ÉTUDE SIMULÉE DE LA LOCALISATION D'UN ROBOT IMMOBILE DANS UNE CARTE

Cette manipulation de Travaux Pratiques a pour but d'illustrer des notions et techniques élémentaires d'estimation stochastique sur l'étude simulée d'un problème de localisation 2D statique en robotique.

Dans l'ensemble du sujet, $\mathbb{I}$, 0 et $\mathbb{O}$ désignent respectivement la matrice identité, tout vecteur constitué de zéros et toute matrice constituée de zéros. Leurs dimensions ne sont pas explicitées lorsqu'elles découlent du contexte. La notation $\text{blkdiag}(A_1, \ldots, A_N)$ désigne la matrice bloc-diagonale constituée de l'assemblage de $A_1, \ldots, A_N$. L'opérateur de transposition est noté ^{T}. Enfin, $\mathsf{x} \sim \mathcal{G}(\bar{x}, P)$ signifie que la variable aléatoire x suit la loi normale dont la moyenne est le vecteur $\bar{x}$ et dont la matrice de covariance est la matrice définie positive P.

POSITION DU PROBLÈME

On considère un monde plan. Celui-ci est muni du repère de référence $\mathcal{F}_0 = (O, \vec{x}_0, \vec{y}_0)$, dont les axes $\vec{x}_0, \vec{y}_0$ sont respectivement orientés vers l'Est et vers le Nord. Une carte est disponible, constituée d'amers ponctuels $M_1, \ldots, M_L$ en nombre L donné et disposés en des positions absolues (i.e., exprimées dans $\mathcal{F}_0$) connues $m_l = (u^l, v^l)^T, l = 1, \ldots, L$. Un robot ponctuel R est placé en une position absolue (i.e., exprimée dans $\mathcal{F}_0$) inconnue, caractérisée par le vecteur caché $x = (u, v)^T$.

La perception de chaque $l^{\text{ème}}$ amer M_l relativement au robot R constitue le vecteur z_l. La superposition $(z_1^T, \ldots, z_L^T)^T$ de toutes les observations effectuées constitue le vecteur de mesure z. L'objectif est bien sûr d'estimer le vecteur $x \in \mathbb{R}^2$ sur la base de $z \in \mathbb{R}^{2L}$. On suppose ne disposer d'aucune connaissance a priori sur x préalablement à l'obtention de z. L'incertitude entâchant z est caractérisée de manière probabiliste : ainsi, si ω désigne le résultat de l'expérience aléatoire en cours, alors $z = \mathsf{Z}(\omega)$ est la réalisation du vecteur aléatoire de mesure $\mathsf{Z} = (\mathsf{Z}_1^T, \ldots, \mathsf{Z}_L^T)^T$ lié à x par le modèle d'observation $p_{\mathsf{Z}|x}(z|x)$. Ce modèle est décrit par l'équation de mesure à bruits additifs Gaussiens suivante :

$$\mathsf{Z} = h(x) + \mathsf{V}\,;\ h(.) \text{ fonction donnée}\,;\ \mathsf{V} \sim \mathcal{G}(\bar{v}, C_v)\,;\ \bar{v}, C_v \text{ donnés.} \tag{1}$$

Divers modèles exprimant le lien entre x caché, $\{u^l, v^l\}_{l=1,\ldots,L}$ connus et Z seront explicités. Divers schémas d'estimation seront considérés, qu'il s'agira d'analyser soigneusement.

On fournit la fonction MATLAB `simulationDonnees.m`, dont le prototype est décrit Figure 1. Cette fonction permet de simuler une expérience aléatoire. Par conséquent, elle renvoie le vecteur de mesure z, stocké dans `Z`, potentiellement différent lors de chacun de ses appels. Elle retourne également le nombre d'amers L (stocké dans `L`), des matrices potentiellement utiles, $\bar{v}, C_v$ (respectivement stockés dans `bar_v`,`C_v`) ainsi que le vecteur x (stocké dans `X`, déterministe caché). Il convient de préciser que si l'expérience était menée sur un robot réel, alors la vérité terrain `X` ne serait pas accessible (sauf à utiliser un système de capture de mouvement par exemple) et `bar_v`,`C_v` devraient être élaborés séparément (en caractérisant les capteurs embarqués sur le robot). On fournit également la fonction `ellipse`, qui,

```
function [Z,L,m,H,Hfull,bar_v,C_v,x] = simulationDonnees(cas_d_etude,plot_p);
%SIMULATIONDONNEES
%  Simulation d'une expérience
%  Syntaxe d'appel :  [Z,L,m,H,Hfull,bar_v,C_v,x] = simulationDonnees(cas_d_etude,plot_p);
%
%  Entrée :
%  . cas_d_etude : index de la section considérée dans le sujet
%  . plot_p : si égal à 1, alors produit un affichage, sinon rien
%
%  Sorties :
%  . Z : réalisation de la variable aléatoire de mesure (2Lx1)
%  . L : nombre d'amers 2D
%  . m : vecteur des positions des amers (2Lx1)
%  . H,Hfull : matrices potentiellement utiles...
%  -> et comme ceci est de la simulation, sont également accessibles
%     . bar_v,C_v : espérance et covariance du bruit de mesure additif Gaussien (2Lx1,2Lx2L)
%     . x         : vecteur de paramètres caché (2x1)
```

