Chapter 25

MIMO Controller Parameterizations

In this chapter, we will extend the, so called **Q** parameterization for SISO design of Chapter 15 to the MIMO case. We will find that many issues are common between the SISO and MIMO cases. However, there are distinctive issues in the MIMO case that warrant separate treatment. The key factor leading to these differences is once again the fact that MIMO systems have spatial coupling, i.e., each input can affect more than one output and each output can be affected by more than one input.

Notwithstanding these differences, the central issue in MIMO control-system design still turns out to be that of (*approximate*) inversion. Again, because of interactions, inversion is more intricate than in the SISO case, and we will thus need to develop more sophisticated tools for achieving this objective.

Our treatment in these slides will be relatively brief because the issues are probably best followed by reading the details in the book.

We thus present a summary that highlights the key end results.

The procedures and results mirror those presented earlier for the SISO case. The only difference here is that we need to deal with matrix, rather than scalar, transfer functions. This raises some issues of a technical nature not met in the SISO case.

Affine Parameterization: Stable MIMO Plants

The generalization of the parameterization of all stabilizing controllers to the multivariable case is straightforward. Indeed, all controllers that yield a stable closed-loop for a given open-loop stable plant having nominal transfer function $\mathbf{G_o}(s)$ can be expressed as

$$\mathbf{C}(s) = [\mathbf{I} - \mathbf{Q}(s)\mathbf{G_o}(s)]^{-1}\mathbf{Q}(s) = \mathbf{Q}(s)[\mathbf{I} - \mathbf{G_o}(s)\mathbf{Q}(s)]^{-1}$$

where $\mathbf{Q}(s)$ is any stable proper transfer-function *matrix*.

The Nominal Sensitivities in **Q** form

The resulting nominal sensitivity functions are

$$\mathbf{T_o}(s) = \mathbf{G_o}(s)\mathbf{Q}(s)$$

$$\mathbf{S_o}(s) = \mathbf{I} - \mathbf{G_o}(s)\mathbf{Q}(s)$$

$$\mathbf{S_{io}}(s) = (\mathbf{I} - \mathbf{G_o}(s)\mathbf{Q}(s))\mathbf{G_o}(s)$$

$$\mathbf{S_{uo}}(s) = \mathbf{Q}(s)$$

These transfer-function matrices are simultaneously stable if and only if **Q**(*s*) is stable. A key property is that they are affine in the matrix **Q**(*s*).

Achieved Sensitivities

The achieved sensitivity is

$$
\begin{aligned}
\mathbf{S}(s) &= \mathbf{S}_o(s)[\mathbf{I} + \mathbf{G}_{\Delta 1}(s)\mathbf{T_o}(s)]^{-1} \\
&= [\mathbf{I} - \mathbf{G_o}(s)\mathbf{Q}(s)][\mathbf{I} + \mathbf{G}_\epsilon(s)\mathbf{Q}(s)]^{-1}
\end{aligned}
$$

where $\mathbf{G}_\epsilon(s)$ is the additive model error, defined in

$$
\mathbf{G}(s) = \mathbf{G_o}(s) + \mathbf{G}_\epsilon(s)
$$

Again the reader will see the similarity with the SISO case.

We next see how one might use the nominal sensitivites expressed in the **Q** form for design.

Use of the **Q** form for design

An idealized target sensitivity function is $\mathbf{T}(s) = \mathbf{I}$. We then see from the equation $\mathbf{T_o}(s) = \mathbf{G_o}(s)\mathbf{Q}(s)$ that the design of $\mathbf{Q}(s)$ reduces to the problem of finding an (*approximate*) right inverse for $\mathbf{G_o}(s)$. Thus, as in the SISO case, we see that inversion is the core issue in control system design.

We remind the reader of the following issues which arose in the SISO problem of finding approximate inverses:

- *nonminimium-phase zeros*

- *model relative degree*

- *disturbance trade-offs*

- *control effort*

- *robustness*

- *uncontrollable modes*

These same issues appear in the MIMO case, but they are compounded by directionality issues.

We will focus on two of the design issues, namely:

- *Dealing with the issue of relative degree*
- *Dealing with non-minimum phase behavior.*

Dealing with Model Relative Degree

We recall from Chapter 15 that, in the SISO case, we dealt with model relative-degree issues by simply introducing extra filtering to render the appropriate transfer-function biproper prior to inversion. This same principle applies to the MIMO case, save that the filter needed to achieve a biproper matrix transfer function is a little more complicated than in the SISO case. In particular, we will need to use the idea of *interactor matrices*.

MIMO Relative Degree

Interactor matrices

We recall that the relative degree of a SISO model, amongst other things, sets a lower limit to the relative degree of the complementary sensitivity. In the SISO case, we say that the relative degree of a (*scalar*) transfer function $G(s)$ is the degree of a polynomial $p(s)$ such that

$$\lim_{s \to \infty} p(s)G(s) = K \qquad \text{where } 0 < |K| < \infty$$

This means that $p(s)G(s)$ is biproper, i.e., $(p(s)G(s))^{-1}$ is also proper.

