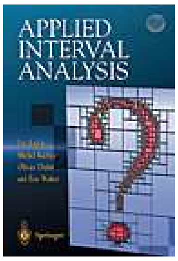
Programmation par contraintes pour la localisation d'un robot sous-marin



Luc Jaulin, ENSIETA, Brest

Demie-journée industrielle, JFPC

Mercredi 4 juin 2008

1 Estimation à erreurs bornées

Modèle : $p_1 e^{-p_2 t}$.

Paramètres : p_1, p_2.

Temps de mesure : $t_1, t_2, \ldots, t_m$

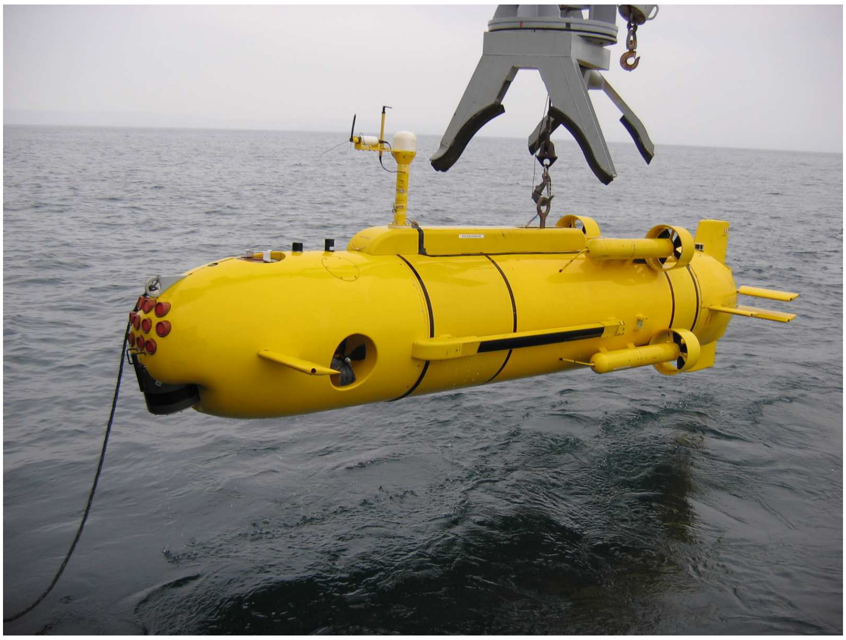
Barres de mesure : $[y_1^-, y_1^+], [y_2^-, y_2^+], \ldots, [y_m^-, y_m^+]$

Système de contraintes :

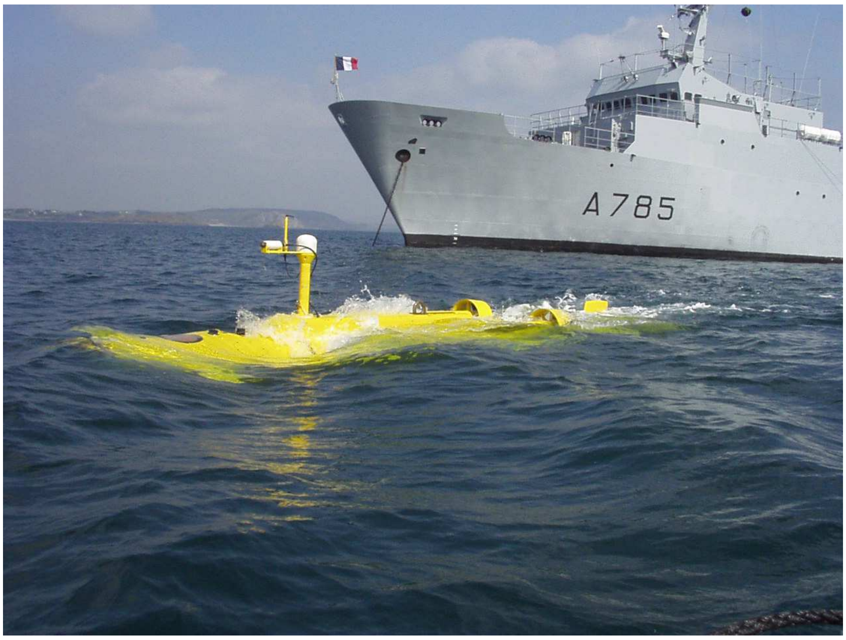
$$\begin{cases} y_1^- \leq p_1 e^{-p_2 t_1} \leq y_1^+ \\ \\ y_m^- \leq p_1 e^{-p_2 t_m} \leq y_m^+ \end{cases}$$

