

Filtrage Particulaire :

Quelques exemples “avec les mains” et matlab

Fabien Campillo (IRISA / INRIA)

`Fabien.Campillo@inria.fr`

Plan

- (i) le filtrage non linéaire et son approximation particulière (une présentation simpliste)
- (ii) exemple 1 : redistribuer ou pas
- (iii) exemple 2 : modèle numérique de terrain
- (iv) exemple 3 : poursuite avec obstacles
- (v) exemple 4 : poursuite dans une séquence d’images

exemple 1:
redistribuer ou pas (un cas 1D)

exemple 1:redistribuer ou pas (un cas 1D)

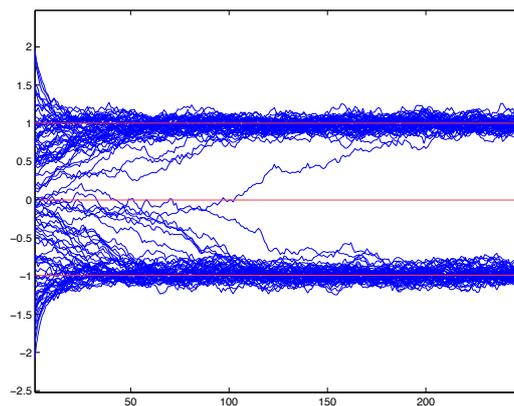
2

le système

$$X_k = X_{k-1} - \alpha X_{k-1} (X_{k-1}^2 - 1) \Delta t + \sqrt{\Delta t} W_{k-1} \quad \dot{X}_t = -\alpha X_t (X_t^2 - 1) + \dot{W}_t$$

$$Y_k = h(X_k) + \sqrt{\Delta t} V_k \quad \text{avec } h(x) = \begin{cases} \alpha^+ x & \text{si } x > 0 \\ \alpha^- x & \text{si } x < 0 \end{cases}$$

deux bassins d'attraction



exemple 2: modèle numérique de terrain

exemple 2:modèle numérique de terrain

4

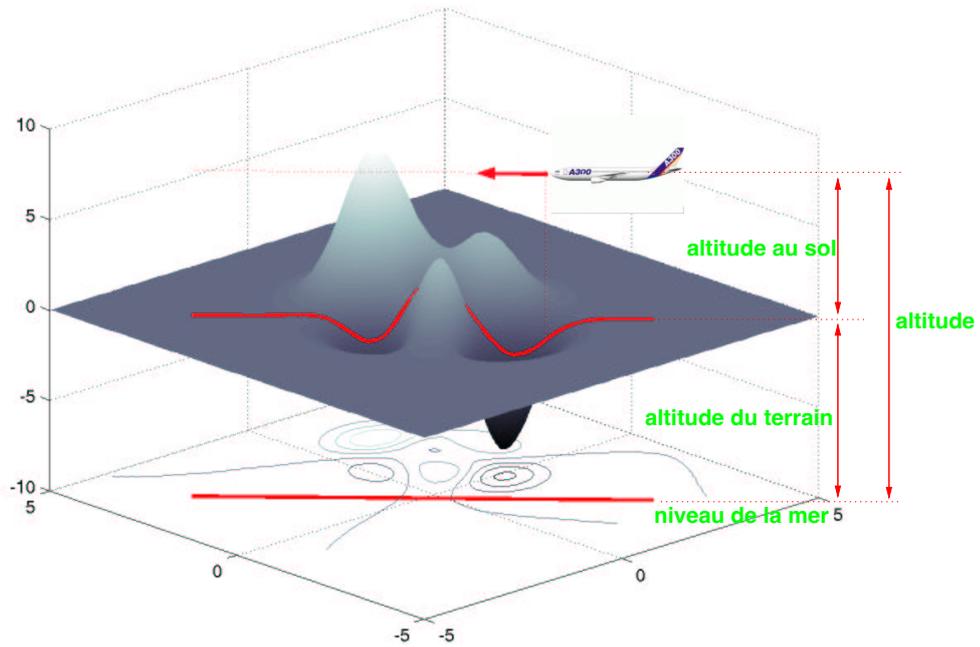
le problème

un avion se déplace sur une zone géographique :

- il mesure l'**altitude du terrain** au point juste au dessous de lui
- il dispose d'une **carte** (avec l'altitude au sol de chaque point)
- il dispose d'une idée (+/- vague) de la **dynamique de l'avion**

ce que l'on cherche : la *position de l'avion sur la carte*

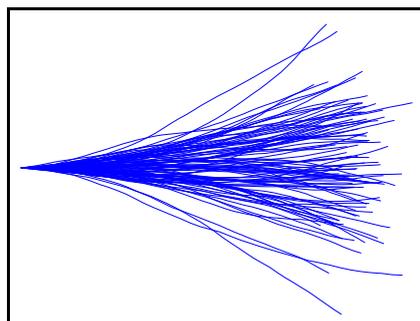
géométrie des mesures



espace/équation d'état

modèle à cap et vitesse constants (bruit)

$$X_k = \begin{cases} r_k^x & \text{coordonnée en } x \\ r_k^y & \text{coordonnée en } y \\ v_k & \text{vitesse} \\ c_k & \text{cap} \end{cases} \begin{cases} r_k^x = r_{k-1}^x + \Delta t v_{k-1} \cos(c_{k-1}) \\ r_k^y = r_{k-1}^y + \Delta t v_{k-1} \sin(c_{k-1}) \\ v_k = v_{k-1} + \Delta t \sigma_v W_{k-1}^1 \\ c_k = c_{k-1} + \Delta t \sigma_c W_{k-1}^2 \end{cases} \quad (\Delta t = 1)$$

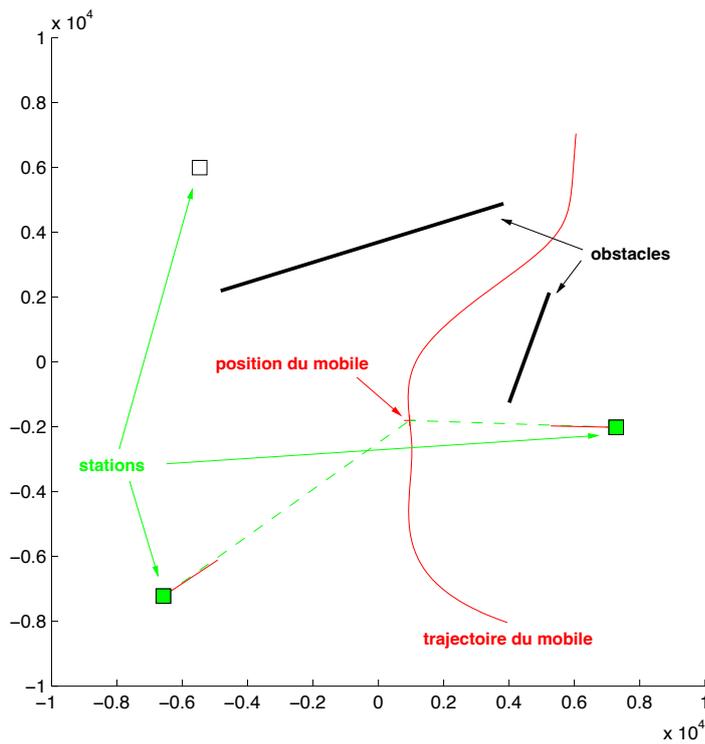


espace/équation d'observation

$h(X_k)$ = altitude du terrain à la position (r_k^x, r_k^y)