In the MIMO case, every entry in the transfer-function matrix $\mathbf{G}(s)$ can have a different relative degree. Thus, to generate a multivariable version of the scalar polynomial $p(s)$ we will need to consider the individual entries and their interactions. To see how this can be done, consider an $m \times m$ matrix $\mathbf{G}(s)$.

We will show that there exist matrices $\boldsymbol{\xi_L}(s)$ and $\boldsymbol{\xi_R}(s)$ such that the following properties hold

$$\lim_{s \to \infty} \boldsymbol{\xi_L}(s)\mathbf{G}(s) = \mathbf{K_L} \qquad\qquad 0 < |\det(\mathbf{K_L})| < \infty$$

$$\lim_{s \to \infty} \mathbf{G}(s)\boldsymbol{\xi_R}(s) = \mathbf{K_R} \qquad\qquad 0 < |\det(\mathbf{K_R})| < \infty$$

Thus $\boldsymbol{\xi_L}(s)$ and $\boldsymbol{\xi_R}(s)$ are the multivariable equivalents of scalar relative degree. We call these matrices "Interactor Matrices". This result is established in the following theorem.

More formal statement of the MIMO relative degree result

Theorem 25.1: Consider a square transfer-function $m \times m$ matrix $\mathbf{G}(s)$, nonsingular almost everywhere in s. Then there exist unique transfer matrices $\boldsymbol{\xi_L}(s)$ and $\boldsymbol{\xi_R}(s)$ (*known as the left and right interactor matrices, respectively*) such that

$$\lim_{s \to \infty} \boldsymbol{\xi_L}(s)\mathbf{G}(s) = \mathbf{K_L} \qquad\qquad 0 < |\det(\mathbf{K_L})| < \infty$$

$$\lim_{s \to \infty} \mathbf{G}(s)\boldsymbol{\xi_R}(s) = \mathbf{K_R} \qquad\qquad 0 < |\det(\mathbf{K_R})| < \infty$$

are satisfied, such that

$$\boldsymbol{\xi_L}(s) = \mathbf{H_L}(s)\mathbf{D_L}(s)$$
$$\mathbf{D_L}(s) = \operatorname{diag}\left(s^{p_1}, \ldots, s^{p_m}\right)$$

$$
\mathbf{H_L}(s) = \begin{bmatrix} 1 & 0 & \cdots & \cdots & 0 \\ h_{21}^L(s) & 1 & \cdots & \cdots & 0 \\ h_{31}^L(s) & h_{32}^L(s) & \ddots & & \vdots \\ \vdots & \vdots & & \ddots & \vdots \\ h_{m1}^L(s) & h_{m2}^L(s) & \cdots & \cdots & 1 \end{bmatrix}
$$

$$
\boldsymbol{\xi_R}(s) = \mathbf{D_R}(s)\mathbf{H_R}(s)
$$

$$
\mathbf{D_R}(s) = \mathrm{diag}\,(s^{q_1}, \ldots, s^{q_m})
$$

$$
\mathbf{H_R}(s) = \begin{bmatrix} 1 & h_{12}^R(s) & h_{13}^R(s) & \cdots & h_{1m}^R(s) \\ 0 & 1 & h_{23}^R(s) & \cdots & h_{2m}^R(s) \\ \vdots & \vdots & \ddots & & \vdots \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & \cdots & 1 \end{bmatrix}
$$

And such that $h_{ij}^L(s)$ and $h_{ij}^R(s)$ are polynomials in s, satisfying $h_{ij}^L(0)=0$ and $h_{ij}^R(0)=0$.

Proof: See the book.

We illustrate by a simple example.

Example 25.2

Consider the transfer-function matrix $\mathbf{G}(s)$ given by

$$\mathbf{G}(s) = \begin{bmatrix} (s+1)^2 & (s+1) \\ 2(s+1) & 1 \end{bmatrix} [(s+1)^3 \mathbf{I}]^{-1}$$

Then

$$\boldsymbol{\xi}_{\mathbf{L}}(s) = \begin{bmatrix} s & 0 \\ -2s^2 & s^3 \end{bmatrix}$$

Remark: It is straightforward to see that the interactors can be defined by using diagonal matrices $\mathbf{D_L}(s)$ and $\mathbf{D_R}(s)$ with arbitrary polynomial diagonal entries with degrees, p_1, p_2, …, p_m, which are invariants of the interactor representation of a given matrix $\mathbf{G}(s)$. This flexibility is important, because we can always choose stable polynomials, implying that the inverses of $\xi_L(s)$ and $\xi_R(s)$ are also stable.

Approximate Inverses

We next show how interactors can be used to construct approximate inverses accounting for relative degree.

A crucial property of $\xi_{\mathbf{L}}(s)$ and $\xi_{\mathbf{R}}(s)$ is that

$$\mathbf{\Lambda_R}(s) \triangleq \mathbf{G_o}(s)\xi_{\mathbf{R}}(s); \qquad \text{and} \qquad \mathbf{\Lambda_L}(s) \triangleq \xi_{\mathbf{L}}(s)\mathbf{G_o}(s)$$

are both biproper transfer functions having nonsingular high-frequency gain. This simplifies the problem of inversion.

