Robust Multi-Objective Control for Linear Systems

Elements of theory and RoMulOC toolbox

Ecole des JD-MACS Juin 2011

Denis Arzelier & Dimitri Peaucelle

LAAS-CNRS, Toulouse, FRANCE

also with Alberto Bortott, Guilherme Chevarria, Yoshio Ebihara, Maud Sevin, Philippe Spiesser...

www.laas.fr/OLOCEP/romuloc/downloads.html

- **Robust control theory**
  - Robustness properties of the feedback loop
  - Aim for guaranteed properties (stability and performances)
  - Uncertain modeling of systems: tradeoff between complexity of systems & simplicity of models
- **Optimization based tools**
  - Linear Matrix Inequalities (LMI) framework [1990's]
  - Efficient fast solvers and nice parser for Matlab [2000's]
  - Possibility of a tool gathering established results : RoMulOC

1. Uncertain LTI systems and performances
   - Control objectives: stability, transient response, perturbation rejection...
   - Structured parametric uncertainties: extremal values, bounded sets...
2. LMIs and convex polynomial-time optimization
   - Semi-Definite Programming and LMIs
   - SDP solvers and parsers
3. Conservative LMI results
   - Methods: Lyapunov, S-procedure, Finsler lemma, Topologic Separation...
   - The ROMULOC toolbox

Linear Time-Invariant State-Space Multi-Input Multi-Output models

$$\begin{array}{rl} \vartheta[x](t) = & Ax(t) + B_u u(t) \\ y(t) = & C_y x(t) + D_{yu} u(t) \end{array} \; , \quad \begin{array}{rl} \vartheta[\eta](t) = & K_A \eta(t) + K_B y(t) \\ u(t) = & K_C \eta(t) + K_D y(t) \end{array}$$

$$x \in \mathbf{C}^n \quad u \in \mathbf{C}^{q_u} \quad y \in \mathbf{C}^{p_y} \quad \eta \in \mathbf{C}^{n_K}$$

- Continuous ($\vartheta[x](t) = \dot{x}(t)$) and discrete-time ($\vartheta[x](t) = x(t+T)$)

analysis problem: For given (K_A, K_B, K_C, K_D) prove closed-loop properties of

$$\vartheta \begin{bmatrix} x \\ \eta \end{bmatrix} (t) = A(K) \left(\begin{array}{c} x \\ \eta \end{array} \right) (t)$$

Design problem: Find (K_A, K_B, K_C, K_D) providing closed loop properties

- For $n_k = 0$: Static output-feedback (SOF)
- For $n_k = 0$ and $y = x$: State feedback problem
- For $n_k = n$: full order output-feedback problem

Control objectives

- Stability of $\vartheta[x](t) = A(K)x(t)$

  - For continuous-time: poles are all in left-hand half of complex plane
  - For discrete-time: poles are all in unit circle of complex plane

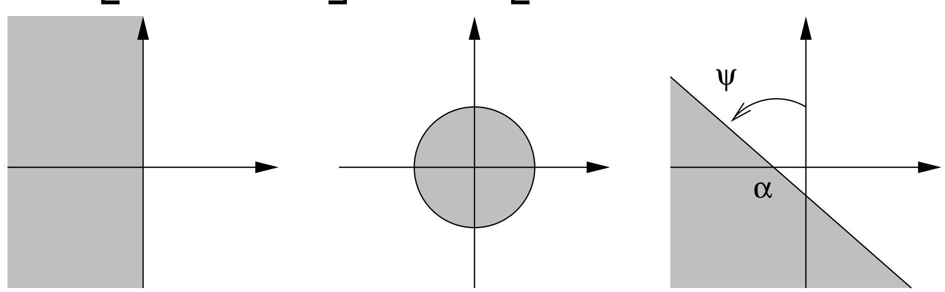
- D_R-Stability of $\vartheta[x](t) = A(K)x(t)$:

  - Poles are in region defined by

$$D_R = \{ \, s \in \mathbb{C} \ : \ r_{11} + s r_{12} + s^* r_{12} + s s^* r_{22} \leq 0 \, \} \quad , \quad R = (r_{ij})$$

Such regions are half-planes and discs

$$R = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \; R = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \; R = \begin{bmatrix} -2\alpha \cos\psi & \cos\psi - i \sin\psi \\ \cos\psi + i\sin\psi & 0 \end{bmatrix}$$



Input/output objectives

$$\begin{array}{rlll} \vartheta[x](t) = & A(K)x(t) + & B_w(K)w(t) & \quad : \quad w \in \mathbf{C}^{q_w} \\ z(t) = & C_z(K)x(t) + & D_{zw}(K)w(t) & \quad\quad\; z \in \mathbf{C}^{p_z} \end{array}$$

- Induced L_2 gain: $\|z\| \leq \gamma_\infty \|w\|$, $(\|w\|^2 = \int_0^\infty w^* w dt)$

  Also known as: H_∞ performance (max singular value $H(j\omega)$, $\omega \in \mathbf{R}$),

  Robustness to unmodeled dynamics $w = \Delta z$, $\|\Delta\| \leq 1/\gamma_\infty$ (bounded-real lemma)

- Impulse-to-norm performance: $\|z\| \leq \gamma_2$ if $w(t) = \delta(t)1$

  Also known as: H_2 performance (mean value $H(j\omega)$, $\omega \in \mathbf{R}$),

  Energy of output in response to Gaussian white noise

  Norm-to-peak performance ($\max |z| \leq \gamma_2 \|w\|$, $z \in \mathbf{R}$)

- Impulse-to-peak performance: $\max |z| \leq \gamma_{\text{i2p}}$ if $w(t) = \delta(t)\alpha$, $\|\alpha\| \leq 1$.

  Also known as: Invariant ellipsoids

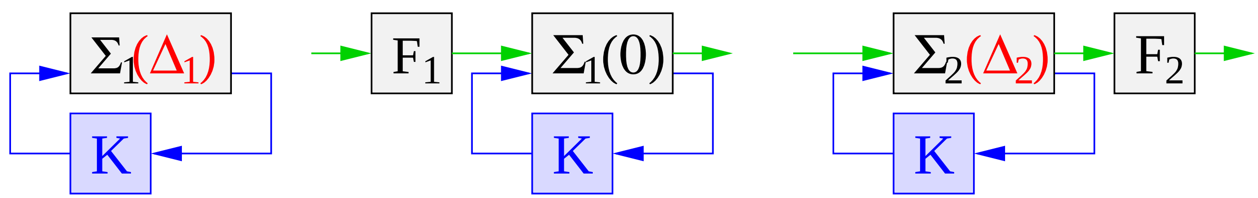
  Non saturating initial conditions

Robust Multi-Objective Control

- Δ: errors in modeling, operating conditions, mass-production...
- Δ: parametric uncertainty, assumed constant, belongs to a set $\mathbf{\Delta}$.

$$
\begin{aligned}
\vartheta[x](t) = &\quad A(\Delta, K)x(t) + &\quad B_w(\Delta, K)w(t) \\
z(t) = &\quad C_z(\Delta, K)x(t) + &\quad D_{zw}(\Delta, K)w(t)
\end{aligned}
$$

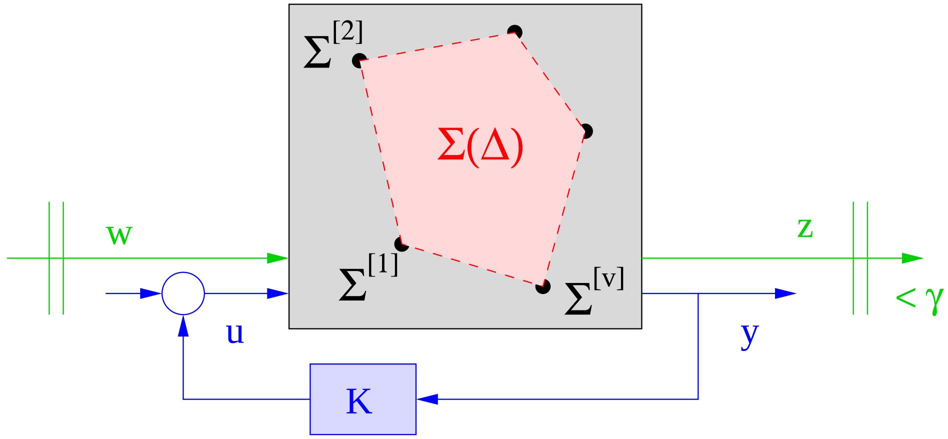
Design: Find a controller K that fulfills all robust specifications $\Pi_{p=1\ldots\bar{p}}$ defined for models $\Sigma_p(\Delta_p)$ subject to uncertainties $\Delta_p \in \mathbf{\Delta}_p$.



Analysis: For given K prove for each $\Sigma_{p=1\ldots\bar{p}}(\Delta_p)$ that the specification Π_p holds for all uncertainties $\Delta_p \in \mathbf{\Delta}_p$.

