Chapter 21

Exploiting SISO Techniques in MIMO Control

In the case of SISO control, we found that one could use a wide variety of synthesis methods. Some of these carry over directly to the MIMO case. However, there are several complexities that arise in MIMO situations. For this reason, it is often desirable to use synthesis procedures that are in some sense automated. This will be the subject of the next few chapters. However, before we delve into the full complexity of MIMO design, it is appropriate that we pause to see when, if ever, SISO techniques can be applied to MIMO problems directly.

We will study

- decentralized control as a mechanism for directly exploiting SISO methods in a MIMO setting

- robustness issues associated with decentralized control.

Completely Decentralized Control

Before we consider a fully interacting multivariable design, it is often useful to check on whether a completely decentralized design can achieve the desired performance objectives. When applicable, the advantage of a completely decentralized controller, compared to a full MIMO controller, is that it is simpler to understand, is easier to maintain, and can be enhanced in a straightforward fashion (*in the case of a plant upgrade*).

Readers having previous exposure to practical control will realize that a substantial proportion of real-world systems will utilize decentralized architectures. Thus, one is led to ask the question, *is there ever a situation in which decentralized control will not yield a satisfactory solution*? We will present several real-world examples later in Chapter 22 that require MIMO thinking to get a satisfactory solution. As a *textbook example* of where decentralized control can break down, consider the following MIMO example.

Example 21.1

Consider a two-input, two-output plant having the transfer function

$$\mathbf{G_o}(s) = \begin{bmatrix} G^o_{11}(s) & G^o_{12}(s) \\ G^o_{21}(s) & G^o_{22}(s) \end{bmatrix}$$

$$G^o_{11}(s) = \frac{2}{s^2 + 3s + 2} \qquad G^o_{12}(s) = \frac{k_{12}}{s + 1}$$

$$G^o_{21}(s) = \frac{k_{21}}{s^2 + 2s + 1} \qquad G^o_{22}(s) = \frac{6}{s^2 + 5s + 6}$$

Let us say that k_{12} and k_{21} depend on the operating point (*a common situation*, *in practice*).

Operating point 1 ($k_{12} = k_{21} = 0$)

Clearly, there is no interaction at this operating point.

Thus, we can safely design two SISO controllers. To be specific, say we aim for the following complementary sensitivities:

$$T_{o1}(s) = T_{o2}(s) = \frac{9}{s^2 + 4s + 9}$$

The corresponding controller transfer functions are $C_1(s)$ and $C_2(s)$, where

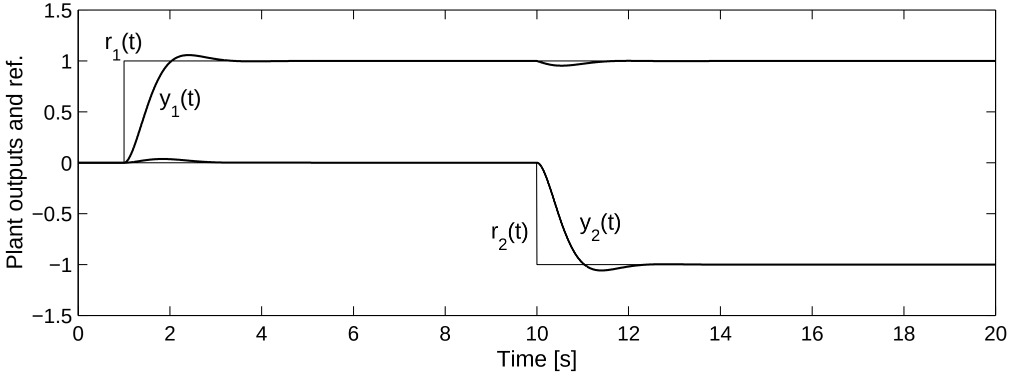
$$C_1(s) = \frac{4.5(s^2 + 3s + 2)}{s(s+4)}; \qquad C_2(s) = \frac{1.5(s^2 + 5s + 6)}{s(s+4)}$$

The two independent loops perform as predicted by the choice of complementary sensitivities.

Operating point 2 ($k_{12} = k_{21} = 0.1$)

We leave the controller as previously designed for operating point 1. We apply a unit step in the reference for output 1 at $t = 1$ and a unit step in the reference for output 2 at $t = 10$. The closed-loop response is shown on the next slide. These results would probably be considered very acceptable, even though the effects of coupling are now evident in the response.

Figure 21.1: *Effects of weak interaction in control loop with SISO design*

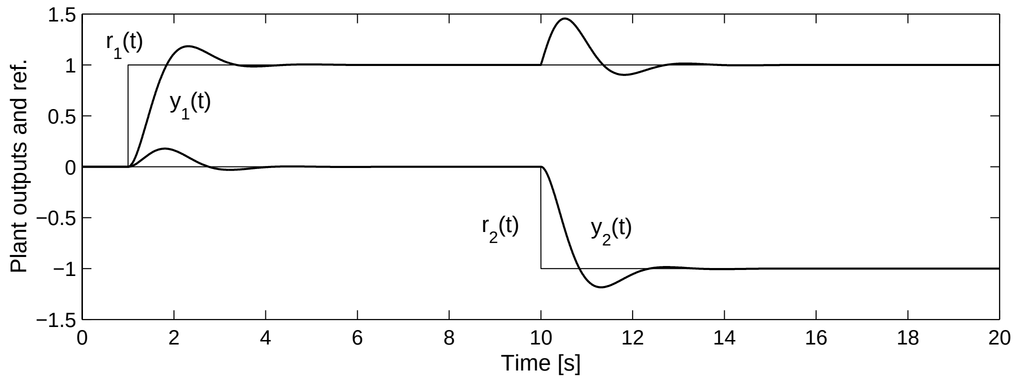


Operating point 3 (k_{12} = -1, k_{21} = 0.5)

With the same controllers and for the same test as used at operating point 2, we obtain the results on the next slide.

We see that a change in the reference in one loop now affects the output in the other loop significantly.

Figure 21.2: *Effects of strong interaction in control loops with SISO design*



Operating point 4 (k_{12} = -2, k_{21} = -1)

Now a simulation with the same reference signals indicates that the whole system becomes unstable. We see that the original SISO design has become unacceptable at this final operating point.

Pairing of Inputs and Outputs

If one is to use a decentralized architecture, then one needs to pair the inputs and outputs. In the case of an $m \times m$ plant transfer function, there are m! possible pairings. However, physical insight can often be used to suggest sensible pairings.

Relative Gain Array

One method that can be used to suggest pairings is a quantity known as the **Relative Gain Array** (RGA). For a system with matrix transfer function $\mathbf{G_o}(s)$, the RGA is defined as a matrix Λ with the ij^{th} element

$$\lambda_{ij} = [\mathbf{G_o}(0)]_{ij}[\mathbf{G_o}^{-1}(0)]_{ji}$$

where $[\mathbf{G_o}(0)]_{ij}$ and $[\mathbf{G_o}^{-1}(0)]_{ij}$ denote the ij^{th} element of the plant d.c. gain and the ji^{th} element of the inverse of the d.c. gain matrix respectively.

Note that $[\mathbf{G_o}(0)]_{ij}$ corresponds to the d.c. gain from the i^{th} input, u_i, to the j^{th} output, y_j, while the rest of the inputs, u_l for $l \in \{1, 2, \ldots, i\text{-}1, i\text{+}1, \ldots, m\}$ are kept constant. Also $[\mathbf{G_o}^{-1}]_{ij}$ is the reciprocal of the d.c. gain from the i^{th} input, u_i, to the j^{th} output, y_j, while the rest of the outputs, y_l for $l \in \{1, 2, \ldots, j\text{-}1, j\text{+}1, \ldots, m\}$ are kept constant. Thus, the parameter λ_{ij} provides an indication of how sensible it is to pair the i^{th} input with the j^{th} output.

One usually aims to pick pairings such that the diagonal entries of Λ are large. One also tries to avoid pairings that result in negative diagonal entries in Λ.