Note that $\mathbf{\Lambda_R}(s)$ and $\mathbf{\Lambda_L}(s)$ both have a state space representation of the form

$$\dot{x}(t) = \mathbf{A}x(t) + \overline{\mathbf{B}}u(t)$$
$$y(t) = \mathbf{C}x(t) + \overline{\mathbf{D}}u(t)$$

where $\overline{\mathbf{D}} \neq 0$.   Note also that $\mathbf{A}$ and $\mathbf{C}$ are the same as in the plant description.

Exact Inversion of $\Lambda(s)$

The key point is that the exact inverse of $\Lambda(s)$ can be obtained by simply reversing the roles of input and output, to yield the following state space realization of $[\Lambda(s)]^{-1}$:

$$\dot{x}(t) = \mathbf{A}x(t) + \overline{\mathbf{B}\mathbf{D}}^{-1}(y(t) - \mathbf{C}x(t))$$

$$= \mathbf{A}_{\lambda}x(t) + \mathbf{B}_{\lambda}\tilde{u}(t)$$

$$\bar{u}(t) = \overline{\mathbf{D}}^{-1}(y(t) - \mathbf{C}x(t))$$

$$= \mathbf{C}_{\lambda}x(t) + \mathbf{D}_{\lambda}\tilde{u}(t)$$

where $\tilde{u}(t)$ denotes the input to the inverse $\tilde{u}(t) = y(t)$, and $\bar{u}(t)$ denotes the output of the inverse.

Also

$$\mathbf{A}_\lambda = \mathbf{A} - \overline{\mathbf{B}}\,\overline{\mathbf{D}}^{-1}\mathbf{C} \qquad \mathbf{B}_\lambda = \overline{\mathbf{B}}\,\overline{\mathbf{D}}^{-1}$$

$$\mathbf{C}_\lambda = -\overline{\mathbf{D}}^{-1}\mathbf{C} \qquad \mathbf{D}_\lambda = \overline{\mathbf{D}}^{-1}$$

Use of the exact inverse for $\mathbf{\Lambda_L}(s)$ and $\mathbf{\Lambda_R}(s)$ to yield approximate inverses for $\mathbf{G_o}(s)$

We can use $[\mathbf{\Lambda_L}(s)]^{-1}$ or $[\mathbf{\Lambda_R}(s)]^{-1}$ to construct various approximations to the inverse of $\mathbf{G_o}(s)$. For example,

$$\mathbf{G_R^{inv}}(s) \triangleq [\boldsymbol{\xi_L}(s)\mathbf{G_o}(s)]^{-1}\boldsymbol{\xi_L}(0)$$

is an approximate *right inverse* with the property

$$\mathbf{G_o}(s)\mathbf{G_R^{inv}}(s) = [\boldsymbol{\xi_L}(s)]^{-1}\boldsymbol{\xi_L}(0)$$

which is lower triangular, and equal to the identity matrix at d.c.

Similarly

$$\mathbf{G}_{\mathbf{L}}^{\mathbf{inv}}(s) \triangleq \boldsymbol{\xi}_{\mathbf{R}}(0)[\mathbf{G_o}(s)\boldsymbol{\xi}_{\mathbf{R}}(s)]^{-1}$$

is an approximate *left inverse* with the property

$$\mathbf{G}_{\mathbf{L}}^{\mathbf{inv}}(s)\mathbf{G_o}(s) = \boldsymbol{\xi}_{\mathbf{R}}(0)[\boldsymbol{\xi}_{\mathbf{R}}(s)]^{-1}$$

which is also lower triangular, and equal to the identity matrix at d.c.

Use of approximate inverses in MIMO Control System Design

With the above tools in hand, we return to the original problem of constructing $\mathbf{Q}(s)$ as an (*approximate*) inverse for $\mathbf{G_o}(s)$. For example, we could choose $\mathbf{Q}(s)$ as

$$\mathbf{Q}(s) = [\boldsymbol{\Lambda}_{\mathbf{L}}(s)]^{-1}\boldsymbol{\xi}_{\mathbf{L}}(0) = [\boldsymbol{\xi}_{\mathbf{L}}(s)\mathbf{G_o}(s)]^{-1}\boldsymbol{\xi}_{\mathbf{L}}(0)$$

Note that for stable, minimum phase plants $\mathbf{G_o}(s)$, then $\mathbf{Q}(s)$ as defined above is proper and stable.

With this choice, we find that

$$\mathbf{T_o}(s) = \mathbf{G_o}(s)\mathbf{Q}(s)$$
$$= \mathbf{G_o}(s)[\mathbf{\Lambda_L}(s)]^{-1}\boldsymbol{\xi_L}(0)$$
$$= [\boldsymbol{\xi_L}(s)]^{-1}\boldsymbol{\xi_L}(0)$$

Thus, by choice of the relative-degree-modifying factors $(s + \alpha)$ we can make $\mathbf{T_o}(s)$ equal to $\mathbf{I}$ at d.c. and triangular at other frequencies, with bandwidth determined by the factors $(s + \alpha)$ used in forming $\boldsymbol{\xi_L}(s)$.