■ Uncertain LTI systems: Affine with scalar parametric uncertainty

● Polytopic models



Convex hull of $\bar{v}$ vertices

$$A(\Delta) = \sum_{v=1}^{\bar{v}} \xi_v A^{[v]} \ , \ \ B_w(\Delta) = \sum_{v=1}^{\bar{v}} \xi_v B_w^{[v]} \ \ \ldots \qquad : \ \xi_v \geq 0 \ , \ \ \sum_{v=1}^{\bar{v}} \xi_v = 1$$

▲ Example: Linear combination of linear models identified on different operating points.

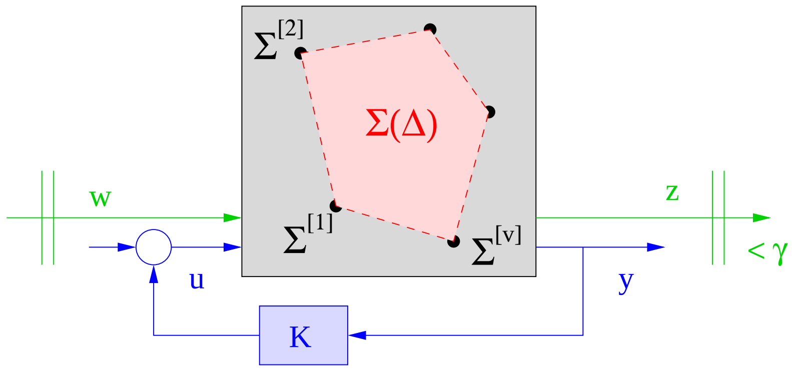
▲ The ξ_v parameters may not have physical meaning

▲ $\bar{v}$ vertices can define a volume in $\bar{v} - 1$ space of parameters

(possible to divide space in polytopes with low number of vertices)

Uncertain LTI systems: Affine with scalar parametric uncertainty

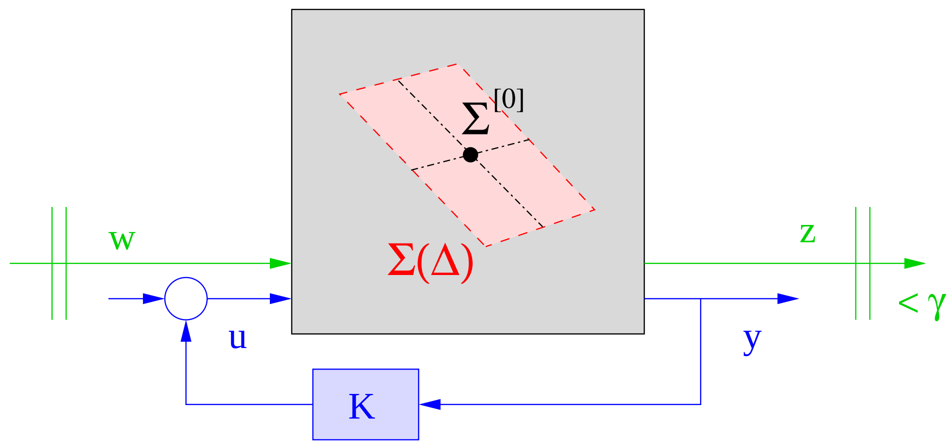
- Polytopic models



Convex hull of $\bar{v}$ vertices

$$A(\Delta) = \sum_{v=1}^{\bar{v}} \xi_v A^{[v]} \;, \quad \ldots \quad : \; \xi_v \geq 0 \;, \; \sum_{v=1}^{\bar{v}} \xi_v = 1$$

- Parallelotopic models with $\bar{\varsigma}$ axes

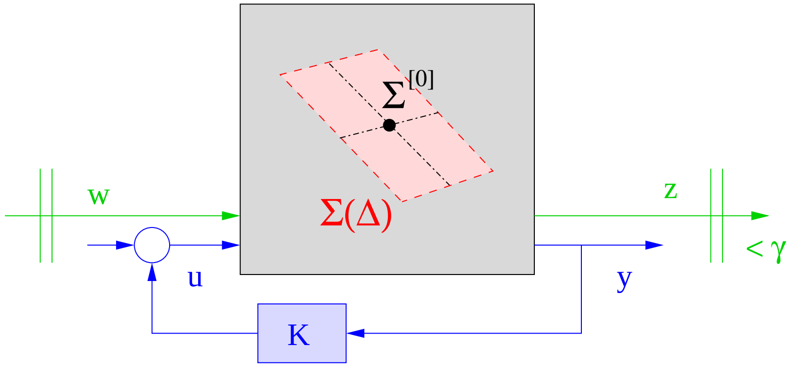


$$A(\Delta) = A^{[0]} + \sum_{\varsigma=1}^{\bar{\varsigma}} \delta_\varsigma A^{[\varsigma]} \;, \quad \ldots \quad : \; |\delta_\varsigma| \leq 1$$

- $\bar{\varsigma}$ independent parameters δ_ς identified in intervals
- polytope with $\bar{v} = 2^{\bar{\varsigma}}$ vertices

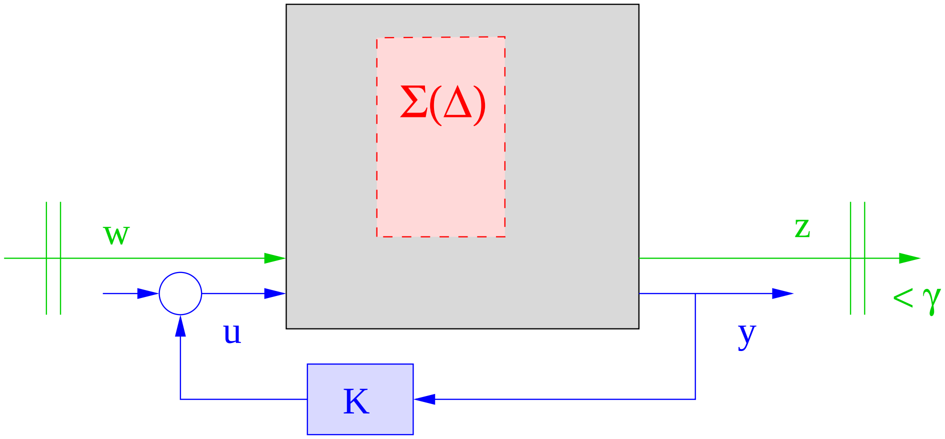
Uncertain LTI systems: Affine with scalar parametric uncertainty

- Parallelotopic models with $\bar{\varsigma}$ axes



$$A(\Delta) = A^{[0]} + \sum_{\varsigma=1}^{\bar{\varsigma}} \xi_\varsigma A^{[\varsigma]} \ , \quad \ldots \quad : \ |\xi_\varsigma| \leq 1$$

- Interval models with $\bar{\varsigma}$ non-equal coefficients



$$\underline{A} \preceq A(\Delta) \preceq \overline{A} \ : \ \underline{a}_{ij} \leq a_{ij}(\Delta) \leq \overline{a}_{ij} \quad \ldots$$

- Parallelotope with axes in the euclidian basis of matrices
- All coefficients independent
- Change of basis does not preserve the structure

- Polytopic models can also be written as

$$A(\Delta) = A + \underbrace{\begin{bmatrix} B_\Delta^{[1]} & \dots & B_\Delta^{[\bar{v}]} \end{bmatrix}}_{B_\Delta} \underbrace{\begin{bmatrix} \xi_1 \mathbf{1}_{q_1} & & 0 \\ & \ddots & \\ 0 & & \xi_{\bar{v}} \mathbf{1}_{q_{\bar{v}}} \end{bmatrix}}_{\Delta} \underbrace{\begin{bmatrix} C_\Delta^{[1]} \\ \vdots \\ C_\Delta^{[\bar{v}]} \end{bmatrix}}_{C_\Delta}$$

where for each vertex $A^{[v]} = A + B_\Delta^{[v]} C_\Delta^{[v]}$ with $B_\Delta^{[v]} \in \mathbf{C}^{n \times q_v}$.

