

Suivi de cibles à l'aide d'un réseau de capteurs

Adrien ICKOWICZ, Jean Pierre LE CADRE

IRISA / CNRS
Rennes, France

ickowicz@irisa.fr, lecadre@irisa.fr

Abstract –

1 Introduction

La problématique principale des réseaux de capteurs consiste à utiliser la topologie de ces réseaux afin d'extraire et de traiter l'information la plus fiable possible. Ces capteurs pouvant être limités en terme de portée, précision, autonomie... L'intérêt d'utiliser une information binaire est qu'elle ne demande pas ou peu de traitement local, et coûte moins en énergie lors de la transmission. De même, la modélisation d'une erreur d'information est très facile. Il existe plusieurs types d'informations de ce genre, et nous avons décidé de nous concentrer sur l'information de mouvement : la cible s'approche d'elle ou s'éloigne d'elle du capteur [1] ? Nous présentons dans cet article une méthode d'estimation de la position et de la vitesse basée sur l'information spatiale et temporelle qui permet un suivi particulièrement efficace du déplacement d'une cible dans un réseau de capteurs binaires.

2 Suivi de cible dans un réseau de capteurs binaires

Nous supposons que notre cible évolue suivant un modèle markovien :

$$\mathbf{x}_t | \mathbf{x}_{t-1} \sim \mathcal{N}(F_t \mathbf{x}_{t-1}, \mathbf{Q}_t) \quad (1)$$

pour tout $t = 1, 2, \dots$ avec $\mathcal{N}(\mu, \sigma^2)$ une loi normale de moyenne μ et de variance σ^2 . La position de départ de la cible est supposée inconnue. A chaque instant, les capteurs nous donnent une information binaire, signifiant si la cible s'approche ou s'éloigne dudit capteur. Nous connaissons dès lors la zone où se situe notre cible à chaque instant.

2.1 Problème de localisation de la cible

Notons A l'ensemble des capteurs dont la cible s'approche et B celui dont elle s'éloigne, et $C(A)$ et $C(B)$ leurs enveloppes convexes :

Proposition 1 $C(A) \cap C(B) = \emptyset$ et $\mathbf{x}_t \notin C(A) \cup C(B)$.

Donc notre cible se trouve entre ces deux espaces. De plus $C(A)$ et $C(B)$ étant deux espaces convexes disjoints, il existe un hyperplan séparateur. Alors, avec t_k la position du capteur k , nous pouvons écrire $\mathbf{t}_k = \lambda \mathbf{v}_t + \mu \mathbf{v}_t^\perp$, tel que :

$$\langle \mathbf{t}_k, \mathbf{v}_t \rangle = \lambda \|\mathbf{v}_t\|^2 > 0 \iff \lambda > 0. \quad (2)$$

Ce qui signifie que la droite générée par le vecteur \mathbf{v}_t^\perp sépare $C(A)$ et $C(B)$. Puis, considérant la base $\{\mathbf{v}_t, \mathbf{v}_t^\perp\}$, nous avons :

$$\begin{cases} \mathbf{t}_k \in A \iff \lambda \|\mathbf{v}_t\|^2 > \langle \mathbf{x}_t, \mathbf{v}_t \rangle, \\ \mathbf{t}_k \in B \iff \lambda \|\mathbf{v}_t\|^2 < \langle \mathbf{x}_t, \mathbf{v}_t \rangle. \end{cases} \quad (3)$$

Ainsi, dans la base $(\mathbf{v}_t, \mathbf{v}_t^\perp)$, la droite passant par le point $\left(\frac{\langle \mathbf{x}_t, \mathbf{v}_t \rangle}{\|\mathbf{v}_t\|^2}, 0\right)$ et de direction \mathbf{v}_t^\perp sépare $C(A)$ et $C(B)$. Cela signifie que si nous estimons cette droite séparatrice, nous obtenons le vecteur vitesse unitaire.

2.2 Estimation de la vitesse

Le but clairement défini par le précédent paragraphe étant d'estimer cette droite séparatrice, nous utilisons la méthode des Séparateurs à Vaste Marge (SVM) [2], largement utilisée dans des contextes de classification. Soient $\{(y_1, \mathbf{x}_1), \dots, (y_l, \mathbf{x}_l)\}$ ($y_i \in \{-1, 1\}$ et \mathbf{x}_i positions des capteurs). Le problème est dit linéairement séparable s'il existe un vecteur \mathbf{w} et un scalaire b tels que :

$$\begin{cases} \langle \mathbf{w}, \mathbf{x}_i \rangle + b \geq 1 & \text{si } : y_i = 1, \\ \langle \mathbf{w}, \mathbf{x}_i \rangle + b \leq -1 & \text{si } : y_i = -1. \end{cases} \quad (4)$$