This gives a simple solution to the MIMO design problem.

What about Non-minimum Phase Systems?

Of course, the above procedure for calculating a suitable value for $\mathbf{Q}(s)$ in MIMO design will only yield a stable $\mathbf{Q}(s)$ if $\mathbf{G_o}(s)$ is minimum phase.

We recall that closed loop stability requires that $\mathbf{Q}(s)$ be stable. Hence, in the case of non-minimum phase systems, we need to find some way of modifying the design so as to render $\mathbf{Q}(s)$ stable. We describe several mechanisms for dealing with non-minimum phase (NMP) zeros below.

Dealing with NMP Zeros

Z-Interactors

We saw earlier that interactor matrices are a convenient way of describing the relative degree of a plant (*or zeros at ∞ of a plant*). Also, we saw that the interactor matrix can be used to precompensate the plant so as to isolate the zeros at ∞, thus allowing a *proper* inverse to be computed for the remainder. The same basic idea can be used to describe the structure of finite zeros. The appropriate transformations are known as *z*-interactors.

They allow a precompensator to be computed that isolates particular finite zeros. In particular, when applied to isolating the nonminimum-phase zeros and combined with interactors for the zeros at ∞, z-interactors allow a *stable* and *proper* inverse to be computed.

We will not spell out the details. However, full details are provided in the book.

Instead, we will describe an alternative procedure for design $\mathbf{Q}(s)$ based on the use of model matching ideas.

Q Synthesis as a Model-Matching Problem

Although the use of interactors and z-interactors gives insight into the principal possibilities and fundamental structure of Q-synthesis for MIMO design, the procedure is usually not appropriate for numerically carrying out the synthesis. In addition to being difficult to automate as a numerical algorithm, the procedure would require the analytical removal of unstable pole-zero cancellations, which is awkward.

We therefore investigate an alternative method for computing a stable approximate inverse by using the model-matching procedures. This circumvents the need for using z-interactors.

We first turn the plant into biproper form by using the usual left interactor, $\xi_L(s)$.

Next, let us assume that the target complementary sensitivity is $\mathbf{T}^*(s)$. We know that, under the MIMO affine parameterization for the controller, the nominal sensitivity is $\mathbf{G}_o(s)\mathbf{Q}(s)$. Hence we can convert the $\mathbf{Q}$-synthesis problem into a model matching problem by seeking to find $\mathbf{Q}(s)$ by minimizing a model matching cost of the form:

$$J = \frac{1}{2\pi} \int_{-\infty}^{\infty} \left\| \mathbf{M}(j\omega) - \mathbf{N}(j\omega)\mathbf{\Gamma}(j\omega) \right\|_F^2 \, d\omega$$

where $\mathbf{M}$, $\mathbf{N}$, and $\mathbf{\Gamma}$ correspond to $\mathbf{T}^*$, $\xi_L \mathbf{G}_o$, and $\mathbf{Q}$, respectively. Also, $\|\cdot\|F$ denotes the Frobeneous norm.

We begin by examining **Q** one column at a time. For the ith column, we have

$$J_i = \frac{1}{2\pi} \int_{-\infty}^{\infty} \left\| [\mathbf{M}(j\omega)]_{*i} - \mathbf{N}(j\omega)[\mathbf{\Gamma}(j\omega)]_{*i} \right\|_F^2 \, d\omega$$

Conversion to Time Domain

As in Chapter 22, we convert to the time domain and use

$$\dot{x}_1(t) = \mathbf{A_1} x_1(t); \qquad x_1(0) = \mathbf{B_1}$$
$$y_1(t) = \mathbf{C_1} x_1(t)$$

to represent the system with transfer function $[\mathbf{M}(s)]_{*i.}$

We also use

$$\dot{\tilde{x}}_2(t) = \mathbf{A_2}\tilde{x}_2(t) + \tilde{\mathbf{B}}_2 u(t)$$
$$z_2(t) = \mathbf{C_2}\tilde{x}_2(t) + \tilde{\mathbf{D}}u(t)$$

to represent the system with biproper transfer function $\xi_{\mathbf{L}}(s)\mathbf{G_o}(s)$. Also, for square plants, we know from the properties of $\xi_{\mathbf{L}}(s)$ that $\det\{\mathbf{D}\} \neq 0$.

This seems to fit the theory given earlier for Model Matching. However, an important difference is that here we have no weighting on the control effort $u(t)$. This was not explicitly allowed in the earlier work. We thus need to extend the earlier results to cover the case where no control weighting is used. This requires a simple transformation.

Full details are given in the book.

The model matching problem can then be solved, as before, using LQR theory via Riccati equations. We illustrate by a simple example.

