

Calcul ensembliste pour l'automatique
et ses applications en estimation,
commande robuste et robotique

Luc Jaulin, Guillaume Baffet,
Xavier Baguenard, Massa Dao, Pau Herrero,
LISA-Angers

Pour les JNAs 2003, Valenciennes,
Jeudi 26 juin 2003, 14h,
Session plénière, axe modélisation.

Plan

I. Motivations

- Ensemble de vraisemblance

- Domaine de stabilité

II. Propagation de contraintes sur les intervalles

- Calcul par intervalles

- Propagation de contraintes

- Décomposition en contraintes primitives

- Résolution d'équations non-linéaires

III. Applications

- Estimation à erreurs bornées

- Stabilité robuste

- Estimation d'état

- Planification de trajectoire

Motivations

Modèle : $y_m(\mathbf{p}, t) = p_1 e^{-p_2 t}$.

Paramètres : p_1, p_2 .

Temps de mesure : $t_1, t_2, \dots, t_m$

Barres de mesure : $[y_1^-, y_1^+], [y_2^-, y_2^+], \dots, [y_m^-, y_m^+]$

Ensemble de vraisemblance

$$\begin{aligned} \mathbb{S} = \{ \mathbf{p} \in \mathbb{R}^2, & \ y_m(\mathbf{p}, t_1) \in [y_1^-, y_1^+], \\ & \dots \\ & \ y_m(\mathbf{p}, t_m) \in [y_m^-, y_m^+] \}. \end{aligned}$$

Logiciel SetDemo
fait par Guillaume Baffet (étudiant en maîtrise de
physique et applications, Angers)

Logiciel disponible sur
[http://www.istia.univ-
angers.fr/~jaulin/demo.html](http://www.istia.univ-angers.fr/~jaulin/demo.html)

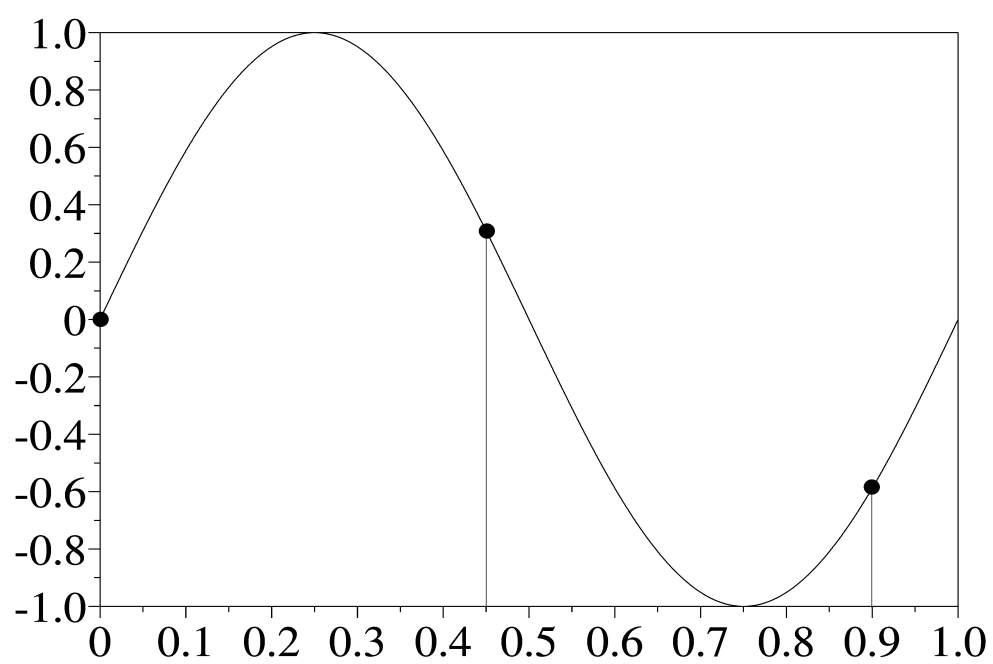
(illustration de l'ensemble de vraisemblance par
Setdemo)

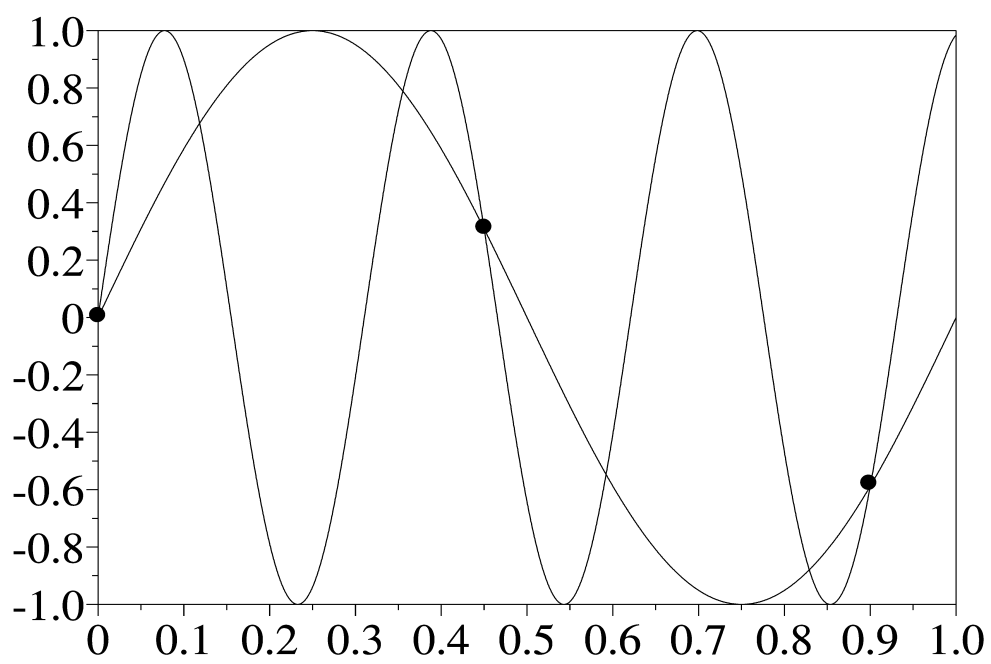
Modèle : $y_m(\mathbf{p}, t) = p_1 \sin(2\pi p_2 t)$.

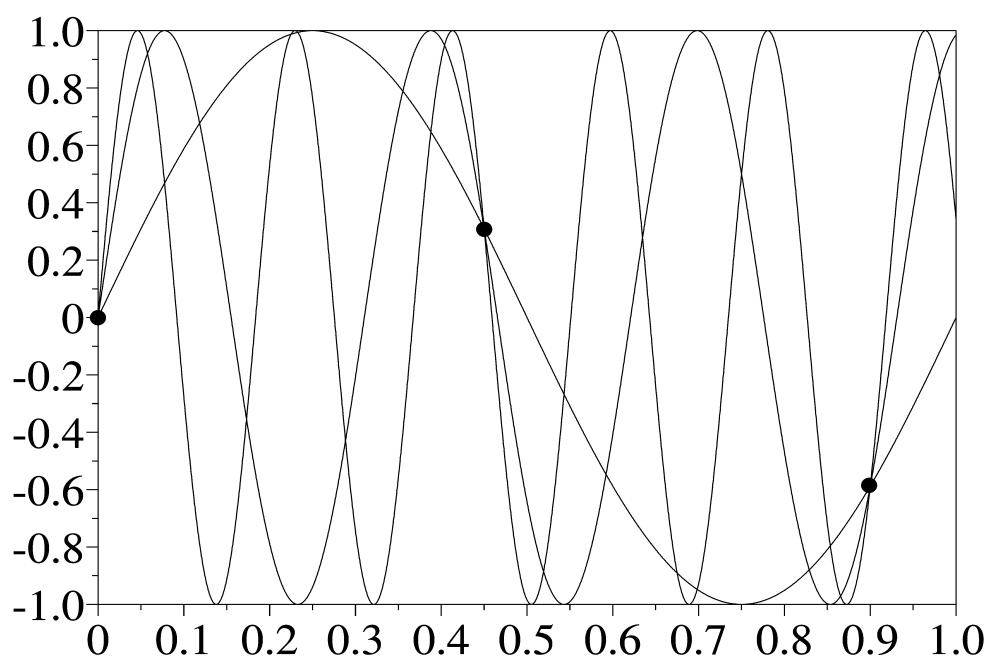
Paramètres : p_1, p_2 .

