Optimization based control for Robots

some solutions for the implementation issue

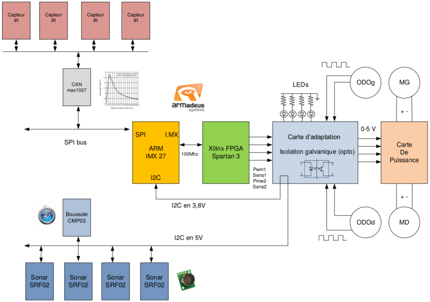
Benoît CLEMENT

June, 30, 2016

Control point of view

Today is dedicated to answer few questions with a control point of view in Robotics.

- Which Robotics?
- How to control a Robot?
- Are the GNC algorithms are a big issue?



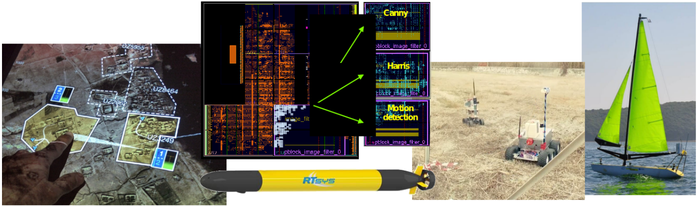
Outline

Ocean Senging and Mapping at ENSTA Bretagne

- OSM: Teaching and Research Department
- **Large scope of teaching activities**: hydrography, oceanography, embedded electronics, signal processing, information technology, computer science, robotics, etc.
- Research topics:
  - Hydrography/Oceanography
  - Underwater robotics
  - Sonar systems
  - Data Processing
- Application field: Maritime environment, civilian and defense.

Focus on Robotics

- Robotics issues:
  - Guidance, Navigation and Control
  - Group of Robots: interaction management
  - Localisation
- Academic tools:
  - Interval Analysis
  - Data processing
  - Global Optimization
  - Robust Control

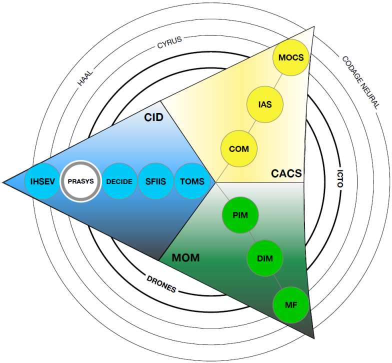


Teaching

- Linear Control and Sensors
- Mobile Robotics
- Localisation and Kalman filter
- Prototyping Robots
- Middleware and Compilation
- Simulation and nonlinear control
- Digital conception
- Robust Control
- Vision
- Robotics Architecture

Lab-STICC

- UMR CNRS 6285
- CID : Connaissance Information Decision
- PRASYS : **P**erception, **R**obotics, **A**utonomous **SYS**tems



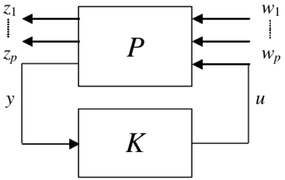
Some Robots

Demonstrations with movies...

What is robust control

Question : find a controller that insures performances to the closed loop

- **P et K : systèmes LTI ou LPV - MIMO**
- **u = commandes, y = mesures**
- **Transferts T_i utilisés pour spécifier différents objectifs de performance ou robustesse :**



$$\left\|T_{z_1/w_1}\right\| < \gamma_1$$
$$\vdots$$
$$\left\|T_{z_p/w_p}\right\| < \gamma_p$$

$\| \; \|_2$ ou $\| \; \|_\infty$

$$P = \left|\begin{array}{c|ccc} A & B_1 & B_2 & B_u \\ \hline C_1 & D_{11} & D_{12} & D_{1u} \\ C_2 & D_{21} & D_{22} & D_{2u} \\ C_y & D_{y1} & D_{y2} & 0 \end{array}\right|$$

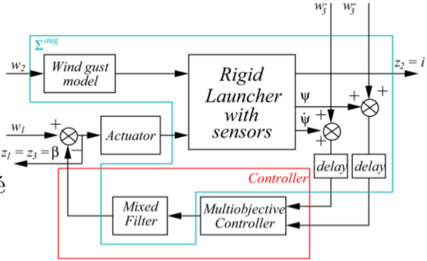
$$K = \left(\begin{array}{c|c} A_K & B_K \\ \hline C_K & D_K \end{array}\right)$$

Norm interpretations

- The H_2-norm:
  - for SISO systems, the induced norm from l_2 to l_∞
  - the square root of the average power (is *RMS value* or *power-norm*) of the response to a white input signal of unit spectral density or the spectrum/power gain.
  - the square root of the energy contained in the impulse response
- The H_∞-norm:
  - the induced norm from l_2 to l_2
  - the power/power gain (RMS)
  - the spectrum/spectrum gain
  - an upper bound on the l_∞/power gain, assuming that the input is a persistent sinusoidal signal
  - the peak amplitude of the Bode singular value plot

Control / Objectives / Optimisation Constraints

- **i2p (Impulse to peak)**
  Influence du vent sur l'incidence
- H_∞
  Sur la fonction de sensibilité pour les marges de stabilité
- H_2
  Influence des bruit et Réduction de consommation
- **Filtrage**
  Contrôle des modes avec réglage séparé



$$\min_{K \in \mathcal{K}} \quad \alpha_i \gamma_{w-i} + \alpha_c \gamma_{cons}$$

s.t.

$$||\Sigma^{aug.}_{mod} \star K||^2_\infty \leq \gamma_{mod}$$

$$||\Sigma^{aug.}_{W-i} \star K||^2_{i2p} \leq \gamma_{w-i}$$

$$||\Sigma^{aug.}_{cons} \star K||^2_2 \leq \gamma_{cons}$$

D. Arzelier, B. Clement et D. Peaucelle. Multi-objective H2/H∞/Impulse-to-Peak Control of a Space Launch Vehicle. *European Journal of Control*, vol. 12, no. 1, 2006

Control / Objectives / Optimisation Constraints - LMI Optimization

$$\min_{\substack{\mathbf{X},\mathbf{Y},\gamma_{cons},\gamma_{w-i},\mathbf{T},\mathbf{S}\\ \mathbf{J_i},\mathbf{H_i},\mathbf{Q_i},\, i=\infty,i2p,2}} \alpha_c\gamma_{cons} + \alpha_i\gamma_{w-i}$$

under

$$\begin{bmatrix} -\mathbf{Q}_\infty & -\mathbf{J}_\infty & A\mathbf{X}+B\hat{\mathbf{C}} & A+B\hat{\mathbf{D}}C & B_\infty + B\hat{\mathbf{D}}D_{y\infty} & \mathbf{0} \\ \star & -\mathbf{H}_\infty & \hat{\mathbf{A}} & \mathbf{Y}A+\hat{\mathbf{B}}C & \mathbf{Y}B_\infty+\hat{\mathbf{B}}D_{y\infty} & \mathbf{0} \\ \star & \star & \mathbf{Q}_\infty-\mathbf{X}-\mathbf{X}' & -\mathbf{1}-\mathbf{S}'+\mathbf{J}_\infty & \mathbf{0} & \mathbf{X}'C'_\infty+\hat{\mathbf{C}}'D'_{\infty u} \\ \star & \star & \star & \mathbf{H}_\infty-\mathbf{Y}-\mathbf{Y}' & \mathbf{0} & C'_\infty+C'\hat{\mathbf{D}}'D'_{\infty u} \\ \star & \star & \star & \star & -\mathbf{1} & D'_\infty+D'_{y\infty}\hat{\mathbf{D}}'D'_{\infty u} \\ \star & \star & \star & \star & \star & -\gamma_{mod}\mathbf{1} \end{bmatrix} < \mathbf{0}$$

$$\begin{bmatrix} -\mathbf{Q_{i2p}} & -\mathbf{J_{i2p}} & A\mathbf{X}+B\hat{\mathbf{C}} & A+B\hat{\mathbf{D}}C \\ \star & -\mathbf{H_{i2p}} & \hat{\mathbf{A}} & \mathbf{Y}A+\hat{\mathbf{B}}C \\ \star & \star & \mathbf{Q_{i2p}}-\mathbf{X}-\mathbf{X}' & \mathbf{J_{i2p}}-\mathbf{S}'-\mathbf{1} \\ \star & \star & \star & \mathbf{H_{i2p}}-\mathbf{Y}-\mathbf{Y}' \end{bmatrix} < \mathbf{0}$$

$$\begin{bmatrix} -\mathbf{Q_{i2p}} & -\mathbf{J_{i2p}} & B_{i2p}+B\hat{\mathbf{D}}D_{i2py} \\ \star & -\mathbf{H_{i2p}} & \mathbf{Y}B_{i2p}+\hat{\mathbf{B}}D_{i2py} \\ \star & \star & \mathbf{1} \end{bmatrix} < \mathbf{0}$$

$$\begin{bmatrix} -\boldsymbol{\gamma_{w-i}} & C_{i2p}\mathbf{X}+D_{i2pu}\hat{\mathbf{C}} & C_{i2p}+D_{i2pu}\hat{\mathbf{D}}C \\ \star & \mathbf{Q_{i2p}}-\mathbf{X}-\mathbf{X}' & \mathbf{J_{i2p}}-\mathbf{S}'-\mathbf{1} \\ \star & \star & \mathbf{H_{i2p}}-\mathbf{Y}-\mathbf{Y}' \end{bmatrix} < \mathbf{0}$$

$$\begin{bmatrix} -\boldsymbol{\gamma_{w-i}} & D_{i2p}+D_{i2pu}\hat{\mathbf{D}}D_{i2py} \\ \star & -\mathbf{1} \end{bmatrix} < \mathbf{0}$$

$$\mathrm{Trace}(\mathbf{T_2}) < \boldsymbol{\gamma_{cons}}$$

$$\begin{bmatrix} -\mathbf{T_2} & C_2\mathbf{X}+D_{2u}\hat{\mathbf{C}} & C_2+D_{2u}\hat{\mathbf{D}}C & D_2+D_{2u}\hat{\mathbf{D}}D_{y2} \\ \star & Q_2-\mathbf{X}-\mathbf{X}' & -\mathbf{1}-\mathbf{S}'+\mathbf{J_2} & \mathbf{0} \\ \star & \star & \mathbf{H_2}-\mathbf{Y}-\mathbf{Y}' & \mathbf{0} \\ \star & \star & \star & -\mathbf{1} \end{bmatrix} < \mathbf{0}$$

