Slides related to the summer school of applied interval analysis in Grenoble, Septembre 2005

Luc Jaulin, Ensieta, E3I2.

Basic notions on set theory

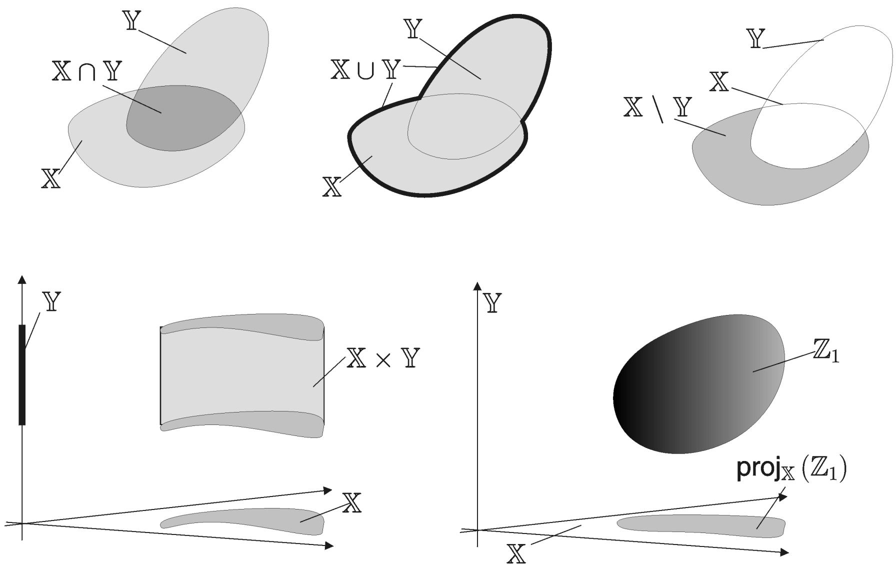
(Luc Jaulin, Monday, 10h00-10h30)

Basic operation on sets

$$
\begin{aligned}
\mathbb{X} \cap \mathbb{Y} &\stackrel{\text{def}}{=} \{x \mid x \in \mathbb{X} \text{ and } x \in \mathbb{Y}\} \\
\mathbb{X} \cup \mathbb{Y} &\stackrel{\text{def}}{=} \{x \mid x \in \mathbb{X} \text{ or } x \in \mathbb{Y}\} \\
\mathbb{X} \setminus \mathbb{Y} &\stackrel{\text{def}}{=} \{x \mid x \in \mathbb{X} \text{ and } x \notin \mathbb{Y}\} \\
\mathbb{X} \times \mathbb{Y} &\stackrel{\text{def}}{=} \{(x, y) \mid x \in \mathbb{X} \text{ and } y \in \mathbb{Y}\}
\end{aligned}
$$

If $\mathbb{Z} = \mathbb{X} \times \mathbb{Y}$, then the *projection* of a subset $\mathbb{Z}_1$ of $\mathbb{Z}$ onto $\mathbb{X}$ (with respect to $\mathbb{Y}$) is defined as

$$\text{proj}_{\mathbb{X}}(\mathbb{Z}_1) \stackrel{\text{def}}{=} \{x \in \mathbb{X} \mid \exists y \in \mathbb{Y} \text{ such that } (x, y) \in \mathbb{Z}_1\}.$$



$\mathbb{Y}$
$\mathbb{X} \cap \mathbb{Y}$
$\mathbb{X}$
$\mathbb{Y}$
$\mathbb{X} \cup \mathbb{Y}$
$\mathbb{X}$
$\mathbb{Y}$
$\mathbb{X}$
$\mathbb{X} \setminus \mathbb{Y}$
$\mathbb{Y}$
$\mathbb{X} \times \mathbb{Y}$
$\mathbb{X}$
$\mathbb{Y}$
$\mathbb{Z}_1$
$\mathrm{proj}_{\mathbb{X}}(\mathbb{Z}_1)$
$\mathbb{X}$

Example 1: If $\mathbb{X} = \{a, b, c, d\}$ and $\mathbb{Y} = \{b, c, x, y\}$, then

$$\begin{aligned}
\mathbb{X} \cap \mathbb{Y} &= \{b, c\} \\
\mathbb{X} \cup \mathbb{Y} &= \{a, b, c, d, x, y\} \\
\mathbb{X} \setminus \mathbb{Y} &= \{a, d\} \\
\mathbb{X} \times \mathbb{Y} &= \{(a, b), (a, c), (a, x), (a, y), \\
& \quad \ldots, (d, b), (d, c), (d, x), (d, y)\}
\end{aligned}$$

If $\mathbb{Z}_1 \overset{\text{def}}{=} \{(a, c), (a, y), (b, c), (d, y)\} \subset \mathbb{X} \times \mathbb{Y}$, we have

$$\begin{aligned}
\text{proj}_{\mathbb{X}}(\mathbb{Z}_1) &= \{a, b, d\}, \\
\text{proj}_{\mathbb{Y}}(\mathbb{Z}_1) &= \{c, y\}.
\end{aligned}$$

Example 2: If

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

then

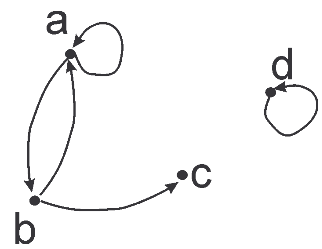
$$\begin{aligned} \text{proj}_{\mathbb{X}}(\mathbb{S}) &= [6, 9] \\ \text{proj}_{\mathbb{Y}}(\mathbb{S}) &= [2, 4] \\ \text{proj}_{\mathbb{Z}}(\mathbb{S}) &= [2, 5]. \end{aligned}$$

Relation (or binary constraint)

A relation in $\mathbb{X}$ is a subset of $\mathbb{X} \times \mathbb{X}$.

Example 1: Consider the relation in $\mathbb{X} = \{a, b, c, d\}$ given by

$$\mathcal{C} = \{(a, a), (a, b), (b, a), (b, c), (d, d)\}$$



$$\left(\begin{array}{ccccc} \nearrow & a & b & c & d \\ a & 1 & 1 & 0 & 0 \\ b & 1 & 0 & 1 & 0 \\ c & 0 & 0 & 0 & 0 \\ d & 0 & 0 & 0 & 1 \end{array}\right)$$

Example 2: The set

$$\mathcal{C} = \left\{(x, y) \in \mathbb{R}^2 | y = \mathsf{sin}(x)\right\}$$

is a relation in $\mathbb{R}$.

This relation can be written as "$y = \mathsf{sin}\, x$", or "$\mathsf{sin}(y, x)$" or "sin".

Example 3: The set

$$\mathcal{C} = \left\{(x, y) \in \mathbb{R}^2 | y \leq x\right\}$$

is a relation in $\mathbb{R}$.

This relation can be written as "$y \leq x$", or "$\leq (y, x)$" or "$\leq$".

Example 4: The set

$$\mathcal{C} = \left\{(x, y) \in \mathbb{R}^2 | \sin(x + y) = 0\right\}$$

is a relation in $\mathbb{R}$.

This relation can be written as "$\sin(x + y) = 0$".

Constraints

A constraint in $\mathbb{X}$ is a subset of $\mathbb{X} \times \mathbb{X} \times \cdots \times \mathbb{X}$.

Example 1: If $\mathbb{X} = \{a, b, c, d\}$, the set

$$\mathcal{C} = \{(a, a, a), (a, b, c), (c, c, a)\}$$

is a ternary constraint in $\mathbb{X}$.

Example 2: The set

$$\mathcal{C} = \left\{(x, y, z) \in \mathbb{R}^3 | z = x + y\right\}$$

is a ternary constraint in $\mathbb{R}$.

It can be written as "$z = x + y$", or "$+(z, y, x)$" or "$+$".

Consider a function $f : \mathbb{X} \longrightarrow \mathbb{Y}$. If $\mathbb{X}_1 \subset \mathbb{X}$, the *direct image* of $\mathbb{X}_1$ by f is

$$f\left(\mathbb{X}_1\right) \stackrel{\text{def}}{=} \{f(x) \mid x \in \mathbb{X}_1\}.$$

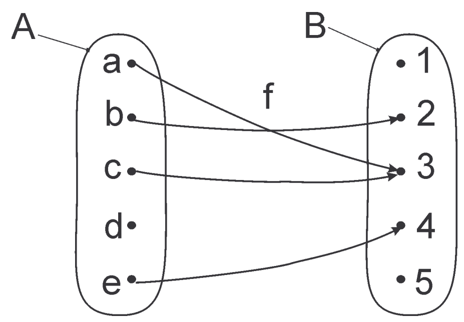
If $\mathbb{Y}_1 \subset \mathbb{Y}$, the *reciprocal image* of $\mathbb{Y}_1$ by f is

$$f^{-1}\left(\mathbb{Y}_1\right) \stackrel{\text{def}}{=} \{x \in \mathbb{X} \mid f(x) \in \mathbb{Y}_1\}.$$

If $\mathbb{X}_1$ and $\mathbb{X}_2$ are subsets of $\mathbb{X}$ and if $\mathbb{Y}_1$ and $\mathbb{Y}_2$ are subsets of $\mathbb{Y}$, then

$$\begin{gathered}
f\left(\mathbb{X}_1 \cap \mathbb{X}_2\right) \subset f\left(\mathbb{X}_1\right) \cap f\left(\mathbb{X}_2\right), \\
f\left(\mathbb{X}_1 \cup \mathbb{X}_2\right) = f\left(\mathbb{X}_1\right) \cup f\left(\mathbb{X}_2\right), \\
f^{-1}\left(\mathbb{Y}_1 \cap \mathbb{Y}_2\right) = f^{-1}\left(\mathbb{Y}_1\right) \cap f^{-1}\left(\mathbb{Y}_2\right), \\
f^{-1}\left(\mathbb{Y}_1 \cup \mathbb{Y}_2\right) = f^{-1}\left(\mathbb{Y}_1\right) \cup f^{-1}\left(\mathbb{Y}_2\right), \\
f\left(f^{-1}\left(\mathbb{Y}\right)\right) \subset \mathbb{Y}, \\
\mathbb{X}_1 \ \subset f^{-1}\left(\mathbb{Y}\right) \Rightarrow f^{-1}\left(f\left(\mathbb{X}_1\right)\right) \supset \mathbb{X}_1 \\
\mathbb{X}_1 \subset \mathbb{X}_2 \Rightarrow f\left(\mathbb{X}_1\right) \subset f\left(\mathbb{X}_2\right), \\
\mathbb{Y}_1 \subset \mathbb{Y}_2 \Rightarrow f^{-1}\left(\mathbb{Y}_1\right) \subset f^{-1}\left(\mathbb{Y}_2\right),
\end{gathered}$$

Example 1: If f is defined as follows



then

$$
\begin{aligned}
f(A) &= \{2,3,4\} = \mathrm{Im}(f). \\
f^{-1}(B) &= \{a,b,c,e\} = \mathrm{dom}(f). \\
f^{-1}(f(A)) &= \{a,b,c,e\} \subset A \\
f^{-1}(f(\{b,c\})) &= \{a,b,c\}.
\end{aligned}
$$

Example 2: If $f(x) = x^2$, then

$$\begin{aligned} f([2,3]) &= [4,9] \\ f^{-1}([4,9]) &= [-3,-2] \cup [2,3]. \end{aligned}$$

This is consistent with the property

$$f\left(f^{-1}(\mathbb{Y})\right) \subset \mathbb{Y}.$$

Example 3: If $f(x) = \log$, then

$$f^{-1}\left(f\left([-3, -2]\right)\right) = \emptyset.$$

Correct the error in the book page 13, line 6 of (2.10).

Interval computation

(Luc Jaulin , Monday, 11h45-12h15).

Intervals

A (closed) *interval* is a connected, closed subset of $\mathbb{R}$.

For example $[1,3]$, $\{1\}$, $]-\infty,6]$, $\mathbb{R}$ and $\emptyset$ are considered as intervals whereas $]1,3[$, $[3,2]$ and $[1,2]\cup[3,4]$ are not.

The *lower bound* of $[x]$ is defined by

$$\underline{x} = \mathsf{lb}([x]) = \inf\{x|x \in [x]\}\,.$$

The *upper bound* of $[x]$ is defined by

$$\bar{x} = \text{ub}([x]) = \sup \{x | x \in [x]\} .$$

By convention, $\text{ub}(\emptyset) = -\infty$ and $\text{lb}(\emptyset) = +\infty$.

The *width* of $[x]$ is

$$w\left([x]\right) = \bar{x} - \underline{x}.$$

The *midpoint* of $[x]$ is

$$\text{mid}\left([x]\right) = \frac{\bar{x} + \underline{x}}{2}.$$

The *enveloping interval* associated $\mathbb{X} \subset \mathbb{R}$ is the smallest interval $[\mathbb{X}]$ containing $\mathbb{X}$. For instance

$$[\, [1,3] \cup [6,7[\,] = [1,7].$$

The *interval union* of $[x]$ and $[y]$ is defined by

$$[x] \sqcup [y] = [[x] \cup [y]] \, .$$

Binary operators

If $\diamond \in \{+, -, *, /, \max, \min\}$, where $*$ is the multiplication, and if $[x]$ and $[y]$ are two intervals, we define

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

Therefore,

$$\begin{array}{rl} [\underline{x}, \bar{x}] + [\underline{y}, \bar{y}] & = [\underline{x} + \underline{y}, \bar{x} + \bar{y}] \\ [\underline{x}, \bar{x}] \cdot [\underline{y}, \bar{y}] & = [\min(\underline{x}\underline{y}, \bar{x}\underline{y}, \underline{x}\bar{y}, \bar{x}\bar{y}), \\ & \quad \max(\underline{x}\underline{y}, \bar{x}\underline{y}, \underline{x}\bar{y}, \bar{x}\bar{y})] \\ \max\left([\underline{x}, \bar{x}], [\underline{y}, \bar{y}]\right) & = [\max(\underline{x}, \underline{y}), \max(\bar{x}, \bar{y})]. \end{array}$$

For instance,

$$\begin{aligned} [-1,3]+[2,5] &= [1,8], \\ [-1,3].[2,5] &= [-5,15], \\ [-1,3]/[2,5] &= [-\tfrac{1}{2},\tfrac{3}{2}], \\ \max([-1,3],[2,5]) &= [2,5]. \end{aligned}$$

We have

$$\begin{aligned} ([1,2]+[-3,4])*[-1,5] &= [-2,6]*[-1,5] \\ &= [-10,30]. \end{aligned}$$

Subdistributivity

$$[x] * ([y] + [z]) \subset [x] * [y] + [x] * [z]$$

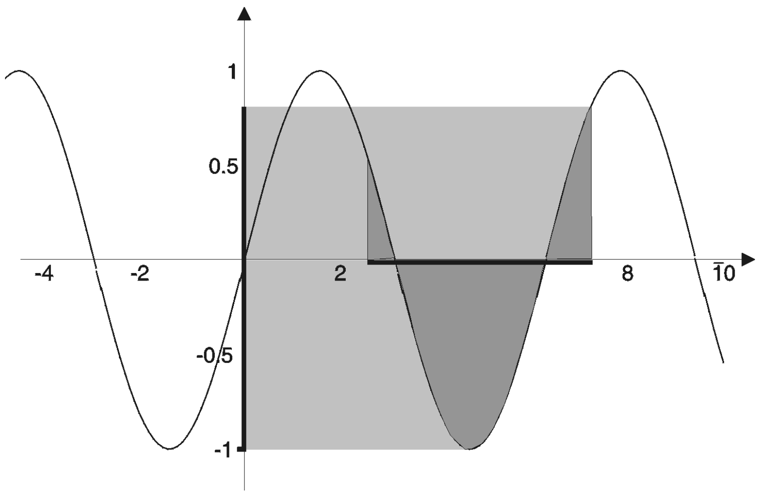
Example:

$$[0, 1]*([-1, -1]+[1, 1]) \subset [0, 1]*[-1, -1]+[0, 1]*[1, 1])$$

Elementary functions

If $f \in \{\cos, \sin, \text{sqr}, \text{sqrt}, \log, \exp, \ldots\}$, its interval extension is

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



For instance

$$
\begin{aligned}
\mathsf{sin}\,([0,\pi]) &= [0,1],\\
\mathsf{sqr}\,([-1,3]) &= [-1,3]^2 = [0,9],\\
\mathsf{abs}\,([-7,1]) &= [0,7],\\
\mathsf{sqrt}\,([-10,4]) &= \sqrt{[-10,4]} = [0,2],\\
\mathsf{log}\,([-2,-1]) &= \emptyset.
\end{aligned}
$$

Interpretation

If f is an expression (such as $xy + x.\,\mathsf{sin}\,x$) then

$$f([x],[y]) = [z] \Rightarrow \forall x \in [x], \forall y \in [y], \exists z \in [z], z = f(x,y)$$

Modal intervals : handle proper intervals (such as $[1, 2]$) and improper intervals (such as $[2, 1]$).

For instance,

$$[1, 4] + [2, 1] = [3, 5]$$

should be interpreted as

$$\forall x \in [1, 4], \exists y [1, 2], \exists z \in [3, 5], z = x + y$$

and

$$[4, 1] + [1, 2] = [5, 3]$$

should be interpreted as

$$\forall y \in [1, 2], \forall z \in [3, 5], \exists x \in [1, 4], z = x + y.$$

Modal interval analysis can be useful to prove propositions such as

$$\begin{aligned} \forall x_1 &\in [x_1], \forall x_2 \in [x_2], \\ \exists y_1 &\in [y_1], \exists y_2 \in [y_2], \exists z \in [z], \\ z &= \sin(x_1 x_2) + x_2 y_1 - y_2 x_2^2. \end{aligned}$$

Boxes

A *box* is the Cartesian product of n intervals

$$[\mathbf{x}] = [\underline{x}_1, \bar{x}_1] \times \cdots \times [\underline{x}_n, \bar{x}_n] = [x_1] \times \cdots \times [x_n].$$

The set of all boxes of $\mathbb{R}^n$ will be denoted by $\mathbb{IR}^n$.

The *width* $w([\mathbf{x}])$ of a box $[\mathbf{x}]$ is the length of its largest side

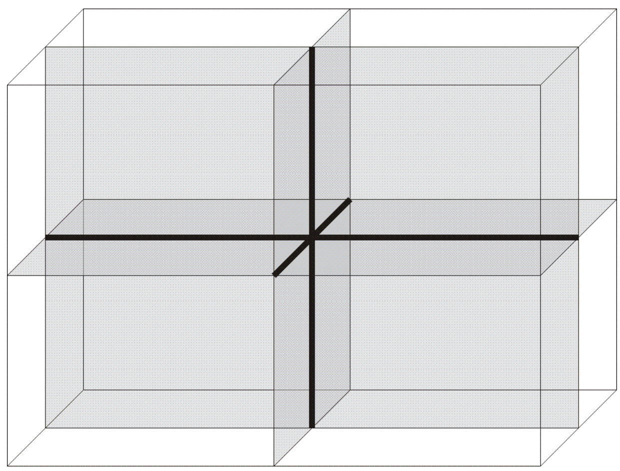
$$w([\mathbf{x}]) = \max_{i \in \{1,\ldots,n\}} w([x_i]).$$

For instance

$$w([1, 2] \times [-1, 3]) = 4.$$

If $w([\mathbf{x}]) = 0$, $[\mathbf{x}]$ is said to be *degenerated*.

The *principal plane* of $[\mathbf{x}]$ is the symmetric plane $[\mathbf{x}]$ perpendicular to its largest side.



To *bisect* a box $[\mathbf{x}]$ means to split it in two parts.

The bisection of $[\mathbf{x}] = [1,2] \times [-1,3]$ generates the boxes:

$$\begin{aligned} \text{Left}\,([\mathbf{x}]) &= [1,2] \times [-1,1] \\ \text{Right}\,([\mathbf{x}]) &= [1,2] \times [1,3]. \end{aligned}$$

Set inversion

(Luc Jaulin, Tuesday, 9h30-10h15).

Subpavings

A *subpaving* of $\mathbb{R}^n$ is a set of non-overlapping boxes of $\mathbb{R}^n$.

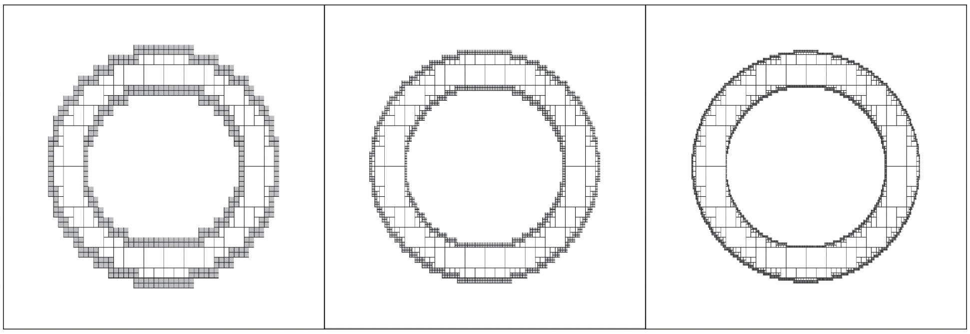
Compact sets $\mathbb{X}$ can be bracketed between inner and outer subpavings:

$$\mathbb{X}^- \subset \mathbb{X} \subset \mathbb{X}^+.$$

The set

$$\mathbb{X} = \{(x_1, x_2) \mid x_1^2 + x_2^2 \in [1, 2]\}$$

can be bracketed between subpavings as follows.



Set operations such as

$$\mathbb{Z} := \mathbb{X} + \mathbb{Y},\ \mathbb{X} := \mathbf{f}^{-1}(\mathbb{Y}), \mathbb{Z} := \mathbb{X} \cap \mathbb{Y} \dots$$

can be approximated by subpaving operations.

Stack-queue

A *queue* is a list on which two operations are allowed :

- add an element at the end (*push*)
- remove the first element (*pull*).

A *stack* is a list on which two operations are allowed :

- add an element at the beginning of the list (*stack*)
- remove the first element (*pop*).

Example: Let $\mathcal{L}$ be an empty queue.

k	operation	result
0		$\mathcal{L} = \emptyset$
1	push $(\mathcal{L}, a)$	$\mathcal{L} = \{a\}$
2	push $(\mathcal{L}, b)$	$\mathcal{L} = \{a, b\}$
3	$x :=$ pull $(\mathcal{L})$	$x = a, \mathcal{L} = \{b\}$
4	$x :=$ pull $(\mathcal{L})$	$x = b, \mathcal{L} = \emptyset.$

If $\mathcal{L}$ is a stack, the table becomes

k	operation	result
0		$\mathcal{L} = \emptyset$
1	stack $(\mathcal{L}, a)$	$\mathcal{L} = \{a\}$
2	stack $(\mathcal{L}, b)$	$\mathcal{L} = \{a, b\}$
3	$x :=$ pop $(\mathcal{L})$	$x = b, \mathcal{L} = \{a\}$
4	$x :=$ pop $(\mathcal{L})$	$x = a, \mathcal{L} = \emptyset.$

Set inversion

Characterize the set

$$\mathbb{X} = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{f}(\mathbf{x}) \in \mathbb{Y}\} = \mathbf{f}^{-1}(\mathbb{Y}),$$

where $\mathbb{Y} \subset \mathbb{R}^m$. and $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^m$.

Two subpavings $\mathbb{X}^-$ and $\mathbb{X}^+$ such that

$$\mathbb{X}^- \subset \mathbb{X} \subset \mathbb{X}^+,$$

can be obtained with the algorithm Sivia.

To test if a box $[\mathbf{x}]$ is inside or outside $\mathbb{X}$, we shall use the following tests.

$$\begin{array}{lll} \text{(i)} & [\mathbf{f}]([\mathbf{x}]) \subset \mathbb{Y} & \Rightarrow \quad [\mathbf{x}] \subset \mathbb{X} \\ \text{(ii)} & [\mathbf{f}]([\mathbf{x}]) \cap \mathbb{Y} = \emptyset & \Rightarrow \quad [\mathbf{x}] \cap \mathbb{X} = \emptyset. \end{array}$$

Show the demo of N. Delanoue.

.

Algo Sivia(in: $[\mathbf{x}]$; out: $\mathcal{L}^-, \mathcal{L}^+$)	
1	$\mathcal{L} := \{[\mathbf{x}]\}$; $\mathcal{L}^- = \emptyset$; $\mathcal{L}^+ := \emptyset$;
2	if $\mathcal{L} \neq \emptyset$, $[\mathbf{x}] := \text{pop}\,(\mathcal{L})$, else end;
3	if $[\mathbf{f}]([\mathbf{x}]) \subset \mathbb{Y}$, push($\mathcal{L}^-, [\mathbf{x}]$);push($\mathcal{L}^+, [\mathbf{x}]$); goto 2;
4	if $[\mathbf{f}]([\mathbf{x}]) \cap \mathbb{Y} = \emptyset$, goto 2;
5	if $w([\mathbf{x}]) < \varepsilon$, push($\mathcal{L}^+, [\mathbf{x}]$); goto 2;
6	stack($\mathcal{L}$,Left($[\mathbf{x}]$) ,Right($[\mathbf{x}]$)); goto 2.

Define

$$\mathbb{X}^- \stackrel{\text{def}}{=} \cup\Big\{[\mathbf{x}] \in \mathcal{L}^-\Big\}$$
$$\mathbb{X}^+ \stackrel{\text{def}}{=} \cup\Big\{[\mathbf{x}] \in \mathcal{L}^+\Big\}.$$

We have

$$\mathbb{X}^- \subset \mathbb{X} \subset \mathbb{X}^+.$$

Contractors

The operator $\mathcal{C}_{\mathbb{S}} : \mathbb{IR}^n \rightarrow \mathbb{IR}^n$ is a contractor for $\mathbb{S} \subset \mathbb{R}^n$ if $\forall [\mathbf{x}] \in \mathbb{IR}^n$,

$$\begin{cases} \mathcal{C}_{\mathbb{S}}([\mathbf{x}]) \subset [\mathbf{x}] & \text{(contractance)}, \\ \mathcal{C}_{\mathbb{S}}([\mathbf{x}]) \cap \mathbb{S} = [\mathbf{x}] \cap \mathbb{S} & \text{(correctness)}, \end{cases}$$

$\mathcal{C}_{\mathbb{S}}$ is *monotonic* iff	$[\mathbf{x}] \subset [\mathbf{y}] \Rightarrow \mathcal{C}_{\mathbb{S}}([\mathbf{x}]) \subset \mathcal{C}_{\mathbb{S}}([\mathbf{y}])$
$\mathcal{C}_{\mathbb{S}}$ is *minimal* iff	$\forall [\mathbf{x}],\ \mathcal{C}_{\mathbb{S}}([\mathbf{x}]) = [[\mathbf{x}] \cap \mathbb{S}]$
$\mathcal{C}_{\mathbb{S}}$ is *thin* iff	$\forall \mathbf{x} \in \mathbb{R}^n,\ \mathcal{C}_{\mathbb{S}}(\mathbf{x}) = \{\mathbf{x}\} \cap \mathbb{S}$
$\mathcal{C}_{\mathbb{S}}$ is *idempotent* iff	$\forall [\mathbf{x}], \mathcal{C}_{\mathbb{S}}\left(\mathcal{C}_{\mathbb{S}}([\mathbf{x}])\right) = \mathcal{C}_{\mathbb{S}}([\mathbf{x}])$

Sivia with contractors

The constraint $\mathbf{f}(\mathbf{x}) \in \mathbb{Y}$ defining $\mathbb{X} \stackrel{\text{def}}{=} \mathbf{f}^{-1}(\mathbb{Y})$ can be translated into nonlinear inequalities:

$$\left\{ \begin{array}{rcl} g_1(x_1, x_2, \ldots, x_n) & \leq & 0, \\ \vdots & \vdots & \vdots \\ g_m(x_1, x_2, \ldots, x_n) & \leq & 0. \end{array} \right.$$

Thus

$$\begin{array}{rcl} \mathbb{X} & = & \{\mathbf{x} \in \mathbb{R}^n \mid \max\left(g_1(\mathbf{x}), \ldots, g_m(\mathbf{x})\right) \leq 0\} \\ \neg\mathbb{X} & = & \{\mathbf{x} \in \mathbb{R}^n \mid \max\left(g_1(\mathbf{x}), \ldots, g_m(\mathbf{x})\right) > 0\}. \end{array}$$

Algorithm SiviaC(in: $[\mathbf{x}]$; out: $\mathcal{L}^-, \mathcal{L}^+$)	
1	$\mathcal{L} := \{[\mathbf{x}]\}$; $\mathcal{L}^- = \emptyset$; $\mathcal{L}^+ := \emptyset$;
2	if $\mathcal{L} \neq \emptyset$ then $[\mathbf{x}]$:= pop$(\mathcal{L})$ else end;
3	$[\mathbf{x}] := \mathcal{C}_{\mathbb{X}}([\mathbf{x}])$; if $[\mathbf{x}] = \emptyset$, goto 2
4	$[\mathbf{a}] := \mathcal{C}_{\neg\mathbb{X}}([\mathbf{x}])$;
5	if $[\mathbf{a}] \neq [\mathbf{x}]$, push$(\mathcal{L}^-, [\mathbf{x}] \backslash [\mathbf{a}])$;push$(\mathcal{L}^+, [\mathbf{x}] \backslash [\mathbf{a}])$;
6	if $(w([\mathbf{a}]) < \varepsilon)$, push$(\mathcal{L}^+, [\mathbf{a}])$; goto 2;
7	stack$(\mathcal{L}$,Left$([\mathbf{a}])$,Right$([\mathbf{a}]))$; goto 2.