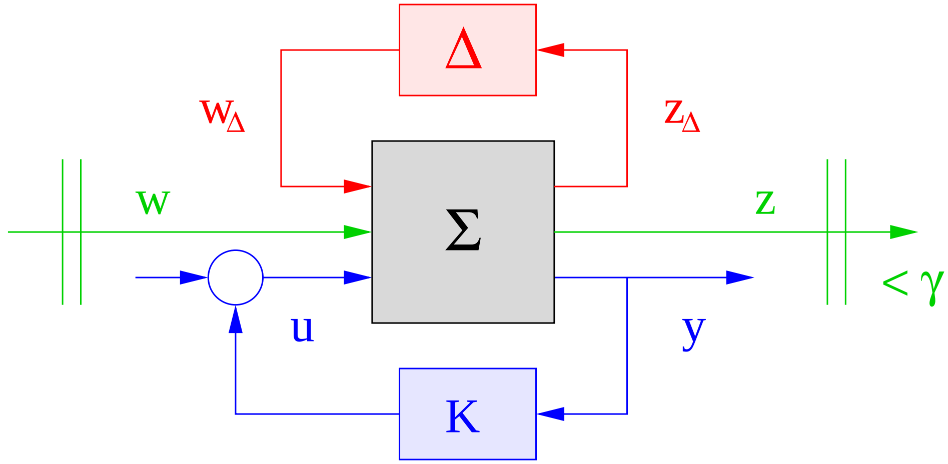
- Parallelotopic models can also be written as

$$A(\Delta) = \underbrace{A^{|0|}}_{A} + \underbrace{\begin{bmatrix} B_\Delta^{|1|} & \dots & B_\Delta^{|\bar{\varsigma}|} \end{bmatrix}}_{B_\Delta} \underbrace{\begin{bmatrix} \delta_1 \mathbf{1}_{p_1} & & 0 \\ & \ddots & \\ 0 & & \delta_{\bar{\varsigma}} \mathbf{1}_{p_{\bar{\varsigma}}} \end{bmatrix}}_{\Delta} \underbrace{\begin{bmatrix} C_\Delta^{|1|} \\ \vdots \\ C_\Delta^{|\bar{\varsigma}|} \end{bmatrix}}_{C_\Delta}$$

where for each axis $A^{|\varsigma|} = B_\Delta^{|\varsigma|} C_\Delta^{|\varsigma|}$ with $B_\Delta^{|\varsigma|} \in \mathbf{C}^{n \times p_\varsigma}$.

⚠ Factorisation as $A(\Delta) = A + B_\Delta \Delta C_\Delta$ is not unique.

Uncertain LTI systems: Linear Fractional Representation (LFR)



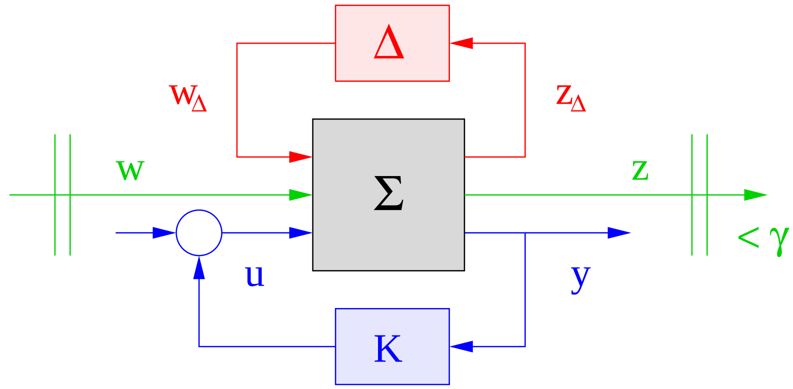
$$\begin{array}{rcccccccc} \vartheta[x](t) = & Ax(t) + & B_\Delta w_\Delta(t) + & B_w w(t) + & B_u u(t) & & \\ z_\Delta(t) = & C_\Delta x(t) + & D_{\Delta\Delta} w_\Delta(t) + & D_{\Delta w} w(t) + & D_{\Delta u} u(t) & \vdots & w_\Delta \in \mathbb{C}^{q_\Delta} \\ z(t) = & C_z x(t) + & D_{z\Delta} w_\Delta(t) + & D_{zw} w(t) + & D_{zu} u(t) & & z_\Delta \in \mathbb{C}^{p_\Delta} \\ y(t) = & C_y x(t) + & D_{y\Delta} w_\Delta(t) + & D_{yw} w(t) + & D_{yu} u(t) & & \end{array}$$

Linear - Fractional Transformation (LFT):

$$A(\Delta) = A + B_\Delta \Delta (1 - D_{\Delta\Delta}\Delta)^{-1} C_\Delta = A + B_\Delta (1 - \Delta D_{\Delta\Delta})^{-1} \Delta C_\Delta,$$

$$B_w(\Delta) = B_w + B_\Delta \Delta (1 - D_{\Delta\Delta}\Delta)^{-1} D_{\Delta w} \quad \ldots$$

Uncertain LTI systems: Linear Fractional Representation (LFR)



$$A(\Delta) = A + B_\Delta \Delta (1 - D_{\Delta\Delta}\Delta)^{-1} C_\Delta = A + B_\Delta (1 - \Delta D_{\Delta\Delta})^{-1} \Delta C_\Delta \quad \ldots$$

- For any model where $A(\Delta), \ldots$ are rational functions of δ_j the LFT exists
- Δ can always be taken as a bloc-diagonal matrix with repeated blocs

$$\Delta = \begin{bmatrix} 1_{r_1} \otimes \Delta_1 & & 0 \\ & \ddots & \\ 0 & & 1_{r_{\bar{j}}} \otimes \Delta_{\bar{j}} \end{bmatrix}$$

- For scalar uncertainties $1_{r_j} \otimes \delta_j = \delta_j 1_{r_j}$
- LFR are not unique

Sets of uncertainties in LFRs

- $\{X, Y, Z\}$−dissipative uncertain matices

$$\left\{ \ \Delta_j \ : \ X + Y\Delta_j + \Delta_j^* Y^* + \Delta_j^* Z \Delta_j \leq 0 \ , \ X \leq 0 \ , \ Z \geq 0 \ \right\}$$

- Norm-bounded : $\|\Delta_j\| \leq \rho 1$ (gain limited operators) $\Rrightarrow \{-\rho^2 1, 0, 1\}$−dissipative
- Positive real : $\Delta_j + \Delta_j^* \geq 0$ (passive operators) $\Rrightarrow \{0, -1, 0\}$−dissipative

Sets of uncertainties in LFRs

- $\{X, Y, Z\}-$dissipative uncertain matices

$$\left\{ \ \Delta_j \ : \ X + Y\Delta_j + \Delta_j^* Y^* + \Delta_j^* Z \Delta_j \leq 0 \ , \ X \leq 0 \ , \ Z \geq 0 \ \right\}$$

- Norm-bounded : $\|\Delta_j\| \leq \rho 1$ (gain limited operators) $\Rrightarrow$ $\{-\rho^2 1, 0, 1\}-$dissipative
- Positive real : $\Delta_j + \Delta_j^* \geq 0$ (passive operators) $\Rrightarrow$ $\{0, -1, 0\}-$dissipative

- Polytopic uncertainties

$$\left\{ \ \Delta_j = \sum \xi_{j,v} \Delta_j^{[v]} \ : \ \xi_{j,v} \geq 0 \ , \ \sum \xi_{j,v} = 1 \ \right\}$$

- Parallelotopic uncertainties

$$\left\{ \ \Delta_j = \Delta_j^{|0|} + \sum \delta_{j,i} \Delta_j^{|i|} \ : \ |\delta_{j,i}| \leq 1 \ \right\}$$

- Interval uncertainties

$$\left\{ \ \underline{\Delta}_j \preceq \Delta_j \preceq \overline{\Delta}_j \ \right\}$$

Example

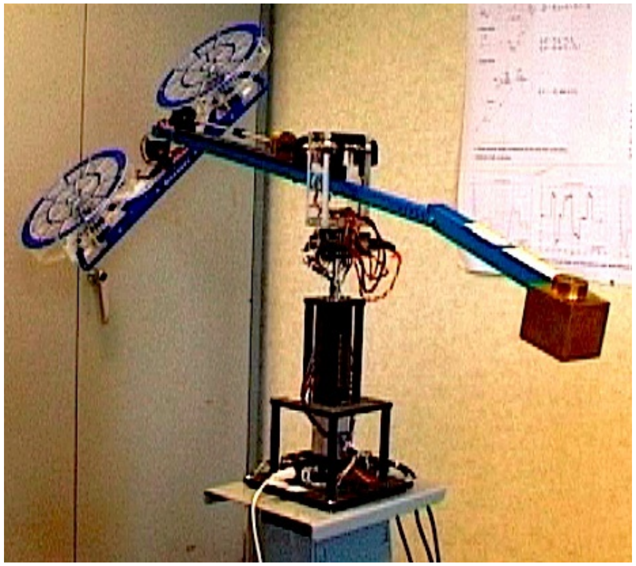
$$\dot{x} = \frac{\delta_1{}^2}{1+\delta_2}x \Rightarrow \begin{cases} \dot{x} = w_{\Delta 1} \\ z_{\Delta 1} = \frac{\delta_1}{1+\delta_2}x \end{cases} \quad w_{\Delta 1} = \delta_1 z_{\Delta 1}$$

$$\Rightarrow \begin{cases} \dot{x} = w_{\Delta 1} \\ z_{\Delta 1} = w_{\Delta 2} \\ z_{\Delta 2} = \frac{1}{1+\delta_2}x \end{cases} \quad \begin{array}{l} w_{\Delta 1} = \delta_1 z_{\Delta 1} \\ w_{\Delta 2} = \delta_1 z_{\Delta 2} \end{array}$$

$$\Rightarrow \begin{cases} \dot{x} = w_{\Delta 1} \\ z_{\Delta 1} = w_{\Delta 2} \\ z_{\Delta 2} = x - w_{\Delta 3} \\ z_{\Delta 3} = x - w_{\Delta 3} \end{cases} \quad \begin{array}{l} w_{\Delta 1} = \delta_1 z_{\Delta 1} \\ w_{\Delta 2} = \delta_1 z_{\Delta 2} \\ w_{\Delta 3} = \delta_2 z_{\Delta 3} \end{array}$$

