

Sondre Myrberg

Assimilering av sonardata med matematiske modeller av fiskeatferd i oppdrettsmerder

Masteroppgave i Kybernetikk og robotikk
Veileder: Martin Føre og Jo Arve Alfredsen
Desember 2019

Sondre Myrberg

Assimilering av sonardata med matematiske modeller av fiskeatferd i oppdrettsmerder

Masteroppgave i Kybernetikk og robotikk
Veileder: Martin Føre og Jo Arve Alfredsen
Desember 2019

Norges teknisk-naturvitenskapelige universitet
Fakultet for informasjonsteknologi og elektroteknikk
Institutt for teknisk kybernetikk



MASTEROPPGAVE

Kandidatens navn: Sondre Myrberg

Fag: Teknisk Kybernetikk

Oppgavens tittel (norsk): Assimilering av sonardata med matematiske modeller av fiskeatferd i oppdrettsmerder

Oppgavens tittel (engelsk): Assimilation of sonar data with mathematical models of fish behaviour in sea-cages

Oppgavens tekst:

Matematisk modellering er en velkjent kybernetisk metode for å strukturere, syntetisere og operasjonalisere eksisterende kunnskap om dynamikken og egenskapene til et system, som har blitt anvendt på fisk med fokus på fiskens energetikk og vekst, atferd eller en kombinasjon av disse. Dette har resultert i at det i dag finnes en individbasert matematiske modell som simulerer fiskens respons mot forholdene den utsettes for i en oppdrettssituasjon. Selv om modellens resultater tidligere har blitt sammenliknet med data fra eksperimenter i små/mellomstore merder, må en validere modellen mot data fra fullskala oppdrettssituasjoner før en med sikkerhet kan si at den gjengir et realistisk bilde av oppdrettssituasjonen.

Atferdsobservasjoner i fullskala fiskeoppdrett gjøres ofte med sonar/ekkolodd. Slike metoder resulterer i datasett som beskriver atferden til den gruppen med fisk som befinner seg innenfor «sonarbeamene» som bare dekker et begrenset sub-volum i merden. Ettersom den matematiske modellen beskriver atferd på individnivå vil en direkte sammenlikning mellom data fra atferdsmodeller og sonardata kreve at en utvikler nye metoder innen dataassimilasjon som tar høyde for forskjellen i datatype. Siden en ikke kan vente at en matematisk modell skal klare å gjengi måledata fra hvert enkelt «sonarskudd» må disse metodene også inneholde et element av aggregering av data fra både sonar og modell, eksempelvis ved hjelp av statistikk.

I tillegg til å muliggjøre validering og videreutvikling av modellens funksjonalitet basert på sonardata, vil vellykket utvikling av slike metoder representere et grunnlag for fremtidige systemer for estimering av fiskens tilstand i merdene basert på assimilering mellom sonarer og simuleringsmodeller i sanntid. For å begrense oppgavens omfang vil fokus holdes mot ekkoloddtypen som anvendes i Ocean Farm 1, SIMRAD EK 80.

Arbeidet skal inkludere:

- Litteraturstudium for å undersøke hva som har blitt gjort for tilsvarende problemstillinger (assimilasjon av punktmålinger med distribuerte modeller) innen andre anvendelsesområder
- Metodeutvikling:
 - Studere egenskapene til EK 80 og dens outputdata for å forstå hva måledata fra denne representerer og hvordan disse forholder seg til modellen
 - Utvikle metoder for å harmonisere EK 80-data og modelldata via for eksempel aggregering i tid og rom

- Anvend metodene på simuleringsdata fra modellen og sammenlikn kvalitativt mot faktiske sonardata fra feltstudier. Vurder:
 - Om modellen gir en tilstrekkelig realistisk romlig distribusjon av fisk til å kunne brukes i en observerstruktur
 - Om det er mulig å gjøre modelloutputs kvalitativt mer like reelle data gjennom enkle parameterjusteringer
- Virtuelle studier der metodene anvendes på rene simuleringsresultater. Basert på dette, vurder:
 - Hvordan antall datapunkter (dvs. sonarenheter) påvirker evnen til å fange dynamikken i simuleringen
 - Hvordan plassering av målinger (dvs. sonarenheter) påvirker evnen til å fange dynamikken i simuleringen

Oppgaven gitt: 4. august 2019

Besvarelsen leveres innen: 22. desember 2019

Utført ved Institutt for teknisk kybernetikk

Veileder: Martin Føre

Trondheim, 5. august 2019

Faglærer

Oppsummering

En viktig del av det å drive med oppdrett er å ha kontroll på tilstanden i merden. Ekkolodd er et hjelpemiddel for å overvåke og følge med på hva som skjer under overflaten, men hva slags informasjon gir egentlig et ekkolodd, og hvordan kan den totale tilstanden observeres ved hjelp av få målepunkter?

Gjennom matematisk modellering av fiskens atferd og matematisk modellering av ekkolodd skal det undersøkes om ekkogrammer fra reelle anlegg kan assimileres inn i den matematiske modellen for å kunne observere den faktiske tilstanden i en merd, og hvordan antall målepunkter og plassering av disse vil påvirke dette.

Basert på et litteraturstudium av hvilke metoder som finnes for assimilering av punktmålinger med distribuerte modeller vises det at et Ensemble Kalmanfilter kan være en god måte å få til dette. Den matematiske modellen representerer imidlertid ikke atferden til simulert fisk på en god nok måte, og må utvikles videre. Modellen er heller ikke testet i så stor skala som Ocean Farm 1, og ekkolodddata herfra er da ikke direkte sammenliknbart med simulerte ekkogrammer. Det vises derimot at for mindre skala at et begrenset antall ekkolodd kan fange opp dynamikken i merden så lenge det observeres over lang nok tid.

Summary

An important part in the field of aquaculture is controlling the state of the cages. Echo sounders is an aid in surveillance and observing what is going on under the surface, but what kind of information exactly does an echosounder provide, and how could the total state of the cage be observed through few measurements?

Through mathematical modelling of the behaviour of the fish and mathematical modelling of the echo sounder, it will be investigated if echograms from real aquaculture facilities could be assimilated into the mathematical model to observe the state of a cage, and changing the number of measurements and placement of these will affect this.

Based on a literature study of what methods assimilation of point measurements of distributed models are available, a clear option to to this is the Ensemble Kalman Filter. However the mathematical model of the fish does not represent the behaviour of the fish in a good enough manner, and needs further development. The models has nor been tested on the scale that is Ocean Farm 1, og data from echo sounders here will not be directly comparable to simulated echograms. On the other hand it is shown that for smaller scale simulations a limited number of echo sounders could observe the dynamics of the cage as long as the observation time is long enough.

Forord

Denne masteroppgaven er utarbeidet som det siste steget på veien mot mastergrad innenfor fagfeltet Kybernetikk og Robotikk ved Norges teknisk-naturvitenskapelige universitet. Arbeidet ble utført høsten 2019 under veiledning fra hovedveileder Førsteamanuensis Martin Føre og medveileder Førsteamanuensis Jo Arve Alfredsen, begge fra Institutt for Teknisk Kybernetikk ved NTNU. Arbeidet bygger til en viss grad videre på en fordypningsrapport skrevet våren 2019. Denne rapporten er ikke publisert, men nødvendig materiale er gjengitt i denne rapporten. Dette gjelder deler av kapittel 2.3 i tillegg noen MATLAB-funksjoner som utnyttes som en del av større operasjoner. Modellen og rammeverket for simulering av fisk er utarbeidet ved SINTEF og NTNU. Fra Kongsberg Maritime og Ocean Farming har jeg fått både tilgang til data, programvare for å behandle data, informasjon, og nøkkelfigurer fra Ocean Farm 1. Fra NTNU har jeg fått arbeidsplass og PC, i tillegg et faglig grunnlag for å kunne lykkes.

Først og fremst har jeg lyst til å takke mine veiledere Martin Føre og Jo Arve Alfredsen for kyndig veiledning og gode diskusjoner, og for å være tilgjengelig for spørsmål stort sett til alle døgnetstider. For all mulig hjelp med FhSim har Biao Su vært helt enestående, og åpenheten ved informasjon om Ocean Farm 1 fra både Kongsberg Maritime ved Thor Hukkelås og Frank Reier Knudsen og Ocean Farming ved Gunnar Myrebø og Arvid Hammernes har vært veldig hjelpsom. I tillegg har stipendiat Kristbjörg Edda Jónsdóttir vært til stor hjelp i bearbeidelsen av data fra Ocean Farm 1. Tusen takk. Til slutt vil jeg takke alle som har vært en del av studietiden min. Uten dere ville det vært kjedelig å være student.

Sondre Myrberg
Trondheim, Desember 2019

Innhold

Oppgavetekst	i
Oppsummering	iii
Summary	iv
Forord	v
Innhold	vii
Tabeller	ix
Figurer	xi
1 Introduksjon	1
2 Teori	3
2.1 Dataassimilering	3
2.1.1 Tilstandsestimering	4
2.1.2 Sekvensiell dataassimilering	7
2.1.3 Assimilering innefor andre disipliner	11
2.2 Simrad EK80	12
2.3 FhSim	16
2.3.1 Grunnleggende om FhSim	16
2.4 Fiskeatferd for et reelt scenario	20
3 Metode	25
3.1 Matematiske målemodeller	25
3.2 Simuleringer	26
3.3 Ekkogrammer	27
3.3.1 Simulerte ekkogrammer	29
3.3.2 Tverrsnitt av merd	30

3.4	Heatmap	32
3.5	Videogenerering	32
4	Resultater	35
4.1	Reelle ekkogrammer fra Ocean Farm 1	35
4.2	Simulerte ekkogrammer	35
4.2.1	Dybdefordeling	35
4.2.2	Ekkogrammer uten miljøpåvirkning	37
4.3	Horisontalfordeling med 100 000 fisk	47
4.3.1	Tverrsnitt	48
4.3.2	Video	48
5	Diskusjon og konklusjon	51
5.1	Diskusjon	51
5.1.1	Ekkogrammer og fiskens atferd	51
5.1.2	Ekkolodd som observer	53
5.2	Oppsummering og konklusjon	54
5.3	Videre arbeid	55
	Bibliografi	57
	Vedlegg	60
A	Kodesnutter	61
A.1	MATLAB	61
A.1.1	inCone	61
A.1.2	SonarPing	61
A.1.3	Video	62
A.1.4	Interpolert tverrsnitt	63