Example 25.6

Consider 2×2 MIMO plant having the nominal model

$$\mathbf{G_o}(s) = \begin{bmatrix} \dfrac{-1}{s+2} & \dfrac{2}{s+1} \\ \dfrac{2}{s+2} & \dfrac{7(-s+1)}{(s+1)(s+2)} \end{bmatrix}$$

This is a stable but nonminimum-phase system, with poles at -1, -2, and -2 and a zero at $s = 5$.

The target sensitivity function is chosen as

$$\mathbf{T}^*(s) = \frac{9}{s^2 + 4s + 9} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

To cast this into the problem formulation outlined above, we next reparameterize $\mathbf{Q}(s)$ to force integration in the feedback loop. We thus use

$$\mathbf{Q}(s) = \mathbf{G_o}(0)^{-1} + s\overline{\mathbf{Q}}(s) = \frac{1}{15}\begin{bmatrix} -14 & 8 \\ 4 & 2 \end{bmatrix} + s\overline{\mathbf{Q}}(s)$$

and introduce a weighting function $\mathbf{W}_s(s) = \mathbf{I}/s$.

(This mirrors a similar idea used in the SISO case).

Then, in terms of the model matching problem,we have that

$$\mathbf{M}(s) = \mathbf{W_s}(s)\left(\mathbf{T}^*(s) - \mathbf{G_o}(s)\mathbf{G_o}(0)^{-1}\right); \qquad \text{and} \quad \mathbf{N}(s) = \mathbf{G_o}(s)$$

Thus,

$$\mathbf{M}(s) = \begin{bmatrix} \dfrac{-1.46s^2 + 1.13s + 5.8}{(s^2 + 4s + 9)(s+1)(s+2)} & \dfrac{0.267}{(s+1)(s+2)} \\ \dfrac{3.73}{(s+1)(s+2)} & \dfrac{-0.13s^2 + 6.466s + 17.8}{(s^2 + 4s + 9)(s+1)(s+2)} \end{bmatrix}$$

To solve the problem by following the approach presented above, we need to first build the left interactor, $\boldsymbol{\xi_L}(s)$, for $\mathbf{N}(s)$. This interactor is given by $\boldsymbol{\xi_L}(s) = s\mathbf{I}$, leading to

$$\boldsymbol{\xi}_L(\alpha)^{-1}\boldsymbol{\xi}_L(s+\alpha)\mathbf{N}(s) = \frac{\tau s + 1}{(s+1)(s+2)} \begin{bmatrix} -(s+1) & 2(s+2) \\ 2(s+1) & 7(-s+1) \end{bmatrix}$$

where $\tau = \alpha^{-1} = 0.1$.

Then solving the model matching problem, we obtain

$$\overline{\mathbf{Q}}(s) = \frac{\begin{bmatrix} \overline{Q}_{11}(s) & \overline{Q}_{12}(s) \\ \overline{Q}_{21}(s) & \overline{Q}_{22}(s) \end{bmatrix}}{s^4 + 19s^3 + 119s^2 + 335s + 450}$$

$$\overline{Q}_{11}(s) = 7.11s^3 + 83.11s^2 + 337.11s + 328.67$$
$$\overline{Q}_{12}(s) = 3.55s^3 + 31.56s^2 + 123.56s + 189.33$$
$$\overline{Q}_{21}(s) = 2.22s^3 + 24.00s^2 + 100.44s + 142.67$$
$$\overline{Q}_{22}(s) = 1.11s^3 + 12.0s^2 + 50.22s + 71.33$$

Finally, we recover **Q**(s) as

$$Q(s) = \frac{\begin{bmatrix} Q_{11}(s) & Q_{12}(s) \\ Q_{21}(s) & Q_{22}(s) \end{bmatrix}}{s^4 + 19s^3 + 119s^2 + 335s + 450}$$

$$Q_{11}(s) = 6.18s^4 + 65.38s^3 + 226.04s^2 + 16.00s - 420.00$$

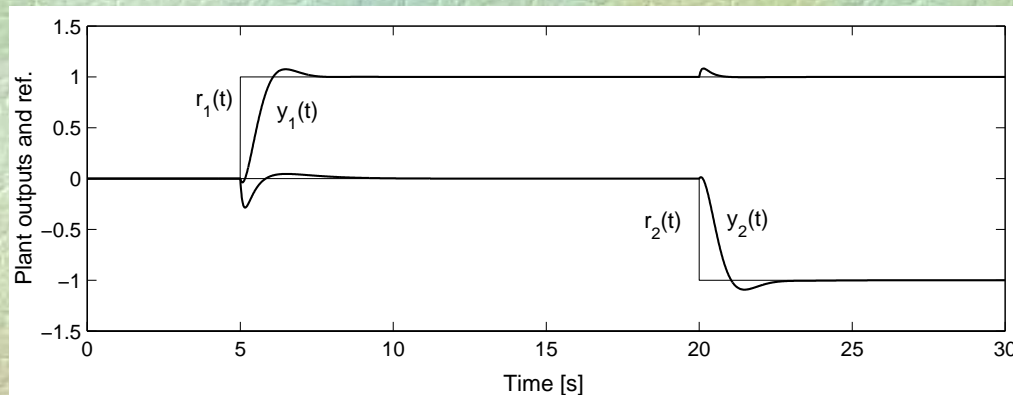$$Q_{12}(s) = 4.09s^4 + 41.69s^3 + 187.02s^2 + 368.00s + 240.00$$

