

INGENIØRHØJSKOLEN AARHUS UNIVERSITET

ITROB2 - GRUPPE 13

AUTONOMOUS MOBILE ROBOTS ASSIGNMENT - 2

DATO: 07/03/2021

VEJLEDER:

PETER AHRENDT

MADS DYRMANN



Studerende:

Dorte Sølund Hansen (SWT)

Stine Falkenberg Gravesen (SWT)

Berat Kaya (SWT)

Casper Fevre Hansen (SWT)

Stud. nummer:

201808180

201608108

201610971

201501949

Introduktion

Dette er den anden obligatoriske øvelse i kurset ITROB2. Her skal der eksperimenteres med path planning med Probabilistic Roadmap (PRM).

Der er i opgaven taget udgangspunkt i dokumentationen for Probabilistic Roadmaps i MathWorks [1] og det tilhørende eksempel. Derudover er Pure Pursuit algoritmen, der blev benyttet i en tidligere øvelse, blevet anvendt [2].

Path Planning med PRM

Ved hjælp af PRM algoritmen blev der genereret en path, som det ses i Listing 1.

```

1 %Calculate a simple path:
2 startLocation = [2 1];
3 endLocation = [12 10];
4 path = findpath(prmSimple,startLocation,endLocation);
5
6 show(prmSimple)

```

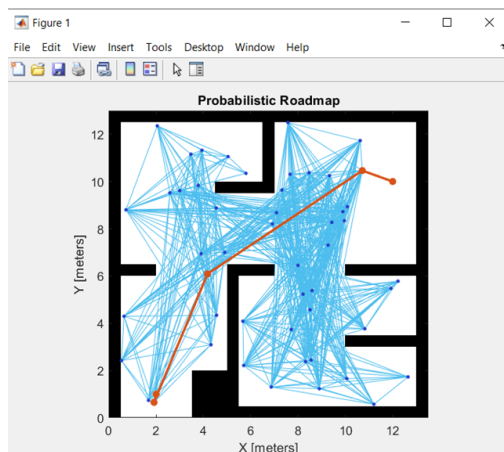
Listing 1: Kodeudsnit for beregning af path

Derefter blev pure pursuit algoritmen brugt, hvor controllerens waypoints sættes til den genererede path fra Listing 1. Controllerens ønskede lineære hastighed sættes til 0,3 m/s, maksimale vinkelhastighed til 1 rad/s og lookaheaddistance til 0,4 m.

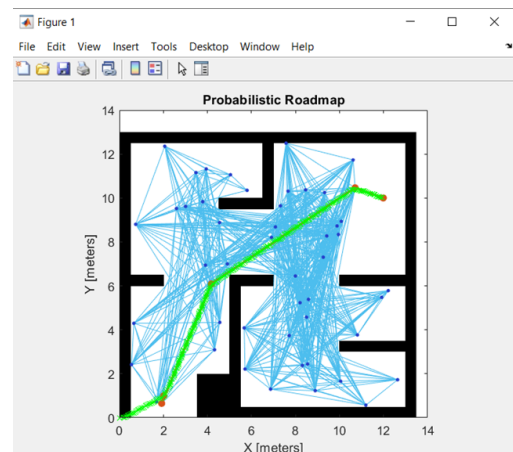
En lav lineær hastighed og høj vinkelhastighed, gør at robotten følger banen godt, da den hurtigt kan rette ind når der kommer et sving i banen.

Som en hovedregel bør lookaheaddistance være større end den lineære hastighed for robotten [2]. Derudover er det vigtigt at lookaheaddistancen ikke er for høj (da robotten så vil afskære hjørnerne af banen) eller for lav, for så kan robottens kørsel blive ustabil [2].

På Figur 1a ses path fra Listing 1. PRM algoritmen beregner en ny path hver gang koden køres på ny. Antallet af knuder afhænger af, hvad kortet er opsat til i koden. I dette eksempel er det et simpelt map med 50 knuder.



(a) PRM uden robot



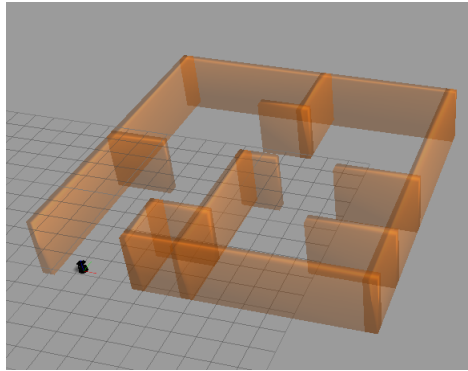
(b) PRM med robot

Figur 1: Screenshot af PRM med og uden robot

Et map med mange knuder vil give en mere direkte path, men samtidig kræver et større antal knuder også mere beregningskraft for computeren [1]. Derfor vil det altid være en afvejning, hvor direkte path skal være i forhold til hvor meget beregningskraft der kan benyttes. Et map med for lidt knuder vil betyde at den ikke ville kunne beregne en path. Det er derfor vigtigt, at vide hvilken computer der arbejdes med, så der kan

benyttet det bedste antal knuder til den benyttetes processer.

For at visualisere at robotten kører den viste path i Figur 1a, er der bygget et map i Building Editor i Gazebo. Dette kan ses på Figur 2.



Figur 2: map bygget ved hjælp af Building Editor i Gazebo

Mappet blev gemt som en .sdf fil, men der var store problemer med at åbne det gemte map i gazebo igen efter programmet var lukket ned. Det er ikke lykkedes at åbne mappet igen, og derfor har robotten ikke kørt i det viste kort på Figur 2, men i en tom verden.

For bedre at kunne følge robottens path er dens x-, og y-koordinater plottet på Figur 1b, hvor de grønne krydser er robottens path fra gazebo.

Litteratur

- [1] MathWorks. Probabilistic roadmaps. [Online]. Available: https://se.mathworks.com/help/robotics/ug/probabilistic-roadmaps-prm.html?searchHighlight=path%20planning%20prm&s_tid=srchtitle
- [2] ——. Pure pursuit controller. [Online]. Available: <https://se.mathworks.com/help/robotics/ug/pure-pursuit-controller.html>