Robust Control Design and Analysis

NASA Workshop: Winter 2025

Lecture 7: Robustness Analysis Examples Using the Structured Singular Value (μ)

Key Takeaways

This lecture presents three examples to illustrate robustness analysis using the structured singular value:

- Input margins for a MIMO system with crossfeed: We study loop-at-a-time margins, multi-loop margins, and robustness to unstructured uncertainty.
- System with uncertain, lightly damped modes: We study the effect of uncertainty in the real parameters associated with the modal frequency and damping ratio.
- MIMO robustness analysis (Matlab Demo): This example includes parametric and dynamic uncertainty. Both robust stability and performance is analyzed.

Along the way we also describe how robust performance can be assessed as a related robust stability problem.

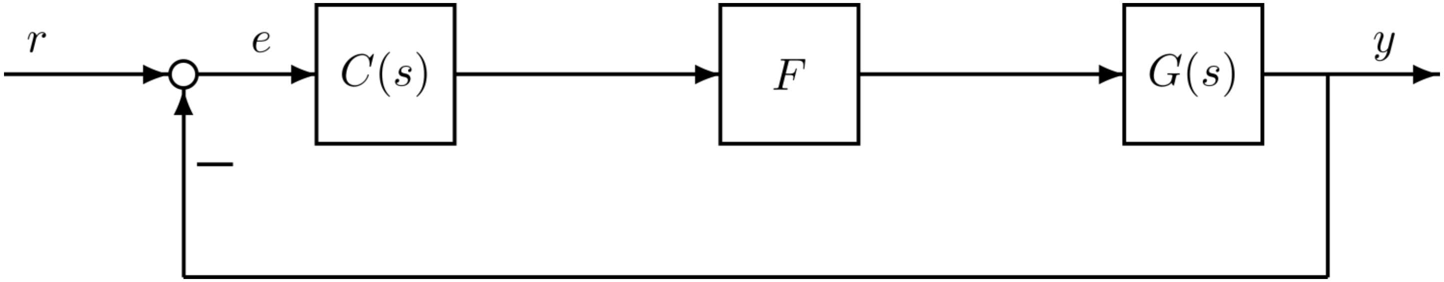
Example: Input Margins for MIMO Crossfeed System

Summary

We'll revisit various stability margins in the context of our new structured singular value results.

- Loop-at-a-time
- Unstructured uncertainty
- Multi-loop disk margins

Each of these cases corresponds to a different assumption on the uncertainty.



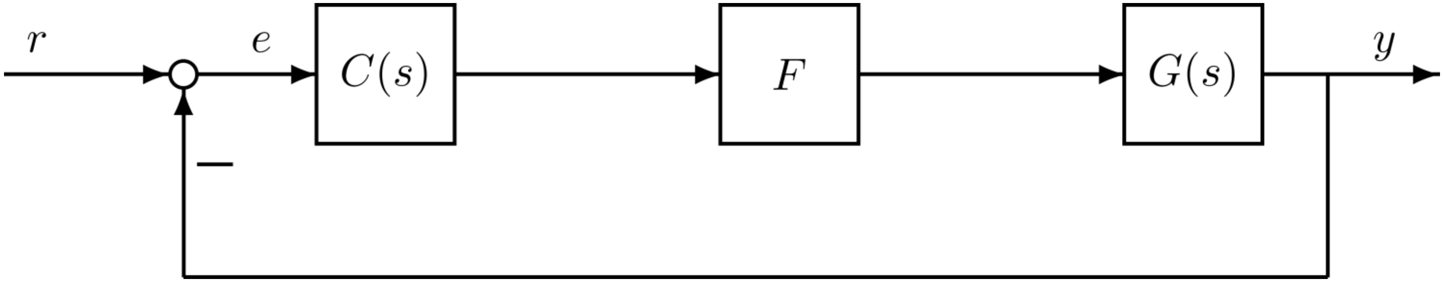
Example: Crossfeed System

Plant: $G(s) = \frac{1}{s} \cdot \begin{bmatrix} 1 & 10 \\ 0 & 1 \end{bmatrix}$

Controller: $C(s) = I_2$

Note that the plant has a large cross feed from the second input to the first output (but no feed in the other direction).

This example is taken from a multivariable control textbook by Prof. James Freudenberg at Michigan.



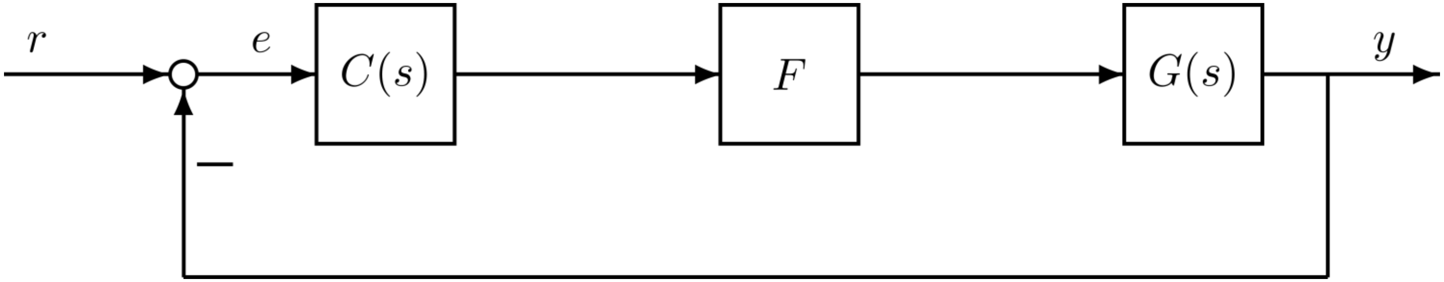
Loop-at-a-Time Margins

Margins on Input 1: $F = \begin{bmatrix} f_1 & 0 \\ 0 & 1 \end{bmatrix}$

- Gain Margin = $[0, \infty)$
- Phase Margin = 90° at 1.0 rad/s
- Delay Margin = 1.57s at 1.0 rad/s
- Symmetric Disk Margin = 2 ⇔ stable for any $Re(f_1) \geq 0$.

Margins on Input 2: $F = \begin{bmatrix} 1 & 0 \\ 0 & f_2 \end{bmatrix}$

- Same.



Loop-at-a-Time Margins

Margins on Input 1: $F = \begin{bmatrix} f_1 & 0 \\ 0 & 1 \end{bmatrix}$

- Gain Margin = $[0, \infty)$
- Phase Margin = 90° at 1.0 rad/s
- Delay Margin = 1.57s at 1.0 rad/s
- Symmetric Disk Margin = 2 ⇔ stable for any $Re(f_1) \geq 0$.

Margins on Input 2: $F = \begin{bmatrix} 1 & 0 \\ 0 & f_2 \end{bmatrix}$

- Same.

System is very robust to perturbations in individual channels at the plant input.

Code: Loop-at-a-Time Margins

```
% Plant and Controller
% State-space models are preferred for calculations)
s = tf('s’); G = ss( [1 10; 0 1]/s );
C = eye(2);

% Loop-at-a-time classical/disk margins at plant input
LI = C*G;
AM = loopmargin(LI);
AM(1)    % AM(2) is similar

DM = diskmargin(LI);
DM(1)    % DM(2) is similar
```

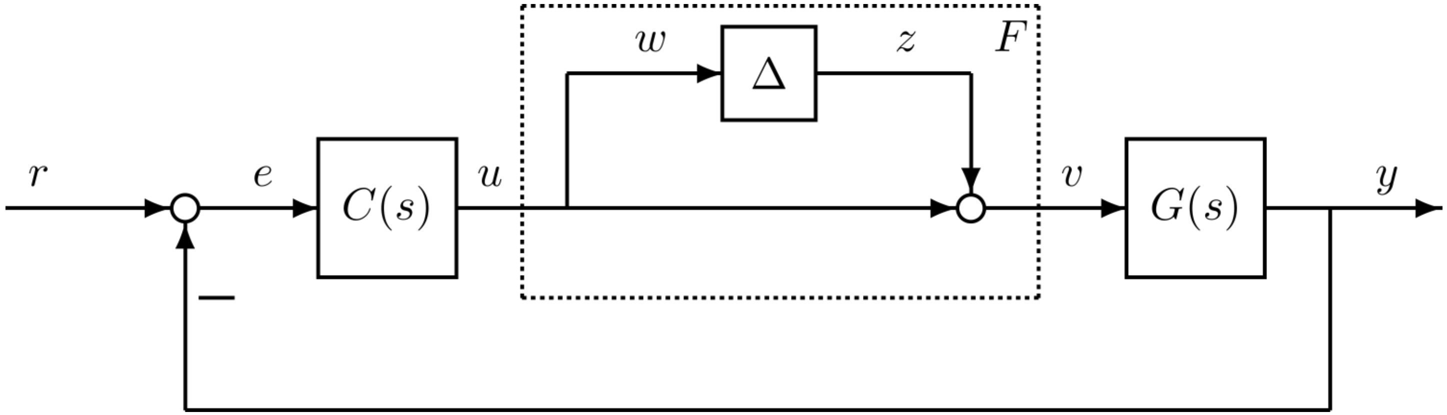
Margins With Unstructured Uncertainty

Unstructured Input Uncertainty: $F = I + \Delta$

$\Delta(s)$ is "full", i.e. all entries can be non-zero. This can model cross-coupling in the uncertainty. In the 2-by-2 case,

$$v = (I + \Delta)u \quad \Rightarrow \quad \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} 1 + \Delta_{11} & \Delta_{12} \\ \Delta_{21} & 1 + \Delta_{22} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

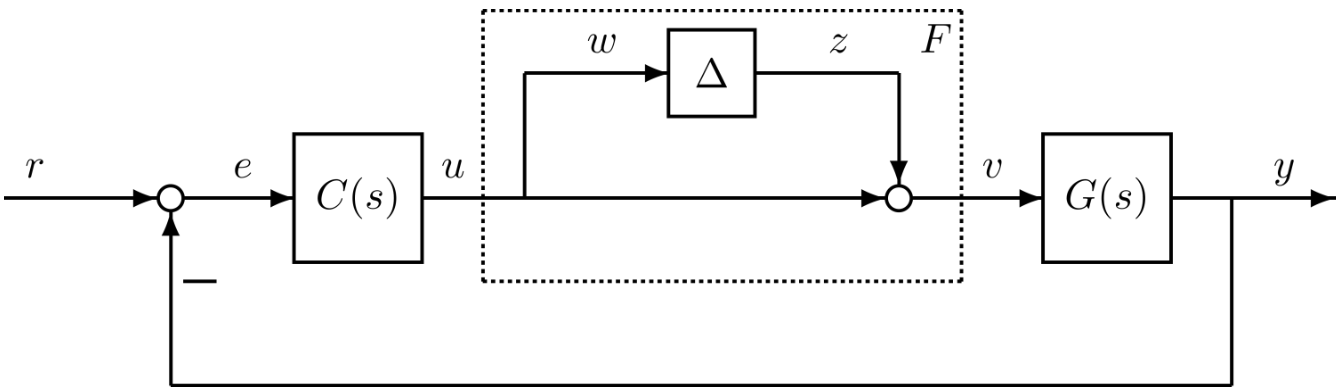
Control command u_1 can affect plant input v_2 and vice versa.

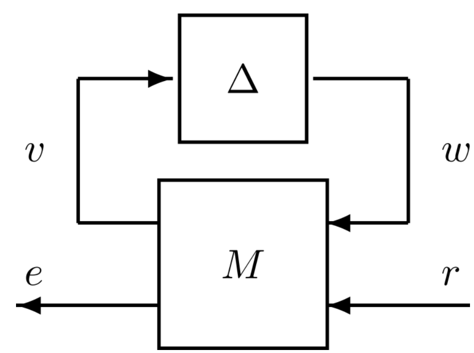


Margins With Unstructured Uncertainty

LFT Representation: $F_U(M, \Delta)$

$$M := \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} = \begin{bmatrix} -C(I+GC)^{-1}G & C(I+GC)^{-1} \\ -(I+GC)^{-1}G & (I+GC)^{-1} \end{bmatrix}$$



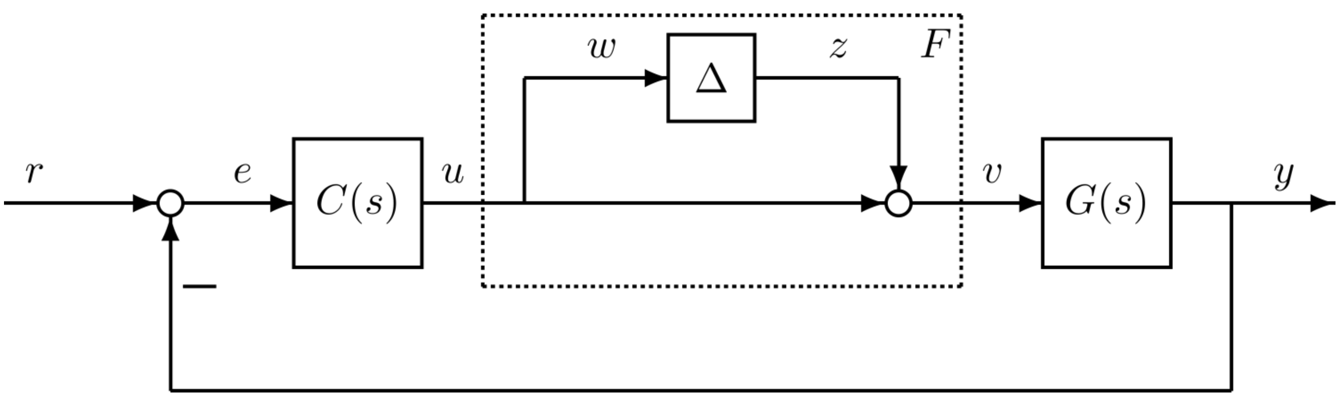


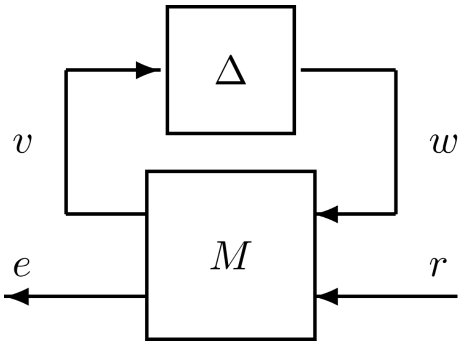
Margins With Unstructured Uncertainty

LFT Representation: $F_U(M, \Delta)$

$$M := \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} = \begin{bmatrix} -C(I+GC)^{-1}G & C(I+GC)^{-1} \\ -(I+GC)^{-1}G & (I+GC)^{-1} \end{bmatrix}$$

Robust Stability: $F_U(M, \Delta)$ is stable for all Δ with $||\Delta||_\infty \leq m$ if and only if $\sup_\omega \mu_\Delta\big(M_{11}(j\omega)\big) < \frac{1}{m}$.





