4 Stability

4.1 Well-posedeness

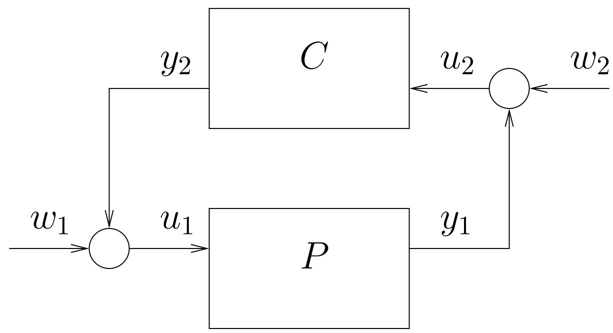

Figure 1: Feedback connection

Definition: A feedback system is said to be *well-posed* if all closed-loop matrices are well-defined and proper.

Lemma: The feedback system in Figure 1 is well-posed if and only if

$$I - C(\infty)P(\infty)$$

is invertible.
Proof: The system equations are

$$\begin{aligned} u_1 &= w_1 + y_2 & \qquad y_1 &= P(s)u_1 \\ u_2 &= w_2 + y_1 & \qquad y_2 &= C(s)u_2 \end{aligned}$$

so that

$$u_1 = w_1 + C(s)[w_2 + P(s)u_1]$$

and

$$[I - C(s)P(s)]u_1 = w_1 + C(s)w_2$$

For u_1 to be well defined $[I - C(s)P(s)]^{-1}$ must exist and be proper. This is the same as having $[I - C(\infty)P(\infty)]$ nonsingular. Note that if u_1 is well defined then u_2 will also be.

4.2 Internal Stability

Definition: The feedback system in Figure 1 is said to be *internally stable* if the transfer function from $\begin{pmatrix} w_1 \\ w_2 \end{pmatrix}$ to $\begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$ is in RH_∞.

Lemma: The feedback system in Figure 1 is said to be *internally stable* if and only if the transfer function

$$T_{wu}(s) = \begin{bmatrix} I & -C(s) \\ -P(s) & I \end{bmatrix}^{-1}$$

is in RH_∞.

Proof: Just determine the transfer function from $\begin{pmatrix} w_1 \\ w_2 \end{pmatrix}$ to $\begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$. From the diagram

$$\begin{aligned} u_1 &= w_1 + C(s)u_2 \\ u_2 &= w_2 + P(s)u_1 \end{aligned}$$

or

$$\begin{pmatrix} w_1 \\ w_2 \end{pmatrix} = \begin{bmatrix} I & -C(s) \\ -P(s) & I \end{bmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$$

Lemma: The transfer function

$$T_{wu}(s) := \begin{bmatrix} I & -C(s) \\ -P(s) & I \end{bmatrix}^{-1}$$

is in RH_∞ if and only if

$$\begin{bmatrix} I + C(s)[I - P(s)C(s)]^{-1}P(s) & C(s)[I - P(s)C(s)]^{-1} \\ [I - P(s)C(s)]^{-1}P(s) & [I - P(s)C(s)]^{-1} \end{bmatrix}$$

is in RH_∞.

Proof: Use the fact that if A is nonsingular

$$\begin{bmatrix} A & D \\ C & B \end{bmatrix}^{-1} = \begin{bmatrix} A^{-1} + A^{-1}DE^{-1}CA^{-1} & -A^{-1}DE^{-1} \\ -E^{-1}CA^{-1} & E^{-1} \end{bmatrix}$$

where

$$E = B - CA^{-1}D$$

is known as *Schur Complement* of the matrix.
In the case of the matrix in the lemma it produces

$$T_{wu}(s) = \begin{bmatrix} I + C(s)E(s)^{-1}P(s) & C(s)E(s)^{-1} \\ -E^{-1}P(s) & E(s)^{-1} \end{bmatrix}, \quad E(s) = I - P(s)C(s).$$

Example: (From Zhou, p. 69) Let

$$P(s) = \frac{s-1}{s+1}, \qquad\qquad C(s) = -\frac{1}{s-1}.$$

Then

$$\begin{aligned} [I - P(s)C(s)]^{-1} &= \left(1 + \frac{1}{s+1}\right)^{-1} = \frac{s+1}{s+2}, \\ 1 + C(s)[I - P(s)C(s)]^{-1}P(s) &= 1 - \frac{1}{s+2} = \frac{s+1}{s+2} \end{aligned}$$

and

$$\begin{bmatrix} I + C(s)[I - P(s)C(s)]^{-1}P(s) & C(s)[I - P(s)C(s)]^{-1} \\ [I - P(s)C(s)]^{-1}P(s) & [I - P(s)C(s)]^{-1} \end{bmatrix}$$
$$= \begin{bmatrix} \frac{s+1}{s+2} & -\frac{s+1}{(s-1)(s+2)} \\ \frac{s-1}{s+2} & \frac{s+1}{s+2} \end{bmatrix}$$

Not internally stable!

Corollary: If $C(s) \in RH_\infty$ then the feedback system in Figure 1 is *internally stable* if and only if $[I - P(s)C(s)]^{-1}P(s) \in RH_\infty$.

Proof: Necessity is immediate since $[I - P(s)C(s)]^{-1}P(s)$ is a block of $T_{wu}(s)$. Sufficiency commes from

$$\begin{aligned}
[I - P(s)C(s)]^{-1} &= [I - P(s)C(s)]^{-1}[I - P(s)C(s) + P(s)C(s)] \\
&= [I - P(s)C(s)]^{-1}[I - P(s)C(s)] + [I - P(s)C(s)]^{-1}P(s)C(s) \\
&= I + [I - P(s)C(s)]^{-1}P(s)C(s) \in RH_\infty
\end{aligned}$$

since $C(s)$ and $[I - P(s)C(s)]^{-1}P(s)$ are in RH_∞.

Corollary: If $P(s) \in RH_\infty$ then the feedback system in Figure 1 is *internally stable* if and only if $C(s)[I - P(s)C(s)]^{-1} \in RH_\infty$.

Proof: Necessity is immediate again.
Sufficiency commes from

$$\begin{aligned}
[I - P(s)C(s)]^{-1} &= [I - P(s)C(s) + P(s)C(s)][I - P(s)C(s)]^{-1} \\
&= [I - P(s)C(s)][I - P(s)C(s)]^{-1} + P(s)C(s)[I - P(s)C(s)]^{-1} \\
&= I + P(s)C(s)[I - P(s)C(s)]^{-1} \in RH_\infty
\end{aligned}$$

since $C(s)$ and $[I - P(s)C(s)]^{-1}P(s)$ are in RH_∞.

Lemma: Let $P(s)$ and $C(s)$ have state space realizations

$$P(s) = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right], \qquad C(s) = \left[\begin{array}{c|c} \hat{A} & \hat{B} \\ \hline \hat{C} & \hat{D} \end{array}\right].$$

Assume that (A, B) and $(\hat{A}, \hat{B})$ are stabilizable and that (A, C) and $(\hat{A}, \hat{C})$ are detectable. Then the feedback system in Figure 1 is said to be *internally stable* if and only if the realization

$$T_{wu}(s) = \left[\begin{array}{cc|cc} A + B\Delta^{-1}\hat{D}C & B\Delta^{-1}\hat{C} & B\Delta^{-1} & B\Delta^{-1}\hat{D} \\ \hat{B}\hat{\Delta}^{-1}C & \hat{A} + \hat{B}\hat{\Delta}^{-1}D\hat{C} & \hat{B}\hat{\Delta}^{-1}D & \hat{B}\hat{\Delta}^{-1} \\ \hline \Delta^{-1}\hat{D}C & \Delta^{-1}\hat{C} & \Delta^{-1} & \Delta^{-1}\hat{D} \\ \hat{\Delta}^{-1}C & \hat{\Delta}^{-1}D\hat{C} & \hat{\Delta}^{-1}D & \hat{\Delta}^{-1} \end{array}\right]$$

where

$$\Delta = (I - \hat{D}D)^{-1}, \qquad \hat{\Delta} = (I - D\hat{D})^{-1}$$

is Hurwitz.

Proof: First show that the above is indeed a realization for $T_{wu}(s)$. Then all we have to prove that this realization is stabilizable and detectable. Before we prove this note that

$$\begin{bmatrix} I & -\hat{D} \\ -D & I \end{bmatrix}^{-1} = \begin{bmatrix} I + \hat{D}\hat{\Delta}^{-1}D & \hat{D}\hat{\Delta}^{-1} \\ \hat{\Delta}^{-1}D & \hat{\Delta}^{-1} \end{bmatrix}$$

Using the identity

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(DA^{-1}B + C^{-1})^{-1}DA^{-1}$$

we have

$$I + \hat{D}\hat{\Delta}^{-1}D = I + \hat{D}(I - D\hat{D})^{-1}D = (I - \hat{D}D)^{-1} = \Delta^{-1}.$$

Also

$$\hat{D} - \hat{D}D\hat{D} = \hat{D}(I - D\hat{D}) = (I - \hat{D}D)\hat{D}$$

so that

$$\hat{D}\hat{\Delta}^{-1} = \hat{D}(I - D\hat{D})^{-1} = (I - \hat{D}D)^{-1}\hat{D} = \Delta^{-1}\hat{D}.$$

We have established that

$$\begin{bmatrix} I & -\hat{D} \\ -D & I \end{bmatrix}^{-1} = \begin{bmatrix} \Delta^{-1} & \Delta^{-1}\hat{D} \\ \hat{\Delta}^{-1}D & \hat{\Delta}^{-1} \end{bmatrix}$$

Define

$$\left[\begin{array}{c|c} \tilde{A} & \tilde{B} \\ \hline \tilde{C} & \tilde{D} \end{array}\right] := \left[\begin{array}{cc|cc} A & 0 & B & 0 \\ 0 & \hat{A} & 0 & \hat{B} \\ \hline 0 & \hat{C} & I & -\hat{D} \\ C & 0 & -D & I \end{array}\right]$$

such that

$$\begin{bmatrix} A + B\Delta^{-1}\hat{D}C & B\Delta^{-1}\hat{C} \\ \hat{B}\hat{\Delta}^{-1}C & \hat{A} + \hat{B}\hat{\Delta}^{-1}D\hat{C} \end{bmatrix} = \tilde{A} + \tilde{B}\tilde{D}^{-1}\tilde{C},$$
$$\begin{bmatrix} \Delta^{-1}\hat{D}C & \Delta^{-1}\hat{C} \\ \hat{\Delta}^{-1}C & \hat{\Delta}^{-1}D\hat{C} \end{bmatrix} = \tilde{D}^{-1}\tilde{C}$$

