

# Dataøving 2

## TTK5 Kalmanfiltrering og navigasjon

### Løsningsforslag

#### Oppgave 1

- a) Sammenhengen mellom pseudorange  $\rho$  og posisjon  $\mathbf{x}$  i ECEF rammen når man har  $n$  satellitter er:

$$\rho_i = \sqrt{(\mathbf{x}_{s_i} - \mathbf{x})^T (\mathbf{x}_{s_i} - \mathbf{x})} + c\tau^* \quad (1)$$

der  $i \in [1, \dots, n]$ ,  $\mathbf{x}_{s_i}$  er satellittposisjon  $i$ .  $c$  er lyshastigheten,  $\tau^*$  er mottakers klokkebias. For enkelhets skyld kaller vi  $c\tau^* = \tau$  som da blir klokkefeil målt i meter. På vektorform får man:

$$\boldsymbol{\rho} = \mathbf{h}(\mathbf{x}, \tau)$$

der  $\boldsymbol{\rho} = [\rho_1, \dots, \rho_n]^T$  og

$$\mathbf{h}(\mathbf{x}, \tau) = \begin{bmatrix} \sqrt{(\mathbf{x}_{s_1} - \mathbf{x})^T (\mathbf{x}_{s_1} - \mathbf{x})} + \tau \\ \vdots \\ \sqrt{(\mathbf{x}_{s_n} - \mathbf{x})^T (\mathbf{x}_{s_n} - \mathbf{x})} + \tau \end{bmatrix}$$

Den klassiske måten å løse dette på er ved hjelp av linearisering og rekursiv minste kvadraters løsning eller et utvidet (eng.: extended) Kalmanfilter. Linearisering av (1) gir:

$$\Delta \rho_i = \mathbf{e}_i^T \Delta \mathbf{x} + \tau$$

der

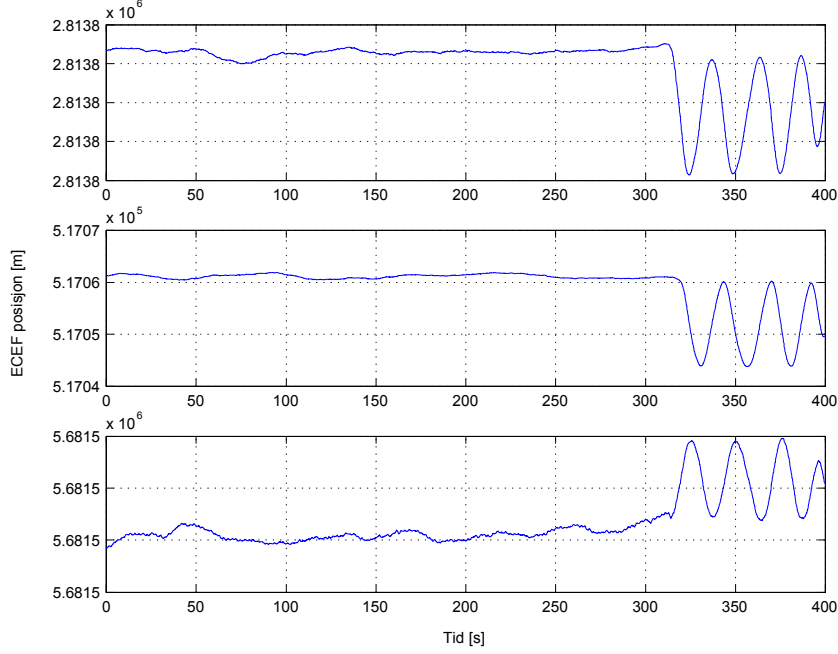
$$\mathbf{e}_i = \frac{\partial \mathbf{h}_i(\mathbf{x})}{\partial \mathbf{x}} = \frac{-(\mathbf{x}_{s_i} - \mathbf{x}_0)}{\sqrt{(\mathbf{x}_{s_i} - \mathbf{x}_0)^T (\mathbf{x}_{s_i} - \mathbf{x}_0)}}$$

