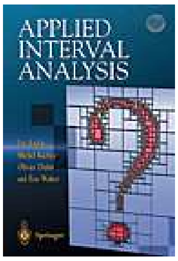
Méthodes ensemblistes pour la robotique



Luc Jaulin, DTN, ENSIETA
Mercredi 19 janvier 2010,

1 Approche ensembliste

1.1 Calcul par intervalles

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

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

Par exemple,

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

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

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

Par exemple,

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

1.2 Projection de contraintes

Soient x, y, z trois variables telles que

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

Les valeurs < 2 pour x, < 1 pour y et > 9 pour z sont inconsistantes.

1.3 Méthode numérique de projection

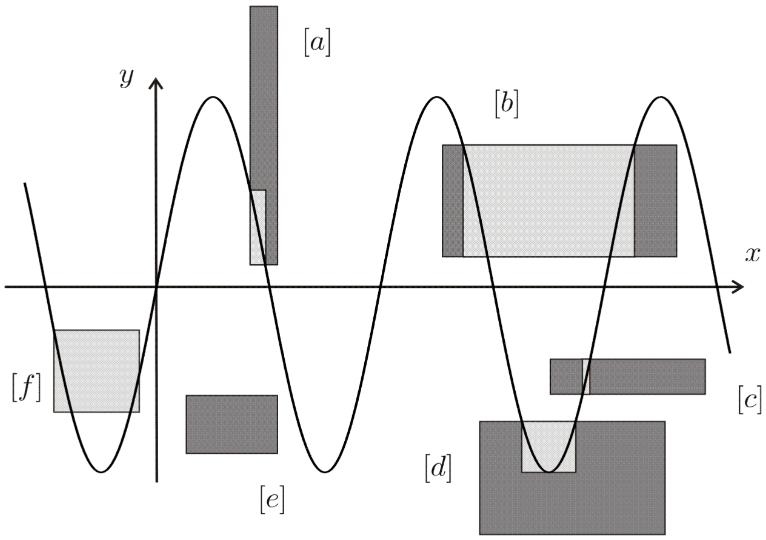
En effet, puisque $x \in [-\infty, 5], y \in [-\infty, 4], z \in [6, \infty]$ et $z = x + y$, nous avons

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

Pour la contrainte

$$y = \sin x,\ \ x \in [x], y \in [y]$$

le problème est un peu plus difficile.



1.4 Algorithme de propagation-bissection

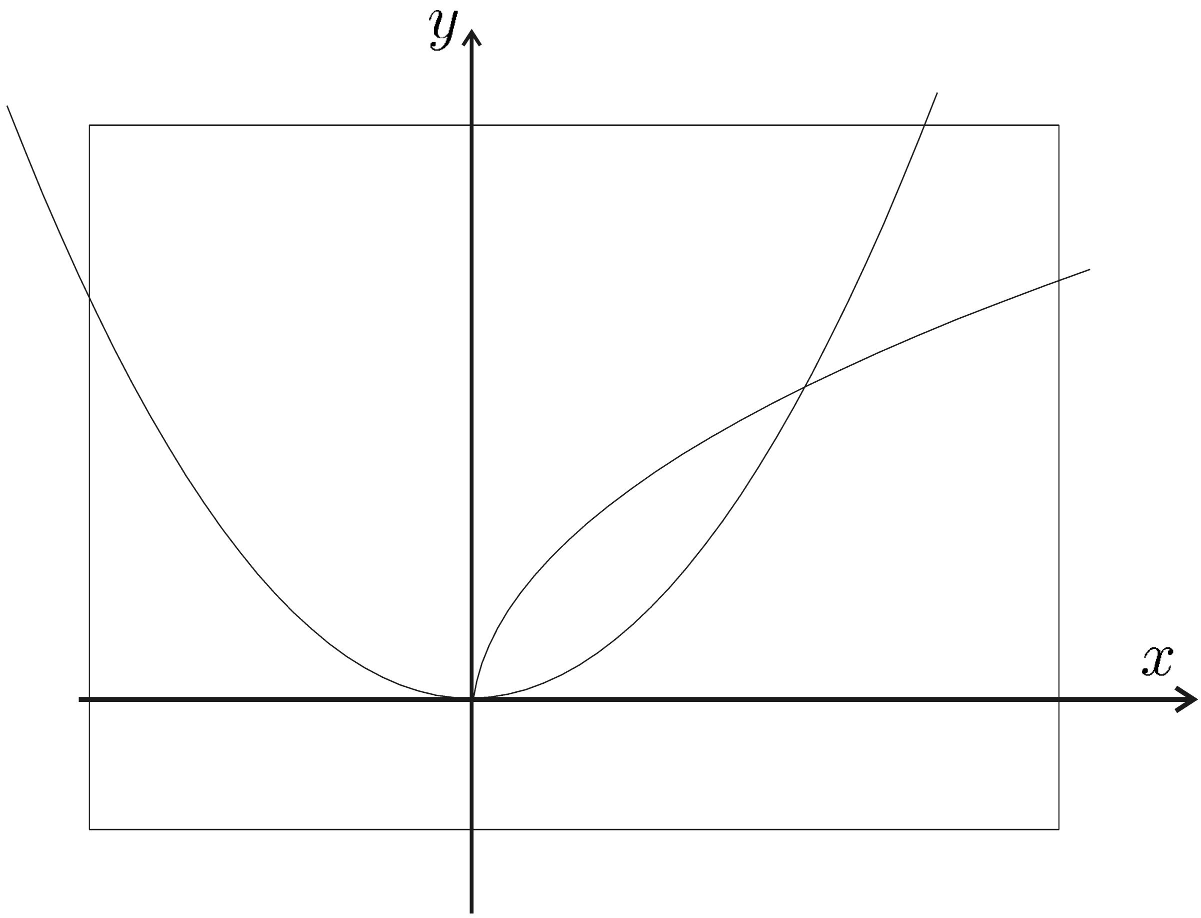
Exemple. Cherchons à résoudre.