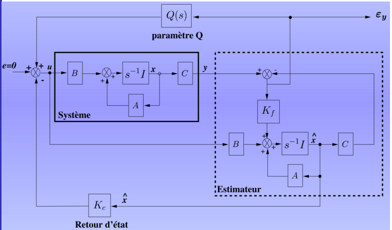
$$\begin{bmatrix} -\mathbf{Q_2} & -\mathbf{J_2} & A\mathbf{X}+B\hat{\mathbf{C}} & A+B\hat{\mathbf{D}}C & B_2+B\hat{\mathbf{D}}D_{y2} \\ \star & -\mathbf{H_2} & \hat{\mathbf{A}} & \mathbf{Y}A+\hat{\mathbf{B}}C & \mathbf{Y}B_2+\hat{\mathbf{B}}D_{y2} \\ \star & \star & \mathbf{Q_2}-\mathbf{X}-\mathbf{X}' & -\mathbf{1}-\mathbf{S}'+\mathbf{J_2} & \mathbf{0} \\ \star & \star & \star & \mathbf{H_2}-\mathbf{Y}-\mathbf{Y}' & \mathbf{0} \\ \star & \star & \star & \star & -\mathbf{1} \end{bmatrix} < \mathbf{0}$$

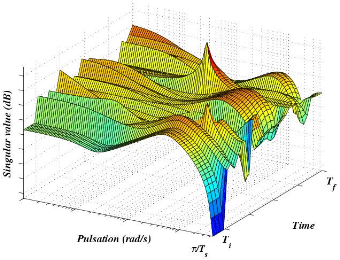
- Conservative solution
- Convex optimization
- Full order controller that needs truncature or/and a posteriori stucturation
- Fragility of the solution

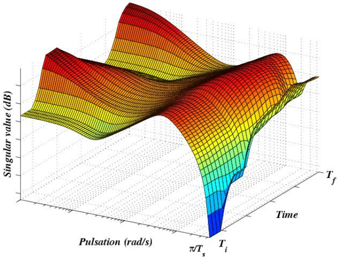
Adding a structural constraint

Structure is good for

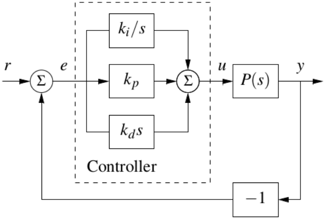
- gains interpolation for LPV systems
- interpretation for physical behavior
- implementation in embedded system (example of PID next slide)







focus on implementation



Full order controller:

- n =sum of the orders of all of systems
- Ordinary differencial equation of order n
- implementation in embedded system

PID pseudo code:

```
% Control algorithm - main loop
while (running)
{
        r=adin(ch1)                % read setpoint from ch1
        y=adin(ch2)                % read process variable from ch2
        P=kp*(r-y)                 % compute proportional part
        D=ad*D-bd*(y-yold)         % update derivative part
        v=P+I+D                    % compute temporary output
        u=sat(v,ulow,uhigh)        % simulate actuator saturation
        daout(ch1)                 % set analog output ch1
        I=I+bi*(r-y)+br*(u-v)      % update integral
        yold=y             % update old process output
        sleep(h)                   % wait until next update interval
}
```

Is there any alternative to tackle the drawbacks?

- make the implementation easy
- keep the constraints formulation for the engineer needs

Yes

- a posteriori structuration with the risk of lack of performance (order reduction, Youla parameter, etc...)
- Choose a structure for the controller (a PID for example)
- Use Optimization... but nonconvex one

Motivation

Observations:

- In Automatic, Robotic, Electronic or Mechanic, engineers know very well their problems.
  $\Longrightarrow$ Physical Sense
- In Optimization, the specification of each solver need to classify a model: LP, NLP, MINLP, SDP, DFO,...
  If the model cannot be classify: Modification, Adaptation, Reformulation, ...
  $\Longrightarrow$ Numerical Sense

Physical Solutions $\Longleftrightarrow$ Numerical Solutions

$\Longrightarrow$ Goal: Propose optimization tools to build the best solver for their own problems.

Definition: Contractor

Let $\mathbb{K} \subseteq \mathbb{R}^n$ be a "feasible" region.

The operator $\mathcal{C}_{\mathbb{K}} : \mathbb{IR}^n \rightarrow \mathbb{IR}^n$ is a contractor for $\mathbb{K}$ if:

$$\forall \mathbf{x} \in \mathbb{IR}^n, \begin{cases} \mathcal{C}_{\mathbb{K}}(\mathbf{x}) \subseteq \mathbf{x}, & \text{(contractance)} \\ \mathcal{C}_{\mathbb{K}}(\mathbf{x}) \cap \mathbb{K} = \mathbf{x} \cap \mathbb{K}. & \text{(completeness)} \end{cases}$$

Example: *Forward-Backward Algorithm*

The operator $\mathcal{C} : \mathbb{IR}^n \rightarrow \mathbb{IR}^n$ is a contractor for the equation $f(x) = 0$, if:

$$\forall \mathbf{x} \in \mathbb{IR}^n, \begin{cases} \mathcal{C}(\mathbf{x}) \subseteq \mathbf{x}, \\ x \in \mathbf{x} \text{ and } f(x) = 0 \Rightarrow x \in \mathcal{C}(\mathbf{x}). \end{cases}$$

General Design

$(\tilde{x}, \tilde{f}) =$ **OptimCtc** $([\mathbf{x}], \mathcal{C}_{out}, \mathcal{C}_{in}, f_{cost})$:

- ⋆ Merging of a Branch&Bound Algorithm based on Interval Analysis (spacialB&B) and a Set Inversion Via Interval Analysis (SIVIA).
- ⋆ $\mathcal{C}_{out}$, $\mathcal{C}_{in}$: contractors designed by the user based on $\mathbb{K}$ and $\overline{\mathbb{K}}$,
- ⋆ $\mathcal{C}_f$: a FwdBwd contractor based on $\{x : f_{cost}(x) \leq \tilde{f}\}$
- ⋆ $\mathcal{B}$: Largest first, smear evaluation, homemade,...