Tabeller

2.1	Forklaring av indeksene som brukes i likning (2.14), forenklet og gjengitt fra [1].	16
2.2	Optimale verdier for fôrspredermodellen. Gjengitt fra [2]	20

Figurer

2.1	Illustrasjon av algoritmen som utgjør et Kalmanfilter, og likningene assosiert med hver enkelt boks. Hentet fra [3].	8
2.2	Strålemønster for transduceren Simrad ES70-7C. Figur hentet fra [4]. . .	13
2.3	Bilde av Ocean Farm 1.	14
2.4	Tegninger av Ocean Farm 1 med plasseringer av ekkolodd. Tilsendt fra Kongsberg og gjengitt med tillatelse.	15
2.5	Sammenlikning av vertikalfordelingen av observert fisk(venstre) og simulert fisk(høyre) der simuleringen har tilsvarende miljøparametre som ble målt i scenariene som ble observert. Figuren er gjengitt fra [1].	17
2.6	Dimensjoner for de to forskjellige fôrsprederne brukt i [5]. Sprederen til venstre er av typen RS-90C og sprederen til høyre av typen RS-90C MK III, begge fra AKVAGroup. Figur er hentet fra [5].	18
2.7	Fordeling av pellets i avstand fra fôrfordeler med tre forskjellige hastigheter på luftstrømmen. 7 til 12 mm betyr hvilken størrelse fôrpellet som ble brukt i de forskjellige forsøkene, SA og SB betyr henholdsvis <i>Spreader A</i> og <i>Spreader B</i> , og forteller hvilken spredertype som er brukt, TU eller TD forteller om sprederen var tiltet opp (<i>U</i>) eller ned (<i>D</i>) og C1 og C2 forteller hvilken merd (<i>cage</i>) som forsøket ble utført i. NS betyr at den statistiske analysen for forsøket ikke var signifikant med $p < 0.05$. Figuren er hentet fra [5].	19
2.8	Sammenlikning med måledata og modellerte data for pelletfordeling ved forskjellige scenarier. Måledata er de heltrukne linjene, mens modellert data er punktene. Tittelen på underfigurene forteller om konfigurasjonen ved den enkelte simuleringen, altså om det var tilt opp eller ned, hvilken fôrtype som ble brukt og lufthastigheten. Figuren er hentet fra [2].	21
2.9	Simulert <i>xy</i> -fordeling av fôr med type 1-spreder. Sprederen er plassert i origo med retning forover, eller <i>dense direction</i> , i positiv <i>x</i> -retning. Figur hentet fra [2].	22
2.10	Akkumulert telemetrisk data fra fire merkede lakseindivider over omtrent tre måneder i et kommersielt anlegg. Figurer hentet fra [6].	23

2.11 Dybdemønster fra teletrisk merket fisk. Hentet fra [6].	24
3.1 Reelt ekkogram fra OF1. Fargekodene antyder reflektert signalstyrke. Dette ekkogrammet viser et gitt punkt i merden fra tidlig morgen til sen ettermiddag/tidlig kveld. Serial No: 377 er ”navnet” på dette ekkoloddet. . . .	28
3.2 Igjen kart over plasseringene til ekkoloddene i Ocean Farm 1, men merket med hvilket serienummer som hører til hvilket ekkolodd.	28
3.3 Samme simulerte scenario, men to forskjellige observasjonsmodeller. Til venstre brukes likning (3.2) til å gi en normalisert observasjon, mens til høyre brukes kun antall observert fisk innenfor kjeglen.	29
3.4 Plassering av simulerte ekkolodd.	30
3.5 Illustrasjonsbilde på hvordan oppsettet en snor med ekkolodd er satt ut på tvers av en merd. Figur er gjengitt fra [7].	31
3.6 Interpolert ekkogram over et tverrsnitt av merden. Prikkete linjer angir hvor ekkoloddene har stått. Gjengitt fra [7].	32
3.7 Plassering på simulerte ekkolodd for observasjon av tverrsnitt.	33
4.1 Ekkogrammer for forskjellige steder i Ocean Farm 1. Disse viser omtrent tolv timer en dag i februar 2018.	36
4.2 Dybdefordeling av all fisken i merden over en dag. Det er to fôringsperioder, første fra 0900 til 1200, siste fra 1500 til 1800.	37
4.3 Dybdefordeling av fisk når miljøet er helt nøytralt. Fisken fordeler seg jevnere utover vannmassen. Simuleringen ble stoppet noe tidligere enn i figuren over, derfor viser den ikke fordelingen over et helt døgn.	38
4.4 Dybdefordeling av fisk når miljøet er helt nøytralt og foretrukket avstand til naboene er justert opp til 2 BL.	38
4.5 Simulert ekkogram med tidssteg på 200 s over et døgn. Under fôrperiodene er det ekstremt tett mellom fisken, men det kan se ut som fisken kun oppholder seg her når det føres.	39
4.6 Samme scenario som i figur 4.5, men her er fargekodingen begrenset til 14 individer per rute. Altså vil alt over 14 gi samme farge. Dette viser at det er fisk her utenom fôringsperioder også.	40
4.7 Simulert ekkogram med tidssteg på 200 s over et døgn. Her kan det også sees stor ansamling av fisk under fôring, med tidspunkter med ekstrem tetthet.	40
4.8 Simulert ekkogram med tidssteg på 200 s over et døgn. Fra dette området forsvinner fisken totalt under fôrperiodene, men holder seg jevnt utenom fôring.	41
4.9 Simulert ekkogram med tidssteg på 200 s over et døgn. Her forsvinner fisken nærmest helt gjennom fôrperiodene, men holder seg stabilt utenom.	41
4.10 Simulert ekkogram med tidssteg på 200 s over et døgn. Det samme som i tidligere figurer kan observeres her, fiskemengden holder seg stabilt utenom fôrperiodene der mesteparten av fisken forsvinner fra området.	42
4.11 Simulert ekkogram med tidssteg på 200 s over et døgn. Her er ekkoloddet plassert noe nærmere sentrum av merden og det kan sees at utenom fôrperiodene er det null aktivitet i dette området.	42

4.12	Simulert ekkogram med tidssteg på 200 s over et døgn. Under fôrperiodene er det tettere mellom fisken, og dette området er sammenliknet med andre steder i merden veldig populært under fôring.	43
4.13	Simulert ekkogram med tidssteg på 200 s over et døgn. Her kan det også sees stor ansamling av fisk under fôring.	44
4.14	Simulert ekkogram med tidssteg på 200 s over et døgn. I dette området forsvinner mesteparten av fisken mens det fôres, men det kan sees at noe fisk dukker opp mot slutten av begge fôrperiodene.	44
4.15	Simulert ekkogram med tidssteg på 200 s over et døgn. Her er det noe fisk gjennom fôring, spesielt i andre fôrperiode.	45
4.16	Simulert ekkogram med tidssteg på 200 s over et døgn. Her forsvinner også mye fisk mens det fôres.	45
4.17	Simulert ekkogram med tidssteg på 200 s over et døgn. Her er ekkoloddet plassert noe nærmere sentrum av merden, men i motsetning til figur 4.11 som viser nøyaktig samme plassering er det aktivitet her også utenom fôrperiodene.	46
4.18	Heatmap av fiskens fordeling i xy -planet med to forskjellige konfigurasjoner. Begge figurer viser de første ni timene med tidssteg på 2000 s.	47
4.19	Heatmap av fiskens fordeling i xy -planet med to forskjellige konfigurasjoner. Begge figurer viser 23 timer med tidssteg på 3600 s og to fôrperioder.	48
4.20	Tverrsnitt av merden åtte timer etter midnatt.	49
4.21	Tverrsnitt av merden etter elleve timer, altså to timer inn i første fôrperiode.	50

Kapittel 1

Introduksjon

I et oppdrettsanlegg er det, som i alle andre næringer, ønskelig å kunne overvåke hva som skjer på en god måte, men når det som skal overvåkes befinner seg under vann byr dette på utfordringer. I dag utnyttes ekkolodd som en måte å observere hva som skjer, men hva sier egentlig resultatene fra et ekkolodd? Basert på leverandørens egne tall vil et ekkoloddene som er montert i Ocean Farm 1 dekke omlag 0.5% av det totale volumet fisken kan bevege seg på[8, 4].

Matematisk modellering er en kybernetisk metode for å strukturere, syntetisere og operasjonalisere eksisterende kunnskap om dynamikken til et eksisterende system. Dette har blant annet blitt utført for eksisterende kunnskap om laksens atferd og biologi[2], og endt med en modell som kan simuleres i et bestemt rammeverk kalt FhSim[9]. Denne modellen er ikke validert mot fullskala oppdrettsanlegg, men det har vist seg at modellen i mindre skala har gjengitt dybdefordelingen for oppdrettslaks på en god måte. Modellen har derimot ikke blitt validert for fiskens respons i xy -planet, og dermed er det ukjent om den romlige fordelingen på fisken er god nok til å gjengi virkeligheten på en god måte.

Gjennom denne oppgaven skal det først utforskes hvilke metoder for dataassimilering av punktmålinger med distribuerte modeller som eksisterer og blir brukt innefor andre fagfelt i tilsvarende problemstillinger. Deretter skal ekkoloddene som brukes i Ocean Farm 1 studeres, for så å modellere disse matematisk til bruk på den allerede eksisterende modellen for atferd. Med dette som bakgrunn skal forskjellige scenarier simuleres og observasjonene fra simulerte ekkolodd skal sammenliknes med reelle ekkolodd. Dette både for å validere modellen og vurdere hvor bra et ekkolodd som observerer gjengir dynamikken i en merd, og dermed hva slags informasjon et ekkolodd gir. Til slutt vil simuleringer, data, resultater og litteraturstudium drøftes, og det vil vurderes hvorvidt den matematiske atferdsmodellen er god nok, om ekkolodd som observererstruktur er gjennomførbart og eventuelt hvordan.

Kapittel 2

Teori

Dette kapittelet vil ta for seg den nødvendige teoretiske bakgrunnen for oppgaven. Først vil kapittelet omhandle hva dataassimilering er, før noen metoder for datassimilering utledes. For å få til dette må en bakgrunn om tilstandsestimering også med. Før assimileringebiten avsluttes vil det også vises til hvilke metoder innenfor assimilering som er brukt i tilsvarende problemstillinger innenfor andre fagfelt. Deretter vil det komme informasjon om ekkolodd, spesifikt Simrad EK80 som er ekkoloddet som er brukt i Ocean Farm 1 og som er utgangspunktet her. Videre kommer det en utledning om fiskesimuleringsmodellen som blir brukt, før kapittelet avsluttes med et kjapt sammendrag av et case hvor fiskeatferd har blitt observert med hydroakustiske telemetrisendere.