$Y_k = h(X_k) +$ bruit d'observation

exemple 3:
poursuite avec obstacles



- un mobile dans le plan
- S stations de mesure : mesures d'angle
- L segments : occultent le déplacement du mobile

équation d'état

- dans le **modèle** :

$$X_k = \begin{pmatrix} r_k^x \\ r_k^y \end{pmatrix} = \begin{cases} r_k^x = r_k^x + \sigma_w w_k^x \\ r_k^y = r_k^y + \sigma_w w_k^y \end{cases}$$

- en **"réalité"** :

- ce que l'on veut à condition que le modèle "suive" (?)
- pas de contrainte de direction

 équation d'observation

pour chacune des stations $s = 1 \dots S$

- si la station s ne voit pas le mobile \rightarrow pas de mesure
- si la station s voit le mobile la mesure est :

$$Y_k^s = h_s(X_k) + \sigma_v V_k$$

avec

$$h_s(X) = \text{atan2}(r^x - r_s^x, r^y - r_s^y)$$

(r_s^x, r_s^y) sont les coordonnées de la station s

 initialisation

on suppose que $\mu_0 \sim \mathcal{N}(m, \sigma_0 I)$

$$\xi_0^{x,i} \leftarrow m^x + \sigma_0 \mathcal{N}(0, 1)$$

$$\xi_0^{y,i} \leftarrow m^y + \sigma_0 \mathcal{N}(0, 1)$$

 mutation

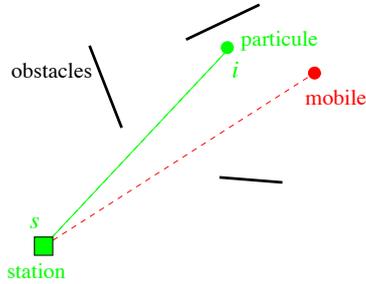
$$\xi_{k|k-1}^{x,i} = \xi_{k-1}^{x,i} + \sigma_w w_k^x$$

$$\xi_{k|k-1}^{y,i} = \xi_{k-1}^{y,i} + \sigma_w w_k^y$$

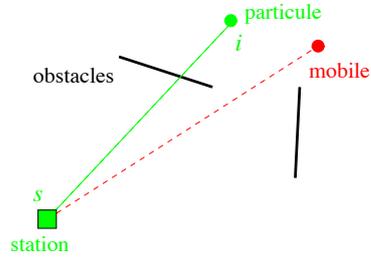
vraisemblance

l'instant k est fixé, on calcule la vraisemblance $\omega_k^{i,s}$ de la particule i pour la station s

cas 1 : la station voit le mobile



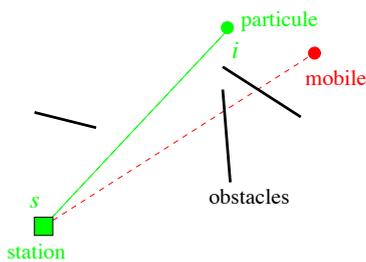
la station voit la particule



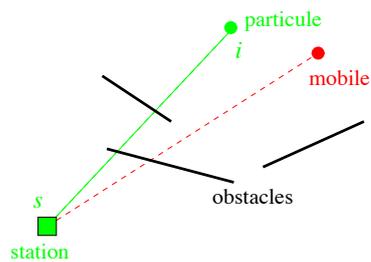
la station ne voit pas la particule

$$\omega_k^{i,s} = \begin{cases} \exp\left(-\frac{|Y_k^s - h_s(\xi_{k|k-1}^i)|^2}{2\sigma_v}\right) & \text{si } s \text{ voit la particule } \xi_{k|k-1}^i \\ 0 & \text{si } s \text{ ne voit pas la particule } \xi_{k|k-1}^i \end{cases} \quad i = 1 \dots N$$

cas 2 : la station ne voit pas le mobile



la station voit la particule



la station ne voit pas la particule

$$\omega_k^{i,s} = \begin{cases} 0 & \text{si } s \text{ voit la particule } \xi_{k|k-1}^i \\ 1 & \text{si } s \text{ ne voit pas la particule } \xi_{k|k-1}^i \end{cases} \quad i = 1 \dots N$$

la vraisemblance de la particule i à l'instant k :

$$\omega_k^i = \prod_{s=1}^S \omega_k^{i,s}$$

exemple 4:
poursuite dans une séquence d'images

Références

D. Salmond, H. Birch, A particle filter for track-before-detect, In *Proc. of the American Control Conference*, June 2001.

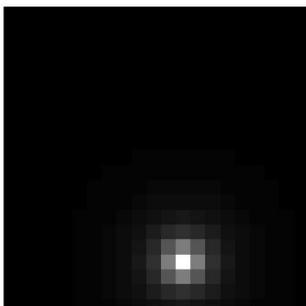
S. Maskell, M.P. Rollason, N.J. Gordon, D.J. Salmond, Efficient particle filtering for multiple target tracking with application to tracking in structured images, *Proc. SPIE Vol. 4728*, p. 251-262, Signal and Data Processing of Small Targets 2002.

poursuite dans une séquence d'images

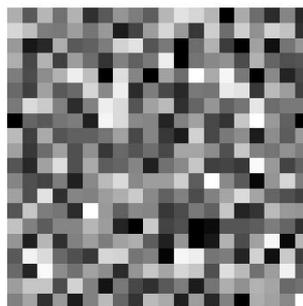
observations

$$Y_1 Y_2 \cdots Y_K$$

chaque image se décompose en $Y_k(s) = I_k^{(mobile)}(s) + I_k^{(bruit)}(s)$ où $s \in \mathcal{S}$ est l'indice de pixel



