

# Localisation et suivi robustes d'un robot mobile grâce à l'analyse par intervalles

## Robust robot localization and tracking using interval analysis

par Michel KIEFFER<sup>1</sup>, Luc JAULIN<sup>1,2</sup>, Éric WALTER<sup>1</sup> et Dominique MEIZEL<sup>3</sup>

<sup>1</sup>Laboratoire des Signaux et Systèmes, UMR CNRS n° 8506 – Supélec – Université Paris-Sud, Plateau de Moulon, 91192 Gif-sur-Yvette, France  
{jaulin, kieffer, walter}@lss.supelec.fr

<sup>2</sup>Laboratoire d'Ingénierie des Systèmes Automatisés, Université d'Angers

<sup>3</sup>UTC/HEUDIASYC, UMR CNRS n° 6599, B.P. 20529, 60205 Compiègne, France  
dominique.meizel@hds.univ-compiegne.fr

### *résumé et mots clés*

Cet article présente une solution originale au problème de localisation et de suivi d'un robot mobile à partir de données télé-métriques fournies par des capteurs à ultrasons. Le problème est traité dans un contexte d'erreurs de mesure et d'incertitudes sur l'état bornées. Grâce à une technique d'estimation de paramètres et d'état utilisant l'analyse par intervalles, les mesures provenant des capteurs à ultrasons sont fusionnées en tenant compte de la présence inévitable de nombreuses données aberrantes. Un *ensemble* contenant la configuration (position et orientation) du robot est fourni à chaque instant. Un exemple simulé réaliste est présenté en détail pour illustrer les propriétés de la technique.

calcul par intervalles, capteurs à ultrasons, données aberrantes, estimation de paramètres, estimation d'état, fusion de données, robotique, robustesse.

### *abstract and key words*

This paper presents a new solution to the problem of localizing and tracking a robot from distance measurements provided by on-board ultrasonic sensors. The measurement errors and state perturbations are assumed to be bounded. A parameter and state estimation technique based on interval analysis is employed to compute, as a function of time, a *set* guaranteed to contain all robot configurations (position and orientation) that are consistent with the measurements provided by the ultrasonic sensors, given bounds on the acceptable errors. The unavoidable presence of many outliers is taken into account to provide robust estimates. A realistic simulated example is treated in detail to illustrate the properties of the technique.

data fusion, interval computation, outliers, parameter estimation, robotics, robustness, state estimation, ultrasonic sensors.

## 1. introduction

La localisation d'un véhicule dans un référentiel pertinent pour la tâche à accomplir est un problème central de la robotique mobile, et en particulier de la robotique d'intérieur qui forme le contexte de cet article. La difficulté est liée autant à la variété des capteurs qui concourent à cette tâche qu'aux incontournables incertitudes et imprécisions des cartes utilisées pour définir la tâche à accomplir. Il s'agit là d'un problème typique de fusion de données [Crowley], [Castellanos].

Le suivi d'un robot en mouvement présente deux facettes, l'initialisation considérée comme la partie la plus complexe, et la poursuite qui l'est moins. Une fois l'initialisation réalisée, et à condition que celle-ci permette une localisation relativement précise, la poursuite peut s'effectuer en alternant des séquences de prédiction et d'estimation [Durieu-a] réalisées souvent par filtrage de Kalman (étendu) comme par exemple dans [Leonard-a] ou bien par des algorithmes elliptiques à erreur bornée comme dans [Hanebeck] et [Meizel]. L'initialisation, lorsqu'aucune connaissance *a priori* n'est disponible, est plutôt du ressort de la reconnaissance de scènes [Drumheller]. Cette reconnaissance a été envisagée par gestion d'arbres d'interprétation des données ([Grimson], [Halbwachs]) et plus récemment par estimation au sens du maximum global de vraisemblance [Olson].

La méthode présentée dans ce papier est développée dans un cadre théorique rigoureux et permet d'apporter une solution globale à ce problème, tant pour l'initialisation que pour la poursuite, même dans le cas où l'initialisation est ambiguë ou très imprécise. Tout au long de l'étude, nous ferons l'hypothèse que les erreurs de mesure et les incertitudes sur l'état sont bornées, tout en admettant que certaines données expérimentales puissent ne pas vérifier cette hypothèse.

Dans un premier temps, quelques résultats concernant l'estimation paramétrique grâce à l'analyse par intervalles seront rappelés au paragraphe 2. Une attention particulière sera accordée au traitement des mesures aberrantes, c'est-à-dire ne satisfaisant pas les hypothèses sur les erreurs. En effet, celles-ci sont pratiquement inévitables lors de l'utilisation de mesures télémétriques fournies par des capteurs à ultrasons. Un exemple de localisation statique sera détaillé au paragraphe 3. Il s'agira de déterminer la position et l'orientation du robot dans un environnement de surface au sol plane. Le paragraphe 4 présentera un algorithme ensembliste d'estimation d'état pour des systèmes non linéaires [Kieffer-b]. Comme pour le filtre de Kalman, une phase de prédiction à partir des informations passées alterne avec une phase de correction pour la prise en compte des nouvelles mesures. Cette étape de correction met en œuvre une technique d'estimation robuste décrite dans [Kieffer-c] et [Kieffer-d]. L'application de cette technique au suivi du robot lors de son déplacement sera ensuite détaillée.

## 2. estimation paramétrique à erreurs bornées

L'objectif de l'estimation paramétrique est de déterminer le vecteur des paramètres  $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^n$  d'un modèle du dispositif étudié à partir de mesures expérimentales, le but étant d'obtenir un modèle dont le comportement soit optimal ou acceptable en un sens à définir.

### 2.1. notations

Soit  $y(k) \in \mathbb{R}$  le résultat de la  $k$ -ième mesure. Dans le cas du robot,  $k$  représente le numéro d'un capteur. En supposant que la sortie  $y(k)$  peut être décrite par le modèle  $y_m(k, \mathbf{x})$ , nous cherchons à estimer le vecteur des paramètres  $\mathbf{x} \in \mathcal{X}$  de ce modèle. L'erreur entre les sorties mesurées et prédites est

$$e(k, \mathbf{x}) = y(k) - y_m(k, \mathbf{x}). \quad (1)$$

Cette erreur correspond d'une part au caractère approximatif de la modélisation des phénomènes physiques intervenant dans le processus à décrire (on parle d'*erreur structurelle*) et d'autre part aux *erreurs (ou bruits) de mesure*, dues aux précisions limitées des appareils de mesure, aux diverses perturbations, etc.

Parmi les nombreuses méthodes d'identification disponibles (voir [Walter-b], par exemple), beaucoup procèdent par optimisation d'un critère. Tout argument  $\hat{\mathbf{x}}$  de l'optimum de ce critère correspond à une valeur *optimale* des paramètres  $\mathbf{x}$ , *au sens du critère retenu*, car la valeur de  $\hat{\mathbf{x}}$  peut varier fortement suivant la fonction coût choisie. Outre le fait que  $\hat{\mathbf{x}}$  n'est pas forcément unique, son obtention est en général difficile. De plus, l'évaluation de l'incertitude avec laquelle  $\hat{\mathbf{x}}$  est estimé reste un problème largement ouvert.

Ici, nous allons rechercher l'ensemble des valeurs de  $\mathbf{x}$  *compatibles* avec les données au sens où toutes les erreurs sont comprises entre des bornes prédéfinies. C'est le contexte d'*erreurs bornées* (voir [Milanese], [Norton-a], [Norton-b], [Walter-a] et [Walter-b]). L'estimée de  $\mathbf{x}$  ne sera alors plus ponctuelle, mais constituée d'un ensemble de valeurs.

### 2.2. estimation à erreurs bornées

Par hypothèse,  $e(k, \mathbf{x})$  appartient à l'intervalle *connu*  $[e(k)] = [\underline{e}(k), \bar{e}(k)]$ . Une valeur  $\mathbf{x}$  du vecteur des paramètres sera compatible avec la  $k$ -ième mesure si  $y_m(k, \mathbf{x}) \in [y(k)] = [y(k) - \bar{e}(k), y(k) + \underline{e}(k)]$ . Cette propriété correspond à un

test  $t_k$  sur  $\mathbf{x}$ , qui permet de définir l'ensemble  $\mathcal{S}_k$  des valeurs du vecteur des paramètres compatibles avec la  $k$ -ième mesure comme

$$\mathcal{S}_k = \{\mathbf{x} \in \mathcal{X} | t_k(\mathbf{x}) = \text{vrai}\}, \quad (2)$$

avec

$$t_k(\mathbf{x}) = (y_m(k, \mathbf{x}) \in [y(k)]). \quad (3)$$

L'intersection des  $\mathcal{S}_k$  pour  $k = 1, \dots, N$  définit l'ensemble  $\mathcal{S}$  des valeurs des paramètres compatibles avec toutes les mesures

$$\mathcal{S} = \bigcap_{k=1}^N \mathcal{S}_k = \{\mathbf{x} \in \mathcal{X} | t(\mathbf{x}) = \text{vrai}\}, \quad (4)$$

avec

$$t(\mathbf{x}) = t_1(\mathbf{x}) \wedge \dots \wedge t_N(\mathbf{x}), \quad (5)$$

où  $\wedge$  représente l'opérateur logique ET (voir la figure 1, pour une situation où seules deux mesures sont disponibles).

$\mathcal{S}$  n'est en général pas un singleton. Il peut être vide, ce qui signifie qu'il n'est pas possible de trouver de valeurs de  $\mathbf{x}$  satisfaisant (3) pour tout  $k$ , ou encore être un sous-ensemble de  $\mathbb{R}^n$ , éventuellement non connexe.

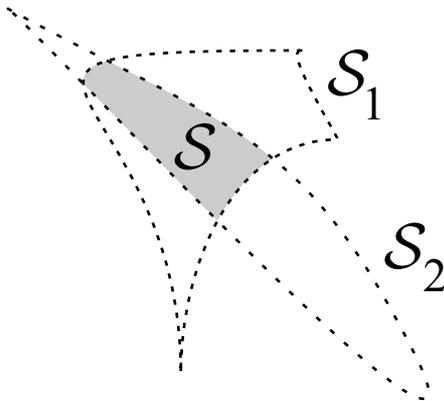


Figure 1. – Ensemble des vecteurs de paramètres compatibles avec deux mesures.

