Olasılıksal Robotik

Dr. Öğr. Üyesi Erkan Uslu

8

Konum Belirleme Problemleri - 1

- **Pozisyon Takibi (Position Tracking):**
  - Önceki konum biliniyor
  - Lokal bir konum belirlemedir
  - Unimodal dağılım ile modellemeye uygun
- **Global Konum Belirleme:**
  - Başlangıç konumu bilinmiyor
  - Unimodal dağılım ile modellemeye uygun değil
- **Kaçırılmış Robot Problemi:**

Konum Belirleme Problemleri - 2

- **Statik Ortamda Konum Belirleme:**
  - Tek durum değişkeni: robot konumu
- **Dinamik Ortamda Konum Belirleme:**
  - Ortamda hareket eden robot ve dinamik objeler var
  - İki şekilde ele alınabilir
    - Durum değişkenleri: Dinamik obje konumları + robot konumu
    - Filtreleme ile dinamik objeler elenebilir

Konum Belirleme Problemleri - 3

- **Pasif Konum Belirleme:**
  - Konum belirleme arkada çalışır
- **Aktif Ortamda Konum Belirleme:**
  - Robot konum belirlemeyi iyileştirecek şekilde kontrol edilir

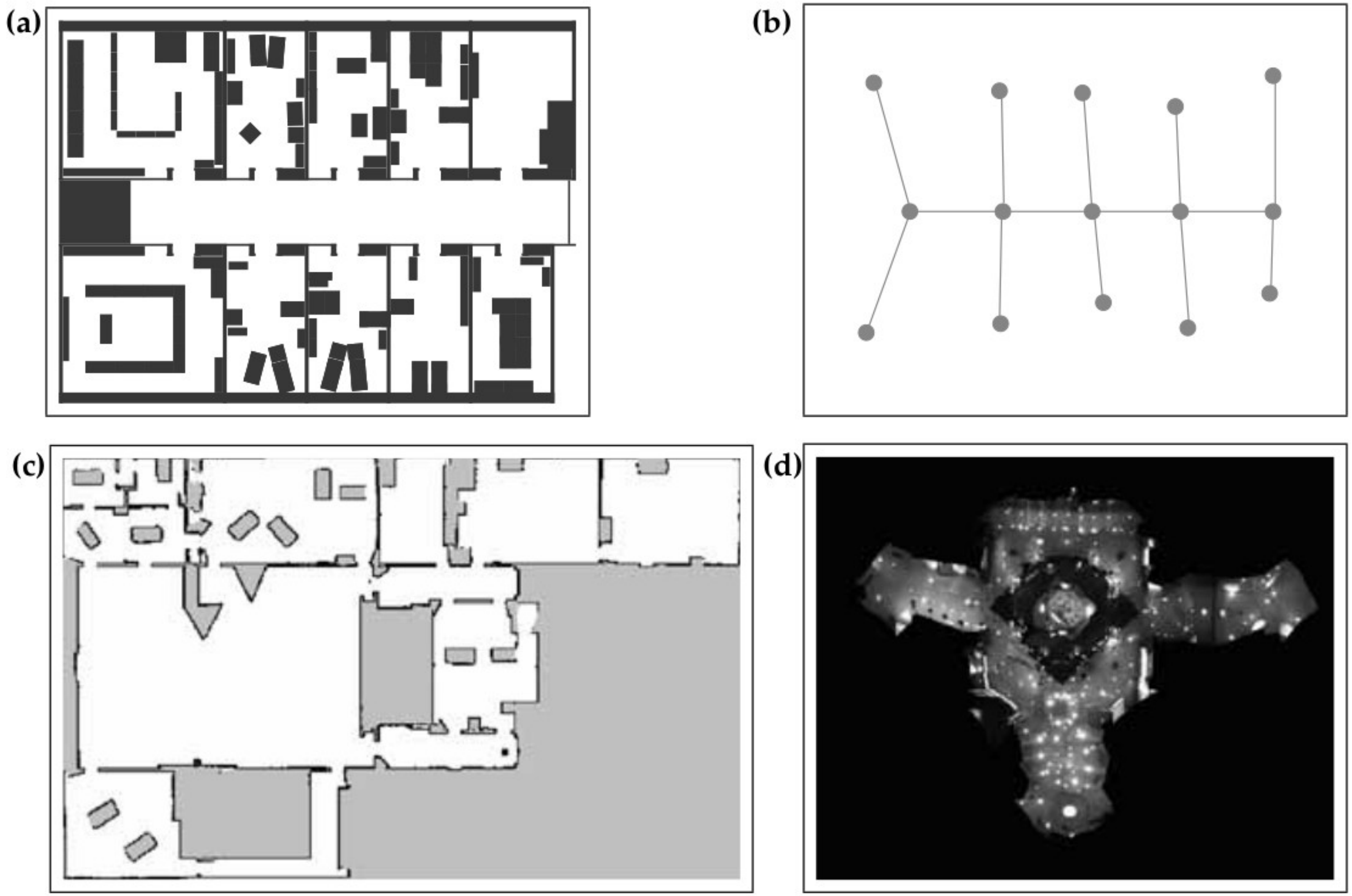
- **Simetrik bir koridorda kumanda ile gezdirilen robot için 2 lokal min oluşur (Pasif). Aktif konum belirlemede robot hatayı azaltmak amacıyla bir odaya girerek konumunu iyileştirebilir.**

