

Dokking av AUV ved bruk av feiltilstands- kalmanfilter

Kristine Elisabeth Aas Herje

Master i kybernetikk og robotikk

Innlevert: juni 2017

Hovedveileder: Edmund Førland Brekke, ITK

Medveileder: Are B. Willumsen, Kongsberg Maritime

Norges teknisk-naturvitenskapelige universitet
Institutt for teknisk kybernetikk

Takk

Takk til veilederene mine Edmund Førland Brekke og Are B. Willumsen for hjelp, støtte og gode innspill.

Sammendrag

Det er et økende behov for algoritmer for dokking av AUVer. Det som finnes i litteraturen i dag er løsninger på konkrete problemer, noe som gjør dem mer komplisert enn dokkingproblemet i seg selv. Denne oppgaven gjør et forsøk på sette opp et feiltilstands-Kalmanfilter som estimerer posisjonen til en AUV som nærmer seg en dokkingstasjon. Det brukes Eulervinkler for å utlede ligningene. Til slutt gjøres det simuleringer som gir en kvantifisering av nøyaktigheten.

Contents

1	Innledning	3
1.1	Bakgrunn	3
1.2	Agenda	3
1.3	Rotasjon	4
1.4	Målinger	5
2	Modellering	6
2.1	Tilstand	6
2.2	Målinger	6
3	Tilstandsligning	8
3.1	Posisjon	8
3.2	Hastighet	8
3.3	Orientering	8
3.4	Full tilstand	9
3.5	Målefeil	9
3.5.1	Feil i posisjon	9
3.5.2	Feil i hastighet	9
3.5.3	Feil i orientering	10
3.5.4	Feilestimat	11
3.6	Diskretisering	11
3.7	Måleligning	11
4	Kalmanfilter	13
4.1	Diskret Kalmanfilter	13
4.2	Feiltilstands Kalmanfilter	13
5	Simuleringer	15
5.1	Simulering 1	15
5.1.1	Trajektor og startbetingelser	15
5.1.2	Posisjonsfeil	15
5.1.3	Hastighetsfeil	17
5.1.4	Orienteringsfeil	18
5.1.5	De siste hundre sekundene	19
5.1.6	Oppsummering av simulering 1	20
5.2	Simulering 2	21
5.2.1	Posisjonsfeil	21
5.2.2	Hastighetsfeil	22
5.2.3	Orienteringsfeil	23
5.2.4	De siste hundre sekundene	24
5.2.5	Samendrag av simulering 2	25
6	Diskusjon	26
7	Konklusjon	27

1 Innledning

1.1 Bakgrunn

I dag brukes de fleste autonome undervannsfartøy (AUVer) til undersøkelsesoppdrag. Det er derimot et nytt bruksområde som stadig gjør seg mer gjeldende. Permanente undervannsinstallasjoner trenger vedlikehold som i dag blir gjort av fjernstyrte fartøy (ROVer). Om en kunne byttet ut ROVen med AUVer, kunne en forenklet mye arbeid. Dette krever imidlertid at en har mulighet til å lade opp batteriene deres. Måten AUVer blir ladet på i dag er ved å laste dem inn i et moderfartøy. Det å laste AUVene inn og ut krever en stor arbeidsmengde, og dermed melder behovet seg for å kunne dokke AUVene autonomtisk til en dokkingstasjon i forbindelse med lading. For ytterligere begrunnelse av hvorfor det er aktuelt å utvikle algoritmer for dokking av AUVer, se [4] og videre referanser derfra.

Det kritiske for å få AUVen til å dokke vellykket er å ha gode estimat på den relative posisjonen mellom AUVen og dokkingstasjonen. Den mest robuste måten å estimere en posisjon på er ved å bruke et GNSS-system, men når en befinner seg under vann er ikke dette et alternativ. Alternativ som utnytter tilsvarende prinsipper som GNSS er "ultra short baseline" (USBL), "Dopler velocity log" (DVL) og akustiske posisjoneringssystemer (APS). Sans-Muntadas og Miranda nærmer seg dokkingproblemet på denne måten i henholdsvis [6] og [3]. Det er også vanlig å utstyre AUVen med en treghetssensorenhet (IMU). En kan ikke estimere den relative posisjonen til AUVen i forhold til dokkingstasjonen basert på IMU-målingene alene, siden denne bare gir bevegelsen til AUVen og ikke sier noe om hvor den er. Sammen med en eller flere av teknikkene nevnt ovenfor kan en dermed få mye informasjon om hvor AUVen befinner seg i forhold til dokkingstasjonen. Gade har i [2] gjort en grundig redegjørelse av denne teknikken. [2] tar imidlertid for seg treghetsnavigasjon av AUVer generelt, og diskuterer ikke det spesifikke dokkingproblemet.

De overnevnte artiklene er bare noen av flere løsninger på AUV dokkingproblemet. Det er altså ikke et helt nytt felt, men det som er felles for alle løsningene er at de tar for seg helt spesielle system. De har hvert sitt bestemte utvalg av sensorer, navigasjonsalgoritmer og dokkingstasjon. De forskjellige løsningene er generelt sett mer kompliserte enn selve essensen av dokkingproblemet.

1.2 Agenda

Agendaen for denne oppgaven er å prøve å sette opp et enkelt system som estimerer posisjonen til en AUV på vei mot en dokkingstasjon. En standard løsning i navigasjonssystemer er et feiltilstands kalmanfilter. Her blir det tatt for seg å sette opp et slikt filter.

Siden systemet bare skal brukes i relativ nærhet til en dokkingstasjon er det en god tilnærming å anse jorden som flat i området der en bruker systemet, med konstant gravitasjon som antas kjent og står vinkelrett på den flate jorda. I denne oppgaven vil det også bli sett vekk fra forstyrrelser i miljøet (f.eks. strøm i vannet). Alle forstyrrelser modelleres som hvit støy på målingene. Systemet skal settes opp på en slik måte at forstyrrelser og flere spesifikasjoner skal kunne innkorporeres i systemet. Dette systemet skal kunne brukes som basis for mer komplekse og spesifikke systemer.

Et Kalmanfilter (KF) finner det optimale estimatet ved å minimere kovariansen til tilstanden. Det er flere grunner for å sette opp et feiltilstands-kalmanfilter (ESKF) i stedet for et vanlig et. Et kalmanfilter forutsetter lineære eller lineariserte systemer. Ved å bruke et ESKF i stedet for et KF kan en, i stedet for å linearisere hele systemet, bare linearisere feiltilstanden til systemet. Dette vil gjøre det lettere å oppdatere kovariansen til feilen og dette kan gjøres sjeldnere enn oppdateringene fra IMUen. ESKF kan også gi en god tredimensional representasjon av orienteringen i form av en rotasjonsmatrise, eulervinkler eller quaternioner.

1.3 Rotasjon

Rotasjonsmatriser, eulervinkler og quaternioner representerer rotasjoner mellom to koordinatsystemer. Her opereres det med to koordinatsystemer. Det ene koordinatsystemet vil ha origo i dokkingstasjonen. Aksene i koordinatsystemet vil være definert som følger: x-aksen peker mot nord, y-aksen peker mot øst og z-aksen peker mot sentrum av jorden eller vinkelrett nedover i og med at jorden her tilnærmes flat. Dette er et NED-system som beskrevet i [1]. Siden en har forenklet problemet til at jorden er flat og i ro med tyngdekraft som virker vinkelrett på planet, kan dette systemet tilnærmes et tregthetssystem. Altså et system der Newtons lover gjelder, et som ikke akselererer eller har ytre påvirkninger. Videre vil dette koordinatsystemet refereres til som \mathbf{I} . Det andre koordinatsystemet er et med origo i AUVens tyngdesenter. Aksene i dette koordinatsystemet vil være definert som: x-aksen fremover, y-aksen mot styrbord og z-aksen nedover. Dette er et body-system som beskrevet i [1] og vil derfor refereres til som \mathbf{B} .

Representasjonen av orienteringen beskriver hvordan \mathbf{B} er vridd i forhold til \mathbf{I} . Siden en vektor ikke har noen fast posisjon i rommet er det bare orienteringen som trengs for å transformere en vektor fra et koordinatsystem til et annet. Når en vektor skal transformeres fra \mathbf{B} til \mathbf{I} premultipliseres den med en rotasjonsmatrise, \mathbf{R}_B^I , som representerer den relative orienteringen. Kapittel 6.4 i [7] tar for seg hvordan en rotasjonsmatrise er definert og hvilke egenskaper den har. Ligning (6.88) i [7] viser at om en ønsker å transformere den andre veien,

premultipliserer en med den inverse rotasjonsmatrisen: $\mathbf{R}_I^B = \mathbf{R}_B^I^{-1} = \mathbf{R}_B^I^T$.

Kapittel 6.4.1 i [7] viser hvordan en kan sette opp en rotasjonsmatrise basert på enhetsvektorene til de to koordinatsystemene. Altså kan en ha et fast sett med enhetsvektorer for referanse-koordinatsystemet og et nominelt sett med enhetsvektorer som oppdateres med systemets feildynamikk for å beskrive det andre og bruke disse til å danne rotasjonsmatrisen.

