

## Applications d'estimation récursive

### I. Estimation d'attitude (filtre de Kalman)

Dans cette première partie, nous étudions l'implémentation d'un filtre de Kalman afin d'estimer l'attitude d'une IMU.

L'objectif est d'utiliser l'accéléromètre embarqué sur l'IMU afin d'estimer l'attitude du système en temps réel.

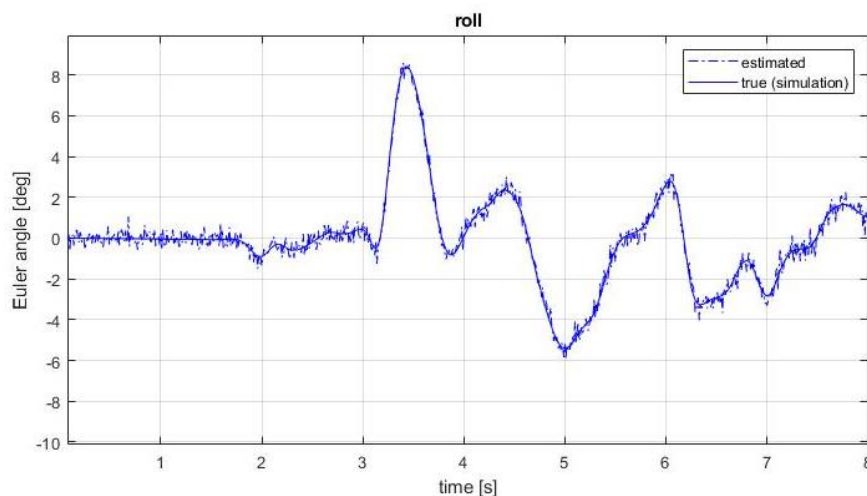
Dans un premier temps, on met en place un filtre de Kalman pour estimer seulement l'angle de roulis  $\theta_x$ . Notre étude se limite donc alors à une étude dans le plan Y, Z. On étend ensuite le principe pour estimer l'attitude de l'objet en trois dimensions.



#### 1. Utilisation du gyromètre

L'IMU dispose, en plus de son accéléromètre, d'un gyromètre qui peut être utilisé pour améliorer la qualité de l'estimation. Le filtre de Kalman peut donc être implémenté de trois manières différentes.

La première consiste à ne pas utiliser le gyromètre. Cette méthode plus générale peut être appliquée lorsque seule l'information de l'accéléromètre est disponible. Elle est la moins performante des trois méthodes étudiées mais est la plus simple à implémenter. De plus, on constate qu'elle donne des résultats tout à fait satisfaisants et peut donc être suffisante pour certaines applications. En effet, en observant le graph ci-dessous, on constate que l'angle de roulis est estimé avec une erreur inférieure à  $1^\circ$ .



La deuxième prend en compte les mesures du gyromètre comme une entrée de contrôle. On suppose qu'il n'y a pas d'erreur dans les mesures fournies par le gyromètre et on les utilise pour améliorer l'estimation de l'attitude à chaque itération. En pratique, là où l'information fournie par le capteur est forcément bruitée, cette approche peut être adoptée lorsque la variance du bruit du gyromètre est beaucoup plus faible que la variance du bruit de l'accéléromètre (i.e. le gyromètre est beaucoup plus fiable que l'accéléromètre).

La troisième consiste à ajouter la vitesse dans le vecteur d'état du filtre de Kalman. Ce cas est à privilégier si les informations transmises par le gyromètre sont peu fiables. Cependant, cette approche augmente la complexité des calculs, ce qui peut être limitant dans certaines applications.

## 2. Réglage des gains Q et R

Le gain Q permet de régler le bruit d'état. L'augmenter permet d'accorder plus d'importance aux mesures qu'au modèle d'évolution des grandeurs. Ceci permet de limiter les risques de divergence liés à une mauvaise modélisation de la dynamique du modèle.

Le gain R modélise les bruits de mesures et doit normalement découler de l'analyse des capteurs utilisés. Cependant on peut artificiellement augmenter ce gain lorsque les hypothèses de bruit blanc Gaussien ne sont pas respectées pour le capteur.

