

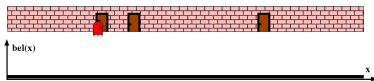
# Pravděpodobnostní metody lokalizace v mobilní robotice

Miroslav Kulich

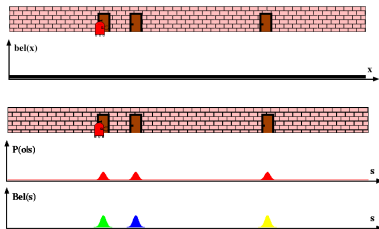
Intelligent and Mobile Robotics Group  
Gerstner Laboratory for Intelligent Decision Making and Control  
Czech Technical University in Prague

27/02/2014

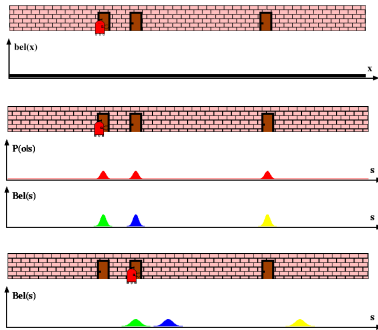
# Motivace



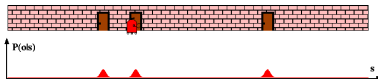
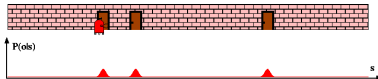
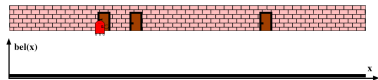
# Motivace



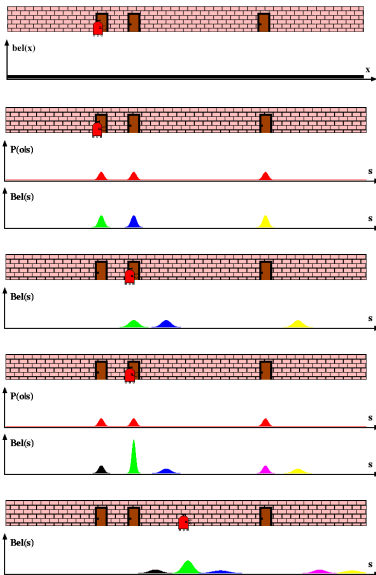
# Motivace



# Motivace



# Motivace



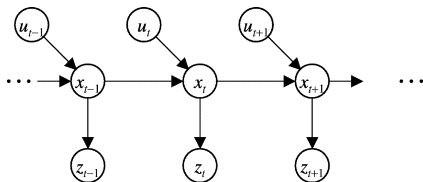
# Bayesův filtr: framework

- Je dáno:
  - Posloupnost pozorování  $z$  a akcí  $u$ :

$$d_t = \{u_1, z_1, \dots, u_t, z_t\}$$

- **Sensorický model**  $p(z|x)$
  - **Model akcí (pohybu)**  $p(x|u, x')$
  - **Počáteční (prior)** pravděpodobnost stavu  $p(x)$
- Chceme:
  - Odhad stavu (**posterior**)
  - Posteriorní pravděpodobnost označujeme jako **belief (víra)**:  
 $Bel(x_t) = p(x_t|u_1, z_1, \dots, u_t, z_t)$

## Markovův předpoklad



Následující stav závisí pouze na aktuálním stavu a nezávisí na stavech předchozích.

$$p(z_t | x_{0:t}, z_{1:t}, u_{1:t}) = p(z_t | x_t)$$

$$p(x_t | x_{1:t-1}, z_{1:t}, u_{1:t}) = p(x_t | x_{t-1}, u_t)$$

S tím souvisí:

- Statický svět (nebo dynamika modelovaná v  $x$ )
- Dokonalý model světa (model senzoru) a model pohybu
- Nezávislý šum
- Přesná reprezentace  $p(x)$



## Bayesův filtr - odvození

$Bel(x_t)$

Bayes

$$= p(x_t | u_1, z_1, \dots, u_t, z_t)$$
$$= \eta p(z_t | x_t, u_1, z_1, \dots, u_t) p(x_t | u_1, z_1, \dots, u_t)$$

Markov

$$= \eta p(z_t | x_t) p(x_t | u_1, z_1, \dots, u_t)$$

Total prob.

$$= \eta p(z_t | x_t) \int p(x_t | u_1, z_1, \dots, u_t, x_{t-1})$$
$$p(x_{t-1} | u_1, z_1, \dots, u_t) dx_{t-1}$$

Markov

$$= \eta p(z_t | x_t) \int p(x_t | u_t, x_{t-1}) p(x_{t-1} | u_1, z_1, \dots, u_t) dx_{t-1}$$

Markov

$$= \eta p(z_t | x_t) \int p(x_t | u_t, x_{t-1}) p(x_{t-1} | u_1, z_1, \dots, z_{t-1}) dx_{t-1}$$

$$= \eta p(z_t | x_t) \int p(x_t | u_t, x_{t-1}) Bel(x_{t-1}) dx_{t-1}$$

## Bayesův filtr

$$Bel(x_t) = \eta p(z_t|x_t) \int p(x_t|u_t, x_{t-1}) Bel(x_{t-1}) dx_{t-1}$$

Algoritmus Bayes\_filter( $Bel(x), d$ )

**if**  $d$  is a **measurement**  $z$  **then**

$$\eta = 0$$

**for** all  $x$  **do**

$$Bel'(x) = p(z|x) Bel(x)$$

$$\eta = \eta + Bel'(x)$$

**end for**

**for** all  $x$  **do**

$$Bel'(x) = \eta^{-1} Bel'(x)$$

**end for**

**end if**

**if**  $d$  is a **action**  $u$  **then**

**for** all  $x$  **do**

$$Bel'(x) =$$

$$\int p(x|u, x') Bel(x') dx'$$

**end for**

**end if**

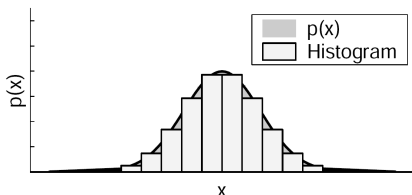
**return**  $Bel'(x)$

# Bayesovy filtry - celá rodina metod

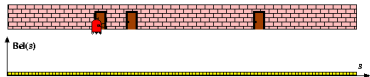
- Kalmanův filtr
- Histogramový filtr
- Částicový (Particle) filtr
- Skryté Markovovy modely
- Dynamické Bayesovy sítě
- Částečně pozorovatelné Markovovy rozhodovací procesy (POMDPs)

# Neparametrické filtry

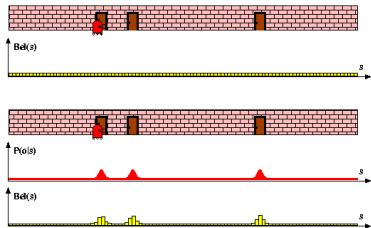
- Aproximace hustoty pravděpodobnosti konečnou množinou hodnot
- Umí reprezentovat libovolné rozložení
- Velikost množiny reprezentantů ovlivňuje rychlost výpočtu a kvalitu filtrace
- Histogramový x částicový filtr