Det går også an å sette opp rotasjonsmatrisen ved å gange sammen rotasjonene for hver av aksene. De klassiske eulervinklene er en vanlig måte å beskrive orienteringen i navigasjonsproblemer. Eulervinklene baserer seg på å gange sammen enkle rotasjoner. Eulervinklene blir kaldt roll, pitch og yaw og beskriver rotasjoner rundt henholdsvis x-, y- og z-aksen. I ligning (2.16) i [1] er rotasjonsmatrisen definert som først en rotasjon om z-aksen, så om y-aksen og til slutt en rotasjon rundt x-aksen. Det største problemet med eulervinklene er at det oppstår en singularitet med en pitch-vinkel på 90° . Mange fartøyer vil ha andre, større problemer om fartøyet skulle nærme seg en pitch-vinkel på 90° og de er derfor mye brukt til tross for singulariteten. For en AUV er det ikke nødvendigvis gitt at fartøyet ikke skal kunne ha 90° pitch, så det er viktig at en holder seg unna 90° pitch eller velger en annen måte å representere orienteringen på.

Orienteringen kan også bli representert ved eulerparametre eller enhetsquaternioner, heretter bare referert til som quaternioner. Quaternionene representerer orienteringen i fire parametre: En skalar og en vektor på tre elementer. Det er en mindre intuitiv måte å representere orienteringen på, men en slipper problemet med singularitet. I motsetning til eulervinklene er ikke quaternionene en minimal representasjon. Kapittel 6.7 i [7] tar grundig for seg hvordan quaternionene er representert og hvilke regneregler som gjelder for dem, samt hvordan en setter opp en rotasjonsmatrise basert på parametrene. Ligningene (6.166)-(6.168) viser hvordan en setter opp rotasjonsmatrisen og den transponerte rotasjonsmatrisen.

1.4 Målinger

Målingen av AUVens relative bevegelse vil komme fra en treghetsnavigasjonssenheter, IMU, som har akser og origo tilnærmet sammenfallende med \mathbf{B} . Disse målingene vil gå inn som pådrag i kalmanfilteret. I tillegg til disse målingene vil det være en posisjonsmåling, som en for eksempel kan få fra en USBL, denne vil brukes som måleoppdatering. Støyen på alle målingene vil modelleres som hvit, ukorrelet og lik i alle retnigner.

2 Modelling

I denne oppgaven vil skalarer oppgis som små bokstaver, vektorer (med unntak av tilstandsvektoren) som små tykke bokstaver og matriser som store tykke bokstaver. Superskript på en vektor indiker hvilket koordinatsystem den er dekomponert i. Subskript angir hvilke koordinatsystem vektoren går fra og til.

Eksempel: posisjonsvektoren \mathbf{p}_{IB}^I er vektoren mellom origo i koordinatsystemene \mathbf{B} og \mathbf{I} , der koordinatsystem \mathbf{B} beveger seg i forhold til referansekoordinatsystem \mathbf{I} , og vektoren er dekomponert i \mathbf{I} .

2.1 Tilstand

Tilstanden til AUVen er satt sammen av AUVens posisjon, hastighet og orientering.

Posisjonen er gitt som vektoren fra origo i koordinatsystem \mathbf{I} til origo i koordinatsystem \mathbf{B} . Altså vil lengden til vektoren ($|\mathbf{p}_{IB}^I|$) være avstanden mellom dokkingstasjonen og AUVen. I tilstandsvektoren vil posisjonen være dekomponert i koordinatsystem \mathbf{I} . Altså får en: $\mathbf{p}_{IB}^I = [x_{IB}^I, y_{IB}^I, z_{IB}^I]^T$.

Hastighetsvektoren er hastigheten koordinatsystem \mathbf{B} beveger seg med i forhold til koordinatsystem \mathbf{I} , dekomponert i \mathbf{I} . Som notasjon for dette brukes: $\mathbf{v}_{IB}^I = [u_{IB}^I, v_{IB}^I, w_{IB}^I]^T$.

Har her valgt å bruke eulervinkler for å representere orienteringen. Dette medfører at om systemet skal realiseres på en AUV som kan ha 90 °pitch, må den holde seg unna dette i dokkingområdet. Notasjonen som brukes for Eulervinklene følger notasjonen fra kapittel 2 i [1]: $\Theta_{IB} = [\phi_{IB}^B, \theta_{IB}^B, \psi_{IB}^B]^T$.

Den totale tilstandsvektoren blir dermed:

$$X = \begin{bmatrix} \mathbf{p}_{IB}^I \\ \mathbf{v}_{IB}^I \\ \Theta_{IB}^B \end{bmatrix} = \begin{bmatrix} x_{IB}^I \\ y_{IB}^I \\ z_{IB}^I \\ u_{IB}^I \\ v_{IB}^I \\ w_{IB}^I \\ \phi_{IB}^B \\ \theta_{IB}^B \\ \psi_{IB}^B \end{bmatrix} \quad (1)$$

2.2 Målinger

En treghetsnavigasjonsenhet er montert slik at akselerometere og gyroskoper avgir målinger av henholdsvis akselerasjon og vinkelhastighet i koordinatsys-

tem \mathbf{B} . Akselerometerene kan ikke merke forskjell på tyngdeakselerasjon og akselerasjon av AUVen. Gyroskopet vil også registrere jordrotasjonen. Både akselerometeret og gyroskopet bygger på Newtons lover for et treghetssystem og målingene er dermed gitt relativ til koordinatsystem \mathbf{I} , altså treghetskoordinatsystemet.

IMUen gir målingene:

- Akselerometermåling:

$$\mathbf{f}_{IB}^B = \mathbf{a}_{IB}^B + \mathbf{g}^B \quad (2)$$

- Gyroskopmåling:

$$\boldsymbol{\omega}_m = \boldsymbol{\omega}_{IB}^B + \boldsymbol{\omega}_e \quad (3)$$

\mathbf{f}_{IB}^b og $\boldsymbol{\omega}_m$ er rådata fra sensorene, \mathbf{g}^B er tyngdeakselerasjonen dekomponert i \mathbf{B} og $\boldsymbol{\omega}_e$ er jordrotasjonen.

Posisjonsmålingen som kommer inn vil gi en måling av $r = |\mathbf{p}_{IB}^I|$, vinkelen spendt mellom x-aksen og \mathbf{p}_{IB}^I , α , og vinkelen mellom x-y-planet og \mathbf{p}_{IB}^I , β .

- Posisjonsmåling:

$$\mathbf{z} = [r \quad \alpha \quad \beta]^T \quad (4)$$

3 Tilstandsligning

Vi ønsker en differensialligning som beskriver AUVens endring i tilstand. For å uttrykke denne ligningen bruker en tilstandsvektoren og målingene fra treghetsnavigasjonsenheten. Setter da opp ligningen:

$$\dot{X} = \mathbf{f}(X, \mathbf{u}) \quad (5)$$

Der

$$\mathbf{u} = [\mathbf{f}_{IB}^B, \boldsymbol{\omega}_{IB}^B]^T. \quad (6)$$

3.1 Posisjon

Ligningen for posisjonsendring er gitt ved:

$$\dot{\mathbf{p}}_{IB}^I = \mathbf{v}_{IB}^I. \quad (7)$$

3.2 Hastighet

Endring i hastighet tilsvarer akselerasjon, men den målingen vi har av akselerasjonen er målt i koordinatsystem B . Dermed har en følgende uttrykk for hastighetsendringen:

$$\dot{\mathbf{v}}_{IB}^I = \mathbf{a}_{IB}^I \quad (8)$$

$$= \mathbf{R}_B^I \mathbf{a}_{IB}^B \quad (9)$$

$$= \mathbf{R}_B^I \mathbf{f}_{IB}^B - \mathbf{R}_B^I \mathbf{g}^B \quad (10)$$

$$= \mathbf{R}_B^I \mathbf{f}_{IB}^B - \mathbf{g}^I \quad (11)$$

Rotasjonsmatrisen er avhengig av eulervinklene og er gitt av Fossen (2.18) [1]:

$$\mathbf{R}_B^I = \begin{bmatrix} \cos \psi \cos \theta & -\sin \psi \cos \phi + \cos \psi \sin \theta \sin \phi & \sin \psi \sin \phi + \cos \psi \cos \phi \sin \theta \\ \sin \psi \cos \theta & \cos \psi \cos \phi + \sin \phi \sin \theta \sin \psi & -\cos \psi \sin \phi + \sin \theta \sin \psi \cos \phi \\ -\sin \theta & \cos \theta \sin \phi & \cos \theta \cos \phi \end{bmatrix} \quad (12)$$

3.3 Orientering

Endring i orientering er gitt av Fossen (2.26) [1]:

$$\dot{\Theta}_{IB}^B = \mathbf{T}_\Theta \boldsymbol{\omega}_{IB}^B \quad (13)$$

$$= \mathbf{T}_\Theta (\boldsymbol{\omega}_{måling} - \boldsymbol{\omega}_e) \quad (14)$$

\mathbf{T}_Θ er avhengig av eulervinklene og gitt ved Fossen (2.28) [1]:

$$\mathbf{T}_\Theta = \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi / \cos \theta & \cos \phi / \cos \theta \end{bmatrix} \quad (15)$$

3.4 Full tilstand

$$\dot{X} = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} X + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \tilde{\mathbf{R}}_B^I & \mathbf{0} \\ \mathbf{0} & T_\Theta \end{bmatrix} u + \begin{bmatrix} \mathbf{0} \\ -\mathbf{g}^B \\ -T_\Theta \omega_e \end{bmatrix} \quad (16)$$

Innfører:

$$\mathbf{F} = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \quad \text{og} \quad \mathbf{G} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \tilde{\mathbf{R}}_B^I & \mathbf{0} \\ \mathbf{0} & T_\Theta \end{bmatrix} \quad (17)$$

3.5 Målefeil

Søker nå å sette opp ligninger som modellerer feilen i estimatene. Setter inn følgende modell for målingene fra IMUen.

$$\tilde{\mathbf{f}}_{IB}^B = \mathbf{f}_{IB}^B - \delta \mathbf{f}_{IB}^B \quad (18)$$

$$\tilde{\omega}_{IB}^B = \omega_{IB}^B - \delta \omega_{IB}^B \quad (19)$$

Der $\tilde{\mathbf{f}}_{IB}^B$ og $\tilde{\omega}_{IB}^B$ er målingene som kommer inn i systemet, \mathbf{f}_{IB}^B og ω_{IB}^B er der de faktiske verdiene til henholdsvis akselerasjon og vinkelhastighet og $\delta \mathbf{f}_{IB}^B$ og $\delta \omega_{IB}^B$ er feilen i målingene.