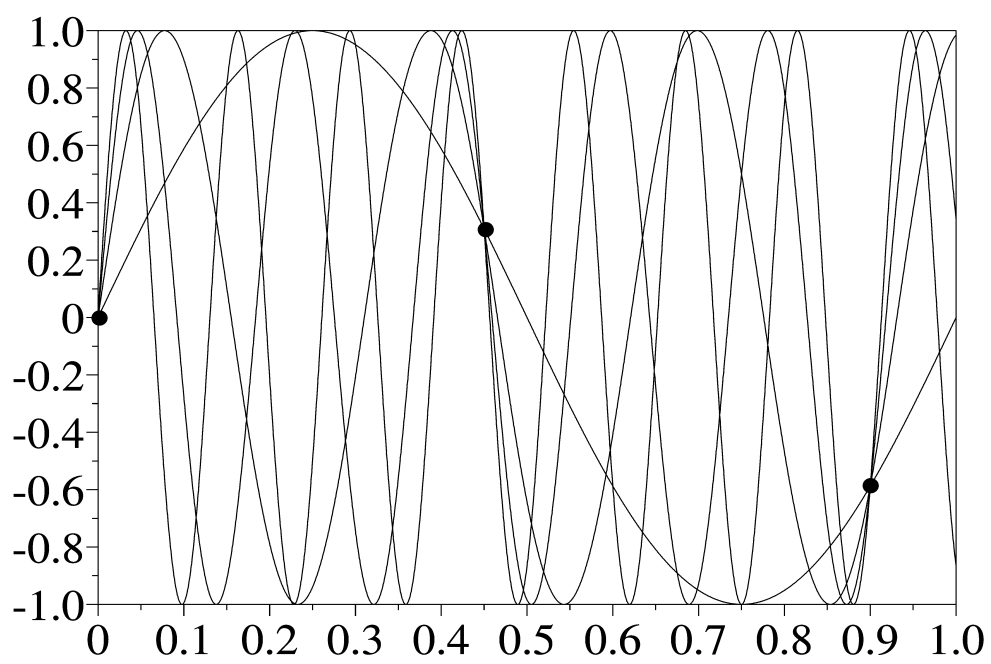
Temps de mesure : $t_1 = 0, t_2 = T_e, t_3 = 2T_e, \dots$

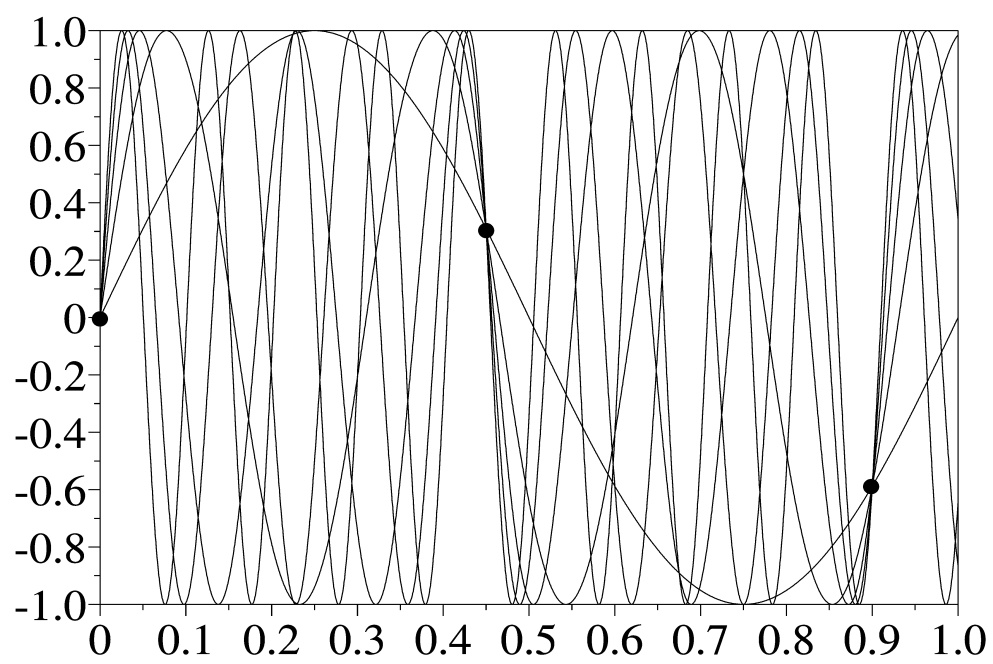
Barres de mesure : $[y_1^-, y_1^+], [y_2^-, y_2^+], [y_3^-, y_3^+], \dots$



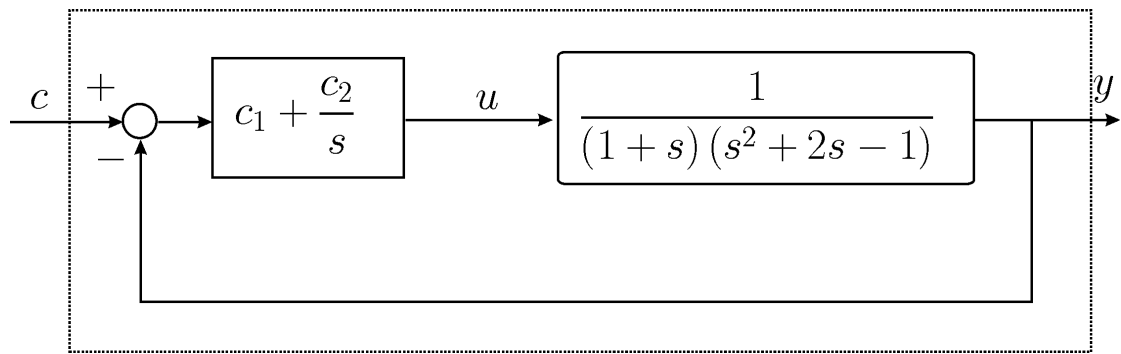








(Montrer l'ensemble de vraisemblance avec SetDemo).



Domaine de stabilité $\mathbb{S}$: ensemble des c_1, c_2 tels que le système bouclé soit stable.

(Montrer le domaine de stabilité avec Setdemo)

Calcul par intervalles

Si $\diamond \in \{+, -, \cdot, /, \max, \min\}$

$$[x] \diamond [y] = [\{x \diamond y \mid x \in [x], y \in [y]\}].$$

Par exemple,

$$[-1, 3] + [2, 5] = [1, 8],$$

$$[-1, 3] \cdot [2, 5] = [-5, 15],$$

$$[-1, 3]/[2, 5] = [-\frac{1}{2}, \frac{3}{2}],$$

$$\max([-1, 3], [2, 5]) = [2, 5].$$

$$\begin{aligned}
[x^-, x^+] + [y^-, y^+] &= [x^- + y^-, x^+ + y^+], \\
[x^-, x^+].[y^-, y^+] &= [\min(x^-y^-, x^+y^-, x^-y^+, x^+y^+), \\
&\quad \max(x^-y^-, x^+y^-, x^-y^+, x^+y^+)], \\
\max([x^-, x^+], [y^-, y^+]) &= [\max(x^-, y^-), \max(x^+, y^+)].
\end{aligned}$$

Si $f \in \{\cos, \sin, \text{sqr}, \text{sqrt}, \log, \exp, \dots\}$

$$f([x]) = [\{f(x) \mid x \in [x]\}].$$

Par exemple,

$$\sin([0, \pi]) = [0, 1],$$

$$\text{sqr}([-1, 3]) = [-1, 3]^2 = [0, 9],$$

$$\text{abs}([-7, 1]) = [0, 7],$$

$$\text{sqrt}([-10, 4]) = \sqrt{[-10, 4]} = [0, 2],$$

$$\log([-2, -1]) = \emptyset.$$

Projection de contraintes

Soient 3 variables x, y, z telles que

$$x \in [1, 5],$$

$$y \in [2, 4],$$

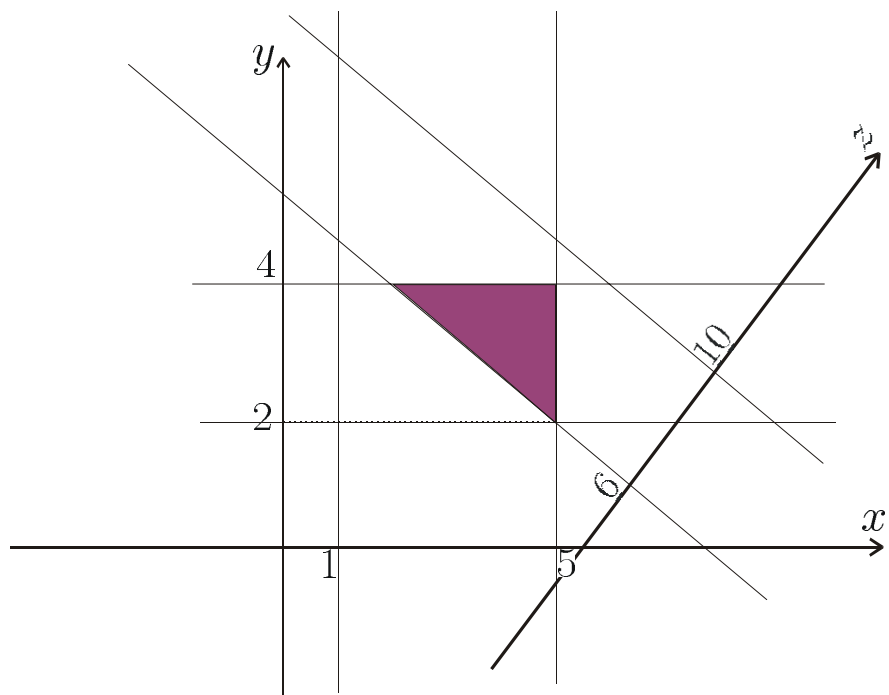
$$z \in [6, 10],$$

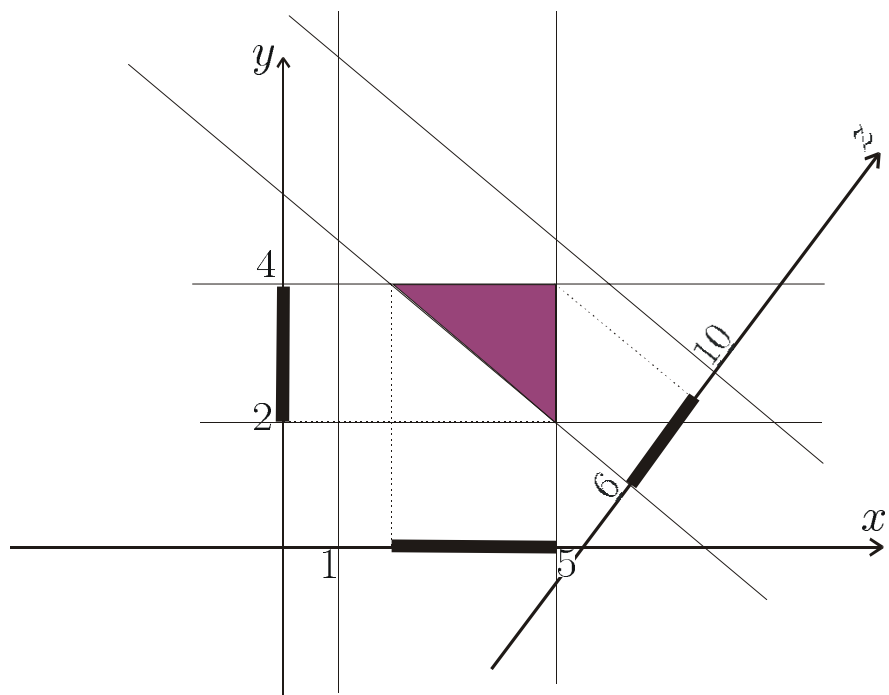
$$z = x + y.$$

Les valeurs 1 pour x et 10 pour z sont dites inconsistantes.