Illustration: $\mathcal{C}_{in}$, $\mathcal{C}_{out}$

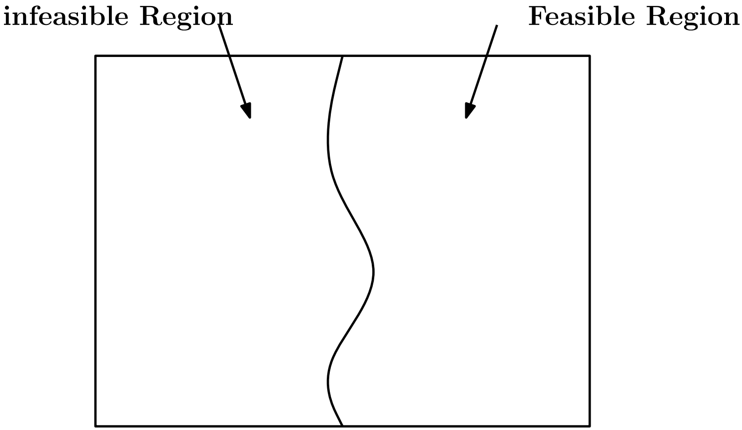


Illustration: $\mathcal{C}_{in}$, $\mathcal{C}_{out}$

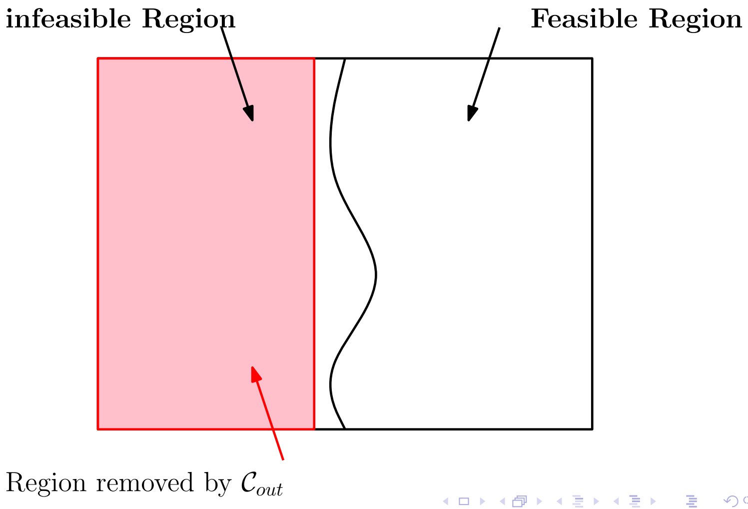
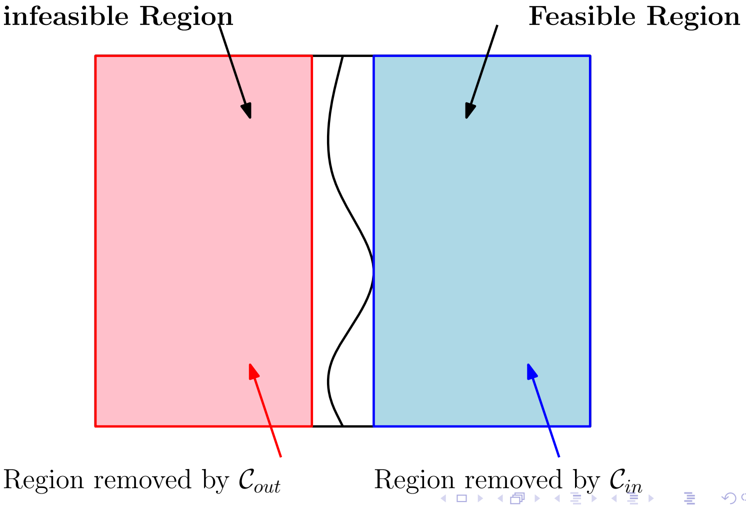


Illustration: $\mathcal{C}_{in}$, $\mathcal{C}_{out}$



The feasability test

Without equation or system,

How to prove that a point is a feasible point?

The feasability test

Without equation or system,

How to prove that a point is a feasible point?

$$\text{Prove that } x \in \mathbb{K} \quad \not\Leftrightarrow \quad \text{Prove that } x \notin \overline{\mathbb{K}}$$

The feasability test

Without equation or system,

How to prove that a point is a feasible point?

Prove that $x \in \mathbb{K}$ $\not\Leftrightarrow$ Prove that $x \notin \overline{\mathbb{K}}$

x is contracted by $\mathcal{C}_{in}$ $\Leftrightarrow$ $x \in \mathbb{K}$ $\Leftrightarrow$ $\mathcal{C}_{out}$ proves that x is in $\mathbb{K}$.

$\mathcal{C}_{in}$ will eliminate all the part of a box which **are not** in $\overline{\mathbb{K}}$.
$\mathcal{C}_{out}$ will eliminate all the part of a box which **are not** in $\mathbb{K}$.

Global Optimization based on Contractor

- $\mathcal{L} := \{(\mathbf{x}, \mathit{false})\}$, The boolean indicate if $\mathbf{x}$ is entirely feasible
- Do
  1. Extract from $\mathcal{L}$ a element $(\mathbf{z}, b)$,
  2. Bisect $\mathbf{z}$ following a bisector $\mathcal{B}$: $(\mathbf{z}_1, \mathbf{z}_2)$
  3. for $j = 1$ to 2 :
     - if $b = \mathit{false}$ *(i.e. **x** is not completly feasible)* then
       Contract the infeasible region using $\mathcal{C}_{out}$ and $\mathcal{C}_f$,
       Extract $\mathbf{z}_{feas}$ a feasible part of $\mathbf{z}_j$ using $\mathcal{C}_{in}$,
       Insert $(\mathbf{z}_{feas}, \mathit{true})$ in $\mathcal{L}$.
       Insert the rest $(\mathbf{z}_j, \mathit{false})$ in $\mathcal{L}$.
     - else *(i.e. **x** is entirely feasible)*
       Contract $\mathbf{z}_j$ using $\mathcal{C}_f$,
       Try to **find** a local optimum without constraint in $[\mathbf{z}_j]$,
       if *succeed* then Update $\tilde{f}$ insert $(\mathbf{z}_j, \mathit{true})$ in $\mathcal{L}$.
- stopping criterion

Global Optimization based on Contractor

- $\mathcal{L} := \{(\mathbf{x}, \mathit{false})\}$, The boolean indicate if $\mathbf{x}$ is entirely feasible
- Do
  1. Extract from $\mathcal{L}$ a element $(\mathbf{z}, b)$,
  2. Bisect $\mathbf{z}$ following a bisector $\mathcal{B}$: $(\mathbf{z}_1, \mathbf{z}_2)$
  3. for $j = 1$ to 2 :
     - if $b = \mathit{false}$ *(i.e. $\mathbf{x}$ is not completly feasible)* then
       Contract the infeasible region using $\mathcal{C}_{out}$ and $\mathcal{C}_f$,
       Extract $\mathbf{z}_{feas}$ a feasible part of $\mathbf{z}_j$ using $\mathcal{C}_{in}$,
       Insert $(\mathbf{z}_{feas}, \mathit{true})$ in $\mathcal{L}$.
       Insert the rest $(\mathbf{z}_j, \mathit{false})$ in $\mathcal{L}$.
     - else *(i.e. $\mathbf{x}$ is entirely feasible)*
       Contract $\mathbf{z}_j$ using $\mathcal{C}_f$,
       Try to **find** a local optimum without constraint in $[\mathbf{z}_j]$,
       if *succeed* then Update $\tilde{f}$ insert $(\mathbf{z}_j, \mathit{true})$ in $\mathcal{L}$.
- stopping criterion

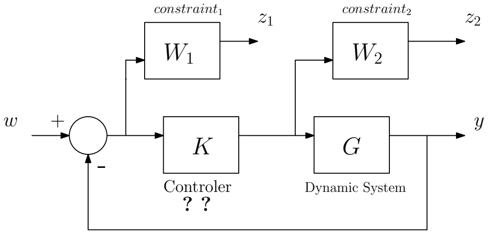
Global Optimization based on Contractor

- $\mathcal{L} := \{(\mathbf{x}, \mathit{false})\}$, The boolean indicate if $\mathbf{x}$ is entirely feasible
- Do
  1. Extract from $\mathcal{L}$ a element $(\mathbf{z}, b)$,
  2. Bisect $\mathbf{z}$ following a bisector $\mathcal{B}$: $(\mathbf{z}_1, \mathbf{z}_2)$
  3. for $j = 1$ to 2 :
     - if $b = \mathit{false}$ *(i.e. **x** is not completly feasible)* then
       Contract the infeasible region using $\mathcal{C}_{out}$ and $\mathcal{C}_f$,
       Extract $\mathbf{z}_{feas}$ a feasible part of $\mathbf{z}_j$ using $\mathcal{C}_{in}$,
       Insert $(\mathbf{z}_{feas}, \mathit{true})$ in $\mathcal{L}$.
       Insert the rest $(\mathbf{z}_j, \mathit{false})$ in $\mathcal{L}$.
     - else *(i.e. **x** is entirely feasible)*
       Contract $\mathbf{z}_j$ using $\mathcal{C}_f$,
       Try to **find** a local optimum without constraint in $[\mathbf{z}_j]$,
       if *succeed* then Update $\tilde{f}$ insert $(\mathbf{z}_j, \mathit{true})$ in $\mathcal{L}$.
- stopping criterion

Global Optimization based on Contractor

- $\mathcal{L} := \{(\mathbf{x}, \mathit{false})\}$, The boolean indicate if $\mathbf{x}$ is entirely feasible
- Do
  1. Extract from $\mathcal{L}$ a element $(\mathbf{z}, b)$,
  2. Bisect $\mathbf{z}$ following a bisector $\mathcal{B}$: $(\mathbf{z}_1, \mathbf{z}_2)$
  3. for $j = 1$ to 2 :
     - if $b = \mathit{false}$ *(i.e.* $\mathbf{x}$ *is not completly feasible)* then
       Contract the infeasible region using $\mathcal{C}_{out}$ and $\mathcal{C}_f$,
       Extract $\mathbf{z}_{feas}$ a feasible part of $\mathbf{z}_j$ using $\mathcal{C}_{in}$,
       Insert $(\mathbf{z}_{feas}, \mathit{true})$ in $\mathcal{L}$.
       Insert the rest $(\mathbf{z}_j, \mathit{false})$ in $\mathcal{L}$.
     - else *(i.e.* $\mathbf{x}$ *is entirely feasible)*
       Contract $\mathbf{z}_j$ using $\mathcal{C}_f$,
       Try to **find** a local optimum without constraint in $[\mathbf{z}_j]$,
       if *succeed* then Update $\tilde{f}$ insert $(\mathbf{z}_j, \mathit{true})$ in $\mathcal{L}$.
- stopping criterion

H_∞ control synthesis under structural constraints



H_∞ control synthesis $\Rightarrow$ Guarantee the robustness and stability

$$||P||_\infty = \sup_\omega(\sigma_{\max}(P(j\omega)))$$

- Classical approach without structural constraint
  $\rightarrow$ LMI system, SDP opimization
- Classical approach **with** structural constraint
  $\rightarrow$ Nonsmooth **local** optimization

Mathematical Modeling

$$\begin{cases} \min_{\mathbf{k},\gamma} & \gamma \\ \forall\omega, & \left\| \dfrac{W_1(j\omega)}{1+G(j\omega)K(j\omega)} \right\|_\infty \leq \gamma, \\ \forall\omega, & \left\| \dfrac{W_2(j\omega)K(j\omega)}{1+G(j\omega)K(j\omega)} \right\|_\infty \leq \gamma, \\ & \text{The closed-loop system must be } \textit{stable}. \end{cases}$$

Stability:

The system is stable iff its poles are strictly negative.