2.1 Dataassimilering

Datassimilering er en disiplin innenfor matematikk som søker å optimalt inkorporere observasjonsdata i en matematisk modell, og anvendelsen er veldig utbredt i matematiske problemstillinger der antall variabler er svært høyt [10, 11]. Meteorologi, hydrologi og oseanografi er tre eksempler på disipliner som bruker dataassimilering som verktøy for å få presise estimater over større områder ved hjelp av distribuerte målepunkter. [12] definerer dataassimilering som

- (1) *optimally incorporating*
- (2) *partial and noisy observational data of a*
- (3) *chaotic and thus necessarily nonlinear (and most often, complex, multiscale) dynamical system, with an*
- (4) *imperfect model of the system dynamics and of the observations, in order to obtain*
- (5) *predictions of and the assoxoated uncertainty for the state of the system, repetively in time.*

Dette vil da si at formålet med dataassimilering er å gi gode estimater for tilstanden til et system og usikkerheten rundt disse estimatene ved å optimalt inkorporere støyete og ikke-

perfekte målinger fra et kaotisk og komplekst dynamisk system som ikke er modellert perfekt. For å få til dette på en god måte er det gjennom tiden utviklet flere metoder. Noen av disse metodene vil bli presentert og utdypet i de påfølgende underkapitlene.

2.1.1 Tilstandsestimering

Tilstandsromanalyse

I [13] gis estimatet for en sann tilstand i et tidsuavhengig system ved

$$\begin{aligned}\psi^f &= \psi^t + p^f \\ d &= \psi^t + \epsilon\end{aligned}\tag{2.1}$$

hvor ψ^f representerer et estimat av den sanne tilstanden ψ^t med ukjent estimatorusikkerhet p^f . d representerer en måling av ψ^t med ukjent målingsavvik eller -støy ϵ . Herfra er målet å finne et forbedret analysert tilstandsestimat ψ^a for ψ^t . For å få til dette må vi gjøre antakelsene

$$\begin{aligned}\mathbf{E}[p^f] &= 0 \\ \mathbf{E}[\epsilon] &= 0 \\ \mathbf{Cov}[\epsilon, p^f] &= \mathbf{E}[(\epsilon - \mathbf{E}[\epsilon])(p^f - \mathbf{E}[p^f])] = 0 \\ \mathbf{Var}[p^f] &= \mathbf{E}[(p^f)^2] - \mathbf{E}[p^f]^2 = C_{\psi^f\psi^f}^f \quad (= Q) \\ \mathbf{Var}[\epsilon] &= \mathbf{E}[\epsilon^2] - \mathbf{E}[\epsilon]^2 = C_{\epsilon\epsilon} \quad (= R)\end{aligned}\tag{2.2}$$

som betyr at usikkerhet og støy følger en normalfordelingskurve. Q og R er alternative notasjoner som brukes for kovariansmatriser når det for eksempel kommer Kalmanfiltrering av tidsuavhengige systemer i én eller (som oftest) flerdimensjonale systemer. En lineær estimator

$$\psi^a = \psi^t + p^a = \alpha_1 \psi^f + \alpha_2 d\tag{2.3}$$

hvor p^a følger samme normalfordelingskurve med

$$\begin{aligned}\mathbf{E}[p^a] &= 0 \\ \mathbf{E}[(p^a)^2] &= C_{\psi^a\psi^a}^a\end{aligned}$$

gir et estimat av den sanne tilstanden med $\mathbf{E}[\psi^a] = \psi^t$. Innsetting av (2.1) i (2.3) gir

$$\begin{aligned}\psi^t + p^a &= \alpha_1(\psi^t + p^f) + \alpha_2(\psi^t + \epsilon) \\ \implies \mathbf{E}[\psi^t + p^a] &= \mathbf{E}[\alpha_1(\psi^t + p^f)] + \mathbf{E}[\alpha_2(\psi^t + \epsilon)] \\ \implies \psi^t &= (\alpha_1 + \alpha_2)\psi^t \\ \implies \alpha_1 + \alpha_2 &= 1\end{aligned}$$

som dermed gir en tilstandsestimator

$$\begin{aligned}\psi^a &= (1 - \alpha_2)\psi^f + \alpha_2 d \\ &= \psi^f + \alpha_2(d - \psi^f)\end{aligned}$$

med estimatoravvik

$$p^a = p^f + \alpha_2(\epsilon - p^f)$$

Variansen for estimatoravviket er da gitt ved

$$\begin{aligned} \mathbf{Var}[p^a] &= C_{\psi\psi}^a = \mathbf{E}[(p^a)^2] \\ &= \mathbf{E}[(p^f + \alpha_2(\epsilon - p^f))^2] \\ &= \mathbf{E}[(p^f)^2] + 2\alpha_2\mathbf{E}[p^f(\epsilon - p^f)] + \alpha_2^2\mathbf{E}[\epsilon^2 - 2\epsilon p^f + (p^f)^2] \\ &= C_{\psi\psi}^f - 2\alpha_2 C_{\psi\psi}^f + \alpha_2^2(C_{\epsilon\epsilon} + C_{\psi\psi}^f) \end{aligned}$$

som minimeres ved

$$\begin{aligned} \frac{dC_{\psi\psi}^a}{d\alpha_2} &= 0 \\ \implies -2C_{\psi\psi}^f + 2\alpha_2(C_{\epsilon\epsilon} + C_{\psi\psi}^f) &= 0 \\ \implies \alpha_2 &= \frac{C_{\psi\psi}^f}{C_{\epsilon\epsilon} + C_{\psi\psi}^f} \end{aligned}$$

Dette gir et analysert tilstandsestimat med minimal varians gitt ved

$$\psi^a = \psi^f + \frac{2C_{\psi\psi}^f}{C_{\epsilon\epsilon} + C_{\psi\psi}^f}(d - \psi^f) \quad (2.4)$$

med varians for estimatoravvik

$$\begin{aligned} C_{\psi\psi}^a &= C_{\psi\psi}^f - 2\frac{C_{\psi\psi}^f}{C_{\epsilon\epsilon} + C_{\psi\psi}^f}C_{\psi\psi}^f + \left(\frac{C_{\psi\psi}^f}{C_{\epsilon\epsilon} + C_{\psi\psi}^f}\right)^2 (C_{\epsilon\epsilon} + C_{\psi\psi}^f) \\ &= C_{\psi\psi}^f \left(1 - \frac{C_{\psi\psi}^f}{C_{\epsilon\epsilon} + C_{\psi\psi}^f}\right) \end{aligned}$$

som i [13] kalles *minimum variance estimate*. Dette er altså en tilstandsestimator med lav varians i usikkerheten i estimatet.

Sannsynlighetsanalyse

Bayes' teorem er et sentralt begrep innen statistikk og sannsynlighetsberegning og er fra [13, 14] gitt ved

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Gitt en sannsynlighetstetthetsfunksjon $f(\psi)$ for første estimat ψ^f og en betinget sannsynlighetsfunksjon $f(d|\psi)$ for måling d gir Bayes' teorem at

$$f(\psi|d) \propto f(\psi)f(d|\psi)$$

som sier forteller at a posteriori sannsynlighetsfunksjon for ψ er proporsjonalt med produktet for a priori sannsynlighetsfunksjon for ψ og den betingete sannsynlighetsfunksjonen for d . Med de samme estimatene for ψ^t gitt i (2.1) og antakelser om at støy følger normalfordelingskurven kan $f(\psi)$ og $f(d|\psi)$ defineres som

$$\begin{aligned}
 f(\psi) &\propto \exp\left(-\frac{1}{2}(\psi - \psi^f)(C_{\psi\psi}^f)^{-1}(\psi - \psi^f)\right) \\
 f(d|\psi) &\propto \exp\left(-\frac{1}{2}(\psi - d)C_{\epsilon\epsilon}^{-1}(\psi - d)\right) \\
 \implies f(\psi|d) &\propto \exp\left(-\frac{1}{2}\mathcal{J}[\psi]\right)
 \end{aligned}$$

der minimering av $\mathcal{J}[\psi]$ nødvendigvis gir maksimal betinget sannsynlighet for $f(\psi|d)$.
 Minimering av \mathcal{J} gir

$$\begin{aligned}
 \frac{d\mathcal{J}[\psi]}{d\psi} &= \frac{d}{d\psi}(\psi - \psi^f)(C_{\psi\psi}^f)^{-1}(\psi - \psi^f) + (\psi - d)C_{\epsilon\epsilon}^{-1}(\psi - d) = 0 \\
 &\implies 2(\psi - \psi^f)(C_{\psi\psi}^f)^{-1} + 2(\psi - d)C_{\epsilon\epsilon}^{-1} = 0 \\
 &\implies \psi = \psi^f + \frac{C_{\psi\psi}^f}{C_{\psi\psi}^f + C_{\epsilon\epsilon}}(d - \psi^f)
 \end{aligned}$$

som gir samme resultat som i (2.4). Dermed er estimatet med minimal varians også estimatet for maksimal betinget sannsynlighet.