Logiciel SetDemo (Guillaume Baffet)

2 Localisation de mines



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



Le *Redermor* à la surface

Montrer la simulation

Pourquoi une approche par intervalles ?

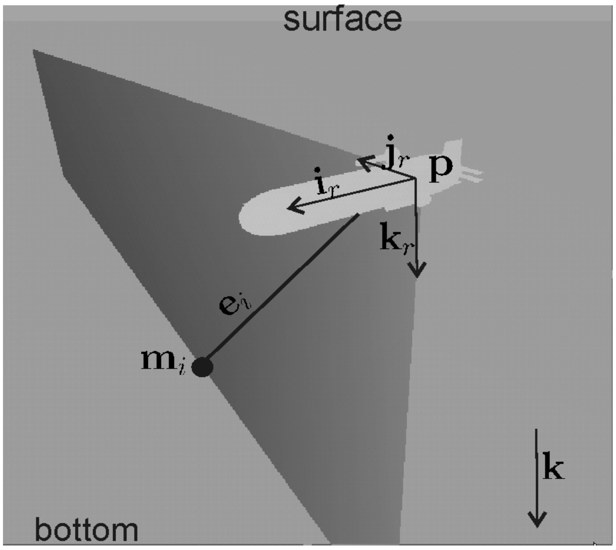
1) Besoin d'une approche fiable.
2) Les équations du robot sont non linéaires.
3) Les bruits de mesure sont non gaussiens.
4) Des bornes sur les erreurs sont fournies par les constructeurs des capteurs.
5) Beaucoup de mesures redondantes sont disponibles.

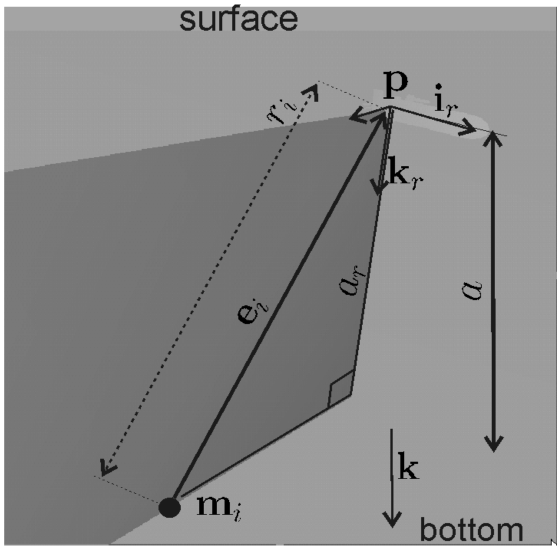
2.1 Capteurs

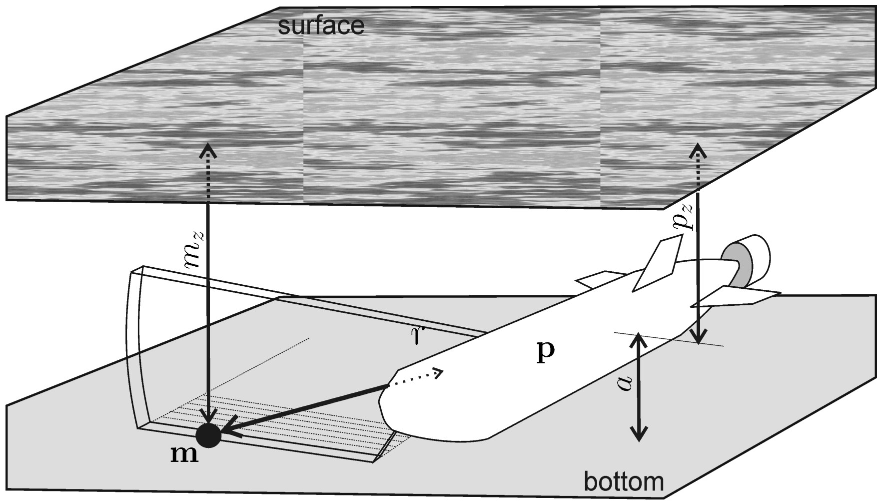
Un GPS (Global positioning system), disponible à la surface.

