Computing the structured distance to instability

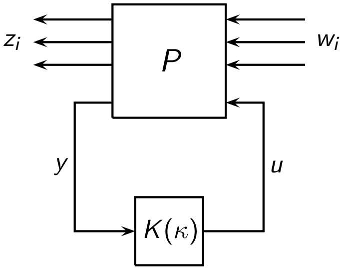
Dominikus Noll

INSTITUT de MATHEMATIQUES de TOULOUSE

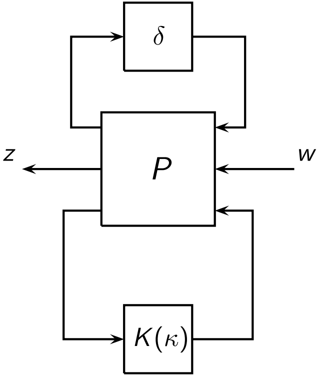
Université de Toulouse

Joint work with:
Pierre Apkarian (ONERA)
Laleh Ravanbod (IMT)
Minh Ngoc Dao (UBC)

Nominal versus parametric robust controller synthesis

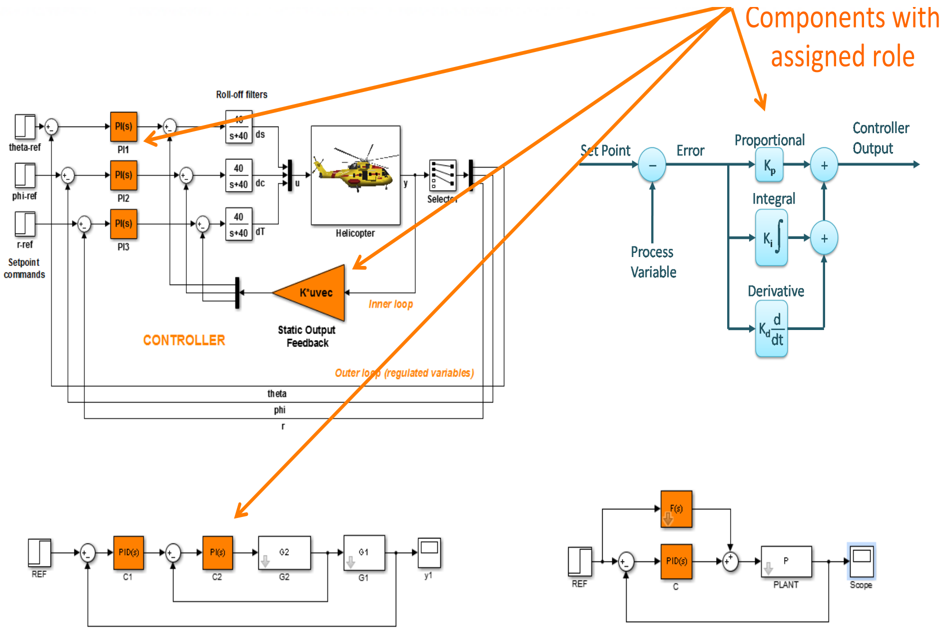


$$\begin{aligned} &\text{minimize} && \|T_{w_i z_i}(\kappa)\|_\infty \\ &\text{subject to} && K(\kappa) \text{ stabilizing} \\ & && \kappa \in \mathbb{R}^n \end{aligned}$$



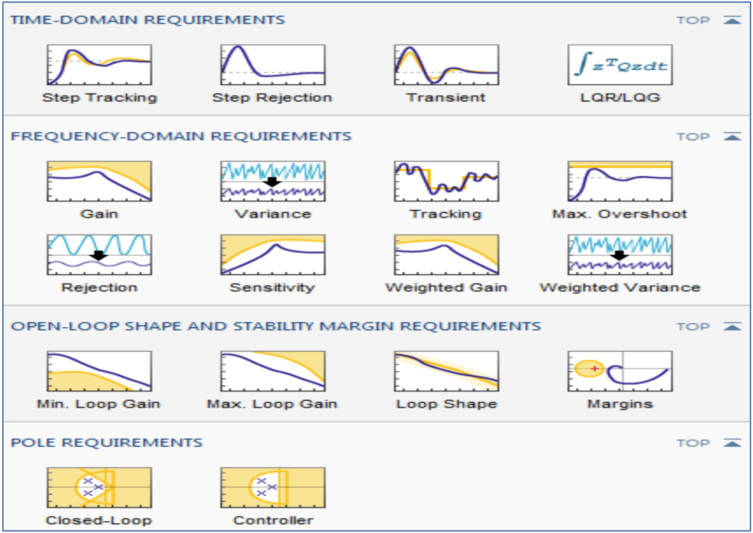
$$\begin{aligned} &\text{minimize} && \max_{\delta \in [-1,1]^m} \|T_{wz}(\kappa, \delta)\|_\infty \\ &\text{subject to} && K(\kappa) \text{ stab. } \delta \in [-1,1]^m \\ & && \kappa \in \mathbb{R}^n \end{aligned}$$