Now assume that

$$\left(\begin{bmatrix} A + B\Delta^{-1}\hat{D}C & B\Delta^{-1}\hat{C} \\ \hat{B}\hat{\Delta}^{-1}C & \hat{A} + \hat{B}\hat{\Delta}^{-1}D\hat{C} \end{bmatrix}, \quad \begin{bmatrix} \Delta^{-1}\hat{D}C & \Delta^{-1}\hat{C} \\ \hat{\Delta}^{-1}C & \hat{\Delta}^{-1}D\hat{C} \end{bmatrix} \right)$$
$$= \left(\tilde{A} + \tilde{B}\tilde{D}^{-1}\tilde{C}, \quad \tilde{D}^{-1}\tilde{C} \right)$$

is not stabilizable, that is, there exists $x \neq 0$ and $\lambda + \lambda^* > 0$ such that

$$\left(\tilde{A} + \tilde{B}\tilde{D}^{-1}\tilde{C} \right) x = \lambda x, \qquad \tilde{D}^{-1}\tilde{C}x = 0$$

which implies, because $\tilde{D}$ is nonsingular,

$$\tilde{A}x = \lambda x, \qquad \tilde{C}x = 0.$$

Expanding we conclude that

$$Ax_1 = \lambda x_1, \qquad Cx_1 = 0, \qquad \hat{A}x_2 = \lambda x_2, \qquad Cx_2 = 0$$

which implies that (A, C) and $(\hat{A}, \hat{C})$ are not detectable, establishing a contradiction.

4.3 Coprime Factorization over RH_∞

Lemma: Two rational matrices $M(s)$ and $N(s)$ in RH_∞ are *right coprime* if and only if there exists rational matrices $\hat{X}(s)$ and $\hat{Y}(s)$ in RH_∞ such that

$$\hat{X}(s)M(s) + \hat{Y}(s)N(s) = \begin{bmatrix} \hat{X}(s) & \hat{Y}(s) \end{bmatrix} \begin{bmatrix} M(s) \\ N(s) \end{bmatrix} = I.$$

Lemma: Two rational matrices $\hat{M}(s)$ and $\hat{N}(s)$ in RH_∞ are *left coprime* if and only if there exists rational matrices $X(s)$ and $Y(s)$ in RH_∞ such that

$$\hat{M}(s)X(s) + \hat{N}(s)Y(s) = \begin{bmatrix} \hat{M}(s) & \hat{N}(s) \end{bmatrix} \begin{bmatrix} X(s) \\ Y(s) \end{bmatrix} = I.$$

Theorem: Let $H(s)$ be a proper rational matrix. There exists matrices $M(s)$, $N(s)$, $\hat{M}(s)$, $\hat{N}(s)$, $X(s)$, $Y(s)$, $\hat{X}(s)$, and $\hat{Y}(s)$ all in RH_∞ such that

$$H(s) = N(s)M^{-1}(s) = \hat{M}^{-1}(s)\hat{N}(s)$$

and

$$\begin{bmatrix} \hat{X}(s) & \hat{Y}(s) \\ -\hat{N}(s) & \hat{M}(s) \end{bmatrix} \begin{bmatrix} M(s) & -Y(s) \\ N(s) & X(s) \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}.$$

Proof: Let $H(s) = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right]$ be a minimal realization[2] for $H(s)$ and define

$$\begin{bmatrix} \hat{X}(s) & \hat{Y}(s) \\ -\hat{N}(s) & \hat{M}(s) \end{bmatrix} = \left[\begin{array}{c|cc} A + LC & -(B + LD) & L \\ \hline K & I & 0 \\ C & -D & I \end{array}\right],$$

$$\begin{bmatrix} M(s) & -Y(s) \\ N(s) & X(s) \end{bmatrix} = \left[\begin{array}{c|cc} A + BK & B & -L \\ \hline K & I & 0 \\ C + DK & D & I \end{array}\right].$$

[2]We just need (A, B) stabilizable and (A, C) detectable.

Then

$$
\begin{aligned}
&\begin{bmatrix} \hat{X}(s) & \hat{Y}(s) \\ -\hat{N}(s) & \hat{M}(s) \end{bmatrix} \begin{bmatrix} M(s) & -Y(s) \\ N(s) & X(s) \end{bmatrix} \\
&\quad = \left[\begin{array}{c|cc} A+LC & -(B+LD) & L \\ \hline K & I & 0 \\ C & -D & I \end{array}\right] \left[\begin{array}{c|cc} A+BK & B & -L \\ \hline K & I & 0 \\ C+DK & D & I \end{array}\right] \\
&\quad = \left[\begin{array}{cc|cc} A+LC & LC-BK & -B & L \\ 0 & A+BK & B & -L \\ \hline K & K & I & 0 \\ C & C & 0 & I \end{array}\right] \\
&\quad = \left[\begin{array}{c|c} \begin{bmatrix} I & -I \\ 0 & I \end{bmatrix} \begin{bmatrix} A+LC & 0 \\ 0 & A+BK \end{bmatrix} \begin{bmatrix} I & I \\ 0 & I \end{bmatrix} & \begin{bmatrix} I & -I \\ 0 & I \end{bmatrix} \begin{bmatrix} 0 & 0 \\ B & -L \end{bmatrix} \\ \hline \begin{bmatrix} K & 0 \\ C & 0 \end{bmatrix} \begin{bmatrix} I & I \\ 0 & I \end{bmatrix} & \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \end{array}\right] \\
&\quad = \left[\begin{array}{cc|cc} A+LC & 0 & 0 & 0 \\ 0 & A+BK & B & -L \\ \hline K & 0 & I & 0 \\ C & 0 & 0 & I \end{array}\right]
\end{aligned}
$$

What is left to prove is that all these transfer matrices are in RH_∞. But since (A, B) is stabilizable and (A, C) is detectable we can chose K and L such that they be all in RH_∞.
Then consider

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

and the feedback law

$$
u = v + Kx
$$

The closed loop system is

$$
\begin{aligned}
\dot{x} &= (A + BK)x + Bv \\
u &= Kx + v \\
y &= (C + DK)x + Dv
\end{aligned}
$$

whose transfer function is

$$\left[\begin{array}{c|c} A+BK & B \\ \hline K & I \\ C+DK & D \end{array}\right] = \begin{bmatrix} M(s) \\ N(s) \end{bmatrix}$$

or

$$y = N(s)\,v, \qquad\qquad u = M(s)\,v$$

which imply

$$y = N(s)\,v = N(s)M^{-1}(s)\,u = H(s)\,u$$

The same argument can be used to show that $\hat{M}(s)^{-1}\hat{N}(s) = H(s)$.

Example (Colaneri, p. 77):

$$H(s) = \begin{bmatrix} \dfrac{s^2+s-1}{s^3-s} & \dfrac{s}{s^2-1} \\[2ex] \dfrac{2s^2-1}{s^3-s} & \dfrac{s^2}{s^2-1} \end{bmatrix} = \left[\begin{array}{ccc|cc} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ \hline 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{array}\right]$$

Note that $\lambda(A) = \{-1, 0, 1\}$. Choose the stabilizing gains

$$K = \begin{bmatrix} 0 & 0 & -1 \\ -1 & -2 & 0 \end{bmatrix}, \qquad L = \begin{bmatrix} -4 & 0 \\ -4 & 0 \\ 1 & 0 \end{bmatrix}$$

and construct

$$\begin{bmatrix} M(s) & -Y(s) \\ N(s) & X(s) \end{bmatrix} = \left[\begin{array}{c|cc} A+BK & B & -L \\ \hline K & I & 0 \\ C+DK & D & I \end{array}\right]$$

$$= \left[\begin{array}{ccc|cccc} -1 & -1 & 0 & 0 & 1 & 4 & 0 \\ 1 & 0 & -1 & 1 & 0 & 4 & 0 \\ 0 & 0 & -1 & 1 & 0 & -1 & 0 \\ \hline 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ -1 & -2 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ -1 & -1 & 1 & 0 & 1 & 0 & 1 \end{array}\right]$$

$$= \begin{bmatrix} \dfrac{s}{s+1} & 0 & \dfrac{1}{(s+1)} & 0 \\[2ex] \dfrac{-2s(s+0.5)}{(s+1)(s^2+s+1)} & \dfrac{(s+1)(s-1)}{(s^2+s+1)} & \dfrac{-12(s+1.384)(s+0.7829)}{(s+1)(s^2+s+1)} & 0 \\[2ex] \dfrac{(s^2+1)}{(s+1)(s^2+s+1)} & \dfrac{s}{(s^2+s+1)} & \dfrac{(s+4.537)(s+1.307)(s-0.8434)}{(s+1)(s^2+s+1)} & 0 \\[2ex] \dfrac{(s+1)}{(s+1)(s^2+s+1)} & \dfrac{s^2}{(s^2+s+1)} & \dfrac{-9(s+0.5556)(s+1)}{(s+1)(s^2+s+1)} & 1 \end{bmatrix}$$

Theorem: Consider the feedback system in Figure 1 where

$$P(s) = N(s)M^{-1}(s) = \hat{M}^{-1}(s)\hat{N}(s), \quad C(s) = U(s)V^{-1}(s) = \hat{V}^{-1}(s)\hat{U}(s)$$

are coprime factorizations in RH_∞. The following are equivalent:

1. It is internally stable.
2. $\begin{bmatrix} M(s) & U(s) \\ N(s) & V(s) \end{bmatrix}$ is invertible in RH_∞.
3. $\begin{bmatrix} \hat{V}(s) & -\hat{U}(s) \\ -\hat{N}(s) & \hat{M}(s) \end{bmatrix}$ is invertible in RH_∞.
4. $\hat{V}(s)M(s) - \hat{U}(s)N(s)$ is invertible in RH_∞.
5. $\hat{M}(s)V(s) - \hat{N}(s)U(s)$ is invertible in RH_∞.