- $z_{\Delta 3} = z_{\Delta 2}$ added to have Δ diagonal
- δ_1 repeated twice
- δ_1, δ_2 if independent can be defined in two intervals, or as norm-bounded
- δ_1, δ_2 if dependent can be defined in polytope

- "Helicopter“ example



- System defined at maximal value of parameters

```
>> sysmax = ssmodel( 'Helicopter' );
>> sysmax.A = [0 1 0 ;0 0 1;0 -2.8 -0.14];
>> sysmax.Bw = [0;0;-14];
>> sysmax.Bu = [0;0;8];
>> sysmax.Dzu = 1
name: Helicopter
               n=3     mw=1     mu=1
 n=3     dx   =   A*x +  Bw*w +  Bu*u
pz=1      z   =                  Dzu*u
continuous time ( dx : derivative operator )
```

- System defined at maximal value of parameters

```
>> sysmax = ssmodel( 'Helicopter' );
>> sysmax.A = [0 1 0 ;0 0 1;0 -2.8 -0.14];
>> sysmax.Bw = [0;0;-14];
>> sysmax.Bu = [0;0;8];
>> sysmax.Dzu = 1;
```

- System defined at minimal value of parameters

```
>> sysmin = ssmodel( 'Helicopter' );
>> sysmin.A = [0 1 0 ;0 0 1;0 -3 -0.2];
>> sysmin.Bw = [0;0;-14];
>> sysmin.Bu = [0;0;8];
>> sysmin.Dzu = 1
name: Helicopter
                n=3    mw=1    mu=1
 n=3    dx   =   A*x +  Bw*w +  Bu*u
pz=1     z   =                 Dzu*u
continuous time ( dx : derivative operator )
```

- Uncertain system defined as interval of max and min

```
>> usys = uinter( sysmin, sysmax )
Uncertain model : interval 2 param
-------- WITH --------
name: Helicopter
               n=3    mw=1    mu=1
 n=3    dx  =   A*x +  Bw*w +  Bu*u
pz=1     z  =                 Dzu*u
continuous time ( dx : derivative operator )
```

- Interval model converted to polytopic model

```
>> usys = u2poly( usys )
Uncertain model : polytope 4 vertices
-------- WITH --------
name: Helicopter
               n=3    mw=1    mu=1
 n=3    dx  =   A*x +  Bw*w +  Bu*u
pz=1     z  =                 Dzu*u
continuous time ( dx : derivative operator )
```

- Declare a state-feedback design problem

```
>> quiz = ctrpb( 'state-feedback', 'Lyap-unique' )
control problem: STATE-FEEDBACK design
Lyapunov function: UNIQUE (quadratic stability)
No specified performance
```

- Add an H_∞ performance objective

```
>> quiz = quiz + hinfty( usys, 4 );
```

- Add a pole location performance objective

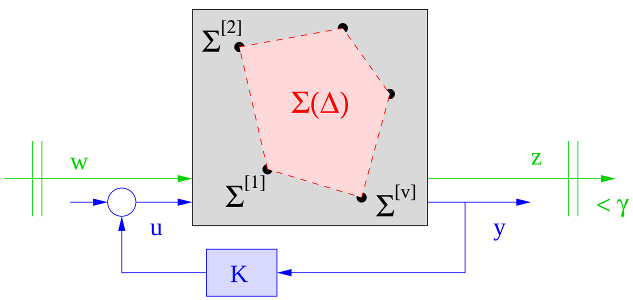
```
>> r = region( 'plane', -0.1 )
Half-plane such that: Re(z)<-0.1
>> quiz = quiz + dstability( usys, r )
```

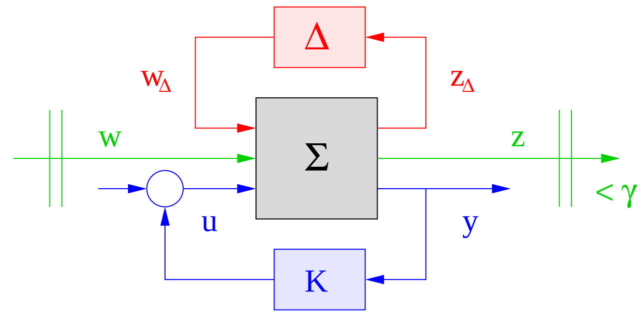
- Add an impulse-to-peak performance minimization objective

```
>> quiz = quiz + i2p( usys )
control problem: STATE-FEEDBACK design
Lyapunov function: UNIQUE (quadratic stability)
Specified performances / systems:
# Hinfty < 4 / Helicopter
# D-stability / Helicopter
# minimize I2P / Helicopter
```

❶ Uncertain LTI systems and performances

- Control objectives: stability, transient response, perturbation rejection...
- Structured parametric uncertainties: extremal values, bounded sets...





❷ LMIs and convex polynomial-time optimization

- Semi-Definite Programming and LMIs
- SDP solvers and parsers

❸ Conservative LMI results

- Methods: Lyapunov, S-procedure, Finsler lemma, Topologic Separation...
- The ROMULOC toolbox

Semi-Definite Programming and LMIs

- Extension of LP to semi-definite matrices

$$\min\ cx \quad : \quad Ax = b \ \ , \quad x_i \geq 0\ (LP) \ \ | \ \ \mathrm{mat}(x) \geq 0\ (SDP)$$

- Convexity, duality, polynomial-time algorithms ($\mathcal{O}(n^{6.5}\log(1/\epsilon))$).

$$\max\ b^T y \quad : \quad A^T y - c^T = z \ \ , \quad \mathrm{mat}(z) \geq 0$$

- 1st developments and 1st results : LMI formalism & Control Theory

$$\min \sum g_i y_i \quad : \quad F_0 + \sum F_i y_i \geq 0$$

▲ The H_∞ norm computation example for $G(s) \sim (A, B, C, D)$:

$$\|G(s)\|_\infty^2 \ = \ \min\ \gamma \ : \ P > 0 \ , \ \begin{bmatrix} A^T P + PA + C_z^T C_z & B_w P + C_z^T D_{zw} \\ P B_w^T + D_{zw}^T C_z & -\gamma 1 + D_{zw}^T D_{zw} \end{bmatrix} \leq 0$$

SDP solvers and parsers

- LMI Control Toolbox $\Rrightarrow$ Control Toolbox

1st solver, dedicated to LMIs issued from Control Theory, Matlab, owner.

- SDP solvers: SP, SeDuMi, SDPT3, CSDP, DSDP, SDPA...

Active field, mathematical programing, C/C++, free.

- Parsers: tklmitool, sdpsol, SeDuMiInterface, YALMIP

Convert LMIs to SDP solver format, Matlab (Scilab), free.

■ SDP-LMI issues and prospectives

● Any SDP representable problem is ”solved” (numerical problems due to size and structure)

▲ Find ”SDP-ables” problems

(linear systems, performances, robustness, LPV, saturations, delays, singular systems...)

▲ Equivalent SDP formulations ⇛ distinguish which are numerically efficient

▲ New SDP solvers: faster, precise, robust (need for benchmark examples)

SDP-LMI issues and prospectives

- Any SDP representable problem is ”solved” (numerical problems due to size and structure)
  - Find ”SDP-ables” problems

(linear systems, performances, robustness, LPV, saturations, delays, singular systems...)

  - Equivalent SDP formulations $\Rrightarrow$ distinguish which are numerically efficient
  - New SDP solvers: faster, precise, robust (need for benchmark examples)

- Any ”SDP-able” problem has a dual interpretation
  - New theoretical results (worst case)
  - New proofs (Lyapunov functions = Lagrange multipliers; related to SOS)
  - SDP formulas numerically stable (KYP-lemma)

SDP-LMI issues and prospectives

- Any SDP representable problem is "solved" (numerical problems due to size and structure)
  - Find "SDP-ables" problems

(linear systems, performances, robustness, LPV, saturations, delays, singular systems...)

  - Equivalent SDP formulations $\Rrightarrow$ distinguish which are numerically efficient
  - New SDP solvers: faster, precise, robust (need for benchmark examples)

- Any "SDP-able" problem has a dual interpretation
  - New theoretical results (worst case)
  - New proofs (Lyapunov functions = Lagrange multipliers; related to SOS)
  - SDP formulas numerically stable (KYP-lemma)

- Non "SDP-able" : Robustesse & Multi-objective & Relaxation of NP-hard problems
  - Optimistic / Pessimistic (conservative) results
  - Reduce the gap (upper/lower bounds) while handling numerical complexity growth.

SDP-LMI issues and prospectives

- Any SDP representable problem is ”solved” (numerical problems due to size and structure)
  - Find ”SDP-ables” problems

(linear systems, performances, robustness, LPV, saturations, delays, singular systems...)

