

Le suivi des obstacles

Christophe BLANC
LASMEA
Université Blaise Pascal
www.christophe-blanc.info

13 janvier 2009

Le rôle du pistage de cible est avant tout d'assurer la cohérence temporelle des observations afin d'éviter les fausses alarmes et de filtrer l'état. La structure haut niveau d'un tel système est donnée figure 1. Les capteurs délivrent des signaux qui sont collectés par un module de traitement du signal délivrant des mesures au traitement des données. Les pistes sont des éléments du sous système de traitement des données dont le rôle est de former et de maintenir les pistes. Le suivi d'obstacles est un

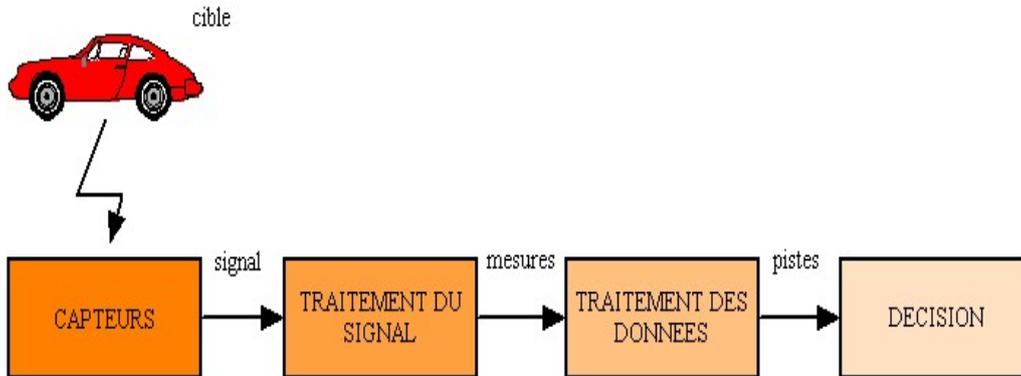


FIG. 1 – Structure haut niveau d'un système de surveillance

élément indispensable pour un système de détection d'obstacles. Son rôle est de déterminer le nombre, la position, et le mouvements des différentes pistes. Un système de pistage repose sur plusieurs briques dont la principale est celle qui permet une estimation récursive de l'état de la piste (position, vitesse, voire accélération). Cette brique s'appuie sur une méthode de filtrage telle que le filtrage de Kalman ou le filtrage particulaire. D'autres briques comme l'initialisation, la destruction, l'association des pistes sont indispensables au bon fonctionnement (voir figure 2). En effet, toutes les mesures délivrées par les

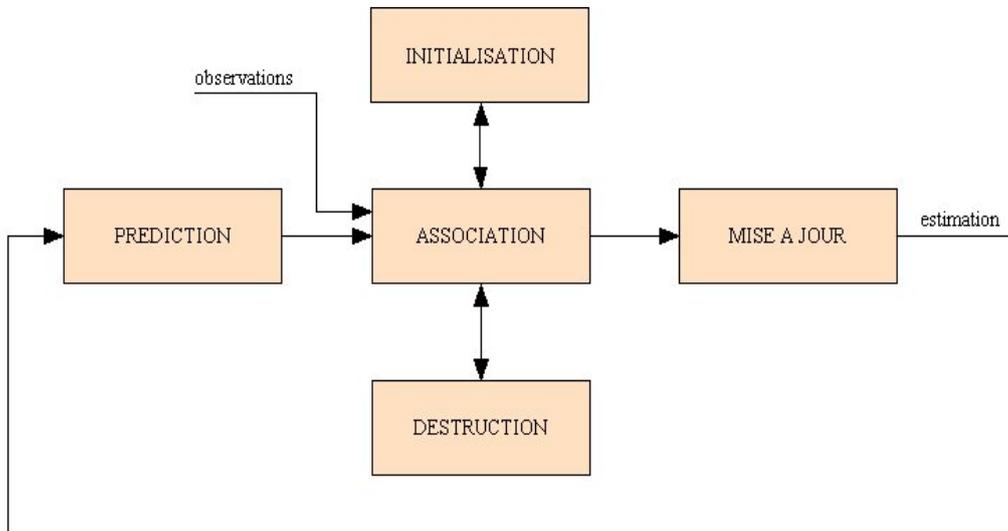


FIG. 2 – Un module de pistage

capteurs ne sont pas forcément des mesures d'obstacles mais peuvent provenir d'interférences aléatoires (conditions atmosphériques, fausses alarmes,...). Il est donc nécessaire de valider les mesures qui correspondent le mieux à l'estimation : c'est l'association. Une piste tentée est typiquement issue d'une mesure non associée avec les pistes existantes. La confirmation d'une piste est basée sur le nombre de mises à jour effectuées pendant un temps fixé : c'est l'initialisation. Une piste est dite supprimée si elle n'est pas mise à jour par une mesure pendant un laps de temps fixé : c'est la destruction.

L'estimation récursive des caractéristiques de l'obstacle à suivre sera abordée dans une première partie. La deuxième partie décrit les autres briques nécessaires à la gestion des pistes (association, initialisation, destruction). Enfin, des résultats seront présentés pour différentes sources d'informations avec des comparaisons sur les estimations faites par les méthodes utilisées.

1 Estimation récursive des caractéristiques de la piste

Le pistage est utilisé afin de fournir la description d'un système dynamique. Il s'agit d'estimer à l'instant k l'état X_k d'un système, à partir d'observations (Z_0, \dots, Z_k) entachées d'un bruit blanc. Les observations seules ne permettent pas d'estimer cet état et il est nécessaire d'établir un modèle suffisamment réaliste pour représenter l'évolution de l'état entre deux instants d'observations. Considérons le modèle d'évolution d'une piste (obstacle à suivre) suivant :

$$X_k = f_{k-1}(X_{k-1}, V_{k-1}) \quad (1)$$

où $f_{k-1} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_v} \rightarrow \mathbb{R}^{n_x}$ est une fonction, pouvant être non linéaire, de l'état X_{k-1} , et de V_{k-1} qui représente le bruit sur l'état dont la distribution de probabilité est $p_{V_{k-1}}(\cdot)$. n_x et n_v sont les dimensions du vecteur d'état et du bruit. Ce modèle décrit un processus aléatoire qui a la propriété que son évolution (passage de X_k à X_{k-1}) ne dépend que de l'état courant et non de son passé. C'est un chaîne de Markov i.e :

$$p(X_k/X_{k-1}, Z_{1:k-1}) = p(X_k/X_{k-1}) \quad (2)$$

L'objectif du suivi est d'estimer récursivement X_k à partir des mesures :

$$Z_k = h_k(X_k, W_k) \quad (3)$$

où $h_k : \mathbb{R}^{n_x} \times \mathbb{R}^{n_w} \rightarrow \mathbb{R}^{n_z}$ est une fonction de X_k pouvant être non linéaire, et W_k est le bruit sur les mesures dont la distribution de probabilité est $p_{W_k}(\cdot)$. n_z et n_w sont les dimensions du vecteur de mesure et du bruit de mesure.

1.1 Estimation bayésienne

Le problème de suivi d'un obstacle, du point de vue bayésien, peut être vu comme le calcul récursif de degrés de croyance en l'état X_k au temps k , en prenant en compte l'historique des mesures $Z_{1:k} = Z_1, \dots, Z_k$. Il est donc nécessaire de construire la densité de probabilité a posteriori (notée pdf dans la suite) $p(X_k/Z_{1:k})$. On considère que la pdf initiale est connue et est donnée par $p(X_0/Z_0) \equiv p(X_0)$. Alors, par principe, la pdf $p(X_k/Z_{1:k})$ peut être obtenue récursivement en deux étapes : la prédiction et la mise à jour.

La première étape consiste en la prédiction à partir de $p(X_{k-1}/Z_{1:k-1})$ (supposée connue à l'instant $k-1$) en utilisant le modèle Markovien définie en 1, afin d'obtenir $p(X_k/Z_{1:k-1})$ via l'équation de Chapman-Kolmogorov [27] [35] :

$$p(X_k/Z_{1:k-1}) = \int p(X_k/X_{k-1})p(X_{k-1}/Z_{1:k-1})dX_{k-1} \quad (4)$$

$p(X_k/X_{k-1})$ est une densité de transition définie par le système d'équation 1. La deuxième étape utilise la règle de Bayes afin de mettre à jour $p(X_k/Z_{1:k})$ en fonction des nouvelles mesures. On a :

$$p(X_k/Z_{1:k}) = \frac{p(Z_k/X_k)p(X_k/Z_{1:k-1})}{p(Z_k/Z_{1:k-1})} \quad (5)$$

où d'après l'équation de Chapman-Kolmogorov :

$$p(Z_k/Z_{1:k-1}) = \int p(Z_k/X_k)p(X_k/Z_{1:k-1})dX_k \quad (6)$$

dépend de la fonction de vraisemblance $p(Z_k/X_k)$ définie par le modèle de mesure 3. Il est aussi possible d'obtenir une formule récursive pour cette distribution :

$$p(X_{k+1}/Z_{1:k+1}) = p(X_k/Z_{1:k}) \frac{p(Z_{k+1}/X_{k+1})p(X_{k+1}/X_k)}{p(Z_{k+1}/Z_{1:k})} \quad (7)$$

La solution optimale bayésienne est basée sur les relations récurrentes 4 et 5. Cependant, ces relations ne permettent pas un calcul analytique de la densité de probabilité. Sous certaines hypothèses, des solutions optimales ou sous-optimales existent. Si les modèles d'état et de mesure sont linéaires et les bruits

considérés comme gaussiens alors les équations de Kalman [18], ou des méthodes numériques (grid-based method [1]) permettent une estimation optimale. Cependant, la condition de linéarité et l'hypothèse gaussienne sont strictes pour certaines applications. D'autres méthodes permettent une estimation sous-optimale dans des cas non-linéaires ou non-gaussiens : EKF (Extended Kalman Filter [5]), UKF (Unscented Kalman Filter [33]), méthodes numériques approximées (approximated grid-based method[1]), filtre à particules [13]. Une description précise de ces méthodes est donnée dans [29]. Les filtres basés sur les méthodes de Kalman (EKF, UKF) supposent la pdf $p(X_k/Z_k)$ gaussienne, ils ne sont donc pas adaptés si celle-ci ne peut pas être considérée comme gaussienne. Les "grid-based method" souffrent de la croissance exponentielle de la complexité de leur algorithmes. Les filtres à particules, quant à eux, basés sur les méthodes de Monte-Carlo, sont des systèmes qui permettent une estimation dans des cas non-linéaires et non-gaussiens. Dans la suite, on présentera les méthodes, utilisées dans notre système, basées sur les filtres de Kalman et les méthodes particulaires.

1.2 Filtre de Kalman

Les filtres de Kalman sont des méthodes récursives qui supposent que la densité de probabilité de l'état conditionné par les mesures est une loi normale d'espérance l'estimation de l'état X_k et de variance la covariance sur cet état P_k .

$$p(X_k/Z_k) \sim N(X_k, P_k)$$

Si on considère que $p(X_{k-1}/Z_{k-1})$ est gaussienne alors $p(X_k/Z_k)$ est aussi gaussienne si :

- les bruits sur l'état, et de mesure sont indépendants et gaussiens. Selon cette hypothèse on a :

$$V_{k-1} \sim N(0, Q_{k-1})$$

$$W_k \sim N(0, C_k)$$

où Q_{k-1} et C_k modélisent respectivement les covariances du bruit sur l'état et du bruit sur la mesure.

- $f_{k-1}(X_{k-1}, V_{k-1})$ est une fonction linéaire connue
- $h_{k-1}(X_k, W_k)$ est une fonction linéaire connue

On a donc, à partir des équations 4 et 5 :

$$p(X_{k-1}/Z_{k-1}) \sim N(\tilde{X}_{k-1/k-1}, P_{k-1/k-1})$$

$$p(X_k/Z_{k-1}) \sim N(\hat{X}_{k/k-1}, P_{k/k-1})$$

$$p(X_k/Z_k) \sim N(\tilde{X}_{k/k}, P_{k/k})$$

De plus, les systèmes 1 et 3 peuvent s'écrire :

$$X_k = F_{k-1}X_{k-1} + V_{k-1} \tag{8}$$

$$Z_k = H_k X_k + W_k \tag{9}$$

où F_{k-1} et H_k sont des matrices connues définies par les fonctions linéaires f_{k-1} et h_k . Sous ces hypothèses, l'utilisation des équations de Kalman [18] permet une estimation optimale de l'état de notre obstacle.