Structured versus full-order controllers


Components with assigned role
Roll-off filters
theta-ref
phi-ref
r-ref
Setpoint commands
PI(s)
PI1
PI2
PI3
40
s+40
ds
dc
dT
u
Helicopter
y
Selector
K*uvec
Inner loop
Static Output Feedback
CONTROLLER
Outer loop (regulated variables)
theta
phi
r
Set Point
Error
Proportional
Controller Output
Integral
Derivative
Process Variable
REF
PID(s)
C1
C2
G2
G1
y1
F(s)
C
P
PLANT
Scope

Why robust control?

- 1968 Kalman filter (LQG control) used in Apollo mission
- Late 1970s failure of LQG-control
- Early 1980s H_∞-problem posed (Zames, Helton, Tannenbaum)
- 1989 Doyle, Glover, Khargonekar, Francis $\Longrightarrow$ unstructured controllers
- 2006 Apkarian, Noll $\Longrightarrow$ structured controllers
- 2010 `hinfstruct` in Robust Control Toolbox (Apkarian, Noll, Gahinet)
- 2012 `systune` (Apkarian, Noll, Gahinet)



Tuning Goals

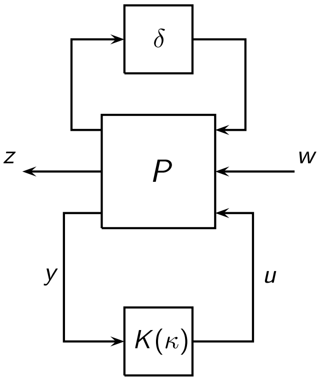
New approach adopted by leading industry

- Airbus Transportation Aircraft
- Airbus Defence & Space (Rosetta mission)
- Dassault Aviation
- Boeing
- Sagem
- CEA Robotics

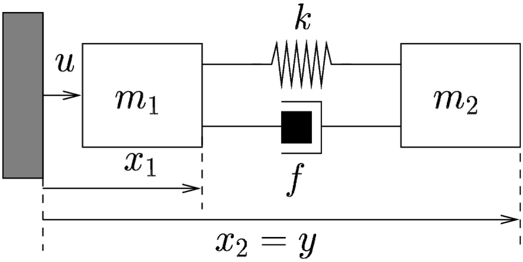
Used for teaching by academia

- Caltech
- MIT
- Supelec
- Supaero (ISEA)

Back to parametric robustness

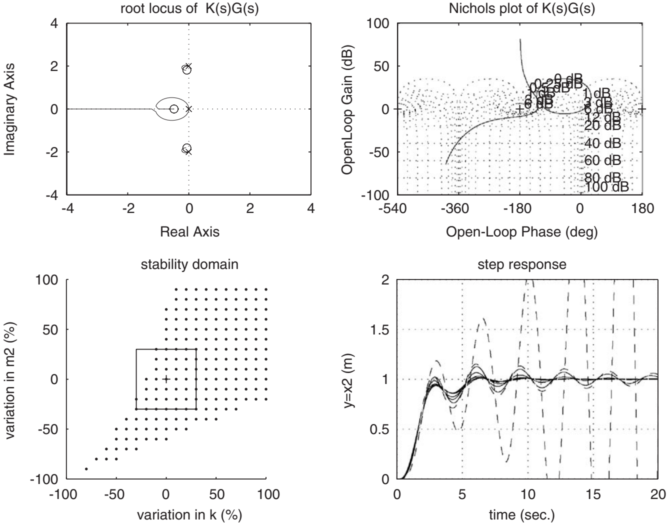


$$\begin{aligned} &\text{minimize} \quad \max_{\delta \in [-1,1]^m} \| T_{wz}(\kappa, \delta) \|_\infty \\ &\text{subject to } K(\kappa) \text{ stab. } \delta \in [-1,1]^m \\ &\qquad\qquad \kappa \in \mathbb{R}^n \end{aligned}$$