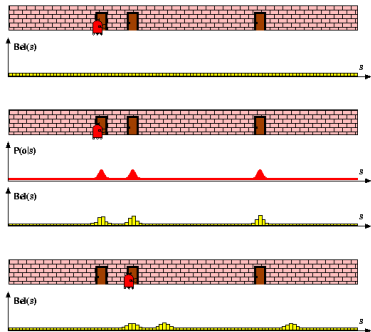
# Histogramový filtr



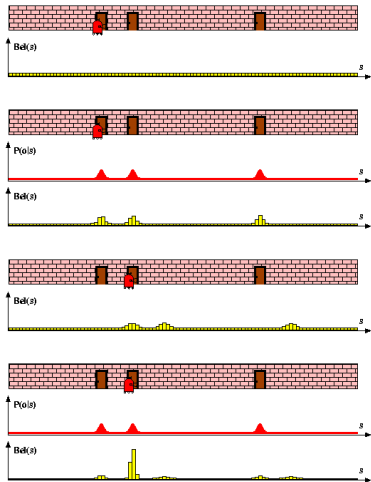
# Histogramový filtr



# Histogramový filtr

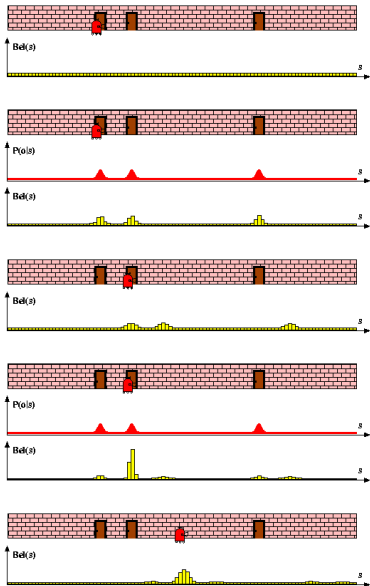


# Histogramový filtr





# Histogramový filtr



## Histogramový filtr

if  $d$  is a measurement  $z$  then

$$\eta = 0$$

for all  $x$  do

$$Bel'(x) = p(z|x)Bel(x)$$

$$\eta = \eta + Bel'(x)$$

end for

for all  $x$  do

$$Bel'(x) = \eta^{-1} Bel'(x)$$

end for

else if  $d$  is an action  $u$  then

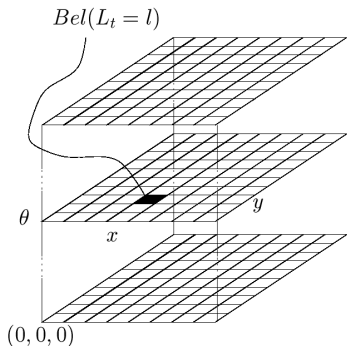
for all  $x$  do

$$Bel'(x) = \int p(x|u, x')Bel(x')dx'$$

end for

end if

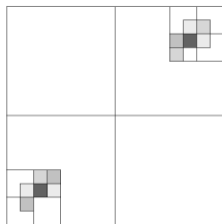
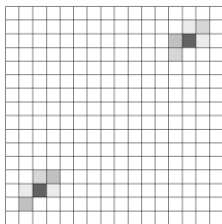
return  $Bel'(x)$



$$Bel(x_t = \langle x, y, \phi \rangle)$$

## Histogramový filtr

- Složitost  $O(n^2)$  – při aktualizaci z modelu pohybu se prochází každá buňka!
- Selektivní update
  - uvažuje se jen část stavového prostoru
  - ale pak je potřeba sledovat, jak úspěšně je robot lokalizován
- Dynamická dekompozice stavového prostoru – kd-stromy (density trees): „jemnost“ dělení závisí na hustotě pravděpodobnosti (vyšší hustota => jemnější dělení)

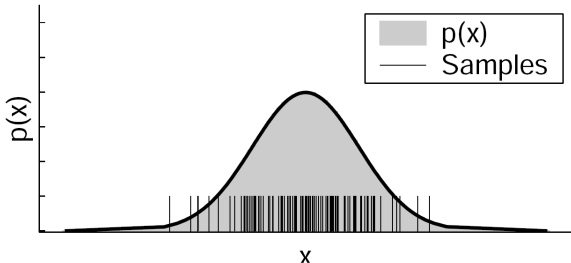


# Částicový filtr (particle filter)

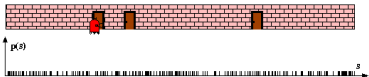
- Hustota pravděpodobnosti je reprezentována „vhodně“ (náhodně umístěnými vzorky):

$$Bel(x_t) \approx \{x^{(i)}, w^{(i)}\}_{i=1, \dots, m}$$

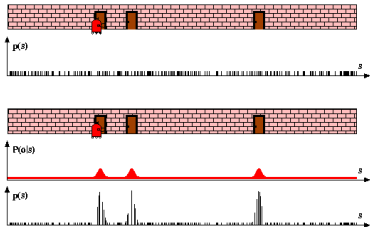
- Vzorky jsou vážené.
- Podle váhy v čase  $t - 1$  se vybírají vzorky pro čas  $t$
- Velice jednoduchý na implementaci
- Nejuniverzálnější z BF: reprezentace **ne-normálního** rozložení a **nelineárních** procesů



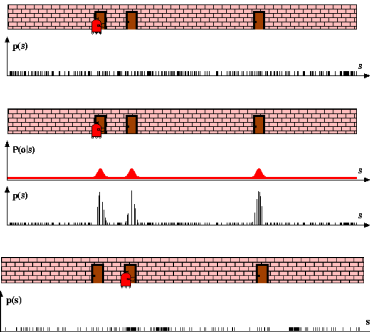
# Částicový filtr



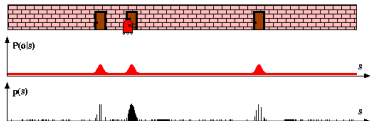
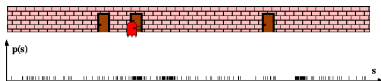
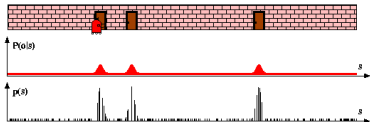
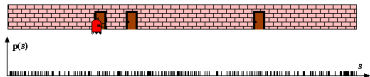
# Částicový filtr



# Částicový filtr

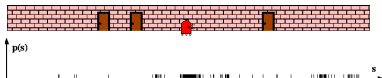
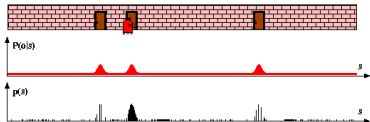
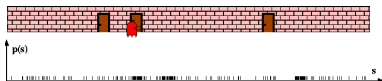
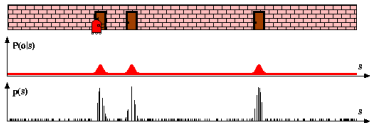
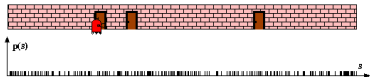


# Částicový filtr





# Částicový filtr



# Algoritmus částicového filtru

Particle\_filter( $S_{t-1}, u_{t-1}, z_t$ )

$S_t = \emptyset, \eta = 0$

**for**  $i = 1, \dots, n$  **do**

Generuj nové vzorky

Vyber index  $j_i$  podle diskrétní distribuce dané  $w_{t-1}$

Vyber  $x_t^i$  z  $p(x_t | x_{t-1}, u_{t-1})$  na základě  $x_{t-1}^{j_i}$  a  $u_{t-1}$