$\mathbf{x}_0$  er til enhver tid det beste estimatet av  $\mathbf{x}$ , mens  $\Delta \mathbf{x}$  og  $\Delta \boldsymbol{\rho}_i$  er de lineariserte variable. På vektorform skrives dette

$$\Delta \boldsymbol{\rho} = \mathbf{E} \Delta \mathbf{x}_*$$

der  $\Delta \mathbf{x}_* = [\Delta \mathbf{x}^T, \Delta \tau]^T$  og

$$\mathbf{E} = \begin{bmatrix} e_1^{x_e} & e_1^{y_e} & e_1^{z_e} & 1 \\ e_2^{x_e} & e_2^{y_e} & e_2^{z_e} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ e_n^{x_e} & e_n^{y_e} & e_n^{z_e} & 1 \end{bmatrix} \quad (2)$$



Figur 1: ECEF posisjon: x-aksen er vist øverst, deretter y-aksen, og nederst vises z-aksen.

$\Delta \mathbf{x}_*$  finnes nå fra

$$\Delta \mathbf{x}_* = \mathbf{E}^\dagger \Delta \boldsymbol{\rho} \quad (3)$$

og posisjon og klokkefeil i iterasjon  $k$  finnes fra:

$$\mathbf{x}_*(k+1) = \mathbf{x}_*(k) + \Delta \mathbf{x}_*(k)$$

Legg merke til at  $k$  *ikke* har noe med tidsskritt å gjøre, alle iterasjonene skjer for hvert sample av data. Når løsningen har konverget, kan man bruke posisjon og klokkefeil i forrige tidsskritt som utgangspunkt for det neste, og da holder det som regel med en iterasjon pr. tidsskritt. Figur 1 viser posisjon dekomponert i ECEF-rammen. Hastighet i ECEF rammen  $\mathbf{v}$  beregnes fra følgende formel:

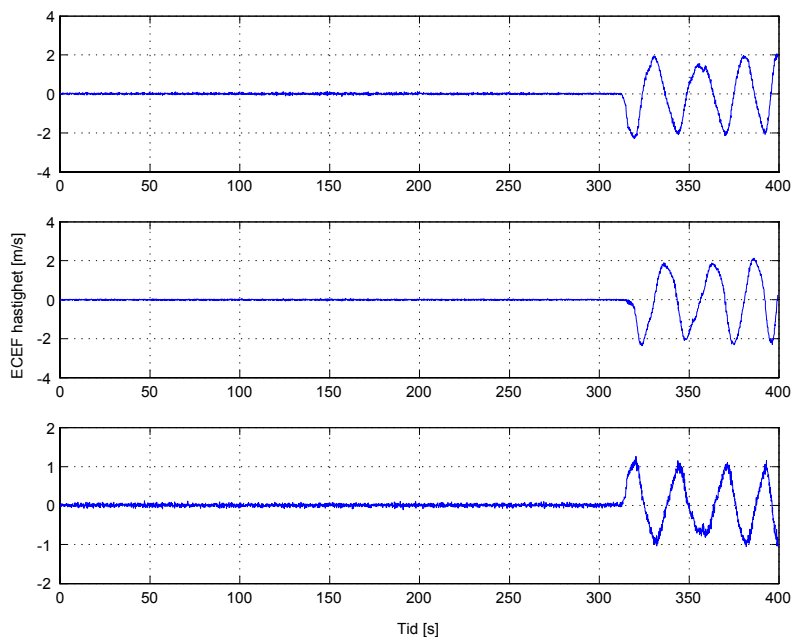
$$\delta_i = \mathbf{e}_i^T (\mathbf{v}_{s_i} - \mathbf{v}) + f$$

der  $\mathbf{v}_{s_i}$  er satellitthastighet i ECEF rammen for satellitt  $i$ ,  $f$  er klokke drift (frekvensbias).  $\mathbf{v}$  og  $f$  kan beregnes direkte fra:

$$\mathbf{v}_* = \mathbf{E}^\dagger (\boldsymbol{\delta} - \boldsymbol{\delta}_{\text{sat}}) \quad (4)$$

