

Úvod do mobilní robotiky — AIL028

Pravděpodobnostní plánování

Zbyněk Winkler a Martin Dlouhý

`zbynek.winkler at mff.cuni.cz, md at robotika.cz`

`http://robotika.cz/guide/umor05/cs`

12. prosince 2005

- 1 Konfigurační prostor
 - Co už umíme a co ne?
 - Jak řešit složitější případy?
 - Definice konfiguračního prostoru
 - Definice plánování cesty
- 2 PRM: Probabilistic Road Map
 - Lokální plánovač
 - Strategie výběru spojovaných vrcholů
 - Heuristiky pro zlepšení mapy
 - Výhody a nevýhody
- 3 RRT: Rapidly-Exploring Random Trees
 - Stavba stromu
 - Vlastnosti stromu
 - Návrh jednostromového plánovače
 - Návrh vícestromového plánovače

Jak na tom jsme

- Co už umíme naplánovat?
- Co ještě neumíme?

Co už umíme naplánovat?

polygonální prostředí

- bod
- kružnice
- konvexní robot bez otáčení
- žebřík s otáčením

Co ještě neumíme?

Například

- robot s otáčením (co není žebřík)
- nekonvexní robot
- několik součástí spojených klouby
- omezení na pohyb (autíčko)
- dynamika pohybu (setrvačnost, omezená akcelerace apod.)
- 3D svět

Čím se tyto skupiny liší?

- počet stupňů volnosti pohybu robota odpovídá dimenzi prostoru, ve kterém plánujeme (až na žebřík)
- řešení jsme nacházeli převedením na „předchozí případ“
- „předchozí případ“ byl pohyb bodu ve 2D

Jak řešit složitější případy?

- potřebujeme nějakou formalizaci, která by nám všechny problémy sjednotila
- algoritmus pracující s touto formalizací potom funguje na všechny problémy
- na naši záchranu přichází

konfigurační prostor — C_{space}

Konfigurační prostor

- C_{space} je n -dimenzionální prostor, kde n je počet parametrů potřebných k úplnému popisu robotovy pozice/stavu
- C_{free} je podprostor všech konfigurací, ve kterých nedochází ke kolizi s žádnou překážkou — **volný konfigurační prostor**
- C_{free} závisí na tvaru a rozměrech robota a distribuci překážek

Plánování cesty

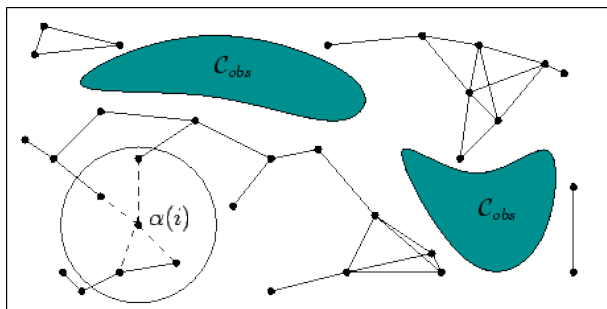
Nechť A je startovní a B cílová pozice. Nalezněte konečně dlouhou spojitou křivku $p \in C_{free}$ takovou, že pro její parametrizaci podle délky l platí, že $p(0) = A$ a $p(l) = B$.

- cesta je spojitá křivka v konfiguračním prostoru
- vyjadřuje pohyb **bodu** v n -dimenzionálním prostoru
- na křivku lze klást další požadavky mimo spojitosti
 - autíčka
 - dynamika

PRM: Probabilistic Road Map

Pravděpodobnostní mapa cest

- pravděpodobnostní dekompozice C_{free}
 - pokrytí C_{free} sítí cest
- dotazy na existenci cesty napojením startu a cíle do sítě



Formální popis

$G = (V, E)$ síť/graf cest

- V vrcholy $\in C_{free}$
- vrcholy v_1 a v_2 jsou spojeny hranou $e = (v_1, v_2) \in E$, pokud mezi nimi existuje cesta v C_{free}
- graf G stavěn inkrementálně postupným přidáváním vrcholů
 - vrcholy generovány náhodně — odtud „pravděpodobnostní“
- hrany mezi vrcholy přidává **lokální plánovač**

Lokální plánovač

Pro každého robota specifický:

- dimenze konfiguračního prostoru
- bere v úvahu různá omezení (např. autíčko)

Vygenerovaná cesta (posloupnost bodů v C_{free}) je zkonvertována do podmnožiny \mathcal{W} (prostor, který by robot zaujímal při pohybu po této cestě), která se otestuje na průnik s překážkami.