Unconstrained global optimization

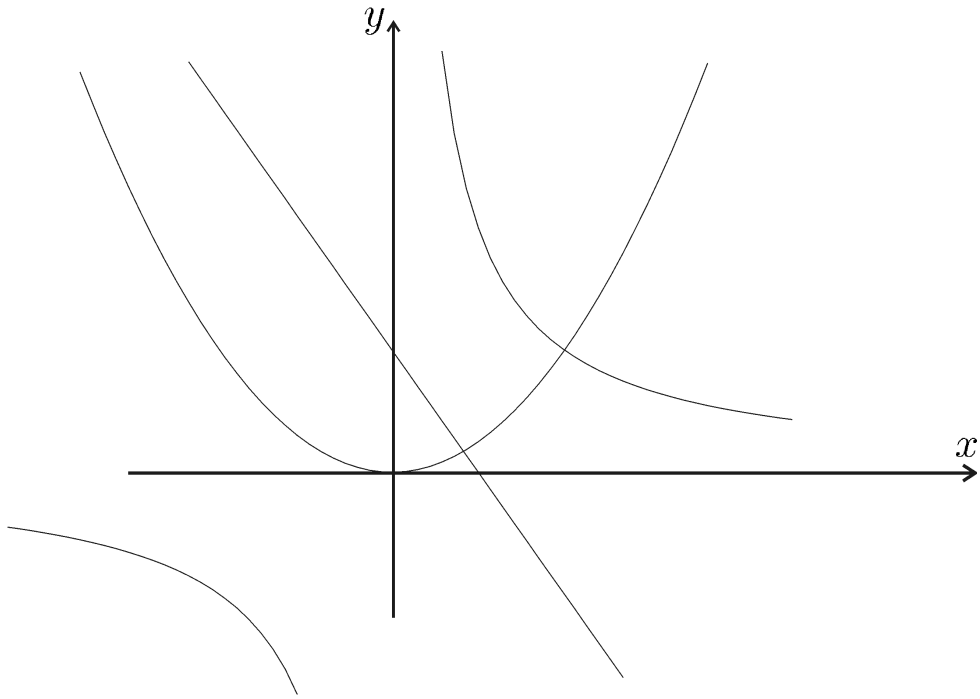
(Luc Jaulin, Tuesday, 11h30-12h15).

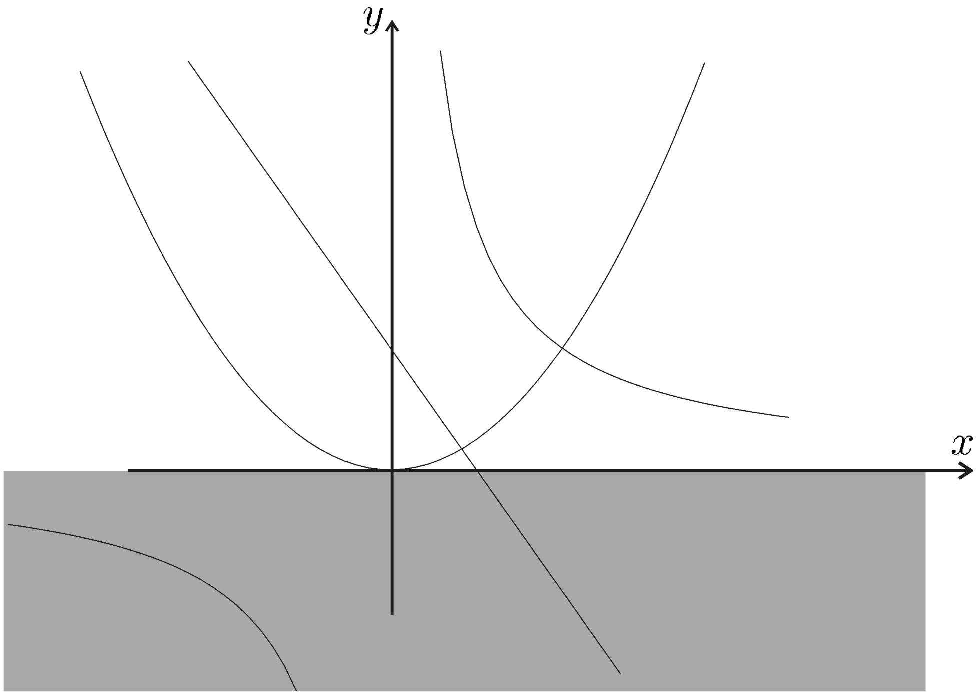
Constraints propagation (reminder)

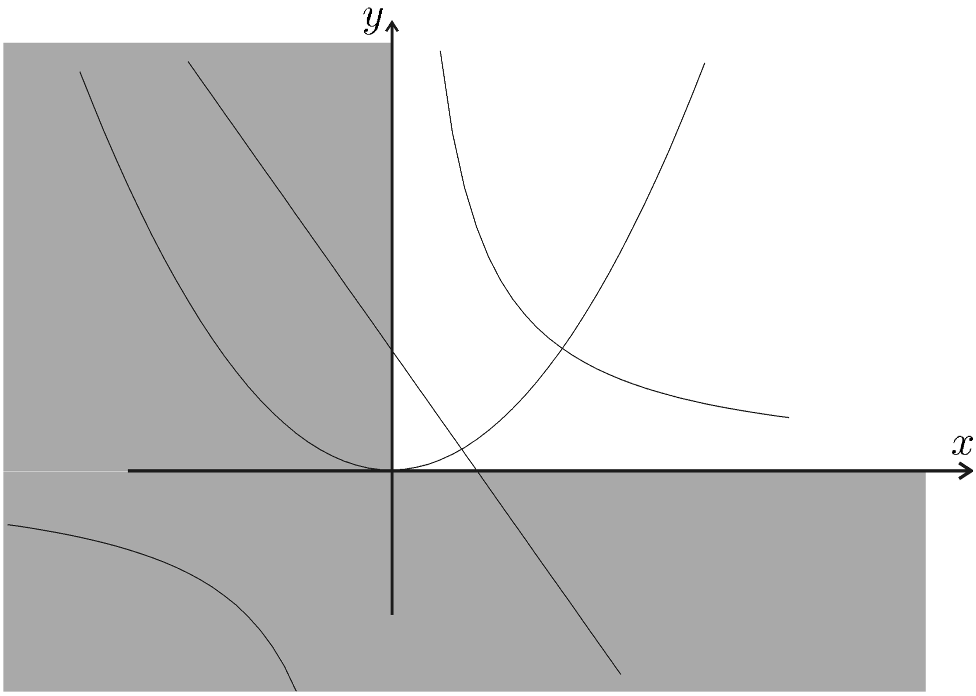
Consider the three following constraints

$$\begin{aligned} (C_1) &: \quad y = x^2 \\ (C_2) &: \quad xy = 1 \\ (C_3) &: \quad y = -2x + 1 \end{aligned}$$

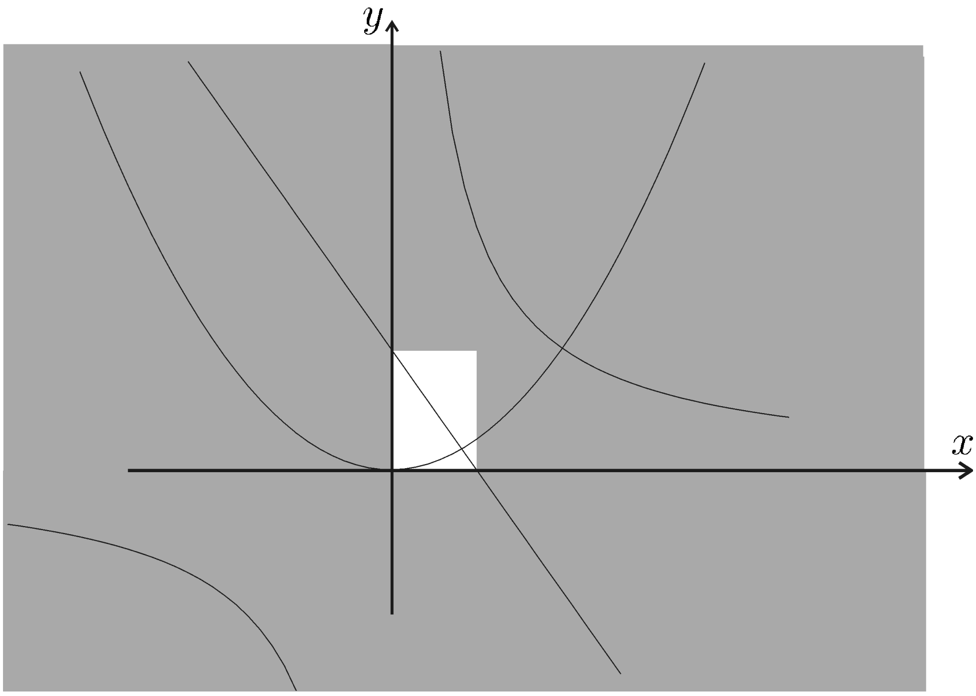
To each variable, we associate the domain $]-\infty, \infty[$.
A constraint propagation consists in projecting all constraints until equilibrium.


y
x

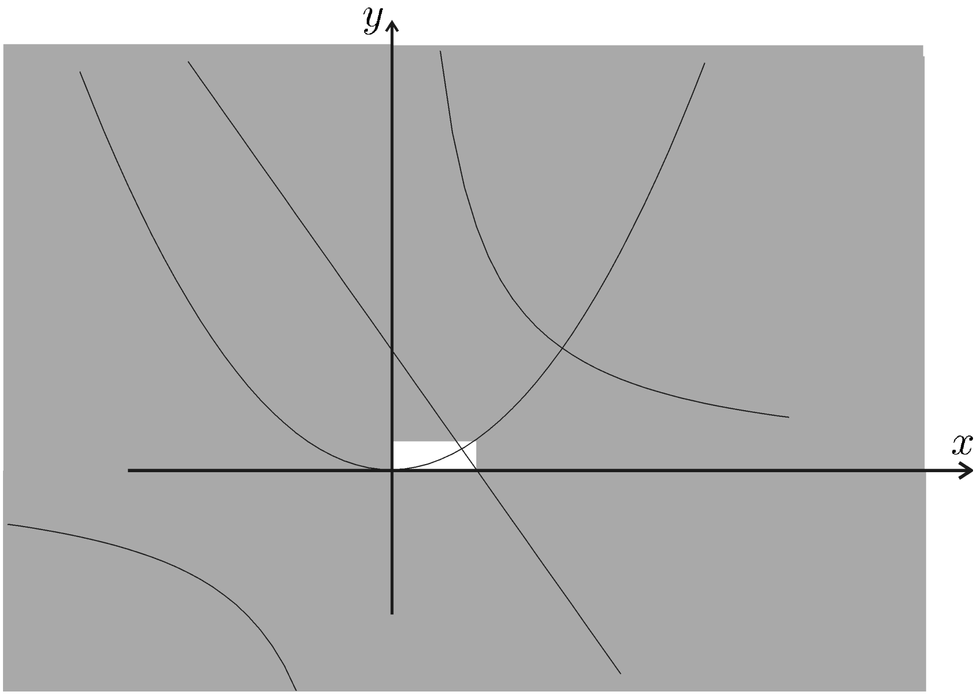

y
x

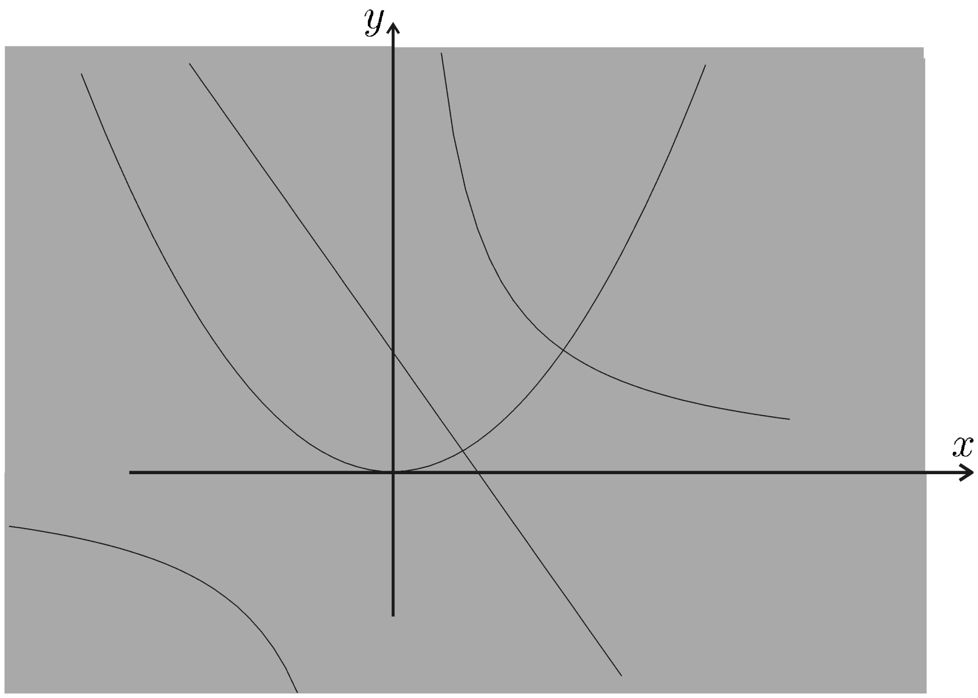


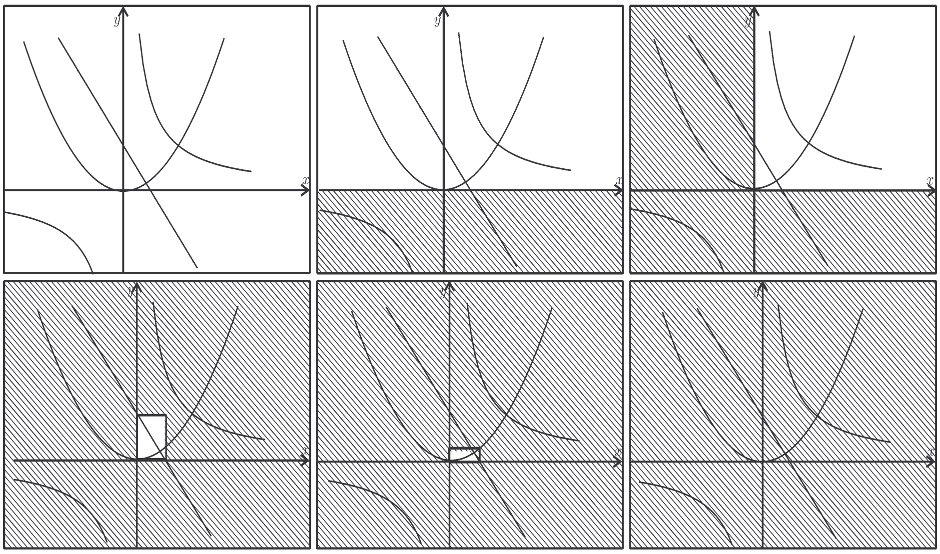
y
x



y
x


y
x


y
x



For more complex constraints, a decomposition is required. For instance, the CSP

$$\begin{array}{l} x + \mathsf{sin}(y) - xz \leq 0, \\ x \in [-1,1], y \in [-1,1], z \in [-1,1] \end{array}$$

can be decomposed into the following one.

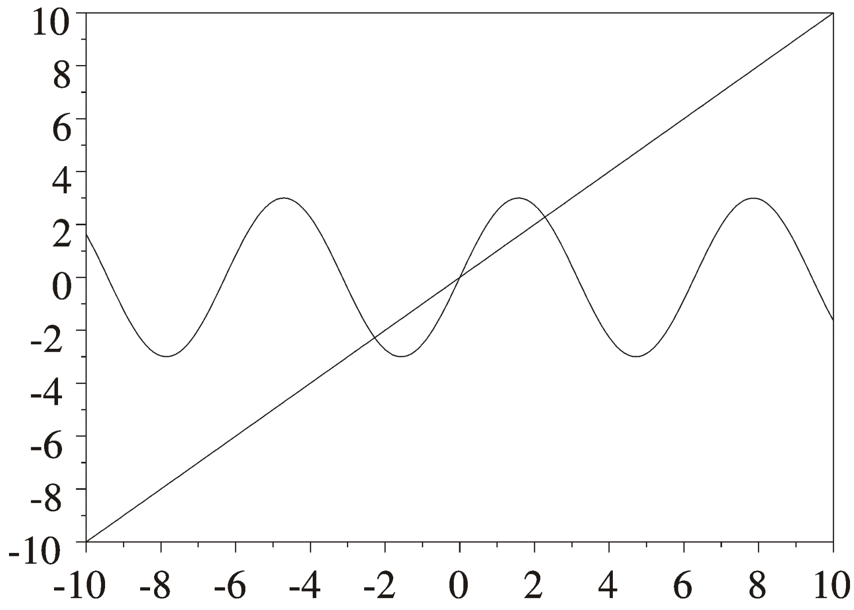
$$\left\{ \begin{array}{l} a = \mathsf{sin}(y) \\ b = x + a \\ c = xz \\ b - c = d \end{array} \right. , \quad \begin{array}{ll} x \in [-1,1] & a \in]-\infty, \infty[\\ y \in [-1,1] & b \in]-\infty, \infty[\\ z \in [-1,1] & c \in]-\infty, \infty[\\ & d \in]-\infty, 0] \end{array}$$

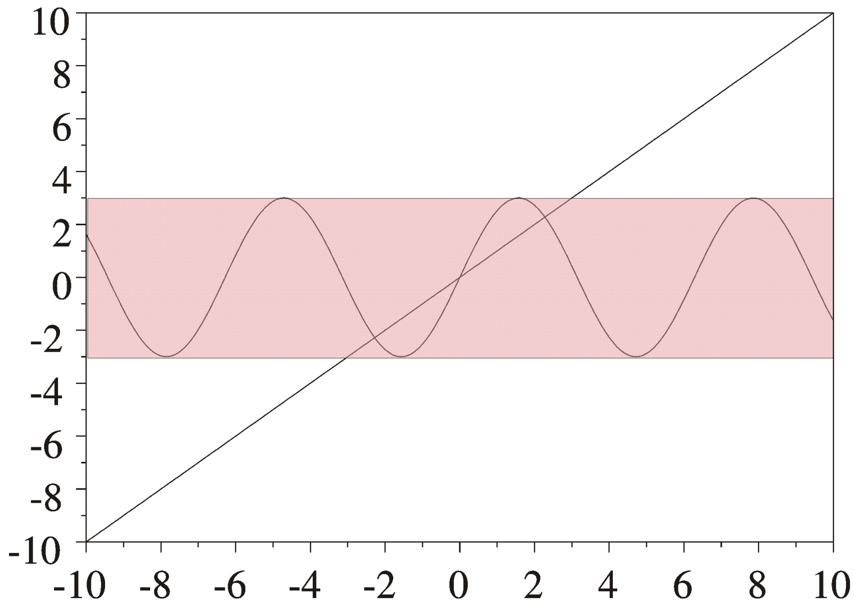
The decomposition introduces pessimism, and should be avoided, if possible.

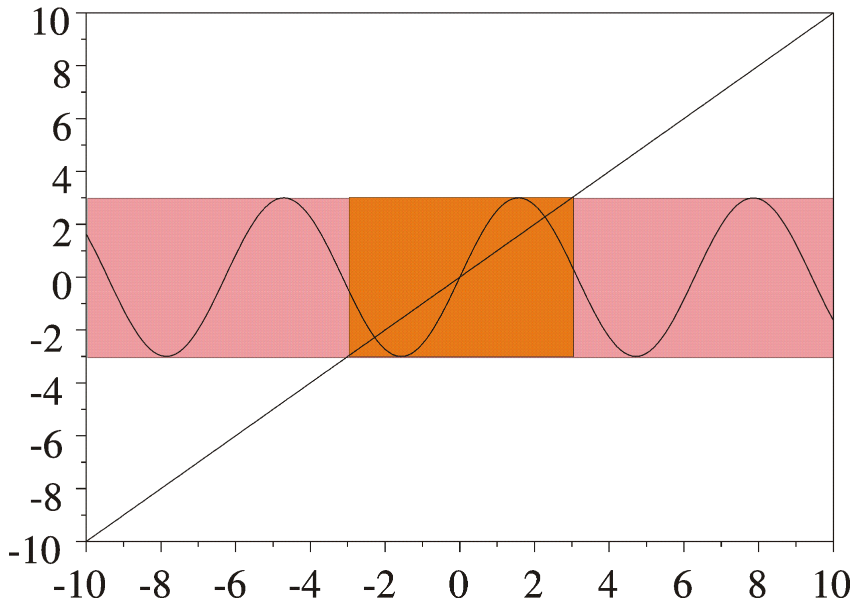
Constraints propagation can be used to solve nonlinear equations.

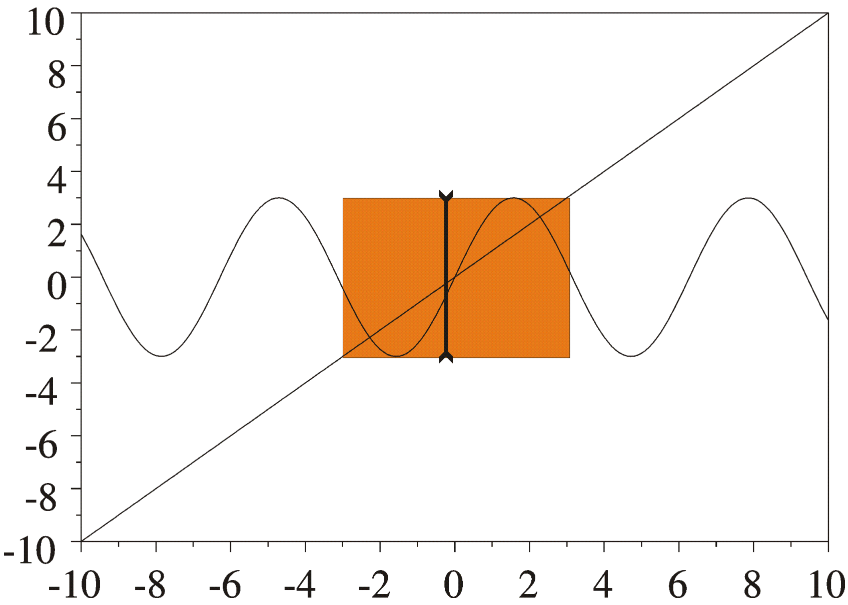
Consider the system

$$\begin{aligned} y &= 3\sin(x) \\ y &= x \end{aligned}$$

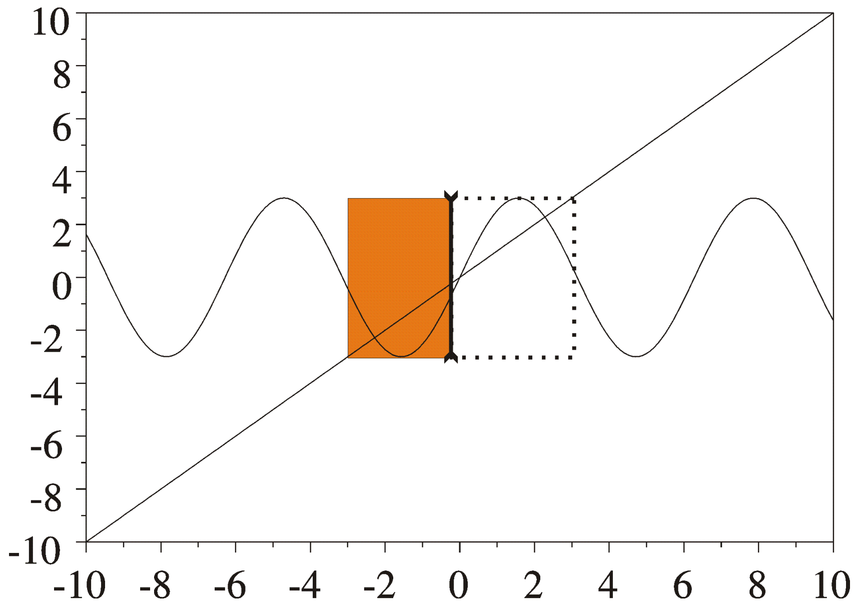

10
8
6
4
2
0
-2
-4
-6
-8
-10
-10 -8 -6 -4 -2 0 2 4 6 8 10


10
8
6
4
2
0
-2
-4
-6
-8
-10
-10 -8 -6 -4 -2 0 2 4 6 8 10

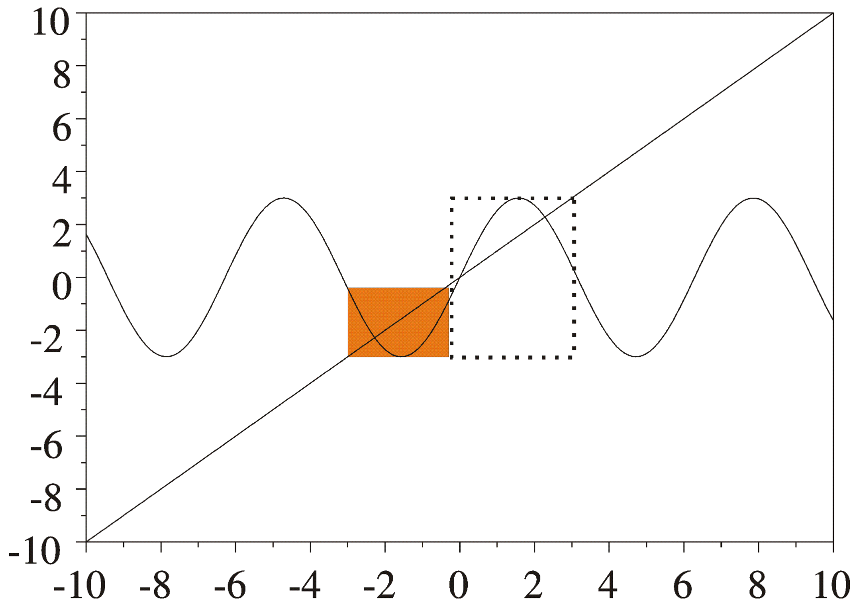

10
8
6
4
2
0
-2
-4
-6
-8
-10
-10
-8
-6
-4
-2
0
2
4
6
8
10



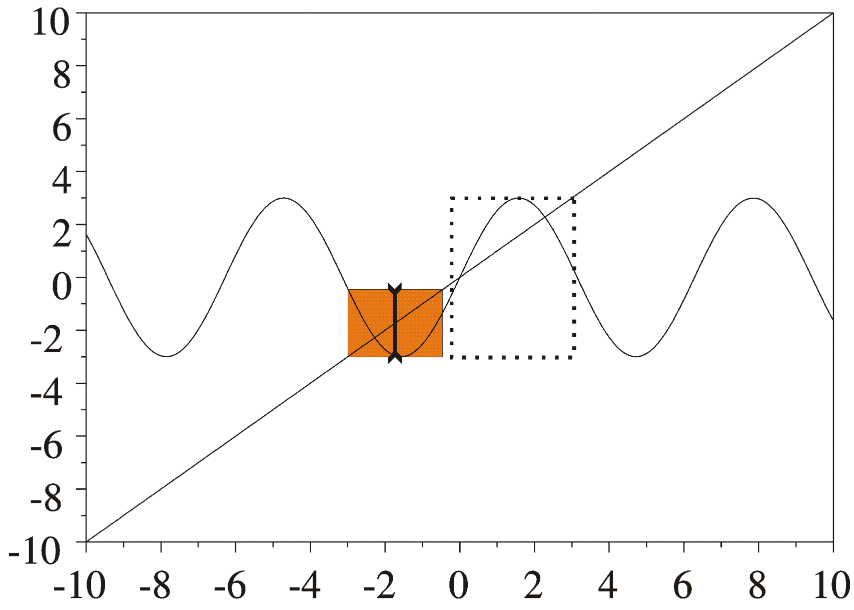
10
8
6
4
2
0
-2
-4
-6
-8
-10
-10 -8 -6 -4 -2 0 2 4 6 8 10