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

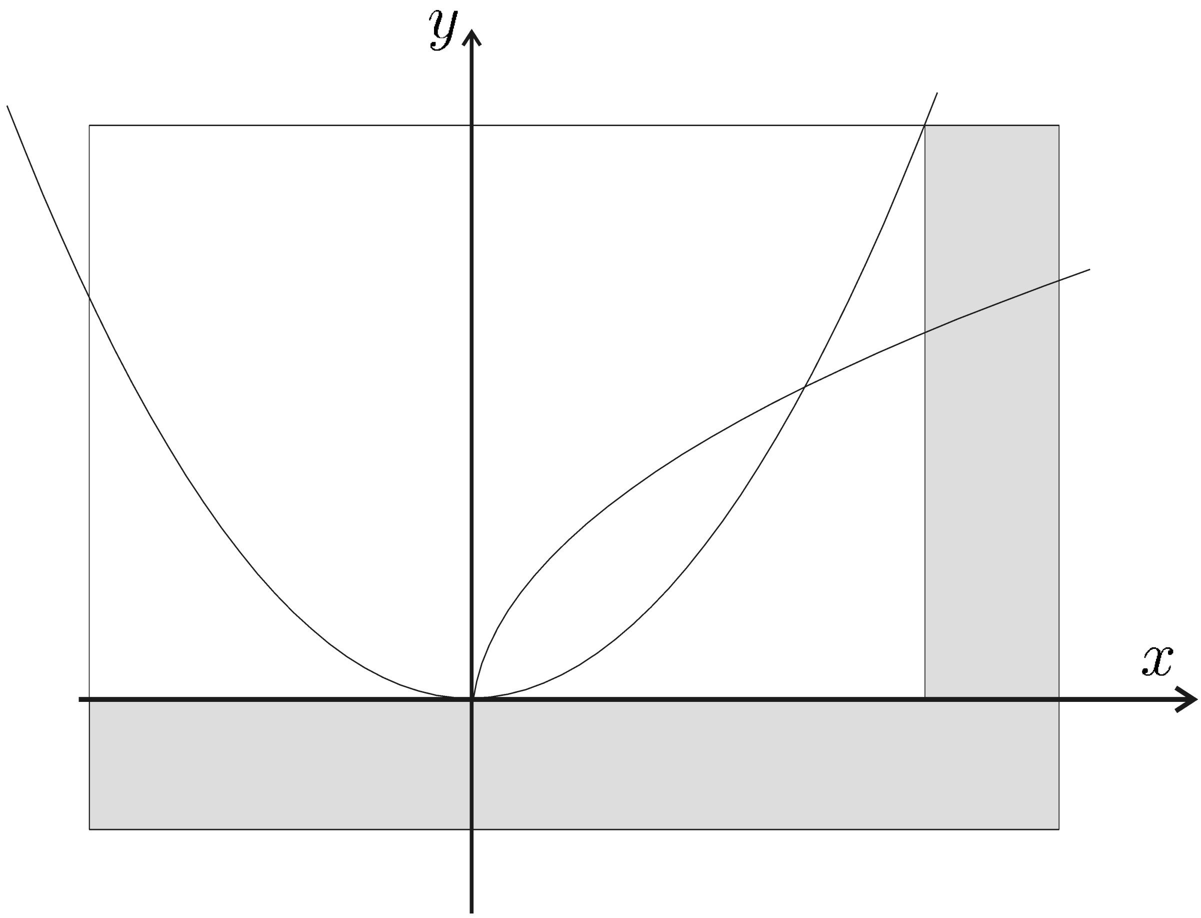
On a deux contracteurs

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

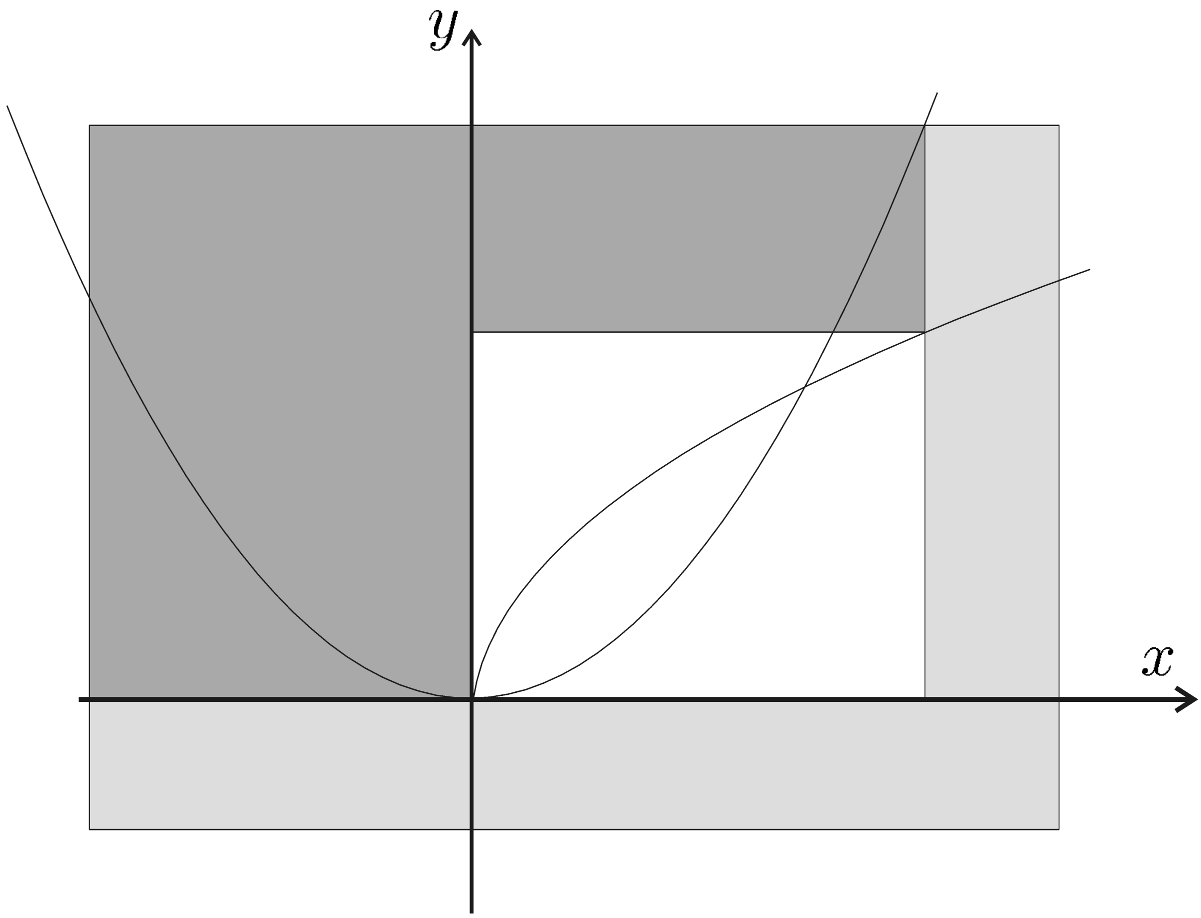
$$\mathcal{C}_2 : \left\{ \begin{array}{l} [y] = [y] \cap \sqrt{[x]} \\ [x] = [x] \cap [y]^2 \end{array} \right. \quad \text{associé à } y = \sqrt{x}$$