der  $\mathbf{v}_* = [\mathbf{v}^T, f]^T$ , og

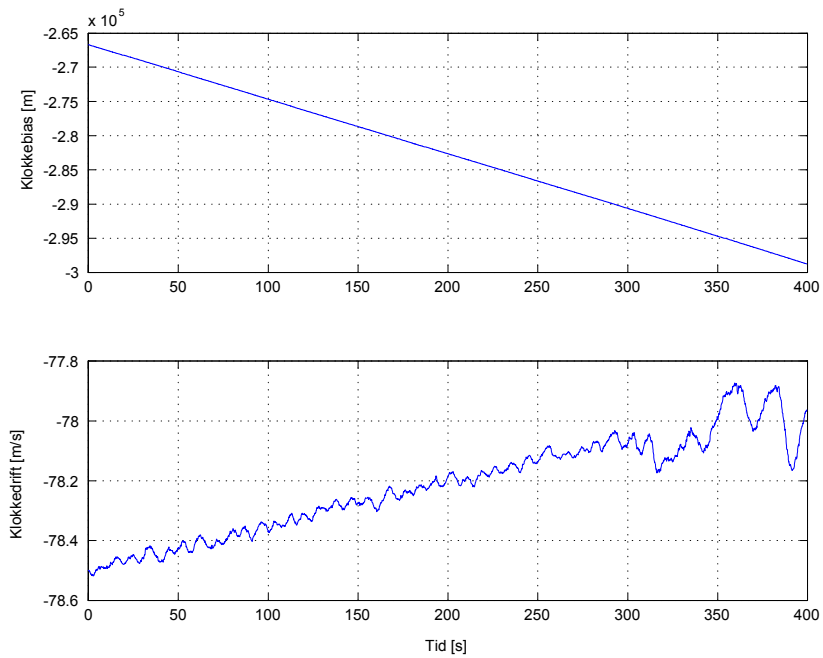
$$\boldsymbol{\delta}_{\text{sat}} = [\mathbf{e}_1^T \mathbf{v}_{s_1}, \dots, \mathbf{e}_n^T \mathbf{v}_{s_n}]^T \quad (5)$$



Figur 2: Hastighet i ECEF-rammen: x-aksen er vist øverst, deretter y-aksen, og nederst vises z-aksen.

er den kjente satellithastighetskomponenten av deltarange vektoren. Legg merke til at man trenger å finne en ca posisjon før man kan løse for hastighet, slik at man vet at  $\mathbf{E}^\dagger$  er korrekt. Posisjonen behøver ikke å være veldig nøyaktig siden  $\mathbf{E}^\dagger$  er lite sensitiv for posisjonsfeil. Hastighetssløsningen er vist i figur 2. Klokkebias og klokke drift er vist i figur 3.