Cet algorithme est une solution optimale sous les conditions de linéarité et gaussienne. Une autre méthode existe si les fonctions h_k ou/et f_{k-1} sont non linéaires : le filtre de Kalman étendu (EKF). Elle consiste en une linéarisation locale en approximant les fonctions non-linéaires par le premier terme du développement en série de Taylor. On a :

$$\hat{X}_{k/k-1} = f_{k-1}(\tilde{X}_{k-1/k-1}) \tag{10}$$

$$\tilde{X}_{k/k} = \hat{X}_{k-1/k} + K_k(Z_k - h_k(\hat{X}_{k/k-1})) \quad (11)$$

et les matrices F_{k-1} et H_k sont les matrices jacobiennes utilisées pour la linéarisation. Ces méthodes supposent toujours que $p(X_k/Z_k)$ est gaussienne. Dans le cas de forte non linéarité, le caractère non gaussien de la pdf devient de plus en plus prononcé et la performance d'un tel filtre sera dégradée. La méthode basée sur le filtre UKF¹ nécessite aussi le caractère gaussien de la pdf $p(X_k/Z_k)$. Cet algorithme, récent, est décrit dans [33, 17]. Contrairement à l'EKF, l'UKF n'approxime pas les fonctions non-linéaires f_{k-1} et h_k mais approxime la pdf $p(X_k/Z_k)$ par une densité gaussienne représentée par un ensemble d'échantillons choisis de façon déterministe. La première phase de ce filtrage sera donc la représentation de cette densité à l'instant $k-1$ par un ensemble de N points X_{k-1}^i pondérés par W_{k-1}^i , $i = 0, \dots, N-1$. L'étape de prédiction est alors réalisée :

$$\hat{X}_{k/k-1} = \sum_{i=0}^{N-1} W_{k-1}^i f_{k-1}(X_{k-1}^i) \quad (12)$$

$$\hat{P}_{k/k-1} = Q_{k-1} + \sum_{i=0}^{N-1} W_{k-1}^i \left[f_{k-1}(X_{k-1}^i) - \hat{X}_{k/k-1} \right] \left[f_{k-1}(X_{k-1}^i) - \hat{X}_{k/k-1} \right]^t \quad (13)$$

La pdf prédite $p(X_k/Z_{k-1})$ est représentée par un ensemble de N échantillons :

$$X_{k/k-1}^i = f_{k-1}(X_{k-1}^i) \quad (14)$$

La mesure prédite est donnée par :

$$\hat{Z}_{k/k-1} = \sum_{i=0}^{N-1} W_{k-1}^i h_k(X_{k/k-1}^i) \quad (15)$$

et la mise à jour est effectuée par les relations suivantes :

$$\tilde{X}_{k/k} = \hat{X}_{k/k-1} + K_k(Z_k - \hat{Z}_{k/k-1}) \quad (16)$$

$$\tilde{P}_{k/k} = \hat{P}_{k/k-1} - K_k S_k K_k^t \quad (17)$$

où

$$K_k = P_{xz} S_k^{-1} \quad (18)$$

$$S_k = R_k + P_{zz} \quad (19)$$

$$P_{xz} = \sum_{i=0}^{N-1} W_{k-1}^i (X_{k/k-1}^i - \hat{X}_{k/k-1}) (h_k(X_{k/k-1}^i) - \hat{Z}_{k/k-1})^t \quad (20)$$

$$P_{zz} = \sum_{i=0}^{N-1} W_{k-1}^i (h_k(X_{k/k-1}^i) - \hat{Z}_{k/k-1}) (h_k(X_{k/k-1}^i) - \hat{Z}_{k/k-1})^t \quad (21)$$

Toutes les méthodes basées sur ce type de filtre ne diffèrent que par la sélection des échantillons (nombre,

¹Unscented Kalman Filter

valeur, poids).

En résumé, si la pdf $p(X_k/Z_k)$ peut être approximée par une gaussienne, les méthodes s'appuyant sur les filtres de Kalman délivrent des estimations optimales/sous optimales selon les conditions de linéarité/non-linéarité des modèles d'état 1 et de mesure 3.

1.3 Filtre à particules

Le filtrage particulaire, développé à l'origine dans [13], connaît actuellement un fort développement dans de nombreux domaines (vision, localisation, navigation, robotique,...), en particulier dans le pistage multi-cibles.

Ce filtrage est une méthode séquentielle de Monte-Carlo dans laquelle des particules parcourent l'espace d'état de manière indépendante, et interagissent sous l'effet d'une fonction de vraisemblance qui concentre automatiquement les particules dans les régions d'intérêt de l'espace d'état. En fait, à chaque instant k , une particule a d'autant plus de chance de se reproduire à l'instant suivant qu'elle est cohérente avec l'observation courante. Cette méthode a l'avantage de ne pas nécessiter de contraintes linéaires ou gaussiennes sur le modèle. De plus, elle est très facile à implémenter, puisqu'il suffit de savoir simuler de manière indépendante différentes trajectoires du modèle.

Pour détailler cet algorithme, supposons que l'on soit capable de simuler N_S échantillons aléatoires à partir de $p(X_{0:k}/Z_{1:k})$:

$$\{X_{0:k}^i\}_{i=0}^{N_S-1}$$

On pourra par exemple prendre cette approximation pour représenter la densité de probabilité a posteriori :

$$p(X_{0:k}/Z_{1:k}) \approx \frac{1}{N_S} \sum_{i=0}^{N_S-1} \delta(X_{0:k} - X_{0:k}^i) \quad (22)$$

où δ est une fonction de Dirac. Cette approximation converge vers la densité de probabilité si N_S est suffisamment grand (loi des grands nombres)[9].

Cependant, il est habituellement impossible d'échantillonner efficacement à partir de $p(X_{0:k}/Z_{1:k})$ à chaque instant k puisque cette distribution n'est pas connue et doit être estimée. Une solution consiste à échantillonner à partir d'une distribution connue $q(X_{0:k}/Z_{1:k})$. C'est le principe d'échantillonnage d'importance (Importance Sampling [10]). On introduit une densité d'importance (fonction d'importance) $q(\cdot)$ telle que $p(\cdot) \propto q(\cdot)$, à partir de laquelle on échantillonnera N_S particules. Alors, une approximation de la distribution quand N_S est grand est donnée par [1] :

$$p(X_{0:k}/Z_{1:k}) = \sum_{i=0}^{N_S-1} w_{0:k}^i \delta(X_{0:k} - X_{0:k}^i) \quad (23)$$

$$\tilde{w}_{0:k}^i = \frac{p(X_{0:k}^i/Z_{1:k})}{q(X_{0:k}^i/Z_{1:k})} \quad (24)$$

$$w_{0:k}^i = \frac{\tilde{w}_{0:k}^i}{\sum_{i=0}^{N_S-1} \tilde{w}_{0:k}^i} \quad (25)$$

où les $\tilde{w}_{0:k}^i, w_{0:k}^i$ sont respectivement les poids non normalisés et les poids normalisés.

Cette méthode n'est cependant pas adéquate pour une estimation récursive. En effet, on a besoin d'avoir toutes les mesures $Z_{1:k}$ pour estimer $p(X_{0:k}/Z_{1:k})$. La méthode précédente peut être modifiée pour qu'il devienne possible de calculer l'estimée de $p(X_{0:k}/Z_{1:k})$ sans utiliser tout l'historique de l'état. Il faut choisir la densité d'importance telle que :

$$q(X_{0:k}/Z_{1:k}) = q(X_{0:k-1}/Z_{1:k-1})q(X_k/X_{0:k-1}, Z_{1:k}) \quad (26)$$

En utilisant les équations 7, 24 et 26 on a :

$$\tilde{w}_k^i = \frac{p(X_{k-1}^i/Z_{1:k-1})p(Z_k/X_k^i)p(X_k^i/X_{k-1}^i)}{q(X_{0:k-1}^i/Z_{1:k-1})p(Z_k/Z_{1:k-1})q(X_k^i/X_{0:k-1}^i, Z_{1:k})} \quad (27)$$

$$\tilde{w}_k^i = \tilde{w}_{k-1}^i \frac{p(Z_k/X_k^i)p(X_k^i/X_{k-1}^i)}{p(Z_k/Z_{1:k-1})q(X_k^i/X_{0:k-1}^i, Z_{1:k})} \propto \tilde{w}_{k-1}^i \frac{p(Z_k/X_k^i)p(X_k^i/X_{k-1}^i)}{q(X_k^i/X_{0:k-1}^i, Z_{1:k})} \quad (28)$$

De plus, on n'a besoin que d'une estimation de $p(X_k/Z_{1:k})$. On ne garde donc en mémoire que X_k^i . Les séquences $X_{0:k-1}^i$ et l'historique des mesures $Z_{1:k-1}$ ne sont plus nécessaires. La formule de récurrence pour les poids modifiés est donc :

$$\tilde{w}_k^i = \tilde{w}_{k-1}^i \frac{p(Z_k/X_k^i)p(X_k^i/X_{k-1}^i)}{q(X_k^i/X_{k-1}^i, Z_k)} \quad (29)$$

Cet algorithme consiste donc en la propagation récursive des particules et de leurs poids associés à chaque réception de mesure.

Cependant, un problème commun à tous ces types d'algorithmes est le phénomène de dégénérescence des particules. En effet, après quelques itérations, les descendants d'une même particule qui a été multipliée occupent toutes la même position. Toutes ces particules ont un poids négligeable sauf une. Une mesure caractéristique de cette dégénérescence est définie par [20] :

$$N_{eff} = \frac{N_S}{1 + Var(w_k^{*i})} \quad (30)$$

où $w_k^{*i} = \frac{p(X_k^i/Z_{1:k})}{q(X_k^i/X_{k-1}^i, Z_k)}$ est défini comme "poids vrai". Cependant, on ne peut pas l'évaluer exactement. On utilise donc un estimée de N_{eff} qui peut être obtenu par :

$$\hat{N}_{eff} = \frac{1}{\sum_{i=0}^{N_S-1} (w_k^i)^2} \quad (31)$$

La dégénérescence est d'autant plus forte que \hat{N}_{eff} est petit. Une approche brute pour réduire ce phénomène est d'augmenter considérablement N_S , ce qui est le plus souvent impraticable. Plusieurs méthodes pour réduire ce phénomène existent. La première consiste à effectuer un choix adéquate pour la fonction d'importance. La seconde repose sur l'utilisation d'un rééchantillonnage. Toutes les variantes du filtrage particulaire se différencient par le choix de la densité d'importance et/ou la modification du rééchantillonnage.

Une méthode consiste à choisir la densité d'importance de telle façon qu'elle minimise la variance des poids d'importance conditionnellement à la trajectoire simulée et aux mesures $Z_{0:k}$. Cette fonction d'importance est $p(X_k/X_{k-1}^i)$ [8] i.e :

$$q(X_k^i/X_{k-1}^i) = p(X_k/X_{k-1}^i) \quad (32)$$

A partir de l'équation 29 on obtient :

$$\tilde{w}_k^i = \tilde{w}_{k-1}^i p(Z_k/X_k^i) \quad (33)$$

Cela semble être le meilleur choix pour la fonction d'importance puisque celle ci permet un calcul des poids de manière intuitive et simple. Dans la suite, cette densité d'importance sera conservée.

Une autre méthode consiste à rééchantillonner les particules quand le phénomène de dégénérescence est significatif. Pour ceci, si $\hat{N}_{eff} < seuil$, on tire (avec remplacement) N_S particules parmi le jeu courant, de façon à favoriser les particules de plus forte vraisemblance (de plus fort poids). Le poids affecté à chaque particule est alors fixé à l'inverse du nombre de particules. Même si cette méthode réduit le phénomène de dégénérescence, d'autres problèmes apparaissent. En effet, les particules de plus fort poids sont statistiquement sélectionnées plusieurs fois ce qui provoque la perte de diversité à travers les particules.

Les particules ne sont concentrées qu'autour d'un seul point. Des techniques existent pour résoudre ce problème : "resample move algorithm [12]", "regularisation [25]".

