

Regbot og håndtuning - del 2

Øvelsesvejledning

Formål

Øvelsen understøtter følgende læringsmål:

1. Anvende heuristiske metoder til regulator design

Indhold

Øvelsen går ud på at designe regulatorer - til styring af hastighed og retning.

Del 1: Hastighedsregulator

Hurtig og præcis PI regulator.

Del 2: Retningsregulator

Så ligeud bliver ligeud.

Del 3: Missionsfirkant

Verificer resultat.

Del 1: Hastighedsregulator

Metoden er grundlæggende at forsøge sig frem ud fra en grundlæggende viden om hvordan en regulator virker og se på konsekvensen.

1. Klargør robot og forbind til PC, til strømforsyning og lader. Det meste af øvelsen foregår med robotten på bordet med hjulene opad. Ideen er at optimere regulatoren med hjulene opad, vel vidende at det nok er anderledes når hjulene også skal trække robotten.

Målsætningen er en P regulator, der opfylder:

- er hurtig (når hurtigt en stabil hastighed),

- har en lille stationær fejl (at hastigheden når referencehastigheden), og
- at motoren kører jævnt - at motorspændingen ikke skifter for meget (og ikke støjer).

2. Lav en mission, der

- Logger motorspænding og hjulhastighed (som minimum).
- Logger hvert ms ($\log=1$).
- kører ligeud med 0.3 m s^{-1} i 0.15 s .
- dernæst skifter til 0.5 m s^{-1} i 0.25 s .
- sætter hastighed til 0 m s^{-1} til der ikke er mere plads i loggen.

P regulator

3. Indstil Velocity regulator,

- så den kun anvender en P-regulator (disable alle andre indstillinger på nær output limit).
- Vælg en K_P i området 1-1000, foretag indledningsvis en vurdering for K_P ud fra at 2 V motorspænding giver ca. en hastighed på 0.5 m s^{-1} .

4. Plot i MATLAB

Hent og gem logfilen og plot i MATLAB

- venstre og højre hjulhastighed som funktion af tiden
- venstre og højre motorspænding.

5. gentag fra punkt 3 med forskellige K_P til der er fundet et rimeligt kompromis, der rimeligt tilgodeser målsætning i punkt 1.

P-feed forward regulator

6. Tilføj et feed-forward led i velocity regulatoren, det burde (i teorien) mindske den stationære fejl.

7. Gentag fra punkt 3 med værdier for K_P og K_{ff} til der er fundet et godt kompromis.

8. Ud fra MATLAB plottet, mål tiden det tager robotten at accelerere fra 0.3 m s^{-1} til 0.45 m s^{-1} . Da målingerne er støjfyldte, så foretages en vurdering af hvornår hastigheden øges fra 0.3 m s^{-1} og hvornår 0.45 m s^{-1} nås.
- $x[0.3]=0.153\text{s}$
 $x[0.45]=0.161$
 $\text{delta } x = 0.161 - 0.153 = 0.008\text{s} = 8 \text{ ms}$
9. udregn den gennemsnitlige acceleration fra 0.3 m s^{-1} til 0.45 m s^{-1} , og gem resultaterne (i Tabel 1).

$$\text{acceleration} = \text{delta } v / \text{delta } t = 0.15 / 0.008 = 18.7 \text{ m s}^{-2}$$

Tabel 1: Parametre for hurtig hastighedsregulering (ikke på gulv)

Robot	
K_P	12 $\left[\frac{\text{volt}}{\text{m s}^{-1}} \right]$
K_{ff}	3.5 $\left[\frac{\text{volt}}{\text{m s}^{-1}} \right]$
Stigetid fra 0.3 til 0.45 m s^{-1}	0.008 [s]
gennemsnitlig acceleration	18.7 $[\text{m s}^{-2}]$

Gulvtest af hastighedsregulator

Det tager længere tid på gulvet, så juster eventuelt log-interval og step tidspunktet (tidspunkt for hastighedsskift).

10. Afprøv med de fundne P-feed-forward indstillinger på gulvet. Den skulle nu køre lige ud (måske bortset fra starten). Placer robotten med bagenden (den med knapperne) ned mod gulvet.

Observer om der er tegn på hjulspin. Prøv eventuelt et par gange.

For et dæk er det typisk at der kun kan opnås en friktionskraft på ca. halvdelen af normalkraften ($M \cdot g$) inden der opstår hjulspin. Det betyder at der maksimalt kan opnås en acceleration på ca. $0.5g \approx 5 \text{ m s}^{-2}$. Hvis accelerationen var højere i testen ovenfor, er der nok tale om hjulspin, bl.a. da ikke hele vægten er på de trækkende hjul.