Konum Belirleme Problemleri - 4

- **Tek Robotlu Konum Belirleme:**
  - Veri tek platformda alınır ve haberleşme yükü yok
- **Çok Robotlu Konum Belirleme:**
  - N adet tek robotlu konum problemi olarak çözülebilir
  - Robotlar birbirlerini algılayabilirlerse konum iyileştime yapılablilir

Konum Belirleme Problemleri - 5

- **Ortam temsili**



Konum Belirleme

- **Markov konum belirleme**
- **EKF konum belirleme**
- **Çok hipotezli konum belirleme → Gaussian mixture model**
- **UKF konum belirleme**
- **Grid Kkonum belirleme → histogram filtresi**
- **Monte Carlo konum belirleme → parçacık filtresi**

Markov Konum Belirleme

Algorithm Markov_localization($bel(x_{t-1}), u_t, z_t, m$**):**

for all x_t *do*

$\overline{bel}(x_t) = \int p(x_t \mid u_t, x_{t-1}, m)\, bel(x_{t-1})\, dx_{t-1}$

$bel(x_t) = \eta\, p(z_t \mid x_t, m)\, \overline{bel}(x_t)$

endfor

return $bel(x_t)$

EKF Konum Belirleme

- **Varsayımlar:**
  - Feature-based map (noktasal landmarklar)
  - Range-bearing measurement model
  - Velocity motion model
  - Known correspondance

EKF Konum Belirleme

- **Durum değişkeni:**

$$\mathbf{X} = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix}$$

- **Kontol işareti:**

$$u = \begin{pmatrix} \nu \\ \omega \end{pmatrix}$$

- **Gerçekleşen kontrol:**

$$\hat{u} = \begin{pmatrix} \hat{\nu} \\ \hat{\omega} \end{pmatrix} = \begin{pmatrix} \nu \\ \omega \end{pmatrix} + \begin{pmatrix} \varepsilon_{\alpha_1 \nu^2 + \alpha_2 \omega^2} \\ \varepsilon_{\alpha_3 \nu^2 + \alpha_4 \omega^2} \end{pmatrix}$$

$$\hat{u} = u + \mathcal{N}\left(0, M_t\right)$$

EKF Konum Belirleme

- **Hareket modeli**

$$\begin{pmatrix} x' \\ y' \\ \theta' \end{pmatrix} = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix} + \begin{pmatrix} -\frac{\nu}{\omega}\sin\theta + \frac{\nu}{\omega}\sin(\theta + \omega \cdot \Delta t) \\ \frac{\nu}{\omega}\cos\theta - \frac{\nu}{\omega}\cos(\theta + \omega \cdot \Delta t) \\ \omega \cdot \Delta t \end{pmatrix} + \mathcal{N}(0, R_t)$$