  - Equivalent SDP formulations $\Rrightarrow$ distinguish which are numerically efficient
  - New SDP solvers: faster, precise, robust (need for benchmark examples)

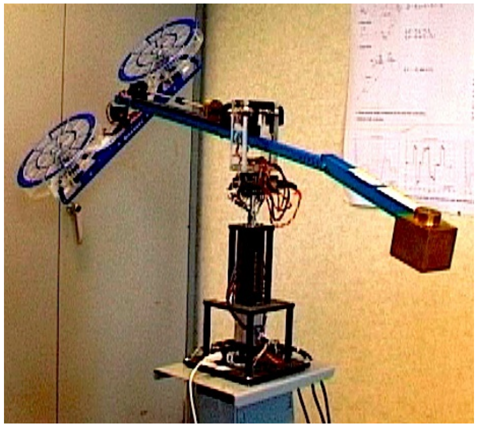
- Any ”SDP-able” problem has a dual interpretation
  - New theoretical results (worst case)
  - New proofs (Lyapunov functions = Lagrange multipliers; related to SOS)
  - SDP formulas numerically stable (KYP-lemma)

- Non ”SDP-able” : Robustesse & Multi-objective & Relaxation of NP-hard problems
  - Optimistic / Pessimistic (conservative) results
  - Reduce the gap (upper/lower bounds) while handling numerical complexity growth.

- Develop software for ”industrial” application / adapted to the application field

$\Rrightarrow$ ROMULOC toolbox

■ "Helicopter" example



● The `quiz` object

```
>> quiz
control problem: STATE-FEEDBACK design
Lyapunov function: UNIQUE (quadratic stability)
Specified performances / systems:
# Hinfty < 4 / Helicopter
# D-stability / Helicopter
# minimize I2P / Helicopter
```

● Contains decision variables

```
>> quiz.vars
    [3x3 sdpvar]    'Lyapunov matrix'
    [1x3 sdpvar]    'S=-K*P'
    [1x1 sdpvar]    'S-procedure scaling'
    [1x1 sdpvar]    'g > (I2P cost)^2'
```

- Constrained by LMIs

```
>> quiz.lmi
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
|     ID|      Constraint|                          Type|            Tag|
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
|     #1|   Numeric value|   Matrix inequality 3x3|         Lyap >0|
|     #2|   Numeric value|   Matrix inequality 4x4|     Var Lyap <0|
|     #3|   Numeric value|   Matrix inequality 4x4|     Var Lyap <0|
|     #4|   Numeric value|   Matrix inequality 4x4|     Var Lyap <0|
|     #5|   Numeric value|   Matrix inequality 4x4|     Var Lyap <0|
|     #6|   Numeric value|   Matrix inequality 3x3|     Var Lyap <0|
|     #7|   Numeric value|   Matrix inequality 3x3|     Var Lyap <0|
|     #8|   Numeric value|   Matrix inequality 3x3|     Var Lyap <0|
|     #9|   Numeric value|   Matrix inequality 3x3|     Var Lyap <0|
|    #10|   Numeric value|   Matrix inequality 3x3|    Constraint 1|
|    #11|   Numeric value|   Matrix inequality 4x4|    Constraint 2|
|    #12|   Numeric value|   Matrix inequality 3x3|    Constraint 3|
|    #13|   Numeric value|        Element-wise 1x1|    Constraint 4|
|    #14|   Numeric value|   Matrix inequality 3x3|    Constraint 1|
|    #15|   Numeric value|   Matrix inequality 4x4|    Constraint 2|
|    #16|   Numeric value|   Matrix inequality 3x3|    Constraint 3|
|    #17|   Numeric value|        Element-wise 1x1|    Constraint 4|
```

```
|    #18|    Numeric value|    Matrix inequality 3x3|    Constraint 1|
|    #19|    Numeric value|    Matrix inequality 4x4|    Constraint 2|
|    #20|    Numeric value|    Matrix inequality 3x3|    Constraint 3|
|    #21|    Numeric value|          Element-wise 1x1|    Constraint 4|
|    #22|    Numeric value|    Matrix inequality 3x3|    Constraint 1|
|    #23|    Numeric value|    Matrix inequality 4x4|    Constraint 2|
|    #24|    Numeric value|    Matrix inequality 3x3|    Constraint 3|
|    #25|    Numeric value|          Element-wise 1x1|    Constraint 4|
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

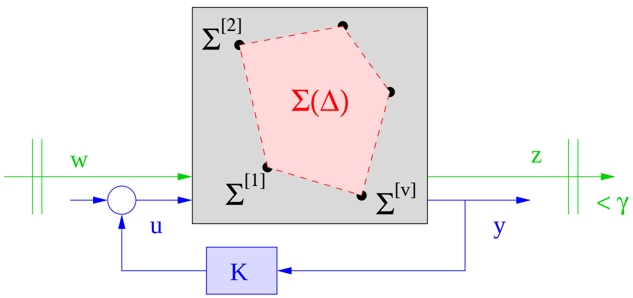
- And can be solved (SeDuMi solver by default)

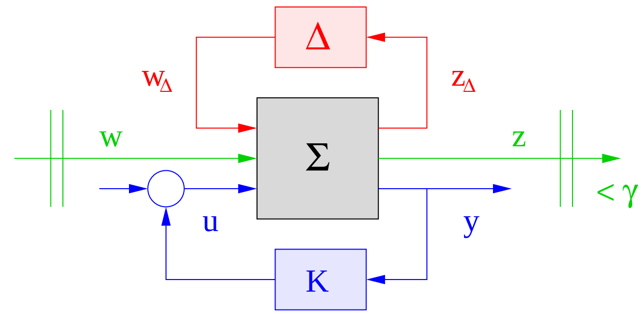
```
>> K = solvesdp( quiz )
SeDuMi 1.1R3 by AdvOL, 2006 and Jos F. Sturm, 1998-2003.
Alg = 2: xz-corrector, theta = 0.250, beta = 0.500
eqs m = 11, order n = 76, dim = 250, blocks = 22
nnz(A) = 282 + 0, nnz(ADA) = 117, nnz(L) = 64
 it :     b*y       gap    delta  rate   t/tP*  t/tD*   feas cg cg  prec
  0 :            6.96E+01 0.000
  1 :  -1.79E+02 1.85E+01 0.000 0.2657 0.9000 0.9000  -0.09  1  1  5.0E+02
  2 :  -1.05E+02 5.96E+00 0.000 0.3223 0.9000 0.9000   1.55  1  1  1.1E+02
  3 :  -2.56E+01 1.38E+00 0.000 0.2312 0.9000 0.9000   1.73  1  1  1.9E+01
  4 :  -5.54E+00 2.62E-01 0.000 0.1902 0.9000 0.9000   1.21  1  1  3.2E+00
  5 :  -1.84E+00 8.00E-02 0.000 0.3050 0.9000 0.9000   1.29  1  1  8.3E-01
  6 :  -7.08E-01 2.90E-02 0.000 0.3621 0.9000 0.9000   1.35  1  1  2.6E-01
  7 :  -2.95E-01 1.05E-02 0.000 0.3637 0.9000 0.9000   1.27  1  1  8.3E-02
  8 :  -2.30E-01 3.57E-03 0.000 0.3393 0.9000 0.9000   1.12  1  1  2.7E-02
  9 :  -1.97E-01 6.73E-04 0.000 0.1882 0.9000 0.9000   1.00  1  1  5.1E-03
 10 :  -1.91E-01 2.02E-05 0.000 0.0300 0.9900 0.9900   0.98  1  1  1.6E-04
 11 :  -1.91E-01 1.13E-06 0.000 0.0558 0.9900 0.9900   1.00  1  1  8.7E-06
 12 :  -1.91E-01 3.08E-07 0.000 0.2737 0.9000 0.9000   1.00  1  1  2.4E-06
 13 :  -1.91E-01 1.33E-08 0.000 0.0433 0.9900 0.9900   1.00  1  1  1.0E-07
 14 :  -1.91E-01 3.01E-09 0.000 0.2261 0.9000 0.9000   1.00  2  2  2.3E-08
```

```
 15 :  -1.91E-01 7.53E-10 0.000 0.2498 0.9000 0.9000   1.00  2  2  5.8E-09
 16 :  -1.91E-01 4.50E-11 0.087 0.0598 0.9900 0.9900   1.00  2  2  3.5E-10

iter seconds digits       c*x                 b*y
 16      0.4   Inf -1.9059654950e-01 -1.9059654919e-01
|Ax-b| =   3.6e-10, [Ay-c]_+ =   2.6E-11, |x|=  5.0e-01, |y|=  3.7e+02

Detailed timing (sec)
   Pre           IPM           Post
1.800E-01    4.000E-01    7.000E-02
Max-norms: ||b||=1, ||c|| = 196,
Cholesky |add|=0, |skip| = 0, ||L.L|| = 42153.4.