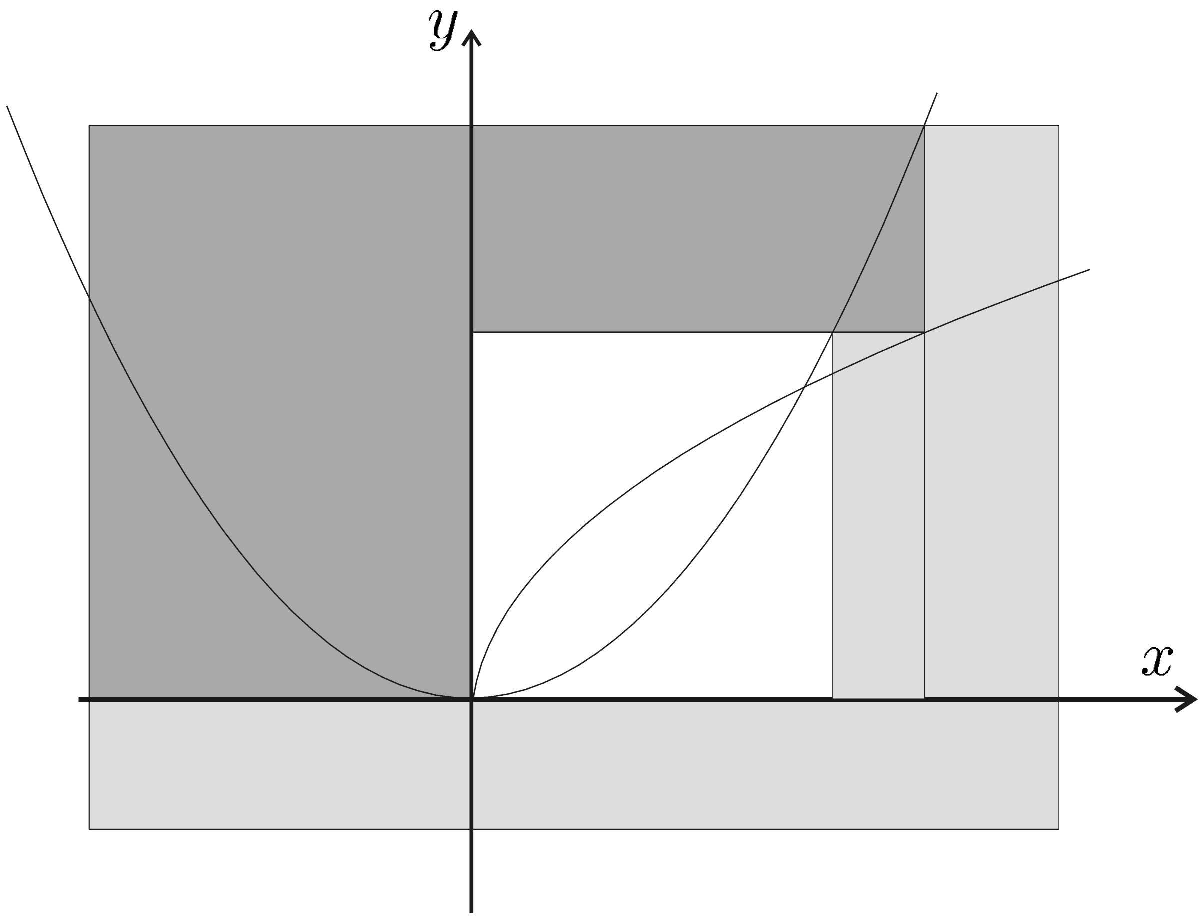


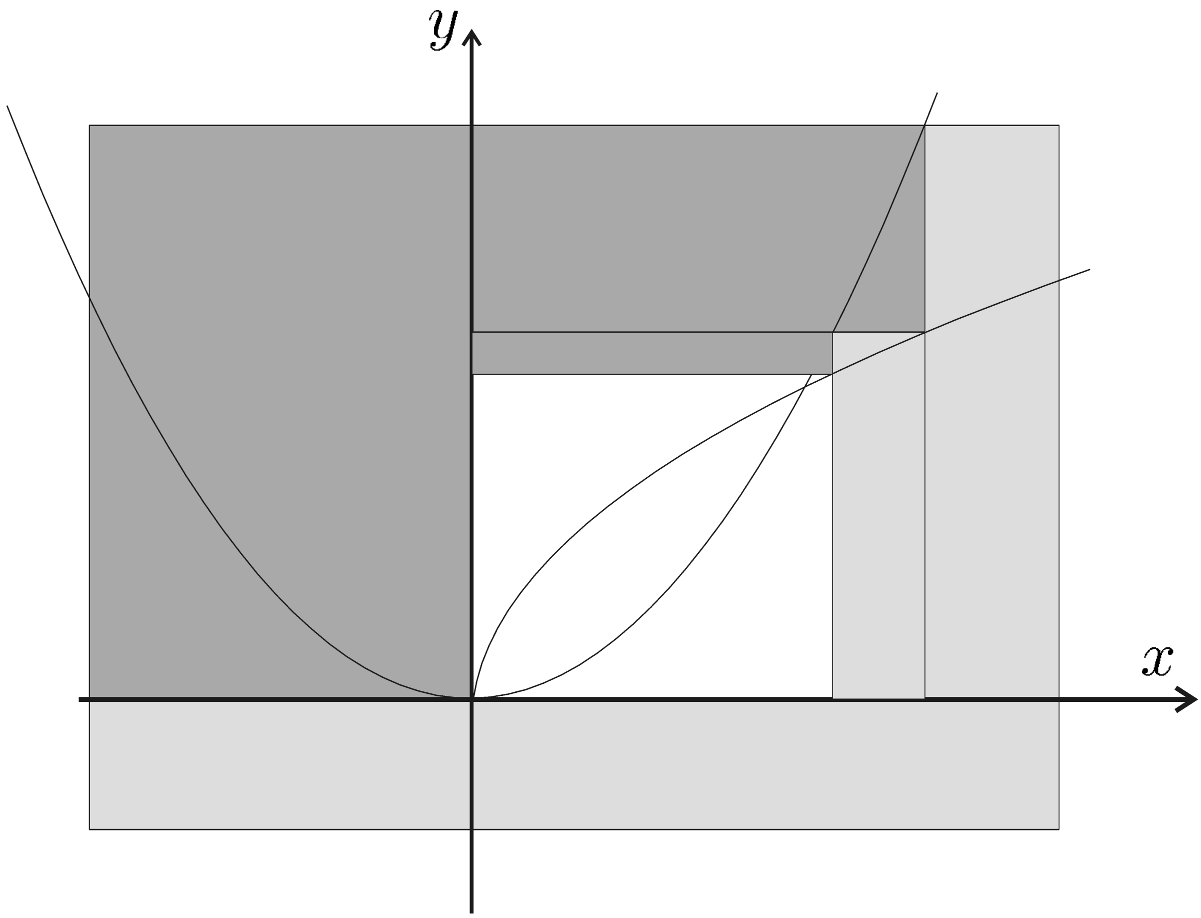
y
x



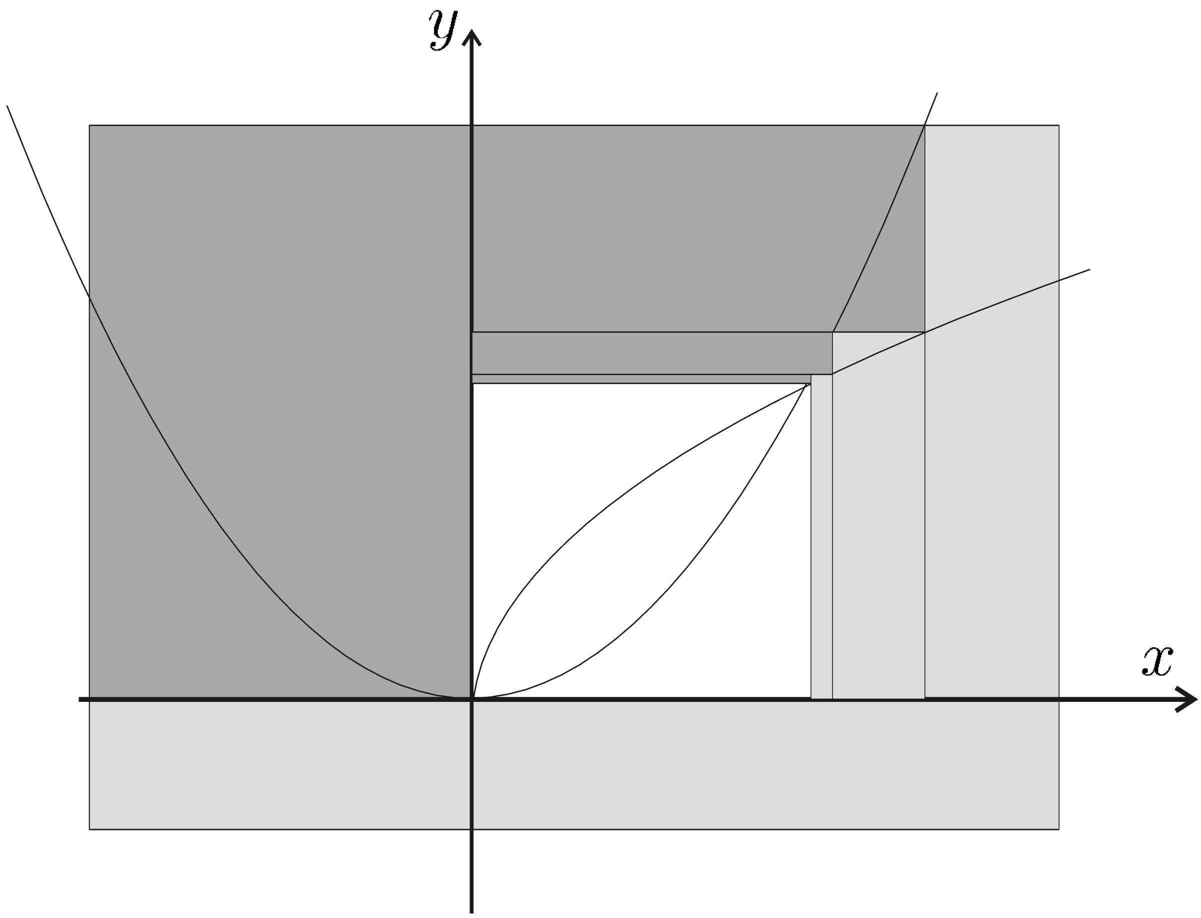
y
x


y
x

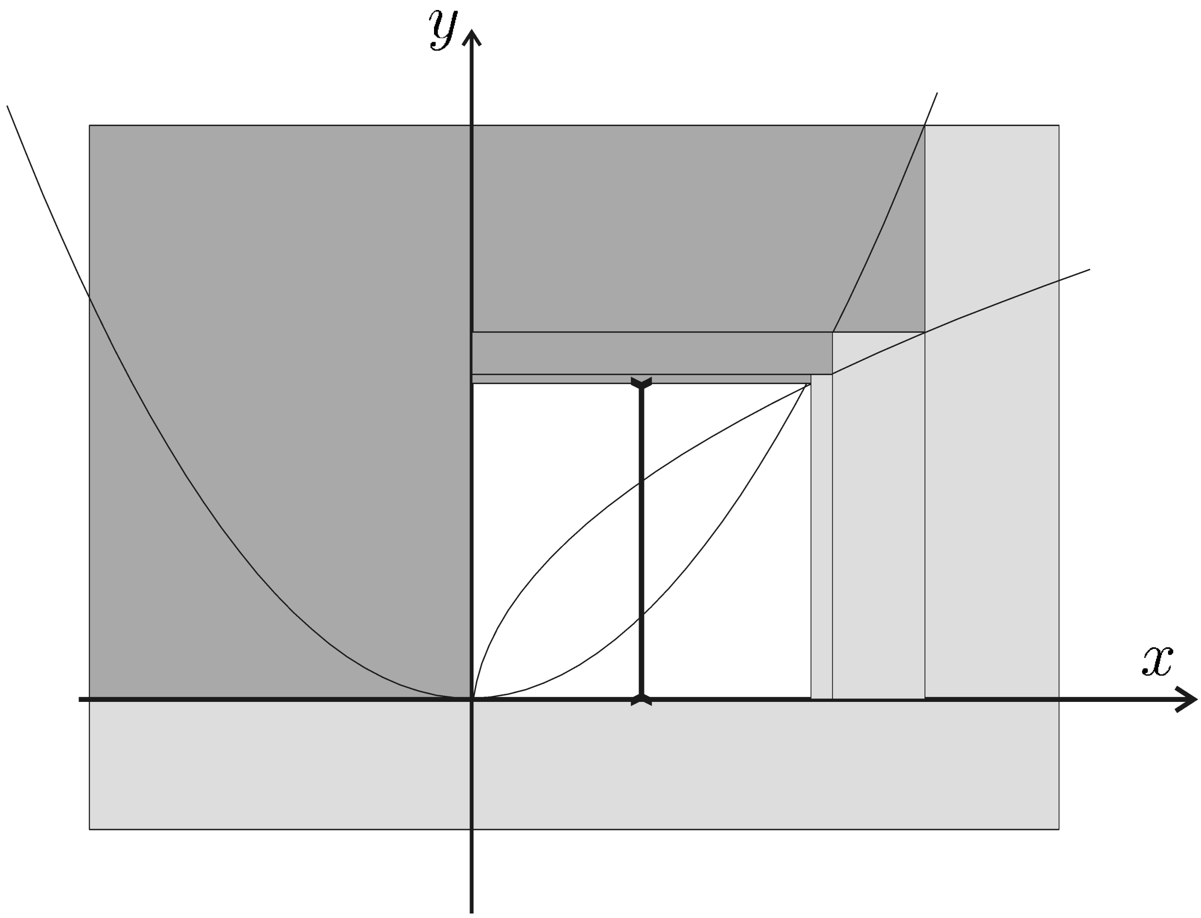

y
x



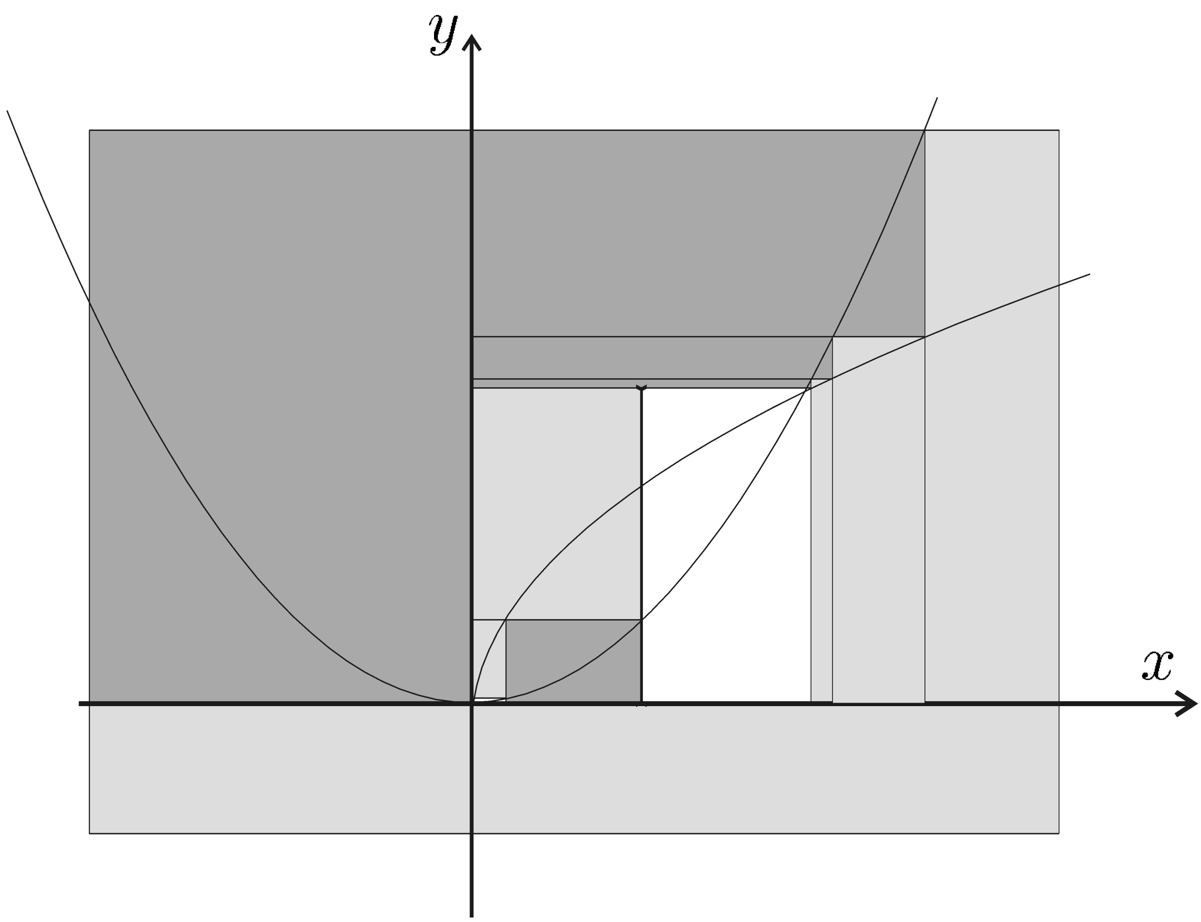
y
x



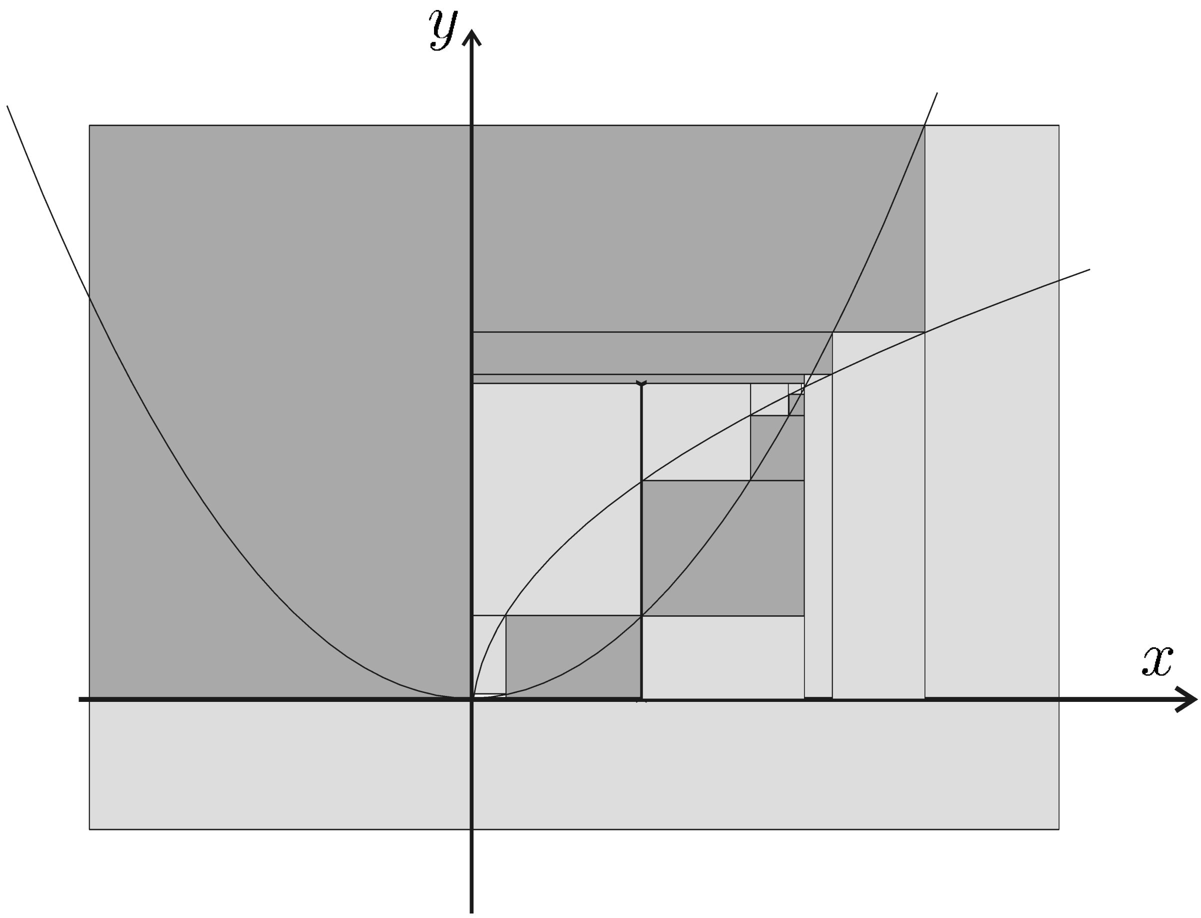
y
x



y
x



y
x


y
x

1.5 Décomposition

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

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

se décompose en

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

1.6 QUIMPER

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

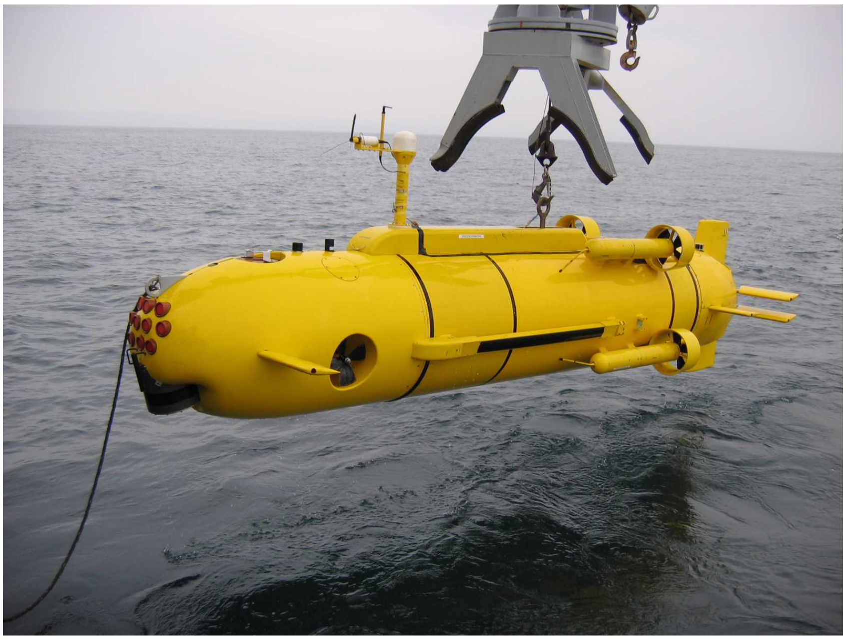
Quimper est un langage interprété pour le calcul ensembliste.

Un programme Quimper se décrit par un ensemble de contracteurs.

Logiciel libre disposible sur

`http://ibex-lib.org/`

2 SLAM



Le *Redermor*, fabriqué par le GESMA