Vi kan dele opp tilstanden på samme måte:

$$X = \tilde{X} + \delta X \quad (20)$$

Der X er den sanne tilstanden, \tilde{X} er den nominelle tilstanden og δX er feiltilstanden.

3.5.1 Feil i posisjon

Ligning 7 med nominelltilstand og feiltilstand:

$$\dot{\tilde{\mathbf{p}}}_{IB}^I + \delta \dot{\mathbf{p}}_{IB}^I = \tilde{\mathbf{v}}_{IB}^I + \delta \mathbf{v}_{IB}^I \quad (21)$$

Antar en at den nominelle tilstanden har samme dynamikk som den sanne tilstanden får en:

$$\delta \dot{\mathbf{p}}_{IB}^I = \delta \mathbf{v}_{IB}^I \quad (22)$$

3.5.2 Feil i hastighet

Ligning 11 blir med nominelltilstand og feiltilstand:

$$\dot{\tilde{\mathbf{v}}}_{IB}^I + \delta \dot{\mathbf{v}}_{IB}^I = \tilde{\mathbf{R}}_B^I \delta \mathbf{R}_B^I (\tilde{\mathbf{f}}_{IB}^B + \delta \mathbf{f}_{IB}^B) - \mathbf{g}^I \quad (23)$$

Fossen (2.19) [1] gir følgende sammenheng:

$$\delta \mathbf{R}_B^I = \mathbf{I} + \mathbf{S}(\delta \Theta) \quad (24)$$

Får da:

$$\dot{\mathbf{v}}_{IB}^I + \delta \dot{\mathbf{v}}_{IB}^I = \tilde{\mathbf{R}}_B^I (\mathbf{I} + \mathbf{S}(\delta \Theta)) (\tilde{\mathbf{f}}_{IB}^B + \delta \mathbf{f}_{IB}^B) - \mathbf{g}^I \quad (25)$$

Første ordens tilnærming gir da:

$$\dot{\mathbf{v}}_{IB}^I + \delta \dot{\mathbf{v}}_{IB}^I = \tilde{\mathbf{R}}_B^I \tilde{\mathbf{f}}_{IB}^B + \tilde{\mathbf{R}}_B^I \delta \mathbf{f}_{IB}^B + \mathbf{S}(\delta \Theta_{IB}^B) \tilde{\mathbf{R}}_B^I \tilde{\mathbf{f}}_{IB}^B - \mathbf{g}^I \quad (26)$$

Regneregler for kryssprodukt gir:

$$\dot{\mathbf{v}}_{IB}^I + \delta \dot{\mathbf{v}}_{IB}^I = \tilde{\mathbf{R}}_B^I \tilde{\mathbf{f}}_{IB}^B + \tilde{\mathbf{R}}_B^I \delta \mathbf{f}_{IB}^B - \mathbf{S}(\tilde{\mathbf{R}}_B^I \tilde{\mathbf{f}}_{IB}^B) \delta \Theta_{IB}^B - \mathbf{g}^I \quad (27)$$

Antar at den nominelle tilstanden har samme dynamikk som den sanne tilstanden, også for hastigheten, og får:

$$\delta \dot{\mathbf{v}}_{IB}^I = -\mathbf{S}(\tilde{\mathbf{R}}_B^I \tilde{\mathbf{f}}_{IB}^B) \delta \Theta_{IB}^B + \tilde{\mathbf{R}}_B^I \delta \mathbf{f}_{IB}^B \quad (28)$$

3.5.3 Feil i orientering

Ligning 14 blir med nominelltilstand og feiltilsand:

$$\dot{\Theta}_{IB}^B + \delta \dot{\Theta}_{IB}^B = \mathbf{T}_\Theta \mathbf{T}_{\delta \Theta} (\tilde{\omega}_{IB}^B + \delta \omega_{IB}^B + \omega_e) \quad (29)$$

Fossen (2.32) [1] gir følgende sammenheng:

$$\mathbf{T}_{\delta \Theta} = \begin{bmatrix} 1 & 0 & \delta \theta \\ 0 & 1 & -\delta \phi \\ 0 & \delta \phi & 1 \end{bmatrix} \quad (30)$$

Tilnærmer:

$$\mathbf{T}_{\delta \Theta} \approx \mathbf{I} + \mathbf{S}(\delta \Theta_{IB}^B) \quad (31)$$

Førsteordens tilnærming gir da:

$$\dot{\Theta}_{IB}^B + \delta \dot{\Theta}_{IB}^B = \mathbf{T}_\Theta \tilde{\omega}_{IB}^B + \mathbf{T}_\Theta \delta \omega_{IB}^B + \mathbf{S}(\delta \Theta_{IB}^B) \mathbf{T}_\Theta \tilde{\omega}_{IB}^B + \mathbf{T}_\Theta \omega_e \quad (32)$$

Setter da inn at:

$$\mathbf{S}(\delta \Theta_{IB}^B) \mathbf{T}_\Theta \tilde{\omega}_{IB}^B = -\mathbf{S}(\mathbf{T}_\Theta \tilde{\omega}_{IB}^B) \delta \Theta_{IB}^B \quad (33)$$

Får da:

$$\dot{\Theta}_{IB}^B + \delta \dot{\Theta}_{IB}^B = \mathbf{T}_\Theta \tilde{\omega}_{IB}^B + \mathbf{T}_\Theta \delta \omega_{IB}^B - \mathbf{S}(\mathbf{T}_\Theta \tilde{\omega}_{IB}^B) \delta \Theta_{IB}^B + \mathbf{T}_\Theta \omega_e \quad (34)$$

Setter inn 14 og står igjen med:

$$\delta \dot{\Theta}_{IB}^B = -\mathbf{S}(\mathbf{T}_\Theta \tilde{\omega}_{IB}^B) \delta \Theta_{IB}^B + \mathbf{T}_\Theta \delta \omega_{IB}^B \quad (35)$$

3.5.4 Feilestimat

Feiltilstanden er altså gitt ved:

$$\delta \mathbf{X} = [\delta \mathbf{p}_{IB}^I \quad \delta \mathbf{v}_{IB}^I \quad \delta \Theta_{IB}^B], \quad (36)$$

mens feilen i målingen er:

$$\delta \mathbf{u} = [\delta \mathbf{f}_{IB}^B \quad \delta \omega_{IB}^B] \quad (37)$$

Oppsummert får vi følgende ligning for feiltilstanden:

$$\delta \dot{\mathbf{X}} = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -S(\tilde{\mathbf{R}}_B^I \tilde{\mathbf{f}}_{IB}^B) \\ \mathbf{0} & \mathbf{0} & -S(\tilde{\omega}_{IB}^B) \end{bmatrix} \delta \mathbf{X} + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \tilde{\mathbf{R}}_B^I & \mathbf{0} \\ \mathbf{0} & \mathbf{T}_\Theta \end{bmatrix} \delta \mathbf{u} \quad (38)$$

Innfører:

$$\delta \mathbf{F} = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -S(\tilde{\mathbf{R}}_B^I \tilde{\mathbf{f}}_{IB}^B) \\ \mathbf{0} & \mathbf{0} & -S(\tilde{\omega}_{IB}^B) \end{bmatrix} \quad (39)$$

3.6 Diskretisering

Pådragsmålingene \mathbf{u} kommer inn som allerede diskretiserte målinger. Eulers metode anvendt på systemligningen (5) gir da:

$$X(k+1) = (\mathbf{I} + \mathbf{F}h)X(k) + h\mathbf{G}\mathbf{u}(k) + \begin{bmatrix} \mathbf{0} \\ -\mathbf{g}^B \\ -\mathbf{T}_\Theta \omega_e \end{bmatrix}, \quad (40)$$

der $\mathbf{F} = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}$ og h er tidssteget.

Feilen $\delta \mathbf{u}$ er modelleres med hvit støy. Betegner dette som:

$$\delta \mathbf{u} = \boldsymbol{\xi}_u. \quad (41)$$

Diskretiserer (38) på samme måte som systemligningen og får:

$$\delta X(k+1) = (\mathbf{I} + \delta \mathbf{F}h)\delta X + \mathbf{G}\boldsymbol{\xi}_u \quad (42)$$

3.7 Måleligning

Målingen av avstanden mellom AUVen og dokkingstasjonen er gitt ved:

$$\mathbf{z} = \begin{bmatrix} r \\ \alpha \\ \beta \end{bmatrix}, \quad (43)$$

der r er lengden på vektoren \mathbf{p}_{IB}^I , α er vinkelen mellom \mathbf{p}_{IB}^I og x-aksen og β er vinkelen mellom x-y-planet og \mathbf{p}_{IB}^I .

Det er to mulige måter å inkorporere denne i systemet. En kan sende \mathbf{z} direkte inn i kalmanfilteret eller en kan omparametrisere \mathbf{z} til kartesiske koordinater før en sender målingen inn i kalmanfilteret.

