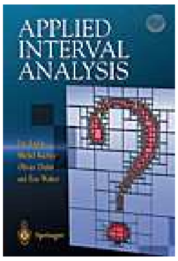
Quelques avancées du calcul par intervalles;

applications en robotique mobile



Luc Jaulin, OSM, ENSTA-Bretagne

Douai, le jeudi 7 avril 2011

1 Approche ensembliste

1.1 Calcul par intervalles

$$
\begin{aligned}
[-1, 3] + [2, 5] &= [1, 8], \\
[-1, 3].[2, 5] &= [-5, 15], \\
[-2, 6]/[2, 5] &= [-1, 3].
\end{aligned}
$$

Si $\mathbf{f}$ est donné par

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;
$y_1 := z$;
$y_2 := \sin(zx_1)$;

alors sa fonction d'inclusion naturelle est

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

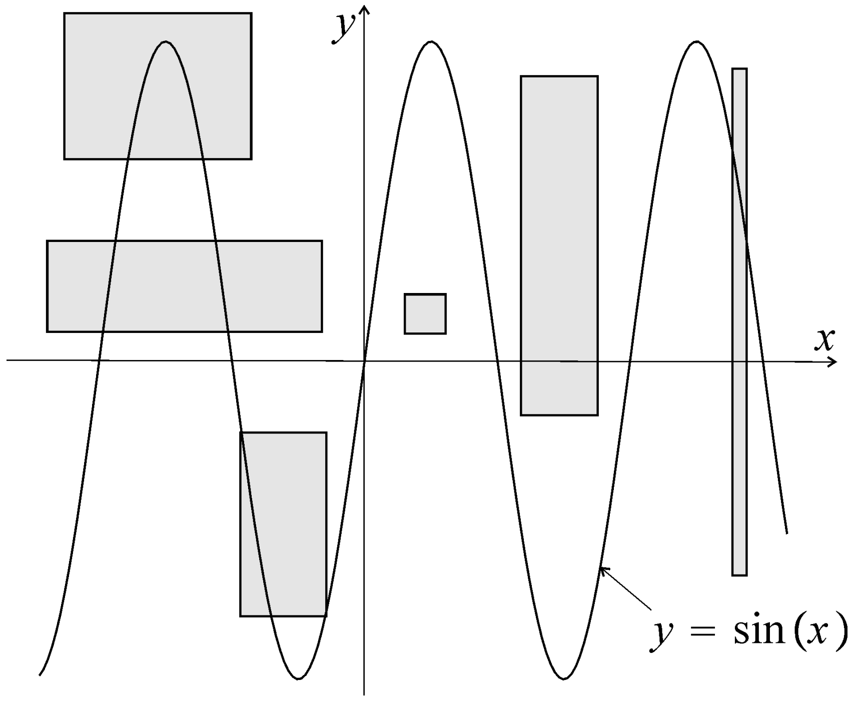
$[z] := [x_1]$;
for $k := 0$ to 100
    $[z] := [x_2] * ([z] + k * [x_3])$;
next;
$[y_1] := [z]$;
$[y_2] :=$ sin$([z] * [x_1])$;

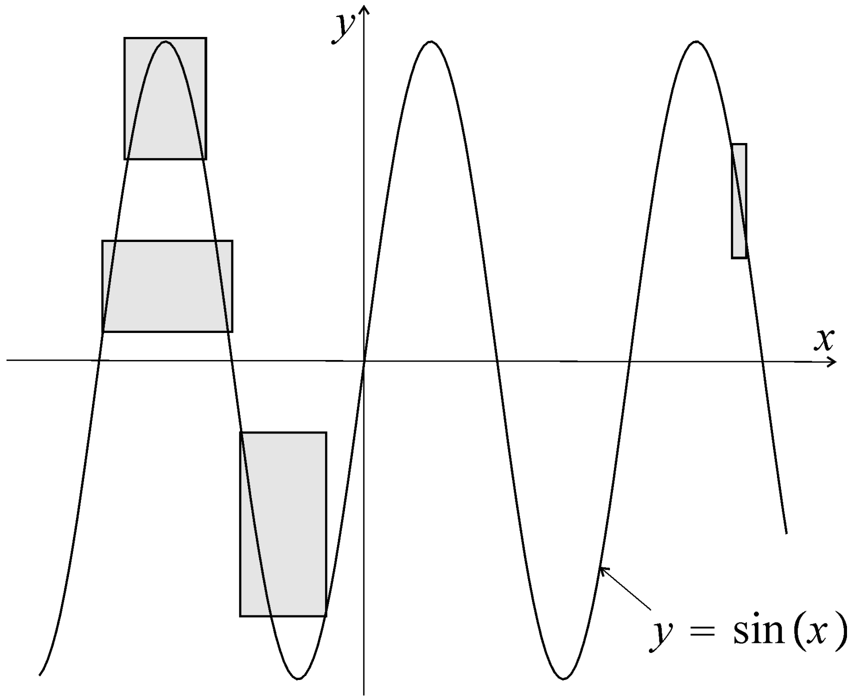
Montrer setdemo

1.2 Contracteurs

L'opérateur $\mathcal{C} : \mathbb{IR}^n \longrightarrow \mathbb{IR}^n$ est un *contracteur* pour l'équation $f(\mathbf{x}) = 0$, si

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


y
x
y = sin(x)


y
x
y = sin(x)

intersection	$(\mathcal{C}_1 \cap \mathcal{C}_2)([\mathbf{x}]) \stackrel{\text{def}}{=} \mathcal{C}_1([\mathbf{x}]) \cap \mathcal{C}_2([\mathbf{x}])$
union	$(\mathcal{C}_1 \cup \mathcal{C}_2)([\mathbf{x}]) \stackrel{\text{def}}{=} [\mathcal{C}_1([\mathbf{x}]) \cup \mathcal{C}_2([\mathbf{x}])]$
composition	$(\mathcal{C}_1 \circ \mathcal{C}_2)([\mathbf{x}]) \stackrel{\text{def}}{=} \mathcal{C}_1(\mathcal{C}_2([\mathbf{x}]))$
répétition	$\mathcal{C}^\infty \stackrel{\text{def}}{=} \mathcal{C} \circ \mathcal{C} \circ \mathcal{C} \circ \dots$

1.3 Algorithme de propagation-bissection

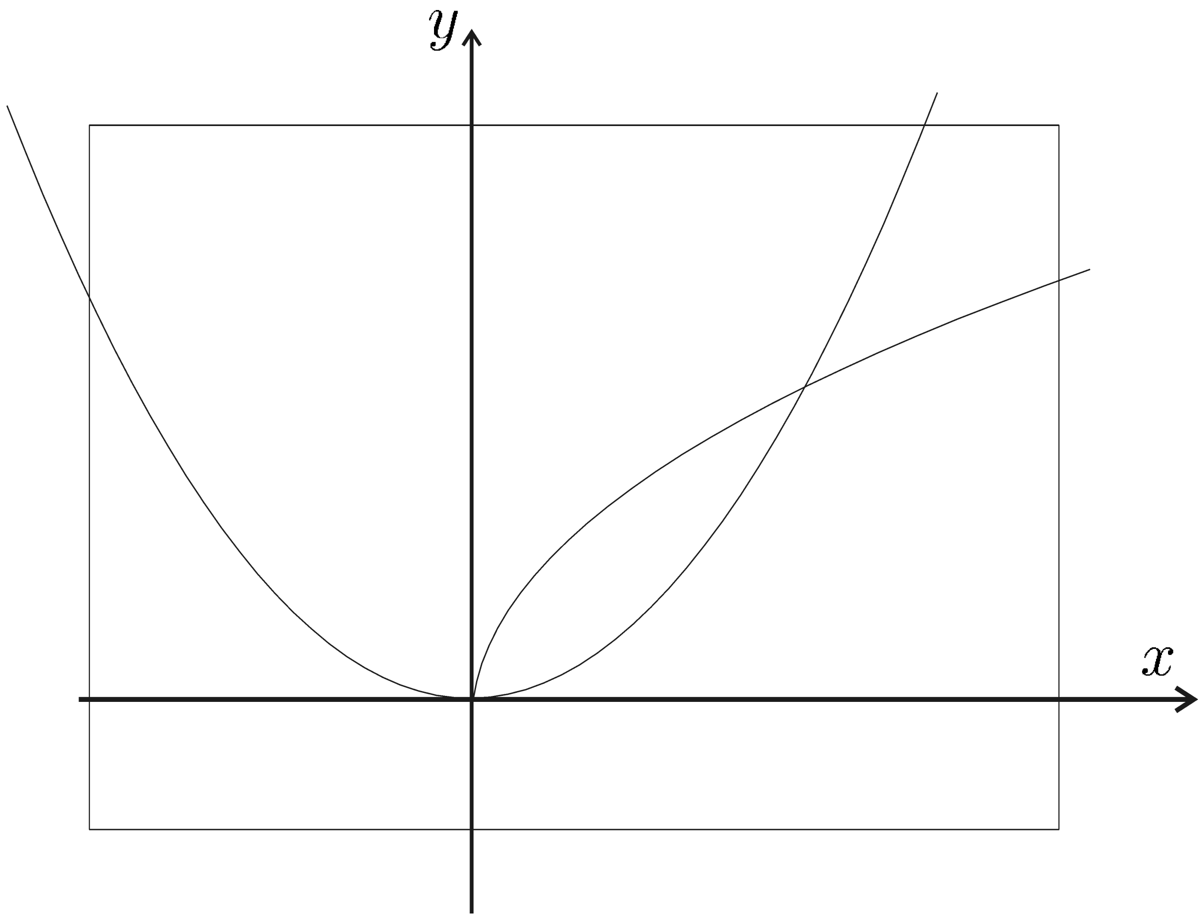
Exemple. Cherchons à résoudre.

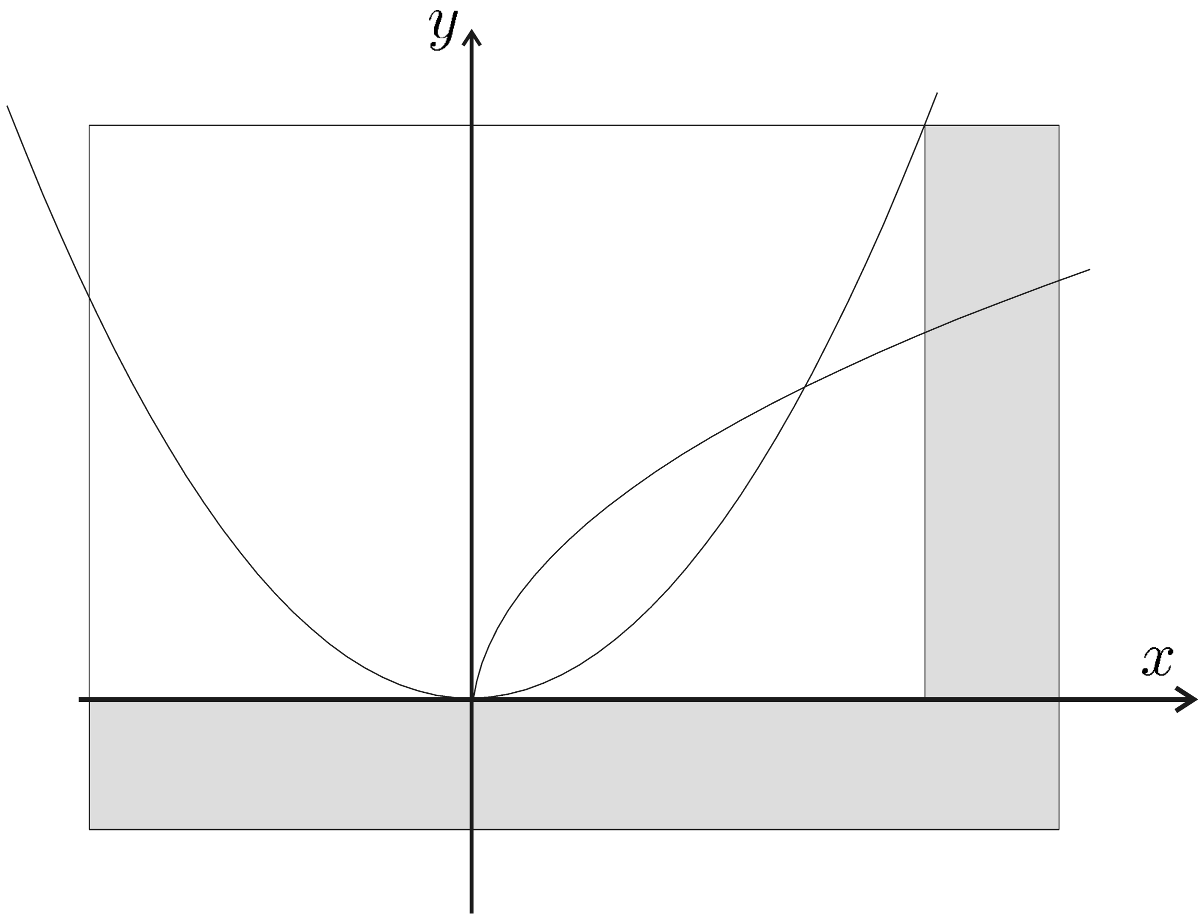
$$\begin{aligned} y &= x^2 \\ y &= \sqrt{x}. \end{aligned}$$

