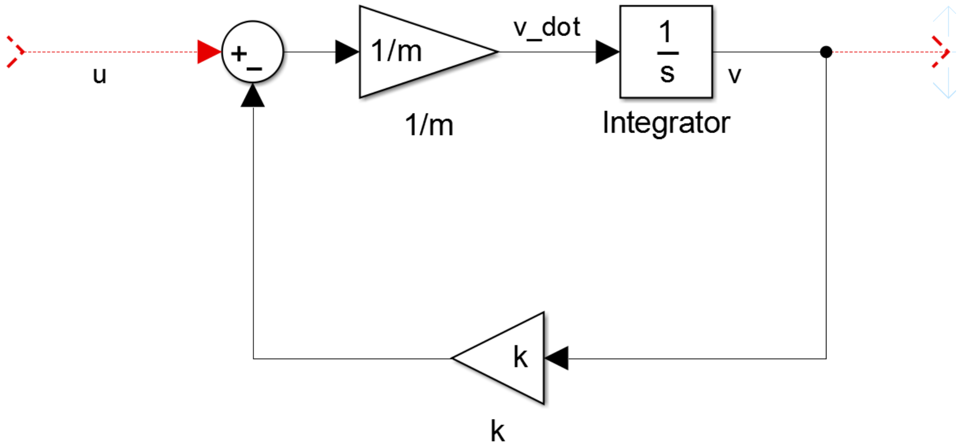
Kybernetikk Intro, Øving 2

Ønsker grundig retting ☺

Oppgave 1: Hastighet

a) Systemet er monovariabelt ettersom det kun har én inngang (u) og én utgang (v).



Figur 1 - Blokkdiagram

Systemet er i åpen sløyfe, så det er ingen tilbakekoblinger.

b) Eulers metode er en måte iterativ og numerisk metode for å tilnærme seg løsningen til en ligning. Den virker slik at man velger et startpunkt, også tar man utgangspunkt i tangenten ved det punktet. Ved hjelp av tangenten flytter man seg til neste x-verdi og lager man en linje som er parallell med funksjonen på ved den x-verdien, og slik fortsetter man så lenge man ønsker. Eulers metode er nok enklere å forstå med formlene enn med ord. Hvis vi har et generelt system $\dot{x} = f(x)$ så kan vi implementere Eulers metode slik:

$$x_{n+1} = x_n + h * f(x_n)$$

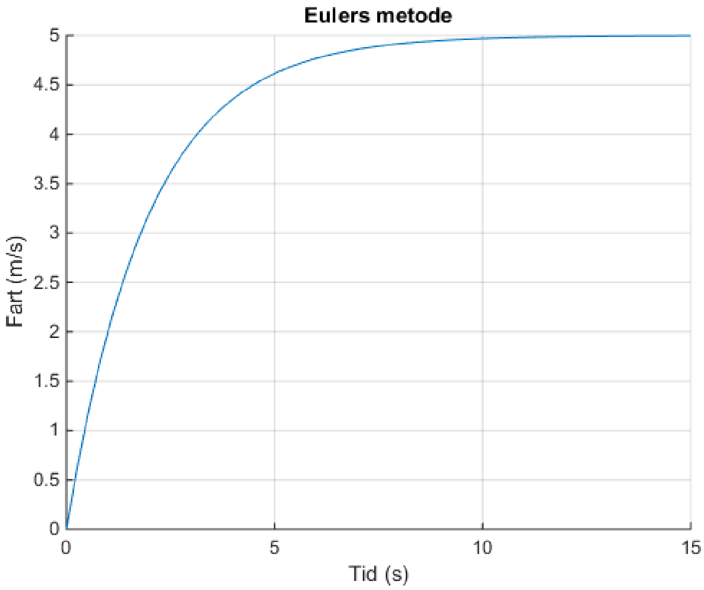
h er skrittlengden, altså lengden mellom hver x-verdi, og det er hovedfaktoren som bestemmer nøyaktigheten til tilnærmingen. Ved å velge en liten h blir tilnærmingen mer og mer nøyaktig.