Margins With Unstructured Uncertainty

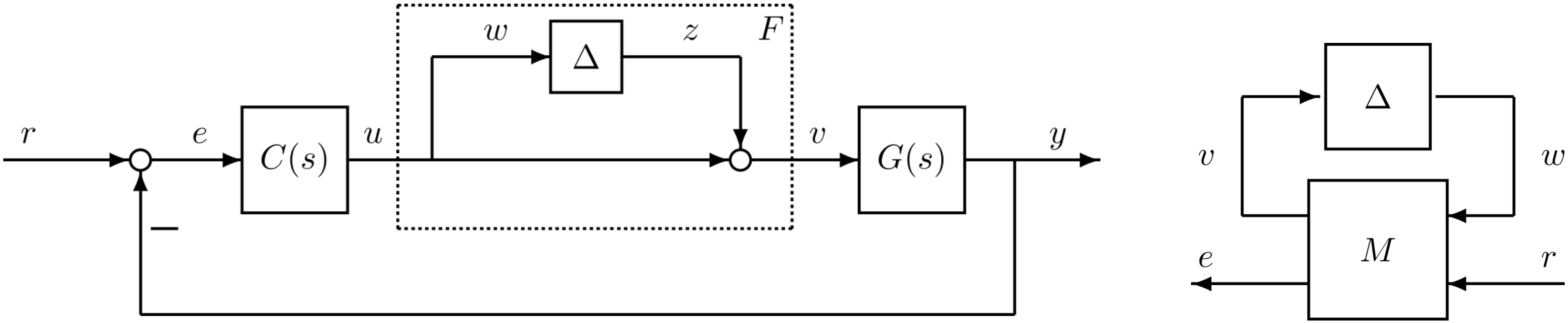
LFT Representation: $F_U(M,\Delta)$

$$M := \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} = \begin{bmatrix} -C(I+GC)^{-1}G & C(I+GC)^{-1} \\ -(I+GC)^{-1}G & (I+GC)^{-1} \end{bmatrix}$$

Robust Stability: $F_U(M,\Delta)$ is stable for all Δ with $||\Delta||_\infty \le m$ if and only if $\sup_\omega \mu_\Delta\left(M_{11}(j\omega)\right) < \frac{1}{m}$.

Unstructured Uncertainty: $\mu_\Delta\left(M_{11}(j\omega)\right) = \bar{\sigma}\left(M_{11}(j\omega)\right)$

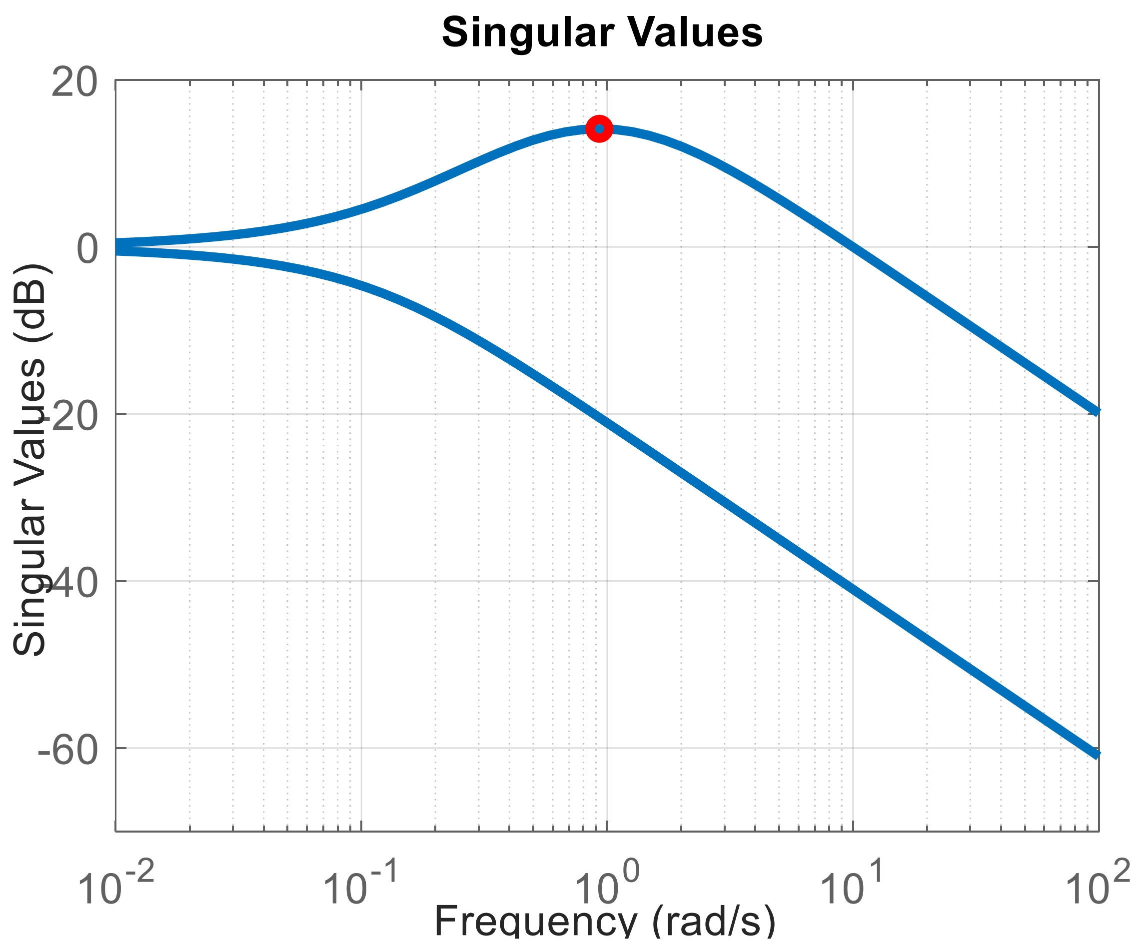
Stability Margin $\Rightarrow m = 1/||M_{11}||_\infty$ where $M_{11} = T_I$



Margins With Unstructured Uncertainty

$||M_{11}||_\infty = ||T_I||_\infty = 5.09$ at 0.97 rad/s.

Margin is $m = 1/||M_{11}||_\infty = 0.196$.



Code: Unstructured Uncertainty

```
% Stability Margin
TI = feedback(LI,I);
[np,wp] = hinfnorm(TI);
StabMarg = 1/np

% Display "sigma" plot for TI
sigma(TI,{1e-2, 1e2});
```

Margins With Unstructured Uncertainty

$||M_{11}||_\infty = ||T_I||_\infty = 5.09$ at 0.97 rad/s.

Margin is $m = 1/||M_{11}||_\infty = 0.196$.

We can construct a destabilizing uncertainty (as described in Lecture 4):

$$\Delta = \begin{bmatrix} -0.0193 - 0.0179i & -0.0002 - 0.0036i \\ -0.1927 + 0.0038i & -0.0193 - 0.0179i \end{bmatrix}$$

This has $\bar{\sigma}(\Delta) = 0.196$ and causes the feedback loop with $\tilde{G} = G(I + \Delta)$ to be unstable with a pole at $j0.97$.

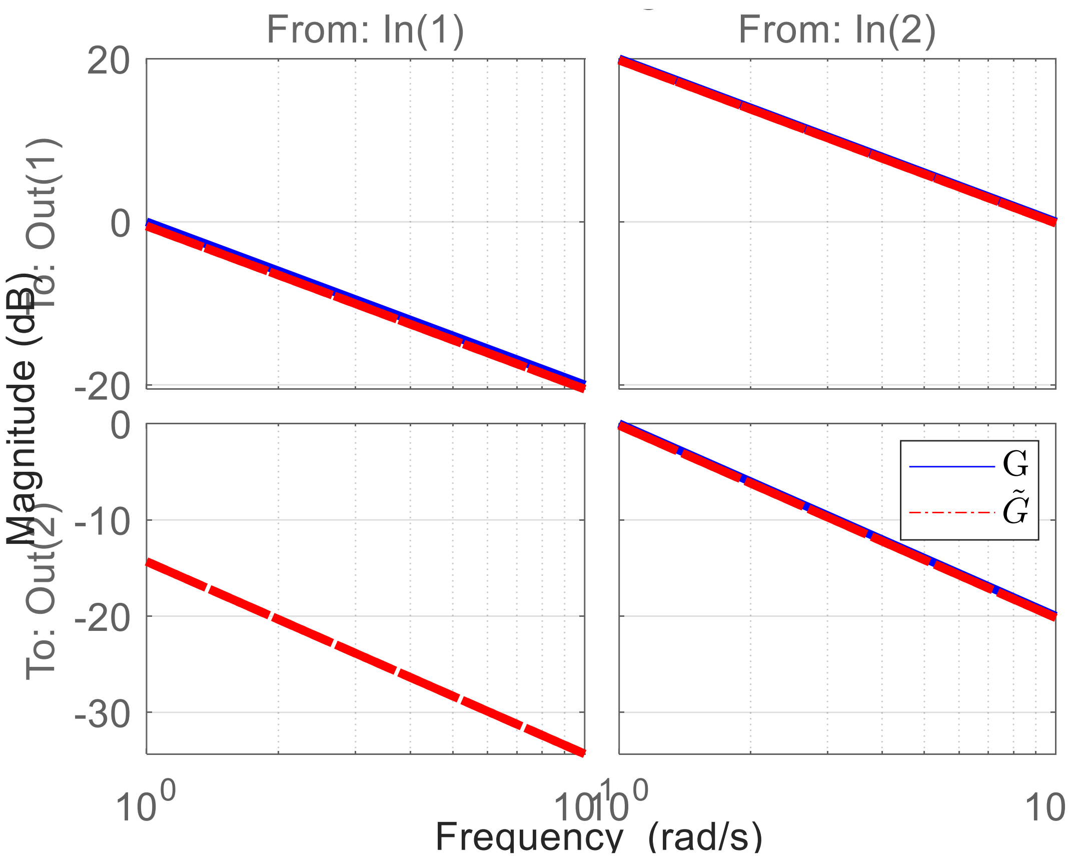
Margins With Unstructured Uncertainty

The destabilizing uncertainty is:

$$\Delta = \begin{bmatrix} -0.0193 - 0.0179i & -0.0002 - 0.0036i \\ -0.1927 + 0.0038i & -0.0193 - 0.0179i \end{bmatrix}$$

G and $\tilde{G} = G(I + \Delta)$ are similar except that Δ causes coupling in the (2,1) entry, i.e. from input 1 to output 2. See Bode mag plots ⇒

The engineer must determine if this a physically meaningful perturbation.



Code: Unstructured Uncertainty

```
% Construct destabilizing perturbation (complex matrix)
TIwp = freqresp(TI,wp);   % Evaluate TI at frequency wp
[U,S,V] = svd( TIwp );  u1 = U(:,1); v1 = V(:,1); s1 = S(1,1);
Delta = -v1*u1'/s1;

% Verify norm(Delta) = 1/||T||inf
[norm(Delta) 1/np]

% Verify Delta causes closed-loop pole at wp
Gpert = G*(I+Delta);
TIpert = feedback(C*Gpert,I);
pole(TIpert1)

% Study perturbed model
bodemag(G,'b',Gpert,'r-.');
```

Unstructured Uncertainty with μ

The analysis with unstructured Δ is simple enough that we can directly solve it with a few lines of code.

We will recompute the input margins for this case using high-level functions: `ultidyn` and `robstab`.

