Intervals for state estimation

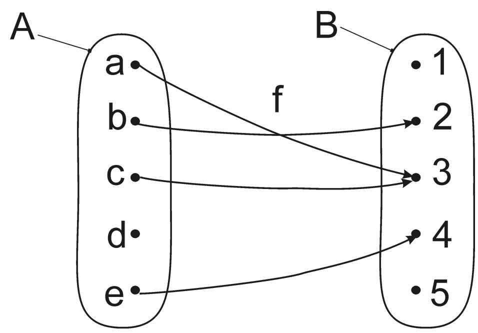
Luc Jaulin, ENSTA Brest

ECA Brest, 23 Mai, 2015

1 Interval analysis

1.1 Basic notions on set theory

If f is defined as follows



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

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

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

1.2 Interval arithmetic

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

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

For instance,

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

$$
\begin{aligned}
[x^-, x^+] + [y^-, y^+] = \;& [x^- + y^-, x^+ + y^+]. \\
[x^-, x^+] \cdot [y^-, y^+] = \;& [x^-y^- \wedge x^+y^- \wedge x^-y^+ \wedge x^+y^+, \\
& x^-y^- \vee x^+y^- \vee x^-y^+ \vee x^+y^+].
\end{aligned}
$$

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

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

For instance,

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

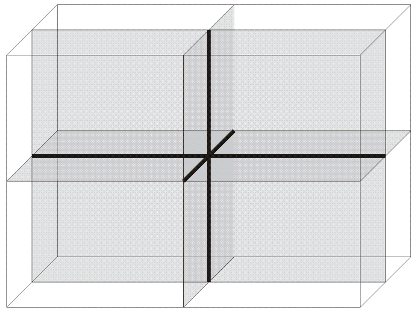
1.3 Boxes

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 $\mathbb{IR}^n$.

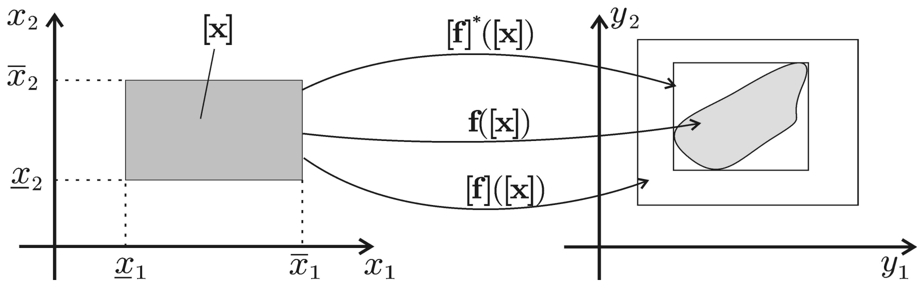
The *principal plane* of $[\mathbf{x}]$ is the symmetric plane $[\mathbf{x}]$ perpendicular to its largest side.



1.4 Inclusion function

The interval function $[\mathbf{f}]$ from $\mathbb{IR}^n$ to $\mathbb{IR}^m$, is an *inclusion function* of $\mathbf{f}$ if

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



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

The natural inclusion function for $f(x) = x^2 + 2x + 4$ is

$$[f]([x]) = [x]^2 + 2[x] + 4.$$

If $[x] = [-3, 4]$, we have

$$\begin{aligned} [f]([-3,4]) &= [-3,4]^2 + 2[-3,4] + 4 \\ &= [0,16] + [-6,8] + 4 \\ &= [-2,28]. \end{aligned}$$

Note that $f([-3, 4]) = [3, 28] \subset [f]([-3, 4]) = [-2, 28]$.

If $\mathbf{f}$ is given by the algorithm

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
    $z := x_2(z + kx_3)$;
next;
$y_1 := z$;
$y_2 := \sin(z \cdot x_1)$;

Its natural inclusion function is

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

Here, $[\mathbf{f}]$ is a convergent, thin and monotonic inclusion function for $\mathbf{f}$.

1.5 Subpavings

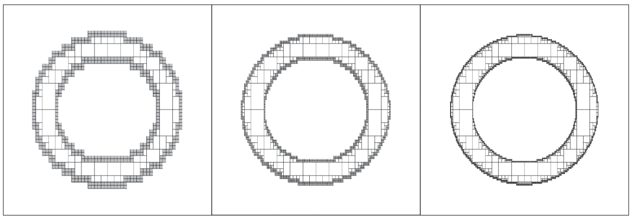
A subpaving of $\mathbb{R}^n$ is a set of non-overlapping boxes of $\mathbb{R}^n$.

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

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

Example.

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



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.

1.6 Set inversion

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.

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

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

.

Algorithm Sivia(in: $[\mathbf{x}](0), \mathbf{f}, \mathbb{Y}$)

1 $\mathcal{L} := \{[\mathbf{x}](0)\}$;
2 pull $[\mathbf{x}]$ from $\mathcal{L}$;
3 if $[\mathbf{f}]([\mathbf{x}]) \subset \mathbb{Y}$, draw($[\mathbf{x}]$, 'red');
4 elseif $[\mathbf{f}]([\mathbf{x}]) \cap \mathbb{Y} = \emptyset$, draw($[\mathbf{x}]$, 'blue');
5 elseif $w([\mathbf{x}]) < \varepsilon$, {draw ($[\mathbf{x}]$, 'yellow')};
6 else bisect $[\mathbf{x}]$ and push into $\mathcal{L}$;
7 if $\mathcal{L} \neq \emptyset$, go to 2

If $\mathbf{\Delta}\mathbb{X}$ denotes the union of yellow boxes and if $\mathbb{X}^-$ is the union of red boxes then :

$$\mathbb{X}^- \subset \mathbb{X} \subset \mathbb{X}^- \cup \mathbf{\Delta}\mathbb{X}.$$

2 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),

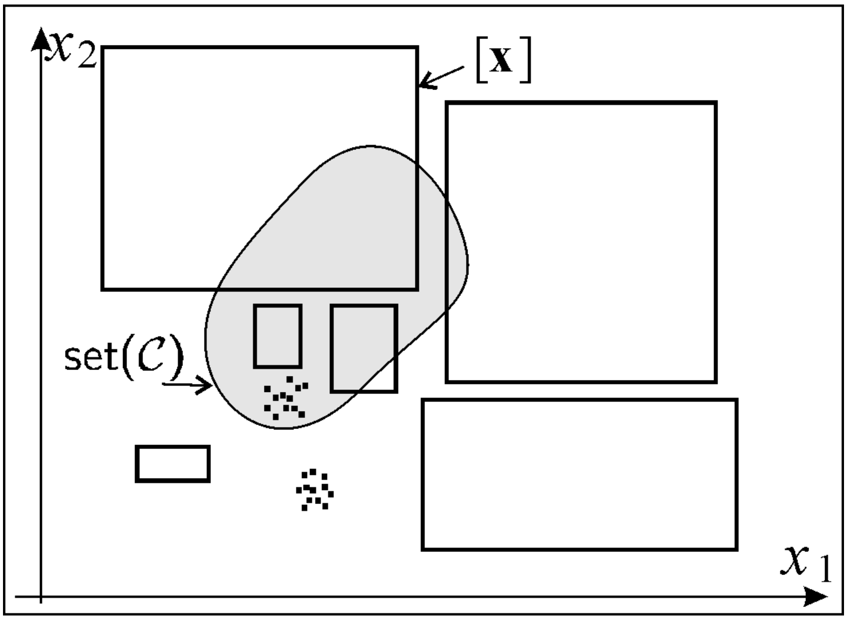
- contraction procedures are used as much as possible,

- bisections are used only as a last resort.

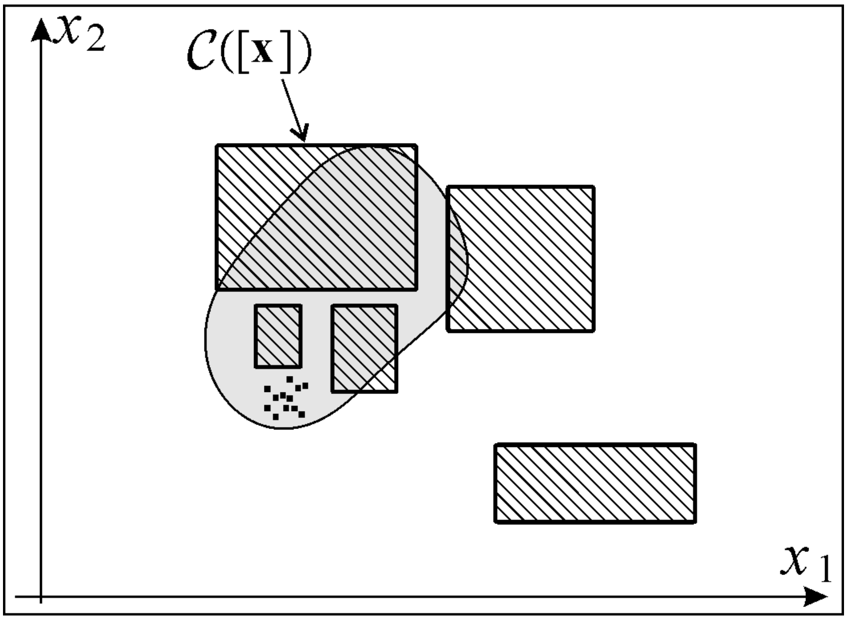
2.1 Definition

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

$$\forall [\mathbf{x}] \in \mathbb{IR}^n, \left\{ \begin{array}{ll} \mathcal{C}_{\mathbb{X}}([\mathbf{x}]) \subset [\mathbf{x}] & \text{(contractance)}, \\ \mathcal{C}_{\mathbb{X}}([\mathbf{x}]) \cap \mathbb{X} = [\mathbf{x}] \cap \mathbb{X} & \text{(completeness)}. \end{array} \right.$$



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



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

$\mathcal{C}_{\mathbb{X}}$ is said to be *convergent* if

$$[\mathbf{x}](k) \to \mathbf{x} \quad \Rightarrow \quad \mathcal{C}_{\mathbb{X}}([\mathbf{x}]\,(k)) \to \{\mathbf{x}\} \cap \mathbb{X}.$$

2.2 Projection of constraints

Let x, y, z be 3 variables such that

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

Which values for x, y, z are consistent.

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

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

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

Algorithm pplus(inout: $[z], [x], [y]$)
$[z] := [z] \cap ([x] + [y])$;
$[x] := [x] \cap ([z] - [y])$;
$[y] := [y] \cap ([z] - [x])$.

The projection procedure developed for plus can be extended to other ternary constraints such as mult: $z = x \cdot y$, or equivalently

$$\text{mult} \triangleq \left\{(x, y, z) \in \mathbb{R}^3 \mid z = x \cdot y\right\}.$$

The resulting projection procedure becomes

Algorithm pmult(inout: $[z], [x], [y]$)	
1	$[z] := [z] \cap ([x] \cdot [y])$;
2	$[x] := [x] \cap ([z] \cdot 1/[y])$;
3	$[y] := [y] \cap ([z] \cdot 1/[x])$.

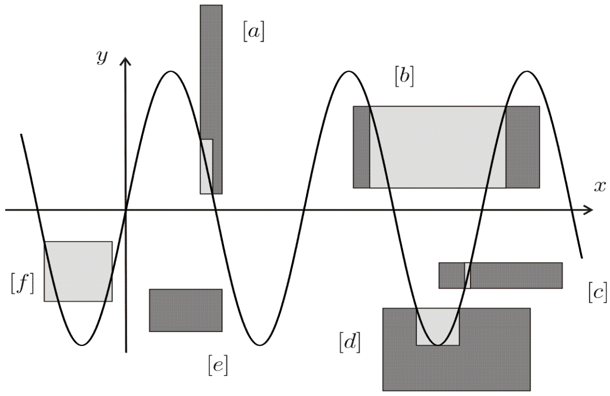
Consider the binary constraint

$$\exp \triangleq \{(x, y) \in \mathbb{R}^n | y = \exp(x)\}.$$

The associated contractor 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*.



Projection of the sine constraint

2.3 Solvers

A CSP (Constraint Satisfaction Problem) 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.

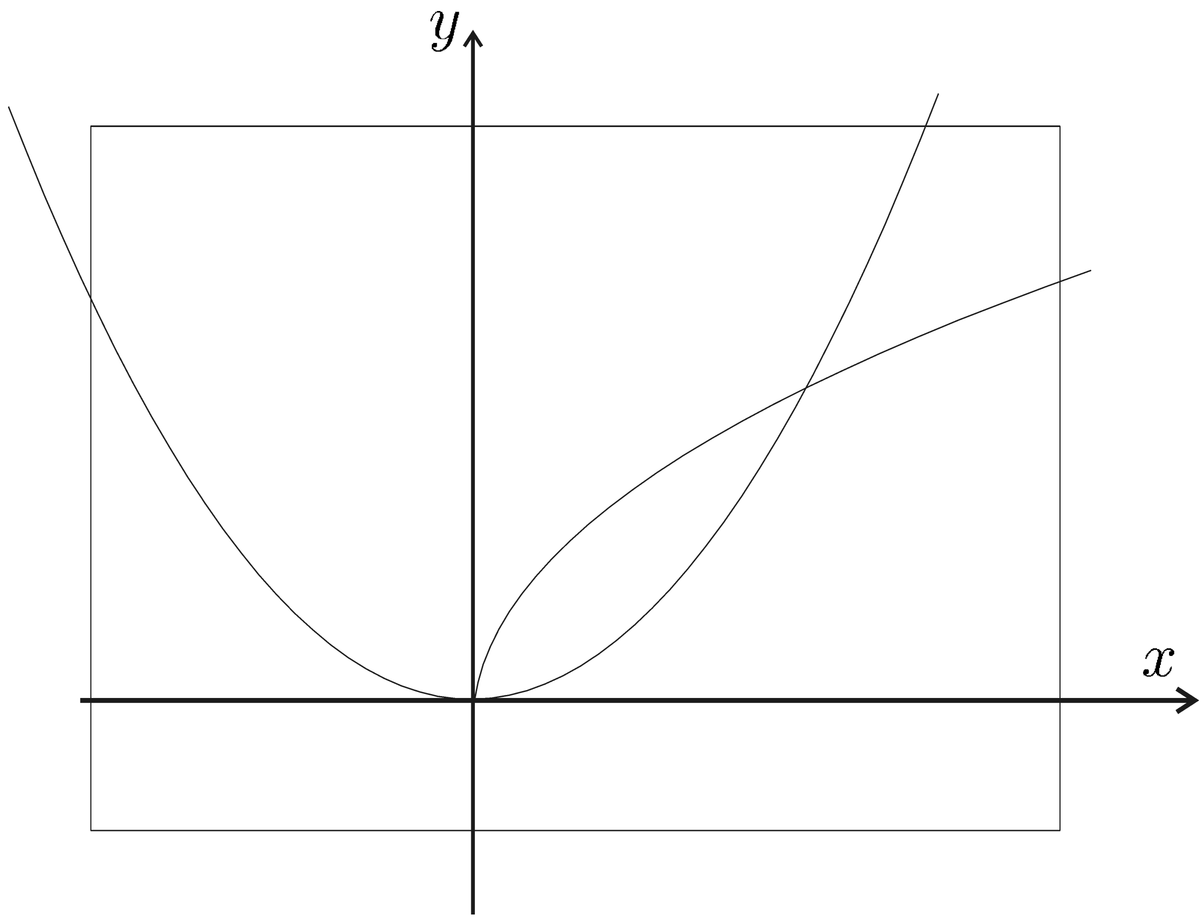
Example. Consider the system.