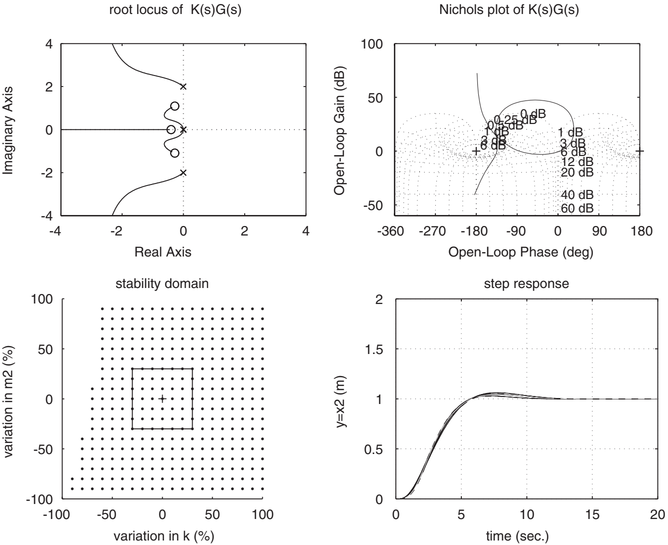


$$
\begin{aligned}
m_1 \ddot{x}_1 &= -kx_1 + kx_2 - f\dot{x}_1 + f\dot{x}_2 + u \\
m_2 \ddot{x}_2 &= \quad kx_1 - kx_2 + f\dot{x}_1 - f\dot{x}_2
\end{aligned}
$$

$$k = k^{\mathrm{nom}} + \delta_k \cdot 30\% k^{\mathrm{nom}} \qquad m_2 = m_2^{\mathrm{nom}} + \delta_{m_2} 30\% m_2^{\mathrm{nom}}.$$



root locus of K(s)G(s)
Imaginary Axis
Real Axis
Nichols plot of K(s)G(s)
OpenLoop Gain (dB)
Open-Loop Phase (deg)
stability domain
variation in m2 (%)
variation in k (%)
step response
y=x2 (m)
time (sec.)



root locus of K(s)G(s)
Imaginary Axis
Real Axis
Nichols plot of K(s)G(s)
Open-Loop Gain (dB)
Open-Loop Phase (deg)
stability domain
variation in m2 (%)
variation in k (%)
step response
y=x2 (m)
time (sec.)

Observe :

- First controller nominally stable, but not robustly stable over parameter variation square (lower left). Step responses show sustained oscillations (lower right).
- Second controller stable over parameter square.
- *But how to synthesize such robustly stable controllers?*

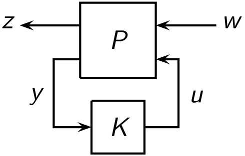
Synthesis as semi-infinite min-max program

Nominal H_∞-synthesis as semi-infinite minimization

$$\min_{\kappa} \; \| T_{wz}(\kappa) \|_\infty$$

$$\min_{\kappa} \; \max_{\omega \in [0,\infty]} \; \overline{\sigma}\left(T_{wz}(\kappa, j\omega) \right)$$

$$\min_{\kappa} \; \max_{\omega \in [0,\infty]} \; \max_{\|x\|_2 = 1} \; \| T_{wz}(\kappa, j\omega) x \|_2$$



convex ☺

non-convex, but computable
by Hamiltonian algorithm ☺

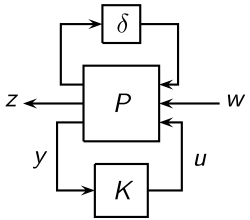
local optimization due to
structured $K = K(\kappa)$