$$\begin{array}{ll} t_0 = 6000 \text{ s}, & \ell^0 = (-4.4582279^o, 48.2129206^o) \pm 2.5m \\ t_f = 12000 \text{ s}, & \ell^f = (-4.4546607^o, 48.2191297^o) \pm 2.5m \end{array}$$

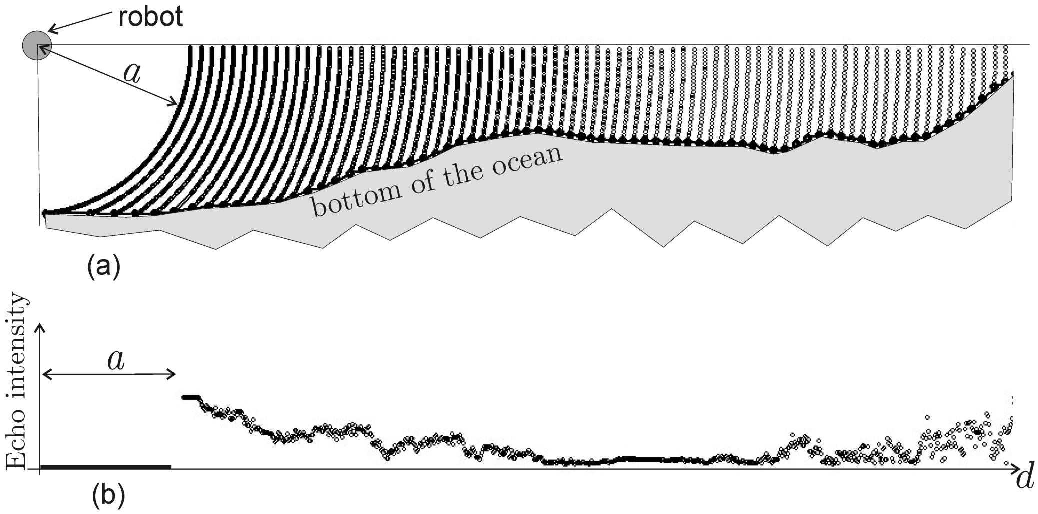
Un sonar (KLEIN 5400 side scan sonar). Donne la distance r entre le robot et la mine