c) Nøyaktigheten bestemmes som nevnt hovedsakelig av skrittlengden fordi man kan velge en vilkårlig liten skrittlengde for å få en vilkårlig bra nøyaktighet (gitt at funksjonen ikke er altfor vill).

d) Koden:

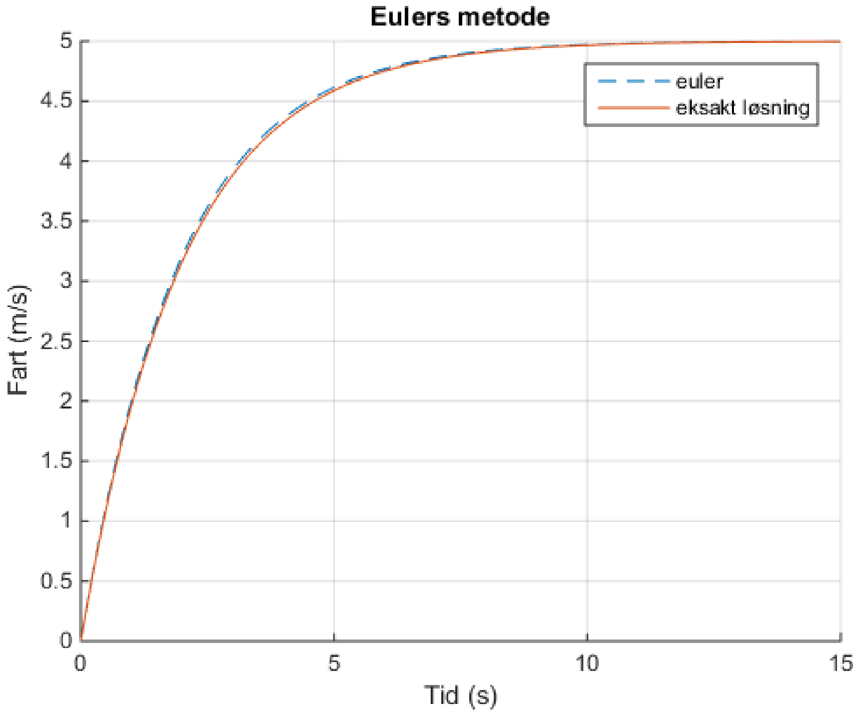
```
%definerer variablene
v(1) = 0;
u = 500;
m = 200;
k = 100;
h = 0.1;
t = 0:h:15;

%implementerer eulers metode
for i=2:(15/h+1),
    v(i)=v(i-1)+h*(u/m - v(i-1)*k/m);
end

plot(t,v);
```

Plottet:



Figur 2 - Eulers metode

e) Plottet av v_eksakt og tilnærmingen:



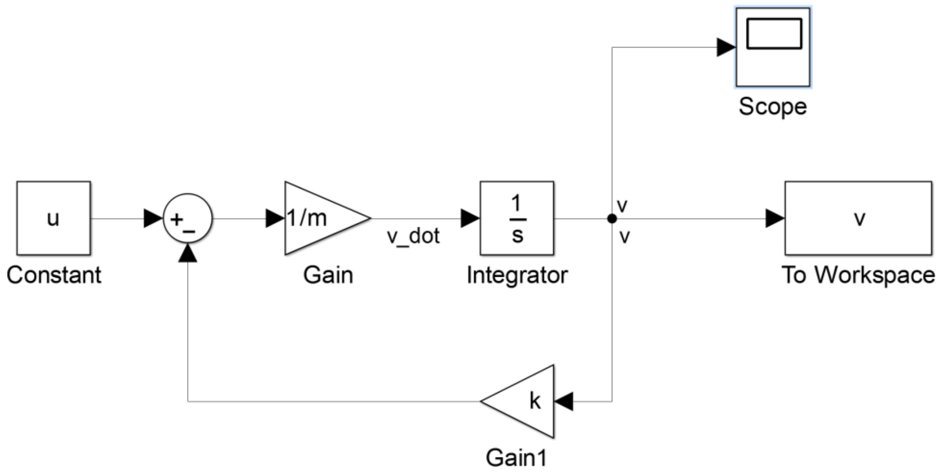
Plottet illustrerer at eulers tilnærming er veldig bra. Den stiplede linjen har aldri noe stort avvik på intervallet 0 til 15.

