Robust Control 2018

Richard Pates and Bo Bernhardsson

Department of Automatic Control LTH,
Lund University

Plan of attack:

Today's topic: Synthesise controllers to meet $\mathcal{H}_\infty$ based robust stability and performance claims.

- General approach.
- $\mathcal{H}_\infty$ loopshaping
- The ν-gap metric.

The basic idea

1. Formulate design specifications.
2. Put into standard form.
3. Compute and check solution.
4. Iterate

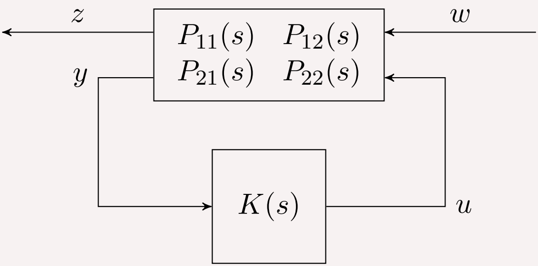
Formulate design specifications

Write performance and robustness specifications as an $\mathcal{H}_\infty$-norm requirement. e.g.

- Robustness to multiplicative uncertainty.
- Good disturbance rejection over control bandwidth.
- Step response tracking.
- ...

Put into standard form

Stack up requirements and write as a big LFT:



- w, z specification inputs and outputs.
- Pull out controller.

Compute and check solution

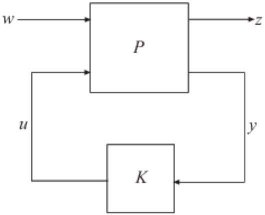
```
[C,CL,GAM,INFO] = hinfsyn(P,NMEAS,NCON)
```

`hinfsyn` computes a stabilizing H_∞ optimal `lti/ss` controller K for a partitioned `lti` plant P.

$$P = \begin{bmatrix} A & B_1 & B_2 \\ C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{bmatrix}$$

The controller, K, stabilizes the P and has the same number of states as P. The system P is partitioned where inputs to B_1 are the disturbances, inputs to B_2 are the control inputs, output of C_1 are the errors to be kept small, and outputs of C_2 are the output measurements provided to the controller. B_2 has column size (NCON) and C_2 has row size (NMEAS). The optional KEY and VALUE inputs determine tolerance, solution method and so forth.

The closed-loop system is returned in CL. This closed-loop system is given by `CL = lft(P,K)` as in the following diagram.



The achieved H_∞ cost γ is returned as GAM. The `struct` array INFO contains additional information about the design.

You get what you asked for

Consider again the problem of tracking a step input when

$$P(s) = \frac{1}{s+1}.$$

You get what you asked for: Step 1

We showed before that we would like:

$$\frac{1}{s}S(s) \approx 0$$

$\implies$ minimise $\|\frac{1}{s+\epsilon}S(s)\|_\infty$!

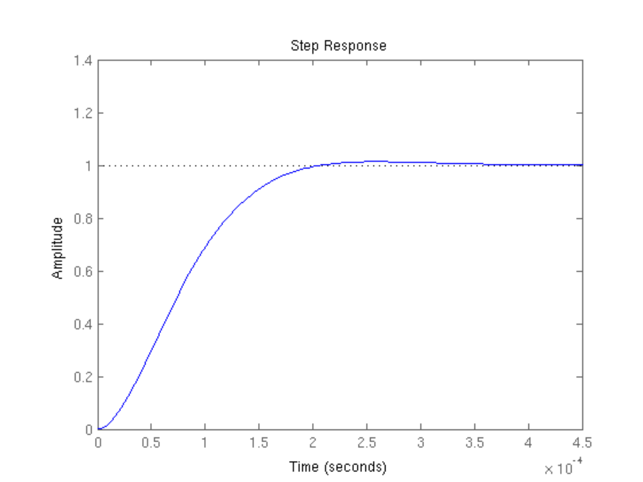
You get what you asked for: Step 2

We showed before that

$$W(s)S(s) = \mathcal{F}_l\left(\begin{bmatrix} W(s) & W(s)P(s) \\ -I & P(s) \end{bmatrix}, C(s)\right)$$