Soit $\mathcal{H}(\mathbf{w}, b) \triangleq \{\mathbf{x} | \langle \mathbf{w}, \mathbf{x} \rangle + b = 0\}$ (\mathbf{w} : vecteur normal) l'hyperplan séparateur optimal, nous définissons la marge comme la distance du capteur le plus proche à l'hyperplan, ce qui nous donne $\text{marge} = \frac{1}{\|\mathbf{w}\|}$. Dès lors, maximiser la marge revient au problème suivant :

$$\begin{cases} \min_{\mathbf{w}, b} \tau(\mathbf{w}) \stackrel{\delta}{=} \|\mathbf{w}\|^2, \\ \text{s.t. } : y_i (\langle \mathbf{w}, \mathbf{x}_i \rangle + b) \geq 1 \quad \forall i = 1, \dots, l \quad y_i = \pm 1. \end{cases} \quad (5)$$

Le problème dual peut être résolu par des méthodes de programmation quadratiques classiques. Le cas un peu moins parfait revient à considérer qu'il ne peut y avoir de séparation sans erreur, et nous remplaçons les contraintes de l'équation 5 par :

$$y_i (\langle \mathbf{w}, \mathbf{x}_i \rangle + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad i = 1, \dots, l. \quad (6)$$

Ceci nous permet donc d'avoir une estimation de notre vecteur vitesse unitaire instant après instant.

2.3 Intégration Multi-Période

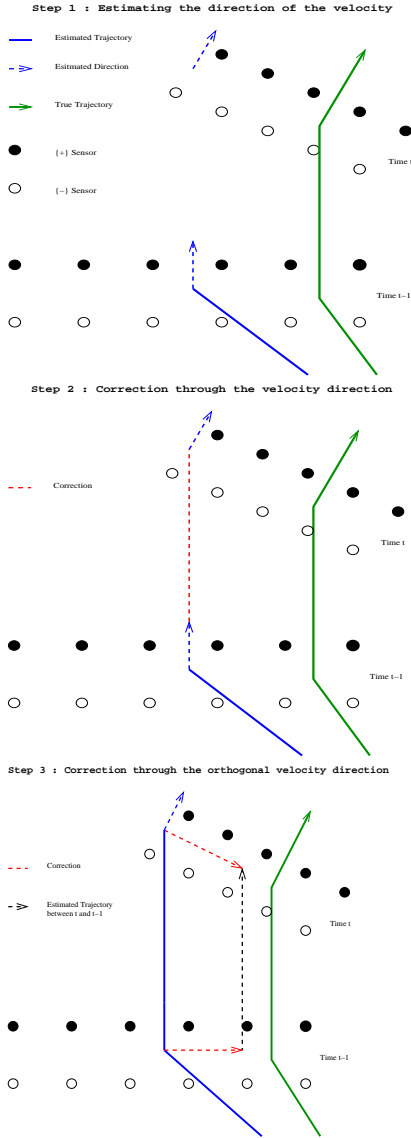


FIG. 1 – Les étapes de l'algorithme.

Il nous reste désormais à estimer la valeur de la vitesse de notre cible ainsi que sa position, ce qui n'est pas un moindre problème. Nous avons décidé de prendre en compte les informations passées pour construire des estimateurs de position et de vitesse. Pour cela, nous avons développé un algorithme en

Algorithm 1 Algorithme de suivi pour un réseau de capteurs binaires

Require: $\forall t, S_t$

- 1: $\hat{X}_0 \sim \mathcal{U}(S_0)$
- 2: $\hat{V} \leftarrow \mathcal{N}(m, \sigma^2)$
- 3: **for** $t = 1$ to T **do**
- 4: $\hat{v}_t \leftarrow SVM(S_t)$
- 5: $\lambda_t \leftarrow \frac{vs(\hat{V}) - \langle \hat{v}_t, \hat{\mathbf{x}}_{t-1} \rangle}{\langle \hat{v}_t, \hat{v}_{t-1} \rangle}$
- 6: $\theta_t \leftarrow \frac{|\hat{V}| - \lambda_t}{\langle \hat{v}_t^\perp, \hat{v}_{t-1} \rangle}$
- 7: $\hat{X}_t \leftarrow \lambda_t (\hat{V}) \hat{v}_t + \theta_t (\hat{V}) \hat{v}_t^\perp$
- 8: $\hat{V} \leftarrow \hat{X}_t - \hat{X}_{t-1}$
- 9: **end for**
- 10: **return** \hat{X}, \hat{V}

deux étapes. Lors de la première, nous proposons une correction le long de la direction de vitesse estimée, appelée λ_t (qui s'avère estimer la norme de la vitesse), puis lors de la deuxième étape, une correction orthogonale est ajoutée, appelée θ_t (qui estime la position). Voir la figure 1 pour une compréhension graphique de l'algorithme.