(Groupe d'Etude Sous-Marine de l'Atlantique)

Pourquoi une approche par intervalles ?

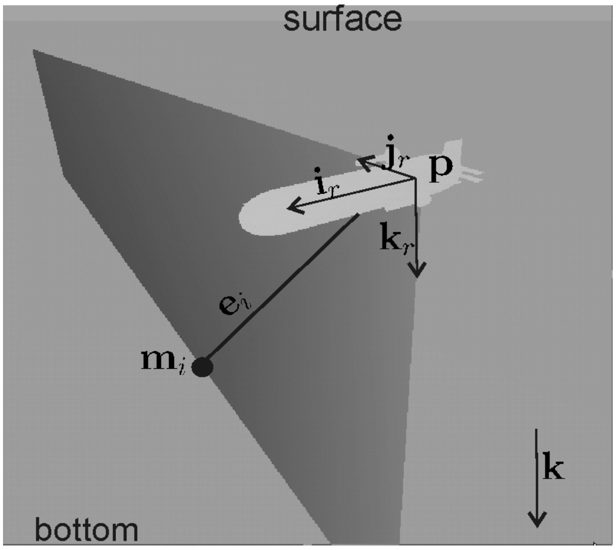
1) Besoin d'une approche fiable.
2) Les équations du robot sont non linéaires.
3) Les bruits de mesure sont non gaussiens.
4) Des bornes sur les erreurs sont fournies par les constructeurs des capteurs.
5) Beaucoup de mesures redondantes sont disponibles.

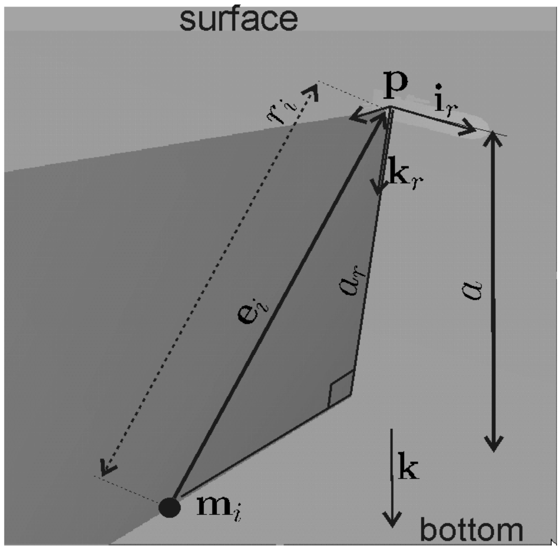
2.1 Capteurs

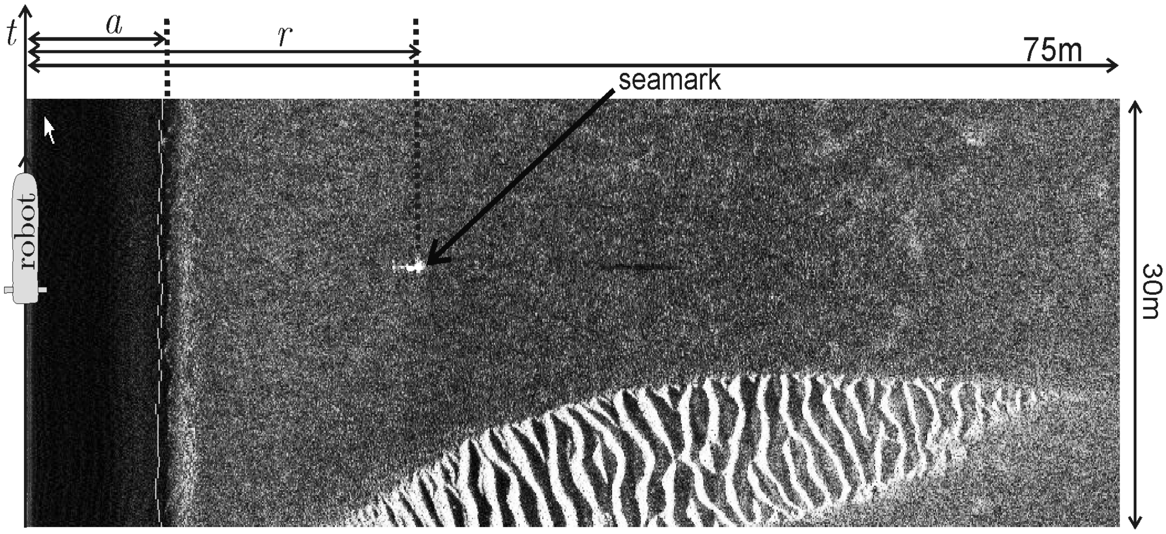
Un GPS (Global positioning system), disponible à la surface.

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

Un sonar (KLEIN 5400 side scan sonar). Donne la distance r entre le robot et la mine







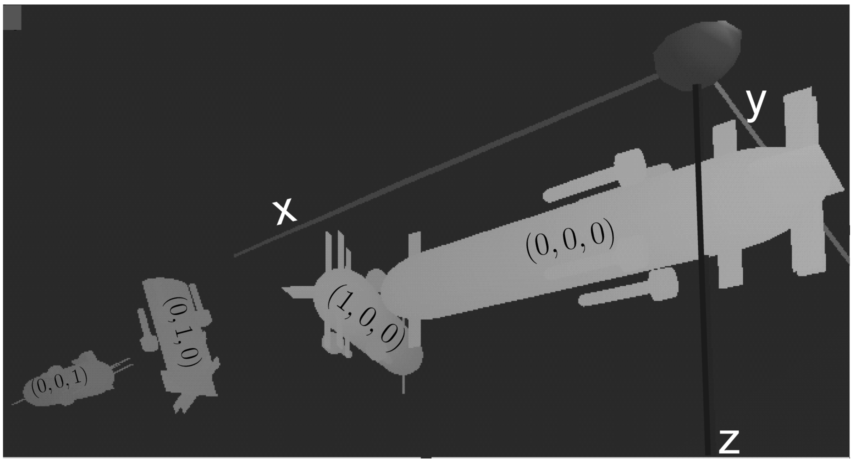
Détection d'une mine à l'aide de SonarPro

Le Loch-Doppler renvoie la vitesse du robot $\mathbf{v}_r$ et son altitude a.

