

0.1 Estimation

0.2 Chaîne de Markov cachée

0.3 Méthode séquentielle - Approche Probabiliste

0.4 Filtrage bayésien

Le filtrage bayésien consiste à écrire la récurrence sur les lois de probabilité, pour estimer, en fonction des observations passées et courante $y_{1:k}$ l'état courant x_k et de prédire l'état future x_{k+1} .

Pour simplifier les notations, l'exposant $^{|k|}$ qui conditionne la densité par les observations $y_{1:N}$. La densité de l'état est initialisée par la densité a priori de l'état initial p_{X_0} .

Puis pour tout $k \geq 0$ les lois de probabilité sont propagées.

L'étape de propagation ou *forecast* loi *a priori* est obtenue grace à la loi des probabilité totales

$$p_{X_{k+1}}^{|k|}(x) = \int p_{X_{k+1}|X_k=x'}(x) p_{X_k}^{|k|}(x') dx' \quad (1)$$

La loi *a priori* de la $k + 1$ observations peut être obtenue de nouveau grace à la loi de probabilité totale

$$p_{Y_{k+1}}^{|k|}(y) = \int p_{Y_{k+1}|X_{k+1}=x}(y) p_{X_{k+1}}^{|k|}(x) dx$$

Après la $k + 1$ observation y_{k+1} , l'étape d'*analyse* permet de déterminer la loi *a posteriori* de l'état avec la loi de Bayes appliquées après mesure de Y_n

$$p_{X_{k+1}}^{|k+1|}(x) = p_{X_{k+1}|Y_{k+1}=y_{k+1}}^{|k|}(x) = \frac{p_{Y_{k+1}|X_{k+1}=x}(y) p_{X_{k+1}}^{|k|}(x)}{p_{Y_{k+1}}^{|k|}(y)}.$$

la loi de Bayes après mesure de Y_n

0.4.1 Filtre de Kalman

Le filtre de Kalman introduit en 1960 [5] est une version du filtre Bayésien appliqué à un modèle linéaire Gaussien. Dans ces conditions, la distribution de l'état a priori de l'état et des observations sont défini par leur deux premiers moment tel que la propagation devient

$$\begin{aligned} \hat{m}_X &= \mathbb{E}[X_{k+1}^{|k|}] = M \mathbb{E}[X_k^{|k|}], \\ \hat{P}_{k+1} &= \mathbb{V}[X_{k+1}^{|k|}] = M \mathbb{E}[X_k^{|k|}] M^T + Q, \end{aligned}$$

et le modèle d'observation donne

$$\begin{aligned} m_Y &= \mathbb{E}[Y_{k+1}^{|k|}] = H \mathbb{E}[X_{k+1}^{|k|}], \\ C_{Y,Y} &= \mathbb{V}[Y_{k+1}^{|k|}] = H \mathbb{V}[X_{k+1}^{|k|}] H^T + R, \\ C_{X,Y} &= \mathbb{C}[X_{k+1}^{|k|}, Y_{k+1}^{|k|}] = P_{k+1} H^T, \end{aligned}$$

De telle sorte que la distribution conditionnelle de Y_{k+1} par $^{|k|}$ et $Y_{k+1} = y_{k+1}$, si cette dernière est non-dégénérée (ce qui est le cas si R n'est pas singulière), est défini par ses deux premiers moments qui sont

$$\begin{aligned} m_X &= \mathbb{E}[X_{k+1}^{|k|} | Y_{k+1}^{|k|}] = \hat{m}_X + C_{X,Y} C_{Y,Y}^{-1} (y_{k+1} - m_Y), \\ P_{k+1} &= \mathbb{V}[X_{k+1}^{|k|} | Y_{k+1}^{|k|}] = \hat{P}_{k+1} - C_{X,Y} C_{Y,Y}^{-1} C_{X,Y}^T. \end{aligned}$$

Ainsi la distribution a posteriori est défini comme un produit matriciel où l'estimateur a priori \hat{m}_X et sa variance \hat{P}_{k+1} sont mis à jour à partir du *gain de Kalman* $K = C_{X,Y} C_{Y,Y}^{-1}$ et du *terme d'innovation* ($y_{k+1} - m_Y$) de telle sorte que les précédentes équations s'écrivent

$$\begin{aligned} m_X &= \hat{m}_X + K(y_{k+1} - m_Y), \\ P_{k+1} &= (I - KH)\hat{P}_{k+1} \end{aligned}$$