2.3.1 Mise en équation et algorithme

Les expressions mathématiques des corrections sont les suivantes :

$$\lambda_t = \frac{vs_{moy}^{(+,-)} - \langle \hat{v}_t, \hat{\mathbf{x}}_{t-1} \rangle}{\langle \hat{v}_t, \hat{v}_{t-1} \rangle}, \quad vs_{moy}^{(+,-)} = \frac{vs_{max}^{(-)} + vs_{min}^{(+)}}{2}, \quad \text{et} \quad \theta_t = \frac{m_{t,k} - \lambda_t}{\langle \hat{v}_t^\perp, \hat{v}_{t-1} \rangle} \quad (7)$$

Ici $m_{t,k}$ correspond à la moyenne des k dernières valeurs de vitesse et vs l'ensemble des capteurs de même information binaire.

L'estimation finale de la position devient alors :

2.3.2 Résultats

$$\hat{\mathbf{x}}_t^{fin} = \lambda_t \hat{v}_t + \theta_t \hat{v}_t^\perp. \quad (8)$$

Nous présentons les résultats de simulations pour une cible se déplaçant dans un espace de 200m x 200m, couvert par 81 capteurs. La variance de la trajectoire est faible, et la durée du suivi est de 30 secondes. Un résultat de suivi est présenté sur la figure 2. En rouge la vraie trajectoire, et en vert l'estimation. La figure de droite présente l'erreur quadratique moyenne de l'estimateur de la position pour trois topologies de réseau différentes. Toutefois, la conclusion reste la même. Nous observons une forte réduction de l'EQM au cours du temps. Des résultats équivalents sont observés quant à l'estimation de la vitesse au cours du temps. Ceci nous amène à conclure sur la pertinence de l'algorithme utilisé.

3 Conclusion

Nous avons développé une nouvelle méthode pour le suivi de cible dans un réseau de capteurs binaires spécifiques. Bien que l'information instantannée soit relativement faible, nous avons réussi à utiliser l'information passée pour pouvoir estimer le

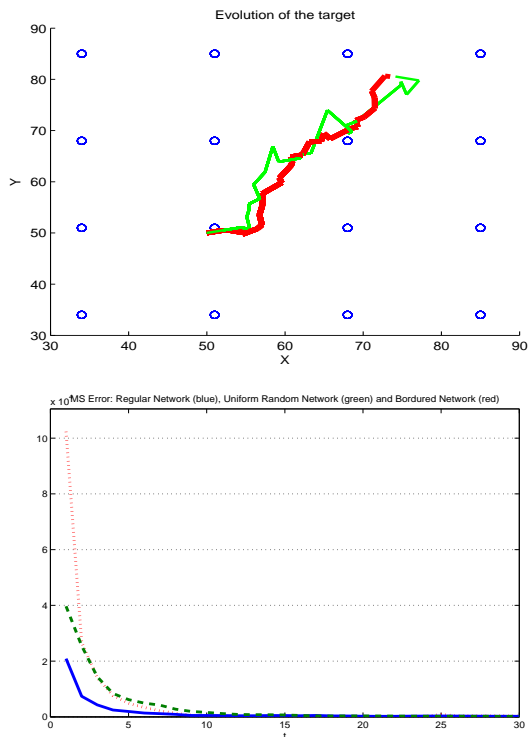


FIG. 2 – Suivi de cible à l'aide de capteurs binaires.

plus finement possible à la fois la position et la vitesse de la cible. De plus, notre algorithme reste rapide et fiable. Il serait intéressant d'analyser l'effet de la fréquence d'acquisition des données sur nos résultats de suivi.

Les prochains développements de cette méthode prévoient l'extension au suivi multicible [3]. Bien que nous ne soyons qu'au début de nos travaux, nous prévoyons que l'analyse spatio-temporelle permettra de s'extraire des problèmes d'associations compliqués inhérents à la dimension multicible. Enfin, nous prévoyons des expériences réelles, basées sur des capteurs binaires classiques (Doppler) ou encore avec d'autres capteurs aux méthodes d'extractions binaires (flot optique pour les caméras vidéos notamment).

Références

- [1] J. ASLAM, Z. BUTLER, F. CONSTANTIN, V. CRESPI, G. CYBENKO, D. RUS : Tracking a moving object with a binary sensor network.
Proc. of the 1st international Conference on Embedded Networked Sensor Systems, Nov 2005, pp. 150–161.
- [2] C. CORTES, V. VAPNIK : Support-Vector Networks.
Machine Learning, **20**, 1995, pp. 273–297.
- [3] X. WANG and B. MORAN : Multitarget Tracking Using Virtual Measurements of Binary Sensor Networks.
Proc. of the 9-th Int. Conf. on Information Fusion, Jul. 2006.