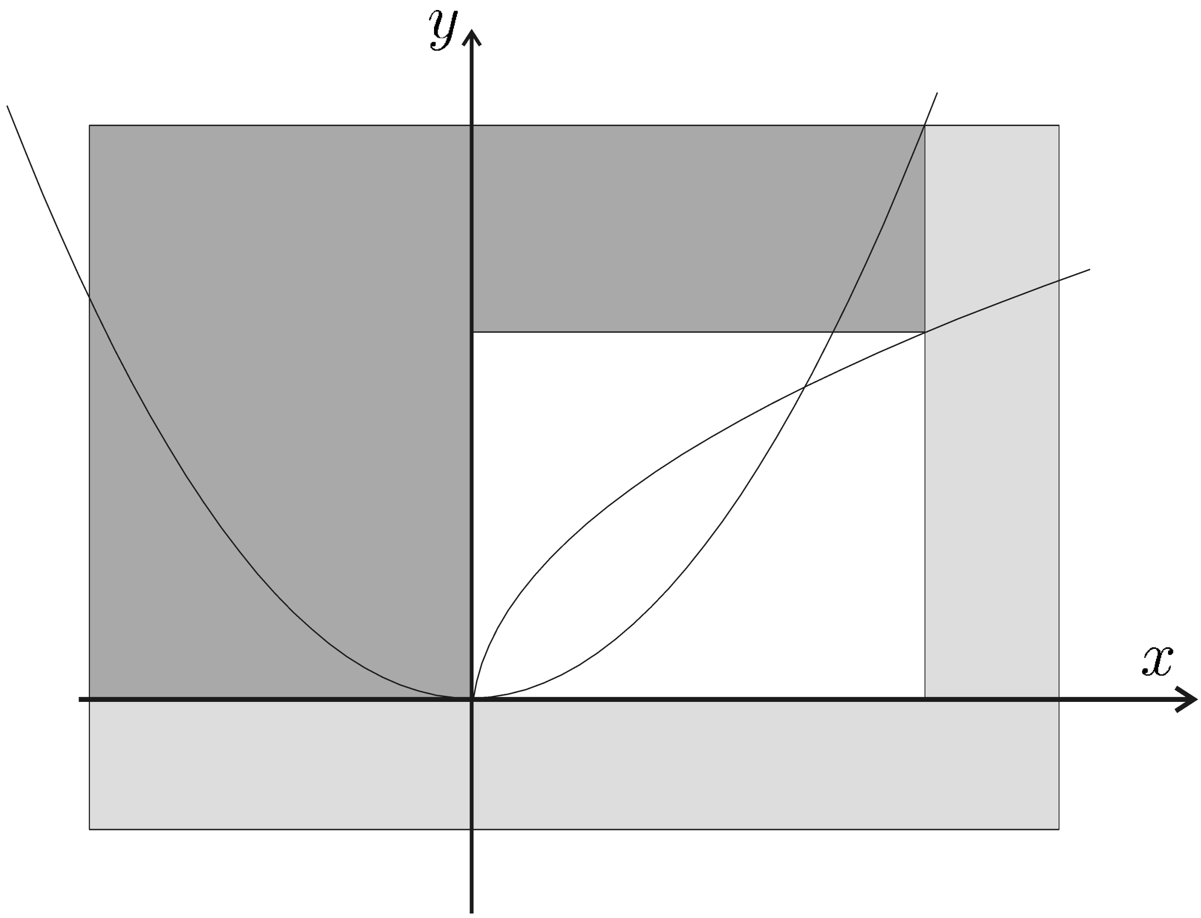
On a deux contracteurs

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

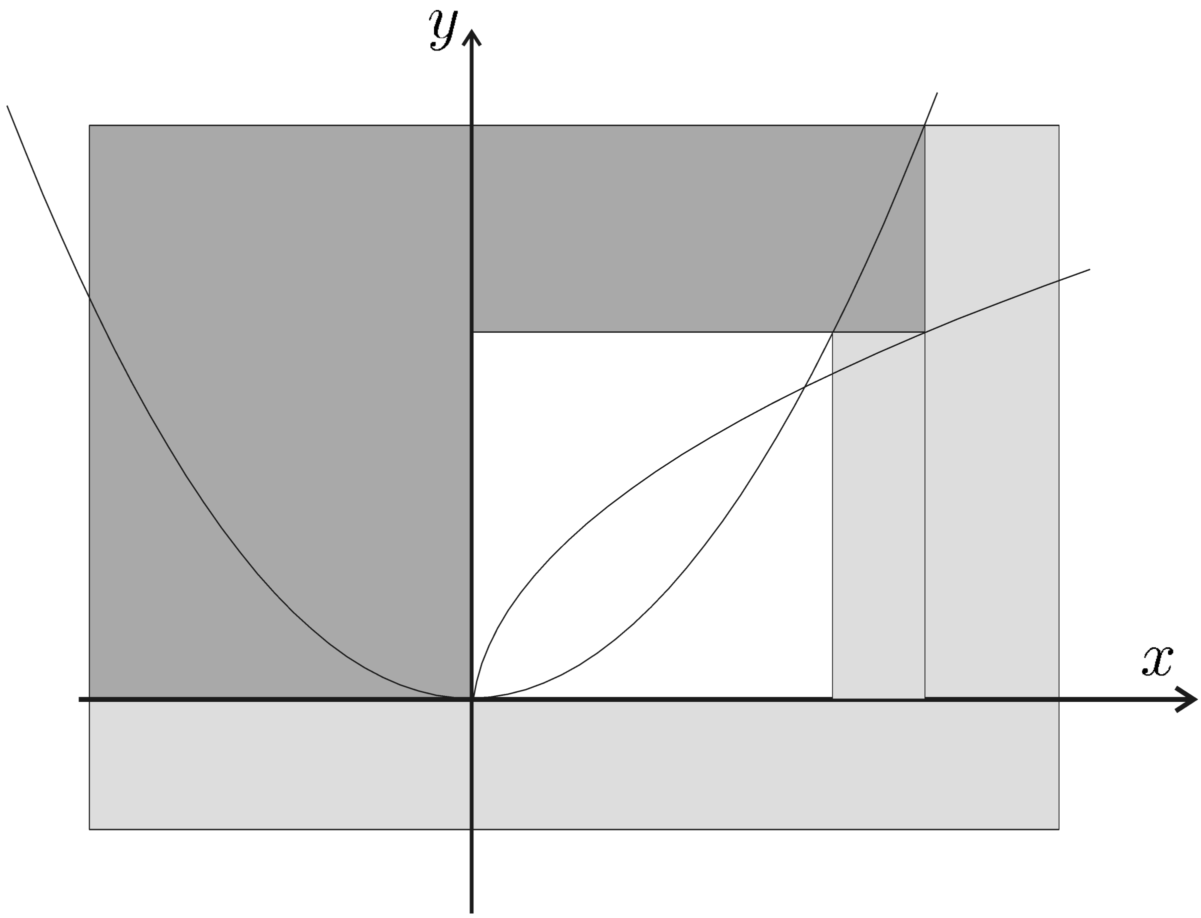
$$\mathcal{C}_2 : \left\{ \begin{array}{l} [y] = [y] \cap \sqrt{[x]} \\ [x] = [x] \cap [y]^2 \end{array} \right. \quad \text{associé à } y = \sqrt{x}$$