Parametric robust H_∞-synthesis as semi-infinite program

$$\min_{\kappa} \max_{\delta\in[-1,1]^m} \|T_{wz}(\delta,\kappa)\|_\infty$$

$$\min_{\kappa} \max_{\delta\in[-1,1]^m} \max_{\omega\in[0,\infty]} \overline{\sigma}\left(T_{wz}(\delta,\kappa,j\omega)\right)$$

$$\min_{\kappa} \max_{\delta\in[-1,1]^m} \max_{\omega\in[0,\infty]} \max_{\|x\|_2=1} \|T_{wz}(\delta,\kappa,j\omega)x\|_2$$



non-convex, not computable ☹

local optimization, $K = K(\kappa)$

Relaxations
Do we want outer or inner?

Outer relaxations

$$\min_{\kappa} \max_{\delta \in [-1,1]^m} \max_{\omega \in [0,\infty]} \max_{\|x\|_2=1} \| T_{wz}(\delta, \kappa, j\omega) x \|_2$$

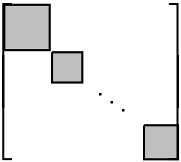
$$\min_{\kappa} \max_{\overline{\sigma}(\Delta) \leq 1} \max_{\omega \in [0,\infty]} \max_{\|x\|_2=1} \| T_{wz}(\Delta, \kappa, j\omega) x \|_2$$

$$\Delta = \begin{bmatrix} \delta_1 I_{r_1} & & & \\ & \delta_2 I_{r_2} & & \\ & & \ddots & \\ & & & \delta_m I_{r_m} \end{bmatrix}$$

Outer relaxations

$$\min_{\kappa} \max_{\bar{\sigma}(\Delta)\leq 1} \max_{\omega\in[0,\infty]} \max_{\|x\|=1} \|T_{wz}(\Delta,\kappa,j\omega)x\|_2$$

$\Delta = \text{full}$ $\Delta =$



- Over-estimation of true objective function
- Easier to compute, but ...
- Larger set of uncertainties $\Longrightarrow$ *conservative*
- Most widely known outer relaxations use μ-upper bounds.

Inner relaxations

$$\min_{\kappa} \max_{\delta\in[-1,1]^m} \max_{\omega\in[0,\infty]} \max_{\|x\|=1} \|T_{wz}(\delta,\kappa,j\omega)x\|_2$$

$$\min_{\kappa} \max_{\delta\in\Delta_d} \max_{\omega\in[0,\infty]} \max_{\|x\|=1} \|T_{wz}(\delta,\kappa,j\omega)x\|_2$$

- $\Delta_d \subset [-1,1]^m \implies$ under-estimation of true objective function
- No stability/performance certificate on $[-1,1]^m$, only on Δ_d
- Δ_d typically finite, yet works on $[-1,1]^m$. Certificate ?

Discussion: inner versus outer relaxation

Observe:

- People concede that inner relaxations work better in practice, *but* insist that outer relaxations are theoretically sounder, as when work give robust stability certificate.

Disenchantment:

♠ Not true because outer relaxations do not come with a guarantee of success.

♠ Without certificate, both approaches are at equal rights theoretically, hence the better in practice wins.

♡ Therefore inner relaxation wins.

In the same vein (still trying to make us believe that outer approximations are better):

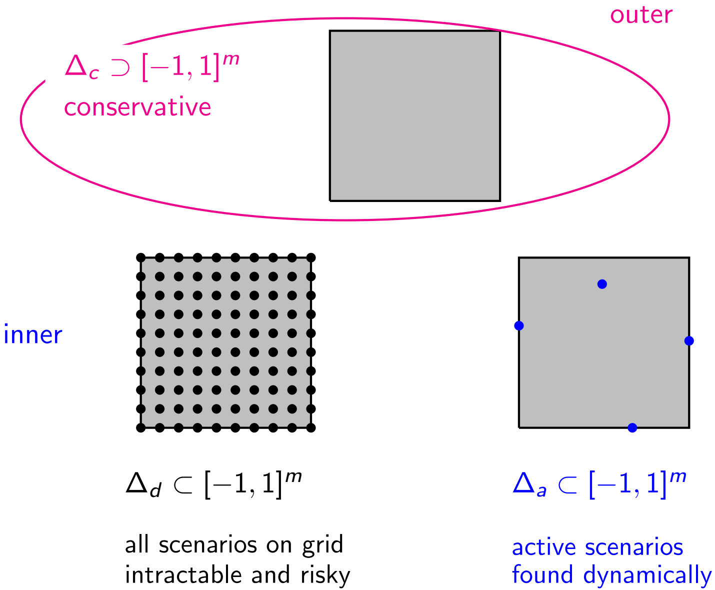
- Cannot we just degrade our performance specification more and more until it becomes possible to obtain a robustly stable controller? And having obtained this certificate, is this then not an advantage over the inner approximation ?

