Úvod do mobilní robotiky — AIL028

Lokalizace

Zbyněk Winkler

zbynek.winkler at mff.cuni.cz

http://robotika.cz/articles/umor/cs

22. listopadu 2007

Opakování z minula
- Senzory související s pohybem
- Modely kolových vozidel
- Odometrie

Jak se vypořádat s nepřesnými vstupními daty?
- Průměr
- Inkrementální výpočet průměru
- Plovoucí průměr délky k
- Odhad plovoucího průměru

Kalmanův filter
- Souvislost s inkrementálním odhadem průměru
- Algoritmus aktualizace
- Podmínky korektnosti

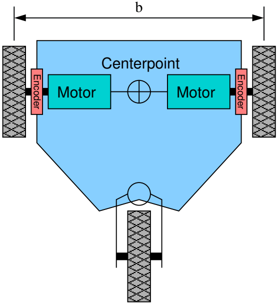
Monte Carlo Lokalizace
- Co je to?
- Slabiny MCL

Senzory související s pohybem

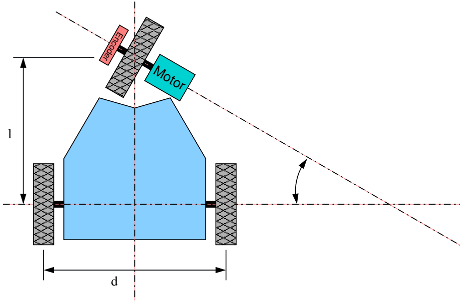
- Optické enkodéry
  - Absolutní
  - Relativní
- Potenciometry
- Magnetické, kapacitní, indukční
- Akcelerometry (MEMS)

Modely kolových vozidel

Diferenční řízení – tank



Tříkolka – Ackermanovo řízení



Odometrie

Dead Reconing

Akumulování relativní informace o změně pohybu:

$$\begin{aligned}\theta_i &= \theta_{i-1} + \Delta\theta_i \\ x_i &= x_{i-1} + \Delta F_i \cos(\theta_i) \\ y_i &= y_{i-1} + \Delta F_i \sin(\theta_i)\end{aligned}$$

Kde $[x_i, y_i, \theta_i]$ je pozice a orientace robota v čase i, $\Delta\theta_i$ je otočení a ΔF_i je posun dopředu.

Vypočítaná pozice začne být časem ovládána více chybou měření než vlastními údaji — je třeba se s tím nějak vypořádat.

Průměr

- asi první věc, co každého napadne
- definice:

$$\overline{x} = \frac{1}{n}\sum_{i=0}^{n} x_i$$

Průměr

- asi první věc, co každého napadne
- definice:

$$\overline{x} = \frac{1}{n}\sum_{i=0}^{n} x_i$$

- v jakých případech nám pomůže?

Průměr

- asi první věc, co každého napadne
- definice:

$$\overline{x} = \frac{1}{n}\sum_{i=0}^{n} x_i$$

- v jakých případech nám pomůže?
- co když nám data chodí postupně?

Průměr

- asi první věc, co každého napadne
- definice:

$$\overline{x} = \frac{1}{n} \sum_{i=0}^{n} x_i$$

- v jakých případech nám pomůže?
- co když nám data chodí postupně?
- co když si nemůžeme pamatovat úplně všechny hodnoty?

Průměr

- asi první věc, co každého napadne
- definice:

$$\overline{x} = \frac{1}{n} \sum_{i=0}^{n} x_i$$

- v jakých případech nám pomůže?
- co když nám data chodí postupně?
- co když si nemůžeme pamatovat úplně všechny hodnoty?
- co když vyžadujeme menší (konstantní) složitost?

Inkrementální výpočet průměru

- definice:

$$\begin{aligned} s_0 &= 0 \\ s_n &= s_{n-1} + x_n \\ \overline{x}_n &= s_n / n \end{aligned}$$

Inkrementální výpočet průměru

- definice:

$$\begin{aligned} s_0 &= 0 \\ s_n &= s_{n-1} + x_n \\ \bar{x}_n &= s_n \big/ n \end{aligned}$$

- v jakých případech nám pomůže?

Inkrementální výpočet průměru

- definice:

$$
\begin{aligned}
s_0 &= 0 \\
s_n &= s_{n-1} + x_n \\
\overline{x}_n &= s_n / n
\end{aligned}
$$

- v jakých případech nám pomůže?
- co když se odhadovaná hodnota mění? (roste/klesá/osciluje)

Plovoucí průměr délky k

- definice:

$$\begin{aligned} x_i, s_0 &= 0 \qquad \text{pro } i < 0 \\ s_n &= s_{n-1} - x_{n-k} + x_n \\ \overline{x}_n &= s_n / k \end{aligned}$$

Plovoucí průměr délky k

- definice:

$$\begin{aligned} x_i, s_0 &= 0 \qquad \text{pro } i < 0 \\ s_n &= s_{n-1} - x_{n-k} + x_n \\ \overline{x}_n &= s_n / k \end{aligned}$$

- v jakých případech nám pomůže?

Plovoucí průměr délky k

- definice:

$$
\begin{aligned}
x_i, s_0 &= 0 \qquad \text{pro } i < 0 \\
s_n &= s_{n-1} - x_{n-k} + x_n \\
\overline{x}_n &= s_n / k
\end{aligned}
$$

- v jakých případech nám pomůže?
- můžeme použít pro filtrování enkodérů?