Feasibility is not strictly determined
Worst constraint residual is -2.59066e-11 < 0

0.436574 (=sqrt(double(CTRPB.vars{4}))) may be a guaranteed I2P norm

K =
    0.0442    0.0091    0.0305
```

❶ Uncertain LTI systems and performances

- Control objectives: stability, transient response, perturbation rejection...
- Structured parametric uncertainties: extremal values, bounded sets...





❷ LMIs and convex polynomial-time optimization

- Semi-Definite Programming and LMIs
- SDP solvers and parsers

$\Rrightarrow$ YALMIP and all solvers

❸ Conservative LMI results

- Methods: Lyapunov, S-procedure, Finsler lemma, Topologic Separation...
- The ROMULOC toolbox

■ Nominal performance analysis: $V(x) = x^T P x$ Lyapunov function ($P > 0$)

- Stability $A^T P + PA < 0 \quad | \quad A^T PA - P < 0$

- D_R-Stability $$\begin{bmatrix} 1 & A^* \end{bmatrix} \begin{bmatrix} r_{11}P & r_{12}P \\ r_{12}^* P & r_{22}P \end{bmatrix} \begin{bmatrix} 1 \\ A \end{bmatrix} < 0$$

- H_∞ norm $$\begin{bmatrix} A^T P + PA + C_z^T C_z & PB_w + C_z^T D_{zw} \\ B_w^T P + D_{zw}^T C_z & -\gamma^2 1 + D_{zw}^T D_{zw} \end{bmatrix} < 0$$

- H_2 norm $$A^T P + PA + C_z^T C_z < 0$$ $$\mathrm{trace}(B_w^T P B_w) < \gamma^2$$

- Impulse-to-peak $$A^T P + PA < 0 \quad B_w^T P B_w < \gamma^2 1$$ $$C_z^T C_z < P \quad D_{zw}^T D_{zw} < \gamma^2 1$$

■ Robust performance analysis: $V(x, \Delta)$ parameter-dependent Lyapunov function.

▲ Nominal analysis (LMI) $\longrightarrow$ Robust analysis (NP-hard)

$$\exists\, P \;:\; \mathcal{L}_{\Sigma}(P) < 0 \quad\longrightarrow\quad \forall\, \Delta \in \mathbf{\Delta}\,,\; \exists\, P(\Delta) \;:\; \mathcal{L}_{\Sigma(\Delta)}(P(\Delta)) < 0$$

▲ Test over sample values $\{\Delta_{1\dots N}\} \in \mathbf{\Delta}$ gives optimistic results

(some results exist if $\{\Delta_{1\dots N}\}$ is uniform distribution of $\mathbf{\Delta}$ and large N)

■ Robust performance analysis: $V(x,\Delta)$ parameter-dependent Lyapunov function.

▲ Nominal analysis (LMI) $\longrightarrow$ Robust analysis (NP-hard)

$$\exists\, P \;:\; \mathcal{L}_{\Sigma}(P) < 0 \quad \rightarrow \quad \forall\, \Delta \in \mathbf{\Delta}\,,\ \exists\, P(\Delta) \;:\; \mathcal{L}_{\Sigma(\Delta)}(P(\Delta)) < 0$$

▲ Test over sample values $\{\Delta_{1\ldots N}\} \in \mathbf{\Delta}$ gives optimistic results

(some results exist if $\{\Delta_{1\ldots N}\}$ is uniform distribution of $\mathbf{\Delta}$ and large N)

● Choice of $P(\Delta)$ for having a finite number of decision variables :

➞ "Quadratic Stability": $P(\Delta) = P$

➞ Polytopic PDLF: $P(\Delta) = \sum \zeta_i P^{[i]}$

➞ $P(\Delta)$ polynomial w.r.t. ζ_i (not coded in RoMulOC)

➞ Quadratic-LFT PDLF: $P(\Delta) = \begin{bmatrix} 1 & \Delta_C^T \end{bmatrix} P \begin{bmatrix} 1 \\ \Delta_C \end{bmatrix} \;:\; \Delta_C = (1 - \Delta D_{\Delta\Delta})^{-1}\Delta C_{\Delta}$

➞ $P(\Delta)$ polynomial w.r.t. Δ_C (not coded in RoMulOC)

■ Robust performance analysis: $V(x, \Delta)$ parameter-dependent Lyapunov function.

▲ Nominal analysis (LMI) $\longrightarrow$ Robust analysis (NP-hard)

$$\exists\, P \;:\; \mathcal{L}_\Sigma(P) < 0 \quad \rightarrow \quad \forall\, \Delta \in \boldsymbol{\Delta}\,,\; \exists\, P(\Delta) \;:\; \mathcal{L}_{\Sigma(\Delta)}(P(\Delta)) < 0$$

▲ Test over sample values $\{\Delta_{1\ldots N}\} \in \boldsymbol{\Delta}$ gives optimistic results

(some results exist if $\{\Delta_{1\ldots N}\}$ is uniform distribution of $\boldsymbol{\Delta}$ and large N)

● Choice of $P(\Delta)$ for having a finite number of decision variables :

- ➙ “Quadratic Stability”: $P(\Delta) = P$
- ➙ Polytopic PDLF: $P(\Delta) = \sum \zeta_i P^{[i]}$
- ➙ Quadratic-LFT PDLF: $P(\Delta) = \begin{bmatrix} 1 & \Delta_C^T \end{bmatrix} P \begin{bmatrix} 1 \\ \Delta_C \end{bmatrix} \;:\; \Delta_C = (1 - \Delta D_{\Delta\Delta})^{-1} \Delta C_\Delta$

▲ LMIs over infinite number of variables

$$\forall\, \Delta \in \boldsymbol{\Delta}\,,\; \exists\, P(\Delta) \;:\; \mathcal{L}_{\Sigma(\Delta)}(P(\Delta)) < 0$$

