Interval analysis for the validation of autonomous vehicles properties

L. Jaulin

Apple Tech Talk, Munich (virtual),
2024, January 18, 11h.

Interval analysis

Problem. Given $f:\mathbb{R}^n \to \mathbb{R}$ and a box $[\mathbf{x}] \subset \mathbb{R}^n$, prove that

$$\forall \mathbf{x} \in [\mathbf{x}], f(\mathbf{x}) \geq 0.$$

Interval arithmetic can solve efficiently this problem.

Example. Is the function

$$f(\mathbf{x}) = x_1 x_2 - (x_1 + x_2)\cos x_2 + \sin x_1 \cdot \sin x_2 + 2$$

always positive for $x_1, x_2 \in [-1, 1]$?

Interval arithmetic

$$
\begin{aligned}
[-1,3]+[2,5] &= ?, \\
[-1,3]\cdot[2,5] &= ?, \\
\mathsf{abs}\,([-7,1]) &= ?
\end{aligned}
$$

Interval arithmetic

$$\begin{aligned} [-1,3]+[2,5] &= [1,8], \\ [-1,3]\cdot[2,5] &= [-5,15], \\ \mathsf{abs}([-7,1]) &= [0,7] \end{aligned}$$

The interval extension of

$$f(x_1, x_2) = x_1 \cdot x_2 - (x_1 + x_2) \cdot \cos x_2 + \sin x_1 \cdot \sin x_2 + 2$$

is

$$\begin{aligned} [f]([x_1],[x_2]) &= [x_1] \cdot [x_2] - ([x_1] + [x_2]) \cdot \cos[x_2] \\ &\quad + \sin[x_1] \cdot \sin[x_2] + 2. \end{aligned}$$

Theorem (Moore, 1970)

$$[f]([\mathbf{x}]) \subset \mathbb{R}^{+} \Rightarrow \forall \mathbf{x} \in [\mathbf{x}], f(\mathbf{x}) \geq 0.$$

Interval arithmetic

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

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

where $[\mathbb{A}]$ is the smallest interval which encloses $\mathbb{A} \subset \mathbb{R}$.

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

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

Inclusion functions

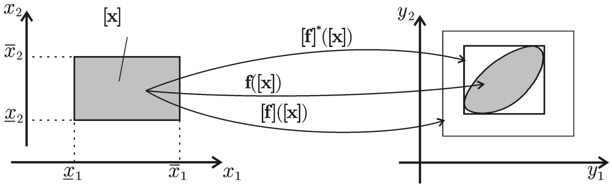
A *box*, or *interval vector* $[\mathbf{x}]$ of $\mathbb{R}^n$ is

$$[\mathbf{x}] = [x_1^-, x_1^+] \times \cdots \times [x_n^-, x_n^+] = [x_1] \times \cdots \times [x_n].$$

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

$[\mathbf{f}] : \mathbb{IR}^n \to \mathbb{IR}^m$ is an *inclusion function* for $\mathbf{f}$ if

$$\forall [\mathbf{x}] \in \mathbb{IR}^n,\ \mathbf{f}([\mathbf{x}]) \subset [\mathbf{f}]([\mathbf{x}]).$$



Inclusion functions $[\mathbf{f}]$ and $[\mathbf{f}]^*$; here, $[\mathbf{f}]^*$ is minimal.

If $\mathbf{f}$ is given by

Algorithm $\mathbf{f}(\text{in} : \mathbf{x} = (x_1, x_2, x_3), \text{ out} : \mathbf{y} = (y_1, y_2))$
$z := x_1$
for$k := 0$ to 100
$\quad\quad z := x_2(z + k \cdot x_3)$
next
$y_1 := z$
$y_2 := \sin(z x_1)$

Its natural inclusion function is

Algorithm $[\mathbf{f}]$(in : $[\mathbf{x}] = ([x_1],[x_2],[x_3])$, out : $[\mathbf{y}] = ([y_1],[y_2])$)
$[z] := [x_1]$ for $k := 0$ to 100 $\quad [z] := [x_2] \cdot ([z] + k \cdot [x_3])$ next $[y_1] := [z]$ $[y_2] := \sin([z] \cdot [x_1])$

Is $[\mathbf{f}]$ convergent? thin? monotonic?

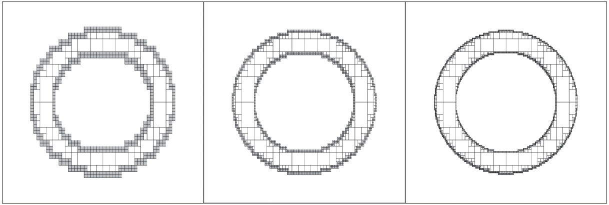
Set inversion

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:

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

Example.