Projeter une contrainte (ici, $z = x + y$), c'est calculer les plus petits intervalles ne contenant que des valeurs consistantes. Pour notre exemple, cela revient à projeter 3 fois (suivant x, y , puis z) le sous-ensemble de $\mathbb{R}^3$ donné par

$$\mathbb{S} = \{(x, y, z) \in [1, 5] \times [2, 4] \times [6, 10] \mid z = x + y\} .$$





Méthode de projection

$$\begin{aligned} z = x + y \Rightarrow z &\in [6, 10] \cap ([1, 5] + [2, 4]) \\ &= [6, 10] \cap [3, 9] = [6, 9]. \end{aligned}$$

$$\begin{aligned} x = z - y \Rightarrow x &\in [1, 5] \cap ([6, 10] - [2, 4]) \\ &= [1, 5] \cap [2, 8] = [2, 5]. \end{aligned}$$

$$\begin{aligned} y = z - x \Rightarrow y &\in [2, 4] \cap ([6, 10] - [1, 5]) \\ &= [2, 4] \cap ([1, 9] - [1, 5]) = [2, 4]. \end{aligned}$$

Propagation de contraintes

Soient les trois contraintes suivantes

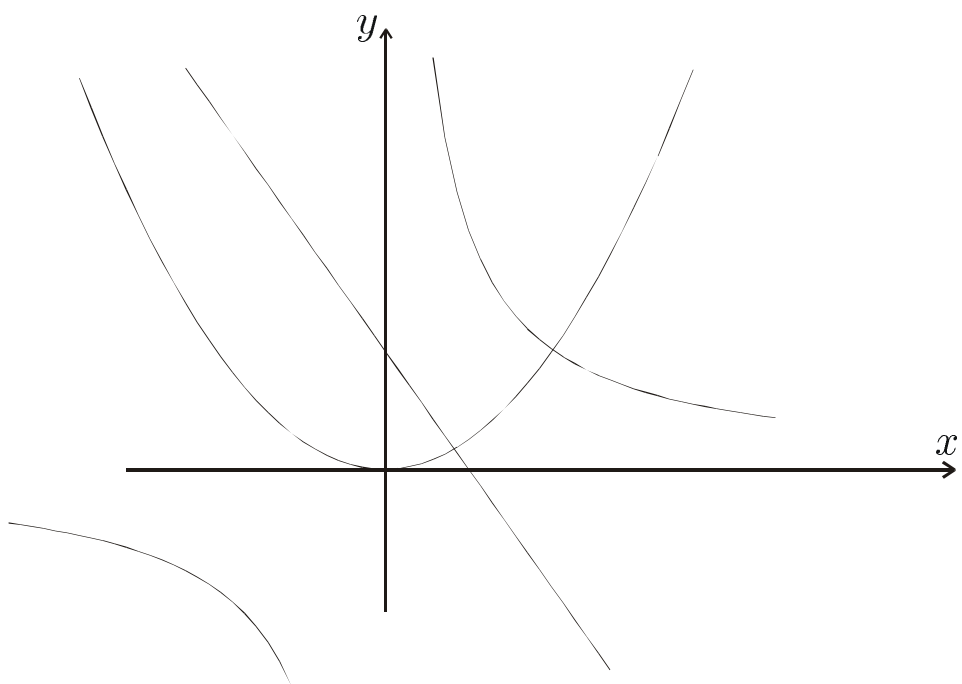
$$(C_1) : y = x^2,$$

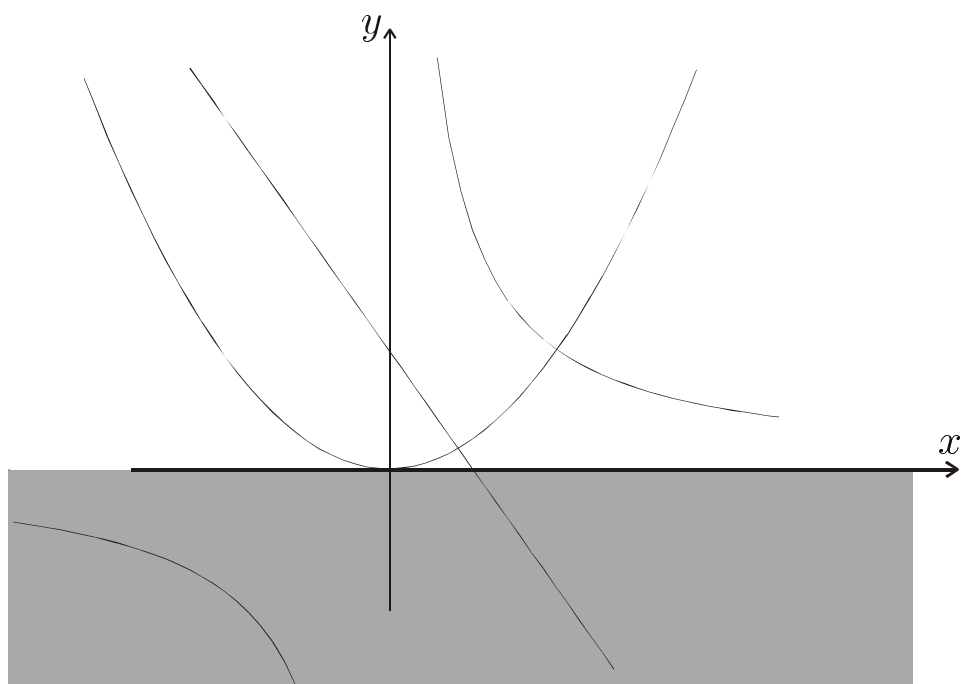
$$(C_2) : xy = 1,$$

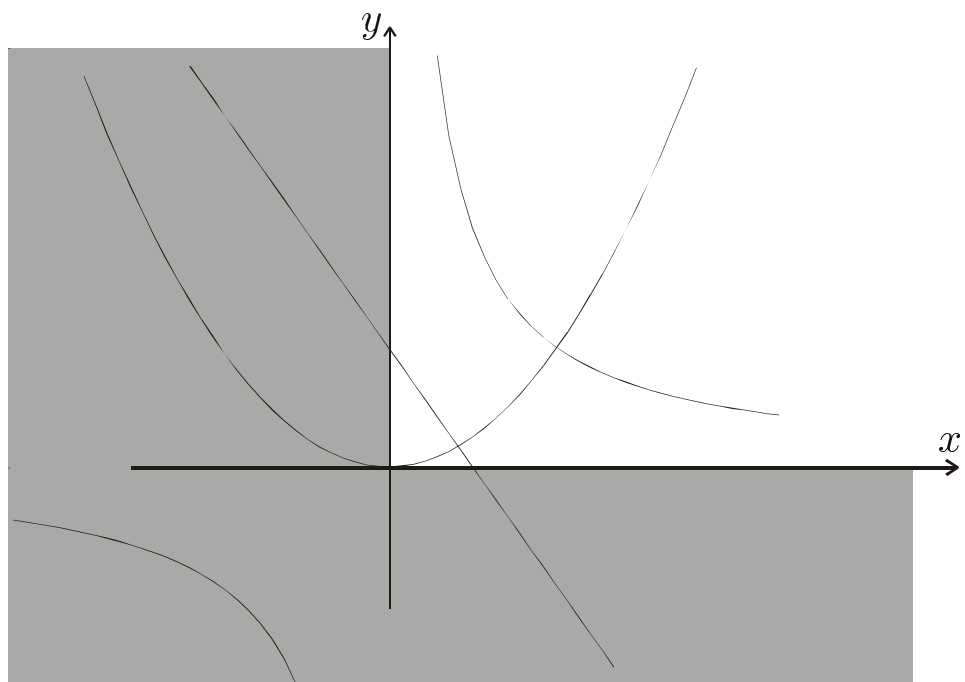
$$(C_3) : y = -2x + 1.$$

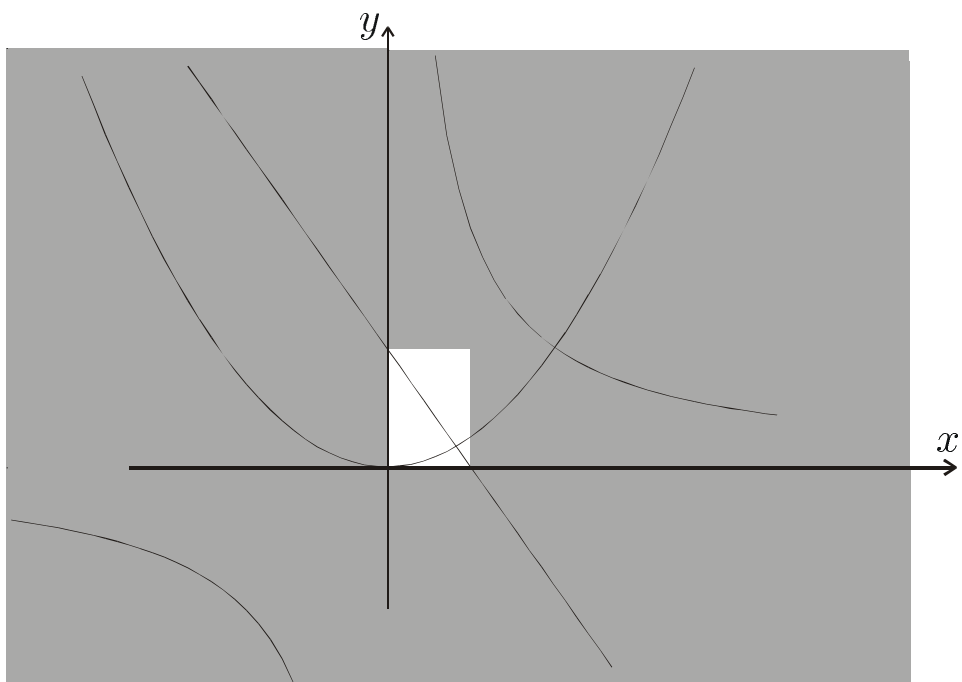
A chaque variable on affecte le domaine $] -\infty, \infty[$.

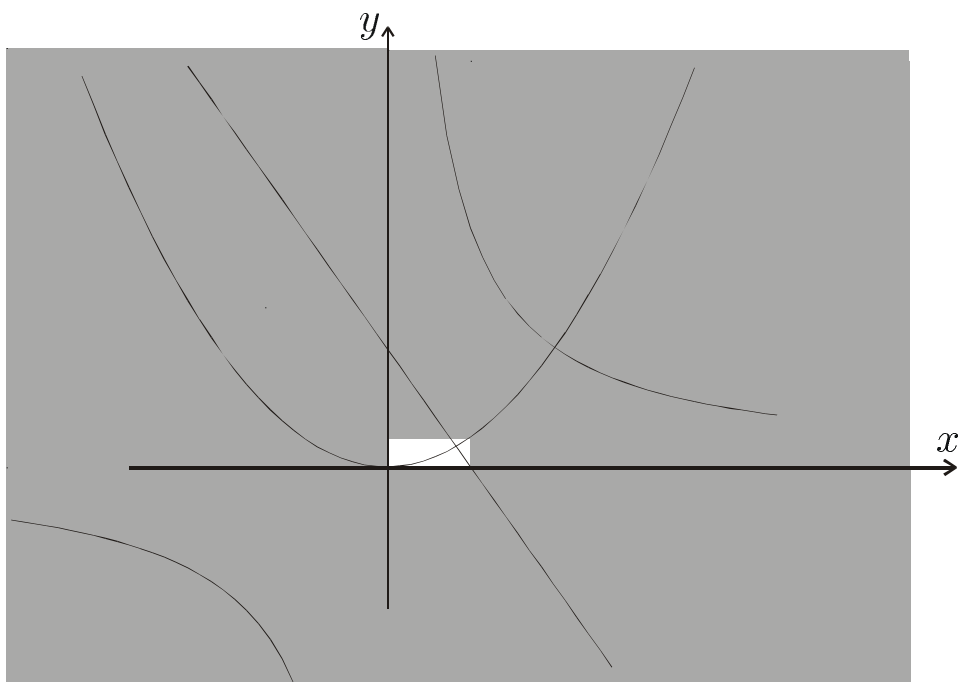
Une propagation de contrainte consiste à projeter les contraintes jusqu'à l'équilibre.











$$(C_1) \Rightarrow y \in]-\infty, \infty[^2 = [0, \infty[$$

$$(C_2) \Rightarrow x \in 1/[0, \infty[= [0, \infty[$$