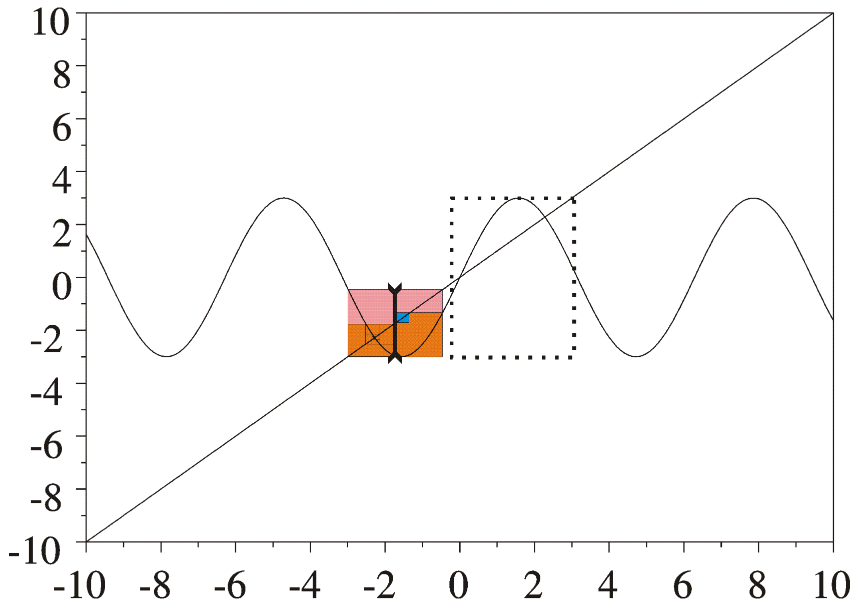
10
8
6
4
2
0
-2
-4
-6
-8
-10
-10 -8 -6 -4 -2 0 2 4 6 8 10

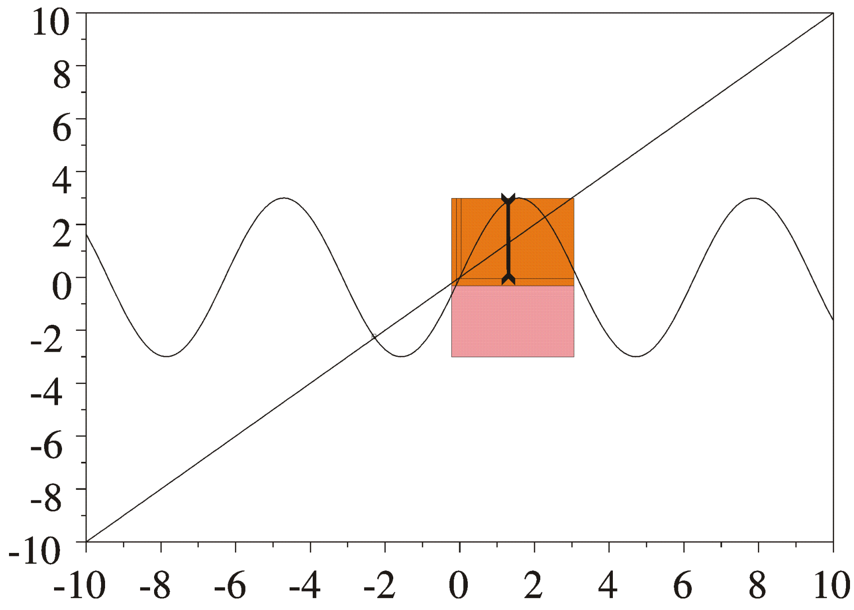


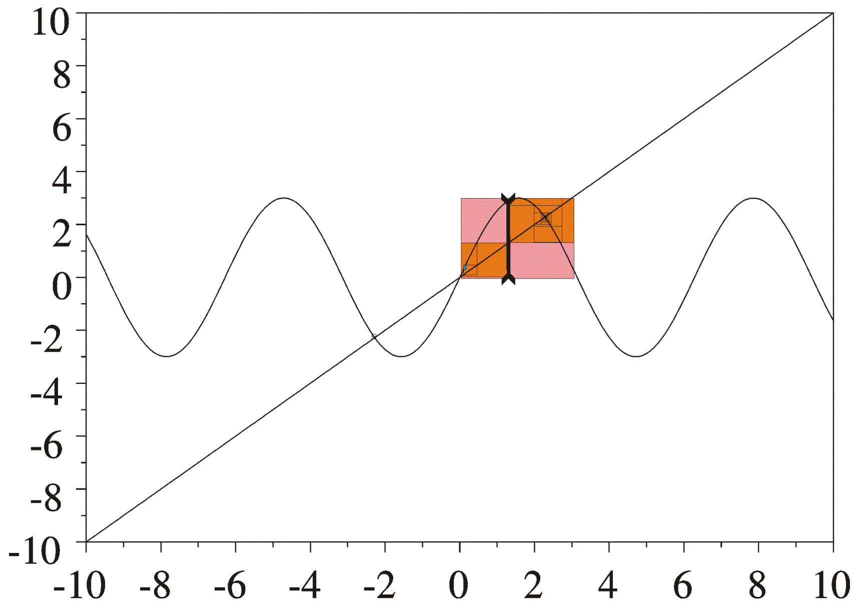
10
8
6
4
2
0
-2
-4
-6
-8
-10
-10 -8 -6 -4 -2 0 2 4 6 8 10



10
8
6
4
2
0
-2
-4
-6
-8
-10
-10
-8
-6
-4
-2
0
2
4
6
8
10


10
8
6
4
2
0
-2
-4
-6
-8
-10
-10 -8 -6 -4 -2 0 2 4 6 8 10


10
8
6
4
2
0
-2
-4
-6
-8
-10
-10 -8 -6 -4 -2 0 2 4 6 8 10



10
8
6
4
2
0
-2
-4
-6
-8
-10
-10
-8
-6
-4
-2
0
2
4
6
8
10

Minimization

Minimize $f(\mathbf{x})$ over a box $[\mathbf{x}] \subset \mathbb{R}^n$:

$$\hat{f} \stackrel{\text{def}}{=} \min_{\mathbf{x} \in [\mathbf{x}]} f(\mathbf{x}).$$

Algorithm Minimize(in: $[\mathbf{x}]$; out: $\mathcal{L}$)

1 $\mathcal{L} := \{[\mathbf{x}]\}$; $f^{+} = \infty$;
2 if $\forall [\mathbf{x}] \in \mathcal{L}$, $w([\mathbf{x}]) < \varepsilon$, return($\mathcal{L}$);
3 $[\mathbf{x}]$: = pull($\mathcal{L}$) ;
4 $f^{+} = \mathbf{min}\left(f^{+}, \text{localmin}\,(f, [\mathbf{x}])\right)$;
5 $[\mathbf{x}] := \mathcal{C}_{\{\mathbf{x} \mid f(\mathbf{x}) \leq f^{+}, \nabla f(\mathbf{x}) = 0, \mathbf{H}_f(\mathbf{x}) \succeq 0\}}([\mathbf{x}])$;
6 if ($w([\mathbf{x}]) < \varepsilon$) then push($\mathcal{L}$, $[\mathbf{x}]$) ; goto 2;
7 push($\mathcal{L}$,Left($[\mathbf{x}]$) ,Right($[\mathbf{x}]$)); goto 2.

Example (Collaboration with D. Henrion)

Consider the three-hump camel function

$$f(\mathbf{x}) = 2x_1^2 - 1.05x_1^4 + \frac{x_1^6}{6} - x_1x_2 + x_2^2.$$

Its gradient is

$$\nabla f(\mathbf{x}) = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \end{pmatrix} = \begin{pmatrix} 4x_1 - x_2 - 4.2x_1^3 + x_1^5 \\ 2x_2 - x_1 \end{pmatrix}.$$

Its Hessian is

$$\mathbf{H}(\mathbf{x}) = \begin{pmatrix} 4 + 5x_1^4 - 12.6x_1^2 & -1 \\ -1 & 2 \end{pmatrix}.$$

After decomposition, the constraints $f(\mathbf{x}) \leq f^+, \nabla f(\mathbf{x}) = 0, \mathbf{H}_f(\mathbf{x}) \succeq 0$ become

(i)$x_{12} = x_1^2$; (ii)$x_{13} = x_1^3$;
(iii)$x_{14} = x_1^4$; (iv)$x_{15} = x_1^5$;
(v)$x_{16} = x_1^6$; (vi)$x_{22} = x_2^2$; (vii)$a = x_1 x_2$

$$
\text{(viii)} \left\{ \begin{array}{ccc} 2x_{12} - 1.05\ x_{14} + \frac{x_{16}}{6} - a + x_{22} & \leq & f^+ \\ 4x_1 - x_2 - 4.2\ x_{13} + x_{15} & = & 0 \\ 2x_2 - x_1 & = & 0 \\ \begin{pmatrix} 4 + 5x_{14} - 12.6x_{12} & -1 \\ -1 & 2 \end{pmatrix} & \succeq & 0 \end{array} \right.
$$

Note that the eigen values of the Hessian matrix are given by:

$$\begin{aligned}\lambda_{1,2} &= \frac{5}{2}x_{14} - 6.3x_{12} + 3 \pm \frac{1}{2}(20x_{14} - 50.4x_{12} \\ &\quad -126x_{12}x_{14} + 158.76x_{12}^2 + 25x_{14}^2 + 8)^{\frac{1}{2}}.\end{aligned}$$

should be positive. Since the 7 variables constraint (viii), is an LMI, it can be projected and should not be decomposed.

LMIs

A linear matrix inequality/

$$\mathbf{A}(\mathbf{x}) \stackrel{\text{def}}{=} \mathbf{A}_0 + x_1\mathbf{A}_1 + \cdots + x_m\mathbf{A}_m \succeq 0,$$

where $\mathbf{x} \in \mathbb{R}^m$ is a vector of variables and the $\mathbf{A}_i$ are symmetric matrices.

An *LMI set* is a subset $\mathbb{X}$ of $\mathbb{R}^m$ defined by an LMI.

Computing $[\mathbb{X}]$ is tractable.

Example 1. A set of linear constraints (equalities or inequalities) is an LMI:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + b_1 & \geq 0 \\ a_{21}x_1 + a_{22}x_2 + b_2 & \geq 0 \end{cases}$$

is equivalent to

$$\begin{pmatrix} a_{11}x_1 + a_{12}x_2 + b_1 & 0 \\ 0 & a_{21}x_1 + a_{22}x_2 + b_2 \end{pmatrix} \succeq 0,$$

i.e.,

$$\begin{pmatrix} b_1 & 0 \\ 0 & b_2 \end{pmatrix} + x_1 \begin{pmatrix} a_{11} & 0 \\ 0 & a_{21} \end{pmatrix} + x_2 \begin{pmatrix} a_{12} & 0 \\ 0 & a_{22} \end{pmatrix} \succeq 0.$$

Example 2. An ellipsoid of $\mathbb{R}^n$ is an LMI set :

$$3x_1^2 + 2x_2^2 - 2x_1x_2 \le 5$$
$$\Leftrightarrow$$
$$\begin{pmatrix} 1 & x_1 & x_2 \\ x_1 & 2 & 1 \\ x_2 & 1 & 3 \end{pmatrix} \succeq 0$$
$$\Leftrightarrow$$
$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 1 \\ 0 & 1 & 3 \end{pmatrix} + x_1 \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} + x_2 \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} \succeq 0.$$

Minimax Optimization

(Luc Jaulin, Tuesday, 12h15-13h00).

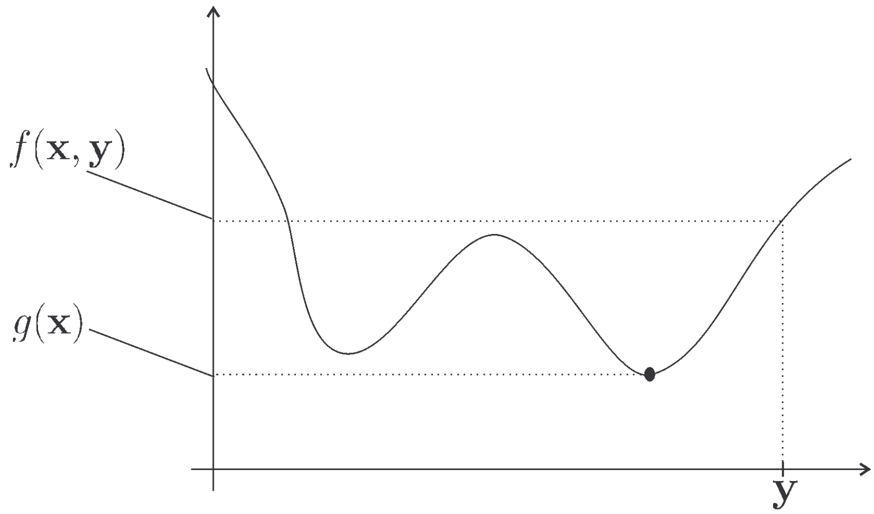
Perturbed minimization

Consider the function

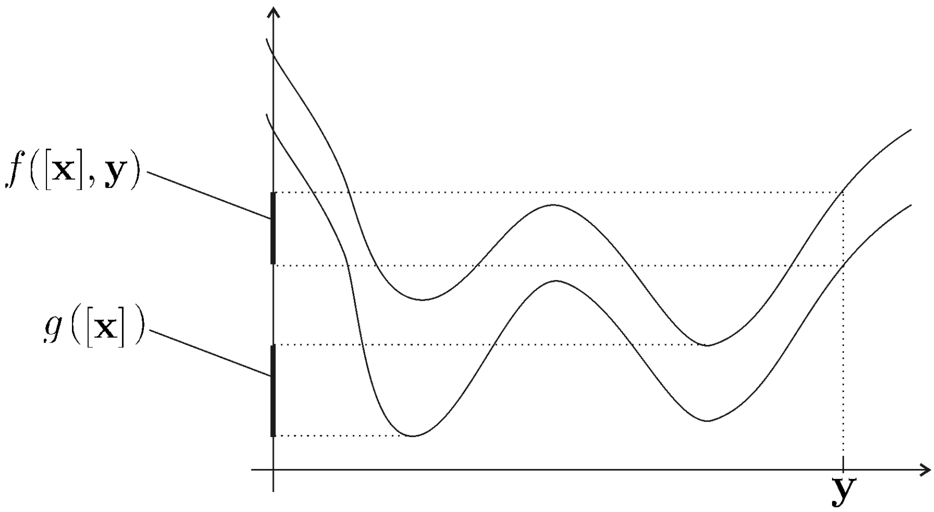
$$g(\mathbf{x}) = \min_{\mathbf{y} \in [\mathbf{y}]} f(\mathbf{x}, \mathbf{y}).$$

We need an inclusion function $[g]([\mathbf{x}])$ for $g(\mathbf{x})$.

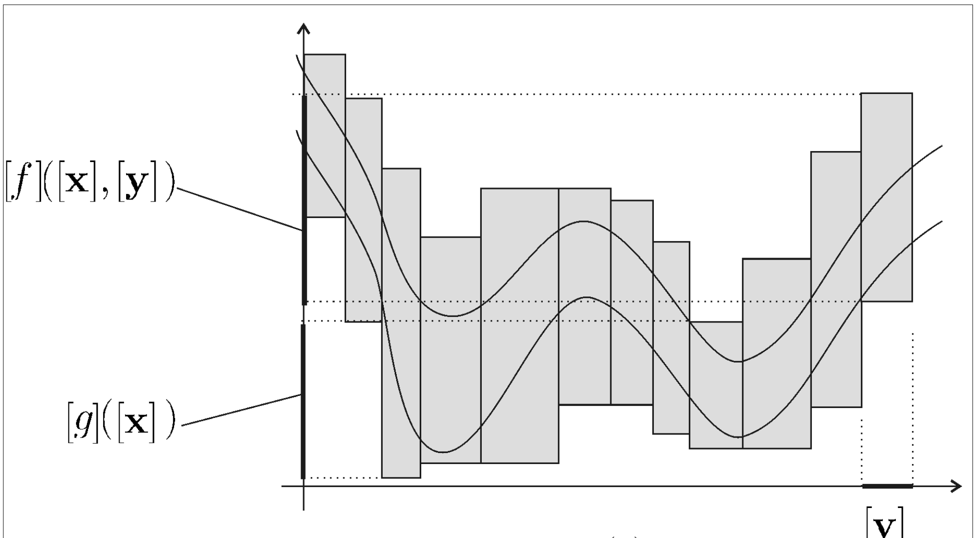
The real number f^+ represents the best known upper bound for $g([\mathbf{x}])$.



(a)



(b)



(c)

Recall that

$$\min([3,7],[2,9],[4,5]) = [2,5]).$$

Algorithm PertMin(in: $[\mathbf{x}], [\mathbf{y}]$,$[f]$,; out: $[g]([\mathbf{x}])$)

1 $\mathcal{L} := \{[\mathbf{y}]\}$; $f^+ = \infty; \varepsilon_y = w([\mathbf{x}]) + \varepsilon$;

2 if $\forall [\mathbf{y}] \in \mathcal{L}$, $w([\mathbf{y}]) < \varepsilon_y$, return($\mathcal{L}$);

3 $[\mathbf{y}] :=$ pull($\mathcal{L}$) ;

4 $f^+ = \min\left(f^+, \text{ub}\left([f]\left([\mathbf{x}], \text{center}\left([\mathbf{y}]\right)\right)\right)\right)$;

5 if $[f]([\mathbf{x}], [\mathbf{y}]) > f^+$, goto 2;

6 if $(w([\mathbf{y}]) \leq \varepsilon_y$, push($\mathcal{L}, [\mathbf{y}]$), goto 2;

7 push($\mathcal{L}$, Left $([\mathbf{y}])$, Right $([\mathbf{x}])$); goto 2.

The perturbed maximization problem can be solved using PertMin: since

$$h(\mathbf{x}) = \max_{\mathbf{y}\in[\mathbf{y}]} f(\mathbf{x},\mathbf{y}) = -\min_{\mathbf{y}\in[\mathbf{y}]} -f(\mathbf{x},\mathbf{y}),$$

an inclusion function for $h(\mathbf{x})$ can be obtained by

$$[h]([\mathbf{x}]) = -\text{PertMin}([\mathbf{x}],[\mathbf{y}],[-f]).$$

Perturbed minimization with constraints

The function

$$g(\mathbf{x}) = \min_{\substack{\mathbf{y} \in [\mathbf{y}] \\ \text{s.t. } \mathbf{h}(\mathbf{x},\mathbf{y}) \leq 0}} f(\mathbf{x}, \mathbf{y})$$

can be rewritten as

$$g(\mathbf{x}) = \min_{\mathbf{y} \in [\mathbf{y}]} f(\mathbf{x}, \mathbf{y}) + \eta(\mathbf{h}(\mathbf{x}, \mathbf{y})),$$

where

$$\eta(\mathbf{v}) = \begin{cases} 0 & \text{if } \mathbf{v} \leq 0 \\ \infty & \text{otherwise} \end{cases}$$

The minimal inclusion function for η is $[\eta] = [\eta(\underline{v}), \eta(\bar{v})]$.

For instance

$$\begin{aligned}
\eta(-1,-3,-2) &= 0; \\
\eta(-1,-3,2) &= \infty; \\
[\eta]([-3,-1],[-3,2],[-2,5]) &= [0,\infty]; \\
[\eta]([-3,-1],[-3,-2],[-2,-1]) &= [0,0]; \\
[\eta]([-3,-1],[1,2],[-2,5]) &= [\infty,\infty];
\end{aligned}$$

An inclusion function for $f(\mathbf{x},\mathbf{y})+\eta(\mathbf{h}(\mathbf{x},\mathbf{y}))$ is

$$[f]([\mathbf{x}],[\mathbf{y}])+[\eta]([\mathbf{h}]([\mathbf{x}],[\mathbf{y}])).$$

PertMin can thus be used to get an inclusion function $[g]([\mathbf{x}])$ for

$$g(\mathbf{x}) \stackrel{\text{def}}{=} \min\{f(\mathbf{x},\mathbf{y}),\mathbf{y}\in[\mathbf{y}],\mathbf{h}(\mathbf{x},\mathbf{y})\leq 0\}$$

if inclusion functions for $f(\mathbf{x},\mathbf{y})$ and $\mathbf{h}(\mathbf{x},\mathbf{y})$ are available.

Minimax optimization

Consider the problem of computing an enclosure for

$$f_3 = \underbrace{\min_{\substack{x_3 \in [x_3] \\ \sin x_3 \le 0}} \overbrace{\max_{\substack{x_2 \in [x_2] \\ x_3^2 + x_2 \le 0}} \overbrace{\min_{\substack{x_1 \in [x_1] \\ x_1^2 x_2 x_3 \le 0}} \overbrace{x_1 x_2 + x_3}^{}}^{}}^{}}_{}$$

$$f_3 = \min_{\substack{x_3 \in [x_3] \\ \sin x_3 \le 0}} \underbrace{\max_{\substack{x_2 \in [x_2] \\ x_3^2 + x_2 \le 0}} \underbrace{\min_{\substack{x_1 \in [x_1] \\ x_1^2 x_2 x_3 \le 0}} \underbrace{x_1 x_2 + x_3}_{f_0(x_1, x_2, x_3)}}_{f_1(x_2, x_3)}}_{f_2(x_3)}.$$

It can be rewritten as

$$f_3 = \min_{x_3 \in [x_3]} \Bigg(\eta\,(\sin x_3) + \max_{x_2 \in [x_2]} \{ -\eta \left(x_3^2 + x_2 \right) + \min_{x_1 \in [x_1]} \left(\eta(x_1^2 x_2 x_3) + x_1 x_2 + x_3 \right) \} \Bigg)$$

An enclosure for the real number f_3 can thus be obtained using PertMin.

Remark: The operators min and max cannot commute in general. For instance,

$$\begin{aligned}\max_{x\in\{-1,1\}} \min_{y\in\{-1,1\}} xy &= ???\\ \min_{y\in\{-1,1\}} \max_{x\in\{-1,1\}} xy &= ???\end{aligned}$$

Remark: The operators min and max cannot commute in general. For instance,

$$\max_{x\in\{-1,1\}} \min_{y\in\{-1,1\}} xy = \max_{x\in\{-1,1\}} x.(-sign(x)) = -1$$
$$\min_{y\in\{-1,1\}} \max_{x\in\{-1,1\}} xy = \min_{y\in\{-1,1\}} sign(y).y = 1.$$

We always have

$$\max_{x\in[x]} \min_{y\in[y]} f(x,y) \leq \min_{y\in[y]} \max_{x\in[x]} f(x,y).$$

Set projection

Problems involving $\exists$ and $\forall$ are closely related to minimax problems. For instance,

$$\begin{array}{rcl} \forall p_3 & \in & [1,3], \exists p_2 \in [1,2], \\ \forall p_1 & \in & [0,1], p_1 + p_2 p_3 \leqslant 1 \end{array}$$

is equivalent to

$$???.$$

Set projection

Problems involving $\exists$ and $\forall$ are closely related to minimax problems. For instance,

$$\begin{aligned} \forall p_3 &\in [1,3], \exists p_2 \in [1,2], \\ \forall p_1 &\in [0,1], p_1 + p_2 p_3 \leqslant 1 \end{aligned}$$

is equivalent to

$$\max_{p_3 \in [1,3]} \min_{p_2 \in [1,2]} \max_{p_1 \in [0,1]} p_1 + p_2 p_3 \leqslant 1.$$

Consider the set

$$\begin{aligned}\mathbb{S} &= \{\mathbf{x} \in [\mathbf{x}] | \exists \mathbf{y} \in [\mathbf{y}], \mathbf{f}(\mathbf{x}, \mathbf{y}) \leq \mathbf{0}\} \\ &= \left\{\mathbf{x} \in [\mathbf{x}] | \min_{\mathbf{y} \in [\mathbf{y}]} \max(f_1(\mathbf{x}, \mathbf{y}), \ldots, f_m(\mathbf{x}, \mathbf{y})) \leq 0\right\}\end{aligned}$$

From PertMin, an inclusion function $[g](\mathbf{x})$ for

$$g(\mathbf{x}) = \min_{\mathbf{y} \in [\mathbf{y}]} \max(f_1(\mathbf{x}, \mathbf{y}), \ldots, f_m(\mathbf{x}, \mathbf{y})).$$

can be obtained and SIVIA can thus be used to characterize $\mathbb{S}$.

Epigraphs (Collaboration with M. Dao, M. Lhommeau)

Consider the optimization problem:

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}) \text{ s.t. } \mathbf{g}(\mathbf{x}) \leq \mathbf{0}.$$

Define its epigraph as

$$\mathbb{S} = \{(\mathbf{x}, a) \in \mathbb{R}^n \times \mathbb{R} \mid a \geq f(\mathbf{x}) \text{ and } \mathbf{g}(\mathbf{x}) \leq \mathbf{0}\}.$$

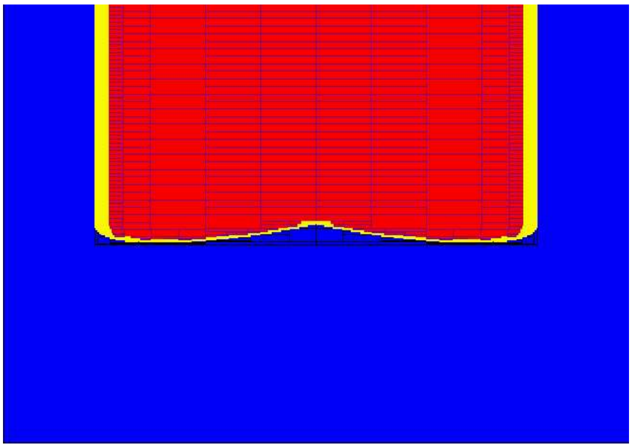
The ith profile of $\mathbb{S}$ is defined by

$$\mathbb{S}_i = \begin{array}{c} \{(x_i, a) \in \mathbb{R} \times \mathbb{R} \mid \exists\,(x_1, \dots, x_{i-1}, x_i, \dots, x_n) \\ \mid a \geq f(\mathbf{x}) \text{ and } \mathbf{g}(\mathbf{x}) \leq \mathbf{0}\} \end{array}.$$

Consider, for instance, the following problem

$$\min_{\mathbf{x}\in\mathbb{R}^n} \sin x_1 x_2 \text{ s.t. } x_1^2 + x_2^2 \in [1, 2].$$

The profiles $\mathbb{S}_1$ (and also $\mathbb{S}_2$) below, has been obtained by Proj2d.