Finalement, on peut réécr

Données : Initialisation de l'état m_x et de sa covariance P ;

pour $k \geq 1$ **faire**

```

    Prédiction;
     $\hat{m}_x = M m_x$ ;
     $\hat{P} = M \mathbb{E}[X_{k+1}^k] M^T + Q$ ;
    Observation de  $Y \rightarrow y$  Analyse;
    Calcul du gain de Kalman :  $K = \hat{P} H^T (H \hat{P} H^T + R)^{-1}$  ;
    Calcul de l'analyse;
     $m_x = \hat{m}_x + K(y - H \hat{m}_x)$ ;
    Calcul de la matrice de covariance de l'état;

```

Algorithme 1 : Filtre de Kalman

0.4.2 Filtre Particulaire

Le filtre particulaire est une implémentation du filtre bayésien qui approxime la PDF à l'aide d'une distribution empirique. Les transformations du filtre, *forecast* et *analysis* sont appliquées sur les membres de cet échantillon. Cette méthode converge vers la distribution exacte lorsque le nombre de particule $N \rightarrow \infty$.

Le prior de l'état $p(x)$ à l'instant k est représenté par un ensemble de N réalisations $\{x_1, x_2, \dots, x_N\}$ de tel sorte que

$$p_{X_k}(x) \simeq \sum_{i=1}^N \omega_k^i \delta(x - x_k^i) \quad \text{with} \quad \sum_{i=1}^N \omega_k^i = 1, \quad \omega_k^i > 0.$$

où δ est la masse de Dirac et ω_k^i les poids associés à chaque membre. Initialement, les échantillons sont supposés tirés de manière uniforme de tel sorte que $\omega_k^i = 1/N$.

Lors de l'étape de *propagation*, les particules sont propagées par le modèle de manière déterministe.

Pour s'en convaincre, le loi de probabilité totale 1 peut être réécrite

$$\begin{aligned} p_{X_{k+1}}^k(x) &= \int p_{X_{k+1}|X_k=x'}(x) p_{X_k}^k(x') dx' \\ &\simeq \int p_{X_{k+1}|X_k=x'}(x) \sum_{i=1}^N \omega_k^i \delta(x' - x_k^i) dx' \\ &\simeq \sum_{i=1}^N \omega_k^i \int p_{X_{k+1}|X_k=x'}(x) \delta(x' - x_k^i) dx' \\ &\simeq \sum_{i=1}^N \omega_k^i \delta(x - \mathcal{M}_{k,k+1}(x_k^i) - \eta_{k,k+1}) = \sum_{i=1}^N \omega_k^i \delta(x - x_{k+1}^i). \end{aligned}$$

Quant à l'étape d'analyse, elle correspond à une mise à jour du poids de chaque membre, qui correspond à sa vraisemblance conditionnée aux données

$$\begin{aligned}
p_{\mathbf{X}_{k+1}}^{[k+1]}(\mathbf{x}) &\propto p_{\mathbf{Y}_{k+1}|\mathbf{X}_{k+1}=\mathbf{x}}^{[k]}(\mathbf{y}) \sum_{i=1}^N \omega_k^i \delta(\mathbf{x} - \mathbf{x}_{k+1}^i) \\
&\propto \sum_{i=1}^N \omega_k^i p_{\mathbf{Y}_{k+1}|\mathbf{X}_{k+1}=\mathbf{x}_{k+1}^i}^{[k]}(\mathbf{y}) \delta(\mathbf{x} - \mathbf{x}_{k+1}^i)
\end{aligned}$$

Leading to

$$\omega_{k+1}^i = \frac{\omega_k^i p_{\mathbf{Y}_{k+1}|\mathbf{X}_{k+1}=\mathbf{x}_{k+1}^i}^{[k]}(\mathbf{y}_{k+1})}{\omega_k^j \sum_{j=1}^N p_{\mathbf{Y}_{k+1}|\mathbf{X}_{k+1}=\mathbf{x}_{k+1}^j}^{[k]}(\mathbf{y}_{k+1})}$$

Où le dénominateur est simplement un terme de normalisation.