Proof: First note that

$$T_{wu}(s)^{-1} = \begin{bmatrix} I & -C(s) \\ -P(s) & I \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & -I \end{bmatrix} \begin{bmatrix} I & C(s) \\ P(s) & I \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & -I \end{bmatrix}$$

so that the system in internally stable if and only if

$$\begin{aligned} \begin{bmatrix} I & C(s) \\ P(s) & I \end{bmatrix}^{-1} &= \begin{bmatrix} I & U(s)V^{-1}(s) \\ N(s)M^{-1}(s) & I \end{bmatrix}^{-1} \\ &= \begin{bmatrix} M(s) & 0 \\ 0 & V(s) \end{bmatrix} \begin{bmatrix} M(s) & U(s) \\ N(s) & V(s) \end{bmatrix}^{-1} \end{aligned}$$

is in RH_∞. Then 1. is equivalent to 2. if we can prove that

$$\begin{bmatrix} M(s) & 0 \\ 0 & V(s) \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} M(s) & U(s) \\ N(s) & V(s) \end{bmatrix}$$

are coprime. Since $M(s)$, $N(s)$, $U(s)$ and $V(s)$ are coprime factorizations in RH_∞ there exists $\hat{X}_M(s)$, $\hat{Y}_N(s)$, $\hat{X}_U(s)$ and $\hat{Y}_V(s)$ all in RH_∞ such that

$$\begin{bmatrix} \hat{X}_M(s) & \hat{Y}_N(s) \end{bmatrix} \begin{bmatrix} M(s) \\ N(s) \end{bmatrix} = I, \qquad \begin{bmatrix} \hat{X}_U(s) & \hat{Y}_V(s) \end{bmatrix} \begin{bmatrix} U(s) \\ V(s) \end{bmatrix} = I$$

and hence

$$\begin{bmatrix} \hat{X}_M(s) & -\hat{Y}_N(s) & 0 & \hat{Y}_N(s) \\ -\hat{Y}_V(s) & \hat{X}_U(s) & \hat{Y}_V(s) & 0 \end{bmatrix} \begin{bmatrix} M(s) & 0 \\ 0 & V(s) \\ M(s) & U(s) \\ N(s) & V(s) \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}$$

proving that they are indeed right coprime. One can prove that 1. is equivalent to 3. in the same way.
Now note that 2. and 3. implies 4. and 5. since

$$\begin{aligned} &\begin{bmatrix} \hat{V}(s) & -\hat{U}(s) \\ -\hat{N}(s) & \hat{M}(s) \end{bmatrix} \begin{bmatrix} M(s) & U(s) \\ N(s) & V(s) \end{bmatrix} \\ &\qquad = \begin{bmatrix} \hat{V}(s)M(s) - \hat{U}(s)N(s) & 0 \\ 0 & \hat{M}(s)V(s) - \hat{N}(s)U(s) \end{bmatrix} \end{aligned}$$

so that the right hand side must be invertible in RH_∞. To show the converse note that

$$\begin{aligned} \begin{bmatrix} I & C(s) \\ P(s) & I \end{bmatrix}^{-1} &= \begin{bmatrix} I & \hat{V}^{-1}(s)\hat{U}(s) \\ N(s)M^{-1}(s) & I \end{bmatrix}^{-1} \\ &= \begin{bmatrix} M(s) & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \hat{V}(s)M(s) & \hat{U}(s) \\ N(s) & I \end{bmatrix}^{-1} \begin{bmatrix} \hat{V}(s) & 0 \\ 0 & I \end{bmatrix} \end{aligned}$$

from which it follows that the system is internally stable if the matrix in the midle on the expression above is invertible in RH_∞. However, note that

$$\begin{bmatrix} \hat{V}(s)M(s) & \hat{U}(s) \\ N(s) & I \end{bmatrix} = \begin{bmatrix} I & \hat{U}(s) \\ 0 & I \end{bmatrix} \begin{bmatrix} \hat{V}(s)M(s) - \hat{U}(s)N(s) & 0 \\ N(s) & I \end{bmatrix}$$

which is invertible in RH_∞ if 4. holds. The proof of sufficiency of 5. follows the same pattern.

4.4 All Stabilizing Controllers

Corollary: Let

$$P(s) = N(s)M^{-1}(s) = \hat{M}^{-1}(s)\hat{N}(s)$$

be coprime factorizations in RH_∞ and $X(s)$, $Y(s)$, $\hat{X}(s)$, and $\hat{Y}(s)$ all in RH_∞ are such that

$$\begin{bmatrix} \hat{X}(s) & -\hat{Y}(s) \\ -\hat{N}(s) & \hat{M}(s) \end{bmatrix} \begin{bmatrix} M(s) & Y(s) \\ N(s) & X(s) \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}.$$

The controller

$$C(s) = Y(s)X^{-1}(s) = \hat{X}^{-1}(s)\hat{Y}(s)$$

stabilizes the feedback system in Figure 1.
Proof: All we have to prove is stability of the closed loop system. Note that

$$\begin{bmatrix} M(s) & Y(s) \\ N(s) & X(s) \end{bmatrix}^{-1} = \begin{bmatrix} \hat{X}(s) & -\hat{Y}(s) \\ -\hat{N}(s) & \hat{M}(s) \end{bmatrix} \qquad \in RH_\infty$$

which proves that $C(s) = Y(s)X^{-1}(s)$ is a stabilizing controller. The relation

$$\begin{bmatrix} \hat{X}(s) & -\hat{Y}(s) \\ -\hat{N}(s) & \hat{M}(s) \end{bmatrix}^{-1} = \begin{bmatrix} M(s) & Y(s) \\ N(s) & X(s) \end{bmatrix} \qquad \in RH_\infty$$

proves that $C(s) = \hat{X}^{-1}(s)\hat{Y}(s)$ is also stabilizing.

Example: From the previous example pick

$$\begin{bmatrix} Y(s) \\ X(s) \end{bmatrix} = \begin{bmatrix} \dfrac{1}{(s+1)} & 0 \\ \dfrac{-12(s+1.384)(s+0.7829)}{(s+1)(s^2+s+1)} & 0 \\ \dfrac{(s+4.537)(s+1.307)(s-0.8434)}{(s+1)(s^2+s+1)} & 0 \\ \dfrac{-9(s+0.5556)(s+1)}{(s+1)(s^2+s+1)} & 1 \end{bmatrix}$$

so that

$$\begin{aligned} C(s) &= Y(s)X^{-1}(s) \\ &= \begin{bmatrix} \dfrac{s^2+s+1}{(s+4.537)(s+1.307)(s-0.8434)} & 0 \\ \dfrac{-12(s+1.384)(s+0.7829)}{(s+4.537)(s+1.307)(s-0.8434)} & 0 \end{bmatrix} \end{aligned}$$

is a stabilizing controller

Theorem: Let

$$P(s) = N(s)M^{-1}(s) = \hat{M}^{-1}(s)\hat{N}(s)$$

be coprime factorizations in RH_∞ and $X(s)$, $Y(s)$, $\hat{X}(s)$, and $\hat{Y}(s)$ all in RH_∞ are such that

$$\begin{bmatrix} \hat{X}(s) & -\hat{Y}(s) \\ -\hat{N}(s) & \hat{M}(s) \end{bmatrix} \begin{bmatrix} M(s) & Y(s) \\ N(s) & X(s) \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}.$$

All controllers that stabilize the feedback system in Figure 1 are of the form

$$\begin{aligned} C(s) &= [Y(s) - M(s)Q(s)]\,[X(s) - N(s)Q(s)]^{-1} \\ &= [\hat{X}(s) - Q(s)\hat{N}(s)]^{-1}\,[\hat{Y}(s) - Q(s)\hat{M}(s)] \end{aligned}$$

where $Q(s) \in RH_\infty$ and $\lim_{s\to\infty}[I - P(s)Q(s)]$ is nonsingular.

Proof: Start with

$$\begin{aligned} \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} &= \begin{bmatrix} I & Q(s) \\ 0 & I \end{bmatrix} \begin{bmatrix} I & -Q(s) \\ 0 & I \end{bmatrix} \\ &= \begin{bmatrix} I & Q(s) \\ 0 & I \end{bmatrix} \begin{bmatrix} \hat{X}(s) & -\hat{Y}(s) \\ -\hat{N}(s) & \hat{M}(s) \end{bmatrix} \begin{bmatrix} M(s) & Y(s) \\ N(s) & X(s) \end{bmatrix} \begin{bmatrix} I & -Q(s) \\ 0 & I \end{bmatrix} \\ &= \begin{bmatrix} \hat{X}(s) - Q(s)\hat{N}(s) & -[\hat{Y}(s) - Q(s)\hat{M}(s)] \\ -\hat{N}(s) & \hat{M}(s) \end{bmatrix} \begin{bmatrix} M(s) & Y(s) - M(s)Q(s) \\ N(s) & X(s) - N(s)Q(s) \end{bmatrix} \end{aligned}$$

The $(1,2)$ block of the above expression is

$$[\hat{X}(s) - Q(s)\hat{N}(s)][Y(s) - M(s)Q(s)] = [\hat{Y}(s) - Q(s)\hat{M}(s)][X(s) - N(s)Q(s)]$$

which multiplied by the left and the right by $[\hat{X}(s) - Q(s)\hat{N}(s)]^{-1}$ and $[X(s) - N(s)Q(s)]^{-1}$ respectively yields

$$\begin{aligned} &[Y(s) - M(s)Q(s)]\,[X(s) - N(s)Q(s)]^{-1} \\ &\qquad = [\hat{X}(s) - Q(s)\hat{N}(s)]^{-1}[\hat{Y}(s) - Q(s)\hat{M}(s)] = C(s). \end{aligned}$$