$$(C_3) \Rightarrow y \in [0, \infty[\cap ((-2) \cdot [0, \infty[+ 1) \\ = [0, \infty[\cap (]-\infty, 1]) = [0, 1]$$

$$x \in [0, \infty[\cap (-[0, 1]/2 + 1/2) \\ = [0, 1/2]$$

$$(C_1) \Rightarrow y \in [0, 1] \cap [0, 1/2]^2 = [0, 1/4]$$

$$(C_2) \Rightarrow x \in [0, 1/2] \cap 1/[0, 1/4] = \emptyset$$

$$y \in [0, 1/4] \cap 1/\emptyset = \emptyset$$

Décomposition en contraintes primitives

Pour les contraintes plus complexes, il nous faut effectuer une décomposition. Par exemple,

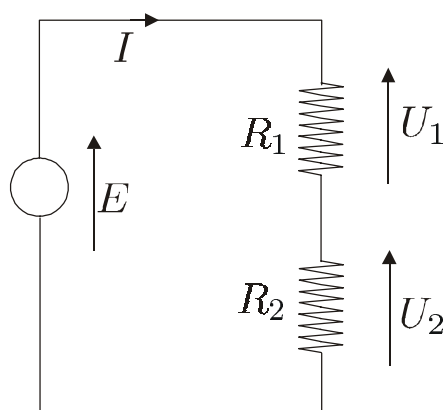
$$x + \sin(y) - xz \leq 0,$$

$$x \in [-1, 1], y \in [-1, 1], z \in [-1, 1]$$

se décompose en

$$\left\{ \begin{array}{ll} a = \sin(y) & x \in [-1, 1] \quad a \in] - \infty, \infty[\\ b = x + a & y \in [-1, 1] \quad b \in] - \infty, \infty[\\ c = xz & z \in [-1, 1] \quad c \in] - \infty, \infty[\\ b - c = d & d \in] - \infty, 0] \end{array} \right.,$$

Problème d'estimation



Contraintes

$$\begin{aligned}P &= EI; \quad E = (R_1 + R_2) I; \\U_1 &= R_1 I; \quad U_2 = R_2 I; \quad E = U_1 + U_2.\end{aligned}$$

Domaines

$$\begin{aligned}R_1 &\in [0, \infty[\Omega, \quad R_2 \in [0, \infty[\Omega, \\E &\in [23, 26]V, \quad I \in [4, 8]A, \quad U_1 \in [10, 11]V, \\U_2 &\in [14, 17]V, \quad P \in [124, 130]W,\end{aligned}$$

Domaines contractés

$$\begin{aligned}R_1 &\in [1.84, 2.31]\Omega, \quad R_2 \in [2.58, 3.35]\Omega, \\E &\in [24; 26]V, \quad I \in [4.769, 5.417]A, \quad U_1 \in [10; 11]V, \\U_2 &\in [14; 16]V, \quad P \in [124, 130]W.\end{aligned}$$

