3. For a full block complex perturbation Δ

$$\mu(M) = \overline{\sigma}(M)$$

4. μ for complex perturbations is bounded by the spectral radius and the singular value

$$\rho(M) \leq \mu(M) \leq \overline{\sigma}(M) \tag{54}$$

5. **Improved lower bound**. Defined $\mathcal{U}$ as the set of all unitary matrices U with the same block diagonal structure as Δ. Then for complex Δ

$$\mu(M) = \max_{U \in \mathcal{U}} \rho(MU) \tag{55}$$

6. **Improved upper bound**. Defined $\mathcal{D}$ as the set of all unitary matrices D that commute with Δ. Then

$$\mu(M) = \min_{D \in \mathcal{D}} \overline{\sigma}(DMD^{-1}) \tag{56}$$

8.9 Robust Stability with Structured Uncertainty

Consider stability of the $M\Delta$-structure for the case where Δ is a set of norm-bounded block-diagonal perturbations. From the determinant stability condition which applies to both complex and real perturbations, we get

$$\text{RS} \;\Leftrightarrow\; \det(I - M\Delta(j\omega)) \neq 0,\ \forall\omega,\ \forall\Delta,\ \|\Delta\|_\infty \leq 1$$

The problem is that this is only a "yes/no" condition. To find the factor k_m by which the system is robustly stable, we scale the uncertainty Δ by k_m, and look for the smallest k_m which yields "borderline instability", namely

$$\det(I - k_m M\Delta) = 0$$

From the definition of μ, this value is $k_m = 1/\mu(M)$, and we obtain the following necessary and sufficient condition for robust stability.

RS for block-diagonal perturbations

Assume that the nominal system M and the perturbations Δ are stable. Then the $M\Delta$-system is stable for all allowed perturbations with $\overline{\sigma}(\Delta) \leq 1,\ \forall\omega$ if on only if

$$\mu(M(j\omega)) < 1,\ \forall\omega \qquad (57)$$

What do $\mu \neq 1$ and skewed-μ mean? A value of $\mu = 1.1$ for robust stability means that **all** the uncertainty blocks must be decreased in magnitude by a factor 1.1 in order to guarantee stability.

But if we want to keep some of the uncertainty blocks fixed, how large can one particular source of uncertainty be before we get instability? We define this value as $1/\mu^s$, where μ^s is called skewed-μ. We may view $\mu^s(M)$ as a generalization of $\mu(M)$.

Example

Let $\Delta = \text{diag}\{\Delta_1, \Delta_2\}$ and assume we have fixed $\|\Delta_1\| \leq 1$ and we want to find how large Δ_2 can be before we get instability. The solution is to select

$$K_m = \begin{bmatrix} I & 0 \\ 0 & k_m I \end{bmatrix}$$

and look at each frequency for the smallest value of k_m which makes $\det(I - K_m M\Delta) = 0$ and we have that skewed-μ is

$$\mu^s(M) \triangleq 1/k_m$$

Note that to compute skewed-μ we must first define which part of the perturbations is to be constant.

8.10 Robust Performance

a Testing RP using μ

To test for RP, we first "pull out" the uncertain perturbations and rearrange the uncertain system into the $N\Delta$-form. Our RP-requirement, is that the $\mathcal{H}_\infty$ norm of the transfer function $F = F_u(N, \Delta)$ remains less than 1 for all allowed perturbations. This may be tested exactly by computing $\mu(N)$.

Theorem - Robust performance

Rearrange the uncertain system into the $N\Delta$-structure. Assume nominal stability such that N is stable. Then

$$\begin{aligned} \text{RP} &\overset{\text{def}}{\Longleftrightarrow} \|F\|_\infty = \|F_u(N,\Delta)\|_\infty < 1,\ \forall\|\Delta\|_\infty < 1 \\ &\Longleftrightarrow \mu_{\hat{\Delta}}(N(j\omega)) < 1,\ \forall\omega \end{aligned}$$

where μ is computed with respect to the structure

$$\hat{\Delta} = \begin{bmatrix} \Delta & 0 \\ 0 & \Delta_P \end{bmatrix}$$

and Δ_P is a full complex perturbation with the same dimensions as F^T.

Some remarks on the theorem:

1. Condition $\mu_{\hat{\Delta}}(N(j\omega)) < 1,\ \forall\omega$ allows us to test if $\|F\|_\infty < 1$ for all possible Δ without having to test each Δ individually. Essential, μ is defined such that it directly addresses the worst case
2. The μ-condition for RP involves the enlarged perturbation $\hat{\Delta} = \text{diag}\{\Delta, \Delta_P\}$. Here Δ, which itself may be a block diagonal matrix, represents the true uncertainty, whereas Δ_P is a full complex matrix stemming from the $\mathcal{H}_\infty$ norm performance specification
3. Since $\hat{\Delta}$ always has structure, the use of $\mathcal{H}_\infty$ norm, $\|N\|_\infty < 1$, is generally conservative for robust performance

b Summary of μ-conditions for NP, RS and RP

Conditions for NS NP RS and RP

Rearrange the uncertain system into the $N\Delta$-structure where the block-diagonal perturbation satisfy $\|\Delta\|_\infty \leq 1$. Introduce

$$F = F_u(N,\Delta) = N_{22} + N_{21}\Delta(I - N_{11}\Delta)^{-1}N_{12}$$

Let the performance requirement be $\|F\|_\infty \leq 1$.

$$\begin{aligned}
\text{NS} &\Leftrightarrow N \text{ (internally) stable} \\
\text{NP} &\Leftrightarrow \text{NS and } \overline{\sigma}(N_{22}) = \mu_{\Delta_P} < 1, \ \forall\omega \\
\text{RS} &\Leftrightarrow \text{NS and } \mu_\Delta(N_{11}) < 1, \ \forall\omega \\
\text{RP} &\Leftrightarrow \text{NS and } \mu_{\tilde{\Delta}}(N) < 1, \ \forall\omega, \ \tilde{\Delta} = \begin{bmatrix} \Delta & 0 \\ 0 & \Delta_P \end{bmatrix}
\end{aligned}$$

Here Δ is a block-diagonal matrix, whereas Δ_P is always a full complex matrix.
Although the structured singular value is not a norm, it is sometimes convenient to refer to the peak μ-value as the "Δ-norm". For a stable rational transfer matrix $H(s)$, with an associated block structure Δ, we therefore define

$$\|H(s)\|_\Delta \triangleq \max_\omega \mu_\Delta(H(j\omega)) \tag{58}$$

For a nominal stable system, we then have

$$\begin{aligned}
\text{NP} &\Leftrightarrow \|N_{22}\|_\infty < 1 \\
\text{RS} &\Leftrightarrow \|N_{11}\|_\Delta < 1 \\
\text{RP} &\Leftrightarrow \|N\|_{\tilde{\Delta}} < 1
\end{aligned}$$

c Worst-case Performance and Skewed-μ

Assume we have a system for which the peak μ-value for RP is 1.1. What does this mean? The definition of μ tells us that our RP-requirement would be satisfied exactly if we reduced **both** the performance requirement and the uncertainty by a factor of 1.1. So μ does not directly give us the worst-case performance $\max_\Delta \overline{\sigma}(F(\Delta))$.
To find the worst-case weighted performance for a given uncertainty, one needs to keep the magnitude of the perturbation fixed ($\overline{\sigma}(\Delta) \leq 1$), that is, **we must compute the skewed-μ** of N. We have, in this case

$$\max_{\overline{\sigma}(\Delta)\leq 1} \overline{\sigma}(F_l(N,\Delta)(j\omega)) = \mu^s(N(j\omega))$$

To find μ^s numerically, we scale the performance part of N by a factor $k_m = 1/\mu^s$ and iterate on k_m until $\mu = 1$. That is, at each frequency skewed-μ is the value $\mu^s(N)$ which solves

$$\mu(K_m N) = 1, \quad K_m = \begin{bmatrix} I & 0 \\ 0 & 1/\mu^s \end{bmatrix}$$

Note that μ underestimate how bad or good the actual worst case performance is. This follows because $\mu^s(N)$ is always further from 1 than $\mu(N)$.

8.11 Application: RP with Input Uncertainty

We will now consider in some detail the case of multiplicative input uncertainty with performance defined in terms of weighted sensitivity (Fig. 31).
The performance requirement is then

$$\text{RP} \quad \overset{\text{def}}{\Longleftrightarrow} \quad \left\|w_P(I + G_pK)^{-1}\right\|_\infty < 1, \quad \forall G_p$$

where the set of plant is given by

$$G_p = G(I + w_I\Delta_I), \quad \|\Delta_I\|_\infty \leq 1$$

Here $w_p(s)$ and $w_I(s)$ are scalar weights, so the performance objective is the same for all the outputs, and the uncertainty is the same for all the inputs.
In this section, we will:

1. Find the interconnection matrix N for this problem
2. Consider the SISO case, so that useful connections can be made with results for SISO systems
3. Consider a multivariable distillation process
4. Find some simple bounds on μ and discuss the role of the condition number
5. Make comparisons with the case where the uncertainty is located at the output

a Interconnection Matrix

On rearranging the system into the $N\Delta$-structure, we get

$$N = \begin{bmatrix} -w_I T_I & -w_I KS \\ w_p SG & w_p S \end{bmatrix} \tag{59}$$

where $T_I = KG(I + KG)^{-1}$, $S = (I + GK)^{-1}$. For simplicity, we can omit the negative signs.
For a given controller K we can now test for NS, NP, RS and RP.

b RP with Input Uncertainty for SISO System

For a SISO system with N as described above:

$$\begin{aligned}
&\text{NS} \Leftrightarrow S, \ SG, \ KS, \text{ and } T_I \text{ are stable} \\
&\text{NP} \Leftrightarrow |w_P S| < 1, \quad \forall\omega \\
&\text{RS} \Leftrightarrow |w_I T_I| < 1, \quad \forall\omega \\
&\text{RP} \Leftrightarrow |w_P S| + |w_I T_I| < 1, \quad \forall\omega
\end{aligned}$$

Robust performance optimization, in terms of weighted sensitivity with multiplicative uncertainty for a SISO system, thus involves minimizing the peak value of $\mu(N) = |w_I T| + |w_P S|$. This may be solved using DK-iteration. A closely related problem, which is easier to

solve is to minimize the peak value ($\mathcal{H}_\infty$ norm) of the mixed sensitivity matrix:

$$N_{\text{mix}} = \begin{bmatrix} w_P S \\ w_I T \end{bmatrix}$$

At each frequency, $\mu(N)$ differs from and $\bar{\sigma}(N_{\text{mix}})$ by at most a factor $\sqrt{2}$. Thus, minimizing $\|N_{\text{mix}}\|_\infty$ is close to optimizing robust performance in terms of $\mu(N)$.

c Robust Performance for 2×2 Distillation Process

Consider a distillation process and a corresponding inverse-based controller:

$$G(s) = \frac{1}{75s+1}\begin{bmatrix} 87.8 & -86.4 \\ 108.2 & -109.6 \end{bmatrix}; \quad K(s) = \frac{0.7}{s} G(s)^{-1}$$

The controller provides a nominally decoupled system:

$$L = lI, \; S = \epsilon I \text{ and } T = tI$$

where

$$l = \frac{0.7}{s}, \; \epsilon = \frac{s}{s+0.7}, \; t = \frac{0.7}{s+0.7}$$

The following weights for uncertainty and performance are used:

$$w_I(s) = \frac{s+0.2}{0.5s+1}; \quad w_P(s) = \frac{s/2+0.05}{s}$$

We now test for NS, NP, RS and RP.

NS with G and K as defined, we find that S, SG, KS and T_I are stable, so the system is nominally stable.

NP with the decoupling controller we have:

$$\bar{\sigma}(N_{22}) = \bar{\sigma}(w_P S) = \left| \frac{s/2+0.05}{s+0.7} \right|$$

and we see from Fig. 34 that the NP-condition is satisfied.

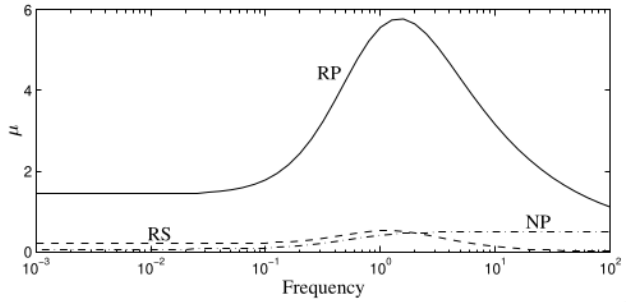


Figure 34 – *μ-plots for distillation process with decoupling controller*

RS In this case $w_I T_I = w_I T$ is a scalar times the identity matrix:

$$\mu_{\Delta_I}(w_I T_I) = |w_I t| = \left| 0.2 \frac{5s+1}{(0.5s+1)(1.43s+1)} \right|$$

and we see from Fig. 34 that RS is satisfied.
The peak value of $\mu_{\Delta_I}(M)$ is 0.53 meaning that we may increase the uncertainty by a factor of $1/0.53 = 1.89$ before the worst case uncertainty yields instability.

RP Although the system has good robustness margins and excellent nominal performance, the robust performance is poor. This is shown in Fig. 34 where the μ-curve for RP was computed numerically using $\mu_{\hat{\Delta}}(N)$, with $\hat{\Delta} = \text{diag}\{\Delta_I, \Delta_P\}$ and $\Delta_I = \text{diag}\{\delta_1, \delta_2\}$. The peak value is close to 6, meaning that even with 6 times less uncertainty, the weighted sensitivity will be about 6 times larger than what we require.

d Robust Performance and the Condition Number

We here consider the relationship between μ for RP and the condition number of the plant or of the controller. We consider unstructured multiplicative uncertainty (i.e. Δ_I is a full matrix) and performance is measured in terms of the weighted sensitivity. With N given by (59), we have:

$$\overbrace{\mu_{\hat{\Delta}}(N)}^{\text{RP}} \leq [\overbrace{\bar{\sigma}(w_I T_I)}^{\text{RS}} + \overbrace{\bar{\sigma}(w_P S)}^{\text{NP}}](1+\sqrt{k})$$

where k is taken as the smallest value between the condition number of the plant and of the controller:

$$k = \min(\gamma(G), \gamma(K))$$

We see that with a "round" controller (i.e. one with $\gamma(K) = 1$), there is less sensitivity to uncertainty. On the other hand, we would expect μ for RP to be large if we used an inverse-based controller for a plant with large condition number, since then $\gamma(K) = \gamma(G)$ is large.

e Comparison with Output Uncertainty

Consider output multiplicative uncertainty of magnitude $w_O(j\omega)$. In this case, we get the interconnection matrix

$$N = \begin{bmatrix} w_O T & w_O T \\ w_P S & w_P S \end{bmatrix}$$

and for any structure of the uncertainty, $\mu(N)$ is bounded as follows:

$$\bar{\sigma}\begin{bmatrix} w_O T \\ w_P S \end{bmatrix} \leq \overbrace{\mu(N)}^{\text{RP}} \leq \sqrt{2}\, \bar{\sigma} \underbrace{\overbrace{\begin{bmatrix} w_O T \\ w_P S \end{bmatrix}}^{\text{RS}}}_{\text{NP}}$$

This follows since the uncertainty and performance blocks both enter at the output and that the difference between bounding the combined perturbations $\bar{\sigma}[\Delta_O\ \Delta_P]$ and the individual perturbations $\bar{\sigma}(\Delta_O)$ and $\bar{\sigma}(\Delta_P)$ is at most a factor $\sqrt{2}$. Thus, we "automatically" achieve RP if we satisfy separately NP and RS. Multiplicative output uncertainty then poses no particular problem for performance.

8.12 μ-synthesis and DK-iteration

The structured singular value μ is a very powerful tool for the analysis of robust performance with a given controller. However, one may also seek to **find the controller that minimizes** a given μ-condition: this is the μ-synthesis problem.

a DK-iteration

At present, there is no direct method to synthesize a μ-optimal controller. However, for complex perturbations, a method known as **DK-iteration** is available. It combines $\mathcal{H}_\infty$ synthesis and μ-analysis and often yields good results.

The starting point is the upper bound on μ in terms of the scaled singular value

$$\mu(N) \leq \min_{D\in\mathcal{D}} \overline{\sigma}(DND^{-1}) \tag{60}$$

The idea is to find the controller that minimizes the peak value over frequency of this upper bound, namely

$$\min_K \left(\min_{D\in\mathcal{D}} \left\| DN(K)D^{-1} \right\|_\infty \right) \tag{61}$$

by alternating between minimizing $\left\| DN(K)D^{-1} \right\|_\infty$ with respect to either K or D (while holding the other fixed).

To start the iterations, one selects an initial stable rational transfer matrix $D(s)$ with appropriate structure. The identity matrix is often a good initial choice for D provided the system has been reasonably scaled for performance.

DK-Procedure

1. **K-step**. Synthesize an $\mathcal{H}_\infty$ controller for the scaled problem, $\min_K \left\| DN(K)D^{-1} \right\|_\infty$ with fixed $D(s)$
2. **D-step**. Find $D(j\omega)$ to minimize at each frequency $\overline{\sigma}(DND^{-1}(j\omega))$ with fixed N
3. Fit the magnitude of each element of $D(j\omega)$ to a stable and minimum phase transfer function $D(s)$ and go to step 1

The iteration may continue until satisfactory performance is achieve, $\left\| DND^{-1} \right\|_\infty < 1$ or until the $\mathcal{H}_\infty$ norm no longer decreases. One fundamental problem with this approach is that although each of the minimization steps are convex, **joint convexity is not guaranteed**. Therefore, the iterations may converge to a **local minimum**.

The order of the controller resulting from each iteration is equal to the number of the states in the plant $G(s)$ plus the number of states in the weights plus twice the number of state in $D(s)$. The obtain μ-optimal controller will usually be of **high order** and will have a flat μ-curve until some high frequency.

The DK-iteration depends heavily on optimal solutions for steps 1 and 2, and also on good fits in step 3. We usually **prefers to have a low-order fit** in step 3 as it will reduces the order of the $\mathcal{H}_\infty$ problem which usually improves the numerical properties of the optimization. In some cases, the iterations converge slowly, the μ-value can even increase. This may be caused by numerical problems and one may consider going back to the initial problem and **rescaling the inputs and outputs**.

b Adjusting the Performance Weight

If μ at a given frequency is different from 1, then the interpretation is that at this frequency we can tolerate $1/\mu$ times more uncertainty and still satisfy our performance objective with a margin of $1/\mu$. In μ-synthesis, the designer will usually adjust some parameter in the performance or uncertainty weights until the weight of the peak μ-value is close to 1.