Noise Free Part

Random noise

$$x_t = g(u_t, x_{t-1}) \qquad \hat{u} = u + \mathcal{N}(0, M_t)$$

EKF Konum Belirleme

$$f(x,y) \approx f(a,b) + f_x(a,b)(x-a) + f_y(a,b)(y-b)$$

$$g(u_t, x_{t-1}) \approx g(\mu_{t_u}, \mu_{t-1_x}) + G_t\left(x_{t-1} - \mu_{t-1_x}\right) + V_t\left(u_t - \mu_{t_u}\right)$$

$$\mathrm{G}_t = \begin{pmatrix} \frac{\partial g_1}{\partial x} & \frac{\partial g_1}{\partial y} & \frac{\partial g_1}{\partial \theta} \\ \frac{\partial g_2}{\partial x} & \frac{\partial g_2}{\partial y} & \frac{\partial g_2}{\partial \theta} \\ \frac{\partial g_3}{\partial x} & \frac{\partial g_3}{\partial y} & \frac{\partial g_3}{\partial \theta} \end{pmatrix} \begin{matrix} \longrightarrow \text{X} \\ \longrightarrow \text{Y} \\ \longrightarrow \text{theta} \end{matrix}$$

$$= \begin{pmatrix} 1 & 0 & -\frac{\nu}{\omega}\cos\mu_{t-1_\theta} + \frac{\nu}{\omega}\cos\left(\mu_{t-1_\theta} + \omega\cdot\Delta t\right) \\ 0 & 1 & -\frac{\nu}{\omega}\sin\mu_{t-1_\theta} + \frac{\nu}{\omega}\sin\left(\mu_{t-1_\theta} + \omega\cdot\Delta t\right) \\ 0 & 0 & 1 \end{pmatrix}$$

EKF Konum Belirleme

$$V_t = \begin{pmatrix} \frac{\partial g_1}{\partial \nu} & \frac{\partial g_1}{\partial \omega} \\ \frac{\partial g_2}{\partial \nu} & \frac{\partial g_2}{\partial \omega} \\ \frac{\partial g_3}{\partial \nu} & \frac{\partial g_m}{\partial \omega} \end{pmatrix}$$

$$= \begin{pmatrix} \frac{-\sin\theta+\sin(\theta+\omega\cdot\Delta t)}{\omega} & \frac{\nu(\sin\theta-\sin(\theta+\omega\cdot\Delta t))}{\omega^2} + \frac{\nu\cos(\theta+\omega\cdot\Delta t)\cdot\Delta t}{\omega} \\ \frac{\cos\theta-\cos(\theta+\omega\cdot\Delta t)}{\omega} & -\frac{\nu(\cos\theta-\cos(\theta+\omega\cdot\Delta t))}{\omega^2} + \frac{\nu\sin(\theta+\omega\cdot\Delta t)\cdot\Delta t}{\omega} \\ 0 & \Delta t \end{pmatrix}$$

EKF Konum Belirleme

1: **Algorithm EKF_localization_known_correspondences**($\mu_{t-1}, \Sigma_{t-1}, u_t, z_t, c_t, m$):

2: $\theta = \mu_{t-1,\theta}$

3: $$G_t = \begin{pmatrix} 1 & 0 & -\frac{v_t}{\omega_t}\cos\theta + \frac{v_t}{\omega_t}\cos(\theta + \omega_t\Delta t) \\ 0 & 1 & -\frac{v_t}{\omega_t}\sin\theta + \frac{v_t}{\omega_t}\sin(\theta + \omega_t\Delta t) \\ 0 & 0 & 1 \end{pmatrix}$$

4: $$V_t = \begin{pmatrix} \frac{-\sin\theta + \sin(\theta+\omega_t\Delta t)}{\omega_t} & \frac{v_t(\sin\theta - \sin(\theta+\omega_t\Delta t))}{\omega_t^2} + \frac{v_t\cos(\theta+\omega_t\Delta t)\Delta t}{\omega_t} \\ \frac{\cos\theta - \cos(\theta+\omega_t\Delta t)}{\omega_t} & -\frac{v_t(\cos\theta - \cos(\theta+\omega_t\Delta t))}{\omega_t^2} + \frac{v_t\sin(\theta+\omega_t\Delta t)\Delta t}{\omega_t} \\ 0 & \Delta t \end{pmatrix}$$