Programme Scilab disponible sur

<http://www.istia.univ-angers.fr/~jaulin/propagR1R2.sce>

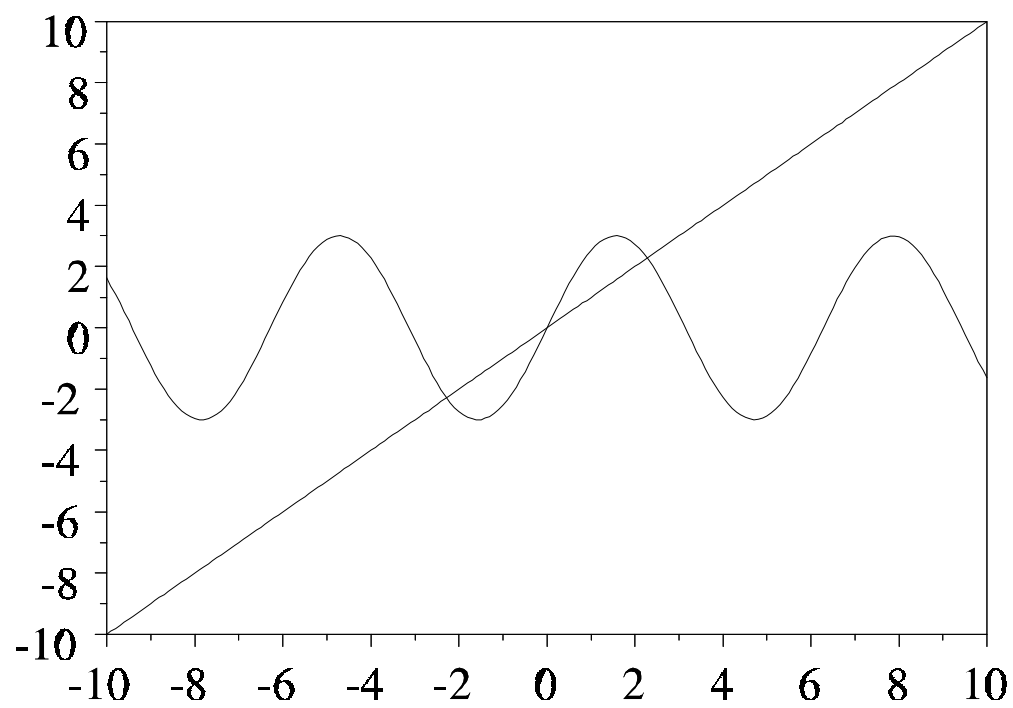
Résolution d'équations non-linéaires

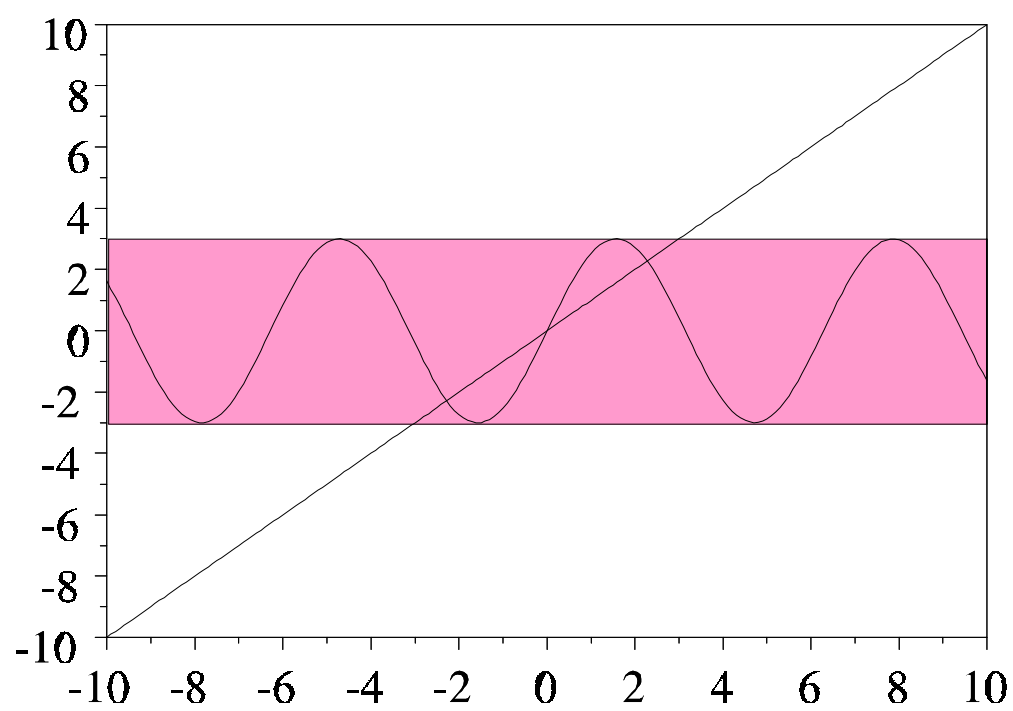
Cherchons par exemple à résoudre

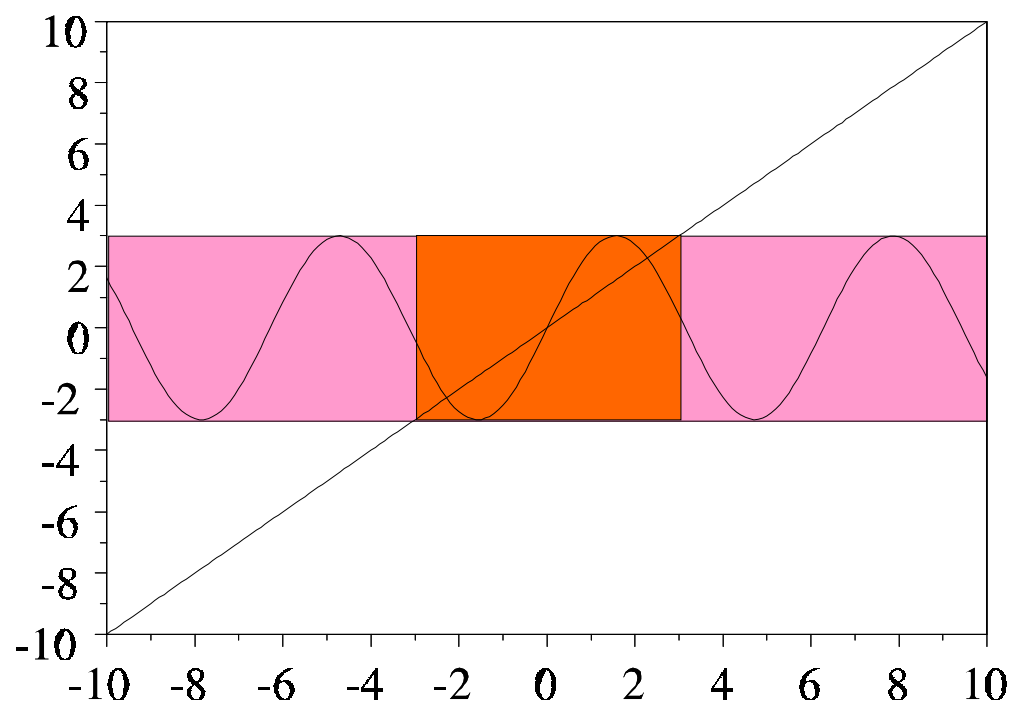
$$3 \sin(x) = x, \quad x \in \mathbb{R}.$$

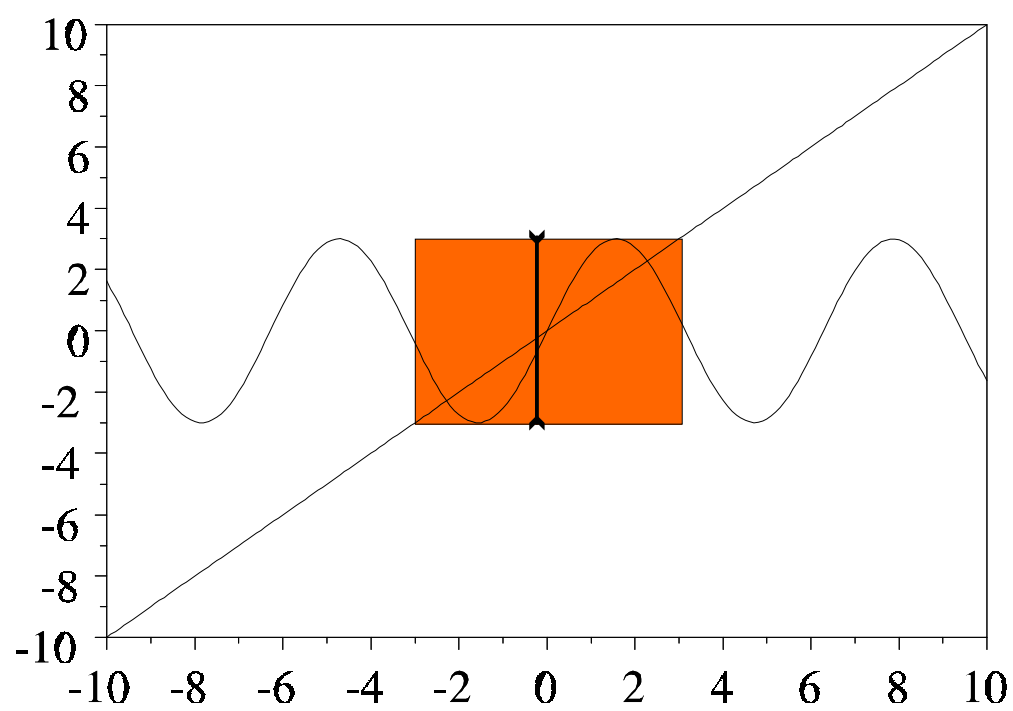
Cette équation se décompose en

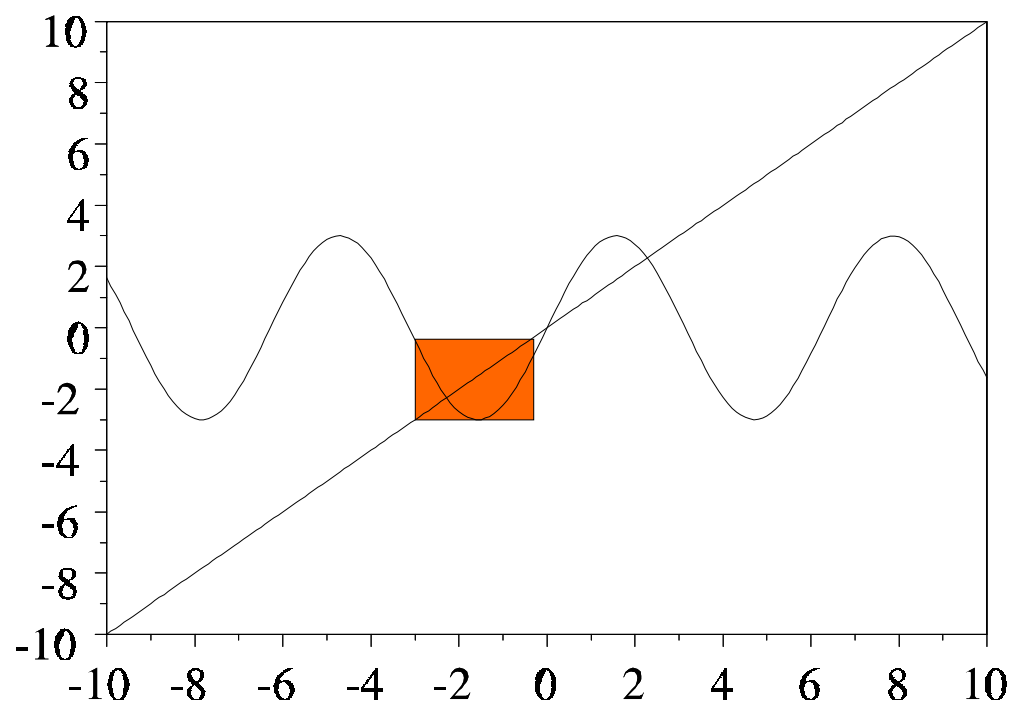
$$\begin{cases} y = 3 \sin(x) \\ y = x \end{cases} \quad x \in \mathbb{R}, y \in \mathbb{R}.$$