Utvidelse til flere dimensjoner

Metodene fra de foregående kapitlene er utledet for endimensjonale problemer, men kan utvides til samme form som i (2.1) ved

$$\begin{aligned}
 \psi^f &= \psi^t + \mathbf{p}^f \\
 \mathbf{d} &= \mathbf{M}\psi^t + \boldsymbol{\epsilon}
 \end{aligned} \tag{2.5}$$

der \mathbf{M} er en målingsmatrise. På samme måte som i en dimensjon blir også her

$$\begin{aligned}
 \mathbf{E}[\mathbf{p}^f] &= 0 \\
 \mathbf{E}[\boldsymbol{\epsilon}] &= 0 \\
 \text{Cov}[\boldsymbol{\epsilon}, \mathbf{p}^f] &= \mathbf{E}[(\boldsymbol{\epsilon} - \mathbf{E}[\boldsymbol{\epsilon}])(\mathbf{p}^f - \mathbf{E}[(\mathbf{p}^f)]^T)] = 0 \\
 \text{Var}[\mathbf{p}^f] &= \mathbf{E}[\mathbf{p}^f(\mathbf{p}^f)^T] - \mathbf{E}[\mathbf{p}^f]^2 = \mathbf{C}_{\psi\psi}^f \\
 \text{Var}[\boldsymbol{\epsilon}] &= \mathbf{E}[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T] - \mathbf{E}[\boldsymbol{\epsilon}]^2 = \mathbf{C}_{\epsilon\epsilon}
 \end{aligned}$$

som fra [13, 3] vil gi likningene

$$\begin{aligned}
 \psi^a &= \psi^f + \mathbf{K}(\mathbf{d} - \mathbf{M}\psi^f) \\
 \mathbf{C}_{\psi\psi}^a &= (\mathbf{I} - \mathbf{K}\mathbf{M})\mathbf{C}_{\psi\psi}^f \\
 \mathbf{K} &= \mathbf{C}_{\psi\psi}^f \mathbf{M}^T (\mathbf{M}\mathbf{C}_{\psi\psi}^f \mathbf{M}^T + \mathbf{C}_{\epsilon\epsilon})^{-1}
 \end{aligned} \tag{2.6}$$

som er kjent som likningene for et *Kalmanfilter* med *Kalman gain* \mathbf{K} . Kalmanfilteret og varianter av dette skal presenteres nærmere i de kommende kapitlene.

2.1.2 Sekvensiell dataassimilering

Mens det forrige kapittelet fokuserte på å finne det besteestimat gitt et a priori estimat i et tidsuavhengig system, vil dette kapittelet fokusere på sekvensiell oppdatering av en tidsavhengig tilstandsmodell ved hjelp av dataassimilering. For å få til dette vil de samme metodene som ble brukt for tidsuavhengige systemer brukes og utvides for å gjelde tidsavhengige systemer. En generell lineær dynamisk modell kan da beskrives som

$$\begin{aligned}\psi_{k+1}^f &= G\psi_k^f + q_k \\ d_k &= M\psi_k^f + \epsilon_k\end{aligned}$$

der q_k er det ukjente modellavviket fra tidssteg k til $k + 1$, G er en lineær modelloperator (matrise) og M er en operator som bestemmer hvilke tilstander som måles. For enkelhetens skyld vil også tilstandsvektorer og matriser ikke utheves i fet skrift, altså vil ψ her representere en tilstandsvektor og nødvendigvis ikke en skalar.

Kalmanfilter, KF

Et Kalmanfilter er en algoritme som ved å bruke en serie av målinger over tid estimerer en tilstand på en bedre måte enn ved å estimere tilstanden ved kun en måling, og ble presentert av Rudolf Kalman så tidlig som i 1960 [15]. Kalmanfilteret er konstruert som et *mean squared error minimiser*, men kan også utledes slik at det kan relateres til statistiske metoder for *maximum likelihood* [3]. Denne sammenhengen ble vist for skalare, tidsuavhengige estimatører i de foregående kapitlene. Kalmanfilteret bruker da disse statistiske beregningene for å lage et variansminimerende estimat, slik at den estimerte tilstanden har lavest mulig varians over lengst mulig tid. Denne beregningen blir da gjort ved hvert målepunkt for så å øke frem til man har en ny måling.

Likningene for et Kalmanfilter ble gitt for et tidsuavhengig system i (2.6). For et tidsavhengig system vil disse likningene fremdeles gjelde, men estimatoren må også kunne endres som tiden går. Dette gjøres rekursivt ved å projisere estimatet for et tidssteg sammen med kovariansmatrisene til neste tidssteg. I [13, 3] defineres dette oppdateringssteget som

$$\begin{aligned}\psi_{k+1}^f &= G\psi_k^a \\ C_{\psi\psi,k+1}^f &= GC_{\psi\psi,k}^f G^T + C_{qq,k}\end{aligned}$$

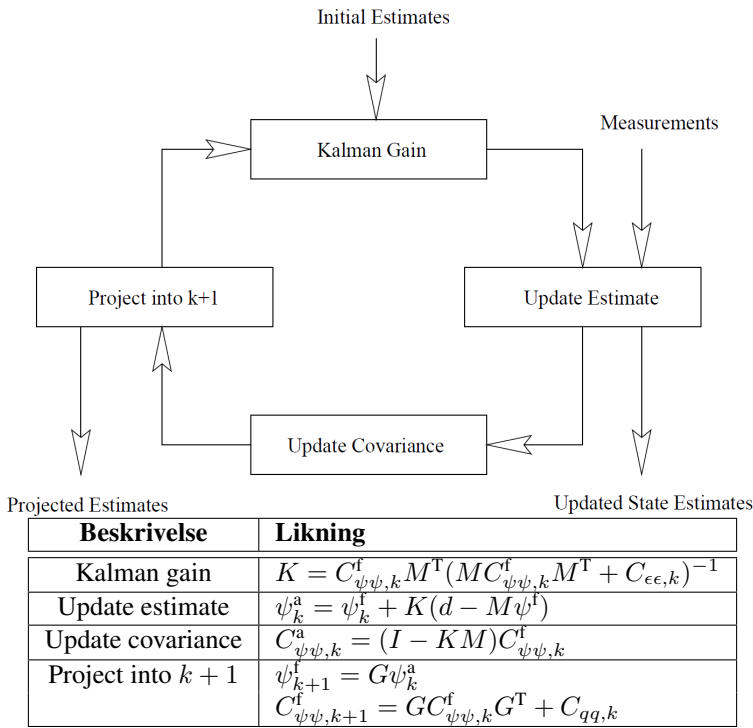
der

$$C_{qq,k} = \mathbf{E}[q_k q_k^T]$$

Dermed blir sammen med et initielt estimat ψ_0 et komplett Kalmanfilter for dynamiske systemer gitt ved likningene

$$\begin{aligned}K &= C_{\psi\psi,k}^f M^T (M C_{\psi\psi,k}^f M^T + C_{\epsilon\epsilon,k})^{-1} \\ \psi_k^a &= \psi_k^f + K(d_k - M\psi_k^f) \\ C_{\psi\psi,k}^a &= (I - KM)C_{\psi\psi,k}^f \\ \psi_{k+1}^f &= G\psi_k^a \\ C_{\psi\psi,k+1}^f &= GC_{\psi\psi,k}^f G^T + C_{qq,k}\end{aligned}\tag{2.7}$$

I figur 2.1 illustreres det hvordan et Kalmanfilter fungerer rekursivt og hvilke likninger som utbyttes i hvert punkt i algoritmen.



Figur 2.1: Illustrasjon av algoritmen som utgjør et Kalmanfilter, og likningene assosiert med hver enkelt boks. Hentet fra [3].

Extended Kalmanfilter, EKF

Der Kalmanfilteret viser seg å være en god estimator for lineære systemer, har det også vist seg at *Extended Kalman Filter*, eller et utvidet Kalmanfilter, er standarden for ulineær tilstandsestimering i for eksempel navigering og GPS[16, 17]. Fra [13] kan et EKF utledes fra et generelt dynamisk system på formen

$$\psi_{k+1} = G(\psi_k) + q_k \tag{2.8}$$

der $G(\cdot)$ er en generell, ulineær modelloperator og q_k er ukjent avvik i modellen. Et a priori estimat av tilstandene følger igjen

$$\psi_{k+1}^f = G(\psi_k^a) \tag{2.9}$$

og kovariansmatrisen for dette estimatet kan som i et vanlig KF gis som

$$\begin{aligned} C_{\psi\psi,k+1}^f &= \mathbf{E}[(\psi_{k+1}^f - \psi_{k+1}^a)(\psi_{k+1}^f - \psi_{k+1}^a)^T] \\ &= \mathbf{E}[(G(\psi_k^f) + q_k - G(\psi_k^a))(G(\psi_k^f) + q_k - G(\psi_k^a))^T] \end{aligned} \tag{2.10}$$

der (2.9) trukket fra (2.8) gir uttrykket som skal beregnes i (2.10). For at det skal kunne gi et godt estimat må en Taylorappriksimasjon av $G(\psi_k^T)$ utnyttes. Dette gir

$$G(\psi_k^t) = G(\psi_k^a) + \left. \frac{\partial G(\psi)}{\partial \psi} \right|_{\psi_k} (\psi_k^t - \psi_k^a) + \frac{1}{2} \left. \frac{\partial^2 G(\psi)}{\partial \psi^2} \right|_{\psi_k} (\psi_k^t - \psi_k^a)^2 + \text{h.o.t.}$$

der h.o.t. er en forkortelse for *higher order terms*. Dette innsatt i (2.10) gir

$$\begin{aligned} C_{\psi\psi, k+1}^f &= G'_k \mathbf{E}[(\psi_k^t - \psi_k^a)(\psi_k^t - \psi_k^a)^T] G_k'^T + \mathbf{E}[q_k q_k^T] + \text{h.o.t.} \\ &= G'_k C_{\psi\psi, k}^a G_k'^T + C_{qq, k} + \text{h.o.t.} \end{aligned} \quad (2.11)$$

der $G'_k = \left. \frac{\partial G(\psi)}{\partial \psi} \right|_{\psi_k}$ er Jacobimatrisen for systemet og at filteret baserer seg på at bidragene fra elementene av høyere orden enn 2 er neglisjerbare. Dermed kan et utvidet Kalmanfilter beskrives av likningene for et ordinært Kalmanfilter, gitt i (2.7), men a priori kovariansmatrise benytter en linearisert variant av G slik det er vist i (2.11) og naturligvis a priori estimat følger den ulineære G som vist i (2.9).

Ensemble Kalmanfilter, EnKF

I [13] nevnes er to store ulemper ved bruk av Kalmanfilter for data assimilering i dynamiske modeller i mange dimensjoner. Det første problemet er i forbindelse med lagring og datakraft. Et system som har n ukjente tilstander vil ha en avvikskovariansmatrise på n^2 elementer, og for systemer med mange ukjente vil behovet for både lagring bli enormt. Det betyr at i formen som er presentert i de foregående avsnittene vil Kalmanfilteret og det utvidete Kalmanfilteret bare være brukbart på systemer av lavere dimensjon. Det andre problemet som nevnes er at ved bruk av EKF kreves det en linearisering ved utledningen av likningen for avvikskovariansen. Dette fører til dårlig avvikskovariansmodell som igjen kan gi ustabil utvikling i avvikskovariansen. Dette kan løses med metoder som tar i bruk andre metoder i høyere dimensjoner, men da faller dette tilbake til det første problemet. Konklusjonen blir dermed at en bedre metode for problemer med mange ukjente er nødvendig. Her kommer Ensemble Kalmanfilteret (EnKF) inn i bildet. Dette er en metode som ble introdusert i [18] med mål om å få bukt med problemene som oppsto ved bruk av EKF på ulineære systemer med mange dimensjoner.