### 2.3. traitement des données aberrantes

Un cas particulier où  $\mathcal{S}$  peut être vide correspond à la situation où les hypothèses sur les erreurs de mesure ne sont pas satisfaites pour toutes les valeurs de  $k$  ; certaines mesures sont alors dites *aberrantes*. Celles-ci sont dues par exemple à un dysfonctionnement temporaire du dispositif de mesure. Une parade à ce problème consiste à rechercher l'ensemble des paramètres qui sont compatibles avec le plus grand nombre possible de mesures.

Le test  $t(\mathbf{x})$  dans (4) doit alors être modifié. Définissons l'opérateur Booléen ET- $q$ -relaxé, noté  $\bigoplus_q$ , tel que

$$\text{Soit } (b_1, b_2, \dots, b_{n_s}) \in \{\text{vrai, faux}\}^{n_s},$$

$$\bigoplus_q(b_1, b_2, \dots, b_{n_s}) = \text{vrai} \iff \text{au moins } n_s - q \text{ des } b_i \text{ valent vrai.}$$

Il est alors possible de tolérer jusqu'à  $q$  mesures aberrantes en caractérisant l'ensemble

$$\mathcal{S}_{r,q} = \{\mathbf{x} \in \mathcal{X} | t_{\text{robuste}}(\mathbf{x}, q) \text{ est vrai}\}, \quad (6)$$

où

$$t_{\text{robuste}}(\mathbf{x}, q) = \bigoplus_q(t_1(\mathbf{x}), \dots, t_{n_s}(\mathbf{x})). \quad (7)$$

La figure 2 illustre une situation où une seule donnée aberrante est tolérée ; l'ensemble solution  $\mathcal{S}_{r,1}$  est constitué ici de deux sous-ensembles disjoints.

Pour déterminer la valeur de  $q$ , une technique possible est de supposer au départ que  $q = 0$ , ce qui revient à utiliser (4), et d'incrémenter  $q$  d'une unité tant que l'ensemble des configurations possibles reste vide. Cette technique correspond à l'estimateur OMNE (*Outlier Minimal Number Estimator*, voir [Lahanier], [Pronzato], et [Walter-c]), mis ici en œuvre de façon garantie (voir [Jaulin-c], [Kieffer-c] et [Kieffer-d]).

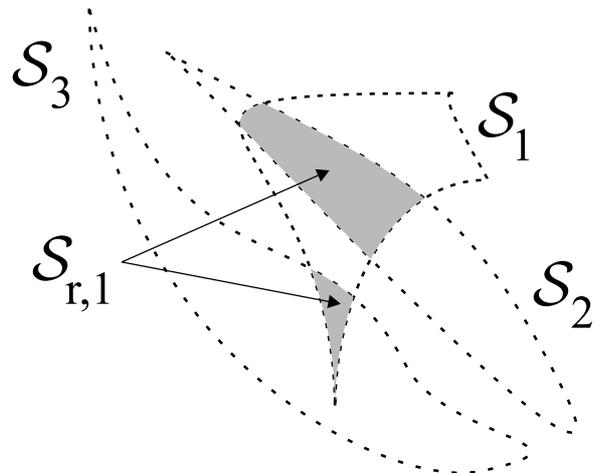


Figure 2. – Estimation robuste avec trois données ; une seule donnée aberrante est tolérée.

### 2.4. inversion ensembliste et analyse par intervalles

La description de  $\mathcal{S}$  (ou  $\mathcal{S}_{r,q}$ ) peut être formulée comme un problème d'*inversion ensembliste*, résolu de manière approchée mais garantie grâce à l'analyse par intervalles et l'algorithme SIVIA (*Set Inversion Via Interval Analysis*, voir [Jaulin-a] et [Jaulin-b]). Cette technique rappelée brièvement ici caractérise

l'ensemble  $\mathcal{S}$  par un sous-pavage, c'est-à-dire une union de vecteurs d'intervalles (ou pavés) disjoints. Grâce à l'analyse par intervalles, il est possible d'évaluer un test  $t(\cdot)$ , tel que (5), non plus sur des vecteurs de réels, mais sur des pavés (voir l'annexe A). Pour un pavé  $[\mathbf{x}]$  donné, trois situations seront possibles :

- $t([\mathbf{x}])$  est faux ; il n'existe donc pas de vecteur dans  $[\mathbf{x}]$  qui soit compatible avec les mesures. Le pavé  $[\mathbf{x}]$  a une intersection vide avec  $\mathcal{S}$ .
- $t([\mathbf{x}])$  est vrai ; tous les vecteurs contenus dans  $[\mathbf{x}]$  sont donc acceptables. Le pavé  $[\mathbf{x}]$  est inclus dans  $\mathcal{S}$ .
- $t([\mathbf{x}])$  est indéterminé ; tout ou partie de  $[\mathbf{x}]$  peut éventuellement appartenir à  $\mathcal{S}$ . On peut alors découper le pavé indéterminé  $[\mathbf{x}]$  et appliquer le test  $t(\cdot)$  sur les sous-pavés obtenus.

SIVIA effectue tout d'abord un test  $t(\cdot)$  sur l'espace de recherche initial  $[\mathbf{x}_0]$ . Si le pavé est indéterminé et est découpé, SIVIA est appliqué récursivement sur les sous-pavés résultants. Afin d'obtenir une solution en temps fini, le découpage n'est pas effectué lorsque la taille d'un pavé indéterminé est inférieure à un facteur de précision  $\varepsilon$ . L'ensemble solution  $\mathcal{S}$  est enfermé dans une sous-pavage  $\hat{\mathcal{S}}$  constitué de tous les pavés compatibles avec les mesures et des pavés indéterminés jugés suffisamment petits pour ne plus être découpés. Cette approximation *extérieure* peut être rendue aussi précise que souhaitée en réduisant  $\varepsilon$ , et elle est *garantie* car toutes les valeurs du vecteur des paramètres  $\mathbf{x}$  compatibles avec les mesures sont incluses dans  $\hat{\mathcal{S}}$ .

**Exemple 2.1.** Soit à caractériser  $\mathcal{S}$  défini par les tests suivants

$$t_1(\mathbf{x}) = \{x_1^2 + x_2^2 - 2|x_1| \in [0.7, 1]\},$$

$$t_2(\mathbf{x}) = \{x_1^2 + x_2^2 \in [0.7, 1]\}.$$

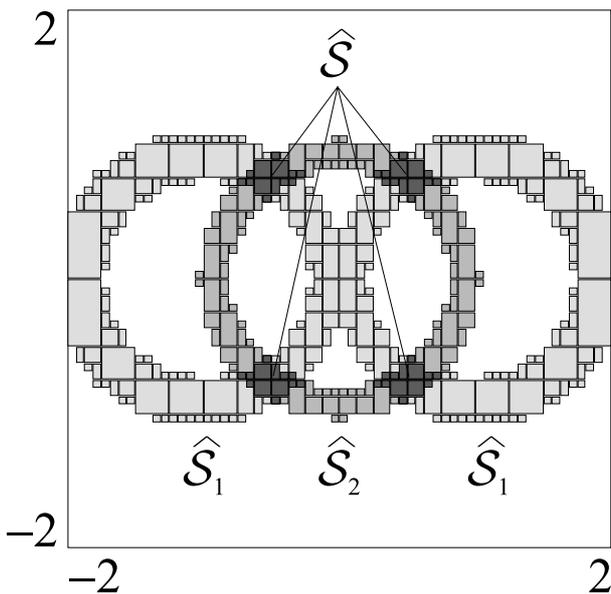


Figure 3. – Utilisation de SIVIA pour l'exemple 2.1. ;  $\hat{\mathcal{S}}$  est en gris foncé.

Si on applique SIVIA à ce problème avec un paramètre de précision  $\varepsilon = 0.1$ , on obtient le sous-pavage représenté en gris foncé sur la figure 3. Les sous-pavages correspondant à  $\mathcal{S}_1$  et  $\mathcal{S}_2$  ont été représentés en gris clair.

Quelques éléments sur la manière de déterminer la valeur d'un test sur un pavé sont donnés dans l'annexe A.

### 3. localisation statique

L'objectif est de déterminer la position  $(x_c, y_c)$  et l'orientation  $\theta$  du repère  $\mathcal{R}$  attaché au robot dans le repère  $\mathcal{W}$  associé à son environnement. Le vecteur des paramètres  $\mathbf{x} = (x_c, y_c, \theta)^T$  à estimer est appelé la *configuration* du robot (par rapport à son environnement) (voir la figure 4).

Les informations disponibles pour la localisation (carte de l'environnement et mesures de distance) seront décrites au paragraphe 3.1. Elles permettront de construire un test, décrit au paragraphe 3.2., applicable à une configuration donnée pour décider si elle est compatible avec les mesures. Ensuite, grâce aux techniques intervalles présentées au paragraphe 2, il sera possible d'évaluer l'ensemble des configurations compatibles avec les informations disponibles, comme illustré au paragraphe 3.3.

#### 3.1. informations disponibles

La carte  $\mathcal{M}$  de l'environnement du robot, constituée de  $n_0$  segments orientés, décrit les obstacles (murs, portes, piliers, etc.) entourant le robot

$$\mathcal{M} = \{[\mathbf{a}_j, \mathbf{b}_j] | j = 1, \dots, n_0\}. \tag{8}$$

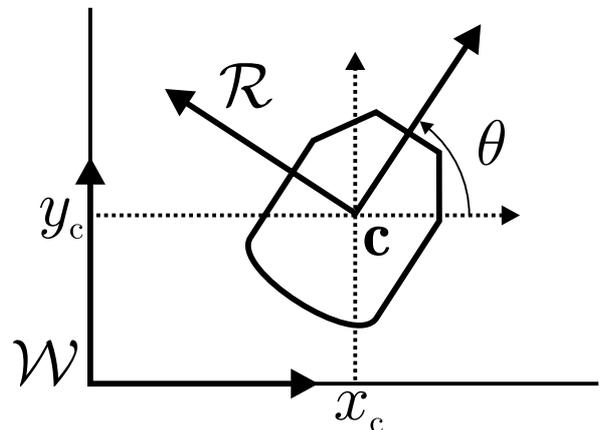


Figure 4. – Configuration du robot.