$$Q_{21}(s) = 2.49s^4 + 29.07s^3 + 132.18s^2 + 232.00s + 120.00$$

$$Q_{22}(s) = 1.24s^4 + 14.53s^3 + 66.09s^2 + 116.00s + 60$$

The design was then simulated with unit step references. The results are shown on the next slide.

Figure 25.1: *Optimal quadratic design-step reference tracking*

Note that there is evidence of the non-minimum phase zero in the undershoot seen in the above responses. Also, note that the closed loop system exhibits some small coupling in the closed loop. We will see in the next chapter how a completely decoupled closed loop response can be obtained using an alternative design for Q(s).

All of the above methodologies for designing $\mathbf{Q}$(s) assumed that the plant was open loop stable.

We next consider the case when the plant is open loop unstable.

Affine Parameterization: Unstable MIMO Plants

We consider a LMFD and RMFD for the plant of the form

$$\mathbf{G_o}(s) = [\overline{\mathbf{G}}_{\mathbf{oD}}(s)]^{-1}\overline{\mathbf{G}}_{\mathbf{oN}}(s) = \mathbf{G_{oN}}(s)[\mathbf{G_{oD}}(s)]^{-1}$$

Observe also that, if $\mathbf{G_o}(s)$ is unstable, then $\overline{\mathbf{G}}_{\mathbf{oD}}(s)$ and $\mathbf{G_{oD}}(s)$ will both be nonminimum phase. Similarly, if $\mathbf{G_o}(s)$ is nonminimum phase, then so will be $\overline{\mathbf{G}}_{\mathbf{oN}}(s)$ and $\mathbf{G_{oN}}(s)$.

Affine Control Law Parameterization for Unstable MIMO Plants

Lemma 25.2 (*Affine parameterization for unstable MIMO plants*):

Consider a plant described in MFD as above where $\overline{\mathbf{G}}_{\mathbf{oN}}(s), \overline{\mathbf{G}}_{\mathbf{oD}}(s), \mathbf{G}_{\mathbf{oN}}(s),$ and $\mathbf{G}_{\mathbf{oD}}(s)$ are a coprime factorization, satisfying

$$\begin{bmatrix} \overline{\mathbf{C}}_{\mathbf{D}}(s) & \overline{\mathbf{C}}_{\mathbf{N}}(s) \\ -\overline{\mathbf{G}}_{\mathbf{N}}(s) & \overline{\mathbf{G}}_{\mathbf{D}}(s) \end{bmatrix} \begin{bmatrix} \mathbf{G}_{\mathbf{D}}(s) & -\mathbf{C}_{\mathbf{N}}(s) \\ \mathbf{G}_{\mathbf{N}}(s) & \mathbf{C}_{\mathbf{D}}(s) \end{bmatrix} = \mathbf{I}$$

Then the class of all stabilizing controllers for the nominal plant can be expressed as

$$\mathbf{C}(s) = \mathbf{C}_{\mathbf{N}\Omega}(s)[\mathbf{C}_{\mathbf{D}\Omega}(s)]^{-1} = [\overline{\mathbf{C}}_{\mathbf{D}\Omega}(s)]^{-1}[\overline{\mathbf{C}}_{\mathbf{N}\Omega}(s)]$$

where

$$\overline{\mathbf{C}}_{\mathbf{D}\Omega}(s) = \overline{\mathbf{C}}_{\mathbf{D}}(s) - \Omega(s)\overline{\mathbf{G}}_{\mathbf{oN}}(s)$$
$$\overline{\mathbf{C}}_{\mathbf{N}\Omega}(s) = \overline{\mathbf{C}}_{\mathbf{N}}(s) + \Omega(s)\overline{\mathbf{G}}_{\mathbf{oD}}(s)$$
$$\mathbf{C}_{\mathbf{D}\Omega}(s) = \mathbf{C}_{\mathbf{D}}(s) - \mathbf{G}_{\mathbf{oN}}(s)\Omega(s)$$
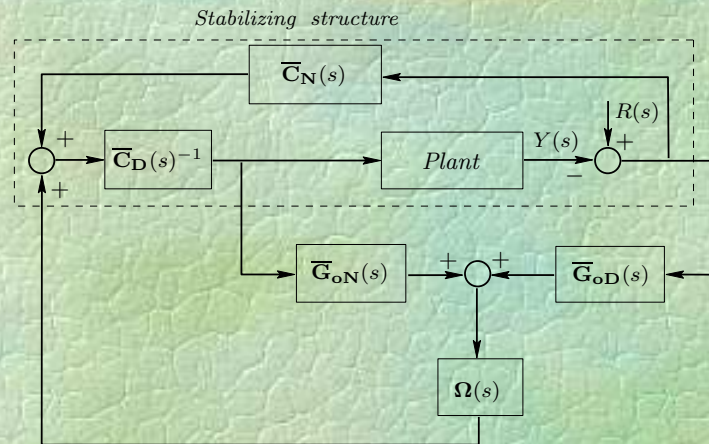$$\mathbf{C}_{\mathbf{N}\Omega}(s) = \mathbf{C}_{\mathbf{N}}(s) + \mathbf{G}_{\mathbf{oD}}(s)\Omega(s)$$

where $\Omega(s)$ is any *stable $m \times m$* proper transfer matrix.

