ver 1.5, 220330 Elling W. Jacobsen, KTH

5 EL2520 Lecture notes 5: Basics of Multivariable Control Systems, Decentralized Control and the RGA

So far the focus of this course has been on SISO systems. However, as stressed before, the main purpose has been to introduce a framework that enables us to more or less treat SISO and MIMO systems in the same way. Historically, the classical methods developed in the input-output, or frequency, domain by the pioneers of the likes of Nyquist, Bode and Black in the 1930s-1940s, were limited to SISO systems and not extendible to the MIMO case. As part of the space race in the 1950s-1960s one needed to solve challenging MIMO control problems and then moved to state space methods combined with optimization theory, resulting in the now famous state space optimal control theory. Some key names in this period were Bellman, Pontryagin and Kalman. Despite being an elegant solution to many control problems, one major drawback of the state space methods were that there was no obvious connection to the classial results in the frequency domain. In particular, while the latter addressed model uncertainty, e.g., in terms of phase and amplitude margins, as well as fundamental limitations, these issues were not easily addressed in the state-space domain. Indeed, in the 1980s George Zames at MIT noted that state space optimal methods for MIMO systems could be highly sensitive to model errors. To address this problem, he introduced an input-output framework suitable for MIMO systems, based on the H_∞-norm. On the basis of this, powerful results on robustness and performance limitations in MIMO control systems could be developed. Other key names from this period are John Doyle and Gunter Stein. It is the latter theory which is the focus of this course. However, as we shall see later in the course, there are also clear links to the state-space optimal control theory, not the least when it comes to solving the H_∞-optimal control problems formulated in the input-output (frequency) domain but typically solved in state-space.

Before extending the SISO results on robust stability, performance specifications and performance limitations to MIMO systems, we need to introduce some basic properties of MIMO systems, such as poles, zeros and gain. We will also briefly discuss more simplistic approaches to MIMO control, namely decentralized control and decoupling.

5.1 Transfer-Matrices, Poles and Zeros

Recall that a multi-input-multi-output (MIMO) system have inputs and outputs that are vector quantities

$$u = \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_m \end{pmatrix} \quad ; \quad y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_p \end{pmatrix}$$

while the corresponding transfer-function $G(s)$ is a $p \times m$ matrix in which the ij-th element $G_{ij}(s)$ is the transfer-function from u_j to y_i. Recall also that if we have a linear time-invariant model on state-space form

$$\begin{aligned} \dot{x} &= Ax(t) + Bu(t) \ ; \quad x \in \mathbb{R}^n, u \in \mathbb{R}^m \\ y(t) &= Cx(t) + Du(t) \ ; \quad y \in \mathbb{R}^p \end{aligned}$$

then the transfer-matrix is

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

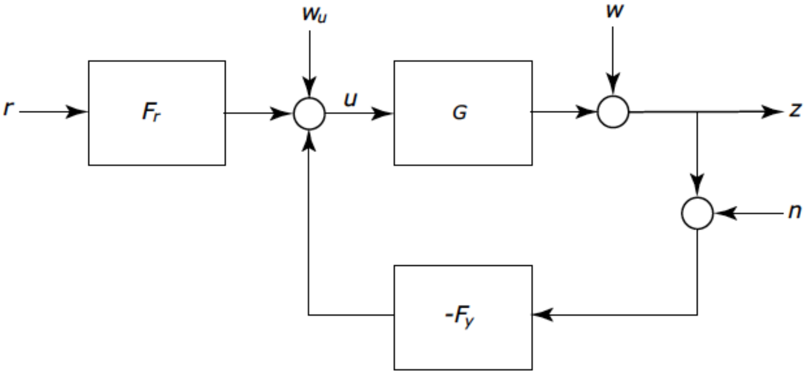

Figure 5.1: Block-diagram for two degree of freedom control system.