Ensemble Kalmanfilteret bruker som navnet antyder et ensemble, eller et utvalg, av tilstandene i systemet. Når størrelsen på utvalget går mot uendelig, eller antall tilstander i det faktiske systemet, vil et EnKF ikke skille seg fra et vanlig Kalmanfilter. Dette kommer frem senere i utledningen. Sammenliknet med et EKF som er deterministisk vil EnKF være stokastisk og utnytte egenskapene ved en Monte Carlo-sampling[13]. For et ensemble med N elementer i n dimensjoner vil hvert av de N elementene i utvalget ansees som et punkt i disse dimensjonene. Jo flere medlemmer det er i utvalget, desto bedre estimater vil kunne beregnes. For en Monte Carlometode vil feilen i metoden avta proporsjonalt med

$\frac{1}{\sqrt{N}}$ [19]. Et enkelt eksempel på praktisk bruk av en Monte Carlo-metode er å beregne

$$\int_0^1 x dx$$

som i praksis er det samme som arealet av en rettvinklet trekant med kateter med lengde 1, eller arealet under den rette linja fra $[x, y] = [0, 0]$ til $[x, y] = [1, 1]$. For å løse dette integralet med en Monte Carlo-metode er å definere et utfallsrom $x, y \in [0, 1]$, for så å generere tilfeldige punkter innenfor dette utfallsrommet og beregne hvor stor andel av disse som befinner seg under linja. Ved få punkter vil dette gi stor usikkerhet i beregningen av integralet, men en økning i antall punkter vil naturligvis gi en mer presis beregning.

For å utlede et EnKF må nå modellen utvides med parameteren $\bar{\psi}$ som defineres til å være gjennomsnittstilstanden til utvalget [13, 18, 20]. Siden utvalget ikke består av alle tilstandene til systemet vil kovariansmatrisene nå bygges rundt dette utvalgsgjennomsnittet. Med likning (2.1) og (2.3) og samme antakelser som er gitt i (2.2) vil da forventningsverdien $E[\cdot]$ byttes ut med et gjennomsnitt over utvalget. Det vil gi avvikskovariansmatriser utledet rundt utvalgsgjennomsnittet $\bar{\psi}$ gitt ved

$$\begin{aligned} C_{\psi\psi}^{e, f} &= \overline{(\psi^f - \bar{\psi}^f)(\psi^f - \bar{\psi}^f)^T} \\ C_{\psi\psi}^{e, a} &= \overline{(\psi^a - \bar{\psi}^a)(\psi^a - \bar{\psi}^a)^T} \end{aligned} \quad (2.12)$$

der f og a betyr det samme som før og e symboliserer at det her handler om et ensemble. Videre har vi målinger

$$d_j = d + \epsilon_j, \quad j \in [1, N]$$

der N er antallet medlemmer i utvalget. Kovariansmatrisen for avviket i målingene blir da

$$C_{\epsilon\epsilon}^e = \overline{\epsilon\epsilon^T}$$

som vil konvergere mot $C_{\epsilon\epsilon}$, altså notasjonen som hører til et vanlig Kalmanfilter, når N går mot uendelig. Videre vil analysesteget i EnKF bestå av likningen

$$\begin{aligned} \psi_j^a &= \psi_j^f + \underbrace{C_{\psi\psi}^{e, f} M^T (M C_{\psi\psi}^{e, f} M^T + C_{\epsilon\epsilon}^e)^{-1}}_{K_e} (d_j - M \psi_j^f) \quad \forall j \in [1, N] \\ \implies \bar{\psi}^a &= \bar{\psi}^f + K_e (\bar{d} - M \bar{\psi}^f) \end{aligned} \quad (2.13)$$

Denne formuleringen er da identisk til formuleringen i et vanlig Kalmanfilter, sett bort ifra bruken av at det er kovariansen for utvalget som brukes. Antas da at gjennomsnittet $\bar{\psi}^a$ er det beste estimatet, kan kovariansmatrisene oppdateres. Først brukes (2.13) som gir

$$\psi_j^a - \bar{\psi}^a = (I - K_e M)(\psi_j^f - \bar{\psi}^f) + K_e (d_j - \bar{d})$$

før dette brukes sammen med (2.12) til å utlede

$$\begin{aligned}
C_{\psi\psi}^{\text{e,a}} &= \overline{(\psi^{\text{a}} - \overline{\psi^{\text{a}}})(\psi^{\text{a}} - \overline{\psi^{\text{a}}})^{\text{T}}} \\
&= \overline{\left((I - K_e M)(\psi^{\text{f}} - \overline{\psi^{\text{f}}}) + K_e(d - \overline{d}) \right) \left((I - K_e M)(\psi^{\text{f}} - \overline{\psi^{\text{f}}}) + K_e(d - \overline{d}) \right)^{\text{T}}} \\
&= (I - K_e M) \overline{(\psi^{\text{f}} - \overline{\psi^{\text{f}}})(\psi^{\text{f}} - \overline{\psi^{\text{f}}})^{\text{T}}} (I - K_e M)^{\text{T}} + K_e \overline{(d - \overline{d})(d - \overline{d})^{\text{T}}} K_e^{\text{T}} \\
&= (I - K_e M) C_{\psi\psi}^{\text{e,f}} (I - K_e M)^{\text{T}} + K_e C_{\epsilon\epsilon}^{\text{e}} K_e^{\text{T}} \\
&= C_{\psi\psi}^{\text{e,f}} - C_{\psi\psi}^{\text{e,f}} (K_e M)^{\text{T}} - K_e M C_{\psi\psi}^{\text{e,f}} + K_e M C_{\psi\psi}^{\text{e,f}} (K_e M)^{\text{T}} + K_e C_{\epsilon\epsilon}^{\text{e}} K_e^{\text{T}} \\
&= (I - K_e M) C_{\psi\psi}^{\text{e,f}} - C_{\psi\psi}^{\text{e,f}} M^{\text{T}} K_e^{\text{T}} + K_e (M C_{\psi\psi}^{\text{e,f}} M^{\text{T}} + C_{\epsilon\epsilon}^{\text{e}}) K_e^{\text{T}} \\
&= (I - K_e M) C_{\psi\psi}^{\text{e,f}}
\end{aligned}$$

der definisjonen av K_e utnyttes til å kansellere de siste leddene. Formuleringen som er utledet her er på samme form som et tradisjonelt Kalmanfilter slik som i (2.5), og impliserer dermed at et EnKF i tillegg til at det minimerer avvikskovariansen konvergerer mot et vanlig KF lineær dynamikk når utvalget blir stort nok. For ulineær dynamikk så tar EnKF høyde for de ulineære leddene i en modell, og det er dermed ikke behov for linearisering eller antakelser slik som i et EKF. Dette gjør at et EnKF er lett å implementere for praktiske formål [13, 20].

2.1.3 Assimilering innefor andre disipliner

Fagfelt der datassimilering av punktmålinger med distribuerte modeller er utbredt er, naturlig nok, disipliner som skal analysere store tilstandsmodeller ved hjelp av få målinger. Eksempler på dette er hydrologi og meteorologi. I de følgende avsnittene vil det kort forklares hva slags metoder som har blitt brukt i forskjellige sammenhenger.

Prognoser for snøsmelting og snødekke

I [21] er utgangspunktet for forskningen at avgjørelser i forbindelse med vannressurser i alpine og høytliggende områder er sterkt knyttet til mengden snø og smeltemraten til denne snøen, og at disse områdene kan være utsatt for mye skyer, alpine områder og andre ting som gjør satelittobservering vanskelig. I forbindelse med dette nyttes distribuerte snømodeller for snødekke og smeltemrate, og spørsmålet som det forsøkes å få et svar på er om assimilering kan være med på å forbedre modellene for disse prognosene. Artikkeforfatterne utforsker et område i Sveits gjennom å bruke utplasserte snødybdemålere og målere av *snow water equivalents*, SWE, og bruker Ensemble Kalmanfilter og statistisk interpolasjon til å teste to forskjellige assimileringmetoder; assimilasjon av fluks, altså regn- og snøfall og avrenninger og en som assimilerer SWE-tilstandene direkte. Resultatene i denne studien viste at en metode som assimilerer fluksene, ga et statistisk signifikant bedre resultat enn en kontrollsimulering som ikke utnyttet filteralgoritmene, mens en alternativ tilnærming som oppdaterte tilstandene direkte ikke ga signifikant bedre resultat. Likevel konkluderes det med at en kan si at sekvensielle datassimileringmetoder kan gi bedre resultater i prosessen av å overføre informasjon fra distribuerte punktmålinger av

snødybde og SWE.

En liknende utgangspunkt ble sett på i [22]. Her var målet å kunne gi gode prognoser for avrenning i høytliggende vassdrag på kort sikt ved hjelp av assimilering av sensordata og meteorologiske data. Dette for å kunne gi informasjon i forbindelse med forvaltning av vannressurser. Det ble brukt en modell kalt *Snowmelt Runoff Model*, SRM, da denne var den eneste modellen som var optimalisert for å få input fra eksterne snødekkensensorer og -følere. SRM beregner så snøsmeltingen med bakgrunn i *snow covered area*, SCA, eller snødekke. Artikkelforfatterne viser til flere andre artikler der slike målinger fra eksterne sensorer brukes som elementer i et Ensemble Kalmanfilter for å gi bedre prognoser for faktisk SCA og avrenning og smelting, og at dette konseptet har stort potensiale. Det ble forsøkt å forbedre disse konseptene med å assimilere inn meteorologiske data fra satellittbilder, slik at daglige avrenninger ble spådd ved deterministiske meteorologisk prognoser sammen med 51 ensemblepreksjoner. Det samme ble funnet her; at *Evaluation of the runoff forecasts reveals good agreement with the measurements, confirming the usefulness of the selected data processing and assimilation scheme for operational use*. Likevel ble det konkludert med at store forbedringer vil være umulig uten utvikling på numeriske meteorologiske prognoser da feil i disse var hovedfeilkilden i denne studien. I tillegg påvirker feil i temperaturprognoser avrenningsprognosen i stor grad.