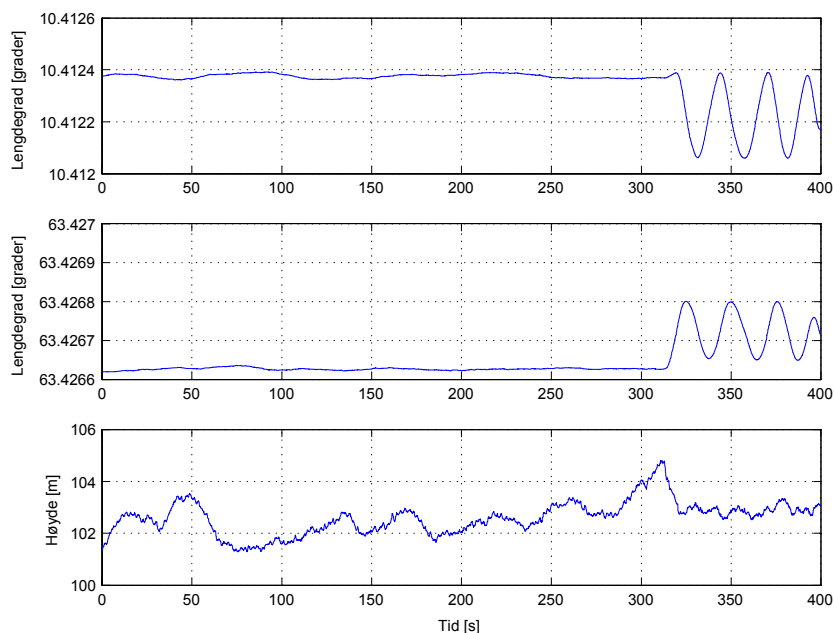
- b) Lengdegrad, breddegrad og høyde finnes ved å bruke algoritmen i kapittel 2.2.3 i læreboka. Resultatet er vist i figur 4. Hastigheter i NED rammen er vist i figur 5. Nord og Øst koordinater relativt til referansepunktet finnes ved å bruke lengde og breddegrader til å rotere ECEF posisjon til NED rammen. Deretter subtraherer man NED referanseposisjonen for å finne avviket. Figur 6 viser den horisontale trajektoren (Nord-Øst koordinater).
- c) NDOP, EDOP, HDOP og VDOP er vist i figur 7. Merk at line-of-sight vektorene i målematrisen må roteres til NED koordinater før man beregner DOP verdiene. Tar vi standardavviket på posisjonene i den statiske perioden får vi følgende resultat som vist i tabell 1. Av tabellen ser vi at Nord og Øst faktorene stemmer overens med standardavvikene på den måten at største standardavvik tilsvarer største DOP. Dersom man dividerer de



Figur 3: Klokkebias er vist øverst, og klokke drift er vist nederst.

	Nord	Øst	Ned
Standardavvik	0.49	0.44	0.42
Dilution of Precision	0.94	0.69	2.12
Tilsvarende std. på deltarange	0.52	0.64	0.20

Tabell 1: Oversikt over standardavvik og DOP. Standardavvik er i meter.

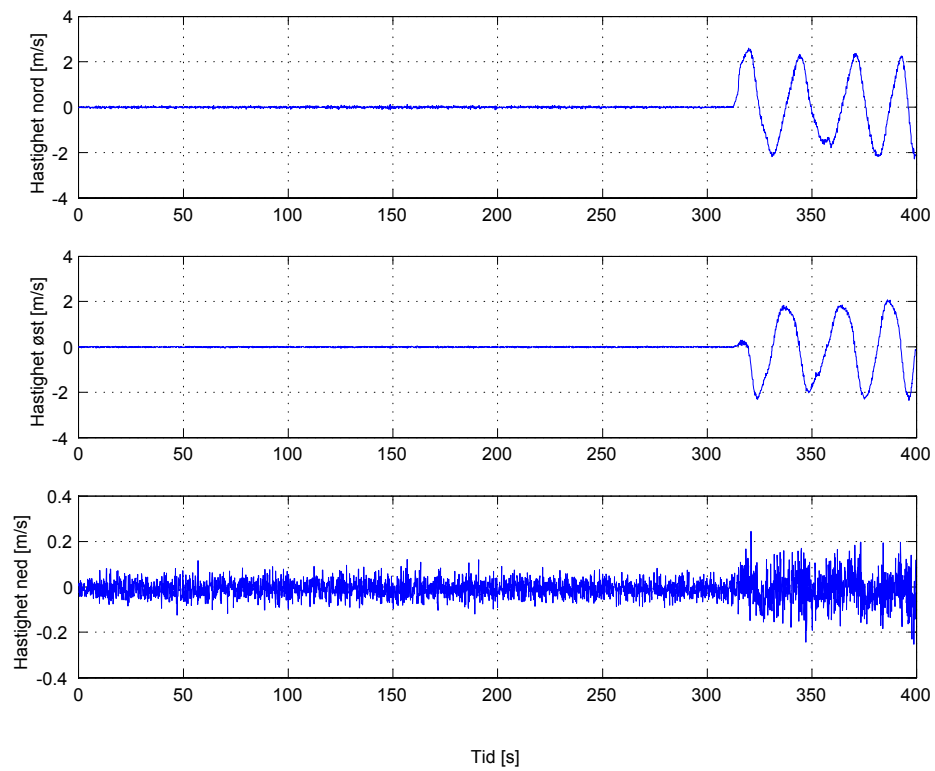


Figur 4: Lengdegrad, breddegrad og høyde.