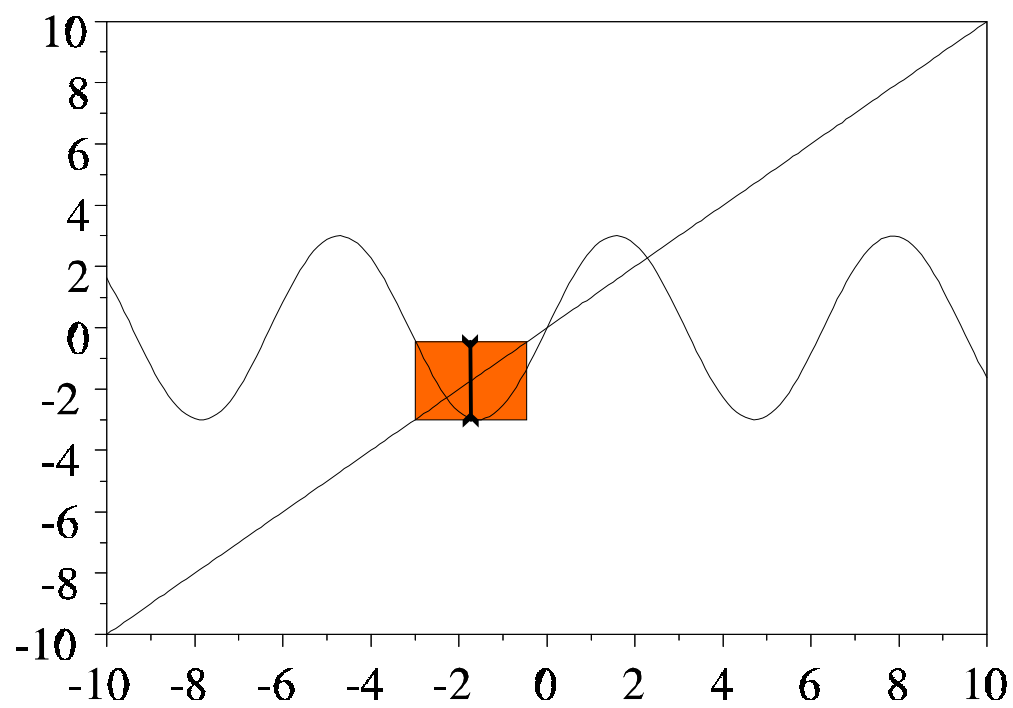


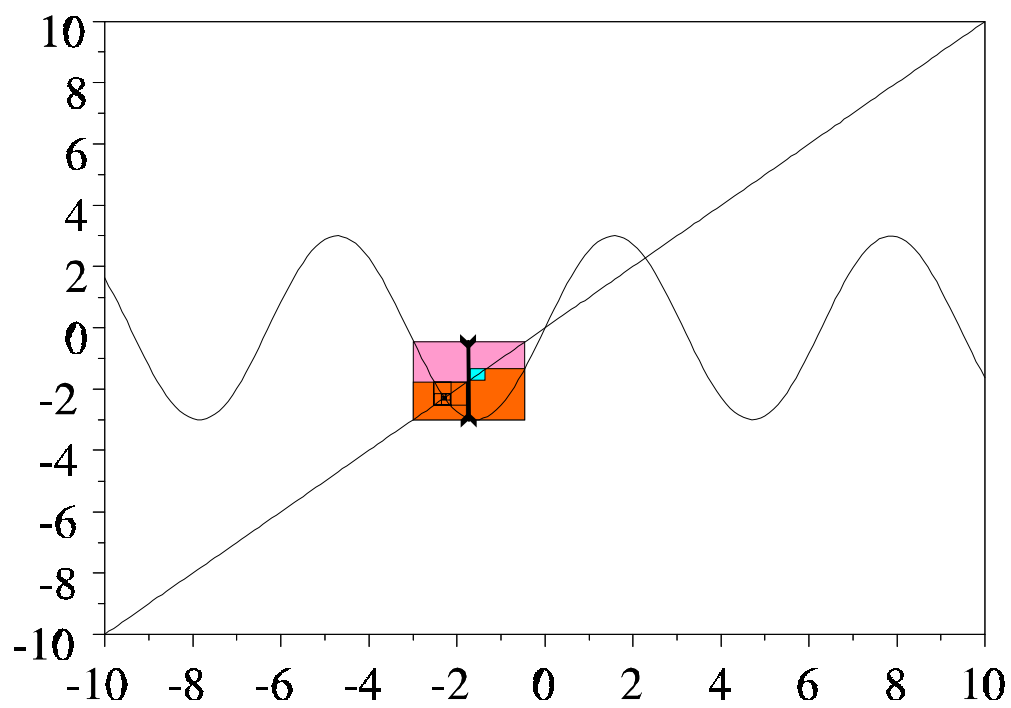












(Illustrer la résolution à l'aide du solveur Proj2D)

Prouver qu'un ensemble de contraintes
est toujours satisfait

Idée : On complémente l'ensemble des contraintes.

Par exemple, montrer que

$$\forall x \in [x], \forall y \in [y], f(x, y) \leq 0 \text{ et } g(x, y) \leq 0,$$

revient à montrer que

$$\{(x, y) \in [x] \times [y] \mid f(x, y) > 0 \text{ ou } g(x, y) > 0\} = \emptyset$$

c'est-à-dire que

$$\{(x, y) \in [x] \times [y] \mid \max(f(x, y), g(x, y)) > 0\} = \emptyset.$$

Stabilité robuste
(exemple tiré de Dorf et Bishop, Modern Control
Systems, p. 313)

Moto roulant à une vitesse constante de 1m/s.

Entrée du système : angle θ du guidon.

Sortie est l'angle de roulis ϕ de la moto.

Fonction de transfert du système :

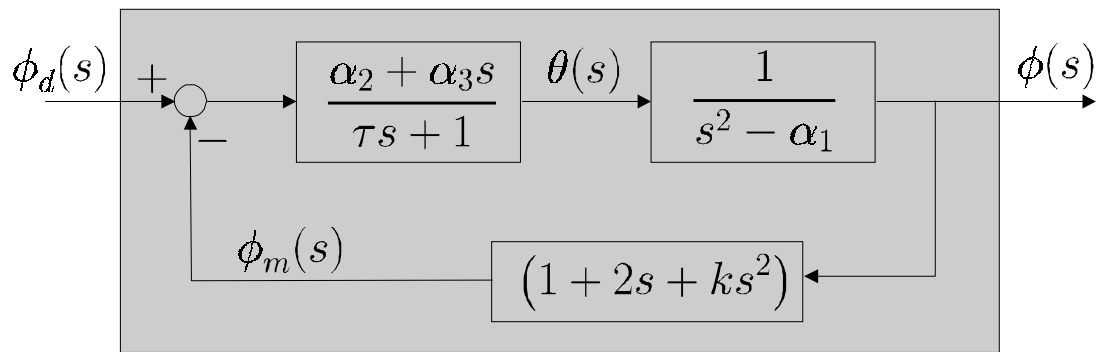
$$\phi(s) = \frac{1}{s^2 - \alpha_1} \theta(s).$$

Capteur :

$$\phi_m(s) = (1 + 2s + ks^2) \phi(s).$$

Régulateur :

$$\theta(s) = \frac{\alpha_2 + \alpha_3 s}{\tau s + 1} (\phi_d(s) - \phi_m(s)).$$



Domaines pour les paramètres

$$[\alpha_1] = [8.8; 9.2], \quad [\alpha_2] = [2.8; 3.2], \quad [\alpha_3] = [0.8; 1.2],$$

$$[\tau] = [1.8; 2.2], \quad [k] = [-3.2; -2.8].$$

Polynôme caractéristique :

$$a_3 s^3 + a_2 s^2 + a_1 s + a_0$$

avec

$$a_0 = -\alpha_1 + \alpha_2,$$

$$a_1 = \alpha_3 - \alpha_1 \tau + 2\alpha_2,$$

$$a_2 = \alpha_2 k + 2\alpha_3 + 1,$$

$$a_3 = \tau + \alpha_3 k.$$

Table de Routh :

a_3	a_1
a_2	a_0
$\frac{a_2 a_1 - a_3 a_0}{a_2}$	0
a_0	0

Stabilité si

$$\text{sign}(a_3) = \text{sign}(a_2) = \text{sign}\left(\frac{a_2 a_1 - a_3 a_0}{a_2}\right) = \text{sign}(a_0).$$

Stabilité robuste si

$$\forall \alpha_1 \in [\alpha_1], \forall \alpha_2 \in [\alpha_2], \forall \alpha_3 \in [\alpha_3], \forall \tau \in [\tau], \forall k \in [k],$$

$$\text{sign}(a_3) = \text{sign}(a_2) = \text{sign}\left(\frac{a_2 a_1 - a_3 a_0}{a_2}\right) = \text{sign}(a_0).$$

c'est-à-dire, si

$$\begin{aligned} \mathbb{S} = \{ (\alpha_1, \alpha_2, \alpha_3, \tau, k) \in [\alpha_1] \times [\alpha_2] \times [\alpha_3] \times [\tau] \times [k] \mid \\ a_0 = -\alpha_1 + \alpha_2, a_1 = \alpha_3 - \alpha_1\tau + 2\alpha_2 \\ a_2 = \alpha_2k + 2\alpha_3 + 1, a_3 = \tau + \alpha_3k \\ b = \frac{a_2a_1 - a_3a_0}{a_2} \\ \max(\min(a_3, a_2, b, a_0), -\max(a_3, a_2, b, a_0)) \leq 0 \} \end{aligned}$$

est vide.