$\Leftrightarrow$

The roots of the denominator of $\frac{1}{1+G(s)K(s)}$ are strictly negative

$\Longrightarrow$ Routh-Hurwitz stability criterion

Routh-Hurwitz stability criterion

$$P(s) = a_n s^n + a_{n-1} s^{n-1} + \cdots + a_1 s + a_0$$

$v_{1,1} = a_n$ $v_{2,1} = a_{n-1}$	$v_{1,2} = a_{n-2}$ $v_{2,2} = a_{n-3}$	$v_{1,3} = a_{n-4}$ $v_{2,3} = a_{n-5}$	$v_{1,4} = a_{n-6}$ $v_{2,4} = a_{n-7}$
$v_{3,1} = \frac{-1}{v_{2,1}} \begin{vmatrix} v_{1,1} & v_{1,2} \\ v_{2,1} & v_{2,2} \end{vmatrix}$	$v_{3,2} = \frac{-1}{v_{2,1}} \begin{vmatrix} v_{1,1} & v_{1,3} \\ v_{2,1} & v_{2,3} \end{vmatrix}$	$v_{3,3} = \frac{-1}{v_{2,1}} \begin{vmatrix} v_{1,1} & v_{1,4} \\ v_{2,1} & v_{2,4} \end{vmatrix}$	$\cdots$
$v_{4,1} = \frac{-1}{v_{3,1}} \begin{vmatrix} v_{2,1} & v_{2,2} \\ v_{3,1} & v_{3,2} \end{vmatrix}$	$v_{4,2} = \frac{-1}{v_{3,1}} \begin{vmatrix} v_{2,1} & v_{2,3} \\ v_{3,1} & v_{3,3} \end{vmatrix}$	$\cdots$	$\cdots$
$v_{5,1} = \frac{-1}{v_{4,1}} \begin{vmatrix} v_{3,1} & v_{3,2} \\ v_{4,1} & v_{4,2} \end{vmatrix}$	$\cdots$	$\cdots$	$\cdots$
$\vdots$	$\ddots$	$\ddots$	$\ddots$

Routh-Hurwitz stability criterion

$$P(s) = a_n s^n + a_{n-1} s^{n-1} + \cdots + a_1 s + a_0$$

$v_{1,1} = a_n$ $v_{2,1} = a_{n-1}$	$v_{1,2} = a_{n-2}$ $v_{2,2} = a_{n-3}$	$v_{1,3} = a_{n-4}$ $v_{2,3} = a_{n-5}$	$v_{1,4} = a_{n-6}$ $v_{2,4} = a_{n-7}$
$v_{3,1} = \frac{-1}{v_{2,1}} \begin{vmatrix} v_{1,1} & v_{1,2} \\ v_{2,1} & v_{2,2} \end{vmatrix}$	$v_{3,2} = \frac{-1}{v_{2,1}} \begin{vmatrix} v_{1,1} & v_{1,3} \\ v_{2,1} & v_{2,3} \end{vmatrix}$	$v_{3,3} = \frac{-1}{v_{2,1}} \begin{vmatrix} v_{1,1} & v_{1,4} \\ v_{2,1} & v_{2,4} \end{vmatrix}$	$\ldots$
$v_{4,1} = \frac{-1}{v_{3,1}} \begin{vmatrix} v_{2,1} & v_{2,2} \\ v_{3,1} & v_{3,2} \end{vmatrix}$	$v_{4,2} = \frac{-1}{v_{3,1}} \begin{vmatrix} v_{2,1} & v_{2,3} \\ v_{3,1} & v_{3,3} \end{vmatrix}$	$\ldots$	$\ldots$
$v_{5,1} = \frac{-1}{v_{4,1}} \begin{vmatrix} v_{3,1} & v_{3,2} \\ v_{4,1} & v_{4,2} \end{vmatrix}$	$\ldots$	$\ldots$	$\ldots$
$\vdots$	$\ddots$	$\ddots$	$\ddots$

If all the value of the **first column** are positive, all roots of P are negative.

Definition of the feasible set

$$\mathbb{K}^1_\omega = \left\{(k,\gamma,\omega) \ : \ \left\| \frac{W_1(i\omega)}{1+G(i\omega)K(i\omega)} \right\|_\infty \leq \gamma \right\},$$

$$\mathbb{K}^2_\omega = \left\{(k,\gamma,\omega) \ : \ \left\| \frac{W_2(i\omega)K(i\omega)}{1+G(i\omega)K(i\omega)} \right\|_\infty \leq \gamma \right\},$$

$$\mathbb{K}^4 = \bigcap_{\omega\in[10^{-2},10^2]} \mathbb{K}^1_\omega \cap \mathbb{K}^2_\omega .$$

The Routh's condition / stability of the closed-loop system:

$$\mathbb{K}^{Routh} = \{(k,\gamma) \ : \ \begin{cases} a_n(k,\gamma) > 0, \\ a_{n-1}(k,\gamma) > 0, \\ v_{2,1}(k,\gamma) > 0, \\ \\ \ldots \end{cases} \}.$$

The feasible set of our problem is $\mathbb{K} = \mathbb{K}^4 \cap \mathbb{K}^{Routh}$.

Contractor Modeling: Properties

Let $\mathcal{A}$ a contractor for the equation $f(x) = 0$, and $\mathcal{B}$ a contractor for the equation $g(x) = 0$, then:

Intersection, Composition

$\mathcal{A} \cap \mathcal{B}$ and $\mathcal{A} \circ \mathcal{B}$ are two contractors for the region:

$$\{x \in \mathbb{R}^n \ : \ f(x) = 0 \ AND \ g(x) = 0\}$$

Union

$\mathcal{A} \cup \mathcal{B}$ is a contractor for the region:

$$\{x \in \mathbb{R}^n \ : \ f(x) = 0 \ OR \ g(x) = 0\}$$

Contractor with Quantifiers

Let $\mathcal{C}$ be a contractor for a set $\mathbb{Z} = \mathbb{X} \times \mathbb{Y}$,
$\pi_{\mathbb{X}}$ the projection of $\mathbb{Z}$ over $\mathbb{X}$.

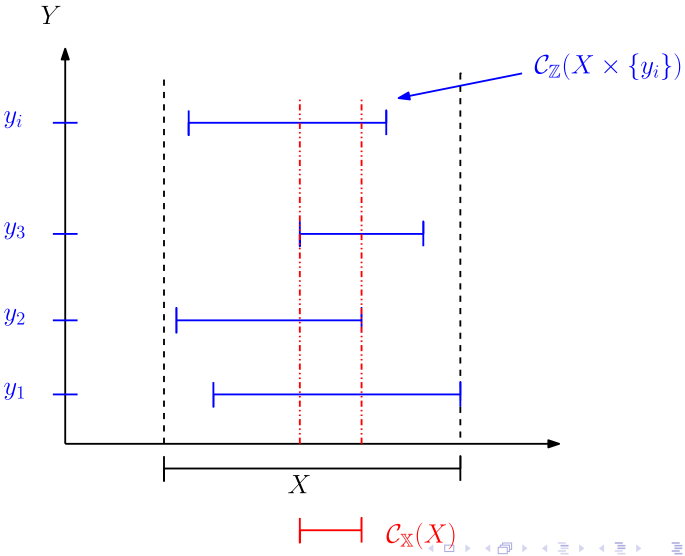
Contractor ForAll / Exists

$$\begin{cases} \mathcal{C}^{\cap\mathbb{Y}}(\mathbf{x}) = \bigcap_{y\in\mathbb{Y}} \pi_{\mathbb{X}}\left(\mathcal{C}(\mathbf{x}\times\{y\})\right), \\ \\ \mathcal{C}^{\cup\mathbb{Y}}(\mathbf{x}) = \bigcup_{y\in\mathbb{Y}} \pi_{\mathbb{X}}\left(\mathcal{C}(\mathbf{x}\times\{y\})\right). \end{cases}$$

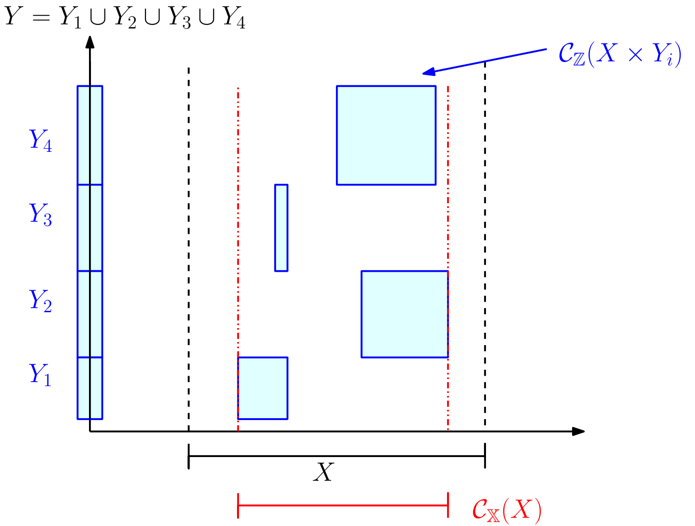
Property

$\mathcal{C}^{\cap\mathbb{Y}}$ is a contractor for $\{x \ : \ \forall y \in \mathbb{Y}, (x,y) \in \mathbb{Z}\}$
$\mathcal{C}^{\cup\mathbb{Y}}$ is a contractor for$\{x \ : \ \exists y \in \mathbb{Y}, (x,y) \in \mathbb{Z}\}$.