f) Brukte et skrip for å definere alle verdiene:

```
%definerer variablene
v_0 = 0;
u = 500;
m = 200;
```

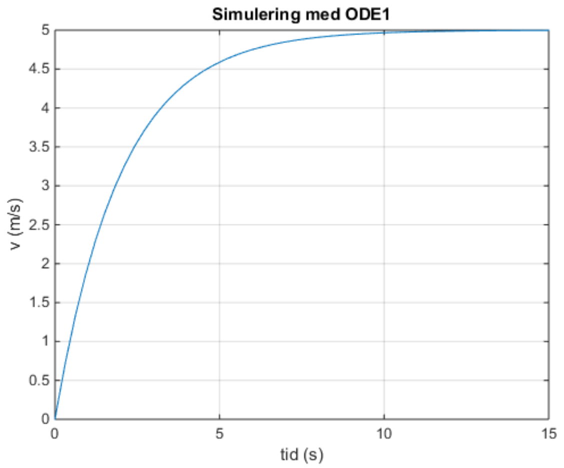
```
k = 100;
t_sim = 15;

sim('simulering', t_sim);

%plotter simuleringsresulatet
figure(1); clf(1);
plot(v);
title('Simulering med ODE1');
xlabel('tid (s)'); ylabel('v (m/s)');
grid on;
```

Simulinkdiagramet er (selvsagt) nesten identisk det i deloppgave a:



Figur 3 - simulering av systemet

Plottet ble:



Figur 4 – ODE1 plot, steptime 0.1

Plottet i figur 4 er identisk med plottet i figur 2 så langt øye kan se.

Oppgave 2: Pitch

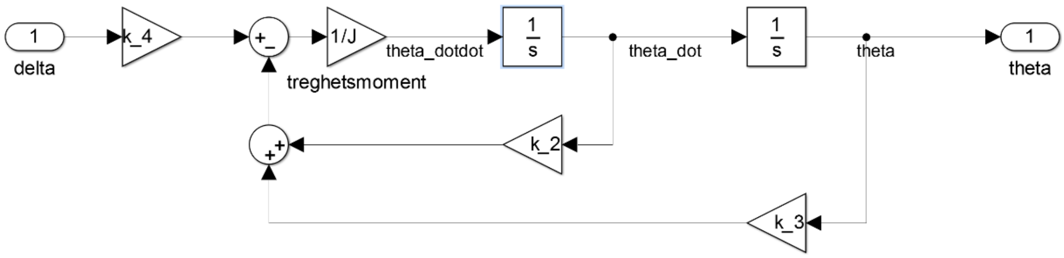
a) Setter opp ligning slik beskrevet i oppgaveteksten:

$$J\ddot{\theta} = -k_2\dot{\theta} - k_3\theta - k_4\delta$$

$$\ddot{\theta} = -\frac{k_2\dot{\theta} + k_3\theta + k_4\delta}{J}$$

Systemet er et andre ordens system.

b)



Figur 5 - blokkdiagram for pitchregulering

Oppgave 3: Posisjon

Farten til fartøyet uavhengig av retningen θ vil være beskrevet av systemet i oppgave 1. Figur 6 (under) viser hvordan hastigheten langs de forskjellige planene påvirkes av vinkelen.

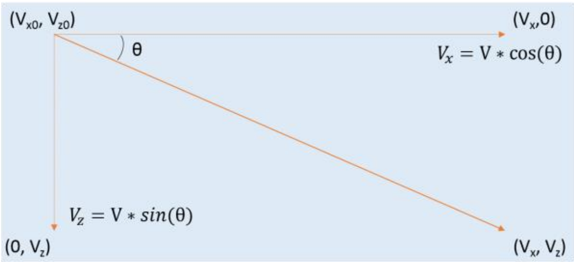
For å finne avstanden langs de forskjellige planene må man integrere farten.