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

Une centrale inertielle (Octans III from IXSEA) renvoie le roulis ϕ, le tangage θ et le cap ψ du robot.

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



2.2 Données

Pour chaque $t \in \{6000.0, 6000.1, 6000.2, \ldots, 11999.4\}$, nous avons des intervalles pour

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

Six mines ont été détectées par un opérateur humain, à l'aide de SonarPro.

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

2.3 Contraintes

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

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

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

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

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

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

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

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

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

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

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

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

```
//------------------------------------------------------
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
//------------------------------------------------------
```

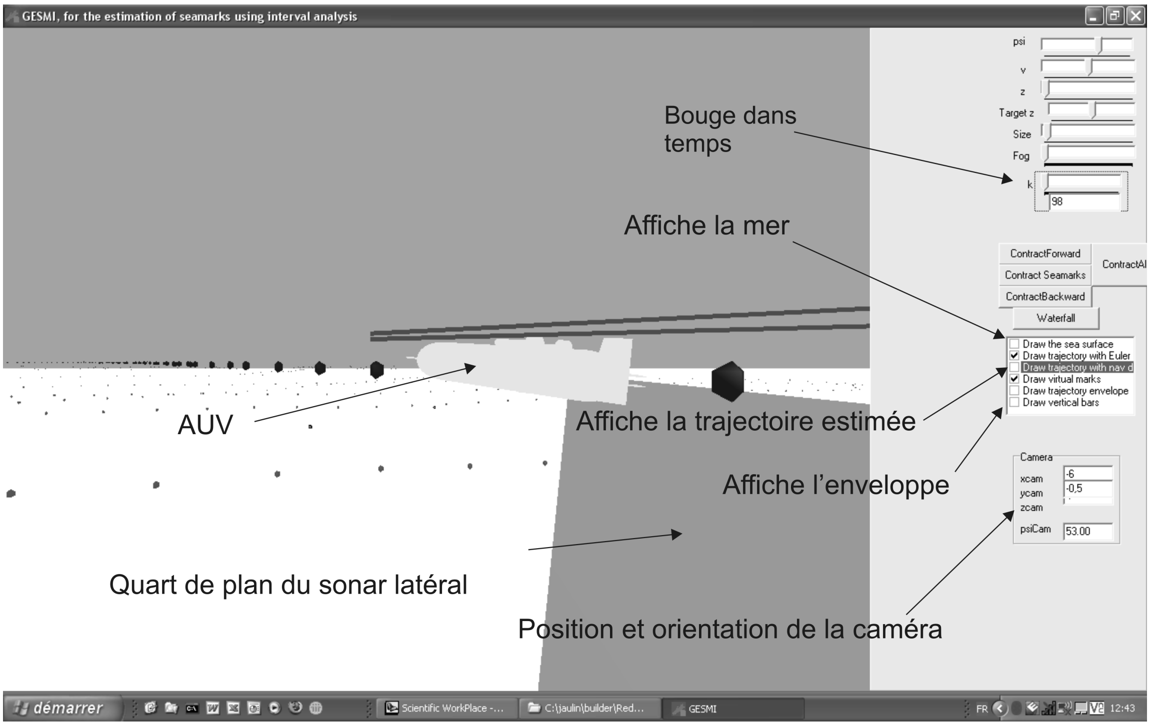
```
function R[3][3]=euler(phi,theta,psi)
  cphi = cos(phi);
  sphi = sin(phi);
  ctheta = cos(theta);
  stheta = sin(theta);
  cpsi = cos(psi);
  spsi = sin(psi);
  R[1][1]=ctheta*cpsi;
  R[1][2]=-cphi*spsi+stheta*cpsi*sphi;
  R[1][3]=spsi*sphi+stheta*cpsi*cphi;
  R[2][1]=ctheta*spsi;
  R[2][2]=cpsi*cphi+stheta*spsi*sphi;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

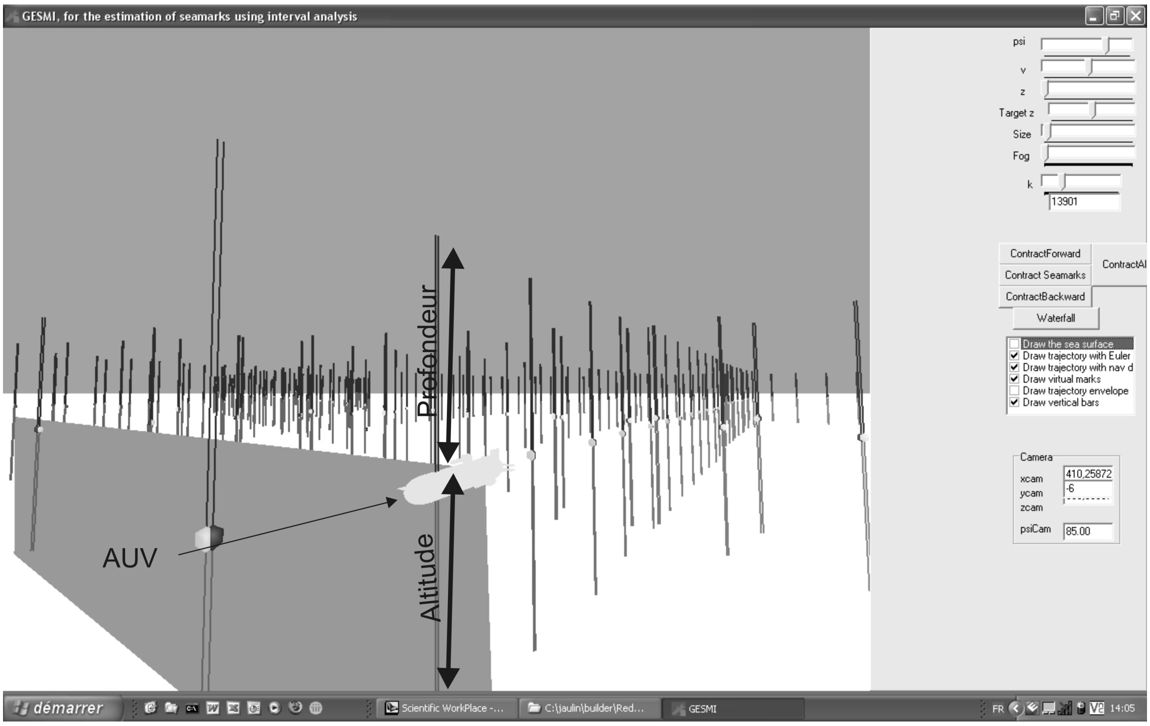
```
contractor fwd
  inter k=1:N-1;
    statequ(k)
  end
end
//------------------------------------------------
contractor bwd
  inter k=1:N-1;
    statequ(N-k)
  end
end
```

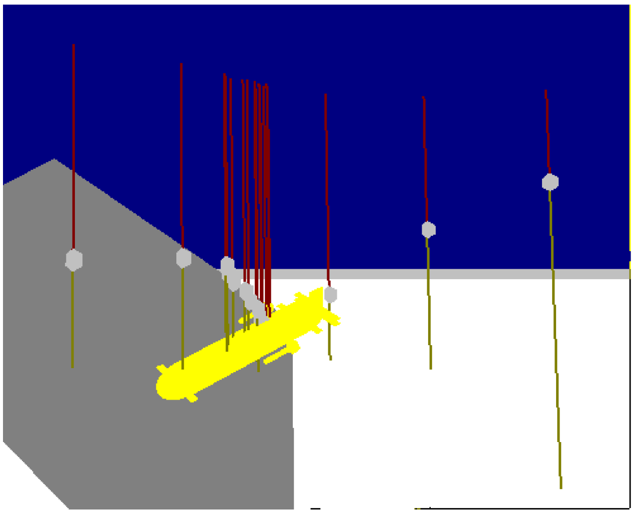
```
main
  p[1] :=read("gps_init.dat");
  v :=read("Quimper_v.dat");
  phi :=read("Quimper_phi.dat");
  theta :=read("Quimper_theta.dat");
  psi :=read("Quimper_psi.dat");
  init;
  fwd;
  bwd;
  column(p,px,1);
  column(p,py,2);
  print("--- Robot positions: ---");
  newplot("gesmi.dat");
  plot(px,py,color(rgb(1,1,1),rgb(0,0,0)));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