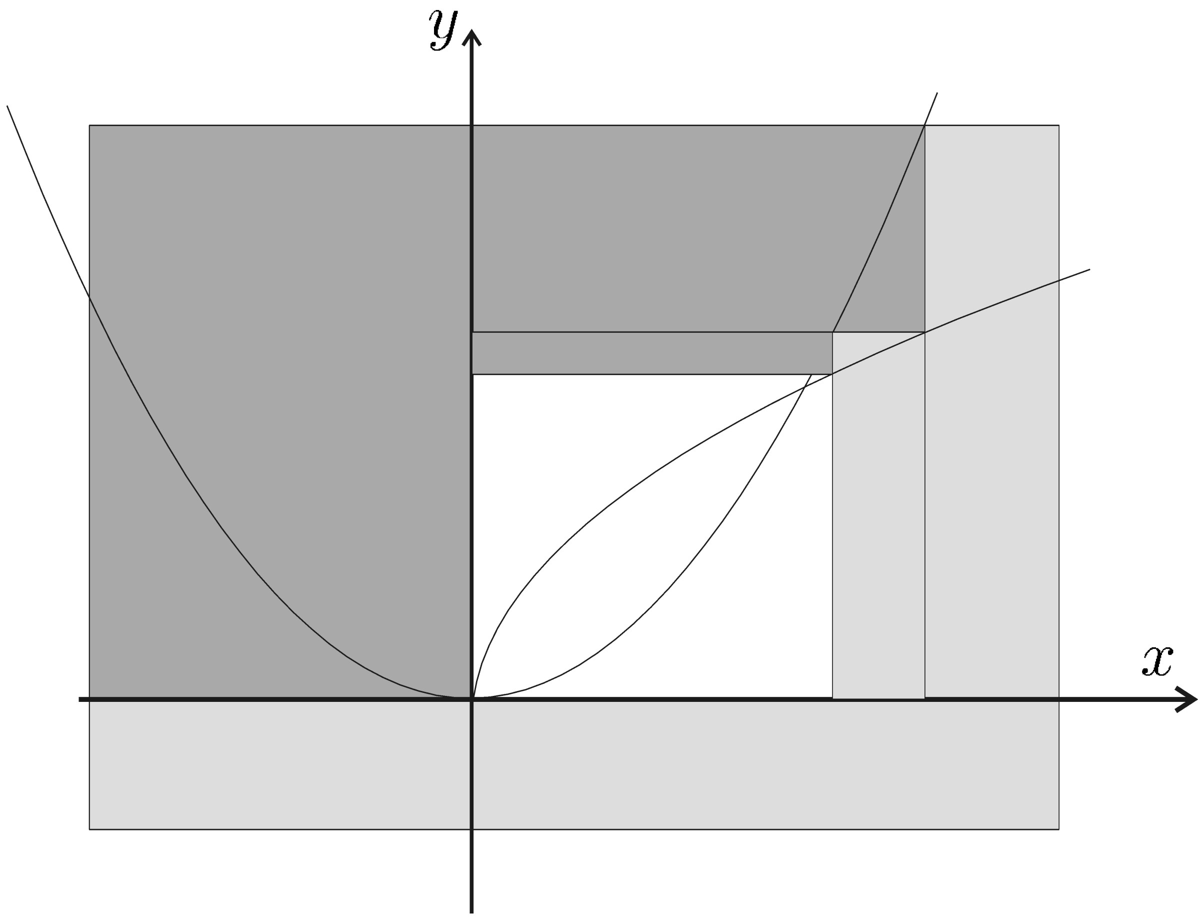

y
x

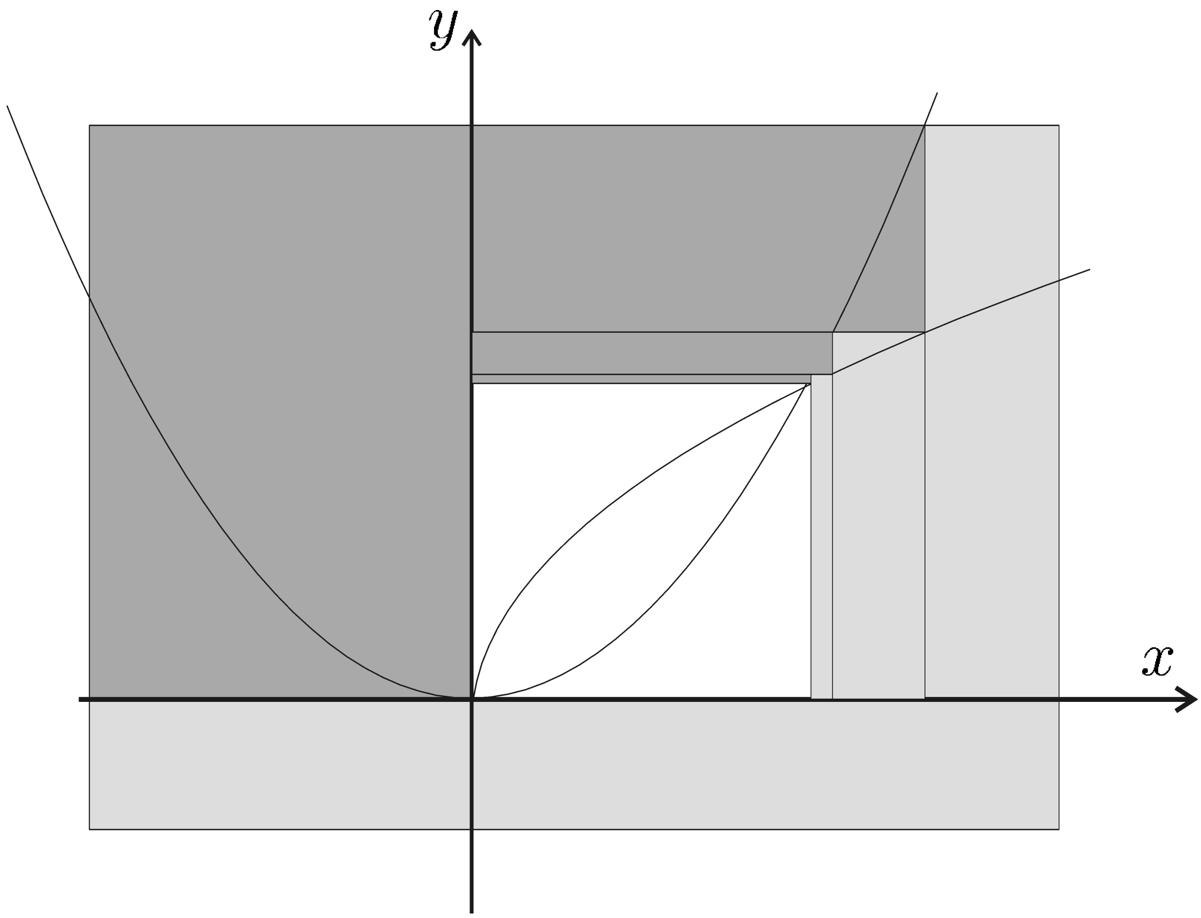

y
x



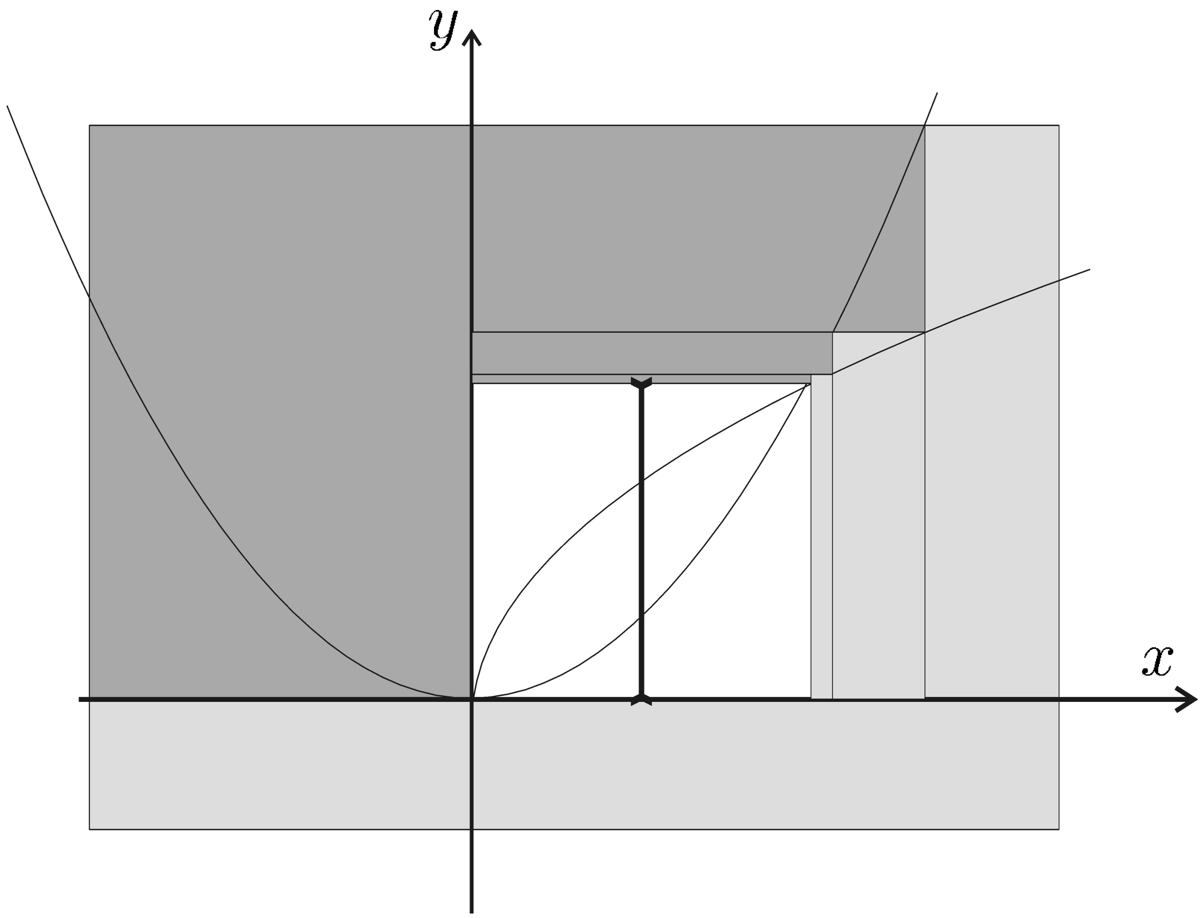
y
x

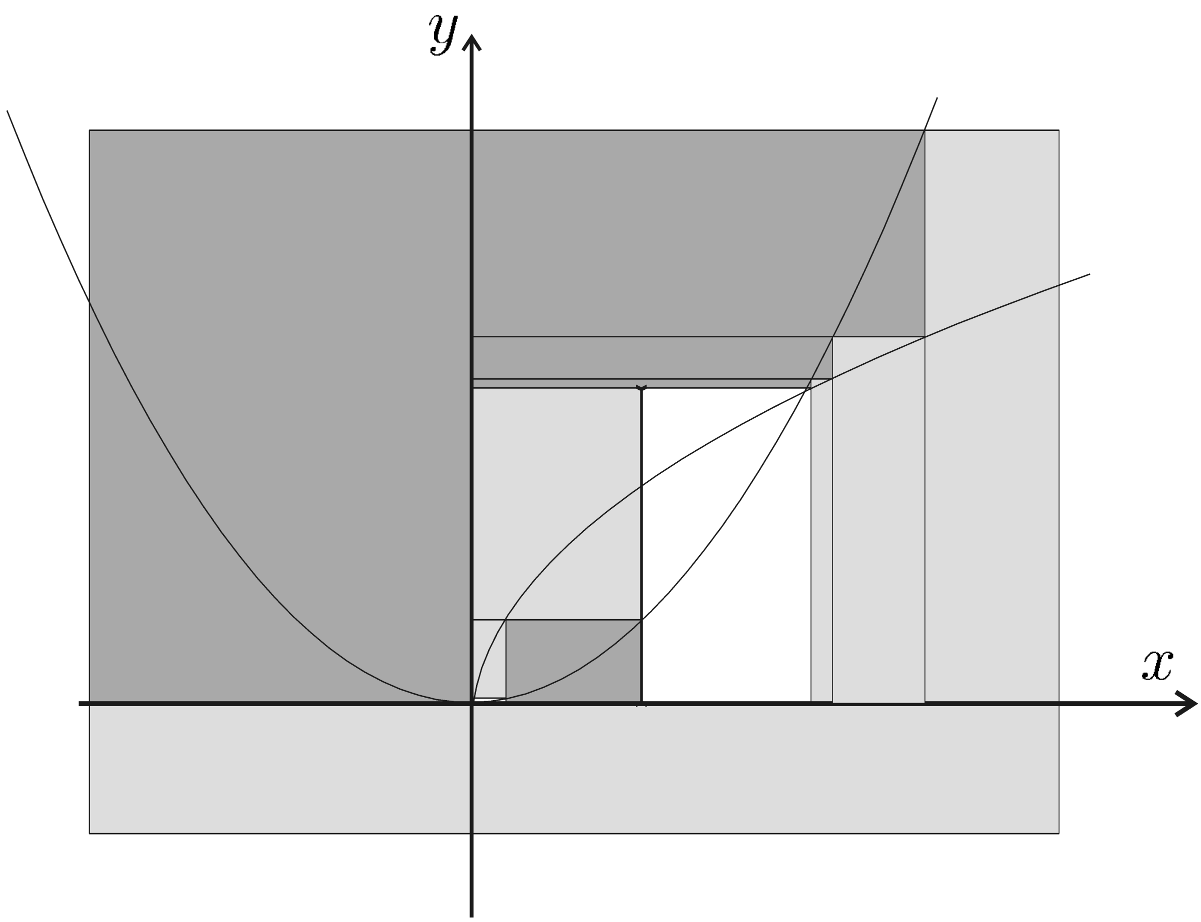

y
x


y
x

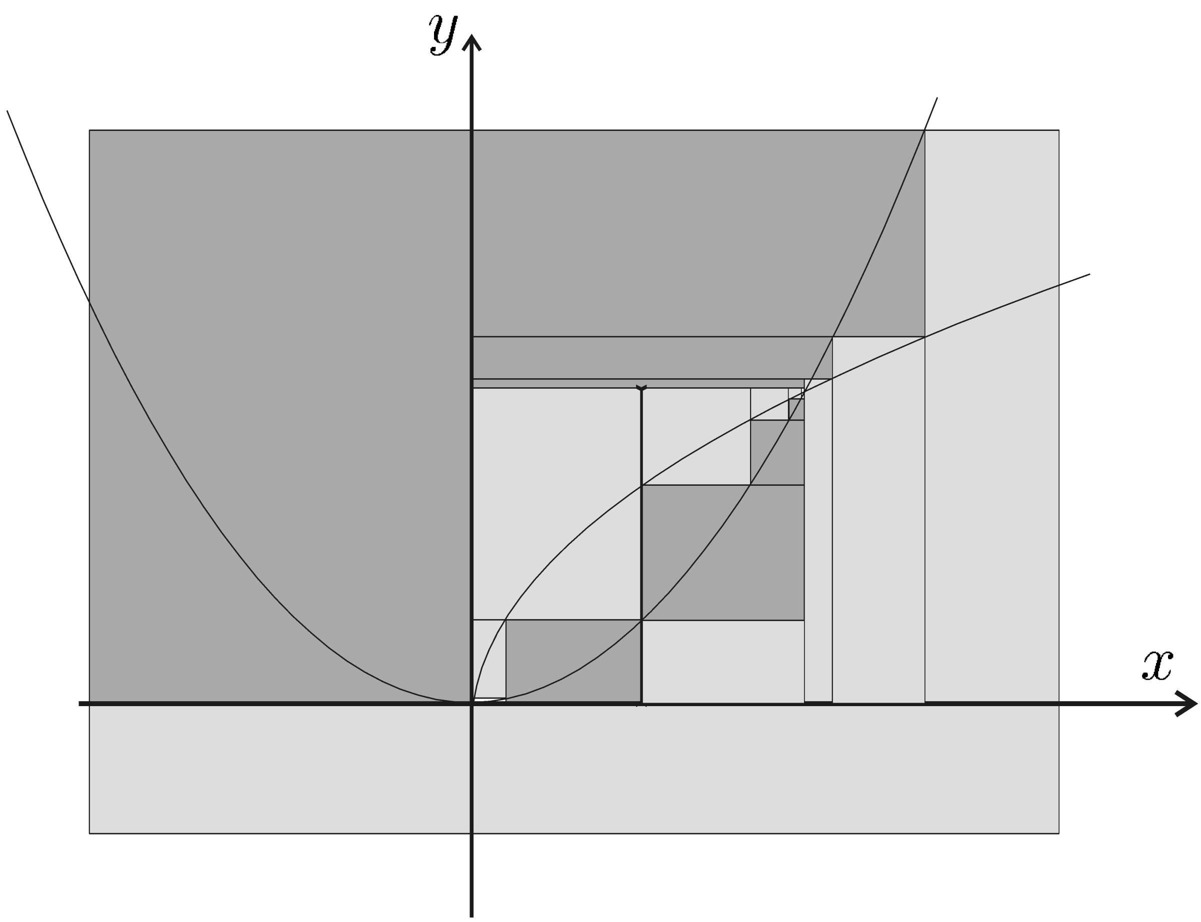


y
x


y
x



y
x



y
x

1.4 Décomposition

Pour les contraintes plus complexes, il nous décomposer :

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

se décompose en

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

1.5 QUIMPER

Quimper : QUick Interval Modeling and Programming in a bounded-ERror context.