Sometimes, uncertainty is fixed and we effectively optimize worst-cast performance by adjusting a parameter in the performance weight. Consider the performance weight

$$w_p(s) = \frac{s/M + \omega_B^*}{s + \omega_B^* A}$$

where we want to keep M constant and find the high achievable bandwidth frequency ω_B^*. The optimization problem becomes

$$\max |\omega_B^*| \quad \text{such that} \quad \mu(N) < 1,\ \forall\omega$$

where N, the interconnection matrix for the RP-problem, depends on ω_B^*. This may be implemented as an **outer loop around the DK-iteration**.

c Fixed Structure Controller

Sometimes it is desirable to find a low-order controller with a given structure. This may be achievable by numerical optimization where μ is minimized with respect to the controller parameters. This problem here is that the optimization is not generally convex in the parameters. Sometimes it helps to switch the optimization between minimizing the peak of μ and minimizing

the integral square deviation of μ away from k (i.e. $\|\mu(j\omega) - k\|_2$) where k is usually close to 1. The latter is an attempt to "flatten out" μ.

d Example: μ-synthesis with DK-iteration

For simplicity, we will consider again the case of multiplicative uncertainty and performance defined in terms of weighted sensitivity. The uncertainty weight $w_I I$ and performance weight $w_P I$ are shown graphically in Fig. 35.

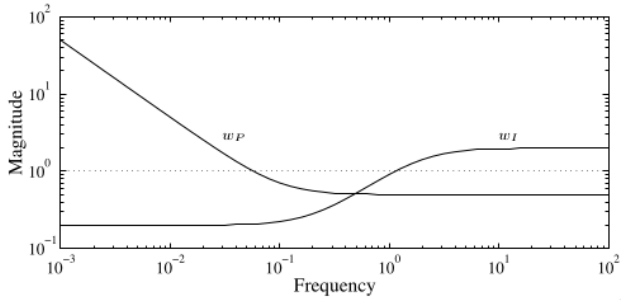


Figure 35 – *Uncertainty and performance weights*

The objective is to minimize the peak value of $\mu_{\tilde{\Delta}}(N)$, $\tilde{\Delta} = \text{diag}\{\Delta_I, \Delta_P\}$. Δ_I is a diagonal 2×2 matrix representing the diagonal input uncertainty and Δ_P is a full 2×2 matrix representing the performance specifications.
First, the generalized plant P is constructed which includes the plant model, the uncertainty weight and the performance weight. Then the block structure is defined, it consists of two 1×1 blocks to represent Δ_I and a 2×2 block to represent Δ_P. The scaling matrix D for DND^{-1} then has the structure $D = \text{diag}\{d_1, d_2, d_3 I_2\}$. We select $d_3 = 1$ and as initial scalings we select $d_1^0 = d_2^0 = 1$. P is then scaled with the matrix $\text{diag}\{D, I_2\}$ where I_2 is associated with the inputs and outputs from the controller (we do not want to scale the controller).

- Iteration No. 1. Step 1: with the initial scalings, the $\mathcal{H}_\infty$ synthesis produced a 6 state controller (2 states from the plant model and 2 from each of the weights). Step 2: the upper μ-bound is shown in Fig. 36. Step 3: the frequency dependent $d_1(\omega)$ and $d_2(\omega)$ from step 2 are fitted using a 4th order transfer function shown in Fig. 37
- Iteration No. 2. Step 1: with the 8 state scalings $D^1(s)$, the $\mathcal{H}_\infty$ synthesis gives a 22 state controller. Step 2: This controller gives a peak value of μ of 1.02. Step 3: the scalings are only slightly changed
- Iteration No. 3. Step 1: The $\mathcal{H}_\infty$ norm is only slightly reduced. We thus decide the stop the iterations.

The final μ-curves for NP, RS and RP with the controller K_3 are shown in Fig. 38. The objectives of RS and NP are easily satisfied. The peak value of μ is

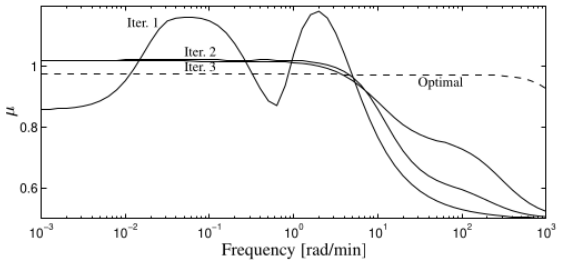


Figure 36 – *Change in μ during DK-iteration*

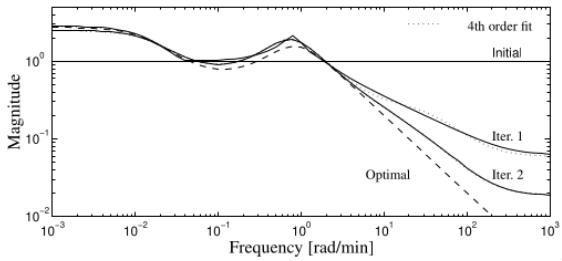


Figure 37 – *Change in D-scale d_1 during DK-iteration*

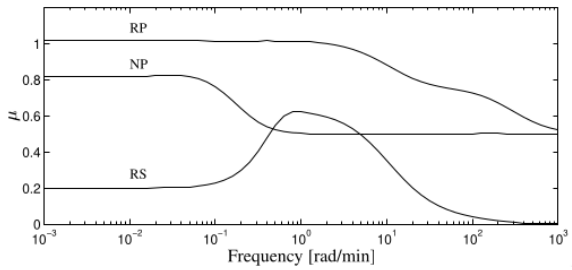


Figure 38 – *mu-plots with μ "optimal" controller K_3*

just slightly over 1, so the performance specification $\bar{\sigma}(w_P S_p) < 1$ is almost satisfied for all possible plants. To confirm that, 6 perturbed plants are used to compute the perturbed sensitivity functions shown in Fig. 39.

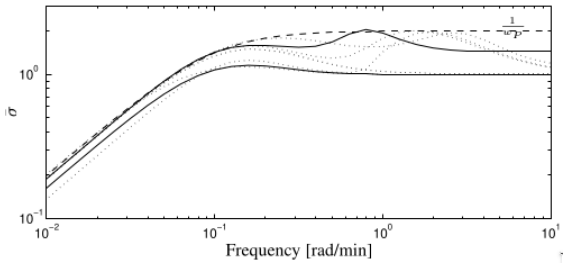


Figure 39 – *Perturbed sensitivity functions $\bar{\sigma}(S')$ using μ "optimal" controller K_3. Lower solid line: nominal plant. Upper solid line: worst-case plant*

8.13 Further Remarks on μ

For complex perturbations, the scaled singular value $\overline{\sigma}(DND^{-1})$ is a tight upper bound on $\mu(N)$ in most cases, and minimizing the upper bound $\left\|DND^{-1}\right\|_\infty$ form the basis for the DK-iteration.
The use of constant D-scales (D is not allowed to vary with frequency), provides a necessary and sufficient condition for robustness to arbitrary fast time varying linear uncertainty. While such perturbations are unlikely in a practical situation, we know that this controller will work very well even for rapid changes in the plant. Moreover, the use of constant D-scales make the computation of μ straightforward and solvable using LMIs.

8.14 Conclusion

We have discussed how to represent uncertainty and how to analyze its effect on stability (RS) and performance (RP) using the **structured singular value** μ.

To analyze robust stability of an uncertain system, we make use of the $M\Delta$-structure where M represents the transfer function for the "new" feedback part generated by the uncertainty. From the small gain theorem

$$RS \quad \Leftarrow \quad \overline{\sigma}(M) < 1, \ \forall\omega$$

which is tight (necessary and sufficient) for the special case where at each frequency any complex Δ satisfying $\overline{\sigma}(\Delta) \leq 1$ is allowed. More generally, the **tight condition is**

$$RP \quad \Leftrightarrow \quad \mu(M) < 1, \ \forall\omega$$

where $\mu(M)$ is the **structured singular value**. The calculation of μ makes use of the fact that Δ has a given block-diagonal structure, where certain blocks may also be real (e.g. to handle parametric uncertainty).

We defined robust performance as $\|F_l(N,\Delta)\|_\infty < 1$ for all allowed Δ. Since we used the $\mathcal{H}_\infty$ norm in both the representation of uncertainty and the definition of performance, we found that RP could be viewed as a special case of RS, and we derived

$$RS \quad \Leftrightarrow \quad \mu(N) < 1, \ \forall\omega$$

where μ is computed with respect to the **block-diagonal structure** $\text{diag}\{\Delta, \Delta_P\}$. Here Δ represents the uncertainty and Δ_P is a fictitious full uncertainty block representing the $\mathcal{H}_\infty$ performance bound.

There are **two main approaches to getting a robust design**:

1. We aim to make the system robust to some **"general" class of uncertainty** which we do not explicitly model. For SISO systems, the classical gain and phase margins and the peaks of S and T provide useful robustness measures. For MIMO systems, normalized coprime factor uncertainty provides a good general class of uncertainty, and the associated Glover-McFlarlane $\mathcal{H}_\infty$ loop-shaping design procedure has proved itself very useful in applications
2. We **explicitly model and quantify the uncertainty** in the plant and aim to make the system robust to this specific uncertainty. Potentially, it yields better designs, but it may require a much larger effort in terms of uncertainty modelling, especially if parametric uncertainty is consider. Analysis and in particular, synthesis using μ can be very involved

In applications, it is therefore recommended to **start with the first approach**, at least for design. The robust stability and performance is then analyzed in simulations and using the structured singular value, for example, by considering first simple sources of uncertainty such as multiplicative input uncertainty. One then iterates between design and analysis until a satisfactory solution is obtained. If resulting control performance is not satisfactory, one may switch to the second approach.

Practical μ-synthesis in practice:

1. Because of the effort involved in deriving detailed uncertainty descriptions, and the subsequent complexity in synthesizing controllers, the rule is to **start simple** with a crude uncertainty description, and then to see whether the performance specifications can be met. Only if they can't, one should

consider more detailed uncertainty descriptions such as parametric uncertainty

2. The use of μ implies a worst-case analysis, so one should be **careful about including too many sources of uncertainty**, noise and disturbances - otherwise it becomes very unlikely for the worst case to occur, and the resulting analysis and design may be **unnecessarily conservative**
3. There is always uncertainty with respect to the inputs and outputs, so **it is generally sage to include diagonal input and output uncertainty**. The relative multiplicative form is very convenient in this case
4. μ is most commonly used for analysis. If μ is used for synthesis, then we recommend that you keep the uncertainty fixed and adjust the parameters in the performance weight until μ is close to 1

9 Controller Design

9.1 Trade-offs in MIMO Feedback Design

The shaping of multivariable transfer functions is based on the idea that a satisfactory definition of gain for a matrix transfer function is given by the **singular values**. By multivariable transfer function shaping, therefore, we mean the shaping of the singular values of appropriate specified transfer functions such as the loop transfer function of one or more closed-loop transfer functions.

The classical loop-shaping ideas can be further generalized to MIMO systems by considering the singular values.
Consider the one degree-of-freedom system as shown in Fig. 40. We have the following important relationships:

$$y(s) = T(s)r(s) + S(s)d(s) - T(s)n(s) \tag{62a}$$
$$u(s) = K(s)S(s)\big(r(s) - n(s) - d(s)\big) \tag{62b}$$

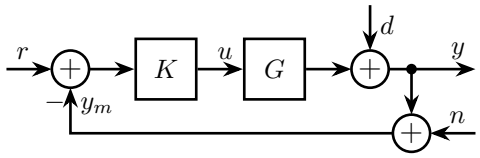


Figure 40 – *One degree-of-freedom feedback configuration*

Typical Closed-Loop Objectives

1. For disturbance rejection make $\overline{\sigma}(S)$ small
2. For noise attenuation make $\overline{\sigma}(T)$ small
3. For reference tracking make $\overline{\sigma}(T) \approx \underline{\sigma}(T) \approx 1$
4. For control energy reduction make $\overline{\sigma}(KS)$ small
5. For robust stability in presence of an additive perturbation ($G_p = G + \Delta$) make $\overline{\sigma}(KS)$ small
6. For robust stability in presence of a multiplicative output perturbation ($G_p = (I + \Delta)G$) make $\overline{\sigma}(T)$ small

The closed-loop requirements cannot all be satisfied simultaneously. Feedback design is therefore a **trade-off over frequency of conflicting objectives**. This is not always as difficult as it sounds because the frequency range over which the objectives are important can be quite different.

In classical loop shaping, it is the magnitude of the open-loop transfer function $L = GK$ which is shaped, whereas the above requirements are all in terms of closed-loop transfer functions. However, we have that

$$\underline{\sigma}(L) - 1 \leq \frac{1}{\overline{\sigma}(S)} \leq \underline{\sigma}(L) + 1$$

from which we see that $\overline{\sigma}(S) \approx 1/\underline{\sigma}(L)$ at frequencies where $\underline{\sigma}(L)$ is much larger than 1. Furthermore, from $T = L(I + L)^{-1}$ it follows that $\overline{\sigma}(T) \approx \overline{\sigma}(L)$ at frequencies where $\overline{\sigma}(L)$ is much smaller than 1.
Thus, over specified frequency ranges, it is relatively easy to approximate the closed-loop requirements by open-loop objectives.

Typical Open-Loop Objectives

1. For disturbance rejection make $\underline{\sigma}(GK)$ large
2. For noise attenuation make $\overline{\sigma}(GK)$ small
3. For reference tracking make $\underline{\sigma}(GK)$ large
4. For control energy reduction make $\overline{\sigma}(K)$ small
5. For robust stability in presence of an additive perturbation make $\overline{\sigma}(K)$ small
6. For robust stability in presence of an multiplicative output perturbation make $\overline{\sigma}(GK)$ small

Typically, the open-loop requirements 1 and 3 are valid and important at low frequencies $0 \leq \omega \leq \omega_l \leq \omega_B$, while conditions 2, 4, 5 and 6 are conditions which are valid and important at high frequencies $\omega_B \leq \omega_h \leq \omega \leq \infty$, as illustrated in Fig. 41.

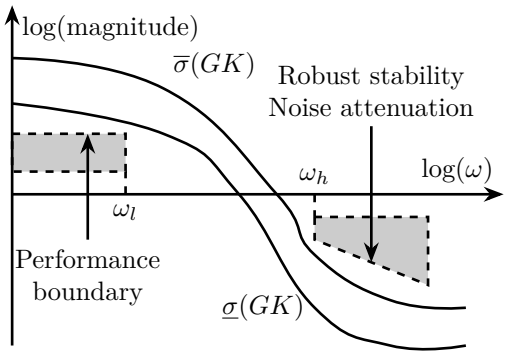


Figure 41 – *Design trade-offs for the multivariable loop transfer function GK*

The control engineer must design K such that $\underline{\sigma}(GK)$ lies above a performance boundary for all ω up to ω_l, and such that $\overline{\sigma}(GK)$ lies below a robustness boundary for all ω above ω_h.

Shaping the singular values of GK by selecting K is relatively easy task, but to do this in a way which also guarantees closed-loop stability is in general difficult as **closed-loop stability cannot be determined**

from open-loop singular values.

For SISO systems, closed-loop stability is closely related to the open-loop roll-off rate from high to low gain at the crossover (which is in practice less than 40 dB/dec). An immediate consequence of this is that there is a lower limit to the difference between ω_h and ω_l.
For MIMO systems, a similar gain/phase relationship holds in the crossover frequency region, but this is in terms of roll-off rate of the magnitude of the eigenvalues of GK and not the singular values. The stability constraint is therefore more difficult to handle.

9.2 LQG Control

LQG control was developed and successfully applied for aerospace problems where accurate plants are available. For other control problems, it was not easy, and the **assumption of white noise disturbance** is not always relevant. As a result, LQG designs were sometimes not robust enough to be used in practice.

It is assumed that the plant dynamics are linear and known, and that the measurement noise and disturbance signals are stochastic with known statistical properties:

$$\dot{x} = Ax + Bu + w_d$$
$$y = Cx + Du + w_n$$

with w_d and w_n are the disturbance and measurement noise which are assumed to be uncorrelated zero-mean Gaussian stochastic processes with constant power spectral density matrices W and V respectively.

LQG control problem

The LQG control problem is to find the optimal control $u(t)$ that minimize:

$$J = E\left\{ \lim_{T\to\infty} \frac{1}{T} \int_0^T \left[x^T Q x + u^T R u\right] dt \right\}$$

Where Q and R are appropriately chosen constant **weighting matrices** (design parameters) such that $Q = Q^T \geq 0$ and $R = R^T > 0$.

The solution to the LQG problem, known as the **Separation Theorem**, is separated into **two problems**. It consists of first determining the **optimal control** to a deterministic **LQR problem** (LQG without w_d and w_n). The solution to this problem is a state feedback law

$$u(t) = -K_r x(t) \tag{63}$$

where K_r is a **constant matrix** that can be easily computed.

The next step is to find an **optimal estimate** $\hat{x}$ of the state x so that $E\{[x-\hat{x}]^T[x-\hat{x}]\}$ is minimized. The optimal state estimate is given by a **Kalman filter**.

The solution to the LQG problem is then found by replacing x by $\hat{x}$ to give $u(t) = -K_r\hat{x}$.

We therefore see that the LQG problem and its solution can be separated into two distinct parts as illustrated in Fig. 42: the optimal state feedback and the optimal state estimator (the Kalman filter).

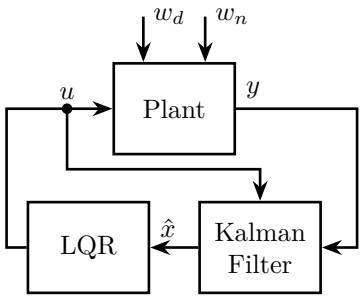


Figure 42 – *The separation theorem*

Optimal State Feedback

The LQR problem, where all the states are known is to find the input signal $u(t)$ that takes the system $\dot{x} = Ax+Bu$ to the zero state ($x = 0$) by minimizing the deterministic cost

$$J_r = \int_0^\infty \left(x(t)^T Q x(t) + u(t)^T R u(t)\right) dt \tag{64}$$

The optimal solution is $u = -K_r x(t)$ with

$$K_r = R^{-1} B^T X \tag{65}$$

and X is the unique positive-semi definite solution of the algebraic Riccati equation:

$$A^T X + XA - XBR^{-1}B^T X + Q = 0 \tag{66}$$

Kalman Filter

The Kalman filter has the structure of an ordinary state-estimator, as shown on Fig. 43, with:

$$\dot{\hat{x}} = A\hat{x} + Bu + K_f(y - C\hat{x}) \tag{67}$$

The optimal choice of K_f, which minimize $E\{[x-\hat{x}]^T[x-\hat{x}]\}$ is given by

$$K_f = YC^T V^{-1} \tag{68}$$

Where Y is the unique positive-semi definite solution of the algebraic Riccati equation

$$YA^T + AY - YC^T V^{-1} CY + W = 0 \tag{69}$$

The structure of the LQG controller is illustrated in

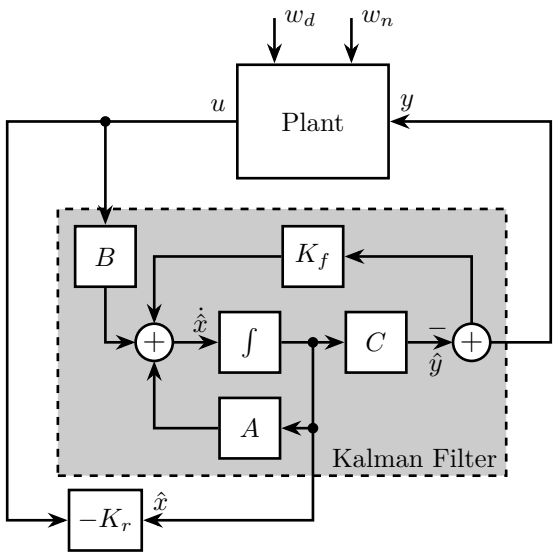


Figure 43 – *The LQG controller and noisy plant*

Fig. 43, its transfer function from y to u is given by

$$L_{\text{LQG}}(s) = \left[\begin{array}{c|c} A - BK_r - K_fC & K_f \\ \hline -K_r & 0 \end{array}\right] = \left[\begin{array}{c|c} A - BR^{-1}B^TX - YC^TV^{-1}C & YC^TV^{-1} \\ \hline -R^{-1}B^TX & 0 \end{array}\right]$$

It has the same degree (number of poles) as the plant.