## 3. Discussions

Pour l'utilisation dans le cadre d'un avion, souvent soumis à des accélérations constantes, le filtre développé précédemment ne conviendrait pas. En effet celui-ci est conçu pour fonctionner dans le champ de gravité terrestre c'est-à-dire avec une accélération constante vers le haut. Toute autre composante de vitesse doit pouvoir être modélisée comme un bruit blanc Gaussien. Or, une accélération constante non nulle ne respecterait pas cette hypothèse. Pour adapter le filtre, il faudrait ajouter les accélérations dans le vecteur d'état du filtre de Kalman afin de pouvoir les estimer et suivre leurs variations. Cependant, cette méthode augmente la quantité de calculs à réaliser à chaque itération.

Dans le cadre d'une utilisation dans un contexte spécifique, on peut extraire des informations supplémentaires afin de raffiner le modèle d'évolution ainsi que les variances de bruits. Par exemple, on peut réduire les variances de bruits sur les mesures d'accélérations si le capteur est tenu à la main par l'opérateur. En effet, on sait qu'un humain qui marche ne peut atteindre qu'un domaine très limité d'accélération. On peut également utiliser ces informations supplémentaires pour élaborer un modèle d'évolution des variables d'états plus représentatif et ainsi s'affranchir du modèle fixe (où les variables restent, à priori, constantes) que l'on utilise par défaut.

S'il le biais du gyromètre n'est pas constant et qu'il doit être estimé, il faut modifier l'algorithme pour intégrer le biais dans les états du filtre de Kalman. Comme on ne peut pas prédire à priori l'évolution de ce biais, on prend un modèle d'évolution constante, on a donc dans le cas 2D :

Variables d'états	Prédiction	Mise à jour
$X = \begin{bmatrix} \theta_x \\ \dot{\theta}_x \\ b_{gyro} \end{bmatrix}$ <p>Avec <math>b_{gyro}</math> le biais du gyromètre à estimer.</p>	$X_{k+1} = \begin{bmatrix} 1 & dt & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} X_k$	$\hat{Y} = \begin{bmatrix} a_y \\ a_z \\ \dot{\theta}_{gyro} \end{bmatrix} = \begin{bmatrix} \sin(X(1)) \\ \cos(X(1)) \\ X(2) + X(3) \end{bmatrix}$ <p>Avec <math>a_y</math> et <math>a_z</math> les accélérations estimées sur chaque axe et <math>\dot{\theta}_{gyro}</math> la prédiction de mesure de vitesse du gyromètre.</p>

Pour choisir la variance du biais du gyromètre, on peut prendre la valeur maximale de biais qui serait admissible dans notre cas d'application.

## II. Estimation de position (filtrage particulaire)

Dans la deuxième partie, nous étudions un cas d'application de filtrage particulaire pour de la localisation sous-marine à partir d'une mesure de profondeur et d'informations toponymiques connues.