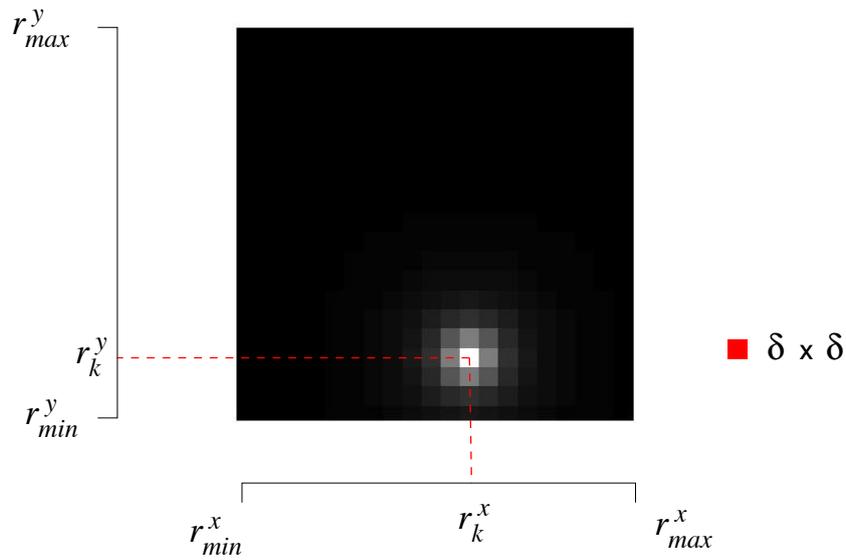
$I_k^{(mobile)}$



$I_k^{(bruit)}$

$$SNR = 10 \times \log_{10} \frac{1}{c^{(bruit)}}$$

image du mobile $Y^r = \{Y^r(s)\}_{s \in \mathcal{S}}$



$r_k = (r_k^x, r_k^y)$ position du mobile à l'instant t_k

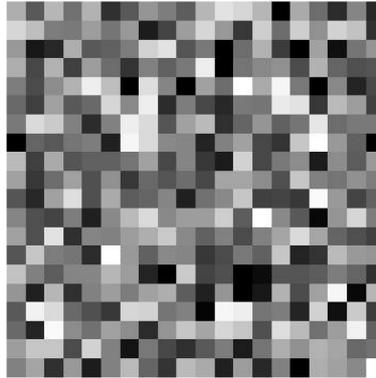
PSF (point spread function)

$$I_k^{(\text{mobile})}(s) = Y^{r_k}(s)$$

$$Y^r(s) = \frac{\delta^2 c}{2\pi\sigma^2} \exp\left(-\frac{(s^x \delta - r^x)^2}{2\sigma^2} - \frac{(s^y \delta - r^y)^2}{2\sigma^2}\right) \underbrace{\mathbf{1}_{|s^x \delta - r^x| < 3} \mathbf{1}_{|s^y \delta - r^y| < 3}}_{=:\mathbf{1}_{C(r)}(s)}$$

$$s = (s^x, s^y) \in \mathcal{S}$$

image bruit



bruit blanc en espace/temps

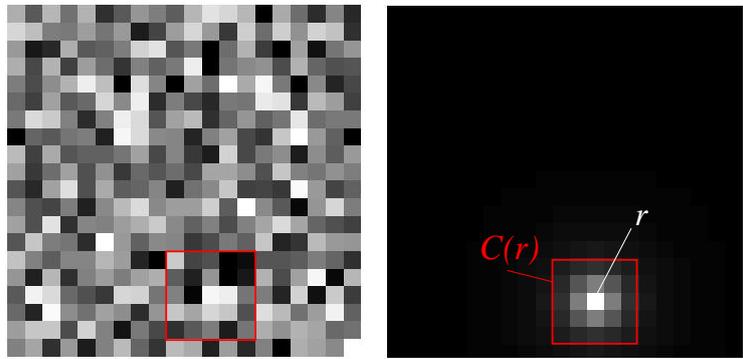
$$I_k^{(\text{bruit})}(s) \sim c^{(\text{bruit})} N(0, 1)$$

fonction de vraisemblance

$$r \mapsto L(Y|r) = \exp \left\{ -\frac{1}{2 c^{(\text{bruit})}} \sum_{s \in \mathcal{S}} [Y^r(s) - Y(s)]^2 \right\} \quad Y = \{Y(s)\}_{s \in \mathcal{S}} \text{ fixée}$$

$$\begin{aligned} L(Y|r) &= \exp \left\{ -\frac{1}{2 c^{(\text{bruit})}} \sum_{s \in \mathcal{S}} Y^r(s) [Y^r(s) - Y(s)] \right\} \exp \left\{ -\frac{1}{2 c^{(\text{bruit})}} \sum_{s \in \mathcal{S}} [Y(s)]^2 \right\} \\ &= \exp \left\{ -\frac{1}{2 c^{(\text{bruit})}} \sum_{s \in C(r)} Y^r(s) [Y^r(s) - Y(s)] \right\} \exp \left\{ -\frac{1}{2 c^{(\text{bruit})}} \sum_{s \in \mathcal{S}} [Y(s)]^2 \right\} \\ &\propto \exp \left\{ -\frac{1}{2 c^{(\text{bruit})}} \underbrace{\sum_{s \in C(r)} Y^r(s) [Y^r(s) - 2 Y(s)]}_{\text{somme locale}} \right\} \\ &\quad \underbrace{\hspace{10em}}_{=: M_{C(r)}(Y^r, Y)} \end{aligned}$$

$$\tilde{L}(Y|r) = \exp \left\{ - \frac{1}{2 \sigma_{\text{bruit}}^2} M_{C(r)}(Y^r, Y) \right\}$$



vecteur d'état

$$X_k = \begin{pmatrix} X_k^x \\ X_k^y \end{pmatrix} = \begin{pmatrix} a_k^x \\ v_k^x \\ r_k^x \\ a_k^y \\ v_k^y \\ r_k^y \end{pmatrix} \begin{matrix} \rightarrow \text{utilisé dans l'observation} \\ \rightarrow \text{utilisé dans l'observation} \end{matrix}$$

bruits indépendants en x et y

$$X_{k+1}^{[x|y]} = F X_k^{[x|y]} + \sigma_W G W_k^{[x|y]}$$

avec

$$F = \begin{pmatrix} 1 & 0 & 0 \\ \Delta & 1 & 0 \\ \frac{\Delta^2}{2} & \Delta & 1 \end{pmatrix} \quad G G^* = \begin{pmatrix} \Delta & \Delta^2/2 & \Delta^3/6 \\ \Delta^2/2 & \Delta^3/3 & \Delta^4/8 \\ \Delta^3/6 & \Delta^4/8 & \Delta^5/20 \end{pmatrix} \quad Q = \sigma_W^2 G G^*$$

algorithme

```

% initialisation
 $\xi_{1-}^i \sim \mu_{1-}$   $i = 1 : N$ 
% itérations
pour  $k = 1 \dots K$  faire
    % vraisemblance (modèle d'observation)
     $\omega_k^i \leftarrow \tilde{L}(Y_k | \xi_{k-}^i)$   $i = 1 : N$ 
    % correction (sélection)
     $\xi_k^i \sim \sum_j \omega_k^j \delta(\xi_{k-}^j)$   $i = 1 : N$ 
    % prédiction (modèle d'état)
     $\xi_{k+1-}^i \leftarrow F \xi_k^i + \sigma_w G W_k^i$   $i = 1 : N$ 
sortie  $\xi_{1:N}$ 
fin pour
    
```