Cependant, lorsque la dimension est grande, le nombre de poids non nulle à tendance à tendre vers 0. Pour éviter cela, des méthodes de rééchantillonnage du *posterior* ont été développées. Le filtre bootstrap [2] consiste à sélectionner les membres de poids les plus élevés, de les cloner de manière proportionnelle à leurs poids. Après échantillonnage, N particules sont rassemblées, dont certaines sont identiques avec des approximativement égaux. Un exemple d'algorithme suivant

pour membre n do faire

 Tirer u dans $\mathcal{U}[0, 1[$;

 Initialiser $j = 1$;

 Affecter $S_w = w^1$;

tant que $S_w < u$ faire

$j = j + 1$;

$S_w = S_w + w(j)$

 Le membre j est conservé et remplace le membre n .

Algorithme 2 : Implémentation du rééchantillonnage par *bootstrap*.

0.5 Material Point Method (MPM)

La méthode MPM est une version de FLIP pour résoudre le problème de mécanique des solides. Elle consiste, comme en élément finis, à résoudre le problème aux valeurs sous sa forme faible, en utilisant conjointement deux discrétisations, une grille et des particules.

Le problème aux conditions limites, sous sa forme forte, se compose des équations d'équilibre, des lois matériaux, de l'équation cinématique et des conditions limites et initiales donnant

$$\begin{cases}
\frac{D\rho}{Dt} + \rho \nabla \cdot \mathbf{v} = 0, & \text{(conservation de la masse)} \\
\rho \frac{D\mathbf{v}}{Dt} = \nabla \cdot \boldsymbol{\sigma} + \rho \mathbf{b}, & \text{(conservation de la quantité de mouvement)} \\
\boldsymbol{\sigma} = LdC(\mathbf{F}), & \text{(loi de comportement)} \\
\mathbf{u}(\mathbf{z}, t) = \bar{\mathbf{u}}, \quad \forall \mathbf{z} \in \Gamma_u, \quad \boldsymbol{\sigma}(\mathbf{z}) \cdot \mathbf{n} = \mathbf{t}, \quad \forall \mathbf{z} \in \Gamma_n, & \text{(conditions limites)} \\
\mathbf{v}(\mathbf{z}, t = 0), \quad \boldsymbol{\sigma}(\mathbf{z}, t = 0) = \boldsymbol{\sigma}_0. & \text{(conditions initiales)}
\end{cases}$$

La forme faible de l'équation de conservation du moment est, en introduisant une fonction test \mathbf{q}

$$\int_{\Omega} \rho \mathbf{a} \cdot \mathbf{q} d\mathbf{z} + \int_{\Omega} \boldsymbol{\sigma} : \nabla \cdot \mathbf{q} d\mathbf{z} = \int_{\Omega} \rho \mathbf{b} d\mathbf{z} + \int_{\partial\Omega_t} \mathbf{t} dS.$$

Le schéma MPM peut être obtenu en utilisant une discrétisation particulière de la densité sur les particules et en utilisant un $\Omega = \bigcup_p \Omega_p$.

0.5.1 Interprétation 1

0.5.2 Interprétation 2

0.5.3 Schéma classique

La méthode MPM est implémentée généralement en trois phases. Dans un premier temps, les quantités définies sur les particules sont transférées sur les noeuds de la grille $p2g$. Le principe fondamentale de la dynamique est alors résolu permettant de déterminer une grille déformée. Finalement, les nouvelles quantités nodales permettent de mettre à jour les quantités particulières dans une phase de transfert grille à particule ($g2p$).

p2g La grille de positions de noeuds x_I est initialisée avec des valeurs nulles.

La masse m_p , la quantité de mouvement $m_p \mathbf{v}_p$ et les forces \mathbf{f}_p sont transférées à la grille à l'aide des fonctions de forme ϕ_I associées à chaque noeuds

$$\begin{aligned} m_I &= \sum_p \varphi_{Ip} m_p, \\ m_I \mathbf{v}_I &= \sum_p \varphi_{Ip} m_p \mathbf{v}_p, \\ \mathbf{f}_I &= \sum_p \varphi_{Ip} \mathbf{f}_p. \end{aligned}$$

Des transferts plus complexes capable de préserver les moments angulaires ont été développés comme APIC [4], Poly-PIC [1], et MLS-MPM [3].