- The code builds the LFT model "under the hood",
- Computes lower/upper bounds on the structured singular value (μ): $L \le \sup_{\omega} \mu_\Delta\big(M_{11}(j\omega)\big) \le U$
- Uses the μ bounds to compute bounds on the margin.

$$\frac{1}{U} \le m = \frac{1}{\sup_{\omega} \mu_\Delta\big(M_{11}(j\omega)\big)} \le \frac{1}{L}$$

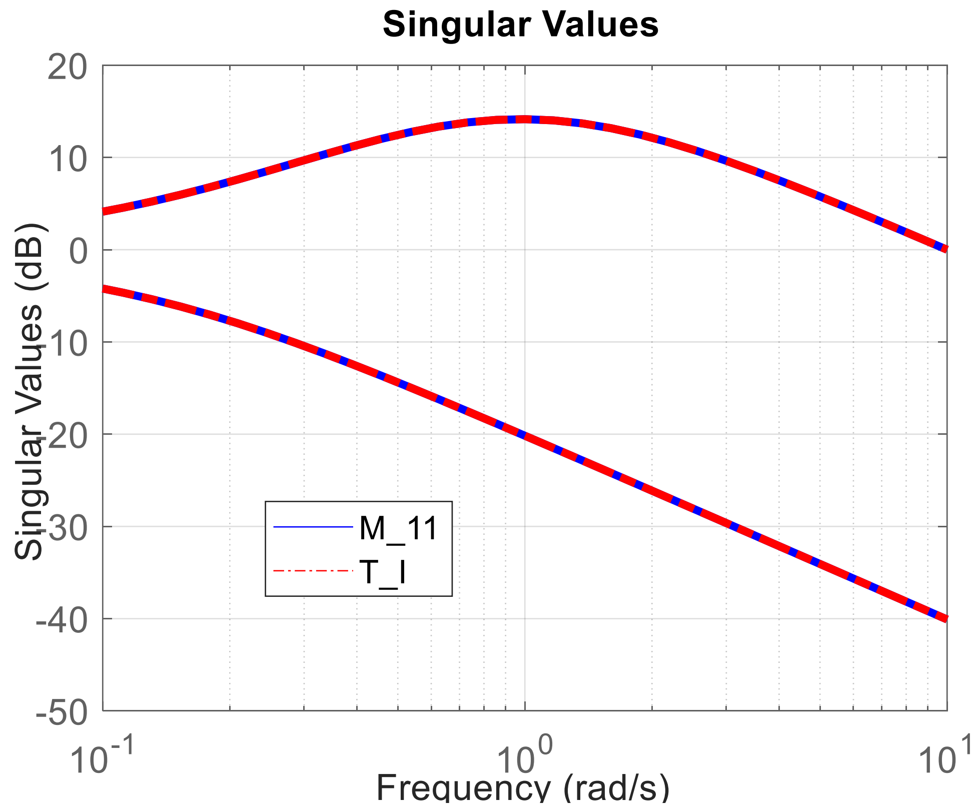
This will illustrate the code that will be used for more general unstructured uncertainties.

Code: Unstructured Uncertainty with μ

```
% Unstructured uncertainty: Stable, 2x2 LTI dynamics
Delta = ultidyn('Delta',[2 2]);

% Construct model with unstructured input uncertainty
F = I+Delta;
TI = feedback(C*G*F,I);

% Access LFT model constructed
% "under the hood"
M = lftdata(TI);
TInom = feedback(C*G,I);
M11 = M(1:2,1:2);
sigma(M11,'b',TInom,'r-.');
```



Code: Unstructured Uncertainty with μ

```
% Use ROBSTAB to compute stability margin
[StabMarg,WCUnc] = robstab(TI);


% StabMarg contains upper/lower bounds on the stability margin
and the critical frequency.
StabMarg
        LowerBound: 0.1958
        UpperBound: 0.1962
  CriticalFrequency: 1
```

These results agree (to within numerical errors) with the results computed in previous slides.

Code: Unstructured Uncertainty with μ

```
% Use ROBSTAB to compute stability margin
[StabMarg,WCUnc] = robstab(TI);
Delta = WCUnc.Delta;

% The perturbation Delta is an LTI system with
%  1) norm(Delta) = StabMarg.UpperBound
%     (This should be ~= 1/||TI||inf for this particular problem)
%  2) closed-loop to have a pole at frequency StabMarg.CriticalFrequency
[StabMarg.UpperBound hinfnorm(Delta)]   % = [ 0.1962    0.1962]
F = (I+Delta); TIpert = feedback(C*G*F,I);
pole(TIpert)    % Poles at +/- 1j (agrees with StabMarg.CriticalFrequency)
```

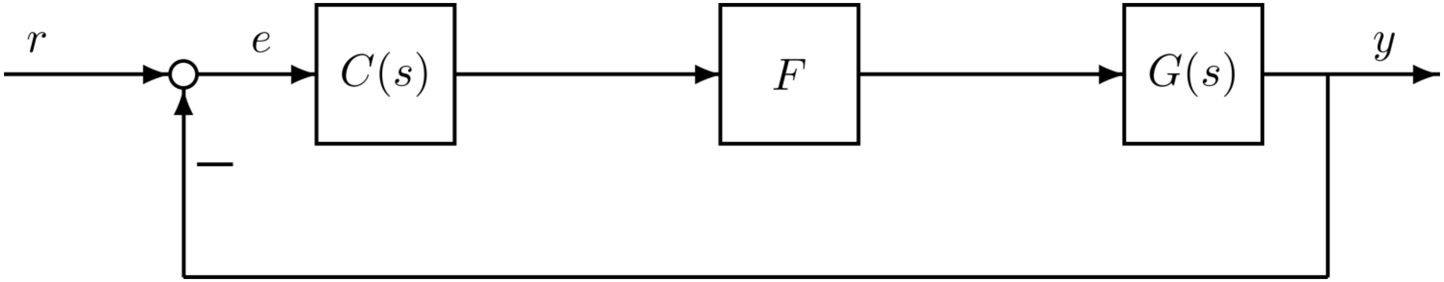
ROBSTAB fits the complex matrix uncertainty to generate an LTI system $\Delta(s)$ with the properties given above.

Multi-Loop Margins

Symmetric Disk Uncertainty: F is diagonal,

$$F = \begin{bmatrix} \frac{1+\delta_1/2}{1-\delta_1/2} & 0 \\ 0 & \frac{1+\delta_2/2}{1-\delta_2/2} \end{bmatrix} \text{ with } \delta_i \in \mathbb{C} \text{ and } |\delta_i| \leq m$$

This specific form models symmetric disks of gain/phase uncertainty on each channel, i.e. no cross-coupling.



Multi-Loop Margins

Symmetric Disk Uncertainty: F is diagonal,

$$F = \begin{bmatrix} \frac{1+\delta_1/2}{1-\delta_1/2} & 0 \\ 0 & \frac{1+\delta_2/2}{1-\delta_2/2} \end{bmatrix} \text{ with } \delta_i \in \mathbb{C} \text{ and } |\delta_i| \leq m$$

This specific form models symmetric disks of gain/phase uncertainty on each channel, i.e. no cross-coupling.

```
[DMI,MMI] = diskmargin(C*G);  % Multi-loop margins, MMI
MMI.DiskMargin = 1.9959
```

The multi-loop margins for this example are almost the same as the loop-at-a-time margins.

This system is sensitive to unstructured uncertainty but robust to multi-loop disk uncertainty.

Multi-Loop Disk Margins with μ

We used the `diskmargin` function to compute the multi-loop disk margins.

We will recompute the multi-loop margins for this case using high-level functions: `ultidyn` and `robstab`.

- The code builds the LFT model "under the hood",
- Computes lower/upper bounds on the structured singular value (μ): $L \le \sup_\omega \mu_\Delta\big(M_{11}(j\omega)\big) \le U$
- Uses the μ bounds to compute bounds on the margin.

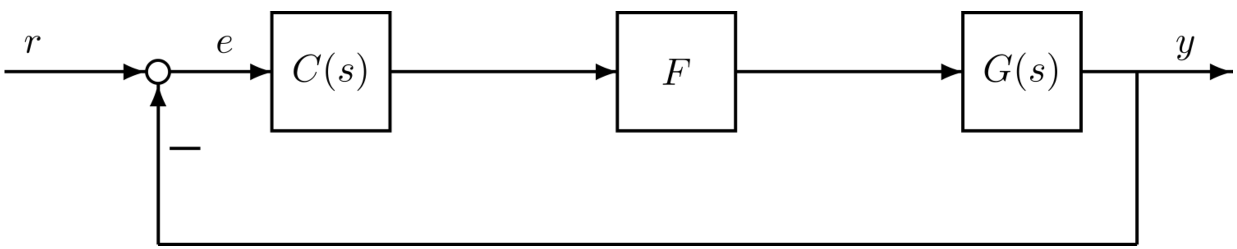
$$\frac{1}{U} \le m = \frac{1}{\sup_\omega \mu_\Delta\big(M_{11}(j\omega)\big)} \le \frac{1}{L}$$

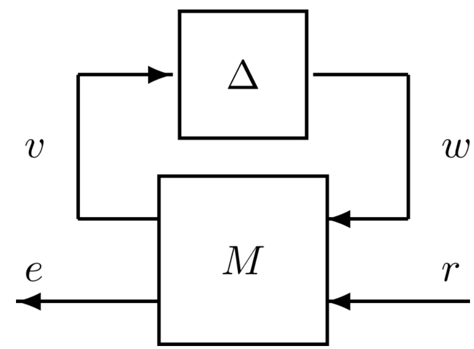
This will illustrate the conceptual code that is used by `diskmargin` (actual implementation is more efficient).

Code: Multi-Loop Disk Margins with μ

The closed-loop can be represented as an LFT model $F_U(M, \Delta)$ with $\Delta = \begin{bmatrix} \delta_1 & 0 \\ 0 & \delta_2 \end{bmatrix}$.

$$F = \begin{bmatrix} \frac{1+\delta_1/2}{1-\delta_1/2} & 0 \\ 0 & \frac{1+\delta_2/2}{1-\delta_2/2} \end{bmatrix}$$

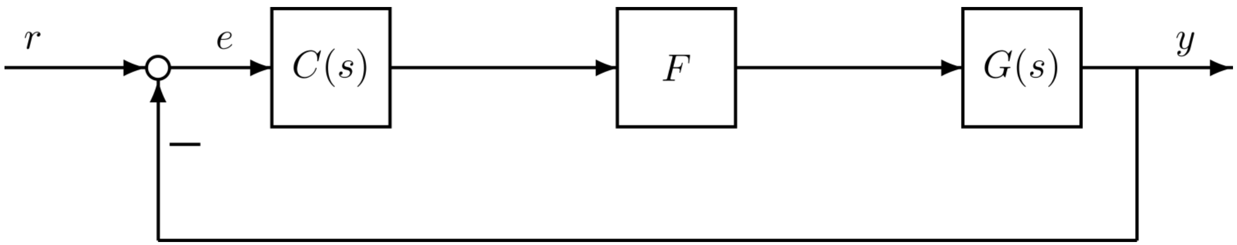


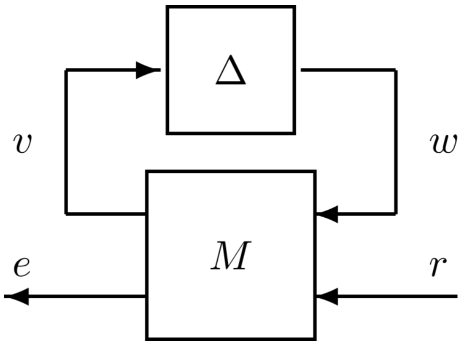


Code: Multi-Loop Disk Margins with μ

The closed-loop can be represented as an LFT model $F_U(M,\Delta)$ with $\Delta = \begin{bmatrix} \delta_1 & 0 \\ 0 & \delta_2 \end{bmatrix}$.

```
d1 = ultidyn('d1',[1 1]);
d2 = ultidyn('d2',[1 1]);
F = [(1+d1/2)/(1-d1/2) 0; 0 (1+d2/2)/(1-d2/2)];
SO = feedback(G*F*C,I);
[M,Delta] = lftdata(SO)
```





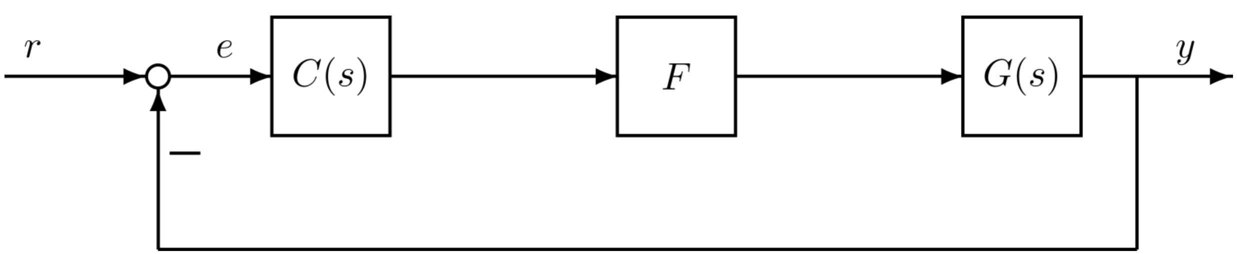
Code: Multi-Loop Disk Margins with μ

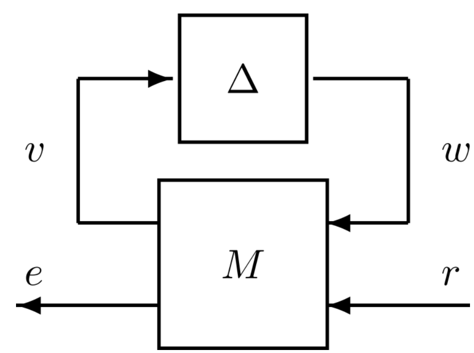
LFT model $F_U(M,\Delta)$ with $\Delta = \begin{bmatrix} \delta_1 & 0 \\ 0 & \delta_2 \end{bmatrix}$.