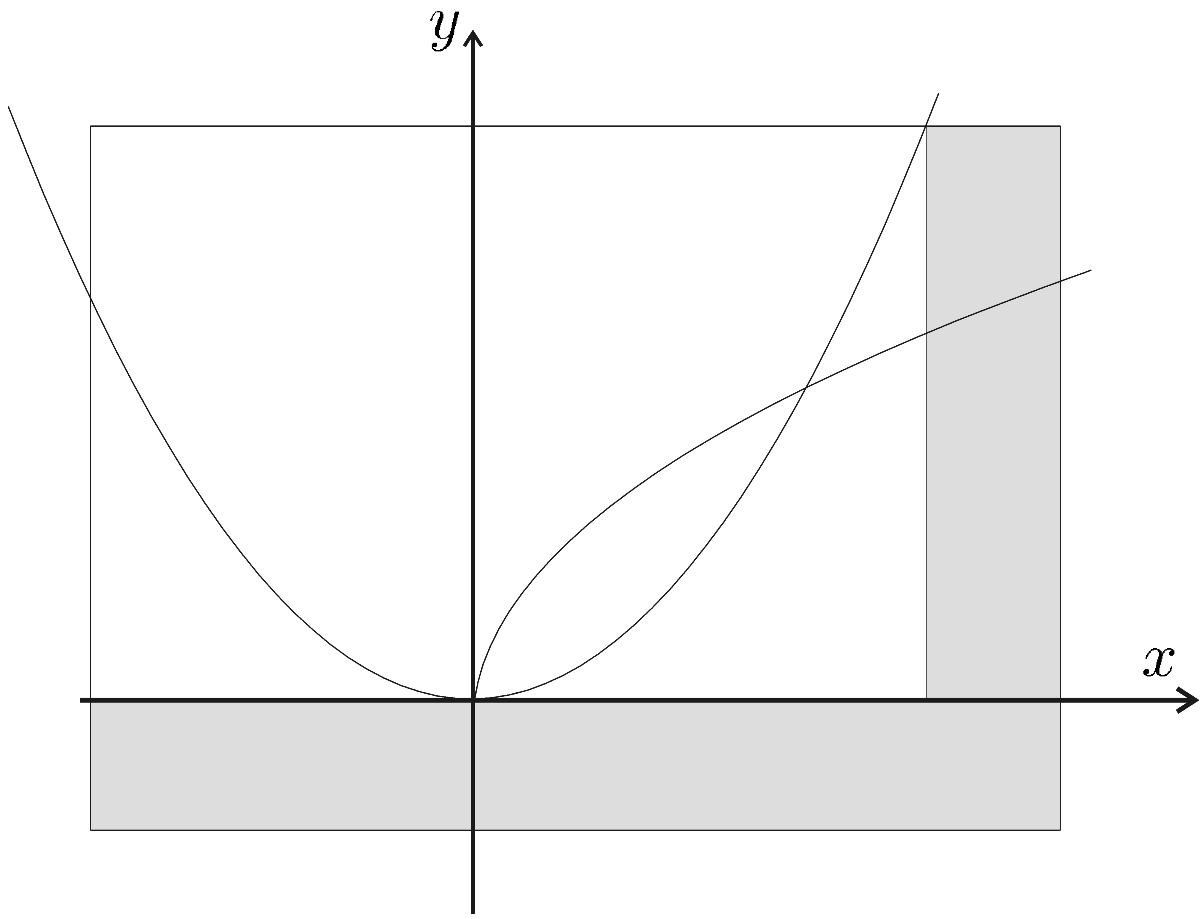
$$\begin{aligned} y &= x^2 \\ y &= \sqrt{x}. \end{aligned}$$

We build two contractors

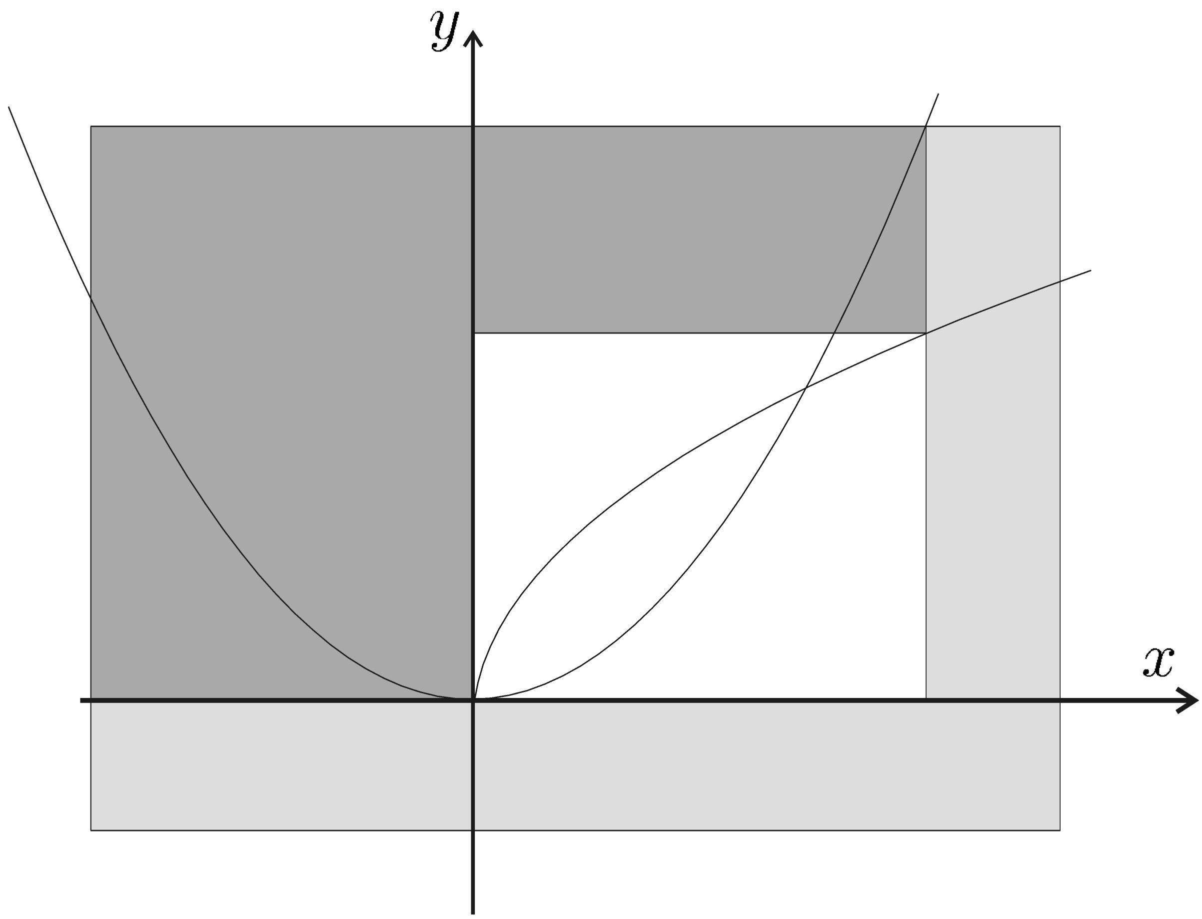
$$\mathcal{C}_1 : \left\{ \begin{array}{l} [y] = [y] \cap [x]^2 \\ [x] = [x] \cap \sqrt{[y]} \end{array} \right. \quad \text{associated to } y = x^2$$

$$\mathcal{C}_2 : \left\{ \begin{array}{l} [y] = [y] \cap \sqrt{[x]} \\ [x] = [x] \cap [y]^2 \end{array} \right. \quad \text{associated to } y = \sqrt{x}$$