Consider the block-diagram in Figure 5.1. To derive closed-loop transfer-functions between inputs and outputs, we can as usual employ simple block-diagram algebra, i.e., write down relations from the block-diagram and solve for the inputs and outputs of interest. However, a simple rule that can be used instead is the following

1. Start from the output and move against the signal flow towards the input.
2. Write down the blocks, left to right, as you meet them
3. When you exit the loop, postmultiply by the term $(I+L)^{-1}$ where L is the loop-gain evaluated from the exit and against the signal flow
4. If there are parallell paths, simply add them toghether

Also useful is the so called "push through" rule (for matrices of appropriate dimensions)

$$A(I + BA)^{-1} = (I + AB)^{-1}A$$

For instance, consider the transfer-function from the noise n to the output z in Figure 5.1. Moving from z we get

$$z = -GF_y(I + GF_y)^{-1}n = -(I + GF_y)^{-1}GF_yn = -Tn$$

Also

$$z = (I + GF_y)^{-1} w = Sw \ ; \quad z = G(I + F_y G)^{-1} w_u = (I + GF_y)^{-1} G w_u = SGw_u$$

and

$$u = (I + F_y G)^{-1} w_u = S_u w_u$$

Note that $(I + GF_y)^{-1} \neq (I + F_y G)^{-1}$ and hence the sensitivity at the input S_u differs from the sensitivity at the output S.

The pole polynomial $\lambda(s)$ of a linear system is the characteristic polynomial of the A matrix in the state space description

$$\lambda(s) = \det(sI - A)$$

The poles p_i are the zeros of the pole polynomial $\lambda(p_i) = 0$. The poles can be computed from the transfer-matrix using the following theorem

Theorem: The pole polynomial $\lambda(s)$ of a system with transfer-matrix $G(s)$ is the least common denominator of all minors of $G(s)$.

Recall that a minor of a matrix M is the determinant of any square matrix obtained by deleting one or more columns and rows of M. For a 2×2 matrix the minors are the matrix elements and the determininant of the matrix itself. Consider the example

$$G(s) = \begin{pmatrix} \frac{2}{s+1} & \frac{3}{s+2} \\ \frac{1}{s+1} & \frac{1}{s+1} \end{pmatrix}$$

which has minors $\frac{2}{s+1}, \frac{3}{s+2}, \frac{1}{s+1}$ and $\det G(s) = \frac{1-s}{(s+1)^2(s+2)}$. The characteristic polynomial is then $\lambda(s) = (s+1)^2(s+2)$ and hence there are two poles in $s = -1$ and one pole in $s = -2$.

The zeros of a SISO transfer-function $G(s)$ are those values of s where $G(s) = 0$. For a MIMO transfer-matrix, the most common corresponding definition of zeros (and the one we use in this course) are the values of s where $G(s)$ looses rank, i.e., has lower rank than for other values of s (except those of possible other zeros).

Theorem: The zero polynomial of $G(s)$ is the greatest common divisor of the maximal minors of $G(s)$, normed so that they have the pole polynomial of $G(s)$ as denominator. The zeros of $G(s)$ are the roots of the zero polynomial.

Note that the maximal minor of a square matrix is the determinant of the matrix. Consider again the 2×2 system above, for which the determinant is

$$\det G(s) = \frac{1-s}{(s+1)^2(s+2)}$$

which is normed so that the denominator is the pole polynomial. Thus, the zero polynomial is $(1 - s)$ and there is one zero at $s = 1$. You can easily check that $G(1)$ has rank 1.

When a matrix has a singularity or becomes rank deficient, these is always an associated subspace. For instance, if $G(z)$ is rank deficient then

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

Here u_z and y_z are the input and output *zero directions*, respectively. Thus, u_z is in the nullspace of $G(z)$ while y_z^H is in the left nullspace of $G(z)$ where superscript H denotes conjugate (Hermetian) transpose. The latter follows from the fact that $y_z^H G(z) = 0 \cdot u_z^H$ where we have assumed that both directions are normalized to have length one, i.e., $y_z^H y_z = 1, u_z^H u_z = 1$. Likewise, at a pole p we have essentially

$$G(p)u_p = \infty \cdot y_p$$

where u_p and y_p are the input and output *pole directions*. We shall see in Lecture 6 that these directions are important when analyzing the limitations imposed on control performance by RHP poles and RHP zeros. Also note that a pole and a zero located at the same position in two transfer-matrices G_1 and G_2, respectively, does not necessarily cancel eachother when forming G_1G_2. They only cancel if the corresponding pole and zero directions coincide.