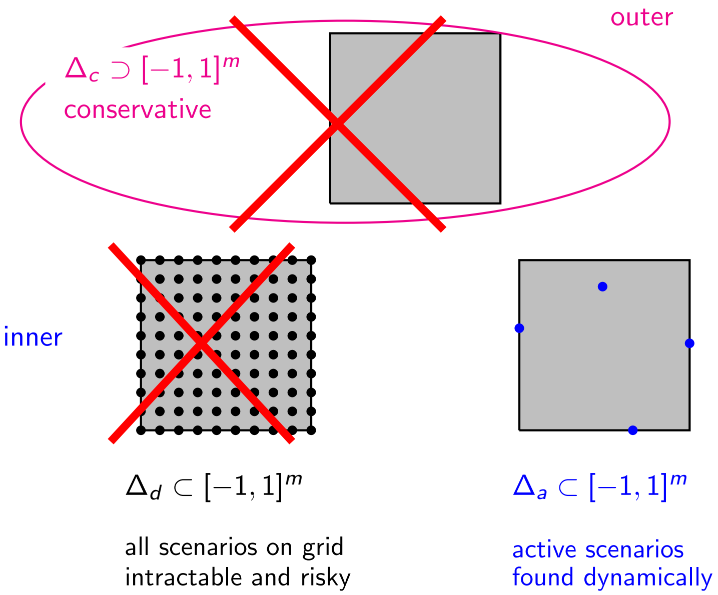
Disenchantment:

♠ No, this is not theoretically better, unless there is a *guarantee* that the degradation of performance leads to the certificate. (No case where this holds is known by the way).

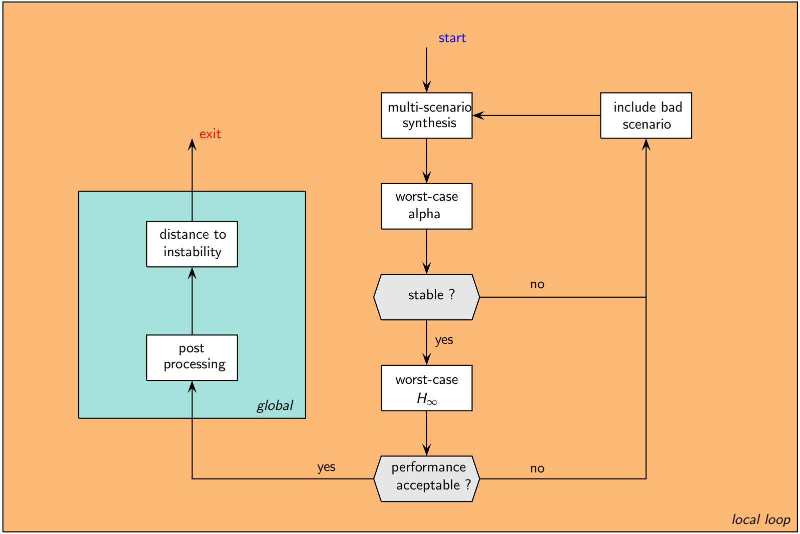
♠ The fact that LMI people present certifcates obtained by degrading does not mean anything. It just means they give in at an early stage and accept a feeble result.