For the LQG-controller, as shown on Fig. 43, it is not easy to see where to position the reference input r and how integral action may be included, if desired. Indeed, the standard LQG design procedure does not give a controller with integral action. One strategy is illustrated in Fig. 44. Here, the control error $r - y$ is integrated and the regulator K_r is designed for the plant augmented with these integral states.

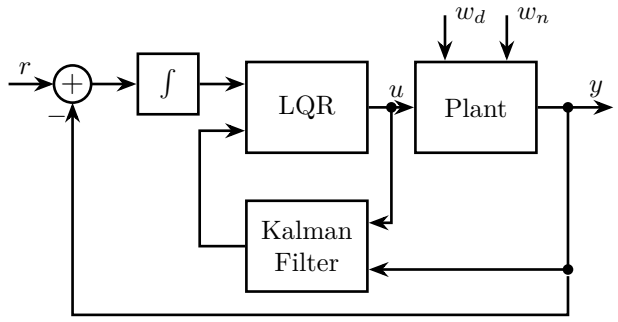


Figure 44 – *LQG controller with integral action and reference input*

For an LQG-controller system with a combined Kalman filter and LQR control law, there are **no guaranteed stability margins**, and there exist LQG combinations with arbitrary small gain margins. However, there are procedures for improving robustness properties of LQG control such as **Loop Transfer Recovery** (LTR).

These procedure are somehow difficult to use in practice. Their main limitation is that they can only be applied to minimum phase plants.

9.3 $\mathcal{H}_2$ and $\mathcal{H}_\infty$ Control

a General Control Problem Formulation

There are many ways in which feedback design problems can be cast as $\mathcal{H}_2$ and $\mathcal{H}_\infty$ optimization problems. It is very useful therefore to have a **standard problem formulation** into which any particular problem may be manipulated.
Such a general formulation is afforded by the general configuration shown in Fig. 45.

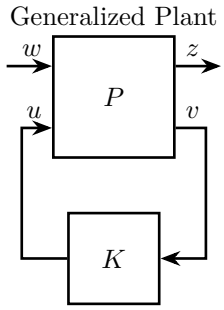


Figure 45 – *General control configuration*

The system is described by

$$\begin{bmatrix} z \\ v \end{bmatrix} = P(s) \begin{bmatrix} w \\ u \end{bmatrix} = \begin{bmatrix} P_{11}(s) & P_{12}(s) \\ P_{21}(s) & P_{22}(s) \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix} \tag{70a}$$

$$u = K(s)v \tag{70b}$$

With a state space realization of the generalized plant P given by

$$P = \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{array}\right] \tag{71}$$

The closed loop transfer function from w to z is given by the linear fractional transformation:

$$\begin{aligned} z &= F_l(P, K)w \\ &= [P_{11} + P_{12}K(I - P_{22}K)^{-1}P_{21}]w \end{aligned}$$

$\mathcal{H}_2$ and $\mathcal{H}_\infty$ control involve the minimization of the $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norms of $F_l(P, K)$.

The most general and widely used algorithms for $\mathcal{H}_2$ and $\mathcal{H}_\infty$ control problems are based on the state space formulation and requires the solution of two Riccati equations.
The following **assumptions** are typically made in $\mathcal{H}_2$ and $\mathcal{H}_\infty$ problems:

1. (A, B_2, C_2) is stabilizable and detectable. This is required for the existence of stabilizing controllers

2. D_{12} and D_{21} have full rank. This is sufficient to ensure that the controllers are proper
3. $\begin{bmatrix} A-j\omega I & B_2 \\ C_1 & D_{12} \end{bmatrix}$ and $\begin{bmatrix} A-j\omega I & B_1 \\ C_2 & D_{21} \end{bmatrix}$ have respectively full column and full row rank for all ω. This ensures that the controller does not cancel poles or zeros in the imaginary axis which would result in closed-loop instability
4. $D_{11}=0$ and $D_{22}=0$ is a conventional requirement for $\mathcal{H}_2$ control. This is not required for $\mathcal{H}_\infty$ control but this significantly simplify algorithm formulas
5. $D_{12}^T C_1 = 0$ and $B_1 D_{12}^T = 0$ is common in $\mathcal{H}_2$ control. $D_{12}^T C_1 = 0$ means that there is no cross terms in the cost function and $B_1 D_{12}^T = 0$ that the process noise and measurement noise are uncorrelated
6. (A, B_1) is stabilizable and (A, C_1) is detectable

If the Matlab Robust Control Toolbox complains, then it probably means that the control problem is not well formulated and that some assumptions are not met.

$\mathcal{H}_\infty$ algorithms, in general, find a **sub-optimal controller**. That is, for a specified γ a stabilizing controller is found for which $\|F_l(P,K)\|_\infty < \gamma$. This contrasts with $\mathcal{H}_2$ theory, in which the optimal controller is **unique** and can be found from the solution of two Riccati equations.

b $\mathcal{H}_2$ Optimal Control

$\mathcal{H}_2$ Optimal Control Problem

The standard $\mathcal{H}_2$ optimal control problem is to find a stabilizing controller K which minimizes

$$\|F(s)\|_2 = \sqrt{\frac{1}{2\pi}\int_{-\infty}^{\infty} tr[F(j\omega)F(j\omega)^H]d\omega}$$

With $F = F_l(P,K)$.

For a particular problem, the generalized plant P will include the plant model, the interconnection structure, and the designer specified weighting functions.

The $\mathcal{H}_2$ norm can be given different **deterministic interpretations**. It also has the following **stochastic interpretation**.

Suppose in the general control configuration that the exogenous input w is white noise of unit density. That is

$$E\{w(t)w(\tau)^T\} = I\delta(t-\tau)$$

Expected power in the error signal z is then given by

$$\begin{aligned} P_z &= E\left\{\lim_{T\to\infty}\frac{1}{2T}\int_{-T}^{T} z(t)^T z(t)dt\right\} \\ &= \text{tr } E\{z(t)z(t)^H\} \\ &= \frac{1}{2\pi}\int_{-\infty}^{\infty} \text{tr}\left[F(j\omega)F(j\omega)^H\right]d\omega \\ &= \|F\|_2^2 = \|F_l(P,K)\|_2^2 \end{aligned}$$

Thus, by minimizing the $\mathcal{H}_2$ norm, the error power of the generalized system, due to a unit intensity white noise input, is minimized. We are **minimizing the Root Mean Square value** of z.

c LQG: a Special $\mathcal{H}_2$ Optimal Controller

An important special case of $\mathcal{H}_2$ optimal control is the LQG problem. For the stochastic system

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

where

$$E\left\{\begin{bmatrix} w_d(t) \\ w_n(t) \end{bmatrix}\begin{bmatrix} w_d(\tau)^T & w_n(\tau)^T \end{bmatrix}\right\} = \begin{bmatrix} W & 0 \\ 0 & V \end{bmatrix}\delta(t-\tau)$$

The LQG problem is to find $u = K(s)y$ such that

$$J = E\left\{\lim_{T\to\infty}\frac{1}{T}\int_0^T [x^T Q x + u^T R u]dt\right\}$$

is minimized with $Q = Q^T \geq 0$ and $R = R^T > 0$.
This problem can be cast as an $\mathcal{H}_2$ optimization in the general framework in the following manner.
Define the error signal z as

$$z = \begin{bmatrix} Q^{\frac{1}{2}} & 0 \\ 0 & R^{\frac{1}{2}} \end{bmatrix}\begin{bmatrix} x \\ u \end{bmatrix}$$

Represent the stochastic inputs as

$$\begin{bmatrix} w_d \\ w_n \end{bmatrix} = \begin{bmatrix} W^{\frac{1}{2}} & 0 \\ 0 & V^{\frac{1}{2}} \end{bmatrix} w$$

where w is a white noise process of unit density.
Then the LQG cost function is

$$K = E\left\{\lim_{T\to\infty}\frac{1}{T}\int_0^T z(t)^T z(t)dt\right\} = \|F_l(P,K)\|_2^2$$

d $\mathcal{H}_\infty$ Optimal Control

With reference to the general control configuration on Fig. 45, the standard $\mathcal{H}_\infty$ optimal control problem is to find all stabilizing controllers K which minimize

$$\|F_l(P,K)\|_\infty = \max_\omega \bar{\sigma}\big(F_l(P,K)(j\omega)\big)$$

The $\mathcal{H}_\infty$ norm has **several interpretations** in terms of performance. One is that it minimizes the peak of the maximum singular value of $F_l(P(j\omega), K(j\omega))$.
It also has a time domain interpretation as the worst-case 2-norm:

$$\|F_l(P,K)\|_\infty = \max_{w(t)\neq 0} \frac{\|z(t)\|_2}{\|w(t)\|_2} \tag{72}$$

where $\|z(t)\|_2 = \sqrt{\int_0^\infty \sum_i |z_i|^2 \, dt}$ is the 2-norm of the vector signal.

In practice, it is usually not necessary to obtain an optimal controller for the $\mathcal{H}_\infty$ problem, and it is simpler to design a **sub-optimal** one.
Let $\gamma_{\min}$ be the minimum value of $\|F_l(P,K)\|_\infty$ over all stabilizing controllers K. Then the $\mathcal{H}_\infty$ **sub-optimal control problem** is: given a $\gamma > \gamma_{\min}$, find all stabilizing controllers K such that

$$\|F_l(P,K)\|_\infty < \gamma \tag{73}$$

By reducing γ in an iterative way, an optimal solution is approached.

General $\mathcal{H}_\infty$ algorithm. For the general control configuration and with assumptions described above, there exists a stabilizing controller $K(s)$ such that $\|F_l(P,K)\|_\infty < \gamma$ if and only if

1. $X_\infty \geq 0$ is a solution to the algebraic Riccati equation $A^T X_\infty + X_\infty A + C_1^T C_1 + X_\infty(\gamma^{-2} B_1 B_1^T - B_2 B_2^T) X_\infty = 0$ such that $\text{Re}\,\lambda_i \left[A + (\gamma^{-2} B_1 B_1^T - B_2 B_2^T) X_\infty\right] < 0, \ \forall i$
2. $Y_\infty \geq 0$ is a solution to the algebraic Riccati equation $AY_\infty + Y_\infty A^T + B_1 B_1^T + Y_\infty(\gamma^{-2} C_1^T C_1 - C_2^T C_2) Y_\infty = 0$ such that $\text{Re}\,\lambda_i \left[A + Y_\infty(\gamma^{-2} C_1^T C_1 - C_2^T C_2) Y_\infty\right] < 0, \ \forall i$
3. $\rho(X_\infty Y_\infty) < \gamma^2$

All such controllers are then given by $K = F_l(K_c, Q)$ where

$$K_c(s) = \left[\begin{array}{c|cc} A_\infty & -Z_\infty L_\infty & Z_\infty B_2 \\ \hline F_\infty & 0 & I \\ -C_2 & I & 0 \end{array}\right], \ L_\infty = -Y_\infty C_2^T$$

$$F_\infty = -B_2^T X_\infty, \ Z_\infty = (I - \gamma^2 Y_\infty X_\infty)^{-1}$$

$$A_\infty = A + \gamma^{-2} B_1 B_1^T X_\infty + B_2 F_\infty + Z_\infty L_\infty C_2$$

and $Q(s)$ is any stable proper transfer function such that $\|Q\|_\infty < \gamma$.

For $Q(s) = 0$, we get

$$K(s) = K_{c11}(s) = -Z_\infty L_\infty (sI - A_\infty)^{-1} F_\infty \tag{74}$$

This is called the **central controller** and has the same number of states as the generalized plant $P(s)$.
The central controller can be separated into a state estimator (observer) of the form

$$\dot{\hat{x}} = A\hat{x} + B_1 \gamma^{-2} B_1^T X_\infty \hat{x} + B_2 u + Z_\infty L_\infty (C_2 \hat{x} - y)$$

and a state feedback $u = F_\infty \hat{x}$.

γ-iteration

If we desire a controller that achieves $\gamma_{\min}$, to within specified tolerance, then we can perform a **bisection** on γ until its value is sufficiently accurate. The above conditions provide a test for each value of γ to determine if $\gamma < \gamma_{\min}$ or $\gamma > \gamma_{\min}$.

There are mainly two methodologies for $\mathcal{H}_\infty$ controller design: the **transfer function shaping approach** and the **signal-based approach**.

In the **shaping approach**, $\mathcal{H}_\infty$ optimization is used to shape the singular values of specified transfer functions over frequency. The maximum singular values are relatively easy to shape by forcing them to lie below user defined bounds, thereby ensuring desirable bandwidth and roll-off rates.

In the **signal-based approach**, we seek to minimize the energy in certain error signal given a set of exogenous input signals.

A difficulty that sometimes arises with $\mathcal{H}_\infty$ control is the **selection of weights** such that the $\mathcal{H}_\infty$ optimal controller provides a good trade-off between conflicting objectives in various frequency ranges. Thus, for practical designs it is sometimes recommended to perform only a few iterations of the $\mathcal{H}_\infty$ algorithm. The justification for this is that the initial design, after one iteration, is similar to an $\mathcal{H}_2$ design which does trade-off over various frequency ranges. Therefore stopping the iterations before the optimal value is achieved gives the design an $\mathcal{H}_2$ flavor which may be desirable.

e Mixed-Sensitivity $\mathcal{H}_\infty$ Control

Mixed-sensitivity is the name given to transfer function shaping problems in which the sensitivity function $S = (I + GK)^{-1}$ is shaped along with one or more other closed-loop transfer functions such as KS or $T = I - S$.

Suppose that we have a regulation problem in which we want to reject a disturbance d entering at the plant output and it is assumed that the measurement noise is relatively insignificant. It makes sense to shape the closed-loop transfer functions S and KS. Recall that S is the transfer function between d and the output, and KS the transfer function from d and the control signal. It is important to include KS as a mechanism for **limiting the size and bandwidth of the controller**, and hence the energy used. The size of KS is also important for robust stability with respect to uncertainty modeled as additive plant perturbations.

The disturbance d is typically a low frequency signal, and therefore it will be successfully rejected if the maximum singular value of S is made small over the same low frequency range. To do this, we could select a scalar low pass filter $w_1(s)$ with a bandwidth equal to that of the disturbance, and then find a stabilizing controller that minimizes $\|w_1 S\|_\infty$. This cost function alone is not very practical, it focuses on just one closed-loop transfer function and the controller may have infinite gain. It is far more useful in practice to minimize

$$\left\| \begin{matrix} w_1 S \\ w_2 K S \end{matrix} \right\|_\infty \tag{75}$$

where $w_2(s)$ is a scalar high pass filter with a crossover frequency approximately equal to that of the desired closed-loop bandwidth.

In general, the scalar weighting functions $w_1(s)$ and $w_2(s)$ can be replaced by matrices $W_1(s)$ and $W_2(s)$. This can be useful for **systems with channels of quite different bandwidths**. In that case, **diagonal weights are recommended** as anything more complicated is usually not worth the effort.

To see how this mixed sensitivity problem can be formulated in the general setting, we can imagine the disturbance d as a single exogenous input and define and error signal $z = [z_1^T \; z_2^T]^T$, where $z_1 = W_1 y$ and $z_2 = -W_2 u$ as illustrated in Fig. 46. We can then see that $z_1 = W_1 S w$ and $z_2 = W_2 K S w$ as required. The elements of the generalized plant are

$$P_{11} = \begin{bmatrix} W_1 \\ 0 \end{bmatrix} \quad P_{12} = \begin{bmatrix} W_1 G \\ -W_2 \end{bmatrix}$$
$$P_{21} = -I \quad P_{22} = -G$$

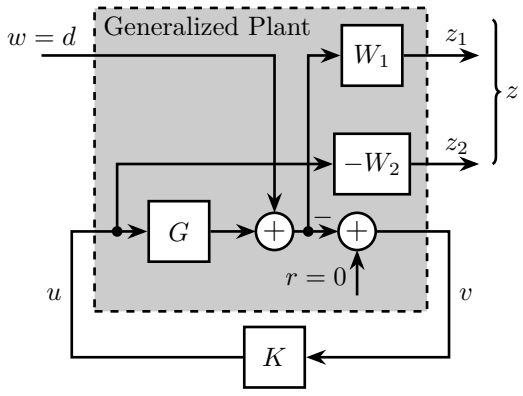


Figure 46 – *S/KS mixed-sensitivity optimization in standard form (regulation)*

Another interpretation can be put on the S/KS mixed-sensitivity optimization as shown in the standard control configuration of Fig. 47. Here we consider a tracking problem. The exogenous input is a reference command r, and the error signals are $z_1 = -W_1 e = W_1(r - y)$ and $z_2 = W_2 u$. As the regulation problem of Fig. 46, we have that $z_1 = W_1 S w$ and $z_2 = W_2 K S w$.