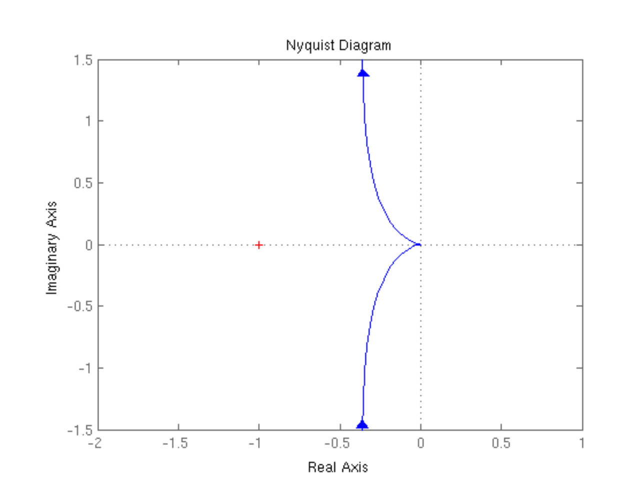
You get what you asked for: Step 3

Try $\epsilon = .01$ and apply `hinfsyn`.



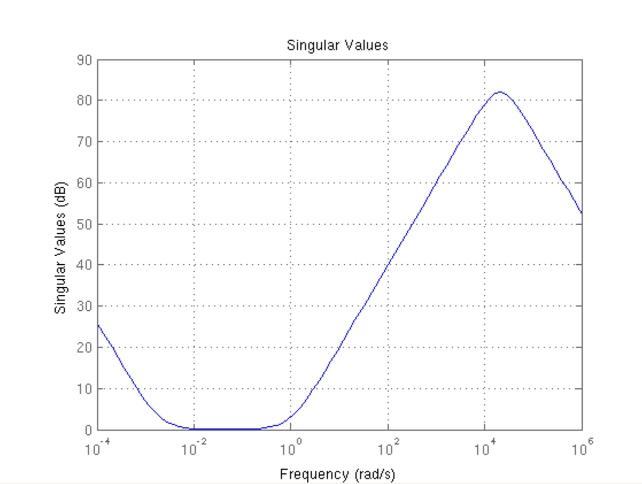
You get what you asked for: Step 3

Try $\epsilon = .01$ and apply `hinfsyn`.



You get what you asked for: Step 3

Try $\epsilon = .01$ and apply `hinfsyn`.



You get what you asked for

Was this a pathological example?

Exercise

You get what you asked for

What happens if we try to maximise robustness to multiplicative uncertainty?

Exercise

Need sensible specifications

Classical control:

- $\overline{\sigma}(S(j\omega)) \leq 2$ (gain and phase margins).
- $\overline{\sigma}(S(j\omega)) \leq \epsilon, \forall \omega < \omega_b$ (performance).
- $\overline{\sigma}(P(j\omega)C(j\omega)) \leq \omega_c^2/\omega^2, \forall \omega > \omega_b$ (robustness).

Need sensible specifications

Bounds on closed loop transfer functions

- $\overline{\sigma}(S(j\omega))$ small over desired control bandwidth, and never too large.
- $\overline{\sigma}(T(j\omega))$ small at high frequencies (robustness).
- Check all the closed loop transfer functions, step responses... $\mathcal{H}_\infty$ norm is just a number!

Matlab

```
[C,CL,GAM,INFO]=mixsyn(P,W1,W2,W3) or

mixsyn H-infinity mixed-sensitivity synthesis method for
robust control design. Controller C stabilizes plant P
and minimizes the H-infinity cost function
              ||   W1*S ||
              || W2*C*S ||
              ||   W3*T ||Hinf
where
    S := inv(I+P*C)         % sensitivity
    T := I-S = P*C/(I+P*C)  % complementary sensitivity
    W1, W2 and  W3 are stable LTI 'weights'
```

Why penalise CS?

Exercise

Glad-Ljung Ex. 10.1: Step 1

Motor control

$$P(s) = \frac{20}{s(s+1)}.$$

Design controller with integral action and specified control bandwidth.