Om en skal sende den opprinnelige målingen direkte inn i kalmanfilteret vil det innføre ulinearitet i kalmanfilteret. Denne må lineariseres og det medfører en jakobimatrise. Om en på den andre siden omparametriserer målingen før en sender den inn i kalmanfilteret må en ta hensyn til at støyen ikke påvirker den omparametriserte målingen lineært. Velger her det siste alternativet, å omparametrisere målingen før den sendes inn til kalmanfilteret.

Ønsker å omparametrisere målingen til kartesiske koordinater, oppgitt i koordinatsystem \mathbf{I} . Oppnår dette ved å sende den opprinnelige målingen gjennom ligningene:

$$x = r \cos \beta \cos \alpha \quad (44)$$

$$y = r \cos \beta \sin \alpha \quad (45)$$

$$z = r \sin \beta \quad (46)$$

Den omformede målingen er da $\mathbf{z} = [x \ y \ z]$. Når det senere refereres til \mathbf{z} er det denne der refereres til.

Som nevnt over vil ikke støyen påvirke den omparametriserte målingen lineært. Derivasjon av ligningene over gir:

$$\delta x = \delta r \cos \beta \cos \alpha - r \cos \beta \sin \alpha \delta \alpha - r \sin \beta \cos \alpha \delta \beta \quad (47)$$

$$\delta y = \delta r \cos \beta \sin \alpha + r \cos \beta \cos \alpha \delta \alpha - r \sin \beta \sin \alpha \delta \beta \quad (48)$$

$$\delta z = \delta r \sin \beta + r \cos \beta \delta \beta \quad (49)$$

Basert på dette kan en sette opp:

$$\text{cov} \left(\begin{bmatrix} \delta x \\ \delta y \\ \delta z \end{bmatrix} \right) = \mathbf{M} \text{cov} \left(\begin{bmatrix} \delta r \\ \delta \alpha \\ \delta \beta \end{bmatrix} \right) \mathbf{M}^T, \quad (50)$$

der:

$$\mathbf{M} = \begin{bmatrix} \cos \beta \cos \alpha & -r \cos \beta \sin \alpha & -r \sin \beta \cos \alpha \\ \cos \beta \sin \alpha & r \cos \beta \cos \alpha & -r \sin \beta \sin \alpha \\ \sin \beta & 0 & r \cos \beta \end{bmatrix} \quad (51)$$

4 Kalmanfilter

I et kalmanfilter trenger en kunnskap om initial tilstanden ($X_{pri}(0)$), kovariansmatrisen assosiert med den initielle feilen ($\mathbf{P}_{pri}(0)$), samt kovariansmatrisene assosiert med støy i prosessligningen (\mathbf{Q}) og støy i måleoppdateringen (\mathbf{R}). Gitt kovariansen til den hvite støyen for de forskjellige målesensorene finner en \mathbf{Q} og \mathbf{R} :

$$\mathbf{Q} = \begin{bmatrix} \sigma_f^2 \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \sigma_\omega^2 \mathbf{I} \end{bmatrix} \quad \text{og} \quad \mathbf{R} = \sigma_p^2 \mathbf{I}, \quad (52)$$

der σ er standardavvik og σ^2 er varians til målingene fra den enkelte sensoren.

Siden dette systemet bare navigerer i nærhet av dokkingstasjonen, kan en anta at det er ett annet navigasjonssystem på AUVen. Dette navigasjonssystemet kan gi oss $X_{pri}(0)$ med relativt liten $\mathbf{P}_{pri}(0)$.

4.1 Diskret Kalmanfilter

Ligningene for et diskret Kalmanfilter på generell form er gitt ved:

$$\mathbf{K}(k) = \mathbf{P}_{pri}(k) \mathbf{H}^T (\mathbf{H} \mathbf{P}_{pri}(k) \mathbf{H}^T + \mathbf{R})^{-1} \quad (53)$$

$$X_{post}(k) = X_{pri}(k) + \mathbf{K}(k)(z(k) - \mathbf{H} X_{pri}(k)) \quad (54)$$

$$\mathbf{P}_{post}(k) = (\mathbf{I} - \mathbf{K}(k) \mathbf{H}) \mathbf{P}_{pri}(k) (\mathbf{I} - \mathbf{K}(k) \mathbf{H})^T + \mathbf{K}(k) \mathbf{R} \mathbf{K}(k)^T \quad (55)$$

$$X_{pri}(k+1) = \Phi X_{post}(k) \quad (56)$$

$$\mathbf{P}_{pri}(k+1) = \Phi \mathbf{P}_{post}(k) \Phi^T + \mathbf{Q} \quad (57)$$

Ligningene (53)-(57) er hentet fra [5]. Kalmanfilteret forutsetter linearitet. Selv om differensialligningen for den nominelle tilstanden (5) er ulineær, så er differensialligningen for feiltilstanden (38) lineær, eller mer presist lineær tidsinvariant (LTV). Derfor kan vi designe et feiltilstnds Kalmanfilter for dokkingproblemet uten å linearisere.

4.2 Feiltilstands Kalmanfilter

Fra den diskretiserte feilutviklingen (42) får vi transisjonsmatrisen:

$$\Phi = \mathbf{I} + \delta \mathbf{F} \mathbf{h} \quad (58)$$

Siden X_{post} i denne oppgaven er et estimat av feilen blir (54) bare:

$$X_{post}(k) = \mathbf{K}(z(k) - \mathbf{H} X_{pri}(k)) \quad (59)$$

Siden både X_{pri} og x_{post} er estimater på feilen, innføres X_{nom} for å ta vare på den nominelle tilstandern:

$$X_{nom}(k+1) = (\mathbf{I} + \mathbf{F} \mathbf{h}) X_{nom}(k) + X_{pri}(k+1) + h \mathbf{G} \mathbf{u}(k) + \mathbf{C}, \quad (60)$$

der \mathbf{C} er kompensering for gravitasjon og jordrotasjon.

Siden målingen oppgis med samme parametrisering som i tilstandsvektoren får vi:

$$\mathbf{H} = [\mathbf{I} \quad \mathbf{0} \quad \mathbf{0}], \quad (61)$$

for alle k .

Siden den hvite støyen ikke legges til som rent additive ledd i prosessligningene, må en ta hensyn til dette når en setter opp kovariansmatrisen \mathbf{Q} . Omparametriseringen av målingen \mathbf{z} må tas hensyn til når en setter opp kovariansmatrisen \mathbf{R} . De nye kovariansmatrisene blir da:

$$\mathbf{Q}^* = \mathbf{G}\mathbf{Q}\mathbf{G}^T \quad \text{og} \quad \mathbf{R}^* = \mathbf{M}\mathbf{R}\mathbf{M}^T \quad (62)$$

Setter en alt dette sammen får en for iterasjoner med måleoppdatering:

$$\mathbf{K}(k) = \mathbf{P}_{pri}(k)\mathbf{H}^T(\mathbf{H}\mathbf{P}_{pri}(k)\mathbf{H}^T + \mathbf{R}^*)^{-1} \quad (63)$$

$$\mathbf{X}_{post}(k) = \mathbf{K}(k)(\mathbf{z}(k) - \mathbf{H}\mathbf{X}_{nom}(k)) \quad (64)$$

$$\mathbf{P}_{post}(k) = (\mathbf{I} - \mathbf{K}(k)\mathbf{H})\mathbf{P}_{pri}(k)(\mathbf{I} - \mathbf{K}(k)\mathbf{H})^T + \mathbf{M}\mathbf{R}^*\mathbf{M}^T \quad (65)$$

$$\mathbf{X}_{pri}(k+1) = \mathbf{\Phi}\mathbf{X}_{post}(k) \quad (66)$$

$$\mathbf{X}_{nom}(k+1) = (\mathbf{I} + \mathbf{F}h)\mathbf{X}_{nom}(k) + \mathbf{X}_{pri}(k+1) + h\mathbf{G}\mathbf{u}(k) + \mathbf{C} \quad (67)$$

$$\mathbf{P}_{pri}(k+1) = \mathbf{\Phi}\mathbf{P}_{post}(k)\mathbf{\Phi}^T + h\mathbf{G}\mathbf{Q}^*\mathbf{G}^T \quad (68)$$

De iterasjonene som kjøres uten måleoppdatering vil ha $\mathbf{X}_{post}(k) = 0$ og dermed $\mathbf{X}_{pri} = 0$. Dermed blir (65) og (67) henholdsvis:

$$\mathbf{P}_{post}(k) = \mathbf{P}_{pri}(k) \quad (69)$$

$$\mathbf{X}_{nom}(k+1) = (\mathbf{I} + \mathbf{F}h)\mathbf{X}_{nom}(k) + h\mathbf{G}\mathbf{u}(k) + \mathbf{C} \quad (70)$$

5 Simuleringer

5.1 Simulering 1

5.1.1 Trajektor og startbetingelser

I denne simuleringene har AUVen en sann startposisjon 500 m direkte nord for dokkingstasjonen. Fra startposisjonen beveger AUVen seg i en rett linje sørover mot dokkingstasjonen med en hastighet på 1 m i sekundet. Det itereres med tidssteg på 0,1 sekund. Måleoppdatering blir gjort hvert hundrede tidssteg, altså hvert tiende sekund. Hastigheten og orienteringen anses som initiell og konstant. Initiell orientering er slik at AUVen allerede er rettet mot dokkingstasjonen.