De nombreuses variantes du filtrage particulaire utilisant le rééchantillonnage sont décrites dans [14]. Une des variantes les plus courantes, consiste à effectuer un rééchantillonnage systématique (même si $\hat{N}_{eff} > seuil$). On parle ici de filtre SIR (Sampling Importance Resampling) ou encore de " bayesian bootstrap " [13]. Les hypothèses émises pour l'utilisation de cet algorithme sont limitées. Il faut premièrement connaître les fonctions caractérisant la dynamique de l'état et des mesures (1 et 3 respectivement). Il faut ensuite pouvoir échantillonner à partir de la distribution du bruit et de $p(X_k/X_{k-1}^i)$ utilisée ici comme densité d'importance. En effet, un échantillon X_k^i de $p(X_k/X_{k-1}^i)$ peut être engendré par la génération d'un échantillon de bruit V_{k-1}^i à partir de $p_{V_{k-1}}(V_{k-1})$ d'une part et par l'utilisation du modèle défini en 1 : $X_k^i = F_k(X_{k-1}^i, V_{k-1}^i)$ d'autre part.

Chaque poids est donc calculé à partir de l'équation 32 et puisque $\forall i, w_{k-1}^i = \frac{1}{N_S}$ on a :

$$w_k^i = p(Z_k/X_k^i) \quad (34)$$

Il faut donc connaître aussi la fonction de vraisemblance. On désigne par :

$$p_{W_k}(Z_k - H_k(X_k^i)) \quad (35)$$

la fonction de vraisemblance, qui quantifie l'adéquation de chaque particule X_k^i par rapport à l'observation courante Z_k : cette fonction dépend de l'écart $Z_k - H_k(X_k^i)$ entre l'observation $H_k(X_k^i)$ à laquelle on s'attend si le système occupait l'état X_k^i , et l'observation réelle Z_k [19] : c'est l'innovation.

Avant chaque rééchantillonnage, les poids calculés à partir de la vraisemblance 34 sont normalisés.

L'algorithme ci-dessous décrit deux variantes du filtrage particulaire suivant la méthode de rééchantillonnage effectuée.

INITIALISATION (i) $k = 0$

Un jeu de N_S particules est g n r e   partir de la valeur initiale du vecteur d' tat X_0

$$X_k^i \sim p(X_0)$$

Chaque poids est calcul  en fonction de la vraisemblance : c'est la pond ration

$$w_0^i = p_{W_0}(Z_0 - H_0(X_0^i)), \forall i = 1, \dots, N_S$$

  tout instant $k \geq 1$

PREDICTION (ii)

On g n re des  chantillons de bruit sur l' tat

$$V_k^i \sim p_{V_k}(\cdot), \forall i = 1, \dots, N_S$$

chaque particule est donn e par

$$X_k^i = F_k(X_{k-1}^i, V_k^i), \forall i = 1, \dots, N_S$$

PONDERATION (iii)

$$w_k^i = w_{k-1}^i p_{W_k}(Z_k - H_k(X_k^i)), \forall i = 1, \dots, N_S$$

normalisation :

$$w_{0:k}^i = \frac{w_{0:k}^i}{\sum_{i=1}^{N_S} w_{0:k}^i}, \forall i = 1, \dots, N_S$$

REECHANTILLONNAGE (iv)

Importance Sampling	Sampling Importance Resampling Bayesian Bootstrap
si $\hat{N}_{eff} = \frac{1}{\sum_{i=1}^{N_S} (w_k^i)^2} < seuil$	on tire (avec remplacement)
on tire (avec remplacement) N_S particules parmi le jeu courant	N_S particules parmi le jeu courant $\forall i, w_k^i = \frac{1}{N_S}$

2 Performances des estimateurs r cursifs : borne de Cramer Rao a posteriori

Le choix de la m thode d'estimation   appliquer est guid  par sa performance. Pour un syst me qui v rifie les conditions gaussienne et de lin arit , alors le filtre de Kalman est optimal. Cependant, dans le cas non lin aire, on ne peut pas conclure sur l'optimalit  du syst me. On utilise donc des comparaisons avec des bornes caract risant la performance optimale. On a vu qu'il existait une borne inf rieure sur la variance des erreurs dans le cas d'estimations de param tres d terministes. Cette notion a  t   tendue dans le cas d'estimation de variables al atoires [32] : borne de Cramer Rao a posteriori². Plus r cemment, dans [31], l'auteur propose une formule r cursive pour le calcul de cette borne.

2.1 Calcul r cursif de la matrice d'information de Fisher

On consid re le syst me d fini par le mod le d' volution et le mod le de mesure d finis respectivement en 1 et 3. Si $\hat{X}_{k/k}$ est un estimateur non biais  de X_k , calcul    partir de la s quence de mesures $Z_k = z_1, \dots, z_k$ et de la connaissance de $p(X_0)$ (pdf initiale), alors la matrice de covariance de $\hat{X}_{k/k}$, not e $P_{k/k}$ admet une borne minimale donn e par :

$$P_{k/k} \triangleq E\{(\hat{X}_{k/k} - X_k)(\hat{X}_{k/k} - X_k)^t\} \geq J_k^{-1} \quad (36)$$

o  la matrice J_k est la matrice d'information de Fisher que l'on cherche   d terminer.

²PCRB : Posterior Cramer Rao Bound

2.1.1 Cas général

Tichavsky et al [31] proposent une méthode pour calculer J_k récursivement :

$$J_{k+1} = D_k^{22} - D_k^{21}(J_k + D_k^{11})^{-1}D_k^{12} \quad (37)$$

où

$$D_k^{11} = E\{-\Delta_{X_k}^{X_k} \log p(X_{k+1}/X_k)\} \quad (38)$$

avec $\Delta_{\Theta}^{\Xi} = \nabla_{\theta} \nabla_{\Xi}^t$

$$D_k^{12} = E\{-\Delta_{X_k}^{X_{k+1}} \log p(X_{k+1}/X_k)\} \quad (39)$$

$$D_k^{21} = E\{-\Delta_{X_{k+1}}^{X_k} \log p(X_{k+1}/X_k)\} = [D_k^{12}]^t \quad (40)$$

$$D_k^{22} = E\{-\Delta_{X_{k+1}}^{X_{k+1}} \log p(X_{k+1}/X_k)\} + E\{-\Delta_{z_{k+1}}^{X_{k+1}} \log p(z_{k+1}/X_{k+1})\} \quad (41)$$

Les espérances $E\{\cdot\}$ dans (38), (39), et (40) sont calculées à partir de X_k et X_{k+1} alors que celle utilisée dans l'équation (41) est évaluée à partir de X_k , X_{k+1} et z_{k+1} .

2.1.2 Cas de Bruit gaussien

Le système est défini par :

$$\begin{aligned} X_{k+1} &= f_k(X_k) + v_k \\ z_{k+1} &= h_{k+1}(X_{k+1}) + w_{k+1} \end{aligned} \quad (42)$$

où les bruits v_k et w_{k+1} sont des bruits blanc, gaussiens et indépendants. Leurs covariances respectives sont données par Q_k et R_{k+1} . On montre facilement que dans ce cas :

$$D_k^{11} = E\{\tilde{F}_k^t Q_k^{-1} \tilde{F}_k\} \quad (43)$$

où

$$\tilde{F}_k = [\nabla_{X_k} f_k^t(X_k)]^t \quad (44)$$

est la matrice jacobienne de $f_k(X_k)$ évaluée à la vraie valeur X_k .

$$D_k^{12} = -E\{\tilde{F}_k^t\} Q_k^{-1} = [D_k^{21}]^t \quad (45)$$

$$D_k^{22} = Q_k^{-1} + E\{\tilde{H}_{k+1}^t R_{k+1}^{-1} \tilde{H}_{k+1}\} \quad (46)$$

où

$$\tilde{H}_{k+1} = [\nabla_{X_{k+1}} h_{k+1}^t(X_{k+1})]^t \quad (47)$$

est la matrice jacobienne de $h_{k+1}(X_{k+1})$ évaluée à la vraie valeur de X_{k+1} . Le problème est de calculer les espérances qui apparaissent dans (46), (45) et (43). Néanmoins, une approximation de Monte Carlo pourra être appliquée. On créera un ensemble de réalisations de vecteurs d'états, trajectoires, et l'espérance sera calculée sur la moyenne de cet ensemble.

2.1.3 Cas linéaire et gaussien

Le cas linéaire et gaussien est donné par le système 42 avec :

$$f_k(X_k) = F_k X_k \quad (48)$$

$$h_{k+1}(X_{k+1}) = H_{k+1} X_{k+1} \quad (49)$$

En remarquant que les jacobiens sont maintenant donnés par :

$$\tilde{F}_k = F_k \quad (50)$$

$$\tilde{H}_{k+1} = H_{k+1} \quad (51)$$

on montre que :

$$D_k^{11} = F_k^t Q_k^{-1} F_k \quad (52)$$

$$D_k^{12} = -F_k^t Q_k^{-1} = [D_k^{21}]^t \quad (53)$$

$$D_k^{22} = Q_k^{-1} + H_{k+1}^t R_{k+1}^{-1} H_{k+1} \quad (54)$$

Finalement, l'équation réursive pour la matrice de Fisher est :

$$J_{k+1} = Q_k^{-1} + H_{k+1}^t R_{k+1}^{-1} H_{k+1} - Q_k^{-1} F_k (J_k + F_k^t Q_k^{-1} F_k)^{-1} F_k^t Q_k^{-1} \quad (55)$$

Si on suppose que la matrice de covariance est telle que $P_{k/k}^{-1} = J_k^{-1}$ alors en utilisant le lemme d'inversion matricielle³ on montre que :

$$P_{k+1/k+1}^{-1} = (Q_k + F_k J_k^{-1} F_k^t)^{-1} + H_{k+1}^t R_{k+1}^{-1} H_{k+1} \quad (56)$$

Une nouvelle application du lemme d'inversion matricielle montre que :

$$P_{k+1/k+1} = P_{k+1/k} - P_{k+1/k} H_{k+1}^t (H_{k+1} P_{k+1/k} H_{k+1}^t + R_{k+1})^{-1} H_{k+1} P_{k+1/k} \quad (57)$$

On retrouve, dans ce cas, l'équation de Kalman pour la covariance sur les erreurs d'estimation définie par :

$$P_{k+1/k+1} = P_{k+1/k} K_{k+1} S_{k+1} K_{k+1}^t \quad (58)$$

avec

$$S_{k+1} = H_{k+1} P_{k+1/k} H_{k+1}^t + R_{k+1} \quad (59)$$

et

$$K_{k+1} = P_{k+1/k} H_{k+1}^t S_{k+1}^{-1} \quad (60)$$

Dans le cas linéaire et gaussien, l'utilisation des équations de Kalman rend l'estimateur efficace, i.e. la borne de Cramer Rao est atteinte pour la covariance sur les erreurs.