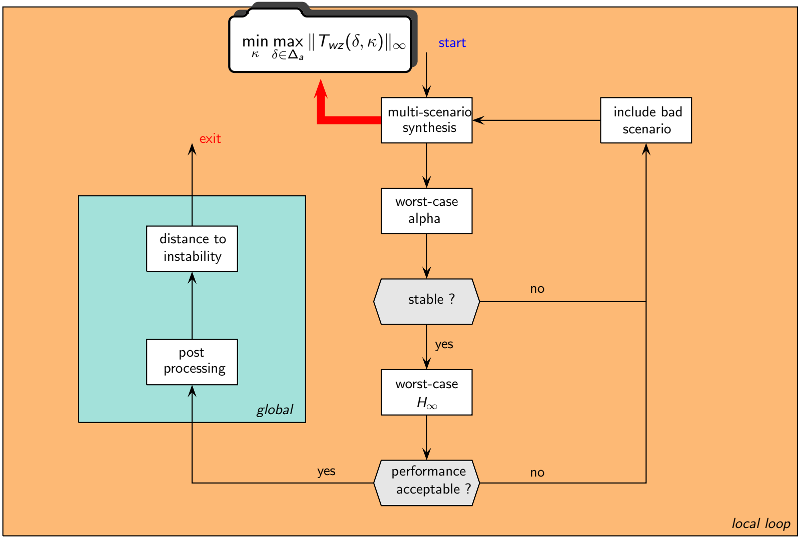
♡ Certificates obtained by degrading performance are useless anyway, as performance is degraded. We can do better.


outer
$\Delta_c \supset [-1,1]^m$
conservative
inner
$\Delta_d \subset [-1,1]^m$
all scenarios on grid
intractable and risky
$\Delta_a \subset [-1,1]^m$
active scenarios
found dynamically

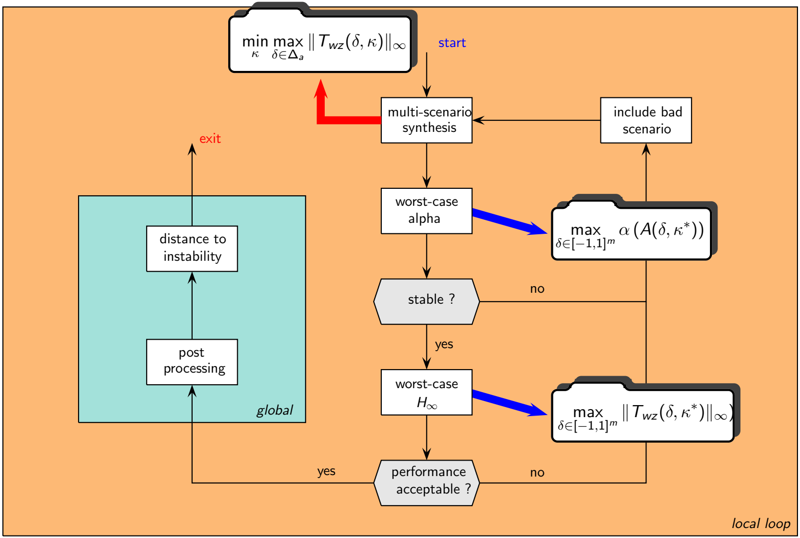


outer
$\Delta_c \supset [-1,1]^m$
conservative
inner
$\Delta_d \subset [-1,1]^m$
all scenarios on grid
intractable and risky
$\Delta_a \subset [-1,1]^m$
active scenarios
found dynamically