Example

Consider again the system

$$\mathbf{G_o}(s) = \begin{bmatrix} \dfrac{2}{s^2+3s+2} & \dfrac{k_{12}}{s+1} \\ \dfrac{k_{21}}{s^2+2s+1} & \dfrac{6}{s^2+5s+6} \end{bmatrix}$$

The RGA is then

$$\mathbf{\Lambda} = \begin{bmatrix} \dfrac{1}{1-k_{12}k_{21}} & \dfrac{-k_{12}k_{21}}{1-k_{12}k_{21}} \\ \dfrac{-k_{12}k_{21}}{1-k_{12}k_{21}} & \dfrac{1}{1-k_{12}k_{21}} \end{bmatrix}$$

For $1 > k_{12} > 0$, $1 > k_{21} > 0$, the RGA suggests the pairing (u_1, y_1), (u_2, y_2). We recall from our earlier study of this example that this pairing worked very well for $k_{12} = k_{21} = 0.1$ and quite acceptably for $k_{12} = -1$, $k_{21} = 0.5$. In the latter case, the RGA is

$$\mathbf{\Lambda} = \frac{1}{3}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

However, for k_{12} = -2, k_{21} = -1 we found that the centralized controller based on the pairing (u_1, y_1), (u_2, y_2) was actually unstable. The corresponding RGA in this case is

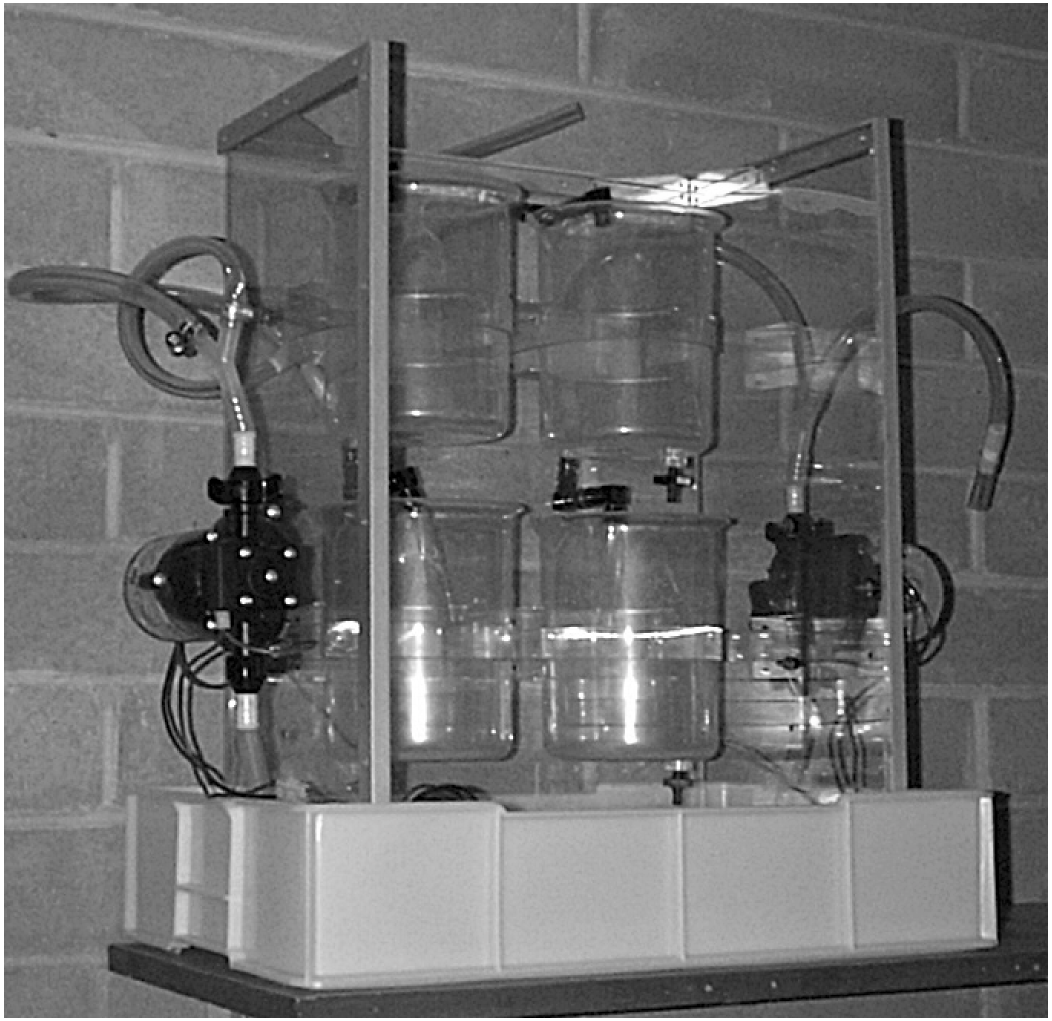
$$\mathbf{\Lambda} = \begin{bmatrix} -1 & 2 \\ 2 & -1 \end{bmatrix}$$

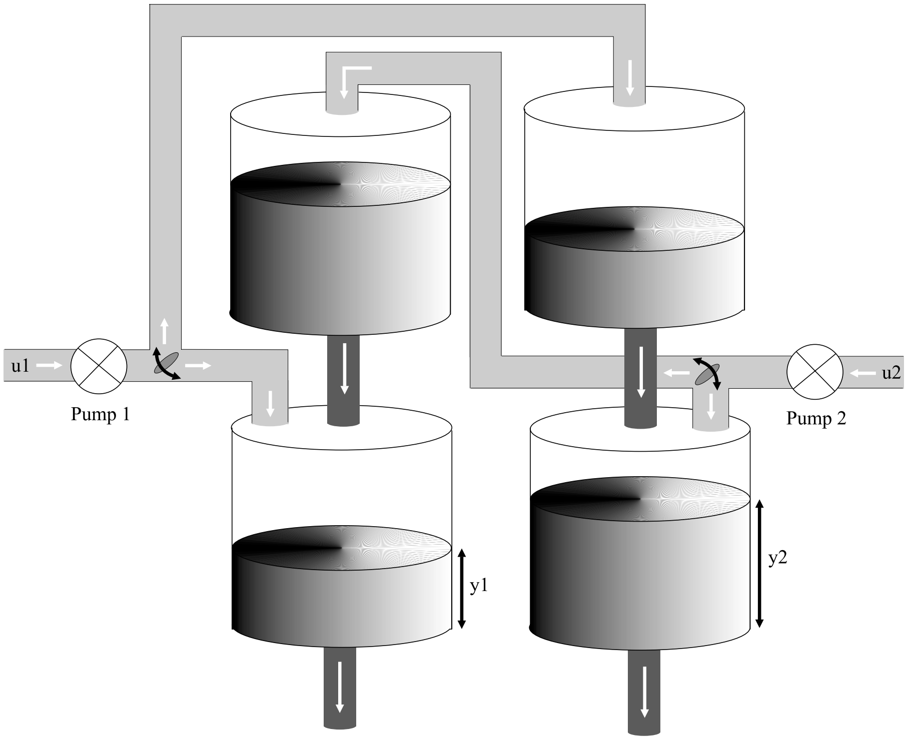
which indicates that we probably should have changed to the pairing (u_1, y_2), (u_2, y_1).

Example 21.3

Quadruple-tank apparatus.

Consider the quadruple-tank apparatus shown on the next two slides.




u1
Pump 1
u2
Pump 2
y1
y2

We recall from Chapter 20 that this system has an approximate transfer function,

$$\mathbf{G}(s) = \begin{bmatrix} \dfrac{3.7\gamma_1}{62s+1} & \dfrac{3.7(1-\gamma_2)}{(23s+1)(62s+1)} \\ \dfrac{4.7(1-\gamma_1)}{(30s+1)(90s+1)} & \dfrac{4.7\gamma_2}{90s+1} \end{bmatrix}$$

The RGA for this system is

$$\mathbf{\Lambda} = \begin{bmatrix} \lambda & 1-\lambda \\ 1-\lambda & \lambda \end{bmatrix} \qquad \text{where} \quad \lambda = \frac{\gamma_1\gamma_2}{\gamma_1\gamma_2 - 1}$$

For $1 < \gamma_1 + \gamma_2 < 2$, we recall from Chapter 20 that the system is of minimum phase. If we take, for example, $\gamma_1 = 0.7$ and $\gamma_2 = 0.6$, then the RGA is

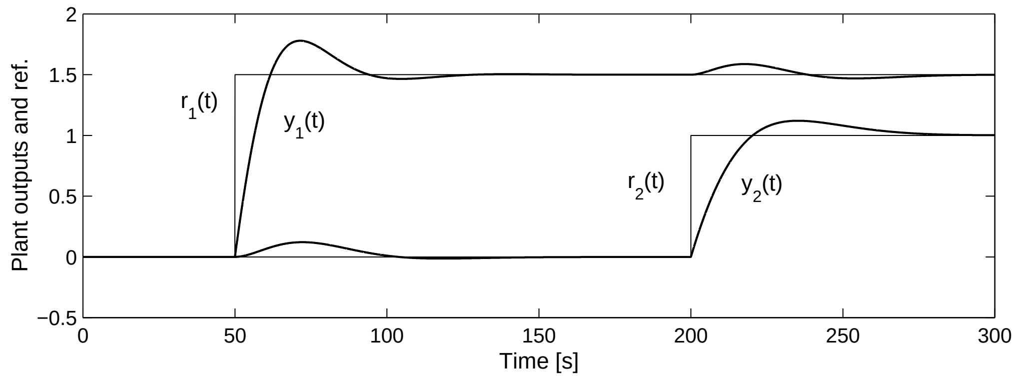
$$\mathbf{\Lambda} = \begin{bmatrix} 1.4 & -0.4 \\ -0.4 & 1.4 \end{bmatrix}$$

This suggests that we can pair (u_1, y_1) and (u_2, y_2).