Stability margins:

- Compute lower/upper bounds on the structured singular value (μ): $L \leq \sup_\omega \mu_\Delta\big(M_{11}(j\omega)\big) \leq U$
- Use the μ bounds to compute bounds on the margin.

$$\frac{1}{U} \leq m = \frac{1}{\sup_\omega \mu_\Delta\big(M_{11}(j\omega)\big)} \leq \frac{1}{L}$$





Code: Multi-Loop Disk Margins with μ

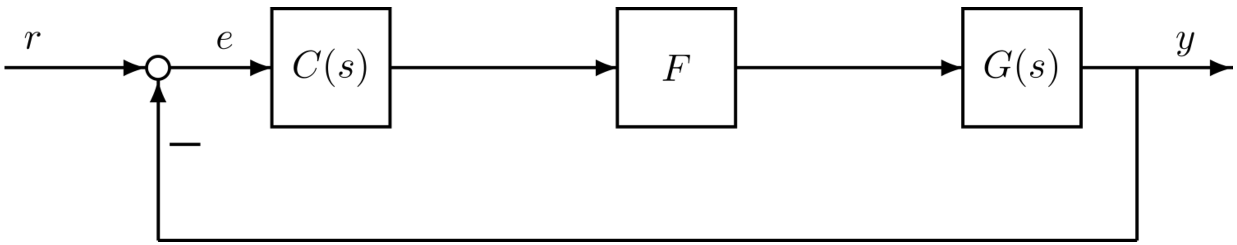
LFT model $F_U(M,\Delta)$ with $\Delta = \begin{bmatrix} \delta_1 & 0 \\ 0 & \delta_2 \end{bmatrix}$.

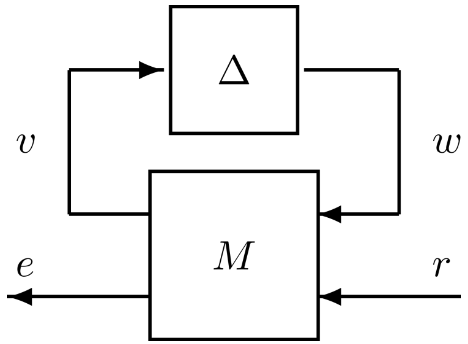
μ lower bound at ω_k is computed by a modified power iteration.

μ upper bound at ω_k:

- The D-scales are $\mathcal{D} = \left\{ \begin{bmatrix} d_1 & 0 \\ 0 & d_2 \end{bmatrix} : \ d_1, d_2 \in \mathbb{R} \right\}$.
- For every $\Delta \in \mathbf{\Delta}$ and nonsingular $D \in \mathcal{D}$, we have $D\Delta = \Delta D$
- Upper bound at frequency ω_k is

$$\mu_{\mathbf{\Delta}}(M_{11}(j\omega_k)) \leq \min_{D \in \mathcal{D}} \bar{\sigma}(D M_{11}(j\omega_k) D^{-1})$$





Code: Multi-Loop Disk Margins with μ

```
% Use ROBSTAB to compute stability margin
[StabMarg,WCUnc] = robstab(SO);


% StabMarg contains upper/lower bounds on the stability margin
and the critical frequency.
StabMarg
        LowerBound: 1.9948
        UpperBound: 2.0000
  CriticalFrequency: 1
```

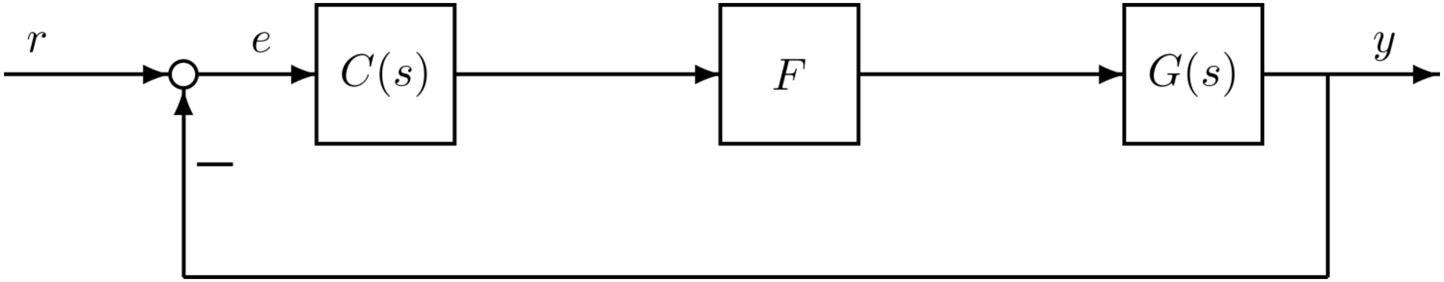
These results agree (to within numerical errors) with the results computed in previous slides using `diskmargin`.

Summary

The crossfeed system is:

- Robust to loop-at-a-time uncertainty. This is rarely, if ever, a good model for uncertainty.
- Sensitivity to unstructured uncertainty (which includes cross-coupling)
- Robust to multi-loop disk margin uncertainty.

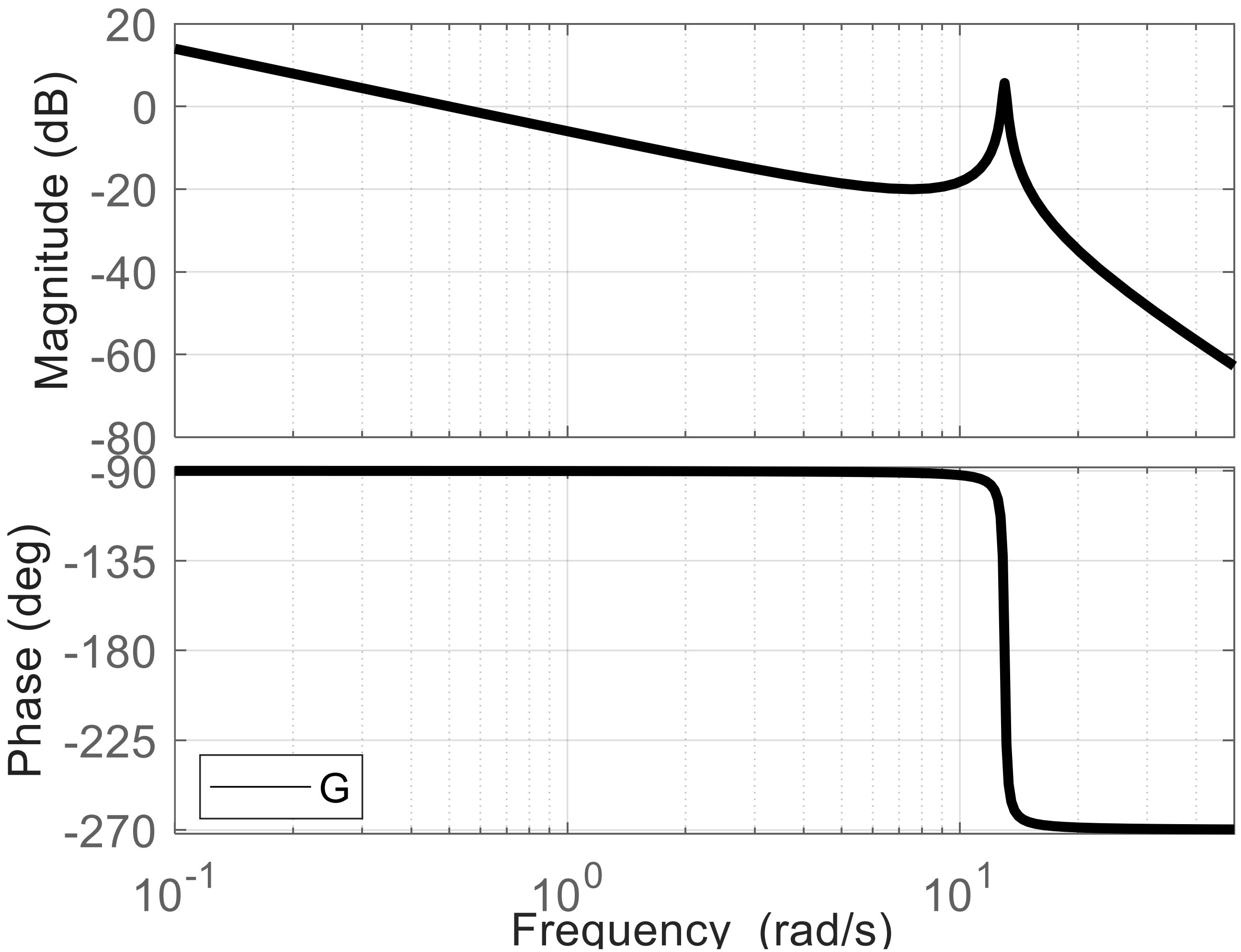
The design engineer must determine if the cross-coupling is a realistic possible uncertainty and, if so, the lack of robustness to this uncertainty would be an issue.



Example: System with Uncertain, Lightly Damped Modes

Example

Plant: $G(s) = \frac{0.5}{s} \cdot \frac{\omega_{n0}^2}{s^2 + 2\zeta_0 \omega_{n0} s + \omega_{n0}^2}$ with $\omega_{n0} = 13\text{rad/s}$, $\zeta_0 = 0.01$



Example

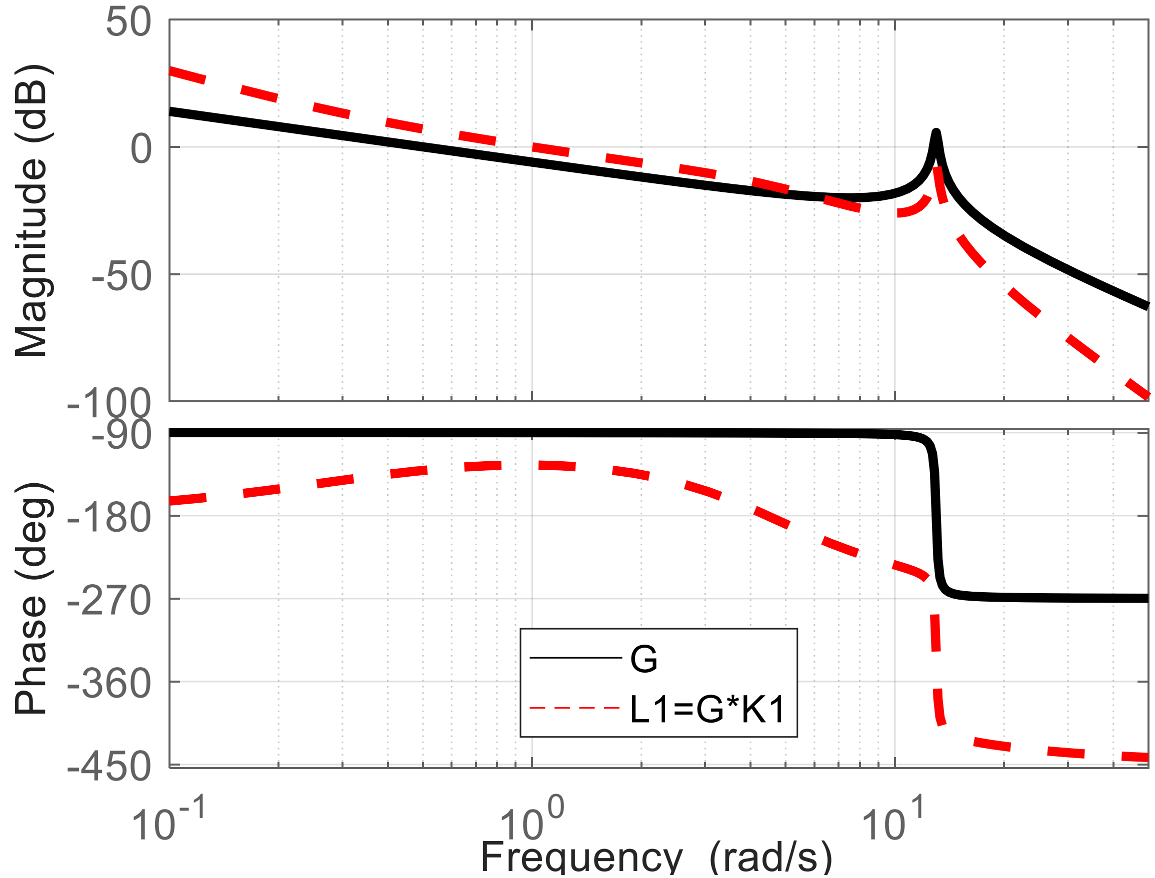
Plant: $G(s) = \frac{0.5}{s} \cdot \frac{\omega_{n0}^2}{s^2 + 2\zeta_0 \omega_{n0}\, s + \omega_{n0}^2}$ with $\omega_{n0} = 13\text{rad/s}, \zeta_0 = 0.01$

Controller 1: $K_1(s) = \frac{6.29\, s + 1.99}{3.32\, s} \cdot \frac{22.1}{0.999\, s^2 + 6.64\, s + 22.1}$

PI controller with second-order roll-off and $\omega_{L1} = 1$ rad/s.