Note that the inverses exist because

$$
\begin{aligned}
X(s) - N(s)Q(s) &= \left[\begin{array}{c|c} A+BK & -L \\ \hline C+DK & I \end{array}\right] - \left[\begin{array}{c|c} A+BK & B \\ \hline C+DK & D \end{array}\right] \left[\begin{array}{c|c} A_Q & B_Q \\ \hline C_Q & D_Q \end{array}\right] \\
&= \left[\begin{array}{c|c} A+BK & -L \\ \hline C+DK & I \end{array}\right] - \left[\begin{array}{cc|c} A+BK & BC_Q & BD_Q \\ 0 & A_Q & B_Q \\ \hline C+DK & DC_Q & DD_Q \end{array}\right] \\
&= \left[\begin{array}{ccc|c} A+BK & 0 & 0 & -L \\ 0 & A+BK & BC_Q & -BD_Q \\ 0 & 0 & A_Q & -B_Q \\ \hline C+DK & C+DK & DC_Q & I-DD_Q \end{array}\right]
\end{aligned}
$$

and $I - DD_Q = \lim_{s\to\infty}[I - P(s)Q(s)]$ is nonsingular. A similar manipulation can show that $\hat{X}(s) - Q(s)\hat{M}(s)$ is also invertible.
Now to prove that $C(s)$ stabilizes the feedaback system recall that

$$
\begin{bmatrix} \hat{X}(s) - Q(s)\hat{N}(s) & -[\hat{Y}(s) - Q(s)\hat{M}(s)] \\ -\hat{N}(s) & \hat{M}(s) \end{bmatrix} \begin{bmatrix} M(s) & Y(s) - M(s)Q(s) \\ N(s) & X(s) - N(s)Q(s) \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}
$$

so that

$$
\begin{bmatrix} \hat{X}(s) - Q(s)\hat{N}(s) & -[\hat{Y}(s) - Q(s)\hat{M}(s)] \\ -\hat{N}(s) & \hat{M}(s) \end{bmatrix}^{-1} = \begin{bmatrix} M(s) & Y(s) - M(s)Q(s) \\ N(s) & X(s) - N(s)Q(s) \end{bmatrix}
$$

The feedback connection is stable because $M(s)$, $N(s)$, $X(s)$ and $Y(s)$ are in RH_∞ and

$$
Y(s) - M(s)Q(s) \qquad \text{and} \qquad X(s) - N(s)Q(s)
$$

are in RH_∞ if $Q(s)$ is in RH_∞.
We now need to prove the converse, that is, that a given stabilizing controller $C(s)$ admits a representation as a function of $Q(s) \in RH_\infty$. Let $U(s)$ and $V(s)$ be right coprime factors of the stabilizing controller $C(s) = U(s)V^{-1}(s)$. Therefore

$$
\begin{bmatrix} \hat{X}(s) & -\hat{Y}(s) \\ -\hat{N}(s) & \hat{M}(s) \end{bmatrix} \begin{bmatrix} M(s) & U(s) \\ N(s) & V(s) \end{bmatrix} = \begin{bmatrix} I & \hat{X}(s)U(s) - \hat{Y}(s)V(s) \\ 0 & -\hat{N}(s)U(s) + \hat{M}(s)V(s) \end{bmatrix}.
$$

Since the left hand side is invertible in RH_∞ (from the Theorem hypothesis and that $C(s)$ is stabilizing) the right hand side also is. Defining

$$
P(s) := [\hat{M}(s)V(s) - \hat{N}(s)U(s)], \;\; Q(s) := -[\hat{X}(s)U(s) - \hat{Y}(s)V(s)]P^{-1}(s)
$$

we have

$$\begin{bmatrix} I & \hat{X}(s)U(s) - \hat{Y}(s)V(s) \\ 0 & \hat{M}(s)V(s) - \hat{N}(s)U(s) \end{bmatrix}^{-1} = \begin{bmatrix} I & -Q(s)P(s) \\ 0 & P(s) \end{bmatrix}^{-1} = \begin{bmatrix} I & Q(s) \\ 0 & P^{-1}(s) \end{bmatrix} \in RH_\infty$$

so that $Q(s)$ is in RH_∞. One can prove that $\lim_{s\to\infty} I - P(s)Q(s)$ is nonsingular by building a realization as before. Now compute

$$\begin{aligned} \begin{bmatrix} M(s) & Y(s) \\ N(s) & X(s) \end{bmatrix} \begin{bmatrix} I & -Q(s)P(s) \\ 0 & P(s) \end{bmatrix} &= \begin{bmatrix} M(s) & Y(s) \\ N(s) & X(s) \end{bmatrix} \begin{bmatrix} \hat{X}(s) & -\hat{Y}(s) \\ -\hat{N}(s) & \hat{M}(s) \end{bmatrix} \begin{bmatrix} M(s) & U(s) \\ N(s) & V(s) \end{bmatrix} \\ &= \begin{bmatrix} M(s) & U(s) \\ N(s) & V(s) \end{bmatrix} \end{aligned}$$

which implies

$$U(s) = [Y(s) - M(s)Q(s)]P(s), \qquad V(s) = [X(s) - N(s)Q(s)]P(s)$$

so that

$$\begin{aligned} C(s) &= U(s)V^{-1}(s) \\ &= [Y(s) - M(s)Q(s)]P(s)P^{-1}(s)[X(s) - N(s)Q(s)]^{-1} \\ &= [Y(s) - M(s)Q(s)]\,[X(s) - N(s)Q(s)]^{-1} \end{aligned}$$

which is of the form given in the theorem. The other formula can be proved similarly using left coprime factors.

Example: From the previous example pick

$$\begin{bmatrix} M(s) \\ N(s) \end{bmatrix} = \begin{bmatrix} \dfrac{s}{s+1} & 0 \\ \dfrac{-2s(s+0.5)}{(s+1)(s^2+s+1)} & \dfrac{(s+1)(s-1)}{(s^2+s+1)} \\ \dfrac{(s^2+1)}{(s+1)(s^2+s+1)} & \dfrac{s}{(s^2+s+1)} \\ \dfrac{(s+1)}{(s+1)(s^2+s+1)} & \dfrac{s^2}{(s^2+s+1)} \end{bmatrix}$$

and

$$Q(s) = \begin{bmatrix} \dfrac{1}{s+3} & 0 \\ \dfrac{1}{s+2} & \dfrac{1}{s+1} \end{bmatrix}$$

so that

$$\begin{aligned} C(s) &= [Y(s) - M(s)Q(s)]\,[X(s) - N(s)Q(s)]^{-1} \\ &= \frac{1}{(s+5.026)(s+2.252)(s-1.399)(s+0.55)(s^2+3.571s+3.673)} \\ &\times \begin{bmatrix} 3(s+2)(s+0.5698)(s^2+0.4302s+1.755) & 0 \\ \begin{matrix} -13(s+2.56)(s+0.7159)(s+0.4773) \\ (s^2+3.785s+3.782) \end{matrix} & \begin{matrix} -(s+1.592)(s+2)(s+3.565) \\ (s^2+1.843s+2.82) \end{matrix} \end{bmatrix} \end{aligned}$$

is a stabilizing controller.

Lemma: Let

$$P(s) = N(s)M^{-1}(s) = \hat{M}^{-1}(s)\hat{N}(s)$$

be coprime factorizations in RH_∞ and $X(s)$, $Y(s)$, $\hat{X}(s)$, and $\hat{Y}(s)$ all in RH_∞ are such that

$$\begin{bmatrix} \hat{X}(s) & -\hat{Y}(s) \\ -\hat{N}(s) & \hat{M}(s) \end{bmatrix} \begin{bmatrix} M(s) & Y(s) \\ N(s) & X(s) \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}.$$

All controllers that stabilize the feedback system in Figure 1 are of the form

$$C(s) = C_0(s) - \hat{X}^{-1}(s)Q(s)\left[I - X^{-1}(s)N(s)Q(s)\right]^{-1} X^{-1}(s)$$

where

$$C_0(s) = Y(s)X^{-1}(s) = \hat{X}^{-1}(s)\hat{Y}(s)$$

and $Q(s) \in RH_\infty$ and $\lim_{s\to\infty}[I - P(s)Q(s)]$ is nonsingular.

Proof: From the previous theorem

$$\begin{aligned} C(s) &= [Y(s) - M(s)Q(s)]\,[X(s) - N(s)Q(s)]^{-1} \\ &= [Y(s) - M(s)Q(s)]\left\{\left[I - N(s)Q(s)X^{-1}(s)\right]X(s)\right\}^{-1} \\ &= [Y(s) - M(s)Q(s)]\,X^{-1}\left[I - N(s)Q(s)X^{-1}(s)\right]^{-1} \\ &= \left[C_0(s) - M(s)Q(s)X^{-1}\right]\left[I - N(s)Q(s)X^{-1}(s)\right]^{-1}. \end{aligned}$$

Now note that

$$\begin{aligned} (I - AB)^{-1} &= (I - AB + AB)(I - AB)^{-1} \\ &= (I - AB)(I - AB)^{-1} + AB(I - AB)^{-1} \\ &= I + AB(I - AB)^{-1} \end{aligned}$$

and because $B - BAB = (I - BA)B = B(I - AB)$ we have

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

and

$$\begin{aligned} (I - AB)^{-1} &= I + AB(I - AB)^{-1} \\ &= I + A(I - BA)^{-1}B \end{aligned}$$

Applying the above formula with $A = N(s)Q(s)$ and $B = X^{-1}(s)$ we obtain

$$\begin{aligned}
C(s) &= \left[C_0(s) - M(s)Q(s)X^{-1}\right]\left[I - N(s)Q(s)X^{-1}(s)\right]^{-1} \\
&= \left[C_0(s) - M(s)Q(s)X^{-1}\right]\left\{I + N(s)Q(s)\left[I - X^{-1}(s)N(s)Q(s)\right]^{-1}X^{-1}(s)\right\} \\
&= C_0(s) + K(s)
\end{aligned}$$

where

$$\begin{aligned}
K(s) &= C_0(s)N(s)Q(s)\left[I - X^{-1}(s)N(s)Q(s)\right]^{-1}X^{-1}(s) \\
&\qquad - M(s)Q(s)X^{-1}\left\{I + N(s)Q(s)\left[I - X^{-1}(s)N(s)Q(s)\right]^{-1}X^{-1}(s)\right\} \\
&= C_0(s)N(s)Q(s)\left[I - X^{-1}(s)N(s)Q(s)\right]^{-1}X^{-1}(s) \\
&\qquad - M(s)Q(s)X^{-1}\left[I - N(s)Q(s)X^{-1}(s)\right]^{-1}
\end{aligned}$$

Using again the fact that $B(I - AB)^{-1} = (I - BA)^{-1}B$ with $A = N(s)Q(s)$ and $B = X^{-1}(s)$

$$\begin{aligned}
K(s) &= C_0(s)N(s)Q(s)\left[I - X^{-1}(s)N(s)Q(s)\right]^{-1}X^{-1}(s) \\
&\qquad - M(s)Q(s)\left[I - X^{-1}(s)N(s)Q(s)\right]^{-1}X^{-1} \\
&= \left[C_0(s)N(s) - M(s)\right]Q(s)\left[I - X^{-1}(s)N(s)Q(s)\right]^{-1}X^{-1}(s).
\end{aligned}$$