2.1.4 Cas déterministe

En l'absence de bruit, l'évolution de l'état est complètement déterministe. Les espérances n'ont plus besoin d'apparaître et on a comme forme réursive l'équation suivante :

$$J_{k+1} = [\tilde{F}_k^{-1}]^t J_k \tilde{F}_k^{-1} + \tilde{H}_{k+1}^t R_{k+1}^{-1} \tilde{H}_{k+1} \quad (61)$$

2.2 PCRB dans le cas linéaire : distribution beta et gaussienne

2.2.1 Calcul de la borne

cas gaussien : voir 2.1.2

cas beta Si les mesures sont bornées par un intervalle, i.e $Z \in [P_L, P_H]$, la distribution la plus appropriée pour modéliser Z est la distribution beta [7]. Sa pdf est donnée par :

$$p(Z) = \frac{\Gamma(\lambda_1 + \lambda_2)}{\Delta P \Gamma(\lambda_1) \Gamma(\lambda_2)} \left(\frac{Z - P_L}{\Delta P} \right)^{\lambda_1 - 1} \left(1 - \frac{Z - P_L}{\Delta P} \right)^{\lambda_2 - 1} \quad (62)$$

où $\Delta P = P_H - P_L$; λ_1, λ_2 sont des paramètres de forme et le symbole Γ représente la fonction gamma. La figure 3 représente cette fonction pour $\lambda_1 = \lambda_2 = 1.1$, $P_L = 22.5$ et $P_H = 45$. On suppose ici que la mesure est une distance comprise entre P_L et P_H (observations de distance radiale donnée par le radar par exemple). On souhaite maintenant calculer la borne de Cramer Rao du système définie ci-dessous :

$$\begin{aligned} X_{k+1} &= F X_k + v_k \\ Z_{k+1} &= H X_{k+1} + w_k \end{aligned} \quad (63)$$

où, pour l'application concernant les observations radar, on a :

$$\begin{aligned} X_k &= \begin{pmatrix} r_k \\ \dot{r}_k \end{pmatrix} \\ Z_k &= \begin{pmatrix} r_{obsk} \\ \dot{r}_{obsk} \end{pmatrix} \end{aligned} \quad (64)$$

$$F = \begin{pmatrix} 1 & T \\ 0 & 1 \end{pmatrix} \quad (65)$$

³ $(A + BCB^t)^{-1} = A^{-1} - A^{-1}B(B^t A^{-1}B + C^{-1})^{-1}B^t A^{-1}$

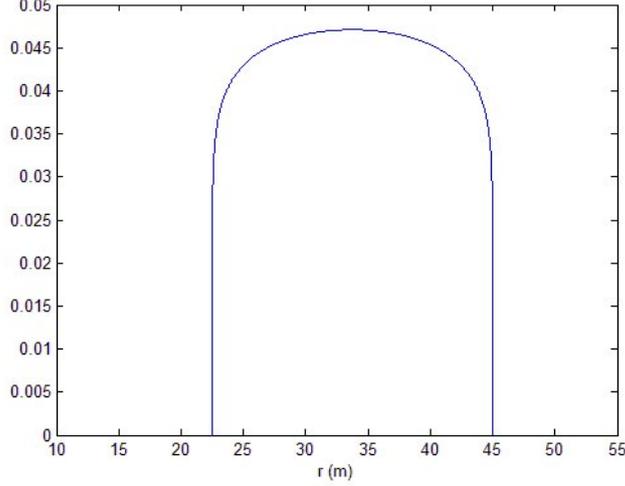


FIG. 3 – La densité de probabilité de la mesure : distribution beta

pour un modèle à vitesse constante et $H = Id_2$. Pour le bruit, on suppose qu'il est gaussien pour l'évolution. Du côté des mesures, le bruit suit une loi de distribution beta pour la distance radiale et une loi normale pour la vitesse radiale. On a donc :

$$\begin{aligned} v_k &\sim N(0, GQG^t) \\ w_k &= \begin{pmatrix} w_{1k} \sim \beta(\lambda_1, \lambda_2) \\ w_{2k} \sim N(0, \sigma_{r_{obs_k}}^2) \end{pmatrix} \end{aligned} \quad (66)$$

où $G = [T^2/2T]^T$ et $Q = \sigma_r^2$. L'utilisation de la formule récursive donnée en 37 et des conditions propres à notre système montre que :

$$D_k^{11} = F^t(GQG^t)^{-1}F \quad (67)$$

$$D_k^{12} = -F^t(GQG^t)^{-1} = [D_k^{21}]^t \quad (68)$$

$$D_k^{22} = (GQG^t)^{-1} + E\{-\Delta_{X_{k+1}}^{X_{k+1}} \log p(Z_{k+1}/X_{k+1})\} \quad (69)$$

En posant $B_k = E\{-\Delta_{X_{k+1}}^{X_{k+1}} \log p(Z_{k+1}/X_{k+1})\}$ et en utilisant le lemme d'inversion matricielle on démontre que :

$$J_{k+1} = B_k + (GQG^t + FJ_k^{-1}F^t)^{-1} \quad (70)$$

Le calcul de B_k est donné ci-dessous. Etant donné notre système, en particulier le modèle de bruit sur les mesures, on a :

$$p(Z_{k+1}/X_{k+1}) = c \left(\frac{r_{k+1} - P_l(r_{obs_{k+1}})}{\Delta P} \right)^{(\lambda_1 - 1)} \left(1 - \frac{r_{k+1} - P_l(r_{obs_{k+1}})}{\Delta P} \right)^{(\lambda_2 - 1)} e^{-\frac{1}{2\sigma_r^2}(r_{obs_{k+1}} - \hat{r}_{k+1})} \quad (71)$$

$$\frac{\partial \log p(Z_{k+1}/X_{k+1})}{\partial r_{k+1}} = (\lambda_1 - 1) \left(\frac{1}{r_{k+1} - P_l(r_{obs_{k+1}})} \right) + (\lambda_2 - 1) \left(\frac{1}{r_{k+1} - P_l(r_{obs_{k+1}}) - \Delta P} \right) \quad (72)$$

$$\frac{\partial^2 \log p(Z_{k+1}/X_{k+1})}{\partial r_{k+1}^2} = -(\lambda_1 - 1) \left(\frac{1}{r_{k+1} - P_l(r_{obs_{k+1}})} \right)^2 - (\lambda_2 - 1) \left(\frac{1}{r_{k+1} - P_l(r_{obs_{k+1}}) - \Delta P} \right)^2 \quad (73)$$

$$\frac{\partial^2 \log p(Z_{k+1}/X_{k+1})}{\partial r_{k+1} \partial \hat{r}_{k+1}} = \frac{\partial^2 \log p(Z_{k+1}/X_{k+1})}{\partial \hat{r}_{k+1} \partial r_{k+1}} = 0 \quad (74)$$

$$\frac{\partial^2 \log p(Z_{k+1}/X_{k+1})}{\partial \hat{r}_{k+1}^2} = -\frac{1}{\sigma_r^2} \quad (75)$$

Finalement, la matrice B_k est définie par :

$$B_k = \begin{pmatrix} E \left(\frac{\lambda_1 - 1}{(r_{k+1} - P_l(r_{obs_{k+1}}))} + \frac{\lambda_2 - 1}{(r_{k+1} - P_l(r_{obs_{k+1}}) - \Delta P)} \right) & 0 \\ 0 & \frac{1}{\sigma_r^2} \end{pmatrix} \quad (76)$$

Cependant, dans le cas où les dénominateurs s'annulent, i.e. l'état rencontre les bornes, alors les termes non nuls de B_k tendent vers l'infini. Il en résulte une convergence rapide de la pcrb de r_k vers zéro. Pour contourner, ce problème on suppose que si $|r_{K+1} - P_l(r_{obs_k})| < \ell$ alors $P_l(r_{obs_k}) = P_l(r_{obs_k}) - \kappa$ et $\Delta P = \Delta P + \kappa$; et si $|r_{K+1} - P_l(r_{obs_k}) - \Delta P| < \ell$ alors $\Delta P = \Delta P + \kappa$.

2.2.2 Simulation : scénario en ligne droite

On propose ici d'évaluer la PCRb pour l'estimation de distance et vitesse radiale dans le cas où la mesure est issue du radar. On considère ici que le bruit sur l'observation de distance radiale suit une distribution beta ou gaussienne, celui de la vitesse radiale est considéré gaussien. Le scénario étudié est un scénario de suivi en ligne droite. On suppose qu'un véhicule s'éloigne à une vitesse relative de 10 m/s. L'initialisation de $J = P_0^{-1}$ est donnée par :

$$P_0 = \begin{pmatrix} 100 & 0 \\ 0 & 10 \end{pmatrix} \quad (77)$$

Pour le problème aux bornes, on utilise $\ell = \kappa = 1$ m. La figure 4 représente la racine carré de la pcrb de la distance radiale pour les deux types de distribution. Pour des scénarios de courte durée (inférieure à 2s ici : dépend du premier changement de porte), la pcrb est légèrement supérieure pour la distribution beta (environ 1 m). Néanmoins, pour les deux types de distributions, les pcrb convergent (ici vers 0.5 m pour une durée de 20s : environ 2500 mesures). Pour la vitesse radiale, la pcrb est indépendante du choix de la distribution effectué pour la distance (figure 5).

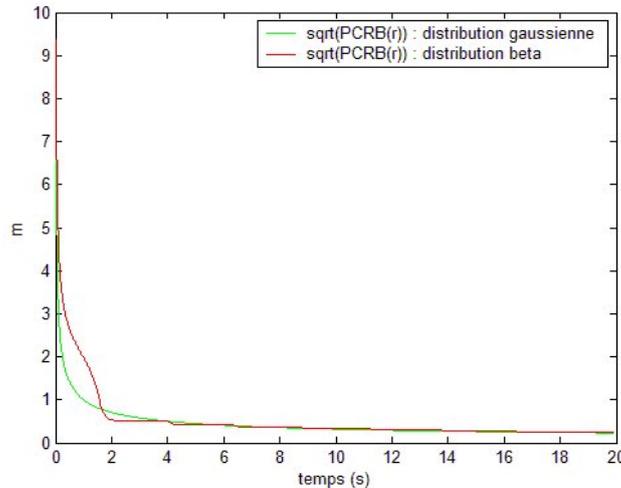


FIG. 4 – Borne De Cramer Rao de la distance radiale selon la distribution du bruit

Etant donné les résultats présentés ci-dessus, on pourra approximer la distribution du bruit sur l'observation de distance radiale par une gaussienne. La figure 6 propose une représentation de la gaussienne utilisée. Ces paramètres sont donnés par [7] :

$$E(r) = Pl(r) + \Delta P \frac{\lambda_1}{\lambda_1 + \lambda_2} \quad (78)$$

$$\sigma_r = \sqrt{\frac{\Delta P^2 \lambda_1 \lambda_2}{(\lambda_1 + \lambda_2)^2 (\lambda_1 + \lambda_2 + 1)}} \quad (79)$$

Si $\Delta P = 22.5$ et $\lambda_1 = \lambda_2 = 1.1$ alors $\sigma_r \approx 6.29$ m.

Le calcul de la pcrb sur r nous a permis de valider le fait que la distribution gaussienne peut être choisie pour caractériser le bruit sur l'observation de distance radiale. Dans la suite, cette approximation sera utilisée.

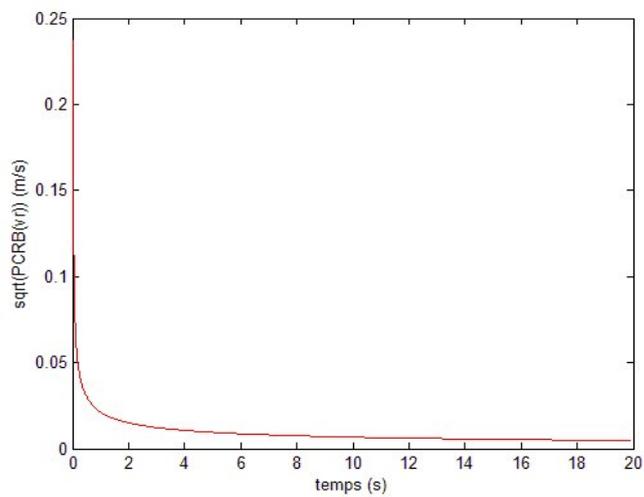


FIG. 5 – Borne De Cramer Rao de la vitesse radiale

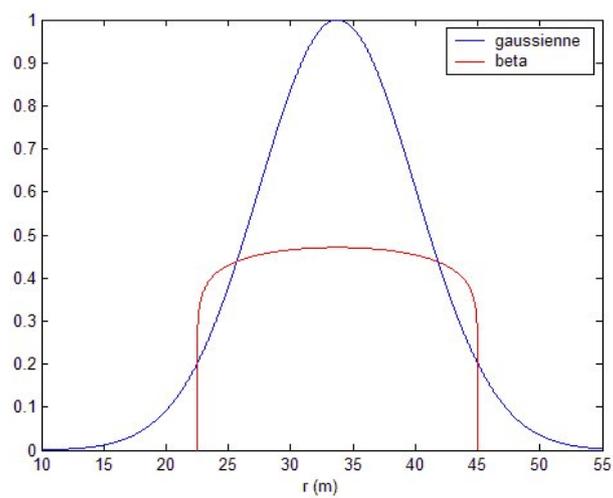


FIG. 6 – Distribution gaussienne et beta

2.3 PCRB dans le cas non linéaire

On propose ici de caractériser la borne dans le cas où on souhaite estimer le vecteur d'état défini par :

$$X_k = \begin{pmatrix} x_k \\ \dot{x}_k \\ y_k \\ \dot{y}_k \end{pmatrix} \quad (80)$$

à partir des observations :

$$Z_k = \begin{pmatrix} r_k \\ \dot{r}_k \\ \omega_k \end{pmatrix} \quad (81)$$

On rappelle que les composantes du vecteur X_k sont les caractéristiques cinématiques de la cible dans un repère cartésien. Les mesures sont les observations de distance et vitesse radiale et de vitesse angulaire du cap. Le modèle d'évolution est défini par

$$X_{k+1} = F X_k + v_k \quad (82)$$

où

$$F = \begin{pmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (83)$$

et $v_k \sim N(0, Q_k)$.