Par convention, lors d'un déplacement de  $a_j$  vers  $b_j$ , la partie réfléchissante du segment est sur la gauche (la matière de l'obstacle est à droite). Une telle carte est représentée sur la figure 5 ; les obstacles sont en gris et les segments en traits pleins.

La perception de l'environnement s'effectue en utilisant 24 télémètres à ultrasons répartis autour du véhicule. Le télémètre d'indice  $i$  émet une onde dans une direction donnée. Cette onde, dont on suppose qu'elle se propage à l'intérieur d'un cône d'émission de demi-ouverture  $\gamma_i$ , sera réfléchiée par les éléments de l'environnement du robot. Lorsque l'obstacle présente une surface irrégulière, la réflexion est à la fois spéculaire et diffuse (voir [Hecht]). L'angle d'incidence sur un obstacle détermine si l'onde réfléchiée est directement reçue par le capteur. Pour un cône d'émission de demi-ouverture nulle, on peut définir un angle d'incidence limite  $\beta_j$  au-delà duquel l'onde réfléchiée n'est plus perçue directement. Lorsque  $\gamma_i$  est non nul, l'angle d'incidence moyen du faisceau doit être inférieur à  $\gamma_i + \beta_j/2$  (voir l'annexe B). La valeur de  $\beta_j \in [0, \pi/2]$  dépend de l'état de surface de l'obstacle représenté par le  $j^{\text{ème}}$  segment de la carte. Cette valeur sera importante pour des surfaces rugueuses, et très faible pour des surfaces lisses. Deux situations sont illustrées sur la figure 6.

En cas de réception, le temps écoulé depuis l'émission est converti en une distance  $d_i$  à l'obstacle le plus proche dans la direction d'émission. Les incertitudes de mesure sont supposées bornées, et l'intervalle  $[d_i] = [(1 - \alpha_i)d_i, (1 + \alpha_i)d_i]$  devrait de ce fait contenir la vraie distance à l'obstacle le plus proche dans le cône d'émission (voir la figure 7-a). Cependant, l'onde

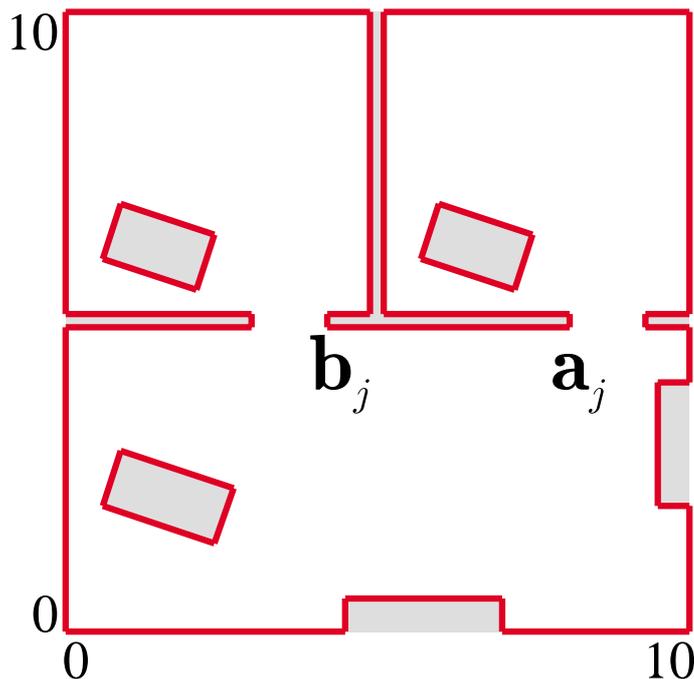


Figure 5. – Carte de l'environnement du robot (les distances sont en m.).

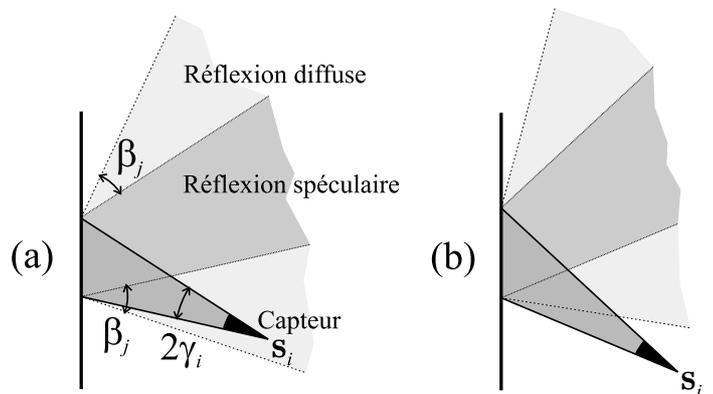


Figure 6. – (a) : L'existence d'une réflexion diffuse permet une réception par le capteur ; (b) : l'onde n'est pas reçue.

pourra être reçue après plusieurs réflexions, voire ne pas être reçue du tout. Dans ces conditions, l'interprétation de la mesure de distance n'est plus aussi facile, et les bornes sur les incertitudes perdent leur sens. Les mesures seront alors considérées comme aberrantes, par rapport à la compréhension qu'autorise notre modèle. La présence, malheureusement à peu près inévitable, de telles mesures aberrantes rend le problème de localisation particulièrement ardu.

L'ensemble des mesures peut être résumé par un diagramme d'émission (voir la figure 7-b). Chaque segment correspond à une distance  $d_i$  mesurée par un capteur. Un obstacle doit se trouver au moins en partie entre les deux arcs de cercle matérialisant l'incertitude relative de mesure  $\alpha_i$  et la demi-ouverture du cône d'émission  $\gamma_i$ .

### 3.2. test sur les configurations

On suppose disposer d'un pavé de recherche  $[x_0]$  dans lequel se trouve la configuration réelle du robot. Ce pavé peut être très grand si nécessaire. L'objectif est de déterminer le sous-ensemble  $\mathcal{S}$  de  $[x_0]$  correspondant à des configurations compatibles avec la carte et les mesures fournies par les capteurs, c'est-

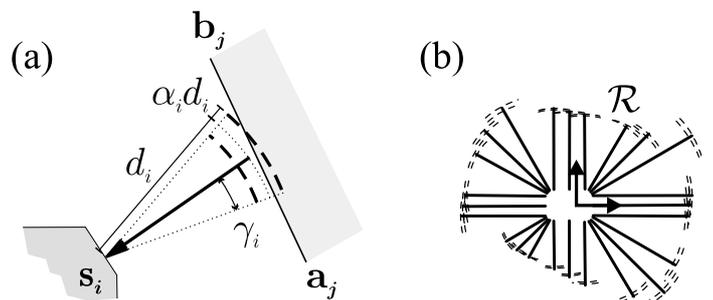


Figure 7. – Incertitude bornée pour une mesure de distance et diagramme d'émission.

à-dire  $\mathcal{S} = \{\mathbf{x} \in [\mathbf{x}]_0 | t(\mathbf{x}) = \text{vrai}\}$ , où  $t(\mathbf{x})$  est un test qui permet de statuer sur la compatibilité d'une configuration.

Pour construire  $t(\mathbf{x})$ , le processus télémétrique est simulé. Il s'agit de déterminer l'ensemble des mesures prédites par le modèle  $\{d_i^m(\mathbf{x}), i = 1, \dots, n_s\}$ , lorsque le robot se trouve dans une configuration  $\mathbf{x}$  donnée. Ces mesures simulées permettent de définir un test élémentaire  $t_i(\mathbf{x})$  pour la  $i^{\text{ème}}$  mesure

$$t_i(\mathbf{x}) = \text{vrai} \iff d_i^m(\mathbf{x}) \in [d_i]. \quad (9)$$

Le test  $t(\mathbf{x})$  peut alors être construit sur le modèle (5), en utilisant les tests élémentaires  $t_i(\mathbf{x})$  associés aux  $n_s$  capteurs. Une première version de  $t_i(\mathbf{x})$  ne prenant pas en compte l'angle d'incidence limite  $\beta$  est donnée dans [Kieffer-a] ou dans [Kieffer-d]. Dans l'annexe B, un test  $t_i(\mathbf{x})$  ne présentant plus cette limitation est construit pas à pas. Pour chaque capteur, nous montrons que la complexité de ce test est linéaire en le nombre de segments de la carte. Le test  $t(\mathbf{x})$  étant évalué à partir du résultat des  $n_s$  tests  $t_i(\mathbf{x})$ , la complexité du test  $t(\mathbf{x})$  est bilinéaire en le nombre de segments de la carte et en le nombre de capteurs.

Si des données aberrantes sont présentes, la technique présentée au paragraphe 2.3. pourra directement être mise en œuvre ; on cherchera alors à caractériser  $\mathcal{S}_{r,q}$  défini par (6) et (7).

### 3.3. exemple

Le robot a été placé dans un environnement dont la carte est représentée sur la figure 5. La configuration réelle (inconnue) du robot est  $(x_c, y_c, \theta) = (2.5 \text{ m}, 8 \text{ m}, 0.2 \text{ rad})$ . Les mesures de distance correspondent au diagramme d'émission de la figure 7-b. Toutes ces mesures ont été simulées de la façon suivante. Les  $n_s = 24$  capteurs sont supposés identiques et l'état de surface est le même pour tous les obstacles, avec  $\alpha = 0.02$ ,  $\gamma = 0.1978 \text{ rad}$  et  $\beta = 0.6 \text{ rad}$ . Afin de simuler un des phénomènes qui peuvent donner naissance à des données aberrantes, lorsque l'angle d'incidence de l'onde émise par un capteur est supérieur à  $\gamma + \beta/2$ , la mesure est remplacée par une distance tirée aléatoirement selon une distribution uniforme entre 0.5 m et 10 m. En l'absence de toute connaissance *a priori* sur la configuration du robot, le pavé de recherche  $[\mathbf{x}]$  est pris égal à  $[0 \text{ m}, 10 \text{ m}] \times [0 \text{ m}, 10 \text{ m}] \times [0 \text{ rad}, 2\pi \text{ rad}]$ . Il pourrait être réduit par exemple à  $[0 \text{ m}, 5 \text{ m}] \times [5 \text{ m}, 10 \text{ m}] \times [0 \text{ rad}, 2\pi \text{ rad}]$  si l'on savait que le robot se trouve dans la pièce représentée dans la partie supérieure gauche de la carte.