La preuve est obtenue en moins de 10^{-3} seconde sur un Pentium 3.

Localisation d'un satellite

$$\left\{ \begin{array}{l} x_1(k+1) = x_1(k) + 0.5 x_3(k) + w_1(k) \\ x_2(k+1) = x_2(k) + 0.5 x_4(k) + w_2(k) \\ x_3(k+1) = x_3(k) - 0.5 \frac{x_1(k)}{\sqrt{x_1^2(k)+x_2^2(k)}} + w_3(k) \\ x_4(k+1) = x_4(k) - 0.5 \frac{x_2(k)}{\sqrt{x_1^2(k)+x_2^2(k)}} + w_4(k) \\ y_1(k) = x_2(k)/x_1(k) \\ y_2(k) = x_1(k)x_3(k) + x_2(k)x_4(k) \\ y_3(k) = x_1^2(k) + x_2^2(k) \end{array} \right.$$

Fichier d'entrée du solveur Realpaver

```
x1_0 in [-oo, oo],  
x2_0 in [-oo, oo],  
x3_0 in [-oo, oo],  
x4_0 in [-oo, oo],  
w1_0 in [-0.01, 0.01],  
w2_0 in [-0.01, 0.01],  
w3_0 in [-0.01, 0.01],  
w4_0 in [-0.01, 0.01],  
y1_0 in [0.107, 0.307],  
y2_0 in [-0.710, -0.510],  
y3_0 in [16.117, 16.317],  
...
```

Constraints

```
x1_1=x1_0+0.5*x3_0+ w1_0,  
x2_1=x2_0+0.5*x4_0+ w2_0,  
x3_1=x3_0-0.5*x1_0/  
      (sqrt(x1_0^2+x2_0^2))^3+w3_0,  
...
```

Fichier en sortie du solveur

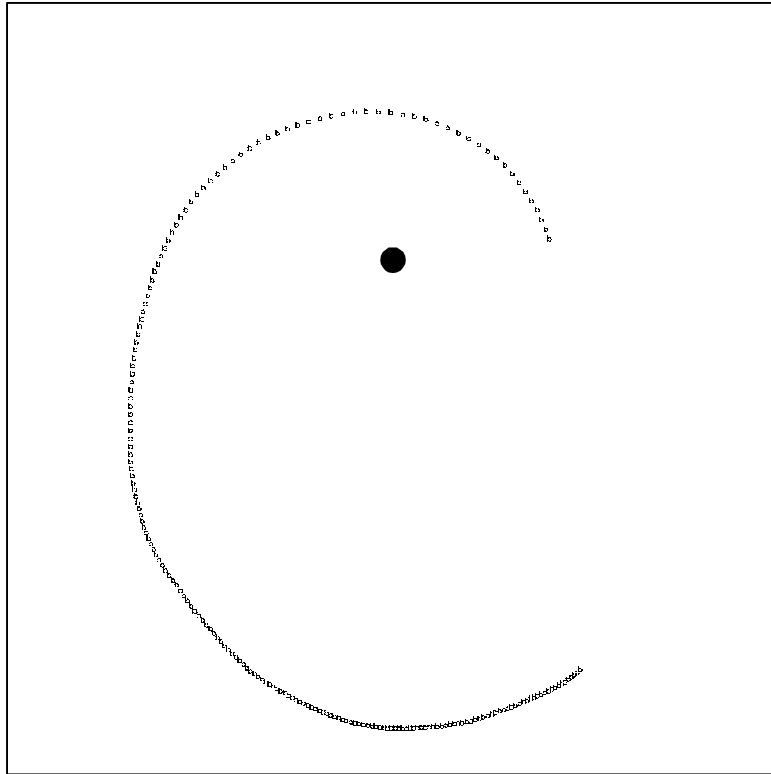
x1_0 in [3.89 , 4.01]
x2_0 in [0.41 , 0.95]
x3_0 in [-0.41 , -0.14]
x4_0 in [0.20 , 0.93]
y1_0 in [0.10 , 0.25]
y2_0 in [-0.71 , -0.51]
y3_0 in [16.11 , 16.31]

...

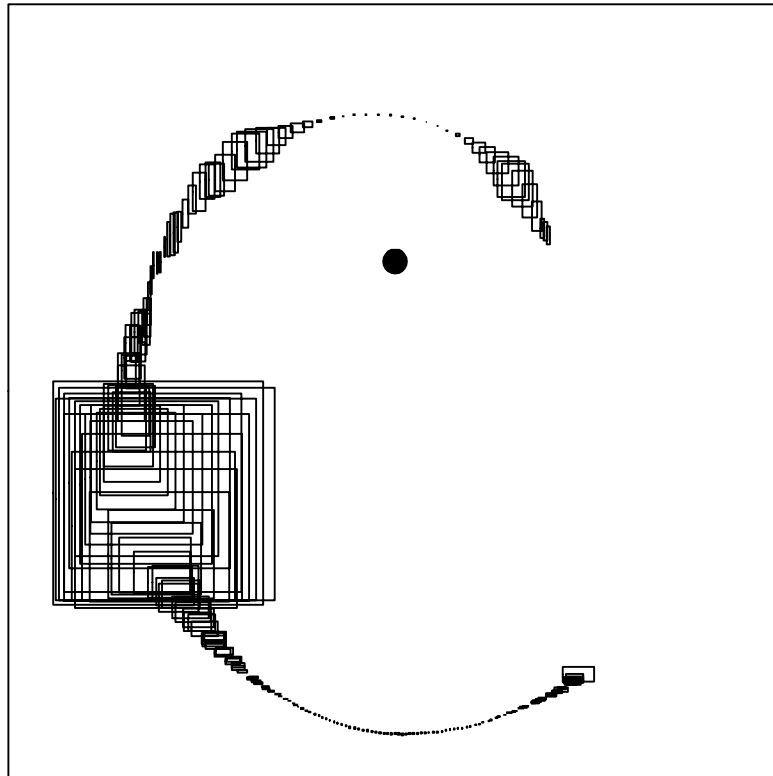
x1_200 in [4.38 , 4.88]
x2_200 in [-10.84 , -10.61]
x3_200 in [-0.14 , 0.54]
x4_200 in [-0.00 , 0.32]
y1_200 in [-2.42 , -2.22]
y2_200 in [-0.92 , -0.72]
y3_200 in [136.61 , 136.81]

...

Elapsed time: 810 ms



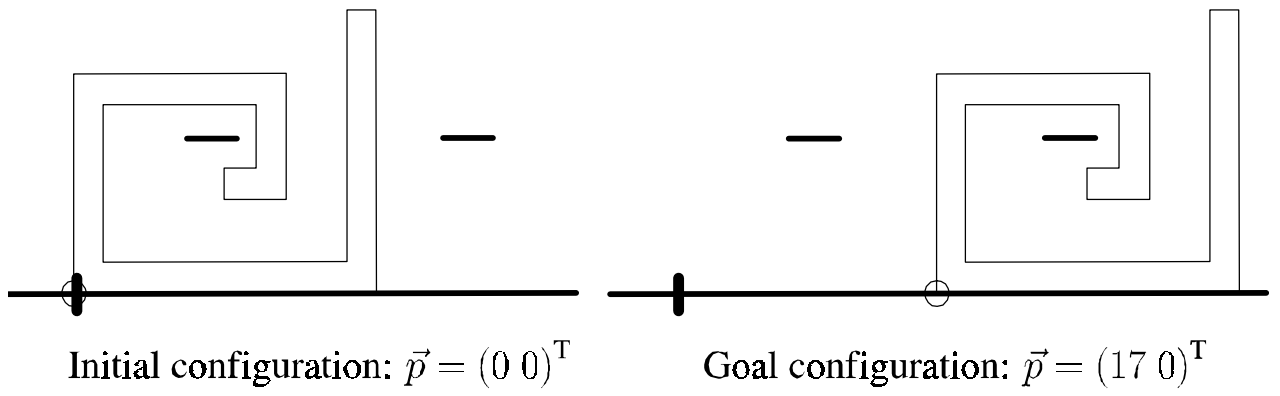
Positions du satellite générées par simulation
pour k allant de 0 à 200

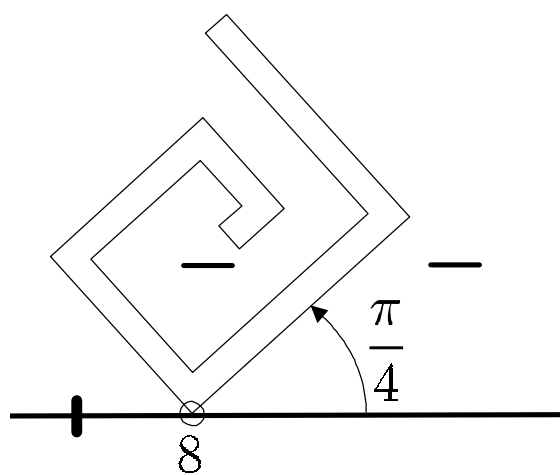


Pavés générés par Realpaver

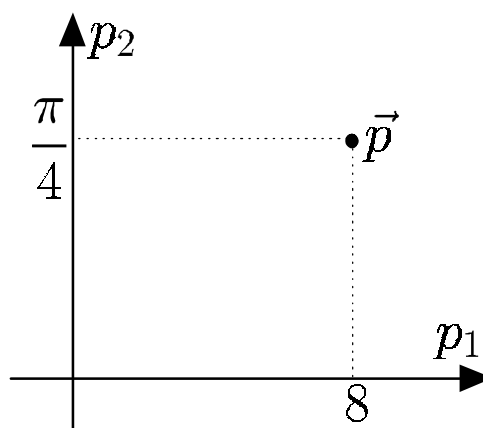
Ce problème est un CSP (constraint satisfaction problem) contenant 8815 variables et 5607 contraintes.