Figure 1 – Prototype de `simulationDonnees.m`

```
function h = ellipse(mx,Px,color)
%ELLIPSE
%  Trace l'ellipse contenant 99% des réalisations d'une variable aléatoire
%  Gaussienne 2D de moyenne mx et de covariance Px.
```

Figure 2 – Prototype de `ellipse.m`

selon le prototype reporté Figure 2, permet de tracer l'ensemble de volume minimal dans lequel se réalise avec une probabilité de 99% une variable aléatoire Gaussienne bivariée de moyenne et matrice de covariance données.

TRAVAIL DEMANDÉ

Cas d'étude 1 – Fonction $h(.)$ affine, bruits indépendants identiquement distribués

L'équation de mesure (1) satisfait

$$h(x) = (h_1(x), \dots, h_L(x))^T, \text{ avec } \forall l \in \{1, \dots, L\},\ h_l(x) = m_l - x = \begin{pmatrix} \overrightarrow{RM_l}.\vec{x}_0 \\ \overrightarrow{RM_l}.\vec{y}_0 \end{pmatrix} = \begin{pmatrix} u_l - u \\ v_l - v \end{pmatrix} \quad (2)$$

$$\bar{v} = 0,\ C_v = \sigma^2 \mathbb{I} \text{ avec } \sigma \text{ donné.} \quad (3)$$

1. Invoquer séquentiellement `simulationDonnees.m` de façon à exécuter un nombre N suffisamment élevé d'expériences aléatoires. Stocker les réalisations (ou « échantillons ») de Z obtenues sur ces expériences. Noter la valeur de σ.
2. Visualiser graphiquement :
   (a) les distributions empiriques des échantillons des sous-vecteurs Z_l, $l = 1, \dots, L$;
   (b) leur cohérence avec les lois théoriques des Z_l, $l = 1, \dots, L$.

On décide d'estimer x caché au moyen de l'estimateur

$$\hat{\mathsf{X}} = \frac{1}{L}\sum_{l=1}^{L} \mathsf{Y}_l, \text{ avec } \mathsf{Y}_l = m_l - \mathsf{Z}_l. \tag{4}$$

3. Calcul théorique – Montrer que $\hat{\mathsf{X}}$ est Gaussien. Calculer son espérance ainsi que sa matrice de covariance $\mathrm{Cov}_{\hat{\mathsf{X}}}$. Déduire son biais.
4. Par exploitation des échantillons de Z sur plusieurs résultats d'expériences, visualiser la distribution empirique des estimés $\hat{x}$ de x. Vérifier qu'elle est cohérente avec la loi théorique de $\hat{\mathsf{X}}$. On pourra utiliser la fonction `ellipse` introduite plus haut.
5. Établir la distribution de probabilité théorique de l'erreur d'estimation sur x. Déduire de ce résultat comment tracer, autour d'un estimé $\hat{x}$ de x produit lors d'une expérience, un ellipse de confiance dans lequel x doit théoriquement se trouver avec une probabilité de 99 %. Vérifier la plausibilité du résultat.
6. Facultatif : calculer la valeur de l'erreur quadratique moyenne.

Cas d'étude 2 – Fonction $h(.)$ affine, bruits mutuellement indépendants mais non identiquement distribués

L'équation de mesure (1) satisfait

$$h(x) = (h_1(x), \ldots, h_L(x))^T, \text{ avec } \forall l \in \{1, \ldots, L\}, \; h_l(x) = m_l - x = \begin{pmatrix} \overrightarrow{RM_l}.\vec{x}_0 \\ \overrightarrow{RM_l}.\vec{y}_0 \end{pmatrix} = \begin{pmatrix} u_l - u \\ v_l - v \end{pmatrix} \tag{5}$$

$$\bar{v} = 0, \; C_v = \mathrm{blkdiag}(C_{v1}, \ldots, C_{vL}), \text{ avec } C_{v1}, \ldots, C_{vL} \text{ donnés.} \tag{6}$$

7. Invoquer séquentiellement `simulationDonnees.m` de façon à exécuter un nombre N suffisamment élevé d'expériences aléatoires. Stocker les réalisations (ou « échantillons ») de Z obtenues sur ces expériences.

On décide d'estimer x caché au moyen de l'estimateur $\hat{\mathsf{X}}$ défini dans la section précédente en (4) ainsi que par l'estimé du maximum de vraisemblance $\hat{\mathsf{X}}_{\mathrm{MLE}}$.