initialisation

$$\xi_{1-}^i \begin{cases} r_{1-}^i = (r_{1-}^{x,i}, r_{1-}^{y,i}) \sim U([r_{\min}^x, r_{\max}^x] \times [r_{\min}^y, r_{\max}^y]) \\ v_{1-}^i = (v_{1-}^{x,i}, v_{1-}^{y,i}) \sim U([v_{\min}^x, v_{\max}^x] \times [v_{\min}^y, v_{\max}^y]) \\ a_{1-}^i = (a_{1-}^{x,i}, a_{1-}^{y,i}) \sim U([a_{\min}^x, a_{\max}^x] \times [a_{\min}^y, a_{\max}^y]) \end{cases}$$

sélection

déjà vu

prédiction

bruits indépendants en x et y

$$\xi_{k+1}^{[x|y],i} \leftarrow F \xi_k^{[x|y],i} + \sigma_w G W_k^{[x|y],i}$$

algorithme

initialisation

$$\begin{aligned}
 r_{-}^{[x|y],1:N} &\leftarrow r_{\min}^{[x|y]} + (r_{\max}^{[x|y]} - r_{\min}^{[x|y]}) \text{rand}(1, N) \\
 v_{-}^{[x|y],1:N} &\leftarrow v_{\min}^{[x|y]} + (v_{\max}^{[x|y]} - v_{\min}^{[x|y]}) \text{rand}(1, N) \\
 a_{-}^{[x|y],1:N} &\leftarrow a_{\min}^{[x|y]} + (a_{\max}^{[x|y]} - a_{\min}^{[x|y]}) \text{rand}(1, N)
 \end{aligned}$$

pour $k = 1 : K$ **faire**

vraisemblance

$$\omega^i \propto \tilde{L}(Y_k | r_{-}^i) \quad i = 1 : N \text{ (+normalisation)}$$

sélection

$$\begin{aligned}
 \text{offsprings} &\leftarrow \text{f_resample}(\omega^{1:N}) \\
 \text{indices} &\leftarrow \text{f_resample_indices}(\text{offsprings}) \\
 r^{[x|y],1:N} &\leftarrow r_{-}^{[x|y],\text{indices}(1:N)} \\
 v^{[x|y],1:N} &\leftarrow v_{-}^{[x|y],\text{indices}(1:N)} \\
 a^{[x|y],1:N} &\leftarrow a_{-}^{[x|y],\text{indices}(1:N)}
 \end{aligned}$$

prédiction

$$\begin{pmatrix} r_{-}^{[x|y],1:N} \\ v_{-}^{[x|y],1:N} \\ a_{-}^{[x|y],1:N} \end{pmatrix} \leftarrow F \times \begin{pmatrix} r^{[x|y],1:N} \\ v^{[x|y],1:N} \\ a^{[x|y],1:N} \end{pmatrix} + \sigma_W G \times \begin{pmatrix} \text{rand}(1, N) \\ \text{rand}(1, N) \\ \text{rand}(1, N) \end{pmatrix}$$

fin

c'est possible

- seul r apparaît dans le calcul de la vraisemblance, i.e. l'observation ne dépend de l'état que par la composante r
- conditionnellement à r l'équation d'état est linéaire/gaussienne

motivations

à nombre de particules fixé c'est **plus coûteux** mais c'est **plus "précis"**:

sans RB N particules pour \mathbb{R}^6

avec RB N particules pour \mathbb{R}^2

rappel: itération $k \rightarrow k + 1$

- (i) vraisemblance $\omega^i \propto \tilde{L}(Y_k | r_-^i)$
- (ii) redistribution $r^i \sim \sum_j \omega^j \delta_{r_-^j}$ (travail sur $\{\omega^i\}_{i=1:N}$)
- (iii) prédiction $r_-^i \sim \text{loi}(r_{k+1} | r_k = r^i)$

ici $\text{loi}(r_{k+1} | r_k = r^i)$ est **gaussienne** (la covariance ne dépend pas de i)

décomposition: $\alpha = (a, v)$ $\beta = r$

$$\begin{pmatrix} \alpha_{k+1} \\ \beta_{k+1} \end{pmatrix} = \begin{pmatrix} F^{\alpha\alpha} & F^{\alpha\beta} \\ F^{\beta\alpha} & F^{\beta\beta} \end{pmatrix} \begin{pmatrix} \alpha_k \\ \beta_k \end{pmatrix} + \begin{pmatrix} W_k^\alpha \\ W_k^\beta \end{pmatrix} \quad \text{cov}(W_k) = \begin{pmatrix} Q^{\alpha\alpha} & Q^{\alpha\beta} \\ Q^{\beta\alpha} & Q^{\beta\beta} \end{pmatrix}$$

il s'agit de simuler selon la loi $r_{k+1}^i \sim \text{loi}(\beta_{k+1} | \beta_k = r_k^i)$
 pour chaque particule r_k^i l'itération $k \rightarrow k + 1$ de RB est

$$\hat{\alpha}_{k+1-}^i = F^{\alpha\alpha} \hat{\alpha}_k^i + F^{\alpha\beta} r_k^i \quad \text{loi}(\alpha_{k+1} | \beta_k = r_k^i) = \mathcal{N}(\hat{\alpha}_{k+1-}^i, R_{k+1-})$$

$$R_{k+1-} = F^{\alpha\alpha} R_k (F^{\alpha\alpha})^* + Q^{\alpha\alpha} \quad \text{entrée}$$

$$\hat{r}_{k+1-}^i = F^{\beta\alpha} \hat{\alpha}_k^i + F^{\beta\beta} r_k^i \quad \text{loi}(\beta_{k+1} | \beta_k = r_k^i) = \mathcal{N}(\hat{r}_{k+1-}^i, \Xi_{k+1})$$

$$\Xi_{k+1} = F^{\beta\alpha} R_k (F^{\beta\alpha})^* + Q^{\beta\beta} \quad \boxed{r_{k+1-}^i \sim \text{loi}(\beta_{k+1} | \beta_k = r_k^i)}$$

$$S_{k+1} = F^{\alpha\alpha} R_k (F^{\beta\alpha})^* + Q^{\alpha\beta}$$

$$\hat{\alpha}_{k+1}^i = \hat{\alpha}_{k+1-}^i + S_{k+1} \Xi_{k+1}^{-1} (r_{k+1}^i - \hat{r}_{k+1-}^i) \quad \text{loi}(\alpha_{k+1} | \beta_1 = r_1^i, \beta_{k+1} = r_{k+1-}^i) = \mathcal{N}(\hat{\alpha}_{k+1}^i, R_{k+1})$$

$$R_{k+1} = R_{k+1-} - S_{k+1} \Xi_{k+1}^{-1} S_{k+1}^* \quad \text{les covariances ne dépendent pas de } i$$

comment ça marche....