Because the system is of minimum phase, the design of a decentralized controller is relatively easy in this case. For example, the following decentralized controller gives the results shown on the next slide

$$C_1(s) = 3\left(1 + \frac{1}{10s}\right); \qquad C_2(s) = 2.7\left(1 + \frac{1}{20s}\right)$$

Figure 21.3: *Decentralized control of a minimum-phase four-tank system*



For $0 < \gamma_1 + \gamma_2 < 1$, we recall from Chapter 20 that the system is nonminimum phase. If we take, for example $\gamma_1 = 0.43$ and $\gamma_2 = 0.34$, then the system has a NMP zero at s = 0.0229, and the relative gain array becomes

$$\boldsymbol{\Lambda} = \begin{bmatrix} -0.64 & 1.64 \\ 1.64 & -0.64 \end{bmatrix}$$

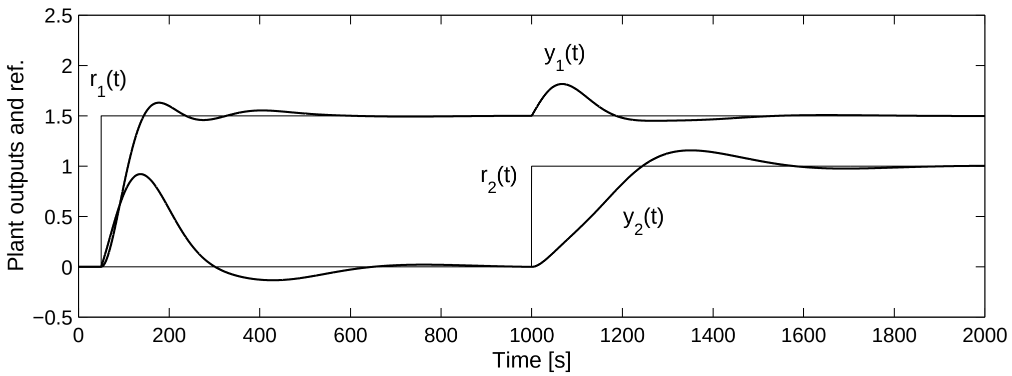
This suggests that (y_1, y_2) should be commuted for the purposes of decentralized control,. This is physically reasonable, given the flow patterns produced in this case. This leads to a new RGA of

$$\mathbf{\Lambda} = \begin{bmatrix} 1.64 & -0.64 \\ -0.64 & 1.64 \end{bmatrix}$$

Note, however, that control will still be much harder than in the minimum-phase case. For example, the following decentralized controllers give the results shown on the next slide.

$$C_1(s) = 0.5\left(1 + \frac{1}{30s}\right); \qquad C_2(s) = 0.3\left(1 + \frac{1}{50s}\right)$$

Figure 21.4: *Decentralized control of a nonminimum-phase four-tank system*



Robustness Issues in Decentralized Control

One way to carry out a decentralized control design is to use a diagonal *nominal* model. The off-diagonal terms then represent under-modelling, in the terminology of Chapter 3.

Thus, say we have a model $\mathbf{G_o}(s)$, then the nominal model for decentralized control could be chosen as

$$\mathbf{G_o^d}(s) = diag\{g_{11}^o, \dots, g_{mm}^o(s)\}$$

and the additive model error would be

$$\mathbf{G_\epsilon}(s) = \mathbf{G_o}(s) - \mathbf{G_o^d}(s); \qquad \mathbf{G_{\Delta l}}(s) = \mathbf{G_\epsilon}(s)[\mathbf{G_o^d}(s)]^{-1}$$

With this as a background, we can employ the robustness checks described in Chapter 20. We recall that a sufficient condition for robust stability is

$$\overline{\sigma}\left(\mathbf{G_{\Delta l}}(j\omega)\mathbf{T_o}(j\omega)\right) < 1 \qquad \forall\omega \in \mathsf{R}$$

where $\overline{\sigma}(\mathbf{G}_{\Delta 1}(j\omega)\mathbf{T_o}(j\omega))$ is the maximum singular value of $\mathbf{G}_{\Delta 1}(j\omega)\mathbf{T_o}(j\omega)$.

Example

Consider again the system

$$\mathbf{G_o}(s) = \begin{bmatrix} \dfrac{2}{s^2+3s+2} & \dfrac{k_{12}}{s+1} \\ \dfrac{k_{21}}{s^2+2s+1} & \dfrac{6}{s^2+5s+6} \end{bmatrix}$$

In this case, the various matrices arising in the centralized design are

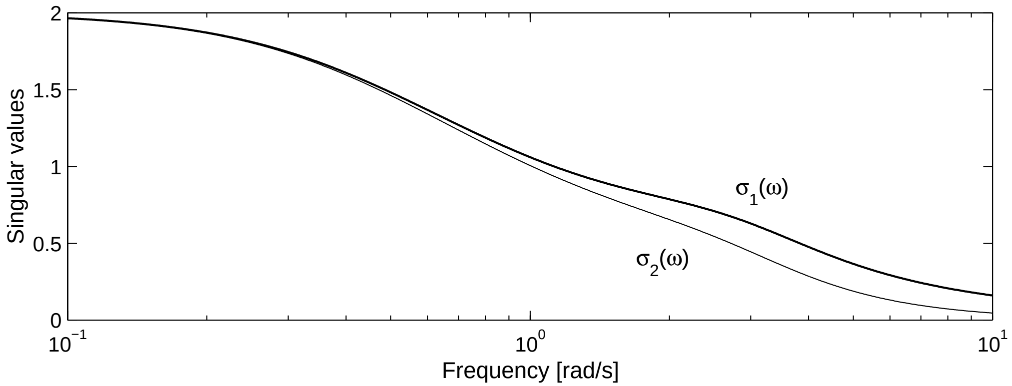
$$\mathbf{T_o}(s) = \frac{9}{s^2+4s+9}\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix} \qquad \mathbf{G_o^d}(s) = \begin{bmatrix}\frac{2}{s^2+3s+2} & 0\\ 0 & \frac{6}{s^2+5s+6}\end{bmatrix}$$

$$\mathbf{G}_\epsilon(s) = \begin{bmatrix}0 & \frac{k_{12}}{s+1}\\ \frac{k_{21}}{s^2+2s+1} & 0\end{bmatrix} \qquad \mathbf{G_{\Delta l}}(s) = \begin{bmatrix}0 & \frac{k_{12}(s^2+5s+6)}{6(s+1)}\\ \frac{k_{21}(s+2)}{2(s+1)} & 0\end{bmatrix}$$

$$\mathbf{G_{\Delta l}}(s)\mathbf{T_o}(s) = \begin{bmatrix}0 & \frac{3k_{12}(s^2+5s+6)}{2(s+1)(s^2+4s+9)}\\ \frac{9k_{21}(s+2)}{2(s+1)(s^2+4s+9)} & 0\end{bmatrix}$$

The singular values, in this case, are simply the magnitudes of the two off-diagonal elements. These are plotted on the next slide for normalized values $k_{12} = k_{21} = 1$.

Figure 21.5: Singular Values of $\mathbf{G}_{\Delta 1}(j\omega)\mathbf{T_o}(j\omega)$



We see that a sufficient condition for robust stability of the decentralized control, with the pairing (u_1, y_1), (u_2, y_2), is that $|k_{12}| < 1$ and $|k_{21}| < 1$. Observe that this is conservative, but consistent with the performance results presented earlier.