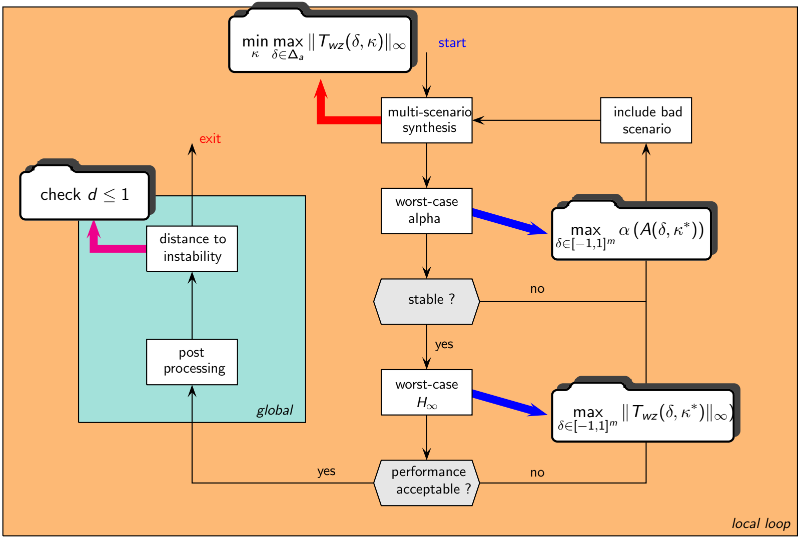

start
multi-scenario synthesis
include bad scenario
exit
worst-case alpha
distance to instability
stable ?
no
yes
post processing
worst-case H_∞
global
yes
performance acceptable ?
no
local loop



$\min_{\kappa} \max_{\delta\in\Delta_a} \|T_{wz}(\delta,\kappa)\|_\infty$
start
multi-scenario synthesis
include bad scenario
exit
worst-case alpha
distance to instability
stable ?
no
yes
worst-case H_∞
post processing
global
yes
performance acceptable ?
no
local loop



$\min_{\kappa} \max_{\delta\in\Delta_a} \|T_{wz}(\delta,\kappa)\|_\infty$
start
multi-scenario synthesis
include bad scenario
exit
worst-case alpha
$\max_{\delta\in[-1,1]^m} \alpha(A(\delta,\kappa^*))$
distance to instability
stable ?
no
yes
post processing
worst-case H_∞
$\max_{\delta\in[-1,1]^m} \|T_{wz}(\delta,\kappa^*)\|_\infty$
global
yes
performance acceptable ?
no
local loop



$\min_{\kappa} \max_{\delta \in \Delta_a} \|T_{wz}(\delta,\kappa)\|_\infty$
start
multi-scenario synthesis
include bad scenario
exit
check $d \leq 1$
worst-case alpha
$\max_{\delta \in [-1,1]^m} \alpha(A(\delta,\kappa^*))$
distance to instability
stable ?
no
yes
post processing
worst-case H_∞
$\max_{\delta \in [-1,1]^m} \|T_{wz}(\delta,\kappa^*)\|_\infty$
global
yes
performance acceptable ?
no
local loop

Instability optimization

Spectral abscissa:

$$\alpha(A) = \max\{\mathrm{Re}\,\lambda : \lambda \text{ eigenvalue of } A\}$$

Stability:

$$A \text{ stable} \iff \alpha(A) < 0$$

Destabilize by bad parameter scenario:

$$\alpha^* = \max_{\delta \in [-1,1]^m} \alpha\left(A(\delta, \kappa^*)\right)$$

- If $\alpha^* < 0$ then stable for all $\delta \in [-1, 1]^m$; otherwise bad scenario.
- $\delta \mapsto \alpha\left(A(\delta, \kappa^*)\right)$ non-smooth and not locally Lipschitz

Spectral abscissa:

$$\alpha(A) = \max\{\mathrm{Re}\,\lambda : \lambda \text{ eigenvalue of } A\}$$

Stability:

$$A \text{ stable} \iff \alpha(A) < 0$$

Destabilize by bad parameter scenario:

$$\alpha^* = \max_{\delta \in [-1,1]^m} \alpha\left(A + B\Delta(I - D\Delta)^{-1}C\right)$$

- If $\alpha^* < 0$ then stable for all $\delta \in [-1, 1]^m$; otherwise bad scenario.
- $\delta \mapsto \alpha\left(A(\delta, \kappa^*)\right)$ non-smooth and not locally Lipschitz

Structured distance to instability:

$$d^* = \inf\{\|\delta\|_\infty : A + B\Delta(I - D\Delta)^{-1}C \text{ unstable}\}$$
$$= \sup\{d : A + B\Delta(I - D\Delta)^{-1}C \text{ stable for all } \|\delta\|_\infty \le d\}$$

- $d^* \le 1 \implies$ parametric robust stability over $[-1, 1]^m$
- Need global optimum