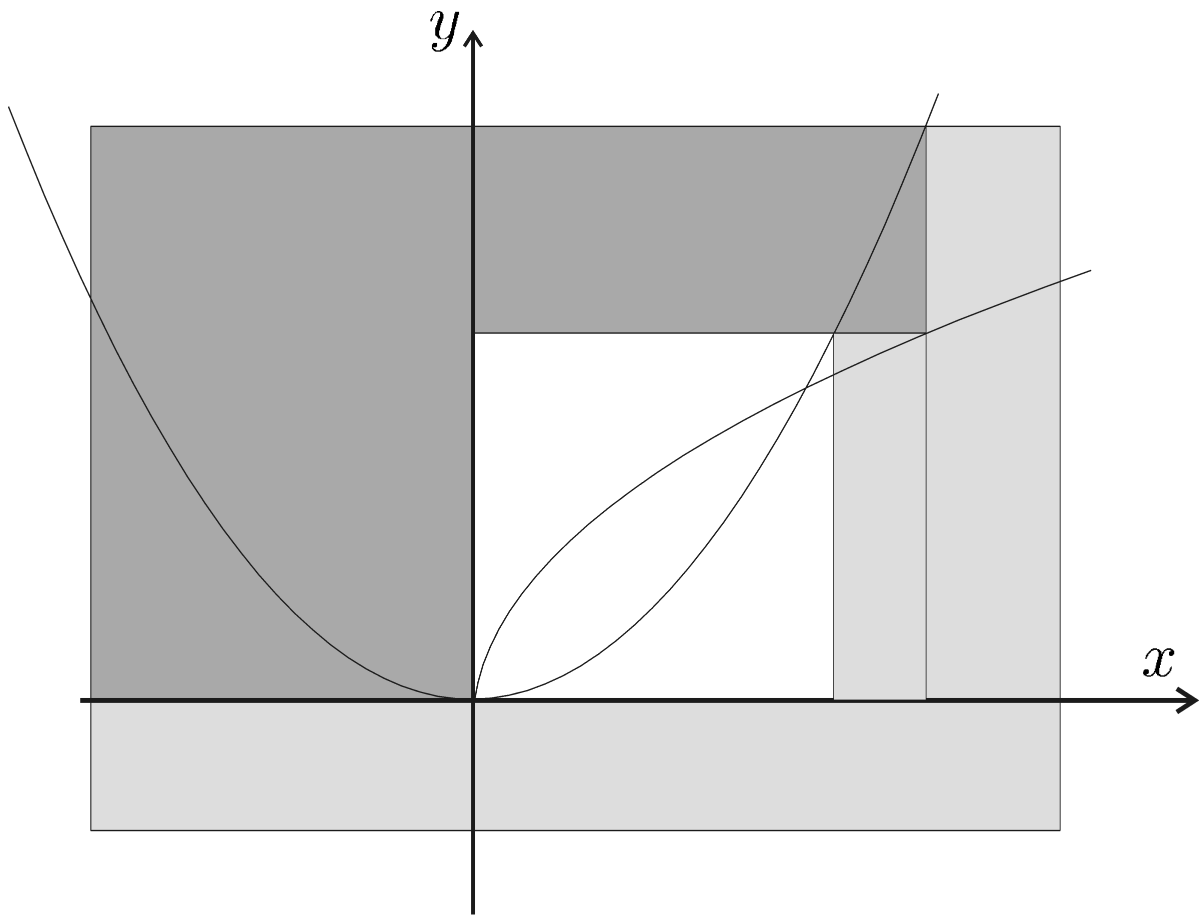

y
x



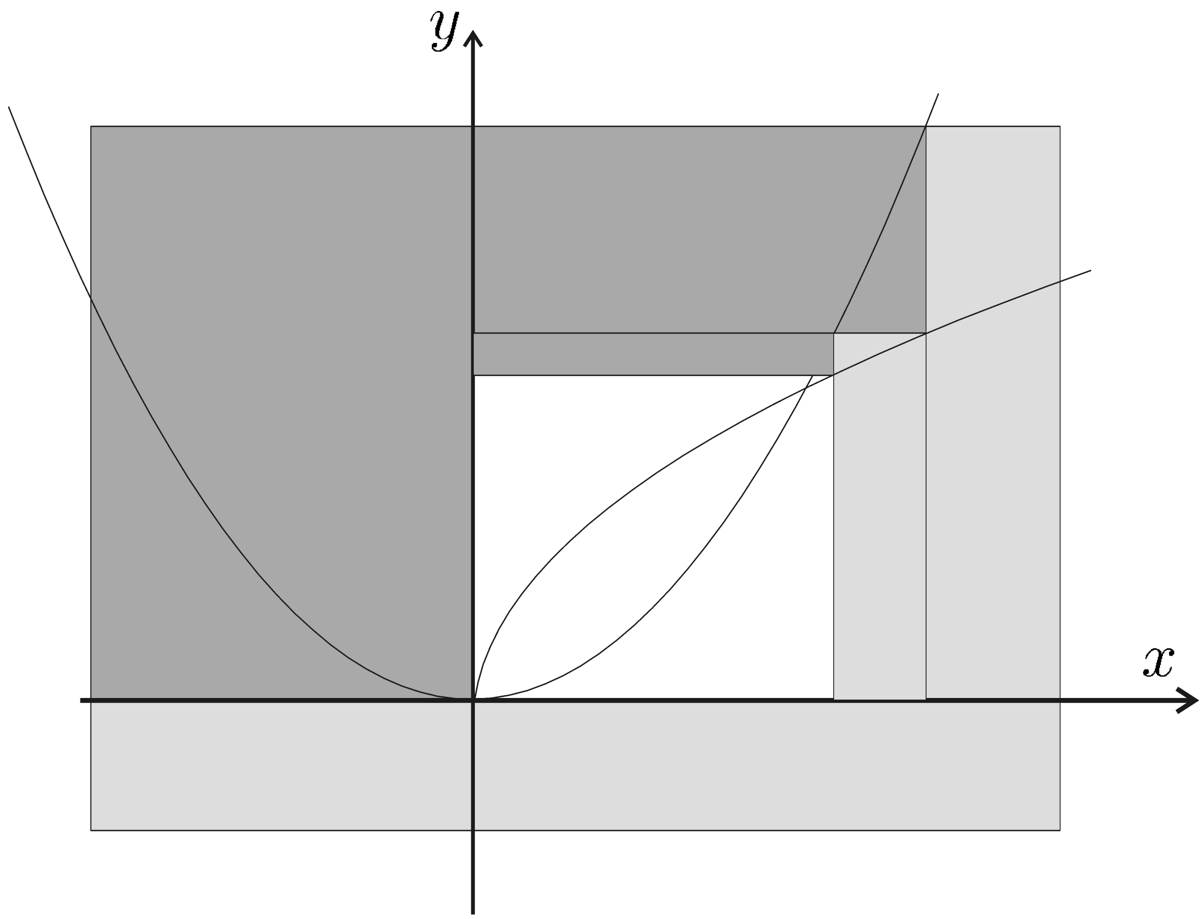
y
x



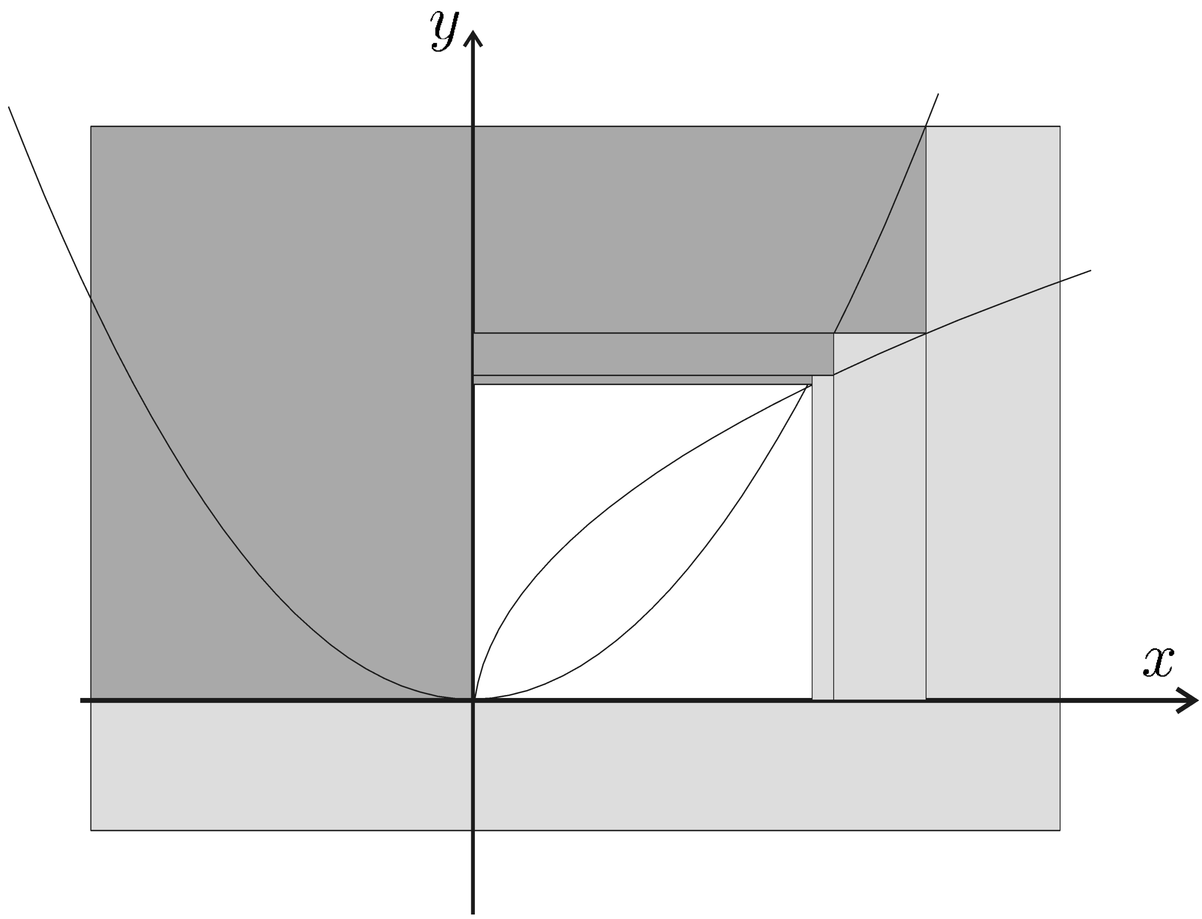
y
x



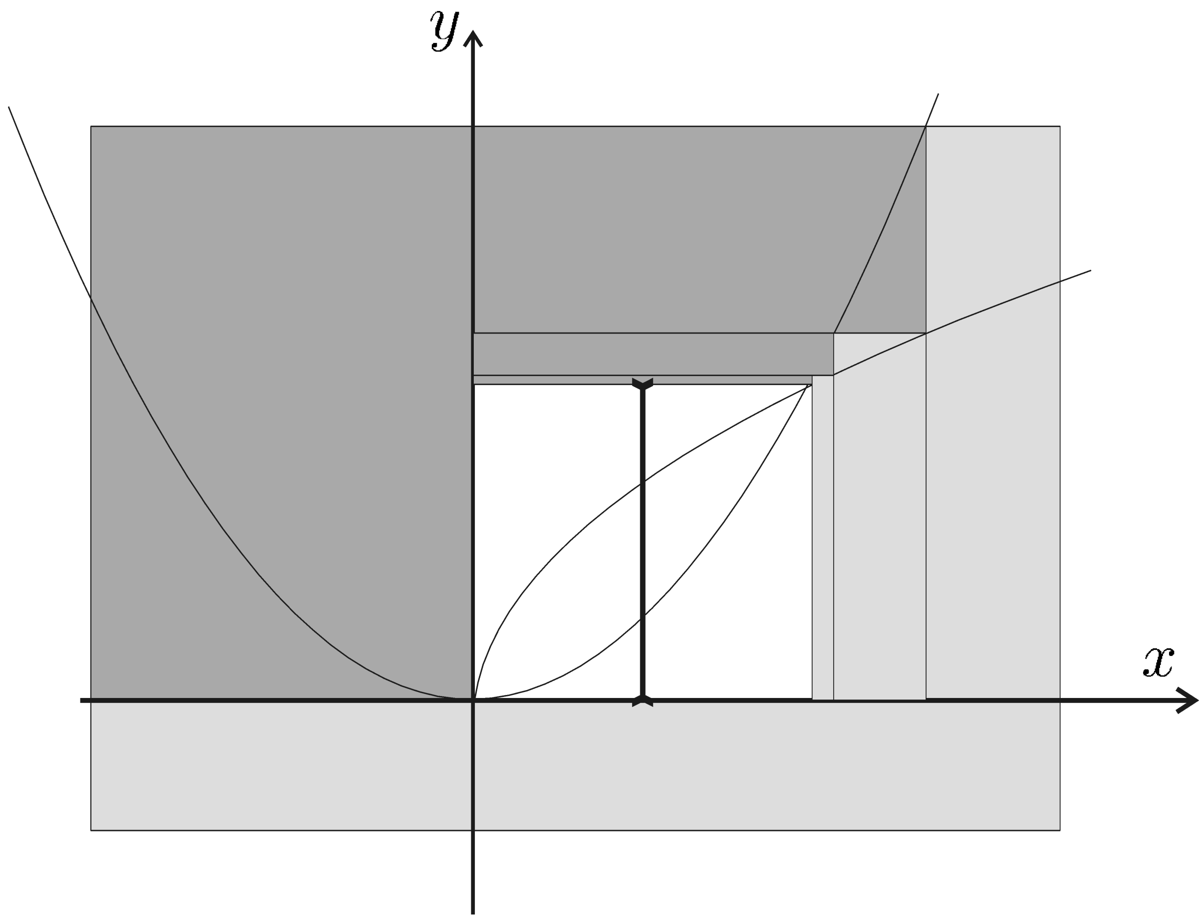
y
x

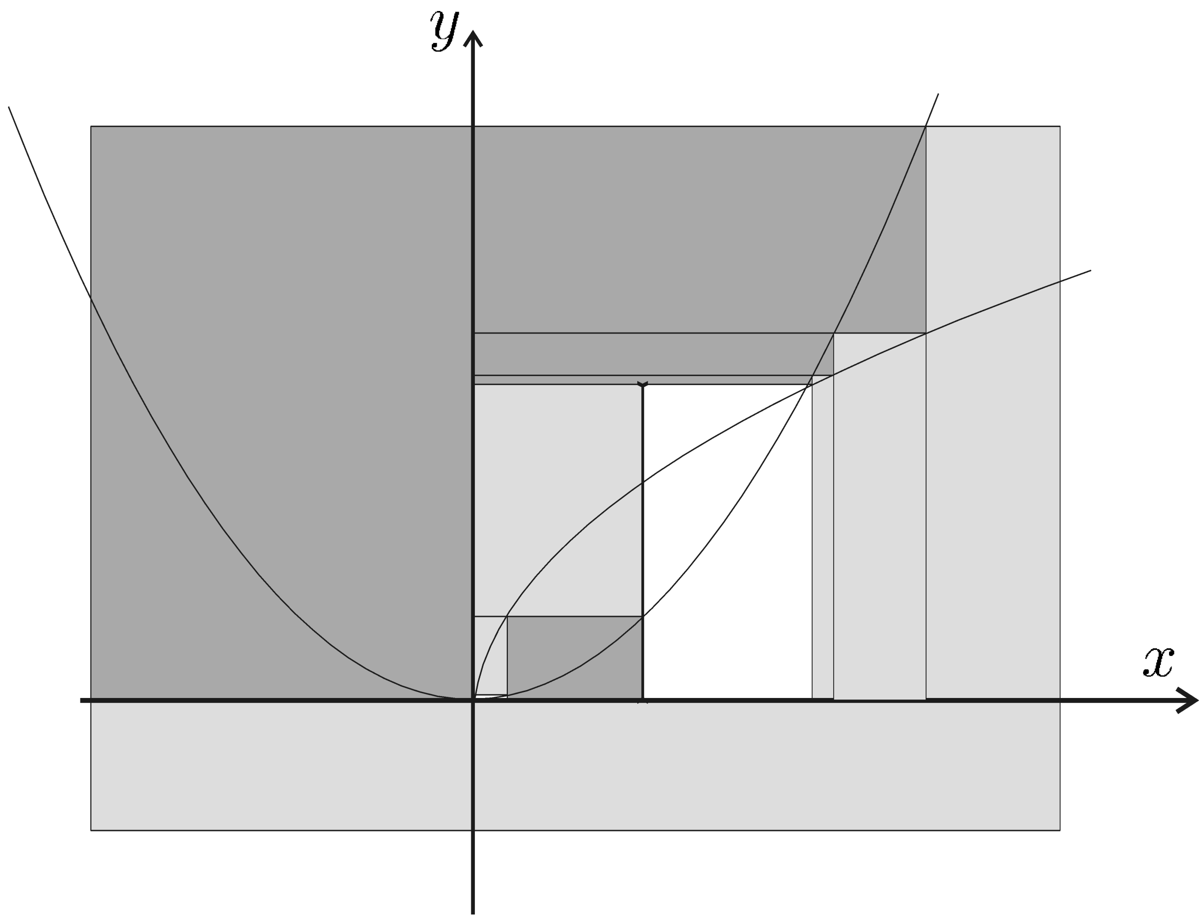


y
x

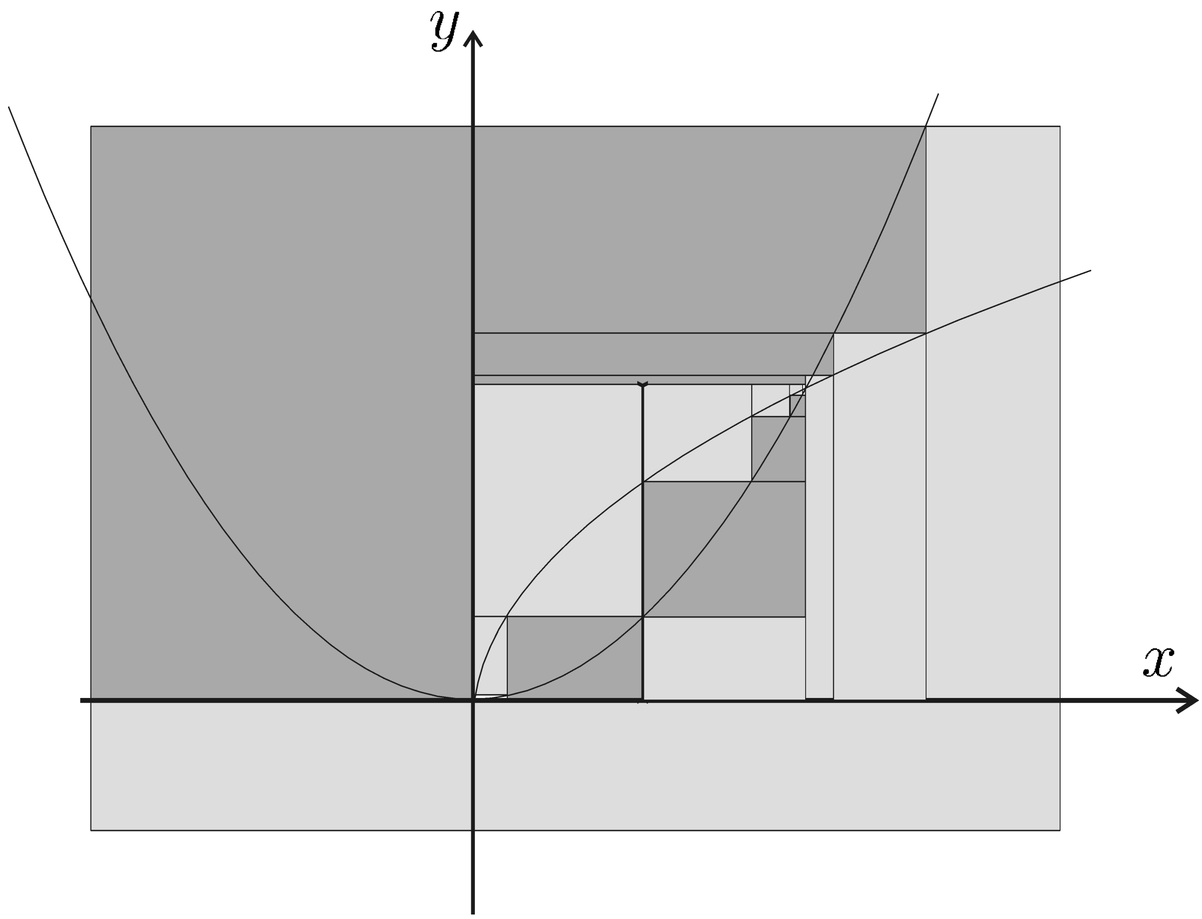


y
x


y
x



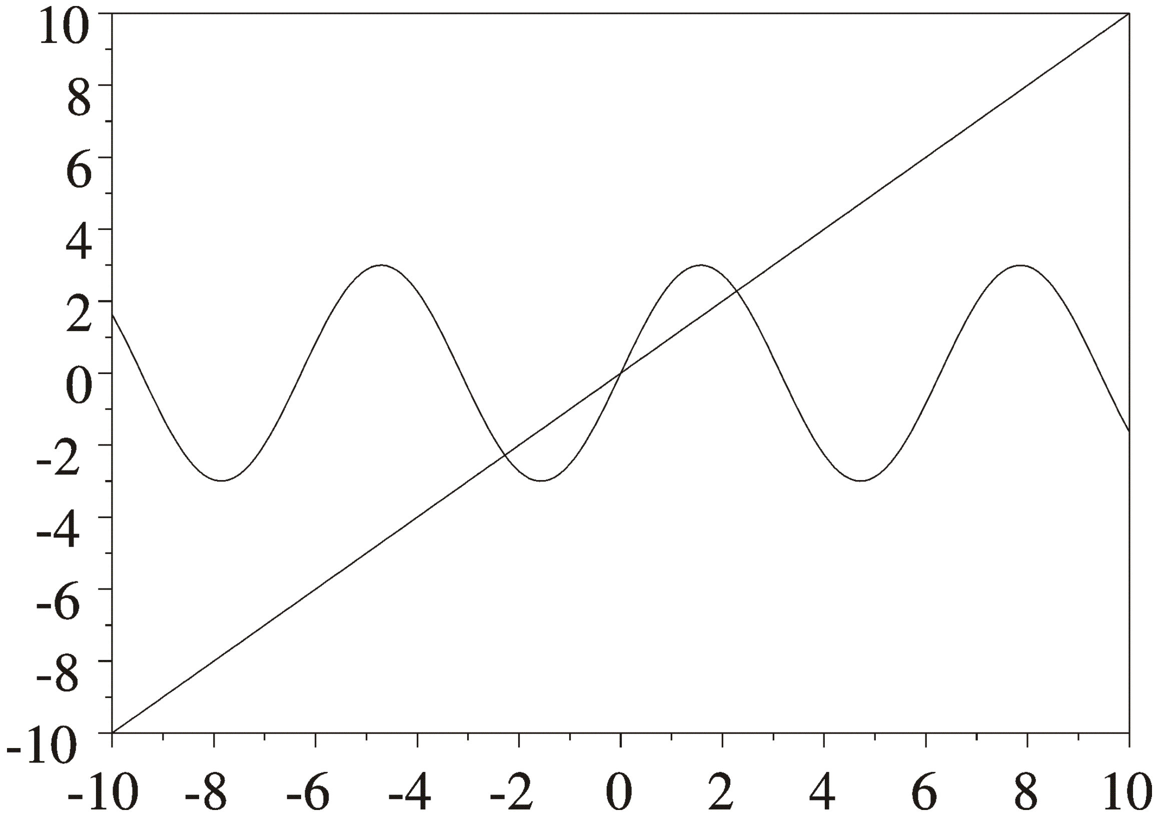