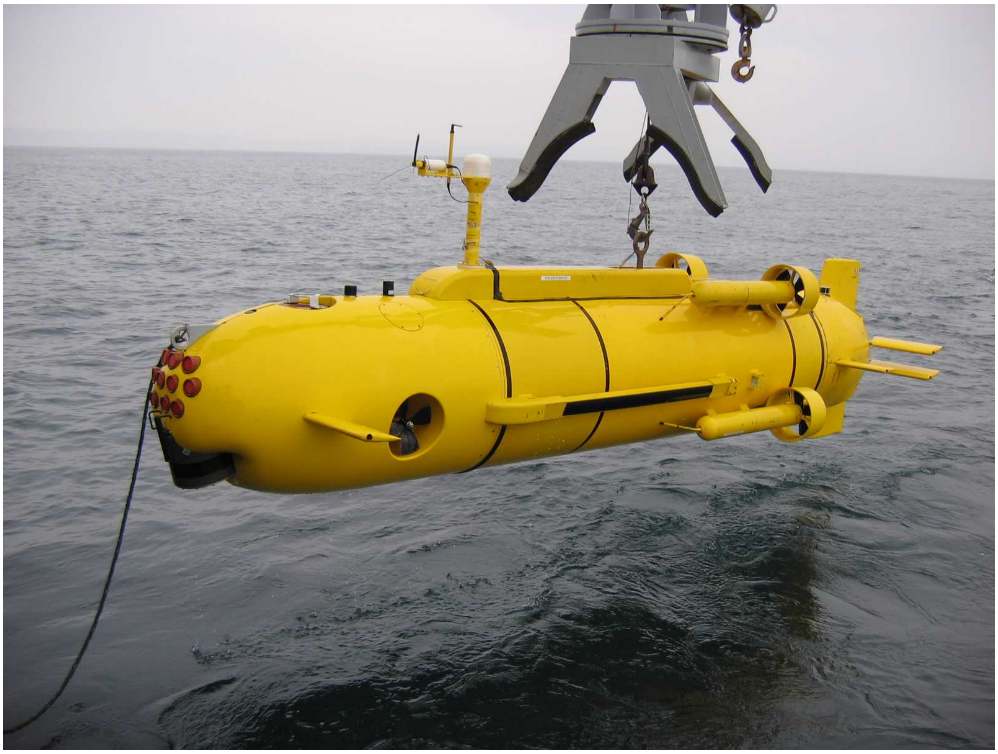
Quimper est un langage interprété pour le calcul ensembliste.

Un programme Quimper se décrit par un ensemble de contracteurs.

Quimper est un logiciel libre.

2 SLAM

2.1 Redermor



Redermor, GESMA
(Groupe d'Etude Sous-Marine de l'Atlantique)


A785

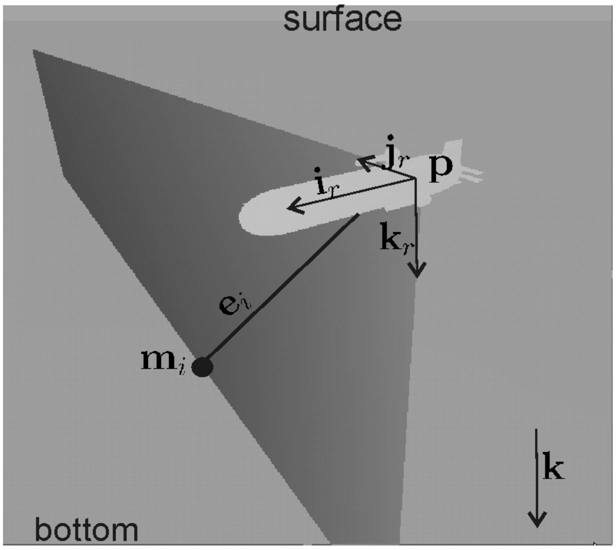
Montrer Daurade

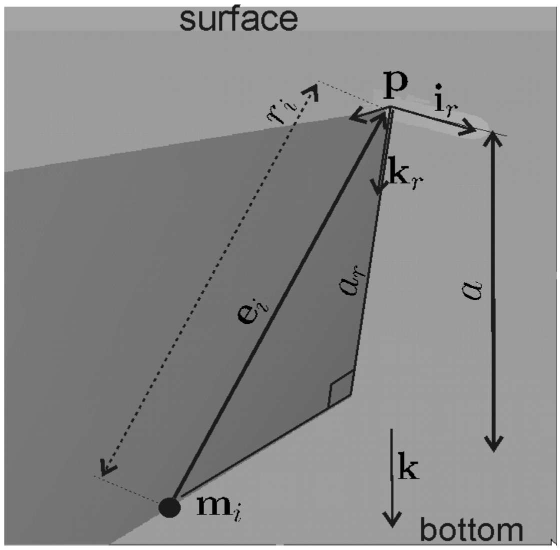
2.1.1 Sensors

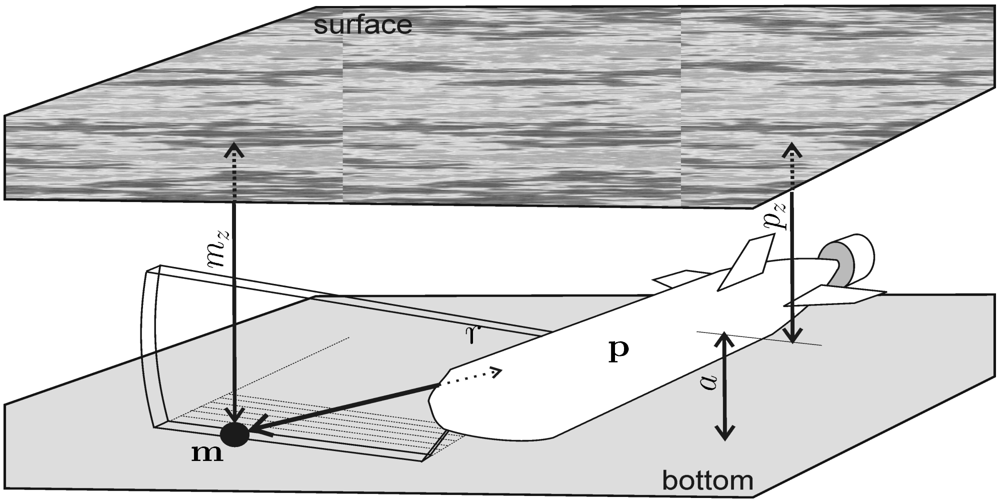
GPS (Global positioning system), only at the surface.

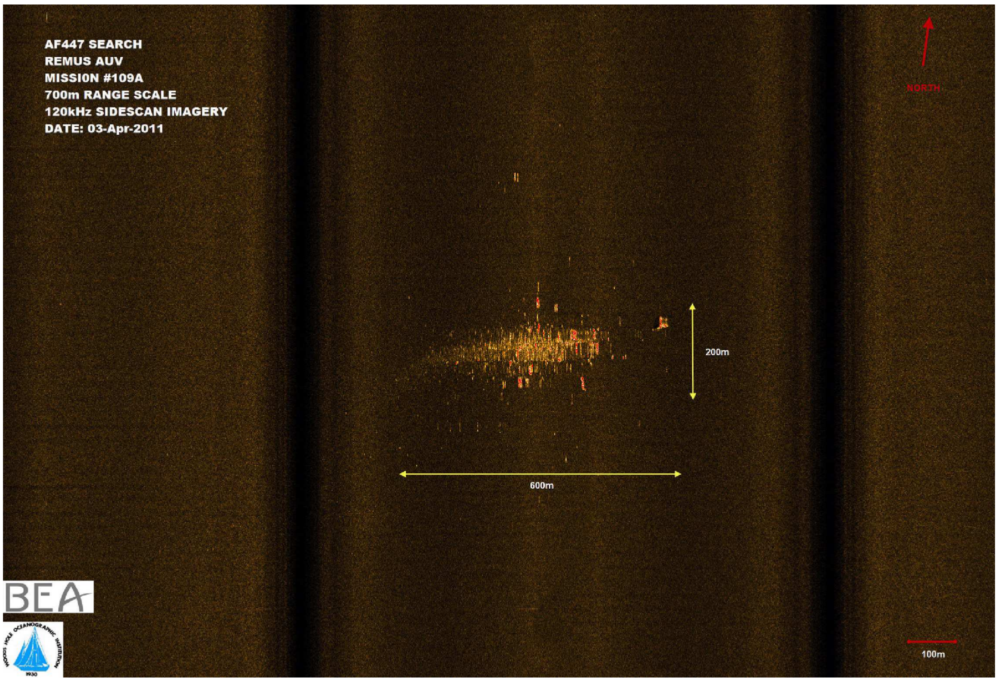
$$t_0 = 6000 \text{ s}, \quad \ell^0 = (-4.4582279^\circ, 48.2129206^\circ) \pm 2.5m$$
$$t_f = 12000 \text{ s}, \quad \ell^f = (-4.4546607^\circ, 48.2191297^\circ) \pm 2.5m$$

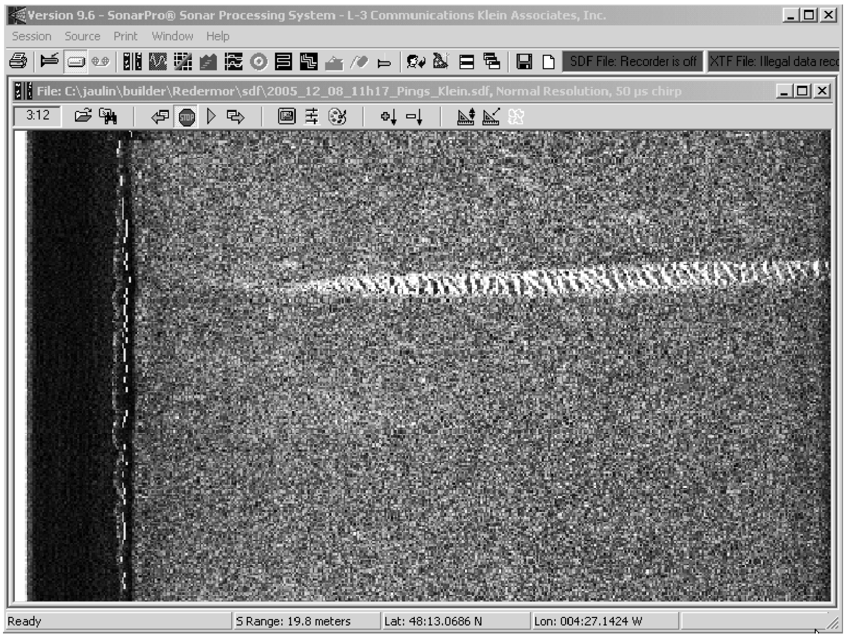
Sonar (KLEIN 5400 side scan sonar).



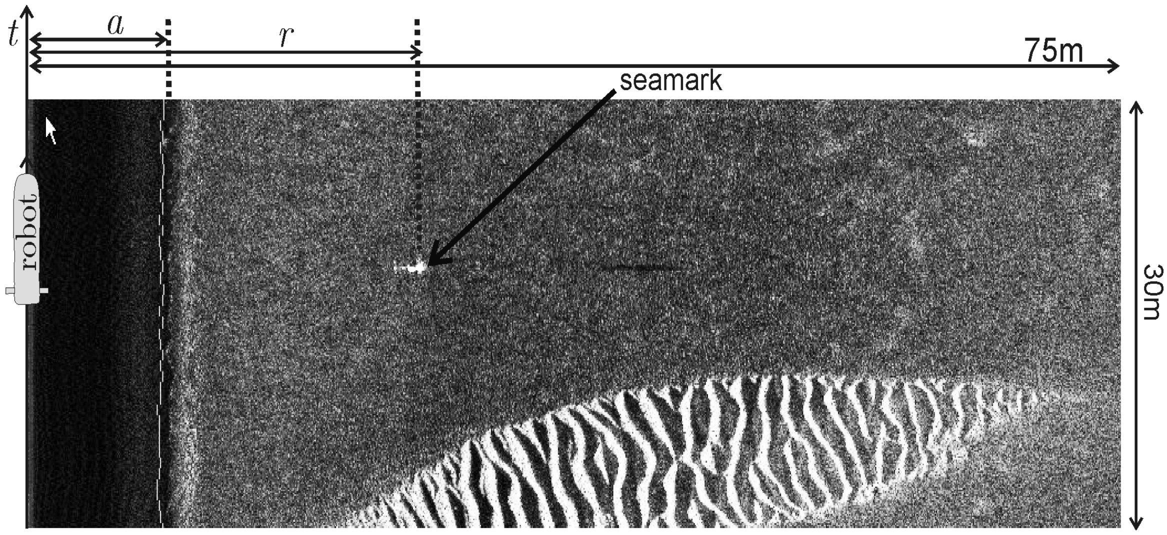



surface
m_z
p_z
r
$\mathbf{p}$
a
$\mathbf{m}$
bottom


AF447 SEARCH
REMUS AUV
MISSION #109A
700m RANGE SCALE
120kHz SIDESCAN IMAGERY
DATE: 03-Apr-2011
NORTH
200m
600m
100m
BEA



Screenshot of SonarPro



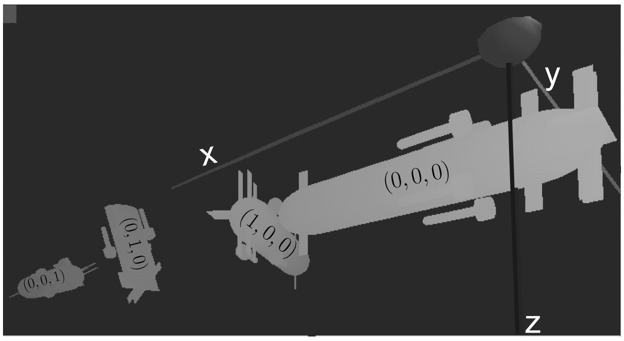
Mine detecttion with SonarPro

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 central (Octans III from IXSEA).

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



Six marks have been detected.