L'ensemble  $\widehat{\mathcal{S}}_{r,q}$  reste vide tant que  $q$  est inférieur à trois, ce qui traduit la présence d'au moins trois données aberrantes (voir le paragraphe 2.3.). L'ensemble solution  $\widehat{\mathcal{S}}_{r,3}$  est constitué de deux sous-ensembles disjoints, tels que

$$\begin{aligned} \widehat{\mathcal{S}}_{r,3} \subset & [2.43 \text{ m}, 2.57 \text{ m}] \times [7.93 \text{ m}, 8.07 \text{ m}] \times [0.125 \text{ rad}, 0.250 \text{ rad}] \\ & \cup [7.50 \text{ m}, 7.69 \text{ m}] \times [7.93 \text{ m}, 8.07 \text{ m}] \times [0.125 \text{ rad}, 0.313 \text{ rad}]. \end{aligned}$$

À condition qu'il n'y ait effectivement pas plus de trois données aberrantes,  $\widehat{\mathcal{S}}_{r,3}$  contient la configuration réelle du robot, comme le montre le raisonnement suivant. Soit  $\mathbf{x}^*$  la configuration réelle du robot. S'il n'y a pas plus de  $q$  données aberrantes,  $t_{\text{robuste}}(\mathbf{x}^*, q)$  est vrai. Par conséquent  $\mathbf{x}^* \in \mathcal{S}_{r,q}$ . Or SIVIA garantit que  $\mathcal{S}_{r,q} \subset \widehat{\mathcal{S}}_{r,q}$ . Par conséquent  $\mathbf{x}^* \in \widehat{\mathcal{S}}_{r,q}$ .

L'ambiguïté de la localisation est une conséquence des symétries locales de la pièce, qui ne doit bien sûr pas être attribuée à un défaut de la méthode. En effet, celle-ci ne fait que révéler un problème d'identifiabilité alors que nombre de techniques de localisation se borneraient à proposer une solution ponctuelle unique.

Lorsque  $q$  augmente, le volume de l'approximation extérieure de l'ensemble solution (calculé en additionnant les volumes des pavés constituant  $\widehat{\mathcal{S}}_{r,q}$ ) peut également augmenter. En effet, tolérer plus de mesures aberrantes revient à ignorer un nombre plus important de données qui auraient pu être informatives. Le tableau 1 résume les résultats de l'estimation pour  $q$  allant de 0 à 6 ; les calculs ont été effectués par un Pentium-II à 450 MHz. Pour  $q = 6$ , le volume de l'ensemble solution augmente sensiblement à cause de l'apparition de nouveaux sous-ensembles disjoints des deux premiers. D'autre part, lorsque  $q$  croît, l'aug-

Tableau 1. – Résultats de la procédure de localisation statique.

$q$	volume de $\widehat{\mathcal{S}}_{r,q}$	temps de calcul	temps cumulé
0	0	59 s	59 s
1	0	87 s	146 s
2	0	115 s	261 s
3	0.008988	144 s	405 s
4	0.009587	195 s	600 s
5	0.009587	243 s	843 s
6	0.014980	322 s	1165 s

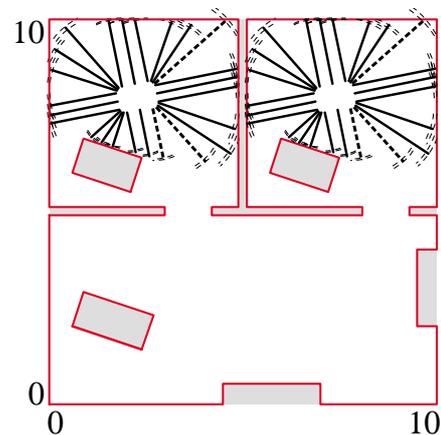


Figure 8. – Deux des configurations possibles du robot après localisation statique ; les mesures jugées aberrantes sont matérialisées par des segments en pointillés.

mentation du temps de calcul de chacune des étapes est globalement linéaire ; on n'observe donc pas la croissance exponentielle qui serait constatée s'il avait fallu tester chacune des combinaisons de  $q$  données aberrantes parmi l'ensemble des mesures.

**Remarque 3.1.** – Dans [Lévêque], les résultats fournis par une version antérieure de cet algorithme ensembliste de localisation sont comparés à ceux obtenus par une technique utilisant le filtrage de Kalman (par exemple aux pages 111 et 181). Les temps de calcul sont bien sûr nettement en faveur de ce dernier, mais le filtre de Kalman suppose l'association entre segments de la carte et mesures déjà effectuée et ne procède qu'à un traitement local, incapable de révéler d'éventuelles ambiguïtés de localisation.

## 4. estimation d'état non linéaire

Lorsqu'un système évolue dans le temps, il peut être intéressant de déterminer l'évolution de son état (ici, la configuration du robot). Pour être efficace cette évaluation doit exploiter, à un instant  $l$  donné, toutes les connaissances acquises jusqu'à  $l$  et de nouvelles mesures lorsqu'elles deviennent disponibles. Nous supposons disposer du modèle non linéaire à temps discret

$$\begin{cases} \mathbf{x}_{l+1} = \mathbf{f}_l(\mathbf{x}_l + \mathbf{u}_l + \mathbf{v}_l) \\ \mathbf{y}_l = \mathbf{h}_l(\mathbf{x}_l) + \mathbf{w}_l \end{cases} \quad l = 0, 1, \dots, \quad (10)$$

où  $\mathbf{x}_l \in \mathbb{R}^n$  est le vecteur d'état,  $\mathbf{u}_l \in \mathbb{R}^m$  est le vecteur d'entrée et  $\mathbf{y}_l \in \mathbb{R}^p$  est le vecteur de mesures. Le vecteur  $\mathbf{w}_l \in \mathbb{R}^p$  représente le bruit sur les mesures, et la perturbation d'état  $\mathbf{v}_l$  permet de tenir compte de dynamiques non modélisées. La valeur initiale de l'état  $\mathbf{x}_0$  appartient par hypothèse à un compact  $\mathcal{X}_0 \subset \mathbb{R}^n$  donné (par exemple un sous-pavage résultant de l'application de l'algorithme de localisation statique).  $\{\mathbf{v}_l\}$  et  $\{\mathbf{w}_l\}$  sont des suites supposées appartenir respectivement aux suites de pavés  $\{[\mathbf{v}_l]\}$  et  $\{[\mathbf{w}_l]\}$  connues. Enfin,  $\mathbf{f}_l$  et  $\mathbf{h}_l$  sont des fonctions connues (éventuellement décrites par des algorithmes finis).

### 4.1. algorithme d'estimation d'état

Dans un premier temps, nous présenterons une version idéalisée d'un algorithme ensembliste d'estimation d'état, puis nous donnerons quelques éléments sur la méthode employée pour réaliser une version *approchée* mais *garantie* de cet algorithme.

L'objectif est de caractériser un ensemble  $\mathcal{X}_l$  contenant toutes les valeurs possibles de l'état  $\mathbf{x}_l$  à un instant  $l$  donné. Le raisonnement est par récurrence. Supposons qu'à l'instant  $l$ ,  $\mathcal{X}_l$  soit

disponible. À l'instant  $l + 1$ , on définit l'ensemble  $\mathcal{X}_{l+1}$  contenant toutes les valeurs de l'état qui sont accessibles à partir d'une valeur quelconque de  $\mathbf{x}_l$  prise dans  $\mathcal{X}_l$ . Cet ensemble est évalué (au moins en théorie) grâce à la première équation de (10). L'ensemble ① de la figure 9 est ainsi obtenu. Les ensembles  $\mathcal{X}_l$  et  $\mathcal{X}_{l+1}$  peuvent être des compacts quelconques.

Supposons qu'à l'instant  $l + 1$ , un nouveau vecteur de mesure  $\mathbf{y}_{l+1}$  devienne disponible. En considérant une erreur de mesure bornée, nous savons que la sortie ne peut appartenir qu'à l'ensemble  $\mathcal{Y}_{l+1}$  représenté sur la figure 9 – ensemble ②. Dans la seconde équation de (10), la fonction  $\mathbf{h}_{l+1}$  décrit comment le vecteur de mesures est obtenu à partir de l'état (dans le cas du robot, c'est la fonction décrite dans l'annexe B qui sera utilisée). De ce fait, l'état ne peut appartenir qu'à  $\mathcal{X}_{l+1}^0 = \mathbf{h}_{l+1}^{-1}(\mathcal{Y}_{l+1})$  [9 – ensemble ③]. Par conséquent, l'ensemble  $\mathcal{X}_{l+1}$  de toutes les valeurs de l'état  $\mathbf{x}_{l+1}$  compatibles avec les informations disponibles à l'instant  $l + 1$  est l'intersection de  $\mathcal{X}_{l+1}$  et de  $\mathcal{X}_{l+1}^0$ , schématisée sur la figure 9 par l'ensemble ④. La récurrence est initialisée à  $\mathcal{X}_0$ , qui rassemble la connaissance sur les valeurs possibles de l'état à l'instant initial. La figure 9 représente ainsi les deux étapes de base d'un algorithme ensembliste et récursif d'estimation d'état. L'étape de *prédiction* détermine l'évolution de l'ensemble contenant l'état et l'étape de *correction* tient compte des nouvelles informations disponibles en éliminant certaines parties de l'ensemble obtenu par prédiction.

