

31300/31301 Regulerings teknik 1

Forår 2018 rev 3/jca

REGBOT i balance — vejledning

Formål

Understøtter følgende af kursets læringsmål:

- Anvende frekvensanalyse til regulator design, herunder PID, lead, lag og feed forward regulatorer.
- Forstå og og kunne anvende begreber som stabilitet, stabilitetsmargin, wind-up, overshoot og stationære fejl.
- Vurdere og vælge standard regulatorer.
- Anvende analyse- dimensionerings- og simuleringsprogrammel

Indhold

Projektet går ud på at designe regulatorer til problematiske overføringsfunktioner og derigennem forhåbentlig bedre forstå PID regulatorens muligheder.

Der skal designs regulatorer, der kan sikre at missioner kan køres med robotten i balance.

Designovervejelser og resultater skal dokumenteres i rapport 3 REGBOT, og indgår i eksamensbedømmelsen.

Denne øvelsesvejledning dækker de resterende øvelsesperioder i kurset.

Del 1: REGBOT balancemodel.

Del 2: Hjulhastighed regulator

Del 3: Balanceregulator.

Del 4: Hastighedsregulator i balance.

Del 5: Positionsregulator i balance (optional).

NB! Læs først hvilke dele der skal med i rapporten, så de rigtige figurer og værdier sikres.

1 REGBOT balancemodel

1.1 Hovedmodel

Udgangspunktet er en simulink simscape mekanisk model (regbot_1mg.slx) (se figur 1).

Der er et tilhørende konfigurationsscript (regbot_mg.m) samme sted.

Modellen er forberedt til linearisering med `linmod` med netop et input og et output.

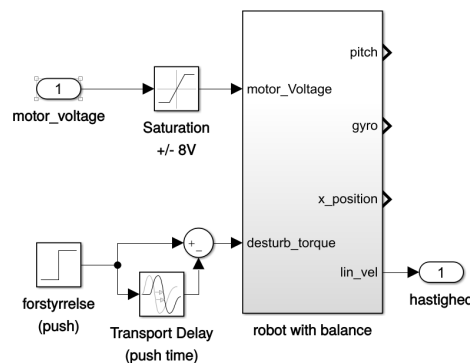
Der er parametre for massen af de enkelte dele af modellen og en `startAngle`, som er den vinkel robotten står i til tiden 0, og som bestemmer arbejds punktet for `linmod` (når der lineariseres til tiden 0).

1.2 Model detaljer

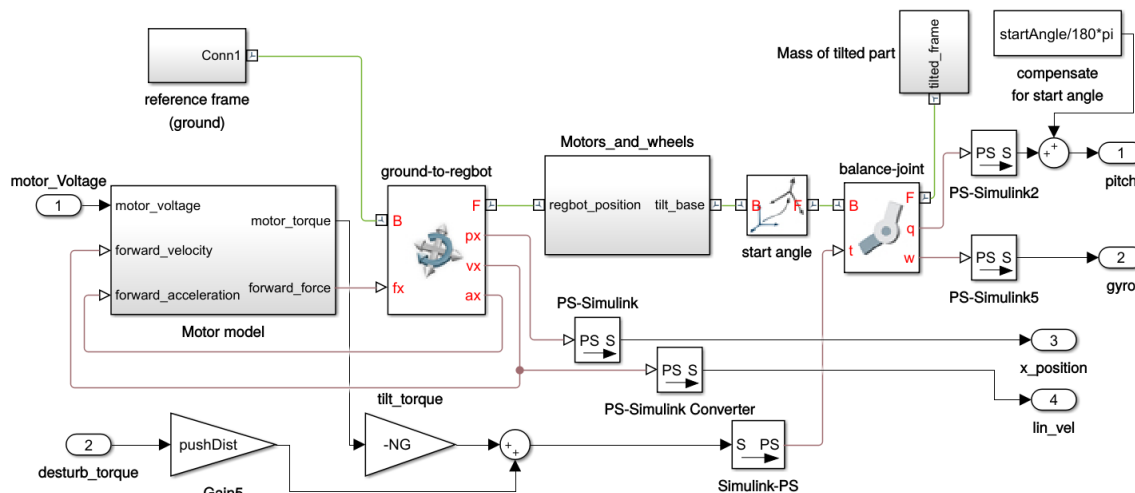
Selve modellen er opbygget omkring matlab simscape sim-mechanics elementer blandet med normale simulink blokke, som vist på figur 2.

Motormodellen “Motor model” svarer til den der er anvendt tidligere, men da robotens inert i nu afhænger af den vinkel (pitch) robotten står i (skal motoren flytte hele robotten, eller kun underdelen) er nogle blokke lidt anderledes. Acceleration “forward_acceleration” og hastighed “forward_velocity” kommer udefra, og motorens moment “motor_torque” er med til at vippe robotten.