Finally

$$\begin{aligned}
C_0(s)N(s) - M(s) &= \hat{X}^{-1}(s)\hat{Y}(s)N(s) - M(s) \\
&= \hat{X}^{-1}(s)\left[\hat{Y}(s)N(s) - \hat{X}(s)M(s)\right] \\
&= -\hat{X}^{-1}(s)
\end{aligned}$$

so that

$$K(s) = -\hat{X}^{-1}(s)Q(s)\left[I - X^{-1}(s)N(s)Q(s)\right]^{-1}X^{-1}(s).$$

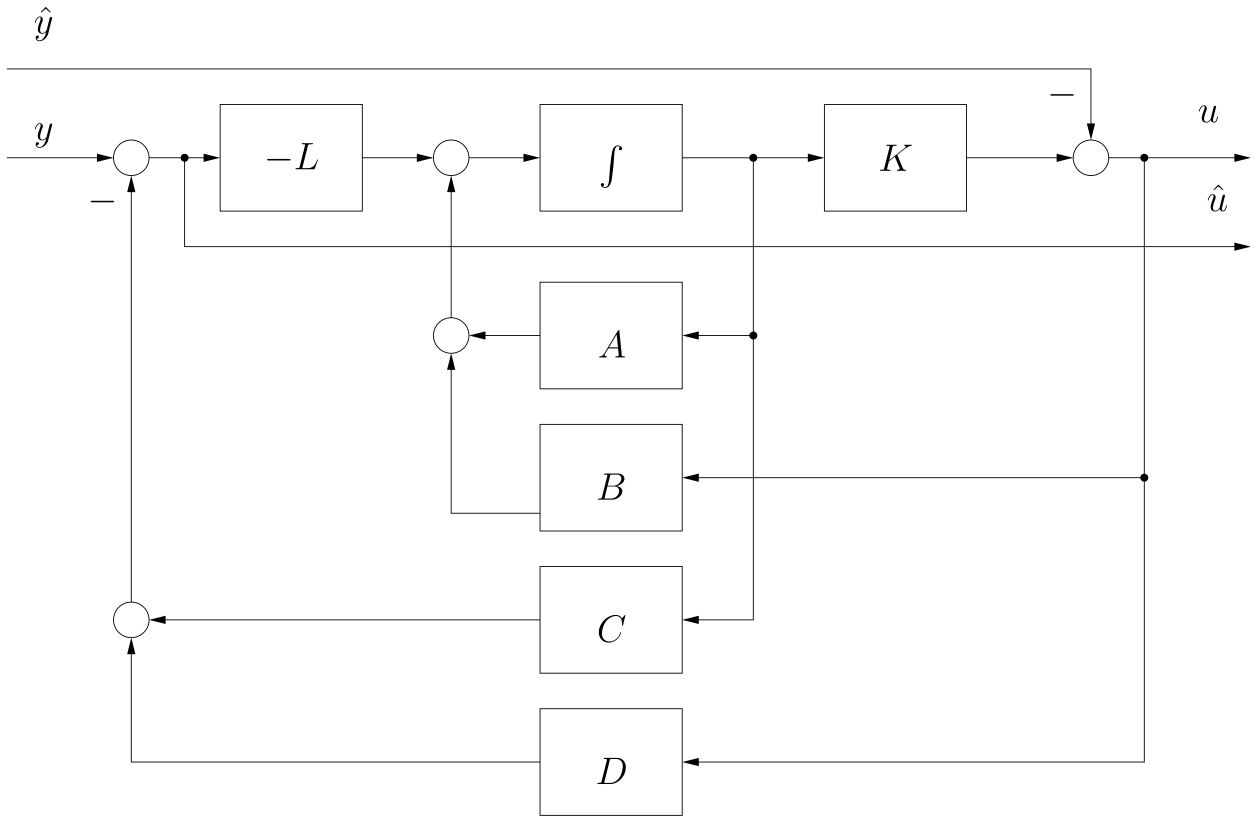

Figure 2: Observer-based controllers

Corollary: All controllers that stabilize the feedback system in Figure 1 are the result of the feedback connection of

$$\begin{pmatrix} U(s) \\ \hat{U}(s) \end{pmatrix} = R(s) \begin{pmatrix} Y(s) \\ \hat{Y}(s) \end{pmatrix} = \begin{bmatrix} C_0(s) & -\hat{X}^{-1}(s) \\ X^{-1}(s) & X^{-1}(s)N(s) \end{bmatrix} \begin{pmatrix} Y(s) \\ \hat{Y}(s) \end{pmatrix}$$

with

$$\hat{Y}(s) = Q(s)\hat{U}(s).$$

Furthermore $R(s)$ admits a realization

$$R(s) = \begin{bmatrix} C_0(s) & -\hat{X}^{-1}(s) \\ X^{-1}(s) & X^{-1}(s)N(s) \end{bmatrix} = \left[\begin{array}{c|cc} A + BK + L(C + DK) & -L & -(B + LD) \\ \hline K & 0 & -I \\ -(C + DK) & I & D \end{array}\right],$$

for which one implementation is given in Figure 2.

Proof: For the first part compute

$$\begin{aligned}\hat{Y}(s) &= Q(s)\hat{U}(s) \\ &= Q(s)X^{-1}(s)\left[Y(s) + N(s)\hat{Y}(s)\right]\end{aligned}$$

so that

$$\left[I - Q(s)X^{-1}N(s)\right]\hat{Y}(s) = Q(s)X^{-1}(s)Y(s)$$

and

$$\begin{aligned}\hat{Y}(s) &= \left[I - Q(s)X^{-1}N(s)\right]^{-1} Q(s)X^{-1}(s)Y(s) \\ &= Q(s)\left[I - X^{-1}N(s)Q(s)\right]^{-1} X^{-1}(s)Y(s).\end{aligned}$$

Hence

$$U(s) = \left\{C_0(s) - \hat{X}^{-1}(s)Q(s)\left[I - X^{-1}N(s)Q(s)\right]^{-1} X^{-1}(s)\right\} Y(s) = C(s)Y(s).$$

For the second part start by constructing

$$X^{-1}(s) = \left[\begin{array}{c|c} A+BK & -L \\ \hline C+DK & I \end{array}\right]^{-1} = \left[\begin{array}{c|c} A+BK+L(C+DK) & L \\ \hline C+DK & I \end{array}\right]$$

and

$$\hat{X}^{-1}(s) = \left[\begin{array}{c|c} A+LC & -(B+LD) \\ \hline K & I \end{array}\right]^{-1} = \left[\begin{array}{c|c} A+LC+(B+LD)K & B+LD \\ \hline K & I \end{array}\right].$$

Then

$$\begin{aligned}X^{-1}(s)N(s) &= \left[\begin{array}{c|c} A+BK+L(C+DK) & L \\ \hline C+DK & I \end{array}\right]\left[\begin{array}{c|c} A+BK & B \\ \hline C+DK & D \end{array}\right] \\
&= \left[\begin{array}{cc|c} A+BK & 0 & B \\ L(C+DK) & A+BK+L(C+DK) & LD \\ \hline C+DK & C+DK & D \end{array}\right] \\
&= \left[\begin{array}{c|c} \begin{bmatrix} I & -I \\ 0 & I \end{bmatrix}\begin{bmatrix} A+BK+L(C+DK) & 0 \\ L(C+DK) & A+BK \end{bmatrix}\begin{bmatrix} I & I \\ 0 & I \end{bmatrix} & \begin{bmatrix} I & -I \\ 0 & I \end{bmatrix}\begin{bmatrix} B+LD \\ LD \end{bmatrix} \\ \hline \begin{bmatrix} C+DK & 0 \end{bmatrix}\begin{bmatrix} I & I \\ 0 & I \end{bmatrix} & D \end{array}\right] \\
&= \left[\begin{array}{cc|c} A+BK+L(C+DK) & 0 & B+LD \\ L(C+DK) & A+BK & LD \\ \hline C+DK & 0 & D \end{array}\right] \\
&= \left[\begin{array}{c|c} A+BK+L(C+DK) & B+LD \\ \hline C+DK & D \end{array}\right]\end{aligned}$$

Similarly

$$
\begin{aligned}
C_0(s) &= Y(s)X^{-1}(s) \\
&= \left[\begin{array}{c|c} A + BK + L(C + DK) & L \\ \hline -K & 0 \end{array}\right].
\end{aligned}
$$

Therefore, since all submatrices share the same poles

$$
\begin{aligned}
R(s) = \begin{bmatrix} C_0(s) & -\hat{X}^{-1}(s) \\ X^{-1}(s) & X^{-1}(s)N(s) \end{bmatrix} &= \left[\begin{array}{cc|c} A + BK + L(C + DK) & L & B + LD \\ \hline -K & 0 & -I \\ C + DK & I & D \end{array}\right] \\
&= \left[\begin{array}{c|cc} A + BK + L(C + DK) & -L & -(B + LD) \\ \hline K & 0 & -I \\ -(C + DK) & I & D \end{array}\right].
\end{aligned}
$$