$$\Leftarrow \exists\, P^{[i]} \;:\; \forall\, \Delta \in \boldsymbol{\Delta}\,,\; \mathcal{L}_{\Sigma(\Delta)}(P(\Delta)) < 0$$

■ Conservative LMIs for polytopic models (Example of stability analysis)

$\dot{x} = A(\xi)x$ with $A(\xi) = \sum_{v=1}^{\bar{v}} \xi_v A^{[v]} : \ \xi \in \Xi = \{\xi_v \geq 0, \ \sum_{v=1}^{\bar{v}} \xi_v = 1\}$

Conservative LMIs for polytopic models (Example of stability analysis)

$\dot{x} = A(\xi)x$ with $A(\xi) = \sum_{v=1}^{\bar{v}} \xi_v A^{[v]}$: $\xi \in \Xi = \{\xi_v \geq 0, \sum_{v=1}^{\bar{v}} \xi_v = 1\}$

- "Quadratic Stability": $P(\Delta) = P$

$$\dot{V}(x) < 0 \;\Leftrightarrow\; A^T(\Delta)P + PA(\Delta) < 0 \;\Leftrightarrow\; {A^{[i]}}^T P + PA^{[i]} < 0$$

Conservative LMIs for polytopic models (Example of stability analysis)

$\dot{x} = A(\xi)x$ with $A(\xi) = \sum_{v=1}^{\bar{v}} \xi_v A^{[v]}$: $\xi \in \Xi = \{\xi_v \geq 0, \ \sum_{v=1}^{\bar{v}} \xi_v = 1\}$

- "Quadratic Stability": $P(\Delta) = P$

$$\dot{V}(x) < 0 \ \Leftrightarrow \ A^T(\Delta)P + PA(\Delta) < 0 \ \Leftrightarrow \ {A^{[i]}}^T P + PA^{[i]} < 0$$

- Polytopic PDLF: $P(\Delta) = \sum \zeta_i P^{[i]}$

$$\begin{pmatrix} x \\ \dot{x} \end{pmatrix}^T \begin{bmatrix} 0 & P(\Delta) \\ P(\Delta) & 0 \end{bmatrix} \begin{pmatrix} x \\ \dot{x} \end{pmatrix} < 0 \ : \ \begin{bmatrix} A(\Delta) & -1 \end{bmatrix} \begin{pmatrix} x \\ \dot{x} \end{pmatrix} = 0$$

$\Leftrightarrow$ Finsler Lemma

$$\begin{bmatrix} 0 & P(\Delta) \\ P(\Delta) & 0 \end{bmatrix} + G(\Delta) \begin{bmatrix} A(\Delta) & -1 \end{bmatrix} + \begin{bmatrix} A^T(\Delta) \\ -1 \end{bmatrix} G^T(\Delta) < 0$$

$\Leftarrow$ $G(\Delta) = G$ & convexity

$$\begin{bmatrix} 0 & P^{[i]} \\ P^{[i]} & 0 \end{bmatrix} + G \begin{bmatrix} A^{[i]} & -1 \end{bmatrix} + \begin{bmatrix} {A^{[i]}}^T \\ -1 \end{bmatrix} G^T < 0$$

■ Conservative LMIs for LFT models (Example of stability analysis)

$$\dot{x} = Ax + B_\Delta w_\Delta \text{ with } w_\Delta = \Delta z_\Delta = \Delta C_\Delta x + \Delta D_{\Delta\Delta} w_\Delta$$

■ Conservative LMIs for LFT models (Example of stability analysis)

$$\dot{x} = Ax + B_\Delta w_\Delta \text{ with } w_\Delta = \Delta z_\Delta = \Delta C_\Delta x + \Delta D_{\Delta\Delta} w_\Delta$$

● "Quadratic Stability": $P(\Delta) = P$

$$\dot{V}(x) = \begin{pmatrix} x \\ w_\Delta \end{pmatrix}^* \begin{bmatrix} 1 & 0 \\ A & B_\Delta \end{bmatrix}^* \begin{bmatrix} 0 & P \\ P & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ A & B_\Delta \end{bmatrix} \begin{pmatrix} x \\ w_\Delta \end{pmatrix} < 0$$

$$: \begin{bmatrix} \Delta & -1 \end{bmatrix} \begin{bmatrix} C_\Delta & D_{\Delta\Delta} \\ 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ w_\Delta \end{pmatrix} = 0$$

$\Leftrightarrow$ Finsler Lemma

$$M_A^* \begin{bmatrix} 0 & P \\ P & 0 \end{bmatrix} M_A < \tau M_C^* \begin{bmatrix} \Delta^* \\ -1 \end{bmatrix} \begin{bmatrix} \Delta & -1 \end{bmatrix} M_C$$

■ Conservative LMIs for LFT models (Example of stability analysis)

$$\dot{x} = Ax + B_\Delta w_\Delta \text{ with } w_\Delta = \Delta z_\Delta = \Delta C_\Delta x + \Delta D_{\Delta\Delta} w_\Delta$$

● "Quadratic Stability": $P(\Delta) = P$

$$\dot{V}(x) = \begin{pmatrix} x \\ w_\Delta \end{pmatrix}^* \begin{bmatrix} 1 & 0 \\ A & B_\Delta \end{bmatrix}^* \begin{bmatrix} 0 & P \\ P & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ A & B_\Delta \end{bmatrix} \begin{pmatrix} x \\ w_\Delta \end{pmatrix} < 0$$

$$: \begin{bmatrix} \Delta & -1 \end{bmatrix} \begin{bmatrix} C_\Delta & D_{\Delta\Delta} \\ 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ w_\Delta \end{pmatrix} = 0$$

$\Leftrightarrow$ Finsler Lemma

$$M_A^* \begin{bmatrix} 0 & P \\ P & 0 \end{bmatrix} M_A < M_C^* \Theta M_C \leq \tau M_C^* \begin{bmatrix} \Delta^* \\ -1 \end{bmatrix} \begin{bmatrix} \Delta & -1 \end{bmatrix} M_C$$

$$\text{with } \begin{bmatrix} 1 & \Delta^* \end{bmatrix} \Theta \begin{bmatrix} 1 \\ \Delta \end{bmatrix} \leq 0$$

❸ Conservative LMI results

Conservative LMIs for LFT models (Example of stability analysis)

$$\dot{x} = Ax + B_\Delta w_\Delta \text{ with } w_\Delta = \Delta z_\Delta = \Delta C_\Delta x + \Delta D_{\Delta\Delta} w_\Delta$$

