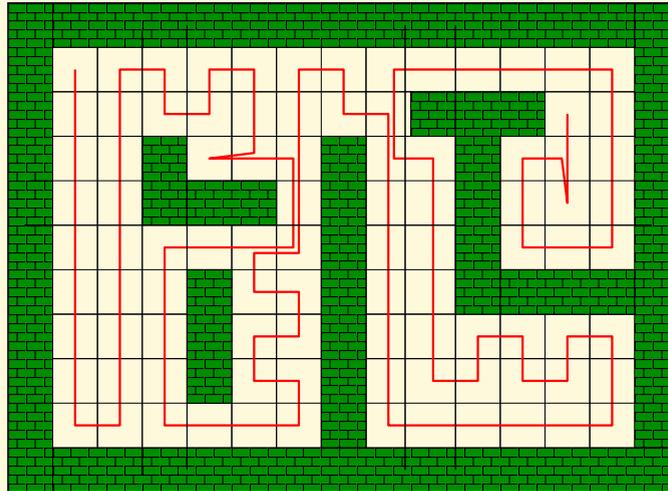


Uklízečí cesta

- cleaning/coverage path
- aplikace uklízení, hledání min, mapování pouze lokálně detekovatelné látky, sekání trávy, hrabání listí. . .
- cíl: efektivně projet všechna dosažitelná místa ve vymezeném prostoru

Rastrové uklízení (1/2)

- prostředí bitová mapa, robot = pixel
- semínkové vyplňování (seed fill)
- optimální řešení - nejkratší cesta?

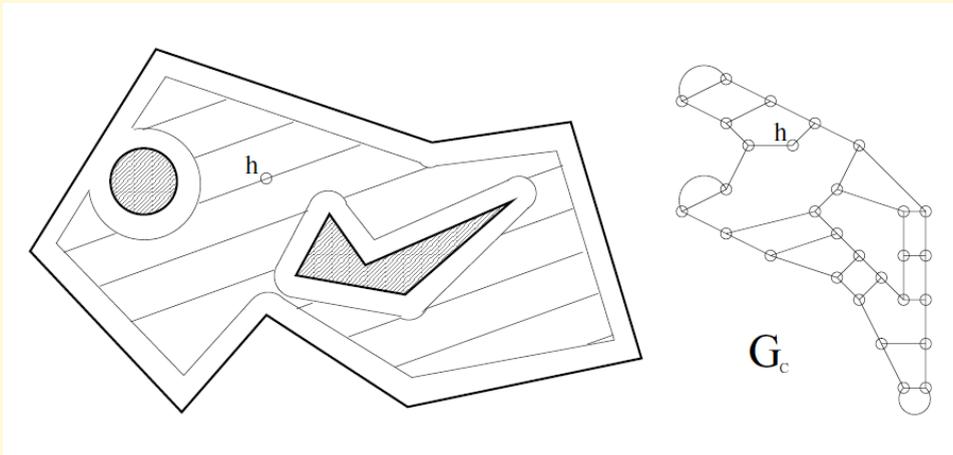


Rastrové uklízení (2/2)

- optimální řešení je NP-C
- suboptimální: dvojitě řádkování
- grafové řešení — uklízení okolo překážek a rovné segmenty

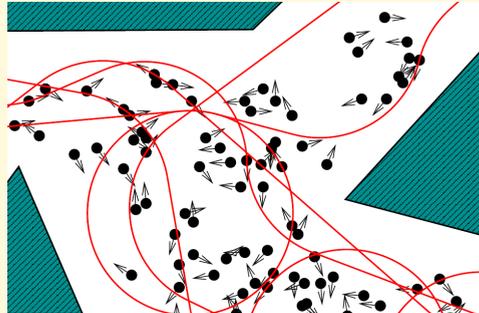
Obecné grafové řešení

- rozdělení prostoru řádkováním v daném směru
- přidat objíždění překážek
- vznikne 3-regulární graf
- zdvojení některých hran — možný Euklidovský tah



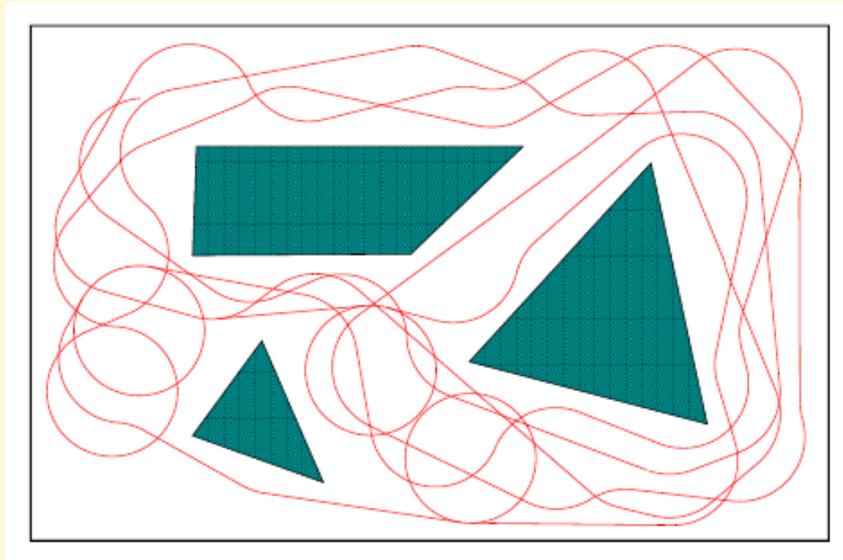
Uklízečí autíčko

- řízení typu auto, nesmí couvat
- těžké už plánování (řeší pravděpodobnostní plánovače)
- možnost segmentace prostoru a řešení problému obchodního cestujícího
- jak uklízet při nedostatku místa pro manévrování?



Genetické algoritmy pro uklízení

- gen = uklízení cesta
- ohodnocovací funkce = stupeň pokrytí
- rekombinace = výměna jednotlivých úseků



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 $d = \{o^{(1)}, u^{(1)}, o^{(2)}, u^{(2)}, \dots, o^{(T)}, u^{(T)}\}$, 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

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

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 žiháním (zabránění lokálnímu minimu)

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)