5.2 Gain of a MIMO system

For a linear SISO system $G(s)$ we know that the amplification of the system depends on the frequency ω of the input, i.e.,

$$\frac{|Y(i\omega)|}{|U(i\omega)|} = |G(i\omega)|$$

Recall from Lecture 1 that the *gain* of a system was defined as the maximum amplification, and for a linear SISO system this is then

$$\sup_u \frac{\|y\|_2}{\|u\|_2} = \sup_\omega |G(i\omega)| = \|G\|_\infty$$

Thus, the gain a linear time invariant SISO system is the $\mathcal{H}_\infty$-norm of the transfer-function.

In a MIMO system, the amplification of the system will also depend on the direction of the input vector. To see this, consider first a static linear system

$$y = Ax$$

where A is real or complex constant matrix of dimension $m \times r$. As before, we consider amplification in terms of the 2-norm of the output over the 2-norm of the input, i.e., $|y|/|x|$ where $|\cdot|$ denotes the Euclidian 2-norm. We have

$$|y|^2 = |Ax|^2 = (Ax)^H Ax = x^H A^H Ax$$

Here $A^H A$ is a symmetric positive definite matrix and hence

$$|x|^2 \lambda_{min}(A^H A) \leq |y|^2 \leq |x|^2 \lambda_{max}(A^H A)$$

(follows from $A^HAx = \lambda x \;\Rightarrow\; x^HA^HAx = \lambda x^Hx = \lambda|x|^2$). We then get the lower and upper bounds on the amplification

$$\underbrace{\sqrt{\lambda_{min}(A^HA)}}_{\underline{\sigma}(A)} \leq \frac{|y|}{|x|} \leq \underbrace{\sqrt{\lambda_{max}(A^HA)}}_{\bar{\sigma}(A)}$$

where $\underline{\sigma}(A)$ and $\bar{\sigma}(A)$ are the smallest and largest *singular values* of A, respectively. The corresponding input and output directions can be obtained from the Singular Value Decomposition (SVD) of A

$$A = U\Sigma V^H = [u_1 \;\cdots\; u_r]\texttt{diag}(\sigma_i)[v_1 \;\cdots\; v_r]^H = \sum_{i=1}^{r} \sigma_i u_i v_i^H$$

Here U and V are orthonormal matrices (all columns orthogonal to each other and each of unit length) and Σ is a diagonal matrix with the singular values in descending order, i.e., $\sigma_1 = \bar{\sigma}$, $\sigma_r = \underline{\sigma}$. The input-output interpretation is that an input in the direction v_i gives an output in the direction u_i[1] and the amplification is σ_i. The maximum amplification is then $\bar{\sigma}(A)$. Thus, if we consider the frequency response of $G(s)$ at a given frequency

$$\underline{\sigma}(G(i\omega)) \leq \frac{|Y(i\omega)|}{|U(i\omega)|} \leq \bar{\sigma}(G(i\omega))$$

Considering the maximum amplification over both frequency and direction we get

$$\sup_u \frac{\|y\|_2}{\|u\|_2} = \sup_\omega \bar{\sigma}(G(i\omega)) = \|G\|_\infty$$

Thus, just like in the SISO case, the gain of a linear MIMO system is given by the $\mathcal{H}_\infty$-norm of the transfer-matrix $G(s)$.