a) x-planet:

$$S_x(t) = \int_0^t \mathrm{V(x)} * \cos(\theta(\mathrm{x}))\, dx$$

b) z-planet:

$$S_z(t) = \int_0^t \mathrm{V(x)} * \sin(\theta(\mathrm{x}))\, dx$$



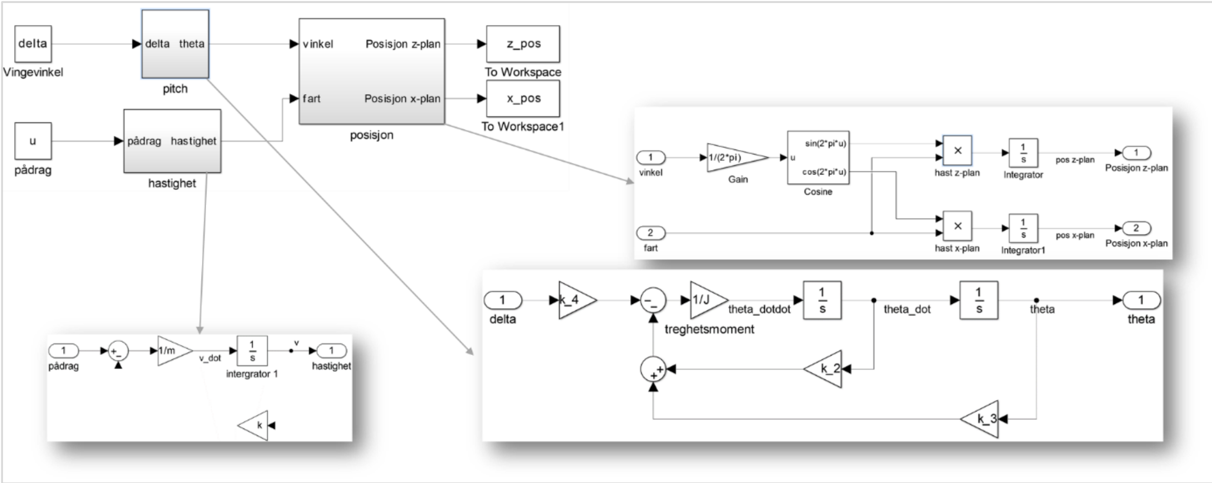
Figur 6 - fart i xz-planet

c) Systemet kan beskrives av tre 1. ordens diff. Ligninger og ett 2. ordens.

$$\dot{S_z} = v * \sin(\theta)$$
$$\dot{S_x} = v * \cos(\theta)$$
$$\dot{v} = \frac{1}{m}u - \frac{k}{m}v$$
$$\ddot{\theta} = -\frac{k_2\dot{\theta} + k_3\theta + k_4\delta}{J} = -\frac{k_2}{J}\dot{\theta} - \frac{k_3}{J}\theta - \frac{k_4}{J}\delta$$

Disse fire diff. Lignengene beskriver AUV-ens bevegelse fullstendig.
Diff. Ligningen for ϑ kan også skrives som to ordens diff. Ligninger, men har valgt her å beholde det som én 2. ordens siden det er det jeg har gått utifra i de andre oppgavene.
Modellen er multivariabel siden den har mer enn én inngang (rorvinkel og pådrag). Den har også to utganger, posisjon for hver akse. Det er ingen kvadratiske ledd i modellen så den er derfor lineær.

d) Fullstendig simulink modell:



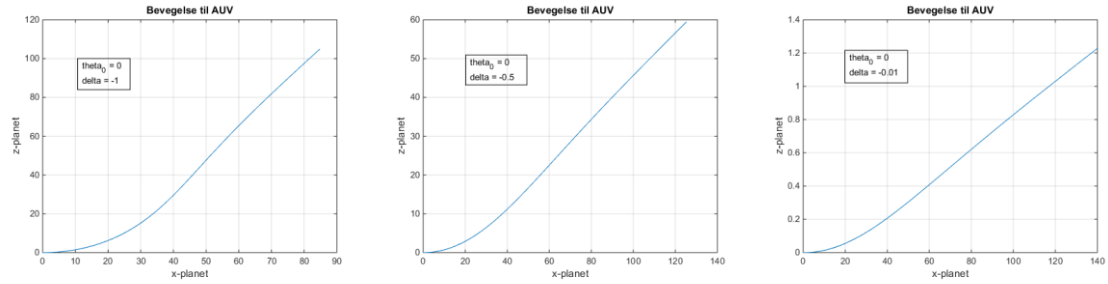
Figur 7 - Hele Simulink Systemet, NB: grafisk feil i hastighetssystemet, skal være en tilbakekobling via k-gainen.