Le modèle non linéaire est :

$$Z_{k+1} = h_{1_{k+1}}(X_{k+1}) = h_{k+1}(X_{k+1}) + w_k \quad (84)$$

avec :

$$r_k = \sqrt{x_k^2 + y_k^2} \quad (85)$$

$$\dot{r}_k = h_{2_{k+1}}(X_{k+1}) = \frac{x_k \dot{x}_k + y_k \dot{y}_k}{\sqrt{x_k^2 + y_k^2}} \quad (86)$$

$$\omega_k = h_{3_{k+1}}(X_{k+1}) = \frac{x_k \dot{y}_k - \dot{x}_k y_k}{x_k^2 + y_k^2} \quad (87)$$

et $w_k \sim N(0, R_k)$. Les bruits v_k et w_k sont considérés indépendants.

2.3.1 Calcul de la borne

Le calcul de la borne est donné dans 2.1.2 puisque l'on travaille dans le cas gaussien. On a donc dans le cas où f_k est linéaire et h_k non linéaire :

$$J_{k+1} = D_k^{22} - D_k^{21} (J_k + D_k^{11})^{-1} D_k^{12} \quad (88)$$

avec

$$D_k^{11} = F_k^t Q_k^{-1} F_k \quad (89)$$

$$D_k^{12} = -F_k^t Q_k^{-1} = [D_k^{21}]^t \quad (90)$$

$$D_k^{22} = Q_k^{-1} + E\{\tilde{H}_{k+1}^t R_{k+1}^{-1} \tilde{H}_{k+1}\} \quad (91)$$

où

$$\tilde{H}_{k+1} = [\nabla_{X_{k+1}} h_{k+1}^t(X_{k+1})]^t = \begin{pmatrix} \frac{\partial h_1}{\partial x} & \frac{\partial h_1}{\partial \dot{x}} & \frac{\partial h_1}{\partial y} & \frac{\partial h_1}{\partial \dot{y}} \\ \frac{\partial h_2}{\partial x} & \frac{\partial h_2}{\partial \dot{x}} & \frac{\partial h_2}{\partial y} & \frac{\partial h_2}{\partial \dot{y}} \\ \frac{\partial h_3}{\partial x} & \frac{\partial h_3}{\partial \dot{x}} & \frac{\partial h_3}{\partial y} & \frac{\partial h_3}{\partial \dot{y}} \end{pmatrix}_{k+1} (X_{k+1}) \quad (92)$$

2.3.2 Simulation : scénario en virage

On propose ici d'évaluer la PCRb pour l'estimation des caractéristiques cinématiques de la cible dans un repère cartésien. Le scénario spécifié permet de préciser les performances optimales dans le cas du suivi d'un obstacle en virage, à partir d'observations de distance et vitesse radiale et vitesse angulaire de cap. La cible se déplace à la vitesse $v = 10 \text{ m/s}$ le long d'une clothoïde définie par le rayon de courbure R . Selon le bruit de mesure sur la vitesse angulaire, on calcule les pcrbs des composantes du vecteur d'état. Les figures 7, 8 décrivent respectivement ces résultats. La pcrb sur x est fortement dépendante de la précision que l'on a sur ω pour des scénarios d'une durée inférieure à 60 s. Pour un scénario d'une durée de 120 s, la pcrb converge vers la même valeur quel que soit la variance sur ω . En début de scénario, le mouvement étant longitudinal, la pcrb sur x augmente. Pour la distance et la vitesse suivant y , on s'aperçoit que les pcrbs sont identiques (à 10 cm près) et ont tendance à augmenter puisqu'en fin de scénario le mouvement du véhicule tend vers un mouvement plus transversal que longitudinal. L'initialisation de la matrice de Fisher J_0 , est donnée à partir de la matrice de covariance initiale P_0 :

$$J_0 = P_0^{-1} = \begin{pmatrix} 1.4 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0.25 \end{pmatrix}^{-1} \quad (93)$$

Ensuite, on analyse l'effet du rayon de courbure sur la pcrb. Les figures 9 et 10 montrent les pcrbs obtenues

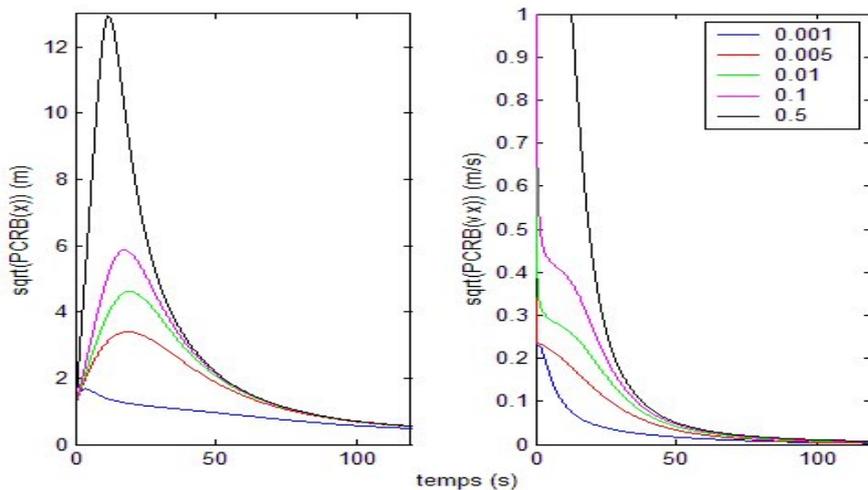


FIG. 7 – PCRb de x et v_x en fonction de σ_ω (rad/s)

pour différents R . Plus on tend vers un scénario en ligne droite, plus la performance sur l'estimation de x diminue. En effet, les mouvements selon l'axe x (transversaux) sont quasiment nuls pour une cible se déplaçant sur une courbe qui tend vers la ligne droite.

3 Gestion des pistes

3.1 Association des pistes

Dans la partie précédente, nous avons présenté deux types de filtres de poursuite qui permettent une estimation de l'état de l'obstacle à suivre. Ces filtres ont pour rôle de suivre une trajectoire étant données les observations qui lui sont associées. Dans de nombreuses applications radar, par exemple, les mesures ne sont pas seulement issues des cibles à suivre mais aussi du bruit et d'objets divers (barrières de sécurité, panneaux, terrain, ...). Il est donc nécessaire de sélectionner les mesures, si elles existent, correspondant le mieux à la piste. Les mesures non désirées sont généralement qualifiées de "clutter". De plus, dans un contexte multipiste, les mesures peuvent être issues de plusieurs cibles. Les cibles peuvent, à tout instant, rentrer et sortir de la zone de surveillance du capteur. Un capteur a rarement une probabilité de détection $P_D = 1$ et une probabilité de fausses alarmes $P_F = 0$. Dans ce contexte, de nombreux critères, pour prendre la décision d'associer des mesures ou une mesure à une piste, existent. Ces critères

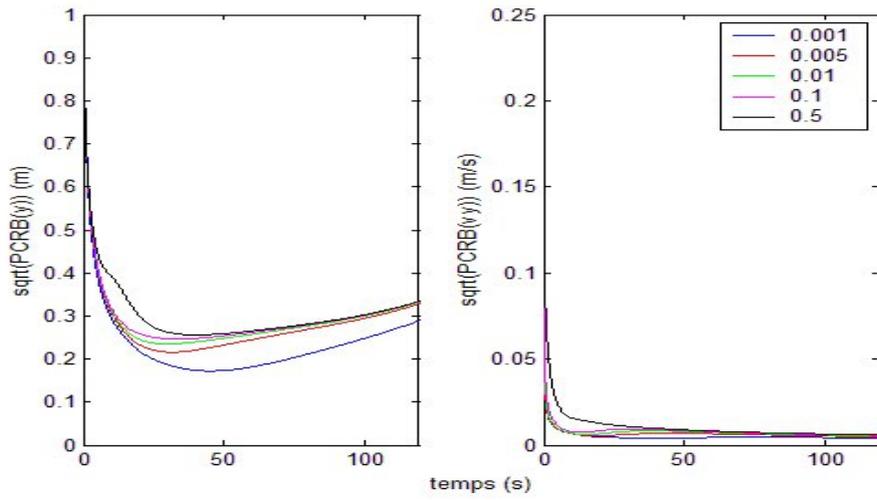


FIG. 8 – PCRB de y et v_y en fonction de σ_ω (rad/s)

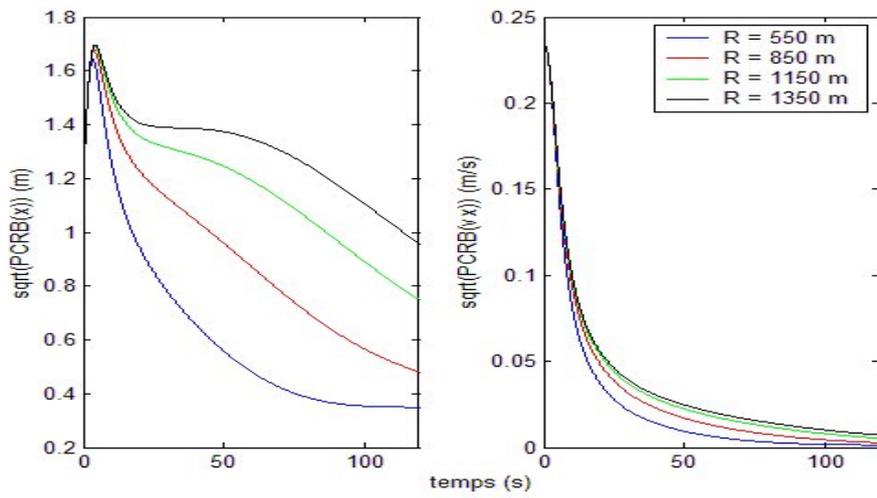


FIG. 9 – PCRB de x et v_x en fonction du rayon de courbure

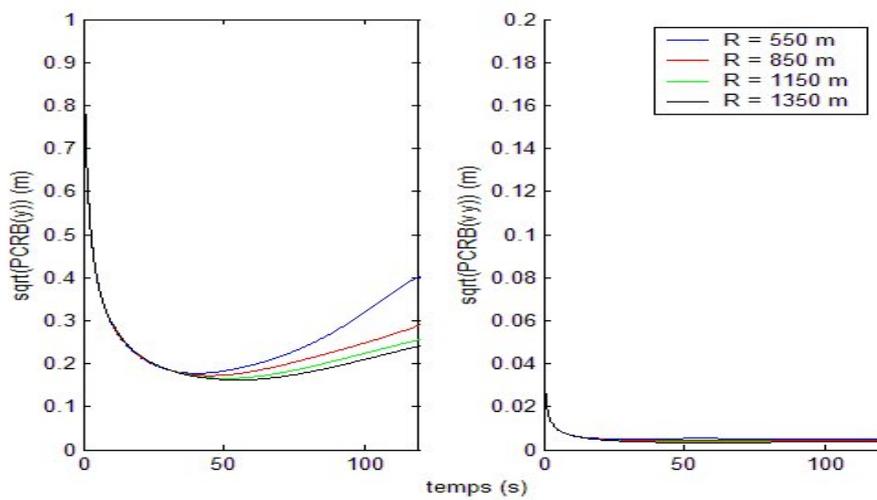


FIG. 10 – PCRB de y et v_y en fonction du rayon de courbure

dépendent essentiellement de la complexité de la tâche du pistage, comme par exemple le nombre de pistes, les manoeuvres effectuées par les obstacles, le nombre de fausses alarmes. Dans un contexte de pistage en présence de "clutter", une des premières phase essentielle pour l'association de donnée est le fenêtrage [3]. Le fenêtrage permet de ne pas prendre en considération les mesures qui sont loin de la mesure prédite et de sélectionner les mesures ayant la plus forte probabilité d'entretenir la piste. Après cette sélection de mesures, il est nécessaire de faire des choix quant aux associations éventuelles. De nombreuses méthodes existent dans la littérature. On pourra citer par exemple comme méthode d'association la méthode simple du plus proche voisin (PPV) [11] qui est une méthode à hypothèse unique [15]. Ce type de méthode sous-optimale, est adapté aux environnements contenant peu d'événements et connaît de faibles performances en présence de "clutter". Dans [3], les auteurs décrivent une méthode optimale, d'un point de vue bayésien, pour l'association de données. Cette méthode nécessite la connaissance de toutes les mesures de l'instant initial à l'instant courant et utilise le théorème des probabilités totales. Cette méthode ne peut donc pas être implémentée puisque le nombre d'hypothèses croient exponentiellement avec le temps.