$w_t^i = p(z_t | x_t^i)$

Spočítej váhu

$\eta = \eta + w_t^i$

Aktualizuj normalizační faktor

$S_t = S_t \cup \{ \langle x_t^i, w_t^i \rangle \}$

Přidej vzorek

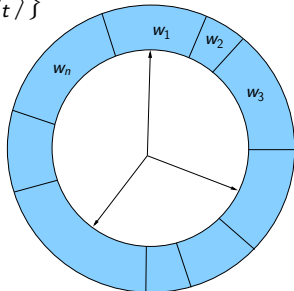
**end for**

**for**  $i = 1, \dots, n$  **do**

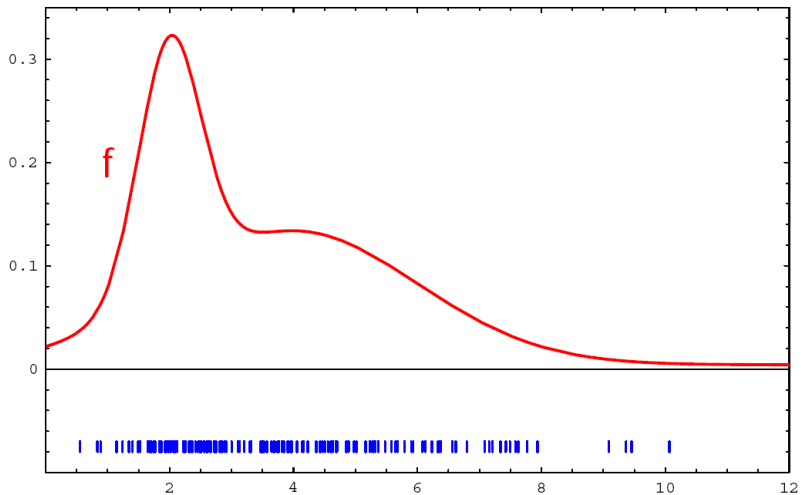
$w_t^i = \frac{w_t^i}{\eta}$

Normalizuj váhy

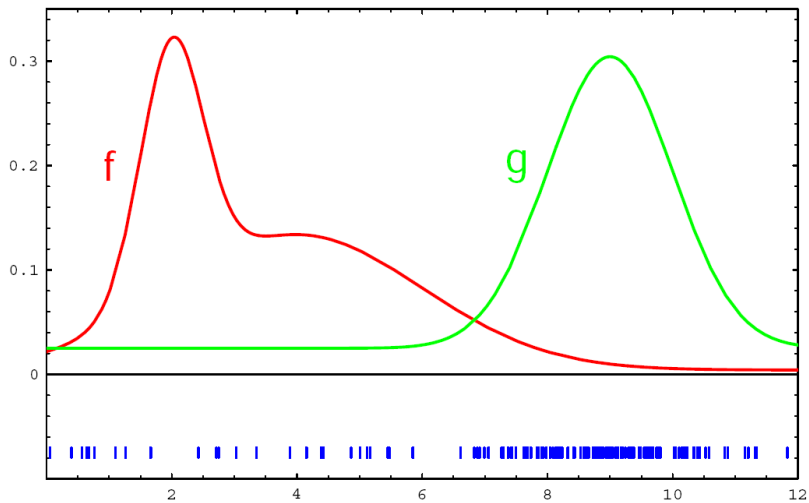
**end for**



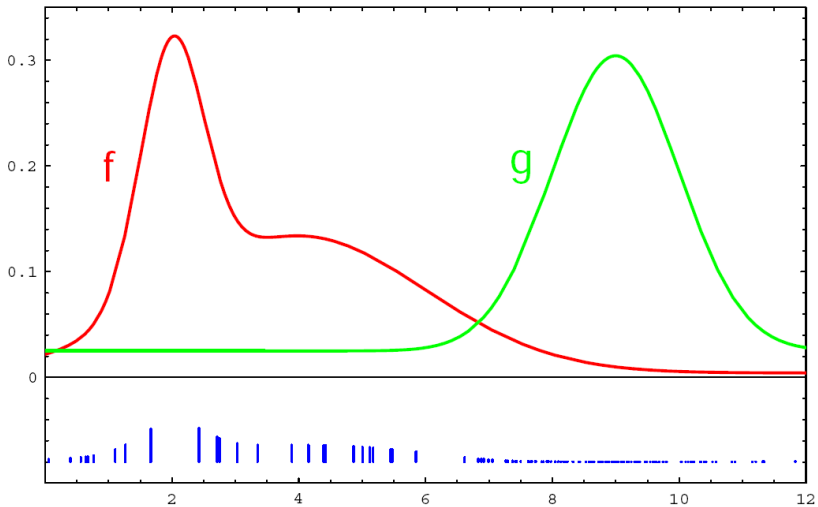
# Proč se váží?



## Proč se váží?



# Proč se váží?



# Kalmanův filtr

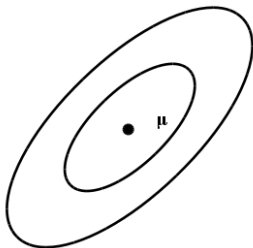
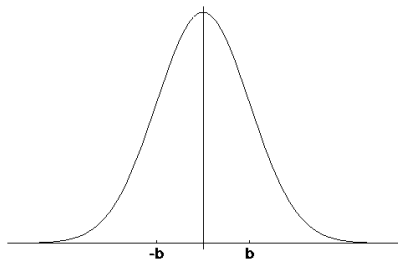
- Unimodální reprezentace hustoty pp. Gaussiánem

$$p(x) \sim N(\mu, \sigma^2)$$

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

$$p(x) \sim N(\mu, \Sigma)$$

$$p(x) = \frac{1}{\sqrt{2\pi|\Sigma|}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)}$$



## Lineární transformace

- Lineární transformace zachovávají normální rozdělení

$$\left. \begin{array}{l} X \sim N(\mu, \sigma^2) \\ Y = aX + b \end{array} \right\} \Rightarrow Y \sim N(a\mu + b, a^2\sigma^2)$$

$$\left. \begin{array}{l} X_1 \sim N(\mu_1, \sigma_1^2) \\ X_2 \sim N(\mu_2, \sigma_2^2) \end{array} \right\} \Rightarrow p(X_1)p(X_2) \sim N\left(\frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2}\mu_1 + \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2}\mu_2, \frac{1}{\sigma_1^{-2} + \sigma_2^{-2}}\right)$$

$$\left. \begin{array}{l} X \sim N(\mu, \Sigma) \\ Y = AX + B \end{array} \right\} \Rightarrow Y \sim N(A\mu + B, A\Sigma A^T)$$

$$\left. \begin{array}{l} X_1 \sim N(\mu_1, \Sigma_1) \\ X_2 \sim N(\mu_2, \Sigma_2) \end{array} \right\} \Rightarrow p(X_1)p(X_2) \sim N\left(\frac{\Sigma_2}{\Sigma_1 + \Sigma_2}\mu_1 + \frac{\Sigma_1}{\Sigma_1 + \Sigma_2}\mu_2, \frac{1}{\Sigma_1^{-1} + \Sigma_2^{-1}}\right)$$