Margins:

- GM=[0, 5.54]
- PM=55°



Example

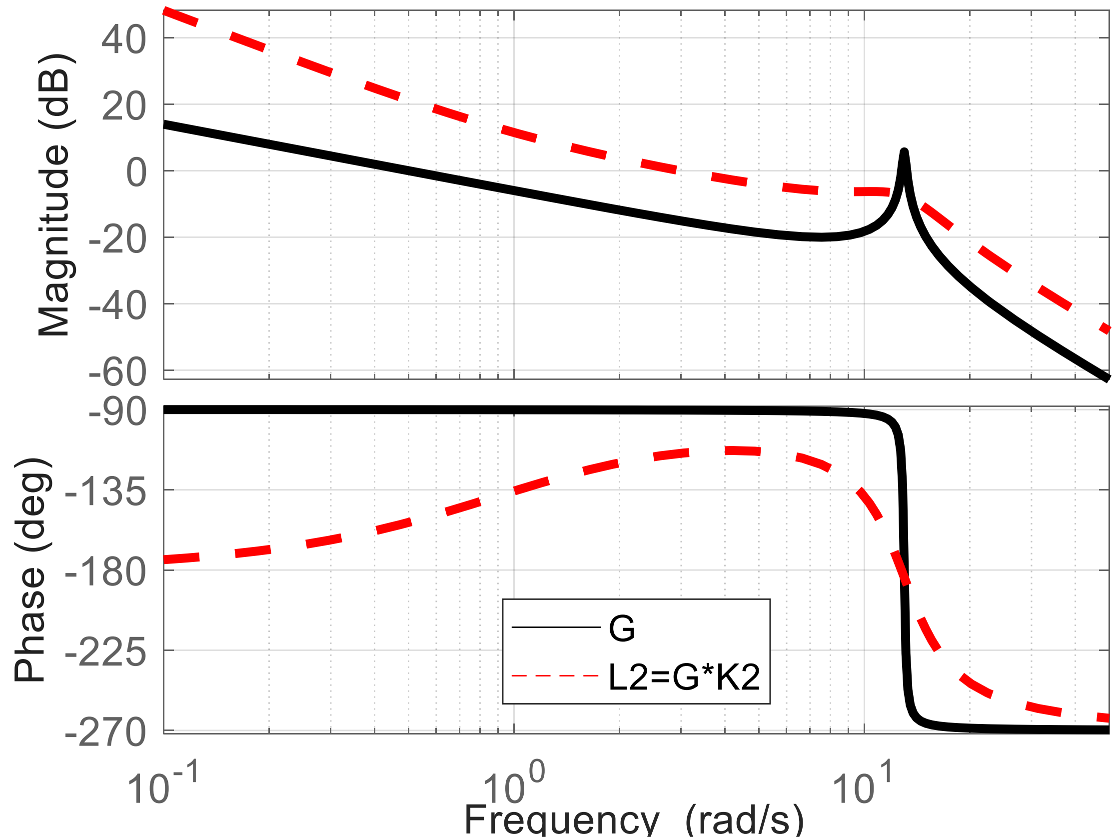
Plant: $G(s) = \frac{0.5}{s} \cdot \frac{\omega_{n0}^2}{s^2 + 2\zeta_0 \omega_{n0} s + \omega_{n0}^2}$ with $\omega_{n0} = 13$rad/s, $\zeta_0 = 0.01$

Controller 2: $K_2(s) = \frac{17.96\, s + 17.04}{3.32\, s} \cdot \frac{s^2 + 0.26\, s + 169}{s^2 + 6.5\, s + 169}$

PI controller with notch and $\omega_{L2} = 2$ rad/s.

Margins:

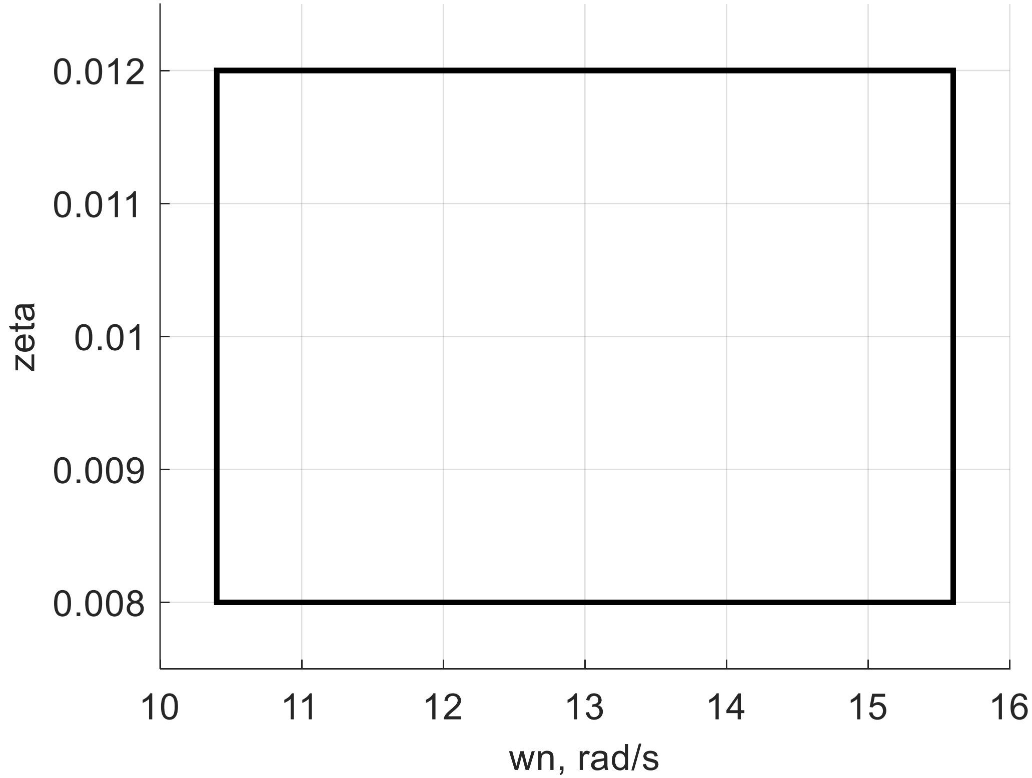
- GM=[0, 2.31]
- PM=65°



Uncertainty Modeling

Model +/-20% uncertainty in the modal parameters:

$$\{(\omega_n, \zeta) : 10.4 \leq \omega_n \leq 15.6, 0.008 \leq \zeta \leq 0.012\}$$



Uncertainty Modeling

Model +/-20% uncertainty in the modal parameters:

$$\{(\omega_n, \zeta) : 10.4 \le \omega_n \le 15.6, 0.008 \le \zeta \le 0.012\}$$

These uncertainties are not directly comparable, e.g. ζ is unitless while ω_n is in units of rad/s.

We can used "normalized" uncertainties (δ_1, δ_2) to make them directly comparable:

$\omega_n = \omega_{n0} \times (1 + 0.2\delta_1)$ where $\omega_{n0} = 13$ rad/s.

$\zeta = \zeta_0 \times (1 + 0.2\delta_2)$ where $\zeta_0 = 0.01$.

The actual parameter uncertainty corresponds to the following normalized set:

$$\{(\delta_1, \delta_2) : \delta_1, \delta_2 \in \mathbb{R}, -1 \le \delta_1, \delta_2 \le 2\}$$

Uncertainty Modeling

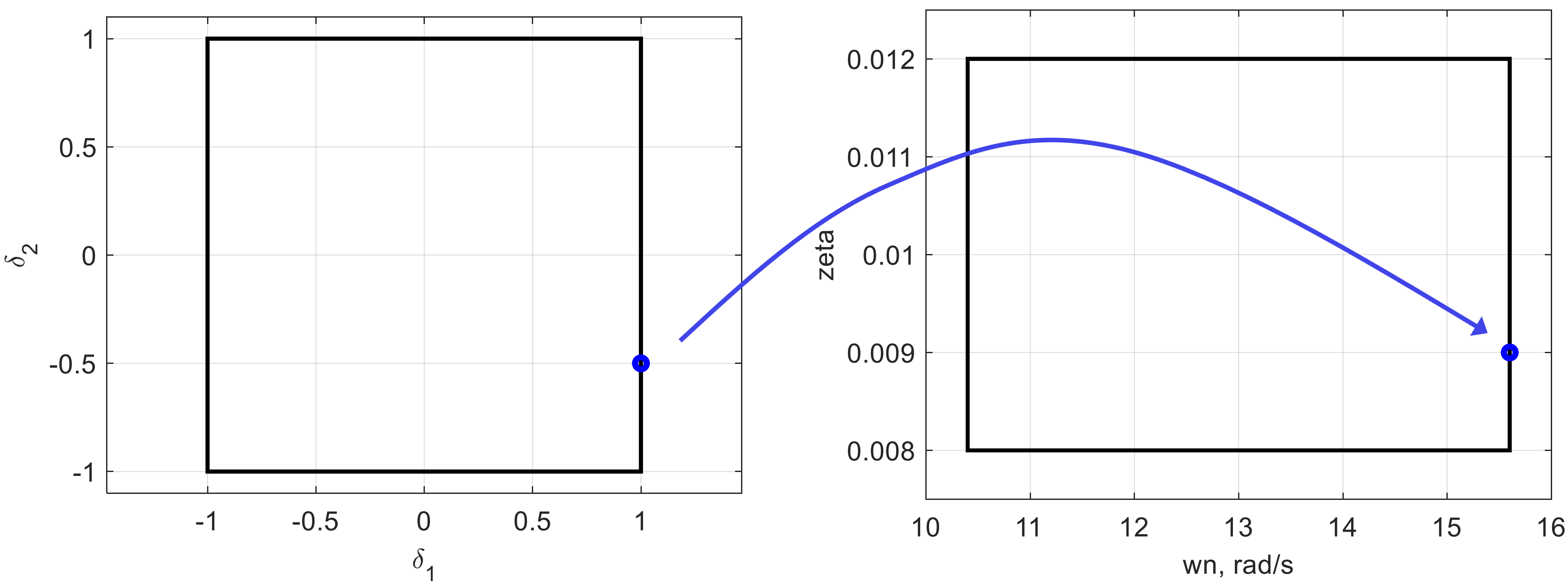
Model +/-20% uncertainty in the modal parameters:

$$\{(\omega_n, \zeta) : 10.4 \leq \omega_n \leq 15.6, 0.008 \leq \zeta \leq 0.012\}$$

Normalized uncertainty:

$$\{(\delta_1, \delta_2) : -1 \leq \delta_1, \delta_2 \leq 2\}$$

Example: $(\delta_1, \delta_2) = (1, -0.5) \Rightarrow (\omega_n \zeta) = (15.6{,}0.009)$.



Code: Uncertainty Modeling

```
% 20% uncertainty in modal frequency and damping:
pvar = 0.2;
zetaU = ureal('zeta',zeta0,'Range',zeta0*[(1-pvar), (1+pvar)]);
wnU = ureal('wn',wn0,'Range',wn0*[(1-pvar), (1+pvar)]);

% Uncertain plant model
GU = tf(0.5,[1 0])*tf(wnU^2,[1 2*zetaU*wnU wnU^2])
Uncertain continuous-time state-space model with 1 outputs, 1 inputs, 3 states.
The model uncertainty consists of the following blocks:
  wn: Uncertain real, nominal = 13, range = [10.4,15.6], 5 occurrences
  zeta: Uncertain real, nominal = 0.01, range = [0.008,0.012], 1 occurrences
```

Minor point: The code uses a total of 5 copies of ω_n: 2 each for ω_n^2 in the num. and den. and one more for the $2\zeta\omega_n$ term in the den. This is not a minimal realization.

Code: Uncertainty Modeling

```
% 20% uncertainty in modal frequency and damping:
pvar = 0.2;
zetaU = ureal('zeta',zeta0,'Range',zeta0*[(1-pvar) (1+pvar)]);
wnU = ureal('wn',wn0,'Range',wn0*[(1-pvar) (1+pvar)]);
wnU.AutoSimplify = 'full';

% Uncertain plant model
GU = tf(0.5,[1 0])*tf(wnU^2,[1 2*zetaU*wnU wnU^2])
Uncertain continuous-time state-space model with 1 outputs, 1 inputs, 3 states.
The model uncertainty consists of the following blocks:
  wn: Uncertain real, nominal = 13, range = [10.4,15.6], 3 occurrences
  zeta: Uncertain real, nominal = 0.01, range = [0.008,0.012], 1 occurrences
```

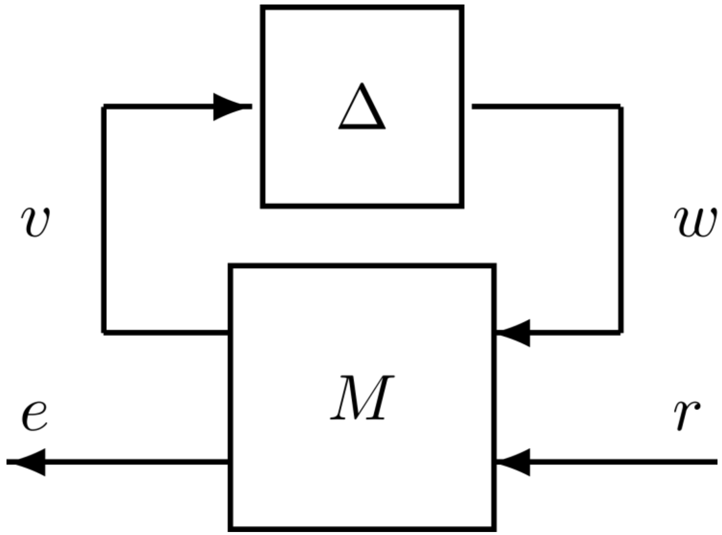
Minor point: The AutoSimplify option finds a realization with less copies and this reduces computation in `robstab`.