Glad-Ljung Ex. 10.1: Step 2

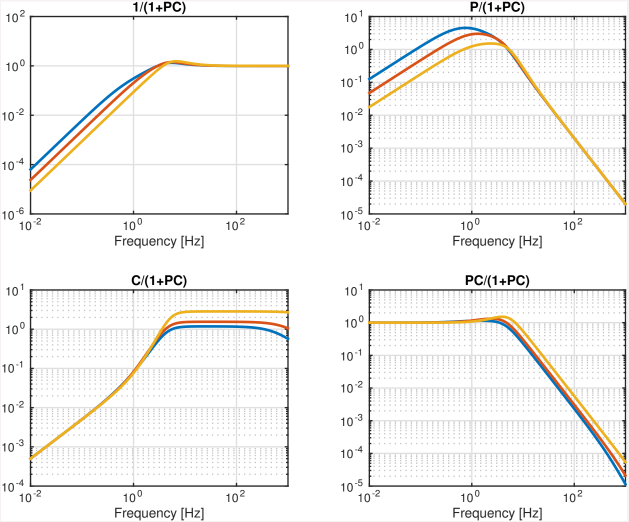
$$\text{Minimize } H_\infty \text{ norm of } \begin{bmatrix} W_1(I+PC)^{-1} \\ W_2C(I+PC)^{-1} \\ W_3PC(I+PC)^{-1} \end{bmatrix}, \quad \text{with} \quad \begin{array}{l} W_1 = \frac{k}{s^2} \\ W_2 = 1 \\ W_3 = 1 \end{array}$$

Increasing k gives higher bandwidth at the cost of larger controller gain

Shape of W_1 will enforce integral action. Try $k = 1, 5, 30$.

Needed to change to $P(s) = \dfrac{20}{(s+\epsilon)(s+1)}$ and $W_1 = \dfrac{k}{(s+\epsilon)^2}$

Glad-Ljung Ex. 10.1: Step 3



Further iteration

- Does the controller really need high gain beyond 10^2 rad/s?
- We didn't enforce rolloff in T.
- Check step responses...

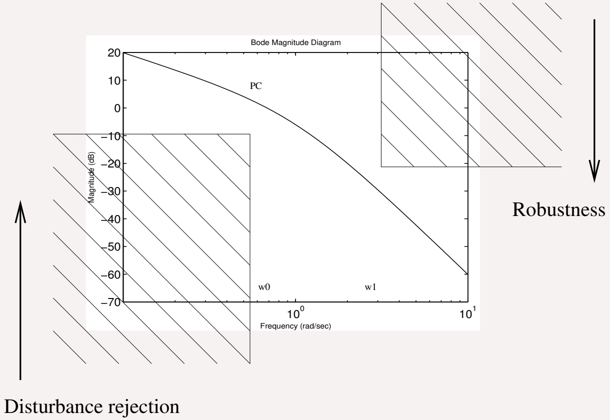
Loopshaping

Classical control:

- $\overline{\sigma}(S(j\omega)) \leq 2$.
- $\overline{\sigma}(S(j\omega)) \leq \epsilon, \forall \omega < \omega_b$.
- $\overline{\sigma}(P(j\omega)C(j\omega)) \leq \omega_c^2/\omega^2, \forall \omega > \omega_b$.

Loopshaping

Formulate as specifications on $L(s) = P(s)C(s)$:



$\mathcal{H}_\infty$-Loopshaping

1. Formulate design specifications.
2. Design suitable loopshape (ignoring slope at crossover).
3. Weight plant to approximate desired loopshape.
4. Compute an optimal controller for weighted gang of four stability margin
5. Iterate

$\mathcal{H}_\infty$-Loopshaping: Step 1

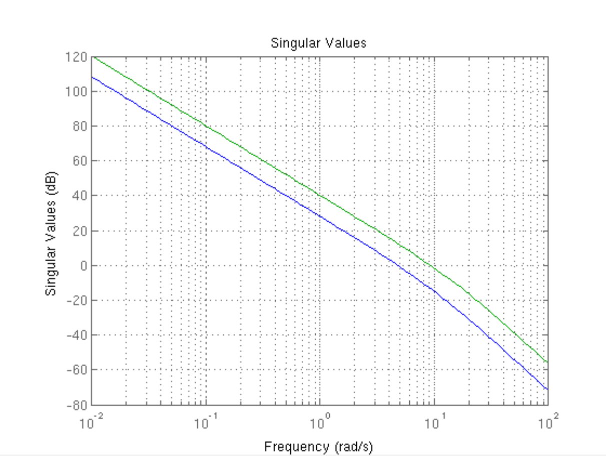
Motor control again...

$$P(s) = \frac{20}{s(s+1)}.$$

Design controller with integral action and specified control bandwidth.