Example: For

$$f(\mathbf{p}) = \max_{t \in \{1,2,3\}} \left| e^{-p_1 t} + 1.01.e^{-p_2 t} - y_t \right|$$

where

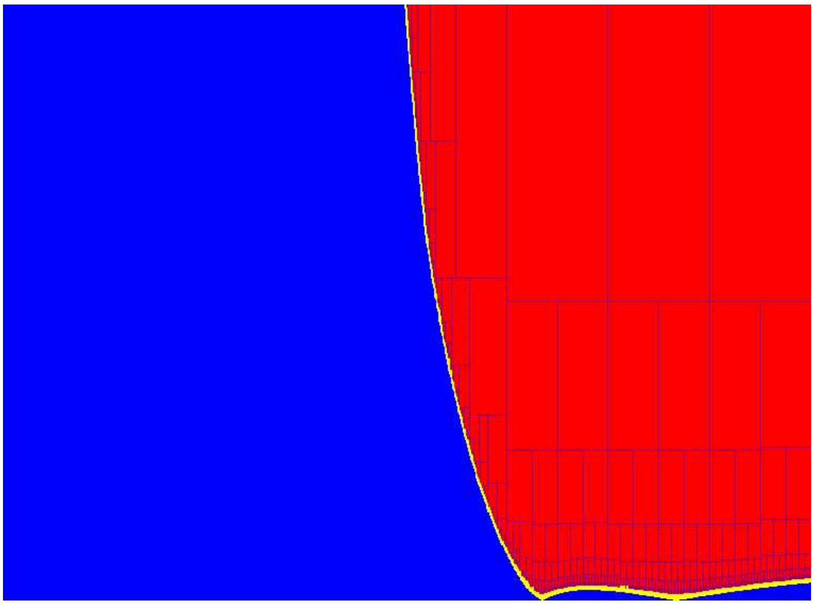
$$y_1 = 0.504,\ y_2 = 0.153 \text{ and } y_3 = 0.052.$$

the problem corresponds to an estimation problem where the model is almost non-identifiable.

Its profiles can be obtained by the following Proj2d program.

```
Variables
  p1 in [-3,3]
  p2 in [-3,3]
  a in [0,1]
Constraints
 max(abs(exp(-p1*1)+1.01*exp(-p2*1)-0.504),
     abs(exp(-p1*2)+1.01*exp(-p2*2)-0.153),
     abs(exp(-p1*3)+1.01*exp(-p2*3)-0.052))
               -a in [-1000,0]
Projected variables
  p1;a;
Epsilon
  0.05
EndOfFile
```

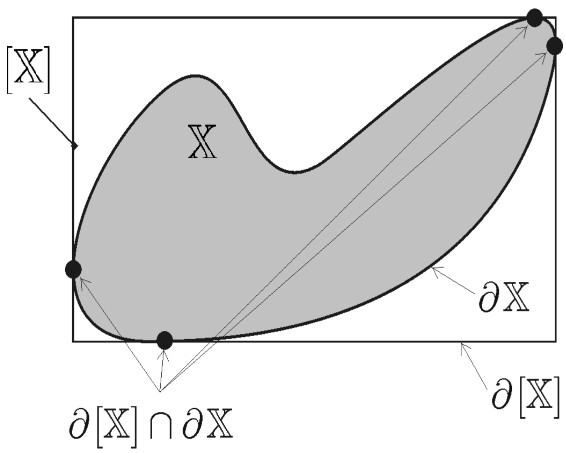
The picture on the (p_1, a) space shows a unique global minimizer $\mathbf{p} \simeq (1, 2)$ and a quasi-global one $\simeq (2, 1)$.



Interval hull

Given a set $\mathbb{X}$, compute two boxes $[\mathbf{x}_{\text{in}}]$ and $[\mathbf{x}_{\text{out}}]$ such that

$$[\mathbf{x}_{\text{in}}] \subset [\mathbb{X}] \subset [\mathbf{x}_{\text{out}}].$$



Since

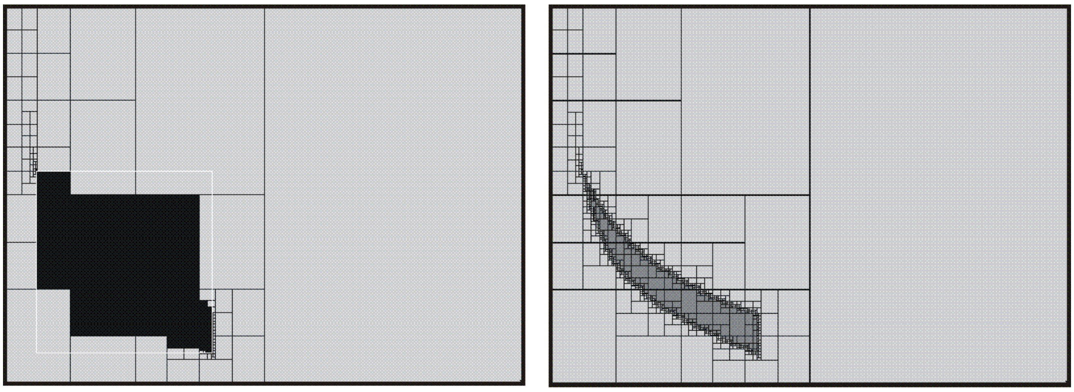
$$[\mathbb{X}] = \left[\min_{\mathbf{x}\in\mathbb{X}} x_1, \max_{\mathbf{x}\in\mathbb{X}} x_1\right] \times \cdots \times \left[\min_{\mathbf{x}\in\mathbb{X}} x_n, \max_{\mathbf{x}\in\mathbb{X}} x_n\right],$$

we can compute the enclosure

$$[\underline{x}_1^-, \underline{x}_1^+] \times \cdots \times [\underline{x}_n^-, \underline{x}_n^+] \subset [\mathbb{X}] \subset [\bar{x}_1^-, \bar{x}_1^+] \times \cdots \times [\bar{x}_n^-, \bar{x}_n^+].$$

Example: Assume that

$$\mathbb{X} = \{(x_1, x_2) \in [0,5]^2 \mid \forall t \in [0,1], \\ |t^2 + 2t + 1 - x_1 e^{x_2 t}| \leqslant 1\}.$$



(a) (b)

Constraints propagation for estimation

(Luc Jaulin, Wednesday, 14h30-15h45).

Constraint propagation (remainder)

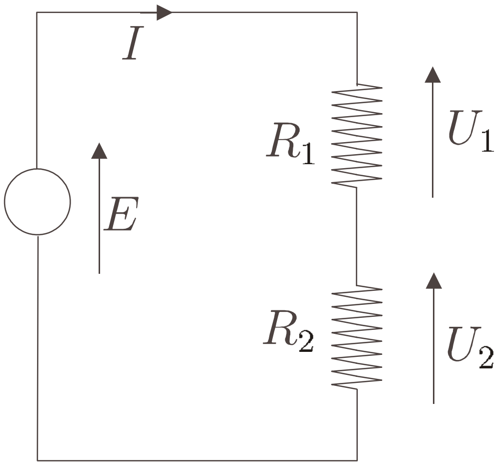
A CSP is composed of
1) a set of variables $\mathcal{V} = \{x_1, \ldots, x_n\}$,
2) a set of constraints $\mathcal{C} = \{c_1, \ldots, c_m\}$ and
3) a set of interval domains $\{[x_1], \ldots, [x_n]\}$.

Principle of propagation techniques: contract $[\mathbf{x}] = [x_1] \times \cdots \times [x_n]$ as follows:

$$((((((([\mathbf{x}] \sqcap c_1) \sqcap c_2) \sqcap \ldots) \sqcap c_m) \sqcap c_1) \sqcap c_2) \ldots,$$

until a steady box is reached.

Constraint propagation for estimation (Collaboration with I. Braems, M. Kieffer, E. Walter)



Assume that

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

where P is the power delivered by the battery. The constraints are

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

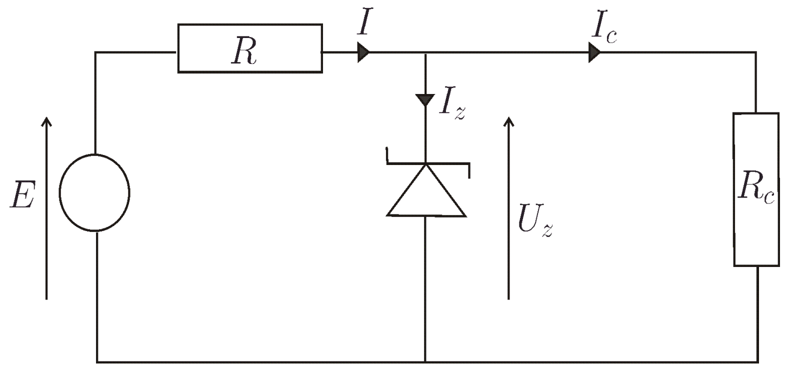
IntervalPeeler gets

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

Question:

Is the contraction optimal ?

How can we check it ?



R
I
I_c
I_z
E
U_z
R_c

It is known that

$$U_z \in [6V, 7V], r \in [7, 8]\Omega, U_0 \in [6, 6.2]V$$
$$R \in [100, 110]\Omega, E \in [18, 20]V, I_z \in [0.001, \infty]A$$
$$I \in]-\infty, \infty[A,\ I_c \in]-\infty, \infty[A, R_c \in [50, 60]\Omega.$$

The constraints are

Zener diode	$I_z = \max(0, \frac{U_z - U_0}{r}),$
Ohm rule	$U_z = R_c I_c,$
Current rule	$I = I_c + I_z,$
Voltage rule	$E = RI + U_z.$

IntervalPeeler contracts the domains into:

$$U_z \in [6,007; 6,518], r \in [7,8]\Omega,$$
$$U_0 \in [6,6.2]V, R \in [100,110]\Omega,$$
$$E \in [18,20]V, I_z \in [0.001,0.398]A$$
$$I \in [0.11; 0.14]A,\ \ I_c \in [0.1; 0,13]A,$$
$$R_c \in [50,60]\Omega$$

Forward-backward propagation

Select the primitive constraints in an optimal order.
Consider the constraint

$$f(\mathbf{x}) \in [y],$$

where

$$f(\mathbf{x}) = x_1 \exp(x_2) + \sin(x_3).$$

First write an algorithm that computes $y = f(\mathbf{x})$, by a finite sequence of elementary operations.

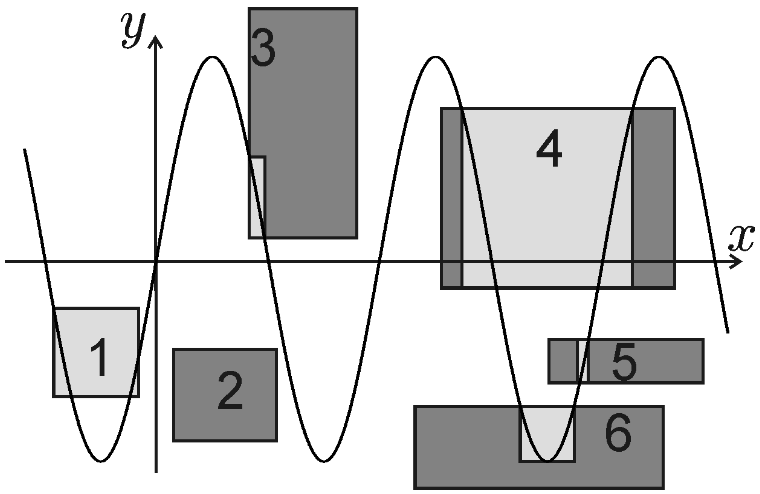
$$\begin{aligned} a_1 &:= \exp(x_2); \\ a_2 &:= x_1 a_1; \\ a_3 &:= \sin(x_3); \\ y &:= a_2 + a_3. \end{aligned}$$

Write an interval counterpart to this algorithm:

$$[a_1] := \mathsf{exp}\,([x_2])\,;$$
$$[a_2] := [x_1] * [a_1];$$
$$[a_3] := \mathsf{sin}\,([x_3])\,;$$
$$[y] := [y] \cap [a_2] + [a_3].$$

5 $[a_2] := ([y] - [a_3]) \cap [a_2];$
6 $[a_3] := ([y] - [a_2]) \cap [a_3];$
7 $[x_3] := \sin^{-1}([a_3]) \cap [x_3];$
8 $[a_1] := ([a_2]/[x_1]) \cap [a_1];$
9 $[x_1] := ([a_2]/[a_1]) \cap [x_1];$
10 $[x_2] := \log([a_1]) \cap [x_2].$

At Step 8, $\sin^{-1}([a_3]) \cap [x_3]$ returns $[\{x_3 \in [x_3] \mid \sin(x_3) \in [a_3]\}]$



y
x
1
2
3
4
5
6

The final contractor is given below

Algorithm $\mathcal{C}_{\downarrow\uparrow}$(inout: $[\mathbf{x}]$)

$[a_1] := \exp([x_2])$;

$[a_2] := [x_1] * [a_1]$;

$[a_3] := \sin([x_3])$;

$[y] := [y] \cap ([a_2] + [a_3])$;

$[a_2] := ([y] - [a_3]) \cap [a_2]$;

$[a_3] := ([y] - [a_2]) \cap [a_3]$;

$[x_3] := \sin^{-1}([a_3]) \cap [x_3]$;

$[a_1] := ([a_2]/[x_1]) \cap [a_1]$;

$[x_1] := ([a_2]/[a_1]) \cap [x_1]$;

$[x_2] := \log([a_1]) \cap [x_2]$.

Application to state estimation

Consider the nonlinear discrete-time system

$$\begin{cases} \begin{pmatrix} x_1(k) \\ x_2(k) \end{pmatrix} = \begin{pmatrix} \frac{x_1(k-1)}{10} + x_2(k-1)\, e^{x_1(k-1)} \\ x_1(k-1) + \frac{x_2^2(k-1)}{10} + \sin k \end{pmatrix}, \\ y(k) = x_2(k)/x_1(k), \end{cases}$$

with $k \in \{1, \ldots, 15\}$.

Simulation: $\mathbf{x}^*(0) = (-1 \ \ 0)^{\mathsf{T}}$ and a random output error with a uniform distribution in $[-e, e]$.

Algo ϕ(in: $x_1(0), x_2(0)$; out: $y(1), \ldots, y(15)$)

for $k := 1$ to 15,

$\quad x_1(k) := 0.1\, x_1(k-1) + x_2(k-1) \,.\, \exp(x_1(k-1))$;

$\quad x_2(k) := x_1(k-1) + 0.1\, x_2^2(k-1) + \sin(k)$;

$\quad y(k) := x_2(k) / x_1(k)$.

This simulator is decomposed

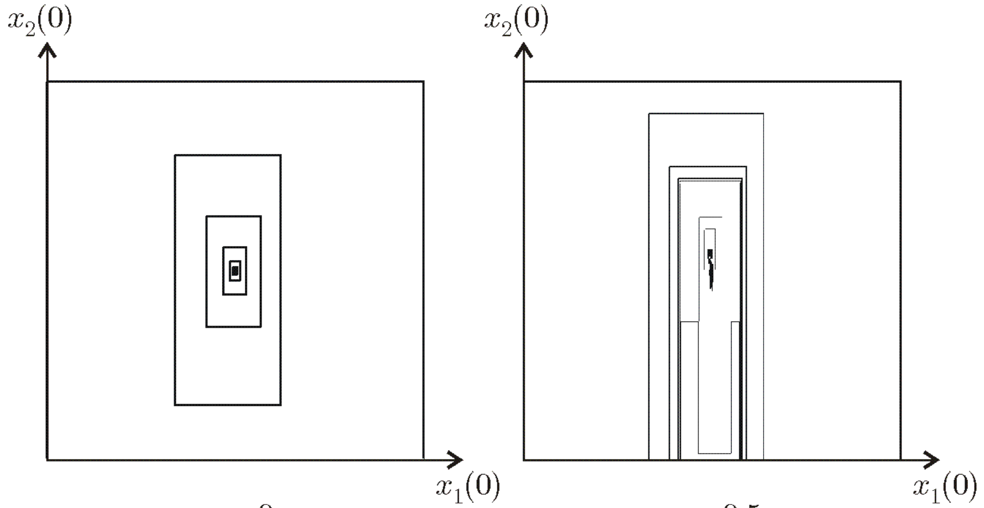
Algo ϕ(in: $x_1(0), x_2(0)$; out: $y(1), \ldots, y(15)$)
for $k := 1$ to 15,
$\quad z_1(k) := \exp(x_1(k-1))$;
$\quad z_2(k) = x_2(k-1) * z_1(k)$;
$\quad x_1(k) := 0.1 * x_1(k-1) + z_2(k)$;
$\quad z_3(k) := 0.1 * \mathrm{sqr}(x_2(k-1))$;
$\quad z_4(k) := z_3(k) + \sin(k)$;
$\quad x_2(k) := x_1(k-1) + z_4(k)$;
$\quad y(k) := x_2(k) / x_1(k)$.

Algo $\mathcal{C}_{\hat{\mathbb{X}}(0)}$(in: $[y(1)], \ldots, [\check{y}(15)]$; inout: $[x_1(0)], [x_2(0)]$)

for $k := 1$ to 15
  $[x_1(k)] := [-\infty, \infty]$; $[x_2(k)] := [-\infty, \infty]$;
  $[z_1(k)] := [-\infty, \infty]$; $[z_2(k)] := [-\infty, \infty]$;
  $[z_3(k)] := [-\infty, \infty]$; $[z_4(k)] := [-\infty, \infty]$;
do
  for $k := 1$ to 15,
    $[z_1(k)] := [z_1(k)] \cap \exp([x_1(k-1)])$;
    $[z_2(k)] := [z_2(k)] \cap ([x_2(k-1)] * [z_1(k)])$;
    $[x_1(k)] := [x_1(k)] \cap (0.1\ [x_1(k-1)] + [z_2(k)])$;
    $[z_3(k)] := [z_3(k)] \cap (0.1\ \mathrm{sqr}([x_2(k-1)]))$;
    $[z_4(k)] := [z_4(k)] \cap ([z_3(k)] + \sin(k))$;
    $[x_2(k)] := [x_2(k)] \cap ([x_1(k-1)] + [z_4(k)])$;
    $[y(k)] := [y(k)] \cap ([x_2(k)]/[x_1(k)])$;
  for $k := 15$ down to 1,
    $[x_2(k)] := [x_2(k)] \cap ([y(k)] * [x_1(k)])$;
    $[x_1(k)] := [x_1(k)] \cap ([x_2(k)]/[y(k)])$;
    $[x_1(k-1)] := [x_1(k-1)] \cap ([x_2(k)] - [z_4(k)])$;
    $[z_4(k)] := [z_4(k)] \cap ([x_2(k)] - [x_1(k-1)])$;
    $[z_3(k)] := [z_3(k)] \cap ([z_4(k)] - \sin(k))$;
    $[x_2(k-1)] := [x_2(k-1)] \cap 0.1\ \sqrt{[z_3(k)]}$);
    $[x_1(k-1)] := [x_1(k-1)] \cap 10\ ([x_1(k)] - [z_2(k)])$;
    $[z_2(k)] := [z_2(k)] \cap ([x_1(k)] - 0.1 * [x_1(k-1)])$;
    $[x_2(k-1)] := [x_2(k-1)] \cap ([z_2(k)]/[z_1(k)])$;
    $[z_1(k)] := [z_1(k)] \cap ([z_2(k)]/[x_2(k-1)])$;
    $[x_1(k-1)] := [x_1(k-1)] \cap \log([z_1(k)])$;
while contraction is significant.

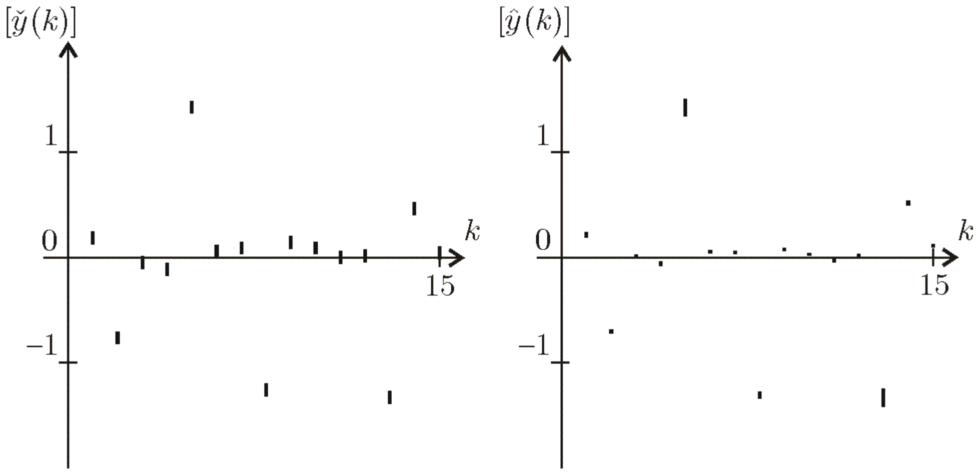
The prior domains for the initial state vector are

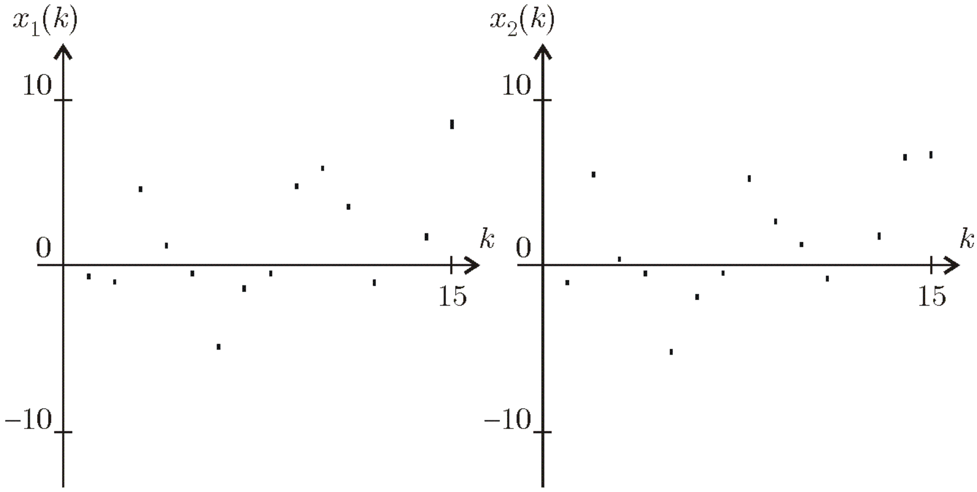
$$[x_1(0)] = [-1.2, -0.8], \qquad [x_2(0)] = [-0.2, 0.2].$$



$e = 0$ $e = 0.5$

For $e = 0.5$:

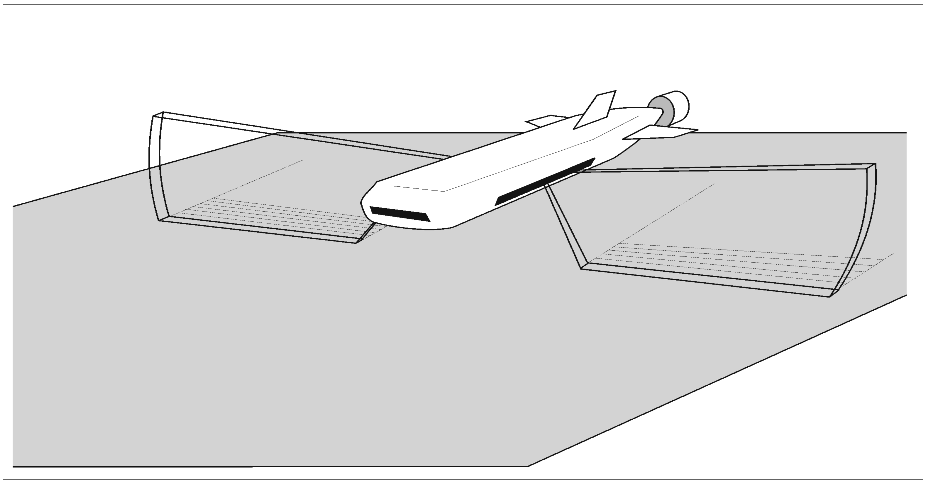




$x_1(k)$
10
0
−10
k
15
$x_2(k)$
10
0
−10
k
15

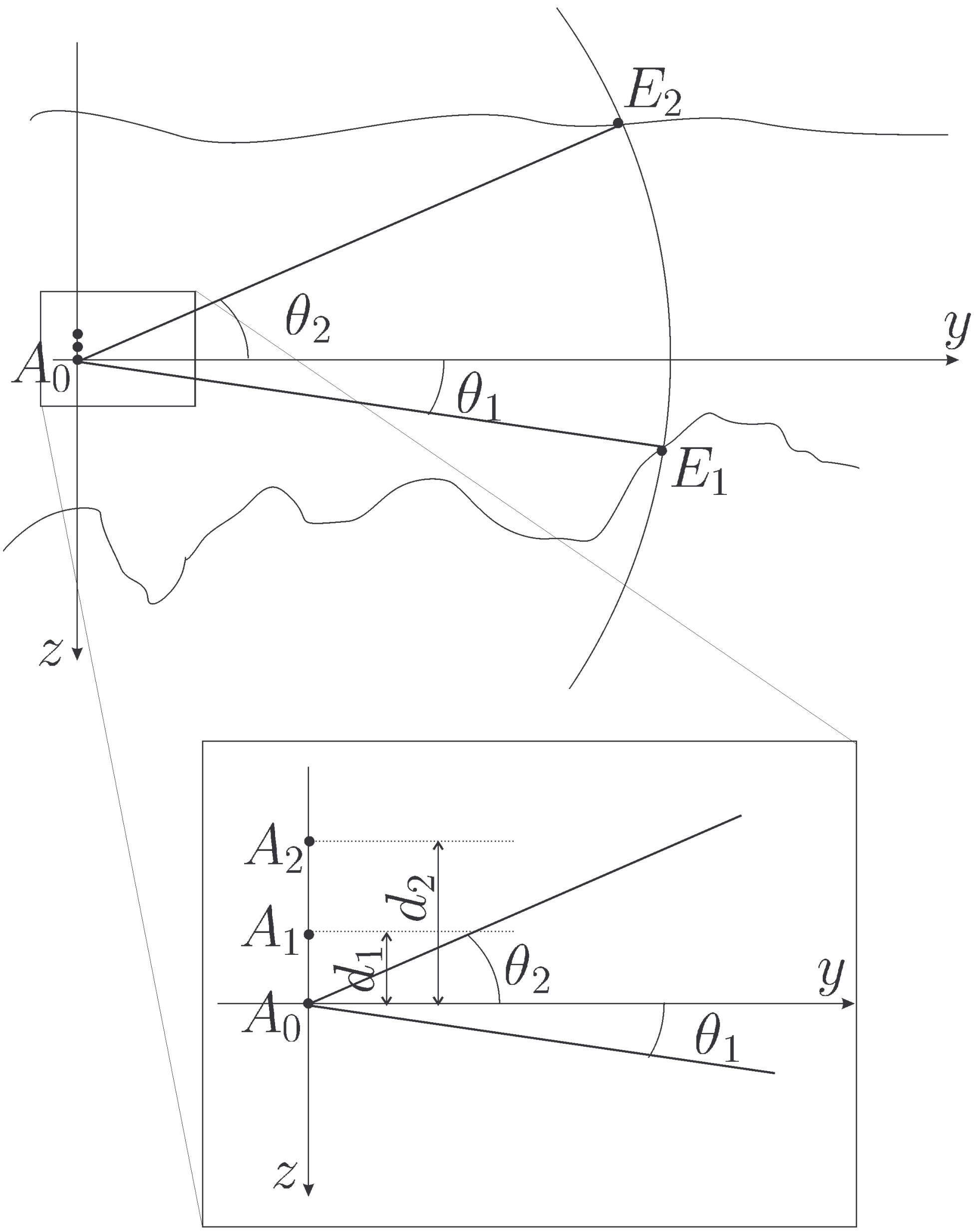
Estimation of the bathymetry of the ocean (Collaboration with M. Legris)

Consider an autonomous underwater vehicle (AUV) with two lateral sonars.
At each sample, the sonar measures an echo signal.



The sonar has three antennas A_0, A_1, A_2. The wave emitted by A_0 is $s(t) = e^{2\pi f_0 t}$.

We have $c = 1500$ ms^{-1}, $\lambda = 3$ mm, $f_0 = 455$ kHz.