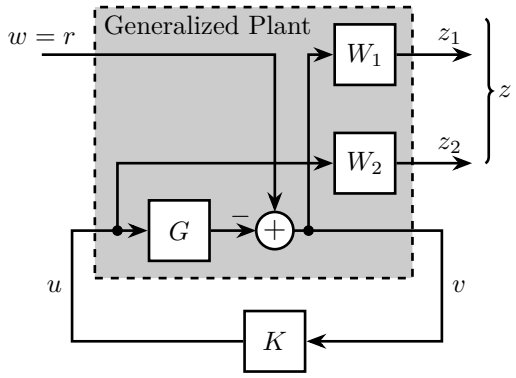


Figure 47 – *S/KS mixed-sensitivity optimization in standard form (tracking)*

Another useful mixed sensitivity optimization problem, is to find a stabilizing controller which minimizes

$$\left\| \begin{matrix} W_1 S \\ W_2 T \end{matrix} \right\|_\infty \tag{76}$$

The ability to shape T is desirable for tracking problems and noise attenuation. It is also important for robust stability with respect to multiplicative perturbations at the plant output.
The S/T mixed-sensitivity minimization problem can be put into the standard control configuration as shown in Fig. 48.
The elements of the generalized plant are

$$P_{11} = \begin{bmatrix} W_1 \\ 0 \end{bmatrix} \quad P_{12} = \begin{bmatrix} -W_1 G \\ W_2 G \end{bmatrix}$$
$$P_{21} = -I \quad P_{22} = -G$$

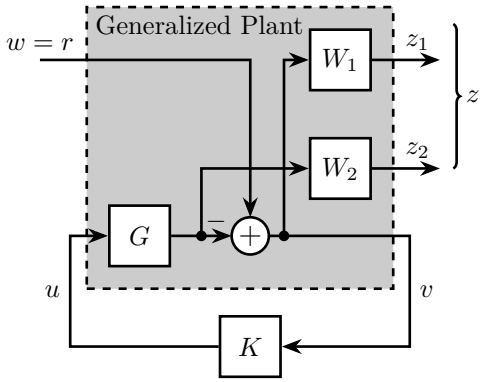


Figure 48 – *S/T mixed-sensitivity optimization in standard form*

The shaping of closed-loop transfer functions as described above with the stacked cost functions becomes difficult with more than two functions whereas with two, the process is relatively easy. The bandwidth requirements on each are usually complementary and simple, stable low-pass and high-pass filters

are sufficient to carry out the required shaping and trade-offs.

The weights W_i in mixed-sensitivity $\mathcal{H}_\infty$ optimal control must all be stable. Therefore, if we wish, for example, to emphasize the minimization of S at low frequency by weighting with a term including integral action, we would have to approximate $\frac{1}{s}$ by $\frac{1}{s+\epsilon}$ where $\epsilon \ll 1$. Similarly, one might be interested in weighting KS with a non-proper weight to ensure that K is small outside of the system bandwidth. The trick is to replace a non proper term such as $(1+\tau_1 s)$ by $\frac{1+\tau_1 s}{1+\tau_2 s}$ where $\tau_2 \ll \tau_1$.

f Signal-Based $\mathcal{H}_\infty$ Control

The signal-based approach to controller design is very general and is appropriate for multivariable problems in which several objectives must be taken into account simultaneously. In this approach, we define the plant, possibly the model uncertainty, the **class of external signals affecting the system** and the **norm of the error signals we want to keep small**.

> The focus of attention has moved to the size of signals and away from the size and bandwidth of selected closed-loop transfer functions.

Weights are used to describe the expected or known frequency content of exogenous signals and the desired frequency content of error signals. Weights are also used if a perturbation is used to model uncertainty, as in Fig. 49, where G represents the nominal model, W is a weighting function that captures the relative model fidelity over frequency, and Δ represents unmodelled dynamics usually normalized such that $\|\Delta\|_\infty < 1$.

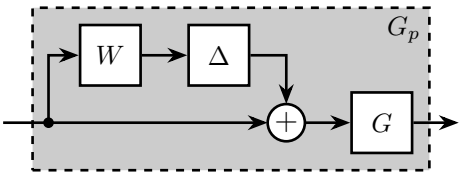


Figure 49 – *Multiplicative dynamic uncertainty model*

LQG control is a simple example of the signal based approach, in which the exogenous signals are assumed to be stochastic and the error signals are measured in terms of the 2-norm. As we have seen, the weights Q and R are constant, but LQG can be generalized to include frequency dependent weights on the signals leading to what is called Wiener-Hopf design or $\mathcal{H}_2$ control.

When we consider a system's response to persistent sinusoidal signals of varying frequency, or when we consider the induced 2-norm between the exogenous input signals and the error signals, we are required to minimize the $\mathcal{H}_\infty$ norm. In the absence of model uncertainty, there does not appear to be an overwhelming case for using the $\mathcal{H}_\infty$ norm rather than the more traditional $\mathcal{H}_2$ norm. However, when uncertainty is addressed, as it always should be, $\mathcal{H}_\infty$ is clearly the more **natural approach** using component uncertainty models as in Fig. 49.

A typical problem using the signal-based approach to $\mathcal{H}_\infty$ control is illustrated in the interconnection diagram of Fig. 50. G and G_d are nominal models of the plant and disturbance dynamics, and K is the controller to be designed. The weights W_d, W_r, and W_n may be constant or dynamic and describe the relative importance and/or the frequency content of the disturbance, set points and noise signals. The weight W_{ref} is a desired closed-loop transfer function between the weighted set point r_s and the actual output y. The weights W_e and W_u reflect the desired frequency content of the error $(y - y_{\text{ref}})$ and the control signals u, respectively. The problem can be cast as a standard $\mathcal{H}_\infty$ optimization in the general control configuration by defining

$$w = \begin{bmatrix} d \\ r \\ n \end{bmatrix}, \quad z = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}, \quad v = \begin{bmatrix} r_s \\ y_m \end{bmatrix}, \quad u = u$$

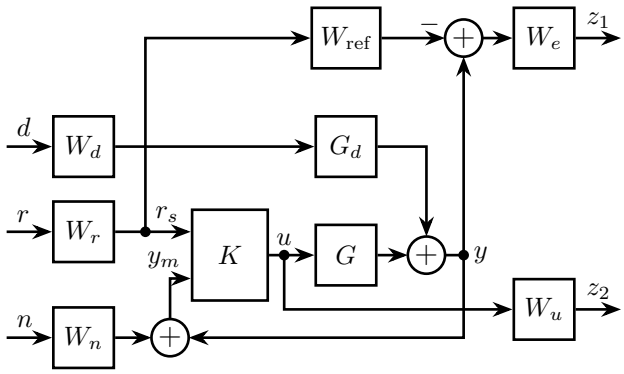


Figure 50 – *A signal-based $\mathcal{H}_\infty$ control problem*

Suppose we now introduce a multiplicative dynamic uncertainty model at the input to the plant as shown in Fig. 51. The problem we now want to solve is: find a stabilizing controller K such that the $\mathcal{H}_\infty$ norm of the transfer function between w and z is less that 1 for all Δ where $\|\Delta\|_\infty < 1$. We have assumed in this statement that the **signal weights have normalized the 2-norm of the exogenous input signals to unity**. This problem is a non-standard $\mathcal{H}_\infty$ optimization. It is a robust performance problem for which the μ-synthesis procedure can be applied where we require the structured singular value:

$$\mu(M(j\omega)) < 1, \quad \forall\omega$$

However, whilst the structured singular value is a useful analysis tool for assessing designs, μ-synthesis is sometimes difficult to use and often too complex for the practical problems.

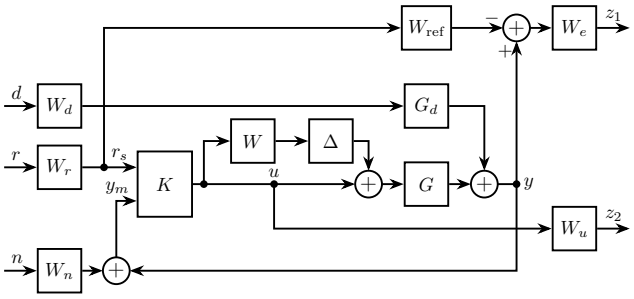


Figure 51 – *A signal-based $\mathcal{H}_\infty$ control problem with input multiplicative uncertainty*

9.4 $\mathcal{H}_\infty$ Loop-Shaping Design

The loop-shaping design procedure described in this section is based on $\mathcal{H}_\infty$ robust stabilization combined with classical loop shaping. It is essentially a **two stage design process**:

- First the open-loop plant is augmented by pre and post compensators to give a desired shape to the singular values of the open-loop frequency response
- Then the resulting shaped plant is robustly stabilized with respect to coprime factor uncertainty using $\mathcal{H}_\infty$ optimization

An important advantage is that no problem-dependent uncertainty modelling, or weight selection, is required in this second step.

a Robust Stabilization

For multivariable systems, **classical gain and phase margins are unreliable indicators of robust stability** when defined for each channel (or loop), taken one at a time, because simultaneous perturbations in more than one loop are not then catered for.

It is now common practice to model uncertainty by stable **norm-bounded** dynamic (complex) **matrix perturbations**. With a single perturbation, the associated robustness tests is in terms of the maximum singular values of various closed-loop transfer functions. Use of a single stable perturbation restricts the plant and perturbed plant models to either have the same number of unstable poles or the same number of RHP zeros.

To overcome this, **two stable perturbations** can be used, one on each of the factors in a **coprime factorization** of the plant. Although this uncertainty description seems unrealistic and less intuitive than the others, it is in fact quite general, and for our purposes it leads to a very useful $\mathcal{H}_\infty$ robust stabilization problem.

Let's consider the stabilization of a plant G which has a normalized left coprime factorization

$$G = M^{-1}N \tag{77}$$

where we have dropped the subscripts from M and N for simplicity.
A perturbed plant model G_p can then we written has

$$G_p = (M + \Delta_M)^{-1}(N + \Delta_N) \tag{78}$$

where Δ_M, Δ_N are stable unknown transfer functions which represent the uncertainty in the nominal plant G.

The objective of robust stabilization is to stabilize not only the nominal model G, but a family of perturbed plants defined by

$$G_p = \{(M + \Delta_M)^{-1}(N + \Delta_N) \ : \ \|\Delta_N \ \Delta_M\|_\infty < \epsilon\} \tag{79}$$

where $\epsilon > 0$ is then the **stability margin**.

For the perturbed feedback system of Fig. 52, the stability property is robust if and only if the nominal feedback system is stable and

$$\gamma \triangleq \left\| \begin{bmatrix} K \\ I \end{bmatrix} (I - GK)^{-1} M^{-1} \right\|_\infty \leq \frac{1}{\epsilon}$$

Notice that γ is the $\mathcal{H}_\infty$ norm from ϕ to $\begin{bmatrix} u \\ y \end{bmatrix}$ and $(I - GK)^{-1}$ is the sensitivity function for this positive feedback arrangement.

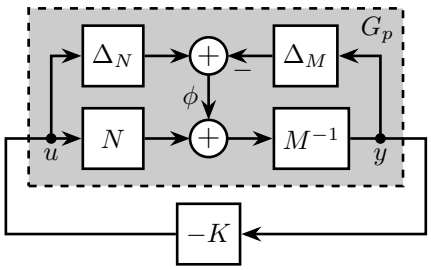


Figure 52 – *$\mathcal{H}_\infty$ robust stabilization problem*

The lowest achievable value of γ and the corresponding maximum stability margin ϵ are given as

$$\gamma_{\min} = \epsilon_{\max}^{-1} = \left\{1 - \|N \ M\|_H^2\right\}^{-\frac{1}{2}} = (1 + \rho(XZ))^{\frac{1}{2}} \tag{80}$$

where $\| \cdot \|_H$ denotes Hankel norm, ρ denotes the spectral radius (maximum eigenvalue), and for a minimal state space realization of G, Z is the unique positive definite solution of the algebraic Riccati equation

$$(A - BS^{-1}D^TC)Z + Z(A - BS^{-1}D^TC)^T - ZC^TR^{-1}CZ + BS^{-1}B^T = 0$$

where

$$R = I + DD^T, \quad S = I + D^TD$$

X is the unique positive definite solution of the following algebraic Riccati equation

$$(A - BS^{-1}D^TC)X + X(A - BS^{-1}D^TC)^T - XBS^{-1}B^TX + C^TR^{-1}C = 0$$

A controller which guarantees that

$$\left\| \begin{bmatrix} K \\ I \end{bmatrix} (I - GK)^{-1} M^{-1} \right\|_\infty \leq \gamma$$

for a specified $\gamma > \gamma_{\min}$, is given by

$$K \triangleq \left[\begin{array}{c|c} A+BF+\gamma^2 L^{-T} Z C^T (C+DF) & \gamma^2 L^{-T} Z C^T \\ \hline B^T X & -D^T \end{array}\right] \quad (81a)$$

$$F = -S^{-1}(D^T C + B^T X) \quad (81b)$$

$$L = (1-\gamma^2) I + XZ \quad (81c)$$

The Matlab function `coprimeunc` can be used to generate the controller in (81a). It is important to emphasize that since we can compute $\gamma_{\min}$ from (80) we get an explicit solution by solving just two Riccati equations and avoid the γ-iteration needed to solve the general $\mathcal{H}_\infty$ problem.

b A Systematic $\mathcal{H}_\infty$ Loop-Shaping Design Procedure

Robust stabilization alone is not much used in practice because the designer is not able to specify any performance requirements.
To do so, **pre and post compensators** are used to **shape the open-loop singular values** prior to robust stabilization of the "shaped" plant.
If W_1 and W_2 are the pre and post compensators respectively, then the shaped plant G_s is given by

$$G_s = W_2 G W_1 \quad (82)$$

as shown in Fig. 53.

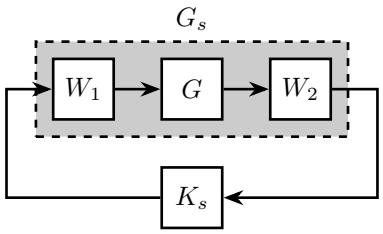


Figure 53 – *The shaped plant and controller*

The controller K_s is synthesized by solving the robust stabilization problem for the shaped plant G_s with a normalized left coprime factorization $G_s = M_s^{-1} N_s$. The feedback controller for the plant G is then $K = W_1 K_s W_2$.

Systematic procedure for $\mathcal{H}_\infty$ loop-shaping design:

1. **Scale the plant outputs and inputs**. This is very important for most design procedures. In general, scaling improves the conditioning of the design problem, it enables meaningful analysis to be made of the robustness properties of the feedback system in the frequency domain, and for loop shaping it can simplify the selection of weights:
   - The outputs are scaled such that equal magnitudes of cross-coupling into each of the outputs is equally undesirable
   - Each input is scaled by a given percentage (say 10 %) of its expected range of operation. That is, the inputs are scaled to reflect the relative actuator capabilities.
2. **Order the inputs and outputs** so that the plant is as diagonal as possible. The relative gain array can be useful here. The purpose of this pseudo-diagonalization is to ease the design of the pre and post compensators which, for simplicity, will be chosen to be diagonal. Next, we discuss the selection of weights to obtain the shaped plant $G_s = W_2 G W_1$ where $W_1 = W_p W_a W_g$
3. **Select the elements of diagonal pre and post compensators** W_p and W_2 so that the singular values of $W_2 G W_p$ are desirable. This would normally mean high gain at low frequencies, a slope of about -1 at the desired bandwidth(s), with higher rates at high frequencies. The weights should be chosen so that no unstable hidden modes are created in G_s
   - W_2 is usually chosen as a constant, reflecting the relative importance of the outputs to be controlled and the other measurements being fed back to the controller
   - W_p contains the dynamic shaping. Integral action, for low frequency performance; phase-advance for reducing the roll-off rates at crossover; and phase-lag to increase the roll-off rates at high frequencies should all be places in W_p is desired
4. *Optional*: Align the singular values at a desired bandwidth using a further constant weight W_a cascaded with W_p
5. *Optional*: Introduce an additional gain matrix W_g cascaded with W_a to provide control over actuator range. W_g is diagonal and is adjusted so that actuator rate limits are not exceeded for reference demands and typical disturbances on the scaled plant outputs
6. **Robustly stabilize the shaped plant** $G_s = W_2 G W_1$ where $W_1 = W_p W_a W_g$
   - First, calculate the maximum stability margin $\epsilon_{\max} = 1/\gamma_{\min}$
   - If the margin is too small, $\epsilon_{\max} < 0.25$, then go back to step 4 and modify the weights. Otherwise, select $\gamma > \gamma_{\min}$, by about 10 %, and synthesize a sub-optimal controller. There is usually no advantage to be gained by using the optimal controller
   - When $\epsilon_{\max} > 0.25$ (respectively $\gamma_{\min} < 4$) the design is usually successful. In this case, at least 25 % coprime factor uncertainty is allowed, and we also find that the shape of the open-loop singular values will not have

changed much after robust stabilization
    - A small value of $\epsilon_{\max}$ indicates that the chosen singular value loop-shapes are incompatible with robust stability requirements
7. **Analyze the design** and if not all the specification are met, make further modifications to the weights
8. **Implement the controller**. The configuration shown in Fig. 54 has been found useful when compared with the conventional setup in Fig. 40. This is because the references do not directly excite the dynamics of K_s, which can result in large amounts of overshoot. The constant prefilter ensure a steady-state gain of 1 between r and y, assuming integral action in W_1 or G

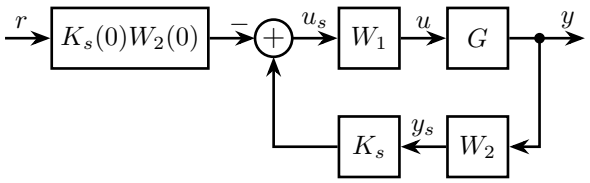


Figure 54 – *A practical implementation of the loop-shaping controller*

We will conclude this section with a summary of the **advantages** offered by the above $\mathcal{H}_\infty$ loop-shaping design procedure:

- It is relatively easy to use, being based on classical loop-shaping ideas
- There exists a closed formula for the $\mathcal{H}_\infty$ optimal cost $\gamma_{\min}$, which in turn corresponds to a maximum stability margin $\epsilon_{\max} = 1/\gamma_{\min}$
- No γ-iteration is required in the solution
- Except for special systems, ones with all-pass factors, there are no pole-zero cancellations between the plant and controller. Pole-zeros cancellations are common in many $\mathcal{H}_\infty$ control problems and are a problem when the plant has lightly damped modes

c Two Degrees-of-freedom Controllers

Many control design problems possess two degrees-of-freedom:

- on one hand, **measurement of feedback signals**
- and on the other hand, **commands and reference**

Sometimes, one degree-of-freedom is left out of the design, and the controller is driven by an error signal i.e. the difference between a command and the output. But in cases where stringent time-domain specifications are set on the output response, a one degree-of-freedom structure may not be sufficient.

A general two degrees-of-freedom feedback control scheme is depicted in Fig. 55. The commands and feedbacks enter the controller separately and are independently processed.

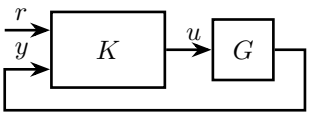


Figure 55 – *General two degrees-of-freedom feedback control scheme*

The presented $\mathcal{H}_\infty$ loop-shaping design procedure in section 9.4.b is a one-degree-of-freedom design, although a **constant** pre-filter can be easily implemented for steady-state accuracy. However, this may not be sufficient and a dynamic two degrees-of-freedom design is required.

The design problem is illustrated in Fig. 56. The feedback part of the controller K_2 is designed to meet robust stability and disturbance rejection requirements. A prefilter is introduced to force the response of the closed-loop system to follow that of a specified model T_{ref}, often called the **reference model**.