Den mekaniske model regner på forskydning af koordinater, derfor er der her en koordinatsystem reference “reference frame”. I forhold til den kan robotten be-



Figur 1: Modellen består af blokken “robot with balance” der indeholder motormodel og øvrig mekanik (se figur 2). Output er bl.a. den lineære hastighed i x-retningen (lin_vel) og tilt vinklen (kaldet pitch). Motorspændingen er her begrænset til 8V (9V er også OK). Til testformål er der mulighed for en forstyrrelse (push) indbygget i modellen.



Figur 2: Modellen består af simulink og sim-mechanics blokke. Input 1 “motor_voltage” er motor spænding som giver en kraft “fx” til joint-blokken “ground-to-regbot” og skubber robotten fremad i x-retningen. Det næste led “balance-joint” tillader modellen at tilte, og tilt-vinklen er tilgængelig som output “pitch”.

væge sig, det er modelleret med et "joint" kaldet "ground-to-regbot". Bevægelsen styres af en kraft "fx" i x-retningen - som er det eneste der bruges her. Ud fra den samme blok fås robotens positionen i x "px", samt hastighed "vx" og acceleration "ax" i x-retningen.

Den masse der flyttes med robottens position består (i hovedsagen) af motorer og hjul, de findes i blokken “Motors _and_wheels”, der på udgangen giver tilt-aksen på robotten.

Tilt akseforskydes med en "startAngle" og endnu et joint "balance-joint" tillader at vippe om denne akse. Ud af dette joint fås tiltvinklen "q" og tilt hastigheden "w". Tilt hastigheden svarer til det vi på robotten kan måle med den indbyggede gyro, ligesom tilt vinklen kan måles på robotten.

Den masse det tiltes findes i blokken “Mass of tilted part”.

1.3 Tilpasning af model

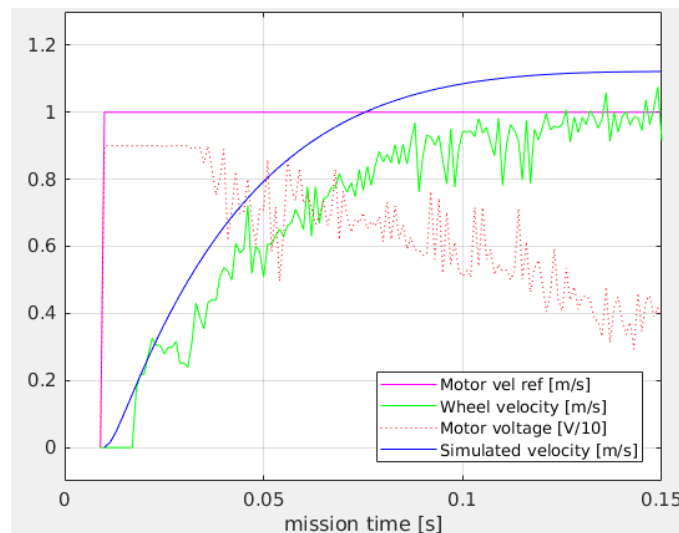
Juster og tilføj parametre så det passer med jeres hidtidige arbejde. Parametrene er i m-filen.

hint: Hver gang der tilføjes en ny regulator til modellen, så gem modellen under et ny navn. Der vil mest sandsynligt være behov for at revurdere de tidligere designede regulatorer.

2 Hjulhastighed regulator

Formål: Hastighedsregulatoren skal være hurtig, men helst uden oversving.

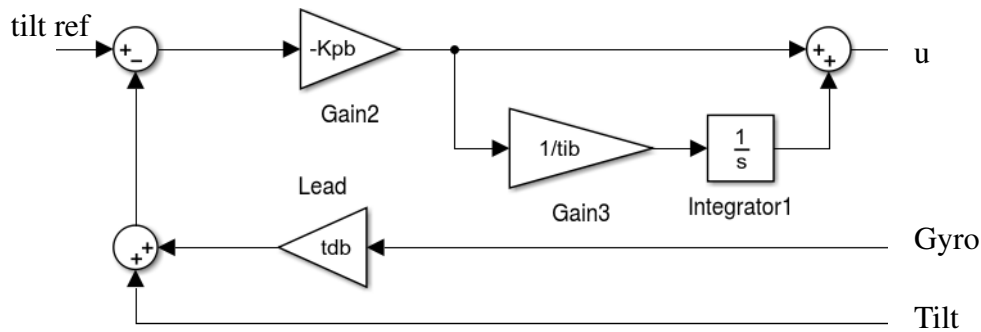
1. Enten brug de regulatorparametre i har fundet tidligere, eller
2. Lad MATLAB estimere overføringsfunktion ud fra model med funktionen `linmod` (se m-filen). Brug den til at designe en ny motor hastighedsregulator efter de nye metoder.
(NB: at den fundne overføringsfunktion er fra modellen med tilt, og vil derfor indeholde poler i højre halvplan. Disse kan motorregulatoren ikke alene fjerne (de kræver at de andre regulatorer er på plads også)).
 - (a) Inspicer bodeplot, vælg en regulatortype,
 - (b) design regulator,
3. vurder step-respons. Step respons skal nok være noget i samme stil som vist i figur 3.
4. indfør hastighedsregulator i simulink, så modellen er klar til næste regulator.