Plovoucí průměr délky k

- definice:

$$\begin{aligned} x_i, s_0 &= 0 \qquad \text{pro } i < 0 \\ s_n &= s_{n-1} - x_{n-k} + x_n \\ \overline{x}_n &= s_n / k \end{aligned}$$

- v jakých případech nám pomůže?
- můžeme použít pro filtrování enkodérů?
  - zachovává integrál?

Plovoucí průměr délky k

- definice:

$$
\begin{aligned}
x_i, s_0 &= 0 \qquad \text{pro } i < 0 \\
s_n &= s_{n-1} - x_{n-k} + x_n \\
\overline{x}_n &= s_n / k
\end{aligned}
$$

- v jakých případech nám pomůže?
- můžeme použít pro filtrování enkodérů?
  - zachovává integrál?
- co když si nemůžeme/nechceme pamatovat k starých měření?

Odhad plovoucího průměru

- x_{n-k} nahradíme pomocí $\overline{x}_{n-1}$ (nejlepší odhad, co máme)
- s_{n-1} nahradíme pomocí $\overline{x}_{n-1} \cdot k$ (ekvivalentní)

$$
\begin{aligned}
x_i, s_0 &= 0 \qquad \text{pro } i < 0 \\
s_n &= s_{n-1} - x_{n-k} + x_n \\
\overline{x}_n &= s_n / k \\
&\Downarrow \\
\overline{x}_n &= (\overline{x}_{n-1} \cdot k - \overline{x}_{n-1} + x_n)/k \\
&\Downarrow \\
\overline{x}_n &= (\overline{x}_{n-1} \cdot (k-1) + x_n)/k
\end{aligned}
$$

Odhad plovoucího průměru

- x_{n-k} nahradíme pomocí $\overline{x}_{n-1}$ (nejlepší odhad, co máme)
- s_{n-1} nahradíme pomocí $\overline{x}_{n-1} \cdot k$ (ekvivalentní)

$$
\begin{aligned}
x_i, s_0 &= 0 \qquad \text{pro } i < 0 \\
s_n &= s_{n-1} - x_{n-k} + x_n \\
\overline{x}_n &= s_n / k \\
&\Downarrow \\
\overline{x}_n &= (\overline{x}_{n-1} \cdot k - \overline{x}_{n-1} + x_n)/k \\
&\Downarrow \\
\overline{x}_n &= (\overline{x}_{n-1} \cdot (k-1) + x_n)/k
\end{aligned}
$$

- v praxi třeba $\overline{x} = (\overline{x} \cdot 3 + x) \gg 2$ – jednoduché, rychlé

Souvislost s inkrementálním odhadem průměru

$$
\begin{aligned}
\overline{x}_n &= (\overline{x}_{n-1} \cdot (k-1) + x_n)/k \\
&= \tfrac{k-1}{k} \cdot \overline{x}_{n-1} + \tfrac{1}{k} \cdot x_n \\
&= \left(1 - \tfrac{1}{k}\right) \cdot \overline{x}_{n-1} + \tfrac{1}{k} \cdot x_n
\end{aligned}
$$

- $\left(1 - \frac{1}{k}\right)$ a $\frac{1}{k}$ zde působí jako určité váhy pro kombinaci předchozího průměru a nové hodnoty

Toto je základní myšlenka, které je celý Kalmanův filter založen!

Definice

Stavová rovnice

$$x_{k+1} = Fx_k + w_k$$

kde x_k je n dimenzionální stavový vektor, F je transformační matice a w_k šum/chyba stavu ($w_k \sim N(0, Q)$)

Rovnice měření

$$z_k = Hx_k + v_k$$

kde z_k je m dimenzionální vektor měření, H je $m \times n$ matice určující vztah mezi stavem a měřením a v_k je šum/chyba měření ($v_k \sim N(0, R)$)

Algoritmus aktualizace

Predikce stavu a chyby — pomocí stavové rovnice

$$
\begin{aligned}
x_{k+1}^- &= Fx_k \\
P_{k+1}^- &= FP_kF^T + Q
\end{aligned}
$$

Korekce pomocí měření — pomocí rovnice měření

$$
\begin{aligned}
z_k &= Hx_k + v_k \\
K_k &= P_k^- H^T (HP_k^- H^T + R)^{-1} \\
x_k &= x_k^- + K_k(z_k + Hx_k^-) \\
P_k &= (I - K_kH)P_k^-
\end{aligned}
$$

Podmínky korektnosti

- diskrétní, lineární model procesu (změny stavu)
- diskrétní, lineární model měření
- všechen šum má gaussovské rozdělení pravděpodobnosti

Monte Carlo Lokalizace

- Pozice reprezentovaná množinou vážených vzorků
- 3 fáze
  - pohyb (predikce)
  - měření (korekce)
  - převzorkování
- Jednoduché ohodnocovací funkce:
  `double eval(Pozice p);`
- Zásadní výhoda - jednoduchá implementace

Slabiny MCL

- Není jedna pozice, tak podle čeho má robot jet?
- Možnost skokových změn
- Problém nastavení parametrů a ladění
- Manévry nezachycené v distribuční funkci (čelní náraz)