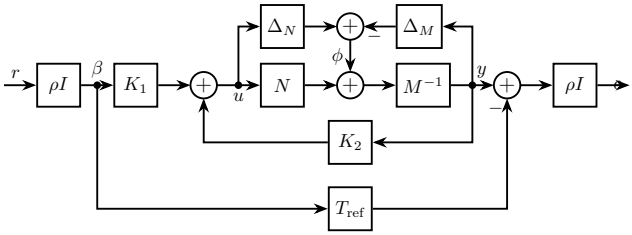


Figure 56 – *Two degrees-of-freedom $\mathcal{H}_\infty$ loop-shaping design problem*

The design problem is to find the stabilizing controller $K = [K_1,\ K_2]$ for the shaped plant $G_s = GW_1$, with a normalized coprime factorization $G_s = M_s^{-1}N_s$, which minimizes the $\mathcal{H}_\infty$ norm of the transfer function between the signals $[r^T\ \phi^T]^T$ and $[u_s^T\ y^T\ e^T]^T$ as defined in Fig. 56. This problem is easily cast into the general configuration.
The control signal to the shaped plant u_s is given by:

$$u_s = \begin{bmatrix} K_1 & K_2 \end{bmatrix} \begin{bmatrix} \beta \\ y \end{bmatrix}$$

where K_1 is the prefilter, K_2 is the feedback controller, β is the scaled reference and y is the measured output. The purpose of the prefilter is to ensure that:

$$\left\| (I - G_s K_2)^{-1} G_s K_1 - T_{\text{ref}} \right\|_\infty < \gamma \rho^2$$

T_{ref} is the desired closed-loop transfer function and ρ is a scalar parameter that the designer can increase to place more emphasis on model matching in the optimization at the expense of robustness.

The main steps required to synthesize a two degrees-of-freedom $\mathcal{H}_\infty$ loop-shaping controller are:

1. Design a one degree-of-freedom $\mathcal{H}_\infty$ loop-shaping controller (section 9.4.b) but without a post-compensator W_2
2. Select a desired closed-loop transfer function T_{ref} between the commands and controller outputs
3. Set the scalar parameter ρ to a small value greater than 1; something in the range 1 to 3 will usually suffice
4. For the shaped $G_s = GW_1$, the desired response T_{ref}, and the scalar parameter ρ, solve the standard $\mathcal{H}_\infty$ optimization problem to a specified tolerance to get $K = [K_1,\ K_2]$
5. Replace the prefilter K_1 by $K_1 W_i$ to give exact model-matching at steady-state.
6. Analyze and, if required, redesign making adjustments to ρ and possibly W_1 and T_{ref}

The final two degrees-of-freedom $\mathcal{H}_\infty$ loop-shaping controller is illustrated in Fig. 57.

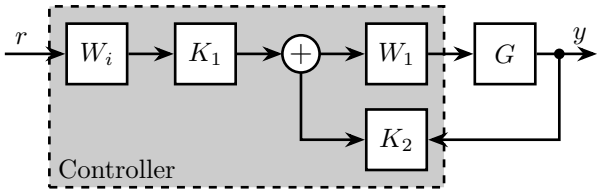


Figure 57 – *Two degrees-of-freedom $\mathcal{H}_\infty$ loop-shaping controller*

d Observer-Based Structure for $\mathcal{H}_\infty$ Loop-Shaping Controllers

$\mathcal{H}_\infty$ designs exhibit an observer/state feedback structure in the controller. The clear structure of the $\mathcal{H}_\infty$ loop-shaping controllers has several advantages:

- It is helpful in describing a controller's function
- It lends itself to implementation in a gain-schedule scheme
- If offers computational savings in digital implementations

Let's assume that the shaped plant is strictly proper, with a stabilizable and detectable state space realization

$$G_s \triangleq \left[\begin{array}{c|c} A_s & B_s \\ \hline C_s & 0 \end{array}\right]$$

The single degree-of-freedom $\mathcal{H}_\infty$ loop-shaping controller can be realized as an observer for the shaped plant plus a state feedback control law:

$$\begin{aligned} \dot{\hat{x}}_s &= A_s \hat{x}_s + H_s(C_s \hat{x}_s - y_s) + B_s u_s \\ u_s &= K_s \hat{x}_s \end{aligned}$$

where $\hat{x}_s$ is the observer state, u_s and y_s are respectively the input and output of the shaped plant, and

$$\begin{aligned} H_s &= -Z_s C_s^T \\ K_s &= -B_s^T [I - \gamma^{-2} I - \gamma^{-2} X_s Z_s]^{-1} X_s \end{aligned}$$

where Z_s and X_s are the appropriate solutions to the generalized algebraic Riccati equations for G_s.
The same can be done for two degrees-of-freedom controllers.

e Implementation Issues

Discrete-time controllers For implementation purposes, discrete-time controllers are usually required. These can be obtained from a continuous-time design using a **bilinear transformation** from the s-domain to the z-domain, but there can be advantages in being able to design directly in discrete time.

Anti-windup In $\mathcal{H}_\infty$ loop-shaping the pre compensator weight W_1 would normally include integral action in order to reject low frequency disturbances acting on the system. However, in the case of actuator saturation, the integrators continue to integrate their input and hence cause **windup** problems. An anti-windup scheme is therefore required on the weighting function W_1. The approach we recommend is to implement the weight W_1 in its self-conditioned or Hanus form. Let the weight W_1 have a realization

$$W_1 \triangleq \left[\begin{array}{c|c} A_w & B_w \\ \hline C_w & D_w \end{array}\right]$$

and let u be the input to the plant actuators and u_s the input to the shaped plant. Then $u = W_1 u_s$. When implemented in Hanus form, the expression for u becomes

$$u = \left[\begin{array}{c|cc} A_w - B_w D_w^{-1} C_w & 0 & B_w D_w^{-1} \\ \hline C_w & D_w & 0 \end{array}\right] \begin{bmatrix} u_s \\ u_a \end{bmatrix}$$

where u_a is the **actual plant input**, that is the measurement at the **output of the actuators** which therefore contains information about possible actuator saturation.
The situation is illustrated in Fig. 58, where the actuators are each modeled by a unit gain and a saturation.

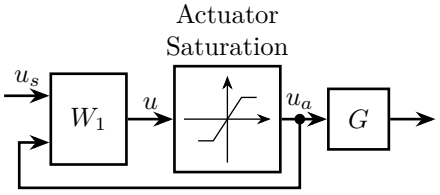


Figure 58 – *Self-conditioned weight W_1*

The Hanus form prevents windup by keeping the states of W_1 consistent with the actual plant input at all times.

When there is no saturation, $u_a = u$, the dynamics of W_1 remains unaffected. But when $u_a \neq u$, the dynamics are inverted and driven by u_a so that the states remain consistent with the actual plant input u_a. Notice that such an implementation requires W_1 to be invertible and minimum phase.

Bumpless transfer When multi-mode switched controller is designed, one should ensure **smooth transition from one controller to the other** (bumpless transfer). It was found useful to condition the reference models and the observers in each of the controllers. When on-line, the observer state evolves according to

$$\dot{\hat{x}}_s = A_s\hat{x}_s + H_s(C_s\hat{x}_s - y_s) + B_s u_s$$

but when off-line, the state equation becomes

$$\dot{\hat{x}}_s = A_s\hat{x}_s + H_s(C_s\hat{x}_s - y_s) + B_s u_{as}$$

where u_{as} is the actual input to the shaped plant governed by the on-line controller.
Doing so ensure that the inputs to the shaped plant for the off-line controller follows the actual shaped plant input u_{as} given by the on-line controller. The observer based structure of the $\mathcal{H}_\infty$ loop-shaping controller is then helpful for such technique.

9.5 Conclusion

Several methods and techniques for controller design have been described. The emphasis has been on $\mathcal{H}_\infty$ loop shaping which is easy to apply and works well in practice. It combines classical loop-shaping ideas with an effective method for robustly stabilizing the feedback loop.

For complex problems, such as unstable plants with multiple gain crossover frequencies, it may not be easy to decide on a desired loop shape. In which case, we would suggest doing an initial LQG design (with simple weights) and using the resulting loop shape as the desired one for the $\mathcal{H}_\infty$ loop shaping.

And alternative to $\mathcal{H}_\infty$ loop shaping is a standard $\mathcal{H}_\infty$ design with a stacked cost function such as in S/KS mixed-sensitivity optimization. In this approach, $\mathcal{H}_\infty$ optimization is used to shape two or sometimes three closed-loop transfer functions. However, with more functions, the shaping becomes increasingly difficult for the designer.

In other design situations where there are several performance objectives, it may be more appropriate to follow a signal-based $\mathcal{H}_2$ or $\mathcal{H}_\infty$ approach. But again, the problem formulations become so complex that the designer has little direct influence on the design.

After a design, the resulting controller should be analyzed with respect to robustness and tested using nonlinear simulations. For the study of robustness, we recommend μ-analysis. If the design is not robust, then the weights should be modified. Sometimes, one might consider synthesizing a μ-optimal controller, but this complexity is rarely necessary in practice. Moreover, one should be careful about combining controller synthesis and analysis into a single step.

10 Controller Structure Design

10.1 Introduction

In previous sections, we considered the general problem formulation in Fig. 59 and stated that the controller design problem is to find a controller K which based on the information in v, generates a control signal u which counteracts the influence of w on z, thereby minimizing the closed loop norm from w to z.

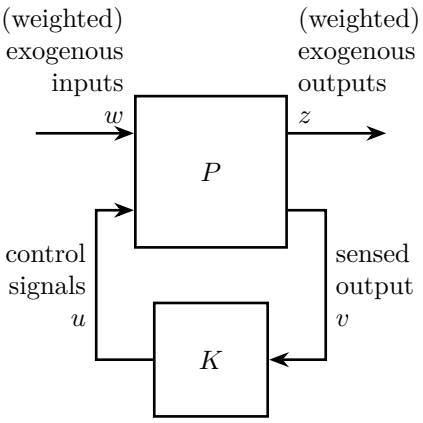


Figure 59 – *General Control Configuration*

In this chapter we are concerned with the **structural decisions** associated with the following selection tasks of control structure design:

- **Controlled outputs**: What are the variables z?
- **Manipulations and measurements**: What are the variable set u and v?
- **Control configuration**: What is the structure of K?
- **Controller type**: What algorithm is used for K?

The distinction between the words under control *structure* and control *configuration* are significant. The *control structure* refers to all structural decisions included in the design of a control system. On the other hand, the *control configuration* refers only to the structuring of the controller K itself.

Ideally, the tasks involved in designing a complete control system are performed sequentially; first a "top down" selection of controller outputs, measurements and inputs, and then a "bottom up" design of the control system in which the selection of the control configuration is the most important decision. However, in practice the tasks are closely related so the procedure may involve iteration.

One important reason for decomposing the control system into a specific *control configuration* is that it may **allow for simple tuning** of the sub-controllers **without the need for a detailed plant model** describing the dynamics and interactions in the process. Multivariable centralized controllers may always outperform decomposed (decentralized) controllers, bus this performance gain must be traded off against the cost of obtaining and maintaining a sufficiently detailed plant model.

The number of possible control structure is usually very large. Fortunately, we can often from physical insight obtain a reasonable choice of controlled outputs, measurements and manipulated inputs.

10.2 Optimization and Control

The selection of controlled outputs involves selecting the variables y to be controlled at given reference values $y \approx r$. The reference value r is usually set at some higher layer in the control hierarchy which is often divided into two layers:

- **Optimization layer**: computes the desired reference commands r
- **Control layer**: implements these commands to achieve $y \approx r$

Additional layers are possible, as is illustrated in Fig. 60 which shows a typical control hierarchy for a chemical plant.

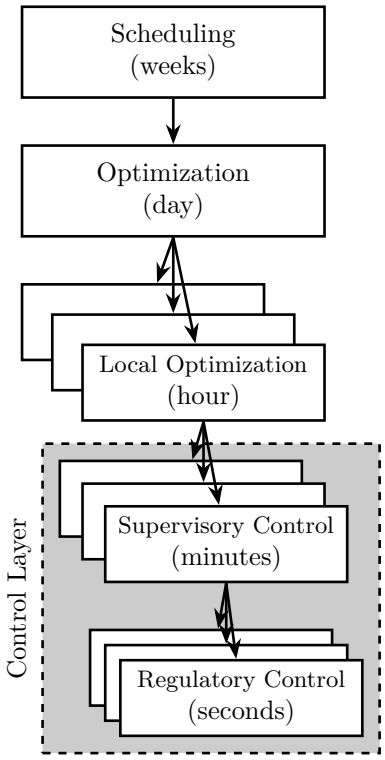
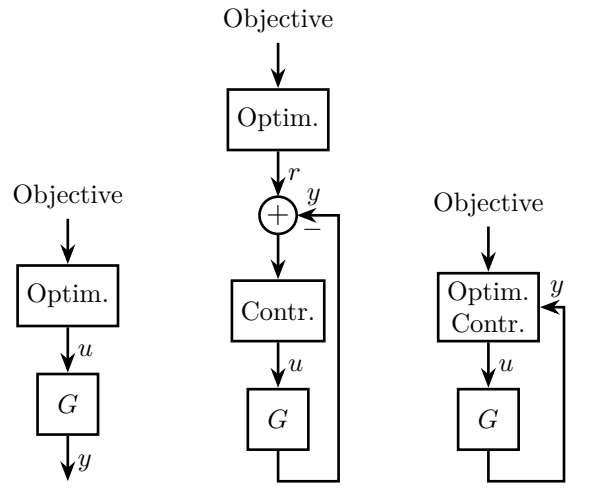


Figure 60 – *Typical control system hierarchy in a chemical plant*

In general, the information flow in such a control hierarchy is based on the higher layer sending reference values (setpoints) to the layer below reporting back any problems achieving this (see Fig. 61b). There is usually a time scale separation between the layers which means that the **setpoints**, as viewed from a given layer, are **updated only periodically**.

The optimization tends to be performed open-loop with limited use of feedback. On the other hand, the control layer is mainly based on feedback information. The **optimization is often based on nonlinear steady-state models**, whereas we often use **linear dynamic models in the control layer**.

From a theoretical point of view, the optimal performance is obtained with a **centralized optimizing controller**, which combines the two layers of optimizing and control (see Fig. 61c). All control actions in such an ideal control system would be perfectly coordinated and the control system would use on-line dynamic optimization based on nonlinear dynamic model of the complete plant. However, this solution is normally not used for a number a reasons, included the cost of modeling, the difficulty of controller design, maintenance, robustness problems and the lack of computing power.



(a) – *Open loop optimization*

(b) – *Closed-loop implementation with separate control layer*

(c) – *Integrated optimization and control*

Figure 61 – *Alternative structures for optimization and control*

10.3 Selection of Controlled Outputs

A **controlled output** is an output variable (usually measured) with an associated control objective (usually a reference value). In many cases, it is clear from a physical understanding of the process what the controlled outputs should be. In other cases, it is less obvious because each control objective may not be associated with a measured output variable.

In the following, we let y denote the selected controller outputs in the control layer. Two distinct questions arise:

1. What variables y should be selected?
2. What is the optimal reference value y_{opt}?

For the first problem, we make the following assumptions:

1. The overall goal can be quantified in terms of a **scalar cost function** J which we want to minimize
2. For a given disturbance d, there exists an optimal value $u_{\text{opt}}(d)$ and corresponding value $y_{\text{opt}}(d)$ which minimizes the cost function J
3. The reference values r for the controlled outputs y should be constant, i.e. r should be independent of the disturbances d

The system behavior is a function of the independent variables u and d: $J = J(u, d)$. For a given disturbance d the optimal value of the cost function is

$$J_{\text{opt}}(d) \triangleq J(u_{\text{opt}}, d) = \min_u J(u, d) \tag{83}$$

In practice $u \neq u_{\text{opt}}$, and we have a loss which can be quantified by $L = J - J_{\text{opt}}$. A reasonable objective for selecting controlled outputs y is to minimize some norm of the loss, for instance the worst-case loss:

$$\Phi \triangleq \max_{d \in \mathcal{D}} | \underbrace{J(u, d) - J(u_{\text{opt}}, d)}_{L} | \tag{84}$$

where $\mathcal{D}$ is the set of possible disturbances.

a Direct Evaluation of Cost

The "brute force" approach for selecting controlled variables is to evaluate the loss for alternative sets of controlled variable. By solving the non linear equations, we evaluate directly the cost function J for various disturbances d. The set of controlled outputs with smallest worst case or average value of J is then preferred. This approach may be time consuming because the solution of the nonlinear equations must be repeated for each candidate set of controlled outputs.

b Linear Analysis

Consider the loss $L = J(u, d) - J_{\text{opt}}(d)$ where d is a fixed disturbance. We make the following additional assumptions:

4. The cost function J is smooth (twice differentiable)
5. The optimization problem is unconstrained. If it is optimal to keep some variable at a constant, then we assume that this is implemented and consider the remaining unconstrained problem

6. The dynamics of the problem can be neglected, that is, **we consider the steady-state control and optimization**

For a fixed d we may express $J(u,d)$ in terms of a Taylor series expansion in u around the optimal point. By neglecting terms of third order and higher, we obtain:

$$J(u,d) = J_{\text{opt}}(d) + \frac{1}{2}(u-u_{\text{opt}}(d))^T \left(\frac{\partial^2 J}{\partial u^2}\right)_{\text{opt}} (u-u_{\text{opt}}(d))$$

This quantifies how $u - u_{\text{opt}}$ affects the cost function. For a fixed d, we have: $y - y_{\text{opt}} = G(u - u_{\text{opt}})$ where G is the steady state gain matrix. Thus, we get:

$$J - J_{\text{opt}} \approx \frac{1}{2}\left(G^{-1}(y-y_{\text{opt}})\right)^T \left(\frac{\partial^2 J}{\partial u^2}\right)_{\text{opt}} G^{-1}(y-y_{\text{opt}})$$

We conclude that we should select y such that:

1. G^{-1} is small: the inputs have a large effect on y
2. $e_{\text{opt}} = r - y_{\text{opt}}(d)$ is small: its optimal value $y_{\text{opt}}(d)$ depends only weakly on the disturbances and other changes
3. $e = y - r$ is small: it is easy to keep the control error e small

Note that $\bar{\sigma}(G^{-1}) = 1/\underline{\sigma}(G)$ and so **we want the smallest singular value of the steady state gain matrix to be large**.

As this depends of scaling, we should first **scale the outputs** such that the expected magnitude of $y_i - y_{i_{\text{opt}}}$ is similar in magnitude for each output, and **scale the inputs** such that the effect of a given deviation $u_j - u_{j_{\text{opt}}}$ on the cost function J is similar for each input.