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

2.1.2 Constraints

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

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

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

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

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

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

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

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

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

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

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

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

```
//---------------------------------------------------------
Constants
N = 59996; // Number of time steps
Variables
  R[N-1][3][3], // rotation matrices
  p[N][3], // positions
  v[N-1][3], // speed vectors
  phi[N-1],theta[N-1],psi[N-1]; // Euler angles
  px[N],py[N]; // for display only
//---------------------------------------------------------
```

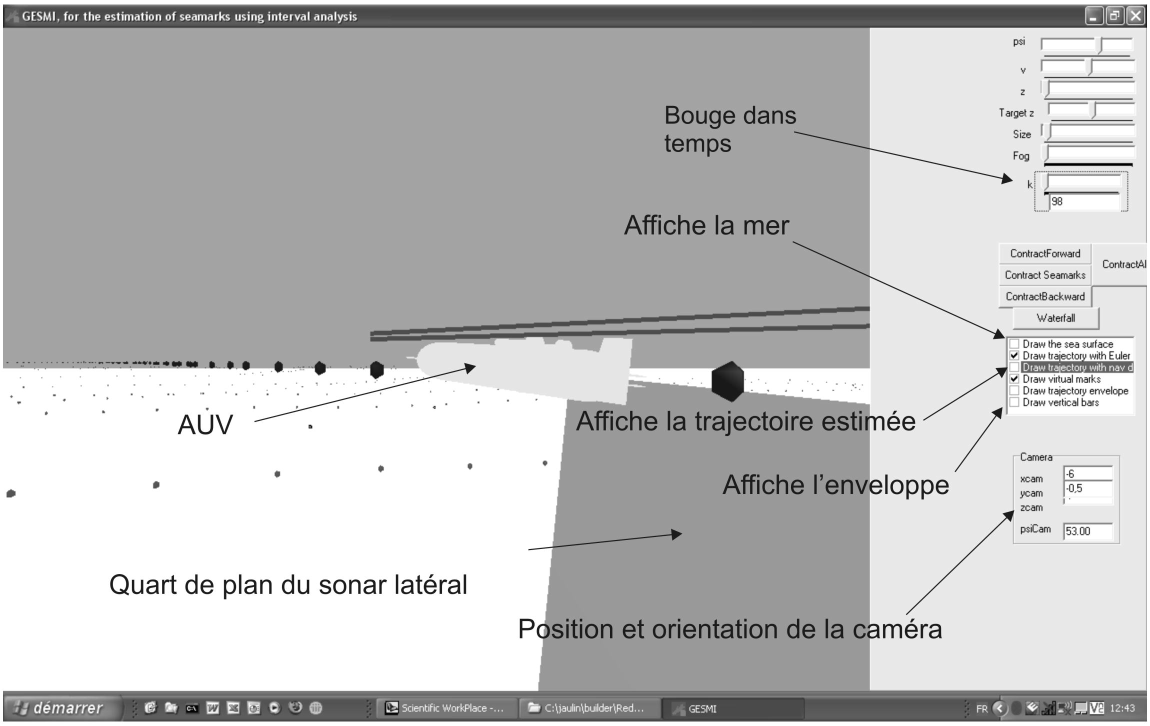
```
function R[3][3]=euler(phi,theta,psi)
  cphi = cos(phi);
  sphi = sin(phi);
  ctheta = cos(theta);
  stheta = sin(theta);
  cpsi = cos(psi);
  spsi = sin(psi);
  R[1][1]=ctheta*cpsi;
  R[1][2]=-cphi*spsi+stheta*cpsi*sphi;
  R[1][3]=spsi*sphi+stheta*cpsi*cphi;
  R[2][1]=ctheta*spsi;
  R[2][2]=cpsi*cphi+stheta*spsi*sphi;
  R[2][3]=-cpsi*sphi+stheta*cphi*spsi;
  R[3][1]=-stheta;
  R[3][2]=ctheta*sphi;
  R[3][3]=ctheta*cphi;
end
```

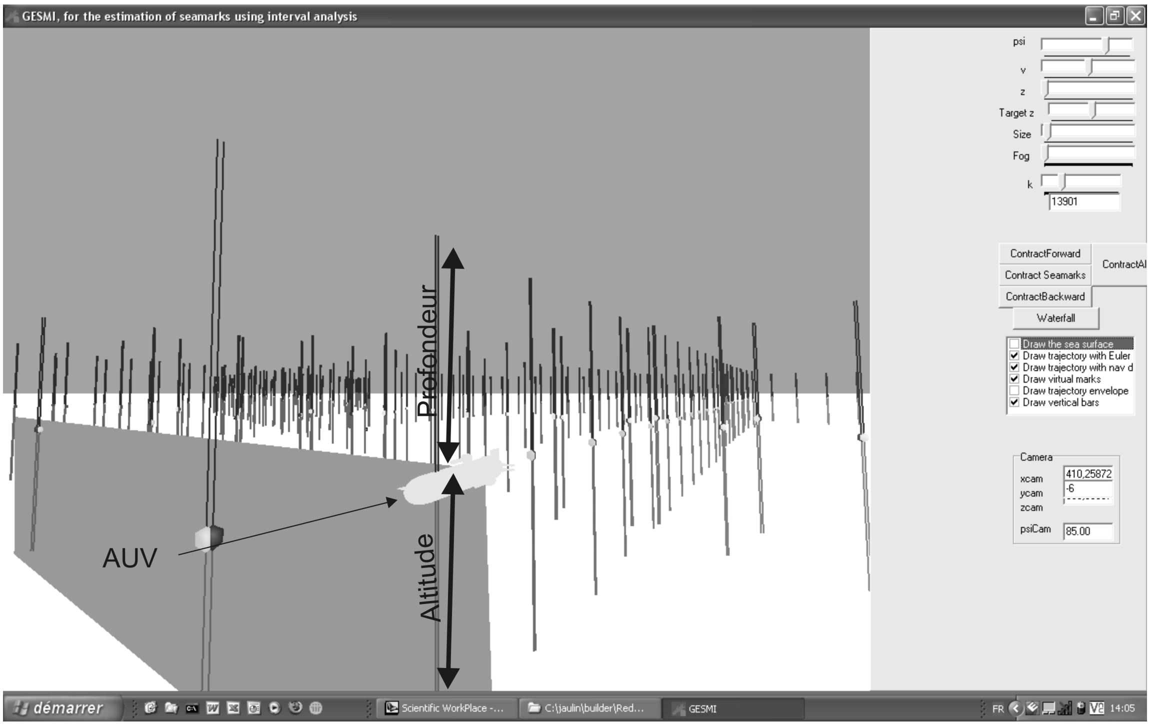
```
contractor-list rotation
  for k=1:N-1;
    R[k]=euler(phi[k],theta[k],psi[k]);
  end
end
//---------------------------------
contractor-list statequ
  for k=1:N-1;
    p[k+1]=p[k]+0.1*R[k]*v[k];
  end
end
//---------------------------------
contractor init
  inter k=1:N-1;
    rotation(k)
  end
end
```

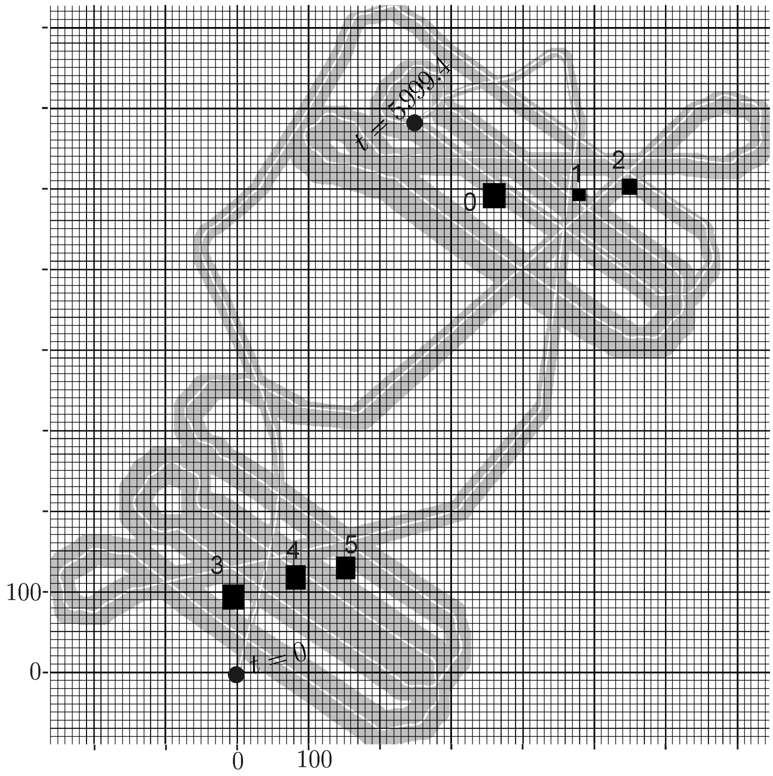
```
contractor fwd
  inter k=1:N-1;
    statequ(k)
  end
end
//--------------------------------
contractor bwd
  inter k=1:N-1;
    statequ(N-k)
  end
end
```

```
main
  p[1] :=read("gps_init.dat");
  v :=read("Quimper_v.dat");
  phi :=read("Quimper_phi.dat");
  theta :=read("Quimper_theta.dat");
  psi :=read("Quimper_psi.dat");
  init;
  fwd;
  bwd;
  column(p,px,1);
  column(p,py,2);
  print("--- Robot positions: ---");
  newplot("gesmi.dat");
  plot(px,py,color(rgb(1,1,1),rgb(0,0,0)));
end
```

2.1.3 GESMI


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

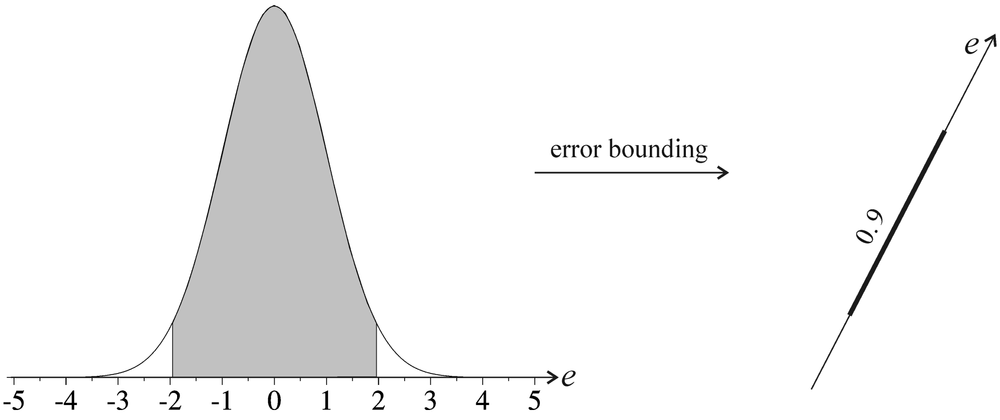

GESMI, for the estimation of seamarks using interval analysis
psi
v
z
Target z
Size
Fog
k
13901
ContractForward
Contract Seamarks
ContractBackward
ContractAl
Waterfall
Draw the sea surface
Draw trajectory with Euler
Draw trajectory with nav d
Draw virtual marks
Draw trajectory envelope
Draw vertical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



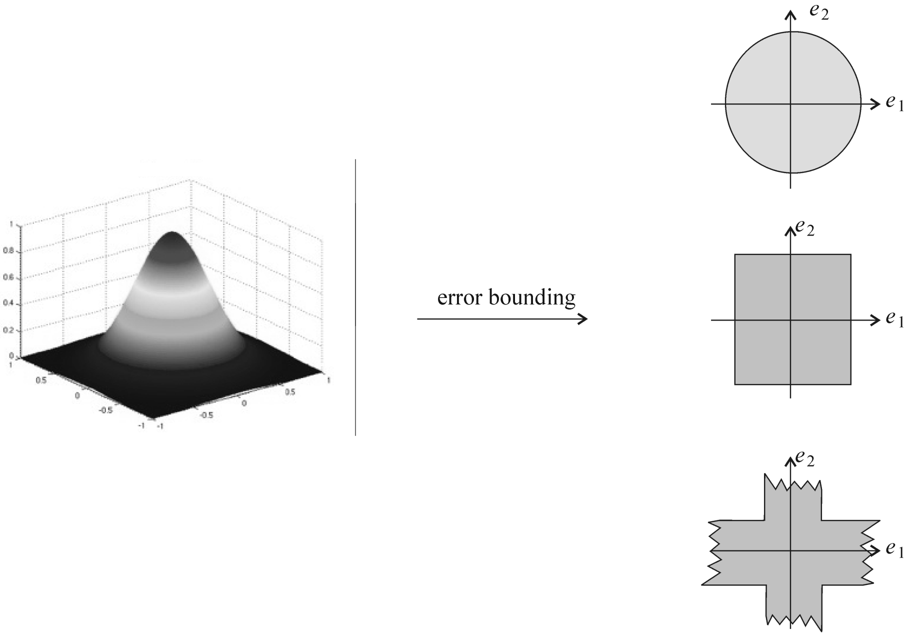
t = 5999.4
0
1
2
3
4
5
t = 0
100
0
0
100

3 Probabilistic-set approach

3.1 Error bounding

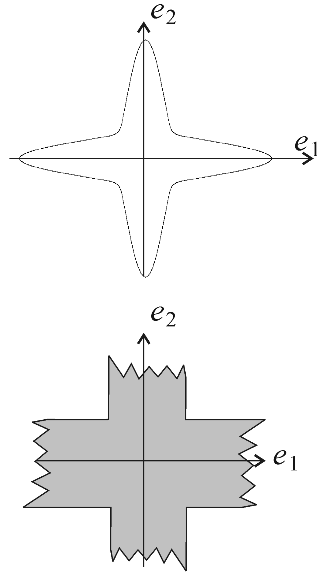


-5 -4 -3 -2 -1 0 1 2 3 4 5
e
error bounding
0.9
e


error bounding
e2
e1
e2
e1
e2
e1

$$\Pi(\mathbf{e}) \propto \left(\exp(-e_1^2) + \exp\left(-\frac{e_1^2}{10}\right)\right) * \left(\exp(-e_2^2) + \exp\left(-\frac{e_2^2}{10}\right)\right)$$

error bounding $\longrightarrow$



3.2 Principle of the approach

Consider the error model

$$\mathbf{e} = \underbrace{\mathbf{y} - \boldsymbol{\psi}(\mathbf{p})}_{\mathbf{f}(\mathbf{y},\mathbf{p})}.$$

y_i is an *inlier* if $e_i \in [e_i]$ and an *outlier* otherwise. We assume that

$$\forall i,\ \mathsf{Pr}\left(e_i \in [e_i]\right) = \pi$$

and that all e_i's are independent.