E_2
θ_2
y
A_0
θ_1
E_1
z
A_2
d_2
A_1
d_1
θ_2
y
A_0
θ_1
z

The sensors $A_m, m = 0, 1, 2$ receive the signal

$$\widehat{s}_m(t) = \sum_{n=1}^{n_{\max}} \alpha_n e^{j2\pi f_0 t + j2\pi f_0 \frac{r}{c} + j\varphi_n} e^{j2\pi f_0 \frac{d_m \sin\theta_n}{c}},$$

where $n_{\max}$ is the number of existing obstacles at a distance $r = ct$ from A_0.

$\varphi_n(r)$ results from the superposition of microscopic reflections.

$d_0 = 0, d_1 = 4.94$mm, $d_2 = 13.187$ mm.

Fresnel transformation:

$$
\begin{aligned}
s_m(r) &= \sum_{n=1}^{n_{\mathsf{max}}} \alpha_n e^{j2\pi f_0 \frac{r}{c} + j\varphi_n} e^{j2\pi f_0 \frac{dm \sin\theta_n}{c}} \\
&= \sum_{n=1}^{n_{\mathsf{max}}} \alpha_n e^{j\rho_n} e^{j2\pi f_0 \frac{dm \sin\theta_n}{c}},
\end{aligned}
$$

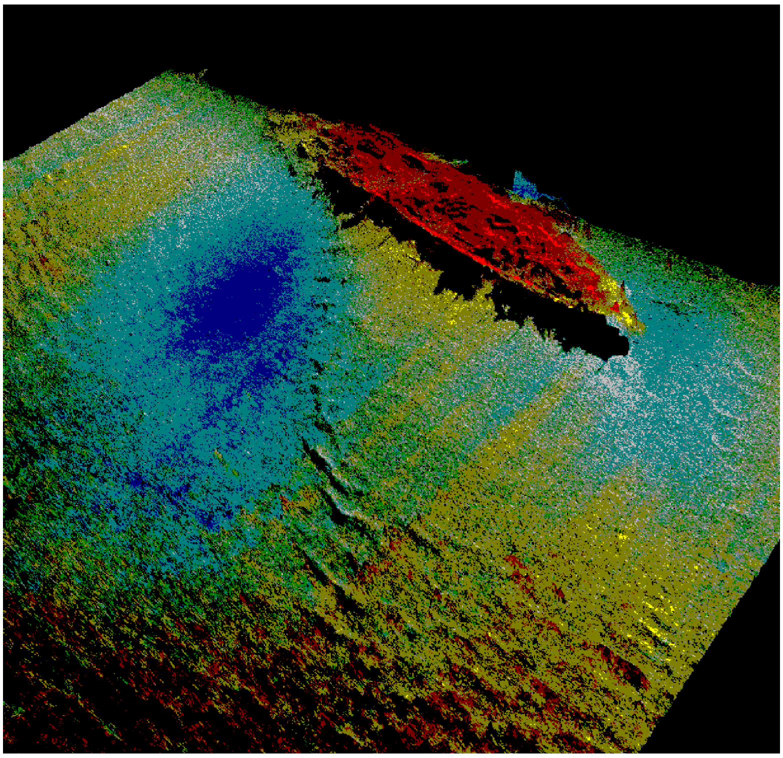
where $\rho_n(r) = 2\pi f_0 \frac{r}{c} + \varphi_n(r)$.

For each r, we have 6 equations with $3n_{\mathsf{max}}$ unknowns (the α_n's, the θ_n's and the ρ_n's).

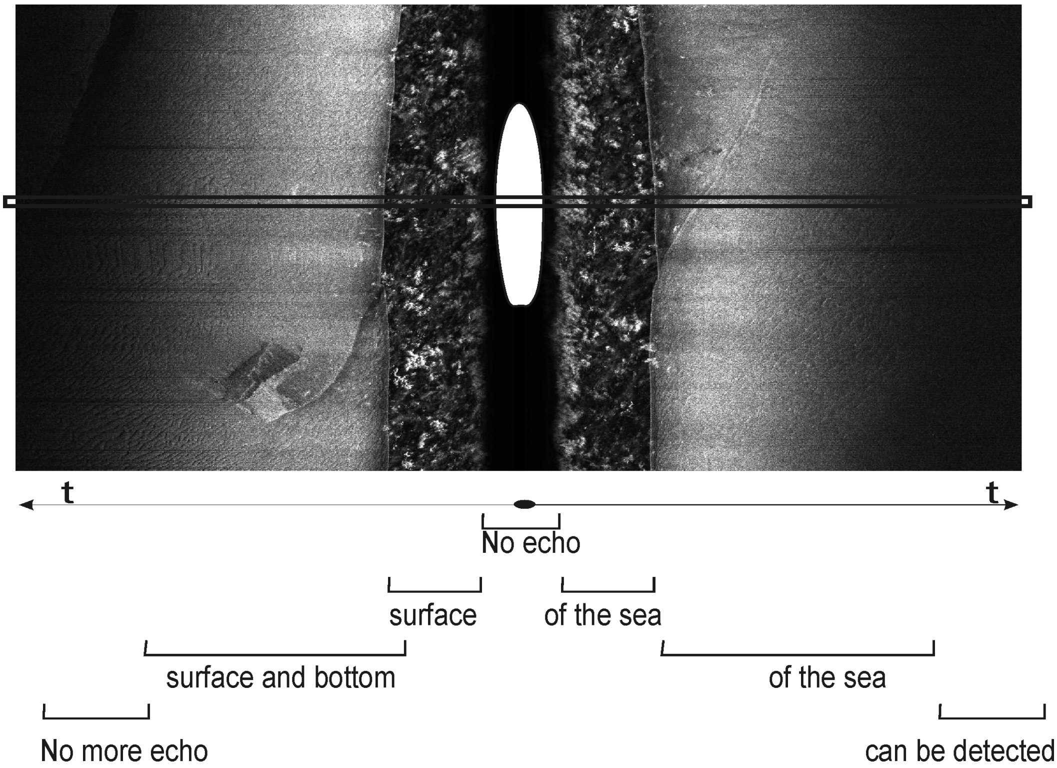
When $n_{\text{max}} = 1$, an analytical resolution can be obtained:

$$
\begin{aligned}
\text{Re}(s_m) &= \alpha_1 \cos\left(\rho_1 + 2\pi f_0 \frac{d_m \sin\theta_1}{c}\right) \\
\text{Im}(s_m) &= \alpha_1 \sin\left(\rho_1 + 2\pi f_0 \frac{d_m \sin\theta_1}{c}\right)
\end{aligned}
$$

From $s_m(r)$ for $m = 1, 2$ we obtain $\alpha_1(r), \rho_1(r), \theta_1(r)$.



In practice, more that one obstacle should be considered.
Data given by the GESMA (Groupe d'Etudes Sous Marines de l'Atlantique).
Sonar: Klein 5000 with $f_0 = 455$ kHz.

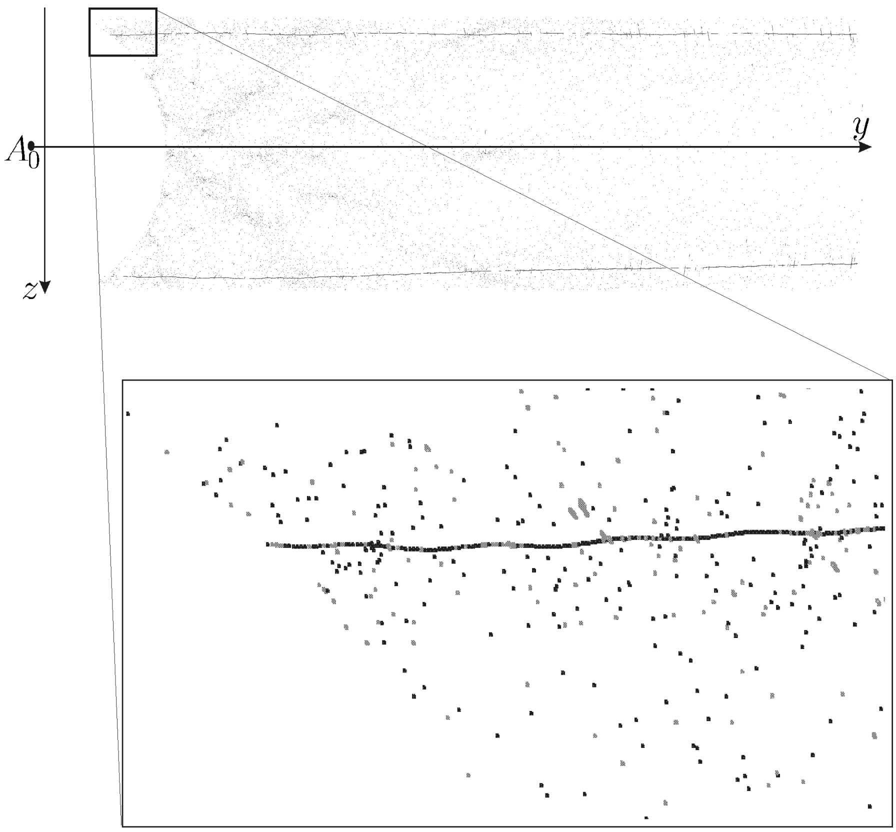

t
t
No echo
surface
of the sea
surface and bottom
of the sea
No more echo
can be detected

For $n_{\text{max}} = 2$, the equations to be solved for each

$$r \in \{15m, 15.03m, 15.06m, \ldots, 150m\}$$

are:

$$
\begin{aligned}
s_0^{\mathrm{Re}} &= \alpha_1 \cos \rho_1 + \alpha_2 \cos \rho_2 \\
s_0^{\mathrm{Im}} &= \alpha_1 \sin \rho_1 + \alpha_2 \sin \rho_2 \\
s_1^{\mathrm{Re}} &= \alpha_1 \cos \left(\rho_1 + 2\pi f_0 \frac{d_1 \sin \theta_1}{c} \right) \\
&\quad + \alpha_2 \cos \left(\rho_2 + 2\pi f_0 \frac{d_1 \sin \theta_2}{c} \right) \\
s_1^{\mathrm{Im}} &= \alpha_1 \sin \left(\rho_1 + 2\pi f_0 \frac{d_1 \sin \theta_1}{c} \right) \\
&\quad + \alpha_2 \sin \left(\rho_2 + 2\pi f_0 \frac{d_1 \sin \theta_2}{c} \right) \\
s_2^{\mathrm{Re}} &= \alpha_1 \cos \left(\rho_1 + 2\pi f_0 \frac{d_2 \sin \theta_1}{c} \right) \\
&\quad + \alpha_2 \cos \left(\rho_2 + 2\pi f_0 \frac{d_2 \sin \theta_2}{c} \right) \\
s_2^{\mathrm{Im}} &= \alpha_1 \sin \left(\rho_1 + 2\pi f_0 \frac{d_2 \sin \theta_1}{c} \right) \\
&\quad + \alpha_2 \sin \left(\rho_2 + 2\pi f_0 \frac{d_2 \sin \theta_2}{c} \right).
\end{aligned}
$$



This estimation amounts to solving 4500 systems of 6 nonlinear equations with 6 unknowns ($\Rightarrow$ 1 hour).

Set $\mathbf{y} = \left(s_0^{\mathsf{Re}}, s_0^{\mathsf{Im}}, s_1^{\mathsf{Re}}, s_1^{\mathsf{Im}}, s_2^{\mathsf{Re}}, s_2^{\mathsf{Im}}\right)$, and $\mathbf{x} = (\theta_1, \theta_2)$, we have the state equations

$$\begin{aligned}
\mathbf{x}(r+dr) &= \mathbf{x}(r) + \mathbf{b}_{\mathbf{x}}(r) \\
\mathbf{y}(r) &= \alpha_1 \begin{pmatrix} \cos\rho_1 \\ \sin\rho_1 \\ \cos\left(\rho_1 + \frac{2\pi f_0 d_1 \sin x_1}{c}\right) \\ \sin\left(\rho_1 + \frac{2\pi f_0 d_1 \sin x_1}{c}\right) \\ \cos\left(\rho_1 + \frac{2\pi f_0 d_2 \sin x_1}{c}\right) \\ \alpha_1 \sin\left(\rho_1 + \frac{2\pi f_0 d_2 \sin x_1}{c}\right) \end{pmatrix} \\
&\quad + \alpha_2 \begin{pmatrix} \cos\rho_2 \\ \sin\rho_2 \\ \cos\left(\rho_2 + \frac{2\pi f_0 d_1 \sin x_2}{c}\right) \\ \sin\left(\rho_2 + \frac{2\pi f_0 d_1 \sin x_2}{c}\right) \\ \cos\left(\rho_2 + \frac{2\pi f_0 d_2 \sin x_2}{c}\right) \\ \sin\left(\rho_2 + \frac{2\pi f_0 d_2 \sin x_2}{c}\right) \end{pmatrix}
\end{aligned}$$

Robust stability of linear systems

(Luc Jaulin, Thursday, 10h00-11h00).

Stability domain

The *stability domain* of

$$P(s, \mathbf{p}) = s^n + a_{n-1}(\mathbf{p})s^{n-1} + \ldots + a_1(\mathbf{p})s + a_0(\mathbf{p})$$

is the set of all $\mathbf{p}$ such that $P(s, \mathbf{p})$ is stable.

The Routh table of

$$P(s,\mathbf{p}) = s^3 + (p_1 + p_2 + 2)s^2 + (p_1 + p_2 + 2)s + 2p_1p_2 + 6p_1 + 6p_2 + 2 + \sigma^2,$$

is given by

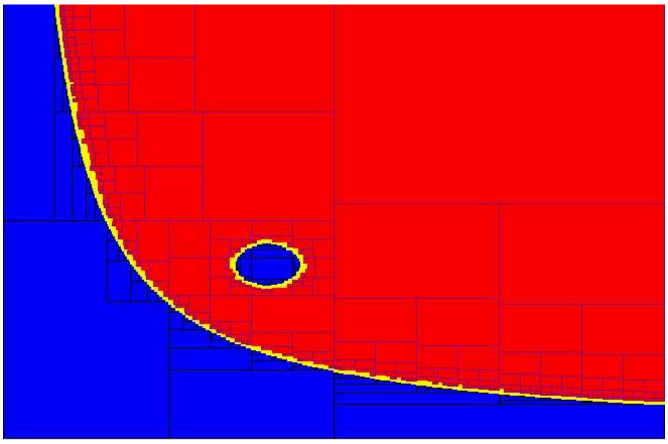
1	$p_1 + p_2 + 2$
$p_1 + p_2 + 2$	$2p_1p_2 + 6p_1 + 6p_2 + 2 + \sigma^2$
$\frac{(p_1-1)^2+(p_2-1)^2-\sigma^2}{p_1+p_2+2}$	0
$2(p_1 + 3)(p_2 + 3) - 16 + \sigma^2$	0

Its stability domain is thus

$$\mathbb{S}_{\mathsf{p}} \stackrel{\text{def}}{=} \{\mathbf{p} \in \mathbb{R}^n \mid \mathbf{r}(\mathbf{p}) > \mathbf{0}\} = \mathbf{r}^{-1}\left(]0, +\infty[^{\times n}\right).$$

where

$$\mathbf{r}(\mathbf{p}) = \begin{pmatrix} p_1 + p_2 + 2 \\ (p_1 - 1)^2 + (p_2 - 1)^2 - \sigma^2 \\ 2(p_1 + 3)(p_2 + 3) - 16 + \sigma^2 \end{pmatrix}.$$



Robust stability of a controlled motorbike (Collaboration with M. Christie, L. Granvilliers, X. Baguenard)

A CSP is *infallible* if any arbitrary instantiation of the variables is a solution.

Consider the CSP

$$\begin{array}{rcl} \mathcal{V} & = & \{x, y\} \\ \mathcal{D} & = & \{[x],[y]\} \\ \mathcal{C} & = & \{\ f(x,y) \leq 0,\ g(x,y) \leq 0\}\,. \end{array}$$

The CSP is infallible if

$$\begin{array}{rl} & \forall x \in [x], \forall y \in [y],\ f(x,y) \leq 0 \text{ and } g(x,y) \leq 0, \\ \Leftrightarrow & \{(x,y) \in [x] \times [y] \mid f(x,y) > 0 \text{ or } g(x,y) > 0\} = \emptyset \\ \Leftrightarrow & \{(x,y) \in [x] \times [y] \mid \max\left(f(x,y), g(x,y)\right) > 0\} = \emptyset. \end{array}$$

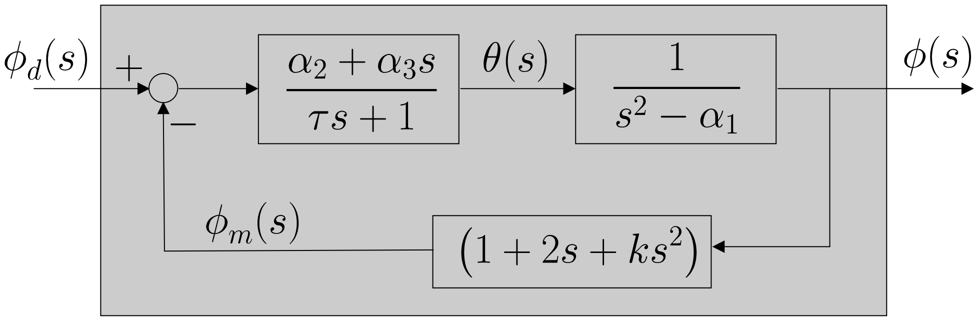
Consider a motorbike with a speed of 1m/s.

Angle of the handlebars: θ.

Rolling angle: ϕ

Wanted rolling angle: ϕ_d

Measured rolling angle: ϕ_m.



The input-output relation of the closed-loop system is :

$$\phi(s) = \frac{\alpha_2 + \alpha_3 s}{\left(s^2 - \alpha_1\right)(\tau s + 1) + (\alpha_2 + \alpha_3 s)\left(1 + 2s + ks^2\right)}\phi_d(s).$$

Its characteristic polynomial is thus

$$\begin{aligned} P(s) &= \left(s^2 - \alpha_1\right)(\tau s + 1) + (\alpha_2 + \alpha_3 s)\left(1 + 2s + ks^2\right) \\ &= a_3 s^3 + a_2 s^2 + a_1 s + a_0, \end{aligned}$$

with

$$\begin{array}{cc} a_3 = \tau + \alpha_3 k & a_2 = \alpha_2 k + 2\alpha_3 + 1 \\ a_1 = \alpha_3 - \alpha_1 \tau + 2\alpha_2 & a_0 = -\alpha_1 + \alpha_2. \end{array}$$

The Routh table is :

a_3	a_1
a_2	a_0
$\frac{a_2a_1-a_3a_0}{a_2}$	0
a_0	0

The closed-loop system is stable if $a_3, a_2, \frac{a_2a_1-a_3a_0}{a_2}$ and a_0 have the same sign.

Assume that it is known that

$$\alpha_1 \in [8.8; 9.2] \qquad \alpha_2 \in [2.8; 3.2]$$
$$\alpha_3 \in [0.8; 1.2] \qquad \tau \in [1.8; 2.2]$$
$$k \in [-3.2; -2.8].$$

The system is robustly stable if.,

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

$a_3,\ a_2,\ \frac{a_2 a_1 - a_3 a_0}{a_2}$ and a_0 have the same sign.

Now, we have the equivalence

$$\begin{aligned} & b_1,\ b_2,\ b_3 \text{ and } b_4 \text{ have the same sign} \\ \Leftrightarrow \quad & \max\left(\min\left(b_1, b_2, b_3, b_4\right), -\max\left(b_1, b_2, b_3, b_4\right)\right) > 0 \end{aligned}$$

The robust stability condition amounts to proving that

$$\begin{aligned} \exists \alpha_1 \in \; & [\alpha_1], \exists \alpha_2 \in [\alpha_2], \exists \alpha_3 \in [\alpha_3], \exists \tau \in [\tau], \exists k \in [k], \\ & \max(\quad \min\left(a_3, a_2, \tfrac{a_2 a_1 - a_3 a_0}{a_2}, a_0\right), \\ & \qquad - \max(a_3, a_2, \tfrac{a_2 a_1 - a_3 a_0}{a_2}, a_0) \quad) \leq 0 \end{aligned}$$

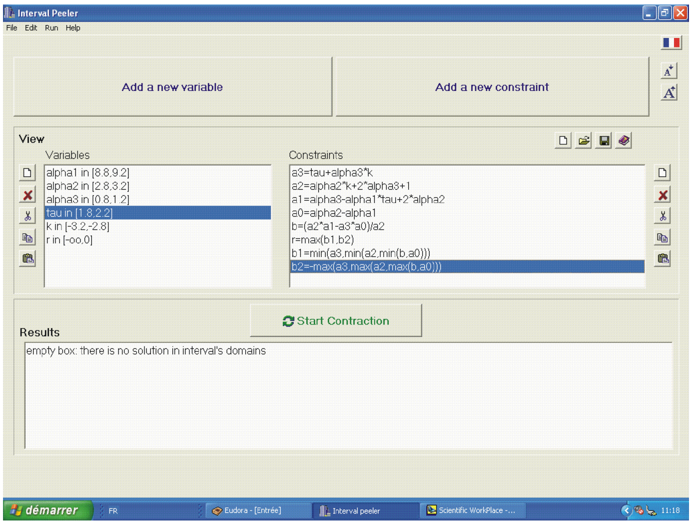
is false,...

i.e., that the CSP

$$\begin{array}{rcl}
\mathcal{V} & = & \{a_0, a_1, a_2, a_3, \alpha_1, \alpha_2, \alpha_3, \tau, k\}, \\
\mathcal{D} & = & \{[\alpha_0], [\alpha_1], [\alpha_2], [\alpha_3], [\alpha_2], [\alpha_3], [\tau], [k]\}, \\
\mathcal{C} & = & \left\{ \begin{array}{l}
a_3 = \tau + \alpha_3 k \ ; \ a_2 = \alpha_2 k + 2\alpha_3 + 1 \ ; \\
a_1 = \alpha_3 - \alpha_1 \tau + 2\alpha_2, \\
a_0 = -\alpha_1 + \alpha_2 \ ; \\
m_1 = \min\left(a_3, a_2, \frac{a_2 a_1 - a_3 a_0}{a_2}, a_0\right) ; \\
m_2 = \max\left(a_3, a_2, \frac{a_2 a_1 - a_3 a_0}{a_2}, a_0\right) \\
\max\left(m_1, -m_2\right) \leq 0.
\end{array} \right\}
\end{array}$$

has no solution.

This task has been performed using IntervalPeeler



Analysis of a time-delay system (Collaboration with M. Dao, M. Di Loreto, J.F. Lafay and J.J. Loiseau)

Consider the linear system

$$\ddot{y}(t) - \ddot{y}(t-1) + 2\dot{y}(t) - \dot{y}(t-1) + y(t) = u(t).$$

The Laplace transform of this equation is

$$s^2 y(s) - s^2 e^{-s} y(s) + 2sy(s) - se^{-s} y(s) + y(s) = u(s)$$

Its transfer function is

$$\begin{aligned} H(s) &= \frac{y(s)}{u(s)} = \frac{1}{s^2 - s^2 e^{-s} + 2s - se^{-s} y(s) + 1} \\ &= \frac{1}{(s+1)\left(s(1-e^{-s}) + 1\right)}. \end{aligned}$$

Its magnitude function is

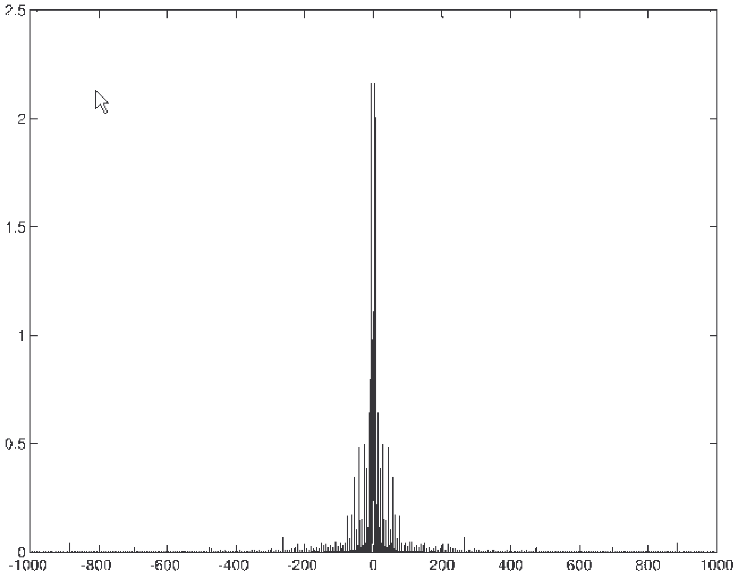
$$\begin{aligned} G(\omega) &= |H(j\omega)| \\ &= \frac{1}{A}\frac{1}{\sqrt{(1-\omega\sin\omega)^2+\omega^2(1-\cos\omega))^2}}. \end{aligned}$$

The Bode diagram is

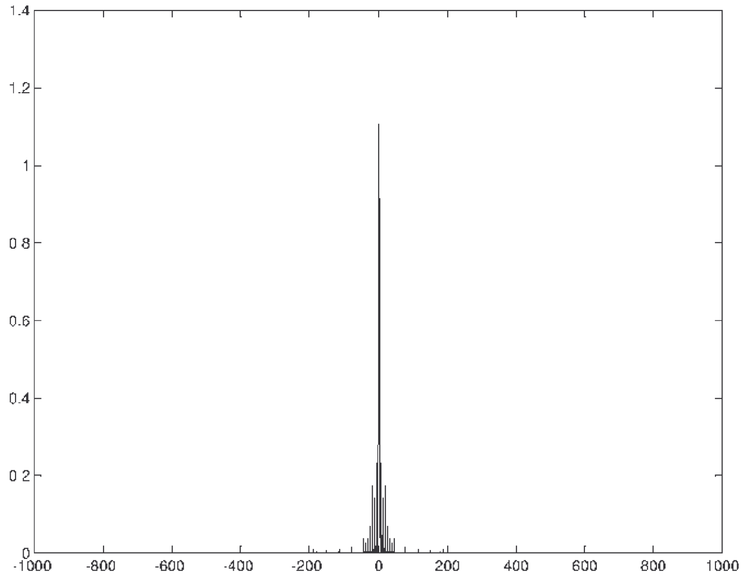
$$\mathbb{S} = \left\{(\omega, h) \in \mathbb{R}^2 | G(\omega) = h\right\}.$$

The Bode diagram has picks every 2π.

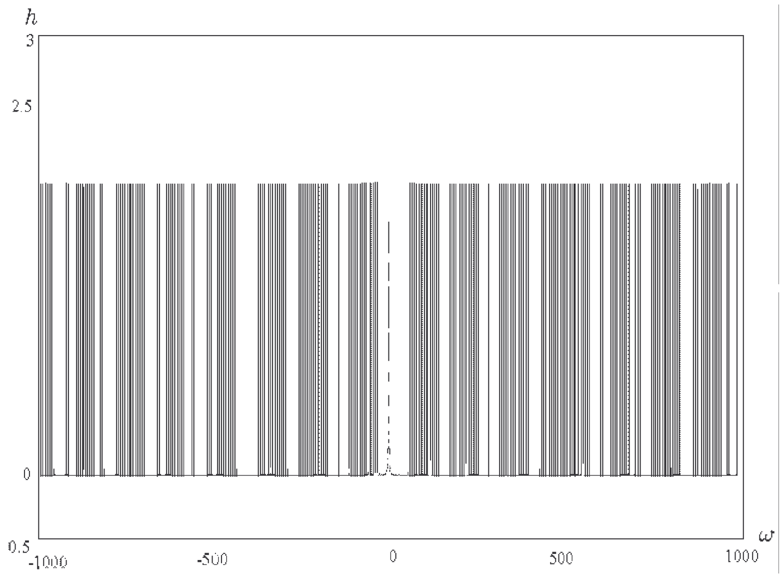
Matlab has some difficulties, even for a very high precision.