Code: Uncertainty Modeling

```
% Access LFT Model where Delta = diag(delta1*I_3, delta2)
[M,Delta] = lftdata(GU);

% Ex.: (delta1,delta2)=(1,-0.5) maps to (wn,zeta) = (15.6,0.009)
d1 = 1;  d2 = -0.5;  D = blkdiag( d1*eye(3), d2);
Gex = lft(D,M);
damp(Gex)
```

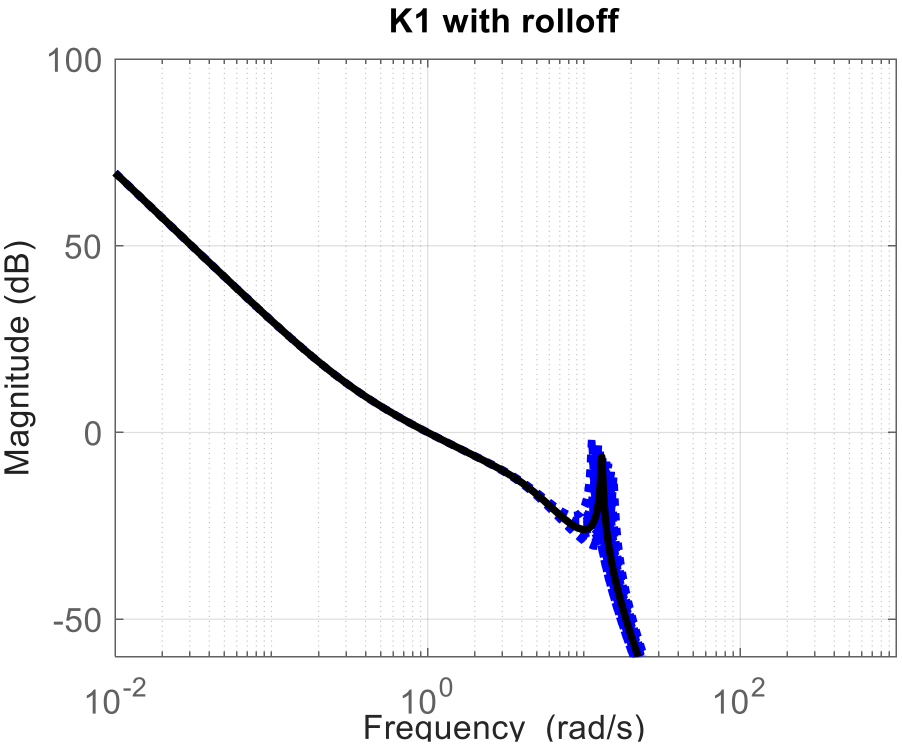
Gex has poles at -0.14 +/- 15.6 j corresponding to ω_n = 15.6 rad/s and ζ = 0.009, as expected.

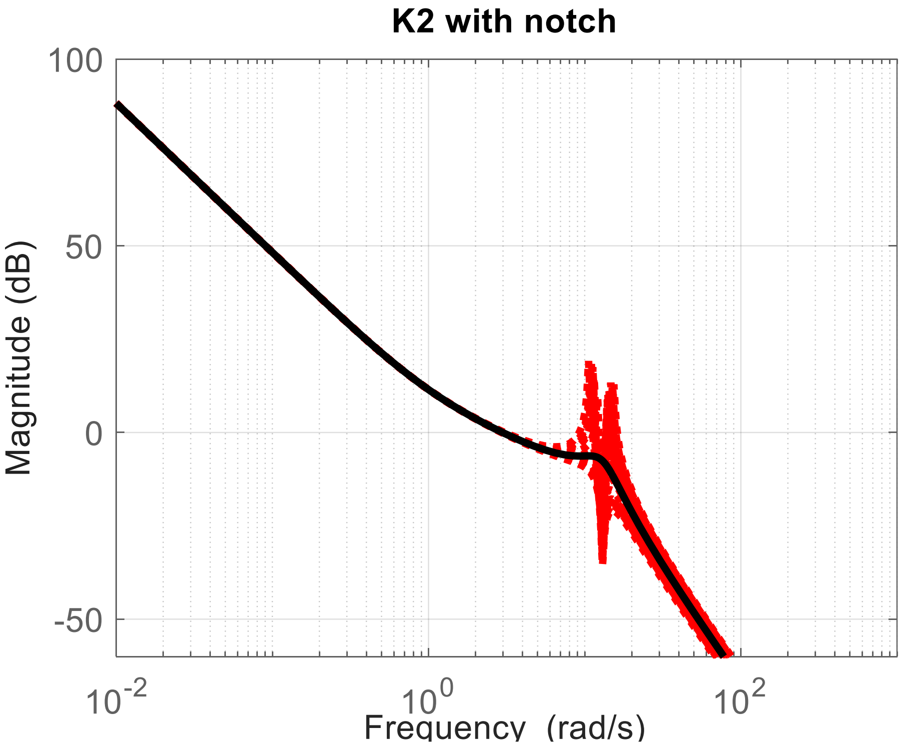


Random Sampling Of Uncertain Parameters

The loops with K1 and K2 are shown below for the nominal (solid black) and samples (dashed colors).

The closed-loops with roll-off are stable for all samples but not with the notches.





Analysis With Structured Singular Value

```
% Stability Margin Code
TU1 = feedback(GU*K1,1);  SM1 = robstab(TU1)
SM1 =
          LowerBound: 3.1704
          UpperBound: 3.1768
   CriticalFrequency: 4.3246
```

The stability margin is ≈3.17. The closed-loop with K_1 can tolerate approximately 3x the modeled uncertainty, i.e. $\approx 3.17 \times 20 = 63.4\%$ uncertainty in the parameters.

The system with controller K_1 (roll-off) is robust to the parameter uncertainty.

Analysis With Structured Singular Value

```
% Stability Margin Code
TU2 = feedback(GU*K2,1);  [SM2,WCU2] = robstab(TU2); SM2
SM2 =
          LowerBound: 0.1858
          UpperBound: 0.1874
   CriticalFrequency: 12.1646
```

The stability margin is ≈ 0.19. The closed-loop with K_2 is:

1) Stable for all parameter combinations $\leq \mathrm{LowerBound} \times 20 \ = 3.7\%$ from nominal.

2) Unstable for a parameter combination at most $\mathrm{UpperBound} \times 20 \ = 3.7\%$ from nominal.

The system with controller K_2 (notch) is NOT robust to the parameter uncertainty.

Analysis With Structured Singular Value

% Stability Margin Code

TU2 = feedback(GU*K2,1); [SM2,WCU2] = robstab(TU2); WCU2

WCU2 =

wn: 12.5127

zeta: 0.0097

WCU2 causes TU2 to be unstable with poles at ± 12.1646j. This agrees with SM2.CriticalFrequency.

The combination in WCU2 corresponds to the normalized uncertainty $(\delta_1, \delta_2) = (-0.1874, -0.1460)$. This agrees with SM2.UpperBound.

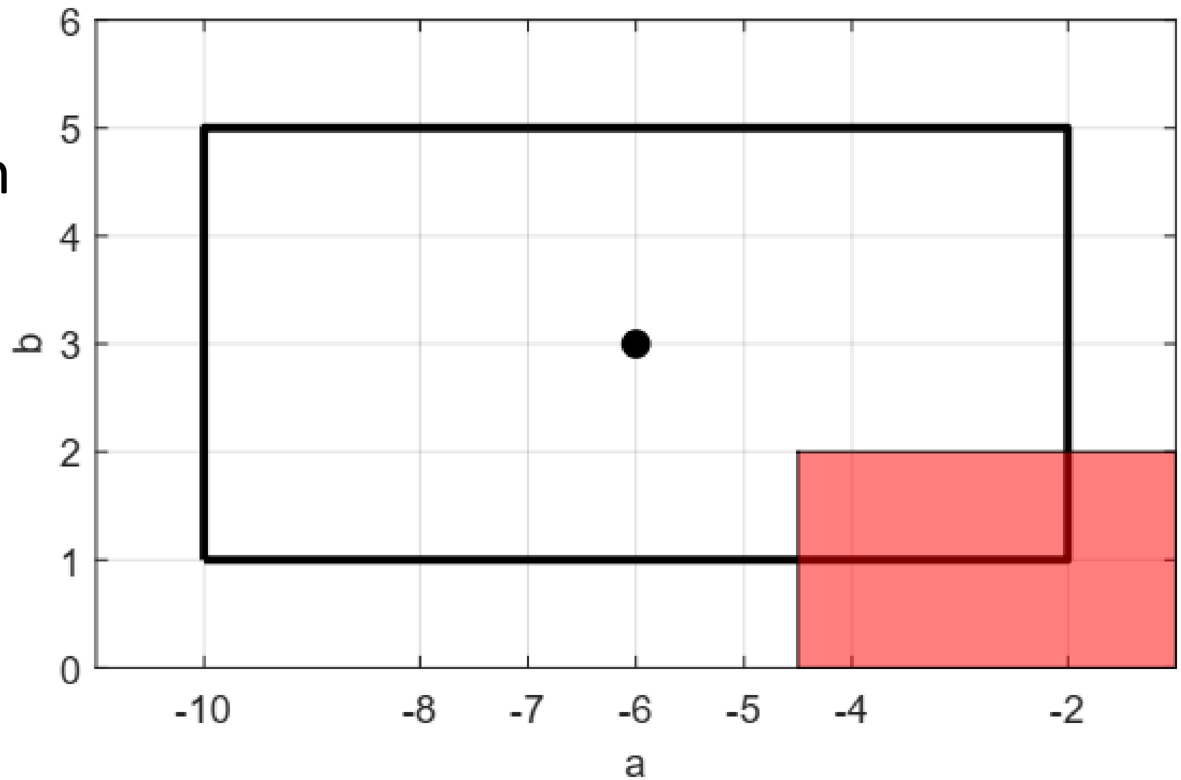
Problem

Consider an LTI system that depends on two uncertain parameters:

- $a \in [-10, -2]$ with nominal $a_0 = -6$, and
- $b \in \ [1, 5]$ with nominal $b_0 = 3$.

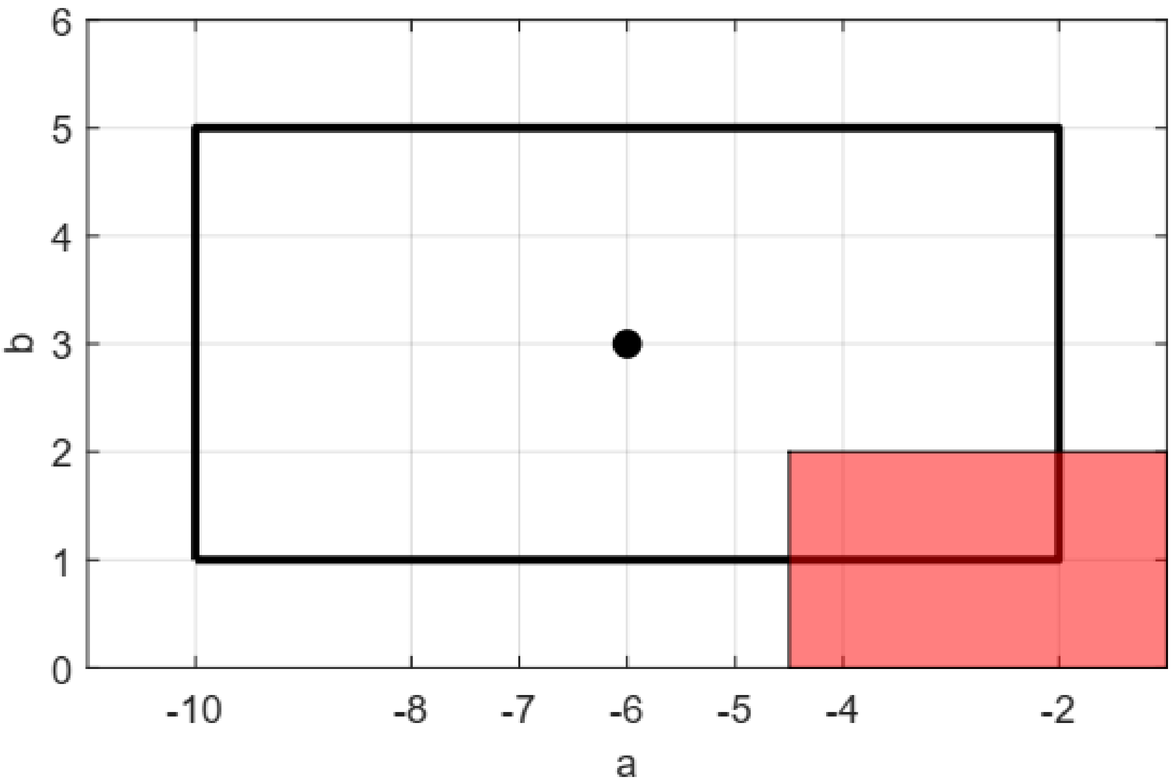
The system is unstable for $\{(a, b) : a \geq -4.5 \text{ and/or } b \leq 2\}$ and stable otherwise.

- Is the uncertain system robustly stable?
- What is the stability margin for the modeled uncertainty?



Solutions

- Is the uncertain system robustly stable?
- What is the stability margin for the modeled uncertainty?



Robust Performance

MIMO Small Gain Condition

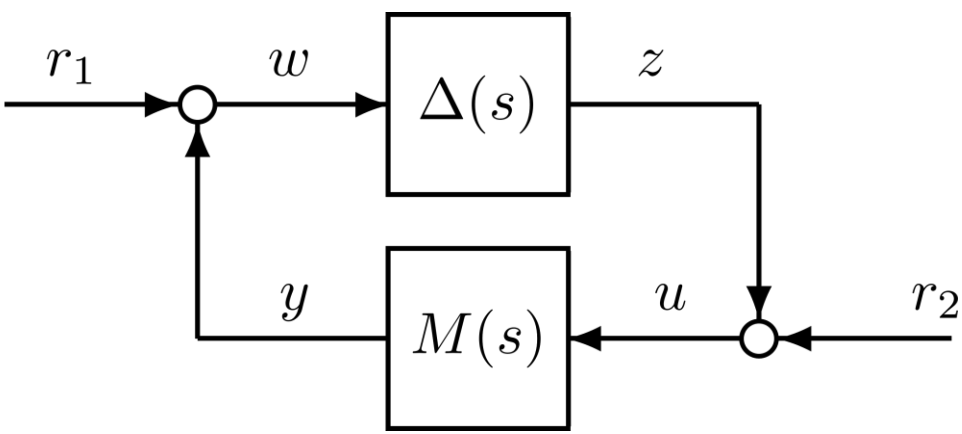
Theorem: Consider the positive feedback system below where $M(s)$ is stable.