y
x

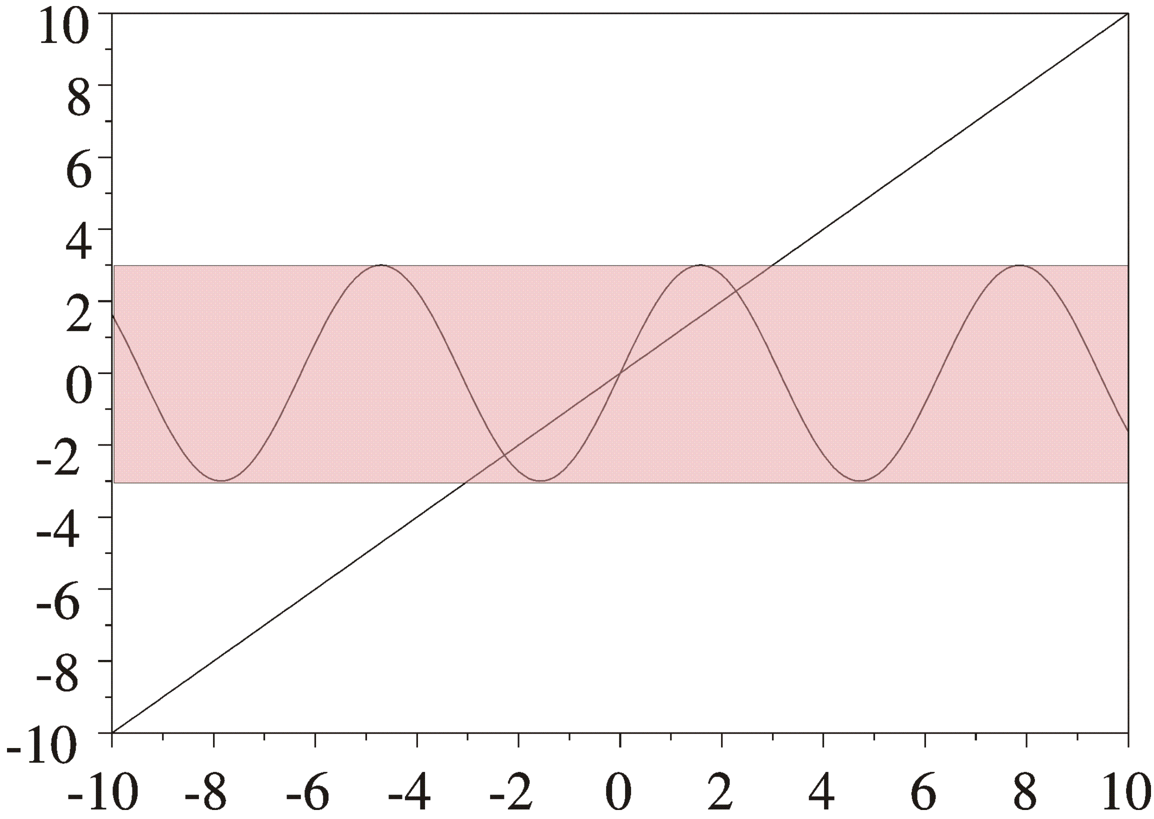


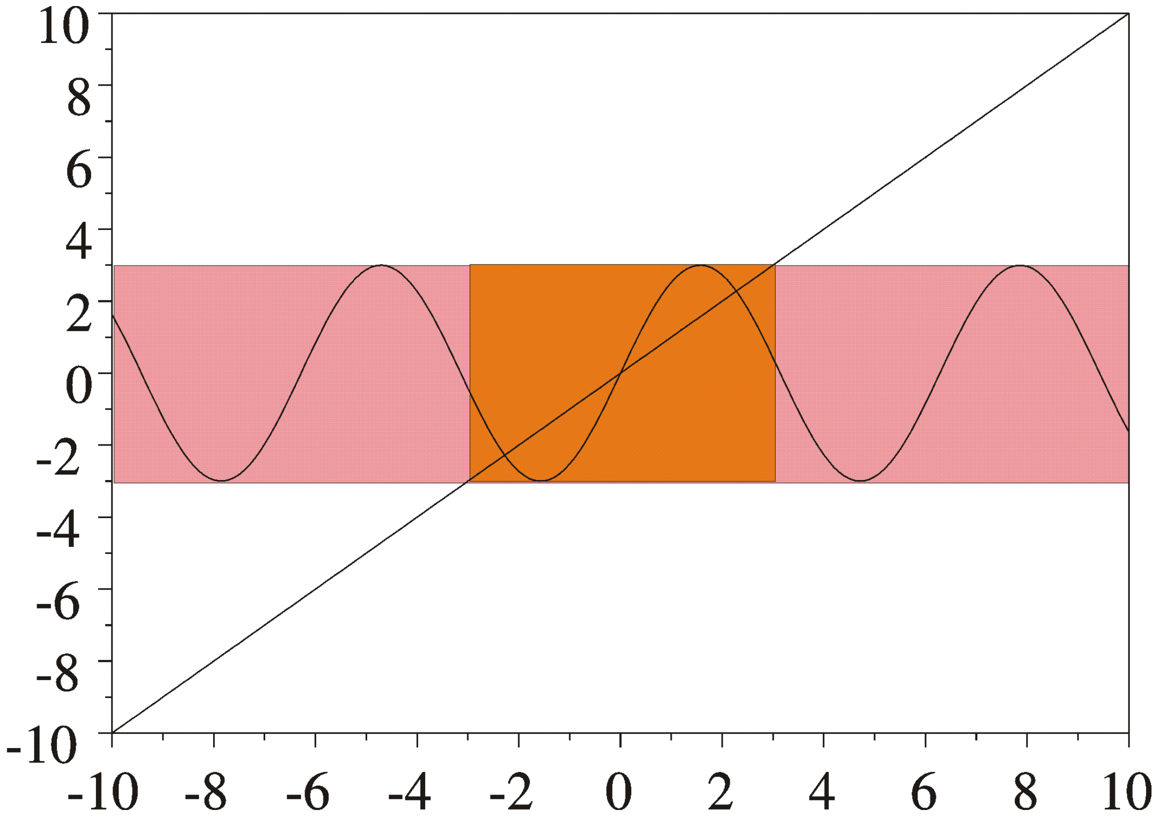
y
x

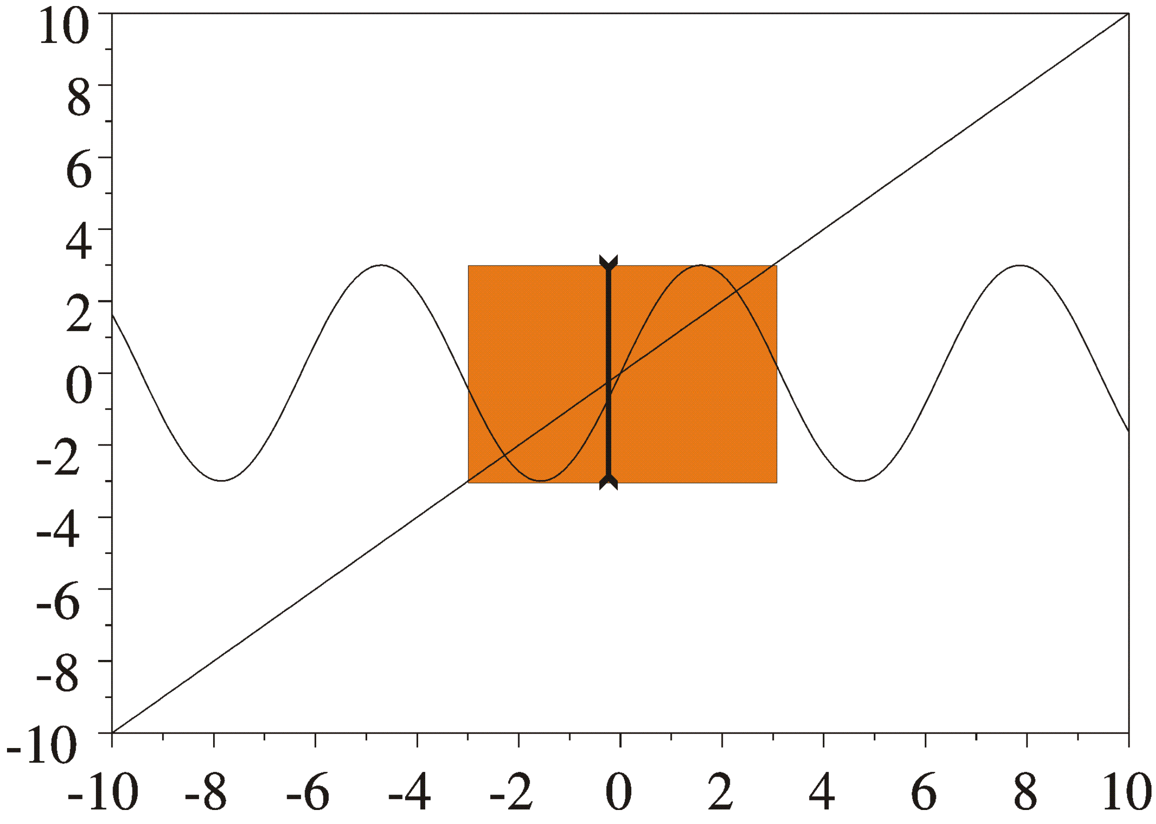
Exemple. Consider the system

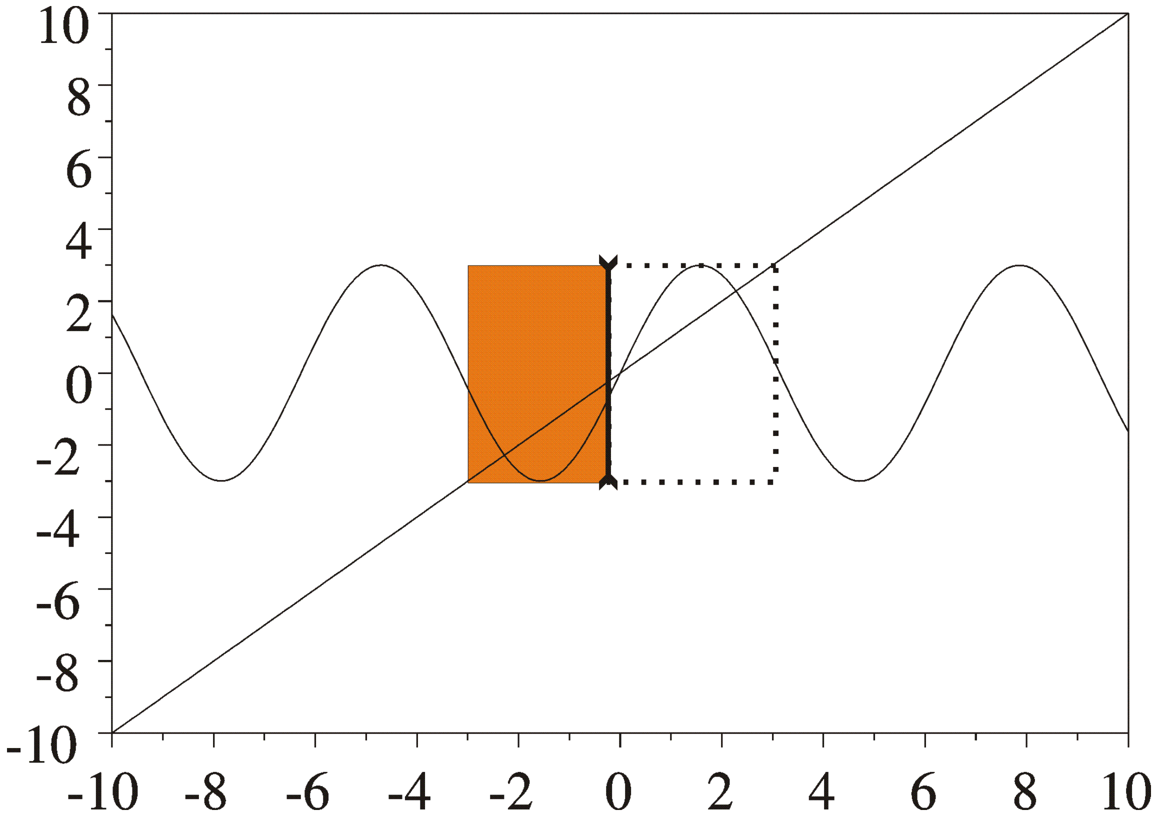
$$\left\{ \begin{array}{lcl} y & = & 3\sin(x) \\ y & = & x \end{array} \right. \qquad x \in \mathbb{R},\ y \in \mathbb{R}.$$