# Základní předpoklady

- Musí platit Markovova podmínka + tři následující:
  - Pravděpodobnost přechodu stavu (model pohybu)  $p(x|u, x')$  je lineární s přidaným Gausovým šumem:

$$x_t = Ax_{t-1} + B_t u_t + \varepsilon_t$$

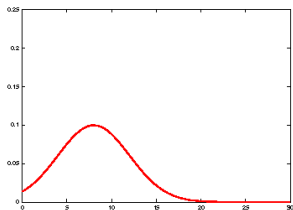
- Model senzoru musí být lineární s Gausovým šumem

$$z_t = Cx_t + \delta_t$$

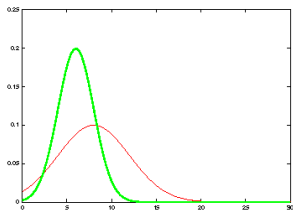
- Počáteční informace (víra) musí mít normální rozložení



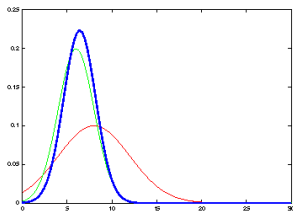
# Kalmanův filtr



Apriorní znalost

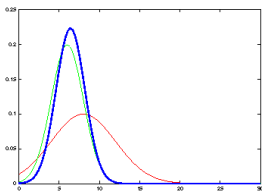


Nové měření

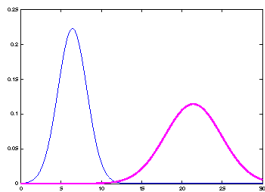


Integrace nového měření

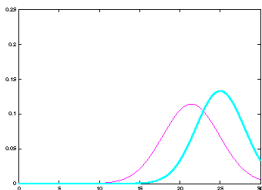
# Kalmanův filtr



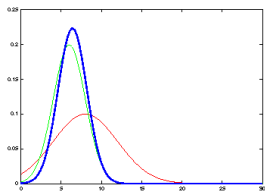
Aktuální stav



Provedení akce



Nové měření



Integrace nového měření

# Algoritmus

Algoritmus Kalman\_filter( $\mu_{t-1}, \Sigma_{t-1}, u_t, z_t$ )

## Predikce (integrace akce)

$$\bar{\mu}_t = A_t \mu_{t-1} + B_t u_t$$

$$\bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + R_t$$

## Korekce (integrace měření)

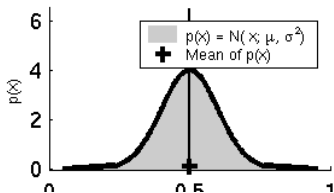
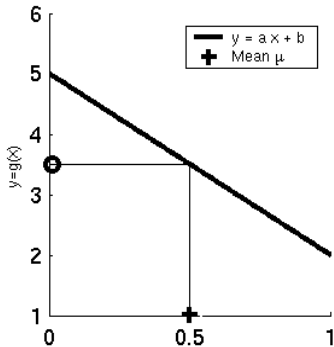
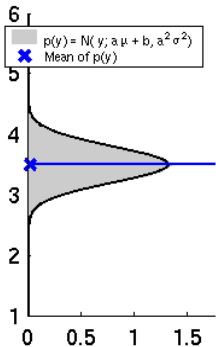
$$K_t = \bar{\Sigma}_t C_t^T (C_t \bar{\Sigma}_t C_t^T + Q_t)^{-1}$$

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

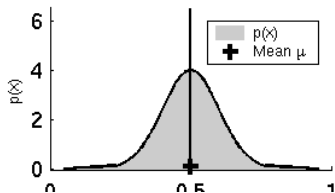
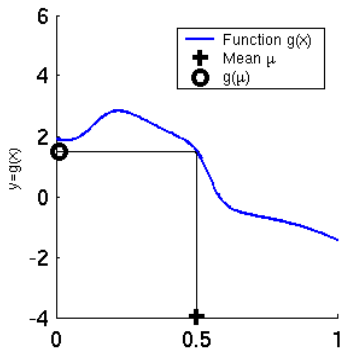
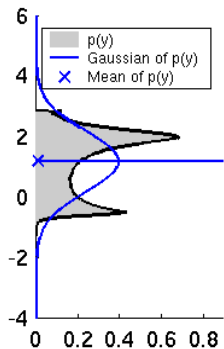
$$\Sigma_t = (E - K_t C_t) \bar{\Sigma}_t$$

**return**  $\mu_t, \Sigma_t$

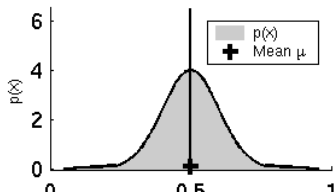
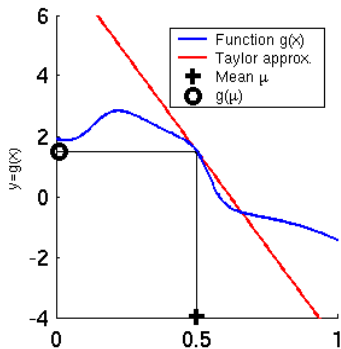
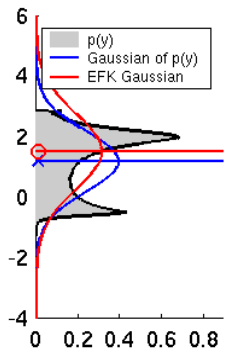
# Lineární transformace Kalmanova filtru



# Nelineární transformace



# Rozšířený Kalmanův filtr



$$x_t = g(u_t, x_{t-1})$$

$$z_t = h(x_t)$$

## Rozšířený Kalmanův filtr

- Pro linearizaci používáme Taylorův rozvoj v  $\mu$ , tj. matice derivací funkcí – Jakobiány
- Predikce:

$$g(u_t, x_t - 1) \approx g(u_t, \mu_{t-1}) + \frac{\partial g(u_t, \mu_{t-1})}{\partial x_{t-1}} (x_{t-1} - \mu_{t-1})$$
$$g(u_t, x_t - 1) \approx g(u_t, \mu_{t-1}) + G_t (x_{t-1} - \mu_{t-1})$$

- Korekce:

$$h(x_t) \approx h(\bar{\mu}_t) + \frac{\partial h(\bar{\mu}_t)}{\partial x_t} (x_t - \bar{\mu}_t)$$
$$h(x_t) \approx h(\bar{\mu}_t) + H_t (x_t - \bar{\mu}_t)$$

# Algoritmus

Algoritmus Extended\_Kalman\_filter( $\mu_{t-1}, \Sigma_{t-1}, u_t, z_t$ )

## Predikce (integrace akce)

$$\bar{\mu}_t = g(u_t, \mu_{t-1})$$

$$\bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + R_t$$

## Korekce (integrace měření)

$$K_t = \bar{\Sigma}_t H_t^T (H_t \bar{\Sigma}_t H_t^T + Q_t)^{-1}$$

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

$$\Sigma_t = (E - K_t H_t) \bar{\Sigma}_t$$

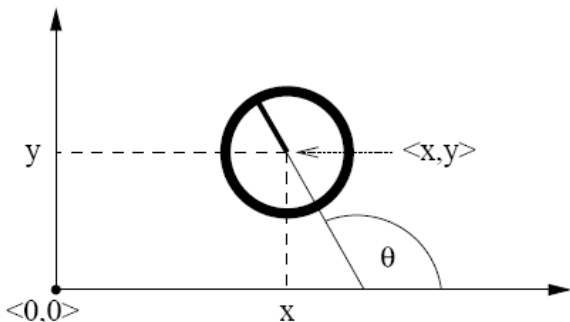
$$H_t = \frac{\partial h(\bar{\mu}_t)}{\partial x_t}$$

$$G_t = \frac{\partial g(u_t, \mu_{t-1})}{\partial x_{t-1}}$$



## Model pohybu

- Pro implementaci Bayesova filtru potřebujeme definovat model pohybu  $p(x|x', u)$ .
- Model pohybu definuje pravděpodobnost, že po provedení akce  $u$  ve stavu  $x'$  bude robot ve stavu  $x$ . Předpokládáme robot pohybující se v rovině, tj  $x = \langle x, y, \phi \rangle$
- Existují různé modely (v závislosti na typu řízení, zahrnutí kinematiky, atd.)