Example

Consider a MIMO system with

$$\mathbf{G}(s) = \begin{bmatrix} \frac{1}{s+1} & 0.25\frac{10s+1}{(s+1)(s+2)} \\ 0.25\frac{10s+1}{(s+1)(s+2)} & \frac{2}{s+2} \end{bmatrix}; \qquad \mathbf{G_o}(s) = \begin{bmatrix} \frac{1}{s+1} & 0 \\ 0 & \frac{2}{s+2} \end{bmatrix}$$

We first observe that the RGA for the nominal model $\mathbf{G_o}(s)$ is given by

$$\boldsymbol{\Lambda} = \begin{bmatrix} 1.0159 & -0.0159 \\ -0.0159 & 1.0159 \end{bmatrix}$$

This value of the RGA might lead to the hypothesis that a correct pairing of inputs and outputs has been made and that the interaction is weak. We thus proceed to do a decentralized design leading to a diagonal controller **C**(s) to achieve a complementary sensitivity $\mathbf{T_o}(s)$, where

$$\mathbf{T_o}(s) = \frac{9}{s^2+4s+9}\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}; \qquad \mathbf{C}(s) = \begin{bmatrix}\dfrac{9(s+1)}{s(s+4)} & 0\\ 0 & \dfrac{9(s+2)}{2s(s+4)}\end{bmatrix}$$

However, this controller, when applied to control the full plant $\mathbf{G}(s)$, leads to closed-loop poles located at -6.00, -2.49 ± j4.69, 0.23 ± j1.36, and -0.50 - an *unstable closed loop* !

The lack of robustness in this example can be traced to the fact that the required closed-loop bandwidth includes a frequency range where the off-diagonal frequency response is significant.

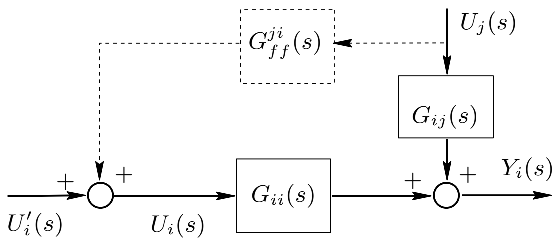
Feedforward Action in Decentralized Control

Although it usually will not aid robust stability, the performance of decentralized controllers is often significantly enhanced by the judicious choice of feedforward action to reduce coupling. Consider, for example, the output response at port #1, i.e.

$$Y_1(s) = G_{11}(s)U_1(s) + \sum_{i=2}^{m} G_{1i}(s)U_i(s)$$

and, for simplicity, we consider only the effect of the j^{th} loop on the i^{th} loop. We can then apply the feedforward ideas developed in Chapter 10 to obtain the architecture shown on the next slide.

Figure 21.6: Feedforward action in decentralized control



The feedforward gain $G_{ff}^{ji}(s)$ should be chosen in such a way that the coupling from the j^{th} loop to the i^{th} loop is compensated *in a particular*, *problem-dependent frequency band* $[0 \;\; \omega_{ff}]$ - i.e.

$$G_{ff}^{ji}(j\omega)G_{ii}(j\omega) + G_{ij}(j\omega) \approx 0 \qquad \forall \omega \in [0 \;\; \omega_{ff}]$$

This can also be written as

$$G_{ff}^{ji}(j\omega) \approx -[G_{ii}(j\omega)]^{-1}G_{ij}(j\omega) \qquad \forall \omega \in [0 \;\; \omega_{ff}]$$

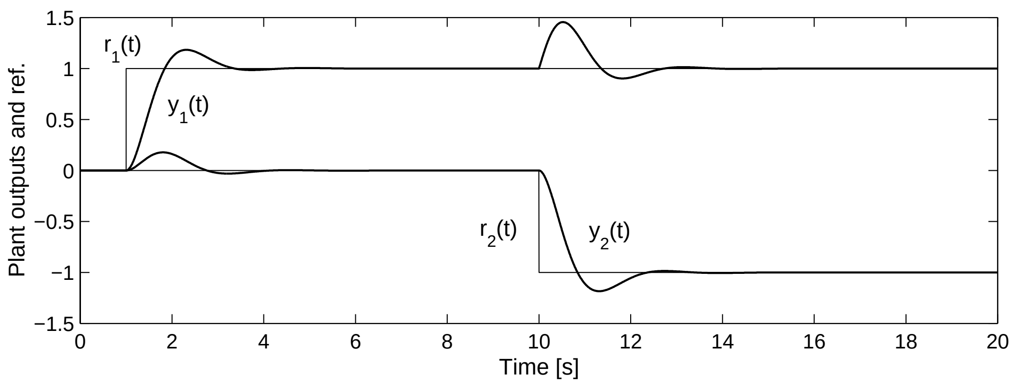
from which we observe the necessity to build an inverse. Hence all of the issues associated with building inverses discussed in earlier chapters arise again.

Example 21.6

Consider again the system

$$\mathbf{G_o}(s) = \begin{bmatrix} \dfrac{2}{s^2+3s+2} & \dfrac{k_{12}}{s+1} \\ \dfrac{k_{21}}{s^2+2s+1} & \dfrac{6}{s^2+5s+6} \end{bmatrix}$$

with k_{12} = -1 and k_{21} = 0.5. We recall the results presented earlier for this case.



We see that there is little coupling from the first to the second loop, but relatively strong coupling from the second to the first loop. This suggests that feedforward from the second input to the first loop may be beneficial. To illustrate, we choose $G_{ff}^{ji}(s)$ to completely compensate the coupling at d.c., i.e. $G_{ff}^{ji}(s)$ is chosen to be a constant $G_{ff}^{ji}(s) = \alpha$, satisfying

$$\alpha G_{11}(0) = -G_{12}(0) \Longrightarrow \alpha = 1$$

The resulting modified MIMO system can be seen to be modeled by

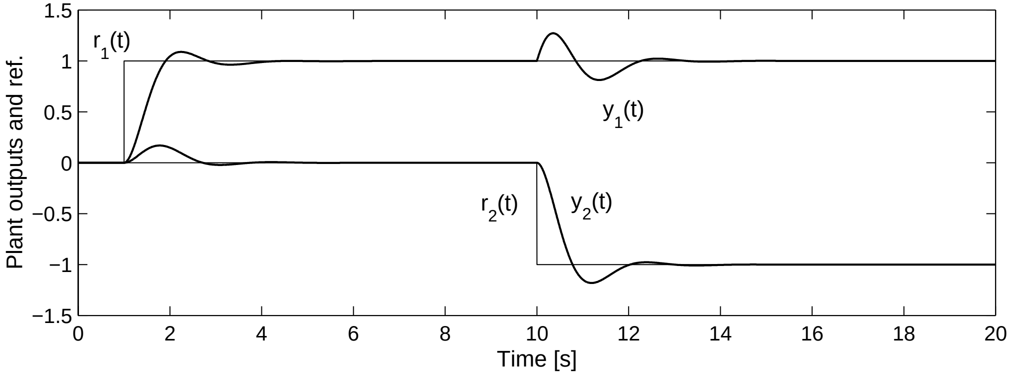
$$Y(s) = \mathbf{G_o}(s)\begin{bmatrix} U_1(s) \\ U_2(s) \end{bmatrix} = \mathbf{G_o}(s)\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} U_1'(s) \\ U_2(s) \end{bmatrix} = \mathbf{G_o'}(s)\begin{bmatrix} U_1'(s) \\ U_2(s) \end{bmatrix}$$

where

$$\mathbf{G_o'}(s) = \begin{bmatrix} \dfrac{2}{s^2+3s+2} & \dfrac{-s}{s^2+3s+2} \\ \dfrac{0.5}{s^2+2s+1} & \dfrac{6.5s^2+14.5s+9}{(s^2+2s+1)(s^2+5s+6)} \end{bmatrix}$$

The RGA is now $\boldsymbol{\Lambda}$ = diag(1, 1) and when we redesign the decentralized controller, we obtain the results presented on the next slide.

Figure 21.7: *Performance of a MIMO decentralized control loop with interaction feedforward*



The above examples indicate that a little coupling introduced into the controller can be quite helpful. This, however, raises the question of how we can systematically design coupled controllers that rigorously take into account multivariable interaction. This motivates us to study the latter topic, which will be taken up in the next chapter. Before ending this chapter, we investigate whether there exist simple ways of converting an inherently MIMO problem to a set of SISO problems.

Converting MIMO problems to SISO Problems

Many MIMO problems can be modified so that decentralized control becomes a more viable (*or attractive*) option. For example, one can sometimes use a precompensator to turn the resultant system into a more nearly diagonal transfer function.

To illustrate, say the nominal plant transfer function is $\mathbf{G_o}(s)$. If we introduce a precompensator $\mathbf{P}(s)$, then the control loop appears as in the figure below.

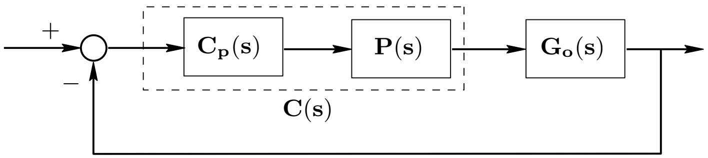


Figure 21.8: *Feedback control with plant precompensation*

The design of $\mathbf{C_p}(s)$ can then be based on the equivalent plant.

$$\mathbf{H}(s) = \mathbf{G_o}(s)\mathbf{P}(s)$$

Several comments are in order regarding this strategy:

(*i*) A first attempt at designing $\mathbf{P}(s)$ might be to approximate $\mathbf{G_o}(s)^{-1}$ in some way. For example, one might use the d.c. gain matrix $\mathbf{G_o}(0)^{-1}$ as a precompensator, assuming this exists.