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



Let $\mathbf{f}:\mathbb{R}^n \to \mathbb{R}^m$ and let $\mathbb{Y}$ be a subset of $\mathbb{R}^m$. Set inversion is the characterization of

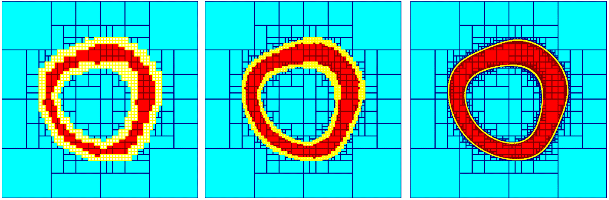
$$\mathbb{X} = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{f}(\mathbf{x}) \in \mathbb{Y}\} = \mathbf{f}^{-1}(\mathbb{Y}).$$

We shall use the following tests.

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

Boxes for which these tests failed, will be bisected, except if they are too small.

Contractors

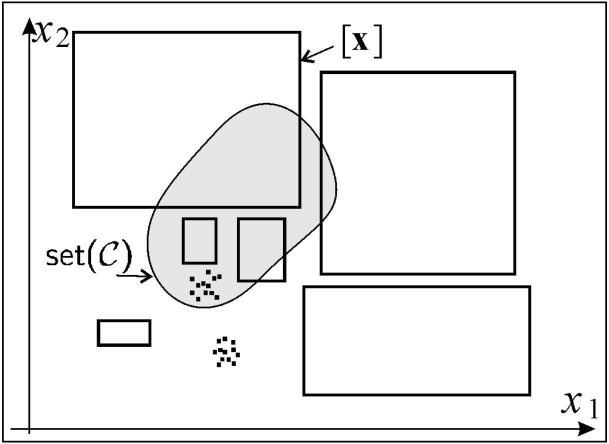


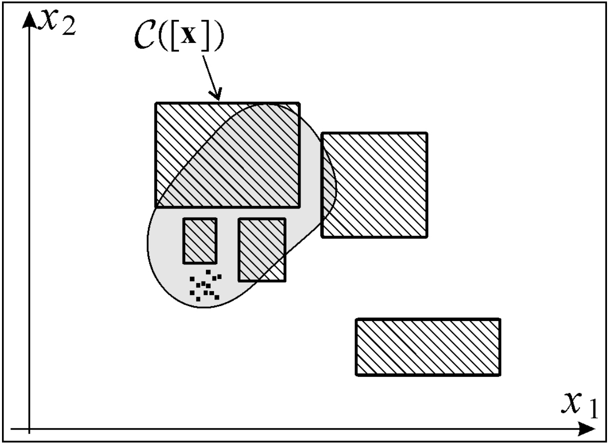
To characterize $\mathbb{X} \subset \mathbb{R}^n$, bisection algorithms bisect all boxes in all directions and become inefficient. Interval methods can still be useful if

- the solution set $\mathbb{X}$ is small (optimization problem, solving equations),
- contraction procedures are used as much as possible,
- bisections are used only as a last resort.

The operator $\mathscr{C} : \mathbb{IR}^n \to \mathbb{IR}^n$ is a *contractor* for $\mathbb{X} \subset \mathbb{R}^n$ if

$$\forall [\mathbf{x}] \in \mathbb{IR}^n, \begin{cases} \mathscr{C}([\mathbf{x}]) \subset [\mathbf{x}] & \text{(contractance),} \\ \mathscr{C}([\mathbf{x}]) \cap \mathbb{X} = [\mathbf{x}] \cap \mathbb{X} & \text{(completeness).} \end{cases}$$


x_2
[x]
set($\mathcal{C}$)
x_1


x_2
$\mathcal{C}([\mathbf{x}])$
x_1

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

$$\forall [\mathbf{x}] \in \mathbb{IR}^n, \begin{cases} \mathscr{C}([\mathbf{x}]) \subset [\mathbf{x}] \\ \mathbf{x} \in [\mathbf{x}] \text{ et } f(\mathbf{x}) = 0 \Rightarrow \mathbf{x} \in \mathscr{C}([\mathbf{x}]) \end{cases}$$

Exercice. Let x, y, z be 3 variables such that

$$\begin{array}{rcl} x & \in & [-\infty, 5] \\ y & \in & [-\infty, 4] \\ z & \in & [6, \infty] \\ z & = & x + y. \end{array}$$

Contract the intervals for x, y, z.