Det er lite realistisk at AUVen har tilgjengelig en eksakt initiell posisjon når den starter kalmanfilteret. Derfor er den initielle posisjonen som gis til kalmanfilteret:

$$X_{nom}(0) = \begin{bmatrix} 499m \\ 1m \\ 1m \\ -0,99m/s \\ 0,01m/s \\ 0,01m/s \\ 0,001rad \\ 0,001rad \\ \pi - 0,001rad \end{bmatrix}. \quad (71)$$

Disse startbetingelsene er innenfor den initielle kovariansen som er satt:

$$P_{pri}(0) = \begin{bmatrix} (\sqrt{10}m)^2 \cdot \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (\sqrt{0,1}m/s)^2 \cdot \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & (\sqrt{0,0001}rad)^2 \cdot \mathbf{I} \end{bmatrix} \quad (72)$$

Parametre for støyen i de forskjellige sensorene er gitt i tabell 1.

Table 1: Støy

Akselerometer:	$50\mu g/\sqrt{Hz}$	=	$4,905 \cdot 10^{-4}m/s\sqrt{Hz}$
Gyroskop:	$0,15^\circ/\sqrt{h}$	=	$4,36 \cdot 10^{-5}rad/\sqrt{s}$
Posisjonsmåling, avstand:	$\sqrt{0,1}m$	=	0,01m
Posisjonsmåling, vinkler:	$0,25^\circ$	=	$4,4 \cdot 10^{-3}rad$

5.1.2 Posisjonsfeil

Ser av figur 1 at kovariansen til feilen i posisjonen p_{IB}^I stiger raskt i starten. Delta x får en stabil kovarians etter de første tre måleoppdateringene. Denne

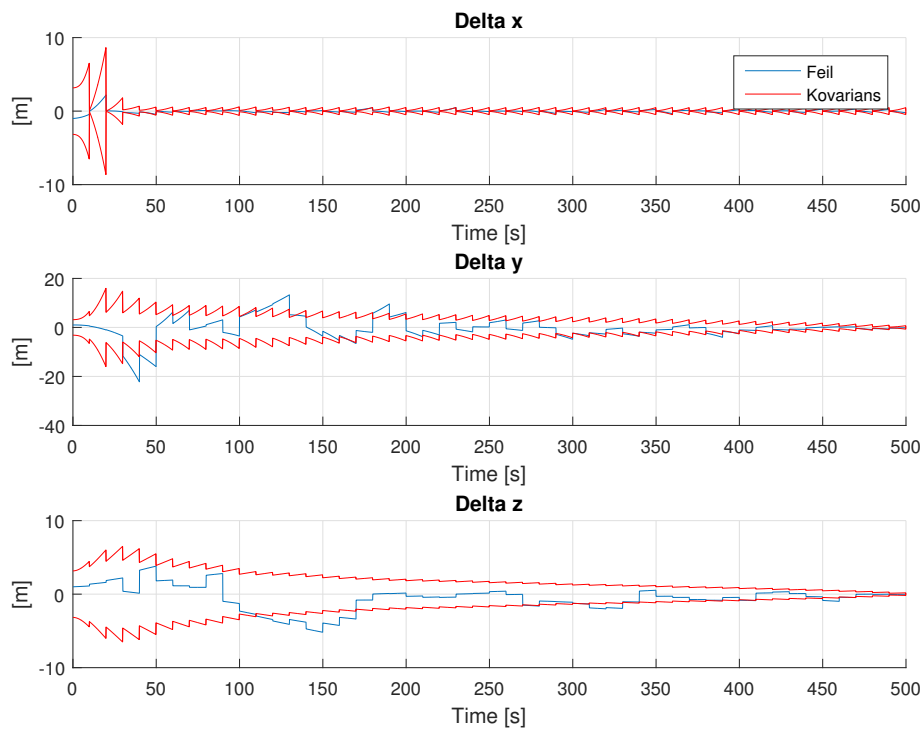


Figure 1: Feilen i estimatet for x, y og z

kovariansen holdes ut hele resten av simuleringen. Estimatet til x har en gjennomsnittsfeil på 1,15 cm og er innenfor kovariansen i 93,72% av tidsstegene. Årsaken til den høye kovariansen er sannsynligvis den urealistisk enkle trajektoren som AUVen følger inn mot dokkingstasjonen.

Kovariansen til delta y stiger i starten dobbelt så mye som kovariansen til x og z. Etter den første måleoppdateringen ser kovariansen ut til å bli gradvis mindre og mindre ut simuleringen. Der stigningen mellom måleoppdateringene aldri blir større enn nedgangen i måleoppdateringene. Den gjennomsnittlige verdien til delta y er -36,01 cm. Dette skyldes nok at den er vesentlig større i første halvdel av simuleringen i forhold til delta x, siden delta x allerede etter tre måleoppdateringer er blitt så liten. Delta y er innenfor kovariansen 76,62% av tidsstegene, noe som er mye mer realistisk enn de 93,72% for delta x, men fremdeles høyere enn forventet.

Kovariansen til delta z stiger nesten like mye som kovariansen til delta x i starten, deretter går også den til å bli lav mot slutten av simuleringen der AUVen kommer frem til dokkingstasjonen. Feilen i z-retning er innenfor kovariansen i 75,64% av tidsskrittene og har en gjennomsnittsverdi på -54,10 cm,

noe som ikke er så ulikt det som er for delta y.

De høye kovariansene i begynnelsen gjør det vanskelig å se av disse plottene hvor nøyaktig posisjonsestimaten \mathbf{p}_{BI}^I er når AUVen nærmer seg dokkingstasjonen. Den enkle trajektoren gjør at feilen i estimatene blir urealistisk små. Estimaten for x-posisjonen er betydelig mer urealistisk enn for y- og z-posisjonen.

5.1.3 Hastighetsfeil

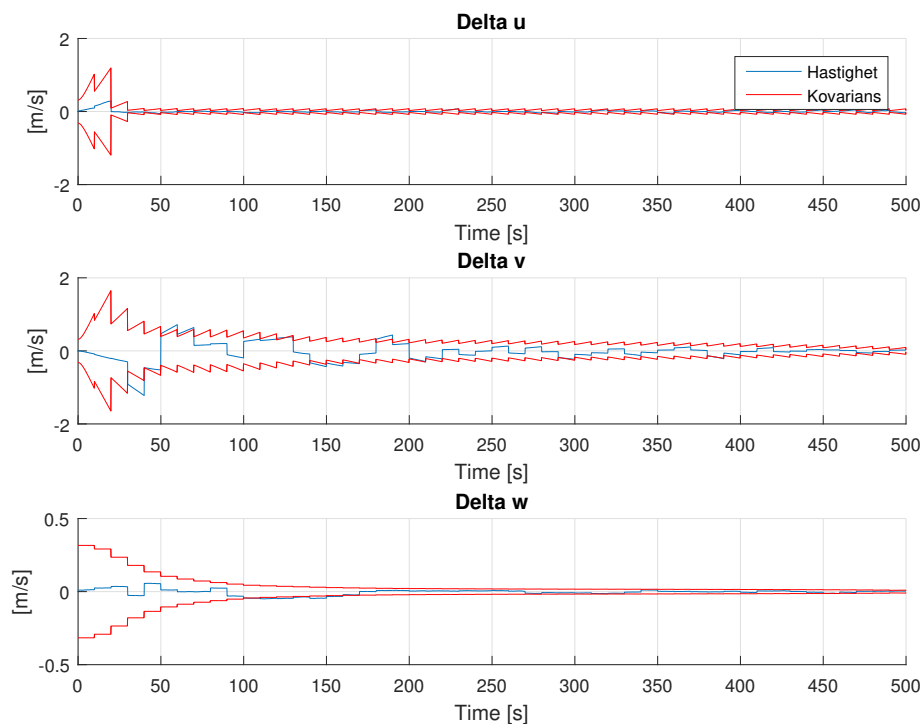


Figure 2: Feilen i estimatet for u, v og w

Figur 2 viser feilene i estimatet for hastigheten \mathbf{v}_{IB}^I . Kovariansen til hastigheten i x- og y-retning stiger en del de første tjue tidsskrittene. Deretter ser delta u ut til å stabiliseres mens delta v fortsetter å gå nedover gjennom hele simuleringen. Kovariansen til delta v går fort nedover de første måleoppdateringene. Etter de første måleoppdateringene ser kovariansen ut til å få en konstant rate nedover.

Delta u er innenfor kovariansen i 99,44% av tidsstegene og har en gjennomsnittsverdi på 0,28 cm/s, mens delta v er innenfor kovariansen 82,56% av tidsskrittene og har en gjennomsnittsverdi op 2,59 cm/s. Estimaten er begge urealistisk mye innenfor kovariansen og derfor kan en også anta at estimatene er urealistisk gode. I seksjonen over ble årsaken til det samme fenomenet lagt på den enkle

trajektoren, og det er nok rimelig å anta at dette er årsaken her også.

Kovariansen til delta w ser ut til å stå stille mellom måleoppdateringene. Når det kommer en måleoppdatering går kovariansen ned, frem til den ser ut til å stabiliseres. I det mindste blir kovariansen så lav at det er vanskelig å se den på dette plottet. Feilestimatet er innenfor kovariansen 87,92% av tidsskrittene i simuleringen og har en gjennomsnittsverdi på 0,36 cm/s. Altså ser vi også her at det er et urealistisk godt estimat.