4.5 Matlab demo

```
>> % MAE 284
>> % Coprime factorizations
>> %
>> % MAE 284 - Robust and Multivariable Control
>> % Mauricio de Oliveira
>> % April 2006
>> %
>>
>> % Define state space matrices for plant
>> A = [
0  1  0
1  0  0
0  0  0
];
>>
>> B = [
0  1
1  0
1  0
];
>>
>> C = [
1  0  1
0  1  1
];
>>
>> D = [
0  0
0  1
];
>>
>> % construct plant
>> p = ss(A, B, C, D);
>>
>> % plant transfer function
>> zpk(p)

Zero/pole/gain from input 1 to output...
      (s+1.618) (s-0.618)
 #1:  -------------------
         s (s+1) (s-1)

      2 (s+0.7071) (s-0.7071)
 #2:  -----------------------
           s (s+1) (s-1)

Zero/pole/gain from input 2 to output...
           s
 #1:  -----------
      (s+1) (s-1)

          s^2
```

```
 #2:  -----------
      (s+1) (s-1)

>>
>> % define stabilizing gains
>> K = [
 0  0 -1
-1 -2 0
];
>>
>> L = [
-4 0
-4 0
 1 0
];
>>
>> % check stability
>> eig(A + B * K)
ans =
  -0.5000 + 0.8660i
  -0.5000 - 0.8660i
  -1.0000
>> eig(A + L * C)
ans =
  -1.0000 + 0.0000i
  -1.0000 - 0.0000i
  -1.0000
>>
>> % construct mnxy
>> mnxy = ss(A + B * K, [B -L], [K; C + D * K], [eye(2) zeros(2); D eye(2)])

a =
       x1  x2  x3
   x1  -1  -1   0
   x2   1   0  -1
   x3   0   0  -1


b =
       u1  u2  u3  u4
   x1   0   1   4  -0
   x2   1   0   4  -0
   x3   1   0  -1  -0


c =
       x1  x2  x3
   y1   0   0  -1
   y2  -1  -2   0
   y3   1   0   1
   y4  -1  -1   1


d =
```

```
       u1  u2  u3  u4
   y1   1   0   0   0
   y2   0   1   0   0
   y3   0   0   1   0
   y4   0   1   0   1

Continuous-time model.
>>
>> % construct mnxyhat
>> mnxyhat = ss(A + L*C, [-(B + L * D) L], [K; C], [eye(2) zeros(2); -D eye(2)])

a =
       x1  x2  x3
   x1  -4   1  -4
   x2  -3   0  -4
   x3   1   0   1


b =
       u1  u2  u3  u4
   x1  -0  -1  -4   0
   x2  -1  -0  -4   0
   x3  -1  -0   1   0


c =
       x1  x2  x3
   y1   0   0  -1
   y2  -1  -2   0
   y3   1   0   1
   y4   0   1   1


d =
       u1  u2  u3  u4
   y1   1   0   0   0
   y2   0   1   0   0
   y3  -0  -0   1   0
   y4  -0  -1   0   1

Continuous-time model.
>>
>> % check Bezout equation
>> minreal(mnxyhat * mnxy)
3 states removed.

a =
            x1        x2        x3
   x1  -0.2022   -1.095   0.3118
   x2    1.006   -1.103   0.3381
   x3   0.9026  -0.1481  -0.6943


b =
```

```
             u1       u2       u3       u4
   x1   -1.352  -0.5702   -7.393        0
   x2   -1.317   0.9995    3.319        0
   x3   0.6632   0.8222   0.5685        0


c =
                x1          x2          x3
   y1   5.172e-14   4.635e-14  -9.862e-14
   y2  -3.464e-13  -4.058e-13    7.55e-13
   y3   6.495e-14   9.483e-14  -1.615e-13
   y4   6.323e-14    8.57e-14  -1.489e-13


d =
       u1  u2  u3  u4
   y1   1   0   0   0
   y2   0   1   0   0
   y3   0   0   1   0
   y4   0   0   0   1

Continuous-time model.
>>
>> % obtain transfer function for mnxy
>> zpk(mnxy)

Zero/pole/gain from input 1 to output...
        s
 #1:  -----
      (s+1)

          -2 s (s+0.5)
 #2:  --------------------
      (s+1) (s^2  + s + 1)

           (s^2  + 1)
 #3:  --------------------
      (s+1) (s^2  + s + 1)

             (s+1)
 #4:  --------------------
      (s+1) (s^2  + s + 1)

Zero/pole/gain from input 2 to output...
 #1:  0

       (s+1) (s-1)
 #2:  -------------
      (s^2  + s + 1)

            s
 #3:  -------------
      (s^2  + s + 1)
```

```
           s^2
 #4:  --------------
      (s^2  + s + 1)

Zero/pole/gain from input 3 to output...
        1
 #1:  -----
      (s+1)

      -12 (s+1.384) (s+0.7829)
 #2:  ------------------------
        (s+1) (s^2  + s + 1)

      (s+4.537) (s+1.307) (s-0.8434)
 #3:  ------------------------------
           (s+1) (s^2  + s + 1)

      -9 (s+0.5556) (s+1)
 #4:  --------------------
      (s+1) (s^2  + s + 1)

Zero/pole/gain from input 4 to output...
 #1:  0

 #2:  0

 #3:  0

 #4:  1

>>
>> % extract x and y
>> y = mnxy(1:2, 3:4);
>> x = mnxy(3:4, 3:4);
>>
>> % compute stabilizing controller
>> c = y * inv(x)

a =
       x1  x2  x3  x4  x5  x6
   x1  -1  -1   0  -4   0  -4
   x2   1   0  -1  -4   0  -4
   x3   0   0  -1   1   0   1
   x4   0   0   0  -5  -1  -4
   x5   0   0   0  -3   0  -5
   x6   0   0   0   1   0   0


b =
       u1  u2
   x1   4   0
   x2   4   0
   x3  -1   0
   x4   4   0
```

```
   x5   4   0
   x6  -1   0


c =
       x1  x2  x3  x4  x5  x6
   y1   0   0  -1   0   0   0
   y2  -1  -2   0   0   0   0


d =
       u1  u2
   y1   0   0
   y2   0   0

Continuous-time model.
>>
>> % compute controller transfer function
>> zpk(minreal(c))
3 states removed.

Zero/pole/gain from input 1 to output...
              (s^2  + s + 1)
 #1:  ------------------------------
      (s+4.537) (s+1.307) (s-0.8434)

         -12 (s+1.384) (s+0.7829)
 #2:  ------------------------------
      (s+4.537) (s+1.307) (s-0.8434)

Zero/pole/gain from input 2 to output...
 #1:  0

 #2:  0

>>
>> % check closed loop stability
>> pole(inv([eye(2) -p; -c eye(2)]))
ans =
  -0.5000 + 0.8660i
  -0.5000 - 0.8660i
  -0.5000 + 0.8660i
  -0.5000 - 0.8660i
  -1.0001 + 0.0001i
  -1.0001 - 0.0001i
  -0.9999 + 0.0001i
  -0.9999 - 0.0001i
  -1.0000
>>
>> % extract x and y
>> m = mnxy(1:2, 1:2);
>> n = mnxy(3:4, 1:2);
>>
>> % define q
```

```
>> q = [tf(1,[1 3]) 0; tf(1,[1 2]) tf(1, [1 1])]

Transfer function from input 1 to output...
        1
 #1:  -----
      s + 3

        1
 #2:  -----
      s + 2

Transfer function from input 2 to output...
 #1:  0

        1
 #2:  -----
      s + 1

>>
>> % compute stabilizing controller
>> c = (y - m * q) * inv(x - n * q)

a =
         x1   x2   x3   x4   x5   x6   x7   x8   x9  x10  x11  x12  x13
   x1    -1   -1    0    0    0    0    0    0    0   -4    0   -4    4
   x2     1    0   -1    0    0    0    0    0    0   -4    0   -4    4
   x3     0    0   -1    0    0    0    0    0    0    1    0    1   -1
   x4     0    0    0   -1   -1    0    0    1    1    0    0    0    0
   x5     0    0    0    1    0   -1    1    0    0    0    0    0    0
   x6     0    0    0    0    0   -1    1    0    0    0    0    0    0
   x7     0    0    0    0    0    0   -3    0    0   -1    0   -1    1
   x8     0    0    0    0    0    0    0   -2    0   -1    0   -1    1
   x9     0    0    0    0    0    0    0    0   -1    1    1   -1   -1
   x10    0    0    0    0    0    0    0    0    0   -5   -1   -4    4
   x11    0    0    0    0    0    0    0    0    0   -3    0   -5    4
   x12    0    0    0    0    0    0    0    0    0    1    0    0   -1
   x13    0    0    0    0    0    0    0    0    0    0    0    0   -1
   x14    0    0    0    0    0    0    0    0    0    0    0    0    1
   x15    0    0    0    0    0    0    0    0    0    0    0    0    0
   x16    0    0    0    0    0    0    0    0    0   -1    0   -1    1
   x17    0    0    0    0    0    0    0    0    0   -1    0   -1    1
   x18    0    0    0    0    0    0    0    0    0    1    1   -1   -1

        x14  x15  x16  x17  x18
   x1     0    4    0    0    0
   x2     0    4    0    0    0
   x3     0   -1    0    0    0
   x4     0    0    0    0    0
   x5     0    0    0    0    0
   x6     0    0    0    0    0
   x7     0    1    0    0    0
   x8     0    1    0    0    0
   x9    -1    1    0    1    1
   x10    0    4    0    0    0
```

```
   x11    0    4    0    0    0
   x12    0   -1    0    0    0
   x13   -1    0    0    1    1
   x14    0   -1    1    0    0
   x15    0   -1    1    0    0
   x16    0    1   -3    0    0
   x17    0    1    0   -2    0
   x18   -1    1    0    1    0


b =
        u1  u2
   x1    4   0
   x2    4   0
   x3   -1   0
   x4    0   0
   x5    0   0
   x6    0   0
   x7    1   0
   x8    1   0
   x9    0   1
   x10   4   0
   x11   4   0
   x12  -1   0
   x13   0   0
   x14   0   0
   x15   0   0
   x16   1   0
   x17   1   0
   x18   0   1


c =
        x1   x2   x3   x4   x5   x6   x7   x8   x9  x10  x11  x12  x13
   y1    0    0   -1   -0   -0    1   -1   -0   -0    0    0    0    0
   y2   -1   -2    0    1    2   -0   -0   -1   -1    0    0    0    0

       x14  x15  x16  x17  x18
   y1    0    0    0    0    0
   y2    0    0    0    0    0


d =
       u1  u2
   y1   0   0
   y2   0   0

Continuous-time model.
>>
>> % compute controller transfer function
>> zpk(minreal(c))
12 states removed.

Zero/pole/gain from input 1 to output...
```

```
               3 (s+2) (s+0.5698) (s^2  + 0.4302s + 1.755)
 #1:  --------------------------------------------------------------
      (s+5.026) (s+2.252) (s-1.399) (s+0.55) (s^2  + 3.571s + 3.673)

        -13 (s+2.56) (s+0.7159) (s+0.4773) (s^2  + 3.785s + 3.782)
 #2:  --------------------------------------------------------------
      (s+5.026) (s+2.252) (s-1.399) (s+0.55) (s^2  + 3.571s + 3.673)

Zero/pole/gain from input 2 to output...
              4.9559e-15 s (s+2) (s^2  + 16.43s + 6.054e14)
 #1:  --------------------------------------------------------------
      (s+5.026) (s+2.252) (s-1.399) (s+0.55) (s^2  + 3.571s + 3.673)

            - (s+1.592) (s+2) (s+3.565) (s^2  + 1.843s + 2.82)
 #2:  --------------------------------------------------------------
      (s+5.026) (s+2.252) (s-1.399) (s+0.55) (s^2  + 3.571s + 3.673)

>>
>> % check closed loop stability
>> pole(inv([eye(2) -p; -c eye(2)]))
ans =
  -3.0000 + 0.0000i
  -3.0000 - 0.0000i
  -2.0000
  -2.0000
  -1.0004
  -1.0000 + 0.0004i
  -1.0000 - 0.0004i
  -0.9996
  -0.5000 + 0.8660i
  -0.5000 - 0.8660i
  -0.5000 + 0.8660i
  -0.5000 - 0.8660i
  -0.5000 + 0.8660i
  -0.5000 - 0.8660i
  -0.5000 + 0.8660i
  -0.5000 - 0.8660i
  -1.0000 + 0.0000i
  -1.0000 - 0.0000i
  -1.0000
  -1.0000
  -1.0000
>>
>> % find stabilizing gains through Riccati
>> R = B * B';
>> Q = eye(3);
>>
>> % find eigenvalues of Hamiltonean
>> eig([A R; -Q -A'])
ans =
        0 + 0.0000i
        0 - 0.0000i
  -0.0000 + 1.2720i
  -0.0000 - 1.2720i
```

```
  -0.7862
   0.7862
>>
>> % there exists eigenvalues on the imaginary axis, flip sign
>> R = -B * B';
>>
>> % find eigenvalues of Hamiltonean
>> [v, d] = eig([A R; -Q -A'])
v =
    0.7071    0.7071   -0.4082   -0.4082    0.4082    0.4082
    0.5000   -0.5000    0.6606   -0.6606   -0.2523    0.2523
         0         0    0.4082   -0.4082    0.4082   -0.4082
   -0.5000    0.5000   -0.0000    0.0000   -0.0000   -0.0000
         0         0    0.4082    0.4082   -0.4082   -0.4082
         0         0    0.2523    0.2523    0.6606    0.6606
d =
    1.4142         0         0         0         0         0
         0   -1.4142         0         0         0         0
         0         0   -1.6180         0         0         0
         0         0         0    1.6180         0         0
         0         0         0         0   -0.6180         0
         0         0         0         0         0    0.6180
>>
>> % go find eigenvectors of Hamiltonean
>> v1 = v(:, [2 3 5])
v1 =
    0.7071   -0.4082    0.4082
   -0.5000    0.6606   -0.2523
         0    0.4082    0.4082
    0.5000   -0.0000   -0.0000
         0    0.4082   -0.4082
         0    0.2523    0.6606
>> x1 = v1(1:3,:)
x1 =
    0.7071   -0.4082    0.4082
   -0.5000    0.6606   -0.2523
         0    0.4082    0.4082
>> x2 = v1(4:6,:)
x2 =
    0.5000   -0.0000   -0.0000
         0    0.4082   -0.4082
         0    0.2523    0.6606
>>
>> % compute are solution
>> X = x2 / x1
X =
    1.9239    1.7208   -0.8604
    1.7208    2.4335   -1.2168
   -0.8604   -1.2168    1.7264
>>
>> % stabilizing gain is
>> K = - B' * X
K =
   -0.8604   -1.2168   -0.5097
```

```
   -1.9239   -1.7208    0.8604
>>
>> eig(A + B * K)
ans =
   -0.6180
   -1.4142
   -1.6180
>>
>> % alternative solution
>> help are

 ARE  Algebraic Riccati Equation solution.

     X = ARE(A, B, C) returns the stablizing solution (if it
     exists) to the continuous-time Riccati equation:

            A'*X + X*A - X*B*X + C = 0

     assuming B is symmetric and nonnegative definite and C is
     symmetric.

     See also  RIC, CARE, DARE.

>>
>> X = are(A, B * B', Q)
X =
    1.9239    1.7208   -0.8604
    1.7208    2.4335   -1.2168
   -0.8604   -1.2168    1.7264
>>
>> % stabilizing filter gain
>> help are

 ARE  Algebraic Riccati Equation solution.

     X = ARE(A, B, C) returns the stablizing solution (if it
     exists) to the continuous-time Riccati equation:

            A'*X + X*A - X*B*X + C = 0

     assuming B is symmetric and nonnegative definite and C is
     symmetric.

     See also  RIC, CARE, DARE.

>>
>> Y = are(A', C' * C, Q)
Y =
    6.2756    5.8614   -3.5549
    5.8614    6.2756   -3.5549
   -3.5549   -3.5549    2.8478
>>
>> % stabilizing gain is
>> L = (- C * Y)'
```

```
L =
   -2.7208   -2.3066
   -2.3066   -2.7208
    0.7071    0.7071
>>
>> eig(A + L * C)
ans =
   -0.7654
   -1.8478
   -1.4142
>>
>> % construct mnxy
>> mnxy = ss(A + B * K, [B -L], [K; C + D * K], [eye(2) zeros(2); D eye(2)])

a =
            x1        x2        x3
   x1   -1.924  -0.7208   0.8604
   x2   0.1396   -1.217  -0.5097
   x3  -0.8604   -1.217  -0.5097


b =
            u1        u2        u3        u4
   x1        0         1     2.721     2.307
   x2        1         0     2.307     2.721
   x3        1         0   -0.7071   -0.7071


c =
            x1        x2        x3
   y1  -0.8604   -1.217  -0.5097
   y2   -1.924   -1.721   0.8604
   y3        1        0        1
   y4   -1.924  -0.7208     1.86


d =
       u1  u2  u3  u4
   y1   1   0   0   0
   y2   0   1   0   0
   y3   0   0   1   0
   y4   0   1   0   1

Continuous-time model.
>>
>> % construct mnxyhat
>> mnxyhat = ss(A + L*C, [-(B + L * D) L], [K; C], [eye(2) zeros(2); -D eye(2)])

a =
           x1       x2       x3
   x1  -2.721  -1.307  -5.027
   x2  -1.307  -2.721  -5.027
   x3  0.7071  0.7071   1.414
```

```
b =
             u1       u2       u3       u4
   x1        -0    1.307   -2.721   -2.307
   x2        -1    2.721   -2.307   -2.721
   x3        -1  -0.7071   0.7071   0.7071


c =
             x1       x2       x3
   y1  -0.8604   -1.217  -0.5097
   y2   -1.924   -1.721   0.8604
   y3        1        0        1
   y4        0        1        1


d =
       u1  u2  u3  u4
   y1   1   0   0   0
   y2   0   1   0   0
   y3  -0  -0   1   0
   y4  -0  -1   0   1

Continuous-time model.
>>
>> % compute stabilizing controller
>> c = y * inv(x)

a =
       x1  x2  x3  x4  x5  x6
   x1  -1  -1   0  -4   0  -4
   x2   1   0  -1  -4   0  -4
   x3   0   0  -1   1   0   1
   x4   0   0   0  -5  -1  -4
   x5   0   0   0  -3   0  -5
   x6   0   0   0   1   0   0


b =
       u1  u2
   x1   4   0
   x2   4   0
   x3  -1   0
   x4   4   0
   x5   4   0
   x6  -1   0


c =
       x1  x2  x3  x4  x5  x6
   y1   0   0  -1   0   0   0
   y2  -1  -2   0   0   0   0
```

```
d =
       u1  u2
   y1   0   0
   y2   0   0

Continuous-time model.
>>
>> % compute controller transfer function
>> zpk(minreal(c))
3 states removed.

Zero/pole/gain from input 1 to output...
              (s^2  + s + 1)
 #1:  ------------------------------
      (s+4.537) (s+1.307) (s-0.8434)

         -12 (s+1.384) (s+0.7829)
 #2:  ------------------------------
      (s+4.537) (s+1.307) (s-0.8434)

Zero/pole/gain from input 2 to output...
 #1:  0

 #2:  0

>>
>> % check closed loop stability
>> pole(inv([eye(2) -p; -c eye(2)]))
ans =
  -0.5000 + 0.8660i
  -0.5000 - 0.8660i
  -0.5000 + 0.8660i
  -0.5000 - 0.8660i
  -1.0001 + 0.0001i
  -1.0001 - 0.0001i
  -0.9999 + 0.0001i
  -0.9999 - 0.0001i
  -1.0000
>>
>> diary off
```

