Úvod do mobilní robotiky — AIL028

SLAM - souběžná lokalizace a mapování

Martin Dlouhý a Zbyněk Winkler

{md|zw} at robotika.cz

http://robotika.cz/guide/umor07/cs

10. ledna 2008

SLAM intro

- SLAM = Simultaneous Localization And Mapping
- problém typu slepice-vejce
- nutné pro průzkum neznámého prostředí
- stále atraktivní/aktivní téma v robotice
- *Robit aneb cesta tam a zase zpátky.*

Vyrovnání polygonu

- použití v zeměměřictví
- měření vzdálenosti a azimutu na další bod
- nelze korigovat, není-li reference
- uzavření cyklu
- minimalizace chyb měření

Navigační značky

- odpovídají vrcholům polygonu
- měřená vzdálenost a azimut z odometrie
- problém s jednoznačností vrcholů

Skládání laserových skenů

- inkrementální
- matching - minimalizace funkce
- po přidání mapa „ztuhne“
- problém s cykly
- chybí „revize historie“

EM metoda

- data $d = \{o^{(1)}, u^{(1)}, o^{(2)}, u^{(2)}, \ldots o^{(T)}, u^{(T)}\}$, kde $o^{(t)}$ je *observation* v čase t a $u^{(t)}$ odometrie
- mapa $m = \{m_{x,y}\}_{x,y}$, kde $m_{x,y}$ popisuje vlastnosti světa na pozici (x, y)
- model pohybu: $P(\xi' | u, \xi)$
- model vnímání: $P(o | m, \xi)$
- inverzní model vnímání: $P(m | o, \xi)$ (z Bayesova vzorce)

$$P(A|B_i) = \frac{P(A \cap B_i)}{P(B_i)}$$

$$P(B_i|A) = \frac{P(B_i)P(A|B_i)}{\sum_{j=1}^{k} P(B_j)P(A|B_j)}$$

EM metoda - cíl

- nalézt nejpravděpodobnější mapu
- $m^* = \arg\max_m P(m|d)$
- EM ve skutečnosti hledá pouze lokální maximum

$$P(m|d) = \lambda \int \dots \int \prod_{t=1}^{T} P(o^{(t)}|m, \xi^{(t)}) \prod_{t=1}^{T-1} P(\xi^{(t+1)}|u^{(t)}, \xi^{(t)}) d\xi^{(1)}, \dots,$$

E step - expectation step

- počítá pravděpodobnost $P(\xi|m, d)$ s dosud nejlepší známou mapou m
- odpovídá lokalizaci
- v první iteraci je mapa prázdná

M step - maximization step

- výpočet nejpravděpodobnější mapy $= \arg\max_m P(m|\xi, d)$
- používá pozici z předešlého kroku
- implementace simulovaným žíháním (zabránění lokálnímu minimu)

Problém?!

Thrun, Burgard, Fox

A Real-Time Algorithm for Mobile Robot Mapping With Applications to Multi-Robot and 3D Mapping (ICRA 2000)

- princip MCL
- mapa je kolekce laserových skenů s pozicí
- matching z každého vzorku
- provádí se zpětné korekce (pozice u skenů)

Tom Duckett, Stephen Marsland, Jonathan Shapiro

Learning Globally Consistent Maps by Relaxation (ICRA 2000)

- používá pouze informace o relativní pozici
- potřebuje kompas
- mapa je graf, každý vrchol má kartézské souřadnice, každá hrana má délku a absolutní orientaci (díky kompasu)

Algoritmus

pro každý vrchol i

1. pro všechny sousedy j vrcholu i spočítej odhad (x'_{ji}, y'_{ji}) pozice vrcholu i

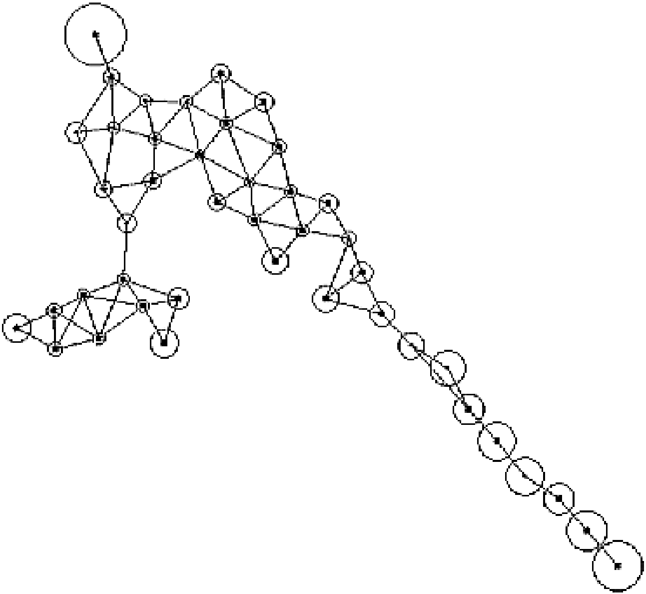
$$\begin{aligned} x'_{ji} &= x_j + d_{ji}\cos\theta_{ji} \\ y'_{ji} &= y_j + d_{ji}\sin\theta_{ji} \end{aligned}$$

   a jeho rozptyl

$$v_{ji} = v_j + u_{ji}$$

2. zkombinuj všechny odhady pozice vrcholu i

$$\frac{1}{v_i} = \sum_j \frac{1}{v_{ji}} \qquad x_i = \sum_j \frac{x'_{ji} v_i}{v_{ji}} \qquad y_i = \sum_j \frac{y'_{ji} v_i}{v_{ji}}$$