5.1.4 Orienteringsfeil

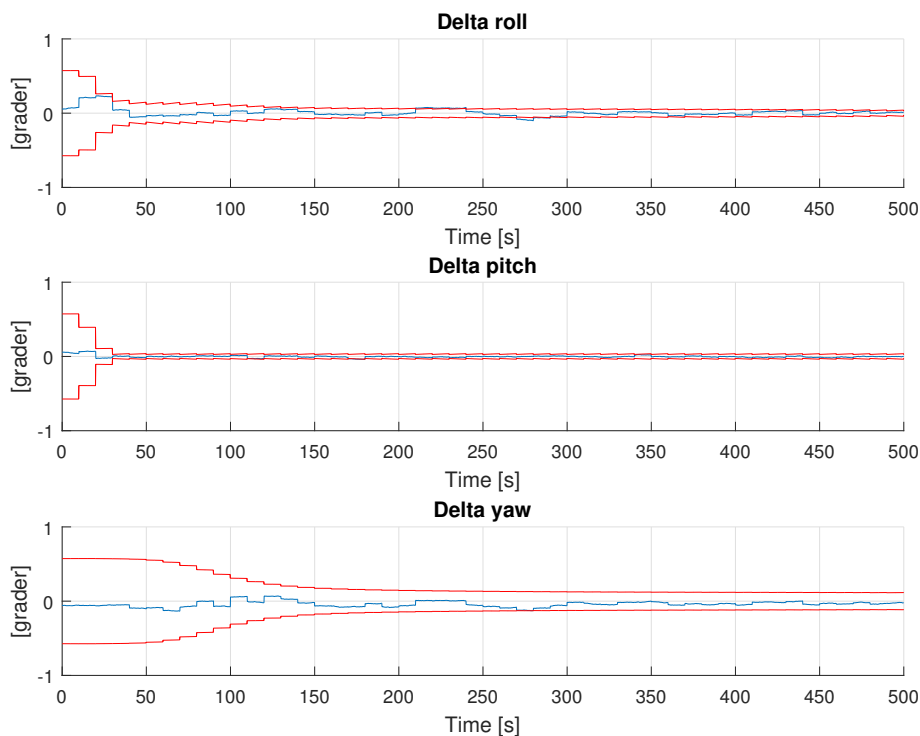


Figure 3: Feilen i estimatet for roll, pitch og yaw

I orienteringen er andelen estimerer innenfor kovariansen enda høyere enn for posisjon og hastighet. Det er 94,60%, 100% og 96,00% for henholdsvis delta roll, delta pitch og delta yaw. De to hovedgrunnene til at disse er så urealistisk høye er er nok den enkle trajektoren og at støyen til alle sensorene er regnet som ren hvit støy.

Kovariansene til feil i orientering har ikke stigningen som er blitt observert for kovariansene til posisjon og hastighet. Det er kovariansen til delta pitch som

er den første som stabiliseres. Denne ser ut til å bruke tre måleoppdateringer før kovariansen er stabil for resten av simuleringen. Kovariansen til delta roll blir tilsvarende fort liten, men etter å ha blitt ganske liten fortsetter den å minke gradvis ut simuleringen. Den gjennomsnittlige verdien til henholdsvis delta roll og delta pitch er $7,4 \cdot 10^{-3}^\circ$ og $3,3 \cdot 10^{-3}^\circ$.

Kovariansen til delta yaw bruker lengre tid på å stabiliseres og blir heller ikke like liten som kovariansen til delta roll og delta pitch. Dette skyldes at det er en langsommere prosess. Feilen i estimatet er innenfor kovariansen 96,00% av simuleringen og har en gjennomsnittsverdi på $0,0431^\circ$. Som for roll og pitch kan en også her anta at hovedgrunnene til de urealistisk gode tallene er den enkle trajektoren og de enkle målingene.

5.1.5 De siste hundre sekundene

Med en fart på 1 m/s er de siste hundre sekundene også de siste hundre metrene. I dette plottet er det mye lettere å lese av hva som skjer på slutten av simuleringen. Det er denne delen som er mest interessant å se på. Det er når AUVen nærmer seg dokkingstasjonen det er kritisk at det er et godt estimat. Som det er diskutert i forbindelse med plottene som viser hele simuleringen er det noe vel optimistiske resultat i simuleringen. Likevel gir simuleringen en pekepinn på hva som er mulig å nærme seg i en mer realistisk situasjon.

Øverst i figur 4 ser en det som bare så vidt kan anes i figur 1, at kovariansen til delta x blir stabil, mens kovariansen til delta y og delta z fortsetter å synke helt til AUVen kommer frem til dokkingstasjonen. De siste metrene frem til dokkingstasjonen er usikkerheten mindre enn en halv meter i både x-, y- og z-retning. En ser også at usikkerheten går mye ned hver gang det kommer en ny måleoppdatering. En måte en kunne ha utnyttet dette ville vært å sette opp frekvensen på måleoppdateringene, når en nærmet seg dokkingstasjonen. På den måten kunne en oppnådd mer nøyaktig estimat, nærme dokkingstasjonen.

Den midterste rekken i figur 4 viser kovarians og feilestimat for hastigheten til AUVen, også i denne rekken kan vi kjenne igjen det vi ante i figuren som viser hele simuleringen, figur 2. Kovariansen til delta u stabiliseres, mens kovariansene til delta v og delta w går nedover helt til AUVen kommer frem til dokkingstasjonen. Hastigheten i z-retning er den som har høyest presisjon med en kovarians under 2 cm/s. Kovariansen til delta v er cirka dobbelt så stor som kovariansen til delta u, når AUVen er hundre meter unna dokkingstasjonen. På de hundre sekundene det tar å komme helt frem til dokkingstasjonen ser det derimot ut til at presisjonen til hastigheten i både x- og y-retning er noenlunde like, under 10 cm/s.

Nederst i figur 4 kan en se et tydelig bilde av den estimerte feilen og den estimerte kovariansen til AUVens orientering. Det er tydelig å se at det er overkonsistens i simuleringen av kovariansen. Som det ble sett i figur 3 er kovariansen lavest

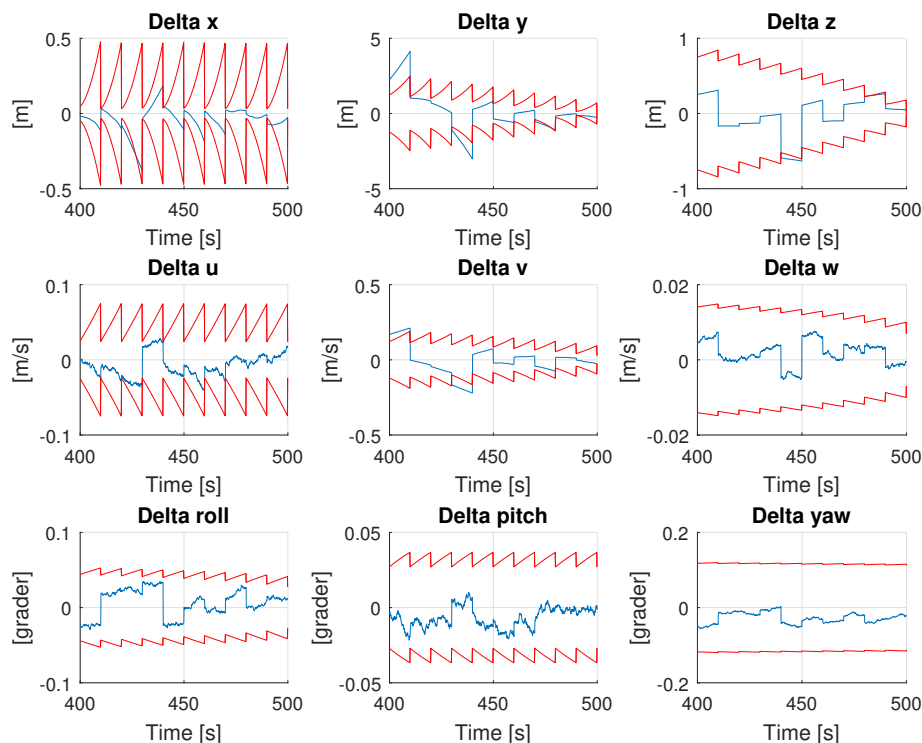


Figure 4: Feilen i alle elementene i tilstandsvektoren for de siste hundre sekundene av simuleringen

for delta pitch og høyest for delta yaw, samt at kovariansen til delta pitch og delta yaw stabiliseres, mens kovariansen til delta roll fortsetter å synke helt til slutten av simuleringen. I og med at det her er sett vekk i fra alle forstyrrelser som kan påvirke AUVens orientering og at trejektoren er slik at orienteringen skal være konstant gjennom hele simuleringen er det kanskje å forvente at det blir overkonsistens.

5.1.6 Oppsummering av simulering 1

Det er en gjennomgående høy kovarianst for alle tilstandene. Feilestimatene av de forskjellige elementene i tilstandsvektoren er innenfor kovariansen mellom 75% og 100% av simuleringene, mens det som skal forventes er at de er innenfor 67% av tiden. Hovedgrunnene til overkonsistensen ligger antagelig på en enkel trajektor, målinger som er anat uten bias og at forstyrrelser fra omgivelsene ikke er tatt i betraktning.

Til tross for forenklingene gjort her peker simuleringene klart mot at det er mulig å få et estimat med høy presisjon i forbindelse med dokkingproblemet.