En général, il n'est pas possible d'évaluer ces ensembles de manière exacte. Par contre, il est possible de réaliser une version approchée mais garantie de l'algorithme d'estimation d'état grâce à l'analyse par intervalles. Celle-ci permet en effet, grâce à la notion de *fonction d'inclusion*, d'obtenir un pavé contenant l'ensemble des valeurs prises par une fonction sur un pavé donné. Ainsi, les ensembles  $\mathcal{X}_l$  et  $\mathcal{X}_{l+1}$  sont enfermés dans des sous-pavages. La technique qui permet de réaliser l'étape de prédiction de manière approchée est la suivante. Les pavés com-

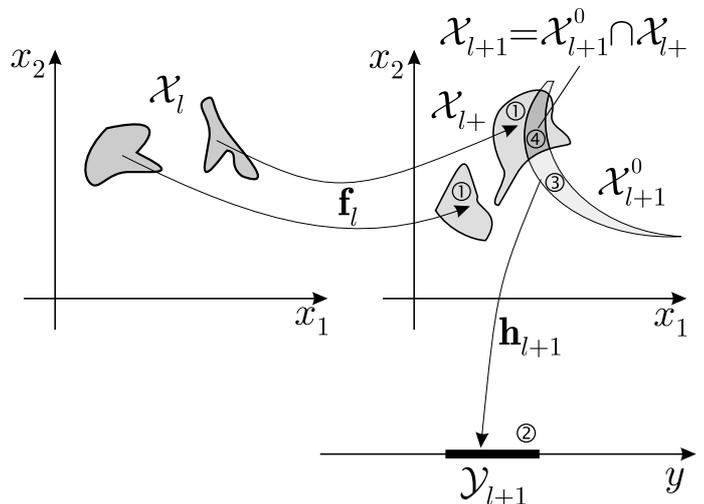


Figure 9. – Algorithme idéalisé d'estimation d'état ensembliste.

posant le sous-pavage  $\widehat{\mathcal{X}}_l$  sont découpés, et l'image de chaque pavé résultant est calculée. Ces images sont ensuite rassemblées pour former un sous-pavage  $\widehat{\mathcal{X}}_{l+}$  approximant  $\mathcal{X}_{l+}$ . L'étape de correction permettant de déterminer  $\widehat{\mathcal{X}}_{l+1}$  se fait ensuite grâce à SIVIA. L'ensemble  $\mathcal{X}_{l+1}^0$  n'est en fait pas calculé, seule son intersection avec  $\widehat{\mathcal{X}}_{l+}$  l'est, ce qui permet d'accélérer de manière spectaculaire la procédure de recherche.

Cet estimateur approché est garanti, au sens où toutes les valeurs de l'état  $x_l$  compatibles avec les hypothèses sont contenues dans  $\widehat{\mathcal{X}}_l$ . Pour plus de détails sur cette technique, voir [Kieffer-a] et [Kieffer-b].

### 4.2. prise en compte de données aberrantes

Les mesures aberrantes éventuelles n'interviennent que dans l'étape de correction. Dans le cas du robot, cette étape correspond en fait à une localisation dans un espace de recherche réduit à l'ensemble  $\widehat{\mathcal{X}}_{l+}$ . Le traitement des données aberrantes se fait donc comme pour la localisation statique. Une des politiques envisageables, qui est celle qui sera suivie ici, consiste dans un premier temps à rechercher l'ensemble  $\widehat{\mathcal{X}}_l$  des états compatibles avec toutes les mesures à l'instant  $l$  ( $q_l = 0$ ), puis à tolérer de plus en plus de données aberrantes en incrémentant  $q_l$  d'une unité tant que  $\widehat{\mathcal{X}}_l$  reste vide.

En pratique, il pourrait s'avérer plus prudent de donner à  $q_l$  une valeur supérieure à la plus petite valeur qui assure que  $\widehat{\mathcal{X}}_l$  soit non-vide. En effet, la procédure n'est garantie que si à chaque étape le nombre réel de données aberrantes est inférieur ou égal au nombre de données aberrantes tolérées  $q_l$ . Cette modification de la politique de choix de  $q_l$  est extrêmement simple à mettre en œuvre, mais cette augmentation de robustesse se fait au détriment de la précision de la localisation, puisqu'un nombre plus important de mesures doit être négligé.

### 4.3. application à la localisation dynamique

Cet algorithme d'estimation d'état est appliqué au suivi du robot. L'équation d'état est obtenue par discrétisation exacte des équations cinématiques. Pour simplifier l'exposé, aucun bruit d'état n'a été considéré, mais la prise en compte d'un tel bruit ne pose aucun problème de principe pourvu que des bornes soient disponibles. L'étape de correction correspond à la technique de localisation statique présentée au paragraphe 3. La pièce est toujours décrite par la carte représentée sur la figure 5. L'ensemble  $\mathcal{X}_0$  correspond au résultat de la localisation statique effectuée au paragraphe 3.3.

Le robot se déplace de la pièce représentée sur la partie supérieure gauche de la figure 5 vers la pièce de la partie inférieure. Toutes

les secondes, une nouvelle série de mesures est effectuée afin de corriger l'ensemble obtenu à l'étape de prédiction précédente. L'évolution de la projection des ensembles  $\mathcal{X}_l$  sur le plan  $(x, y)$  est représentée sur la partie inférieure de la figure 10. Sur la partie supérieure de cette figure, le robot est représenté dans une ou deux configurations appartenant à l'ensemble solution trouvé. Jusqu'à  $t = 6$  s, l'ensemble solution est composé de deux sous-ensembles disjoints. Après cet instant, le sous-ensemble droit a pu être éliminé. En effet, les configurations du sous-ensemble gauche sont compatibles avec toutes les mesures sauf 8, alors que pour conserver les configurations appartenant au sous-ensemble droit il faudrait tolérer, à cet instant, au moins 11 données aberrantes.

La simulation d'un scénario de 19 s nécessite moins de 18 s sur un Pentium-II à 450 MHz. Ce temps est beaucoup plus faible que ceux constatés dans la phase de localisation statique, car à chaque pas l'ensemble de recherche est de taille bien plus petite. Ceci permet donc de poursuivre un robot avec une actualisation des mesures toutes les secondes, ce qui suppose que le robot se déplace relativement lentement. Le tableau 2 indique l'évolution du nombre de données aberrantes estimé au cours du processus de localisation ; ces dernières sont représentées en pointillés sur la figure 10. Transitoirement, le nombre minimum de données aberrantes  $q_l$  se révèle très important (plus de la moitié des mesures), ce qui correspond ici à des configurations où de

Tableau 2. – Evolution de  $q_l$ , le nombre minimum de données aberrantes estimé par l'algorithme, en fonction du temps.

Instant $l$	0	1	2	3	4	5	6	7	8	9
$q_l$	3	2	3	6	9	4	8	5	7	13
Instant $l$	10	11	12	13	14	15	16	17	18	19
$q_l$	4	7	5	6	4	4	4	3	6	7

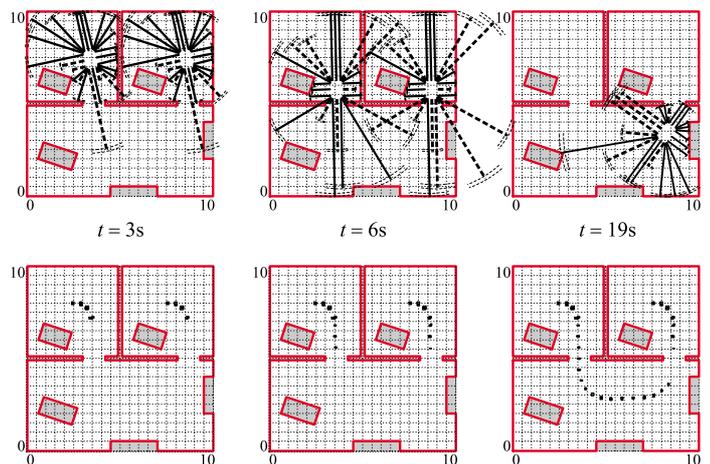


Figure 10. – Evolution de deux configurations possibles du robot et projection des ensembles solution sur le plan  $(x, y)$  ; les mesures jugées aberrantes sont matérialisées par des segments en pointillés ; à partir de  $t = 6$  s, l'une des deux configurations est jugée incompatible avec les données.

nombreuses incidences se font au-delà de l'angle limite. Le temps de calcul n'augmente pas significativement dans ce type de situation du fait de la faible taille des ensembles de recherche.

**Remarque 4.1.** – Pour une application réelle, l'interrogation des capteurs est souvent séquentielle (les capteurs sont par exemple interrogés par groupe de 4). Cette contrainte peut être prise en compte en n'utilisant que les dernières mesures acquises (au lieu de la totalité des mesures) pour actualiser la position prédite.

## 5. conclusions

Cet article a présenté une technique d'estimation (de paramètres ou d'état) qui se révèle très robuste vis-à-vis de données aberrantes comme celles qui sont inévitables en robotique mobile. Une solution originale est ainsi apportée à certaines limitations dont souffraient les techniques de localisation et de suivi précédemment employées. Le résultat est garanti, contrairement à ceux des techniques fondées sur le filtrage de Kalman (voir [Crowley], [Leonard-a] et [Leonard-b]) ou à ceux des techniques à erreur bornée nécessitant une linéarisation préalable, comme celles décrites dans [Durieu-b] et [Maksarov]. Aucune étape préalable d'association des mesures avec les segments de la carte n'est nécessaire, contrairement à ce qui est fait dans [Drumheller] et [Grimson]. Pour le suivi, la gestion d'hypothèses de localisation multiples est naturellement effectuée, contrairement à ce qui est fait dans [Halbwachs]. Enfin, la procédure de localisation étant incluse dans l'estimation d'état, il n'est pas nécessaire de faire appel à une procédure de localisation séparée contrairement à ce qui se passe dans [Leonard-a] et [Neira].

L'un des principaux défauts des techniques ensemblistes utilisant l'analyse par intervalles qui viennent d'être présentées est leur complexité exponentielle en la dimension de  $x$ . Dans le contexte de la localisation et du suivi du robot, ce point ne pose cependant pas de problème insurmontable. Une démonstration du suivi du robot peut être trouvée à l'adresse

<http://www.supelec.fr/lss/perso/kieffer/robotloc.htm>

Ces premiers résultats très encourageants sont à confirmer par une expérimentation plus systématique sur robot réel. Une comparaison des résultats avec ceux de méthodes traditionnelles comme le filtrage de Kalman a été abordée dans [Lévêque]. Une comparaison plus exhaustive reste à conduire. Il convient également d'étendre la procédure proposée à d'autres types de capteurs (lasers à balayage, GPS...) et à la robotique d'extérieur.

Les techniques intervalles dont nous avons parlé peuvent servir à de nombreuses autres tâches de robotique, y compris à l'établissement de cartes en environnement inconnu et à la planification

de trajectoire ; une suite naturelle de ce travail consisterait donc à combiner tous ces éléments dans un contexte d'exploration.