The controller described above is depicted on the following slide.

Figure 25.2: $\mathbf{Q}$ *Parameterisation for MIMO unstable plants*

Stabilizing structure

Note that the above arrangement is reminiscent of the result described earlier for the SISO case. Of course, in the MIMO case, all transfer functions are matrices of appropriate dimensions.
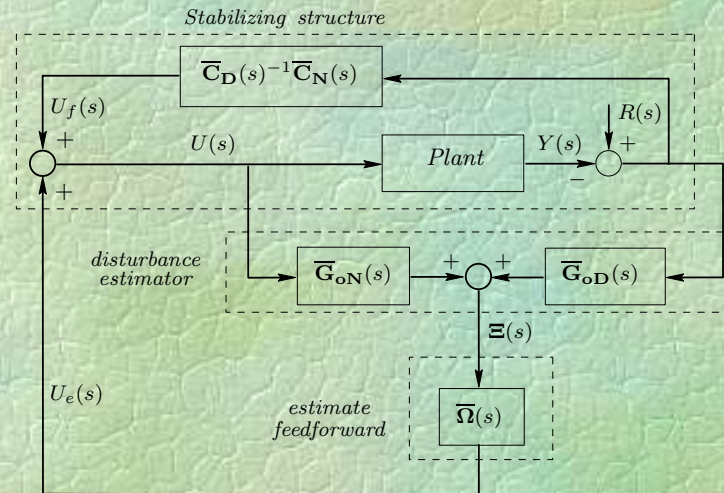
If we make the special choice

$$\boldsymbol{\Omega}(s) = \overline{\mathbf{C}_{\mathbf{D}}}(s)\overline{\boldsymbol{\Omega}}(s)$$

we can represent the system as on the next slide.

Figure 25.3: *Alternative* **Q** *parameterization for MIMO unstable plants with restricted* $\Omega(s)$

Again this is reminiscent of the structure developed for the SISO case.

In this case, we have

$$\mathbf{S_o}(s) = (\mathbf{C_D}(s) - \mathbf{G_{oN}}(s)\overline{\mathbf{C}}_{\mathbf{D}}(s)\overline{\mathbf{\Omega}}(s))\overline{\mathbf{G}}_{\mathbf{oD}}(s)$$
$$= (\mathbf{I} - [\mathbf{C_D}(s)]^{-1}\mathbf{G_{oN}}(s)\overline{\mathbf{C}}_{\mathbf{D}}(s)\overline{\mathbf{\Omega}}(s))\mathbf{C_D}(s)\overline{\mathbf{G}}_{\mathbf{oD}}(s)$$
$$= (\mathbf{I} - [\mathbf{C_D}(s)]^{-1}(\mathbf{C_D}(s)\overline{\mathbf{G}}_{\mathbf{oN}}(s))\overline{\mathbf{\Omega}}(s))\mathbf{C_D}(s)\overline{\mathbf{G}}_{\mathbf{oD}}(s)$$
$$= (\mathbf{I} - \overline{\mathbf{G}}_{\mathbf{oN}}(s)\overline{\mathbf{\Omega}}(s))\mathbf{C_D}(s)\overline{\mathbf{G}}_{\mathbf{oD}}(s)$$
$$= \mathbf{S}_{\overline{\mathbf{\Omega}}}(s)\mathbf{S_C}(s)$$

where $\mathbf{S_C}(s)$ is the sensitivity achieved with the prestabilizing controller and $\mathbf{S}_{\overline{\mathbf{\Omega}}}(s)$ in the sensitivity function

$$\mathbf{S}_{\overline{\mathbf{\Omega}}}(s) = \mathbf{I}(s) - \overline{\mathbf{G}}_{\mathbf{oN}}(s)\overline{\mathbf{\Omega}}(s)$$

We recognize $\mathbf{S}_{\overline{\Omega}}(s)$ as having the form $\mathbf{S_o}(s) = \mathbf{I} - \mathbf{G_o}(s)\mathbf{Q}(s)$ for the stable equivalent plant $\overline{\mathbf{G}}_{\mathbf{oN}}(s)$. This suggests that, the techniques developed earlier in this chapter for $\mathbf{Q}$ design in the stable open-loop case can be simply applied here to design $\overline{\Omega}(s)$ so as to render $\mathbf{S_o}(s)$ *small* in some suitable sense.

