

OBLIGATORISK OPGAVE 2

ITROB2 - AUTONOME MOBILE ROBOTTER
AARHUS UNIVERSITET

08. MARTS, 2020

Pure-Pursuit algorithm

Gruppe: 7

Studienr.:
201705815
201707435
201705707

Navn:
Mathias Gade
Nikolaj Richardy Molzen
Axel Aaby Ulriksen

Retning:
SW
SW
SW

Indhold

Pure-Pursuit algorithm	1
1 Introduktion	2
2 Simulering	3
2.1 Matlab simulering	3
2.1.1 Eksperimentering med parametre	4
3 TurtleBot	5
3.1 Kørsel med Turtlebot	5
Referencer	6

1 Introduktion

I denne opgave skal der bruges "Pure Pursuit-algorithm" til at følge en rektangulær form (2mx5m) med en TurtleBot. Dette gøres først ved hjælp af en simulering i MatLab, således at der er et udgangspunkt og en demo af den ønskede funktionalitet. Der blev her taget udgangspunkt i et eksempel fra mathworks [1]

Når simulationen virker så vil gruppen forsøge at få robotten til at køre ruten og se hvor tæt dens kørsel kommer på den som ville kunne ses i simuleringen. Dette vil give formodning om hvor præcis TurtleBot'en er i forhold til det input den modtager fra programmet i MatLab.

2 Simulering

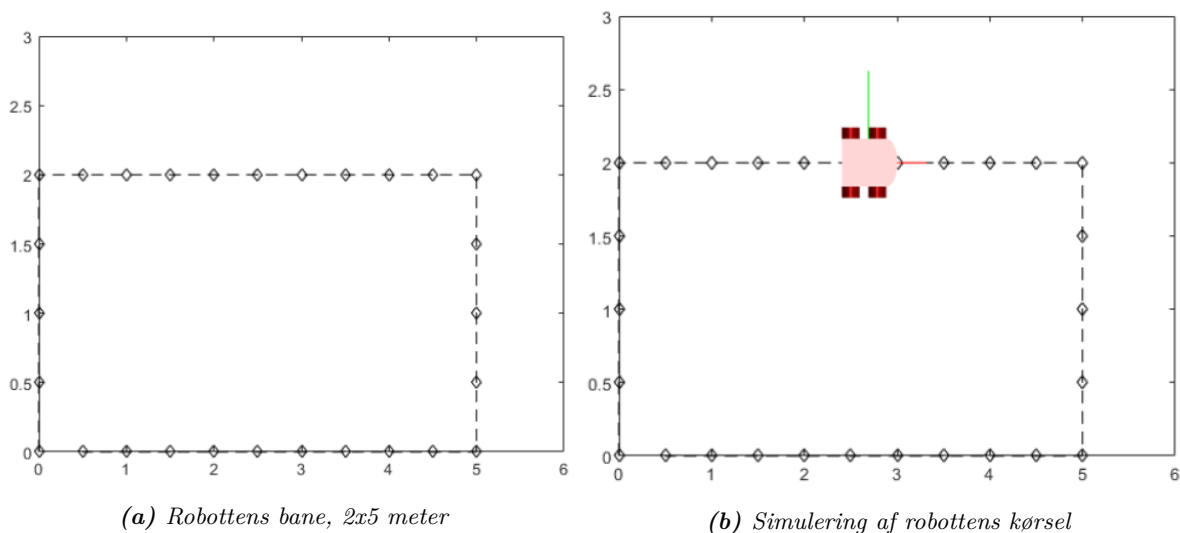
I dette afsnit vil simuleringen blive beskrevet. Her vil programmet blive forklaret i korte træk, med fokus på de væsentlige elementer, som gør simuleringen mulig.

2.1 Matlab simulering

Til simuleringen af pure pursuit algoritmen, blev der først lavet en path, som robotten skulle følge, denne blev indtastet manuelt og fulgte beskrivelsen efter opgaven på de 2m x 5m.

Til dette blev der lavet en pure pursuit controller, denne havde nogle forskellige parametre man kunne eksperimentere med, disse var bla.

- DesiredLinearVelocity
- MaxAngularVelocity
- LookAheadDistance



Figur 2.1: Simulering af TurtleBot

På figur 2.1a ses den bane som simuleringen ved prøve at følge. Her er det punkterne der danner banen og som programmet vil styre efter. På figur 2.1b ses simuleringen igang. Her er robotten blevet afbildet at den kører rundt på banen. Her kan det også ses at ved en simulering kører robotten meget præcist på ruten.