(*ii*) If dynamic precompensators are used, then one needs to check that no unstable pole-zero cancellations are introduced between the compensator and the original plant.

(*iii*) Various measures of resultant interactions can be introduced. For example, the following terminology is frequently employed in this context.

Dynamically decoupled

Dynamically decoupled: Here, every output depends on one and only one input. The transfer-function matrix $\mathbf{H}(s)$ is diagonal for all s. In this case, the problem reduces to separate SISO control loops.

Band-decoupled and statically decoupled systems: When the transfer-function matrix $\mathbf{H}(j\omega)$ is diagonal only in a finite frequency band, we say that the system is decoupled in that band. In particular, we will say, when $\mathbf{H}(0)$ is diagonal, that the system is statically decoupled.

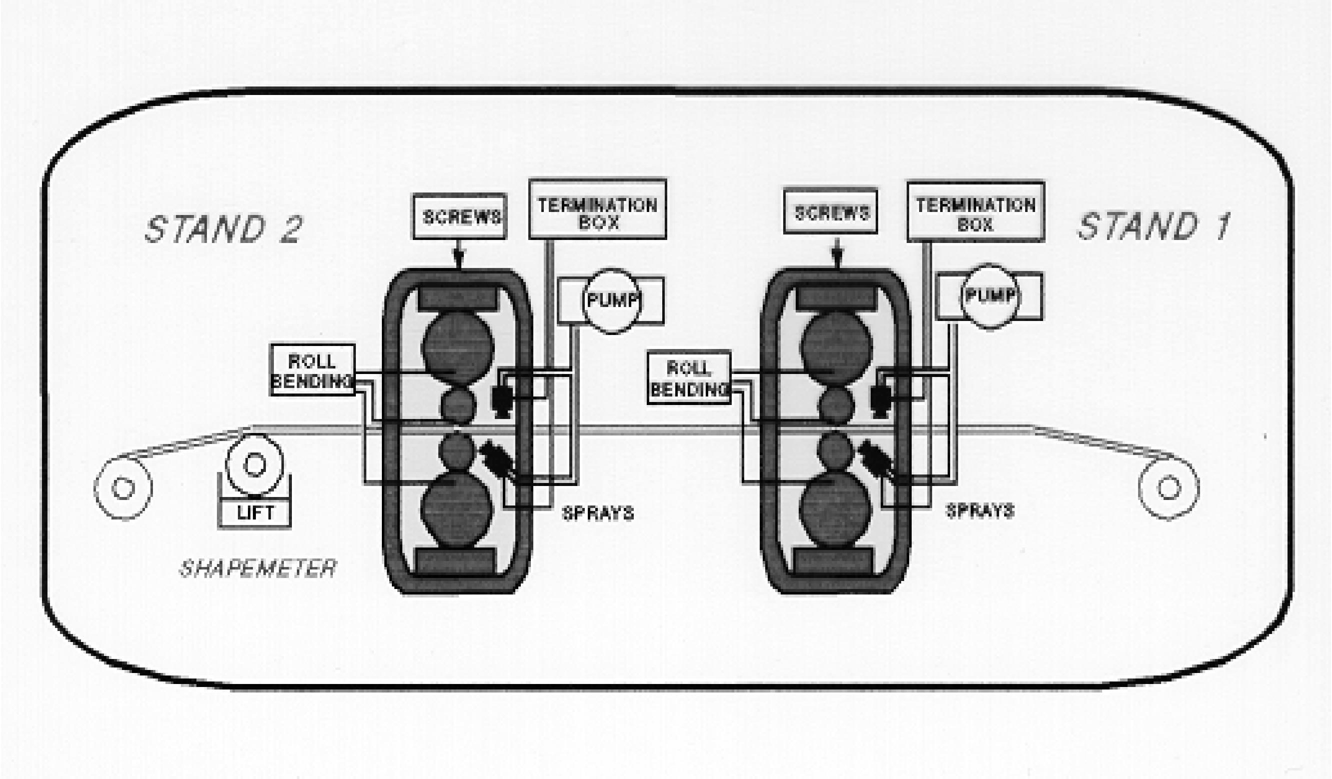
Triangularly coupled systems: A system is triangularly coupled when the inputs and outputs can be ordered in such a way that the transfer-function matrix $\mathbf{H}(s)$ is either upper or lower triangular, for all s. The coupling is then hierarchical.

Industrial Case Study (*Strip Flatness Control*)

An an illustration of the use of simple precompensators to convert a MIMO problem into one in which SISO techniques can be employed, we consider the problem of strip flatness control in rolling mills. Actually, very similar issues arise in many other problems including paper making and plastic extrusion.

The next slide shows a typical rolling stand configuration.

Typical rolling stand configuration



What is flatness in a Rolling Mill?

If rolling results in a nonuniform reduction of the strip thickness across the strip width, then a residual stress will be created, and buckling of the final product may occur. A practical difficulty is that flatness defects can be *pulled out* by the applied strip tensions, so that they are not visible to the mill operator. However, the buckling will become apparent as the coil is unwound or after it is slit or cut to length in subsequent processing operations.

Source of Flatness Problems

There are several sources of flatness problems, including the following:

- roll thermal cambers
- incoming fed disturbances (*profile*, *hardness*, *thickness*)
- transverse temperature gradients
- roll stack deflections
- incorrect ground roll cambers
- roll wear
- inappropriate mill setup (*reduction*, *tension*, *force*, *roll bending*)
- lubrication effects.

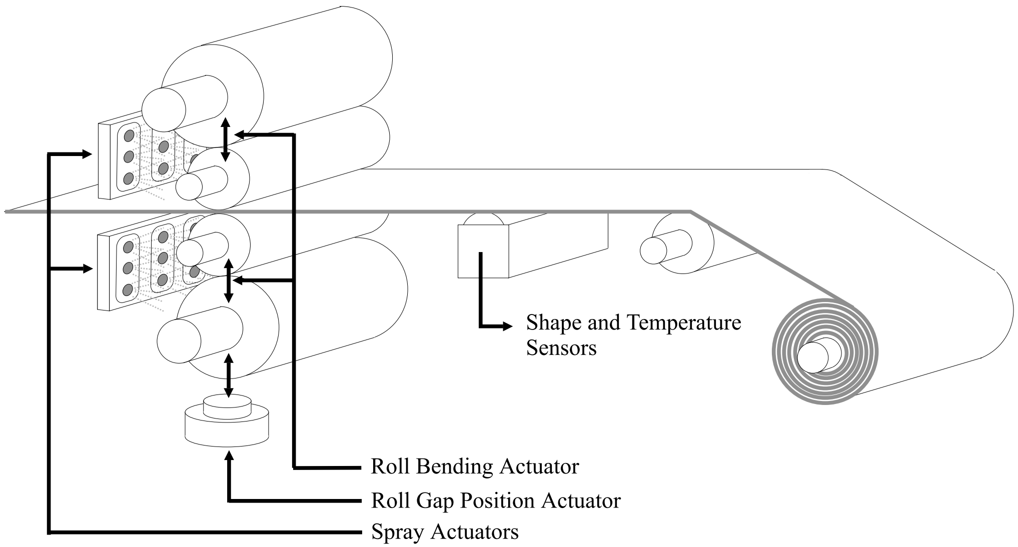
On the other hand, there are strong economic motives to control strip flatness, including the following:

- improved yield of prime-quality strip
- increased throughput, due to faster permissible acceleration, reduced threading delay, and higher rolling speed on shape-critical products
- more efficient recovery and operation on such downstream units as annealing and continuous-process lines
- reduced reprocessing of material on tension-leveling lines or temper-rolling mills.

Control Options

In this context, there are several control options to achieve improved flatness. These include roll tilt, roll bending, and cooling sprays. These typically can be separated by preprocessing the measured shape. Here, we will focus on a particular aspect of the cooling spray option. Note that flatness defects can be measured across the strip by using a special instrument called a Shape Meter. A typical control configuration is shown on the next slide.