$\mathcal{H}_\infty$-Loopshaping: Step 2

Desired loopshape:



$\mathcal{H}_\infty$-Loopshaping: Step 3

Can choose

$$W = \frac{\omega_c(s+1)}{20s(s/T+1)} \Longrightarrow WP = \frac{\omega_c}{s^2(s/T+1)}.$$

Can also use Matlab function `loopsyn`.

$\mathcal{H}_\infty$-Loopshaping: Step 4

Define $\bar{P} = WP$, and solve

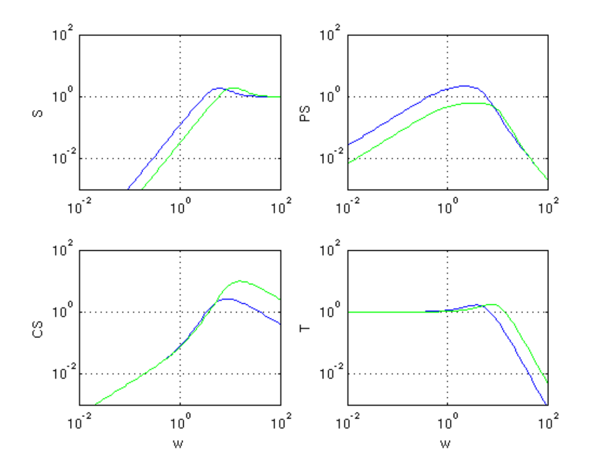
$$b_{\text{opt}}(\bar{P}) = \min_{\bar{C}} b_{\bar{P},\bar{C}}$$

Optimal controller is then $C = \bar{C}W$.

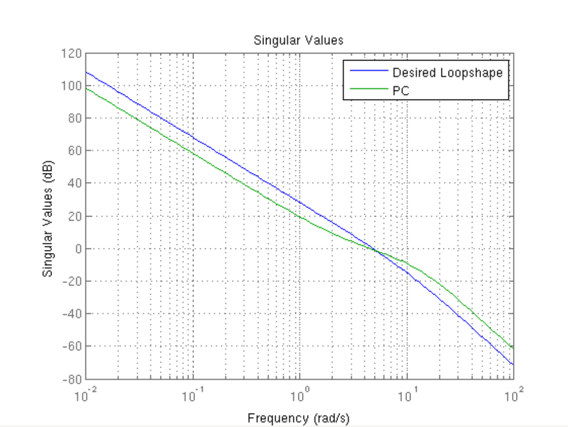
Use Matlab function `C=ncfsyn(P,W)`.

$\mathcal{H}_\infty$-Loopshaping: Step 5

Check GoF



$\mathcal{H}_\infty$-Loopshaping: Justification

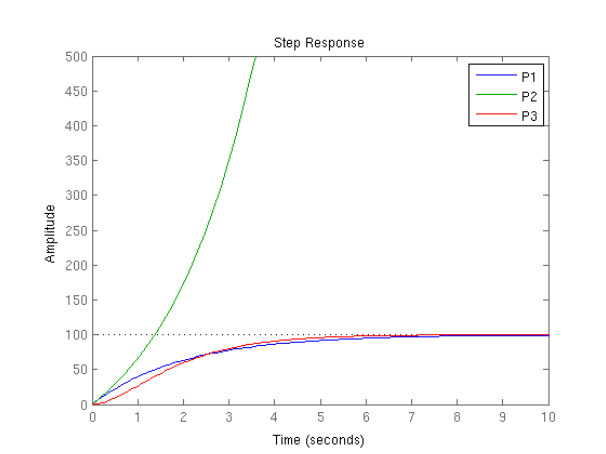


If $b_{\text{opt}} \approx 0.3$, then will approximately match desired loopshape, and have good stability margins.