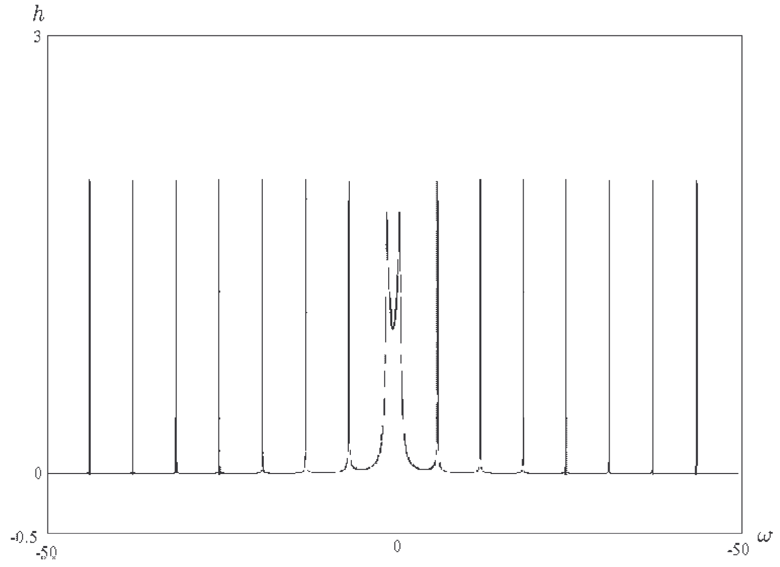
Bode diagram $h = G(\omega)$ with Matlab with
$\Delta\omega = 0.1 rad.s^{-1}$



Bode diagram $h = G(\omega)$ with Matlab with
$\Delta\omega = 0.001 rad.s^{-1}$



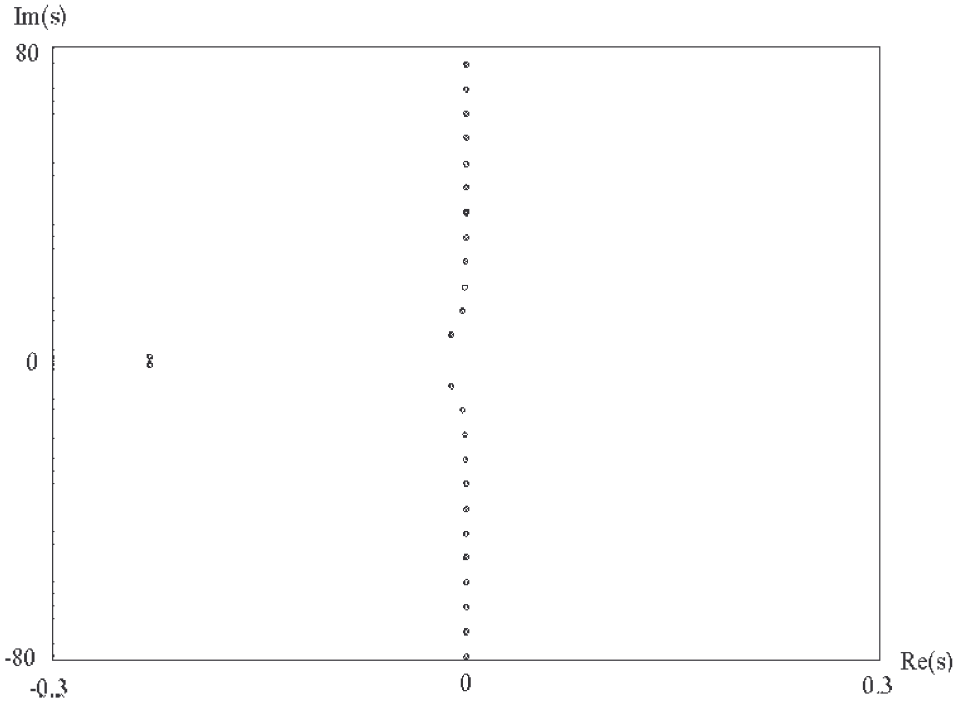
Bode diagram obtained by Proj2d for
$\omega \in [-1000, 1000]$



Bode diagram obtained by Proj2d for $\omega \in [-50, 50]$

The set of all feasible roots (or *root locus*) of the system is given by

$$\begin{aligned}
\mathbb{S} &= \left\{ s \in \mathbb{C} | \, (s+1) \left(s(1-e^{-s}) + 1 \right) = 0 \right\} \\
&= \left\{ x + jy \in \mathbb{C} | \begin{pmatrix} x - (x \cos y + y \sin y) e^{-x} + 1 \\ x + (x \sin y - y \cos y) e^{-x} \end{pmatrix} = \mathbf{0} \right\}.
\end{aligned}$$

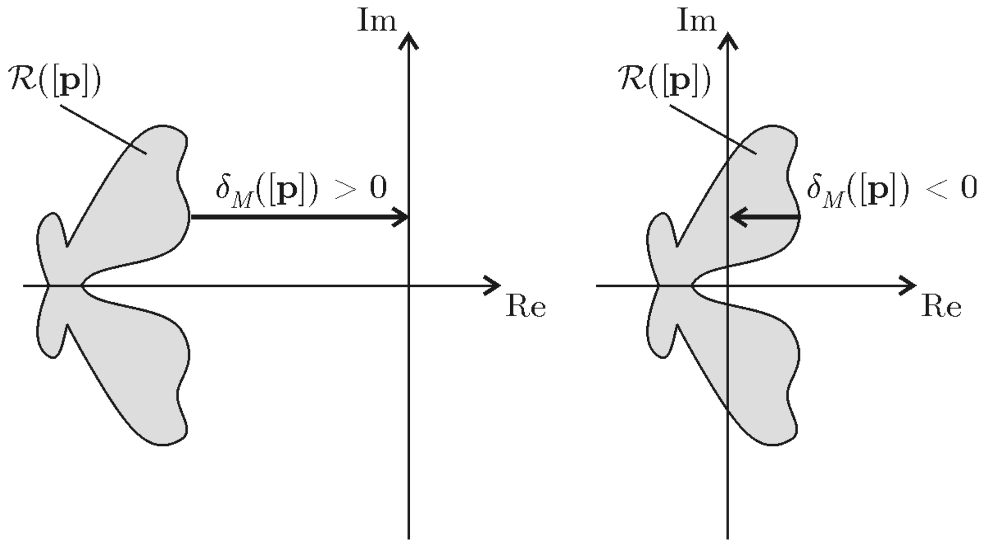


Im(s)
80
0
-80
-0.3
0
0.3
Re(s)

The *robust stability degree* $\delta_M([\mathbf{p}])$ is

$$\delta_M([\mathbf{p}]) = \min_{\mathbf{p}\in[\mathbf{p}]} \max_{\mathbf{r}(\mathbf{p},\delta)\geqslant\mathbf{0}} \delta.$$

If $\delta_M([\mathbf{p}]) > 0$, all roots of $\mathbf{\Sigma}(\mathbf{p})$ are in $\mathbb{C}^-$ and $\mathbf{\Sigma}([\mathbf{p}])$ is robustly stable.



Consider the uncertain system

$$\dot{\mathbf{x}} = \begin{pmatrix} \dfrac{p_2}{1+p_2} & 2 \\ \dfrac{p_2}{1+p_1} & \dfrac{p_1}{1+p_2^2} \end{pmatrix} .\mathbf{x}.$$

For $[\mathbf{p}] = [1, 2] \times [0, 0.5]$, the interval algorithm Pert-Min finds that the robust stability degree satisfies

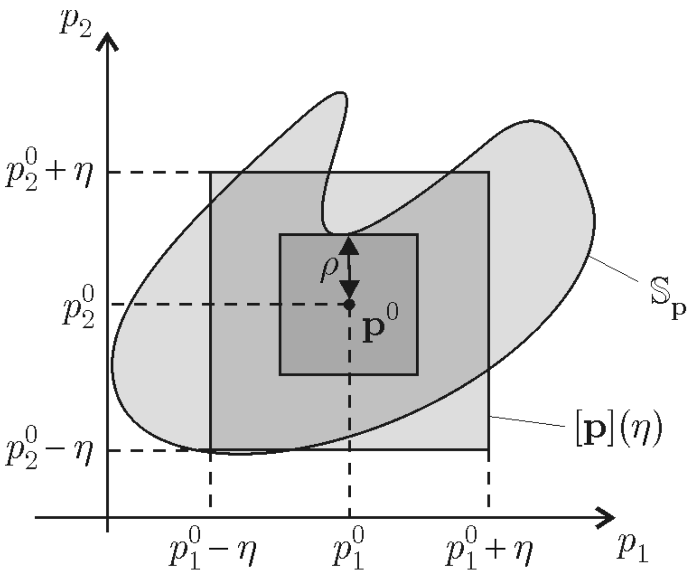
$$-2.01590 \leqslant \delta_M([\mathbf{p}]) \leqslant -2.01451.$$

The system is thus not robustly stable.

Stability radius

The *stability radius* of $\mathbf{\Sigma}(\mathbf{p})$ at $\mathbf{p}^0$ is

$$\begin{aligned}\rho &\stackrel{\text{def}}{=} \sup\{\eta \geqslant 0 \mid \mathbf{\Sigma}(\mathbf{p}) \text{ is stable for all } \mathbf{p} \in [\mathbf{p}](\eta)\}\,,\\ &= \min\{\eta \geqslant 0 \mid \mathbf{\Sigma}(\mathbf{p}) \text{ is unstable for one } \mathbf{p} \in [\mathbf{p}](\eta)\}\,,\end{aligned}$$



Since

$$\mathbf{p} \in [\mathbf{p}](\eta) \Leftrightarrow \forall j \in \{1, \ldots, n\}, (p_j^0 - \eta \leqslant p_j \leqslant p_j^0 + \eta)$$

and since

$$\begin{aligned} \mathbf{\Sigma}(\mathbf{p}) \text{ is unstable} \quad & \Leftrightarrow \quad \exists i \text{ such that } r_i(\mathbf{p}) \leqslant 0 \\ & \Leftrightarrow \quad (r_1(\mathbf{p}) \leqslant 0) \vee \cdots \vee (r_n(\mathbf{p}) \leqslant 0) \\ & \Leftrightarrow \quad \max\left(r_1(\mathbf{p})\right), \ldots, r_n(\mathbf{p})) \leqslant 0, \end{aligned}$$

the stability radius can also be defined as

$$\left\{ \begin{array}{l} \rho = \min\limits_{\eta \geqslant 0} \eta, \\ \text{s.t.} \left\{ \begin{array}{l} \exists \mathbf{p}, \max\left(r_1(\mathbf{p})\right), \ldots, r_n(\mathbf{p})) \leqslant 0 \\ \quad \wedge \left(\forall j \in \{1, \ldots, n\}, \left\{ \begin{array}{l} p_j^0 - \eta - p_j \leqslant 0 \\ -p_j^0 - \eta + p_j \leqslant 0 \end{array} \right. \right) \end{array} \right. \end{array} \right.$$

Example 1: Consider the polynomial

$$P(s, \mathbf{p}) = s^3 + (p_1 + p_2 + 2)s^2 + (p_1 + p_2 + 2)s + 2p_1p_2 + 6p_1 + 6p_2 + 2 + \sigma^2.$$

For $\mathbf{p}^0 = (1.4, 0.85)^\mathsf{T}$, we get

σ	10^{-1}	10^{-3}	10^{-5}	10^{-7}
Computing time (s)	0.44	0.55	0.44	0.49
Solution boxes	1	5	1	2
Stability radius	0.2727	0.3627	0.3636	0.3636

Example 2: Consider the polynomial

$$P(s, \mathbf{p}) = s^3 + a_2(\mathbf{p})s^2 + a_1(\mathbf{p})s + a_0(\mathbf{p}),$$

with

$$\begin{aligned} a_0(\mathbf{p}) &= \sin(p_2)e^{p_2} + p_1p_2 - 1, \\ a_1(\mathbf{p}) &= 2p_1 + 0.2p_1e^{p_2}, \\ a_2(\mathbf{p}) &= p_1 + p_2 + 4. \end{aligned}$$

We get $\rho \simeq 2.025$ at $\mathbf{p}^0 = (1.5, 1.5)^\mathsf{T}$.

Nonlinear control of a sailboat

(Luc Jaulin, Thursday, 11h15-11h45).

Projection of an equality.

Consider the set

$$\mathbb{S} \overset{\text{def}}{=} \{\mathbf{p} \in \mathbf{P} \mid \exists \mathbf{q} \in \mathbf{Q}, f(\mathbf{p}, \mathbf{q}) = 0\}.$$

where $\mathbf{P}$ and $\mathbf{Q}$ are boxes and f is continuous.

Since $\mathbf{Q}$ is a connected set and f is continuous, we have

$$\mathbb{S} = \{\mathbf{p} \in \mathbf{P} \mid \exists \mathbf{q}_1 \in \mathbf{Q}, f(\mathbf{p}, \mathbf{q}_1) \leq 0, \exists \mathbf{q}_2 \in \mathbf{Q}, f(\mathbf{p}, \mathbf{q}_2) \geq 0\}.$$

i.e.,

$$\begin{array}{rcl} \mathbb{S} & = & \{\mathbf{p} \in \mathbf{P} \mid (\exists \mathbf{q}_1 \in \mathbf{Q}, f(\mathbf{p}, \mathbf{q}_1) \leq 0)\} \\ & & \cap \{\mathbf{p} \in \mathbf{P} \mid (\exists \mathbf{q}_2 \in \mathbf{Q}, f(\mathbf{p}, \mathbf{q}_2) \geq 0)\}. \end{array}$$

or equivalently

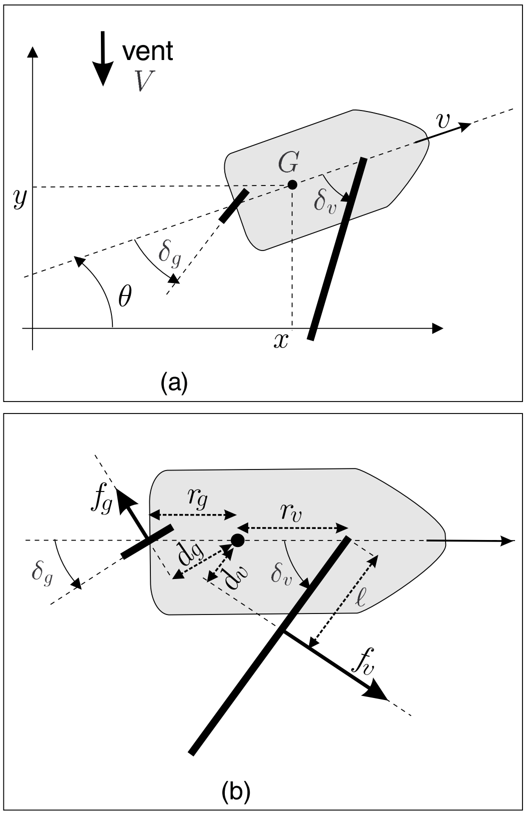
$$\mathbb{S} = \left\{\mathbf{p} \in \mathbf{P} \mid \exists (\mathbf{q}_1, \mathbf{q}_2) \in \mathbf{Q}^2, f(\mathbf{p}, \mathbf{q}_1) \leq 0 \text{ and } f(\mathbf{p}, \mathbf{q}_2) \geq 0\right\}$$

Polar speed diagram of a sailboat (with M. Dao, M. Lhommeau, P. Herrero, J. Vehi and M. Sainz)

State equations of a sailboat:

$$\left\{\begin{array}{ccc}\dot{x} & = & v\cos\theta,\\ \dot{y} & = & v\sin\theta-\beta V,\\ \dot{\theta} & = & \omega,\\ \dot{\delta}_s & = & u_1,\\ \dot{\delta}_r & = & u_2,\\ \dot{v} & = & \frac{f_s\sin\delta_s-f_r\sin\delta_r-\alpha_f v}{m},\\ \dot{\omega} & = & \frac{(\ell-r_s\cos\delta_s)f_s-r_r\cos\delta_r f_r-\alpha_\theta\omega}{J},\\ f_s & = & \alpha_s\left(V\cos\left(\theta+\delta_s\right)-v\sin\delta_s\right),\\ f_r & = & \alpha_r v\sin\delta_r.\end{array}\right.$$

The state vector $\mathbf{x}=(x,y,\theta,\delta_s,\delta_r,v,\omega)^\mathsf{T}\in\mathbb{R}^7$.The inputs u_1 and u_2 of the system are the derivatives of the angles δ_s and δ_r.


vent
V
v
G
y
δ_v
δ_g
θ
x
(a)
f_g
r_g
r_v
d_g
d_v
δ_v
δ_g
ℓ
f_v
(b)

The *polar speed diagram* is the set $\mathbb{S}$ of all feasible (θ, v).

$$\dot{\theta} = 0, \dot{\delta}_s = 0, \dot{\delta}_r = 0, \dot{v} = 0, \dot{\omega} = 0,$$

implies that

$$\begin{cases} 0 &= \frac{f_s \sin \delta_s - f_r \sin \delta_r - \alpha_f v}{m}, \\ 0 &= \frac{(\ell - r_s \cos \delta_s) f_s - r_r \cos \delta_r f_r}{J}, \\ f_s &= \alpha_s \left(V \cos\left(\theta + \delta_s\right) - v \sin \delta_s \right), \\ f_r &= \alpha_r v \sin \delta_r. \end{cases}$$

An elimination of f_s, f_r and δ_r yields

$$\Big(\big(\alpha_r + 2\alpha_f\big)\, v - 2\alpha_s V \cos(\theta + \delta_s) \sin \delta_s + 2\alpha_s v \sin^2 \delta_s\Big)^2 + \Big(\frac{2\alpha_s}{r_r}(\ell - r_s \cos \delta_s)(V \cos(\theta + \delta_s) - v \sin \delta_s)\Big)^2 - \alpha_r^2 v^2 = 0$$

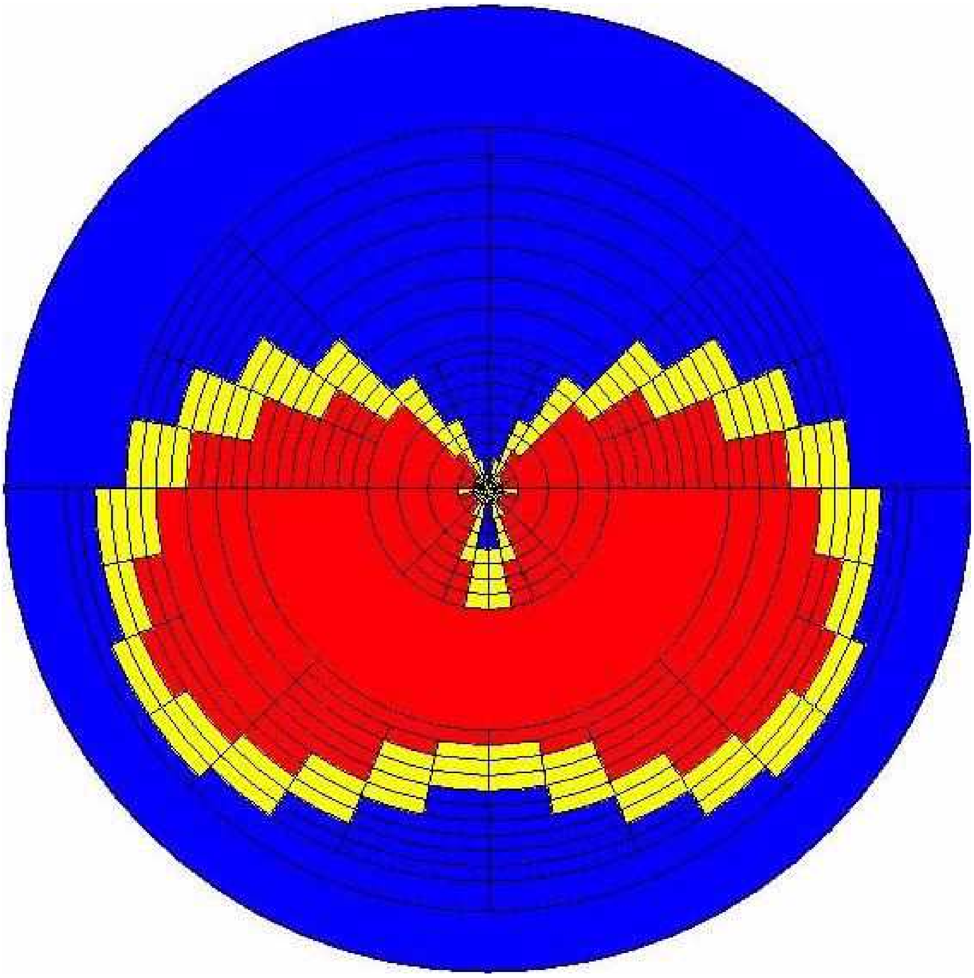
The polar speed diagram can thus be written as

$$\mathbb{S} = \left\{(\theta, v) | \exists \delta_s \in [-\frac{\pi}{2}, \frac{\pi}{2}] \mid f(\theta, v, \delta_s) = 0\right\}.$$

For the parameters

$$\begin{aligned} L &= 1, \alpha_f = 60, \alpha_\theta = 500, \alpha_s = 500, \\ \alpha_r &= 300, \beta = 0.05, r_s = 1, \\ r_r &= 2, V = 10, m = 1000, J = 2000, \end{aligned}$$

the polar speed diagram is given by



Feedback linearization

Consider a normalized version of the sailboat:

$$\left\{\begin{array}{ccc}\dot{\theta} & = & \omega, \\ \dot{\delta}_s & = & u_1, \\ \dot{\delta}_r & = & u_2, \\ \dot{v} & = & f_s \sin\delta_s - f_r \sin\delta_r - v, \\ \dot{\omega} & = & (1-\cos\delta_s)\, f_s - \cos\delta_r f_r - \omega, \\ f_s & = & \cos(\theta+\delta_s) - v\sin\delta_s, \\ f_r & = & v\sin\delta_r. \end{array}\right.$$

Denote by $\mathcal{F}(\mathbf{x},\mathbf{u})$ the set of all variables that are algebraic functions of $\mathbf{x}$ and $\mathbf{u}$. We have

$$\left(\dot{\theta}, \dot{\delta}_s, \dot{\delta}_r, \dot{v}, \dot{\omega}, f_s, f_r\right) \in \mathcal{F}(\mathbf{x},\mathbf{u}),$$

but $\ddot{\delta}_s = \dot{u}_1 \notin \mathcal{F}(\mathbf{x},\mathbf{u})$.

Since

$$\left\{\begin{array}{lcc} \ddot{\theta} & = & \dot{\omega}, \\ \ddot{v} & = & \dot{f}_s \sin\delta_s + f_s u_1 \cos\delta_s - \dot{f}_r \sin\delta_r - f_r u_2 \cos\delta_r - \\ \ddot{\omega} & = & u_1 \sin\delta_s f_s + (1-\cos\delta_s)\,\dot{f}_s + u_2 \sin\delta_r f_r - \cos\delta_r \dot{f}_r \\ \dddot{\theta} & = & \ddot{\omega} \\ \dot{f}_s & = & -(\omega+u_1)\sin(\theta+\delta_s) - \dot{v}\sin\delta_s - v u_1 \cos\delta_s \\ \dot{f}_r & = & \dot{v}\sin\delta_r + v u_2 \cos\delta_r. \end{array}\right.$$

we have

$$\left(\ddot{\theta}, \ddot{v}, \ddot{\omega}, \dot{f}_s, \dot{f}_r, \dddot{\theta}\right) \in \mathcal{F}(\mathbf{x}, \mathbf{u}).$$

Denote by $\mathcal{F}_u(\mathbf{x},\mathbf{u})$ the variables of $\mathcal{F}(\mathbf{x},\mathbf{u})$ which depend on $\mathbf{u}$. We have

$$\left(u_1, u_2, \dot{\delta}_s, \dot{\delta}_r, \ddot{v}, \ddot{\omega}, \dot{f}_s, \dot{f}_r, \dddot{\theta}\right) \in \mathcal{F}_u(\mathbf{x},\mathbf{u}).$$

but $\omega, \dot{\omega} \notin \mathcal{F}_u(\mathbf{x},\mathbf{u})$.

Take two state variables and store them into $\mathbf{y}$. For instance

$$\mathbf{y} = (\delta_s, \theta)^\mathsf{T}.$$

We have $\left(\dot{\delta}_s, \dddot{\theta}\right) \in \mathcal{F}_u(\mathbf{x},\mathbf{u})$:

$$\begin{aligned}
\begin{pmatrix} \dot{y}_1 \\ \dddot{y}_2 \end{pmatrix} &= \begin{pmatrix} \dot{\delta}_s \\ \dddot{\theta} \end{pmatrix} \\
&= \Bigg(\begin{matrix} 1 \\ f_s \sin\delta_s + (\cos\delta_s - 1)\left(v\cos\delta_s + \sin\left(\theta+\delta_s\right)\right) \end{matrix} \\
&\quad + \begin{pmatrix} 0 & 0 \\ 1-\cos\delta_s & -\cos\delta_r \end{pmatrix} \Bigg(\begin{matrix} -\omega\sin\left(\theta+\delta_s\right) - \dot{v} \\ \dot{v}\sin\delta_r \end{matrix} \\
&= \mathbf{A}(\mathbf{x})\mathbf{u} + \mathbf{b}(\mathbf{x}).
\end{aligned}$$

If we take

$$\mathbf{u} = \mathbf{A}^{-1}(\mathbf{x})\left(\mathbf{v} - \mathbf{b}(\mathbf{x})\right),$$

The closed loop system becomes linear:

$$\begin{cases} \ddot{\delta}_s & = & v_1, \\ \dddot{\theta} & = & v_2. \end{cases}$$

This linear and decoupled system should now be stabilized.

Linear control.

Denote by $\mathbf{w} = (w_1, w_2) = \left(\hat{\delta}_s, \hat{\theta}\right)$ the wanted values for $\mathbf{y} = (\delta_s, \theta)$. Classical PD^n controllers are given by :

$$\left\{ \begin{array}{lcl} v_1 & = & \alpha_{\mathsf{P}} \left(w_1 - \delta_s\right), \\ v_2 & = & \beta_{\mathsf{P}} \left(w_2 - \theta\right) + \beta_{\mathsf{D}} \left(\dot{w}_2 - \dot{\theta}\right) + \beta_{\mathsf{D}^2} \left(\ddot{w}_2 - \ddot{\theta}\right). \end{array} \right. \tag{1}$$

If w_2 is assumed to be constant, the closed loop system can be written:

$$\left\{ \begin{array}{lcl} \dot{\delta}_s & = & \alpha_{\mathsf{P}} \left(w_1 - \delta_s\right) \\ \dddot{\theta} & = & \beta_{\mathsf{P}} \left(w_2 - \theta\right) - \beta_{\mathsf{D}} \dot{\theta} - \beta_{\mathsf{D}^2} \ddot{\theta}. \end{array} \right.$$

The transfer matrix is

$$\mathbf{M}(s) = \begin{pmatrix} \frac{\alpha_{\mathsf{P}}}{s + \alpha_{\mathsf{P}}} & 0 \\ 0 & \frac{\beta_{\mathsf{P}}}{s^3 + \beta_{\mathsf{D}^2} s^2 + \beta_{\mathsf{D}} s + \beta_{\mathsf{P}}} \end{pmatrix}.$$

The characteristic polynomial is

$$P(s) = (s + \alpha_{\mathsf{P}})\left(s^3 + \beta_{\mathsf{D}^2} s^2 + \beta_{\mathsf{D}} s + \beta_{\mathsf{P}}\right),$$

If we want all roots to be equal to -1, we should solve:

$$\begin{aligned}(s + \alpha_{\mathsf{P}})\left(s^3 + \beta_{\mathsf{D}^2} s^2 + \beta_{\mathsf{D}} s + \beta_{\mathsf{P}}\right) &= (s+1)^4 \\ &= (s+1)\left(s^3 + 3s^2 + 3\right.\end{aligned}$$

Thus the linear controller to be taken is

$$\begin{cases} v_1 &= w_1 - \delta_s, \\ v_2 &= (w_2 - \theta) - 3\dot{\theta} - 3\ddot{\theta}. \end{cases}$$

Control with wanted inputs

The system to be controlled is described by :

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}).$$

For specific output vector $\mathbf{y} = \mathbf{g}(\mathbf{x})$, feedback linearization methods make it possible to find a controller of the form

$$\mathbf{u} = \mathcal{R}_u(\mathbf{x}, \bar{\mathbf{y}}),$$

such that the output $\mathbf{y}$ converges to $\bar{\mathbf{y}}$.