A) If $||\mathrm{M}||_\infty \leq 1$ then the feedback system is stable for all $\Delta(s)$ that are stable and is norm-bounded $||\Delta||_\infty < 1$.

B) If $||\mathrm{M}||_\infty > 1$ then there is a stable $\Delta(s)$ with $||\Delta||_\infty < 1$ such that the feedback system is unstable.

We used this to prove robust stability based on $||\mathrm{M}||_\infty$.

We can use this in reverse: Prove a bound on $||\mathrm{M}||_\infty$ by checking for robust stability.

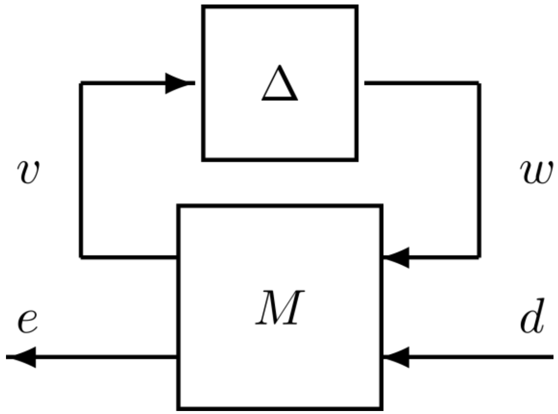


Robust Performance

Assume M and Δ are both stable. Also assume $\|M_{11}\|_\infty < 1$ so that the LFT is robustly stable.

What condition on M ensures robust performance?

$$\sup_{\|\Delta\|_\infty \leq 1} \|F_U(M, \Delta)\|_\infty < 1$$



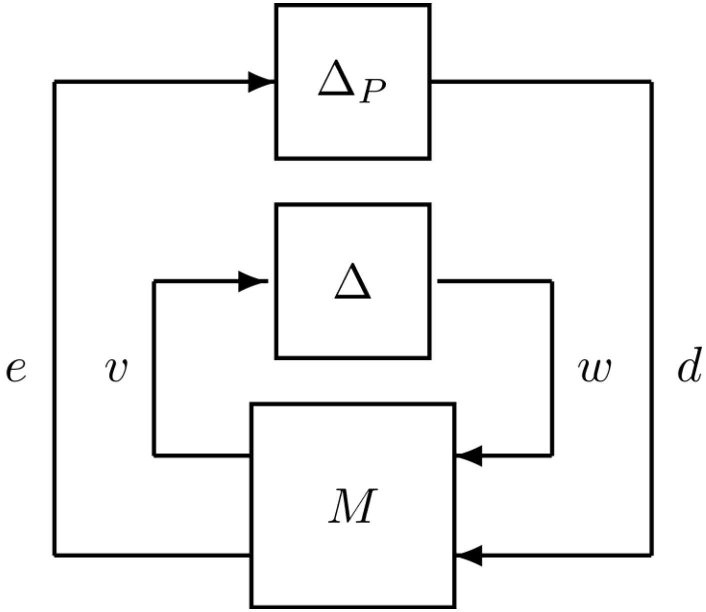
Robust Performance

Assume M and Δ are both stable. Also assume $\|M_{11}\|_\infty < 1$ so that the LFT is robustly stable.

What condition on M ensures robust performance?

$$\sup_{\|\Delta\|_\infty \leq 1} \|F_U(M,\Delta)\|_\infty < 1$$

Robust performance is equivalent to robust stability of M with an additional unstructured uncertainty block Δ_P on the performance channel.



Robust Performance

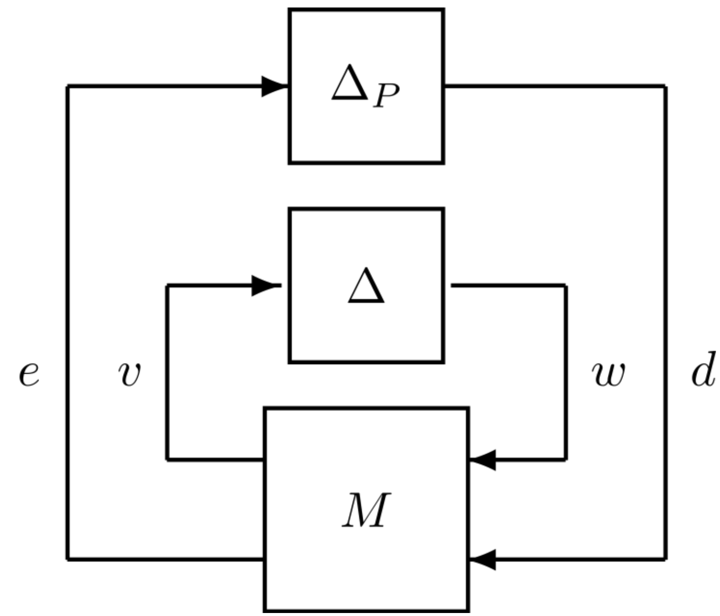
Assume M and Δ are both stable. Also assume $\|M_{11}\|_\infty < 1$ so that the LFT is robustly stable.

What condition on M ensures robust performance?

$$\sup_{\|\Delta\|_\infty \leq 1} \|F_U(M, \Delta)\|_\infty < 1$$

We analyze robust performance for the original problem by checking robust stability of M with respect to the larger, augmented uncertainty:

$$\left\| \begin{bmatrix} \Delta & 0 \\ 0 & \Delta_P \end{bmatrix} \right\|_\infty \leq 1$$



Example: MIMO Robustness Analysis

(Matlab Demo: MIMORobustnessAnalysisExample.mlx)

Uncertain Plant

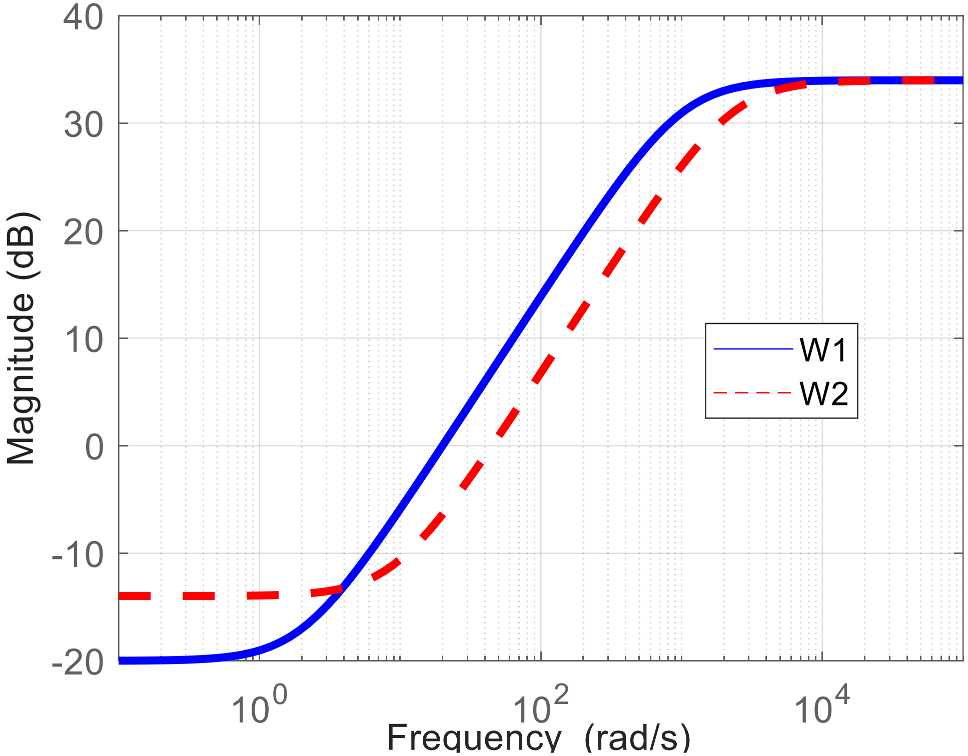
Plant: $G(s) = H(s) \times \begin{bmatrix} 1 + W_1(s)\Delta_1(s) & 0 \\ 0 & 1 + W_2(s)\,\Delta_2(s) \end{bmatrix}$

where

$H(s) = \frac{1}{s^2+p^2}\begin{bmatrix} s-p^2 & p(s+1) \\ -p(s+1) & s-p^2 \end{bmatrix}$ and $p = 10 \pm 10\%$.

$W_1(s) = \frac{50\,s + 100.5}{s+1005}$

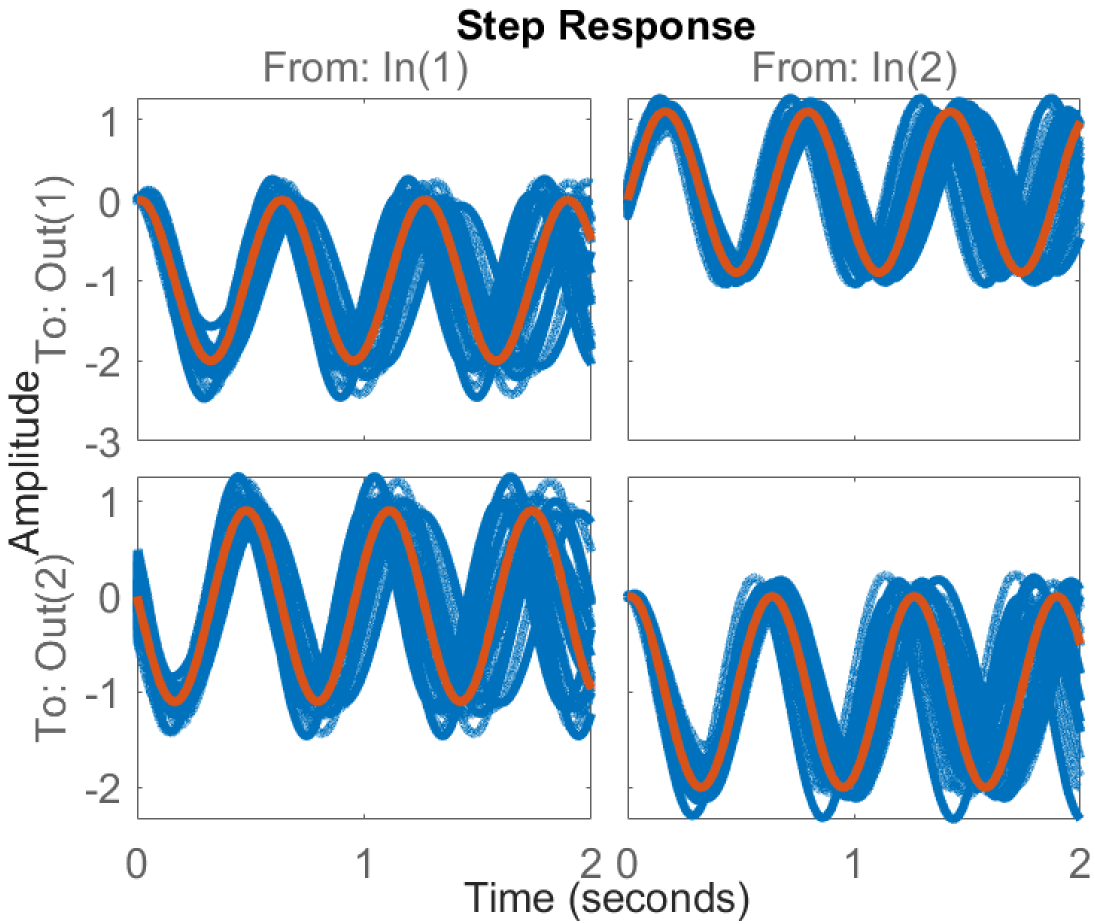
$W_2(s) = \frac{50\,s + 459.2}{s+2296}$



Uncertain Plant

Plant: $G(s) = H(s) \times \begin{bmatrix} 1 + W_1(s)\Delta_1(s) & 0 \\ 0 & 1 + W_2(s)\,\Delta_2(s) \end{bmatrix}$