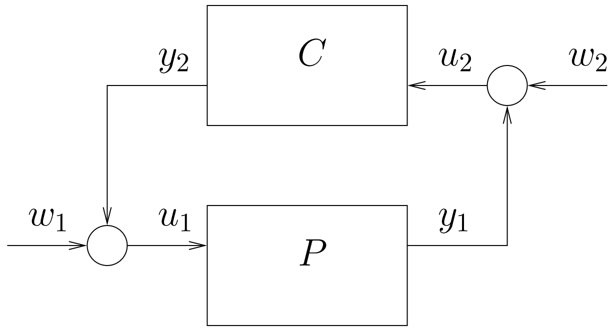

Figure 3: Feedback connection

Lemma: Let

$$H(s) = \begin{bmatrix} H_{zw}(s) & H_{zu}(s) \\ H_{yw}(s) & H_{yu}(s) \end{bmatrix} = \left[\begin{array}{c|cc} A & B_w & B_u \\ \hline C_z & D_{zw} & D_{zu} \\ C_y & D_{yw} & D_{yu} \end{array}\right]$$

where (A, B_u) is stabilizable and (A, C_y) is detectable. A controller $C(s)$ stabilizes the feedback system in Figure 3 if and only if it stabilizes $H_{yu}(s)$

Proof: Necessity is obvious. Sufficiency comes from stabilizability of (A, B_u) and detectability of (A, C_y).