- "Quadratic Stability": $P(\Delta) = P$

$$\dot{V}(x) = \begin{pmatrix} x \\ w_\Delta \end{pmatrix}^* \begin{bmatrix} 1 & 0 \\ A & B_\Delta \end{bmatrix}^* \begin{bmatrix} 0 & P \\ P & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ A & B_\Delta \end{bmatrix} \begin{pmatrix} x \\ w_\Delta \end{pmatrix} < 0$$

$$: \begin{bmatrix} \Delta & -1 \end{bmatrix} \begin{bmatrix} C_\Delta & D_{\Delta\Delta} \\ 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ w_\Delta \end{pmatrix} = 0$$

$\Leftrightarrow$ Finsler Lemma

$$M_A^* \begin{bmatrix} 0 & P \\ P & 0 \end{bmatrix} M_A < M_C^* \Theta M_C \leq \tau M_C^* \begin{bmatrix} \Delta^* \\ -1 \end{bmatrix} \begin{bmatrix} \Delta & -1 \end{bmatrix} M_C$$

$$\text{with } \begin{bmatrix} 1 & \Delta^* \end{bmatrix} \Theta \begin{bmatrix} 1 \\ \Delta \end{bmatrix} \leq 0$$

- Quadratic-LFT PDLF - same methodology (yet needs many matrix manipulations).

Conservative LMIs for LFT models

- LMI constraints on Quadratic Separators Θ

- $\{X,Y,Z\}-$dissipative matrices $\mathbf{\Delta} = \{\Delta \ : \ X + Y\Delta + \Delta^* Y^* + \Delta^* Z\Delta \leq 0\}$

$$\begin{bmatrix} 1 & \Delta^* \end{bmatrix} \Theta \begin{bmatrix} 1 \\ \Delta \end{bmatrix} \leq 0 \ : \ \forall \Delta \in \mathbf{\Delta} \quad \Leftrightarrow \quad \Theta \leq \tau \begin{bmatrix} X & Y \\ Y^* & Z \end{bmatrix}, \ \tau \geq 0$$

Conservative LMIs for LFT models

- LMI constraints on Quadratic Separators Θ

- $\{X,Y,Z\}-$dissipative matrices $\mathbf{\Delta} = \{\Delta \ : \ X + Y\Delta + \Delta^* Y^* + \Delta^* Z \Delta \leq 0\}$

$$\begin{bmatrix} 1 & \Delta^* \end{bmatrix} \Theta \begin{bmatrix} 1 \\ \Delta \end{bmatrix} \leq 0 : \ \forall \Delta \in \mathbf{\Delta} \quad \Leftrightarrow \quad \Theta \leq \tau \begin{bmatrix} X & Y \\ Y^* & Z \end{bmatrix}, \ \tau \geq 0$$

- $\{X,Y,Z\}-$dissipative real repeated scalars $\mathbf{\Delta} = \left\{\Delta = \delta 1 \ : \ x + 2y\delta + z\delta^2 \leq 0 \right\}$

$$\Theta \leq \begin{bmatrix} xD & yD + G \\ yD - G & zD \end{bmatrix}, \quad \begin{matrix} Q \geq 0 \\ G = -G^* \end{matrix}$$

Conservative LMIs for LFT models

- LMI constraints on Quadratic Separators Θ

▲ $\{X,Y,Z\}$−dissipative matrices $\mathbf{\Delta} = \{\Delta \;:\; X + Y\Delta + \Delta^* Y^* + \Delta^* Z \Delta \leq 0\,\}$

$$\begin{bmatrix} 1 & \Delta^* \end{bmatrix} \Theta \begin{bmatrix} 1 \\ \Delta \end{bmatrix} \leq 0 : \; \forall \Delta \in \mathbf{\Delta} \quad \Leftrightarrow \quad \Theta \leq \tau \begin{bmatrix} X & Y \\ Y^* & Z \end{bmatrix}, \; \tau \geq 0$$

▲ $\{X,Y,Z\}$−dissipative real repeated scalars $\mathbf{\Delta} = \{\Delta = \delta 1 \;:\; x + 2y\delta + z\delta^2 \leq 0\,\}$

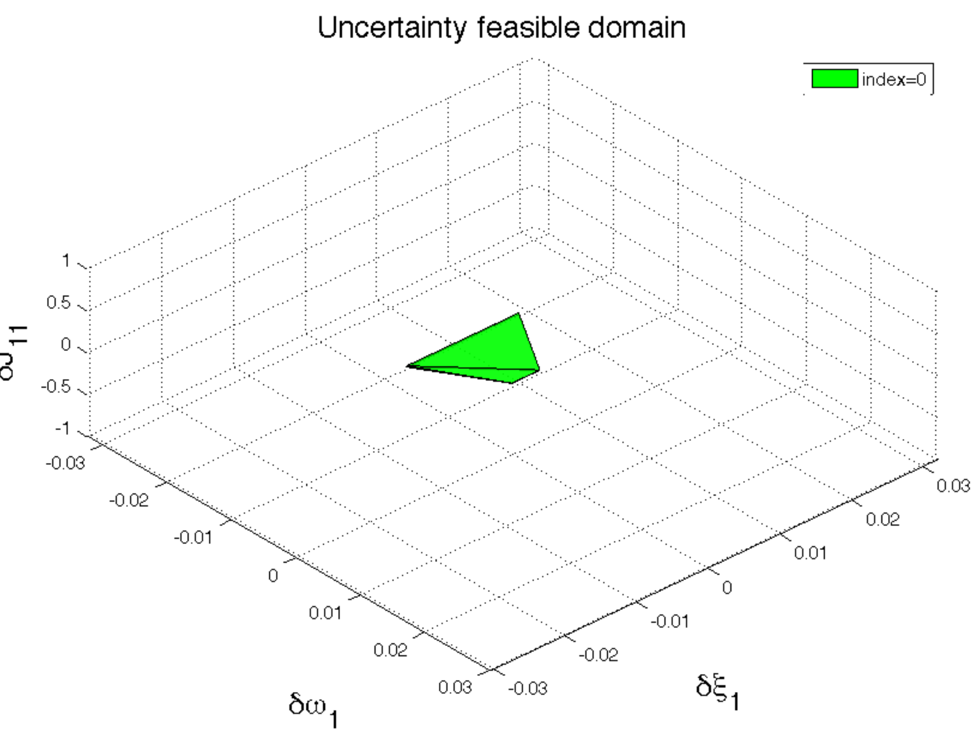
$$\Theta \leq \begin{bmatrix} xD & yD + G \\ yD - G & zD \end{bmatrix}, \quad \begin{matrix} Q \geq 0 \\ G = -G^* \end{matrix}$$

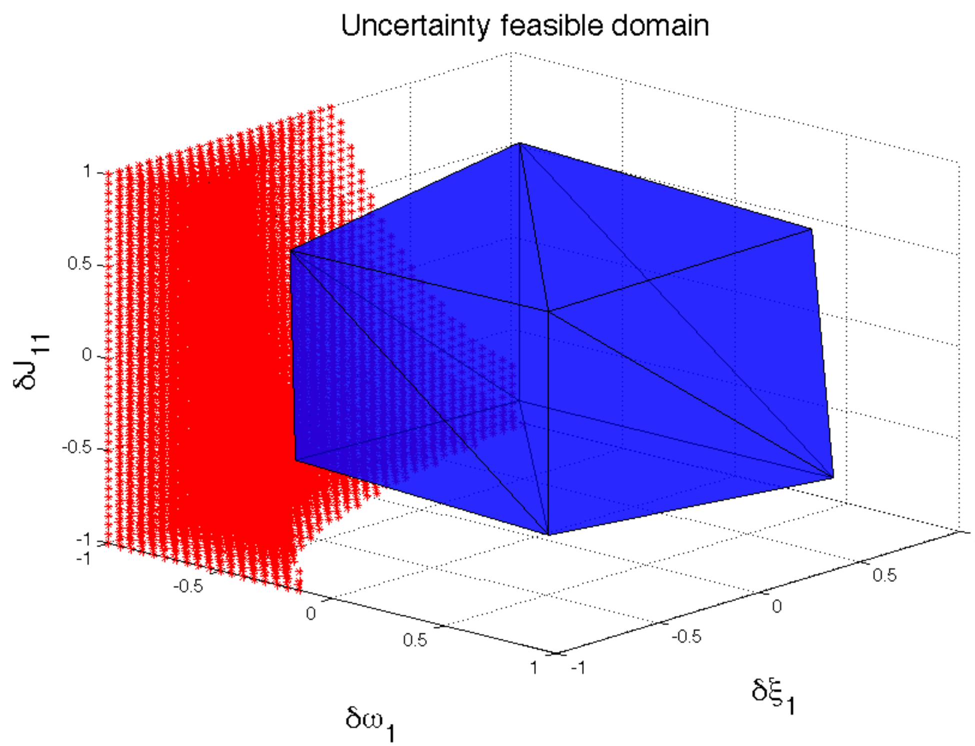
▲ Polytopic uncertainties $\mathbf{\Delta} = \{\, \Delta = \sum \xi_i \Delta^{[i]} \;:\; \xi_i \geq 0 \;,\; \sum \xi_i = 1 \,\}$

$$\begin{bmatrix} 1 & \Delta^{[i]*} \end{bmatrix} \Theta \begin{bmatrix} 1 \\ \Delta^{[i]} \end{bmatrix} \leq 0 \;, \quad \begin{bmatrix} 0 & 1 \end{bmatrix} \Theta \begin{bmatrix} 0 \\ 1 \end{bmatrix} \geq 0$$

■ RoMulOC today: `www.laas.fr/OLOCEP/romuloc/downloads.html`

- Large variety of uncertain models associated with multiple performances
- Several associated LMI-based theoretical results coded
- Access to efficient LMI solvers (thanks to YALMIP)
- Testing being done on applications

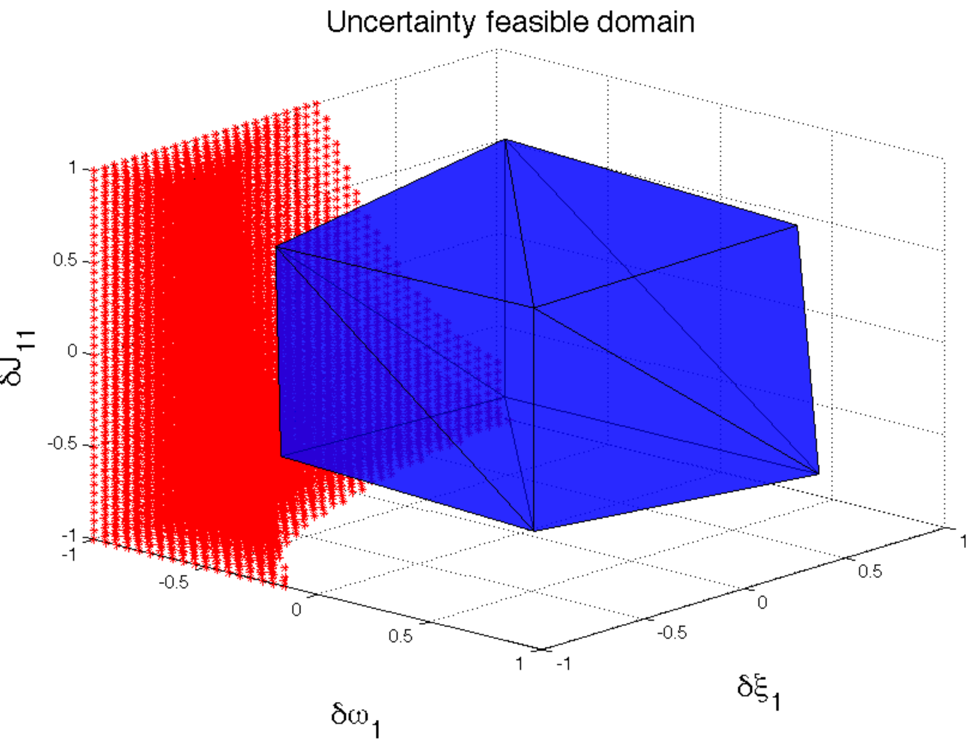


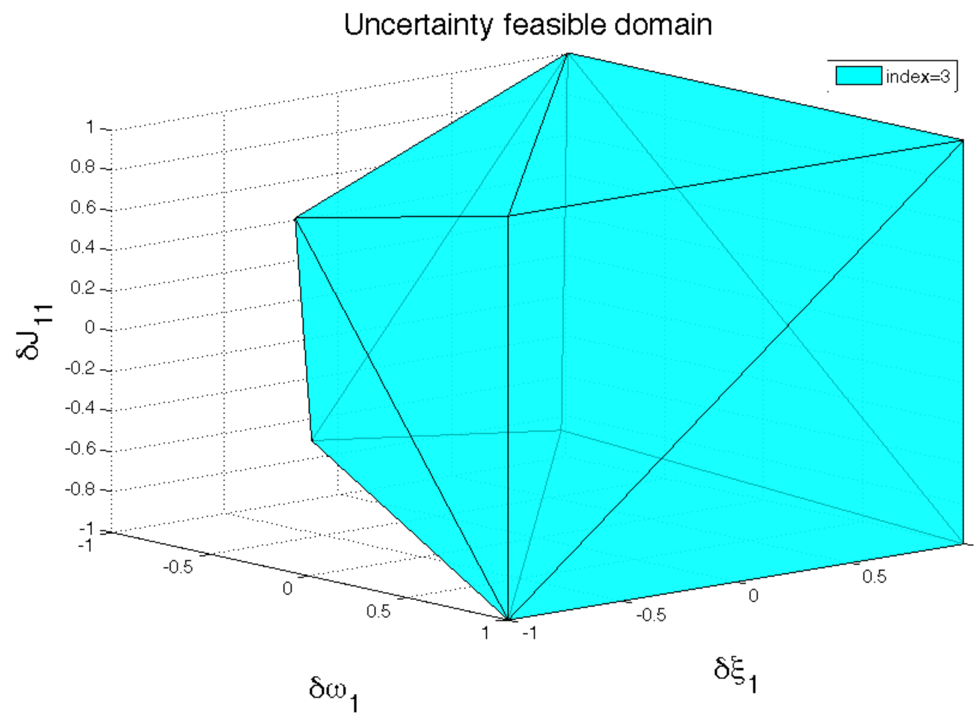


- Design: limited to state-feedback

■ <u>RoMulOC in the future</u>

- Other design problems: Dynamic output-feedback (LMI) Static output-feedback (not LMI)
- Time-varying uncertainties (with bounded derivatives)
- Time-delay systems (constant or time-varying)
- Non-linearities (Saturations, dead-zone ...)
- Other Lyapunov functions





■ Descriptor systems: $E(\Delta)\dot{x} = A(\Delta)x \Rightarrow$ Romuald