Solution. We have

$$\begin{aligned} x &\in [x] = [2,5] \\ y &\in [y] = [1,4] \\ z &\in [z] = [6,9], \end{aligned}$$

Since $x \in [-\infty, 5], y \in [-\infty, 4], z \in [6, \infty]$ and $z = x + y$, we have

$$\begin{array}{rl} z = x + y \Rightarrow \quad z \in & [6,\infty] \cap ([-\infty,5] + [-\infty,4]) \\ & = [6,\infty] \cap [-\infty,9] = [6,9]. \\ x = z - y \Rightarrow \quad x \in & [-\infty,5] \cap ([6,\infty] - [-\infty,4]) \\ & = [-\infty,5] \cap [2,\infty] = [2,5]. \\ y = z - x \Rightarrow \quad y \in & [-\infty,4] \cap ([6,\infty] - [-\infty,5]) \\ & = [-\infty,4] \cap [1,\infty] = [1,4]. \end{array}$$

The contractor associated with $z = x + y$ is:

Algorithm pplus(inout: $[z],[x],[y]$)

$[z] := [z] \cap ([x]+[y])$ // $z = x + y$
$[x] := [x] \cap ([z]-[y])$ // $x = z - y$
$[y] := [y] \cap ([z]-[x])$ // $y = z - x$

The contractor associated with $z = x \cdot y$ is:

Algorithm pmult (inout: $[z],[x],[y]$)

$[z] := [z] \cap ([x] \cdot [y])$	// $z = x \cdot y$
$[x] := [x] \cap ([z] \cdot 1/[y])$	$//x = z/y$
$[y] := [y] \cap ([z] \cdot 1/[x])$	$//\ y = z/x$

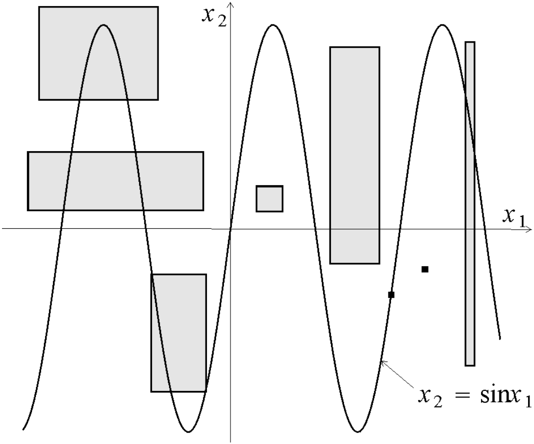
The contractor associated with $y = \exp x$ is:

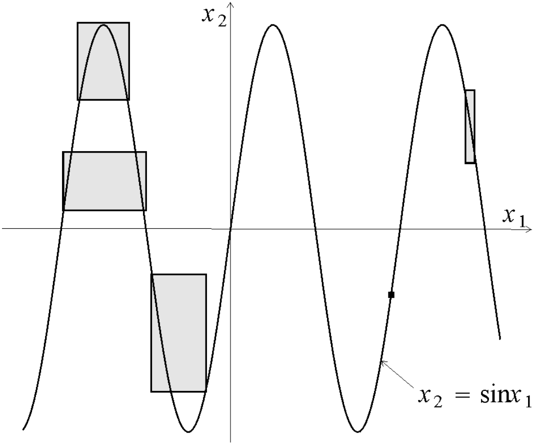
	Algorithm pexp (inout: $[y], [x]$)
1	$[y] := [y] \cap \exp([x])$
2	$[x] := [x] \cap \log([y])$

Any constraint for which such a projection procedure is available will be called a *primitive constraint*.

Example. Consider the primitive equation:

$$x_2 = \sin x_1.$$


x_2
x_1
$x_2 = \sin x_1$


x_2
x_1
$x_2 = \sin x_1$

Decomposition

$$x + \sin(xy) \leq 0,$$
$$x \in [-1,1], y \in [-1,1]$$

Decomposition

$$x + \sin(xy) \leq 0,$$
$$x \in [-1,1], y \in [-1,1]$$

can be decomposed into

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

Forward Backward contractor

For the equation

$$(x_1 + x_2) \cdot x_3 \in [1, 2],$$

we decompose into

$$\begin{array}{rcl} a & = & x_1 + x_2 \\ b & = & a \cdot x_3 \\ b & \in & [1, 2] \end{array}$$