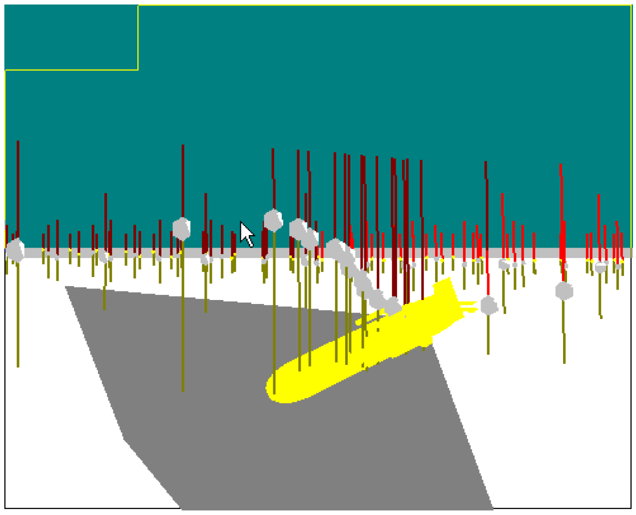
2.4 Logiciel GESMI

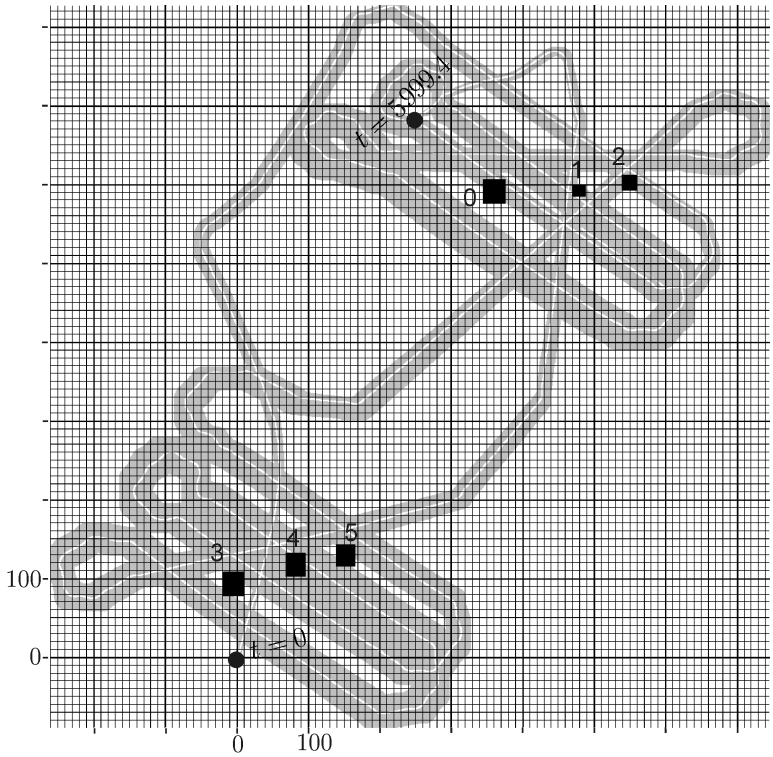


```
% This file has been generated by a generator
% and will be used by GESMI to solve a SLAM problem
% Note : every line starting by a '%' is considered as a comment by GESMI.
% ------------------------------------------------
%
% This file contains
% 1) Some basic information about the scope of the sonar and the sampling time
% 2) The prior domains for the seamarks
% 3) The coordinates for some virtual marks
% 4) sensor data (angles, speeds, depth, altitude, position) with bounds
% 5) ping table : at the end of this file
%
% ------------------------------------------------
% maximal distance of the lateral sonar and error error interval [min,max]
  75.0    1.0
% Sampling time
0.1
% ------------------------------------------------
% Domain for seamarks detected using the software SonarPro
% ------------------------------------------------
%  Initial domains for the mines
% xmin    xmax    ymin     ymax     zmin    zmax
-10000   10000  -10000   10000      0       100
-10000   10000  -10000   10000      0       100
-10000   10000  -10000   10000      0       100
-10000   10000  -10000   10000      0       100
-10000   10000  -10000   10000      0       100
-10000   10000  -10000   10000      0       100
$
%---------------------------------------------
% Virtual marks (only for graphism, not used for computation)
%---------------------------------------------
% x              y              z     colorRed   ColorGreen   ColorBlue  LocalFrame
594.0533723    374.7293035     19       0.1         0          0          0      % seamark 0
599.6093723    484.5528643     19       0.1        0.1         0          0      % seamark 1
601.4613723    557.3573086     19       0.1         0         0.1         0      % seamark 2
94.01337232    -2.868189501    19        0         0.1         0          0      % seamark 3
119.9413723    84.75155672     20        0          0         0.1         0      % seamark 4
127.3493723    156.3280366     20        0          0          0          0      % seamark 5
0             0          0       1       0         0     1    % origin of the local frame
689.96        264.10     0       0       0         1     1    % final GPS
$
%---------------------------------------------
%---------------------------------------------
%t       phi        dphi          theta      dtheta          psi       dpsi          vx         dvx
0    -0.011505    0.0001745329252    0.033556    0.0001745329252    0.415613    0.0052679
0.1     -0.012272    0.0001745329252    0.034994    0.0001745329252    0.416284    0.00526
0.2     -0.012847    0.0001745329252    0.037583    0.0001745329252    0.417051    0.00526
0.3     -0.013422    0.0001745329252    0.040938    0.0001745329252    0.418202    0.00526
0.4     -0.014093    0.0001745329252    0.044773    0.0001745329252    0.419831    0.00526
0.5     -0.014285    0.0001745329252    0.0488    0.0001745329252    0.422228    0.0052679
0.6     -0.013998    0.0001745329252    0.052539    0.0001745329252    0.425488    0.00526
0.7     -0.012943    0.0001745329252    0.055703    0.0001745329252    0.42961    0.005267
0.8     -0.011121    0.0001745329252    0.0581    0.0001745329252    0.434308    0.0052679
0.9     -0.008725    0.0001745329252    0.059825    0.0001745329252    0.439773    0.00526
1    -0.005944    0.0001745329252    0.060496    0.0001745329252    0.445334    0.0052679
1.1     -0.003547    0.0001745329252    0.0604    0.0001745329252    0.450511    0.0052679
1.2     -0.001917    0.0001745329252    0.059729    0.0001745329252    0.455017    0.00526
1.3     -0.001438    0.0001745329252    0.058867    0.0001745329252    0.458564    0.00526
1.4     -0.002013    0.0001745329252    0.058387    0.0001745329252    0.461057    0.00526
1.5     -0.003356    0.0001745329252    0.058483    0.0001745329252    0.462303    0.00526
1.6     -0.005561    0.0001745329252    0.059346    0.0001745329252    0.462591    0.00526
1.7     -0.007958    0.0001745329252    0.060976    0.0001745329252    0.462112    0.00526
```

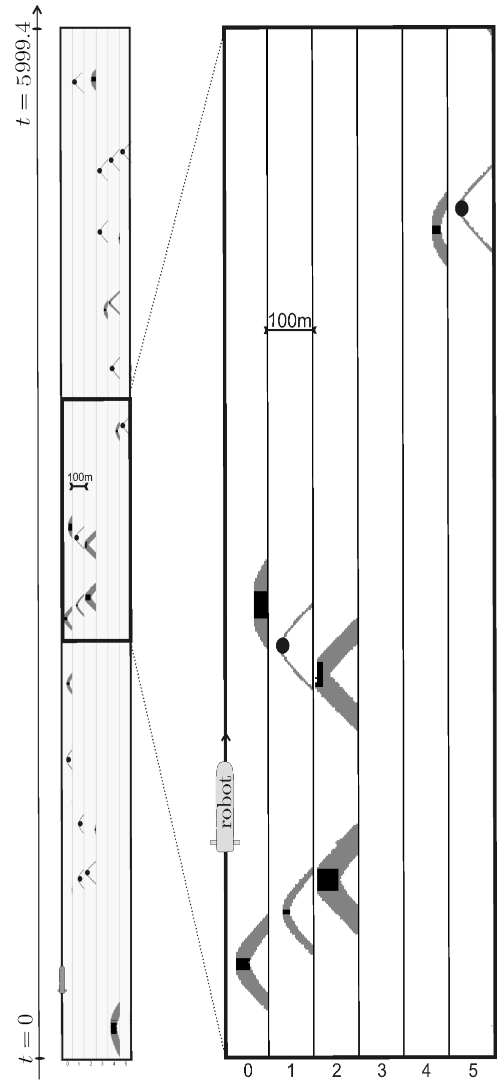

GESMI, for the estimation of seamarks using interval analysis
Profondeur
AUV
Altitude
psi
v
z
Target z
Size
Fog
k
13901
ContractForward
Contract Seamarks
ContractBackward
ContractAl
Waterfall
Draw the sea surface
Draw trajectory with Euler
Draw trajectory with nav d
Draw virtual marks
Draw trajectory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







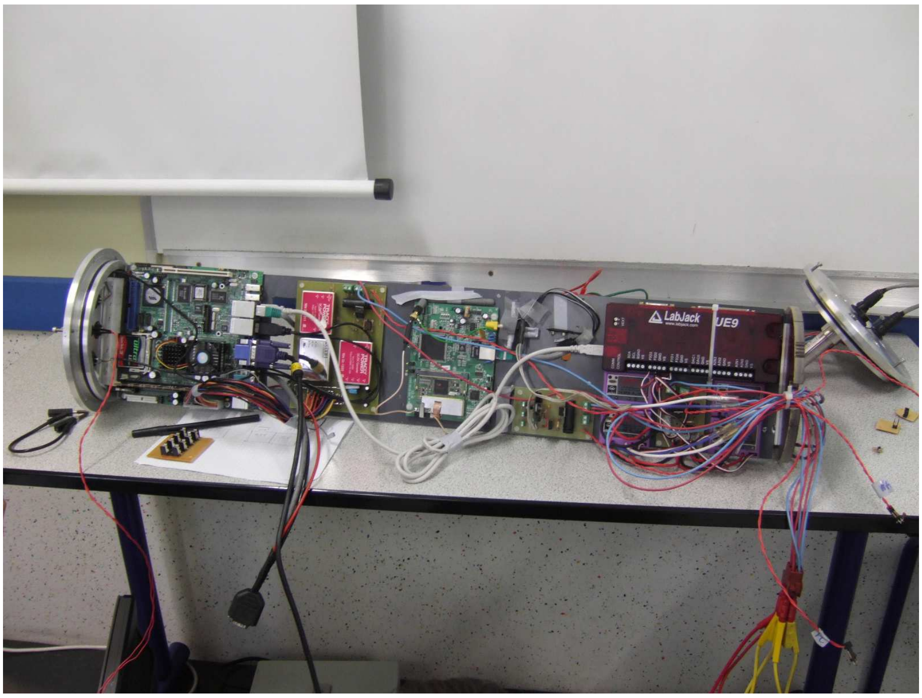
t = 5999.4
0
1
2
3
4
5
100
0
t = 0
0
100

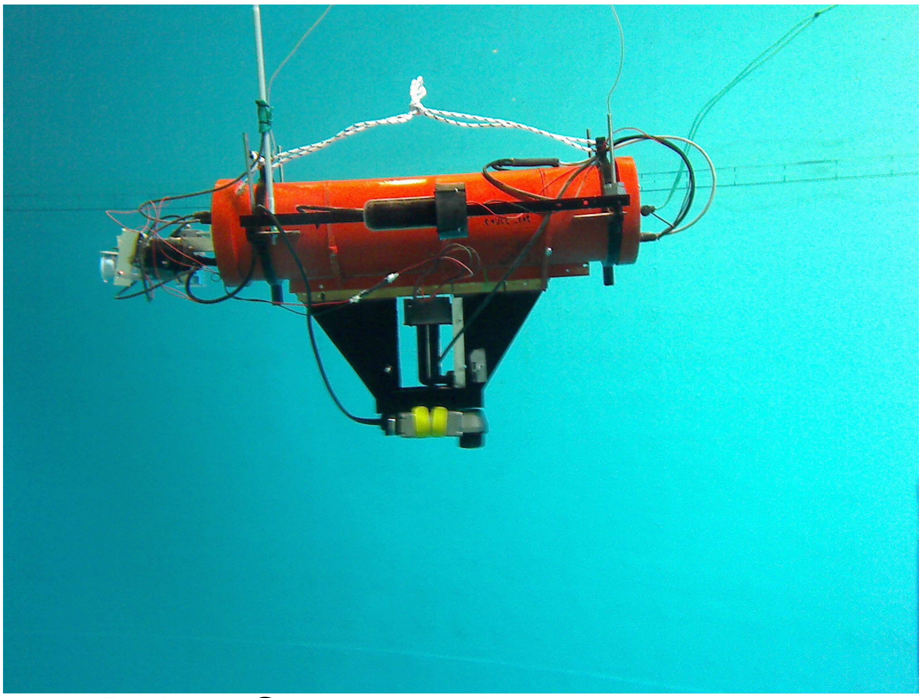


Reconstruction de la waterfall.