Contractor CtcForAll: $\mathbb{X} = \{x \ : \forall y \in \mathbb{Y}, (x, y) \in \mathbb{Z}\}$



Contractor CtcExist: $\mathbb{X} = \{x \;:\; \exists y \in \mathbb{Y}, (x, y) \in \mathbb{Z}\}$



Construction of Contractors $\mathcal{C}_{out}$ of the feasible set $\mathbb{K}$

$\mathcal{C}_{out}$ will eliminate all the part of a box which are not in $\mathbb{K}$.

$$\mathbb{K}^1_\omega = \left\{(k,\gamma,\omega) \ : \ \left\| \frac{W_1(i\omega)}{1+G(i\omega)K(i\omega)} \right\|_\infty \leq \gamma \right\},$$

$$\mathbb{K}^2_\omega = \left\{(k,\gamma,\omega) \ : \ \left\| \frac{W_2(i\omega)K(i\omega)}{1+G(i\omega)K(i\omega)} \right\|_\infty \leq \gamma \right\},$$

$$\mathbb{K} = \left(\bigcap_{\omega \in [10^{-2},10^2]} \mathbb{K}^1_\omega \cap \mathbb{K}^2_\omega \right) \cap \mathbb{K}^{Routh}.$$

Construction of Contractors $\mathcal{C}_{out}$ of the feasible set $\mathbb{K}$

$\mathcal{C}_{out}$ will eliminate all the part of a box which are not in $\mathbb{K}$.

$$\mathbb{K} = \left(\bigcap_{\omega \in [10^{-2}, 10^{2}]} \mathbb{K}^1_\omega \cap \mathbb{K}^2_\omega \right) \cap \mathbb{K}^{Routh}.$$

Construction of Contractors $\mathcal{C}_{out}$ of the feasible set $\mathbb{K}$

$\mathcal{C}_{out}$ will eliminate all the part of a box which are not in $\mathbb{K}$.

$$\mathbb{K} = \left(\bigcap_{\omega \in [10^{-2}, 10^{2}]} \mathbb{K}^1_\omega \cap \mathbb{K}^2_\omega \right) \cap \mathbb{K}^{Routh}.$$

1. Create the contractor $\mathcal{C}_1$, $\mathcal{C}_2$ and $\mathcal{C}_{Routh}$ based on $\mathbb{K}^1_\omega$, $\mathbb{K}^2_\omega$ and $\mathbb{K}^{Routh}$:
   Contractor based on inequality system: Forward-Backward algorithm, HC4-revise, PolytopeHull, ...

Construction of Contractors $\mathcal{C}_{out}$ of the feasible set $\mathbb{K}$

$\mathcal{C}_{out}$ will eliminate all the part of a box which are not in $\mathbb{K}$.

$$\mathbb{K} = \left(\bigcap_{\omega \in [10^{-2}, 10^2]} \mathbb{K}^1_\omega \cap \mathbb{K}^2_\omega \right) \cap \mathbb{K}^{Routh}.$$

1. Create the contractor $\mathcal{C}_1$, $\mathcal{C}_2$ and $\mathcal{C}_{Routh}$ based on $\mathbb{K}^1_\omega$, $\mathbb{K}^2_\omega$ and $\mathbb{K}^{Routh}$:
   Contractor based on inequality system: Forward-Backward algorithm, HC4-revise, PolytopeHull, . . .
2. Inter: $\mathcal{C}_3(\mathbf{k}, \gamma, \omega) = \mathcal{C}_1(\mathbf{k}, \gamma, \omega) \cap \mathcal{C}_2(\mathbf{k}, \gamma, \omega)$.

Construction of Contractors $\mathcal{C}_{out}$ of the feasible set $\mathbb{K}$

$\mathcal{C}_{out}$ will eliminate all the part of a box which are not in $\mathbb{K}$.

$$\mathbb{K} = \left(\bigcap_{\omega \in [10^{-2}, 10^{2}]} \mathbb{K}^1_\omega \cap \mathbb{K}^2_\omega \right) \cap \mathbb{K}^{Routh}.$$

1. Create the contractor $\mathcal{C}_1$, $\mathcal{C}_2$ and $\mathcal{C}_{Routh}$ based on $\mathbb{K}^1_\omega$, $\mathbb{K}^2_\omega$ and $\mathbb{K}^{Routh}$:
   Contractor based on inequality system: Forward-Backward algorithm, HC4-revise, PolytopeHull, ...
2. Inter: $\mathcal{C}_3(\mathbf{k}, \gamma, \omega) = \mathcal{C}_1(\mathbf{k}, \gamma, \omega) \cap \mathcal{C}_2(\mathbf{k}, \gamma, \omega)$.
3. CtcForAll: $\mathcal{C}^{\cap\omega}(\mathbf{k}, \gamma) = \bigcap_{\omega \in [10^{-2}, 10^{2}]} \mathcal{C}_3(\mathbf{k}, \gamma, \omega)$.

Construction of Contractors $\mathcal{C}_{out}$ of the feasible set $\mathbb{K}$

$\mathcal{C}_{out}$ will eliminate all the part of a box which are not in $\mathbb{K}$.

$$\mathbb{K} = \left(\bigcap_{\omega \in [10^{-2}, 10^{2}]} \mathbb{K}^1_\omega \cap \mathbb{K}^2_\omega \right) \cap \mathbb{K}^{Routh}.$$

1. Create the contractor $\mathcal{C}_1$, $\mathcal{C}_2$ and $\mathcal{C}_{Routh}$ based on $\mathbb{K}^1_\omega$, $\mathbb{K}^2_\omega$ and $\mathbb{K}^{Routh}$:
   Contractor based on inequality system: Forward-Backward algorithm, HC4-revise, PolytopeHull, . . .
2. Inter: $\mathcal{C}_3(\mathbf{k}, \gamma, \omega) = \mathcal{C}_1(\mathbf{k}, \gamma, \omega) \cap \mathcal{C}_2(\mathbf{k}, \gamma, \omega)$.
3. CtcForAll: $\mathcal{C}^{\cap\omega}(\mathbf{k}, \gamma) = \bigcap_{\omega \in [10^{-2}, 10^{2}]} \mathcal{C}_3(\mathbf{k}, \gamma, \omega)$.
4. Inter: $\mathcal{C}_{out} = \mathcal{C}^{\cap\omega} \cap \mathcal{C}_{Routh}$.

Construction of Contractors $\mathcal{C}_{in}$ of the unfeasible set $\overline{\mathbb{K}}$

$\mathcal{C}_{in}$ will eliminate all the part of a box which are not in $\overline{\mathbb{K}}$.

$$\overline{\mathbb{K}^1_\omega} = \left\{(k,\gamma,\omega) \; : \; \left\| \frac{W_1(i\omega)}{1+G(i\omega)K(i\omega)} \right\| > \gamma \right\},$$

$$\overline{\mathbb{K}^2_\omega} = \left\{(k,\gamma,\omega) \; : \; \left\| \frac{W_2(i\omega)K(i\omega)}{1+G(i\omega)K(i\omega)} \right\| > \gamma \right\},$$

$$\overline{\mathbb{K}} = \left(\bigcup_{\omega \in [10^{-2}, 10^2]} \overline{\mathbb{K}^1_\omega} \cup \overline{\mathbb{K}^2_\omega} \right) \cup \overline{\mathbb{K}^{Routh}}.$$

Construction of Contractors $\mathcal{C}_{in}$ of the unfeasible set $\overline{\mathbb{K}}$

$\mathcal{C}_{in}$ will eliminate all the part of a box which are not in $\overline{\mathbb{K}}$.

$$\overline{\mathbb{K}} = \left(\bigcup_{\omega \in [10^{-2}, 10^{2}]} \overline{\mathbb{K}^{1}_{\omega}} \cup \overline{\mathbb{K}^{2}_{\omega}} \right) \cup \overline{\mathbb{K}^{Routh}}.$$

Construction of Contractors $\mathcal{C}_{in}$ of the unfeasible set $\overline{\mathbb{K}}$

$\mathcal{C}_{in}$ will eliminate all the part of a box which are not in $\overline{\mathbb{K}}$.

$$\overline{\mathbb{K}} = \left(\bigcup_{\omega \in [10^{-2}, 10^2]} \overline{\mathbb{K}^1_\omega} \cup \overline{\mathbb{K}^2_\omega} \right) \cup \overline{\mathbb{K}^{Routh}}.$$

1. Create the contractor $\mathcal{C}_{\overline{1}}$, $\mathcal{C}_{\overline{2}}$ and $\mathcal{C}_{\overline{Routh}}$ based on $\overline{\mathbb{K}^1_\omega}$, $\overline{\mathbb{K}^2_\omega}$ and $\overline{\mathbb{K}^{Routh}}$:
   Contractor based on inequality system: Forward-Backward algorithm, HC4-revise, PolytopeHull, . . .