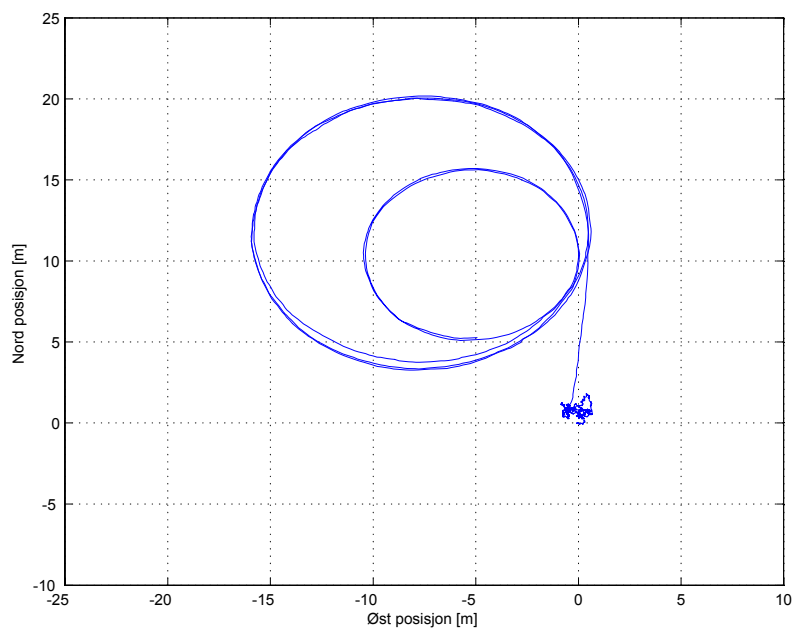
ulike standardavvikene i posisjon med tilsvarende DOP skal man komme frem til en omtrentlig nøyaktighet på pseudorangemålingene. Vi ser at det stemmer ganske bra for Nord og Øst posisjonene, og mindre bra på Ned posisjonen. Grunnen til at det ikke stemmer helt er at de ulike pseudorangemålingene ikke har likt standardavvik. For eksempel så har signalene fra satellitter med høy elevasjonsvinkel vanligvis en del lavere standardavvik enn signaler fra satellitter med lav elevasjonsvinkel. Når posisjonene blir beregnet blir det i pseudoinversmatrisen lagt ulik vekt på pseudorangemålingene for de forskjellige posisjonene. Den vertikale posisjonen er mer avhengig av satellitter med høy elevasjon, mens de horisontale posisjonene er mer avhengig av satellitter med lav elevasjon. Dermed blir den vertikale posisjonen bedre enn man skulle tro ut i fra DOP tallene.

## Oppgave 2

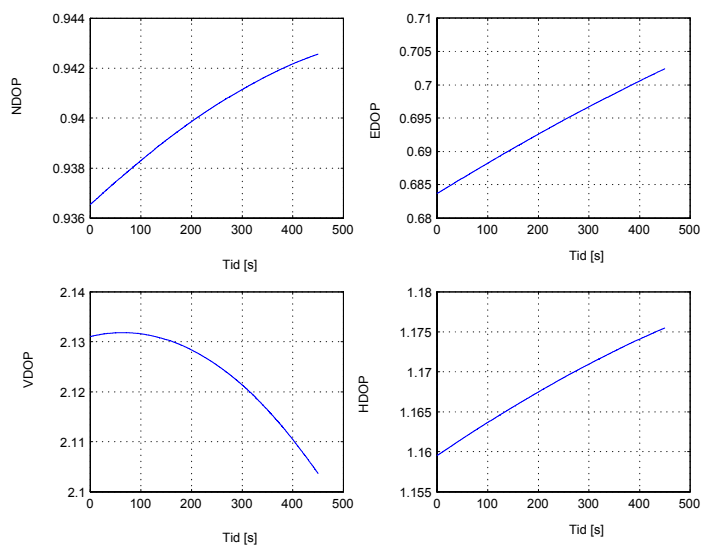
- Rull og stamp beregnes fra akselerometerdata ved å bruke ligning (6.153) i Farrell & Barth. Resultatet er vist i figur 8.
- Man kan bruke mange ulike filter til denne oppgaven. I korte trekk må man parametrisere vinklene ved hjelp av enhetskvaternioner, eller Eulervinkler, bruke den korrekte differensiallikningen (f.eks. ligning 6.51 i Farrell