**Remerciements :** Les auteurs remercient INTAS pour son soutien financier partiel. (Contrat INTAS-RFBR-97-10782).

## BIBLIOGRAPHIE

- [Castellanos] J.A. Castellanos, J.M.M. Monteil, J. Neira, et J.D. Tardos. « The SMAP : A probabilistic framework for simultaneous localization and map building ». *IEEE Trans. on Robotics and Automation*, 15(5) : 948-952, 1999.
- [Crowley] J. Crowley. « World modeling and position estimation for a mobile robot using ultrasonic ranging ». Dans *Proc. IEEE Int. Conf. on Robotics and Automation*, pages 674-680, Scottsdale, Arizona, 1989.
- [Durieu-a] C. Durieu, M.J. Aldon, et D. Meizel. « La fusion de données multi-sensorielles pour la localisation en robotique mobile ». *Traitement du Signal*, 13(2) : 144-166, 1996.
- [Durieu-b] C. Durieu, B. Polyak, et E. Walter. « Trace versus determinant in ellipsoidal outer bounding with application to state estimation ». Dans *Proc. 13th IFAC World Congress*, volume I, pages 43-48, San Francisco, 1996.
- [Drumheller] M. Drumheller. « Mobile robot localization using sonar ». *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 9(2) : 325-332, 1987.
- [Grimson] W.E. Grimson et T. Lozano-Perez. « Localizing overlapping parts by searching the interpretation tree ». *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 9(4) : 469-482, 1987.
- [Hecht] E. Hecht. *Optics*. Addison-Wesley, Reading, second edition, 1987.
- [Hammer] R. Hammer, M. Hocks, U. Kulisch, et D. Ratz. *C++ Toolbox for Verified Computing*. Springer Verlag, Berlin, 1995.
- [Halbwachs] E. Halbwachs et D. Meizel. « Multiple hypothesis management for mobile vehicle localization ». Dans *CD-ROM of the European Control Conference*, Louvain, 1997.
- [Hanebeck] U.D. Hanebeck et G. Schmidt. « Set theoretic localization of fast mobile robots using an angle measurement technique ». Dans *Proc. IEEE Int. Conf. on Robotics and Automation*, pages 1387-1394, Minneapolis, 1996.
- [Jaulin-a] L. Jaulin et E. Walter. « Set inversion via interval analysis for nonlinear bounded-error estimation ». *Automatica*, 29(4) : 1053-1064, 1993.
- [Jaulin-b] L. Jaulin et E. Walter. « Guaranteed nonlinear parameter estimation from bounded-error data via interval analysis ». *Math. and Comput. in Simulation*, 35 : 1923-1937, 1993.
- [Jaulin-c] L. Jaulin, E. Walter, et O. Didrit. « Guaranteed robust nonlinear parameter bounding ». Dans *Proc. CESA'96 IMACS Multiconference (Symposium on Modelling, Analysis and Simulation)*, pages 1156-1161, Lille, 1996.
- [Kieffer-a] M. Kieffer. *Estimation ensembliste par analyse par intervalles, application à la localisation d'un véhicule*. Thèse de doctorat, Université Paris-Sud, Orsay, 1999.
- [Kieffer-b] M. Kieffer, L. Jaulin, et E. Walter. « Guaranteed recursive nonlinear state estimation using interval analysis ». Dans *Proc. 37th IEEE Conf. on Decision and Control*, pages 3966-3971, Tampa, 1998.
- [Kieffer-c] M. Kieffer, L. Jaulin, E. Walter, et D. Meizel. « Nonlinear identification based on unreliable priors and data, with application to robot localization ». Dans A. Garulli, A. Tesi, et A. Vicino, éditeurs, *Robustness in Identification and Control*, pages 190-203, LNCS 245, Londres, 1999. Springer.
- [Kieffer-d] M. Kieffer, L. Jaulin, E. Walter, et D. Meizel. « Robust autonomous robot localization using interval analysis ». *Reliable Computing* 6(3) : 337-362, 2000.
- [Leonard-a] J.J. Leonard et H.F. Durrant-Whyte. « Mobile robot localization by tracking geometric beacons ». *IEEE Trans. on Robotics and Automation*, 7(3) : 376-382, 1991.

[Leonard-b] J.J. Leonard et H.F. Durrant-Whyte. *Directed Sonar Sensing for Mobile Robot Navigation*. Kluwer Academic Publishers, Boston, 1992.

[Lévêque] O. Lévêque. *Méthodes ensemblistes pour la localisation de véhicules*. Thèse de doctorat, Université de Technologie, Compiègne, 1998.

[Lahanier] H. Lahanier, E. Walter, et R. Gomeni. « OMNE : a new robust membership-set estimator for the parameters of nonlinear models ». *J. of Pharmacokinetics and Biopharmaceutics*, 15 : 203-219, 1987.

[Maksarov] D. Maksarov et J.P. Norton. « State bounding with ellipsoidal set description of the uncertainty ». *Int. J. of Control*, 65(5) : 847-866, 1996.

[Milanese] M. Milanese, J. Norton, H. Piet-Lahanier, et E. Walter (éditeurs). *Bounding Approaches to System Identification*. Plenum Press, New York, 1996.

[Moore] R.E. Moore. « Practical aspects of interval computation ». *Appl. Math.*, (13) : 52-92, 1968.

[Meizel] D. Meizel, A. Preciado-Ruiz, et E. Halbwachs. « Estimation of mobile robot localization: geometric approaches ». Dans M. Milanese, J. Norton, H. Piet-Lahanier, et E. Walter, éditeurs, *Bounding Approaches to System Identification*, Plenum Press, New York, 1996, pages 463-489.

[Neira] J. Neira, J. Horn, J. D. Tardoz, et G. Schmidt. « Multisensor mobile robot localization ». Dans *Proc. IEEE Int. Conf. on Robotics and Automation*, pages 673-679, Mineapolis, 1996.

[Norton-a] J.P. Norton (Ed.). « Special issue on bounded-error estimation 1 ». *Int. J. of Adaptive Control and Signal Processing*, 8(1) : 1-118, 1994.

[Norton-b] J.P. Norton (Ed.). « Special issue on bounded-error estimation 2 ». *Int. J. of Adaptive Control and Signal Processing*, 9(1) : 1-132, 1995.

[Olson] C.F. Olson. « Probabilistic self-localization for mobile robots ». *IEEE Trans. on Robotics and Automation*, 16(1) : 55-66, 2000.

[Pronzato] L. Pronzato et E. Walter R. « Robustness to outliers of bounded-error estimators and consequences on experiment design ». Dans M. Milanese, J. Norton, H. Piet-Lahanier, et E. Walter, éditeurs, *Bounding Approaches to System Identification*, Plenum Press, New York, 1996, pages 199-212.

[Walter-a] E. Walter (Ed.). « Special issue on parameter identification with error bounds ». *Mathematics and Computers in Simulation*, 32(5&6) : 447-607, 1990.

[Walter-b] E. Walter et L. Pronzato. *Identification of Parametric Models from Experimental Data*. Springer-Verlag, Londres, 1997.

[Walter-c] E. Walter et H. Piet-Lahanier. « Estimation of the parameter uncertainty resulting from bounded-error data ». *Mathematical Biosciences*, 92 : 55-74, 1988.

## ANNEXE

### A. éléments de calcul sur les intervalles

Un intervalle  $[x] = [\underline{x}, \bar{x}]$  est un ensemble fermé borné de  $\mathbb{R}$  ; il est caractérisé par ses bornes inférieure  $\underline{x}$  et supérieure  $\bar{x}$ . L'ensemble des intervalles réels est noté  $\mathbb{I}\mathbb{R}$ . Il est possible de redéfinir les opérations de base sur les intervalles : soit  $\circ \in \{+, -, \cdot, / \}$  et  $([x], [y]) \in \mathbb{I}\mathbb{R}^2$ ,

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

Caractériser  $[z] = [x] \circ [y]$  est aisé ; par exemple

$$[x] + [y] = [\underline{x} + \underline{y}, \bar{x} + \bar{y}],$$

$$[x]/[y] = [\inf(\underline{x}/\underline{y}, \underline{x}/\bar{y}, \bar{x}/\underline{y}, \bar{x}/\bar{y}), \sup(\underline{x}/\underline{y}, \underline{x}/\bar{y}, \bar{x}/\underline{y}, \bar{x}/\bar{y})],$$

lorsque  $0 \notin [y]$ .

On peut également décrire par un intervalle l'ensemble des valeurs prises par une fonction élémentaire monotone sur un intervalle, par exemple

$$\exp([x]) = [\exp(\underline{x}), \exp(\bar{x})].$$

Pour des fonctions élémentaires non monotones (comme la fonction cosinus ou l'élevation à une puissance paire), l'obtention de l'ensemble des valeurs prises par cette fonction se fait par un algorithme simple (voir [Hammer], [Moore]). Dans le cas de fonctions quelconques, il reste possible dans de très nombreux cas d'évaluer un encadrement des valeurs prises par la fonction grâce aux *fonctions d'inclusion*. Une fonction d'inclusion  $[f]$  de  $\mathbb{I}\mathbb{R}$  vers  $\mathbb{I}\mathbb{R}$  associée à une fonction scalaire  $f$  définie sur  $D \subset \mathbb{R}$  est telle que

$$\forall [x] \subset D, f([x]) \subset [f]([x]).$$

Une technique simple pour obtenir une telle fonction d'inclusion consiste à remplacer toutes les occurrences de la variable réelle par la variable intervalle correspondante. On obtient ainsi une fonction d'inclusion dite *naturelle*. Si, par exemple, on veut évaluer une approximation extérieure de

$$f(x) = x^2 - 0.6x \exp(x^2)$$

sur  $[-1, 1]$ , la fonction d'inclusion naturelle vaut

$$\begin{aligned} [f]([-1, 1]) &= [-1, 1]^2 - 0.6[-1, 1]\exp([-1, 1]^2) \\ &= [0, 1] - 0.6[-1, 1]\exp([0, 1]) \\ &= [0, 1] - 0.6[-e, e] \\ &= [-0.6e, 1 + 0.6e] \subset [-1.64, 2.64]. \end{aligned}$$

Or  $f(x)$  est strictement décroissante sur  $[-1, 1]$ , par conséquent  $f([-1, 1]) = [1 - 0.6e, 1 + 0.6e] \subset [-0.64, 2.64]$ , on a donc bien  $f([-1, 1]) \subset [f]([-1, 1])$ .