Construction of Contractors $\mathcal{C}_{in}$ of the unfeasible set $\overline{\mathbb{K}}$

$\mathcal{C}_{in}$ will eliminate all the part of a box which are not in $\overline{\mathbb{K}}$.

$$\overline{\mathbb{K}} = \left(\bigcup_{\omega \in [10^{-2}, 10^2]} \overline{\mathbb{K}^1_\omega} \cup \overline{\mathbb{K}^2_\omega} \right) \cup \overline{\mathbb{K}^{Routh}}.$$

1. Create the contractor $\mathcal{C}_{\overline{1}}$, $\mathcal{C}_{\overline{2}}$ and $\mathcal{C}_{\overline{Routh}}$ based on $\overline{\mathbb{K}^1_\omega}$, $\overline{\mathbb{K}^2_\omega}$ and $\overline{\mathbb{K}^{Routh}}$.
   Contractor based on inequality system: Forward-Backward algorithm, HC4-revise, PolytopeHull, ...
2. Union: $\mathcal{C}_{\overline{3}}(\mathbf{k}, \gamma, \omega) = \mathcal{C}_{\overline{1}}(\mathbf{k}, \gamma, \omega) \cup \mathcal{C}_{\overline{2}}(\mathbf{k}, \gamma, \omega)$.

Construction of Contractors $\mathcal{C}_{in}$ of the unfeasible set $\overline{\mathbb{K}}$

Control for Robots

SHARC 2016

$\mathcal{C}_{in}$ will eliminate all the part of a box which are not in $\overline{\mathbb{K}}$.

$$\overline{\mathbb{K}} = \left(\bigcup_{\omega \in [10^{-2}, 10^{2}]} \overline{\mathbb{K}^1_\omega} \cup \overline{\mathbb{K}^2_\omega} \right) \cup \overline{\mathbb{K}^{Routh}}.$$

1. Create the contractor $\mathcal{C}_{\overline{1}}$, $\mathcal{C}_{\overline{2}}$ and $\mathcal{C}_{\overline{Routh}}$ based on $\overline{\mathbb{K}^1_\omega}$, $\overline{\mathbb{K}^2_\omega}$ and $\overline{\mathbb{K}^{Routh}}$:
   Contractor based on inequality system: Forward-Backward algorithm, HC4-revise, PolytopeHull, . . .
2. Union: $\mathcal{C}_{\overline{3}}(\mathbf{k}, \gamma, \omega) = \mathcal{C}_{\overline{1}}(\mathbf{k}, \gamma, \omega) \cup \mathcal{C}_{\overline{2}}(\mathbf{k}, \gamma, \omega)$.
3. CtcExist: $\mathcal{C}^{\cup\omega}(\mathbf{k}, \gamma) = \bigcup_{\omega \in [10^{-2}, 10^{2}]} \mathcal{C}_{\overline{3}}(\mathbf{k}, \gamma, \omega)$.

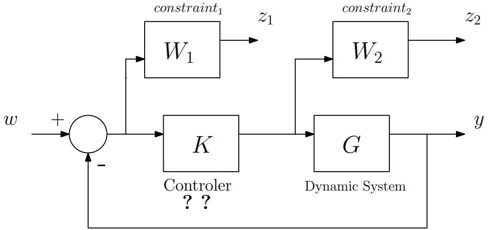
Construction of Contractors $\mathcal{C}_{in}$ of the unfeasible set $\overline{\mathbb{K}}$

$\mathcal{C}_{in}$ will eliminate all the part of a box which are not in $\overline{\mathbb{K}}$.

$$\overline{\mathbb{K}} = \left(\bigcup_{\omega\in[10^{-2},10^{2}]} \overline{\mathbb{K}^1_\omega} \cup \overline{\mathbb{K}^2_\omega} \right) \cup \overline{\mathbb{K}^{Routh}}.$$

1. Create the contractor $\mathcal{C}_{\overline{1}}$, $\mathcal{C}_{\overline{2}}$ and $\mathcal{C}_{\overline{Routh}}$ based on $\overline{\mathbb{K}^1_\omega}$, $\overline{\mathbb{K}^2_\omega}$ and $\overline{\mathbb{K}^{Routh}}$:
   Contractor based on inequality system: Forward-Backward algorithm, HC4-revise, PolytopeHull, ...
2. Union: $\mathcal{C}_{\overline{3}}(\mathbf{k},\gamma,\omega) = \mathcal{C}_{\overline{1}}(\mathbf{k},\gamma,\omega) \cup \mathcal{C}_{\overline{2}}(\mathbf{k},\gamma,\omega)$.
3. CtcExist: $\mathcal{C}^{\cup\omega}(\mathbf{k},\gamma) = \bigcup_{\omega\in[10^{-2},10^{2}]} \mathcal{C}_{\overline{3}}(\mathbf{k},\gamma,\omega)$.
4. Union: $\mathcal{C}_{in} = \mathcal{C}^{\cup\omega} \cup \mathcal{C}_{\overline{Routh}}$.

First Application with second order dynamic system



The transfer function of the dynamic system:

$$G(s) = \frac{1}{s^2 + 1.4s + 1}, \quad K(s) = k_p + \frac{k_i}{s} + \frac{k_d s}{1+s}.$$

$$W_1(s) = \frac{s+100}{100s+1}, \qquad W_2(s) = \frac{10s+1}{s+10}.$$

Overview of the equation

$$\forall \omega \in [10^{-2}, 10^{2}], \left\| \frac{W_1(j\omega)}{1 + G(j\omega)K(j\omega)} \right\|_\infty \leq \gamma.$$

$$\Longleftrightarrow$$

$$\forall \omega \in [10^{-2}, 10^{2}], \frac{w^2\left(w^2+1.0\right)\left(w^2+10000.0\right)\left(25.0\,w^4-1.0\,w^2+25.0\right)}{\left(10000.0\,w^2+1.0\right) f_1(\mathbf{k},\gamma,\omega)} \leq \gamma$$

Overview of the equation

$$\forall \omega \in [10^{-2}, 10^{2}], \left\| \frac{W_1(j\omega)}{1 + G(j\omega)K(j\omega)} \right\|_\infty \leq \gamma.$$

$$\Longleftrightarrow$$

$$\forall \omega \in [10^{-2}, 10^{2}], \frac{w^2\left(w^2+1.0\right)\left(w^2+10000.0\right)\left(25.0\,w^4-1.0\,w^2+25.0\right)}{\left(10000.0\,w^2+1.0\right) f_1(\mathbf{k},\gamma,\omega)} \leq \gamma$$

$$f_1(\mathbf{k},\gamma,\omega) = 25.0\mathrm{kd}^2 w^4 + 25.0\mathrm{kp}^2 w^2 + 25.0\mathrm{kp}^2 w^4 - 1.0\mathrm{ki}\left(50.0\mathrm{kd} w^2 + 70.0 w^2 + 70.0 w^4\right) + \mathrm{ki}^2\left(25.0 w^2 + 25.0\right) + 120.0\mathrm{kd} w^4 - 50.0\mathrm{kd} w^6 + 50.0\mathrm{kp} w^2 - 50.0\mathrm{kp} w^6 + 25.0 w^2 + 24.0 w^4 + 24.0 w^6 + 25.0 w^8 + 50.0\mathrm{kdkp} w^4$$

Overview of the equation

$$\forall \omega \in [10^{-2}, 10^{2}], \left\| \frac{W_1(j\omega)}{1 + G(j\omega)K(j\omega)} \right\|_\infty \leq \gamma.$$

$$\Longleftrightarrow$$

$$\forall \omega \in [10^{-2}, 10^{2}], \frac{w^2\left(w^2+1.0\right)\left(w^2+10000.0\right)\left(25.0\,w^4-1.0\,w^2+25.0\right)}{\left(10000.0\,w^2+1.0\right) f_1(\mathbf{k},\gamma,\omega)} \leq \gamma$$

$$f_1(\mathbf{k}, \gamma, \omega) = 25.0\mathrm{kd}^2 w^4 + 25.0\mathrm{kp}^2 w^2 + 25.0\mathrm{kp}^2 w^4 - 1.0\mathrm{ki}\left(50.0\mathrm{kd}w^2 + 70.0w^2 + 70.0w^4\right) + \mathrm{ki}^2\left(25.0w^2 + 25.0\right) + 120.0\mathrm{kd}w^4 - 50.0\mathrm{kd}w^6 + 50.0\mathrm{kp}w^2 - 50.0\mathrm{kp}w^6 + 25.0w^2 + 24.0w^4 + 24.0w^6 + 25.0w^8 + 50.0\mathrm{kdkp}w^4$$

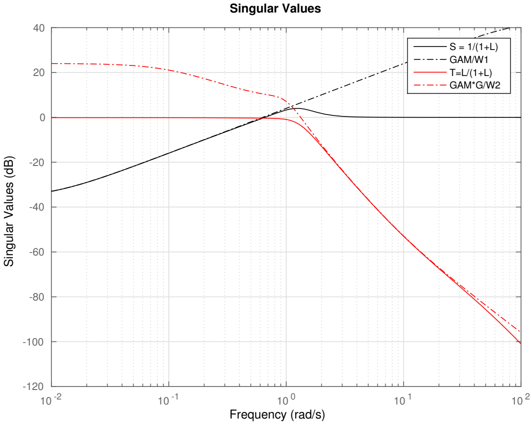
$$\forall u \in [-2, 2], \omega = 10^u.$$

Comparing Results

The same problem is proposed with 2 existant tools and compared with the new approach.