Mise à jour sur la grille La grille à chaque étape est initialisée dans un état non déformée. A l'aide du principe fondamentale de la dynamique, la vitesse sur la grille est mise à jour de manière explicite tel que

$$\begin{aligned} m_I \mathbf{a}_I &= \mathbf{f}_I + \mathbf{f}_g, \\ m_I \mathbf{v}^{n+1} &= \mathbf{v}^n + \Delta t (\mathbf{f}_I + \mathbf{f}_g) / m_I, \\ \mathbf{x}_I^{n+1} &= \mathbf{x}_I^n + \Delta t \mathbf{v}^{n+1}. \end{aligned}$$

C'est durant cette étape que les conditions limites ou les collisions avec un objet peuvent être prise en compte.

g2p Les particules vont suivre la déformation de la grille. Cela aura deux conséquences : La mise à jour de la matrice de déformation \mathbf{F}_p et de leurs positions \mathbf{x}_p et leur vitesses \mathbf{v}_p .

La mise à jour de \mathbf{F}_p est réalisée avec la déformée de la grille \mathbf{x}_I^{n+1} de manière implicite en utilisant \mathbf{v}^{n+1} de telle sorte que

$$\mathbf{F}_p^{n+1} = \left(\mathbf{I} + \Delta t \sum_I \mathbf{v}_I^{n+1} (\nabla \varphi_{Ip}^T) \right) \mathbf{F}_p^n.$$

En ce qui concerne l'étape d'advection des particules, le schéma PIC suggérerait l'interpolation des vitesses tel que

$$\mathbf{v}_{PIC}^{n+1} = \sum_I \varphi_{Ip} \mathbf{v}_I^{n+1}$$

Si ce schéma est stable, il est toutefois dissipatif. Inversement la mise à jour FLIP propose de mettre à jour la vitesse \mathbf{v}_{PIC}^n en interpolant l'accélération tel que

$$\mathbf{v}_{FLIP}^{n+1} = \mathbf{v}_p^n \sum_I \varphi_{Ip} (\mathbf{v}_I^{n+1} - \mathbf{v}_I^n)$$

Dans ce cas, le transfert est conservatif mais instable. Ainsi, il est recommandé d'utiliser pour mettre à jour la vitesse v^{n+1} une combinaison linéaire des deux formulations tel que

$$v_p^{n+1} = \alpha \left(v_p^n \sum_I \varphi_{Ip} (v_I^{n+1} - v_I^n) \right) + (1 - \alpha) \sum_I \varphi_{Ip} v_I^{n+1}$$

avec $\alpha \in [0, 1]$.

Les schémas de type APIC, PolyPIC ou MLS-MPM, utilisant de plus un transfert du gradient de v , utilise une mise à jour PIC tout en restant conservatif.

La position est elle mise jour en interpolant la déformation de la grille de telle sorte que

$$x_p^{n+1} = x_p^n + \sum_I \varphi_{Ip} v^{n+1}$$

Finalement, la grille de calcul peut être effacée et réinitialisée.

La force interne de la particule f_p dépend de la loi de comportement qui lui est associée.

Elle dépend généralement de la contrainte σ_p qui peut être mise à jour au début ou à la fin du schéma donnant deux formulations différence USF (*Update Stress First*) et USL (*Update Stress Last*).

Références

- [1] Chuyuan Fu, Qi Guo, Theodore Gast, Chenfanfu Jiang, and Joseph Teran. A polynomial particle-in-cell method. *ACM Transactions on Graphics*, 36(6) :1–12, November 2017.
- [2] N.J. Gordon. Novel approach to nonlinear/non-gaussian bayesian state estimation. *IEE Proceedings F (Radar and Signal Processing)*, 140 :107–113(6), April 1993.
- [3] Yuanming Hu, Yu Fang, Ziheng Ge, Ziyin Qu, Yixin Zhu, Andre Pradhana, and Chenfanfu Jiang. A moving least squares material point method with displacement discontinuity and two-way rigid body coupling. *ACM Transactions on Graphics*, 37(4) :1–14, August 2018.
- [4] Chenfanfu Jiang, Craig Schroeder, Andrew Selle, Joseph Teran, and Alexey Stomakhin. The affine particle-in-cell method. *ACM Transactions on Graphics*, 34(4) :1–10, July 2015.
- [5] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(1) :35–45, March 1960.