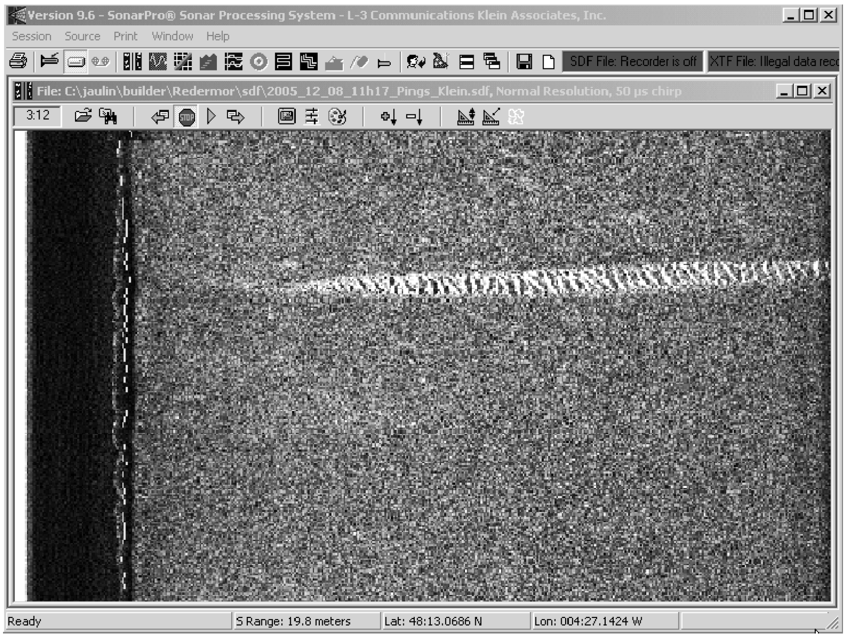




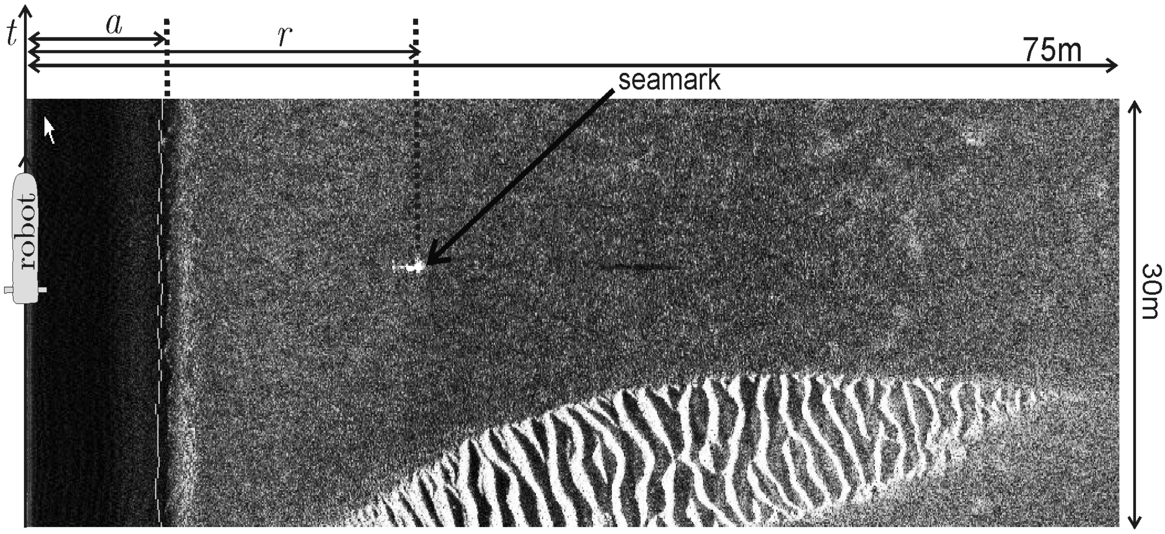


surface
m_z
p_z
r
$\mathbf{p}$
a
$\mathbf{m}$
bottom


robot
a
bottom of the ocean
(a)
Echo intensity
a
d
(b)



Screenshot du logiciel SonarPro

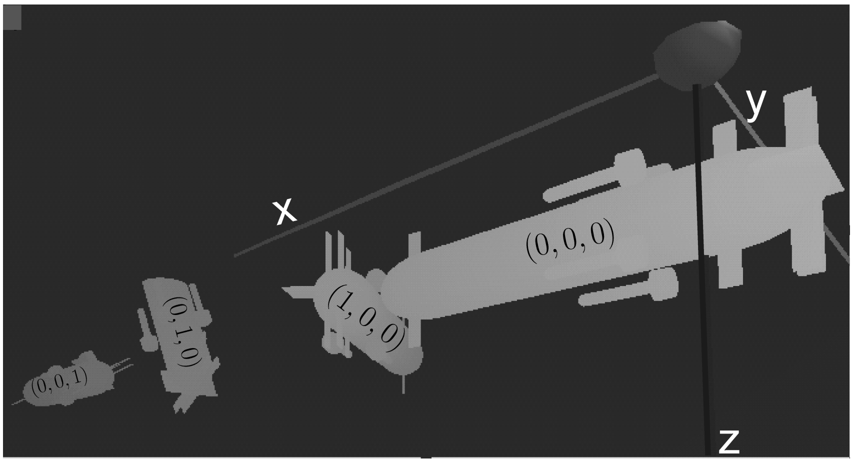


Détection d'une mine à l'aide de SonarPro

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

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
$\tau(i)$	7054	7092	7374	7748	9038	9688
$\sigma(i)$	1	2	1	0	1	5
$\tilde{r}(i)$	52.42	12.47	54.40	52.68	27.73	26.98