Pour évaluer le résultat d'un test sur un intervalle, il est aisé de construire des *tests d'inclusion* à partir des fonctions d'inclusion. Par exemple, soit le test  $t(x) = (f(x) \in [y])$ , le test d'inclusion  $[t]$  correspondant utilisera une fonction d'inclusion  $[f]$  associé à  $f$  :

$$[t]([x]) = \text{faux si } [f]([x]) \cap [y] = \emptyset,$$

$$[t]([x]) = \text{vrai si } [f]([x]) \subset [y],$$

$$[t]([x]) = \text{indéterminé dans les autres cas.}$$

### B. Test de localisation

Cette annexe explique comment est construit le test de compatibilité d'une configuration  $\mathbf{x}$  avec les données fournies par un capteur  $\mathbf{s}$ . Dans le repère  $\mathcal{R}$  du robot, les coordonnées de  $\mathbf{s}$  sont notées  $\tilde{\mathbf{s}}$ , et la direction d'émission  $\tilde{\theta}$ . L'onde émise par ce capteur se propage dans un cône noté  $C(\tilde{\mathbf{s}}, \tilde{\theta}, \gamma)$ , où  $\gamma$  représente la demi-ouverture.

Pour bâtir le test associé à  $s$ , les informations disponibles dans  $\mathcal{R}$  doivent être traduites dans le repère de l'environnement  $\mathcal{W}$ . Pour une configuration  $x$  donnée,  $C$  est décrit de manière équivalente dans  $\mathcal{W}$  (en utilisant des formules de changement de repère) par les coordonnées de son sommet  $s(x)$  et deux vecteurs unitaires  $\vec{u}_1(x, \tilde{\theta}, \gamma)$  et  $\vec{u}_2(x, \tilde{\theta}, \gamma)$  correspondant à ses arêtes. Ainsi, on peut aussi noter le cône  $C(s, \vec{u}_1, \vec{u}_2)$  (en omettant la dépendance en  $x, \tilde{\theta}$  et  $\gamma$ ). Par convention,  $\vec{u}_1$  et  $\vec{u}_2$  sont choisis de telle manière que  $\vec{u}_2$  résulte de  $\vec{u}_1$  par une rotation d'angle  $2\gamma$  dans le sens trigonométrique. Comme  $\gamma$  est toujours inférieur à  $\pi/2$ , la condition d'appartenance d'un point quelconque  $m \in \mathbb{R}^2$  au cône d'émission se traduit par

$$m \in C(s, \vec{u}_1, \vec{u}_2) \Leftrightarrow (\det(\vec{u}_1, \vec{sm}) \geq 0) \wedge (\det(\vec{u}_2, \vec{sm}) \leq 0). \quad (11)$$

L'algorithme testant la cohérence d'une configuration avec les données est fondé sur la notion d'éloignement entre un capteur et la carte. Nous allons dans un premier temps définir ce concept pour un capteur et un segment  $[a, b]$  de la carte pris isolément. On suppose que ce segment représente une partie d'un obstacle dont l'angle limite de réflexion est  $\beta$ .

**Définition 1.** – L'éloignement  $r(s, \vec{u}_1, \vec{u}_2, a, b, \beta)$  entre le cône  $C(s, \vec{u}_1, \vec{u}_2)$  et le segment  $[a, b]$  est évalué comme suit

$$r(s, \vec{u}_1, \vec{u}_2, a, b) = \infty \text{ si } \det(\vec{ab}, \vec{as}) < 0, \quad (12)$$

$$= \infty \text{ si } \det\left(\frac{\vec{u}_1 + \vec{u}_2}{\|\vec{u}_1 + \vec{u}_2\|}, \frac{\vec{ab}}{\|\vec{ab}\|}\right) < \cos(\gamma + \beta/2), \quad (13)$$

$$= \infty \text{ si } [a, b] \cap C = \emptyset, \quad (14)$$

$$= \min_{m \in [a, b] \cap C} \|\vec{sm}\| \text{ dans les autres cas.} \quad (15)$$

La figure 11 illustre cette définition. Pour plus de lisibilité, les rayons réfléchis sont représentés du côté non réfléchissant de  $[a, b]$ , en appliquant une symétrie d'axe  $(a, b)$ . L'image du cap-

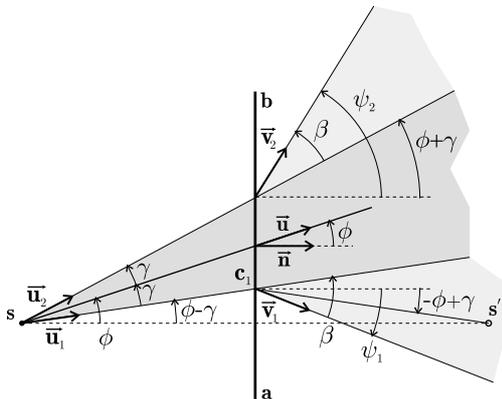


Figure 11. – L'angle d'incidence doit être inférieur à  $\gamma + \beta/2$  pour qu'une onde réfléchie soit perçue.

teur  $s$  est notée  $s'$ . En outre, le vecteur unitaire  $\vec{n}$ , défini par  $\det(\vec{n}, \vec{ab}) = \|\vec{ab}\|$  est introduit comme référence pour les mesures d'angles.

Une première condition nécessaire d'apparition d'une onde réfléchie est contenue dans (12). En effet, le capteur doit se trouver dans le demi-plan correspondant au côté réfléchissant de  $[a, b]$  pour qu'il y ait réflexion. Cela se traduit par

$$\det(\vec{ab}, \vec{as}) \geq 0. \quad (16)$$

Si (16) est satisfaite, on peut introduire l'angle d'incidence moyen

$$\phi = (\vec{n}, \vec{u}), \text{ avec } \vec{u} = \frac{\vec{u}_1 + \vec{u}_2}{\|\vec{u}_1 + \vec{u}_2\|}.$$

L'onde réfléchie, lorsqu'elle existe, se propage elle aussi dans un cône. Lorsque  $\phi \geq 0$ , soit  $c_1$  l'intersection des droites  $(s, \vec{u}_1)$  et  $(a, b)$  et lorsque  $\phi \leq 0$ , soit  $c_2$  l'intersection des droites  $(s, \vec{u}_2)$  et  $(a, b)$ . Considérons les vecteurs  $\vec{v}_1$  et  $\vec{v}_2$ , construits à partir de  $\vec{u}_1$  et  $\vec{u}_2$  par une rotation respective de  $-\beta$  et  $+\beta$ . Les angles

$$\psi_1 = (\vec{n}, \vec{v}_1) = -\beta + \phi - \gamma \quad (17)$$

et

$$\psi_2 = (\vec{n}, \vec{v}_2) = \beta + \phi + \gamma \quad (18)$$

encadrent alors l'ensemble des valeurs des angles formés par les rayons réfléchis avec  $\vec{n}$ . Deux cas doivent être considérés :

1. si  $\phi \geq 0$ , une onde peut être perçue si  $\psi_1 \leq (\vec{n}, \vec{c}_1 s')$ , c'est-à-dire si  $\psi_1 \leq -\phi + \gamma$  (ce cas correspond à la figure 11) ;
2. si  $\phi \leq 0$ , une onde peut être perçue si  $\psi_2 \geq (\vec{n}, \vec{c}_2 s')$ , c'est-à-dire si  $\psi_2 \geq -\phi - \gamma$ .

Dans les deux cas, en utilisant (17) et (18), on obtient

$$\gamma + \beta/2 \geq |\phi|. \quad (19)$$

Cette condition se traduit indépendamment de tout repère par

$$\langle \vec{n}, \vec{u} \rangle \geq \cos(\gamma + \beta/2), \quad (20)$$

ou encore par

$$\det\left(\frac{\vec{u}_1 + \vec{u}_2}{\|\vec{u}_1 + \vec{u}_2\|}, \frac{\vec{ab}}{\|\vec{ab}\|}\right) \geq \cos(\gamma + \beta/2), \quad (21)$$

d'où la condition contenue dans (13).

Les conditions d'incidence (16) et (21) sont seulement nécessaires à la présence d'une onde réfléchie effectivement perçue par le récepteur. La dernière condition d'incidence ( $[a, b] \cap C \neq \emptyset$ ) sera testée en calculant explicitement cette intersection lors de la recherche du minimum de  $\|\vec{sm}\|$  lorsque  $m$  décrit  $[a, b] \cap C$ . Cette minimisation impose l'examen de plusieurs situations. Soit

$\mathbf{h}$  la projection orthogonale de  $\mathbf{s}$  sur la droite  $(\mathbf{a}, \mathbf{b})$ . Si  $\mathbf{h} \in [\mathbf{a}, \mathbf{b}] \cap C$ , alors  $r(\mathbf{s}, \vec{\mathbf{u}}_1, \vec{\mathbf{u}}_2, \mathbf{a}, \mathbf{b}, \beta) = \|\vec{\mathbf{sh}}\|$ . Pour tester si  $\mathbf{h} \in [\mathbf{a}, \mathbf{b}] \cap C$ , il faut déterminer dans un premier temps si  $\mathbf{h} \in [\mathbf{a}, \mathbf{b}]$ . Comme  $\mathbf{h}$  est la projection orthogonale de  $\mathbf{s}$  sur  $[\mathbf{a}, \mathbf{b}]$ , on a

$$\mathbf{h} \in [\mathbf{a}, \mathbf{b}] \iff (\langle \vec{\mathbf{ab}}, \vec{\mathbf{as}} \rangle \geq 0) \wedge (\langle \vec{\mathbf{ba}}, \vec{\mathbf{bs}} \rangle \geq 0). \quad (22)$$

De plus, il faut que  $\mathbf{h} \in C$ . En utilisant (11), on obtient

$$\mathbf{h} \in C \iff (\det(\vec{\mathbf{u}}_1, \vec{\mathbf{sh}}) \geq 0) \wedge (\det(\vec{\mathbf{u}}_2, \vec{\mathbf{sh}}) \leq 0). \quad (23)$$