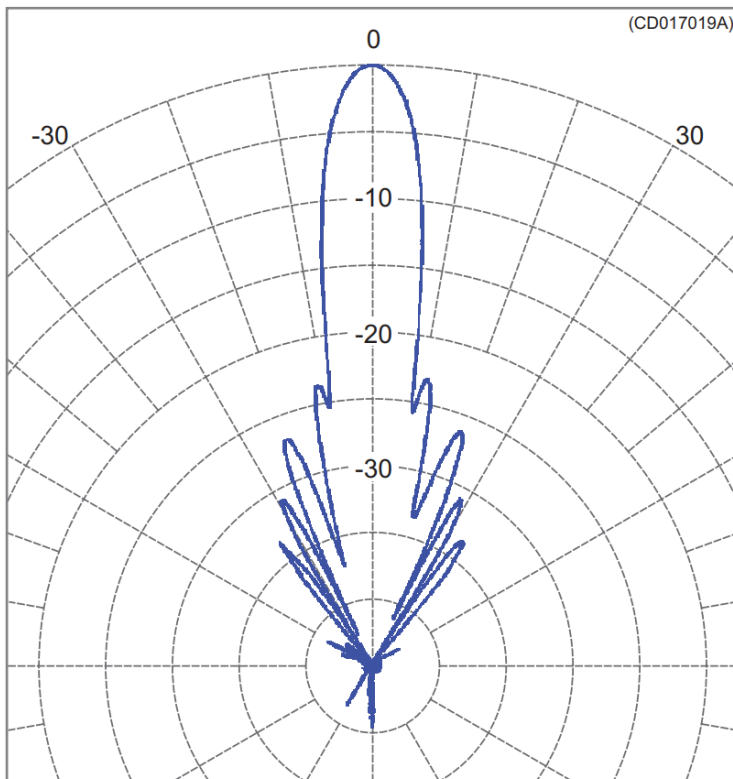
I tillegg til de to artiklene som er utdypet over finnes det mange gode eksempler på bruk av datassimilering med bruk av Ensemble Kalmanfilter. I [23] er det igjen prognoser for snødekke og -smelting som ligger til grunn for studien. Her vises det at EnKF er en både effektiv og mulig løsning da filteret med suksess oppdaterte modellert *snow cover extent*, SCE, til å bedre etterlikne observasjonene gjort med teknikken *moderate resolution imaging spectroradiometry*, MODIS, enn for eksempel simuleringer gjort med åpen sløyfe. I [24] er også forvaltning av vannressurser utgangspunktet, men her i nedlagsfeltet til elvevassdrag. Det konkluderes med at et EnKF gir tilfredsstillende resultater på avrenning, jordfuktighet og fordampningsestimater, men at resultatet kunne blitt enda bedre ved bruk av blant annet jordfuktighetsmålinger. Innenfor andre fagfelt som for eksempel oppskalering av punktmålinger av jordfuktighet, slik studien i [25] har som utgangspunkt, viser det seg også at dataassimilering av målinger inn i matematiske modeller gir tydelig bedre nøyaktighet i modellen. Som en konklusjon på dette viser det seg at dataassimilering gir forbedrede resultater og at EnKF er en god måte å gjøre denne datassimileringen på.

2.2 Simrad EK80

I [26] ble det til en viss grad utledet forskjeller mellom forskjellige typer ekkolodd og sonarer og deres bruksområder. Hovedområdene det ble fokusert på var den generelle forskjellen på aktiv og passiv sonar, i tillegg til forskjellen på single-, multi- og split-beamsendere. I studeiene utført i både [27] og [28] brukes Simrads EK60 ekkolodd til deteksjon og observasjon av fisk. EK60 er forgjengeren til Simrads nye toppmodell EK80, som i følge egen referansemanual er *the most modern high-end split-beam scientific echo sounder in the scientific market*, og kan brukes både innen fiskeoppdrett, planktonforskning, habitatkartlegging, olje- og gassdeteksjon, i tillegg til andre relevante disipliner i følge egen

manual[29]. Det er dette ekkoloddet som det tas utgangspunkt i videre i denne oppgaven.

Ekkoloddene som brukes i Ocean Farm 1 er som nevnt over av typen Simrad EK80 med transducerhode av typen Simrad ES70-7C. Dette er en split-beam transducer som i følge databladet [4] opererer på 70 kHz med 7° beambredde. Videre har den et operativt temperaturområde på -20 til 70 °C og kan plasseres ned på 20 m. Alle disse tallene tas fra Kongsbergs eget datablad, men dette er da typen som plasseres så dypt som 30 m i Ocean Farm 1. Dydbegrensningen kan da tas med en liten klype salt. I figur 2.2 vises det hvordan signalstyrken avtar med vinkel, og det er tydelig at sidelobene er små, og dermed er det responsen fra hovedloben som er dominerende.

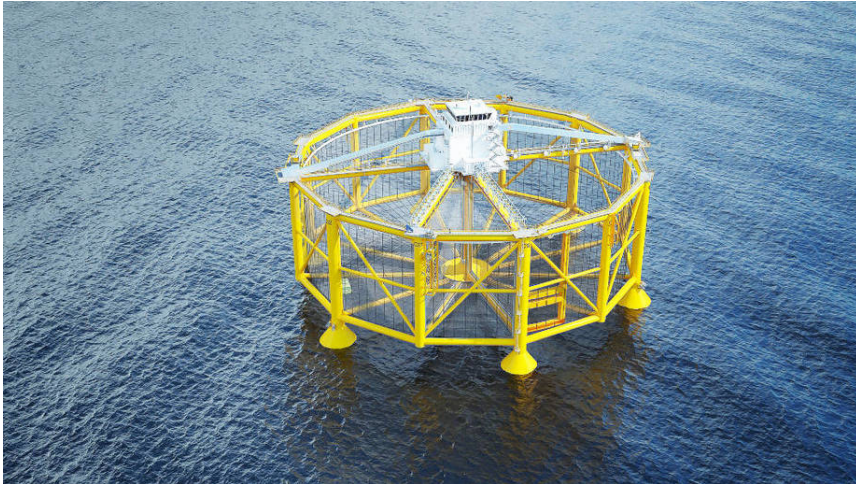


Figur 2.2: Strålemønster for transduceren Simrad ES70-7C. Figur hentet fra [4].

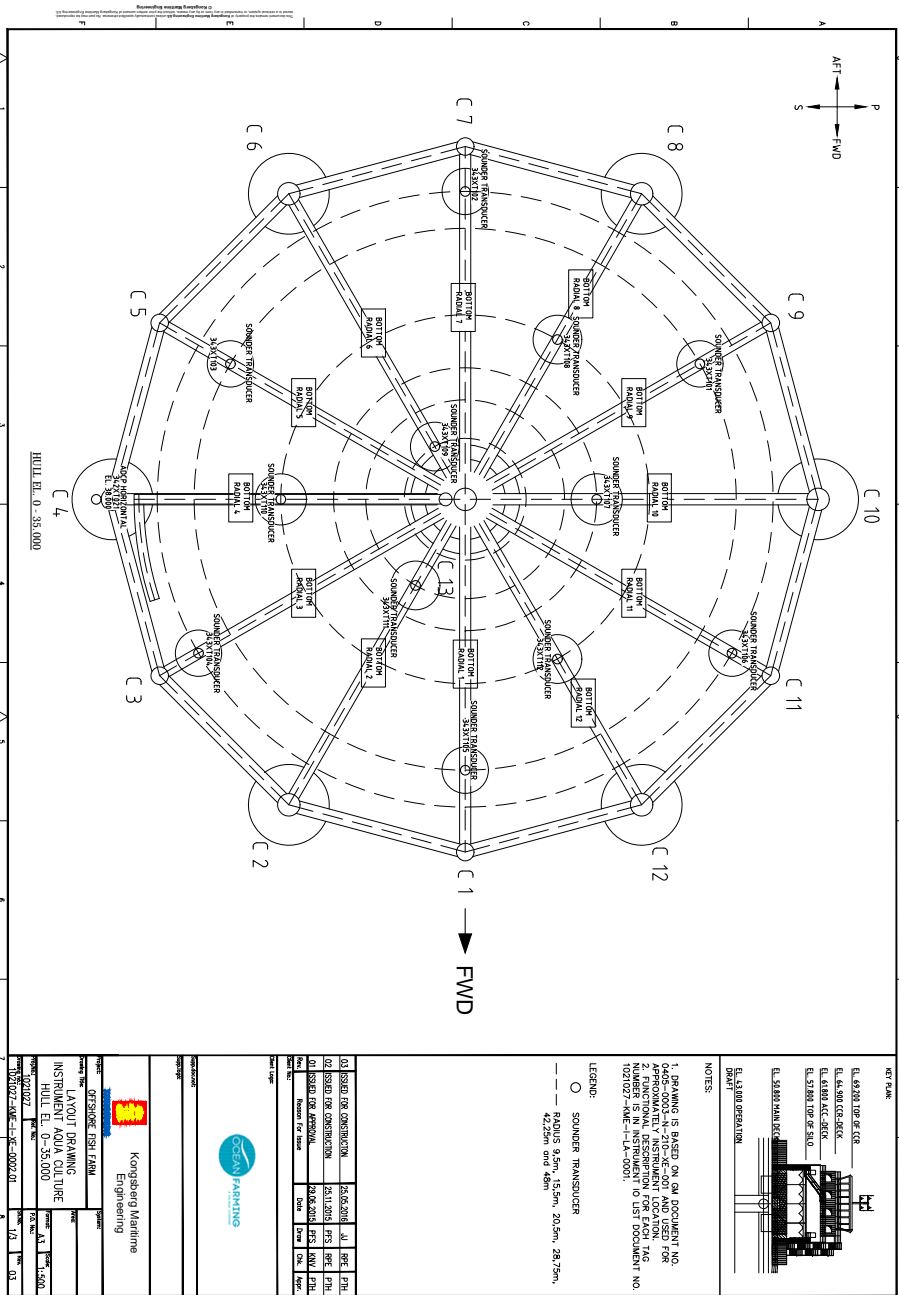
Et ekkolodd som dette, og ikke minst tolv, vil levere enorme mengder data til prosessering. Dataene leveres som .raw-filer som må behandles av egnet programvare. Til EK80-ekkoloddet finnes det egen programvare, ellers er det utviklet MATLAB-programmer som leser inn rådata og gir tilbake ekkogrammer. Et eksempel på et slikt MATLAB-prosessert ekkogram kan sees i figur 3.1.