8. Calcul théorique – Établir l'expression de $\hat{\mathsf{X}}_{\mathrm{MLE}}$. Montrer que $\hat{\mathsf{X}}$ et $\hat{\mathsf{X}}_{\mathrm{MLE}}$ sont tous deux Gaussiens. Calculer leurs espérances ainsi que leurs matrices de covariance $\mathrm{Cov}_{\hat{\mathsf{X}}}$,$\mathrm{Cov}_{\hat{\mathsf{X}}_{\mathrm{MLE}}}$. Déduire le biais de chacun d'eux.
9. Par exploitation des échantillons de Z sur plusieurs résultats d'expériences, visualiser les distributions empiriques des estimés respectifs $\hat{x}$ et $\hat{x}_{\mathrm{MLE}}$ de x. Vérifier qu'elles sont cohérentes avec les lois théoriques de $\hat{\mathsf{X}}$ et $\hat{\mathsf{X}}_{\mathrm{MLE}}$. On pourra utiliser la fonction `ellipse` introduite plus haut.
10. Établir la distribution de probabilité théorique de l'erreur d'estimation sur x pour chaque estimateur $\hat{\mathsf{X}}$ et $\hat{\mathsf{X}}_{\mathrm{MLE}}$. Déduire de ce résultat comment tracer, autour de chaque estimé $\hat{x}$,$\hat{x}_{\mathrm{MLE}}$ de x produit lors d'une expérience, un ellipse de confiance dans lequel x doit théoriquement se trouver avec une probabilité de 99 %. Vérifier la plausibilité du résultat.

Les bruits associés aux perceptions des différents amers par le robot étant mutuellement indépendants, on montre qu'on peut implémenter un calcul équivalent à l'estimé du maximum de vraisemblance $\hat{x}_{\mathrm{MLE}}$ consistant en l'assimilation récursive (séquentielle) de chaque observation $y_l = \mathsf{Y}_l(\omega)$. La démarche consiste à produire la séquence des estimés $\overset{\circ}{x}^{(1)}, \ldots, \overset{\circ}{x}^{(L)}$ et matrices de covariances $P^{(1)}, \ldots, P^{(L)}$

définis comme suit, avec $H = \mathbb{I}_{2x2}$.

$$\begin{aligned}
\text{Initialisation} \qquad & \overset{\circ}{x}^{(1)} = y_1 && (7)\\
& P^{(1)} = C_{v1} && (8)\\
\forall l \in \{2, \ldots, L\}, \qquad & K^{(l)} = P^{(l-1)} H^T \left(C_{vl} + H P^{(l-1)} H^T\right)^{-1} && (9)\\
& \overset{\circ}{x}^{(l)} = \overset{\circ}{x}^{(l-1)} + K^{(l)} (y_l - H \overset{\circ}{x}^{(l-1)}) && (10)\\
& P^{(l)} = P^{(l-1)} - K^{(l)} H P^{(l-1)} && (11)\\
\text{Estimé/Covariance finaux} \qquad & \hat{x}_{\text{MLE}} = \overset{\circ}{x}^{(L)} && (12)\\
& \text{Cov}_{\hat{\mathsf{X}}_{\text{MLE}}} = P^{(L)} && (13)
\end{aligned}$$

11. Implémenter le calcul. Vérifier le bon fonctionnement.
12. Que se passerait-il si les bruits associés aux perceptions des différents amers par le robot n'étaient pas mutuellement indépendants ? (invoquer par exemple `simulationDonnees` avec l'argument `cas_d_etude` fixé à 0)

Cas d'étude 3 – Fonction $h(.)$ non linéaire

L'équation de mesure (1) satisfait

$$h(x) = (h_1(x), \ldots, h_L(x))^T, \text{ avec } \forall l \in \{1, \ldots, L\}, \; h_l(x) = \begin{pmatrix} \|RM_l\| \\ \widehat{\vec{x}_0, \overrightarrow{RM_l}} \end{pmatrix} = \begin{pmatrix} \sqrt{(u_l-u)^2+(v_l-v)^2} \\ \text{atan2}(v_l-v, u_l-v) \end{pmatrix} \quad (14)$$

$$\bar{v} = 0, \; C_v \text{ donnés.} \quad (15)$$

11. Invoquer séquentiellement `simulationDonnees.m` de façon à exécuter un nombre N suffisamment élevé d'expériences aléatoires. Stocker les réalisations (ou « échantillons ») de Z obtenues sur ces expériences.

On décide d'estimer x caché au moyen de l'estimateur du maximum de vraisemblance $\hat{\mathsf{X}}_{\text{MLE}}$.

12. Calcul théorique – Établir l'expression du problème d'optimisation permettant le calcul de l'estimé du maximum de vraisemblance $\hat{x}_{\text{MLE}}$ sur la base du vecteur z. Développer l'algorithme de Gauss-Newton permettant sa résolution.
13. Procéder à l'implémentation de l'algorithme de Gauss-Newton. Tracer autour de l'estimé $\hat{x}_{\text{MLE}}$ obtenu un ellipse de confiance dans lequel x doit se trouver *approximativement* avec une probabilité de 99 %. Vérifier la plausibilité.