Not to confuse with unstructured distance to instability

$\beta(A) = \inf\{\overline{\sigma}(E) : A + E \text{ unstable}\}$

(Trefethen, Kressner, Kanzow, Benner, ...) is too easy

Branch and bound

Global maximum

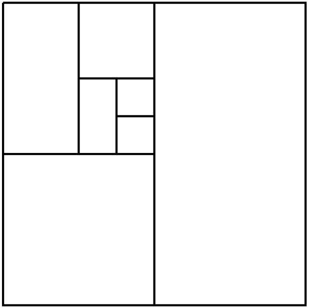
$$\alpha^* = \max_{\delta \in [-1,1]^m} \alpha\left(A(\delta)\right)$$

Lower bound by local solver

$$\underline{\alpha} = \max_{\delta \in [-1,1]^m} \alpha\left(A(\delta)\right) \leq \alpha^*$$

Upper bound on box Δ

$$\alpha^*(\Delta) = \max_{\delta \in \Delta} \alpha\left(A(\delta)\right)$$



Pruning test:

$$\alpha^*(\Delta) \leqslant \underline{\alpha} \implies \Delta \text{ can be pruned}$$

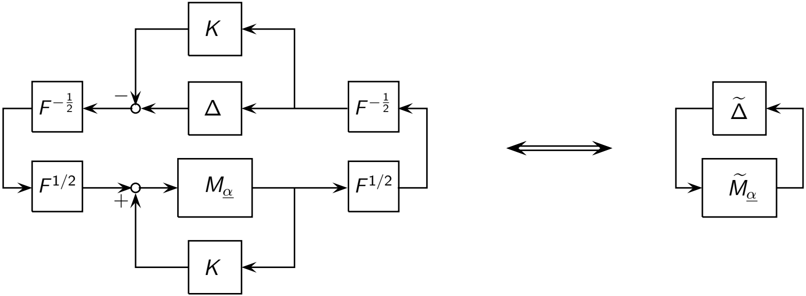
☺

Branching:

If Δ not pruned, then halved

☹

Crucial elements:



- Loop transform: Test $\alpha^*(\Delta) \leq 1 \iff \alpha^* \leq 1$ for $M_{\underline{\alpha}}$.
- Then use conservative μ-upper bound (Fan, Tits, Doyle).
- Can store partial stability on frequency bands for daughter boxes (frequency axis sweep).
- Need high performance local solver to get good $\underline{\alpha}$.

Some information about the dimension of problems

Experiments:

Technique	States	decision variables κ	uncertain parameters δ	repetitions	CPU
nominal H_∞ synthesis	200	50	-	-	seconds
nominal multi-objective	200	20	-	-	seconds
parametric robust H_∞	25	20	10	8	seconds to minutes

Technique	States	decision variables δ	repetitions	CPU
worst-case H_∞	35	11	6	seconds
worst-case α^*	35	10	6	seconds
distance d^* to instability	70	14	39	seconds

Technique	States	decision variables δ	repetitions	CPU
branch & bound H_∞	35	11	6	seconds to hours
branch & bound α^*	35	12	6	seconds to hours
branch & bound d^*	70	14	39	seconds to hours

Existing tools used to check results:

method	type	appreciation	bottleneck
Integral Global Optimization Zheng method	probabilistic global	fast, reliable	-
SOS tools Parillo	global	useless	-
Lasserre's method	global	works for toy problems	is its own bottleneck
branch & bound Balak. & Boyd	global	slow	inefficient lower bound
`wcgain`	computes h^*	not always reliable	dedicated
SMAC Toolbox ONERA	computes d^*	fast, reliable	dedicated
`dksyn`	parametric robust K	conservative	controllers often not practical
hifoo	nominal H_∞	-	not all controller structures

Conclusions for nominal H_∞-synthesis:

- Solved 2006: `hinfstruct`, `systune`
- Solving LMIs, Riccati equations practically *obsolete*.
- Hamiltonian linear algebra stunted to computing H_∞-norm.
- Non-smooth optimization techniques prevail.

Conclusions for parametric robust H_∞-synthesis:

- Solved 2015: $\Longrightarrow$ *robust control toolbox*
- Non-smooth optimization techniques again key to success.
- μ-singular value upper bounds still needed for pruning test in B&B.
- Lower bounds by fast non-smooth solver.
- DK-iteration (`dksyn`) outdated.
- Inner approximation beats outer approximation.