D'autres méthodes sous-optimales, ont donc été proposées. Ces méthodes peuvent être classées en deux classes [24] : mono-scan et multi-scan. Les algorithmes mono-scan (PDAF, JPDAF [3]) délivrent un état estimé courant à partir d'un état précédent et d'hypothèses sur l'origine de la détection à l'instant courant. Au contraire les algorithmes multi-scan (Multiple Hypothesis Tracking [28], VDA [30]) construisent l'état estimé courant à partir d'hypothèses sur les mesures courantes et précédentes. Le nombre d'hypothèses à sauvegarder est fixé. Ces algorithmes sont généralement plus efficaces quand la probabilité de détection P_D est faible et que les "clutters" sont nombreux ou non homogènes.

Pour nos capteurs, dont les probabilités de fausses alarmes sont faibles, on utilisera essentiellement une méthode d'association basée sur la recherche du plus proche voisin.

3.1.1 Fenêtrage

Le but de cette sélection de mesure est d'intégrer au filtre les bonnes mesures et éviter les calculs inutiles en cas de présence de "clutters". Cette méthode élimine les mesures qui sont "loin" de la mesure prédite issue du filtrage. Le fenêtrage est effectué pour chaque piste et à chaque instant par la définition d'une zone de surveillance appelée fenêtre (ou région d'intérêt). Une seule ou l'ensemble des mesures présentes dans cette région est sélectionné pour l'association, les autres sont ignorées. La taille de la fenêtre influence le nombre de mesures validées. Si on choisit une fenêtre trop petite alors la probabilité d'éliminer la mesure provenant de la cible augmente, ce qui peut provoquer la perte de la piste. Au contraire, si la taille est trop grande, un grand nombre de mesures ne provenant pas de la piste est utilisé. Cela tend à augmenter les calculs et à détériorer la précision du pistage. Un résumé de diverses méthodes de fenêtrage est donné dans [6]. La plupart de ces méthodes utilise des styles de fenêtres ellipsoïdaux (voir

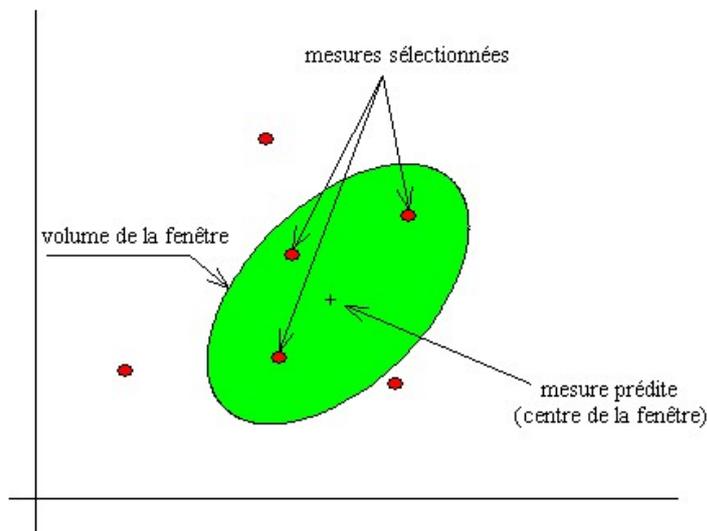


FIG. 11 – Une fenêtre ellipsoïdale

figure 11).

La fenêtre est formée de telle façon que la probabilité qu'une mesure, issue de la cible, soit dans la fenêtre de validation, à condition que la cible existe et soit détectée, est donnée par une probabilité P_G évaluée à

partir des caractéristiques statistiques de la piste. Puisque la taille ou le volume de la région dépendent de la précision du pistage, la fenêtre varie à chaque instant et pour chaque piste.

En supposant que la mesure provienne de la cible est un événement de densité de probabilité normale, centrée sur la mesure prédite ($\hat{Z}_k = H(\hat{X}_k)$) i.e. : $p(Z_k/Z_{1:k-1}) = N(\hat{Z}_k, S_k)$, la fenêtre de validation G est définie par :

$$G = \{Z/(Z - \hat{Z}_k)^t S_k^{-1} (Z - \hat{Z}_k) < \gamma\} \quad (94)$$

où S_k représente la covariance de l'innovation $\nu_k = Z - \hat{Z}_k$. Le volume de la fenêtre est donnée par [3] :

$$V_m = C_m |\gamma S_k|^{1/2} \quad (95)$$

où $|\gamma S_k|$ est le déterminant de γS_k et C_m vaut $\frac{\pi^{m/2}}{2^m}$ si m est pair et $\frac{2^{m+1}(\frac{m+1}{2})!}{(m+1)!} \pi^{\frac{m-1}{2}}$ si m est impair. Le seuil γ est un paramètre directement issu de la probabilité P_G . Il est issu des tables du χ^2 à m degrés de liberté avec un niveau de confiance de $N\%$.

Des techniques de fenêtrage, où les cibles ne suivent pas un modèle à vitesse constante ("maneuvering target"), sont détaillées dans [34]. Dans la plupart des méthodes de pistage, en présence de "clutter", le volume de la fenêtre est utilisé pour mesurer la densité de "clutter" qui est nécessaire pour la caractérisation des probabilités d'associations. Le calcul de ce volume, pour un pistage mono-cible considérant une approximation gaussienne (tel que le filtre à association probabiliste des données : PDAF), est donné par l'équation 95 [16]. Cependant, pour un pistage qui approxime la densité de probabilité a posteriori par une combinaison de gaussiennes ou par un ensemble de particules, le volume est l'union de toutes les régions d'intérêt. Ceci s'applique aussi pour le pistage multi-cible, puisque même si on utilise une approximation gaussienne (filtre à association conjointe des données : JPDAF), les fenêtres de validation de deux cibles proches se chevauchent. Le calcul du volume d'une région composée de plusieurs hyper-ellipses est donc nécessaire. Dans ce contexte, des méthodes de calcul de ce volume sont présentées dans [23]. Une méthode consiste à employer une simulation de Monte-Carlo qui génère des échantillons dans une région englobant la fenêtre et qui compte la proportion d'échantillons appartenant à la fenêtre. Une autre méthode [21] considère comme mesure de chevauchement le nombre de mesures partagées entre les différentes fenêtres. Après avoir sélectionner les mesures valides, il faut utiliser une méthode d'association afin de mettre à jour l'estimation de l'état. Ces méthodes sont diverses suivant le contexte du pistage. Quelques unes sont présentées dans la suite.

3.1.2 Méthode du plus proche voisin (PPV)

L'algorithme du plus proche voisin [3], noté PPV dans la suite, détermine, par l'utilisation de distances (traditionnellement euclidienne ou de Mahalanobis), quelle mesure est la plus proche de la prédiction et associe cette mesure à la piste. Cependant, puisque l'on n'associe qu'une seule mesure, il est parfois possible de ne pas associer la mesure issue de la cible dans différents cas. Le premier cas est celui où le capteur délivre des mesures non issues d'un obstacle (voir figure 12). Si une telle mesure appartient à la fenêtre de validation, rien ne dit qu'elle ne sera pas la plus proche.

De plus, si un capteur délivre plusieurs mesures pour une seule et même cible, on ne prendra pas en compte toutes les données de la cible, ce qui peut entraîner une perte de précision du système de pistage. Le choix du calcul de distance (euclidienne, Mahalanobis) peut conduire à des associations incorrectes. Dans le cas de la figure 13, la donnée sera associée à la piste de gauche en considérant une distance de Mahalanobis alors qu'elle est plus proche de celle de droite en considérant une métrique euclidienne. Au contraire, pour le cas de la figure 14, la donnée sera associée avec la piste de droite en considérant une distance euclidienne alors qu'elle est plus proche de celle de gauche en considérant une distance de Mahalanobis. Afin de palier ce problème, une nouvelle distance a été proposée dans [15] : c'est une pondération entre les distances euclidienne et de Mahalanobis, en fonction de la taille de la fenêtre de validation.

Dans le cas d'un suivi de deux cibles proches où les modules de pistage ne communiquent pas entre eux, cet algorithme peut conduire, dans des cas critiques en particulier le croisement de deux obstacles, à la perte de suivi d'une piste. Une étude réalisée par [26] montre que la méthode PPV a des performances acceptables pour un système de pistage présentant très peu de "clutters". Les auteurs proposent une comparaison des résultats respectifs aux méthodes d'association utilisées. Cette comparaison se base

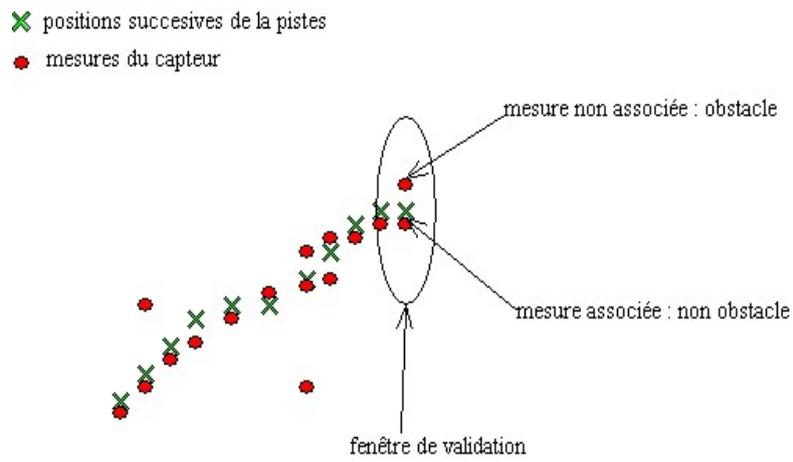


FIG. 12 – Cas 1 : association PPV

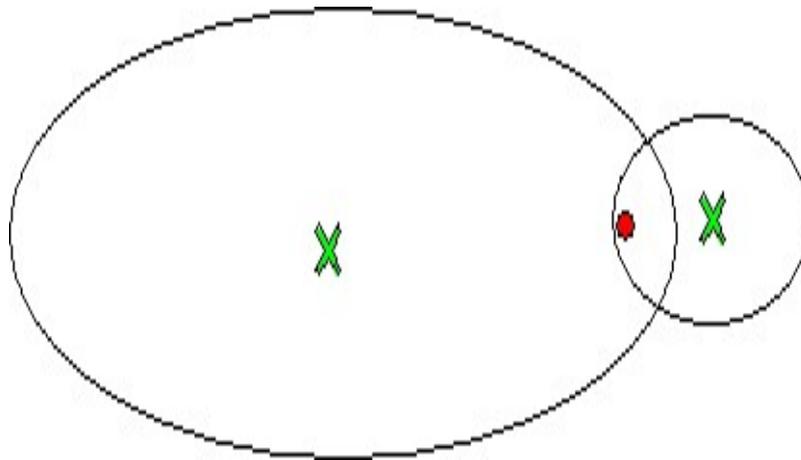


FIG. 13 – Cas 2 : association PPV

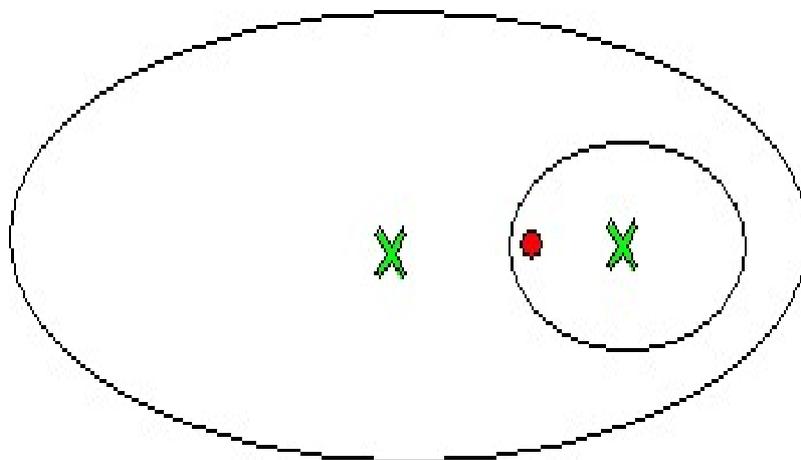


FIG. 14 – Cas 3 : association PPV

sur le temps de vie d'une piste, sur l'erreur moyenne de l'estimation de position et sur la complexité des calculs. Pour des niveaux de "clutter" faibles, les résultats prouvent que la méthode PPV propose un temps de vie supérieure aux autres méthodes, une erreur d'estimation de position équivalente aux autres méthodes, et une complexité moindre. Dans le cas d'une présence importante de "clutters" ces performances se dégradent puisqu'on n'utilise qu'une seule mesure de la fenêtre de validation.