5: $$M_t = \begin{pmatrix} \alpha_1 v_t^2 + \alpha_2 \omega_t^2 & 0 \\ 0 & \alpha_3 v_t^2 + \alpha_4 \omega_t^2 \end{pmatrix}$$

6: $$\bar{\mu}_t = \mu_{t-1} + \begin{pmatrix} -\frac{v_t}{\omega_t}\sin\theta + \frac{v_t}{\omega_t}\sin(\theta + \omega_t\Delta t) \\ \frac{v_t}{\omega_t}\cos\theta - \frac{v_t}{\omega_t}\cos(\theta + \omega_t\Delta t) \\ \omega_t\Delta t \end{pmatrix}$$

7: $\bar{\Sigma}_t = G_t\ \Sigma_{t-1}\ G_t{}^T + V_t\ M_t\ V_t{}^T$

EKF Konum Belirleme

- **($m_{j,x}$, $m_{j,y}$) konumundaki j. LANDMARK için, (x,y,θ) konumundaki robotun elde etmiş olduğu i. Özellik vektörü şu şekilde hesaplanır**

$$\begin{pmatrix} r_t^i \\ \phi_t^i \\ s_t^i \end{pmatrix} = \begin{pmatrix} \sqrt{(m_{j,x}-x)^2 + (m_{j,y}-y)^2} \\ tan^{-1}\left(\frac{m_{j,y}-y}{m_{j,x}-x}\right) - \theta \\ m_{j,s} \end{pmatrix} + \begin{pmatrix} \varepsilon_{\sigma_r^2} \\ \varepsilon_{\sigma_\phi^2} \\ \varepsilon_{\sigma_s^2} \end{pmatrix}$$

- **Burada** $\varepsilon_{\sigma_r^2}, \varepsilon_{\sigma_\phi^2}, \varepsilon_{\sigma_s^2}$ **sıfır ortalamalı** $\sigma_r, \sigma_\phi, \sigma_s$ **standart sapmalı hata terimleridir**

EKF Konum Belirleme

$$h(x_t, j, m) \approx h(\bar{\mu}_t, j, m) + H_t \left(x_t - \bar{\mu}_t\right)$$

$$\mathrm{H}_t = \begin{pmatrix} \frac{\partial h_1}{\partial x} & \frac{\partial h_1}{\partial y} & \frac{\partial h_1}{\partial \theta} \\ \frac{\partial h_2}{\partial x} & \frac{\partial h_2}{\partial y} & \frac{\partial h_2}{\partial \theta} \\ \frac{\partial h_3}{\partial x} & \frac{\partial h_3}{\partial y} & \frac{\partial h_3}{\partial \theta} \end{pmatrix} \begin{array}{l} \longrightarrow \text{R} \\ \longrightarrow \text{Fi} \\ \longrightarrow \text{s} \end{array}$$

$$= \begin{pmatrix} -\frac{m_{j,x} - \bar{\mu}_{t_x}}{\sqrt{q}} & -\frac{m_{j,y} - \bar{\mu}_{t_y}}{\sqrt{q}} & 0 \\ \frac{m_{j,y} - \bar{\mu}_{t_y}}{q} & -\frac{m_{j,x} - \bar{\mu}_{t_x}}{q} & -1 \\ 0 & 0 & 0 \end{pmatrix}$$

$$q = \left(m_{j,x} - \bar{\mu}_{t_x}\right)^2 + \left(m_{j,y} - \bar{\mu}_{t_y}\right)^2$$

EKF Konum Belirleme

$$Q_t = \begin{pmatrix} \sigma_r^2 & 0 & 0 \\ 0 & \sigma_\phi^2 & 0 \\ 0 & 0 & \sigma_s^2 \end{pmatrix}$$

for all observed features $z_t^i = (r_t^i \;\; \phi_t^i \;\; s_t^i)^T$ *do*

$$j = c_t^i$$

$$q = (m_{j,x} - \bar{\mu}_{t,x})^2 + (m_{j,y} - \bar{\mu}_{t,y})^2$$

$$\hat{z}_t^i = \begin{pmatrix} \sqrt{q} \\ \operatorname{atan2}(m_{j,y} - \bar{\mu}_{t,y}, m_{j,x} - \bar{\mu}_{t,x}) - \bar{\mu}_{t,\theta} \\ m_{j,s} \end{pmatrix}$$