Controlled Outputs Selection - Procedure

The use of the minimum singular value to select controlled outputs may be summarized in the following procedure:

1. From a (nonlinear) model compute the optimal parameters (inputs and outputs) for various conditions (disturbances, operating points). This yields a "look-up" table for optimal parameter values as a function of the operating conditions
2. From this data, obtain for each candidate output the variation in its optimal value
$$v_i = \frac{(y_{i_{\text{opt,max}}} - y_{i_{\text{opt,min}}})}{2}$$
3. Scale the candidate outputs such that for each output the sum of the magnitudes of v_i and the control error (e_i, including measurement noise n_i) is similar (e.g. $|v_i| + |e_i| = 1$)
4. Scale the inputs such that a unit deviation in each input from its optimal value has the same effect on the cost function J
5. Select as candidates those sets of controlled outputs which corresponds to a large value of $\underline{\sigma}(G)$. G is the transfer function for the effect of the scaled inputs on the scaled outputs

c Summary

Generally, the optimal values of all variables will change with time during operation. If the loss imposed by keeping constant setpoints is acceptable, then we have self-optimizing control. The objective of the control layer is then to keep the controlled outputs at their reference values (which are computed by the optimization layer).

The controlled outputs are often measured, but we may also estimated their values based on other measured variables. We may also use other measurements to improve the control of the controlled outputs, for example, by use of cascade control. Thus, the selection of controlled and measured outputs are two separate issues.

10.4 Selection of Manipulations and Measurements

We are here concerned with the variable sets u and v in Fig. 59. Note that **the measurements** v used by the controller **are in general different from the controlled variables** z because we may not be able to measure all the controlled variables and we may want to measure and control additional variables in order to:

- Stabilize the plant, or more generally change its dynamics
- Improve local disturbance rejection

Stabilization We usually start of controller design by designing a lower-layer controller to stabilize the plant. The issue is then: which outputs and inputs should be used for stabilization? A reasonable objective is to minimize the required input usage of the stabilizing control system.

Local disturbance rejection For measurements, the rule is generally to select those which have a **strong relationship with the controlled outputs**, or which may **quickly detect a major disturbance**. The selected manipulations should have a **large effect on the controlled outputs** and should be located "close" (in terms of dynamic response) to the outputs and measurements.

To evaluate the combinations of manipulations and measurements, one may perform an **input-output controllability analysis** for each combination (e.g. consider the minimum singular values, RHP-zeros, interactions, etc). A more involved approach would be to perform a achievable robust performance analysis. An even more involved (and exact) approach would be to synthesize controllers for optimal robust performance for each candidate combination. However, the number of combination has a combinatorial growth and the analysis may become very time-consuming.

10.5 RGA for Non-Square Plant

A simple but effective tool for selecting inputs and outputs, which avoids to combinatorial problem is the **Relative Gain Array** (RGA) of the "big" transfer matrix G_{all} with all candidates inputs and outputs included:

$$\Lambda = G_{\text{all}} \times G_{\text{all}}^{\dagger^T} \tag{85}$$

Essentially, one may consider not using those manipulations u corresponding to columns in the RGA where the sum of the elements is much smaller than 1.
Similarly, one may consider not using those outputs v corresponding to rows in the RGA where the sum of the elements is much small than 1.

10.6 Control Configuration Elements

We now assume that the measurements, manipulations and controlled outputs are fixed. The available synthesis theories presented in this book result in a *multivariable controller* K which connects all available measurements v with all available manipulations u:

$$u = Kv$$

However, such a "big" controller may not be desirable.

> We define the **control configuration** to be the restrictions imposed on the overall controller K by decomposing it into a set of **local controllers** with predetermined links and with a possibly predetermined design sequence where subcontrollers are designed locally.

Some elements used to build up a specific control configuration are:

- **Cascade controllers**. The output from one controller is the input to another
- **Decentralized controllers**. The control system consists of independent feedback controllers which interconnect a subset of the output measurements with a subset of the manipulated inputs. These subsets should not be used by any other controller
- **Feedforward elements**. Link measured disturbances and manipulated inputs
- **Decoupling elements**. Link one set of manipulated inputs with another set of manipulated inputs. They are used to improve the performance of decentralized control systems.
- **Selectors**: used to select for control, depending on the conditions of the system, a subset of the manipulated inputs or a subset of the outputs

In addition to restrictions on the structure of K, we may impose restrictions on **in which sequence the subcontrollers are designed**. For most decomposed control systems, we design the controllers sequentially, starting with the "fast" or "inner" or "lower-layer" control loops.

The choice of control configuration leads to two different ways of partitioning the control system:

- **Vertical decomposition**. This usually results from a sequential design of the control system
- **Horizontal decomposition**. This usually involves a set of independent decentralized controllers

Of course, a **performance loss** is inevitable if we decompose the control system. For example, if we select a poor configuration at the lower control layer, then this may pose fundamental limitations on the achievable performance (RHP zeros, strong interactions, etc).

a Cascade Control Systems

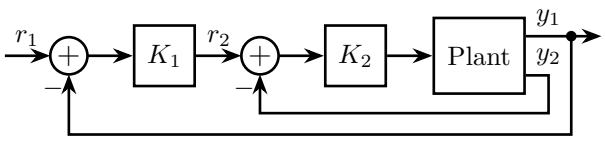
We here use SISO controllers of the form

$$u_i = K_i(s)(r_i - y_i) \tag{86}$$

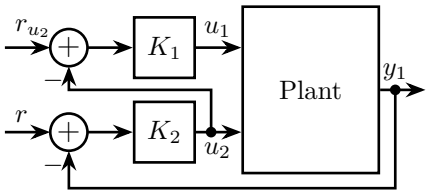
where $K_i(s)$ is a scalar. Then when a SISO control loop is closed, we lose the input u_i as a degree-of-freedom but the reference r_i becomes a new degree-of-freedom.

A cascade control structure results when either of the following two situations arise:

- The reference r_i is an output from another controller. This is the **conventional cascade control** (Fig. 62a)
- The "measurement" y_i is an output from another controller. This is referred to as **input resetting** (Fig. 62b)



(a) – *Extra measurements y_2*



(b) – *Extra inputs u_2*

Figure 62 – *Cascade Implementations*

b Cascade Control: Extra Measurements

Let u be the manipulated input, y_1 the controlled outputs and y_2 the extra measurement. In many cases, we may use y_2 to provide **local disturbance rejection**, **linearization**, or to **reduce the effect of measurement noise**. For example, velocity feedback is frequently used in mechanical systems.

Centralized (parallel) implementation A centralized implementation where K is a 2-inputs-1-output controller may be written

$$u = K(s)(r - y)$$
$$u = K_{11}(s)(r_1 - y_1) + K_{12}(s)(r_2 - y_2)$$

where in most cases $r_2 = 0$ since we do not have a degree-of-freedom to control y_2.

Cascade implementation To obtain an implementation with two SISO controllers, we may cascade the controllers as illustrated in Fig. 62a:

$$r_2 = K_1(s)(r_1 - y_1)$$
$$u_2 = K_2(s)(r_2 - y_2),\ r_2 = \hat{u}_1$$

Note that the output r_2 from the slower primary controller K_1 is not a manipulated plant input, but rather the reference input to the faster secondary controller K_2. Cascades based on measuring the actual manipulated variable ($y_2 = u_m$) are commonly used to **reduce uncertainty and non-linearity at the plant input**. In the general case (Fig. 62a) y_1 and y_2 are not directly related to each other, and this is sometimes referred to as *parallel cascade control*. However, it is common to encounter the situation in Fig. 63 where the primary output y_1 depends directly on y_2 which is a special case of Fig. 62a.

Use of Extra Measurements

With reference to the special (but common) case of cascade control shown in Fig. 63, the use of extra measurements is useful under the following circumstances:

- The disturbance d_2 is significant and G_1 is non-minimum phase. If G_1 is minimum phase, the input-output controllability of G_2 and G_1G_2 are the same and there is no fundamental advantage in measuring y_2
- The plant G_2 has considerable uncertainty associated with it and the inner loop serves to remove the uncertainty. The inner loop $L_2 = G_2K_2$ removes the uncertainty if it is sufficiently fast and yields a transfer function $(I + L_2)^{-1}L_2$ close to I at frequencies where K_1 is active.

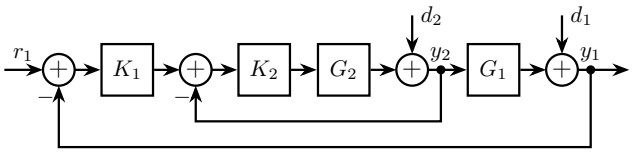


Figure 63 – *Common case of cascade control where the primary output y_1 depends directly on the extra measurement y_2*

In terms of design, it is recommended to first design K_2 to minimize the effect of d_2 on y_1 and then to design K_1 to minimize the effect of d_1 on y_1.

c Cascade Control: Extra Inputs

In some cases we have more manipulated inputs than controlled outputs. These may be used to improve control performance.

Centralized implementation A centralized implementation where K is a 1-input-2-outputs controller may be written

$$u_1 = K_{11}(s)(r - y); \quad u_2 = K_{21}(s)(r - y)$$

Here two inputs are used to control one output. We usually let K_{11} have integral control whereas K_{21} does not. Then $u_2(t)$ will only be used for **transient control** and will return to 0 as $t \to \infty$.

Cascade implementation To obtain an implementation with two SISO controllers we may cascade the controllers as shown in Fig. 62b. We again let input u_2

take care of the **fast control** and u_1 of the **long-term control**. The fast control loop is then

$$u_2 = K_2(s)(r - y)$$

The objective of the other slower controller is then to use input u_1 to reset input u_2 to its desired value r_{u_2}:

$$u_1 = K_1(s)(r_{u_2} - y_1),\ y_1 = u_2$$

and we see that the output from the fast controller K_2 is the "measurement" for the slow controller K_1.
The cascade implementation again has the **advantage of decoupling the design of the two controllers**. It also shows more clearly that r_{u_2}, the reference for u_2, may be used as a degree-of-freedom at higher layers in the control system.

Example: Two layers of cascade control

Consider the system in Fig. 64 with two manipulated inputs (u_2 and u_3), one controlled output (y_1 which should be close to r_1) and two measured variables (y_1 and y_2). Input u_2 has a more direct effect on y_1 than does input u_3 (there is a large delay in $G_3(s)$). Input u_2 should only be used for transient control as it is desirable that it remains close to $r_3 = r_{u_2}$. The extra measurement y_2 is closer than y_1 to the input u_2 and may be useful for detecting disturbances affecting G_1.
Controller K_1 controls the primary output y_1 at its reference r_1 by adjusting the "input" $\hat{u}_1$, which is the reference value for y_2. Controller K_2 controls the secondary output y_2 using input u_2. Finally, controller K_3 manipulates u_3 slowly in order to reset input u_2 to its desired value r_3. We would probably tune the three controllers in the order K_2, K_3, and K_1.

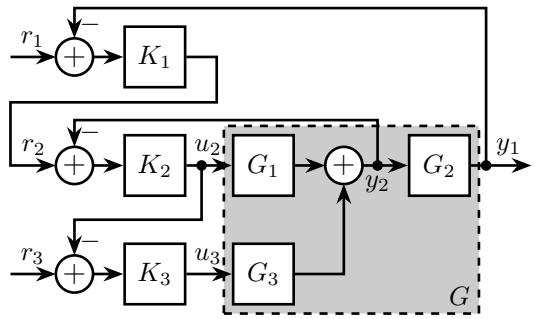


Figure 64 – *Control configuration with two layers of cascade control*

d Selectors

Slip-range control for extra input Sometimes the input constraints make it necessary to add a manipulated input. In this case the control range is often split such that, for example, u_1 is used for control when $y \in [y_{\min}, y_1]$ and u_2 is used when $y \in [y_1, y_{\max}]$.

Selector for too few inputs A completely different situation occurs if there are fewer inputs than outputs. In such case, we cannot control all the outputs independently, so we either need to control all the outputs in some average manner, or we need to make a choice about which outputs are the most important to control. Selectors are often used for the latter option.

e Why use Cascade and Decentralized Control?

Decomposed control configuration can easily become quite complex and difficult to maintain and understand. It may therefore be both simpler and better in terms of control performance to set up the controller design problem as an optimization problem and let the computer do the job, resulting in a **centralized multivariable controller**.

However, there are a **number of reason why cascade and decentralized control are used in practice**. The most important one is the **cost associated with obtaining good plant models**, which are a prerequisite for applying multivariable control. Since cascade and decentralized control systems depend more strongly on feedback rather than models as their source of information, it is usually more important (relative to centralized multivariable control) that the fast control loops be tuned to respond quickly.
The cascade and decentralized control are often easier to understand, their tuning parameters have a direct and "localized" effect, and they tend to be **less sensitive to uncertainty**.

The **main challenge** is then to find a control configuration which allows the controllers to be tuned independently based on a minimum of model information. To be able to tune the controllers independently, we must require that the loops interact only to a limited extent. For example, one desirable property is that the steady-state gain from u_i to y_i in an "inner" loop does not change too much as outer loops are closed.

10.7 Hierarchical and Partial Control

a Partial Control

Partial control involves controlling only a subset of the outputs for which there is a control objective.

We divide the outputs y into two classes:

- y_1 - (temporarily) uncontrolled output
- y_2 - (locally) measured and controlled output

We also subdivide the available manipulated inputs u:

- u_2 - inputs used for controlling y_2

- u_1 - remaining inputs

Four applications of partial control are:

1. **Sequential design on decentralized controllers.** Both y_1 and y_2 have an associated control objective. First, a controller K_2 is designed to control y_2. Then, a controlled K_1 may be designed for the remaining outputs.
2. **Sequential design of conventional cascade control.** The outputs y_2 are additional measured variables which are not important variables in themselves. The reason for controlling y_2 is to improve the control of y_1. The references r_2 are used as degrees-of-freedom for controlling y_1.
3. **"true" partial control.** Both y_1 and y_2 have an associated control objective. We consider whether by controlling only the subset y_2 we can indirectly achieve acceptable control of y_1.
4. **Indirect control.** The outputs y_1 have an associated control objective but are not measured. Instead, we aim at indirectly controlling y_1 by controlling the secondary measured variables y_2.

The table 4 shows clearly the differences between the four applications of partial control. In all cases, there is a control objective associated with y_1 and a feedback involving measurement and control of y_2 and we want:

- The effect of disturbances on y_1 to be small (when y_2 is controlled)
- The control of y_2 using u_2 to be (dynamically) easy

Table 4 – *Applications of partial control*

Control	Meas. and control of y_1?	Control objective for y_2?
Sequ. decentralized	Yes	Yes
Sequ. cascade	Yes	No
"True" partial	No	Yes
Indirect	No	No

By partitioning the inputs and outputs, the overall model $y = Gu$ can be written

$$\begin{aligned} y_1 &= G_{11}u_1 + G_{12}u_2 + G_{d1}d \\ y_2 &= G_{21}u_1 + G_{22}u_2 + G_{d2}d \end{aligned} \tag{87}$$

Assume now that feedback control $u_2 = K_2(r_2 - y_2 - n_2)$ is used for the "secondary" subsystem involving u_2 and y_2 (Fig. 65). We get:

$$\begin{aligned} y_1 =& (G_{11} - G_{12}K_2(I + G_{22}K_2)^{-1}G_{21})u_1 \\ &+ (G_{d1} - G_{12}K_2(I + G_{22}K_2)^{-1}G_{d2})d \\ &+ G_{12}K_2(I + G_{22}K_2)^{-1}(r_2 - n_2) \end{aligned} \tag{88}$$

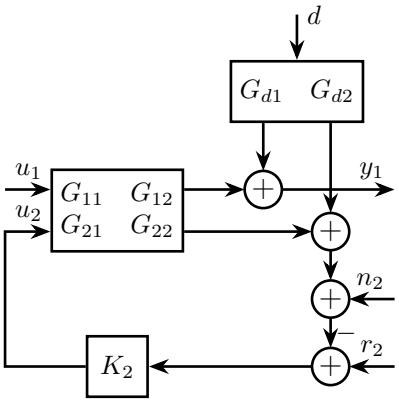


Figure 65 – *Partial Control*

Tight control of y_2 In some cases, we can assume that the control of y_2 is fast compared to the control of y_1 so we may let $K_2 \to \infty$ to get:

$$u_2 = -G_{22}^{-1}G_{d2}d - G_{22}^{-1}G_{21}u_1 + G_{22}^{-1}y_2$$

The dynamics of the system becomes:

$$\begin{aligned} y_1 =& \underbrace{(G_{11} - G_{12}G_{22}^{-1}G_{21})}_{\triangleq P_u}u_1 \\ &+ \underbrace{(G_{d1} - G_{12}G_{22}^{-1}G_{d2})}_{\triangleq P_d}d + \underbrace{G_{12}G_{22}^{-1}}_{\triangleq P_r}\underbrace{(r_2 - e_2)}_{y_2} \end{aligned} \tag{89}$$

where

- P_d is called the **partial disturbance gain**, which is the disturbance gain for a system under perfect partial control
- P_u is the effect of u_1 on y_1 with y_2 perfectly controlled

The obtained dynamics is independent of K_2, but this only applies at frequencies where y_2 is tightly controlled.

b Hierarchical Control and Sequential Design

A **hierarchical control system** results when we design the subcontrollers in a **sequential manner**, usually **starting with the fast loops**. This means that the controller at some higher layer in the hierarchy is designed based on a partially controlled plant.
The idea is to first implement a local lower-layer control system for controlling the outputs y_2. Next, with this lower-layer in place, we design a controller K_1 to control y_1.

The objectives for this hierarchical decomposition are:

- to allow for simple or even on-line tuning of K_2
- to allow the use of longer sampling intervals for K_1
- to allow simple models when designing K_1
- to "stabilize" the plant using K_2 such that it is amenable to manual control

Selection of u_2 and y_2 - Critera

The selection of u_2 and y_2 for use in the lower-layer control system can be done with the following criteria:

- The lower-layer must quickly implement the setpoints computed by the higher layers, that is, the input-output controllability of the subsystem involving the use of u_2 to control y_2 should be good (consider G_{22} and G_{d2})
- The control of y_2 using u_2 should provide local disturbance rejection, that is, it should minimize the effect of disturbances on y_1
- The control of y_2 using u_2 should not impose unnecessary control limitations (RHP-zero, ill-conditioning, etc.) on the remaining control problem which involves using u_1 to control y_1

Sequential design of cascade control systems Consider the conventional cascade control system in Fig. 62a where we have additional "secondary" measurements y_2 with no associated control objective, and the objective is to improve the control of y_1 by locally controlling y_2. The idea is that this should reduce the effect of disturbances and uncertainty on y_1.
From (88), it follows that we should select y_2 and u_2 such that $\|P_d\|$ is small and at least smaller than $\|G_{d1}\|$. These arguments particularly apply at high frequencies. More precisely, we want the input-output controllability of $[P_u\ P_r]$ with disturbance model P_d to be better that of the plant $[G_{11}\ G_{12}]$ with disturbance model G_{d1}.

c "True" Partial Control

We here consider the case where we attempt to leave a set of primary outputs y_1 uncontrolled. This may be possible in cases where the outputs are correlated such that controlling the outputs y_2 indirectly gives acceptable control of y_1.