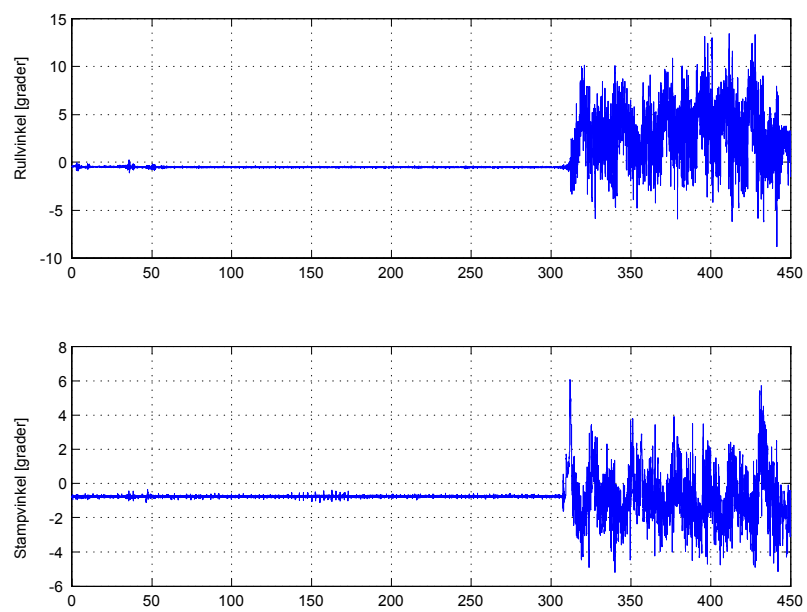
Figur 5: Hastighet i NED rammen.



Figur 6: Trajektoren sett ovenfra.

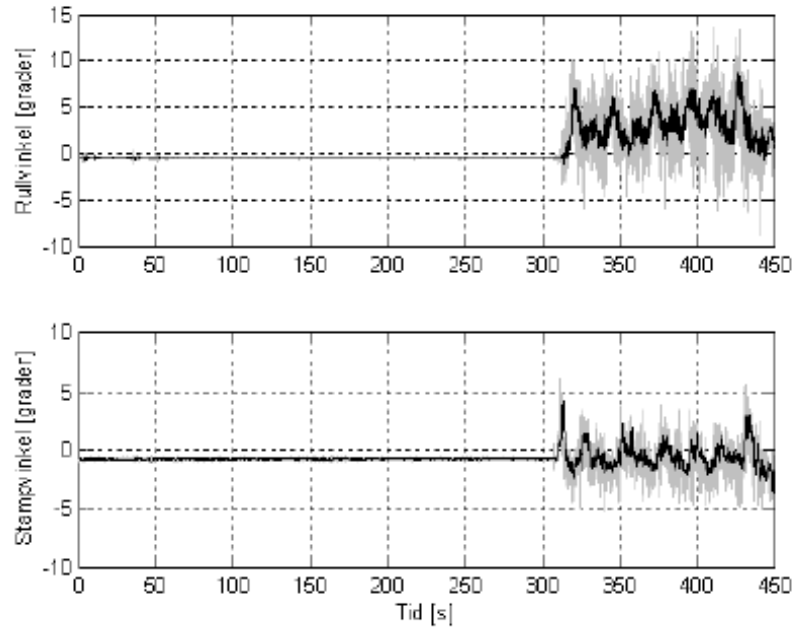


Figur 7: NDOP,EDOP,VDOP og HDOP.