3.2 Initialisation et mort des pistes : vraies ou fausses pistes

Ces briques d'un système de pistage dépendent fortement du contexte et des capteurs. En effet, les systèmes d'association de données évoluent dans des situations où les mesures ne sont pas certaines. On sait que dans de nombreuses applications, notamment radar, les mesures peuvent provenir de cibles ou de fausses détections (divers objets, terrain, bruit thermique,...). Dans des contextes multicibles, les mesures peuvent provenir d'une cible ou d'une autre. Dans de telles conditions, un système d'initialisation des pistes crée des vraies pistes (à partir des mesures d'une cible) et des fausses pistes (à partir des fausses mesures). Durant l'opération de maintenance des pistes, une vraie piste peut devenir fausse si la détection n'a pas eu lieu, ou si le filtre utilisé ne modélise pas les manoeuvres effectuées par la piste. De façon équivalente, une fausse piste peut devenir une vraie piste si les mesures issues d'une cible sont utilisées pour la mise à jour. Il est donc nécessaire de distinguer les vraies des fausses pistes. Quand une piste est considérée comme vraie, elle est intégrée au traitement suivant. Quand une piste est considérée comme fausse, on dit qu'elle est morte ou terminée. Pour distinguer un fausse piste d'une vraie, les systèmes de pistage utilisent en général une mesure de qualité pour la piste considérée. Les divers algorithmes de pistage utilisent des noms différents pour cette mesure de qualité. On parlera d'une fonction de score de piste ("track score function") pour le MHT (Multiple Hypothesis Tracking)[6], d'une probabilité d'existence de piste pour le IPDA (Integrated Probability Data Association) [22] ou de probabilité de détection de piste pour le IMM-PDA (Integrated Multiple Model Probability Data Association)[2]. La procédure usuelle est de considérer une piste comme vraie si sa qualité est au-dessus d'un seuil : on parlera de seuil de confirmation. Elle considère, de plus, qu'une piste est fausse si sa qualité est inférieure à un seuil : on parlera de seuil de terminaison. Dans de nombreux cas, ces différents seuils sont constants. Cependant, dans [4], les auteurs proposent des seuils qui évoluent avec l'âge de la piste. L'initialisation et la terminaison des pistes sont donc basées sur une mesure de qualité. Cette mesure dépend fortement du contexte et sera décrite dans nos applications suivant les capteurs utilisés.

4 Application : simulation de suivi d'obstacles par Lidar et Radar

Pour les deux types de capteur, on choisit de construire un composant fonctionnel pour chaque tâche différente du pistage. On aura donc une brique pour la "vie des pistes", une brique pour l'association de données et une brique pour l'estimation. On traite ici les résultats sur l'estimation des caractéristiques cinématiques de l'obstacle. On suppose que l'initialisation et que l'association de données sont réalisées. On admet donc que les observations sont issues d'une seule et même cible et que la probabilité de non détection des capteurs est nulle. On aborde ici deux cas de scénario : la ligne droite et le virage. On supposera, parfois, que l'on a accès à la courbure de la route à l'endroit considéré ; en particulier dans le cas de l'estimation à partir des observations radar seules.

4.1 Estimation par lidar

On rappelle que le lidar délivre des observations dans un repère cartésien. Le vecteur de mesure est donc :

$$Z = \begin{pmatrix} x \\ y \end{pmatrix} \quad (96)$$

Le vecteur d'état est définie par :

$$X = \begin{pmatrix} x \\ \dot{x} \\ y \\ \dot{y} \end{pmatrix} \quad (97)$$

pour un modèle d'évolution à vitesse constante. Pour un modèle à accélération constante on inclut l'accélération longitudinale et transversale au vecteur d'état. Les deux modèles d'évolution utilisés sont donc :

- un modèle à vitesse constante :

$$X_{k+1} = F_{VC}X_k + G_{VC}V_k \quad (98)$$

où

$$F_{VC} = \begin{pmatrix} 1 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (99)$$

$$G_{VC} = \begin{pmatrix} \frac{\Delta t^2}{2} & 0 \\ \Delta t & 0 \\ 0 & \frac{\Delta t^2}{2} \\ 0 & \Delta t \end{pmatrix} \quad (100)$$

Le bruit sur le modèle est considéré ici comme étant l'accélération. Il est caractérisé par sa matrice de covariance Q .

- un modèle à accélération constante :

$$X_{k+1} = F_{AC}X_k + G_{AC}V_k \quad (101)$$

où

$$F_{AC} = \begin{pmatrix} 1 & \Delta t & \frac{\Delta t^2}{2} & 0 & 0 & 0 \\ 0 & 1 & \Delta t & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \Delta t & \frac{\Delta t^2}{2} \\ 0 & 0 & 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (102)$$

$$G_{AC} = \begin{pmatrix} \frac{\Delta t^3}{6} & 0 \\ \frac{\Delta t^2}{2} & 0 \\ \Delta t & 0 \\ 0 & \frac{\Delta t^3}{6} \\ 0 & \frac{\Delta t^2}{2} \\ 0 & \Delta t \end{pmatrix} \quad (103)$$

Le bruit sur le modèle est considéré ici comme étant le jerk (dérivée de l'accélération). Il est caractérisé par sa matrice de covariance Q .

4.1.1 scénario ligne droite

On suppose ici qu'un cible s'éloigne à la vitesse $v = 5 \text{ m/s}$ de l'observateur. Sa trajectoire suit une ligne droite. On génère des observations toutes les $\Delta t_{lidar} = 600 \text{ m/s}$. Les mesures sont calculées à partir de la vraie valeur en affectant du bruit à celle ci selon les caractéristiques du capteur. La figure 15 illustre les mesures générées et la vraie trajectoire. Les figures 16 17 représentent les erreurs au sens des moindres carrés sur l'estimation des composantes cinématiques. Au vu de ces figures, les deux modèles sont équivalents pour l'estimation de la position. Néanmoins, le modèle à accélération constante génère une erreur moindre pour l'estimation des vitesses. La précision sur les vitesses étant faible, quelque soit le modèle, le choix du modèle sera guidé par l'application considérée.