6	7	8	9	10	11
10024	10817	11172	11232	11279	11688
4	3	3	4	5	1
37.90	36.71	37.37	31.03	33.51	15.05

2.3 Contraintes

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

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

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

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

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

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

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

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

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

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

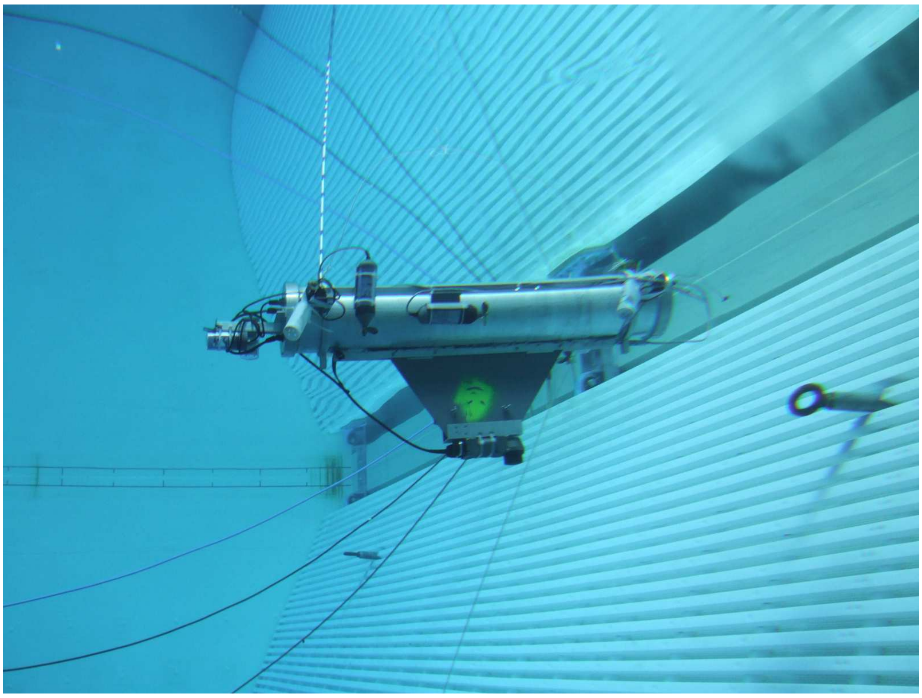
$$\mathbf{R}^{\mathsf{T}}(\tau(i))\,(\mathbf{m}(\sigma(i))-\mathbf{p}(\tau(i)))\in[0]\times[0,\infty]^{\times 2},$$

$$m_{z}(\sigma(i))-p_{z}(\tau(i))-a(\tau(i))\in[-0.5,0.5]$$

Un langage adapté à la programmation par contraintes sur les domaines continus avec des variables vectorielles et matricielles est le langage QUIMPER (Gilles Chabert).