Figur 8: Rullvinkel ( øverst) og stampvinkel beregnet direkte fra akselerometer-data.

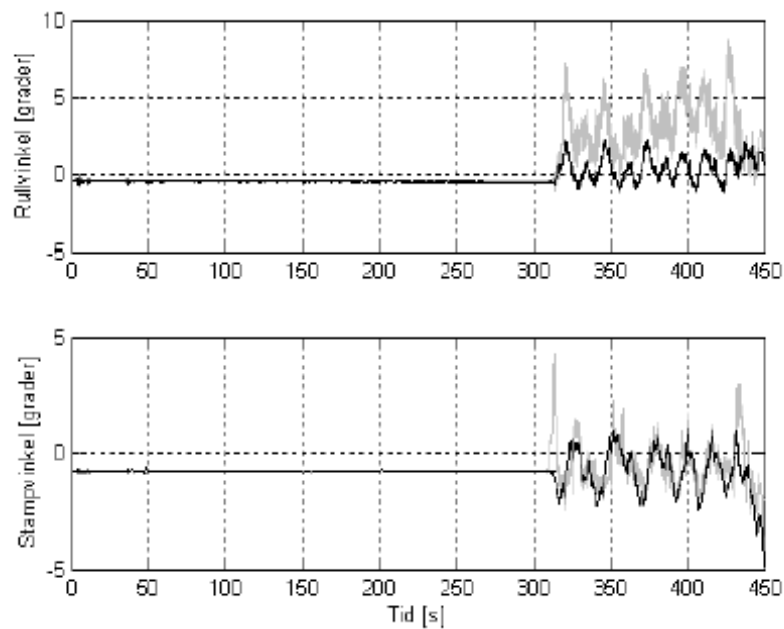




Figur 9: Rull og stampvinkler før integrasjon med gyro (grå), og etter integrasjon med gyro (svart)

& Barth for Eulervinkler) til å integrere den målte vinkelhastigheten og deretter bruke vinklene fra oppgave 2a) til å kompensere for drift på grunn av gyrofeil. Man bør også legge inn estimering av gyrobias. Resultatet for denne dataserien er vist i figur 9, og man ser at reduksjonen i støy er klar i forhold til oppgave 2a). Dersom man bare lavpassfiltrerer vinkelmålingene kan man oppnå det samme resultatet, men da går det ut over båndbredden. Med gyroer skjer ikke dette siden gyroene måler høyfrekvent vinkelbevegelse.

- c) Av posisjons og hastighetsdataene i oppgave 1 ser man at trajektoren er en sirkel som blir kjørt i noenlunde konstant fart. Da får man en sentripetalakselerasjon som har retning langs y-aksen til BODY rammen. Dette gir en stor bias på rullmålingen siden akselerometre ikke kan skille mellom gravitasjon (som brukes til rullmåling) og sentripetalakselerasjonen. Hastighet fra GPS og vinkelhastighet kan brukes til å regne ut denne sentripetalakselerasjonen, og dermed kan man kompensere for den. Dersom man ikke har hastighetsmålinger tilgjengelig bør man bruke veldig lav forsterkning i oppdatering fra akselerometrene under slike manøvre. Har man gode gyroer, kan man også vurdere å kutte oppdateringen i filteret



Figur 10: De svarte plottene viser vinkler når oppdatering fra akselerometrene er kuttet ved ca 270 sekund, mens de grå kurvene viser vinklene ved oppdatering fra akselerometrene.

helt mens slike manøvre pågår. Effekten av sentripetalakselerasjonen er vist i figur 10. Man kan se biasen i rull på nesten 3 grader, mens det ikke er noen synlig bias på stampvinkelen. Derimot kan man se en kraftig spiker i stampvinkelen på grunn av akselerasjon i x-retning i det man starter bevegelsen ved ca 310 sekund.