For the equation

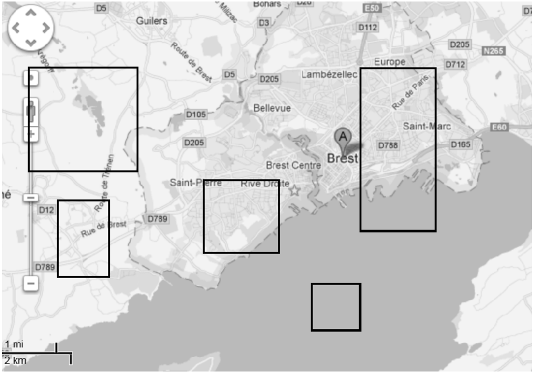
$$(x_1 + x_2) \cdot x_3 \in [1,2],$$

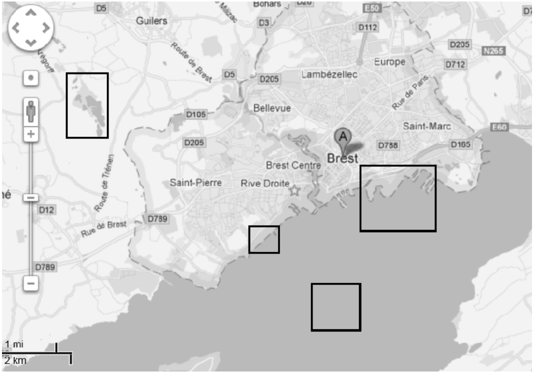
we have the following contractor:

Algorithm $\mathscr{C}$ (inout$[x_1],[x_2],[x_3]$)	
$[a] = [x_1] + [x_2]$	// $a = x_1 + x_2$
$[b] = [a] \cdot [x_3]$	// $b = a \cdot x_3$
$[b] = [b] \cap [1,2]$	// $b \in [1,2]$
$[x_3] = [x_3] \cap \frac{[b]}{[a]}$	// $x_3 = \frac{b}{a}$
$[a] = [a] \cap \frac{[b]}{[x_3]}$	// $a = \frac{b}{x_3}$
$[x_1] = [x_1] \cap [a] - [x_2]$	// $x_1 = a - x_2$
$[x_2] = [x_2] \cap [a] - [x_1]$	// $x_2 = a - x_1$

Contractor on images

The robot with coordinates (x_1, x_2) is in the water.


Guilers
Lambézellec
Europe
Bellevue
Saint-Marc
Brest Centre
Brest
Saint-Pierre
Rive Droite
Route de Brest
Rue de Paris
Rue de Brest
D5
D3
D112
E50
D205
N265
D712
D105
D788
D165
E60
D12
D789
1 mi
2 km


Guilers
Bohars
D5
D3
D112
E50
D205
N265
Europe
D712
Lambézellec
Route de Brest
D205
Bellevue
Rue de Paris
D105
Saint-Marc
E60
D205
D788
D165
Brest
Brest Centre
Route de Trénen
Saint-Pierre
Rive Droite
D12
D789
Rue de Brest
D789
1 mi
2 km

Solving equations

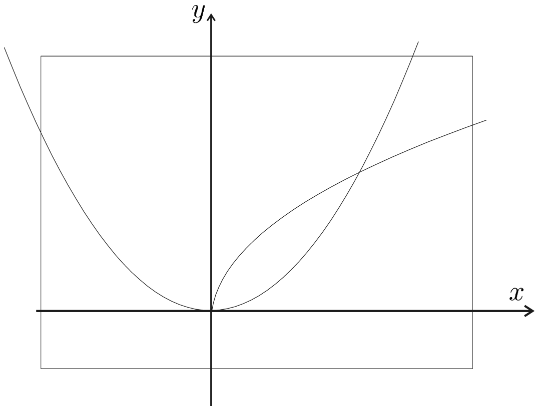
Consider the system of two equations.

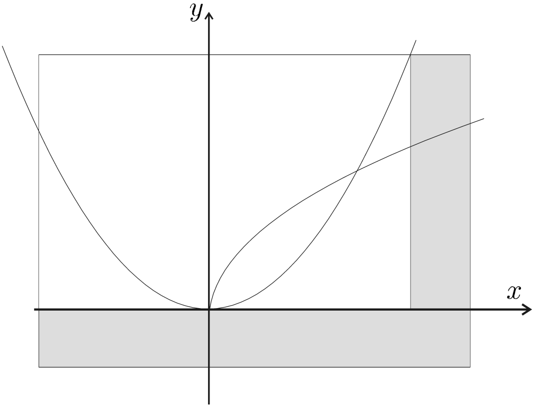
$$\begin{aligned} y &= x^2 \\ y &= \sqrt{x}. \end{aligned}$$