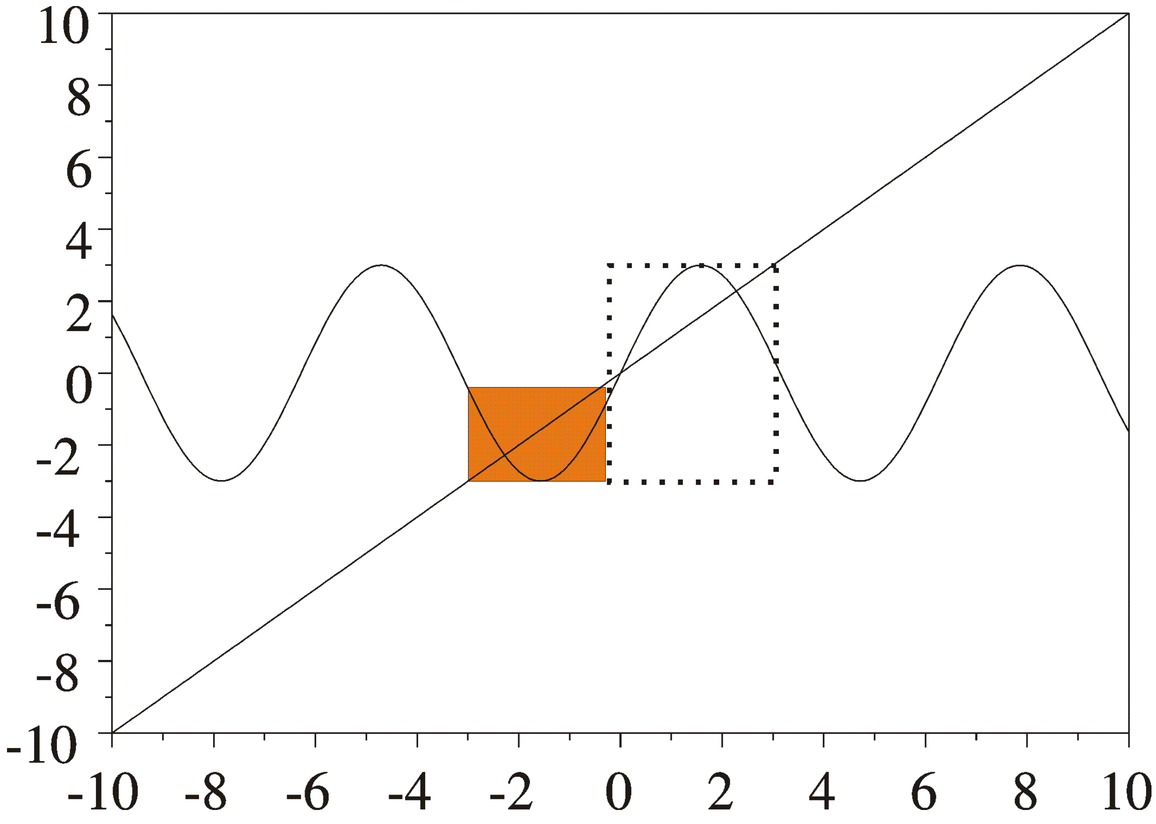

10
8
6
4
2
0
-2
-4
-6
-8
-10
-10 -8 -6 -4 -2 0 2 4 6 8 10


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

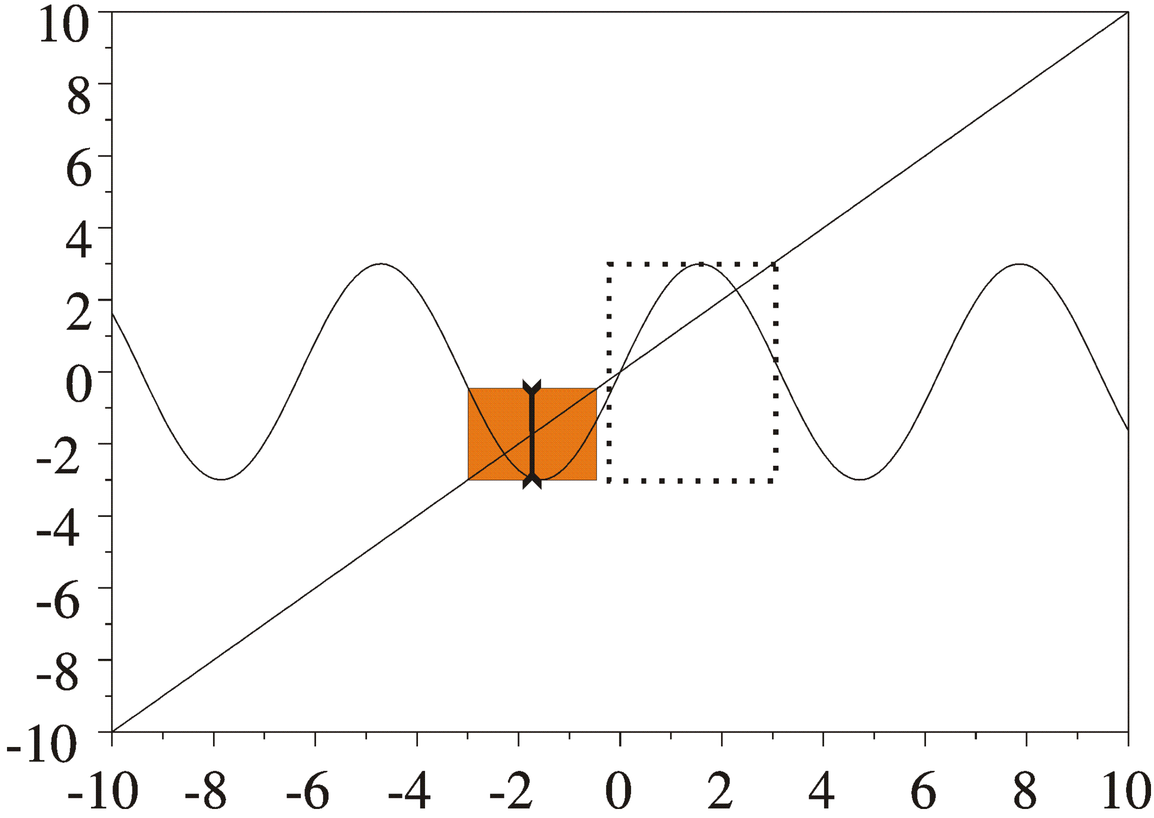

10
8
6
4
2
0
-2
-4
-6
-8
-10
-10
-8
-6
-4
-2
0
2
4
6
8
10


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

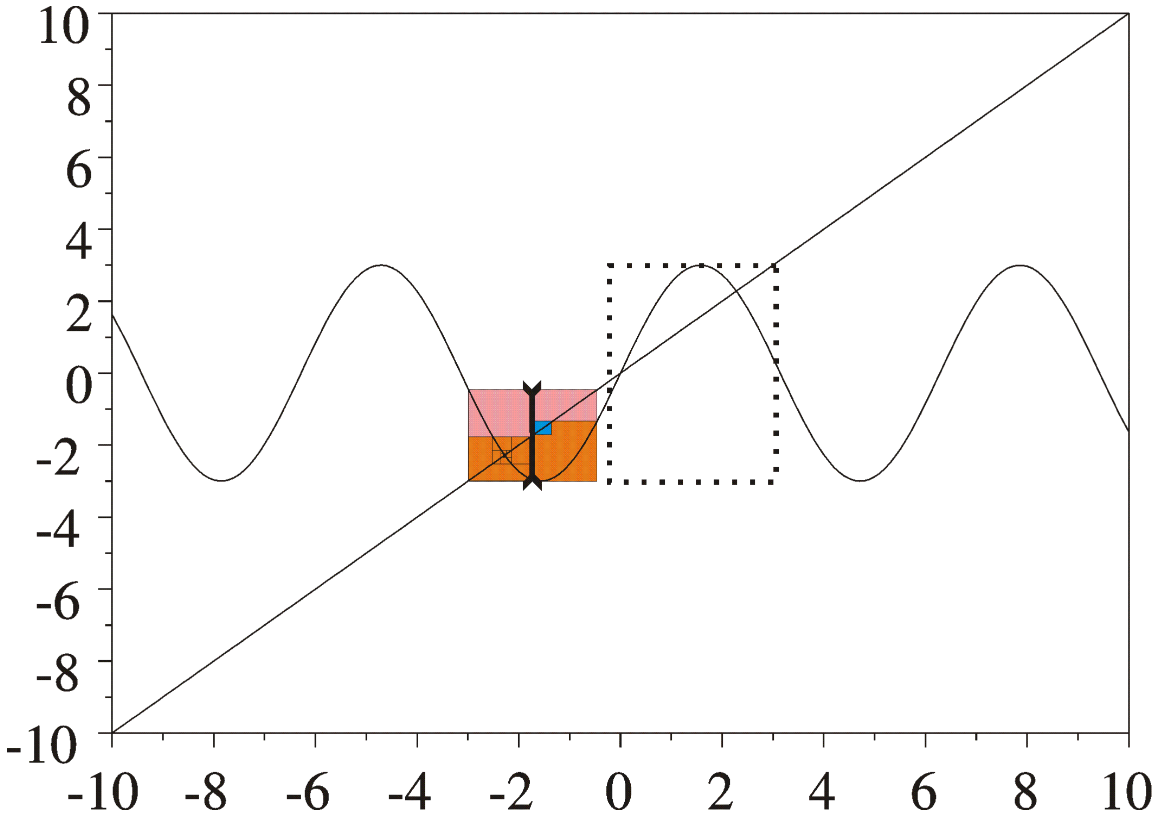

10
8
6
4
2
0
-2
-4
-6
-8
-10
-10
-8
-6
-4
-2
0
2
4
6
8
10

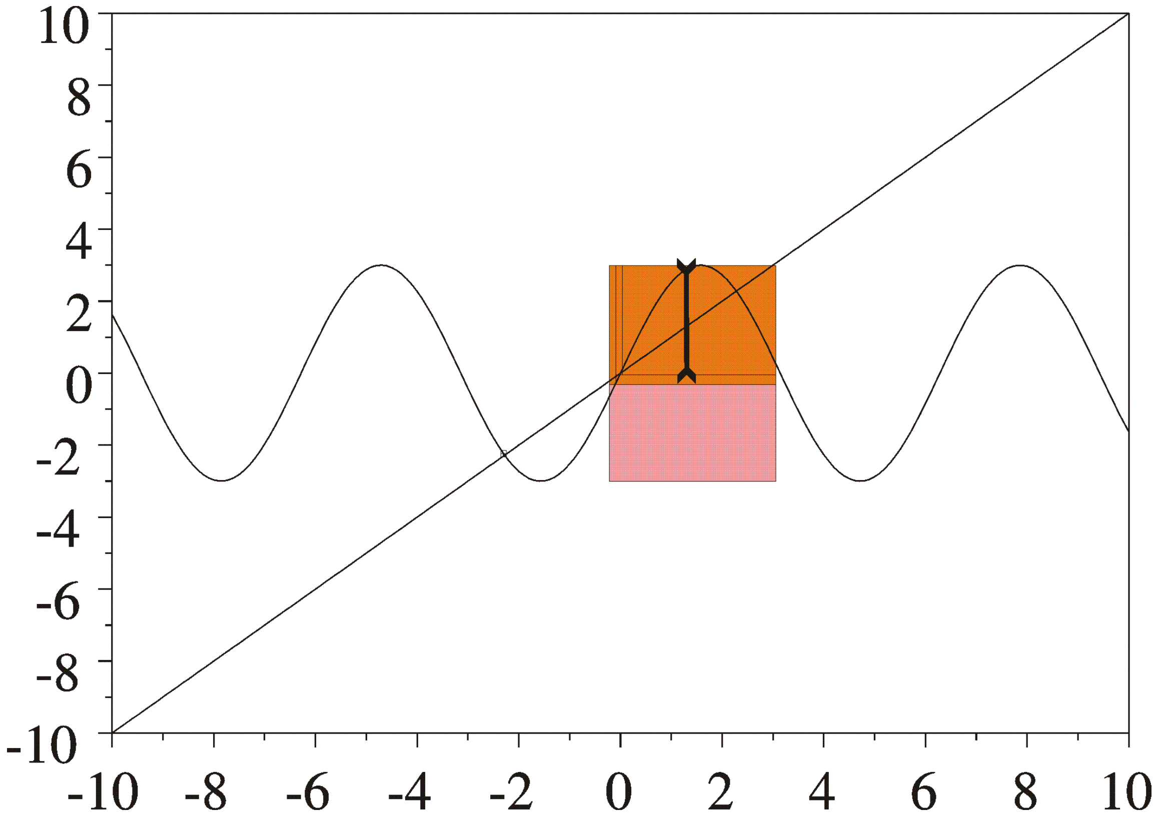


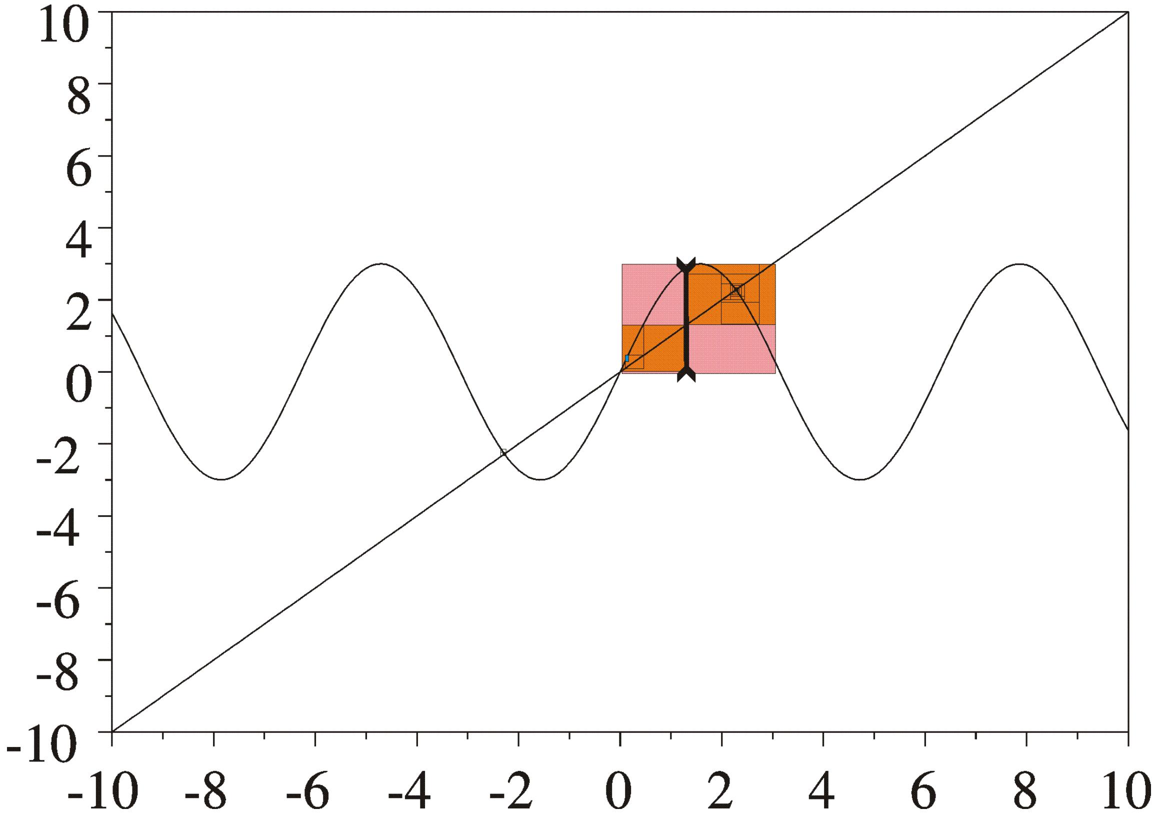
10
8
6
4
2
0
-2
-4
-6
-8
-10
-10
-8
-6
-4
-2
0
2
4
6
8
10



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

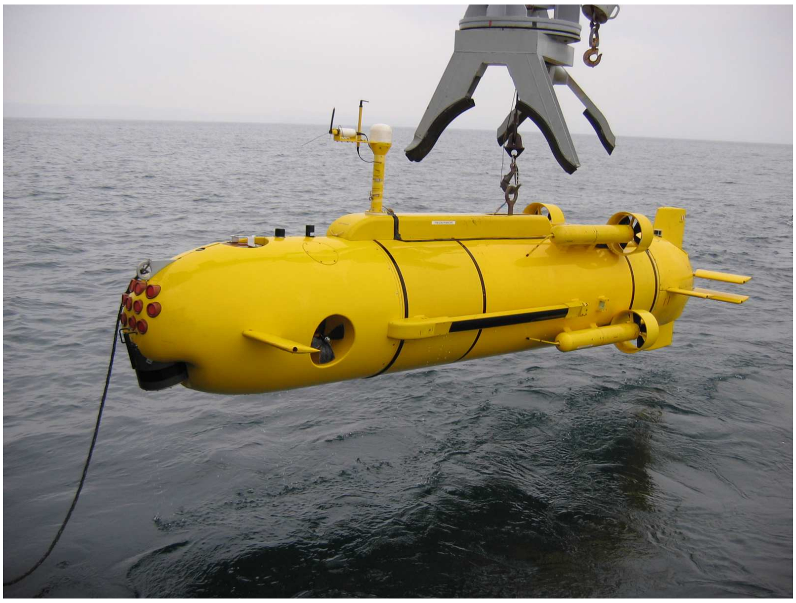
2.4 Decomposition into primitive constraints

$$
\begin{aligned}
&x + \mathsf{sin}(xy) \leq 0, \\
&x \in [-1, 1], y \in [-1, 1]
\end{aligned}
$$

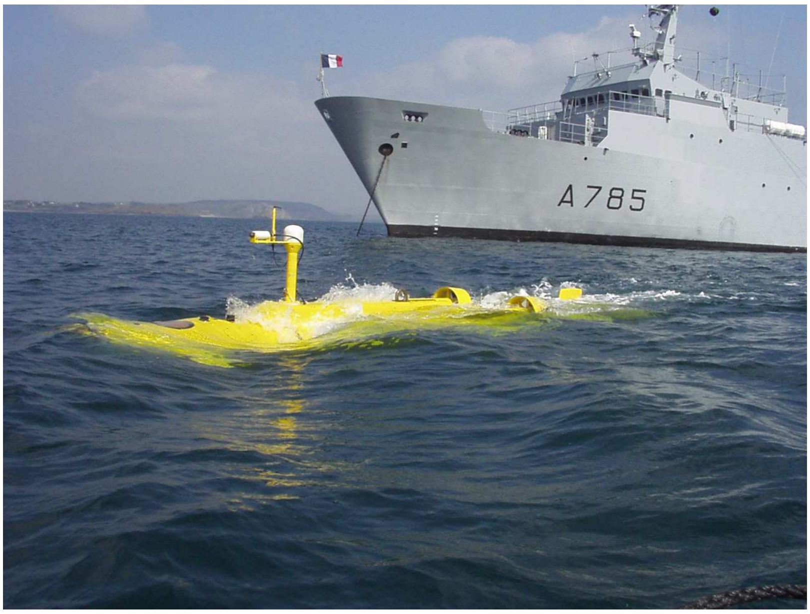
can be decomposed into

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

3 Redermor



The *Redermor*, GESMA



The *Redermor* at the surface

Why choosing an interval constraint approach for SLAM ?

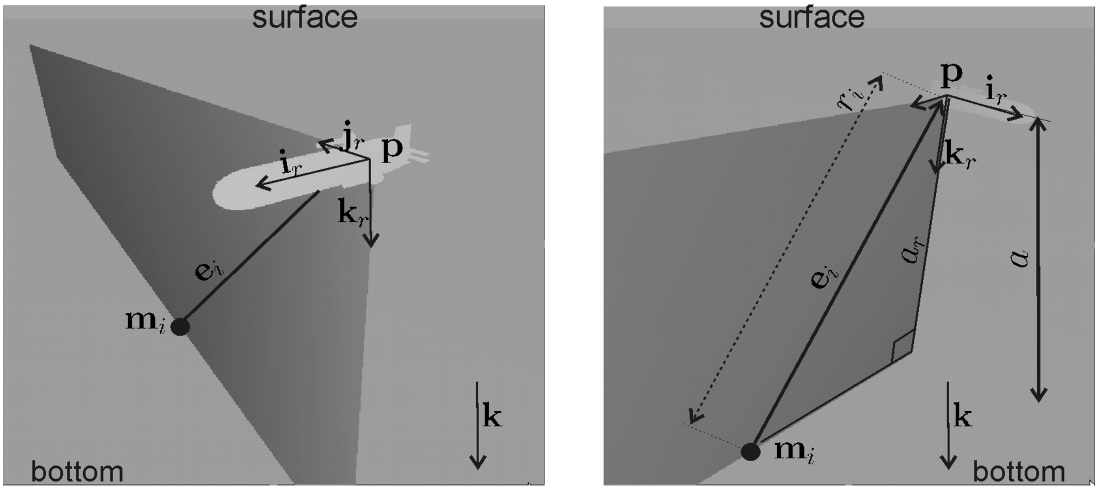
1) A reliable method is needed.
2) The model is nonlinear.
3) The pdf of the noises are unknown.
4) Reliable error bounds are provided by the sensors.
5) A huge number of redundant data are available.

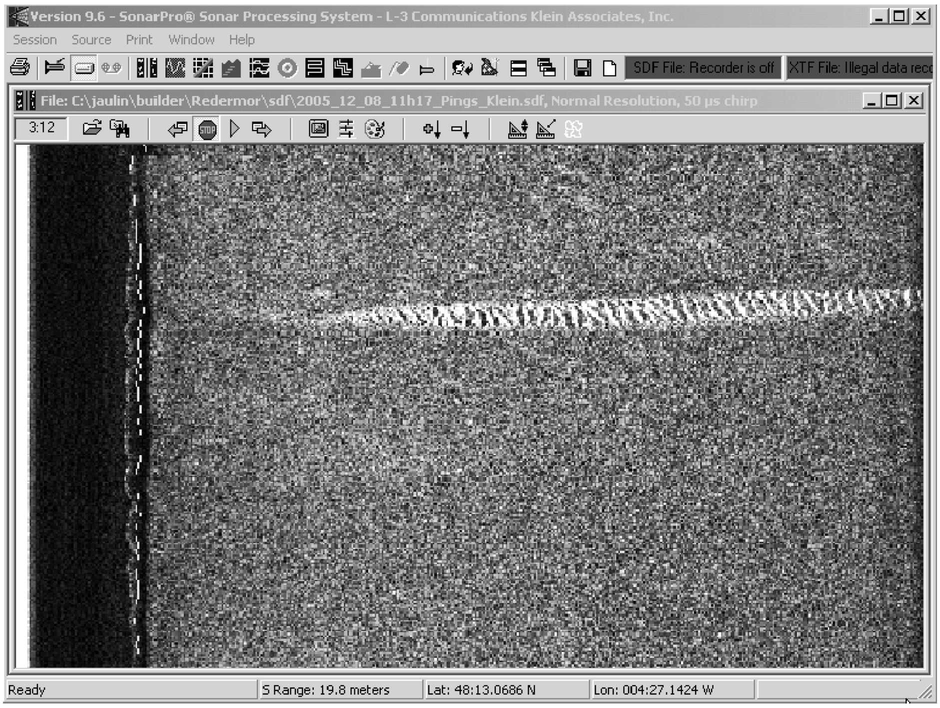
3.1 Sensors

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

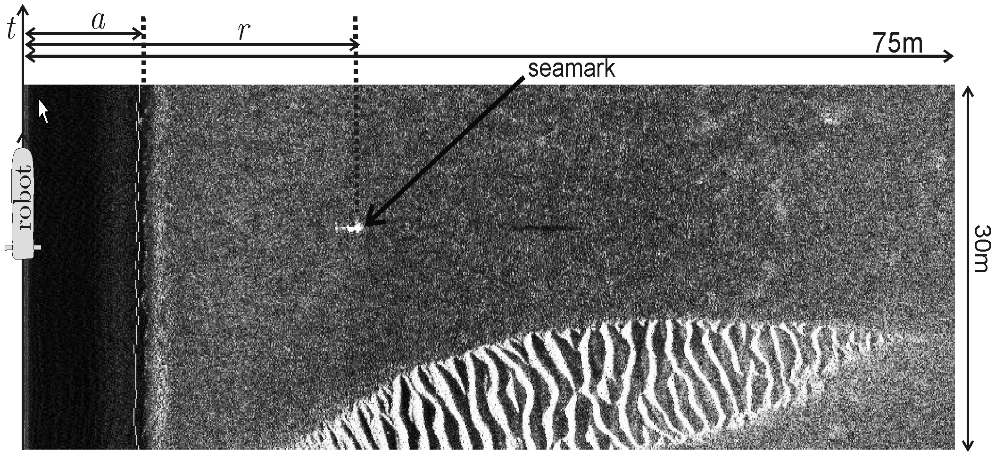
$$t_0 = 6000 \text{ s}, \quad \ell^0 = (-4.4582279^\circ, 48.2129206^\circ) \pm 2.5m$$
$$t_f = 12000 \text{ s}, \quad \ell^f = (-4.4546607^\circ, 48.2191297^\circ) \pm 2.5m$$

A sonar (KLEIN 5400 side scan sonar). Gives the distance r between the robot to the detected object.





Screenshot of SonarPro



Detection of a mine using SonarPro

A Loch-Doppler. Returns the speed of the robot $\mathbf{v}_r$ and the altitude a of the robot $\pm$ 10cm.