A set of outputs y_1 may be left uncontrolled only if the effects of all disturbances (including r_2) on y_1, as expressed by the elements in the corresponding partial disturbance gain matrix P_d are less than 1 in magnitude at all frequencies.

To evaluate the feasibility of partial control, one must for each choice of y_2 and u_2, rearrange the system as in (87) and (88), and compute P_d using (89).

d Measurement Selection for Indirect Control

Assume the overall goal is to keep some variable y_1 at a given value r_1, e.g. our objective is to minimize $J = \|y_1 - r_1\|$. We assume that we cannot measure y_1, and instead we attempt to achieve our goal by controlling y_2 at a constant value r_2. For small changes, we may assume linearity and write:

$$y_1 = G_1 u + G_{d1} d$$
$$y_2 = G_2 u + G_{d2} d$$

With feedback control of y_2 we get $y_2 = r_2 + e_2$ where e_2 is the control error. From the above two equations, we obtain

$$y_1 = (G_{d1} - G_1 G_2^{-1} G_{d2}) d + G_1 G_2^{-1} (r_2 + e_2)$$

With $e_2 = 0$ and $d = 0$ this gives $y_1 = G_1 G_2^{-1} r_2$, so r_2 must be chosen such that

$$r_1 = G_1 G_2^{-1} r_2$$

The control error in the primary output is then

$$y_1 - r_1 = \underbrace{(G_{d1} - G_1 G_2^{-1} G_{d2})}_{P_d} d + \underbrace{G_1 G_2^{-1}}_{P_r} e_2 \qquad (90)$$

To minimize J, we should therefore select controlled outputs such that $\|P_d d\|$ and $\|P_r e_2\|$ are small. Note that P_d depends on the scaling of d and y_1. Also the magnitude of e_2 depends on the choice of outputs y_2.

Selecting Controlled Outputs y_2

Scale the disturbances d to be of magnitude 1, and scale the outputs y_2 so that the expected control error e_2 (measurement noise) is of magnitude 1 for each outputs. Then to minimize the control error for the primary output, $J = \|y_1 - r_1\|$, we should select sets of controlled outputs which minimizes $\|[P_d\ P_r]\|$.

10.8 Decentralized Feedback Control

In this section, $G(s)$ is a square plant which is to be controlled using a diagonal controller (Fig. 66).

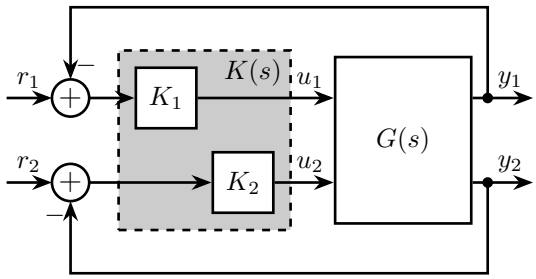


Figure 66 – *Decentralized diagonal control of a 2×2 plant*

The design of **decentralized diagonal control systems** involves two steps:

1. The choice of pairing (control configuration selection)
2. The design of each controller $k_i(s)$

$$K(s) = \text{diag}\{k_i(s)\} = \begin{bmatrix} k_1(s) & & & \\ & k_2(s) & & \\ & & \ddots & \\ & & & k_m(s) \end{bmatrix}$$

a Notations for decentralized diagonal control

$G(s)$ denotes a square $m \times m$ plant with elements g_{ij}. $G^{ij}(s)$ denotes the remaining $(m-1) \times (m-1)$ plant obtained by removing row i and column j in $G(s)$. We introduce:

$$\tilde{G} \triangleq \text{diag}\{g_{ii}\} = \begin{bmatrix} g_{11} & & & \\ & g_{22} & & \\ & & \ddots & \\ & & & g_{mm} \end{bmatrix}$$

The loop transfer function in loop i is denoted $L_i = g_{ii}k_i$.

b RGA as a Measure of the Interaction for Decentralized Control

Let u_j and y_i denote a particular input and output for the multivariable plant $G(s)$ and assume that our task is to use u_j to control y_i. There are two extreme cases:

- **Other loops open**: $u_k = 0, \forall k \neq j$
- **Other loops closed**: $y_k = 0, \forall k \neq i$. It is assumed that the other loop are closed with perfect control which is a good approximation at frequencies within the bandwidth of each loop

We now evaluate the effect $\partial y_i / \partial u_j$ for the two cases:

$$\left(\frac{\partial y_i}{\partial u_j}\right)_{u_k=0, k\neq j} = g_{ij} = [G]_{ij} \tag{91a}$$

$$\left(\frac{\partial y_i}{\partial u_j}\right)_{y_k=0, k\neq i} \triangleq \hat{g}_{ij} = 1/[G^{-1}]_{ji} \tag{91b}$$

The ratio between the gains corresponding the two extreme cases is a useful **measure of interactions** and is defined as the ij'th **relative gain**:

$$\lambda_{ij} \triangleq \frac{g_{ij}}{\hat{g}_{ij}} = [G]_{ij}[G^{-1}]_{ji} \tag{92}$$

The **Relative Gain Array** (RGA) is the corresponding matrix of relative gains:

$$\Lambda(G) = G \times (G^{-1})^T \tag{93}$$

where $\times$ denotes element-by-element multiplication.

Intuitively, we would like to pair variables u_j and y_i so that λ_{ij} is close to 1, because this means that the gain from u_j to y_i is unaffected by closing the other loops. More precisely, we would like to pair such that the rearranged system, with the pairings along the diagonal, has a RGA matrix close to identity.

c Factorization of Sensitivity Function

The magnitude of the off-diagonal elements in G (the interactions) relative to its diagonal elements are given by the matrix

$$E \triangleq (G - \tilde{G})\tilde{G}^{-1} \tag{94}$$

An important relationship for decentralized control is:

$$\underbrace{(I + GK)}_{\text{overall}} = \underbrace{(I + E\tilde{T})}_{\text{interactions}} \quad \underbrace{(I + \tilde{G}K)}_{\text{individual loops}} \tag{95}$$

or equivalently in terms of the sensitivity function:

$$S = \tilde{S}(I + E\tilde{T})^{-1} \tag{96}$$

with

$$\tilde{S} \triangleq (I + \tilde{G}K)^{-1} = \text{diag}\left\{\frac{1}{1 + g_{ii}k_i}\right\}$$

$$\tilde{T} = I - \tilde{S}$$

which contain the sensitivity and complementary sensitivity functions for the individual loops. Note that $\tilde{S}$ is not equal to the matrix of diagonal elements of S.

d Stability of Decentralized Control Systems

Consider a $m \times m$ plant with single-loop controllers. There are $m!$ alternative pairings possible. Thus tools are needed for quickly evaluating alternative pairings. In this section, we first derive **sufficient conditions for stability** which may be used to select promising pairings. We then derive **necessary conditions for stability** which may be used to eliminate undesirable pairings.

Sufficient conditions for stability For decentralized diagonal control, it is desirable that the system can be tuned and operated one loop at a time. Assume therefore that G is stable and each individual loop is stable by itself ($\tilde{S}$ and $\tilde{T}$ are stable). Using the **spectral radius condition** on the factorized S in (96), we have that the overall system is stable (S is stable) if

$$\rho(E\tilde{T}(j\omega)) < 1, \forall \omega \tag{97}$$

Sufficient conditions in terms of E. Assume G is stable and that the individual loops are stable ($\tilde{T}$

is stable). The least conservative approach is to use $\rho(E\tilde{T}) \leq \mu(E)\overline{\sigma}(\tilde{T})$. Then the entire system is closed-loop stable (T is stable) if

$$\overline{\sigma}(\tilde{T}) = \max_i |\tilde{t}_i| < 1/\mu(E) \quad \forall\omega \tag{98}$$

$\mu(E)$ is called the **structured singular value interaction measure**, and is computed with respect to the diagonal structure of $\tilde{T}$ where we may view $\tilde{T}$ as the "design uncertainty".

We usually would like to use integral action in the loops, that is we want $\tilde{T} \approx I$ at low frequencies, i.e. $\overline{\sigma}(\tilde{T}) \approx 1$. Thus, we prefer pairings for which we have $\mu(E) < 1$ at low frequencies where we have tight control. This ensures a "generalized diagonal dominance".

Sufficient conditions in terms of RGA. Suppose the plant $G(s)$ is stable. If the RGA-matrix $\Lambda(G) = I\ \forall\omega$ (which can only arise for a triangular plant $G(s)$), then stability of each of the individual loops implies stability of the entire system.

In most cases, it is sufficient for overall stability to require that $G(j\omega)$ is close to triangular (or $\Lambda(G) \approx I$) at crossover frequencies. This gives the "first pairing rule".

Pairing Rule 1

To achieve stability with decentralized control, prefer pairings such that at frequencies ω around crossover, the rearranged matrix $G(j\omega)$ (with the paired elements along the diagonal) is close to triangular. This is equivalent to requiring $\Lambda(G(j\omega)) \approx I$, i.e. the RGA-number $\|\Lambda(G(j\omega)) - I\|_{\text{sum}}$ should be small.

Necessary steady-state conditions for stability
A desirable property of a decentralized control system is that it has **integrity**, i.e. the closed loop system should remain stable as subsystem controllers are brought in and out of service. Mathematically, the system possesses integrity if it remains stable when the controller K is replace by $\mathbb{E}K$ where $\mathbb{E} = \text{diag}\{\epsilon_i\},\ \epsilon_i = 0, 1$.

An even stronger requirement is that the system remains stable as the gain in various loops are reduced: $0 \leq \epsilon_i \leq 1$.

Decentralized Integral Controllability

The plant $G(s)$ (corresponding to a given pairing with the paired elements along its diagonal) is **Decentralized Integral Controllability** (DIC) if there exists a stabilizing decentralized controller with **integral action in each loop** such that each individual loop may be detuned independently by a factor ϵ_1 $(0 \leq \epsilon_i \leq 1)$ without introducing instability.

Steady-State RGA and DIC. Consider a stable square plant G and a diagonal controller K with integral action in all elements, and assume that the loop transfer function GK is strictly proper. If a pairing of outputs and manipulated inputs corresponds to a **negative steady-state relative gain**, then the closed-loop system has at least one of the following properties:

- The overall closed-loop system is unstable
- The loop with the negative relative gain is unstable by itself
- The closed-loop system is unstable if the loop with the negative relative gain is opened

This can be summarized as follows:

A stable (reordered) plant $G(s)$ is DIC only if $\lambda_{ii}(0) \geq 0$ for all i (99)

e The RGA and RHP-zeros: Further reasons for not pairing on negative RGA elements

With decentralized control, we usually design and implement the controller by tuning and closing one loop at a time in a sequential manner. Assume that we pair on a negative steady-state RGA-element, $\lambda_{ij}(0) < 0$, assume that $\lambda_{ij}(\infty)$ is positive, and assume that the element g_{ij} has no RHP-zero. We have the following implications:

- If we start by closing the loop involving input u_i and y_j, then we will get a RHP-zero in $G^{ij}(s)$ which will limit the performance in the other outputs
- If we end by closing this loop, then we will get a RHP-zero in $\hat{g}_{ij}(s)$ which will limit the performance in output y_i

Pairing Rule 2

For a stable plant, avoid pairings that corresponds to negative steady-state RGA-elements $\lambda_{ij}(0) < 0$

Example - 3×3 plant

$$G(0) = \begin{bmatrix} 10.2 & 5.6 & 1.4 \\ 15.5 & -8.4 & -0.7 \\ 18.1 & 0.4 & 1.8 \end{bmatrix}$$

$$\Lambda(0) = \begin{bmatrix} 0.96 & 1.45 & -1.41 \\ 0.94 & -0.37 & 0.43 \\ -0.90 & -0.07 & 1.98 \end{bmatrix}$$

For a 3×3 plant there are 6 alternative pairings. From the steady state RGA, we see that there is only one positive element in columns 2, and only positive element in row 3, and therefore there is only on possible pairing if we require DIC:

$$u_1 \leftrightarrow y_2, \; u_2 \leftrightarrow y_1, \; u_3 \leftrightarrow y_3$$

Example

$$G(s) = \frac{-s+1}{(5s+1)^2} \begin{bmatrix} 1 & 4 & -26 \\ 6.2 & 1 & -26 \\ 1 & 1 & 1 \end{bmatrix}$$

$$\Lambda(G) = \begin{bmatrix} 1 & 5 & -5 \\ -5 & 1 & 5 \\ 5 & -5 & 1 \end{bmatrix}$$

Only two of the six possible pairings gives positive steady-state RGA-elements: the diagonal pairing on all $\lambda_{ii} = 1$ or the pairing on all $\lambda_{ii} = 5$. Intuitively, one may expect pairing with $\lambda_{ii} = 1$ since it corresponds to pairing on RGA-elements equal to 1. However, the RGA matrix is far from identify, and the RGA-number $\|\Lambda - I\|_{\text{sum}} = 30$ for both alternative. Thus none of the two alternatives satisfy *Pairing Rule 1*, and decentralized control should not be used for this plant.

f Performance of Decentralized Control Systems

To study performance, we use the following factorization

$$S = (I + \tilde{S}(\Gamma - I)^{-1})\tilde{S}\Gamma \tag{100}$$

where Γ is the **Performance Relative Gain Array** (PRGA)

$$\boxed{\Gamma(s) \triangleq \tilde{G}(s)G^{-1}(s)} \tag{101}$$

which is a scaled inverse of the plant.

At frequencies where feedback is effective ($\tilde{S} \approx 0$), $S \approx \tilde{S}\Gamma$ which shows that Γ is important when evaluating performance with decentralized control.

Note that the diagonal elements of the PRGA-matrix are equal to the diagonal elements of the RGA and that the off-diagonal elements of the PRGA depend on the relative scaling on the outputs which is not the case for the RGA.

We will also use the related **Closed-Loop Disturbance Gain** (CLDG) matrix:

$$\boxed{\tilde{G}_d(s) \triangleq \Gamma(s)G_d(s) = \tilde{G}(s)G^{-1}(s)G_d(s)} \tag{102}$$

which depends on both output and disturbance scaling.

Suppose the system has been scaled such that:

- Each disturbance magnitude is less than 1, $|d_k| < 1$
- Each reference change is less than the corresponding diagonal element in R, $|r_j| < R_j$
- For each output the acceptable control error is less than 1, $|e_i| < 1$

Single disturbance Consider a single disturbance, in which case G_d is a vector, and let g_{di} denote the i'th element of G_d. Let $L_i = g_{ii}k_i$ denote the loop transfer function in loop i. Consider frequencies where feedback is effective so $\tilde{S}\Gamma$ is small. Then for **acceptable disturbance rejection** ($|e_i| < 1$) we must with decentralized control required for each loop i

$$\boxed{|1 + L_i| > |\tilde{g}_{di}| \quad \forall i} \tag{103}$$

which is the same as the SISO-condition except that G_d is replaced by the CLDG. In words, $\tilde{g}_{di}$ gives the "apparent" disturbance gain as seen from the loop i when the system is controlled using decentralized control.

Single reference change Consider a change in reference for output j of magnitude R_j. Consider frequencies where feedback is effective. Then for **acceptable reference tracking** ($|e_i| < 1$) we must require for each loop i

$$\boxed{|1 + L_i| > |\gamma_{ij}| \cdot |R_j| \quad \forall i} \tag{104}$$

which is the same as the SISO-condition except for the PRGA-factor $|\gamma_{ij}|$.
Consequently, for performance it is desirable to have small elements in Γ, at least at frequencies where feedback is effective. However, at frequencies close to crossover, stability is the main issue and since the diagonal elements of the PRGA and RGA are equal, we usually prefer to have γ_{ii} close to 1.

g Summary: Controllability Analysis for Decentralized Control

When considering decentralized diagonal control of a plant, one should first check that the plant is controllable with any controller. The next step is to compute

the RGA matrix as a function of frequency, and to determine if one can find a good set of input-output pairs bearing in mind the following:

1. Prefer pairings which have the **RGA-matrix close to identity at frequencies around crossover**, i.e. the RGA-number $\|\Lambda(j\omega) - I\|$ should be small
2. Avoid a pairing ij with negative steady-state RGA elements $\lambda_{ij}(G(0)$
3. Prefer a pairing ij where $g_{ij}(s)$ puts minimal restrictions on the achievable bandwidth. Specifically, the frequency ω_{uij} where $\angle g_{ij}(j\omega_{uij}) = -180°$ should be as large as possible This rule favors parings on variables "close to each other"

When a reasonable choice of pairings have been made, one should rearrange G to have the **paired elements along the diagonal** and perform a **controllability analysis**:

4. Compute the CLDG and PRGA, and plot these as a function of frequency
5. For systems with many loops, it is best to perform the analysis one loop at the time, that is, for each loop i, plot $|\tilde{g}_{dik}|$ for each disturbance k and plot $|\gamma_{ij}|$ for each reference j. For performance, we need $|1 + L_i|$ to be larger than each of these:

$$|1 + L_i| > \max_{k,j}\{|\tilde{g}_{dik}|, |\gamma_{ij}|\} \tag{105}$$

   To achieve stability of the individual loops, one must analyze $g_{ii}(s)$ to ensure that the bandwidth required by (105) is achievable. Note that RHP-zeros in the diagonal elements may limit achievable decentralized control, whereas they may not pose any problems for a multivariable controller. Since with decentralized control, we usually want to use simple controllers, the achievable bandwidth in each loop will be limited by the frequency where $\angle g_{ii}$ is $-180°$
6. Check for constraints by considering the elements of $G^{-1}G_d$ and make sure that they do not exceed one in magnitude within the frequency range where control is needed. Equivalently, one may for each loop i, plot $|g_{ii}|$ and the requirement is then that

$$|g_{ii}| > |\tilde{g}_{dik}| \quad \forall k \tag{106}$$

   at frequencies where $|\tilde{g}_{dik}|$ is larger than 1. This provides a direct generalization of the requirement $|G| > |G_d|$ for SISO systems.

If the plant is not controllable, then one may consider another choice of pairing and go back to Step 4. If one still cannot find any pairing which are controllable, then one should consider multivariable control.

7. If the chosen pairing is controllable, then (105) tells us how large $|L_i| = |g_{ii}k_i|$ must be. This can be used as a basis for designing the controller $k_i(s)$ for loop i

h Sequential Design of Decentralized Controllers

Usually the local controllers $k_i(s)$ are designed locally and then all the loops are closed. One problem with this is that the **interactions** may cause the overall system T so be unstable, even though the local loops $\tilde{T}$ are stable. This will not happen if the plant is **diagonally dominant**, such that we satisfy, for example $\overline{\sigma}(\tilde{T}) < 1/\mu(E)$.