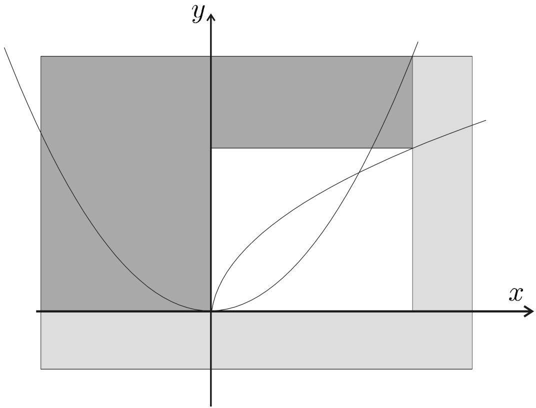
We can build two contractors

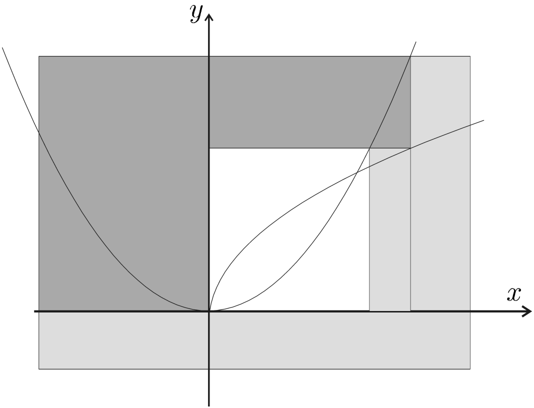
$$\mathscr{C}_1 : \begin{cases} [y] = [y] \cap [x]^2 \\ [x] = [x] \cap \sqrt{[y]} \end{cases} \quad \text{associated to } y = x^2$$

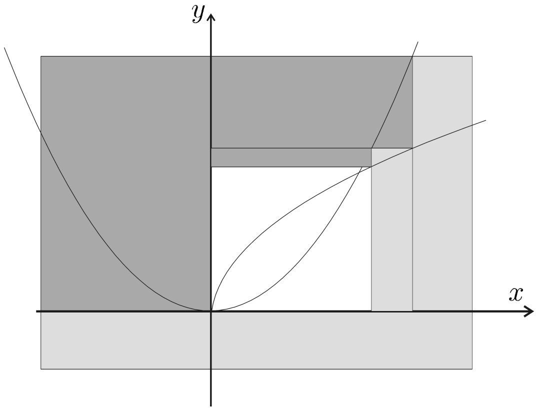
$$\mathscr{C}_2 : \begin{cases} [y] = [y] \cap \sqrt{[x]} \\ [x] = [x] \cap [y]^2 \end{cases} \quad \text{associated to } y = \sqrt{x}$$