Koden som styrer dette:

```
J = 15;
m = 200;

%innstillinger
u = 500;
delta = 0;
theta_0 = 0;
theta_dot0 = 0;
v_0 = 0;
t_sim = 100;

%proporsjonalitetskonstanter
k = 100;
k_2 = 5;
k_3 = 1;
k_4 = 1;

sim('pitch_hastighet', t_sim);

figure(1); clf(1);
plot(x_pos.data, z_pos.data);
xlabel('x-planet'); ylabel('z-planet')
title('Bevegelse til AUV');
grid on;
```

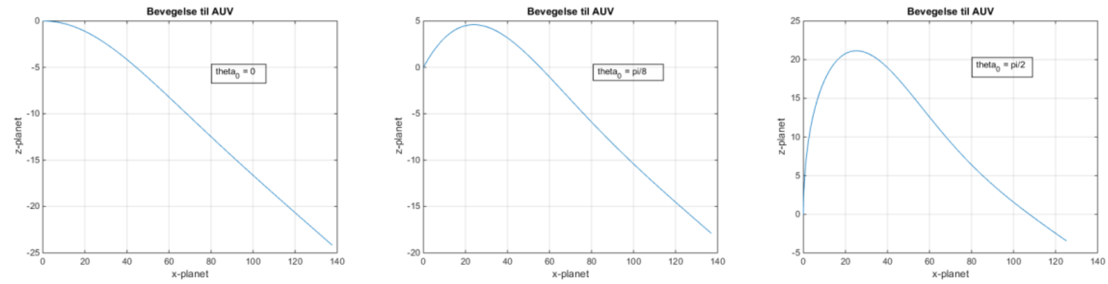
Modellen har fem integraler og er derfor et 5. ordens system. Det er ingen kvadratiske (e.l.) ledd i systemet så det er derfor også et lineært system.

e) Endrer %innstillinger i koden over for å teste de ulike kondisjonene.
   1. Setter vinkelen til å peke horisontalt og endrer δ.



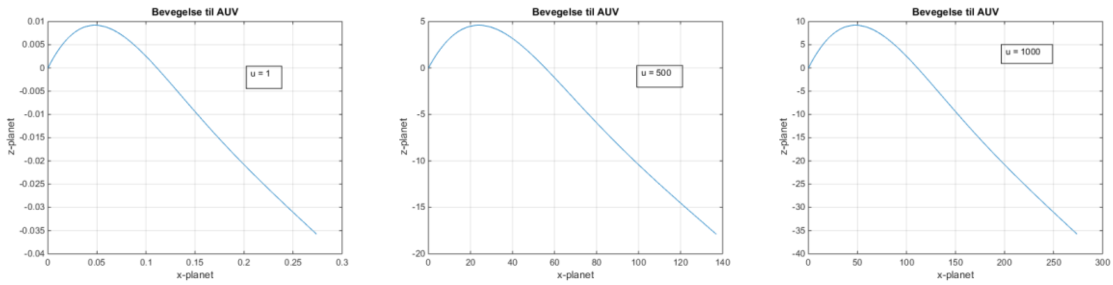
Formen på alle plottene er like, men hvis vi ser på akseverdiene så ser vi at med større (negativ) delta blir z-aksen også stor. Det gir mening fordi en større (negativ) delta betyr en større vinkel oppover (så lenge delta er mindre enn pi/2).

2. Setter $\delta = \pi/16$ og u = 500, endrer på θ_0.



Når θ_0 øker positivt starter AUV-en pekende oppover og en $\theta_0 = \pi/2$ vil i teorien tilsi at AUV-en starter pekende oppover, og i simuleringen stemmer dette slik vi kan se i grafene over.

3. Holder delta og theta_0 konstant, og endrer pådraget u.



I grafene er tre veldig ulike eksempler på positive pådrag, men formen er lik på alle. Det tilsier at pådraget kun påvirker avstanden langs de forskjellige aksene, men ikke formen på trajektoren.