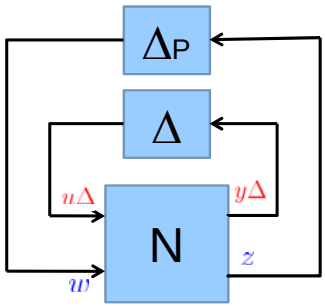
Robust Controller Synthesis

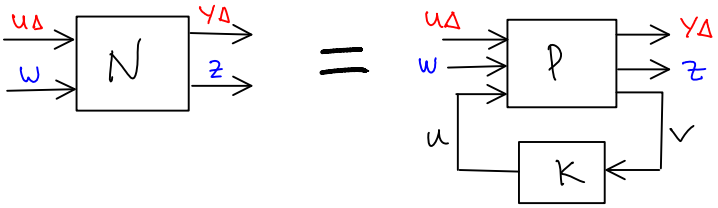
Requirement for Robust Performance (RP): N is internally stable (NS) and



has to be robustly stable for any allowed Δ, $\|\Delta\|_\infty \le 1$ and unstructured Δ_P, $\|\Delta_P\|_\infty \le 1$.

From robust stability analysis, we know that this is equivalent to

$$\mu(N(j\omega)) < 1 \quad, \forall \omega$$



Therefore, in designing a robust controller, we want to achieve
* N is stable
* $\mu(N(j\omega))$ is minimal (or at least less than 1)

Define: $\hat{\Delta} = \begin{bmatrix} \Delta & 0 \\ 0 & \Delta_P \end{bmatrix}$, $\mathcal{D}$ is the set of all matrices D such that $D\hat{\Delta} = \hat{\Delta} D$

For each ω, $\mu(N)$ can be upperbounded by

$$\mu(N) \le \min_{D \in \mathcal{D}} \bar{\sigma}_{max}(D N D^{-1})$$, which is a convex optimization problem

Generally, for complex uncertainty with three or fewer blocks, this upper bound is tight.

DK Iteration (8.12)

Generally, we are interested in designing a controller K according to the following minimization:

$$\min_{K,\, D \in \mathcal{D}} \quad \bar{\sigma}_{max}\left(D\, N(K)\, D^{-1}\right)$$

In practice, rather than solving this optimization for both D and K simultaneously we perform an iteration:

Step 0: Fix a $D(\omega) \in \mathcal{D}$, say $D(\omega) = I$

Step 1: Obtain K by solving

$$\min_{K} \quad \left\| D\, N(K)\, D^{-1} \right\|_\infty \qquad \cdots\cdots (*)$$

Step 2: Using K from step 1, for every ω find $D(\omega)$ from

$$\min_{D(\omega) \in \mathcal{D}} \quad \bar{\sigma}_{max}\left(D(\omega)\, N(j\omega)\, D^{-1}(\omega)\right) \quad \cdots (**)$$

Step 3: Fit the magnitude of $D(\omega)$ to a stable and minimum phase transfer function $D(s)$, repeat step 1.

Remarks:

- Step 1 is essentially an H_∞ synthesis with D as extra weight
- For each ω, step 2 is a convex optimization problem
- The order of the controller in Step 1 depends on the orders of the plant, weights, and D. For better performance, $D(s)$ is usually high order, which leads to high order controller.

Model Reduction (Ch 11)

General idea: Suppose that we are given a very complex plant model, $G(s)$. How can we come up with a simpler model $G_a(s)$ that is still a good representation of the plant?

Complexity: The complexity of a linear proper model $G(s)$ is typically measured as the minimum number of states required to represent it. This number is called the McMillan degree, which is also the number of system poles.

Reasoning: Computational: the number of states corresponds to the complexity of numerical simulation of the system. Also, many controller synthesis technique use the state-space representation

Physical: The Mc Millan degree is roughly related to the number of energy bearing variables in the system

Example: $G(s) = \frac{1}{s}\begin{bmatrix} 1 & 0 \\ 1 & 2 \end{bmatrix}$

has pole polynomial $\phi(s) = s^2$, thus this is a second order system

A minimal realization:

$\dot{x}_1 = u_1 \qquad \dot{x}_2 = 2u_2 \qquad \rightarrow A = 0 \; ; B = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix} \; ; C = \begin{bmatrix} 1 & 0 \\ 1 & 2 \end{bmatrix}$

$y_1 = x_1 \qquad y_2 = x_1 + x_2$

Compare with $G(s) = \frac{1}{s}\begin{bmatrix} 1 & 0 \\ 1 & s \end{bmatrix}$, whose pole polynomial is $\phi(s) = s$

A minimal representation:

$\left.\begin{aligned} \dot{x}_1 &= u_1 \\ y_1 &= x_1 \\ y_2 &= x_1 + u_2 \end{aligned}\right\} \quad A = 0 \; ; B = 1 \; ; C = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \; ; D = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

Quality of approximation is measured by the distance between G and G_a. In this course, we use H_∞ norm.

Thus, the general idea can be concretized as: Given a plant model with high order, how can we come up with a lower order / reduced order model that is close to the original model (in H_∞ sense)?