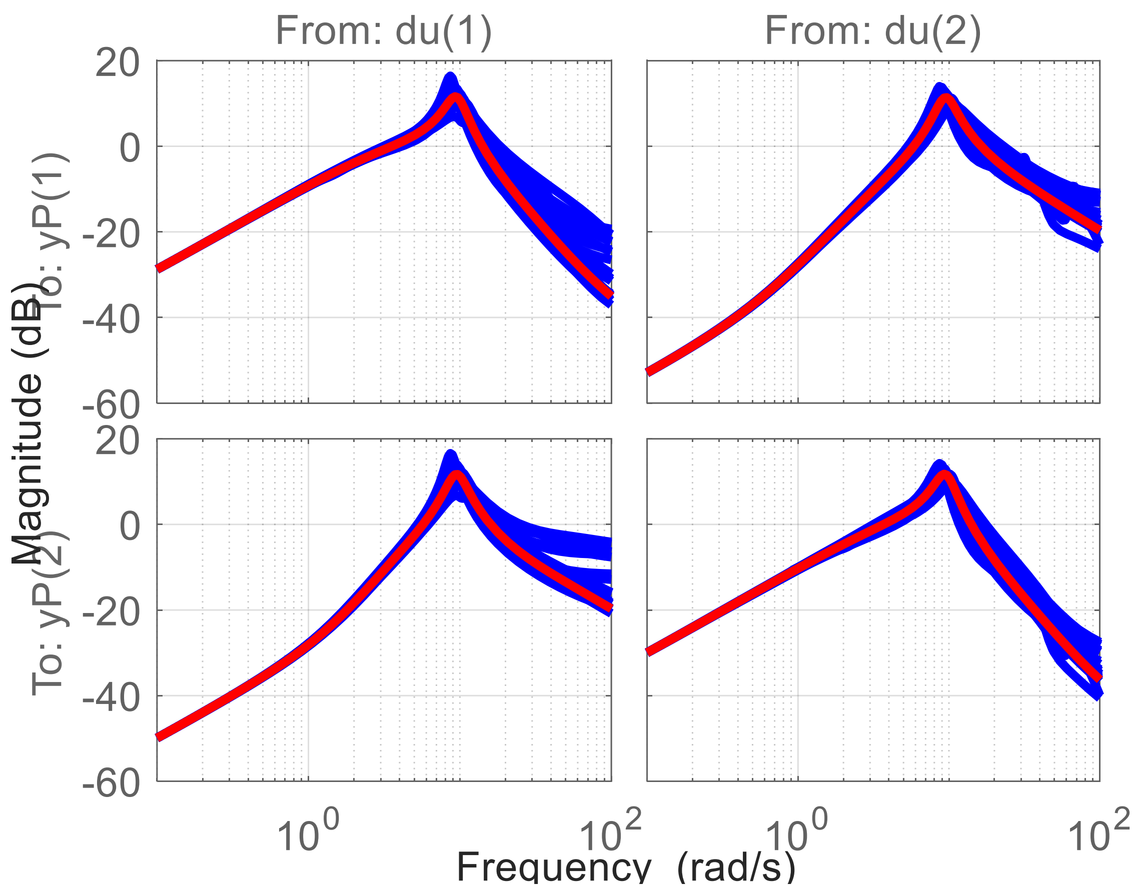
Samples of step response (blue) and nominal (red). The samples show the 10% variation in natural frequency.



Closed-Loop

A 2-by-2 MIMO controller is loaded in the demo.
(Details of the design are not important for this analysis.)

Samples of closed-loop response from input disturbance to plant output (blue) and nominal (red). The samples show the 10% variation in natural frequency and non-parametric uncertainty.



Code: Uncertain Closed-Loop

```
% Uncertain Plant
p = ureal('p',10,'Percentage',10);
A = [0 p;-p 0]; B = eye(2); C = [1 p;-p 1];
H = ss(A,B,C,[0 0;0 0])
W1 = makeweight(.1,20,50); W2 = makeweight(.2,45,50);
Delta1 = ultidyn('Delta1',[1 1]); Delta2 = ultidyn('Delta2',[1 1]);
G = H*blkdiag(1+W1*Delta1,1+W2*Delta2)
stepplot(G,2)

% Controller
load('mimoKexample.mat’)

% Robust Stability
GSi = feedback(G,K);
bodemag(GSi);
```

Robust Stability Margin

The lower and upper bounds on the robust stability margin returned by `robstab` are [2.213,2.217].

- The closed-loop is stable for all uncertainties up to 2.213 times the modeled uncertainty.
- There is a combination of (p, Δ_1, Δ_2) within 2.217 of the modeled uncertainty that causes the closed-loop to be unstable with poles at $\pm 13.6j$.

Code

```
% Robust Stability
So = feedback(eye(2),G*K);
[stabmarg,wcuSM] = robstab(So); stabmarg
         LowerBound: 2.2129
         UpperBound: 2.2173
   CriticalFrequency: 13.6336

pwc = pole( usubs(So,wcuSM) )
pwc(11:12)
  0.0000 +13.6336i
  0.0000 -13.6336i
```

Robust Performance: Worst-Case Gain

The nominal gain from reference to error is $||S_o||_\infty = 1.13$ achieved at 7.13 rad/sec.

What is the largest (worst-case) gain achieved over all the modeled uncertainty? The corresponds to:

$$\sup_{||\Delta||_\infty \leq 1} \|F_U(M, \Delta)\|_\infty$$

This can be converted to a related robust stability problem. The bounds on worst-case gain are [2.1599,2.1642].

- The closed-loop gain is ≤ 2.1642 for all the modeled uncertainty.
- There is a combination of (p, Δ_1, Δ_2) that achieves a gain 2.1599 with the peak gain achieved at 8.33 rad/s.

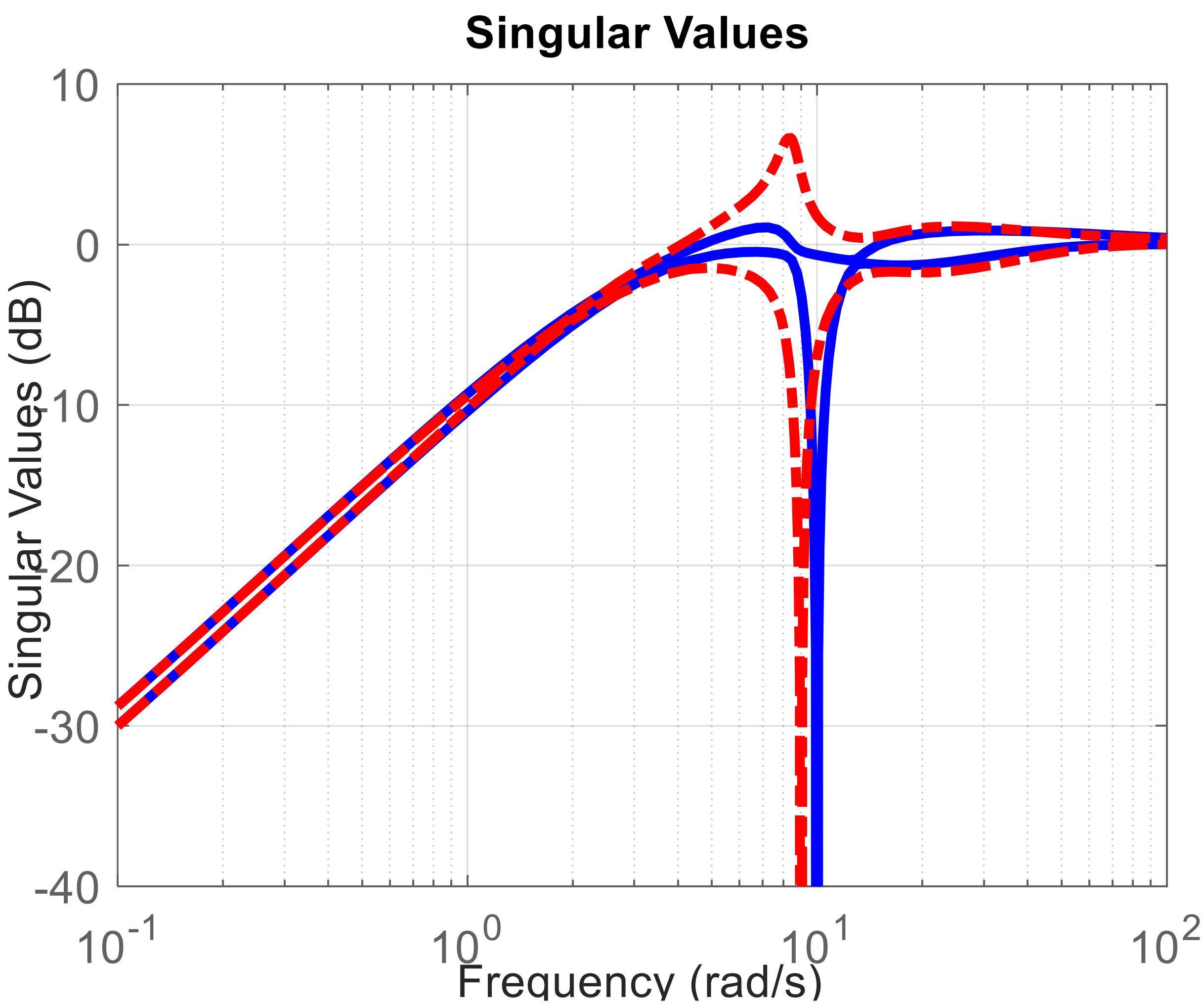
Code

```
% Worst-case gain
[maxgain,wcu] = wcgain(So); maxgain
        LowerBound: 2.1599
        UpperBound: 2.1643
   CriticalFrequency: 8.3354

% Check lower bound
[np,wp] = hinfnorm( usubs(So,wcu) )
np = 2.1599
wp =  8.3390
```

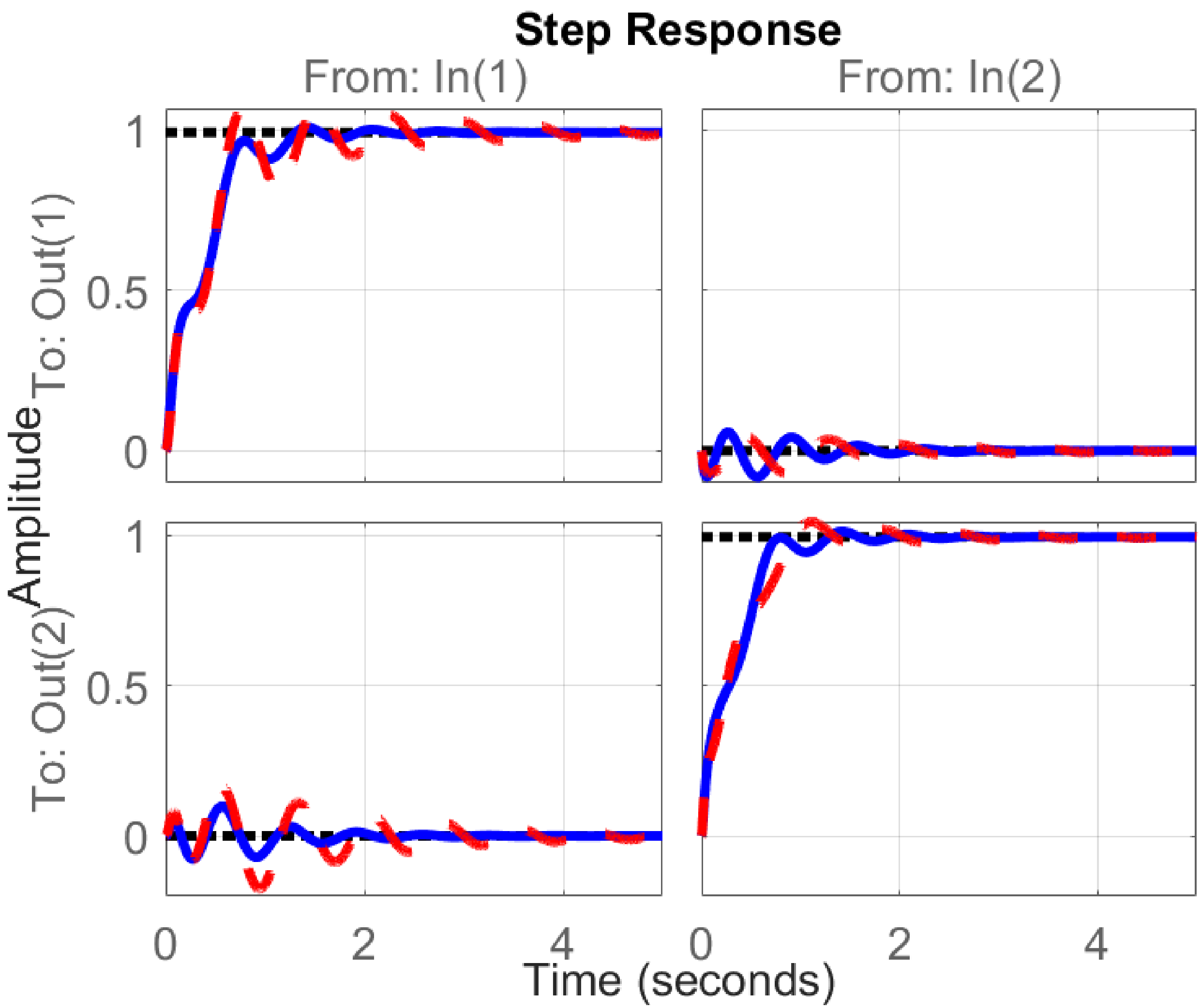
Closed-Loop Frequency Response

Frequency responses from reference to error for nominal (blue) and with worst-case parameters (red).



Step Responses

Step responses from reference to output for nominal (blue) and with worst-case parameters (red).



Code

```
% Closed-loop frequency responses from reference to error
So = feedback(eye(2),G*K);
SoNom = So.NominalValue;
SoWC = usubs(So,wcu);
sigma(SoNom,'b', SoWC,'r-.')

% Closed-loop step responses from reference to output
To = feedback(G*K,eye(2));
ToNom = To.NominalValue;
ToWC = usubs(To,wcu);
step(ToNom,'b', ToWC,'r-.',5)
```

Conclusions

This lecture presented three examples to illustrate robustness analysis using the structured singular value:

- Input margins for a MIMO system with crossfeed: We studied loop-at-a-time margins, multi-loop margins, and robustness to unstructured uncertainty.
- System with uncertain, lightly damped modes: We studied the effect of uncertainty in the real parameters associated with the modal frequency and damping ratio.
- MIMO robustness analysis (Matlab Demo): This example included parametric and dynamic uncertainty. Both robust stability and performance were analyzed.

Along the way we also described how robust performance can be assessed as a related robust stability problem.