Equivalently,

$$\begin{cases} f_1(\mathbf{y}, \mathbf{p}) \in [e_1] & \text{with a probability } \pi \\ \vdots & \vdots \\ f_m(\mathbf{y}, \mathbf{p}) \in [e_m] & \text{with a probability } \pi \end{cases}$$

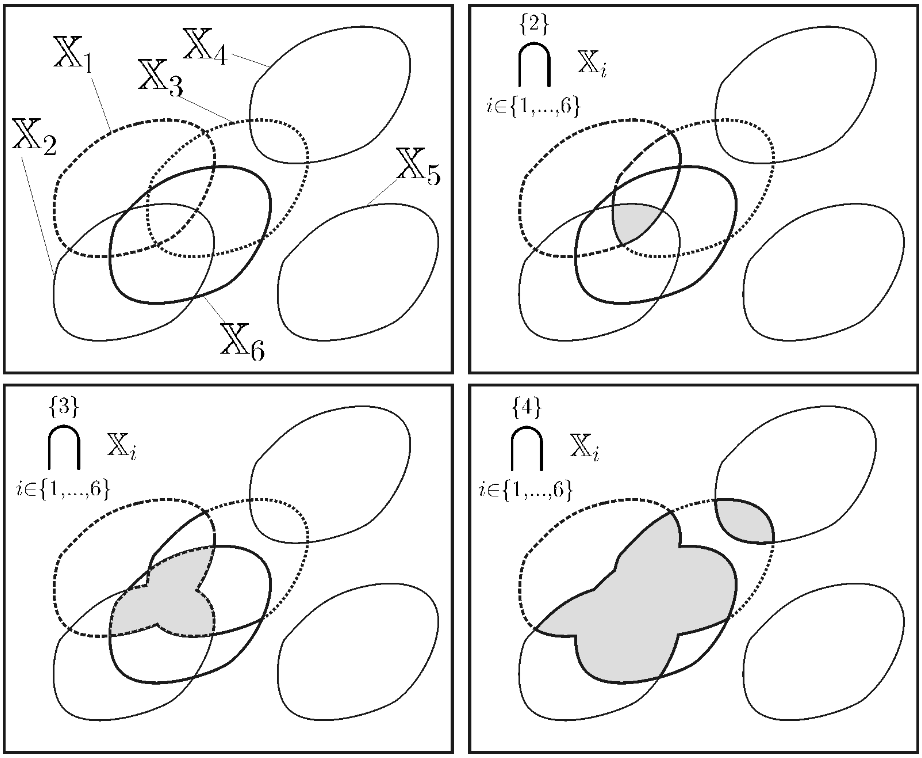
The number k of inliers follows a binomial distribution

$$\frac{m!}{k!\,(m-k)!}\pi^k.\,(1-\pi)^{m-k}\,.$$

The probability of having more than q outliers is thus

$$\gamma(q,m,\pi) \stackrel{\text{def}}{=} \sum_{k=0}^{m-q-1} \frac{m!}{k!\,(m-k)!}\pi^k.\,(1-\pi)^{m-k}.$$

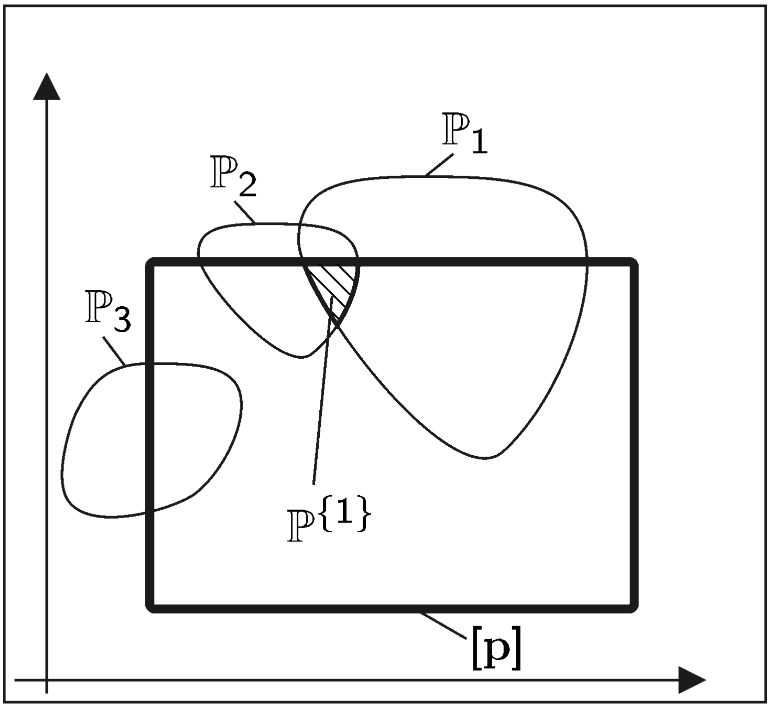
Example. For instance, if $m = 1000$, $q = 900, \pi = 0.2$, we get $\gamma(q,m,\pi) = 7.04 \times 10^{-16}$. Thus having more than 900 outliers can be seen as a rare event.

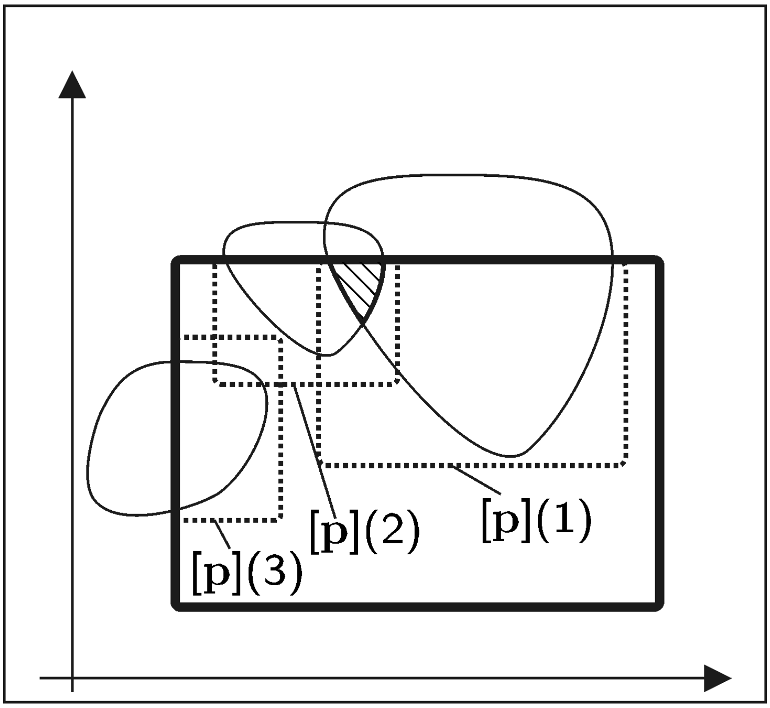


q-intersection

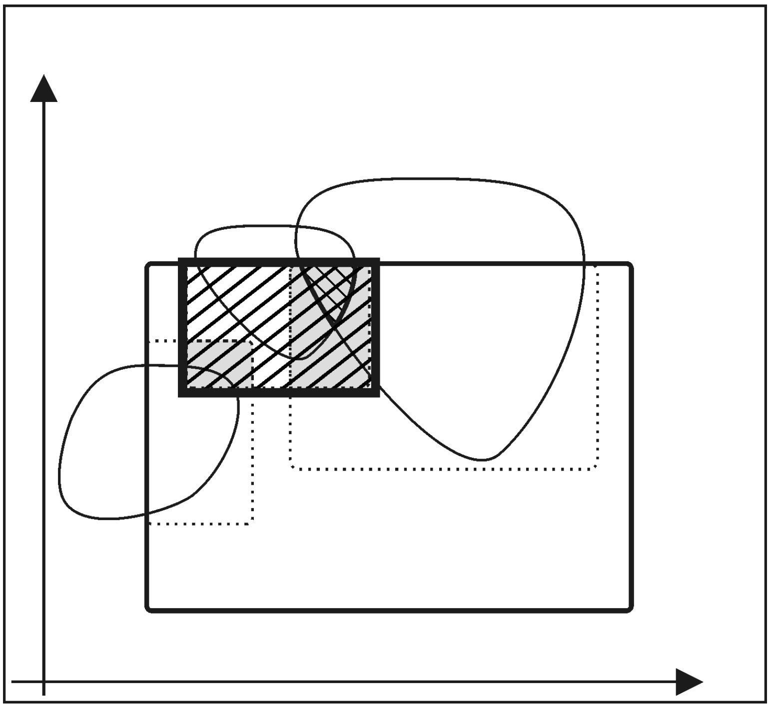
q-intersection de contracteurs.

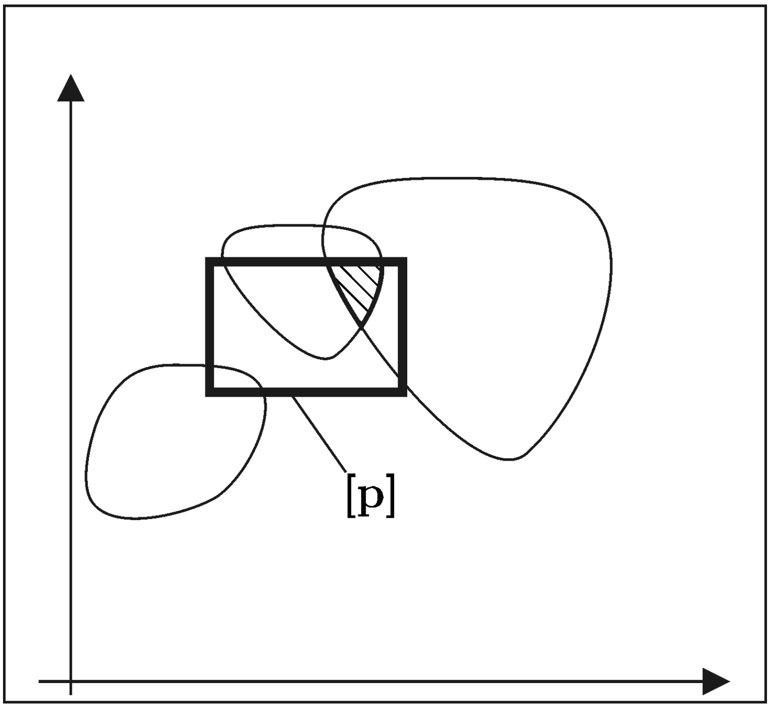
$$\mathcal{C} = \bigcap_{i \in \{1,2,3\}}^{\{1\}} \mathcal{C}_i$$

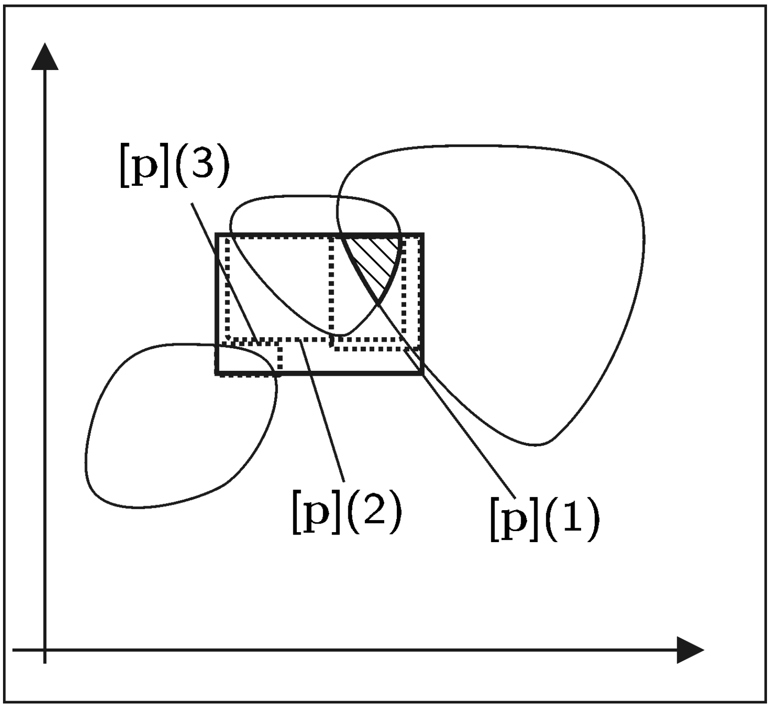

$\mathbb{P}_1$
$\mathbb{P}_2$
$\mathbb{P}_3$
$\mathbb{P}^{\{1\}}$
[p]