Now, in many cases, the user wants to choose its own output vector $\mathbf{w} = \mathbf{h}(\mathbf{x})$.

The problem of interest is to find a controller

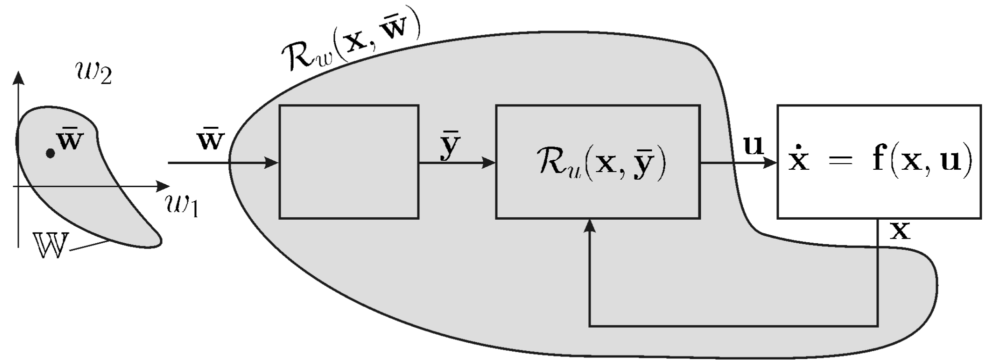
$$\mathbf{u} = \mathcal{R}_w(\mathbf{x}, \bar{\mathbf{w}})$$

such that the $\mathbf{w}$ converges to the *wanted vector* $\bar{\mathbf{w}}$.
The set of all feasible wanted vectors by

$$\mathbb{W} = \{\mathbf{w} \in \mathbb{R}^m | \exists \mathbf{x} \in \mathbb{R}^n, \exists \mathbf{u} \in \mathbb{R}^m, \mathbf{f}(\mathbf{x}, \mathbf{u}) = \mathbf{0}, \mathbf{w} = \mathbf{h}(\mathbf{x})\}.$$

Methodology:

1) Compute an inner and an outer approximation of $\mathbb{W}$.
2) The user choose any point $\bar{\mathbf{w}}$ inside $\mathbb{W}$.
3) From $\bar{\mathbf{w}}$, compute $\bar{\mathbf{x}}$ and $\bar{\mathbf{u}}$ such that $\mathbf{f}(\bar{\mathbf{x}}, \bar{\mathbf{u}}) = \mathbf{0}, \bar{\mathbf{w}} = \mathbf{h}(\bar{\mathbf{x}})$.
4) From $\bar{\mathbf{x}}$, we shall compute $\bar{\mathbf{y}} = \mathbf{g}(\bar{\mathbf{x}})$.
5) The controller $\mathcal{R}_u(\mathbf{x}, \bar{\mathbf{y}})$ will compute $\mathbf{u}$ such that $\mathbf{y}$ converges to $\bar{\mathbf{y}}$. As a consequence, $\mathbf{x}$ will tend to $\bar{\mathbf{x}}$ and $\mathbf{w}$ to $\bar{\mathbf{w}}$.



For the normalized sailboat, with $\mathbf{y} = (\boldsymbol{\delta}_s, \theta)$ a feedback linearization method leads to the controller $\mathcal{R}_u(\mathbf{x}, \bar{\mathbf{y}}) = \mathcal{R}_u(\mathbf{x}, \hat{\delta}_s, \hat{\theta})$ given by

$$\mathbf{u} = \mathbf{A}^{-1}(\mathbf{x}) \left(\left(\begin{array}{c} \hat{\delta}_s - \delta_s \\ \hat{\theta} - \theta - 3\omega - 3\dot{\omega} \end{array} \right) - \mathbf{b}(\mathbf{x}) \right)$$

where

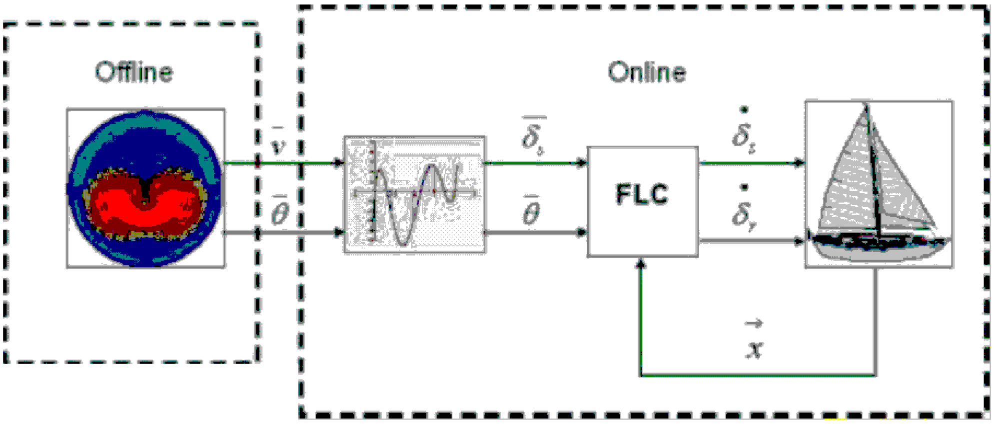
$$\begin{array}{rcl} \mathbf{A}(\mathbf{x}) & = & \left(\begin{array}{c} 1 \\ f_s \sin\delta_s + (v\cos\delta_s + \sin(\theta+\delta_s))(\cos\delta_s - 1) \quad \end{array} \right. \\ \mathbf{b}(\mathbf{x}) & = & \left(\begin{array}{c} 0 \\ (v\sin\delta_s + \omega\sin(\theta+\delta_s))(\cos\delta_s - 1) - v\cos\delta \end{array} \right. \end{array}$$

Principle of the control

1) Choose $\bar{\mathbf{w}} = \left(\bar{v}, \bar{\theta}\right)$ in the polar speed diagram.

2) Compute $\bar{\mathbf{y}} = (\bar{\delta}_s, \bar{\theta})$ such that $\exists \mathbf{x}, \mathbf{f}(\mathbf{x}, \bar{\mathbf{y}}) = \mathbf{0}$ and $\mathbf{w} = \mathbf{h}(\mathbf{x})$.

3) Apply the control based on feedback linearization.



Control of a wheeled stair-climbing robot

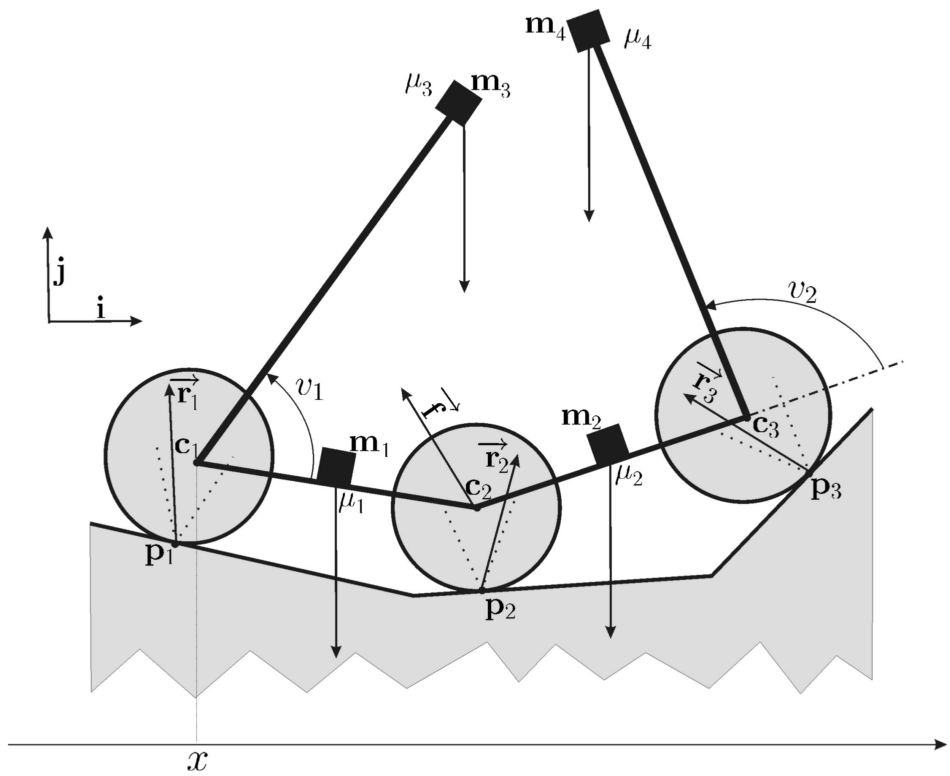
(Luc Jaulin, Thursday, 11h45-12h15).

(Collaboration with students and colleagues from ENSIETA)

Consider the class of constrained dynamic systems:

$$\begin{array}{ll} \text{(i)} & \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) \\ \text{(ii)} & (\mathbf{x}(t), \mathbf{v}(t)) \in \mathbb{V}, \end{array}$$

where $\mathbf{v}(t) \in \mathbb{R}^{n_v}$ is the *viable input vector* and $\mathbb{V}$ is the *viable set*.



Assume that the robot has a quasi-static motion.

1) When the robot does not move, we have

$$\left\{\begin{array}{rcl} -\overrightarrow{\mathbf{p_1m_1}} \wedge \mu_1\mathbf{j} + \overrightarrow{\mathbf{p_1c_2}} \wedge \overrightarrow{\mathbf{f}} - \overrightarrow{\mathbf{p_1m_3}} \wedge \mu_3\mathbf{j} & = & 0 \\ -\overrightarrow{\mathbf{p_2m_2}} \wedge \mu_2\mathbf{j} - \overrightarrow{\mathbf{p_2c_2}} \wedge \overrightarrow{\mathbf{f}} + \overrightarrow{\mathbf{p_2p_3}} \wedge \overrightarrow{\mathbf{r}}_3 & & \\ -\overrightarrow{\mathbf{p_2m_4}} \wedge \mu_4\mathbf{j} & = & 0 \\ \overrightarrow{\mathbf{r}}_1 - (\mu_1 + \mu_3)\mathbf{j} + \overrightarrow{\mathbf{f}} & = & 0 \\ \overrightarrow{\mathbf{r}}_2 - \overrightarrow{\mathbf{f}} - (\mu_2 + \mu_4)\mathbf{j} + \overrightarrow{\mathbf{r}}_3 & = & 0, \end{array}\right.$$

This system can be written into a matrix form as

$$\mathbf{A}_1(x).\mathbf{y} = \mathbf{b}_1(x),$$

where

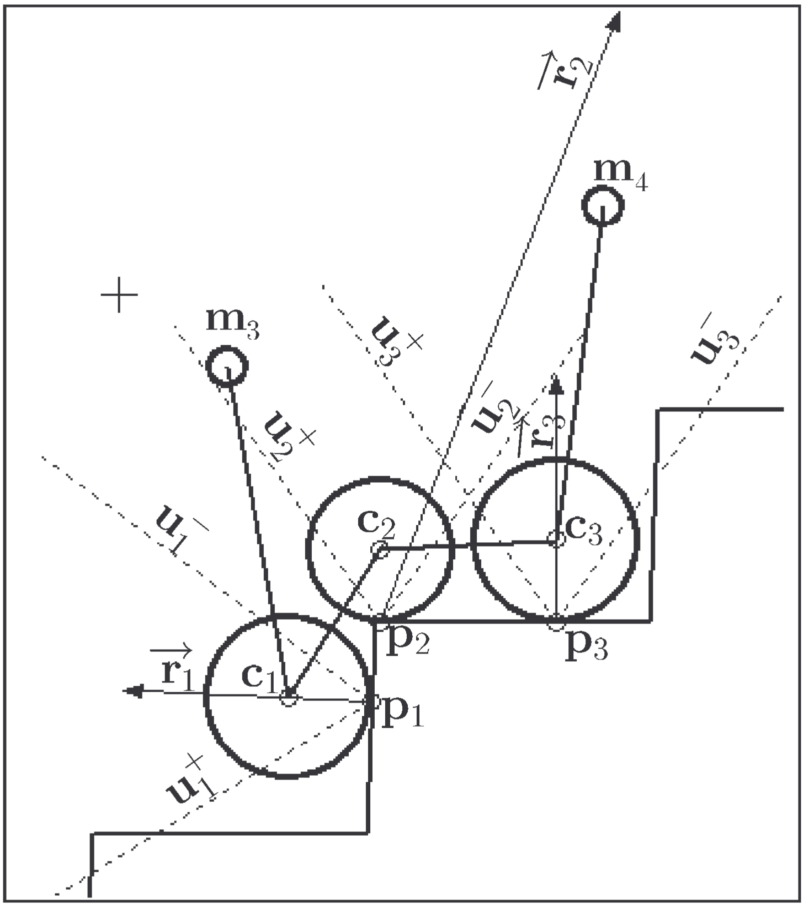
$$\mathbf{y} \;=\; \left(r_{1x}, r_{1y}, r_{2x}, r_{2y}, r_{3x}, r_{3y}, f_x, f_y, m_{3x}, m_{4x}\right)^{\mathsf{T}}.$$

2) None of the wheels will slide if all $\overrightarrow{\mathbf{r}}_i$ belong to their corresponding Coulomb cones:

$$\det(\overrightarrow{\mathbf{r}}_i, \mathbf{u}_i^-) \leq 0 \text{ and } \det(\mathbf{u}_i^+, \overrightarrow{\mathbf{r}}_i) \leq 0,$$

where $\mathbf{u}_i^-$ and $\mathbf{u}_i^+$ denote the two vectors supporting the ith Coulomb cone $\mathcal{C}_i$. These inequalities can be rewritten into

$$\mathbf{A}_2(x).\mathbf{y} \leq \mathbf{0}.$$

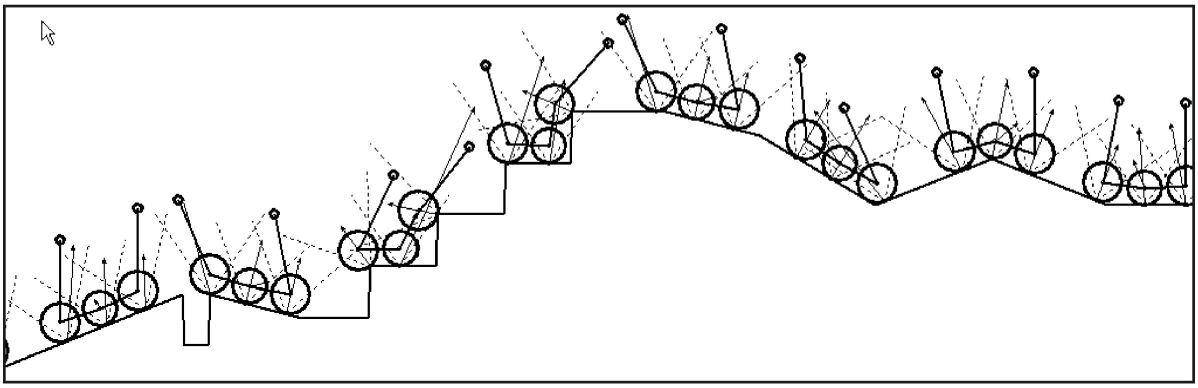


$\overrightarrow{\mathbf{r}}_2$
$\mathbf{m}_4$
$\mathbf{m}_3$
$\mathbf{u}_3^+$
$\mathbf{u}_3^-$
$\mathbf{u}_2^-$
$\overrightarrow{\mathbf{r}}_3$
$\mathbf{u}_2^+$
$\mathbf{u}_1^-$
$\mathbf{c}_2$
$\mathbf{c}_3$
$\mathbf{p}_2$
$\mathbf{p}_3$
$\overrightarrow{\mathbf{r}}_1$
$\mathbf{c}_1$
$\mathbf{p}_1$
$\mathbf{u}_1^+$

3) There is a relation between $\mathbf{y}$ and $\mathbf{v}$ of the form $\mathbf{v} = \mathbf{c}(\mathbf{y})$.

Finally,

$$\begin{aligned}
\mathbf{A}_1(x).\mathbf{y} &= \mathbf{b}_1(x) \\
\mathbf{A}_2(x).\mathbf{y} &\leq \mathbf{0}. \\
\mathbf{v} &= \mathbf{c}(\mathbf{y})
\end{aligned}$$



Robot built by the robotics team of the ENSIETA engineering school that has won the 2005 robot cup ETAS.

Robust control

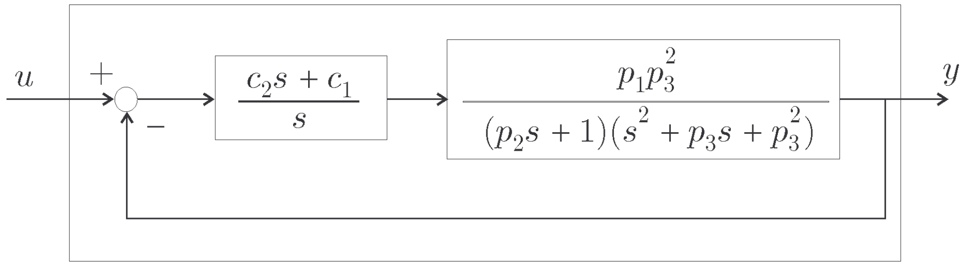
(Luc Jaulin and Michel Kieffer, Thursday,
14h00-15h15).

Robust control of a linear system

$\mathbf{p} \in [\mathbf{p}] = [0.9, 1.1]^{\times 3}$, $[\mathbf{c}] = [0, 1]^2$.

Using the Routh criterion we are able to find r: $\mathbb{R}^2 \times \mathbb{R}^3 \longrightarrow \mathbb{R}$ such that

$$\boldsymbol{\Sigma}(\mathbf{p}, \mathbf{c}) \text{ is stable } \Leftrightarrow r(\mathbf{c}, \mathbf{p}) > 0.$$



Finding all robust controllers amounts to characterizing the set

$$\mathbb{T}_{\mathsf{c}} = \{\mathbf{c} \in [\mathbf{c}] \mid \forall \mathbf{p} \in [\mathbf{p}], r(\mathbf{c}, \mathbf{p}) > 0\} \,.$$

The complementary set of $\mathbb{T}_{\mathsf{c}}$ in $[\mathbf{c}]$:

$$\neg\mathbb{T}_{\mathsf{c}} = \{\mathbf{c} \in [\mathbf{c}] \mid \exists \mathbf{p} \in [\mathbf{p}], r(\mathbf{c}, \mathbf{p}) \leq 0\}$$

is thus the projection of a set defined by a nonlinear inequality.

The transfer function of $\boldsymbol{\Sigma}(\mathbf{p}, \mathbf{c})$ is

$$H(s) = \frac{\left(c_2 s + c_1\right) p_1}{\frac{p_2}{p_3^2} s^4 + \left(\frac{p_2}{p_3} + \frac{1}{p_3^2}\right) s^3 + \left(p_2 + \frac{1}{p_3}\right) s^2 + \left(1 + c_2 p_1\right) s + c_1}$$

The first column of the Routh table is

$$\begin{pmatrix} p_2 \\ p_2p_3+1 \\ p_2p_3^2+p_3-\frac{p_2\left(p_3^2+c_2p_1p_3^2\right)}{p_2p_3+1} \\ p_3^2+c_2p_1p_3^2-\frac{(p_2p_3+1)^2\left(c_1p_1p_3^2\right)}{\left(p_2p_3^2+p_3\right)(p_2p_3+1)-p_2\left(p_3^2+c_2p_1p_3^2\right)} \\ c_1p_1p_3^2 \end{pmatrix}.$$

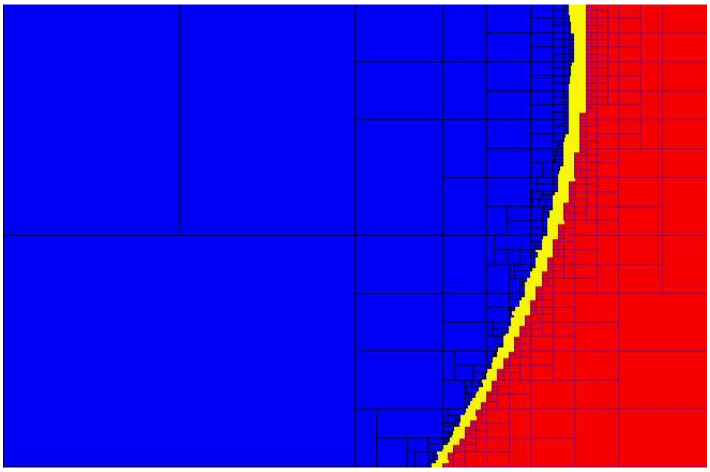
Since $p_2 > 0$, the closed-loop system $\mathbf{\Sigma}(\mathbf{p}, \mathbf{c})$ is asymptotically stable if and only if

$$r\left(\mathbf{c}, \mathbf{p}\right) \stackrel{\text{def}}{=} \min \left(\begin{array}{c} p_2 p_3 + 1 \\ p_2 p_3^2 + p_3 - \frac{p_2 (p_3^2 + c_2 p_1 p_3^2)}{p_2 p_3 + 1} \\ p_3^2 + c_2 p_1 p_3^2 - \frac{(p_2 p_3 + 1)^2 (c_1 p_1 p_3^2)}{(p_2 p_3^2 + p_3)(p_2 p_3 + 1) - p_2 (p_3^2 + c_2 p_1 p_3^2} \\ c_1 p_1 p_3^2 \end{array} \right.$$

The complementary set

$$\neg\mathbb{T}_c \stackrel{\text{def}}{=} \left\{\mathbf{c} \in [0,1]^2 \mid \exists \mathbf{p} \in [0.9;1.1]^{\times 3}, r(\mathbf{c},\mathbf{p}) \leq 0\right\}$$

of the set of robust controller is:



Optimal robust control

Compute of the set $\mathbb{S}_{\mathsf{c}}$ of the vectors $\mathbf{c}$ that maximize the stability degree in the worst case.

This set satisfies

$$\mathbb{S}_{\mathsf{c}} = \arg \max_{\mathbf{c}\in[\mathbf{c}]} \min_{\mathbf{p}\in[\mathbf{p}]} \max_{\mathbf{r}(\mathbf{p},\mathbf{c},\delta)\geqslant\mathbf{0}} \delta.$$

The rightmost *max* corresponds to the definition of the stability degree. The *min* ensures the worst-case conditions. The leftmost *max* corresponds to the optimality requirement.

For the closed-loop system $\mathbf{\Sigma}(\mathbf{p}, \mathbf{c})$ considered immediately before, PertMin gives the results of the table below. In this table, $[\delta_M^c]$ is an interval guaranteed to contain the associated optimal robust stability degree. The times are indicated for a Pentium 90.

$[\mathbf{p}]$	Time (s)	$[\delta_M^c]$
$(1,1,1)^\mathsf{T}$	5.5	$[0.300, 0.326]$
$[0.99, 1.01]^{\times 3}$	85	$[0.288, 0.299]$
$[0.95, 1.05]^{\times 3}$	339	$[0.261, 0.282]$
$[0.9, 1.1]^{\times 3}$	345	$[0.230, 0.246]$

■

Control of a time-delay system (With M. Dao, M. Di Loreto, J.F. Lafay and J.J. Loiseau)

Consider the unstable system

$$\dot{x}(t) = x(t) + u(t-1).$$

Let us try to stabilize this system using the following control law:

$$u(t) = \alpha x(t) + \beta x(t-1).$$

We have

$$\dot{x}(t) = x(t) + \alpha x(t-1) + \beta x(t-2).$$

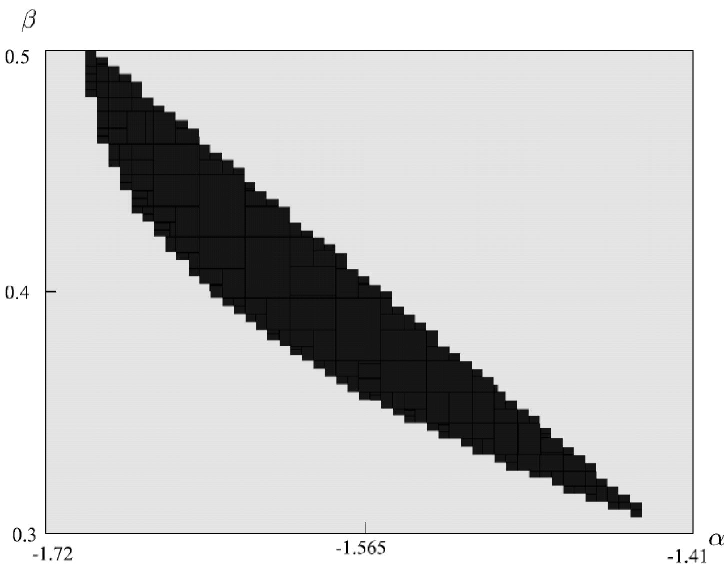
The characteristic equation is

$$s - 1 - \alpha e^{-s} - \beta e^{-2s} = 0.$$

The stability domain is

$$\mathbb{S} = \left\{ (\alpha, \beta) | \nexists s \in \mathbb{C}^+, s - 1 - \alpha e^{-s} - \beta e^{-2s} = 0 \right\}.$$

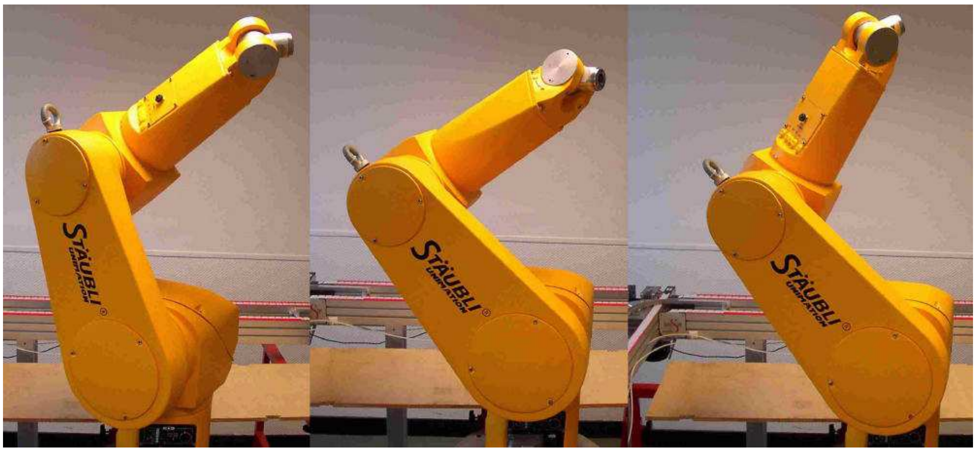
Proving that a box is inside $\mathbb{S}$ can be performed using Proj2d.

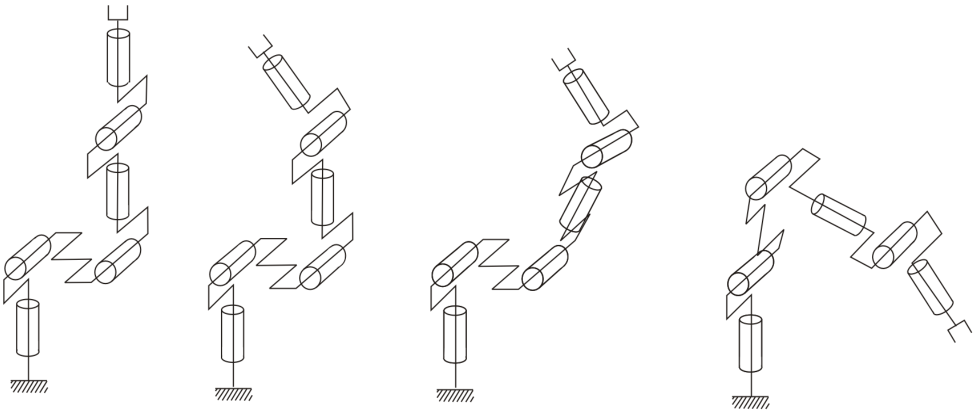


Robot Calibration

(Luc Jaulin and Nacim Ramdani, Friday, 14h00-14h30).