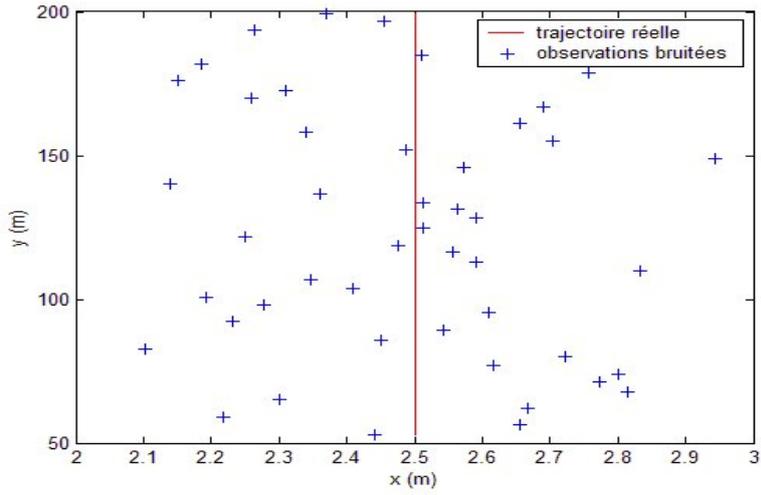


FIG. 15 – Description de la trajectoire et des observations bruitées

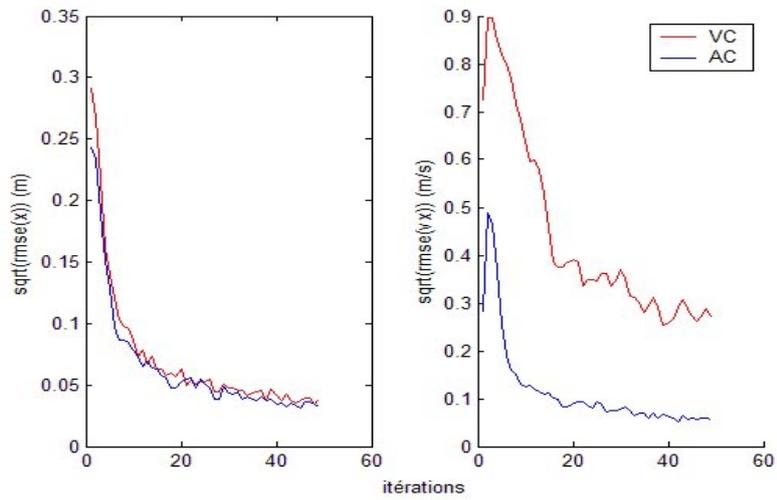


FIG. 16 – Erreur sur x et \dot{x} pour les deux modèles

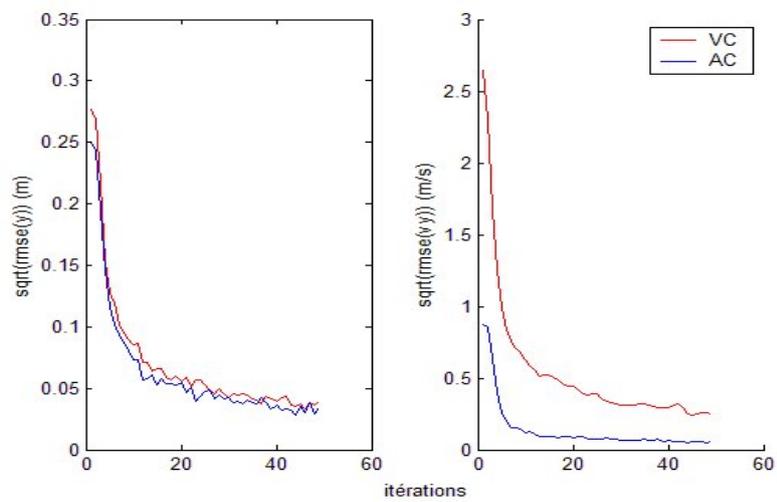


FIG. 17 – Erreur sur y et \dot{y} pour les deux modèles

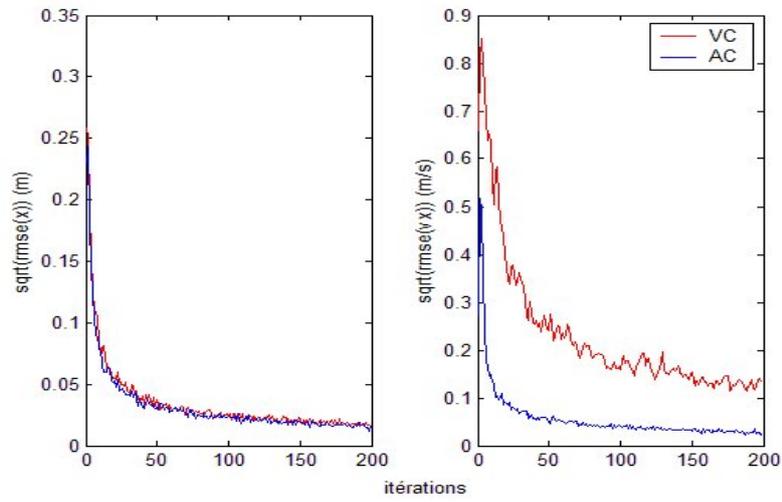


FIG. 18 – Erreur sur x et \dot{x} pour les deux modèles

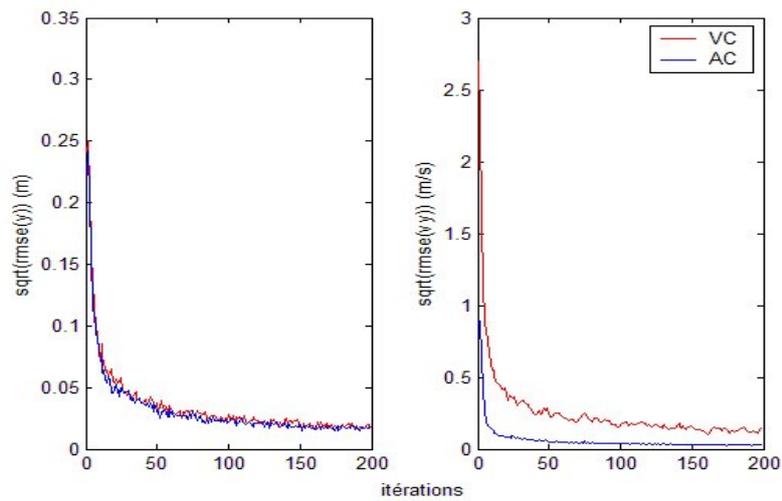


FIG. 19 – Erreur sur y et \dot{y} pour les deux modèles

4.1.2 scénario en virage

Les figures 18 et 19 renseignent sur la rmse sur les différentes composantes du vecteur d'état. De manière identique au scénario en ligne droite, on peut dire que le choix du modèle est pratiquement sans influence sur l'erreur de position. Cependant, la précision sur la vitesse étant moindre pour le modèle à accélération constante pour un scénario en virage, on utilisera ce type de modèle.

4.2 Estimation par radar

On rappelle que le radar délivre des observations radiales toutes les 8 *ms*. La mesure de distance est caractérisée par une observation de porte distance et la mesure de vitesse par un indice vitesse. Le vecteur de mesures est :

$$z = \begin{pmatrix} r \\ \dot{r} \end{pmatrix} \quad (104)$$

La définition du modèle d'évolution est dépendante de l'application à réaliser. En effet, si le scénario est un scénario en ligne droite, l'estimation de r et \dot{r} pourra être suffisante dans un contexte d'anti-collision. Par contre pour la gestion des interdistances, en particulier pour le suivi en virage, l'estimation des positions et des vitesses cartésiennes sera nécessaire.

4.2.1 Estimation linéaire à partir des observations de distance et vitesse radiale

On utilise ici un modèle d'évolution à vitesse constante dont le vecteur d'état est équivalent au vecteur de mesures. Le problème est donc linéaire, on exploitera donc un filtre de Kalman. La figure 20 décrit la définition du scénario et en particulier les observations effectuées (mesure de porte distance). La figure

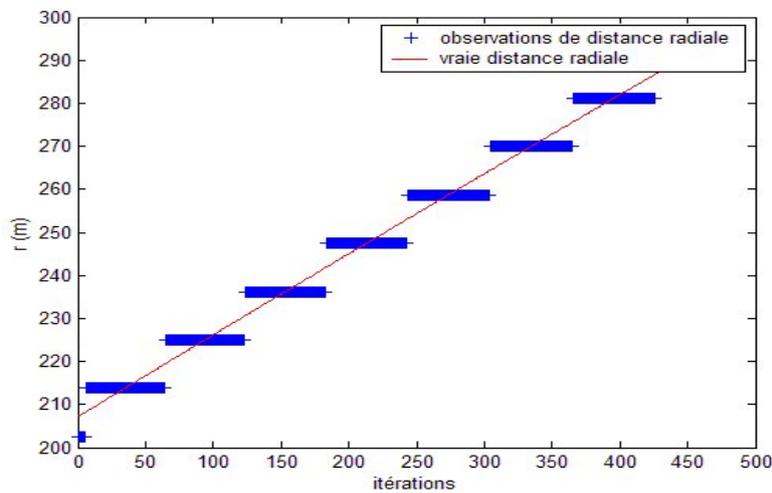


FIG. 20 – Scénario pour la distance radiale

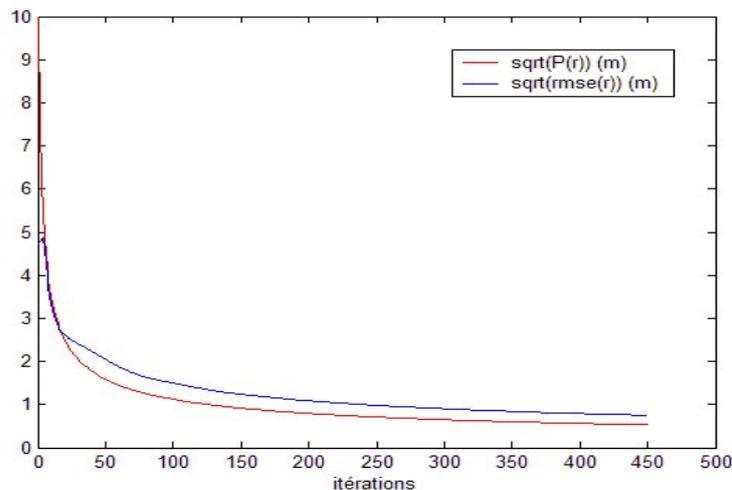


FIG. 21 – Erreur sur r : covariance kalman et rmse

21 renseigne sur la rmse de r . On remarque que l'on est capable d'estimer la distance radiale avec environ 1 m de précision pour un scénario de 5 s environ.

Références

- [1] M.S. Arulampalam, S. Mksell, N.J. Gordon, and T. Clapp. A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking. *Special issue on Monte Carlo Methods for Statistical Signal Processing, IEEE Transactions on signal Processing*, SP-50(2), 2002.
- [2] Y. Bar-Shalom and K.C. Chang. Joint probabilistic data association for multiple target tracking with possibly unresolved measurements and manoeuvres. *IEEE Trans. on Automatic Control*, vol. 29 :p. 585–594, july 1994.
- [3] Y. Bar-Shalom and T.E. Fortmann. *Tracking and data association*, volume 179. Mathematics in science and engineering ISBN 0-12-079760-7, 1988.
- [4] Y. Bar-Shalom and X.R. Li. *Estimation and Tracking : Principles, Techniques and Software*. MA : Artech House, 1993.
- [5] Y. Bar-Shalom, X.R. Li, and T. Kirubarajan. Estimation with applications to tracking and navigation. In *New York : John Wiley and Sons*. 2001.
- [6] S. Blackman and R. Popoli. *Design and Analysis of Modern Tracking Systems*. Boston, MA : Artech House, 1999.
- [7] K. Bury. Statistical distributions in engineering. In *Cambridge University Press*. 1999.
- [8] A. Doucet. *Monte Carlo Methods for Bayesian Estimation of Hidden Markov Models. Application to Radiation Signals*. PhD thesis, Université Paris-Sud, Orsay (France), 1997.
- [9] A. Doucet. On sequential monte carlo methods for bayesian filtering. Rapport technique, université de cambridge, département d'engineering, 1998.
- [10] A. Doucet, N. de Freitas, and N.J. Gordon. *Sequential Monte Carlo Methods in Practice*. Number ISBN 0-387-95146-6. Statistics for Engineering and Information Science, Springer, 2001.
- [11] A. Farina and S. Pardini. Track while scan algorithm in a clutter environment. *IEEE Trans. on AES*, pages p. 769–779, septembre 1978.
- [12] W.R. Gilks and C. Berzuini. Following a moving target-monte carlo inference for dynamic bayesian models. *Journal of the Royal Statistical Society, B*, 63 :p. 127–146, 2001.
- [13] N.J. Gordon, D.J. Salmond, and A.F.M. Smith. Novel approach to nonlinear-non-gaussian bayesian state estimation. *IEE Proc.-F*, vol. 140, no. 2 :p. 107–113, 1993.
- [14] J.D. Hol. Resampling in particle filters. Rapport technique, université de linköping, suède, department of electrical engineering, 2004.
- [15] D. Hutber. *Suivi multi-capteurs de cibles multiples en vision par ordinateur, appliqué à un véhicule dans un environnement routier*. PhD thesis, Université de Nice, Sophia-Antipolis (France), 1995.
- [16] S. Jouannin. *Association et fusion de données : Application au suivi et à la localisation d'obstacles par radar à bord d'un véhicule routier intelligent*. PhD thesis, Université BLAISE-PASCAL, Clermont-Ferrand (France), janvier 1999.
- [17] S. J. Julier and J. K. Uhlmann. A new extension of the kalman filter to nonlinear systems. *Proc. of AeroSense : The 11th Int. Symp. on Aerospace/Defence Sensing, Simulation and Controls*, 1997.
- [18] R.E. Kalman. A new approach to linear filtering and prediction problems. *Trans. ASME, Journal of Basic Engineering*, vol. 82 :p. 35–45, mars 1960.
- [19] F. LeGland. Filtrage particulaire. Mathstic chaînes de markov cachées et filtrage particulaire, méthodes particulières (as 67) du département stic, 2002.
- [20] J.S. Liu and R. Chen. Sequential monte carlo methods for dynamic systems. *Journal of the American Statistical Association*, 93 (443) :p. 1032–1044, 1998.
- [21] D. Musicki and R.J. Evans. Joint integrated probabilistic data association. In *ISIF 2002*, Annapolis, Maryland (E-U), 2002.
- [22] D. Musicki, R.J. Evans, and S. Stankovic. Integrated probabilistic data association. *IEEE Trans. on Automatic Control*, vol. 39(6) :p. 1237–1241, 1994.

- [23] D. Musicki and M.R. Morelande. Gate volume estimation for target tracking. In *ISIF 2004*, Stockholm (Suède), June 28–1st july 2004.
- [24] D. Musicki, B.F. La Scala, and R.J. Evans. Integrated track splitting filter for manoeuvring targets. In *ISIF 2004*, Stockholm (Suède), June 28–1st july 2004.
- [25] C. Musso, N. Oudjane, and F. LeGland. Improving regularised particle filters. *Sequential Monte Carlo Methods in Practice*, 2001.
- [26] L.Y. Pao and R.M. Powers. A comparison of several different approaches for target tracking with clutter. *American Control Conference*, pages p. 3919–3924, june 2003.
- [27] A. Papoulis. Probability, random variables, and stochastic processes, 2nd ed. In *New York : McGraw-Hill*. 1984.
- [28] D.B. Reid. An algorithm for tracking multiple targets. *IEEE Trans. on Automatic Control*, vol. 24 :p. 843–854, décembre 1979.
- [29] B. Ristic, M.S. Arulampalam, and N.J. Gordon. *Beyond the Kalman Filter-Particle Filters for Tracking Applications*. Number ISBN 1-58053-631-X. Artech-House, 2004.
- [30] B.F. La Scala and G.W. Pulford. A viterbi algorithm for data association. *35th Conf. on Decision and Control*, décembre 1996.
- [31] P. Tichavsky, C. Muravchik, and A. Nehorai. Posterior cramer-rao bounds for discrete-time nonlinear filtering. *IEEE Trans. Signal Processing*, vol. 46(5) :p. 1386–1396, mai 1998.
- [32] H.L. Van Trees. Detection, estimation, and modulation theory (part i). In *John Wiley and Sons*. 1968.
- [33] E.A. Wan and R. van der Merwe. The unscented kalman filter for nonlinear estimation. In *IEEE Symp. Adaptive Systems for Signal Processing, Comm. and control (AS-SPCC)*, pages p. 153–158, Lake Louise (Canada), 2000.
- [34] X. Wang, S. Challa, and R.J. Evans. Gating techniques for manoeuvring target tracking in clutter. *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38 (3) :p. 1087–1097, july 2002.
- [35] E.W. Weisstein. Chapman-kolmogorov equation. In *Mathworld : A Wolfram Web Ressource*. <http://mathworld.wolfram.com/Chapman-KolmogorovEquation.html>.