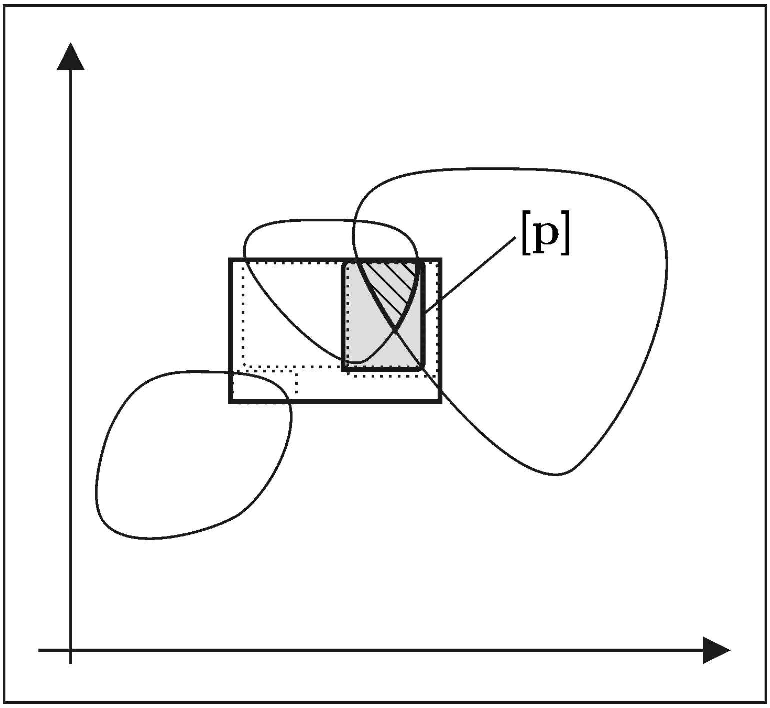


[p](1)
[p](2)
[p](3)




[p]


[p](3)
[p](2)
[p](1)


[p]

3.3 Test case

Generation of data. $m = 500$ data

$$\begin{cases} y_i = p_1 \sin(p_2 t_i) + e_i, & \text{with a probability } 0.2. \\ y_i = r_1 \exp(r_2 t_i) + e_i, & \text{with a probability } 0.2. \\ y_i = n_i \end{cases}$$

where $t_i = 0.02.i$, $i \in \{1, 500\}$, $e_i : \mathcal{U}([-0.1, 0.1])$ and $n_i : \mathcal{N}(2, 3)$.

We took $\mathbf{p}^* = (2, 2)^\mathsf{T}$ and $\mathbf{r}^* = (4, -0.4)^\mathsf{T}$.

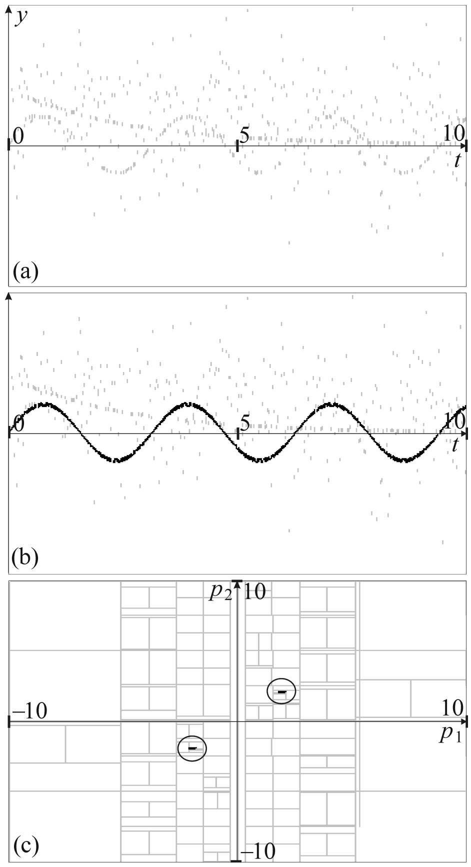
Estimation. We only know that

$$y_i = p_1 \sin(p_2 t_i) + e_i, \text{with a probability } 0.2.$$

We want

$$\Pr\left(\mathbf{p}^* \in \widehat{\mathbb{P}}\right) \geq 0.95$$

Since $\gamma(414, 500, 0.2) = 0.0468$ and $\gamma(413, 500, 0.2) = 0.12$, we should assume $q = 414$ outliers.



y
0
5
10
t
(a)
0
5
10
t
(b)
p_2
10
−10
10
p_1
−10
(c)

3.4 State estimation

$$\begin{cases} \mathbf{x}(k+1) & = & \mathbf{f}_k(\mathbf{x}(k), \mathbf{n}(k)) \\ \mathbf{y}(k) & = & \mathbf{g}_k(\mathbf{x}(k)), \end{cases}$$

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

Without outliers

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

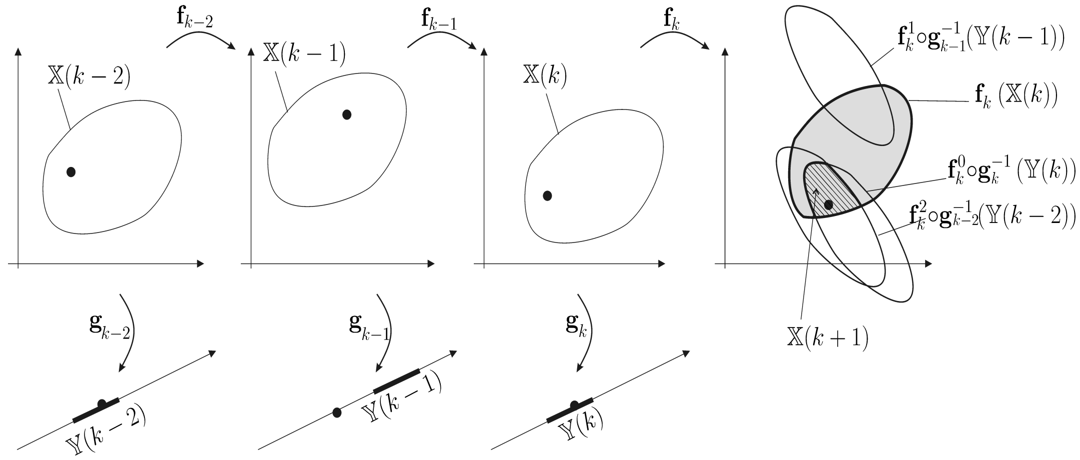
Define

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

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

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



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

We assume
(i) within any time window of length m we have less than q outliers and
(ii) $\mathbb{X}(0)$ contains $\mathbf{x}(0)$, then $\mathbb{X}(k)$ encloses $\mathbf{x}(k)$.

What is the probability of this assumption ?

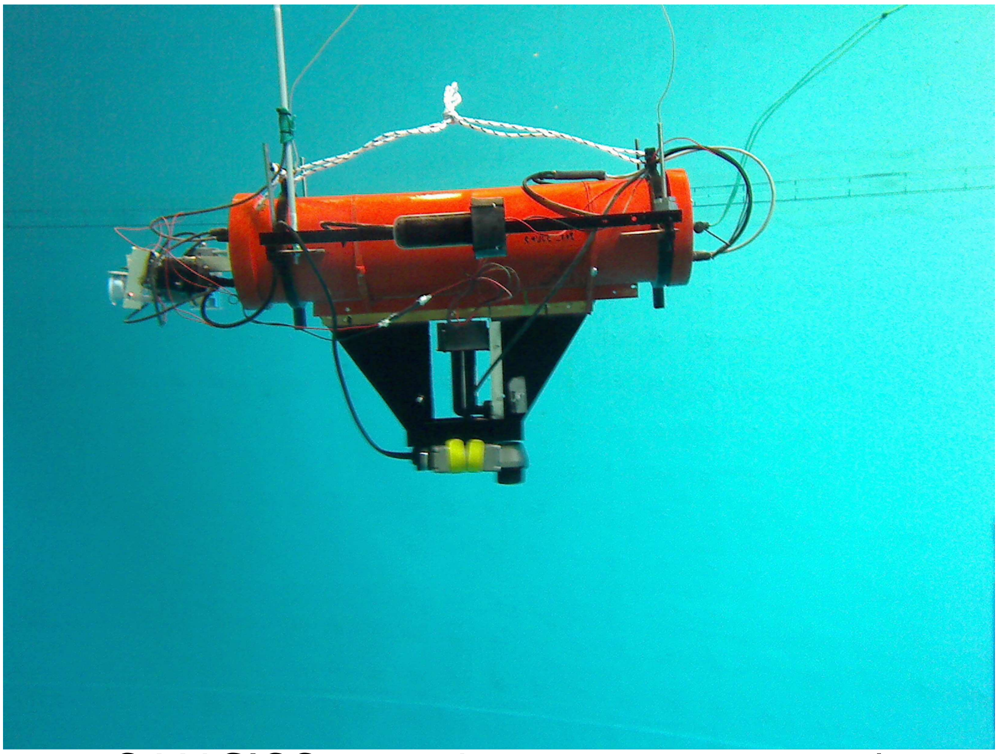
Theorem. Consider the sequence of sets $\mathbb{X}(0), \mathbb{X}(1), \ldots$ built by the set observer. We have

$$\Pr\left(\mathbf{x}\left(k\right) \in \mathbb{X}(k)\right) \geq \alpha \; * \; \Pr\left(\mathbf{x}\left(k-1\right) \in \mathbb{X}(k-1)\right)$$

where

$$\alpha = \sqrt[m]{\sum_{i=m-q}^{m} \frac{m!\; \pi^i . \left(1-\pi\right)^{m-i}}{i!\left(m-i\right)!}}.$$