5.3 Decentralized Control and the RGA

The rest of this course will focus on analyzing and designing multivariable controllers for multivariable plants, based on the framework introduced for SISO systems in Lecture 1-4. However, before moving on to discuss the extension of these results to the MIMO case, we shall briefly discuss one specific type of control structure which is relatively common in practical applications, namely *decentralized control.* In decentralized control one chooses to control each output y_i with only one input u_j, i.e,

$$u_j = F_{yi}(s)(r_i - y_i)$$

where the controller $F_{yi}(s)$ is a SISO transfer-function. Rearranging the inputs and outputs this implies that the full controller can be written on diagonal form

$$F_y(s) = \texttt{diag}(F_{y1}(s) \ldots F_{y_p}(s))$$

[1] It may be somewhat confusing that we use U and u_j to denote output directions, since u normally is used for inputs in control, but we do this to follow the standard nomenclature for SVD.

Even though $F_y(s)$ is diagonal, the fact that the plant $G(s)$ in general is a full matrix implies that the closed-loop transfer-matrix

$$G_c(s) = G(s)F_y(s)(I + GF_y(s))^{-1}$$

also will be a full matrix. Thus, a setpoint change in any r_i will give a response in all outputs and all controllers F_{yi} will respond. This is usually termed *interactions* between the loops and can have a significant impact on both closed-loop performance and stability. Recall the examples discussed in Lecture 1. The strength of the interactions is in general dependent on how the inputs and outputs are paired, and it is advisable to choose a pairing that minimizes the interactions. The Relative Gain Array (RGA) is a model based tool that can be used to select such a pairing. We next derive the RGA and then discuss its use in selecing input-output pairings in decentralized control.

The idea behind the RGA is to quantify the impact of other control loops on the loop involving input u_j and output y_i. The Relative Gain for input u_j and output y_i is defined as the ratio

$$\lambda_{ij} = \frac{(y_i/u_j)_{\texttt{all loops open}}}{(y_i/u_j)_{\texttt{all other loops closed}}}$$

Here the numerator is simply the corresponding transfer-function element $G_{ij}(s)$. To determine the transfer-function (y_i/u_j) when all other loops are closed we must make some assumption on the controllers used. In the RGA, it is assumed that all other outputs are perfectly controlled, i.e., $y_{k,k\neq i} = 0$. Note that $u = G^{-1}y$ and hence the transfer-function we seek is

$$\left(\frac{y_i}{u_j}\right)_{y_{k,k\neq i}} = \frac{1}{(G^{-1})_{ji}}$$

Thus, the Relative Gain for the loop with output y_i and input u_j is

$$\lambda_{ij} = G_{ij}(G^{-1})_{ji} \tag{1}$$

and the Relative Gain Array (RGA) for all possible pairings become

$$\Lambda(G) = G(s) \times (G^{-1}(s))^T \tag{2}$$

where × denotes the Hadamard product, i.e., element-wise product. Since we prefer pairings with weak interactions with other loops we should choose pairings with a relative gain close to 1. In general, the RGA will be frequency dependent and then the most criticial is that interactions does not have a large impact around the crossover frequency ω_c of the loop considered. That is, we should prefer pairings with $\lambda_{ij}(i\omega_c) \approx 1$. Also note that if $\lambda_{ij}(0) < 0$ it implies that the steady-state gain of the loop changes sign as the other loops are closed, and negative feedback becomes positive feedback in the loop considered. Hence we should always avoid such pairings. This leads to the common rules for selecting pairings based on the RGA

1. Avoid pairings with $\lambda_{ij}(0) < 0$.
2. Prefer pairings with $\lambda_{ij}(i\omega_c)$ close to 1 (values between 0.5 and 3 are typically considered acceptable).

Note that if it is not possible to find any pairings that satisfy this rule, then it probably means it will be hard to obtain acceptable performance with decentralized control and one should instead consider true multivariable control, i.e., a full $F_y(s)$.