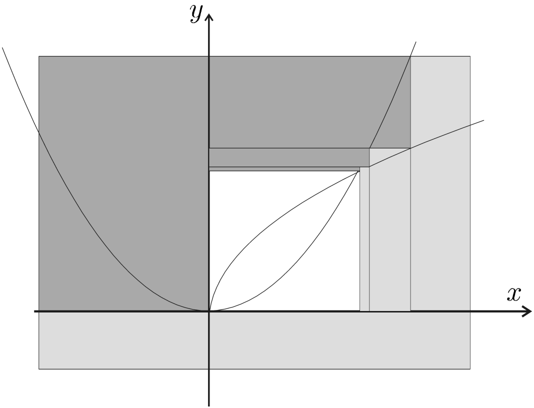

y
x

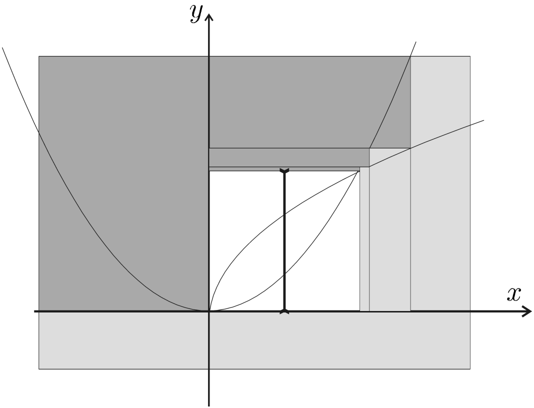

y
x

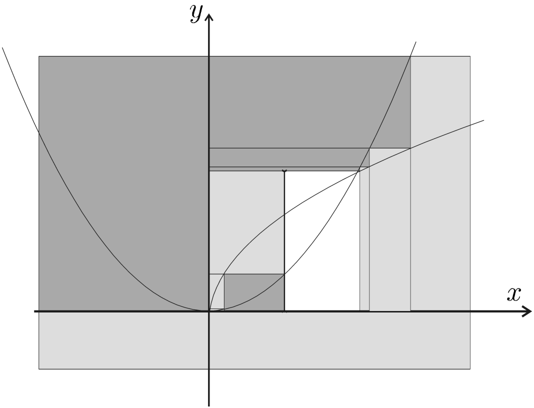

y
x

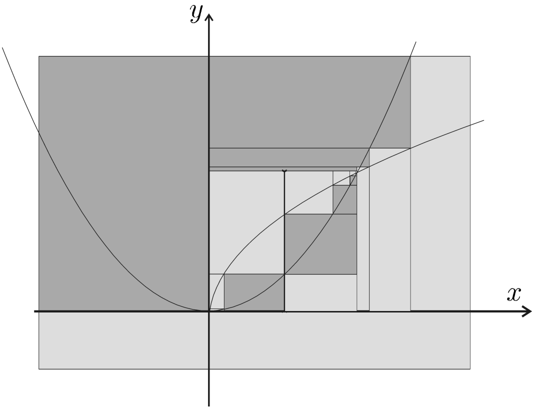

y
x


y
x


y
x

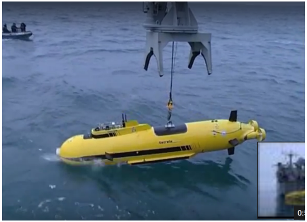

y
x


y
x


y
x

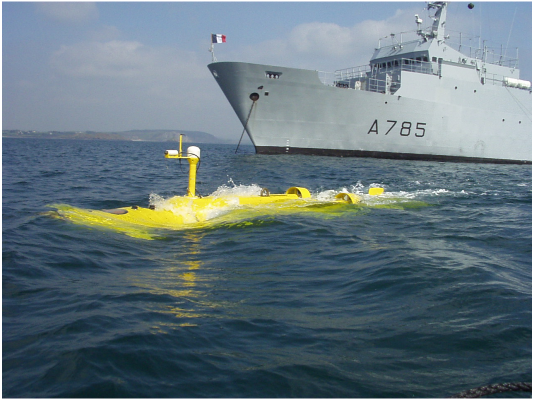
SLAM

$$\begin{cases} \dot{\mathbf{x}} & = & \mathbf{f}(\mathbf{x},\mathbf{u}) & \text{(evolution equation)} \\ \mathbf{y} & = & \mathbf{g}(\mathbf{x},\mathbf{u}) & \text{(observation equation)} \\ \mathbf{z}_i & = & \mathbf{h}(\mathbf{x},\mathbf{u},\mathbf{m}_i) & \text{(mark equation)} \end{cases}$$



Redermor (GESMA, Brest)

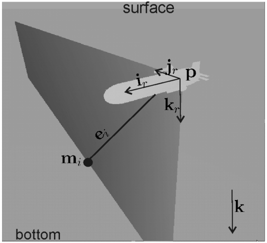
https://youtu.be/X0IqZxb-tFs

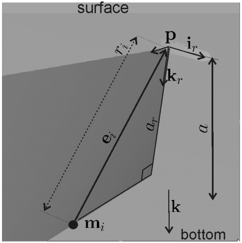

A785

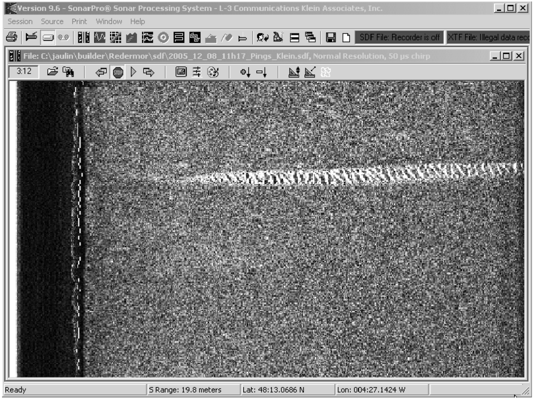
GPS (Global positioning system), only at the surface.

$$t_0 = 0000 \text{ s}, \quad \ell^0 = (-4.4582279°, 48.2129206°) \pm 2.5m$$
$$t_f = 6000 \text{ s}, \quad \ell^f = (-4.4546607°, 48.2191297°) \pm 2.5m$$

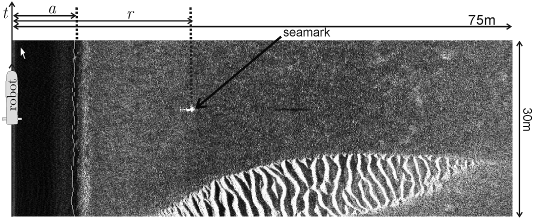
Sonar (KLEIN 5400 side scan sonar).


surface
$\mathbf{j}_r$
$\mathbf{p}$
$\mathbf{i}_r$
$\mathbf{k}_r$
$\mathbf{e}_i$
$\mathbf{m}_i$
$\mathbf{k}$
bottom


surface
$\mathbf{p}$
r_i
$\mathbf{i}_r$
$\mathbf{k}_r$
a_r
a
$\mathbf{e}_i$
$\mathbf{k}$
$\mathbf{m}_i$
bottom