$$\omega^i \propto \tilde{L}(Y_k | r_-^i)$$

$$r^i \sim \sum_j \omega^j \delta_{r_-^j} \quad (\text{on sélectionne } \hat{\alpha}^i, r^i)$$

$$\hat{\alpha}_-^i = F^{\alpha\alpha} \hat{\alpha}^i + F^{\alpha\beta} r^i = F^{\alpha\alpha} \hat{\alpha}^i \quad 2$$

$$R_- = F^{\alpha\alpha} R (F^{\alpha\alpha})^* + Q^{\alpha\alpha} \quad 2 \times 2$$

$$\hat{r}_-^i = F^{\beta\alpha} \hat{\alpha}^i + F^{\beta\beta} r^i = F^{\beta\alpha} \hat{\alpha}^i + r^i \quad 1$$

$$\Xi = F^{\beta\alpha} R (F^{\beta\alpha})^* + Q^{\beta\beta} \quad 1 \times 1$$

$$r_-^i \sim N(\hat{r}_-^i, \Xi)$$

$$S = F^{\alpha\alpha} R (F^{\beta\alpha})^* + Q^{\alpha\beta} \quad 2 \times 1$$

$$\hat{\alpha}^i = \hat{\alpha}_-^i + S \Xi^{-1} (r_-^i - \hat{r}_-^i) = \hat{\alpha}_-^i + \frac{1}{\Xi} (r_-^i - \hat{r}_-^i) S \quad 2$$

$$R = R_- - S \Xi^{-1} S^* = R_- - \frac{1}{\Xi} S S^* \quad 2 \times 2$$

initialisation RB

- particules $r_{-}^{[x|y],1:N} \leftarrow r_{\min}^{[x|y]} + (r_{\max}^{[x|y]} - r_{\min}^{[x|y]}) \mathbf{rand}(1, N)$
- moyennes $\hat{\alpha}_{-}^{[x|y],1:N}$
- covariances $R_{-}^{[x|y]}$

algorithme

initialisation

$$r_{-}^{[x|y],1:N} \leftarrow r_{\min}^{[x|y]} + (r_{\max}^{[x|y]} - r_{\min}^{[x|y]}) \mathbf{rand}(1, N)$$

$$\alpha_{-}^{[x|y],1:N} \leftarrow \begin{pmatrix} a_{\min}^{[x|y]} + (a_{\max}^{[x|y]} - a_{\min}^{[x|y]}) \mathbf{rand}(1, N) \\ v_{\min}^{[x|y]} + (v_{\max}^{[x|y]} - v_{\min}^{[x|y]}) \mathbf{rand}(1, N) \end{pmatrix}$$

$$R_{-}^{[x|y]} \leftarrow \dots$$

pour $k = 1 : K$ **faire**

vraisemblance

$$\omega^i \propto \tilde{L}(Y_k | r_-^i) \quad i = 1 : N \text{ (+normalisation)}$$

sélection

$$\text{offsprings} \leftarrow \text{f_resample}(\omega^{1:N})$$

$$\text{indices} \leftarrow \text{f_resample_indices}(\text{offsprings})$$

$$r^{[x|y],1:N} \leftarrow r_-^{[x|y],\text{indices}(1:N)}$$

$$\alpha^{[x|y],1:N} \leftarrow \alpha_-^{[x|y],\text{indices}(1:N)}$$

$$R^{[x|y],1:N} \leftarrow R_-^{[x|y],\text{indices}(1:N)}$$

prédiction

$$\hat{\alpha}_-^i = F^{\alpha\alpha} \hat{\alpha}^i + F^{\alpha\beta} r^i$$

$$R_- = F^{\alpha\alpha} R (F^{\alpha\alpha})^* + Q^{\alpha\alpha}$$

$$\hat{r}_-^i = F^{\beta\alpha} \hat{\alpha}^i + F^{\beta\beta} r^i$$

$$\Xi = F^{\beta\alpha} R (F^{\beta\alpha})^* + Q^{\beta\beta}$$

$$r_-^i \sim N(\hat{r}_-^i, \Xi)$$

$$S = F^{\alpha\alpha} R (F^{\beta\alpha})^* + Q^{\alpha\beta}$$

$$\hat{\alpha}^i = \hat{\alpha}_-^i + S \Xi^{-1} (r_-^i - \hat{r}_-^i)$$

$$R = R_- - S \Xi^{-1} S^*$$

fin

Filtrage Particulaire :
Quelques exemples “avec les mains” (et matlab)
suite

Fabien Campillo + François Le Gland (IRISA / INRIA)
campillo@irisa.fr + legland@irisa.fr

Simon

Intérêt de la vectorisation

Références

D. Salmond, H. Birch, A particle filter for track-before-detect, In *Proc. of the American Control Conference*, June 2001.

S. Maskell, M.P. Rollason, N.J. Gordon, D.J. Salmond, Efficient particle filtering for multiple target tracking with application to tracking in structured images, *Proc. SPIE Vol. 4728*, p. 251-262, Signal and Data Processing of Small Targets 2002.

Plan

- (i) problème
- (ii) algorithme
- (iii) rao–blackwell

problème

problème

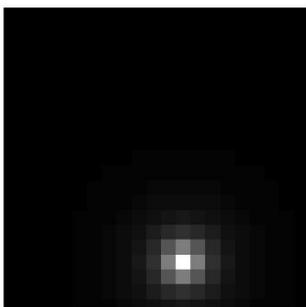
2

poursuite dans une séquence d'images

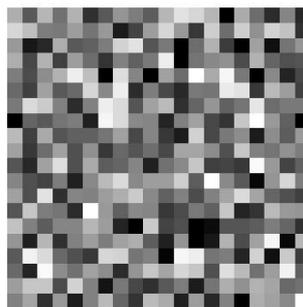
observations

$$Y_1 Y_2 \cdots Y_K$$

chaque image se décompose en $Y_k(s) = I_k^{(\text{mobile})}(s) + I_k^{(\text{bruit})}(s)$ où $s \in \mathcal{S}$ est l'indice de pixel