Planification de chemin

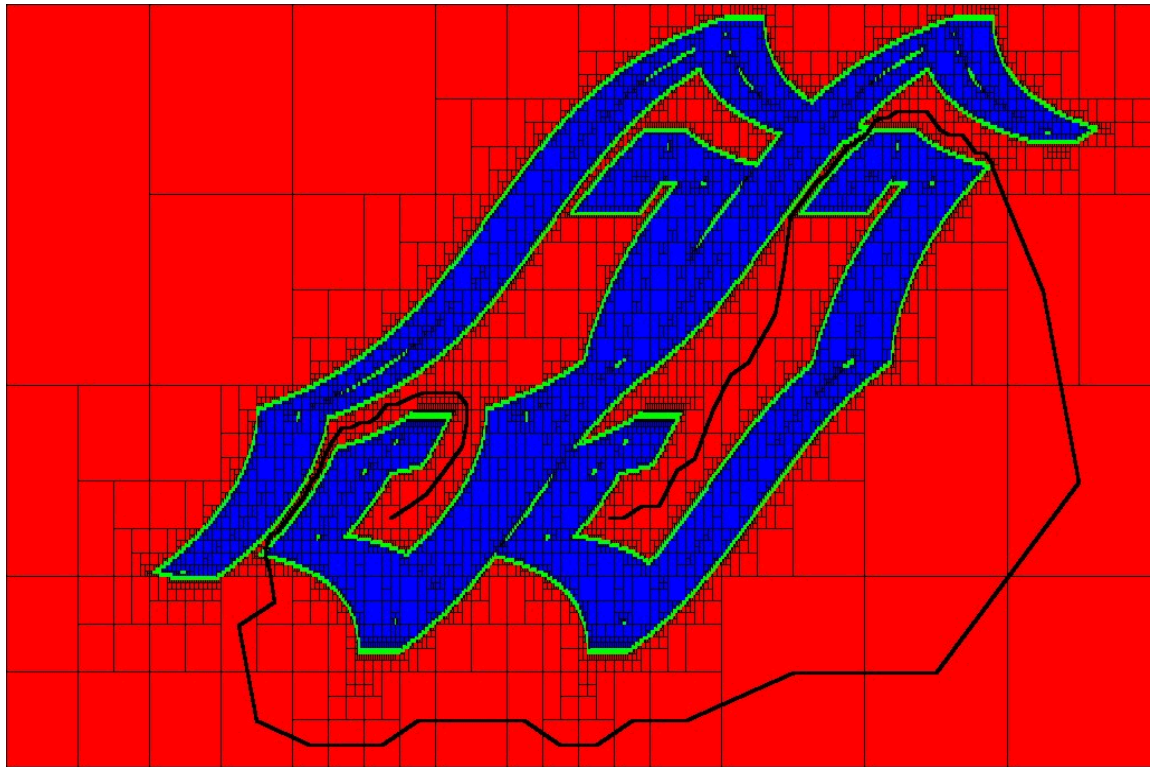


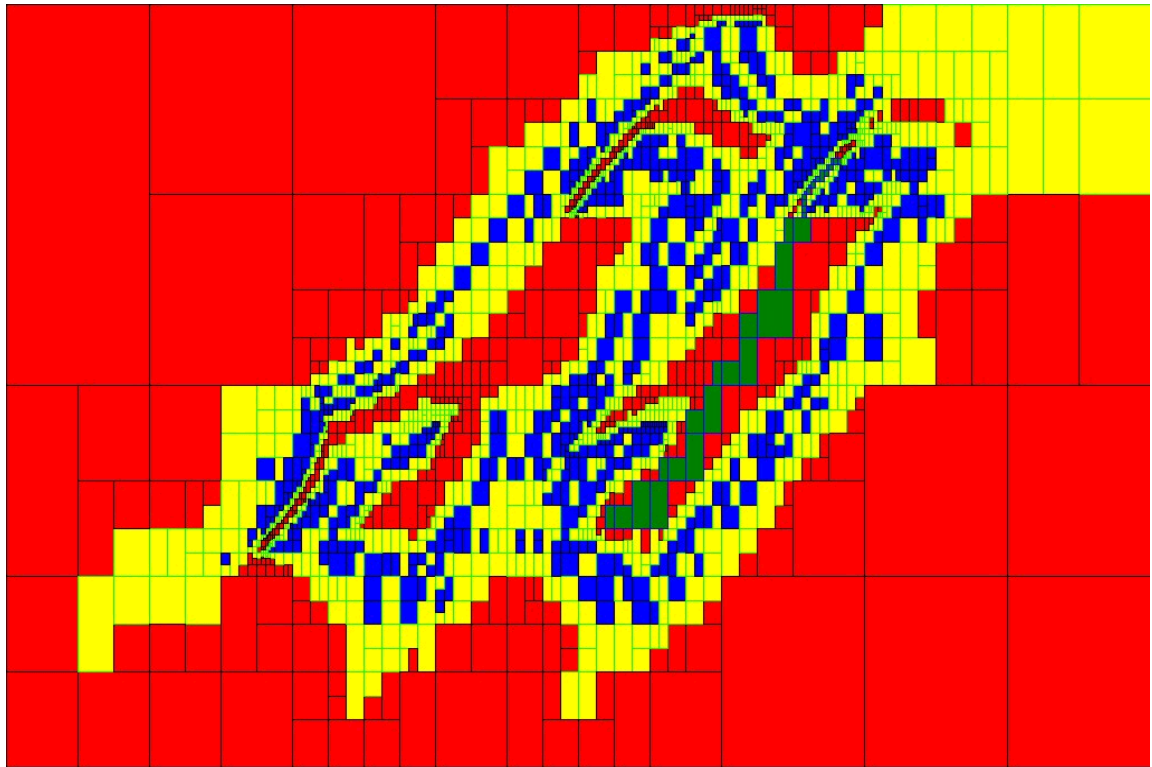


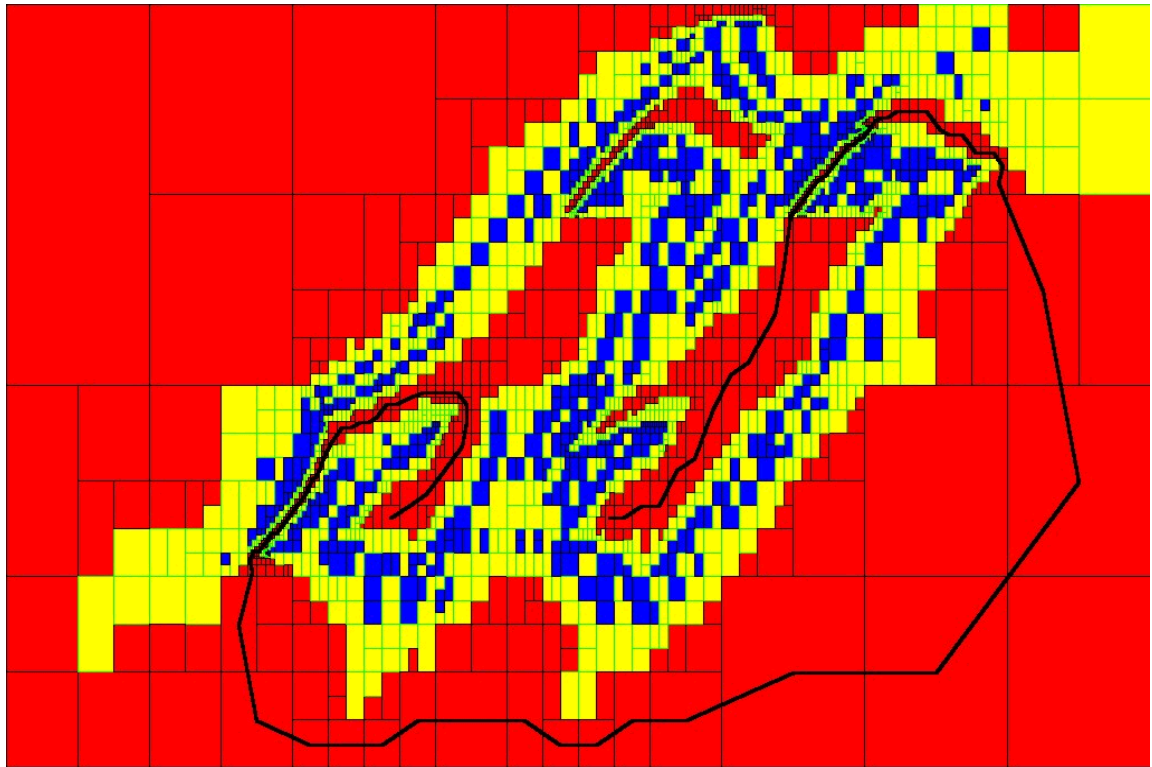
Room



Configuration space







Objectifs

- 1) Développer des solveurs ensemblistes pour les automaticiens et les roboticiens,
- 2) Augmenter l'efficacité des solveurs en faisant collaborer les méthodes intervalles, ellipsoïdales, LMIs, ...
- 3) Résoudre des problèmes reconnus comme difficiles.