$$H_t^i = \begin{pmatrix} -\frac{m_{j,x} - \bar{\mu}_{t,x}}{\sqrt{q}} & -\frac{m_{j,y} - \bar{\mu}_{t,y}}{\sqrt{q}} & 0 \\ \frac{m_{j,y} - \bar{\mu}_{t,y}}{q} & -\frac{m_{j,x} - \bar{\mu}_{t,x}}{q} & -1 \\ 0 & 0 & 0 \end{pmatrix}$$

$$S_t^i = H_t^i \, \bar{\Sigma}_t \, [H_t^i]^T + Q_t$$

$$K_t^i = \bar{\Sigma}_t \, [H_t^i]^T [S_t^i]^{-1}$$

$$\bar{\mu}_t = \bar{\mu}_t + K_t^i (z_t^i - \hat{z}_t^i)$$

$$\bar{\Sigma}_t = (I - K_t^i \, H_t^i) \, \bar{\Sigma}_t$$

endfor

EKF Konum Belirleme

$\mu_t = \bar{\mu}_t$

$\Sigma_t = \bar{\Sigma}_t$

$p_{z_t} = \prod_i \det\left(2\pi S_t^i\right)^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}\,(z_t^i - \hat{z}_t^i)^T [S_t^i]^{-1} (z_t^i - \hat{z}_t^i)\right\}$

return μ_t, Σ_t, p_{z_t}

UKF Konum Belirleme

1: **Algorithm UKF_localization(**$\mu_{t-1}, \Sigma_{t-1}, u_t, z_t, m$**):**

Generate augmented mean and covariance

2: $M_t = \begin{pmatrix} \alpha_1 v_t^2 + \alpha_2 \omega_t^2 & 0 \\ 0 & \alpha_3 v_t^2 + \alpha_4 \omega_t^2 \end{pmatrix}$

3: $Q_t = \begin{pmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\phi^2 \end{pmatrix}$

4: $\mu_{t-1}^a = (\mu_{t-1}^T \quad (0\ \ 0)^T \quad (0\ \ 0)^T)^T$

5: $\Sigma_{t-1}^a = \begin{pmatrix} \Sigma_{t-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & M_t & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & Q_t \end{pmatrix}$

Generate sigma points

6: $\mathcal{X}_{t-1}^a = (\mu_{t-1}^a \quad \mu_{t-1}^a + \gamma\sqrt{\Sigma_{t-1}^a} \quad \mu_{t-1}^a - \gamma\sqrt{\Sigma_{t-1}^a})$

Pass sigma points through motion model and compute Gaussian statistics

7: $\bar{\mathcal{X}}_t^x = g(u_t + \mathcal{X}_t^u, \mathcal{X}_{t-1}^x)$

8: $\bar{\mu}_t = \sum_{i=0}^{2L} w_i^{(m)} \bar{\mathcal{X}}_{i,t}^x$

9: $\bar{\Sigma}_t = \sum_{i=0}^{2L} w_i^{(c)} (\bar{\mathcal{X}}_{i,t}^x - \bar{\mu}_t)(\bar{\mathcal{X}}_{i,t}^x - \bar{\mu}_t)^T$

UKF Konum Belirleme

Predict observations at sigma points and compute Gaussian statistics

$$\bar{\mathcal{Z}}_t = h(\bar{\mathcal{X}}_t^x) + \mathcal{X}_t^z$$

$$\hat{z}_t = \sum_{i=0}^{2L} w_i^{(m)} \bar{\mathcal{Z}}_{i,t}$$

$$S_t = \sum_{i=0}^{2L} w_i^{(c)} (\bar{\mathcal{Z}}_{i,t} - \hat{z}_t)(\bar{\mathcal{Z}}_{i,t} - \hat{z}_t)^T$$

$$\Sigma_t^{x,z} = \sum_{i=0}^{2L} w_i^{(c)} (\bar{\mathcal{X}}_{i,t}^x - \bar{\mu}_t)(\bar{\mathcal{Z}}_{i,t} - \hat{z}_t)^T$$