Screenshot of SonarPro



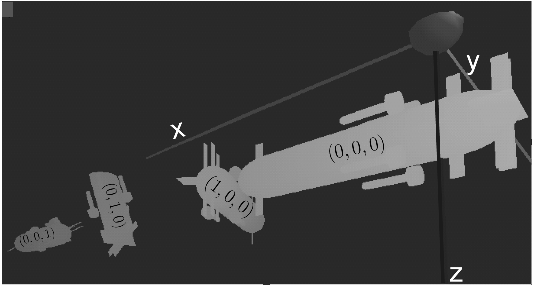
Mine detection with SonarPro

Loch-Doppler returns the speed robot $\mathbf{v}_r$.

$$\mathbf{v}_r \in \tilde{\mathbf{v}}_r + 0.004 * [-1,1] . \tilde{\mathbf{v}}_r + 0.004 * [-1,1]$$

Inertial unit (Octans III).

$$\begin{pmatrix} \phi \\ \theta \\ \psi \end{pmatrix} \in \begin{pmatrix} \tilde{\phi} \\ \tilde{\theta} \\ \tilde{\psi} \end{pmatrix} + \begin{pmatrix} 1.75 \times 10^{-4}.[-1,1] \\ 1.75 \times 10^{-4}.[-1,1] \\ 5.27 \times 10^{-3}.[-1,1] \end{pmatrix}.$$


x
y
z
(0, 0, 0)
(1, 0, 0)
(0, 1, 0)
(0, 0, 1)

Six landmarks have been detected.

i	0	1	2	3	4	5	6	7	8	9	10	11
$\tau(i)$	1054	1092	1374	1748	3038	3688	4024	4817	5172	5232	5279	5688
$\sigma(i)$	1	2	1	0	1	5	4	3	3	4	5	1
$\tilde{r}(i)$	52.42	12.47	54.40	52.68	27.73	26.98	37.90	36.71	37.37	31.03	33.51	15.05

Contraint network

$$t \in \{.0, 0.1, 0.2, \ldots, 5999.9\}$$

$$i \in \{0, 1, \ldots, 11\}$$

$$\begin{pmatrix} p_x(t) \\ p_y(t) \end{pmatrix} = 111120 \begin{pmatrix} 0 & 1 \\ \cos\left(\ell_y(t) * \frac{\pi}{180}\right) & 0 \end{pmatrix} \begin{pmatrix} \ell_x(t) - \ell_x^0 \\ \ell_y(t) - \ell_y^0 \end{pmatrix}$$

$$\mathbf{p}(t) = (p_x(t), p_y(t), p_z(t))$$

$$\mathbf{R}_{\psi}(t)=\begin{pmatrix} \cos\psi(t) & -\sin\psi(t) & 0 \\ \sin\psi(t) & \cos\psi(t) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\mathbf{R}_{\theta}(t)=\begin{pmatrix} \cos\theta(t) & 0 & \sin\theta(t) \\ 0 & 1 & 0 \\ -\sin\theta(t) & 0 & \cos\theta(t) \end{pmatrix}$$

$$\mathbf{R}_{\varphi}(t)=\begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi(t) & -\sin\varphi(t) \\ 0 & \sin\varphi(t) & \cos\varphi(t) \end{pmatrix}$$

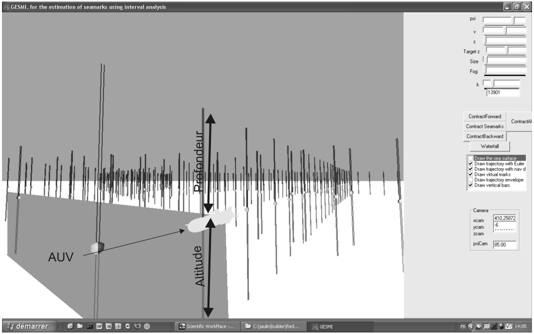
$$\mathbf{R}(t) = \mathbf{R}_{\psi}(t)\mathbf{R}_{\theta}(t)\mathbf{R}_{\varphi}(t)$$

$$\dot{\mathbf{p}}(t) = \mathbf{R}(t).\mathbf{v}_r(t)$$

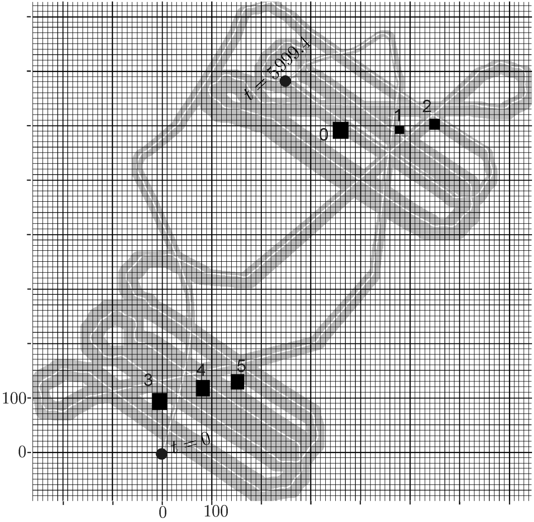
$$||\mathbf{m}(\sigma(i)) - \mathbf{p}(\tau(i))|| = r(i)$$