Často používané heuristiky/zjednodušení:

- konfigurace jsou spojovány úsečkami
- plánovač pouze kontroluje zda celé leží v C_{free}
 - nafoukne úsečku na polygon
 - otestovat prázdný průnik s překážkami

Strategie výběru spojovaných vrcholů

k nejbližších

typicky $k = 15$

k nejbližších z každé komponenty souvislosti

tj. stavíme strom, typicky $k = 1$

blíže než r

všechny body do vzdálenosti r

Heuristiky pro zlepšení mapy

- lepší pokrytí
 - generování vrcholů ve dvou fázích
 - první stejná, ale navíc se ke každému vrcholu generuje statistika úspěšnosti lokálního plánovače
 - ve druhé se nové vrcholy generují tam, kde byla úspěšnost lokálního plánovače menší
 - cesty typu „náhodná procházka“
- lepší topologie
 - přidáváme hrany, které tvoří cykly
 - vrcholy jsou *blízko* v C_{free} , ale *daleko* v grafu
- lepší tvar cesty
 - vybranou cestu iterativně vyhlazujeme a zkracujeme vybráním dvou bodů a jejich přímějším napojením

Výhody a nevýhody

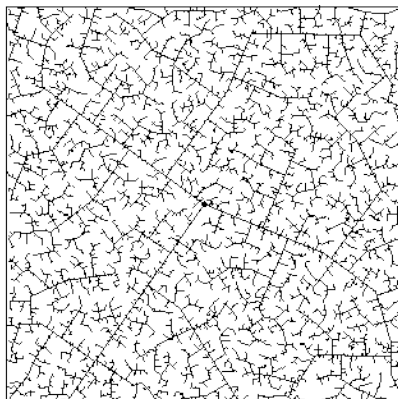
- rychlost generování grafu / rychlost vyhledávání cest
- neznámé prostředí
- dynamické prostředí

RRT: Rapidly-Exploring Random Trees

- roste strom (nebo stromy) ze startu (nebo z cíle)
- cesta nalezena pokud strom dorostl až do druhého konce cesty



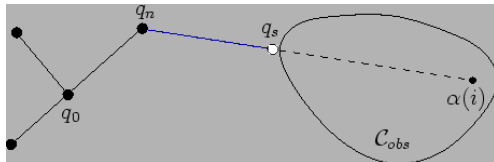
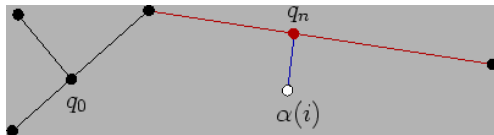
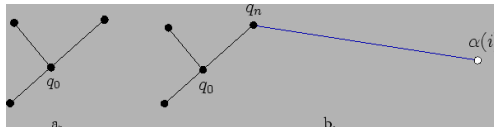
45 iterations



2345 iterations

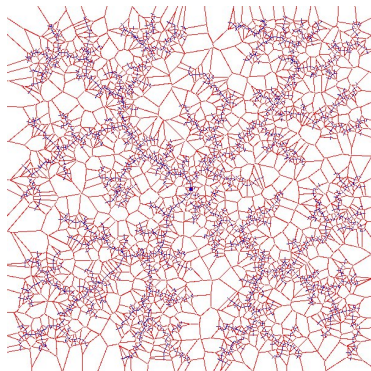
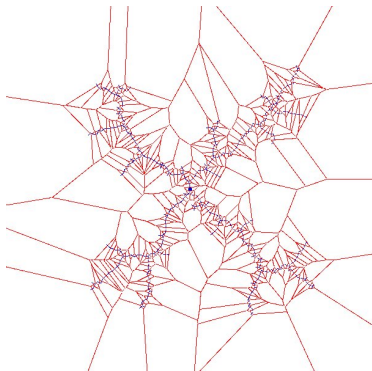
Stavba stromu

- q_{init} startovní konfigurace
- q_{rand} náhodná konfigurace
- q_{goal} cílová konfigurace
- algoritmus staví strom s q_{init} postupným „roztahováním“
- v každém kroku se roztahuje do q_{rand} (extend)



Vlastnosti stromu

- pravděpodobnost, že je vrchol vybrán procedurou `extend` pro rozšíření je přímo úměrná velikosti jeho voronoi regionu
- konverguje k rovnoměrnému pokrytí prostoru



Návrh jednostromového plánovače

- základní verze roste rovnoměrně na všechny strany
 - pomalu konverguje k cíli
- verze GoalBias
 - vybíráme q_{rand} z nerovnoměrné distribuce, házet mincí, jedna strana q_{rand} , druhá q_{goal}
 - i s pravděpodobností $p = 0.05$ pro q_{goal} konverguje k cíli daleko rychleji, ale pozor na lokální minima pro příliš velká p !
- verze GoalZoom
 - místo přímo q_{goal} vybíráme z regionu se středem v q_{goal} , velikost regionu je určena nejbližším vrcholem u q_{goal}
 - nejlepší je ale vybírat z distribuce, která je hladká a k cíli směřuje postupně

Návrh vícestromového plánovače

- strom z q_{init} i z q_{goal}
- algoritmus se dělí na dvě části
 - extend T_a ke q_{rand}
 - extend T_b ke q_{new} a prohod' T_a s T_b
- limitní varianta, kdy z každého q_{rand} stavíme vlastní strom
 - jednovrcholové stromy spojujeme pomocí connect
 - simulace původního algoritmu pravděpodobnostní mapy cest