2.1.1 Eksperimentering med parametre

Ved at ændre på de forskellige parametre for pure pursuit controlleren, kunne man ændre måden hvorpå simuleringen blev kørt samt hvordan robotten kørte. Ved at ændre på LookAheadDistance ændrede man på hvordan robotten bl.a. opførte sig ved sving, ved at have en lav LookAheadDistance vil robotten svinge/"resonere" mere omkring dens path, hvor den ved en f.eks. 0.5, kørte mere smooth uden at svinge for meget efter et sving, men derimod begynde at dreje tidligere. Ved at have en for stor en LookAheadDistance, vil den begynde at dreje for tidligt, og reelt ikke kører ret meget på den path man har givet den, men nærmere i en cirkel inde og ude for rektanglen.

Man kunne som sagt og ændre på DesiredLinearVelocity og MaxAngularVelocity. Ved at have en for høj DesiredLinearVelocity, kunne controlleren ikke nå at korrigere efter den path som den skulle følge, da den kørte for hurtigt og ikke kunne nå at dreje i tide. Ved at ændre på MaxAngularVelocity, kunne man bestemme hvor hurtigt robotten må dreje.

Vi endte her ud med at have følgende parametre for den mest optimale kørsel:

- $\text{DesiredLinearVelocity} = 0.6$
- $\text{MaxAngularVelocity} = 2$
- $\text{LookAheadDistance} = 0.5$

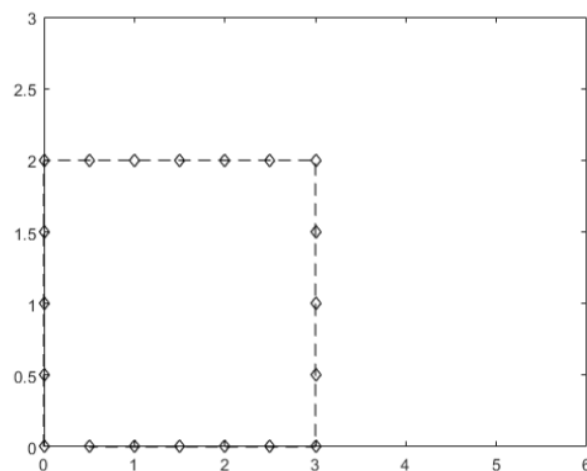
3 TurtleBot

I dette afsnit vil kørslen af TurtleBot blive beskrevet. Her vil programmet blive forklaret i korte træk, med fokus på de væsentlige elementer, som gør kørslen mulig. Programmet til at få TurtleBot til at køre er ved hjælp af koden fra simuleringen hvor man her blot sender kommandoerne til TurtleBot i stedet for et plot.

3.1 Kørsel med Turtlebot

For at få selve TurtleBot til at køre banen som vist på figur 2.1a på side 3, skulle simuleringen blot laves om til at sende kommandoerne til TurtleBot i stedet for at plotte kørslen.

Størrelsen på banen blev dog gjort en smule mindre, da pladsen var begrænset. Derfor gik det fra en bane på 2x5 meter til en bane på 2x3 meter. Den nye bane kan ses på figur 3.1.



Figur 3.1: Bane til kørsel af den rigtige TurtleBot

Til kørslen af TurtleBot skal det noteres at robottens egen position ikke blev brugt, men derimod simuleringen. Med dette betyder det at når programmet prøver at finde ud af hvor det er på banen, så går den ud fra simuleringens position, og altså ikke der hvor robotten selv tror den er.

Efter et par kørsler kunne det hurtigt konkluderes at TurtleBot kørte meget tæt på simuleringen. Den stoppede altid cirka en halv meter fra hvor den startede hvilket den skulle i forhold til de punkter banen er opbygget af. Dertil kørte TurtleBot også flot i svingen. Simuleringen drejede blødt i svingene for at komme tilbage på banen. Her gjorde TurtleBot blot et skarpt knæk uden at køre fremad, og dermed forblev den på rette spor.

Referencer

- [1] MathWorks. *MathWorks Pure Pursuit example*. URL: <https://www.mathworks.com/help/robotics/examples/path-following-for-differential-drive-robot.html>. (accessed: 07.03.2020).