$$\mathbf{R}^{\mathsf{T}}(\tau(i))\left(\mathbf{m}(\sigma(i)) - \mathbf{p}(\tau(i))\right) \in [0] \times [0,\infty]^{\times 2}$$

$$m_z(\sigma(i)) - p_z(\tau(i)) - a(\tau(i)) \in [-0.5, 0.5]$$



youtu.be/lzJtAfAT7h4


t = 5999.4
0
1
2
3
4
5
t = 0
100
0
0
100

tubex-lib
1.0
Search docs

The following is an extract from the paper Guaranteed computation of robot trajectories.

Definition

A tube $[\mathbf{x}](\cdot)$ is defined as an envelope enclosing an uncertain trajectory $\mathbf{x}(\cdot):\mathbb{R}\rightarrow\mathbb{R}^n$. It is built as an interval of two functions $[\mathbf{x}^-(\cdot),\mathbf{x}^+(\cdot)]$ such that $\forall t,\ \mathbf{x}^-(t)\leqslant\mathbf{x}^+(t)$. A trajectory $\mathbf{x}(\cdot)$ belongs to the tube $[\mathbf{x}](\cdot)$ if $\forall t,\ \mathbf{x}(t)\in[\mathbf{x}](t)$. Fig. 1 illustrates a tube implemented with a set of boxes. This sliced implementation is detailed hereinafter.

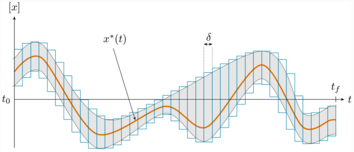
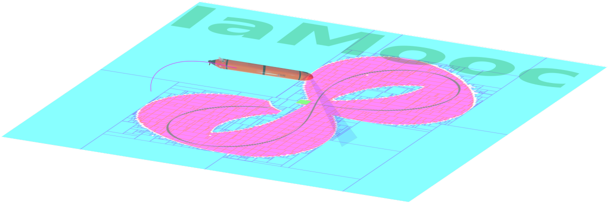


Fig. 1 A tube $[x](\cdot)$ represented by a set of slices. This representation can be used to enclose signals such as $x^*(\cdot)$.

Code example:

```
float timestep = 0.1;
Interval domain(0,10);
Tube x(domain, timestep, Function("t", "(t-5)^2 + [-0.5,0.5]"));
```

http://www.codac.io/



https://www.ensta-bretagne.fr/iamooc/

References

1. Interval analysis [7, 4, 5]
2. Localization with intervals : [6]
3. SLAM with intervals : [3]
4. Interval tubes [8], [2], [1][9]
5. IAMOOC [5]

F. L. Bars, J. Sliwka, O. Reynet, and L. Jaulin.
State estimation with fleeting data.
Automatica, 48(2):381–387, 2012.

A. Bethencourt and L. Jaulin.
Solving non-linear constraint satisfaction problems involving time-dependant functions.
Mathematics in Computer Science, 8(3), 2014.

L. Jaulin.
Range-only SLAM with occupancy maps; A set-membership approach.
IEEE Transaction on Robotics, 27(5):1004–1010, 2011.

L. Jaulin, M. Kieffer, O. Didrit, and E. Walter.
Applied Interval Analysis, with Examples in Parameter and State Estimation, Robust Control and Robotics.
Springer-Verlag, London, 2001.

L. Jaulin, O. Reynet, B. Desrochers, S. Rohou, and J. Ninin.
IaMOOC, Interval analysis with applications to parameter estimation and robot localization ,
www.ensta-bretagne.fr/iamooc/.
ENSTA-Bretagne, 2019.

M. Kieffer, L. Jaulin, E. Walter, and D. Meizel.
Robust autonomous robot localization using interval analysis.
Reliable Computing, 6(3):337–362, 2000.

R. Moore.
Methods and Applications of Interval Analysis.
Society for Industrial and Applied Mathematics, jan 1979.

S. Rohou, L. Jaulin, M. Mihaylova, F. L. Bars, and S. Veres.
Guaranteed Computation of Robots Trajectories.
Robotics and Autonomous Systems, 93:76–84, 2017.

J. A. D. Sandretto and A. Chapoutot.

Dynibex: a differential constraint library for studying dynamical systems.
In *Conference on Hybrid Systems: Computation and Control*, Vienne, Austria, 2016.