The stability problem is avoided if the controllers are **designed sequentially** when, for example, the bandwidths of the loops are quite different. In this case, the outer loops are tuned with the inner loops in place, and each step may be considered as a SISO control problem. In particular, overall stability is determined by m SISO stability conditions. However, the issue of performance is more complicated because the closing of a loop may cause "disturbances" (interactions) into a previously designed loop. The engineer must then go back and redesign a loop that has been designed earlier. Thus sequential design may involve many iterations.

i Conclusion on Decentralized Control

A number of **conditions for the stability**, e.g. (98) and (99), and **performance**, e.g. (103) and (104), of decentralized control systems have been derived.
The conditions may be useful in **determining appropriate pairings of inputs and outputs** and the **sequence in which the decentralized controllers should be designed**.
The conditions are also useful in an **input-output controllability analysis** for determining the viability of decentralized control.

11 Model Reduction

11.1 Introduction

Modern controller design methods such as $\mathcal{H}_\infty$ and LQG, produce controllers of order at least equal to that of the plant, and usually higher because of the inclusion of weights. These control laws may be **too complex** with regards to **practical implementation** and simpler designs are then sought. For this purpose, one can **either reduce the order of the plant model prior to controller design, or reduce the controller in the final stage**.

Model Reduction Problem

Given a high-order linear time-invariant stable model G, find a low-order approximation G_a such that the infinity ($\mathcal{H}_\infty$ or $\mathcal{L}_\infty$) norm of the difference $\|G - G_a\|_\infty$ is small.

By model order, we mean the dimension of the state vector in a minimal realization. This is sometimes called the **McMillan degree**.

So far we have only been interested in the infinity ($\mathcal{H}_\infty$) norm of stable systems. But the error $G - G_a$ may be unstable and **the definition of the infinity norm needs to be extended to unstable systems**.

$\mathcal{L}_\infty$ - Definition

$\mathcal{L}_\infty$ defines the set of rational functions which have no poles on the imaginary axis, it includes $\mathcal{H}_\infty$, and its norm (like $\mathcal{H}_\infty$) is given by

$$\|G\|_\infty = \sup_\omega \overline{\sigma}(G(j\omega)) \tag{107}$$

We will describe three main methods for this problem:

- **Balanced truncation**
- **Balanced residualization**
- **Optimal Hankel norm approximation**

Each method gives a **stable approximation** and a **guaranteed bound on the error in the approximation**. We will further show how the methods can be employed to reduce the order of an **unstable** model G.

All these methods start from a special state-space realization of G referred to as **balanced**. We will describe this realization, but first we will show how the techniques of truncation and residualization can be used to remove the high frequency or fast modes of a state-space realization.

11.2 Truncation and Residualization

Let (A, B, C, D) be a minimal realization of a stable system $G(s)$, and partition the state vector x, of dimension n, into $\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ where x_2 is the vector of $n-k$ states we wish to remove. With approximate partitioning of A, B and C, the state space equations become

$$\begin{aligned} \dot{x}_1 &= A_{11}x_1 + A_{12}x_2 + B_1 u \\ \dot{x}_2 &= A_{21}x_1 + A_{22}x_2 + B_2 u \\ y &= C_1 x_1 + C_2 x_2 + Du \end{aligned} \tag{108}$$

a Truncation

A k-th order truncation of the realization $G \triangleq (A, B, C, D)$ is given by $G_a \triangleq (A_{11}, B_1, C_1, D)$. The truncated model G_a is equal to G at infinite frequency $G(\infty) = G_a(\infty) = D$, but apart from this, we cannot say anything for the general case about the relationship between G and G_a.

If however, A is in **Jordan form**, then it is easy to **order the states** so that x_2 corresponds to **high frequency or fast modes**.

Modal Truncation For simplicity, assume that A has been diagonalized so that

$$A = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_n \end{bmatrix}, \quad B = \begin{bmatrix} b_1^T \\ b_2^T \\ \vdots \\ b_n^T \end{bmatrix}$$
$$C = \begin{bmatrix} c_1, c_2, \dots, c_n \end{bmatrix}$$

Then, if the λ_i are ordered so that $|\lambda_1| < |\lambda_2| < \dots$, the fastest modes are removed from the model after truncation. The difference between G and G_a following a k-th order model truncation is given by

$$G - G_a = \sum_{i=k+1}^{n} \frac{c_i b_i^T}{s - \lambda_i}$$

and therefore

$$\|G - G_a\|_\infty \leq \sum_{i=k+1}^{n} \frac{\overline{\sigma}(c_i b_i^t)}{|\mathrm{Re}(\lambda_i)|} \tag{109}$$

It is interesting to note that the error depends on the residues $c_i b_i^T$ as well as the λ_i. The distance of λ_i from the imaginary axis is therefore not a reliable indicator of whether the associated mode should be included in the reduced order model or not.

An advantage of modal truncation is that the poles of the truncated model are a subset of the poles of the original model and therefore **retain any physical interpretation** they might have.

b Residualization

In truncation, we discard all the states and dynamics associated with x_2. Suppose that instead of this, we simply set $\dot{x}_2 = 0$, i.e. we residualize x_2, in the state-space equations. One can then solve for x_2 in terms of x_1 and u, and back substitution of x_2, then gives

$$\dot{x}_1 = (A_{11} - A_{12}A_{22}^{-1}A_{21})x_1 + (B_1 - A_{12}A_{22}^{-1}B_2)u$$
$$y = (C_1 - C_2A_{22}^{-1}A_{21})x_1 + (D - C_2A_{22}^{-1}B_2)u$$

And let assume A_{22} is invertible and define

$$A_r \triangleq A_{11} - A_{12}A_{22}^{-1}A_{21} \quad B_r \triangleq B_1 - A_{12}A_{22}^{-1}B_2$$
$$C_r \triangleq C_1 - C_2A_{22}^{-1}A_{21} \quad D_r \triangleq D - C_2A_{22}^{-1}B_2$$

The reduced order model $G_a(s) = (A_r, B_r, C_r, D_r)$ is called a **residualization** of $G(s) = (A, B, C, D)$. Usually (A, B, C, D) will have been put into **Jordan form**, with the eigenvalues ordered so that x_2 contains the fast modes.

Model reduction by residualization is then equivalent to singular perturbation approximation, where **the derivatives of the fastest states are allowed to approach zero** with some parameter ϵ.

An important property of residualization is that **it preserves the steady-state gain of the system**:

$$G_a(0) = G(0) \tag{110}$$

This should be no surprise since the residualization process sets derivatives to zero, which are zero anyway at steady-state. But it is in stark contrast to truncation which retains the system behavior at infinite frequency. This contrast between truncation and residualization follows from the simple bilinear relationship $s \to \frac{1}{s}$ which relates the two.

It is clear that **truncation is to be preferred when accuracy is required at high frequencies**, whereas **residualization is better for low frequency modelling**.

Both methods depend to a large extent on the original realization and we have suggested to use of the Jordan form. A better realization, with many useful properties, is the **balanced realization**.

11.3 Balanced Realization

A balanced realization is an asymptotically stable minimal realization in which the **controllability and observability Gramiams are equal and diagonal**.

Let (A, B, C, D) be a minimal realization of a stable, rational transfer function $G(s)$, then (A, B, C, D) is called **balanced** if the solutions to be following Lyapunov equations

$$AP + PA^T + BB^T = 0 \tag{111a}$$
$$A^TQ + QA + C^TC = 0 \tag{111b}$$

are $P = Q = \text{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n) \triangleq \Sigma$, where $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n > 0$. P and Q are the **controllability and observability Gramiams**, also defined by

$$P \triangleq \int_0^\infty e^{At}BB^Te^{A^Tt}dt \tag{112a}$$
$$Q \triangleq \int_0^\infty e^{A^Tt}C^TCe^{At}dt \tag{112b}$$

Σ is therefore simply referred to as the Gramiam of $G(s)$. The σ_i are the **ordered Hankel singular values** of $G(s)$, more generally defined as $\sigma_i \triangleq \lambda_i^{\frac{1}{2}}(PQ)$, $i = 1, \ldots, n$. Notice that $\sigma_1 = \|G\|_H$ is the Hankel norm of $G(s)$.

In balanced realization the value of each σ_i is associated with a state x_i of the balanced system.

> The size of σ_i is a relative measure of the contribution that x_i makes to the input-output behavior of the system.

Therefore if $\sigma_1 \gg \sigma_2$, then the state x_1 affects the input-output behavior much more than x_2, or indeed any other state because of the ordering of the σ_i.

After balancing a system, each state is just as controllable as it is observable, and a measure of a state's joint observability and controllability is given by its associated Hankel singular value. This property is fundamental to the model reduction methods in the remainder of this chapter which work by removing states having little effect on the system's input-output behavior.

11.4 Balanced Truncation and Balanced Residualization

Let the balanced realization (A, B, C, D) of $G(s)$ and the corresponding Σ be partitioned compatibly as

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \quad B = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}$$
$$C = \begin{bmatrix} C_1 & C_2 \end{bmatrix}, \quad \Sigma = \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} \tag{113}$$

where

$$\Sigma_1 = \text{diag}(\sigma_1, \sigma_2, \ldots, \sigma_k)$$
$$\Sigma_2 = \text{diag}(\sigma_{k+1}, \sigma_{k+2}, \ldots, \sigma_n), \ \sigma_k > \sigma_{k+1}$$

Balanced Truncation The reduced order model given by (A_{11}, B_1, C_1, D) is called a **balanced truncation** of the full order system $G(s)$. The idea of

balancing truncation is thus to first make a balanced realization of the system and then to discard the states corresponding to small Hankel singular values.
A balanced truncation is also a balanced realization, and the infinity norm of the error between $G(s)$ and the reduced order system $G_a(s)$ is bounded by twice the sum of the last $n-k$ Hankel singular values, i.e. twice the trace of Σ_2:

$$\|G(s) - G_a(s)\|_\infty \leq 2 \cdot \mathrm{Tr}(\Sigma_2) \tag{114}$$

For the case of repeated Hankel singular values, each repeated Hankel singular value is to be counted only once in calculating the sum.
Useful algorithms that compute balanced truncations without first computing a balanced realization still require the computation of the observability and controllability Gramiam, which can be a problem if the system to be reduced is of very high order.

Balanced Residualization In balanced truncation above, we discarded the least controllable and observable states corresponding to Σ_2. In balanced residualization, we simply set to zero the derivatives of all these states.

Theorem Let $G(s)$ be a stable rational transfer function with Hankel singular values $\sigma_1 > \sigma_2 > \cdots > \sigma_N$ where each σ_i has multiplicity r_i and let $G_a^k(s)$ be obtained by truncating or residualizing the balanced realization of $G(s)$ to the first $(r_1 + r_2 + \cdots + r_k)$ states. Then

$$\|G(s) - G_a^k(s)\|_\infty \leq 2(\sigma_{k+1} + \sigma_{k+2} + \cdots + \sigma_N) \tag{115}$$

11.5 Optimal Hankel Norm Approximation

In this approach to model reduction, the problem that is directly addressed is the following: given a stable model $G(s)$ of order n, find a reduced order model $G_h^k(s)$ of degree k such that the Hankel norm of the approximation error, $\|G(s) - G_h^k(s)\|_H$, is minimized.

Hankel Norm - Definition

The Hankel norm of any stable transfer function $E(s)$ is defined as

$$\|E(s)\|_H \triangleq \rho^{\frac{1}{2}}(PQ) \tag{116}$$

where P and Q are the controllability and observability Gramiams of $E(s)$.

So in the optimization we seek an error which is in some sense closest to being completely unobservable and completely uncontrollable.
The infinity norm bound on the approximate error for the optimal Hankel norm approximation is better than for balanced truncation and residualization. This is shown with the following theorem.

Theorem Let $G(s)$ be a stable, square, transfer function $G(s)$ with Hankel singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k \geq \sigma_{k+1} = \sigma_{k+2} = \cdots = \sigma_{k+l} > \sigma_{k+l+1} \geq \cdots \geq \sigma_n > 0$. An optimal Hankel norm approximation of order k, $G_h^k(s)$, can be constructed as follows.
Let (A, B, C, D) be a balanced realization of $G(s)$ with the Hankel singular values reordered so that the Gramiam matrix is

$$\begin{aligned}\Sigma &= \mathrm{diag}(\sigma_1, \ldots, \sigma_k, \sigma_{k+l+1}, \ldots, \sigma_n, \sigma_{k+1}, \ldots, \sigma_{k+l}) \\ &\triangleq \mathrm{diag}(\Sigma_l, \sigma_{k+1} I)\end{aligned}$$

Partition (A, B, C, D) to conform with Σ

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \; B = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}, \; C = \begin{bmatrix} C_1 & C_2 \end{bmatrix}$$

Define $(\hat{A}, \hat{B}, \hat{C}, \hat{D})$ by

$$\hat{A} \triangleq \Gamma^{-1}\left(\sigma_{k+1}^2 A_{11}^T + \sigma_1 A_{11} \Sigma_1 - \sigma_{k+1} C_1^T U B_1^T\right) \tag{117a}$$

$$\hat{B} \triangleq \Gamma^{-1}\left(\sigma_1 B_1 + \sigma_{k+1} C_1^T U\right) \tag{117b}$$

$$\hat{C} \triangleq C_1 \Sigma_1 + \sigma_{k+1} U B_1^T \tag{117c}$$

$$\hat{D} \triangleq D - \sigma_{k+1} U \tag{117d}$$

where U is a unitary matrix satisfying

$$B_2 = -C_2^T U \quad \text{and} \quad \Gamma \triangleq \Sigma_1^2 - \sigma_{k+1}^2 I$$

The matrix $\hat{A}$ has k "stable" eigenvalues; the remaining ones are in the open right-half plane. Then

$$G_h^k(s) + F(s) = \left[\begin{array}{c|c} \hat{A} & \hat{B} \\ \hline \hat{C} & \hat{D} \end{array}\right]$$

where $G_h^k(s)$ is a stable optimal Hankel norm approximation of order k, and $F(s)$ is an anti-stable (all poles in the open right-half plane) transfer function of order $n-k-l$. The Hankel norm of the error between G and the optimal approximation G_h^k is equal to the $(k+1)$'th Hankel singular value of G:

$$\|G - G_h^k\|_H = \sigma_{k+1}(G) \tag{118}$$

11.6 Model Reduction - Practical Summary

a Reduction of model

Three reduction techniques have been discussed here: balanced residualization, balance truncation and optimal Hankel norm approximation.
It is sometimes desirable to have the steady-state gain of the reduced plant model the same as the full order model. For instance, this is the case if we want to use feedforward control. The truncated and optimal Hankel norm approximated systems do not preserve the

steady-state gain and they have to be **scaled**, i.e. the model approximation G_a is replaced by $G_a W_s$ where $W_a = G_a(0)^{-1}G(0)$, $G(s)$ being the full order model. However, this scaling generally introduced **large model errors at other frequencies**.

> Hence **residualization** is to be preferred whenever low frequency matching is desired.

b Reduction of a 2 degrees-of-freedom controller

Let's consider a 2 degrees-of-freedom controller $K = [K_1 \ K_2]$. In order ensure perfect steady-state tracking, i.e. to match T_{ref} at steady-state, a prefilter W_i is added to scale the controller: $K = [K_1 W_i \ K_2]$.
There are two approaches for order reduction:

1. the scaled controller $[K_1 W_i \ K_2]$ is reduced. A balanced residualization of the controller preserves the controller's steady state gain and would not need to be scaled again. Reductions via truncation and optimal Hankel norm approximation techniques, however, lose the steady-state gain and reduced controllers would need to be re-scaled to match $T_{\text{ref}}(0)$
2. the full order controller $[K_1 \ K_2]$ is reduced without first scaling the prefilter. In which case, scaling is done after reduction. A larger scaling is generally required for the truncated and optimal Hankel norm approximated controllers and this gives poorer model matching at other frequencies.

In both cases, the balanced residualization is preferred.

11.7 Reduction of Unstable Models

Balanced truncation, balanced residualization and optimal Hankel norm approximation only apply to stable models. In this section we briefly present two approaches for reducing the order of an unstable model.

a Stable Part Model Reduction

The unstable model can be first decomposed into its stable and anti-stable parts:

$$G(s) = G_u(s) + G_s(s) \tag{119}$$

where $G_u(s)$ has all its poles in the closed right-half plane and $G_s(s)$ has all its poles in the open left-half plane. Balanced truncation, balanced residualization or optimal Hankel norm approximation can then be applied to the stable part $G_s(s)$ to find a reduced order approximation $G_{sa}(s)$. This is then added to the anti-stable part to give

$$G_a(s) = G_u(s) + G_{sa}(s) \tag{120}$$

as an approximation to the full order model $G(s)$.

b Coprime Factor Model Reduction

The coprime factors of a transfer function $G(s)$ are stable, and therefore we could reduce the order of these factors using balanced truncation, balanced residualization or optimal Hankel norm approximation:

- Let $G(s) = M^{-1}(s)N(s)$, where $M(s)$ and $N(s)$ are stable left-coprime factors of $G(s)$
- Approximate $[N \ M]$ of degree n by $[N_a \ M_a]$ of degree $k < n$, using balanced truncation, balanced residualization or optimal Hankel norm approximation
- Realize the reduced order transfer function $G_a(s)$, or degree k, by $G_a(s) = M_a^{-1}(s)N_a(s)$

Theorem Let (N, M) be a normalized left-coprime factorization of $G(s)$ of degree n. Let $[N_a, \ M_a]$ be a degree k **balanced truncation** of $[N \ M]$ which has Hankel singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k \geq \sigma_{k+1} \geq \cdots \geq \sigma_n > 0$. Then (N_a, M_a) is a normalized left-coprime factorization of $G_a = M_a^{-1}N_a$, and $[N_a, \ M_a]$ has Hankel singular values $\sigma_1, \sigma_2, \ldots, \sigma_k$.

11.8 Conclusion

We have presented and compared three main methods for model reduction based on balanced realizations: **balanced truncation**, **balanced residualization** and **optimal Hankel norm approximation**.

Residualization, unlike truncation and optimal Hankel norm approximation, preserves the steady-state gain of the system, and like truncation, it is simple and computationally inexpensive. It is observed that truncation and optimal Hankel norm approximation perform better at high frequencies, where residualization performs better at low and medium frequencies, i.e. up to the critical frequencies.

Thus **for plant model reduction**, where models are not accurate at high frequencies to start with, **residualization would seem to be a better option**. Further, if the steady state gains are to be kept unchanged, truncated and optimal Hankel norm approximated systems require scaling, which may result in large errors. In such a case, too, residualization would be preferred choice.

For **controller reduction**, we have shown in a two degrees-of-freedom example, the importance of scaling and steady-state gain matching.
In general, steady-state gain matching may not be crucial, but the matching should usually be good near the desired closed-loop bandwidth. Balanced residualization has been seen to perform close to the full order system in this frequency range. Good approximation at high frequencies may also sometimes be desired. In such a case, using truncation or optimal Hankel norm

approximation with appropriate frequency weightings may yield better results.