11. Prøv en kørsel med begrænset acceleration ved at tilføje $a_{cc}=4$ (4 m s^{-2}) i missionens første linje (den er så gyldig for hele missionen (indtil den ændres)).

Accelerationsbegrænsningen er implementeret ved at begrænse hvor hurtigt referencen må ændrer sig (et step bliver således til en rampe med denne hældning).

Blev kørslen kønnere?

At begrænse ændringshastigheden er en typisk metode til at reducere slid og forlænge levetiden for maskiner i industrien.

Del 2. Drejestyring

Der skal nu findes en regulator til drejefunktionen (de fundne parametre for hastighedsregulator bevares).

1. Tilføj *Turnrate* og *Pose* til log options. Heading er "h" delen af pose.

2. Opbyg en ny mission, hvor der drejes og logges under drejet.
 - Log interval 10 ms (eller deromkring, så det passer med missionen).
 - Kør med 0.4 m s^{-1} i hele missionen og accelerationsbegrænsning,
 - start ligeud 0.4 m (brug `dist=0.4` som betingelse)
 - Drej til f.eks. 10° ved at sætte *heading* referencen med `head=10` i yderligere 0.3 m, f.eks. med linjen:
`head=10 : dist=0.4`
 - Drej tilbage til 0 grader (`head=0`) i 0.4 m og stop (når loggen er fuld)
3. Indstil heading regulatoren (under *control* fanen og *heading*) med en K_P værdi mellem 0.1 og 100. Husk at enable regulatoren. Alle andre indstillinger i heading regulator disables.
4. Kør først missionen med hjulene opad.
5. Gem loggen og plot missionen i MATLAB, plot *turnrate* og *heading* (h i *pose*) og eventuelt motorspænding og hjulhastighed i en anden figur, alt som funktion af tiden. Heading er i radianer, så konverter heading til grader i plottet, f.eks.
`plot(dd(:,1), dd(:,9)*180/pi, 'c')`, hvis data er i `dd` og heading er kolonne 9.

Retningsstyring fungerer ved at det ene hjul bevarer hastigheden og det andet kører langsommere. PS ved kørsel i retning 180 grader er der i visse regulatorkonfigurationer en mindre fejl (skift fra 180° til -180° er ikke alle steder håndteret rigtigt), men det burde ikke give problemer her.

Kurven skulle vise at der drejes til retning 10° (og tilbage), at den nye kurs falder til ro hurtigt og stabilt, og at *turnrate* falder til nul hurtigt.

Find den højeste værdi af K_P der giver et godt drej, se eventuelt også lidt på motorspændingen.

Del 3. Firkantkørsel

Det skal nu prøves om firkantkørslen er bedre med aktiv retningsstyring.

1. Opret en mission, hvor drej nu bruger `head=XX : dist=0.4` (i stedet for `tr=0.1 : turn=90`). Log relevante data med et interval så hele turen er dækket.
2. Prøv først firkantkørsel med hjulene opad. Plot data i matlab.
3. Hvis det ikke set pænt ud - tydelige dæmpede svingninger - så juster:
 - Heading-regulator K_P - kompromis mellem oversving og hastighed/nøjagtighed.
 - Brug accelerationsbegrænsning (i første missionslinje).
 - Begræns output af drejeregulator (*output limit*).
 - Overvej om feed forward kan være en ide.

4. Når firkantkørsel ser godt ud, så prøv på gulvet.
5. Vurder om kørslen visuelt svarede til forventningerne. Hent log og plot kørslen.
6. Svarer plot til virkeligheden? og er drej 90° og er lige stræk lige? ellers gå tilbage til punkt 3.
7. Noter resultatet i tabel 2, tilføj, hvis der er flere anvendte regulatorparametre.
resultaterne skal bruges til sammenligning, når vi når til resultat af frekvensanalysemetoden.

Tabel 2: Håndtunet regulator - regulatorparametre

Robot		
	Text hastighedsregulator	retningsregulator
Kp	$[V m^{-1} s]$	$[m s^{-1} rad^{-1}]$
Output limit	$[V]$	$[m s^{-1}]$
Feed forward	$[V m^{-1} s]$	$[m s^{-1} rad^{-1}]$

(Det var såmænd det hele for denne gang)