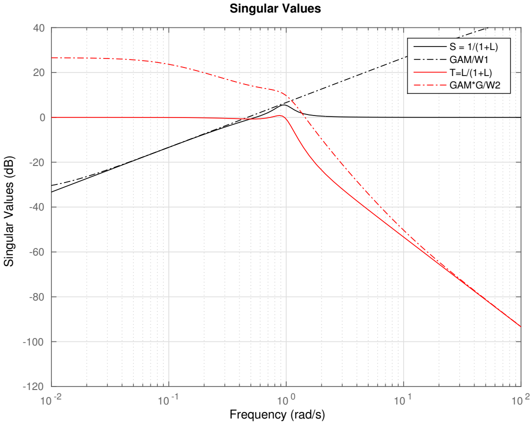
1. HINFSYN of Matlab - full order controller with convex optimization based on LMI ($\gamma = 1.5887$);
2. HINFSTRUCT of Matlab - structured controller with local optimization ($\gamma = 2.1414$).
3. Global Optimization of IBEX ($\gamma = 2.1414$)

Results with HINFSYN of Matlab



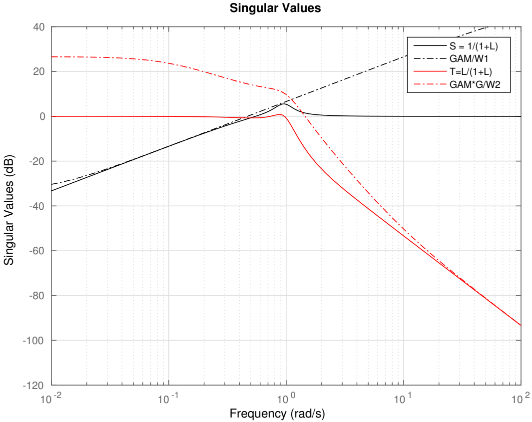
$$\gamma = 1.5887$$

Results with HINFSTRUCT of Matlab



$$\gamma = 2.1414$$

Results with Global Optimization of IBEX

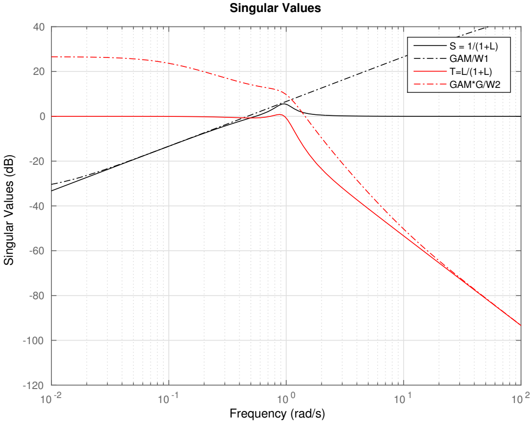


$$\gamma = 2.1414$$

$\Longrightarrow$ same result as with HINFSTRUCT,

but with a global optimality proof!

Results with Global Optimization of IBEX



$$\gamma = 2.1414$$

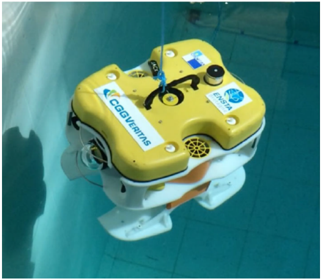
$\Longrightarrow$ same result as with HINFSTRUCT,
but with a global optimality proof!

To keep in mind

Contractor Programming:

- Generates the Modeling and the adapted Solver in the same time,
- Consider heterogeneous constraints without changing the solver,
- Give all the tools to the expert of the application.

AUV CISCREA



Size	0.525m (L) 0.406m (W) 0.395m (H)
Weight in air	15.56kg (without payload and floats)
Degrees of Freedom	Surge, Sway, Heave and Yaw
Propulsion	2 vertical and 4 horizontal propellers
Speed	2 knots (Surge) and 1 knot (Sway, Heave)
Depth Rating	50m
On-board Battery	2-4 hours

AUV CISCREA model

Rigid-body dynamic:

$$M_{RB}\dot{\nu} + C_{RB}(\nu)\nu = \tau_{env} + \tau_{hydro} + \tau_{pro} \quad (1)$$

Hydrodynamic formulations:

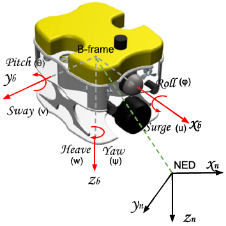
$$\tau_{hydro} = -M_A\dot{\nu} - C_A(\nu)\nu - D(|\nu|)\nu - g(\eta) \quad (2)$$

Damping:

$$D(|\nu|) = D_L + D_N|\nu|\nu \quad (3)$$

Parameter	Description
M_{RB}	AUV rigid-body mass and inertia matrix
M_A	Added mass matrix
C_{RB}	Rigid-body induced coriolis-centripetal matrix
C_A	Added mass induced coriolis-centripetal matrix
$D(\|\nu\|)$	Damping matrix
$g(\eta)$	Restoring forces and moments vector
τ_{env}	Environmental disturbances(wind,waves and currents)
τ_{hydro}	Vector of hydrodynamic forces and moments
τ_{pro}	Propeller forces and moments vector

AUV CISCREA Yaw model



We consider that there are no dependencies between the yaw dynamic and dynamics along other axis.
Resulting Yaw dynamic:

$$(I_{YRB} + I_{YA})\ddot{x} + D_{YN}|\dot{x}|\dot{x} + D_{YL}\dot{x} = K_t\tau_i \quad (4)$$

However, H_∞ synthesis requires a linear system. Thus, the CISCREA yaw model could be linearized as:

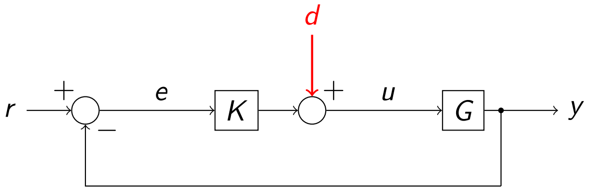
$$(I_{YRB} + I_{YA})\ddot{x} + (D_{YLA} + \delta)\dot{x} = K_t\tau_i, \quad (5)$$

Control objectives specifications

We aim to synthesize a controller to meet the following objectives:

1. Small tracking error e.
2. External perturbation rejection.

External perturbation can be modeled as a control disturbance signal d.



H_∞ formulation

The two objectives can be formulate as H_∞ constraints:

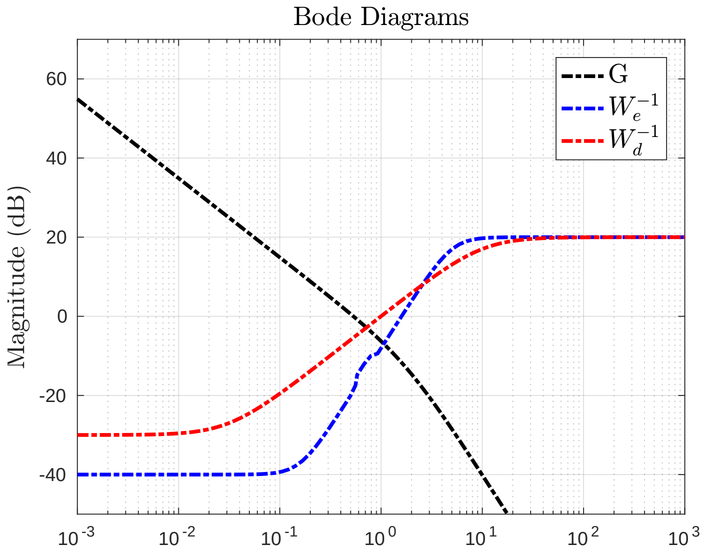
1. Small tracking error:

$$\frac{|e(i\omega)|}{|r(i\omega)|} \leq |W_e^{-1}(i\omega)| \iff ||T_{r\rightarrow e}(i\omega)W_e(i\omega)||_\infty \leq 1$$

2. External perturbation rejection:

$$\frac{|y(i\omega)|}{|d(i\omega)|} \leq |W_y^{-1}(i\omega)| \iff ||T_{d\rightarrow y}(i\omega)W_y(i\omega)||_\infty \leq 1$$

Bode diagrams of Weighted functions



Min Max Problem

- The controller $K(k,s)$ depends on free parameters k.
- $T_{r\to e}(k,s) = \frac{1}{1+G(s)K(k,s)}$ depends on k
- $T_{d\to y}(k,s) = \frac{G(s)}{1+G(s)K(k,s)}$ depends on k

The constraint satisfaction problem is:

$$\text{Find } k,\ \max(||T_{r\to e}(k,s)W_e(s)||_\infty, ||T_{d\to y}(k,s)W_y(s)||_\infty) \leq 1$$

- $||T(s)||_\infty = \sup_\omega |T(i\omega)|$

The Min Max problem is:

$$\min_k \sup_{\omega\geq 0}\{\max(|T_{r\to e}(k,s)W_y(s)|, |T_{d\to y}(k,s)W_y(s)|)\}$$

Solving the Min Max problem

We solve the Min Max problem with Global optimization based on interval analysis.