5.4 Decoupling

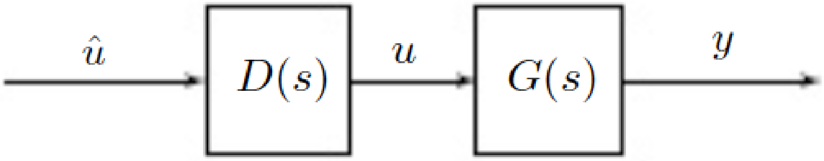

Figure 5.2: Pre-compensator for decoupling

When a plant has strong interactions, an apparantly simply way to resolve the problem is to remove the interactions by means of a compensator. Consider a plant $y = G(s)u$ and introduce the pre-compensator $u = D(s)\hat{u}(s)$. See Figure 5.2. Then

$$y = G(s)D(s)\hat{u}(s)$$

We can now choose the compensator $D(s)$ so that $G(s)D(s)$ has weak interactions. In particular, we can choose to have $G(s)D(s)$ diagonal, i.e., no interactions and $\Lambda(GD) = I$. A simple choice is then $D(s) = d(s)G^{-1}(s)$ where $d(s)$ are dynamics added to make $D(s)$ proper, i.e., have at least as many poles as zeros. The result is

$$G(s)D(s) = d(s)I$$

i.e., a diagonal system corresponding to a collection of r SISO systems.

Example 1: Consider the 2×2 system

$$G(s) = \frac{1}{s+1}\begin{pmatrix} 1 & -1 \\ 1.1 & -1 \end{pmatrix}$$

which has RGA

$$\Lambda = \begin{pmatrix} -10 & 11 \\ 11 & -10 \end{pmatrix}$$

at all frequencies. Thus, it is not recommended to use decentralized control. A decoupler is then

$$D(s) = \begin{pmatrix} 1 & -1 \\ 1.1 & -1 \end{pmatrix}^{-1} = \begin{pmatrix} -10 & 10 \\ -11 & 10 \end{pmatrix}$$

which yields

$$G(s)D(s) = \frac{1}{s+1}\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

The decoupling above may at first sight seem to more or less solve all problems related to MIMO control; simply use a decoupler and you then have a number of SISO problems. However, it is not as simple as that. First, decoupling may require large input moves which can be costly. Second, the decoupling may not be a good solution when considering disturbance attenuation. Third, one can not employ the inverse of a model which has RHP poles or zeros, or time delays, since then internal stability is lost. However, the most important reason why one should be careful about using decoupling is uncertainty.

If one does not have a perfect model of the plant, then the result of the decoupler may be far from the expected. Consider the example above again, but now add 10% uncertainty to each of the elements of the model G when computing the decoupler

Example 1, cont'd: Assume our model of G is

$$\hat{G} = \frac{1}{s+1}\begin{pmatrix} 1.1 & -0.9 \\ 1.2 & -0.9 \end{pmatrix}$$

Then the decoupler becomes

$$D(s) = \hat{G}^{-1} = \begin{pmatrix} -10 & 10 \\ -13.3 & 12.2 \end{pmatrix}$$

and the compensated plant

$$G(s)D(s) = \frac{1}{s+1}\begin{pmatrix} 3.3 & -2.2 \\ 2.3 & -1.2 \end{pmatrix}$$

which is far from diagonal.

In fact, it can be shown that plants with large RGA numbers are highly sensitive to uncertainty in their inverse. One result that reflects this is the following

Theorem: A matrix G becomes singular if we make a relative change of $-1/\lambda_{ij}$ in element G_{ij}, where λ_{ij} is element i, j of the RGA of G. That is, if we change element G_{ij} to $G_{p,ij} = G_{ij}(1 - \frac{1}{\lambda_{ij}})$ then the matrix becomes singular.

For the example above we have $\lambda_{11} = 10$ and hence changing G_{11} by a relative factor 0.1 to $G_{11} = 1.1$ makes the model singular (with infinite RGA-elements).

In summary, one should be careful with using decoupling due to sensitivity to model uncertainty. In particular, this applies to plant with large RGA numbers (which somewhat ironically are plants for which decoupling is needed the most). Thus, it is better to solve the overall control problem, including some description of the model uncertainty, using optimization based approaches as we will discuss later in this course.