Důkaz konvergence

- každá hrana představuje pružinu
- pružina je nejméně napnutá, když vzdálenost vrcholů odpovídá její délce
- globální minimum, když celková energie minimální

$$E = \sum_i \sum_j \left[(x_i - x_j + d_{ij} \cos \theta_{ij})^2 + (y_i - y_j + d_{ij} \sin \theta_{ij})^2 \right]$$

1. každá aktualizace zmenšuje energii E
2. energie E je zdola omezená 0
3. je kvadratická, takže má pouze jedno minimum

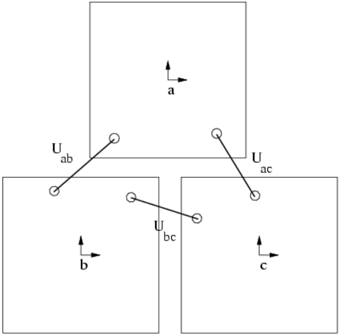
Andrew Howard, Maja J Mataric and Gaurav Sukhatme

Relaxation on a Mesh: a Formalism for Generalized Localization (IROS 2001)

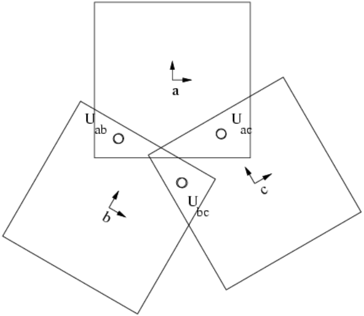
- nepotřebuje kompas
- zavádí lokální soustavy souřadnic
- znak pozorovaný ve dvou různých soustavách mezi nimi vytváří hranu nulové délky (taky pružina)

Algoritmus

- v podstatě stejný jako u Duckett.
- není dokázáno, že konverguje do globálního minima



$$U_{ab} + U_{bc} + U_{ac} > 0$$



$$U_{ab} + U_{bc} + U_{ac} = 0$$

Zbyněk Winkler

Plánování cesty pro mobilní roboty
(Diplomová práce, MFF UK, 2002)

- rozšíření a zobecnění Duckett
- nepotřebuje kompas
- u pozorovaných znaků potřebuje navíc i orientaci
- dokázána konvergence do globálního minima
- vrcholy mají kromě pozice i orientaci
- hrany vyjadřují relativní pozici vrcholů vůči sobě

Algoritmus

1. Pro každý vrchol k aktualizujeme souřadnice (x_k, y_k, θ_k) (souřadnice ostatních vrcholů jsou fixované). Nové souřadnice vrcholu k získáme minimalizací funkce

$$f_k(x_k, y_k, \theta_k) = \sum_j^l \begin{bmatrix} (x_j + u_{jk}\cos\theta_j - v_{jk}\sin\theta_j - x_k)^2 + \\ (y_j + u_{jk}\sin\theta_j + v_{jk}\cos\theta_j - y_k)^2 + \\ (x_k + u_{kj}\cos\theta_k - v_{kj}\sin\theta_k - x_j)^2 + \\ (y_k + u_{kj}\sin\theta_k + v_{kj}\cos\theta_k - y_j)^2 \end{bmatrix}$$

2. Předchozí krok opakujeme, dokud maximální změna pozice neklesne pod požadovanou hranici nebo dokud není splněno jiné vhodné ukončovací kritérium.

Důkaz konvergence

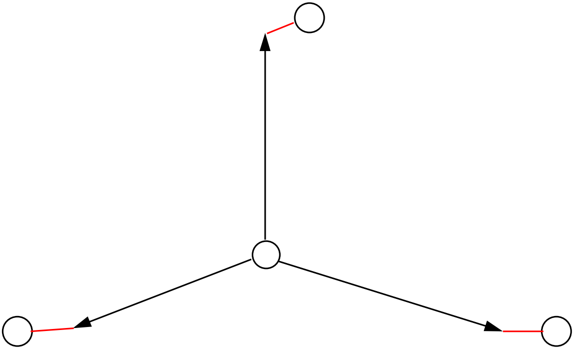
Definujeme chybovou funkci vrcholu i jako součet přes všechny sousedy

$$e_i = \sum_{j}^{l} \begin{bmatrix} (x_i + u_{ij}\cos\theta_i - v_{ij}\sin\theta_i - x_j)^2 + \\ (y_i + u_{ij}\sin\theta_i + v_{ij}\cos\theta_i - y_j)^2 \end{bmatrix} \tag{1}$$

Každý člen součtu (1) vyjadřuje rozdíl v přesvědčení vrcholu i na to, kde by se měl soused j nacházet a toho, kde se ve skutečnosti nachází.
Chybová funkce celého grafu je potom součet chyb e_i všech vrcholů.

$$E = \sum_{i}^{n} e_i \tag{2}$$

Grafické znázornění chyby vrcholu



Grafické znázornění chyby sousedů