# Odometrický model pohybu

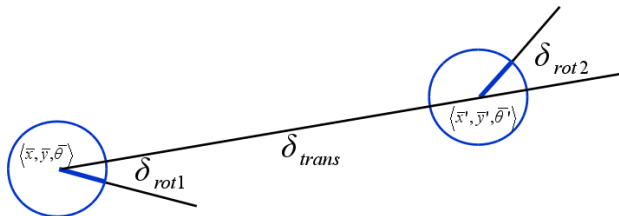
## Ideální případ

- Robot se pohybuje z  $\langle \bar{x}, \bar{y}, \bar{\phi} \rangle$  do  $\langle \bar{x}', \bar{y}', \bar{\phi}' \rangle$
- Z odometrie získáme  $u = \langle \delta_{rot1}, \delta_{rot2}, \delta_{trans} \rangle$

$$\delta_{trans} = \sqrt{(\bar{x}' - \bar{x})^2 + (\bar{y}' - \bar{y})^2}$$

$$\delta_{rot1} = \text{atan2}(\bar{y}' - \bar{y}, \bar{x}' - \bar{x}) - \bar{\phi}$$

$$\delta_{rot2} = \bar{\phi}' - \bar{\phi} - \delta_{rot1}$$



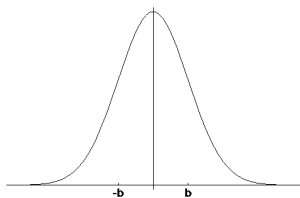
## Přidáme šum

$$\begin{aligned}\hat{\delta}_{rot_1} &= \delta_{rot_1} + \varepsilon_{\alpha_1}|\delta_{rot_1}| + \alpha_2|\delta_{trans}| \\ \hat{\delta}_{trans} &= \delta_{trans} + \varepsilon_{\alpha_3}|\delta_{trans}| + \alpha_4(|\delta_{rot_1}| + |\delta_{rot_2}|) \\ \hat{\delta}_{rot_2} &= \delta_{rot_2} + \varepsilon_{\alpha_1}|\delta_{rot_2}| + \alpha_2|\delta_{trans}|\end{aligned}$$

- Šum je dán čtyřmi parametry
- Na celém modelu je nejtěžší získat právě parametry šumu – experimentálně

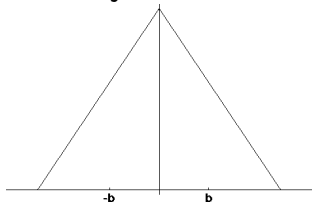
# Typické distribuce šumu

Normální



$$\varepsilon_{\sigma^2}(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}}$$

Trojúhelníková



$$\varepsilon_{\sigma^2}(x) = \begin{cases} 0 & \text{pokud } |x| > \sqrt{6\sigma^2} \\ \frac{\sqrt{6\sigma^2} - |x|}{6\sigma^2} & \text{jinak} \end{cases}$$

# Odometrický model pro vzorkování

$$u = \langle \delta_{rot1}, \delta_{rot2}, \delta_{trans} \rangle, x = \langle x, y, \phi \rangle \Rightarrow x' = \langle x', y', \phi' \rangle$$

## Náhodné řízení

$$\begin{aligned}\hat{\delta}_{rot1} &= \delta_{rot1} + \text{sample}(\alpha_1 |\delta_{rot1}| + \alpha_2 |\delta_{trans}|) \\ \hat{\delta}_{trans} &= \delta_{trans} + \text{sample}(\alpha_3 |\delta_{trans}| + \alpha_4 (|\delta_{rot1}| + |\delta_{rot2}|)) \\ \hat{\delta}_{rot2} &= \delta_{rot2} + \text{sample}(\alpha_1 |\delta_{rot2}| + \alpha_2 |\delta_{trans}|)\end{aligned}$$

## Výpočet pozice

$$\begin{aligned}x' &= x + \hat{\delta}_{trans} + \cos(\phi + \hat{\delta}_{rot1}) \\ y' &= x + \hat{\delta}_{trans} + \sin(\phi + \hat{\delta}_{rot1}) \\ \phi' &= \phi + \hat{\delta}_{rot1} + \hat{\delta}_{rot2} \\ &\text{return } \langle x', y', \phi' \rangle\end{aligned}$$

**sample** (normální rozložení):  $\frac{1}{2} \sum_{i=1}^{12} \text{rand}(-b, b)$

# Výpočet $p(x|u, x')$

## Hodnoty z odometry ( $u$ )

$$\delta_{trans} = \sqrt{(x' - \bar{x})^2 + (y' - \bar{y})^2}$$

$$\delta_{rot1} = \text{atan2}(y' - \bar{y}, x' - \bar{x}) - \bar{\phi}$$

$$\delta_{rot2} = \phi' - \bar{\phi} - \delta_{rot1}$$

## Řízení pro $x$

$$\hat{\delta}_{trans} = \sqrt{(x' - x)^2 + (y' - y)^2}$$

$$\hat{\delta}_{rot1} = \text{atan2}(y' - y, x' - x) - \phi$$

$$\hat{\delta}_{rot2} = \phi' - \phi - \delta_{rot1}$$

## Výpočet pravděpodobnosti

$$p_1 = \text{prob}(\delta_{rot1} - \hat{\delta}_{rot1}, \alpha_1 |\delta_{rot1}| + \alpha_2 \delta_{trans})$$

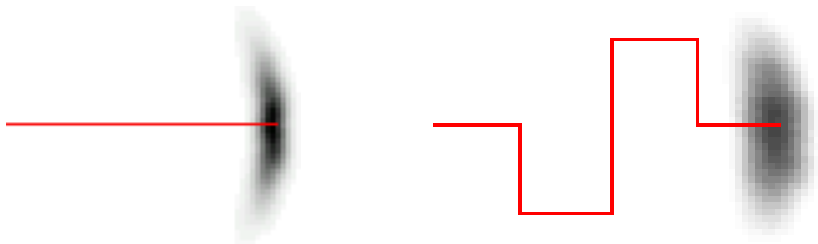
$$p_2 = \text{prob}(\delta_{trans} - \hat{\delta}_{trans}, \alpha_3 \delta_{trans} + \alpha_4 (|\delta_{rot1}| + |\delta_{rot2}|))$$

$$p_3 = \text{prob}(\delta_{rot2} - \hat{\delta}_{rot2}, \alpha_1 |\delta_{rot2}| + \alpha_2 \delta_{trans})$$