Cette condition n'est guère exploitable, car elle nécessite la connaissance des coordonnées de  $\mathbf{h}$ , relativement complexes à obtenir. Or, si la condition d'incidence (21) est respectée, l'angle entre  $\vec{\mathbf{sh}}$  et  $\vec{\mathbf{ab}}$  est de  $\frac{\pi}{2}$  et (23) devient, après quelques manipulations simples,

$$\mathbf{h} \in C \iff (\langle \vec{\mathbf{ab}}, \vec{\mathbf{u}}_1 \rangle \leq 0) \wedge (\langle \vec{\mathbf{ab}}, \vec{\mathbf{u}}_2 \rangle \geq 0). \quad (24)$$

En combinant (22) et (24), on obtient finalement une condition d'appartenance de  $\mathbf{h}$  à  $[\mathbf{a}, \mathbf{b}] \cap C$  indépendante des coordonnées de  $\mathbf{h}$

$$\mathbf{h} \in [\mathbf{a}, \mathbf{b}] \cap C \iff (\langle \vec{\mathbf{ab}}, \vec{\mathbf{as}} \rangle \geq 0) \wedge (\langle \vec{\mathbf{ba}}, \vec{\mathbf{bs}} \rangle \geq 0) \wedge (\langle \vec{\mathbf{ab}}, \vec{\mathbf{u}}_1 \rangle \leq 0) \wedge (\langle \vec{\mathbf{ab}}, \vec{\mathbf{u}}_2 \rangle \geq 0). \quad (25)$$

Si  $\mathbf{h} \notin [\mathbf{a}, \mathbf{b}] \cap C$ , la distance minimale peut être infinie (lorsque  $[\mathbf{a}, \mathbf{b}] \cap C = \emptyset$ ), ou être obtenue pour un point  $\mathbf{m}$  correspondant à l'une des extrémités du segment  $[\mathbf{a}, \mathbf{b}] \cap C$ . Soit  $\mathbf{h}_1$  et  $\mathbf{h}_2$  les intersections de la droite  $(\mathbf{a}, \mathbf{b})$  avec les droites  $(\mathbf{s}, \vec{\mathbf{u}}_1)$  et  $(\mathbf{s}, \vec{\mathbf{u}}_2)$ . L'ensemble des extrémités potentielles à distance finie de  $[\mathbf{a}, \mathbf{b}] \cap C$  est donc  $K = \{\mathbf{a}, \mathbf{b}, \mathbf{h}_1, \mathbf{h}_2\}$ . De ce fait, si  $\mathbf{h} \notin [\mathbf{a}, \mathbf{b}] \cap C$ ,  $r(\mathbf{s}, \vec{\mathbf{u}}_1, \vec{\mathbf{u}}_2, \mathbf{a}, \mathbf{b}, \beta)$  est soit infini, soit égal à  $\|\vec{\mathbf{sm}}\|$ , avec  $\mathbf{m}$  appartenant à  $K$ . Dans le cas de la figure 12,  $r(\mathbf{s}, \vec{\mathbf{u}}_1, \vec{\mathbf{u}}_2, \mathbf{a}, \mathbf{b}, \beta) = \|\vec{\mathbf{sb}}\|$ .

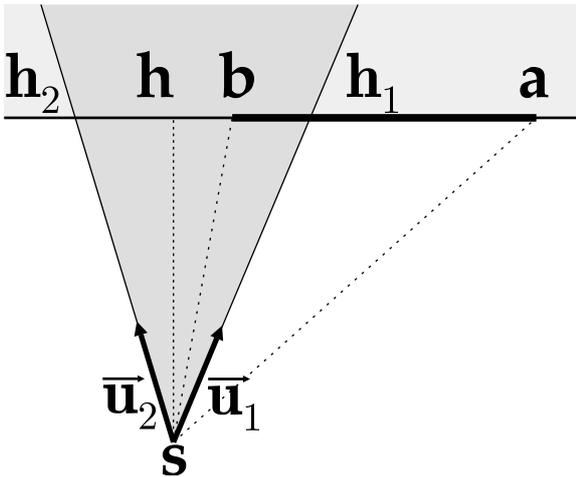


Figure 12. – Espaceur entre un capteur et un segment  $[\mathbf{a}, \mathbf{b}]$  isolé.

Il faut encore pouvoir tester si un élément de  $K$  appartient à  $[\mathbf{a}, \mathbf{b}] \cap C$ . Soit  $\mathbf{m} \in [\mathbf{a}, \mathbf{b}]$ , (11) permet de tester si  $\mathbf{m} \in C$ . Par construction,  $\mathbf{h}_1$  et  $\mathbf{h}_2$  appartiennent à  $C \cap (\mathbf{a}, \mathbf{b})$ , il suffit alors de tester s'ils appartiennent à  $[\mathbf{a}, \mathbf{b}]$ , ce qui revient à tester s'ils appartiennent à  $C(\mathbf{s}, \frac{\vec{\mathbf{sa}}}{\|\vec{\mathbf{sa}}\|}, \frac{\vec{\mathbf{sb}}}{\|\vec{\mathbf{sb}}\|})$ . Ainsi, pour  $i = 1, 2$ , (11) implique que

$$\mathbf{h}_i \in [\mathbf{a}, \mathbf{b}] \cap C \iff \left( \det \left( \frac{\vec{\mathbf{sa}}}{\|\vec{\mathbf{sa}}\|}, \vec{\mathbf{sh}}_i \right) \geq 0 \right) \wedge \left( \det \left( \frac{\vec{\mathbf{sb}}}{\|\vec{\mathbf{sb}}\|}, \vec{\mathbf{sh}}_i \right) \leq 0 \right). \quad (26)$$

Comme  $\frac{\vec{\mathbf{sh}}_i}{\|\vec{\mathbf{sh}}_i\|} = \vec{\mathbf{u}}_i$ , on peut écrire de façon équivalente

$$\mathbf{h}_i \in [\mathbf{a}, \mathbf{b}] \cap C \iff (\det(\vec{\mathbf{sa}}, \vec{\mathbf{u}}_i) \geq 0) \wedge (\det(\vec{\mathbf{sb}}, \vec{\mathbf{u}}_i) \leq 0). \quad (27)$$

En résumé, lorsque ni  $\mathbf{h}$  ni aucun élément de  $K$  n'appartient à  $[\mathbf{a}, \mathbf{b}] \cap C$ , il faut conclure que  $[\mathbf{a}, \mathbf{b}] \cap C = \emptyset$ , et que l'éloignement entre le cône et le segment est infini. C'est ce qu'exprime la définition 1.

Dans la situation normale où la carte comprend  $n_0$  segments, il faut prendre en compte le fait qu'un segment puisse ne pas réfléchir l'onde sonore parce qu'il est masqué par un autre segment, plus proche du capteur. Soit  $r_{ij}(\mathbf{x})$  l'éloignement entre le  $j^{\text{ème}}$  segment, considéré comme isolé, et le cône associé au  $i^{\text{ème}}$  capteur lorsque la configuration est  $\mathbf{x}$ ; cet éloignement est exprimé par

$$r_{ij}(\mathbf{x}) = r(\mathbf{s}_i(\mathbf{x}), \vec{\mathbf{u}}_{1i}(\mathbf{x}, \tilde{\theta}_i, \tilde{\gamma}_i), \vec{\mathbf{u}}_{2i}(\mathbf{x}, \tilde{\theta}_i, \tilde{\gamma}_i), \mathbf{a}_j, \mathbf{b}_j, \beta_j). \quad (28)$$

L'éloignement  $r_i$  entre le capteur  $i$  et la carte est donné par

$$r_i(\mathbf{x}) = \min_{j=1, \dots, n_0} r_{ij}(\mathbf{x}). \quad (29)$$

La complexité de cette procédure de calcul dépend donc linéairement du nombre de segments de la carte.

Manuscrit reçu le 25 février 2000

## LES AUTEURS

Michel KIEFFER



Michel Kieffer est né à Sarreguemines, en 1972. Ancien élève de l'École Normale Supérieure de Cachan, il est titulaire de l'agrégation de physique, option physique appliquée. Il a obtenu son titre de docteur en automatique à l'université Paris-Sud, Orsay en 1999. Actuellement, il est Maître de conférences à l'université Paris-Sud. Ses principaux centres d'intérêt sont l'estimation robuste de paramètres et d'état à l'aide d'outils issus de l'analyse par intervalles, avec pour application la robotique et le traitement du signal

Luc JAULIN



Luc Jaulin est né à Nevers en 1967. Il a obtenu son titre de docteur en automatique à l'université Paris-Sud, Orsay en 1993. Depuis 1993, il est Maître de conférences à l'université d'Angers. Ses principaux centres d'intérêt sont l'estimation et la commande robuste à l'aide d'outils tels que l'analyse par intervalles et la propagation de contraintes.

Éric WALTER



Éric Walter est né à Saint Mandé en 1950. Il a obtenu son doctorat d'état en automatique en 1980. Actuellement, il est Directeur de recherches au CNRS. Ses activités de recherche gravitent autour de problèmes d'estimation paramétrique avec des applications en commande robuste, en électrochimie, en génie chimique, en médecine, en pharmacocinétique, en traitement d'images et en robotique. Avec Luc Pronzato, il a écrit *Identification of Parametric Models from Experimental Data* (Springer, Londres, 1997).

Dominique MEIZEL



Dominique Meizel est né en 1952. Après son diplôme d'ingénieur IDN (devenu École Centrale de Lille) en 1976, il a soutenu sa thèse de Docteur-Ingénieur en 1979 puis de Doctorat d'État en 1984 à propos d'analyse de la stabilité de systèmes non-linéaires et de synthèse paramétrique de contrôleurs stabilisants. Il est Professeur à l'UTC (Université de Technologie de Compiègne) depuis 1988 et travaille depuis lors sur des questions relatives à la commande et à l'observation de robots mobiles et d'automobiles. C'est dans ce cadre d'application qu'il s'est intéressé aux méthodes ensemblistes. Depuis 1999, il est co-directeur du GdR CNRS n°1865 concernant la Coopération Humain-Machine pour l'aide à la conduite automobile.