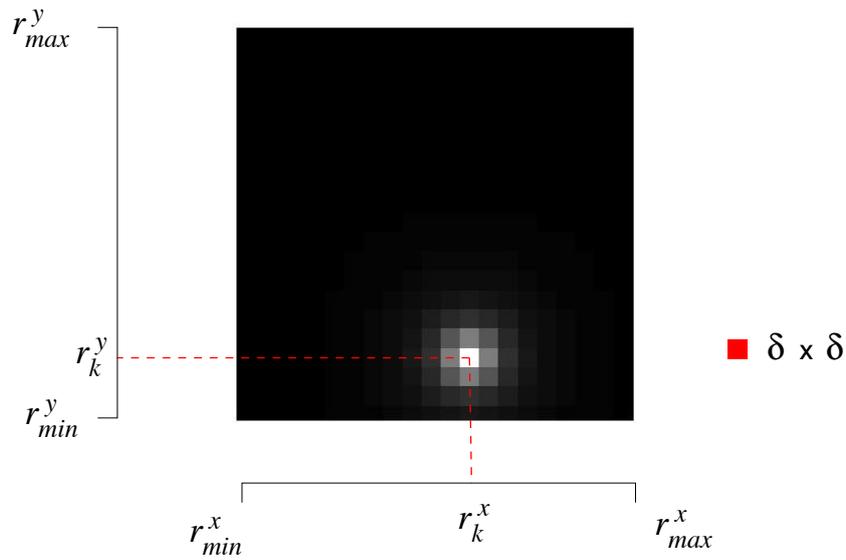
$I_k^{(\text{mobile})}$



$I_k^{(\text{bruit})}$

$$\text{SNR} = 10 \times \log_{10} \frac{1}{c^{(\text{bruit})}}$$

image du mobile $Y^r = \{Y^r(s)\}_{s \in \mathcal{S}}$



$r_k = (r_k^x, r_k^y)$ position du mobile à l'instant t_k

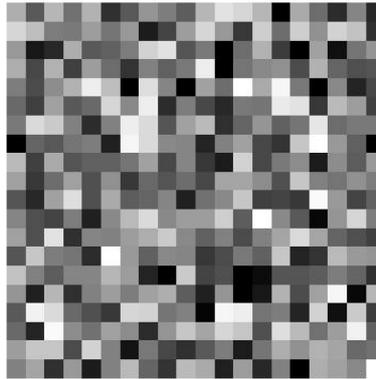
PSF (point spread function)

$$I_k^{(mobile)}(s) = Y^{r_k}(s)$$

$$Y^r(s) = \frac{\delta^2 c}{2\pi\sigma^2} \exp\left(-\frac{(s^x \delta - r^x)^2}{2\sigma^2} - \frac{(s^y \delta - r^y)^2}{2\sigma^2}\right) \underbrace{\mathbf{1}_{|s^x \delta - r^x| < 3} \mathbf{1}_{|s^y \delta - r^y| < 3}}_{=:\mathbf{1}_{C(r)}(s)}$$

$$s = (s^x, s^y) \in \mathcal{S}$$

image bruit



bruit blanc en espace/temps

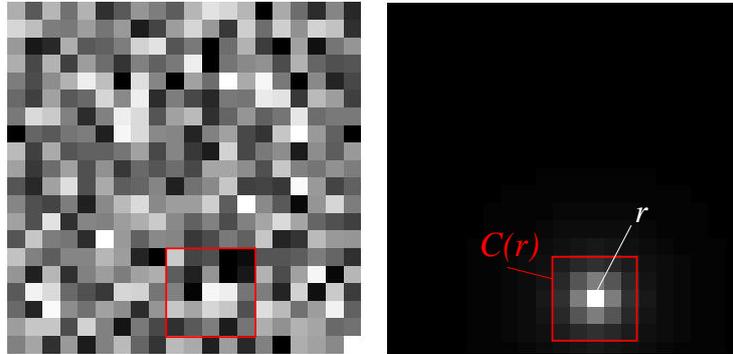
$$I_k^{(\text{bruit})}(s) \sim c^{(\text{bruit})} N(0, 1)$$

fonction de vraisemblance

$$r \mapsto L(Y|r) = \exp \left\{ -\frac{1}{2c^{(\text{bruit})}} \sum_{s \in \mathcal{S}} [Y^r(s) - Y(s)]^2 \right\} \quad Y = \{Y(s)\}_{s \in \mathcal{S}} \text{ fixée}$$

$$\begin{aligned} L(Y|r) &= \exp \left\{ -\frac{1}{2c^{(\text{bruit})}} \sum_{s \in \mathcal{S}} Y^r(s) [Y^r(s) - Y(s)] \right\} \exp \left\{ -\frac{1}{2c^{(\text{bruit})}} \sum_{s \in \mathcal{S}} [Y(s)]^2 \right\} \\ &= \exp \left\{ -\frac{1}{2c^{(\text{bruit})}} \sum_{s \in C(r)} Y^r(s) [Y^r(s) - Y(s)] \right\} \exp \left\{ -\frac{1}{2c^{(\text{bruit})}} \sum_{s \in \mathcal{S}} [Y(s)]^2 \right\} \\ &\propto \exp \left\{ -\frac{1}{2c^{(\text{bruit})}} \underbrace{\sum_{s \in C(r)} Y^r(s) [Y^r(s) - 2Y(s)]}_{\text{somme locale}} \right\} \\ &\quad \underbrace{\hspace{10em}}_{=: M_{C(r)}(Y^r, Y)} \end{aligned}$$

$$\tilde{L}(Y|r) = \exp \left\{ - \frac{1}{2\sigma_{\text{bruit}}^2} M_{C(r)}(Y^r, Y) \right\}$$



vecteur d'état

$$X_k = \begin{pmatrix} X_k^x \\ X_k^y \end{pmatrix} = \begin{pmatrix} a_k^x \\ v_k^x \\ r_k^x \\ a_k^y \\ v_k^y \\ r_k^y \end{pmatrix} \begin{matrix} \rightarrow \text{utilisé dans l'observation} \\ \rightarrow \text{utilisé dans l'observation} \end{matrix}$$

bruits indépendants en x et y

$$X_{k+1}^{[x|y]} = F X_k^{[x|y]} + \sigma_W G W_k^{[x|y]}$$

avec

$$F = \begin{pmatrix} 1 & 0 & 0 \\ \Delta & 1 & 0 \\ \frac{\Delta^2}{2} & \Delta & 1 \end{pmatrix} \quad G G^* = \begin{pmatrix} \Delta & \Delta^2/2 & \Delta^3/6 \\ \Delta^2/2 & \Delta^3/3 & \Delta^4/8 \\ \Delta^3/6 & \Delta^4/8 & \Delta^5/20 \end{pmatrix} \quad Q = \sigma_W^2 G G^*$$

algorithme

algorithme

10

algorithme

```
% initialisation
 $\xi_{1-}^p \sim \mu_{1-} \quad p = 1 : P$ 
% itérations
pour  $k = 1 \dots K$  faire
    % vraisemblance (modèle d'observation)
     $\omega_k^p \leftarrow \tilde{L}(Y_k | \xi_{k-}^p) \quad p = 1 : P$ 
    % correction (sélection)
     $\xi_k^p \sim \sum_q \omega_k^q \delta(\xi_{k-}^q) \quad p = 1 : P$ 
    % prédiction (modèle d'état)
     $\xi_{k+1-}^p \leftarrow \xi_k^p \quad p = 1 : P$ 
sortie  $\xi_{1:P}$ 
fin pour
```