3 Un robot avec des intervalles embarqués

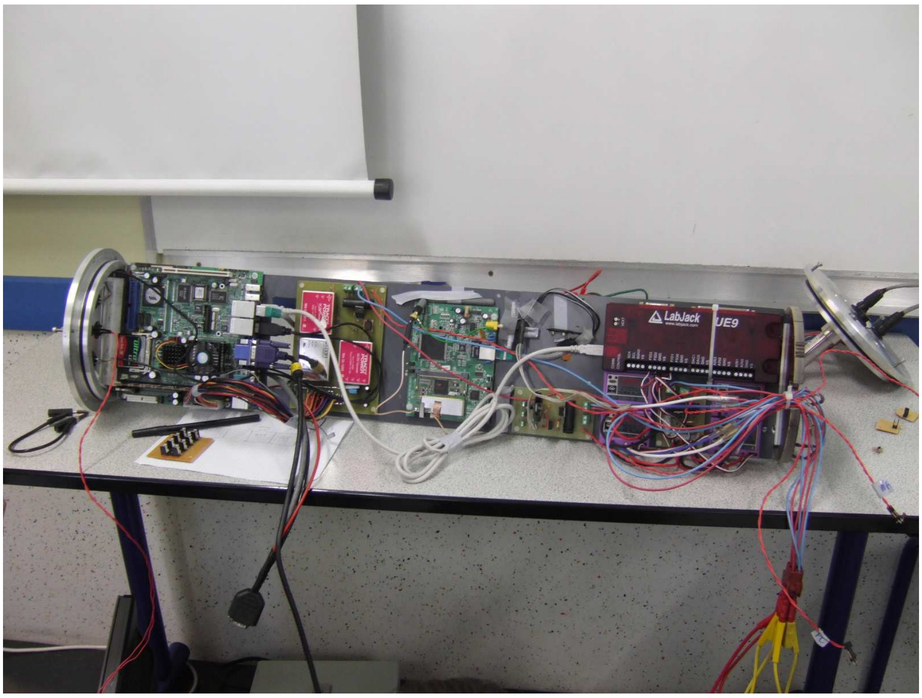
3.1 Présentation du robot SAUC'ISSE



Sauc'isse = SAUCe Interval Submerged Submarine of Ensieta

SAUCE = Student Autonomous Underwater Competition European


SAUC'ISSE
PILOT



LabJack
UE9

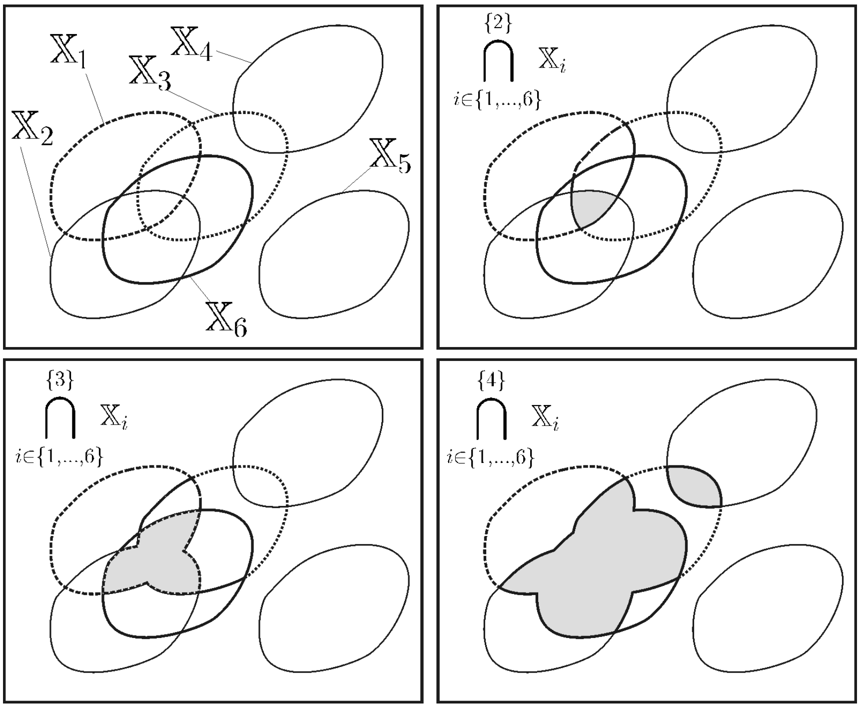
3.2 Système de contraintes

Hypothèse : Sur les ℓ dernières mesures faites par le sonar, il ne peut y avoir plus de q données aberrantes.