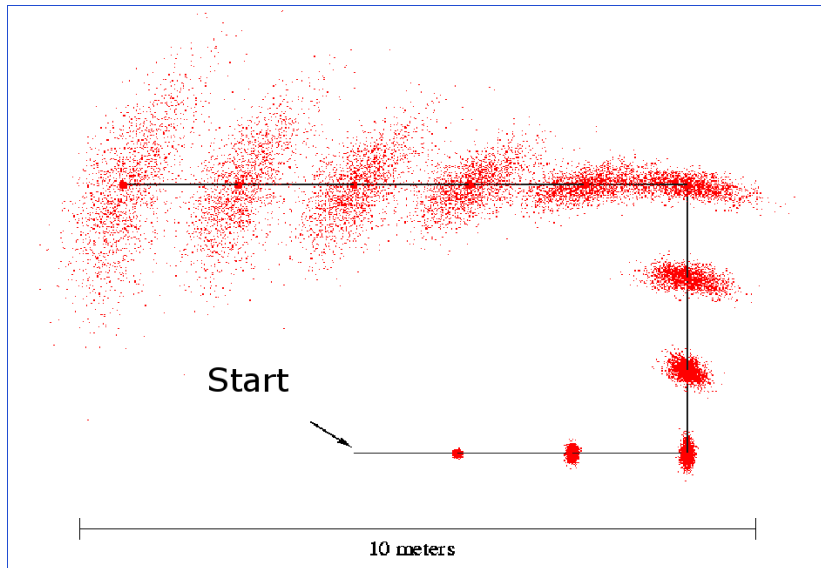
return  $p_1 p_2 p_3$  ← předpokládáme nezávislost

# Aplikace

- Výsledná hustota pravděpodobnosti závisí na ujeté trajektorii, ne pouze na výsledné pozici!
- Pro složitější pohyby se opakovaně provádí předchozí algoritmus.
- Typický případ, jak vypadá distribuce (2D projekce)



## Skládání trajektorií

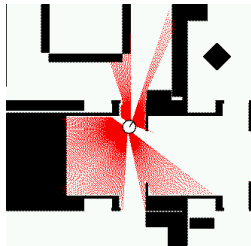
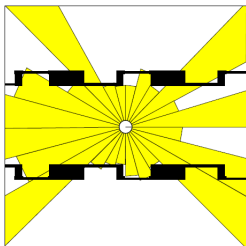




## Model senzoru

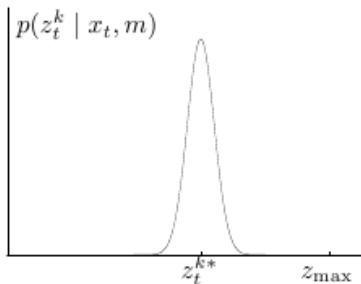
- Cílem je určit  $p(z|m, x)$ .
- Budeme používat proximní senzory (laser).
- Scan se skládá z  $k$  měření (paprsků):  $z = \{z_1, z_2, \dots, z_k\}$
- Předpokládáme, že paprsky jsou nezávislé:

$$P(z|x, m) = \prod_{k=1}^k p(z_k|x, m)$$



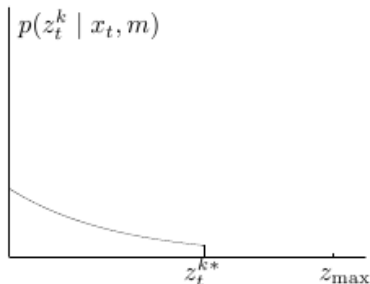
# Paprskový model - komponenty

## Normální rozložení



$$p_{hit}(z|x, m) = \begin{cases} \eta N(z, z^*, \sigma_{hit}^2) & \text{pokud } 0 \leq z \leq z_{\max} \\ 0 & \text{jinak} \end{cases}$$

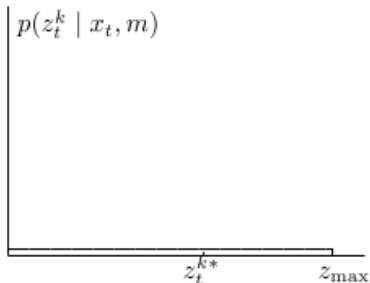
## Poissonovo rozložení



$$p_{short}(z|x, m) = \begin{cases} \eta \lambda_{short} e^{-\lambda_{short} z} & \text{pokud } 0 \leq z \leq z^* \\ 0 & \text{jinak} \end{cases}$$

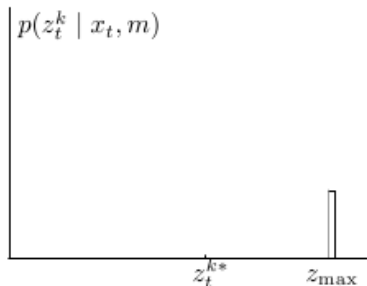
# Paprskový model - komponenty

Rovnoměrné rozložení



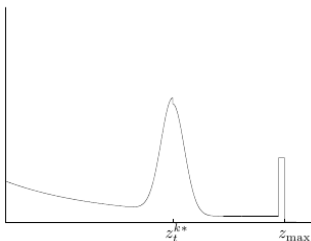
$$p_{\text{rand}}(z|x, m) = \begin{cases} \frac{1}{z_{\max}} & \text{pokud } 0 \leq z \leq z_{\max} \\ 0 & \text{jinak} \end{cases}$$

Diskrétní rozložení :-)



$$p_{\text{max}}(z|x, m) = \begin{cases} 1 & \text{pokud } z = z_{\max} \\ 0 & \text{jinak} \end{cases}$$

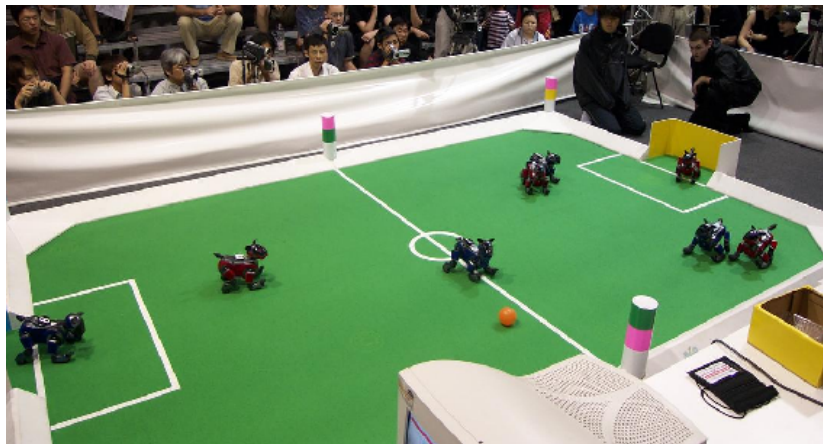
## Sloučení



$$p(z|x, m) = \begin{pmatrix} \alpha_{hit} \\ \alpha_{short} \\ \alpha_{rand} \\ \alpha_{max} \end{pmatrix}^T \begin{pmatrix} p_{hit}(z|x, m) \\ p_{short}(z|x, m) \\ p_{rand}(z|x, m) \\ p_{max}(z|x, m) \end{pmatrix}$$

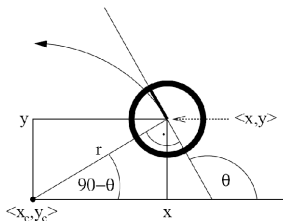
- Parametry modelu se určí nějakým učícím algoritmem na základě naměřených dat (E-M, genetické algoritmy)
- $z_{exp}$  se počítá raytracing, což je časově náročné.
- Lze počítat jen některé paprsky (např. 8); navíc větší nezávislost.
- Lze použít předpočítané mapy (pro každé  $\langle x, y, z \phi \rangle$ )
- Vynásobením modelu senzoru  $\lambda < 1$  omezíme vliv senzoru.
- Model je nespojitý  $\Rightarrow$  aproximační vyjádření hustoty pp. může minout daný stav.