Note, however, that it is desirable to ensure that $\mathbf{S_C}(s)$ is also sensible, or else this will negatively interact with the choice of $\mathbf{S}_{\overline{\Omega}}(s)$. For example, if $\mathbf{S_C}(s)$ is not diagonal, then making $\mathbf{S}_{\overline{\Omega}}(s)$ diagonal does not give dynamic decoupling. We take this topic up in the next chapter.

State Space Implementation

We recall that in Chapter 15 we showed that there exists a nice state space interpretation of the class of all stabilizing controllers for the open-loop unstable SISO case. A similar interpretation applies to the MIMO case. This interpretation is particularly useful in the MIMO case, where a state space format greatly facilitates design and implementation.

Details are given in the book.

Summary

❖ The generalization of the affine parameterization for a stable multivariable model $\mathbf{G_o}(s)$ is given by the controller representation

$$\mathbf{C}(s) = [\mathbf{I} - \mathbf{Q}(s)\mathbf{G_o}(s)]^{-1}\mathbf{Q}(s) = \mathbf{Q}(s)[\mathbf{I} - \mathbf{G_o}(s)\mathbf{Q}(s)]^{-1}$$

yielding the nominal sensitivities

$$\mathbf{T_o}(s) = \mathbf{G_o}(s)\mathbf{Q}(s)$$
$$\mathbf{S_o}(s) = \mathbf{I} - \mathbf{G_o}(s)\mathbf{Q}(s)$$
$$\mathbf{S_{io}}(s) = [\mathbf{I} - \mathbf{G_o}(s)\mathbf{Q}(s)]\mathbf{G_o}(s)$$
$$\mathbf{S_{uo}}(s) = \mathbf{Q}(s)$$

❖ The associated achieved sensitivity, when the controller is applied to $\mathbf{G}(s)$, is given by

$$\mathbf{S}(s) = \mathbf{S_o}(s)[\mathbf{I} + \mathbf{G}_\epsilon(s)\mathbf{Q}(s)]^{-1}$$

where $\mathbf{G}_\epsilon(s) = \mathbf{G}(s) - \mathbf{G_o}(s)$ is the additive modeling error.

❖ In analogy to the SISO case, key advantages of the affine parameterization include the following

- explicit stability of the nominal closed loop if and only if $\mathbf{Q}(s)$ is stable;

- highlighting the fundamental importance of invertibility, i.e., the achievable and achieved properties of $\mathbf{G_o}(s)\mathbf{Q}(s)$ and $\mathbf{G}(s)\mathbf{Q}(s)$; and

- sensitivities that are affine in $\mathbf{Q}(s)$ - this facilitates criterion-based synthesis, which is particularly attractive for MIMO systems.

- ❖ Again in analogy to the SISO case, inversion of stable MIMO systems involves two key issues:
  - ◆ relative degree - i.e., the structure of zeros at infinity; and
  - ◆ inverse stability - i.e. , the structure of NMP zeros.

- ❖ Because of directionality, both of these attributes exhibit additional complexity in the MIMO case.

- ❖ The structure of zeros at infinite is captured by the left or right interactor ($\xi_L(s)$ *or* $\xi_R(s)$, *respectively*).

- ❖ Thus $\xi_L(s)G_o(s)$ is biproper, i.e., its determinant is a nonzero bounded quantity for s $\rightarrow \infty$.

❖ The structure of NMP zeros is captured by the left or right z-interactor ($\varphi_{\mathbf{L}}(s)$ *or* $\varphi_{\mathbf{R}}(s)$, *respectively*).

❖ Thus, *analytically*, $\varphi_{\mathbf{L}}(s)\mathbf{G_o}(s)$ is a realization of the inversely stable portion of the model - i.e., the equivalent to the minimum-phase factors in the SISO case.

❖ However, the realization $\varphi_{\mathbf{L}}(s)\,\mathbf{G_o}(s)$

  ◆ if nonminimal, and

  ◆ generally involves cancellations of unstable pole-zero dynamics (the NMP zero dynamics of $\mathbf{G_o}(s)$).

❖ Thus, the realization $\varphi_{\mathbf{L}}(s)$Go(s)

  ◆ is useful for analyzing the fundamentally achievable properties of the key quantity $\mathbf{G_o}(s)\mathbf{Q}(s)$, subject to the stability of $\mathbf{Q}(s)$, and

  ◆ is generally not suitable for either implementation or inverse implementation, because it involves unstable pole-zero cancellation.

❖ A stable inverse suitable for implementation is generated by model matching, which leads to a particular linear quadratic regulator (LQR) problem which is solvable via Riccati equations.

❖ If the plant model is unstable, controller design can be carried out in two steps:

(i) prestabilization, for example via LQR; then

(ii) detailed design, by applying the theory for stable models to the prestabilized system.

❖ All of the above results can be interpreted equivalently in either a transfer-function or a state space framework; for MIMO systems, the state space framework is particularly attractive for numerical implementation.