A Gyrocompass (Octans III from IXSEA). Returns the roll ϕ, the pitch θ and the head ψ.

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

3.2 Data

For each time $t \in \{6000.0, 6000.1, 6000.2, \ldots, 11999.4\}$, we get intervals for

$$\phi(t), \theta(t), \psi(t), v_r^x(t), v_r^y(t), v_r^z(t), a(t).$$

Six mines have been detected by the sonar:

i	0	1	2	3	4	5	6	7	8	9	10	11
$\tau(i)$	7054	7092	7374	7748	9038	9688	10024	10817	11172	11232	11279	11688
$\sigma(i)$	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

3.3 Constraints satisfaction problem

$$t \in \{6000.0, 6000.1, 6000.2, \ldots, 11999.4\},$$

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

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

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

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

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

$$\mathbf{R}_\varphi(t) = \left(\begin{array}{ccc} 1 & 0 & 0 \\ 0 & \cos\varphi(t) & -\sin\varphi(t) \\ 0 & \sin\varphi(t) & \cos\varphi(t) \end{array} \right),$$

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

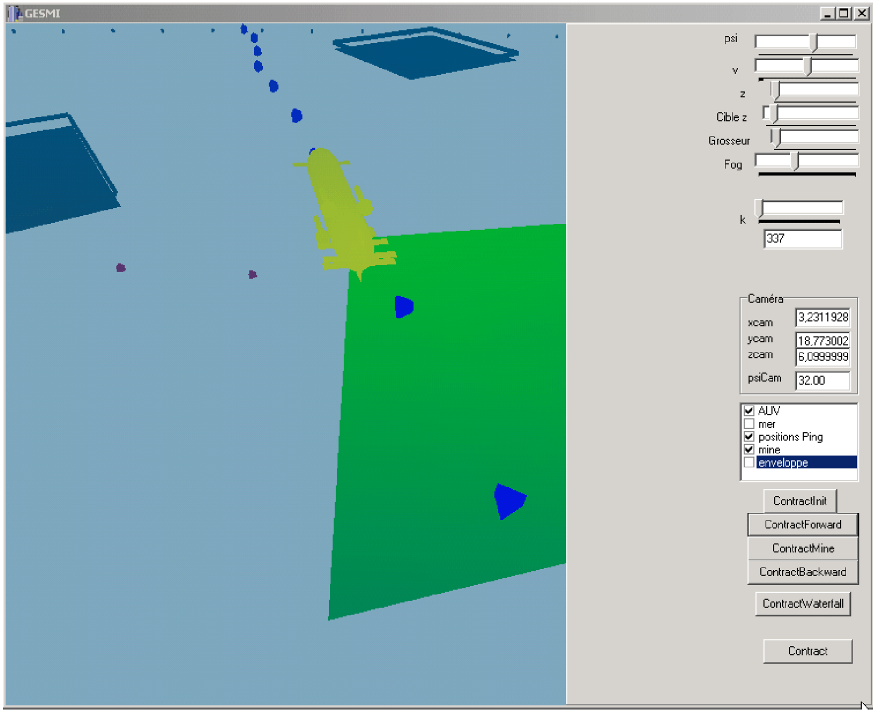
$$\dot{\mathbf{p}}(t) = \mathbf{R}(t).\mathbf{v}_r(t)$$

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

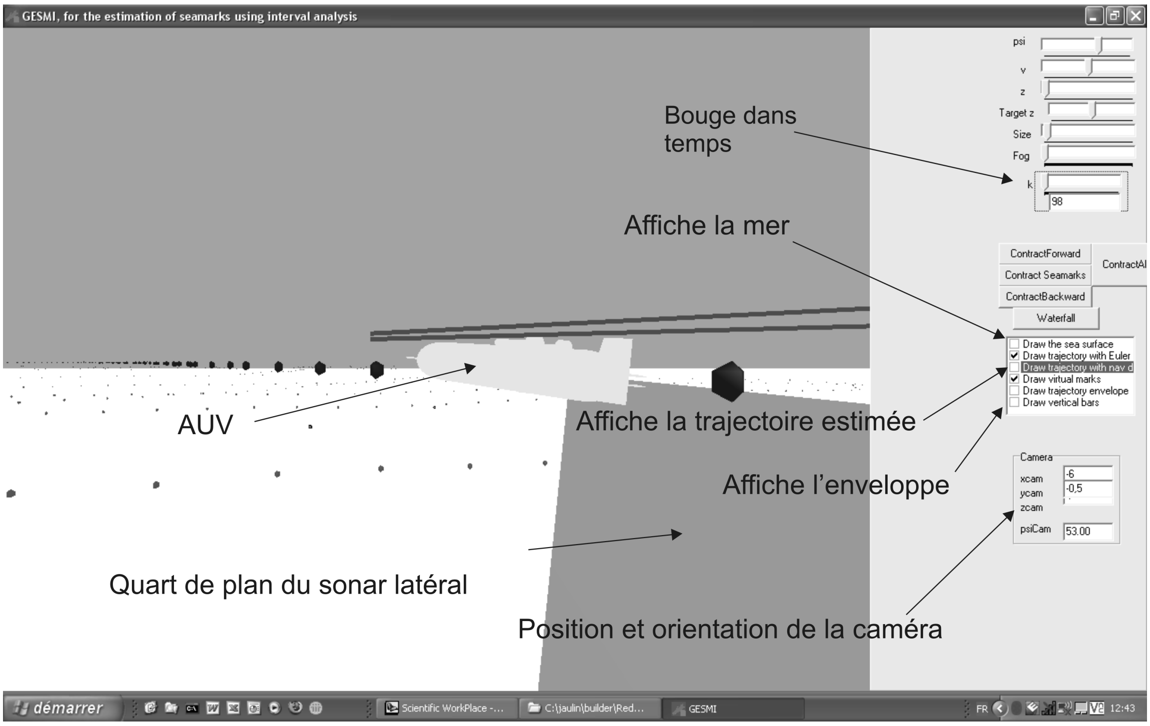
$$\mathbf{R}^{\mathsf{T}}(\tau(i))\,(\mathbf{m}(\sigma(i)) - \mathbf{p}(\tau(i))) \in [0] \times [0, \infty]^{\times 2},$$

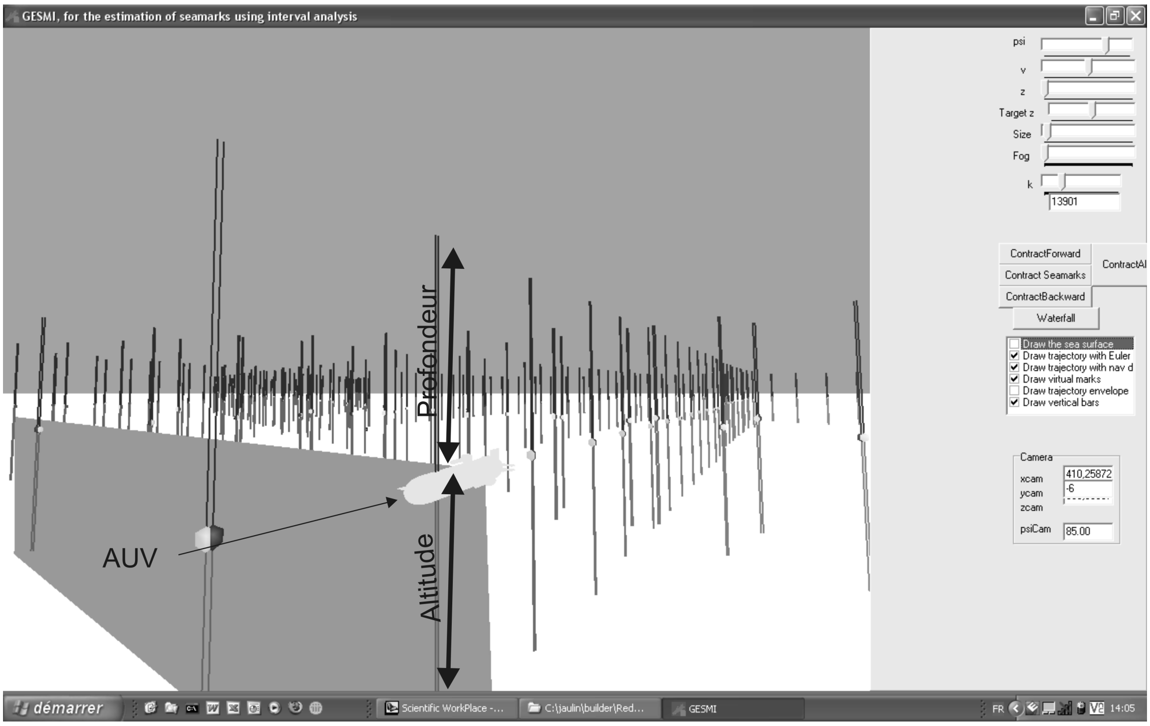
$$m_z(\sigma(i)) - p_z(\tau(i)) - a(\tau(i)) \in [-0.5, 0.5].$$

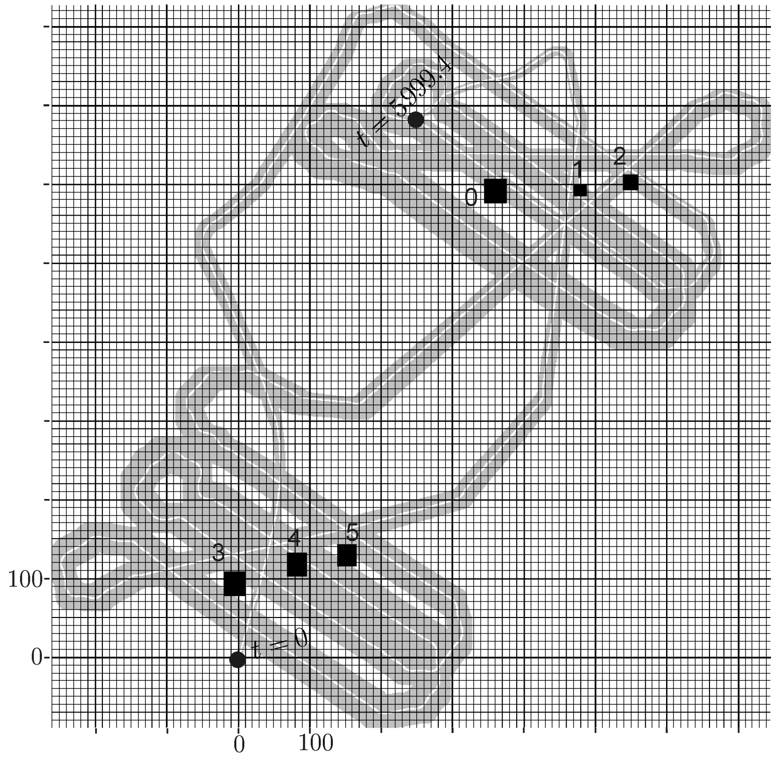
3.4 GESMI



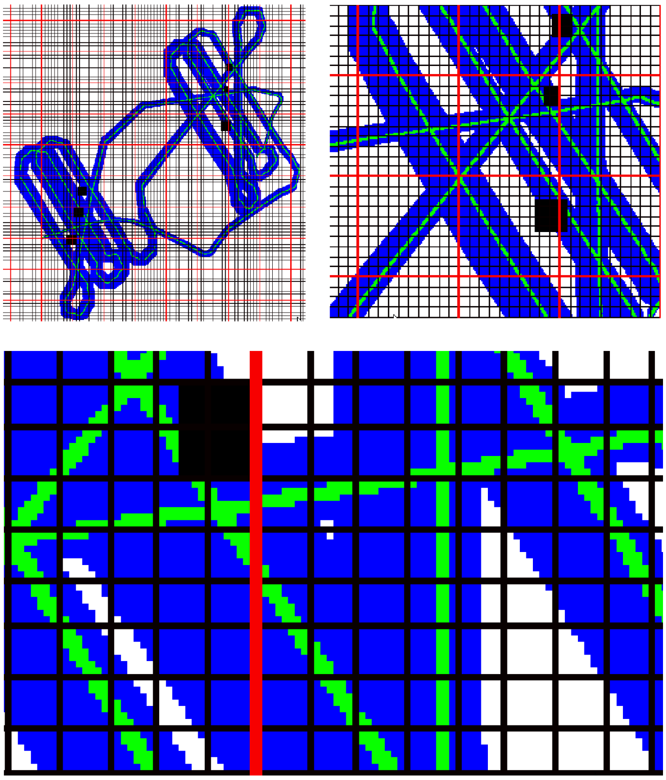
GESMI (Guaranteed Estimation of Sea Mines with Intervals)


GESMI, for the estimation of seamarks using interval analysis
Bouge dans
temps
Affiche la mer
AUV
Affiche la trajectoire estimée
Affiche l’enveloppe
Quart de plan du sonar latéral
Position et orientation de la caméra
psi
v
z
Target z
Size
Fog
k
98
ContractForward
Contract Seamarks
ContractBackward
Waterfall
Draw the sea surface
Draw trajectory with Euler
Draw virtual marks
Draw trajectory envelope
Draw vertical bars
Camera
xcam
ycam
zcam
psiCam
-6
-0,5
53.00
démarrer
Scientific WorkPlace -...
GESMI
FR
12:43


GESMI, for the estimation of seamarks using interval analysis
psi
v
z
Target z
Size
Fog
k
13901
ContractForward
Contract Seamarks
ContractBackward
Waterfall
Draw the sea surface
Draw trajectory with Euler
Draw trajectory with nav d
Draw virtual marks
Draw trajectory envelope
Draw vertical bars
Camera
xcam
410,25872
ycam
-6
zcam
psiCam
85.00
Profondeur
AUV
Altitude
démarrer
Scientific WorkPlace -...
C:\jaulin\builder\Red...
GESMI
FR
14:05