# Motivace



# EKF lokalizace

- Rychlostní model pohybu



$$\underbrace{\begin{pmatrix} x' \\ y' \\ \theta' \end{pmatrix}}_{x_t} = \underbrace{\begin{pmatrix} x_c + \frac{v}{\omega} \sin(\theta + \omega \Delta t) \\ y_c + \frac{v}{\omega} \cos(\theta + \omega \Delta t) \\ \theta + \omega \Delta t \end{pmatrix}}_{g(u_t, x_{t-1})} + N(0, M_t)$$

$$\begin{pmatrix} x_c \\ y_c \end{pmatrix} = \begin{pmatrix} x - \frac{v}{\omega} \sin \theta \\ y + \frac{v}{\omega} \cos \theta \end{pmatrix}$$

# EKF lokalizace

- Mapa (seznam příznaků  $\langle m_{j,x}, m_{j,y} \rangle$ )

$$\underbrace{\begin{pmatrix} r_i \\ \phi_i \end{pmatrix}}_{z_t^i} = \underbrace{\begin{pmatrix} \sqrt{(m_{j,x} - x)^2 + (m_{j,y} - y)^2} \\ \text{atan2}(m_{j,y} - y, m_{j,x} - x) - \theta \end{pmatrix}}_{h(x_t, j, m)} + N(0, Q_t)$$

Korespondence jsou známé!

# Algoritmus

Algoritmus Extended\_Kalman\_filter( $\mu_{t-1}, \Sigma_{t-1}, u_t, z_t$ )

## Predikce (integrace akce)

$$\begin{aligned}\bar{\mu}_t &= g(u_t, \mu_{t-1}) \\ \bar{\Sigma}_t &= G_t \Sigma_{t-1} G_t^T + R_t\end{aligned}$$

## Korekce (integrace měření)

$$\begin{aligned}K_t &= \bar{\Sigma}_t H_t^T (H_t \bar{\Sigma}_t H_t^T + Q_t)^{-1} \\ \mu_t &= \bar{\mu}_t + K_t (z_t - h(\bar{\mu}_t))\end{aligned}$$

$$\begin{aligned}H_t &= \frac{\partial h(\bar{\mu}_t)}{\partial x_t} \\ G_t &= \frac{\partial g(u_t, \mu_{t-1})}{\partial x_{t-1}}\end{aligned}$$



# Predikce

$g(u_t, x_{t-1})$

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

Šum pohybu v prostoru řízení

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

Jacobian g vzhledem k pozici

$$G_t = \frac{\partial g(u_t, \mu_{t-1})}{\partial x_{t-1}} = \begin{pmatrix} \frac{\partial x'}{\partial \mu_{t-1,x}} & \frac{\partial x'}{\partial \mu_{t-1,y}} & \frac{\partial x'}{\partial \mu_{t-1,\theta}} \\ \frac{\partial y'}{\partial \mu_{t-1,x}} & \frac{\partial y'}{\partial \mu_{t-1,y}} & \frac{\partial y'}{\partial \mu_{t-1,\theta}} \\ \frac{\partial \theta'}{\partial \mu_{t-1,x}} & \frac{\partial \theta'}{\partial \mu_{t-1,y}} & \frac{\partial \theta'}{\partial \mu_{t-1,\theta}} \end{pmatrix}$$
$$= \begin{pmatrix} 1 & 0 & -\frac{v_t}{\omega_t} \cos(\theta) + \frac{v_t}{\omega_t} \cos(\theta + \omega \Delta t) \\ 0 & 1 & -\frac{v_t}{\omega_t} \sin(\theta) + \frac{v_t}{\omega_t} \sin(\theta + \omega \Delta t) \\ 0 & 0 & 1 \end{pmatrix}$$

# Predikce

Predikovaná střední hodnota

$$\bar{\mu}_t = g(u_t, \mu_{t-1})$$

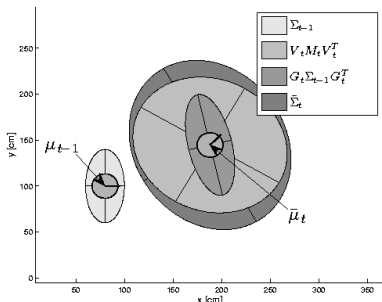
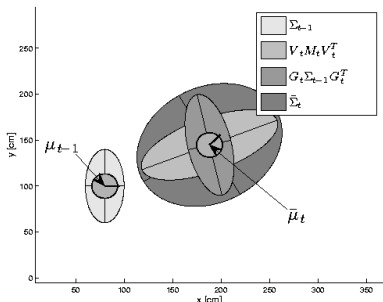
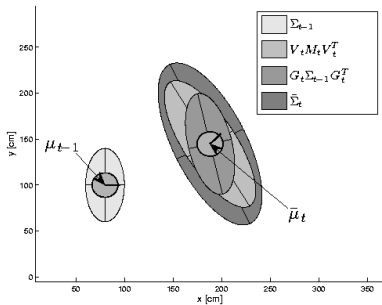
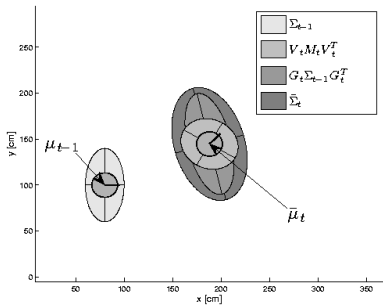
Predikovaná kovariance

$$\bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + \underbrace{V_t M_t V_t^T}_{R_t}$$

Jacobian  $g$  vzhledem k řízení

$$V_t = \frac{\partial g(u_t, \mu_{t-1})}{\partial u_t} = \begin{pmatrix} \frac{\partial x'}{\partial v_t} & \frac{\partial x'}{\partial \omega_t} \\ \frac{\partial y'}{\partial v_t} & \frac{\partial y'}{\partial \omega_t} \\ \frac{\partial \theta'}{\partial v_t} & \frac{\partial \theta'}{\partial \omega_t} \end{pmatrix}$$
$$= \begin{pmatrix} -\frac{\sin \theta + \sin(\theta + \omega_t \Delta t)}{\omega_t} & \frac{v_t(\sin \theta - \sin(\theta + \omega_t \Delta t))}{\omega_t^2} + \frac{v_t \cos(\theta + \omega_t \Delta t) \Delta t}{\omega_t} \\ \frac{\cos \theta - \cos(\theta + \omega_t \Delta t)}{\omega_t} & -\frac{v_t(\cos \theta - \cos(\theta + \omega_t \Delta t))}{\omega_t^2} + \frac{v_t \sin(\theta + \omega_t \Delta t) \Delta t}{\omega_t} \\ 0 & \Delta t \end{pmatrix}$$