Figur 3: Simuleret step på hastighedsreference (blå) for lineær model af balancemodel (NB! plot kun for meget kort tid, da pol i højre halvplan hurtigt vil få output til at gå mod ∞). Her sammenlignet med hjulhastighed (grøn) for et kort step “vel=1:time=0.15” når robotten startes ca. i samme vinkel som den simulerede, og på en godt underlag for at undgå hjulspin.

3 Balanceregulator

Formål: Balanceregulator skal kunne bringe modellen (og robotten) til at balance fra en startvinkel på ca. 30 grader.



Figur 4: PID regulator, hvor D-leddet tages fra en gyro, hvor den afledede af tilt-vinklen måles direkte af gyro og blot mangler at blive multipliceret med τ_d ("tdb"). Lead leddets pol anvendes således ikke. Resten af regulatoren er helt normal.

1. Modifier en kopi af model med motorhastighedsregulatoren. Flyt input til referenceinput til hastighedsregulatoren, og flyt output til tilt-vinkel, og find nu overføringsfunktion fra referenceinput af motorregulatoren til tiltvinkel ("pitch").
2. Denne overføringsfunktion vil have netop en pol i højre halvplan.
3. Amplitudedel af bodeplot er langt fra monotomt faldende.
Der kunne der være behov for at bruge et I-led til at forme bodeplot mere hensigtsmæssigt.
Fysisk forklaring: Det svarer til at sætte en integrator foran hastighedsregulatoren (når vi ser bort fra nulpunktet). Input til integrator vil så være acceleration, når output er hastighed. Det kan tolkes som vi styrer kraften ($f = M \cdot a$) i stedet for hastigheden. At få robotten i balance med en kraft er måske mere forståeligt end at styre med en hastighed.
I REBOT regulatoren er der en "post integrator", der kan anvendes til dette formål.
4. Brug et Nyquist plot til at vurdere om en positiv K_p kan sikre at -1 punktet omkredsnes netop en gang, når frekvensen s går fra $-j\infty \rightarrow +j\infty$, som som stabilitetskriteriet kræver.
5. D-led (Lead-led) kan designes som sædvanligt, men brug en lille $\alpha \leq 0.1$.
Lead leddet implementeres som et rigtigt D-led ved at anvende "gyro" output. Gyroen giver tilt hastighed som jo allerede er en differentieret udgave af tilt vinklen. Se figur 4 for implementering. Det betyder at den eneste indflydelse α har på designet er placeringen af Lead-leddets nulpunkt.
6. Opret en simulink model, hvor balanceregulatoren er indført, og verificer at balancen nu kan holdes. Hastigheden er der ikke styr på endnu, men hastigheden skulle være konstant (og som en mulighed nul).
7. Prøv den nye regulator på robotten.
NB! vær klar til at gribe, hvis det går galt!

4 Hastighedsregulator i balance

Formål: Hastighedsregulatoren skal sikre at der er styr på den stabile hastighed, som der kan opnås med balanceregulatoren. Således at hastigheden kan styres med `vel=0.xx` fra en mission.

Metodehint:

1. Find overføringsfunktion af systemet fra reference tilt vinkel til hjulhastighed. Overvej igen (ud fra bodeplot) hvordan der designes en regulator.
2. En krydsfrekvens på måske 3-5 radianer/sek (≈ 0.5 Hz) er nok tæt på det maksimale.
3. Prøv på simulatoren, og revurder til det ser fornuftigt ud.
4. Prøv på den virkelige robot (igen er forsigtighed en god ting).

5 Positionsregulator (optional)

Der er også mulighed for en positionsregulator, så robotten kan holde positionen (og ikke bare hastighed 0).

```
vel=0.0, log=25.0: time=0.02
vel=1.0, bal=1, topos=0.0: time=3.0
vel=1.0, topos=1.0: time=8.0
vel=0.0
```

Første linje med `log=25` er på separat linje, da det tager lidt tid at starte logging, som ikke skal forstyrre balancen.

6 Video

Optag en lille video af resultatet (f.eks. fra stilstand til balance og med et step på hastighed eller position). Placer gerne videoen på youtube eller tilsvarende, og tilføj en link i rapporten, så censor kan se resultatet.

Resultaterne skal med i REGBOT rapporten (rapport 3)

(slut)