Ocean Farm 1

Med utgangspunktet oppgaven har er det naturlig med et lite avsnitt om hvordan Ocean Farm 1 er bygget opp. I motsetning til de vanlige merden en vanligvis ser i sjøen er Ocean Farm 1 en rigid struktur beregnet for mer værutsatte lokasjoner. I tillegg er den vesentlig større enn en vanlig merd med diameter på 110 m og volum på 250 000 m³[8]. Ocean Farm 1 er også i følge egne nettsider et fullskala pilotprosjekt som har blitt *utrustet for forskningsoppgaver, med særlig fokus på biologiske forhold og fiskevelferd*. På denne usedvanlig store konstruksjonen er det som nevnt plassert ut tolv EK80 ekkolodd. Disse er plassert som det er vist i figur 2.4. Et bilde av Ocean Farm 1 er vist i figur 2.3.



Figur 2.3: Bilde av Ocean Farm 1.



Figur 2.4: Tegninger av Ocean Farm 1 med plasseringer av ekkolodd. Tilsendt fra Kongsberg og giengitt med tillatelse.

2.3 FhSim

2.3.1 Grunnleggende om FhSim

For at en simulert modell skal kunne representere virkeligheten på en god måte, er det viktig at de forskjellige komponentene er korrekte og gir et godt bilde av virkeligheten. I de følgende avsnittene utdypes de to (kanskje) viktigste av disse forskjellige komponentene som en fullverdig simulering består av, nemlig selve atferdsmodellen for fisk og distribusjonen av fôr. Modelleringen av miljøet neglisjeres, da videre arbeid i denne rapporten stort sett ser bort ifra miljøpåvirkning.

Fiskemodellen

Den viktigste komponenten for en god simulering er modellen av selve fisken. I [1] presenteres en individsbasert matematisk atferdsmodell for laks utformet gjennom forskning hos SINTEF, NTNU og Havforskningsinstituttet som er gitt ved

$$\dot{r}_{\text{ref}} = \tau \dot{r}_{\text{prev}} + (1 - \tau) (k_C v_C + k_F v_F + k_T v_T + k_L v_L + k_{SO} v_{SO} + k_{ST} v_{ST}) \quad (2.14)$$

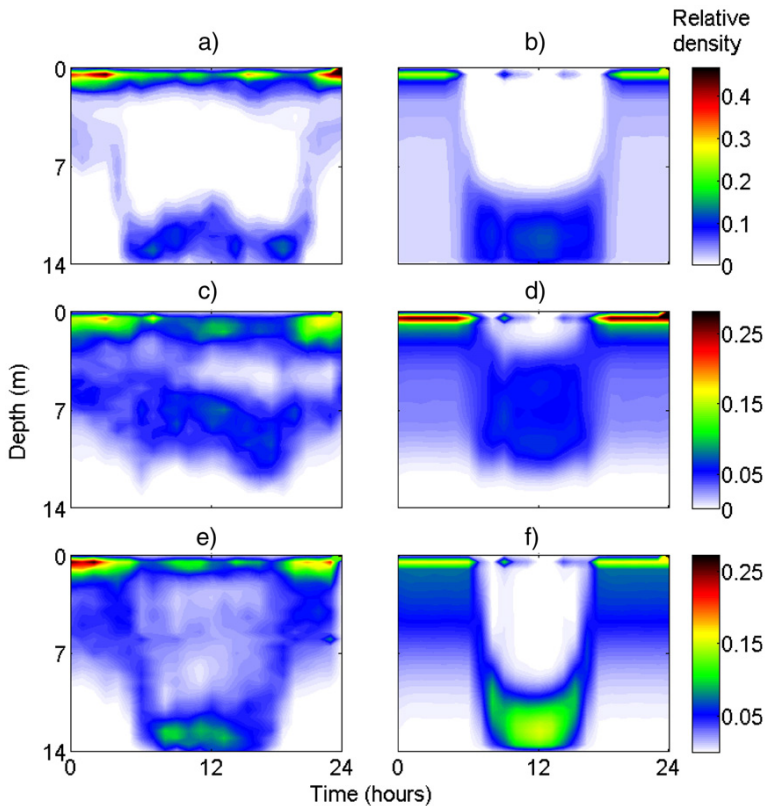
hvor \dot{r}_{ref} representerer en svømmehastighetsreferanse i x -, y - og z -retning for det enkelte lakseindividet. Denne referansen regnes ut med bakgrunn i forrige svømmehastigheten i forrige tidssteg, \dot{r}_{prev} og vektete responser mot forskjellige faktorer som påvirker laksens atferd, for eksempel k_C som er hvor mye responsen mot notveggen vil være. Disse er listet opp i kronologisk rekkefølge med den tyngst vektete faktoren først, og er forklart i tabell 2.1. I tillegg er det en vektingsparameter τ som bestemmer hvor tungt forrige tidssteg skal vektes mot andre påvirkninger.

Tabell 2.1: Forklaring av indeksene som brukes i likning (2.14), forenklet og gjengitt fra [1].

Forkortelse	Hva den representerer
C	Cage. Respons til notvegger, overflate og bunnot.
F	Food. Respons til fôring.
T	Temperature. Temperaturrespons.
L	Light. Lysrespons.
SO	Social. Respons på andre individers oppførsel.
ST	Stochastic. Et innslag av tilfeldighet i oppførsel.

Denne modellen har gjennom observasjon av småskala oppdrettsmerd vist seg å stemme bra når det gjelder vertikalfordelingen av fisk. Med et ekkolodd plassert under merden slik at hele merden kom innenfor beamen ble det i [30] og [1] observert at mens fisken på kvelds- og nattetid beveger seg høyere i vannmassen så trekker den mot dypet på dagtid, og figur 2.5 kan nettopp dette sees. I tre forskjellige scenarier fordelt på tre dager i september og oktober 2002 gjorde [30] observasjoner av vertikalfordelingen til fisk. Målte miljødata fra disse dagene ble utgangspunktet for simuleringen i [1], og igjen; simuleringene viser at modellen gjengir vertikalfordelingen godt. Simuleringene er naturlig nok noe "smoother", noe som også er å forvente av en matematisk modell av et biologisk system som ikke er deterministisk. Et punkt som også er verdt å nevne i forbindelse med figur 2.5

er at det er perioder midt på dagen der fisken trekker mot overflaten både i den observerte og den simulerte modellen. Dette er respons på at det skjer fôring, og at fisken da beveger seg mot fôret selv om det befinner seg i et område som er utenfor foretrukket område med tanke på lys og temperatur. Fra 2.14 kan det sees at dette er en respons som er å forvente i den simulerte modellen, da respons til fôr er vektet tyngre enn respons fra temperatur og lys.

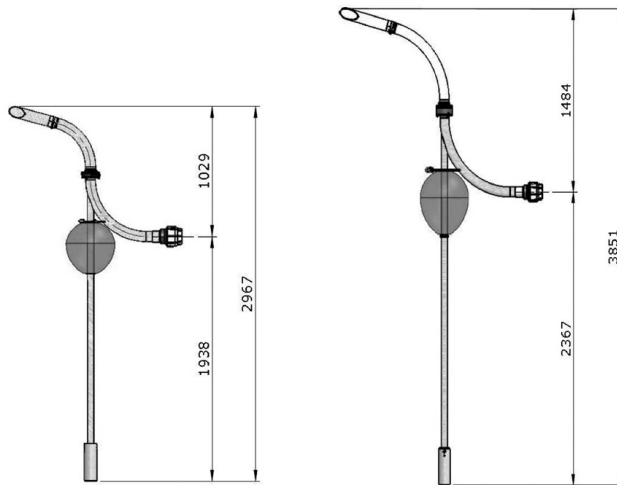


Figur 2.5: Sammenlikning av vertikalfordelingen av observert fisk (venstre) og simulert fisk (høyre) der simuleringen har tilsvarende miljøparametre som ble målt i scenariene som ble observert. Figuren er gjengitt fra [1].

Distribusjon av fôr

En annen komponent som må på plass for at simuleringen kan være sammenlignbar med virkeligheten er modellen av fôrdistribusjon. Da fôrkostnaden ved lakseoppdrett i 2018 i følge Fiskeridirektoratet i rapporten *Lønnsomhetsundersøkelse for produksjon av laks og regnbueørret 2018*[31] stod for nesten 50% av de total kostnadene gjennom en produksjonssyklus er det selvsagt ønskelig å maximere tilgjengeligheten for fôr for fisken og

samtidig minimere fôrtap. I [5] ble det testet to forskjellige typer pneumatiske, roterende fôrspredere ved å sende ut fôrpelletts ved forskjellige forskjellige konfigurasjoner og fôrtyper, for å finne ut hvordan fôret fordeler seg i en merd med disse to typene sprederer. I figur 2.6 illustreres det hvilke typer fôrsprederer som er brukt i forsøket. Resultatene fra disse forsøkene viser at fordelingen av pellets ikke fordeler seg uniformt i merden, men som funksjon av retning og lufthastighet. I figur 2.7 illustreres resultatene av disse forsøkene, og det kan leses at spredningen av pellets som sagt avhenger av luftstrøm og retning, og danner et sirkelformet mønster der tettheten er forskjøvet mot den ene siden. Dette blir kalt *dense direction*, mens retningen med lavere konsentrasjon blir kalt *dispersed direction*. På $20 \frac{\text{m}}{\text{s}}$ er ikke forskjellen på avstanden til fordeleren signifikant for noen fôrtyper eller konfigurasjoner på spreder, mens for høyere lufthastigheter kan det observeres større spredning avhengig av konfigurasjon. Disse dataene danner grunnlaget for videre matematisk modellering og parametrisering av fôrdistribusjon i [32] og [2].