Update mean and covariance

$$K_t = \Sigma_t^{x,z} \; S_t^{-1}$$

$$\mu_t = \bar{\mu}_t + K_t(z_t - \hat{z}_t)$$

$$\Sigma_t = \bar{\Sigma}_t - K_t \; S_t \; K_t^T$$

$$p_{z_t} = \det\left(2\pi S_t\right)^{-\frac{1}{2}} \exp\left\{-\frac{1}{2} \left(z_t - \hat{z}_t\right)^T S_t^{-1} \left(z_t - \hat{z}_t\right)\right\}$$

return μ_t, Σ_t, p_{z_t}

Grid Konum Belirleme

Algorithm Grid_localization($\{p_{k,t-1}\}, u_t, z_t, m$**):**

for all k *do*

$$\bar{p}_{k,t} = \sum_i p_{i,t-1} \ \textbf{motion\_model}(\text{mean}(\mathbf{x}_k), u_t, \text{mean}(\mathbf{x}_i))$$

$$p_{k,t} = \eta \ \bar{p}_{k,t} \ \textbf{measurement\_model}(z_t, \text{mean}(\mathbf{x}_k), m)$$

endfor

return $\{p_{k,t}\}$

Monte Carlo Konum Belirleme

Algorithm MCL($\mathcal{X}_{t-1}, u_t, z_t, m$)**:**

$\bar{\mathcal{X}}_t = \mathcal{X}_t = \emptyset$

for $m = 1$ *to* M *do*

$x_t^{[m]} = \textbf{sample\_motion\_model}(u_t, x_{t-1}^{[m]})$

$w_t^{[m]} = \textbf{measurement\_model}(z_t, x_t^{[m]}, m)$

$\bar{\mathcal{X}}_t = \bar{\mathcal{X}}_t + \langle x_t^{[m]}, w_t^{[m]} \rangle$

endfor

for $m = 1$ *to* M *do*

draw i *with probability* $\propto w_t^{[i]}$

add $x_t^{[i]}$ *to* $\mathcal{X}_t$

endfor

return $\mathcal{X}_t$

Arttırılmış (Augmented) Monte Carlo Konum Belirleme

Algorithm Augmented_MCL($\mathcal{X}_{t-1}, u_t, z_t, m$)**:**

static $w_{\text{slow}},\ w_{\text{fast}}$

$\bar{\mathcal{X}}_t = \mathcal{X}_t = \emptyset$

for $m = 1$ *to* M *do*

$x_t^{[m]} =$ **sample_motion_model**$(u_t, x_{t-1}^{[m]})$

$w_t^{[m]} =$ **measurement_model**$(z_t, x_t^{[m]}, m)$

$\bar{\mathcal{X}}_t = \bar{\mathcal{X}}_t + \langle x_t^{[m]}, w_t^{[m]} \rangle$

$w_{\text{avg}} = w_{\text{avg}} + \frac{1}{M}\, w_t^{[m]}$

endfor

$w_{\text{slow}} = w_{\text{slow}} + \alpha_{\text{slow}}(w_{\text{avg}} - w_{\text{slow}})$

$w_{\text{fast}} = w_{\text{fast}} + \alpha_{\text{fast}}(w_{\text{avg}} - w_{\text{fast}})$

for $m = 1$ *to* M *do*

with probability $\max\{0.0,\ 1.0 - w_{\text{fast}}/w_{\text{slow}}\}$ *do*

add random pose to $\mathcal{X}_t$

else

draw $i \in \{1, \ldots, N\}$ *with probability* $\propto w_t^{[i]}$

add $x_t^{[i]}$ *to* $\mathcal{X}_t$

endwith

endfor

return $\mathcal{X}_t$

Karşılaştırma

	EKF	MHT	Coarse (topological) grid	fine (metric) grid	MCL
Measurements	landmarks	landmarks	landmarks	raw measurements	raw measurements
Measurement noise	Gaussian	Gaussian	any	any	any
Posterior	Gaussian	mixture of Gaussians	histogram	histogram	particles
Efficiency (memory)	++	++	+	−	+
Efficiency (time)	++	+	+	−	+
Ease of implementation	+	−	+	−	++
Resolution	++	++	−	+	+
Robustness	−	+	+	++	++
Global localization	no	yes	yes	yes	yes