5.2 Simulering 2

Perfekt initialisering

I denne simuleringen er alt likt som i den forrige utenom initialtilstanden som blir gitt til kalmanfilteret og den initielle usikkerheten til denne. Det er tidligere nevnt at et annet navigasjonssystem navigerer AUVen frem til den kommer til området der dokkingen begynner. Da er det også rimelig å anta at dette andre navigasjonssystemet kan gi en initiell posisjon med liten usikkerhet. I denne simuleringen er det gitt eksakt riktig initialtilstand og en nærmest neglisjerbar usikkerhet.

$$\mathbf{P}_{pri}(0) = \begin{bmatrix} (0,0032\text{m})^2 \cdot \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (0,001\text{m/s})^2 \cdot \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & (0,00032\text{rad})^2 \cdot \mathbf{I} \end{bmatrix} \quad (73)$$

5.2.1 Posisjonsfeil

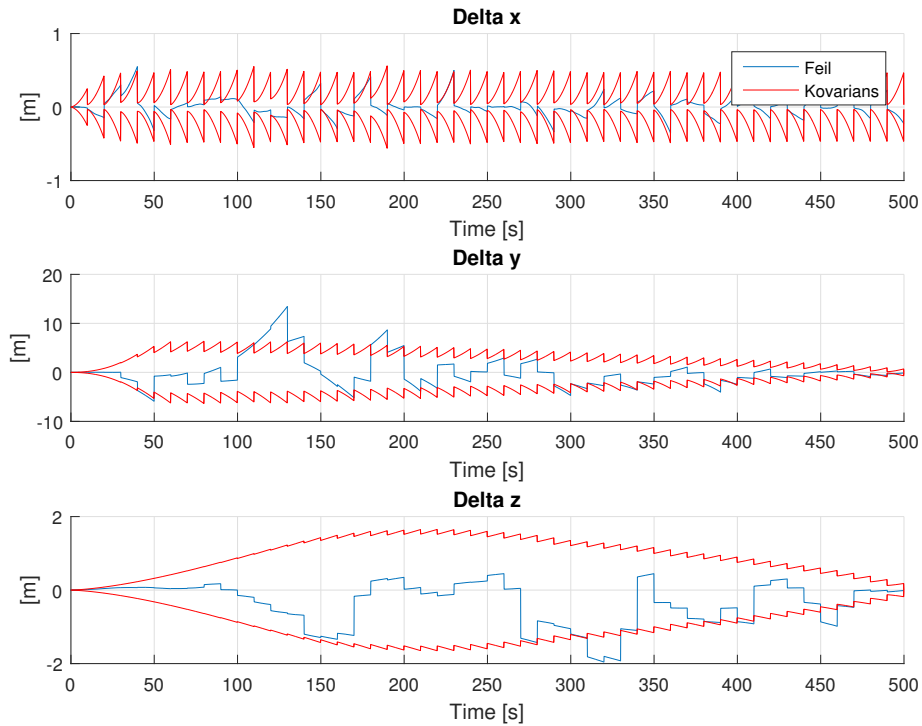


Figure 5: Feilen i estimatet for x, y og z

Figur 5 er helt uten de høye kovariansene som kunne observeres i figur 1, noe som medfører at det i figur 5 er mye tydeligere hvordan kovariansen og delta-estimatet utvikler seg gjennom simuleringen. Denne simuleringen gir også en

mye høyere andel av feilestimatene innenfor kovariansen, med 88,76% for delta x, 78,36% for delta y og 89,76% for delta z. Den gjennomsnittlige feilen i estimatene ser derimot ut til å være mindre i denne simuleringen der henholdsvis delta x, delta y og delta z har gjennomsnittsverdier på 0,72 cm, 0,86 cm og 35,84 cm.

Kovariansen til delta x ser ut til å bruke to måleoppdateringer før den stabiliseres. Deretter ser den ut til å ha den samme karakteristikken ut simuleringen. For delta y stiger kovariansen gradvis de første femti sekundene, for deretter å synke jevnt ut simuleringen. Kovariansen til delta z stiger gradvis de første to hundre sekundene, for så å synke gradvis.

5.2.2 Hastighetsfeil

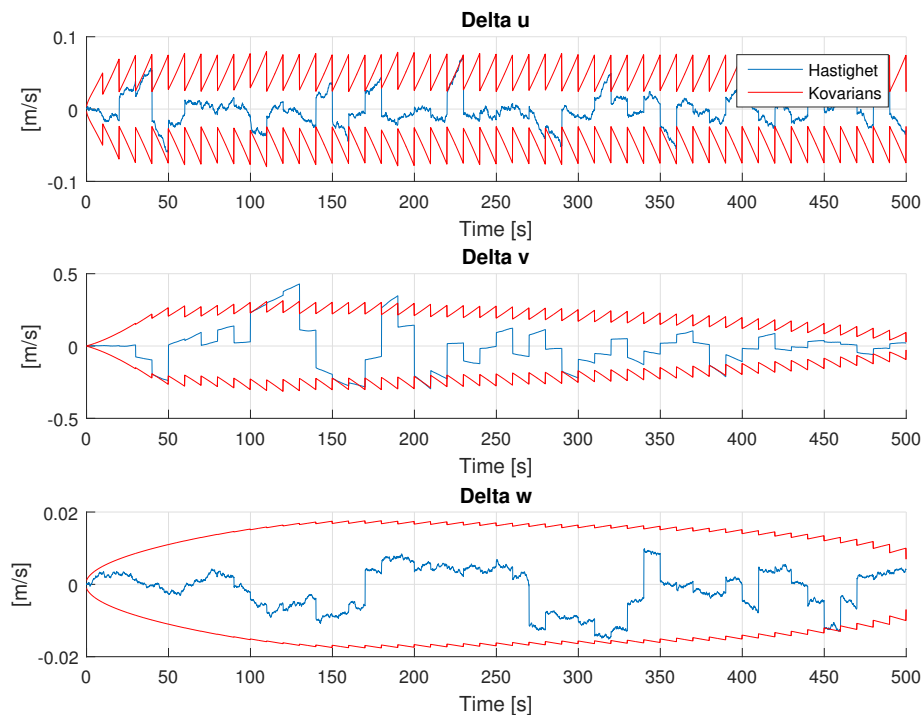


Figure 6: Feilen i estimatet for u, v og w

I figur 6 ser en også at en ikke får den høye kovariansen i starten, men heller at det er noe stigning ut over de første måleoppdateringene. Karakteristikken er også mye tydeligere gjennom simuleringen sammenlignet med figur 2. Som for posisjonsfeilen er også hastighetsfeilen mye mer innenfor kovariansen enn det en vanligvis kan forvente. Delta u er innenfor 99,30%, delta v er innenfor 88,62%, og delta w er innenfor 99,66% av tidsskrittene. Som for posisjonsfeilen er også den

gjennomsnittlige verdien for feilen i estimatet av hastigheten betraktelig mindre i denne simuleringen. For henholdsvis x, y og z retning er den gjennomsnittlige hastighetsfeilen $-0,093$ cm/s, $-0,49$ cm/s og $0,13$ cm/s.

5.2.3 Orienteringsfeil

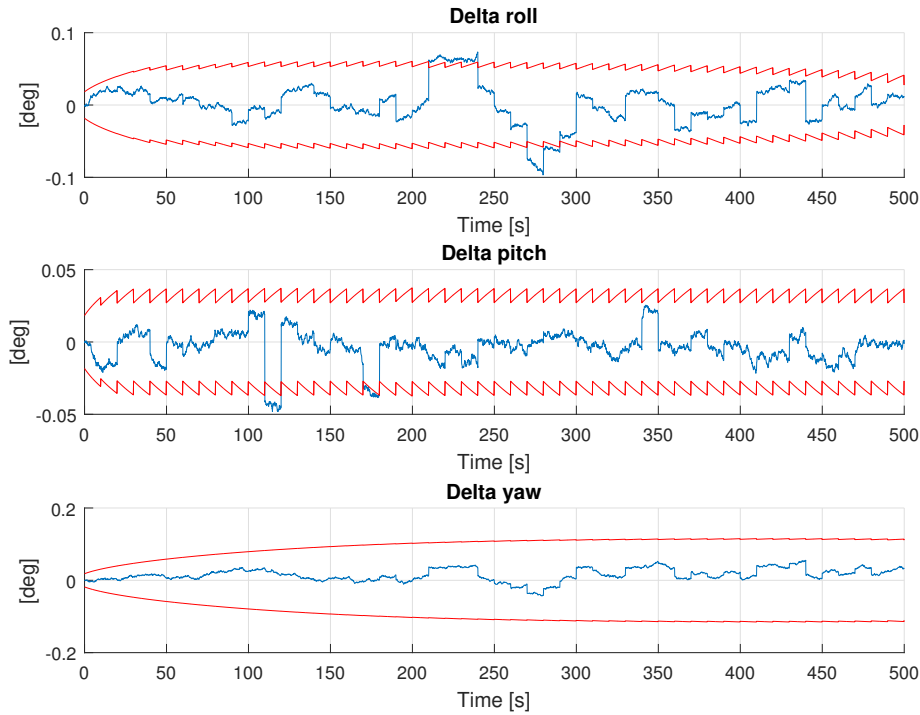


Figure 7: Feilen i estimatet for roll, pitch og yaw

Figur 7 viser den estimerte orienteringsfeilen og dens kovarians. Her er det helt tydelig ut fra bildet at det er overkonsistens i simuleringene der alle estimatene, med unntak av 0,76% av delta roll, er innenfor kovariansen. Den gjennomsnittlige orienteringsgeilen rund henholdsvis x-, y- og z-aksen er $0,0013^\circ$, $0,0003^\circ$ og $0,025^\circ$. Det er ingenting som tilsier at dette skal ha noen andre årsaker enn de som allerede er diskutert i forrige simulering.