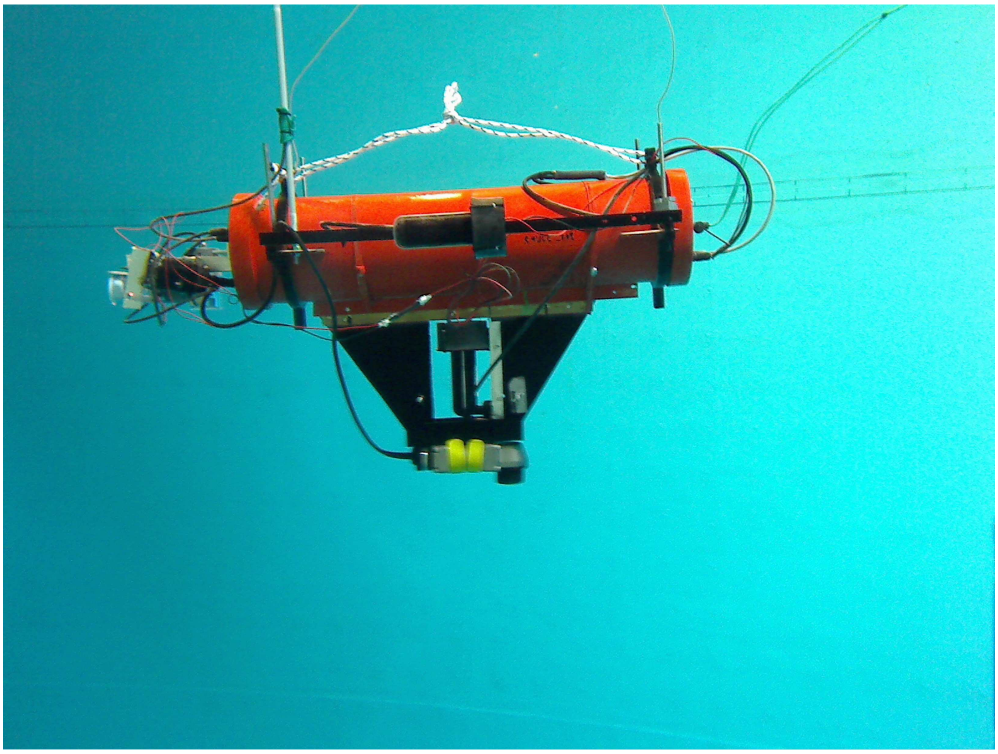


t = 5999.4
0
1
2
3
4
5
t = 0
100
0
0
100



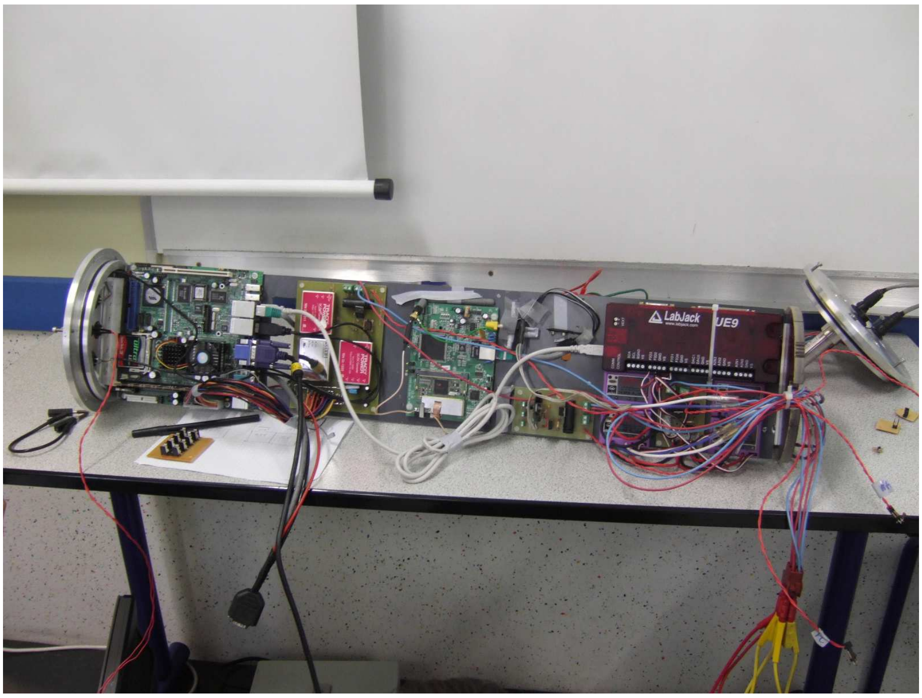
Trajectory reconstructed by GESMI

4 SAUC'ISSE

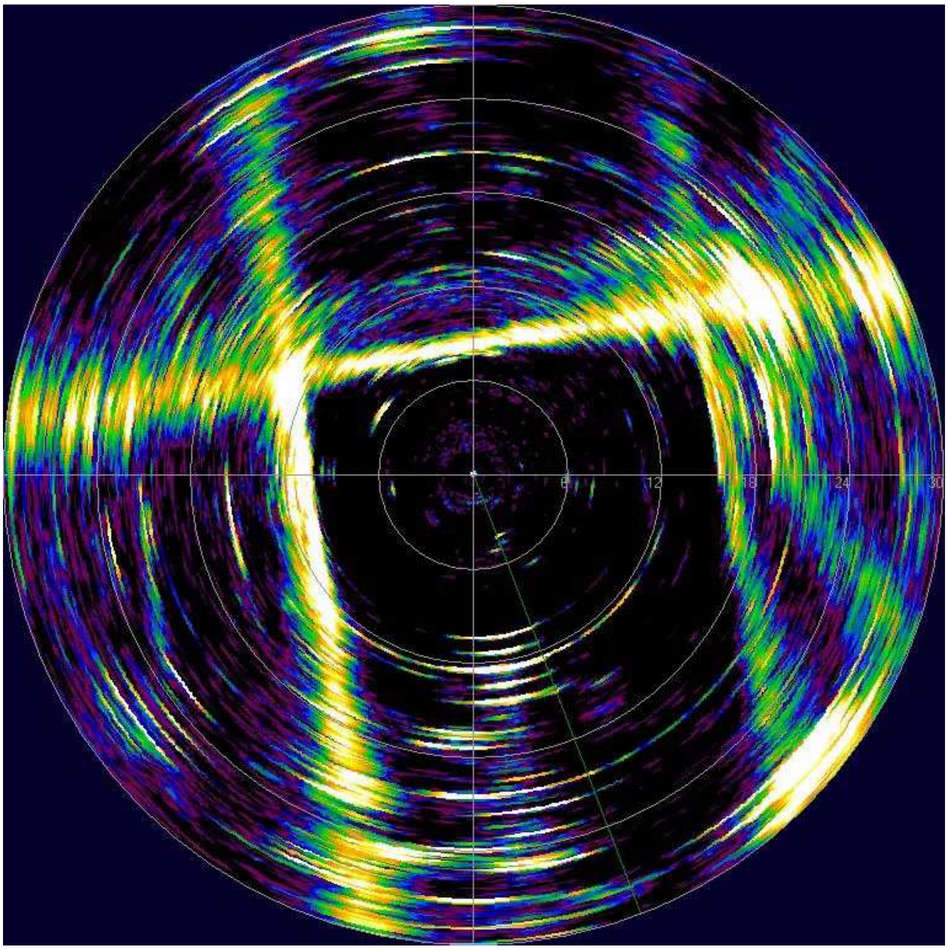


Robot SAUC'ISSE

Portsmouth, July 12-15, 2007.


LabJack
UE9

4.1 Localization with sonar


6
12
18
24
30

4.2 Set-membership approach

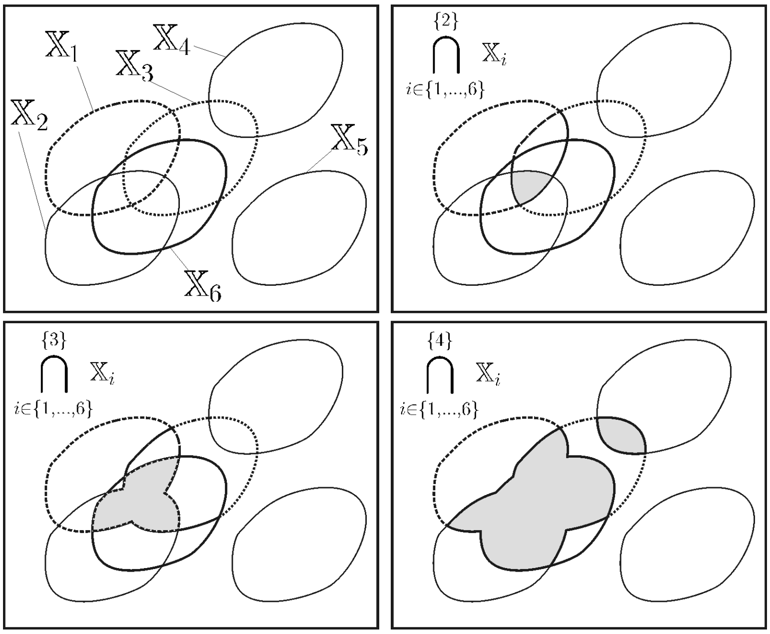
$$\left\{ \begin{array}{lcl} \mathbf{x}(k+1) & = & \mathbf{f}_k(\mathbf{x}(k), \mathbf{n}\,(k)) \\ \mathbf{y}(k) & = & \mathbf{g}_k(\mathbf{x}(k)), \end{array} \right.$$

with $\mathbf{n}\,(k) \in \mathbb{N}\,(k)$ and $\mathbf{y}\,(k) \in \mathbb{Y}\,(k)$.

Without outliers

$$\mathbb{X}(k+\mathbf{1}) = \mathbf{f}_k\left(\mathbb{X}(k) \cap \mathbf{g}_k^{-1}\left(\mathbb{Y}(k)\right),\ \ \mathbb{N}\left(k\right)\right).$$

4.3 Relaxed intersection


$\mathbb{X}_1$
$\mathbb{X}_2$
$\mathbb{X}_3$
$\mathbb{X}_4$
$\mathbb{X}_5$
$\mathbb{X}_6$
$\bigcap_{i\in\{1,\ldots,6\}}^{\{2\}} \mathbb{X}_i$
$\bigcap_{i\in\{1,\ldots,6\}}^{\{3\}} \mathbb{X}_i$
$\bigcap_{i\in\{1,\ldots,6\}}^{\{4\}} \mathbb{X}_i$

4.4 Robust localization

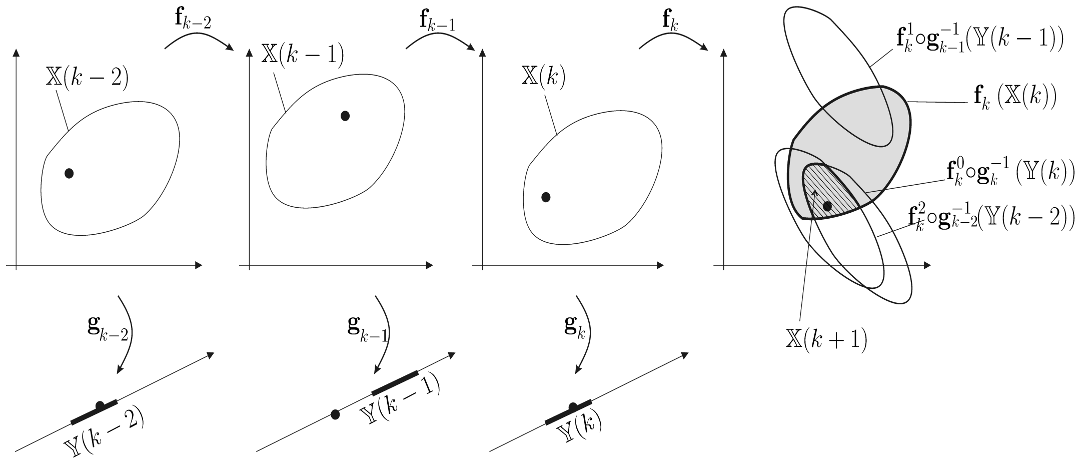
Define

$$\begin{cases} \mathbf{f}_{k:k}(\mathbb{X}) & \stackrel{\text{def}}{=} & \mathbb{X} \\ \mathbf{f}_{k_1:k_2+1}(\mathbb{X}) & \stackrel{\text{def}}{=} & \mathbf{f}_{k_2}(\mathbf{f}_{k_1:k_2}(\mathbb{X}), \mathbb{N}(k_2)), \ k_1 \leq k_2. \end{cases}$$

The set $\mathbf{f}_{k_1:k_2}(\mathbb{X})$ represents the set of all $\mathbf{x}(k_2)$, consistent with $\mathbf{x}(k_1) \in \mathbb{X}$.

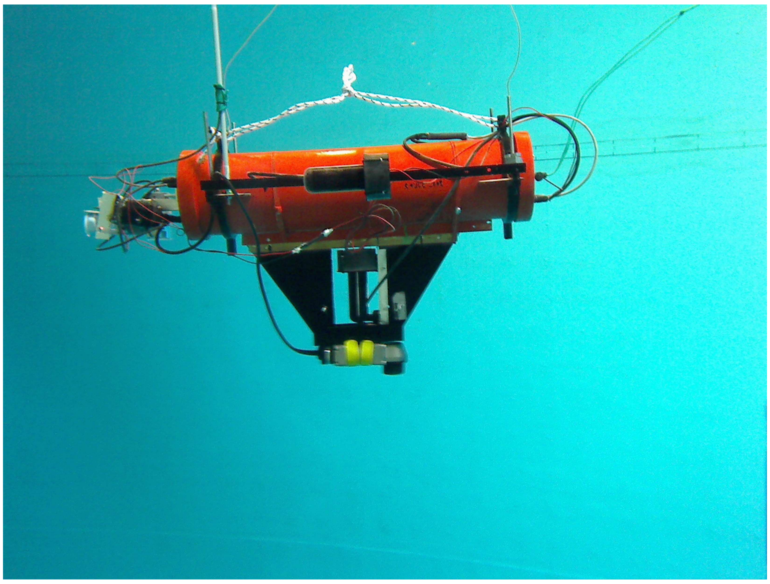
Consider the set state estimator

$$
\begin{cases}
\mathbb{X}(k) = \mathbf{f}_{0:k}(\mathbb{X}(0)) \quad \text{if } k < m, \ \text{(initialization step)} \\
\mathbb{X}(k) = \mathbf{f}_{k-m:k}(\mathbb{X}(k-m)) \cap \\
\qquad \bigcap_{i \in \{1,\dots,m\}}^{\{q\}} \mathbf{f}_{k-i:k} \circ \mathbf{g}_{k-i}^{-1}(\mathbb{Y}(k-i)) \quad \text{if } k \geq m
\end{cases}
$$



$\mathbf{f}_{k-2}$
$\mathbf{f}_{k-1}$
$\mathbf{f}_k$
$\mathbb{X}(k-2)$
$\mathbb{X}(k-1)$
$\mathbb{X}(k)$
$\mathbf{f}_k^1 \circ \mathbf{g}_{k-1}^{-1}(\mathbb{Y}(k-1))$
$\mathbf{f}_k\,(\mathbb{X}(k))$
$\mathbf{f}_k^0 \circ \mathbf{g}_k^{-1}\,(\mathbb{Y}(k))$
$\mathbf{f}_k^2 \circ \mathbf{g}_{k-2}^{-1}(\mathbb{Y}(k-2))$
$\mathbb{X}(k+1)$
$\mathbf{g}_{k-2}$
$\mathbf{g}_{k-1}$
$\mathbf{g}_k$
$\mathbb{Y}(k-2)$
$\mathbb{Y}(k-1)$
$\mathbb{Y}(k)$

4.5 Application to localization



Sauc'isse robot inside a swimming pool

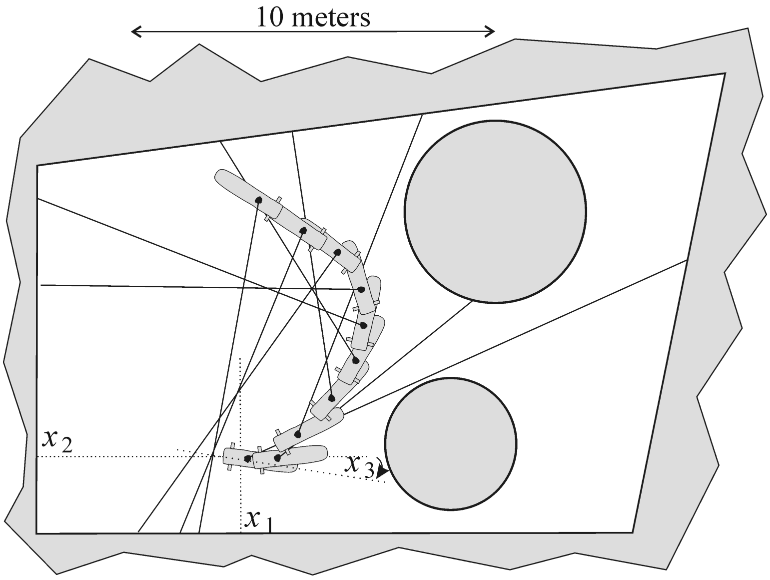
The robot evolution is described by

$$\begin{cases} \dot{x}_1 &= x_4 \cos x_3 \\ \dot{x}_2 &= x_4 \sin x_3 \\ \dot{x}_3 &= u_2 - u_1 \\ \dot{x}_4 &= u_1 + u_2 - x_4, \end{cases}$$

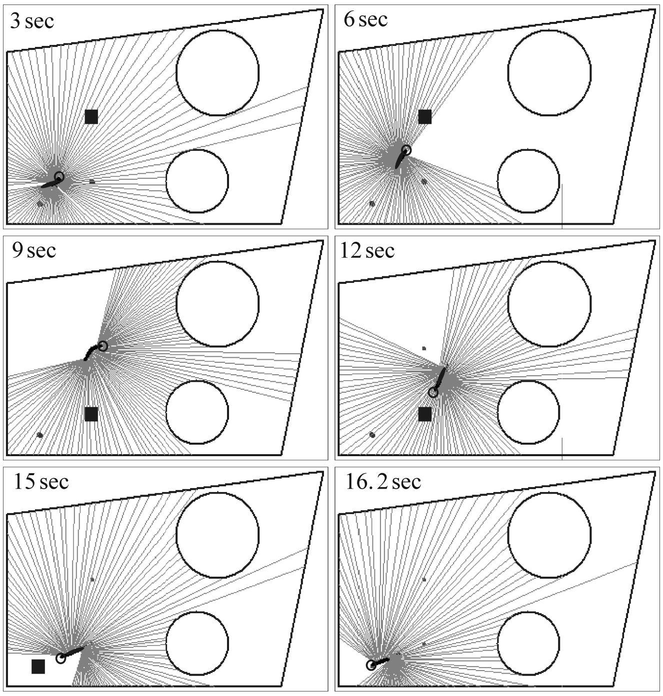
where x_1, x_2 are the coordinates of the robot center, x_3 is its orientation and x_4 is its speed. The inputs u_1 and u_2 are the accelerations provided by the propellers.