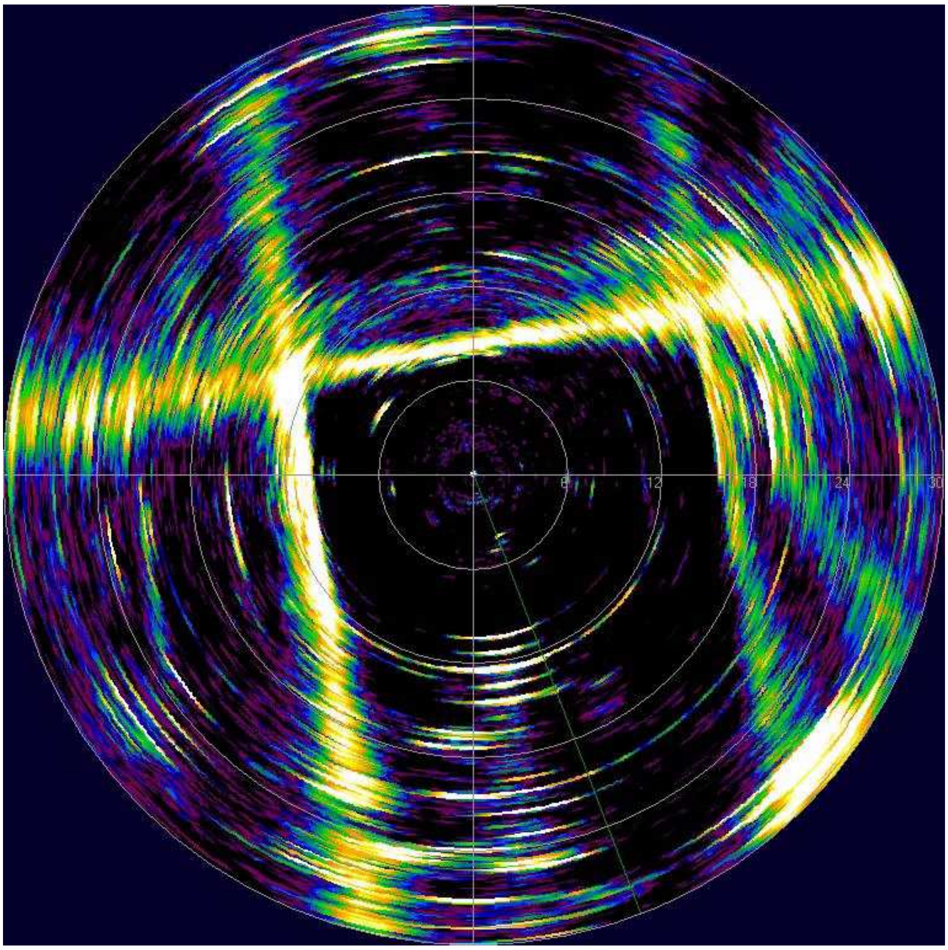
3 SAUCE

Portsmouth, July 12-15, 2007.

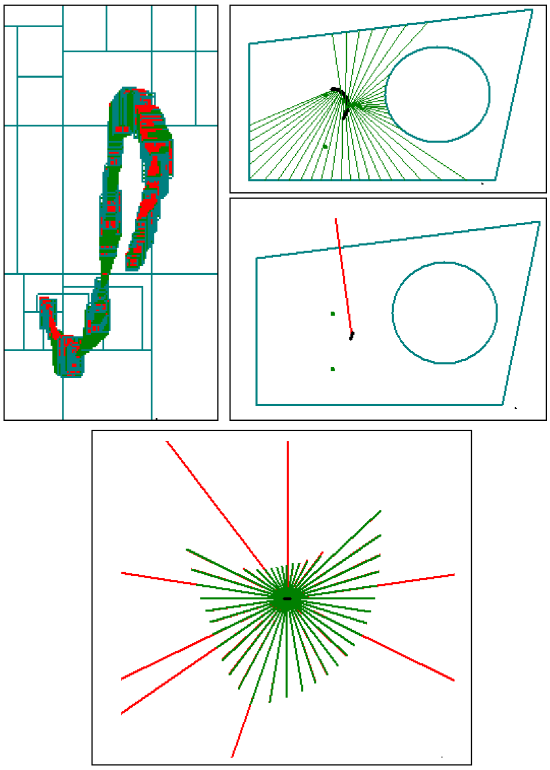

LabJack
UE9



Robot Sauc'isse dans une piscine



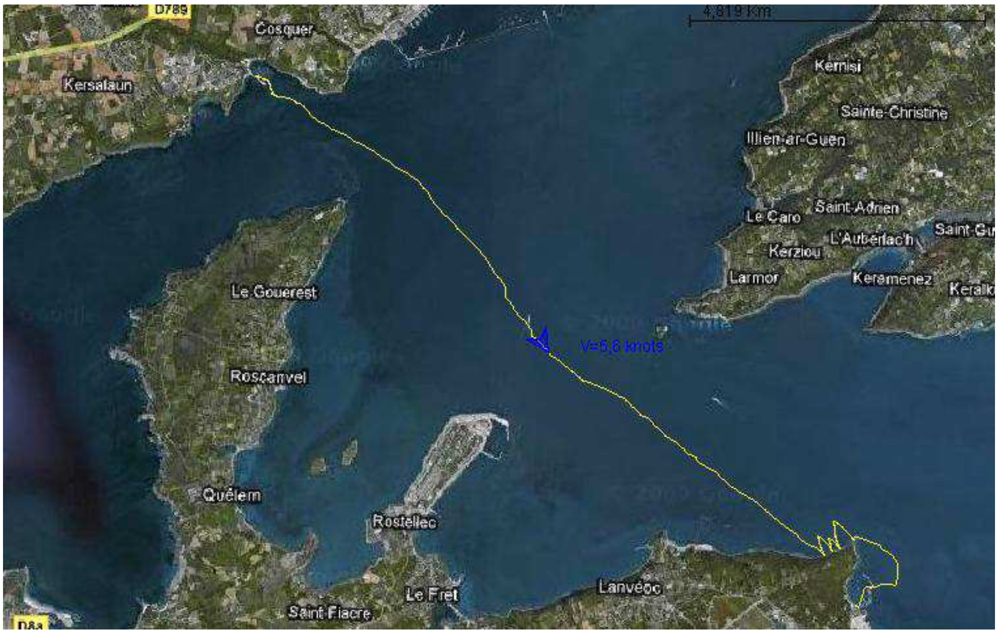
6
12
18
24
30



Montrer les images de la Marina

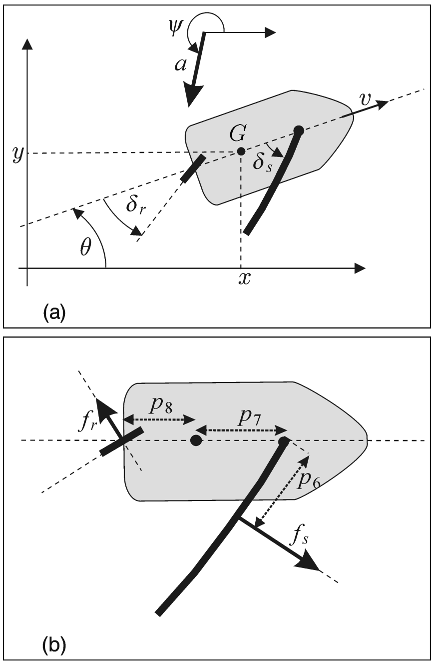
4 Breizh Spririt


BREIZH
ENSIETA
SPIRIT


4,819 Km
D789
Cosquer
Kersalaun
Kernisi
Sainte-Christine
Illien-ar-Guen
Le Caro
Saint-Adrien
L'Auberlac'h
Kerziou
Larmor
Keramenez
Le Gouérest
V=5,6 knots
Roscanvel
Quélern
Rostellec
Lanvéoc
Le Fret
Saint-Fiacre

4.1 Normalized State equations

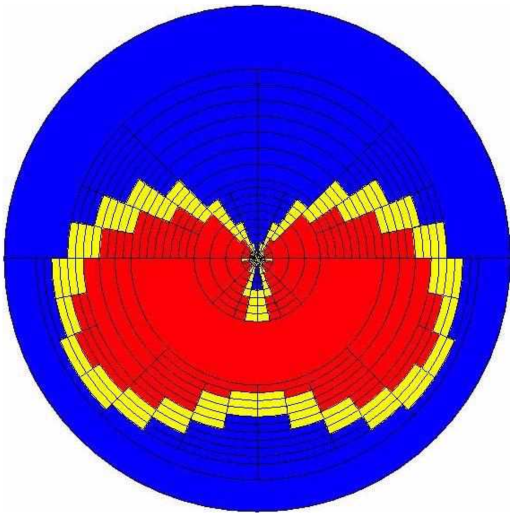
$$
\begin{cases}
\dot{x} & = & v \cos\theta + a \cos\psi \\
\dot{y} & = & v \sin\theta + a \sin\psi \\
\dot{\theta} & = & \omega \\
\dot{v} & = & f_s . \sin\delta_s - f_r . \sin u_1 - v \\
\dot{\omega} & = & f_s . (1 - \cos\delta_s) - f_r . \cos u_1 - \omega \\
f_s & = & a \sin(\theta - \psi + \delta_s) \\
f_r & = & v \sin u_1 \\
\gamma & = & \cos(\theta - \psi) + \cos(u_2) \\
\delta_s & = & \begin{cases} \pi - \theta + \psi & \text{if } \gamma \leq 0 \\ sign(\sin(\theta - \psi)) . u_2 & \text{otherwise.} \end{cases}
\end{cases}
$$

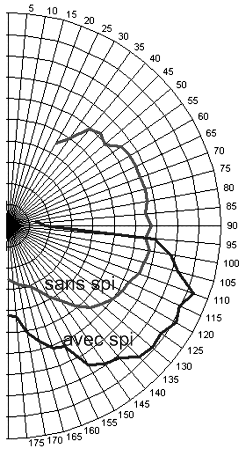


ψ
a
v
G
y
δs
δr
θ
x
(a)
fr
p8
p7
p6
fs
(b)