Presentation of the robot (Staubli RX90). (With X. Baguenard, P. Lucidarme and W. Khalil)





Its configuration vector is

$$\mathbf{q} = (q_1, ..., q_6) \in \mathbb{R}^6,$$

where the q_i's are the angles of the articulations.

The tool, is represented by 3 points A_1, A_2, A_3 forming the vector

$$\mathbf{x} = \left(a_x^1, a_y^1, a_z^1, a_x^2, a_y^2, a_z^2, a_x^3, a_y^3, a_z^3\right)^t \in \mathbb{R}^9.$$

The parameter vector of the robot is given by

$$\begin{aligned}\mathbf{p} \;=\; & (r_0, \alpha_1, d_1, r_1, \dots, \alpha_5, d_5, r_5, \alpha_6, d_6, \theta_0, \theta_1^o, \\ & \dots, \theta_5^o, b_x^1, b_y^1, b_z^1, b_x^2, b_y^2, b_z^2, b_x^3, b_y^3, b_z^3)\end{aligned}$$

contains all geometric constants which characterize the robot. For instance, d_i correspond to the length of the ith arm.

The *direct geometric model* is given by

$$\mathbf{x} = \mathbf{f}(\mathbf{p}, \mathbf{q}).$$

where ...

Algorithm f

inputs : $\mathbf{q} = (q_1, \ldots, q_6)^{\mathsf{T}}$,

$$\mathbf{p} = \left(\alpha_j, d_j, r_j, \theta_0, \theta_j^o, b_x^i, b_y^i, b_z^i, \ldots\right)^{\mathsf{T}}.$$

outputs : $\mathbf{x} = \left(a_x^1, a_y^1, a_z^1, a_x^2, a_y^2, a_z^2, a_x^3, a_y^3, a_z^3\right)^{\mathsf{T}}$.

1 $\mathbf{M} := \begin{pmatrix} 1 & 0 & 0 & d_6 \\ 0 & \cos\alpha_6 & -\sin\alpha_6 & 0 \\ 0 & \sin\alpha_6 & \cos\alpha_6 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$

1 $\mathbf{M} := \begin{pmatrix} 1 & 0 & 0 & d_6 \\ 0 & \cos\alpha_6 & -\sin\alpha_6 & 0 \\ 0 & \sin\alpha_6 & \cos\alpha_6 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.\mathbf{M}$

2 for $j := 5$ to 1,

3 $\quad \theta := \theta_j^o + q_j$;

4 $\quad \mathbf{M} := \begin{pmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & r_j \\ 0 & 0 & 0 & 1 \end{pmatrix}.\mathbf{M}$

$\quad \mathbf{M} := \begin{pmatrix} 1 & 0 & 0 & d_j \\ 0 & \cos\alpha_j & -\sin\alpha_j & 0 \\ 0 & \sin\alpha_j & \cos\alpha_j & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.\mathbf{M}$

5 endfor

6 $\mathbf{M} := \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} . \begin{pmatrix} \cos\theta_0 & -\sin\theta_0 & 0 & 0 \\ \sin\theta_0 & \cos\theta_0 & 0 & 0 \\ 0 & 0 & 1 & r_0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.\mathbf{M}$

7 for $i := 1$ to 3, $\quad \mathbf{b}^i = \begin{pmatrix} b_x^i & b_y^i & b_z^i & 1 \end{pmatrix}^{\mathsf{T}}$

8 $\mathbf{x} := \begin{pmatrix} \mathbf{M} & 0 & 0 \\ 0 & \mathbf{M} & 0 \\ 0 & 0 & \mathbf{M} \end{pmatrix} \begin{pmatrix} \mathbf{b}^1 \\ \mathbf{b}^2 \\ \mathbf{b}^3 \end{pmatrix}.$

Principle of the calibration

1. Choose r different configuration vectors $\mathbf{q}(1), ..., \mathbf{q}(r)$.

2. Measure the coordinates of : $\mathbf{x}(1), ..., \mathbf{x}(r)$,

3. Generate the constraints

$$\mathbf{x}(k) = \mathbf{f}(\mathbf{p}, \mathbf{q}(k)), \ k = \{1, ..., r\}.$$

4. Contract the prior domains for all variables $\mathbf{p}, \mathbf{q}(k), \mathbf{x}(k)$ $k = \{1, ..., r\}$.

DAG (Directed Acyclic Graph)

Our problem is a CSP with a huge number of variables and constraints. It is important to rewrite our constraints in an optimal way in order to make the propagation more efficient. Consider for instance the constraints

$$\begin{aligned} y_1 &= \cos(i_1 + i_2).\sin(i_1 + i_2), \\ y_2 &= i_3.\sin^2(i_1 + i_2). \end{aligned}$$

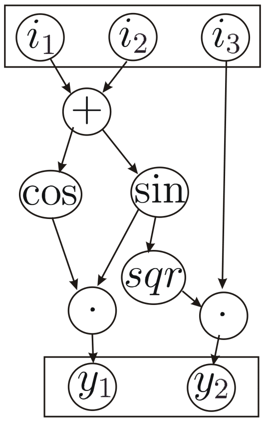
They can be decomposed into primitive constraints as follows

$$\begin{array}{ll} a_1 = i_1 + i_2, & \\ a_2 = \cos\left(a_1\right), & a_5 = i_1 + i_2, \\ a_3 = i_1 + i_2, & a_6 = \sin(a_5), \\ a_4 = \sin(a_3), & a_7 = a_6^2, \\ y_1 = a_2.a_4. & y_2 = i_3.a_7. \end{array}$$

A more efficient representation is

$$
\begin{aligned}
a_1 &= i_1 + i_2, \\
a_2 &= \cos(a_1), \\
a_4 &= \sin(a_1), \qquad & a_7 &= a_4^2, \\
y_1 &= a_2 . a_4 . \qquad & y_2 &= i_3 . a_7 .
\end{aligned}
$$

which is associated to the following DAG



An automatic way to get an optimal decomposition use the notions of DAG (Directed Acyclic Graph) and hatching table.

Generation of simulated measurements

The nominal values chosen for the geometric parameters of the robot STAUBLI RX90 are

j	α_j	d_j	θ_j^o	r_j
0	-	-	$\frac{\pi}{2}$	0.5
1	0.1	0	0	0
2	$-\frac{\pi}{2}$	0	0	0
3	0	0.5	0	0
4	$\frac{\pi}{2}$	0	0	0.5
5	$-\frac{\pi}{2}$	0	0	0
6	$\frac{\pi}{2}$	0	-	-

The three points of the tool with coordinates b_x^i, b_y^i and b_y^i in the terminal arm frame are chosen as

i	b_x^i	b_y^i	b_z^i
1	0.1	0.2	0.1
2	0.1	0.1	0.2
3	0.2	0.1	0.1

For 50 random configuration vector $\mathbf{q}(k)$, we computes $\mathbf{x}(k) = \mathbf{f}(\mathbf{p}, \mathbf{q}(k)) + \mathbf{e}(k)$ where $\mathbf{e}(k)$ is a random bounded noise.

Results

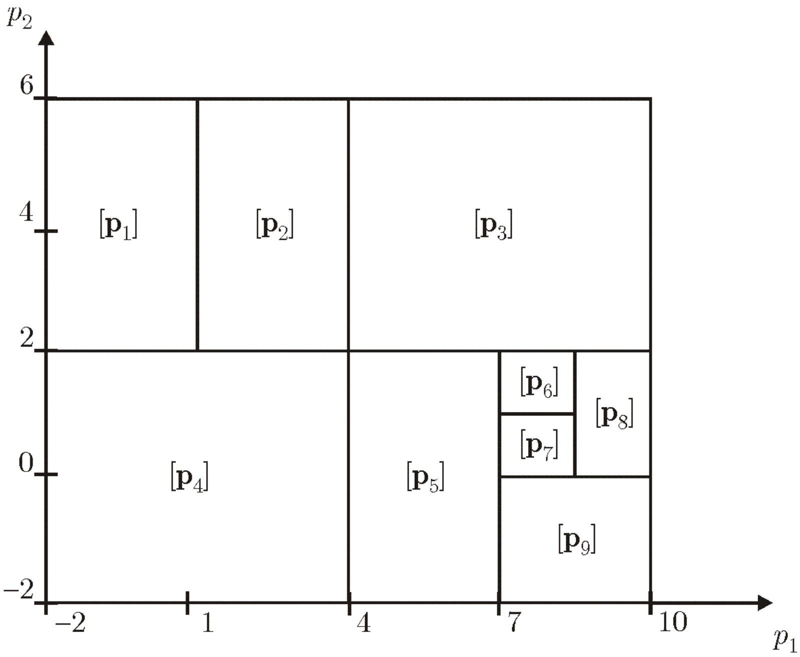
The file containing all constraints takes about 837Ko.
The results obtained are given below.

	initial domains	contracted domains
r_0	$[0.4, 0.6]$	$[0.494046, 0.50101]$
d_1	$[0, 0.1]$	$[0, 0.000558009]$
r_1	$[0, 0.1]$	$[0, 0.00693694]$
d_3	$[0.49, 0.51]$	$[0.498385, 0.501133]$
r_4	$[0.49, 0.51]$	$[0.499216, 0.50114]$
b_x^1	$[0, 0.2]$	$[0.0996052, 0.100629]$
b_y^1	$[0.1, 0.3]$	$[0.199502, 0.200455]$
b_z^1	$[0, 0.2]$	$[0.0997107, 0.100714]$
b_x^2	$[0, 0.2]$	$[0.0996747, 0.100712]$
b_y^2	$[0, 0.2]$	$[0.0994585, 0.10031]$
b_z^2	$[0.1, 0.3]$	$[0.199535, 0.200642]$
b_x^3	$[0.1, 0.3]$	$[0.199689, 0.200578]$
b_y^3	$[0, 0.2]$	$[0.0997562, 0.100319]$
b_z^3	$[0, 0.2]$	$[0.0995661, 0.100557]$

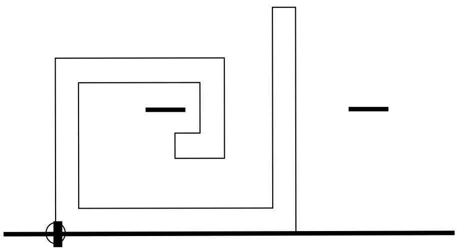
Path planning

(Luc Jaulin, Friday, 15h15-16h00).

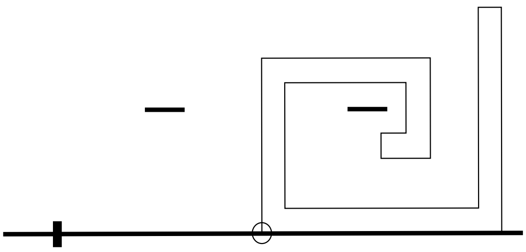
Graph discretization of the configuration space



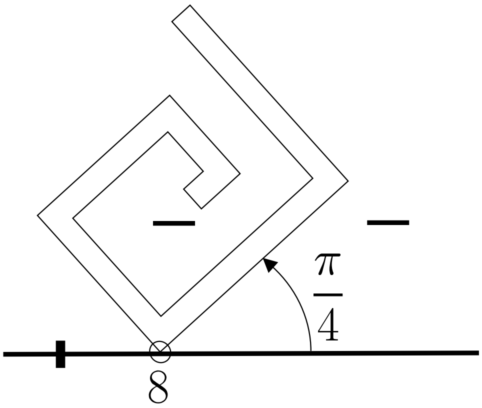
Test case



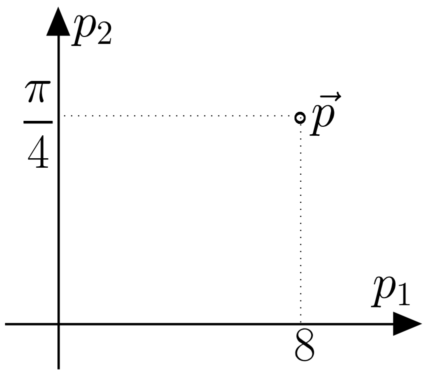
Initial configuration: $\vec{p} = (0\ 0)^{\mathrm{T}}$



Goal configuration: $\vec{p} = (17\ 0)^{\mathrm{T}}$

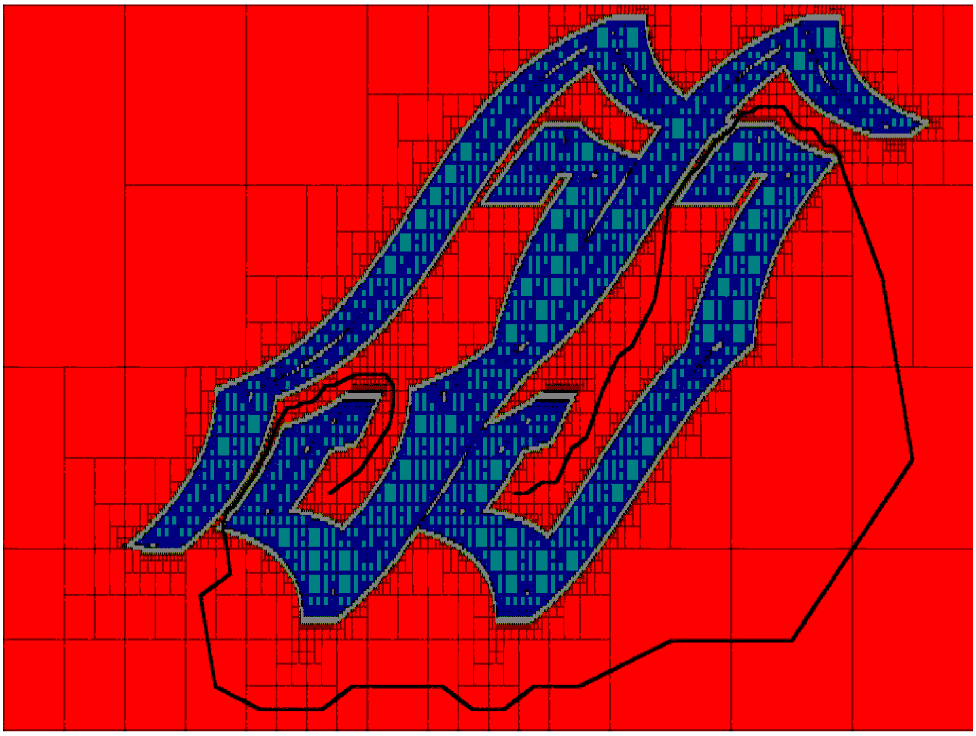


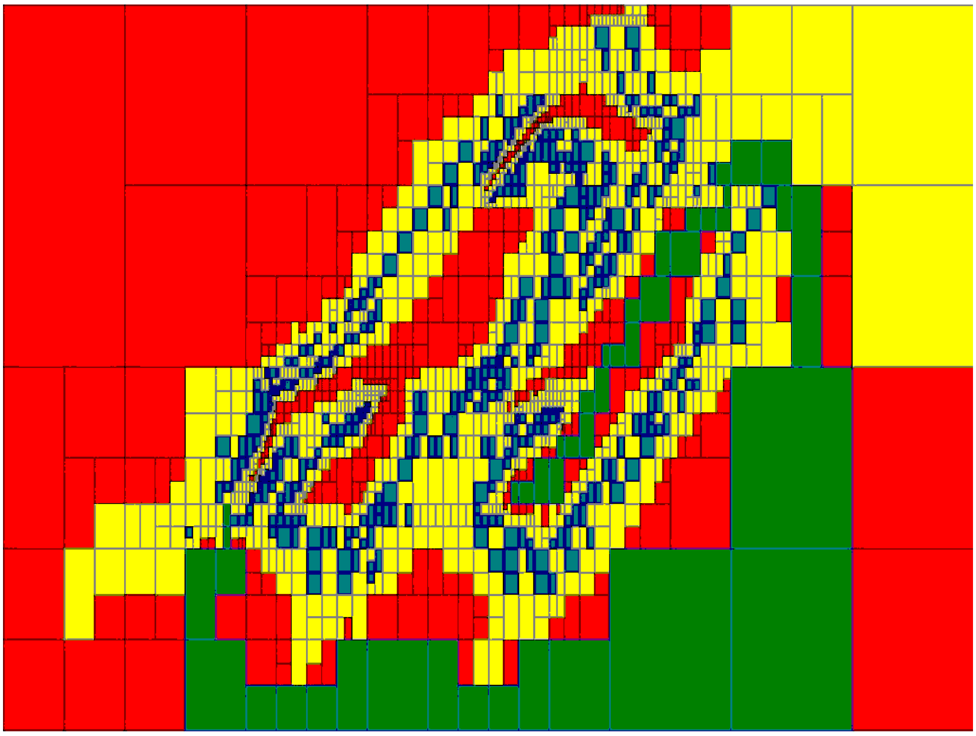
Room

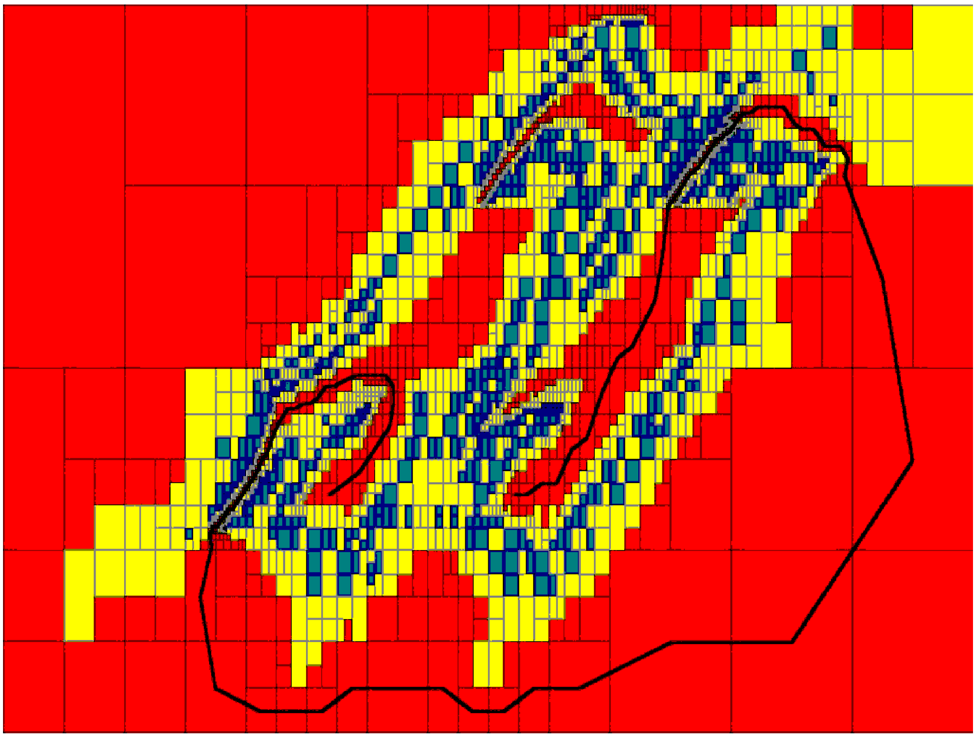


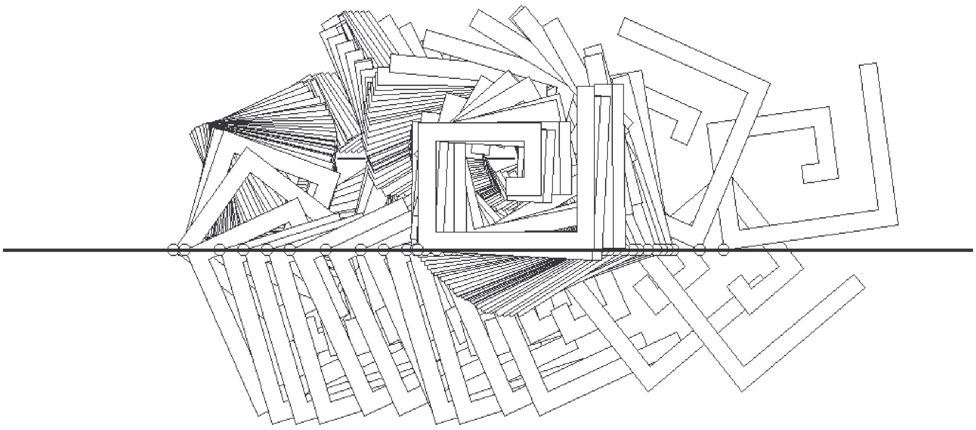
Configuration space

$$\mathbf{p} \in \mathbb{S} \Leftrightarrow \left(\begin{array}{l} \forall i \in \mathcal{I}, \forall j \in \mathcal{J},\ [\mathbf{s}_i, \mathbf{s}_{i+1}] \cap\ [\mathbf{a}_j, \mathbf{b}_j] = \emptyset \\ \text{and } \mathbf{a}_j \text{ and } \mathbf{b}_j \text{ are outside the object} \end{array} \right).$$





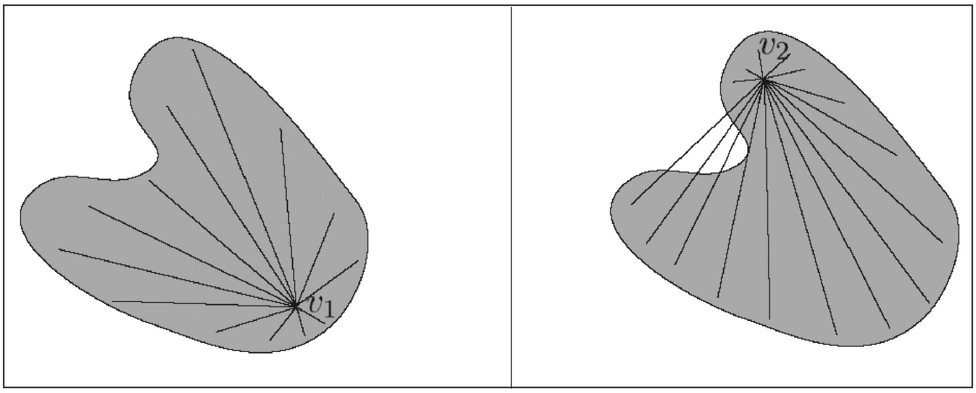




Computing the number of connected components of a set (Collaboration with N. Delanoue and B. Cottenceau)

The point $\mathbf{v}$ is a *star* for $\mathbb{S} \subset \mathbb{R}^n$ if $\forall \mathbf{x} \in \mathbb{S}, \forall \alpha \in [0,1]$, $\alpha \mathbf{v} + (1-\alpha)\mathbf{x} \in \mathbb{S}$.

For instance, in the figure below $\mathbf{v}_1$ is a star for $\mathbb{S}$ whereas $\mathbf{v}_2$ is not.



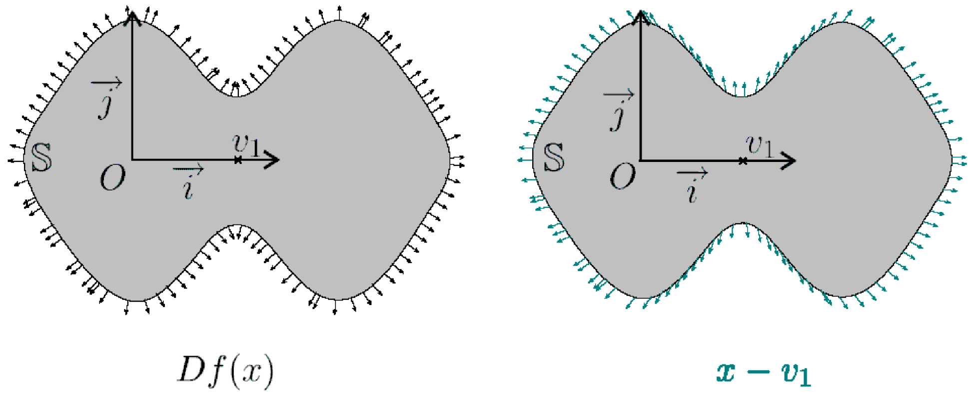
The set $\mathbb{S} \subset \mathbb{R}^n$ is *star-shaped* is there exists $\mathbf{v}$ such that $\mathbf{v}$ is a star for $\mathbb{S}$.

Theorem: Define

$$\mathbb{S} \stackrel{\text{def}}{=} \{\mathbf{x} \in [\mathbf{x}] | f(\mathbf{x}) \leq \mathbf{0}\}$$

where f is differentiable. We have the following implication

$$\{\mathbf{x} \in [\mathbf{x}] \mid f(\mathbf{x}) = 0, Df(\mathbf{x}).(\mathbf{x} - \mathbf{v}) \leq 0\} = \emptyset \Rightarrow \mathbf{v} \text{ is a star for}$$



If $\mathbf{v}$ is a star for $\mathbb{S}_1$ and a star for $\mathbb{S}_2$ then it is a star for $\mathbb{S}_1 \cap \mathbb{S}_2$ and for $\mathbb{S}_1 \cup \mathbb{S}_2$. Thus, one can be used to prove that a point $\mathbf{v}$ is a star for a set defined by a conjunction or a disjunction of inequalities.

Consider a subpaving $\mathcal{P} = \{[\mathbf{p}_1], [\mathbf{p}_2], \ldots\}$ covering $\mathbb{S}$. The relation $\mathcal{R}$ defined by

$$[\mathbf{p}]\mathcal{R}[\mathbf{q}] \Leftrightarrow \mathbb{S} \cap [\mathbf{p}] \cap [\mathbf{q}] \neq \emptyset$$

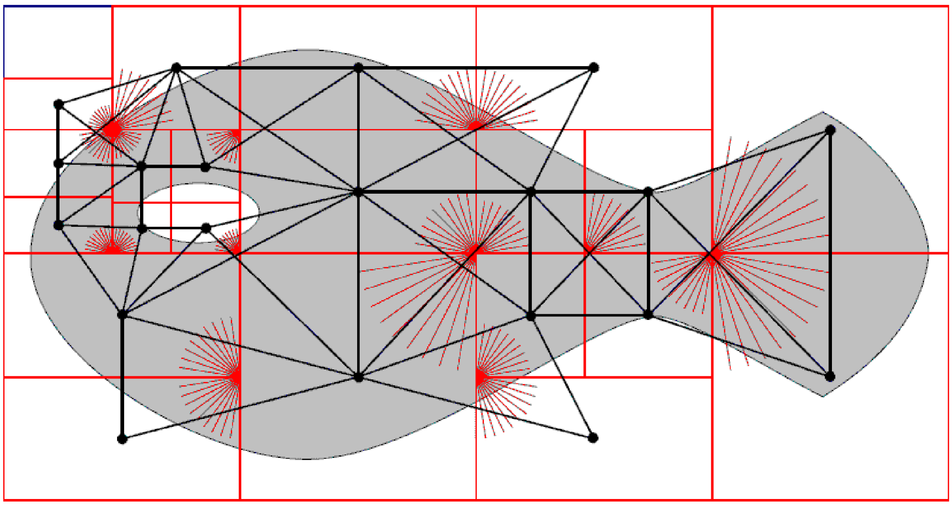
is *star-spangled graph* of $\mathbb{S}$ if

$$\forall [\mathbf{p}] \in \mathcal{P}, \mathbb{S} \cap [\mathbf{p}] \text{ is star-shaped.}$$

For instance, a star-spangled graph for the set

$$\mathbb{S} \stackrel{\text{def}}{=} \left\{ (x,y) \in \mathbb{R}^2 \mid \begin{pmatrix} x^2 + 4y^2 - 16 \\ 2\sin x - \cos y + y^2 - \frac{3}{2} \\ -(x+\frac{5}{2})^2 - 4(y-\frac{2}{5})^2 + \frac{3}{10} \end{pmatrix} \leq 0 \right\},$$

obtained using the solver CIA (`http://www.istia.univ-angers.`
is



Theorem: The number of connected components of the star-spangled graph of $\mathbb{S}$ is equal to that of $\mathbb{S}$.

An extension of this approach has also been developed by N. Delanoue to compute a triangulation homeomorphic to $\mathbb{S}$.