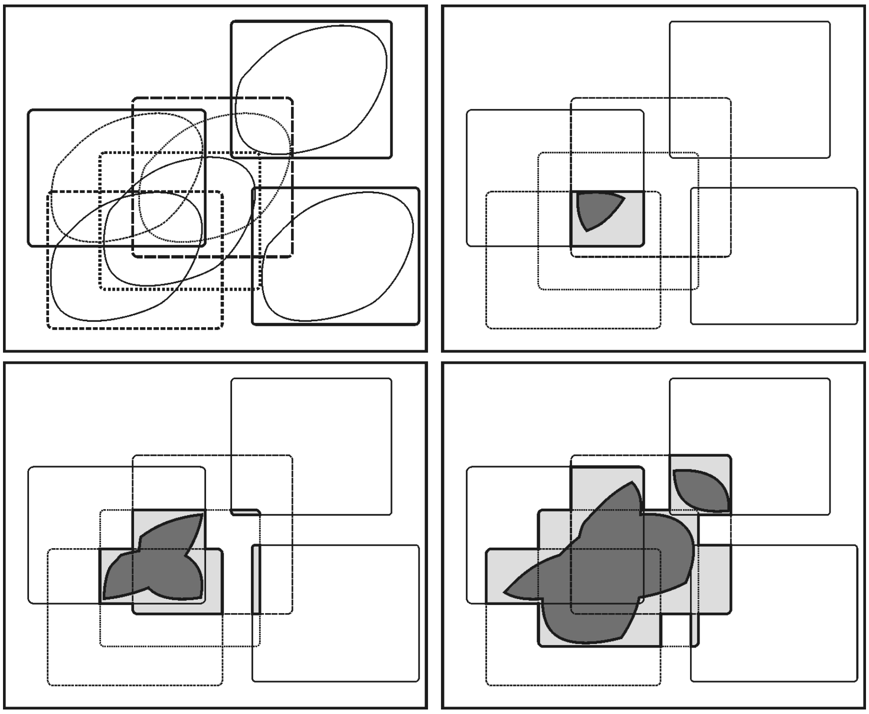
Le système de contraintes s'écrit :

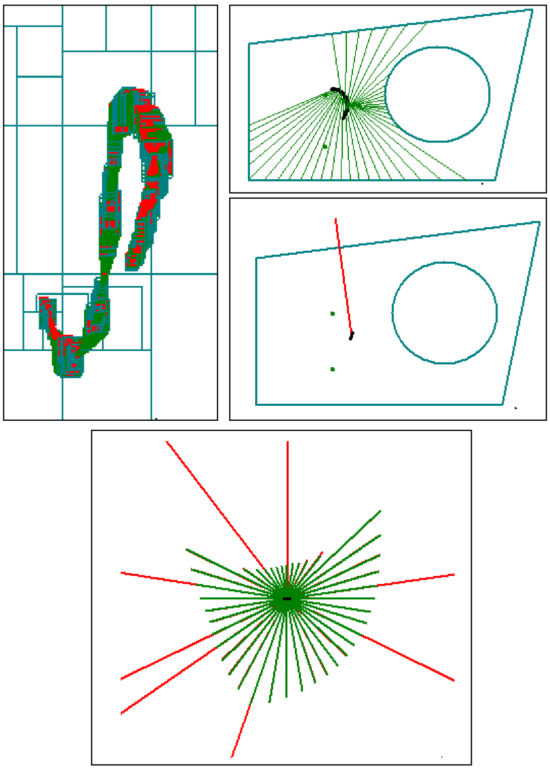
$$\begin{cases} \dot{x} = v \cos\theta \\ \dot{y} = v \sin\theta \\ \dot{\theta} = u_2 - u_1, \\ \dot{v} = u_1 + u_2 \\ d(t_k) = \text{dist}\left(x\left(t_k\right), y\left(t_k\right), \theta\left(t_k\right), \alpha_k\left(t_k\right)\right). \\ u_1 \in [u_1](t), u_2 \in [u_2](t) \\ \bigwedge_q \left((d(t_k) \in [d](t_k)) \ldots, (d(t_{k-\ell}) \in [d](t_{k-\ell}))\right). \end{cases}$$

où $t_{k,}$ est l'instant de la $k$ième mesure.



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





4 SWIM08

SWIM 08

Small Workshop on Interval Methods

Montpellier, France

June 2008, Thursday 19 and Friday 20

http://www.ensieta.fr/e3i2/Jaulin/swim08.html