Kovariansen til både delta roll og delta pitch stiger gjennom de første måleoppdateringene, for så at kovariansen til delta roll begynner å synke gradvis og at kovariansen til delta pitch stabiliseres. Kovariansen til delta yaw bruker nesten halve simuleringen på å stige, for så å bli mer eller mindre konstant ut resten av simuleringen.

5.2.4 De siste hundre sekundene

I figur 8 ser vi de siste hundre sekundene av simulering to for alle elementene på tilstandsvektoren. Selv om det i denne simuleringen har vært mulig å se tydeligere hvordan feilestimatene og deres kovarianser har utviklet seg når AUVen nærmer seg dokkingstasjonen, blir det her valgt å sette blikket på et plott som bare tar med de siste hundre sekundene også for denne simuleringen. Grunnen til dette er hovedsakelig for å kunne sammenligne figur 8 med figur 4.

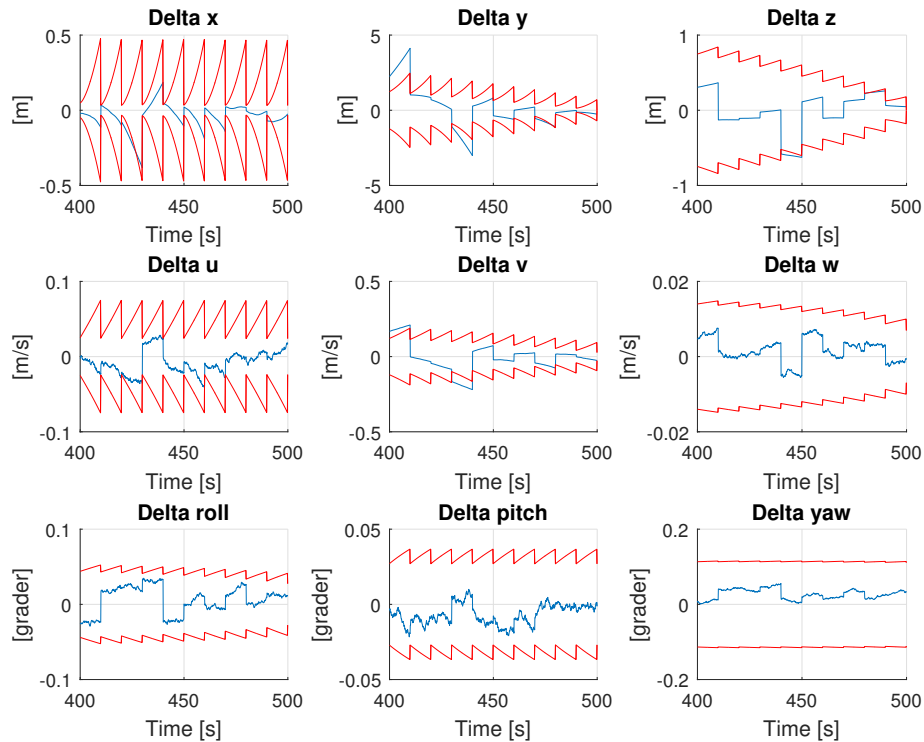


Figure 8: Feilen i alle elementene i tilstandsvektoren for de siste hundre sekundene av simuleringen

Sammenligning av plottene for de siste hundre sekundene i simulering 1 og simulering 2 viser at kovariansen til de forskjellige feilestimatene er identisk uavhengig av om en har en eksakt start posisjon med liten usikkerhet eller ikke. Den eneste forskjellen som er å se i de to figurene, er at delta yaw er mindre i figur 8 enn den er i figur 4.

5.2.5 Samendrag av simulering 2

Når en ser på begynnelsen av simuleringen får en andre karakteristikk for kovariansene til de forskjellige feilestimatene og en får mer presise estimater. Det er derimot området nærmere dokkingstasjonen som er det som er av størst interesse her. I dette området er det bare feilestimatet av orienteringen om AUVens z-akse som er forbedret i denne simuleringen. Altså har det ikke så mye å si for estimatet når en nærmer seg dokkingstasjonen om en hadde en god initiell startposisjon for Kalmanfilteret.

6 Diskusjon

Det er tydelig mulig å få gode estimat av AUVens posisjon, hastighet og orientering ved å bruke et kalmanfilter. Selv om det i simuleringene er brukt en enkel trajektor, er ligningene utledet på bakgrunn av at AUVen skal kunne ankomme på mer komplekse trajektorer. Det en må være klar over i valg av trajektor, er at AUVen ikke på innta en orientering der pitch risikerer å bli 90° .

Simuleringene viser at større nøyaktighet i den initiale tilstanden og dens kovarians, ikke har noe å si for estimatet når AUVen nærmer seg dokkingstasjonen. Her er det valgt et ganske stort område for dokking, 500 m. Hvis området der en tok i bruk navigasjonssystemet for dokking, i stedet for det som styrer AUVen ellers, hadde vært mindre kunne dette fått utslag. Om området for dokking hadde vært så lite at kalmanfilteret ikke hadde rukket å få ned den høye kovariansen, ville det gitt bedre resultat med større nøyaktighet i den initiale tilstanden og dens kovarians.

Dokkingstasjonen er her generalisert til et punkt som sender ut signalet som gir posisjonsmålingen. Dette gjør at det er mulig å bruke matematikken i denne oppgaven som grunnlag for et mer generelt navigasjonssystem. Hvis det er jevnlig punkter som sender ut signaler som gir posisjonsmålingen og AUVen kjenner igjen hver enkelt og vet hvor de er plassert. Dette vil kunne virke på omtrent samme måte som et GNSS-system. I dette tilfellet må det tas hensyn til jordkrumningen og at gravitasjonen ikke er like sterk over alt.

Til videre arbeid er det er det flere ting som kan inkorporeres i kalmanfilteret. Det er lite realistisk å ha målinger uten bias. Det kan derfor være lurt å legge inn biasene som tilstander i tilstandsvektoren og estimere disse også. Her er det også sett vekk i fra forstyrrelser fra omgivelsene. Det er derimot bare sett på estimeringen av tilstanden til AUVen. Alle sensorene måler uavhengig av hvordan AUVen helst skulle bevegde seg. Kompensering for forstyrrelsene fra miljøet vil være opp til valg av trajektor og systemet som styrer motorene til AUVen.

For å tilpasse dette systemet til et mer spesifikt system er det en del å gjøre. Karakteristikken for IMUen kan settes rett inn, men for måleoppdateringen kan det være en må utlede nye ligninger for geometrien til måleoppdateringen. Om en er veldig langt nord eller veldig langt sør, blir det nødvendig å ta hensyn til at nord-sør-linjene ikke kan tilnærmes som vinkelrette på øst-vest-linjene.

Ønsker en større nøyaktighet i estimatene, er det flere ting som kan gjøres. Det kan brukes en IMU som er utstyrt med kompass måling. Dette vil gi en måling av yaw, som er mye raskere observerbar. Når en nærmer seg dokkingstasjonen, kan en senke farten til AUVen og øke frekvensen på måleoppdateringen. Det kan uansett være lurt å senke farten på AUVen når den nærmer seg dokkingstasjonen, i hvert fall om den kommer i høy fart. På den måten kan en unngå at

AUVen treffer dokkingstasjonen med stor kraft om estimatene ikke er gode nok.

7 Konklusjon

Her er det vist at det er mulig å få gode estimat av posisjon, hastighet og orientering til en AUV på vei mot en dokkingstasjon ved hjelp av et kalmanfilter. Det er lagt et grunnarbeid som kan bygges videre på. Det er likevel mer som kan gjøres på den generelle løsningen av estimeringsproblemet og det vil være mer å gjøre for å tilpasse det til et mer spesifikt problem

I simuleringene er det tydelig at det er samsvar mellom feilen i estimatene og kovariansen. Simuleringene viser imidlertid at feilestimatene er noe vel mye innenfor kovariansen. Dette kan skyldes alle forenklingene som er gjort og den enkle trajektoren.

A Akronymer

AUV Autonomous Underwater Vehicle (autonomt undervansfartøy)

ROV Remotely operated vehicle (fjernstyrt fartøy)

GNSS Global Navigation Satellite System

USBL Ultra short baseline

DVL Dopler velocity log

APS Akustisk posisjonssystem

IMU Inertial measurement unit (treghetssensorenhet)

KF Kalmanfilter

ESKF Error-state Kalman filter (Feiltilstands-kalmanfilter)

NED North-East-Down

References

- [1] Thor I. Fossen. *Handbook of Marine Craft Hydrodynamics and Motion Control*. Wiley, 2011.
- [2] Kenneth Gade. “Integrering av treghetsnavigasjon i en autonom undervannsfarkost”. In: Teknisk rapport nummer 97/03179, FFI (1997).
- [3] M. Miranda et al. “Homing an unmanned underwater vehicle equipped with a DUSBL to an unmanned surface platform: A feasibility study”. In: *Oceans - San Diego, 2013*. Sept. 2013, pp. 1–10.
- [4] Narcís Palomeras. “Autonomus I-AUV Docking for Fixed-base Manipulation”. In: *Preprints of the 19th World Congress, The International Federation of Automatic Control* (2014), p. 6.
- [5] Robert Brown og Patrick Hwang. *Introduction to Random Signals and Applied Kalman Filtering*. Wiley, 2012.
- [6] Albert Sans-Muntadas et al. “Navigation and probability assessment for successful AUV docking using USBL”. In: *10th IFAC Conference on Manoeuvring and Control of Marine Craft (MCMC)*. Vol. 48. 16. Copenhagen, Denmark, 2015, pp. 204–209.
- [7] Olav Egeland Tommy Gravdal. *Modeling and Simulation for Automatic Control*. Wiley, 2004.