Can be made rigorous: Exercise

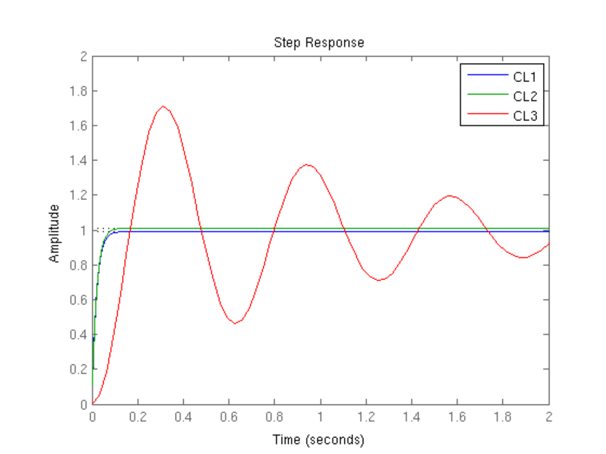
Robust performance and the ν-gap metric

Open and closed loop can be very different:



Robust performance and the ν-gap metric

Open and closed loop can be very different:



Robust performance and the ν-gap metric

Consider

$$\delta'(P_1, P_2) = \sin \sup_C \left| \arcsin b_{P_1,C} - \arcsin b_{P_2,C} \right|.$$

Robust performance and the ν-gap metric

Consider

$$\delta'(P_1, P_2) = \sin \sup_C |\arcsin b_{P_1,C} - \arcsin b_{P_2,C}|.$$

Maximum possible difference between closed loop performance of two systems.

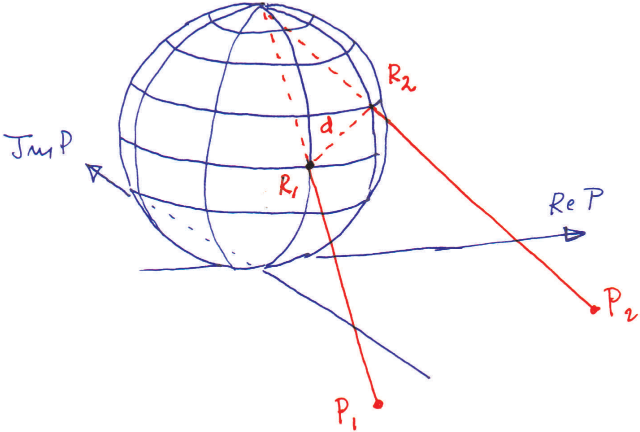
Robust performance and the ν-gap metric

Introduce the ν-gap metric:

$$\delta_\nu(P_1, P_2) = \begin{cases} \|(I+P_2P_2^*)^{-\frac{1}{2}}(P_1-P_2)(I+P_1^*P_1)^{-\frac{1}{2}}\|_{\mathcal{L}_\infty} \\ \quad \text{if } \det(I + P_2^* P_1) \neq 0 \text{ on } j\mathbb{R} \text{ and} \\ \quad \operatorname{wno}\det(I+P_2^*P_1) + \eta(P_1) = \overline{\eta}(P_2), \\ 1 \quad \text{otherwise} \end{cases}$$

where $\overline{\eta}$ (η) is the number of closed (open) RHP poles and wno is winding number.

Geometric Interpretation



Robust performance and the ν-gap metric

It turns out that:

$$\delta'(P_1, P_2) = \min\{\delta_\nu(P_1, P_2), \max\{b_{\mathsf{opt}}(P_1), b_{\mathsf{opt}}(P_2)\}\}$$

Robust performance and the ν-gap metric

It turns out that:

$$\delta'(P_1, P_2) = \min\{\delta_\nu(P_1, P_2), \max\{b_{\mathsf{opt}}(P_1), b_{\mathsf{opt}}(P_2)\}\}$$

ν-gap measures distance between systems from the perspective of closed loop performance.

Robust performance and the ν-gap metric

ν-gap also gives the following robust performance condition: Let

$$P_\Delta = \{P : \delta_\nu(P, P_1) \leq \beta\}.$$

Suppose that $b_{P_1,C} = \arcsin\gamma + \arcsin\beta$. Then

$$\min_{P \in P_\Delta} b_{P,C} = \gamma.$$

Robust performance and the ν-gap metric

The philosophy:

- $b_{P,C}$ is a good measure of performance.
- ν-gap balls give the largest uncertainty balls w.r.t. degradation of $b_{P,C}$.
- Cover actual model uncertainty with smallest possible ν-gap ball.

Exercise