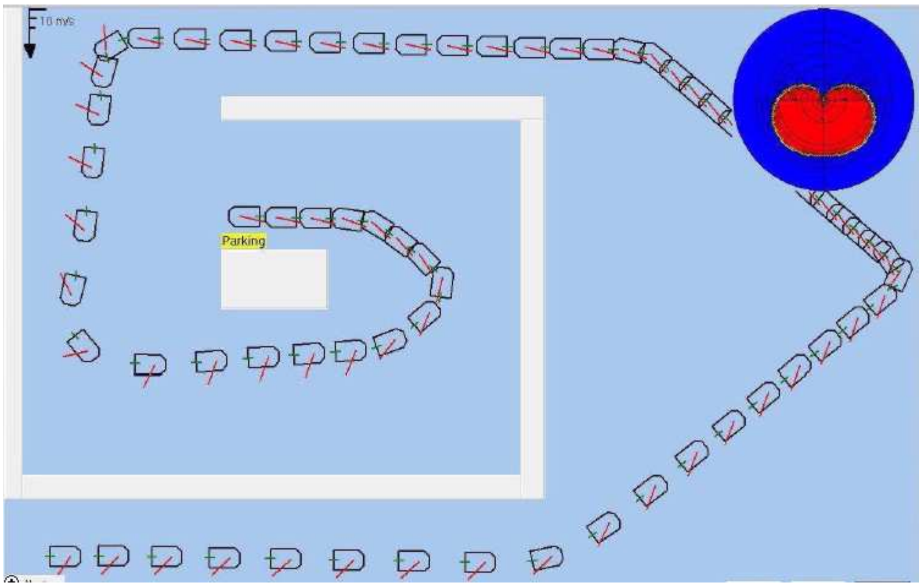
4.2 Polar speed diagram

$$
\begin{array}{lll}
\mathbb{W} = \{ & (\theta, v) \mid & \exists(\omega, u_1, u_2, f_s, f_r, \delta_r, \delta_s) \\
 & & \omega = 0, u_1 = 0, u_2 = 0 \\
 & & f_s \sin\delta_s - f_r \sin\delta_r - v = 0 \\
 & & (1 - \cos\delta_s)\, f_s - \cos\delta_r f_r = 0 \\
 & & f_s = a\cos(\theta + \delta_s) - v\sin\delta_s \\
 & & f_r = v\sin\delta_r \qquad \}.
\end{array}
$$



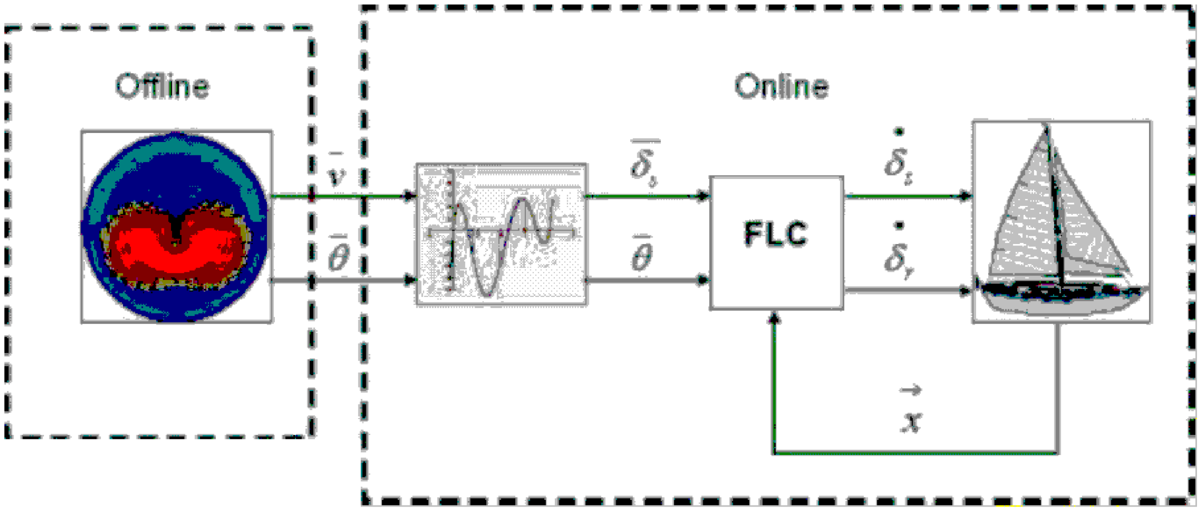


5
10
15
20
25
30
35
40
45
50
55
60
65
70
75
80
85
90
95
100
105
110
115
120
125
130
135
140
145
150
155
160
165
170
175
sans spi
avec spi



10 m/s
Parking

4.3 Control



4.4 Observer

From the state equations of the sailboat, it is easy to check that

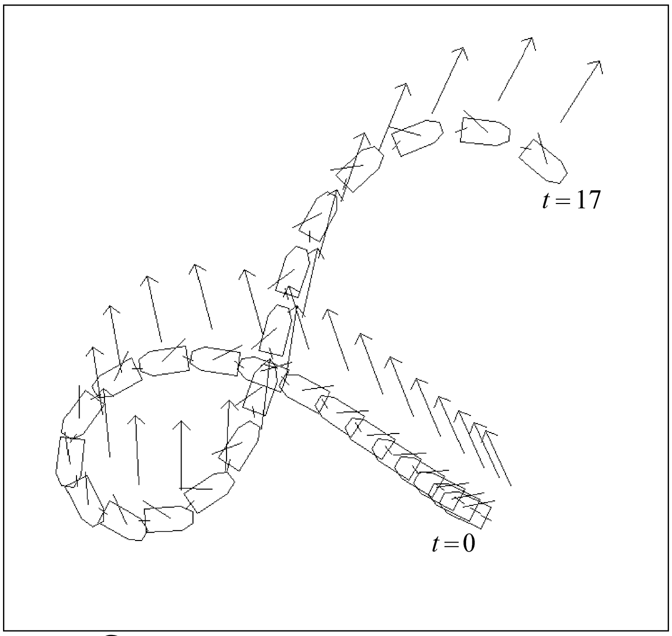
$$\underbrace{\begin{pmatrix} \theta \\ \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \ddot{x} \\ \ddot{y} \\ \ddot{\theta} \end{pmatrix}}_{\mathbf{z}} = \mathbf{\Psi}_t \underbrace{\begin{pmatrix} \theta \\ v \\ \omega \\ a \\ \psi \end{pmatrix}}_{\mathbf{x}}$$

with

$$\mathbf{\Psi}_t\left(\mathbf{x}\right)=\left(\begin{array}{c}\theta\\ v\sin\theta+a\sin\psi\\ v\sin\theta+a\sin\psi\\ \omega\\ \left(f_s.\sin\delta_s-f_r\sin u_1-v\right)\cos\theta-\omega v\sin\theta\\ \left(f_s\sin\delta_s-f_r\sin u_1-v\right)\sin\theta+\omega v\cos\theta\\ f_s\left(1-\cos\delta_s\right)-f_r\cos u_1-\omega\end{array}\right)$$

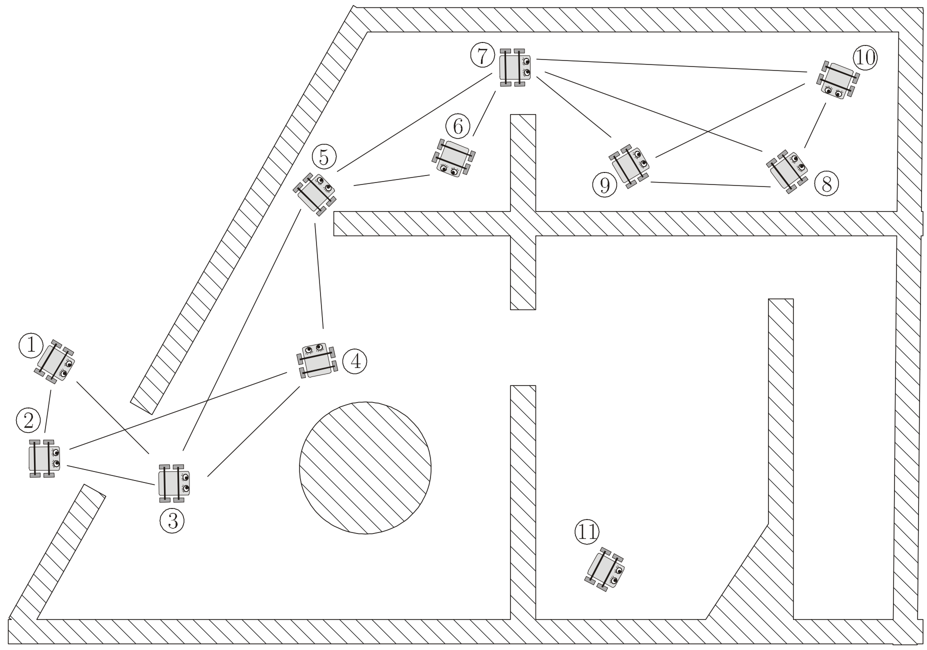
and

$$\begin{cases} f_s(\mathbf{x}) & = & a \sin(\theta - \psi + \delta_s) \\ f_r(\mathbf{x}, t) & = & v \sin u_1 \\ \delta_s(\mathbf{x}, t) & = & \begin{cases} \pi - \theta + \psi & \text{if } \gamma(\mathbf{x}, t) \leq 0 \\ sign(\sin(\theta - \psi)) .u_2 & \text{otherwise} \end{cases} \\ \gamma(\mathbf{x}, t) & = & \cos(\theta - \psi) + \cos(u_2) . \end{cases}$$



Simulated experiment

5 CAROTTE



1
2
3
4
5
6
7
8
9
10
11

6 CSPs

Un CSP est constitué.

- d'un ensemble de variables $\mathcal{V} = \{x_1, \ldots, x_n\}$,
- d'un ensemble de contraintes $\mathcal{C} = \{c_1, \ldots, c_m\}$
- d'un ensemble d'intervalles $\{[x_1], \ldots, [x_n]\}$.

1) Les informations sur la localisation d'un robot et sur sa carte peuvent être représentées par un CSP.

2) Les contraintes c_i représentent les relations entre les variables. On leur associe un contracteur.

3) Les CSP se distribuent facilement entre différents robots.

4) La méthode ne linéarise pas.

5) Elle permet de prendre en compte des variables discrètes (entières, booléennes, . . .).

6) Elle est robuste par rapport aux outliers.

7) Elle se parallélise aisément.