Theorem: Let

$$H(s) = \begin{bmatrix} H_{zw}(s) & H_{zu}(s) \\ H_{yw}(s) & H_{yu}(s) \end{bmatrix} = \left[\begin{array}{c|cc} A & B_w & B_u \\ \hline C_z & D_{zw} & D_{zu} \\ C_y & D_{yw} & D_{yu} \end{array}\right]$$

where (A, B_u) is stabilizable and (A, C_y) is detectable. All stabilizing controllers $C(s)$ are parametrized as the feedback connection of

$$\begin{aligned} R(s) &= \begin{bmatrix} C_0(s) & -\hat{X}^{-1}(s) \\ X^{-1}(s) & X^{-1}(s)N(s) \end{bmatrix} \\ &= \left[\begin{array}{c|cc} A + B_uK + L(C_y + D_{yu}K) & -L & -(B_u + LD_{yu}) \\ \hline K & 0 & -I \\ -(C_y + D_{yu}K) & I & D_{yu} \end{array}\right] \end{aligned}$$

with $Q(s) \in RH_\infty$ and $\lim_{s\to\infty}[I - H_{yu}(s)Q(s)]$ is nonsingular. Furthermore the closed loop transfer function from w to z is

$$G(s) = T_{11}(s) + T_{12}(s)Q(s)T_{21}(s)$$

where

$$T(s) = \begin{bmatrix} T_{11}(s) & T_{12}(s) \\ T_{21}(s) & 0 \end{bmatrix} = \left[\begin{array}{cc|cc} A + B_uK & -B_uK & B_w & -B_u \\ 0 & A + LC_y & B_w + LD_{yw} & 0 \\ \hline C_z + D_{zu}K & -D_{zu}K & D_{zw} & -D_{zu} \\ 0 & C_y & D_{yw} & 0 \end{array}\right]$$

with $T_{ij} \in RH_\infty$, $i, j = \{1, 2\}$.

Proof: Because (A, B_u) is stabilizable and (A, C_y) is detectable, all stabilizing controllers are the ones that stabilize $H_{yu}(s)$, which are given in the theorem. The feedback connection of

$$\begin{pmatrix} \dot{x} \\ \hline z \\ y \end{pmatrix} = \left[\begin{array}{c|cc} A & B_w & B_u \\ \hline C_z & D_{zw} & D_{zu} \\ C_y & D_{yw} & D_{yu} \end{array}\right] \begin{pmatrix} x \\ \hline w \\ u \end{pmatrix}$$

with

$$\begin{pmatrix} \dot{\hat{x}} \\ \hline u \\ \hat{u} \end{pmatrix} = \left[\begin{array}{c|cc} \hat{A} & \hat{B}_y & \hat{B}_{\hat{y}} \\ \hline \hat{C}_u & 0 & \hat{D}_{u\hat{y}} \\ \hat{C}_{\hat{u}} & \hat{D}_{\hat{u}y} & \hat{D}_{\hat{u}\hat{y}} \end{array}\right] \begin{pmatrix} \hat{x} \\ \hline y \\ \hat{y} \end{pmatrix}$$

is given by

$$
\begin{pmatrix} \dot{x} \\ \dot{\hat{x}} \\ \hline z \\ \hat{u} \end{pmatrix} = \left[\begin{array}{cc|cc} A & B_u \hat{C}_u & B_w & B_u \hat{D}_{u\hat{y}} \\ \hat{B}_y C_y & \hat{A} + \hat{B}_y D_{yu} \hat{C}_u & \hat{B}_y D_{yw} & \hat{B}_{\hat{y}} + \hat{B}_y D_{yu} \hat{D}_{u\hat{y}} \\ \hline C_z & D_{zu} \hat{C}_u & D_{zw} & D_{zu} \hat{D}_{u\hat{y}} \\ \hat{D}_{\hat{u}y} C_y & \hat{C}_{\hat{u}} + \hat{D}_{\hat{u}y} D_{yu} \hat{C}_u & \hat{D}_{\hat{u}y} D_{yw} & \hat{D}_{\hat{u}\hat{y}} + \hat{D}_{\hat{u}y} D_{yu} \hat{D}_{u\hat{y}} \end{array}\right] \begin{pmatrix} x \\ \hat{x} \\ \hline w \\ \hat{y} \end{pmatrix}
$$

Substituting

$$
\left[\begin{array}{c|cc} \hat{A} & \hat{B}_y & \hat{B}_{\hat{y}} \\ \hline \hat{C}_u & 0 & \hat{D}_{u\hat{y}} \\ \hat{C}_{\hat{u}} & \hat{D}_{\hat{u}y} & \hat{D}_{\hat{u}\hat{y}} \end{array}\right] = \left[\begin{array}{c|cc} A + B_u K + L(C_y + D_{yu} K) & -L & -(B_u + L D_{yu}) \\ \hline K & 0 & -I \\ -(C_y + D_{yu} K) & I & D_{yu} \end{array}\right]
$$

we obtain

$$
\begin{aligned}
T(s) &= \left[\begin{array}{cc|cc} A & B_u K & B_w & -B_u \\ -LC_y & A + B_u K + LC_y & -LD_{yw} & -B_u \\ \hline C_z & D_{zu} K & D_{zw} & -D_{zu} \\ C_y & -C_y & D_{yw} & 0 \end{array}\right] \\
&= \left[\begin{array}{c|c} \begin{bmatrix} I & 0 \\ I & -I \end{bmatrix} \begin{bmatrix} A & B_u K \\ -LC_y & A + B_u K + LC_y \end{bmatrix} \begin{bmatrix} I & 0 \\ I & -I \end{bmatrix} & \begin{bmatrix} I & 0 \\ I & -I \end{bmatrix} \begin{bmatrix} B_w & -B_u \\ -LD_{yw} & -B_u \end{bmatrix} \\ \hline \begin{bmatrix} C_z & D_{zu} K \\ C_y & -C_y \end{bmatrix} \begin{bmatrix} I & 0 \\ I & -I \end{bmatrix} & \begin{matrix} D_{zw} & -D_{zu} \\ D_{yw} & 0 \end{matrix} \end{array}\right] \\
&= \left[\begin{array}{cc|cc} A + B_u K & -B_u K & B_w & -B_u \\ 0 & A + LC_y & B_w + LD_{yw} & 0 \\ \hline C_z + D_{zu} K & -D_{zu} K & D_{zw} & -D_{zu} \\ 0 & C_y & D_{yw} & 0 \end{array}\right].
\end{aligned}
$$

The closed loop transfer function $G(s)$ is the feedback connection of $T(s)$ with $Q(s)$, that is,

$$
\begin{aligned}
Z(s) &= T_{11}(s) W(s) + T_{12}(s) \hat{Y}(s) \\
\hat{U}(s) &= T_{21}(s) W(s)
\end{aligned}
$$

with

$$
\hat{Y}(s) = Q(s) \hat{U}(s)
$$

which produces

$$
Z(s) = G(s) W(s) = \left[T_{11}(s) + T_{12}(s) Q(s) T_{21}(s) \right] W(s).
$$

4.6 Linear Fractional Transformation

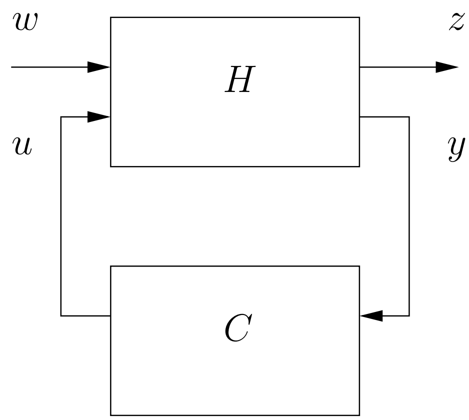

Figure 4: Feedback connection

Let

$$H(s) = \begin{bmatrix} H_{zw}(s) & H_{zu}(s) \\ H_{yw}(s) & H_{yu}(s) \end{bmatrix}$$

and the control

$$U(s) = C(s)Y(s).$$

Then

$$\begin{aligned} H_{yw}(s)W(s) &= Y(s) - H_{yu}(s)U(s) \\ &= \left[I - H_{yu}(s)C(s)\right] Y(s) \end{aligned}$$

and if the feeback loop is well-posed, i.e.,

$$\lim_{s \to \infty} I - H_{yu}(s)C(s) \qquad \text{is invertible}$$

then

$$Y(s) = \left[I - H_{yu}(s)C(s)\right]^{-1} H_{yw}(s)W(s)$$

and

$$\begin{aligned} Z(s) &= H_{zw}(s)W(s) + H_{zu}C(s)Y(s) \\ &= H_{zw}(s)W(s) + H_{zu}C(s)\left[I - H_{yu}(s)C(s)\right]^{-1} H_{yw}(s)W(s) \end{aligned}$$

That is

$$Z(s) = \mathcal{F}_l\left(H(s), C(s)\right) W(s)$$

where

$$\mathcal{F}_l\left(H(s), C(s)\right) := H_{zw}(s) + H_{zu}(s)C(s)\left[I - H_{yu}(s)C(s)\right]^{-1} H_{yw}(s)$$

is called a (Lower) Linear Fractional Transformation.

Simmilarly

$$Z(s) = \mathcal{F}_u\left(H(s), C(s)\right) W(s)$$

where

$$\mathcal{F}_u\left(H(s), C(s)\right) := H_{zw}(s) + H_{zu}(s)\left[I - C(s)H_{yu}(s)\right]^{-1} C(s)H_{yw}(s)$$

is called an (Upper) Linear Fractional Transformation.

For us, LFT is for convinience of notation!

A final remark on the parametrization of all stabilizing controllers. If $H(s) \in RH_\infty$ then

$$\begin{aligned} Z(s) &= \mathcal{F}_l\left(H(s), C(s)\right) W(s) \\ &= \left[H_{zw}(s) + H_{zu} \underbrace{C(s)\left[I - H_{yu}(s)C(s)\right]^{-1}}_{Q(s)} H_{yw}(s)\right] W(s) \end{aligned}$$

or

$$\begin{aligned} Z(s) &= \mathcal{F}_u\left(H(s), C(s)\right) W(s) \\ &= \left[H_{zw}(s) + H_{zu}(s) \underbrace{\left[I - C(s)H_{yu}(s)\right]^{-1} C(s)}_{Q(s)} H_{yw}(s)\right] W(s) \end{aligned}$$

and

$$Q(s) = C(s)\left[I - H_{yu}(s)C(s)\right]^{-1} = \left[I - C(s)H_{yu}(s)\right]^{-1} C(s) \in RH_\infty$$

(Recall the previously studied case $H_{yu}(s) \in RH_\infty$?)