initialisation

$$\xi_{1-}^p \begin{cases} r_{1-}^p = (r_{1-}^{x,p}, r_{1-}^{y,p}) \sim U([r_{\min}^x, r_{\max}^x] \times [r_{\min}^y, r_{\max}^y]) \\ v_{1-}^p = (v_{1-}^{x,p}, v_{1-}^{y,p}) \sim U([v_{\min}^x, v_{\max}^x] \times [v_{\min}^y, v_{\max}^y]) \\ a_{1-}^p = (a_{1-}^{x,p}, a_{1-}^{y,p}) \sim U([a_{\min}^x, a_{\max}^x] \times [a_{\min}^y, a_{\max}^y]) \end{cases}$$

sélection

déjà vu

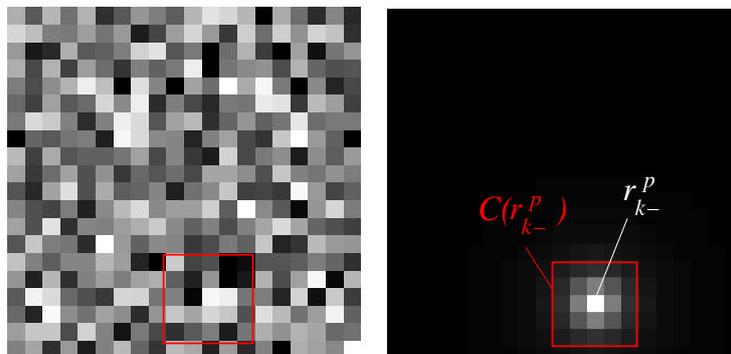
prédiction

bruits indépendants en x et y

$$\xi_{k+1}^{[x|y],p} \leftarrow F \xi_k^{[x|y],p} + \sigma_W G W_k^{[x|y],p}$$

fonction de vraisemblance

$$\omega_k^p = \tilde{L}(Y_k | \xi_{k-}^p) = \exp \left(- \frac{1}{2 C^{(\text{bruit})}} M_{C(r_{k-}^p)}(Y^{r_{k-}^p}, Y_k) \right)$$



algorithme

initialisation

$$\begin{aligned}
r_{-}^{[x|y],1:P} &\leftarrow r_{\min}^{[x|y]} + (r_{\max}^{[x|y]} - r_{\min}^{[x|y]}) \text{rand}(1, P) \\
v_{-}^{[x|y],1:P} &\leftarrow v_{\min}^{[x|y]} + (v_{\max}^{[x|y]} - v_{\min}^{[x|y]}) \text{rand}(1, P) \\
a_{-}^{[x|y],1:P} &\leftarrow a_{\min}^{[x|y]} + (a_{\max}^{[x|y]} - a_{\min}^{[x|y]}) \text{rand}(1, P)
\end{aligned}$$

pour $k = 1 : K$ **faire**vraisemblance

$$\omega^p \propto \tilde{L}(Y_k | r_{-}^p) \quad p = 1 : P \text{ (+normalisation)}$$

sélection

$$\text{offsprings} \leftarrow \text{f_resample}(\omega^{1:P})$$

$$\text{indices} \leftarrow \text{f_resample_indices}(\text{offsprings})$$

$$r^{[x|y],1:P} \leftarrow r_{-}^{[x|y],\text{indices}(1:P)}$$

$$v^{[x|y],1:P} \leftarrow v_{-}^{[x|y],\text{indices}(1:P)}$$

$$a^{[x|y],1:P} \leftarrow a_{-}^{[x|y],\text{indices}(1:P)}$$

prédiction

$$\begin{pmatrix} r_{-}^{[x|y],1:P} \\ v_{-}^{[x|y],1:P} \\ a_{-}^{[x|y],1:P} \end{pmatrix} \leftarrow F \times \begin{pmatrix} r^{[x|y],1:P} \\ v^{[x|y],1:P} \\ a^{[x|y],1:P} \end{pmatrix} + \sigma_W G \times \begin{pmatrix} \text{rand}(1, P) \\ \text{rand}(1, P) \\ \text{rand}(1, P) \end{pmatrix}$$

fin

```

function offsprings=f_resample_comb_randshift(weights)
% function offsprings=f_resample_comb(weights)
% the "comb" algo with a random shift
n_weights      = length(weights);
weights        = cumsum(weights)+rand/n_weights;
weights        = weights/weights(end);
weights        = floor(weights*n_weights);
offsprings     = [weights(1) weights(2:n_weights)-weights(1:n_weights-1)];

```

```

function indices = f_resample_indices(offsprings)
% function indices = f_resample_indices(offsprings)
% to be used after a reseampling algo
% INPUT a table [i1 i2 ... in]
%       ij : means that point #j has been drawn ij times
%       property : i1+...+in=n
% OUTPUT [1*ones(1,i1) 2*ones(1,i2) ... n*ones(1;in)]
ooffsprings    = cumsum(offsprings+1);
a              = zeros(1,ooffsprings(end));
a(ooffsprings) = 1;
aa            = cumsum(a);
indices       = aa(~a)+1;

```

vectorisation

- la vectorisation du générateur
- initialisation: OK (simple)
- sélection: OK (module le travail sur l'algorithme de sélection)
- prédiction: OK (simple)
- calcul de la vraisemblance: demande du travail dans cet exemple

rao-blackwellisation

c'est possible

- seul r apparaît dans le calcul de la vraisemblance, i.e. l'observation ne dépend de l'état que par la composante r
- conditionnellement à r l'équation d'état est linéaire/gaussienne

motivations

à nombre de particules fixé c'est **plus coûteux** mais c'est **plus précis**:

sans RB P particules pour \mathbb{R}^6

avec RB P particules pour \mathbb{R}^2

rappel: itération $k \rightarrow k + 1$

- (i) vraisemblance $\omega^p \propto \tilde{L}(Y_k | r_-^p)$
- (ii) redistribution $r^p \sim \sum_q \omega^q \delta_{r_-^q}$ (travail sur $\{\omega^p\}_{p=1:P}$)
- (iii) prédiction $r_-^p \sim \text{loi}(r_{k+1} | r_k = r^p)$

ici $\text{loi}(r_{k+1} | r_k = r^p)$ est **gaussienne** (la covariance ne dépend pas de p)

décomposition: $\alpha = (a, v)$ $\beta = r$

$$\begin{pmatrix} \alpha_{k+1} \\ \beta_{k+1} \end{pmatrix} = \begin{pmatrix} F^{\alpha\alpha} & F^{\alpha\beta} \\ F^{\beta\alpha} & F^{\beta\beta} \end{pmatrix} \begin{pmatrix} \alpha_k \\ \beta_k \end{pmatrix} + \begin{pmatrix} W_k^\alpha \\ W_k^\beta \end{pmatrix} \quad \text{cov}(W_k) = \begin{pmatrix} Q^{\alpha\alpha} & Q^{\alpha\beta} \\ Q^{\beta\alpha} & Q^{\beta\beta} \end{pmatrix}$$