# Predikce



# Korekce

Predikovaná střední hodnota měření

$$z_t = \underbrace{\begin{pmatrix} \sqrt{(m_x - \bar{\mu}_{t,x})^2 + (m_y - \bar{\mu}_{t,y})^2} \\ \text{atan2}(m_y - \bar{\mu}_{t,y}, m_x - \bar{\mu}_{t,x}) - \bar{\mu}_{t,\theta} \end{pmatrix}}_{h(x_t, j, m)} + \mathcal{N}(0, Q_t)$$

Jacobian vzhledem k pozici

$$\begin{aligned} H_t &= \frac{\partial h(\bar{\mu}_t, m)}{\partial x_t} = \begin{pmatrix} \frac{\partial r_t}{\partial \bar{\mu}_{t,x}} & \frac{\partial r_t}{\partial \bar{\mu}_{t,y}} & \frac{\partial r_t}{\partial \bar{\mu}_{t,\theta}} \\ \frac{\partial \phi_t}{\partial \bar{\mu}_{t,x}} & \frac{\partial \phi_t}{\partial \bar{\mu}_{t,y}} & \frac{\partial \phi_t}{\partial \bar{\mu}_{t,\theta}} \end{pmatrix} \\ &= \begin{pmatrix} -\frac{m_{j,x} - \bar{\mu}_{t,x}}{\sqrt{q}} & -\frac{m_{j,y} - \bar{\mu}_{t,y}}{\sqrt{q}} & 0 \\ \frac{m_{j,y} - \bar{\mu}_{t,y}}{\sqrt{q}} & -\frac{m_{j,x} - \bar{\mu}_{t,x}}{\sqrt{q}} & -1 \end{pmatrix}, \text{ kde} \\ q &= (m_{j,x} - \bar{\mu}_{t,x})^2 + (m_{j,y} - \bar{\mu}_{t,y})^2 \end{aligned}$$

# Korekce

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

Predikovaná kovariance měření

$$S_t = H_t \bar{\Sigma}_t H_t^T + Q_t$$

Zesílení

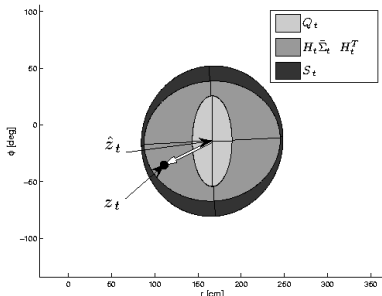
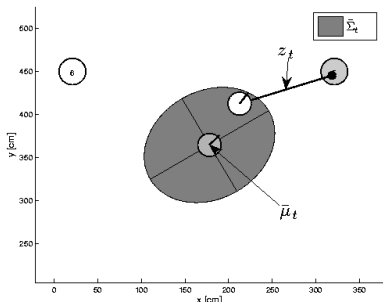
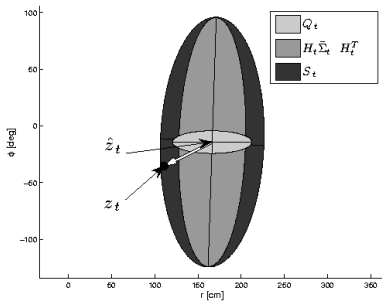
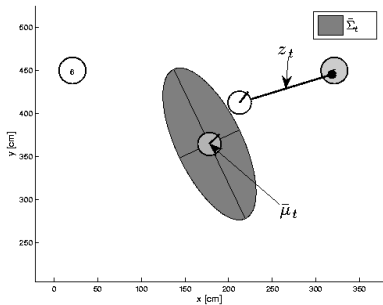
$$K_t = \bar{\Sigma}_t H_t^T S_t^{-1}$$

Aktualizovaná střední hodnota a kovariance

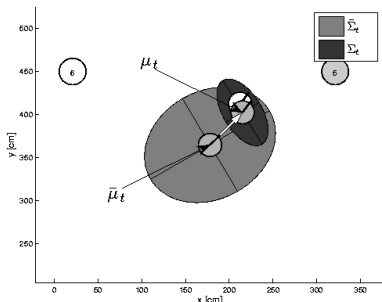
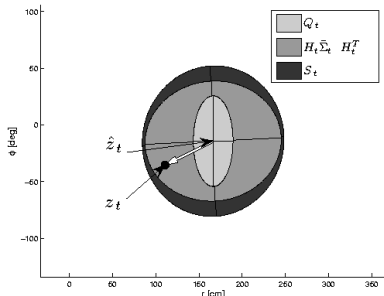
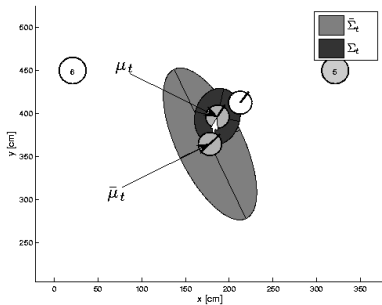
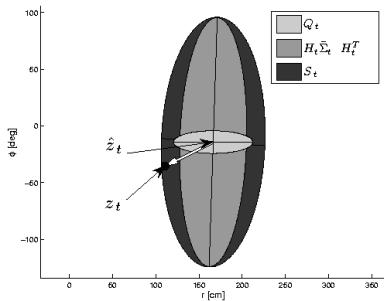
$$\mu_t = \bar{\mu}_t + K_t(z_t - \hat{z}_t)$$

$$\Sigma_t = (E - K_t H_t) \bar{\Sigma}_t$$

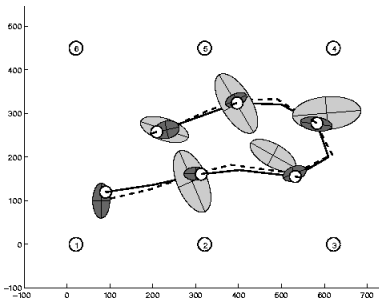
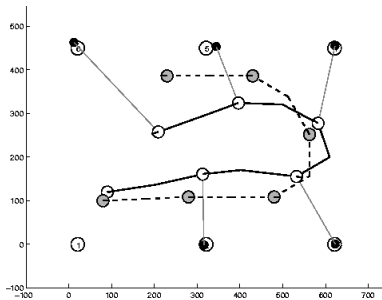
# Pozorování



# Korekce

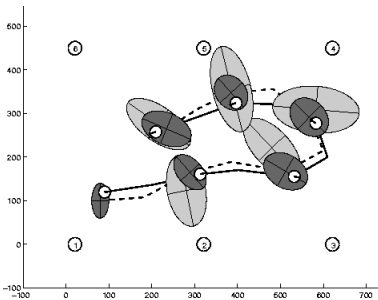
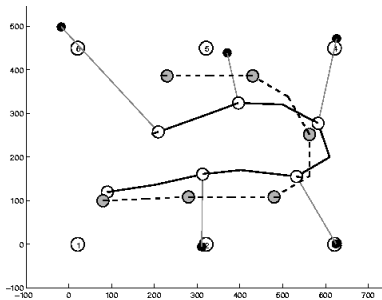


# Odhad - sekvence 1





## Odhad - sekvence 2



# Poděkování

Při přípravě slidů jsem vycházel z přednášek Sebastina Thruna, z kterých pochází i většina prezentovaných obrázků. Tyto (a další) je možné volně stáhnout ze stránek

<http://www.probabilistic-robotics.org/>.

Pro vlastní studium doporučuji knihu

S. Thrun, W. Burgard, and D. Fox: *Probabilistic Robotics*. MIT Press, Cambridge, MA, 2005.