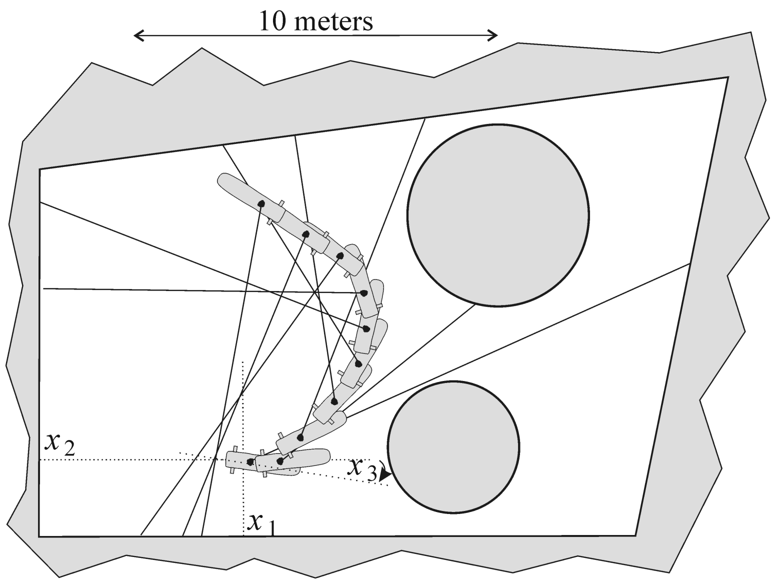
4 Application to localization



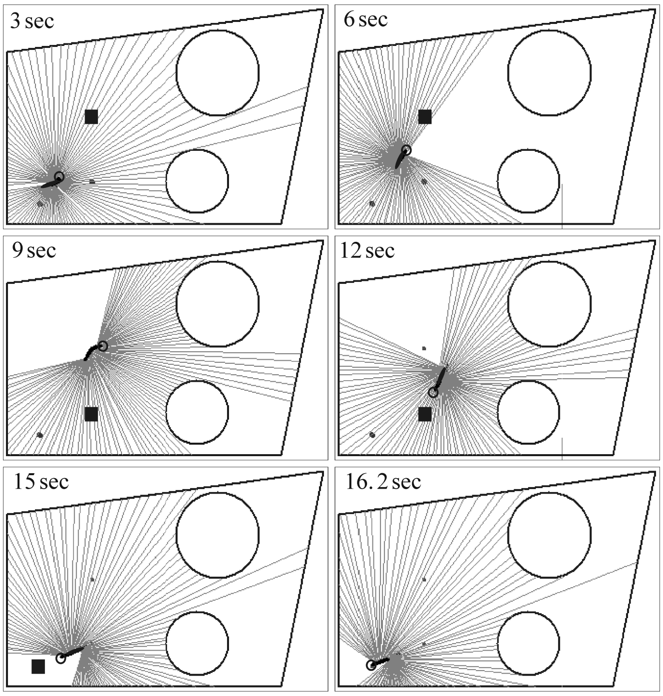
SAUCISSE inside a swimming pool

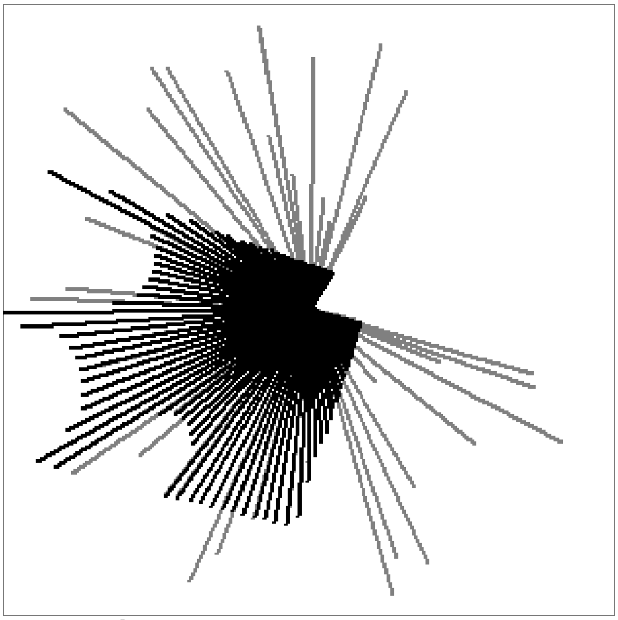
The robot evolution is

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

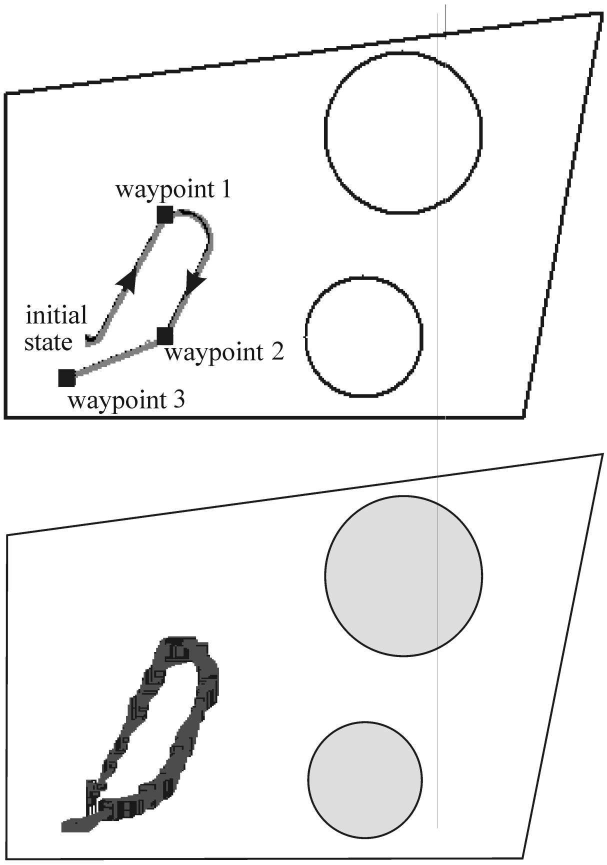


Underwater robot moving inside a pool


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



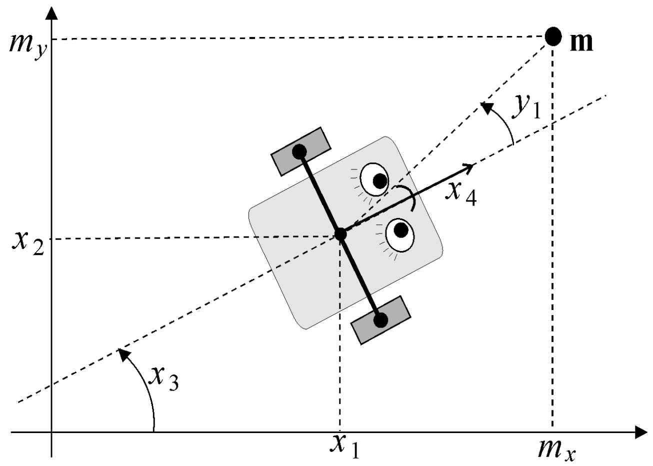
Emmision diagram at time $t = 16.2\,\text{sec}$



waypoint 1
initial state
waypoint 2
waypoint 3

t(sec)	$\Pr(\mathbf{x} \in \mathbb{X})$	Outliers
3.0	≥ 0.965	58
6.0	≥ 0.932	50
9.0	≥ 0.899	42
12.0	≥ 0.869	51
15.0	≥ 0.838	51
16.2	≥ 0.827	49

5 Comparison with the Kalman filter



A robot (unicycle type) which measures the angle y_1 corresponding to the mark $\mathbf{m}$

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

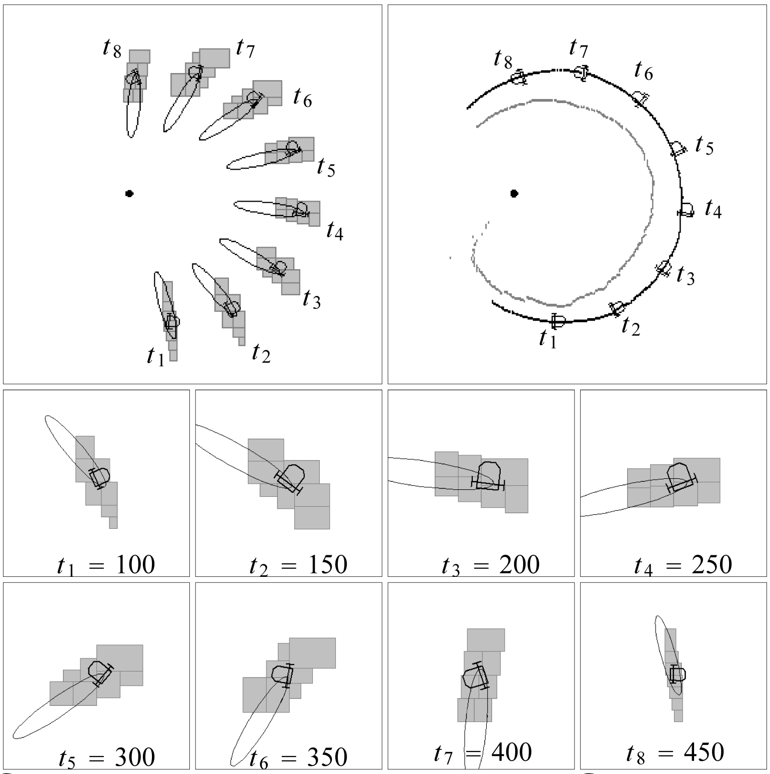
$$\begin{cases} y_1 & = & \text{atan2}\left(m_y - x_2, m_x - x_1\right) + x_3, \quad k \in \mathbb{Z} \\ y_2 & = & x_3 \\ y_3 & = & x_4. \end{cases}$$

Scenario 1. The measurement noises as well as the state noises are all Gaussian and centered with a variance of 0.01.

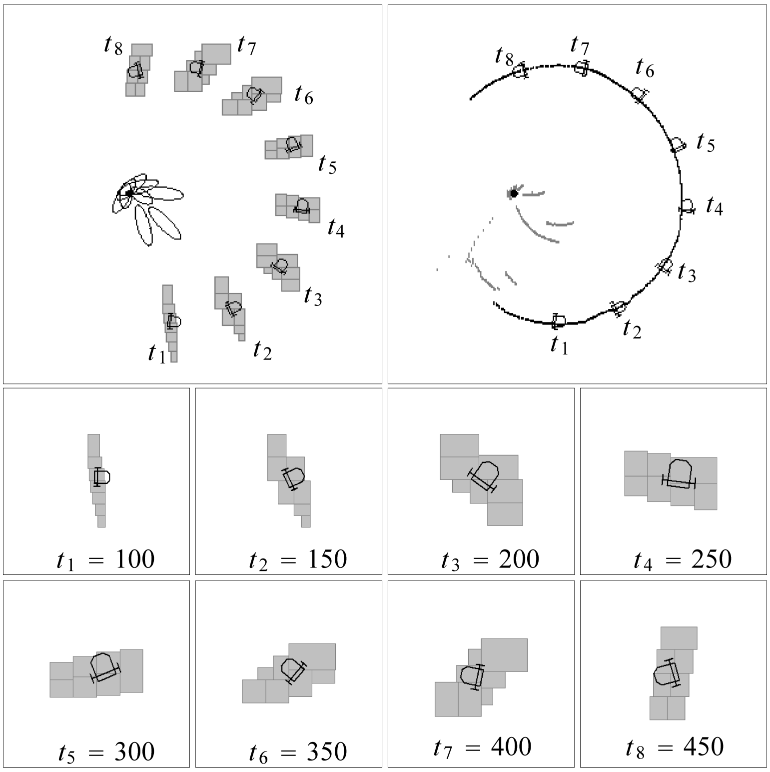
Scenario 2. With a probability of 5%, an outlier for y_1 is generated.

Scenario 3. This scenario is similar to Scenario 1 but a bias of 0.5 is added to y_1.

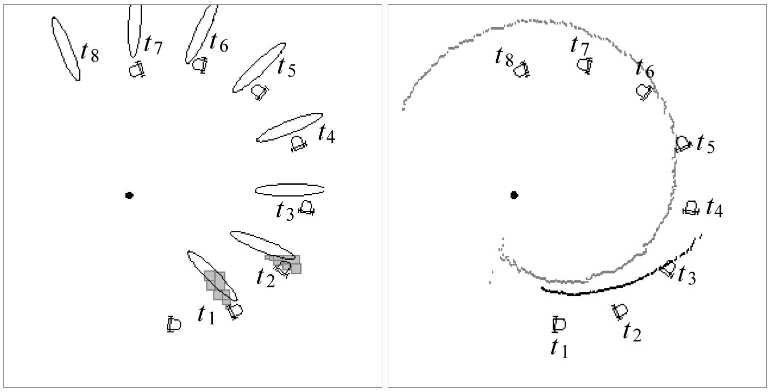
For RSO, $m = 50, q = 10$.



Scenario 1: All noises are Gaussian



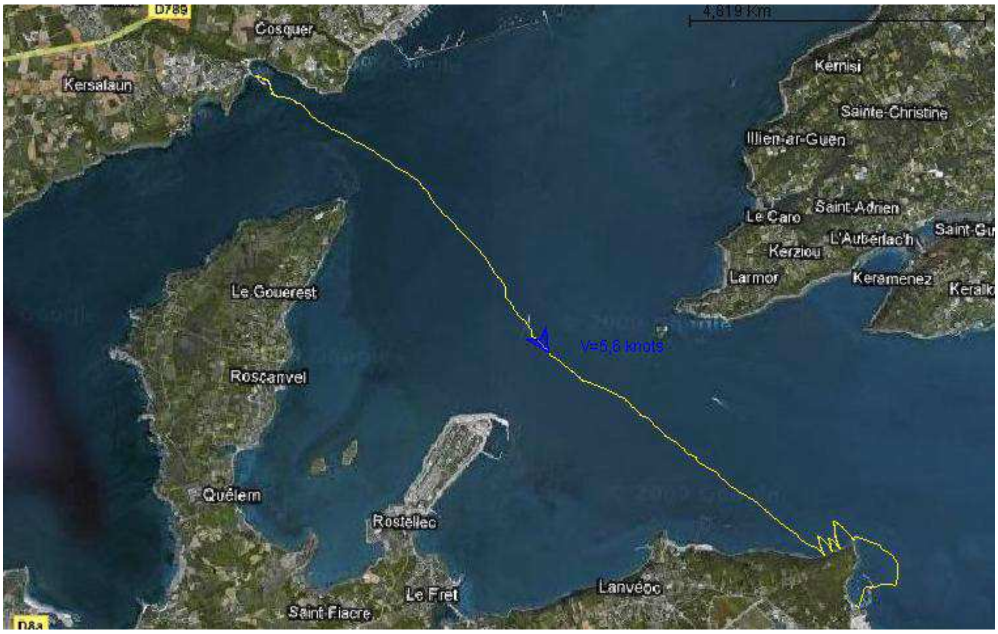
Scenario 2: **1%** of the data are outliers



Scenario 3. An unknown bias has been added to y_1.

6 Robot voilier


BREIZH
ENSIETA
SPIRIT


D789
Cosquer
4,819 Km
Kersalaun
Kernisi
Sainte-Christine
Illien-ar-Guen
Le Caro
Saint-Adrien
L'Auberlac'h
Kerziou
Larmor
Keramenez
Le Gouérest
Roscanvel
Quélern
Rostellec
Lanvéoc
Le Fret
Saint Fiacre

7 Conclusion

1) Les équations représentent les relations entre des variables mal connues. On leur associe un contracteur.
2) La méthode se distribue et se parallélise facilement.
3) La méthode ne linéarise pas.
4) Elle permet de prendre en compte des variables discrètes (entières, booléennes, . . .).
5) Elle est robuste par rapport aux outliers.
6) Venez tous à SWIM'11, June 14-15, 2011 à Bourges.