- Existing methods are based on local optimization. They only provide an upper bound of the objective function.
- Global optimization provides an enclosure of the objective function. It is possible to prove that the CSP (*Constraint Satisfaction Problem*) is not feasible.

Uncertainties

The model of the CISCREA carries uncertainties. The controller is synthesize from a nominal model, and robustness to uncertainties must be analyzed.

- An uncertainty is represented by an interval: $\mathbf{p}$ is the vector of uncertainties.
- $G_\Delta(s, p)$, $p \in \mathbf{p}$ describe the uncertain system.
- The closed loop system stability and performances are robust if and only if: $\forall p \in \mathbf{p}$,

$$\max(||T_{\Delta r \to e}(p, s) W_e(s)||_\infty, ||T_{\Delta d \to y}(p, s) W_y(s)||_\infty) \leq 1$$

- The robustness condition can be validated with interval analysis in a reliable way.

Controller synthesis

- PID controller:

$$K(k,s) = k_p + \frac{k_i}{s} + \frac{k_d s}{1+\tau s}$$

- $k = (k_p, k_i, k_d, \tau)$
- CISCREA model:

$$G(s) = \frac{6.725}{s^2 + 2s}$$

- Weighting functions:

$$W_e = \frac{0.1s^2 + 0.7109s + 2.527}{s^2 + 0.2248s + 0.02527}, \quad W_y = \frac{0.1s + 0.9935}{s + 0.03142}$$

- k is searched in $[0,2]^4$

Controller synthesis

- Solution to the Min Max problem computed: $k^* = (1.987, 1.731, 0.638, 0.001)$
- $||T_{r\rightarrow e}(k^*, s)||_\infty = 0.325$
- $||T_{d\rightarrow y}(k^*, s)||_\infty = 0.154$
- $\min_{k} \sup_{\omega\geq 0}\{\max(|T_{r\rightarrow e}(k,s)W_y(s)|, |T_{d\rightarrow y}(k,s)W_y(s)|)\} \in [0.225, 0.325]$

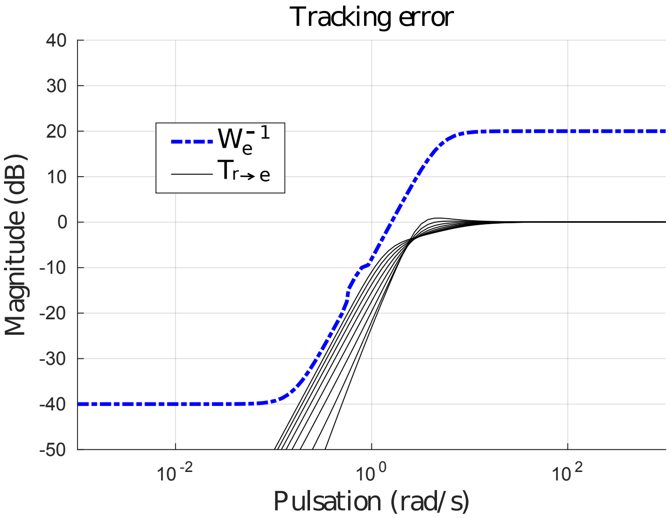
Robustness Analysis

Uncertain CISCREA model:

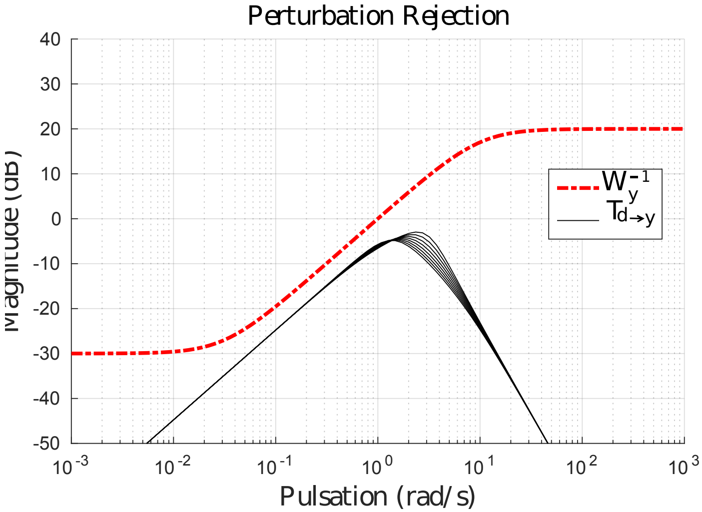
$$G_\Delta(s,p) = \frac{6.725}{s^2 + ps},\ p \in [0,4]$$

- $||T_{\Delta r \to e}(k^*, s, p)||_\infty \leq 0.82$
- $||T_{\Delta d \to y}(k^*, s, p)||_\infty \leq 0.162$
- $||T_{r \to e}(k^*, s)||_\infty = 0.325$
- $||T_{d \to y}(k^*, s)||_\infty = 0.154$

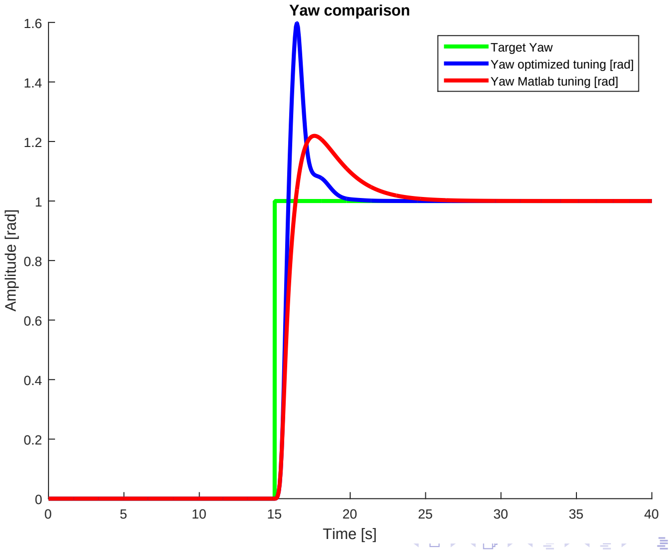
Tracking error constraint



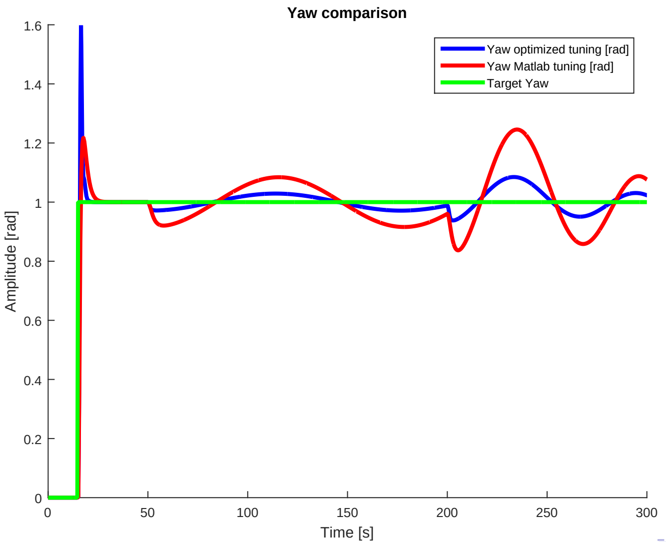
Perturbation rejection



Step response without perturbation



Step response with perturbation



Conclusion for the robot control

- Robust control synthesis method based on global optimization: the optimal PID
- Robustness analysis with respect to uncertainties with experiments on a real underwater robot

Conclusions

- Need: structured control based on end-used demand
- Answers : an original approach based on global optimization (change the hegemony of SPD)
- Perspectives: generalization of the concept for nonlinear control, temporal specifications, etc...
- Others applications:

References

These results are obtained with the collaboration of Jordan NININ (Associate Professor), Dominique MONNET (PhD Student) and Juan Luis ROSENDO (PhD Student)

- D. Arzelier, B. Clement et D. Peaucelle. Multi-objective H_2/H_∞/Impulse-to-Peak Control of a Space Launch Vehicle. European Journal of Control, vol. 12, no. 1, 2006
- N. Imbert et B. Clement. Launcher Attitude control : some answers to the robustness issue. In Proc. of *16th IFAC Symposium on Automatic Control in Aerospace*, Saint-Petersburg, 2004
- D. Monnet, J. Ninin and B. Clement. *Mathematical Aspects of Computer and Information Sciences*, chapter Global Optimization of Hinfinity Problems : Application to Robust Control Synthesis Under Structural Constraints. Springer 2016
- J. Ninin, D. Monnet, and B. Clement. Global Optimization based on Contractor Programming. In *8th Small Workshop on Interval Methods*, Prague, 2015
- J.L. Rosendo, B. Clement and F. Garelli, Sliding mode reference conditioning for path following applied to an AUV. In Proc. of *10th IFAC Conference on Control Applications in Marine Systems*, Norway, 2016.