Figure 21.9: *Typical flatness-control set-up for rolling mill*

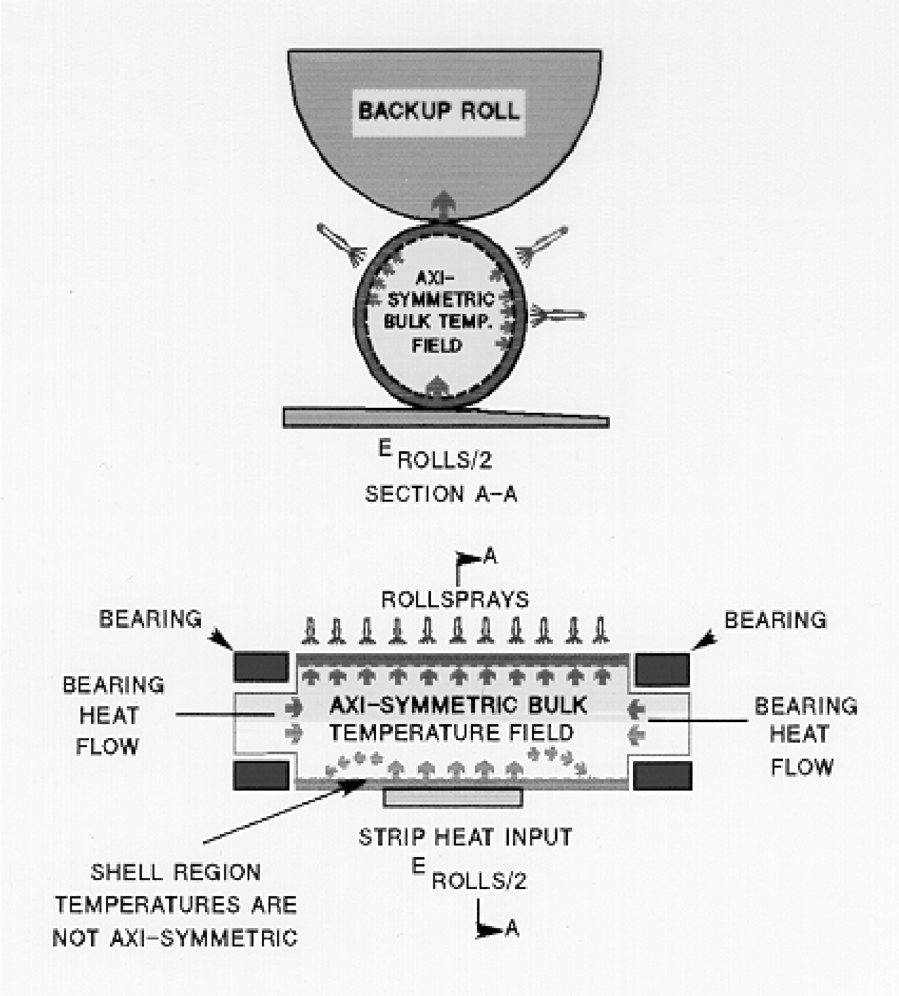


In this configuration, numerous cooling sprays are located across the roll, and the flow through each spray is controlled by a valve. The *cool* water sprayed onto the roll reduces the thermal expansion. The interesting thing is that each spray affects a large section of the roll, not just the section directly beneath it. This leads to an interactive MIMO system, rather than a series of decoupled SISO systems.

The thermal properties of the roll can be modeled using basic laws of physics. This leads to a partial differential equation, however, this can be discretized to give a finite dimenional model. Such a model can then be used as a *calibration model* to test control system design strategies.

The main components of the heat flow inside a typical roll are shown on the next slide.

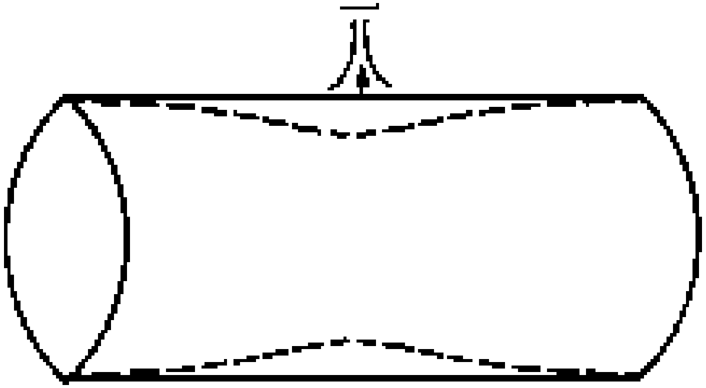
Internal roll heat flows



For the purpose of control system design, it suffices to use a simpler model. Such a model can be developed by approximating the observed behavior of the more complex *calibration model*. A key feature of the observed behavior is that a single cooling spray (*one of the actuators*) effects the radial diameter of the roll and hence the measured strip shape over a extended spatial area. This is diagrammatically shown on the next slide.

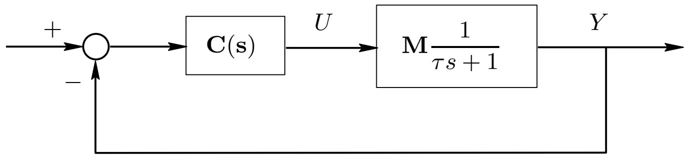
Effect of a single spray on roll diameter

Action of single spray



Based on the above discussion, a simplified model for this system (*ignoring nonlinear heat-transfer effects*, *etc*.) is shown in the block diagram on the next slide, where U denotes a vector of spray valve positions and Y denotes the roll-thickness vector. (*The lines indicate vectors rather than single signals*).

Figure 21.10: *Simplified flatness-control feedback loop*



The sprays affect the roll in a roughly exponential fashion as described by the matrix **M**:

$$\mathbf{M} = \begin{bmatrix} 1 & \alpha & \alpha^2 & \cdots \\ \alpha & 1 & & \\ \alpha^2 & & \ddots & \vdots \\ \vdots & & 1 & \alpha \\ & \cdots & \alpha & 1 \end{bmatrix}$$

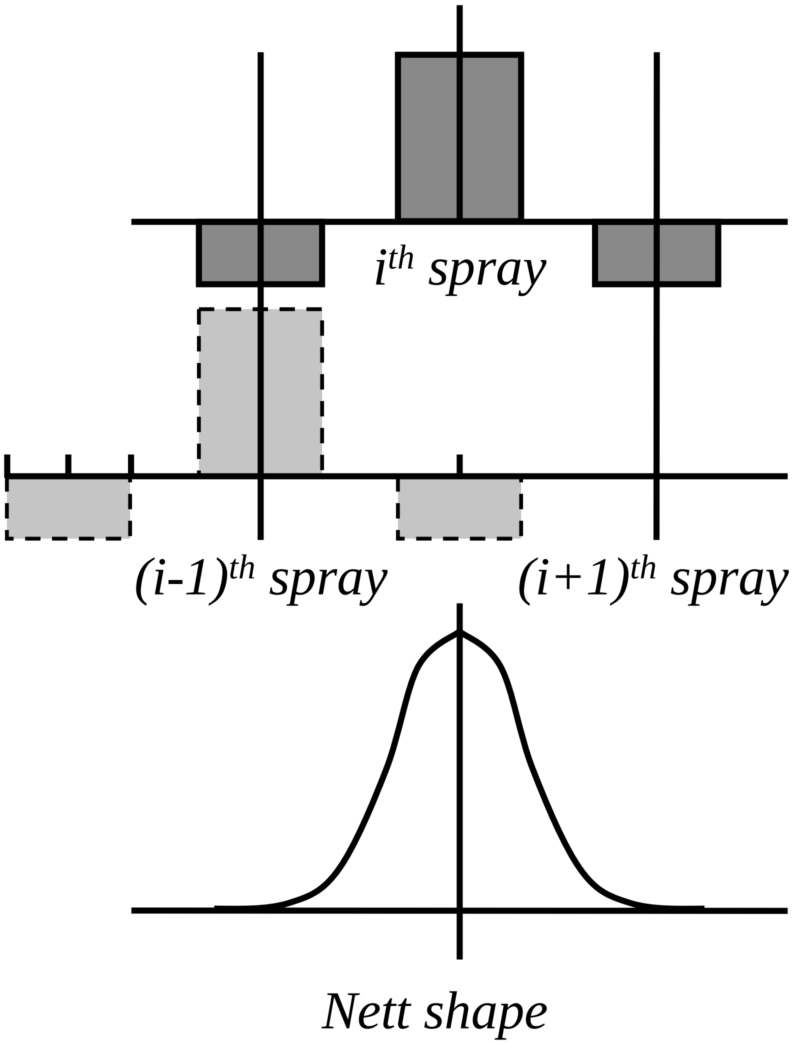
The parameter α represents the level of interactivity in the system and is determined by the number of sprays present and how close together they are.