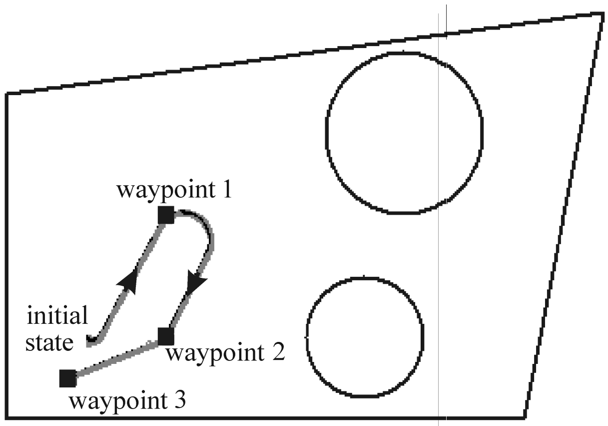
The system can be discretized by $\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k)$, where,

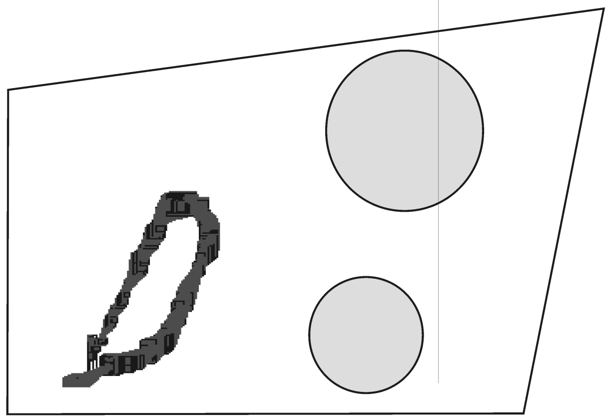
$$\mathbf{f}_k \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} x_1 + \delta.x_4.\cos(x_3) \\ x_2 + \delta.x_4.\sin(x_3) \\ x_3 + \delta.(u_2(k) - u_1(k)) \\ x_4 + \delta.(u_1(k) + u_2(k) - x_4) \end{pmatrix}$$



Underwater robot moving inside a pool


3 sec
6 sec
9 sec
12 sec
15 sec
16. 2 sec


waypoint 1
initial
state
waypoint 2
waypoint 3



https://youtu.be/c-8ZW8nUh7U

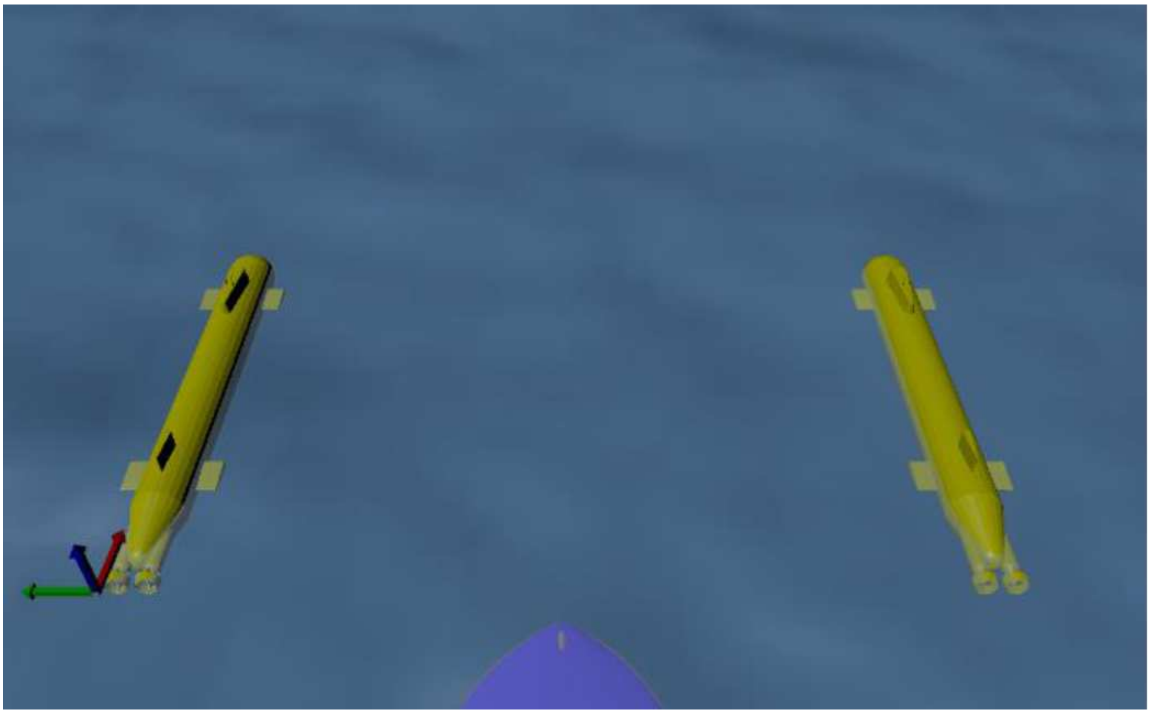
5 Scout project

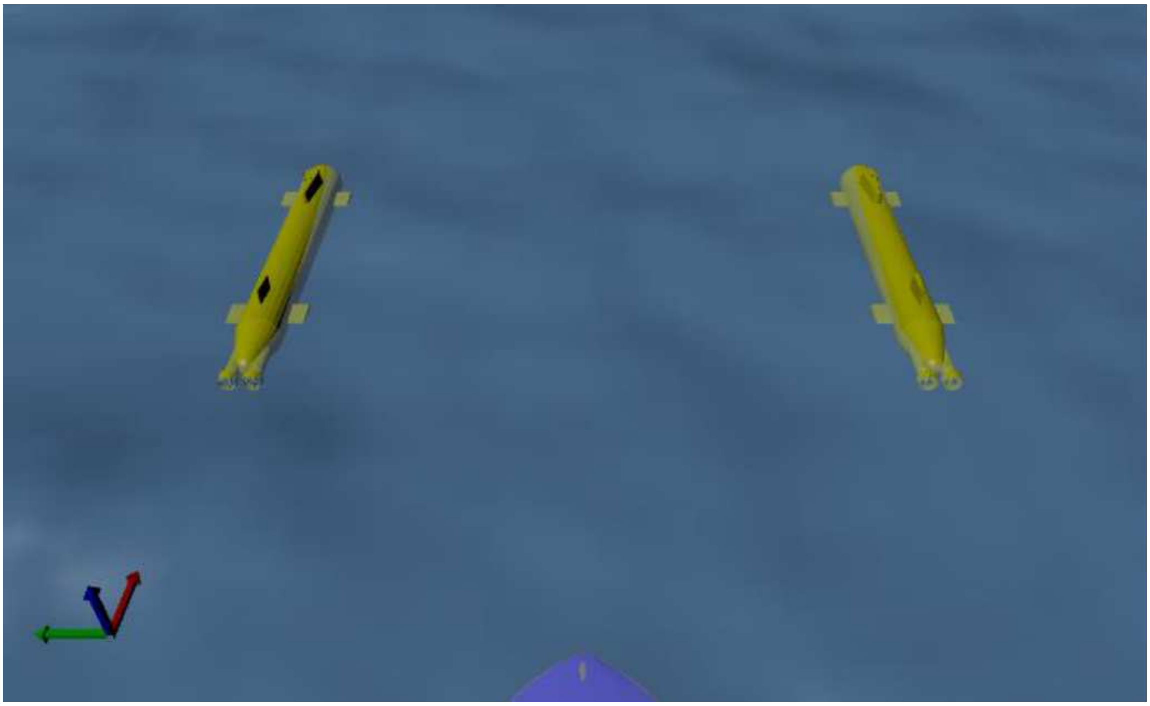
Goal : (i) coordination of underwater robots ; (ii) collaborative behavior.

Supervisors: L. Jaulin, C. Aubry, S. Rohou, B. Zerr, J. Nicola, F. Le bars

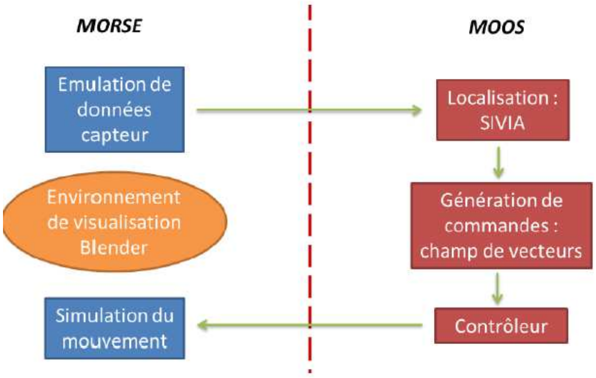
Compagny: RTsys (P. Raude)

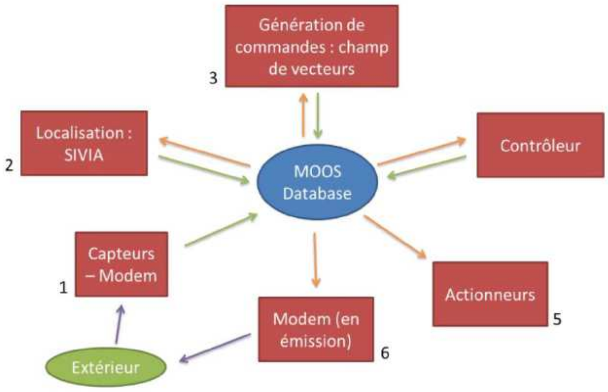
Students: G. Ricciardelli, L. Devigne, C. Guillemot, S. Pommier, T. Viravau, T. Le Mezo, B. Sultan, B. Moura, M. Fadlane, A. Bellaiche, T. Blanchard, U. Da rocha, G. Pinto, K. Machado.



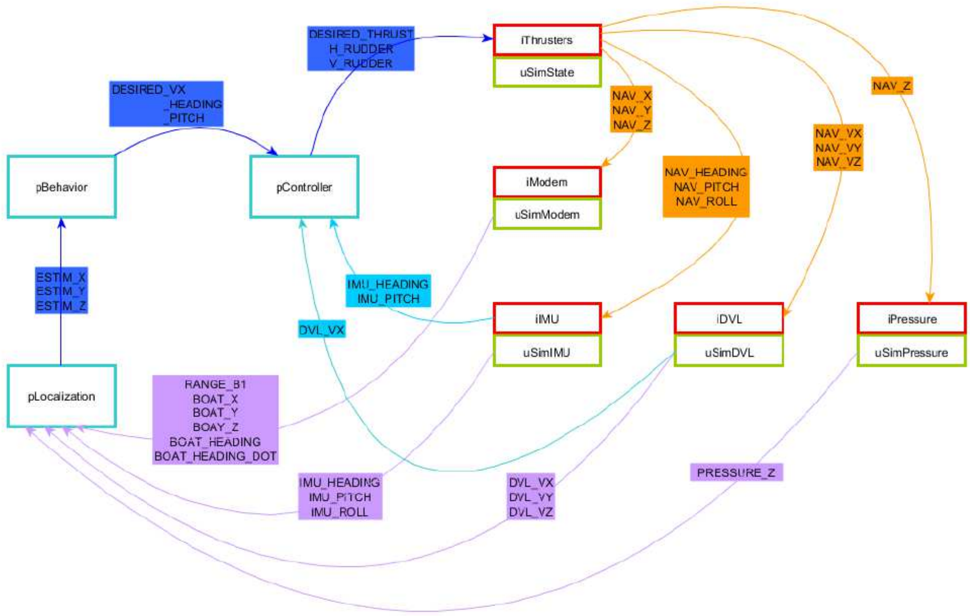


5.1 Simulator



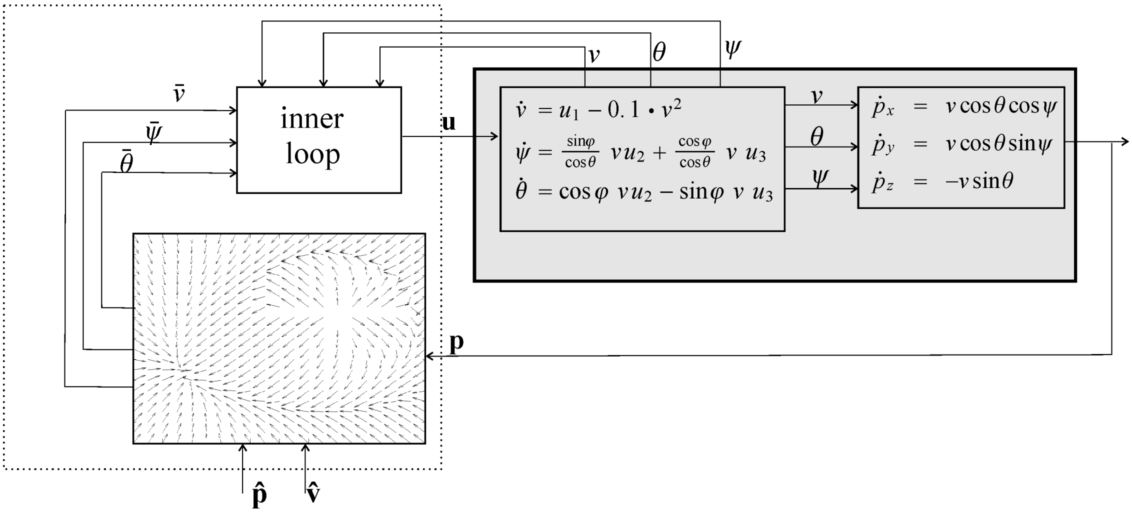


MOOS, MORSE, BLENDER, IBEX, Vibes, GIT



MOOS architecture

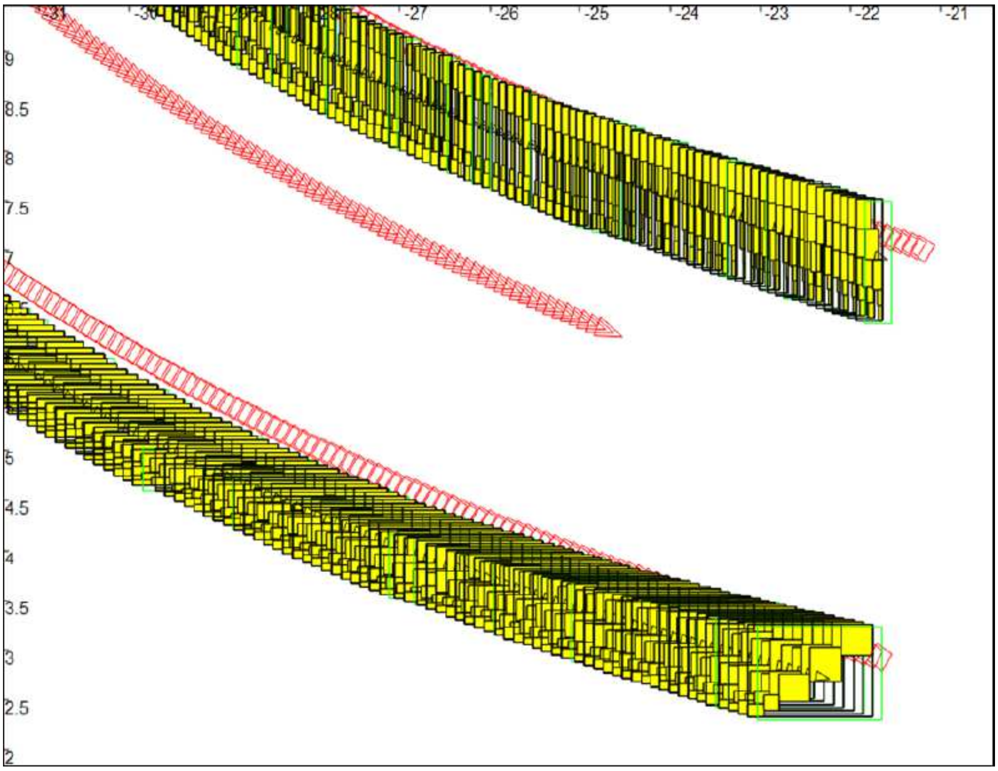
5.2 Controller



$\bar{v}$
$\bar{\psi}$
$\bar{\theta}$
inner loop
$\mathbf{u}$
v
θ
ψ
$\dot{v} = u_1 - 0.1 \cdot v^2$
$\dot{\psi} = \frac{\sin\varphi}{\cos\theta}\ v u_2 + \frac{\cos\varphi}{\cos\theta}\ v\ u_3$
$\dot{\theta} = \cos\varphi\ v u_2 - \sin\varphi\ v\ u_3$
$\dot{p}_x = v\cos\theta\cos\psi$
$\dot{p}_y = v\cos\theta\sin\psi$
$\dot{p}_z = -v\sin\theta$
$\mathbf{p}$
$\hat{\mathbf{p}}$
$\hat{\mathbf{v}}$

5.3 Localization

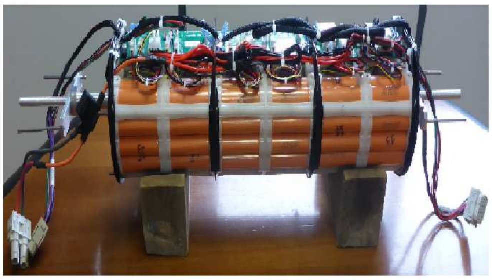
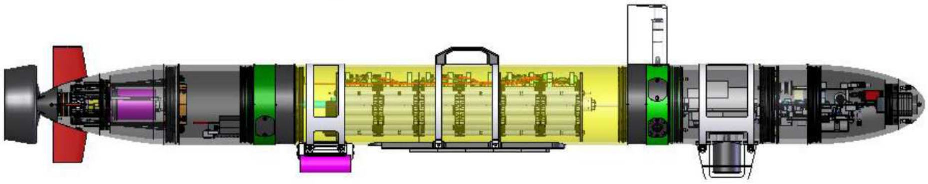
Range only
Based on interval analysis
Robust with respect to outliers
Distributed computation
Low rate communication


-31
-30
-27
-26
-25
-24
-23
-22
-21
9
8.5
8
7.5
7
5
4.5
4
3.5
3
2.5
2

Presentation of the scout project

http://youtu.be/ATPabRHz0LA

5.4 Tests


ZTI z1050



www.ensta-bretagne.fr/jaulin/easibex.html