Filtre de Kalman

Part 1 : Filtre de Kalman dans un cas théorique simple

Exercice On cherche à estimer un vecteur $\mathbf{x} = (x_1, x_2)$ qui vérifie les équations d'état suivantes

$$y(k) = \mathbf{C}(k)\,\mathbf{x} + \beta(k),$$

avec

k	0	1	2	3	4
$\mathbf{C}(k)$	$(4\ ,\ 0)$	$(10\ ,\ 1)$	$(10\ ,\ 5)$	$(13\ ,\ 5)$	$(15\ ,\ 3)$
$y(k)$	5	10	11	14	17

On suppose que la variance pour l'erreur de mesure β vaut 9 et que $x_1 \simeq 1$ et $x_2 \simeq -1$ avec une variance de 4. En utilisant le filtre de Kalman, calculer une estimation pour $\mathbf{x}$ et donner la matrice de covariance associée à cette estimation.

Remarque : on pourra s'aider des scripts (notamment kalman.m, draw_ellipse.m) disponibles sur http://www.ensta-bretagne.fr/lebars/Share/kalman_files_matlab.zip.

Part 2 : Filtre de Kalman pour l'estimation de la position du buggy simulé

Grâce au filtre de Kalman, on peut estimer notre position au mieux tout en dessinant les ellipsoïdes de confiance pour avoir une idée de notre erreur d'estimation…

Dans un premier temps, on supposera que le buggy est en intérieur et donc que le GPS n'est pas disponible. La boussole du buggy sera considérée comme très précise donc theta pourra être considéré comme connu, au même titre que les entrées u1 et u2. Par contre, il nous faut estimer la position x,y.

En repartant du simulateur de buggy (http://www.ensta-bretagne.fr/lebars/Share/buggy_simu.zip), appliquer la même démarche que dans l'exercice du tricycle du TD précédent pour le buggy simulé. Pour y arriver, il faut se poser les questions suivantes :

1. Réécrire les équations du buggy pour qu'elles deviennent compatibles avec les hypothèses du filtre de Kalman.

2. La position initiale sera considérée comme assez bien connue car en général, l'opérateur sait assez précisément où il a démarré son robot. Rajouter le code nécessaire pour indiquer que l'on connait la position initiale du buggy avec une variance de 0.1 (environ 0.5 m d'erreur avec une probabilité de 99%).

3. Dans le scénario actuel, nous n'avons pas de mesures directes de x,y, donc nous n'avons pas de bruit de mesure non plus. Cependant, notre modèle étant forcément imprécis, nous avons du bruit d'état : on peut considérer par exemple qu'il est de variance 0.001 sur x et y (à ajuster en fonction des tailles des ellipses qui seront dessinées par la suite). En fonction de toutes ces informations, utiliser la fonction

kalman pour estimer à chaque instant la position du buggy avec sa matrice de covariance associée.

4. Dessiner les ellipsoïdes de confiance contenant la position du buggy avec une probabilité de 99%.

5. Modifier ensuite le code pour rajouter la prise en compte du GPS (de la même manière que pour les odomètres dans l'exercice du tricycle du TD précédent). Pour un GPS à bas coût, on peut considérer que le bruit de mesure est avec une variance de 2 (environ 5 m d'erreur avec une probabilité de 99%).