An interesting thing about this simplified model is that the interaction is captured totally by the d.c. gain matrix **M**. This suggests that we could design an approximate precompensator by simply inverting this matrix. This leads to

$$\mathbf{M}^{-1} = \begin{bmatrix} \frac{1}{1-\alpha^2} & \frac{-\alpha}{1-\alpha^2} & 0 & \cdots & 0 \\ \frac{-\alpha}{1-\alpha^2} & \frac{1+\alpha^2}{1-\alpha^2} & & & \vdots \\ 0 & & \ddots & & 0 \\ \vdots & & & \frac{1+\alpha^2}{1-\alpha^2} & \frac{-\alpha}{1-\alpha^2} \\ 0 & \cdots & 0 & \frac{-\alpha}{1-\alpha^2} & \frac{1}{1-\alpha^2} \end{bmatrix}$$

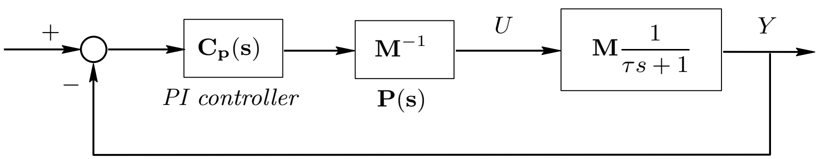
Using this matrix to decouple the system has a nice physical interpretation. Namely, it amounts to turning off surrounding sprays when a spray is turned on. This makes sense physically since we are preventing the spread of the cooling effect by use of adjacent sprays. The essential idea of the decoupling control strategy is shown on the next slide.

Pictorial representation of coordinated use of sprays to decouple control action



In summary, we can (*approximately*) decouple the system simply by multiplying the control vector by the appropriate inverse. This set-up is shown in the block diagram on the next slide.

Figure 21.11: *Flatness control with precompensation*



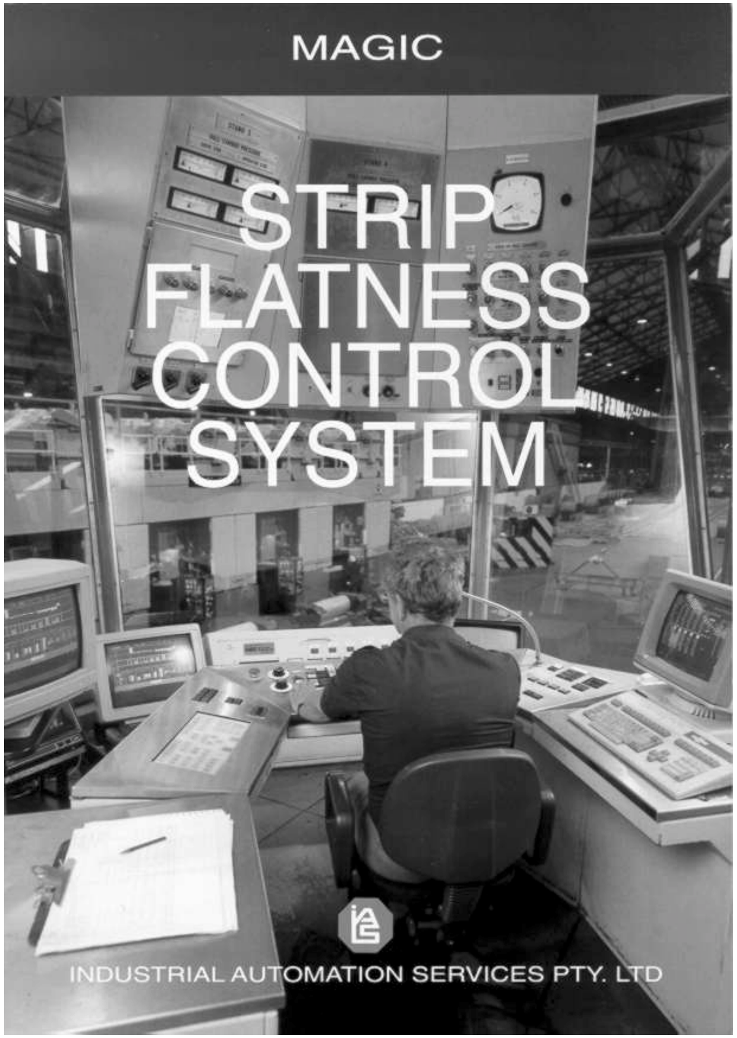
The nominal decoupled system then becomes simply $H(s) = diag \frac{1}{(\tau s+1)}$. With this new model, the controller can be designed by using SISO methods. For example a set of simple PI controllers linking each shape meter with the corresponding spray would seem to suffice. (*We assume that the shape meters measure the shape of the rolls perfectly*).

This idea is routinely used in this particular application and leads to excellent results. (*Of course, the practical problem has many other features that we leave aside so as not to distract from our key point here*).

Actually, control problems almost identical to the above can be found in many alternative industrial situations where there are longitudinal and traverse effects. Examples are paper making and plastic extrusion.

Strip flatness systems of the type (*briefly*) described here are available commercially. The following slides have been made from pamphlets describing a commercial system sold by *Industrial Automation Services Pty. Ltd.*

www.indauto.com.au

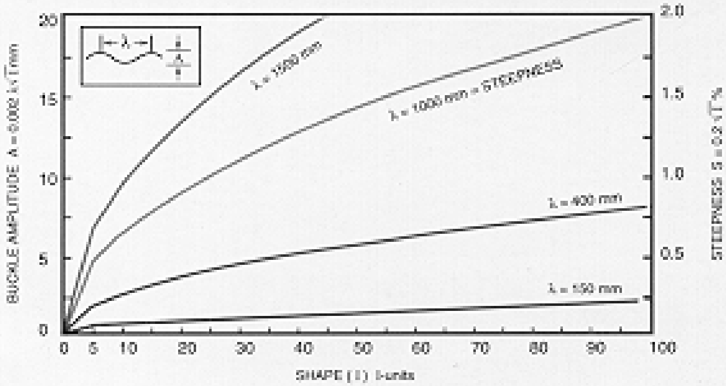

MAGIC
STRIP
FLATNESS
CONTROL
SYSTEM
INDUSTRIAL AUTOMATION SERVICES PTY. LTD

WHAT CAUSES POOR FLATNESS?

If rolling results in a non-uniform reduction of the strip thickness across the strip width, then a residual stress will be created and buckling of the final product may occur. A practical difficulty is that flatness defects can be "pulled out" by the applied strip tensions, so that they are not visible to the mill operator. The buckling may become apparent as the coil is unwound or after it is slit or cut to length in subsequent processing operations. Cutting the rolled strip into multiple narrow strips permits the residual longitudinal stress to be relieved and the fractional length strains to be measured and converted to "I"-units; a commonly accepted measure of online flatness.

Another measure of flatness is "steepness" which is the ratio of peak-to-peak buckle amplitude to longitudinal buckle wavelength, expressed in percent. The relationship between the different measures of flatness can be represented graphically as a function of the wavelength.

RELATIONSHIP BETWEEN SHAPE (I), BUCKLE AMPLITUDE (A) AND STEEPNESS (S) FOR VARIOUS WAVELENGTHS (λ)



Examples of the numerous factors which cause flatness problems:

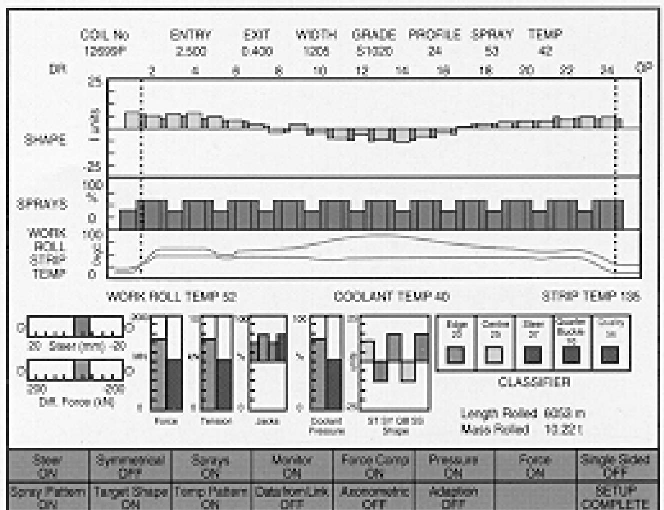
- roll thermal cambers
- incoming feed disturbances (profile, hardness, thickness)
- transverse temperature gradients
- roll stack deflections
- incorrect ground roll cambers
- roll wear
- inappropriate mill setup (reduction, tension, force, roll bending)
- lubrication effects

WHY CONTROL FLATNESS?