il s'agit de simuler selon la loi $r_{k+1}^p \sim \text{loi}(\beta_{k+1} | \beta_k = r_k^p)$
 pour chaque particule r_k^p l'itération $k \rightarrow k + 1$ de RB est

$$\hat{\alpha}_{k+1-}^p = F^{\alpha\alpha} \hat{\alpha}_k^p + F^{\alpha\beta} r_k^p \quad \text{loi}(\alpha_{k+1} | \beta_k = r_k^p) = \mathcal{N}(\hat{\alpha}_{k+1-}^p, R_{k+1-})$$

$$R_{k+1-} = F^{\alpha\alpha} R_k (F^{\alpha\alpha})^* + Q^{\alpha\alpha} \quad \text{entrée}$$

$$\hat{r}_{k+1-}^p = F^{\beta\alpha} \hat{\alpha}_k^p + F^{\beta\beta} r_k^p \quad \text{loi}(\beta_{k+1} | \beta_k = r_k^p) = \mathcal{N}(\hat{r}_{k+1-}^p, \Xi_{k+1})$$

$$\Xi_{k+1} = F^{\beta\alpha} R_k (F^{\beta\alpha})^* + Q^{\beta\beta} \quad \boxed{r_{k+1-}^p \sim \text{loi}(\beta_{k+1} | \beta_k = r_k^p)}$$

$$S_{k+1} = F^{\alpha\alpha} R_k (F^{\beta\alpha})^* + Q^{\alpha\beta}$$

$$\hat{\alpha}_{k+1}^p = \hat{\alpha}_{k+1-}^p + S_{k+1} \Xi_{k+1}^{-1} (r_{k+1}^p - \hat{r}_{k+1-}^p) \quad \text{loi}(\alpha_{k+1} | \beta_1 = r_1^p, \beta_{k+1} = r_{k+1-}^p) = \mathcal{N}(\hat{\alpha}_{k+1}^p, R_{k+1})$$

$$R_{k+1} = R_{k+1-} - S_{k+1} \Xi_{k+1}^{-1} S_{k+1}^* \quad \text{les covariances ne dépendent pas de } i$$

comment ça marche....

$$\omega^p \propto \tilde{L}(Y_k | r_-^p)$$

$$r^p \sim \sum_q \omega^q \delta_{r^q} \quad (\text{on sélectionne } \hat{\alpha}^p, r^p)$$

$$\hat{\alpha}_-^p = F^{\alpha\alpha} \hat{\alpha}^p + F^{\alpha\beta} r^p = F^{\alpha\alpha} \hat{\alpha}^p \quad 2$$

$$R_- = F^{\alpha\alpha} R (F^{\alpha\alpha})^* + Q^{\alpha\alpha} \quad 2 \times 2$$

$$\hat{r}_-^p = F^{\beta\alpha} \hat{\alpha}^p + F^{\beta\beta} r^p = F^{\beta\alpha} \hat{\alpha}^p + r^p \quad 1$$

$$\Xi = F^{\beta\alpha} R (F^{\beta\alpha})^* + Q^{\beta\beta} \quad 1 \times 1$$

$$r_-^p \sim \mathcal{N}(\hat{r}_-^p, \Xi)$$

$$S = F^{\alpha\alpha} R (F^{\beta\alpha})^* + Q^{\alpha\beta} \quad 2 \times 1$$

$$\hat{\alpha}^p = \hat{\alpha}_-^p + S \Xi^{-1} (r_-^p - \hat{r}_-^p) = \hat{\alpha}_-^p + \frac{1}{\Xi} (r_-^p - \hat{r}_-^p) S \quad 2$$

$$R = R_- - S \Xi^{-1} S^* = R_- - \frac{1}{\Xi} S S^* \quad 2 \times 2$$

initialisation RB

- particules $r_{-}^{[x|y],1:P} \leftarrow r_{\min}^{[x|y]} + (r_{\max}^{[x|y]} - r_{\min}^{[x|y]}) \mathbf{rand}(1, P)$
- moyennes $\hat{\alpha}_{-}^{[x|y],1:P}$
- covariances $R_{-}^{[x|y]}$

algorithme

initialisation

$$r_{-}^{[x|y],1:P} \leftarrow r_{\min}^{[x|y]} + (r_{\max}^{[x|y]} - r_{\min}^{[x|y]}) \mathbf{rand}(1, P)$$

$$\alpha_{-}^{[x|y],1:P} \leftarrow \begin{pmatrix} a_{\min}^{[x|y]} + (a_{\max}^{[x|y]} - a_{\min}^{[x|y]}) \mathbf{rand}(1, P) \\ v_{\min}^{[x|y]} + (v_{\max}^{[x|y]} - v_{\min}^{[x|y]}) \mathbf{rand}(1, P) \end{pmatrix}$$

$$R_{-}^{[x|y]} \leftarrow \dots$$

pour $k = 1 : K$ **faire**

vraisemblance

$$\omega^p \propto \tilde{L}(Y_k | r_-^p) \quad p = 1 : P \text{ (+normalisation)}$$

sélection

$$\text{offsprings} \leftarrow \text{f_resample}(\omega^{1:P})$$

$$\text{indices} \leftarrow \text{f_resample_indices}(\text{offsprings})$$

$$r^{[x|y],1:P} \leftarrow r_{\text{indices}(1:P)}^{[x|y]}$$

$$\alpha^{[x|y],1:P} \leftarrow \alpha_{\text{indices}(1:P)}^{[x|y]}$$

$$R^{[x|y],1:P} \leftarrow R_{\text{indices}(1:P)}^{[x|y]}$$

prédiction

$$\hat{\alpha}_-^p = F^{\alpha\alpha} \hat{\alpha}^p + F^{\alpha\beta} r^p$$

$$R_- = F^{\alpha\alpha} R (F^{\alpha\alpha})^* + Q^{\alpha\alpha}$$

$$\hat{r}_-^p = F^{\beta\alpha} \hat{\alpha}^p + F^{\beta\beta} r^p$$

$$\Xi = F^{\beta\alpha} R (F^{\beta\alpha})^* + Q^{\beta\beta}$$

$$r_-^p \sim N(\hat{r}_-^p, \Xi)$$

$$S = F^{\alpha\alpha} R (F^{\beta\alpha})^* + Q^{\alpha\beta}$$

$$\hat{\alpha}^p = \hat{\alpha}_-^p + S \Xi^{-1} (r_-^p - \hat{r}_-^p)$$

$$R = R_- - S \Xi^{-1} S^*$$

fin