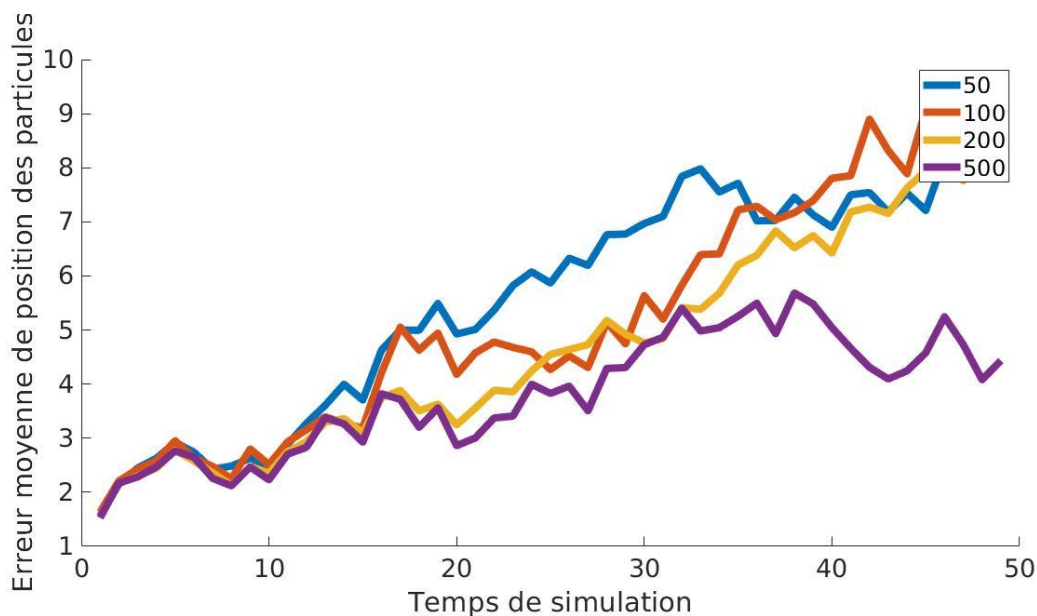
### 1. Importance du resampling

L'algorithme standard sans resampling ne fonctionne pas : il y a dégénérescence de la population de particule. Au bout de quelques itérations, seul un nombre très limité de particule ont un poids non négligeable. Ceci engendre des erreurs numériques de calculs qui empêchent le bon fonctionnement de l'algorithme. Afin de corriger ce problème, une phase de resampling est introduite. On élimine les particules les moins probables et on duplique celle ayant une probabilité plus importante. Ainsi on évite les problèmes de dégénérescence massive qui mènent à des erreurs numériques. Cette amélioration suffit pour produire des résultats satisfaisants. En effet, on constate qu'en prenant des bruits de mesures et des bruits d'évolution égaux aux bruits réels, on obtient pour une centaine de particules, d'ores et déjà des résultats probants. L'erreur entre la moyenne des particules et la position réelle du sous-marin reste contenue.

### 2. Nombre de particules

Afin de fixer le nombre de particules utilisées dans l'algorithme, il prend en compte les contraintes liées aux capacités de calculs, les performances exigées ainsi que d'autres paramètres évoqués ci-après (notamment la valeur des bruits considérés dans le modèle du filtre particulaire).

On peut lire sur le graphique ci-dessous l'évolution de l'erreur entre la position du sous-marin et la moyenne des particules en fonction du temps de simulation en abscisse et pour différentes tailles de population. Les bruits de mesures et d'évolution considérés dans le filtre sont pris égaux aux bruits réels.



Ces courbes ont été obtenues à l'aide 50 simulations pour chaque configuration. Même si ce nombre est relativement faible pour une approche de Monte-Carlo rigoureuse, il permet toutefois d'établir les tendances d'évolution tout en gardant des temps de calculs raisonnables.

La partie la plus sensible à l'augmentation de la taille de la population des particules est le resampling que ce soit au niveau de l'augmentation de la complexité algorithmique que de la pertinence des

estimations. En effet, le resampling fait appel à des fonctions de tris sur la population des particules ce qui induit une complexité a minima de  $n.\log(n)$ . En outre, une plus grande population permet de conserver plus de diversités dans les particules, c'est-à-dire une plage d'exploration plus important et donc plus de chances de réussir à lever des ambiguïtés ou de faire revenir l'erreur à zéro.

### 3. Bruits de mesures et d'évolution

Utiliser des bruits de mesures ou d'évolution supérieurs aux bruits réels permet de limiter l'impact d'incertitudes ou d'erreurs concernant le modèle réel. De plus, l'augmentation de ces bruits permet de conserver une plage d'exploration plus importante pour la population de particules (ce que l'on pourrait appeler de la diversité génétique pour utiliser la terminologie propre aux algorithmiques génétiques). Dans les deux cas, le filtre particulaire a moins de chance de se retrouver bloqué sur une fausse trajectoire (par exemple dans le cas de deux vallées parallèles pour notre relief sous-marin considéré).

Pour éviter le problème de dérive de l'estimation conduisant à l'impossibilité de retrouver la trajectoire du sous-marin, on peut, à l'image des algorithmes génétiques, introduire des mutations dans la population des particules. Plus précisément, il s'agit à chaque itération d'ajouter à la population, un certain nombre de particules générées aléatoirement autour de la dernière estimation avec un écart-type relativement important permettant, le cas échéant de retrouver la trajectoire perdue. Lors de cette opération, il faut veiller à conserver une taille de population stable.

### 4. Influence du point de départ

Pour des réglages fixés du filtre particulaire, on constate que le point de départ ainsi que la trajectoire suivie par le sous-marin ont un impact non négligeable sur les performances de l'algorithme. En effet, si le sous-marin suit une unique vallée profonde, l'algorithme donnera une estimation précise de la position. Cependant, si à un moment donné, le chemin du sous-marin ce se sépare en deux vallées distinctes, l'algorithme pourra être mis en difficultés par l'ambiguïté à lever. Enfin dans le cas d'une topologie relativement uniforme et de faible variation, la dispersion des particules sera plus importante et l'erreur moyenne de l'estimation sera élevée. Le filtrage particulaire sera donc beaucoup moins performant si le sous-marin se déplace au-dessus d'un plateau que dans une vallée.