If you are a manufacturer of flat rolled metal products then there is likely to be sound justification for installing a flatness control system on your cold mill or temper mill.

Benefits a flatness automation system can provide include:

- Improved yield of prime quality strip
- Statistically improved flatness throughout all products leading to increased market share in a competitive environment.
- Coil performance statistics available in graphical or summary form for presentation to customers.
- Increased throughput due to faster permissible acceleration, reduced threading delay and higher rolling speed on shape critical products



OPERATOR INTERFACE DISPLAY

- More efficient recovery and operation on downstream units such as annealing and continuous process lines
- Reduced reprocessing of material on tension levelling lines or temper rolling mills
- Less scrapping of product and hence more reliable, on-time, shipment of product
- Faster detection and diagnosis of flatness related problems due to statistical process control reporting facilities
- Long term storage and retrieval of quality information to meet statutory and client requirements
- Reduced operator fatigue
- More effective process management

STRIP FLATNESS CONTROL SYSTEM

Where is the MAGIC in Flatness Control?

Superficially, many flatness control systems appear similar, however a closer examination will often reveal significant differences in concept and detailed implementation. The MAGIC flatness control system was created with the objective of achieving a consistently high level of performance for any combination of mill type, product range and operating practice.

CONTROL TECHNOLOGY HIGHLIGHTS

Adaptive Setup: Actuator setup references for each coil are calculated with models and take into account conditions observed on previous coils by the application of adaptive, learning algorithms.

Temperature compensation: Where possible, measured temperatures or product dependent temperature profile models are used to compensate for thermal expansions across the strip width.

Asymmetric Control: Asymmetric shape errors are controlled by a combination of differential roll bending and rollgap adjustments. It is tailored to take account of particular mill characteristics such as screw clutching, actuator response times, and offset strip tracking.

Single-sided Control: Fast, single-sided control of localised edge errors is achieved by a combination of single-sided roll bending and rollgap adjustment.

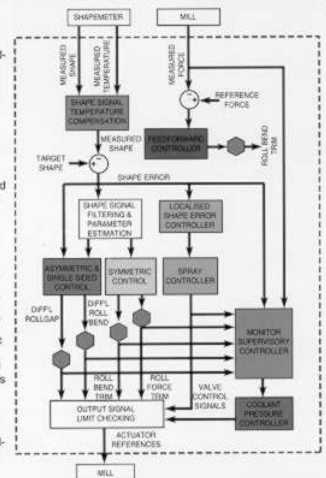
Symmetric Control: Symmetric shape errors are controlled by symmetric adjustment of the roll bending forces, side-shifting rolls and, when required, roll force.

Schedule Dependent Gains: All control system gains which are product or schedule-dependent are calculated within the adaptive setup calculation.

Non-Interactive Control: NIC design techniques are used to decouple control loops and prevent the operation of one loop interfering with another.

Actuator monitors: The dynamic responses of all actuators are monitored continuously to detect any malfunction.

Feed-forward Control: This compensates for the effect of measured force fluctuations on strip flatness by open loop roll bending adjustments.



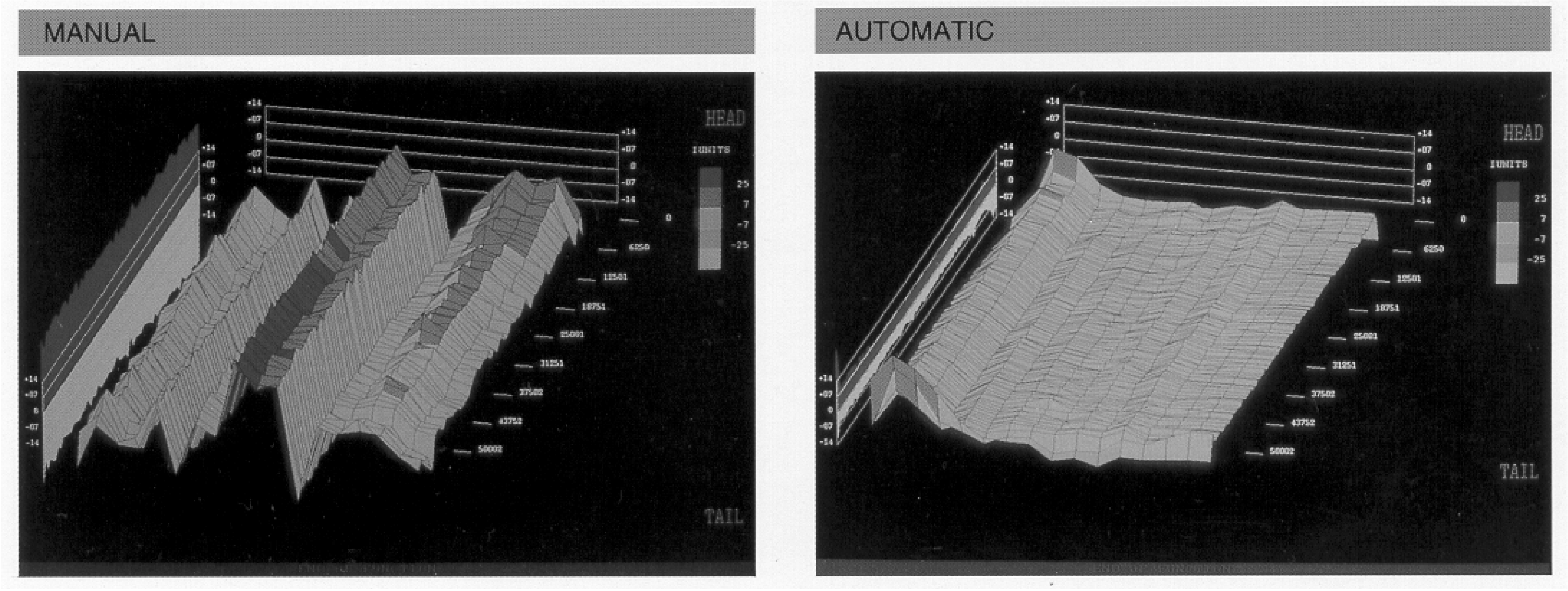
Localised Shape Error Control: Shape errors remaining after the asymmetric and symmetric controls have performed their functions, may be controlled if a suitable actuator is available. This may take the form of coolant controls or multiple backup roll support bearings such as NIPCO rolls on 4-high mills or eccentric backup supports on Sendzimir mills. Thermal camber is controlled by adjusting coolant pressure and spray patterns to ensure:

- compensation for zone interaction
- adequate lubrication of all zones to prevent friction marks or "dirty" strip defects
- limitation of the number of nozzles switched simultaneously to prevent excessive pressure variations
- accurate edge zone control
- compensation for cold rolls
- dynamic optimisation of valve switching to increase valve life.

Monitor Supervisory Control: This co-ordinates the various control functions to maintain the most effective operating conditions and prevent roll bending saturation and unacceptable spray control asymmetry.

Coolant Pressure Control: Adjusts the coolant pump speed or a control valve to achieve the desired pressure reference and prevent spray control saturation.

Impact of MIMO Controller



Summary

- A fundamental decision in MIMO synthesis pertains to the choice of decentralized versus full MIMO control.
- Completely decentralized control
  - In completely decentralized control, the MIMO system is approximated as a set of independent SISO systems
  - To do so, multivariable interactions are thought of as disturbances; this is an approximation, because the interactions involve feedback, whereas disturbance analysis actually presumes disturbances to be independent inputs.
  - When applicable, the advantage of completely decentralized control is that one can apply the simpler SISO theory.

- Applicability of this approximation depends on the neglected interaction dynamics, which can be viewed as modelilng errors; robustness analysis can be applied to determine their impact.
- Chances of success are increased by judiciously pairing inputs and outputs (for example, by using the Relative Gain Array, RGA) and by using feedforward.
- Feedforward is often a very effective tool in MIMO problems.
- Some MIMO problems can be better treated as SISO problems if a precompensator is first used.

- There are several ways to quantify interactions in multivariable systems, including their structure and their strength.
  - Interactions can have a completely general structure (*every input potentially affects every output*) or display particular patterns, such as *triangular* or *dominant diagonal*; they can also display frequency-dependent patterns, such as being *statistically decoupled* or *band-decoupled*.
  - The lower the strength of interaction, the more nearly a system behaves like a set of independent systems that can be analyzed and controlled separately.

- Weak coupling can be due to the nature of the interacting dynamics or to a separation in frequency range or time scale.
- The stronger is the interaction, the more important it becomes to view the multi-input multi-output system and its interactions as a whole.
- Compared to the SISO techniques discussed so far, viewing the MIMO systems and its interactions as a whole requires generalized synthesis and design techniques and insight. These will be the topics of the following two chapters.