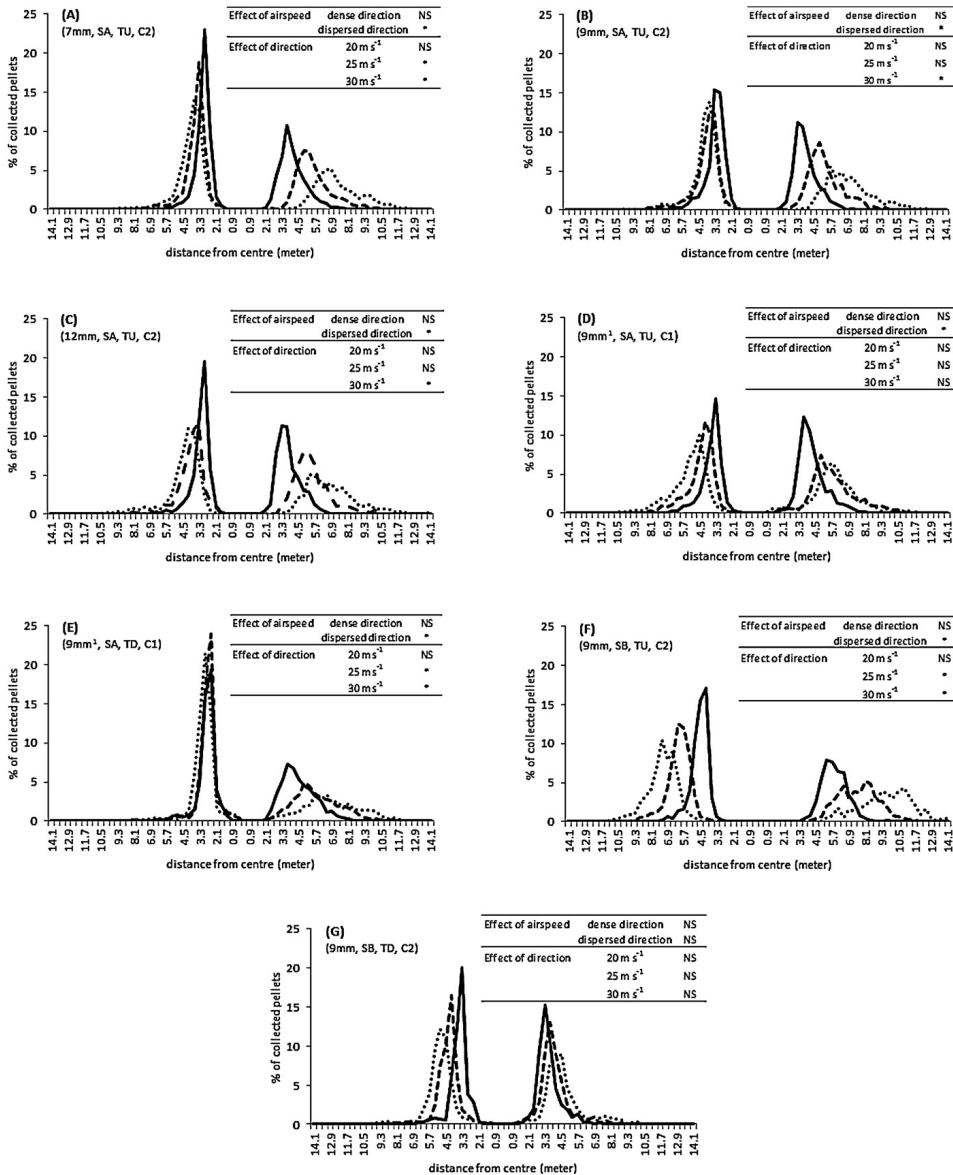


Figur 2.6: Dimensjoner for de to forskjellige fôrsprederne brukt i [5]. Sprederen til venstre er av typen RS-90C og sprederen til høyre av typen RS-90C MK III, begge fra AKVAGroup. Figur er hentet fra [5].

Med bakgrunn i forsøkene om hvordan fôrpelletts distribueres i oppdrettssammenheng vil [32] og [2] modellere og parametrisere dette matematisk, for så å validere modellen opp mot de observerte data. I [2] introduseres en sannsynlighetsfordeling gitt som funksjon av lufthastighet v_{air} , avstand til sprederen d og vinkel relativ til fordeleren ψ . Denne fordelingen er gitt ved

$$P(v_{\text{air}}, d, \psi) = \frac{1}{X_2 \sqrt{2\pi}} \exp\left(-\frac{(d' - X_1)^2}{2X_2^2}\right)$$

som er en normalfordeling med forventningsverdi X_1 og standardavvik X_2 som bestem-



Figur 2.7: Fordeling av pellets i avstand fra fôrfordeler med tre forskjellige hastigheter på luftstrømmen. 7 til 12 mm betyr hvilken størrelse fôrpellet som ble brukt i de forskjellige forsøkene, SA og SB betyr henholdsvis *Spreader A* og *Spreader B*, og forteller hvilken spredertype som er brukt, TU eller TD forteller om sprederen var tiltet opp (*U*) eller ned (*D*) og C1 og C2 forteller hvilken merd (*cage*) som forsøket ble utført i. NS betyr at den statistiske analysen for forsøket ikke var signifikant med $p < 0.05$. Figuren er hentet fra [5].

mes ved likningene

$$X_1 = (p_1 + p_2 v_{\text{air}}) p_{1,\text{tilt}} \left(1 - \frac{\psi}{180}\right) + (p_3 + v_{\text{air}} p_4) p_{2,\text{tilt}} \frac{\psi}{180}$$

$$X_2 = (p_1 + p_2 v_{\text{air}}) p_5 p_{1,\text{tilt}} \left(1 - \frac{\psi}{180}\right) + (p_3 + v_{\text{air}} p_4) p_6 p_{2,\text{tilt}} \frac{\psi}{180}$$

der

$$p_{1,\text{tilt}} = \begin{cases} 1 & : \text{for tilt up} \\ p_7 & : \text{for tilt down} \end{cases}$$

$$p_{2,\text{tilt}} = \begin{cases} 1 & : \text{for tilt up} \\ p_8 & : \text{for tilt down} \end{cases}$$

p_1 - p_8 beskriver her fordelingsmønstre for fremover ($\psi = 0$) og bakover ($\psi = 180$) for sprederen. For $\psi \in [0, 180]$ interpoleres fordelingen med den vektet forskjellen mellom fremover og bakover, mens for å etterligne den forskjøvete fordelingen settes

$$d' = d^{p_9}$$

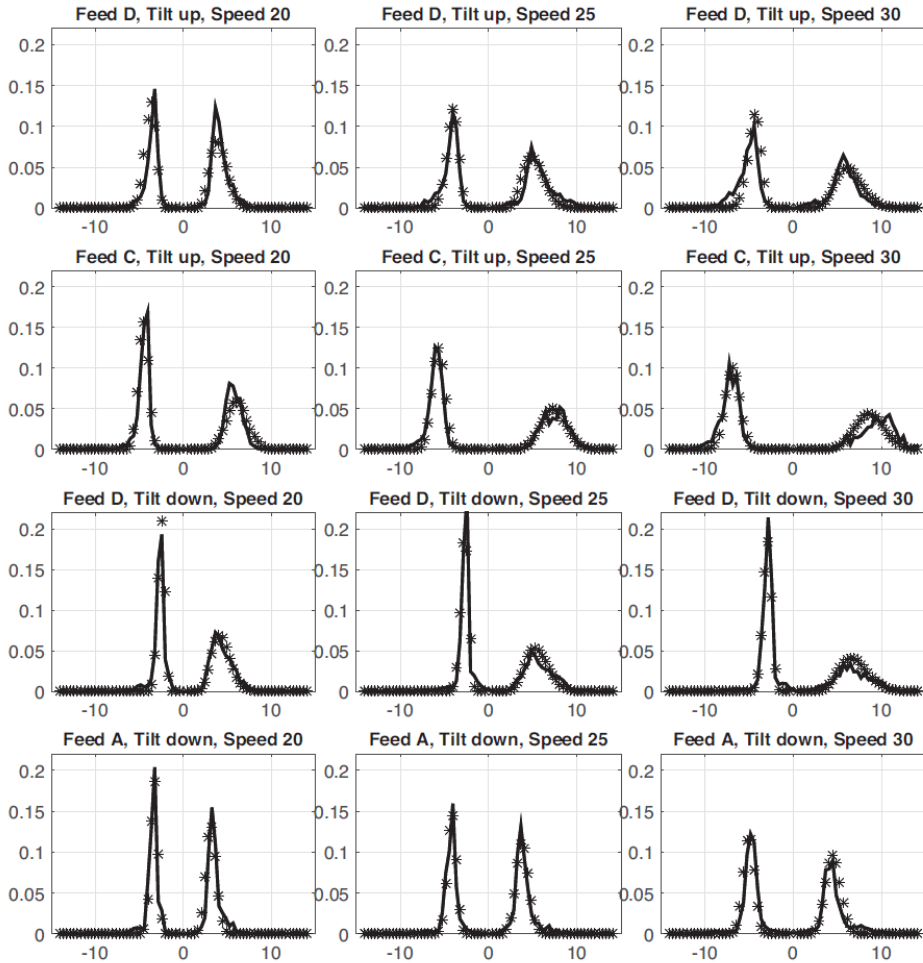
Verdiene for p_1 - p_9 ble funnet til å være optimale med verdiene i tabell 2.2, og med denne parametriseringen ble det gjort simuleringer med tilsvarende oppsett som i [5]. Resultatene viste at modellen stemte svært godt mot observert data, noe som kan leses fra figur 2.8. I figur 2.9 illustreres det hvordan fôret vil fordele seg rundt en fôrspreder med lufthastighet lik $25 \frac{\text{m}}{\text{s}}$ både med tilt opp og ned.

Tabell 2.2: Optimale verdier for fôrspredermodellen. Gjengitt fra [2]

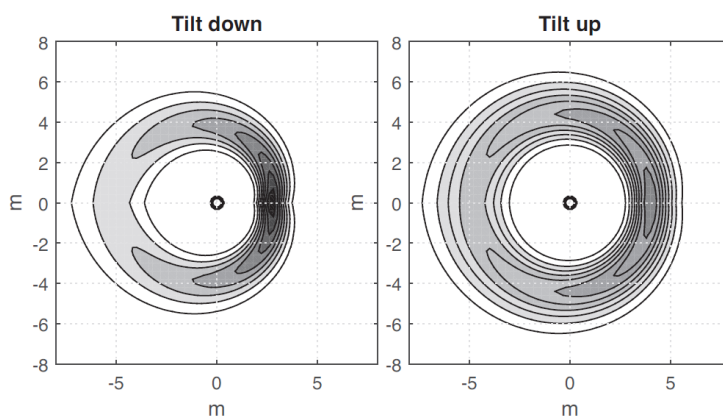
	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9
Type 1	1.4145	0.0101	0.0602	1.0645	0.0297	0.0873	0.8645	1.0270	0.3608
Type 2	0.7373	0.1302	0.0971	1.5705	0.1351	0.1489	0.7823	0.5950	0.7912

2.4 Fiskeatferd for et reelt scenario

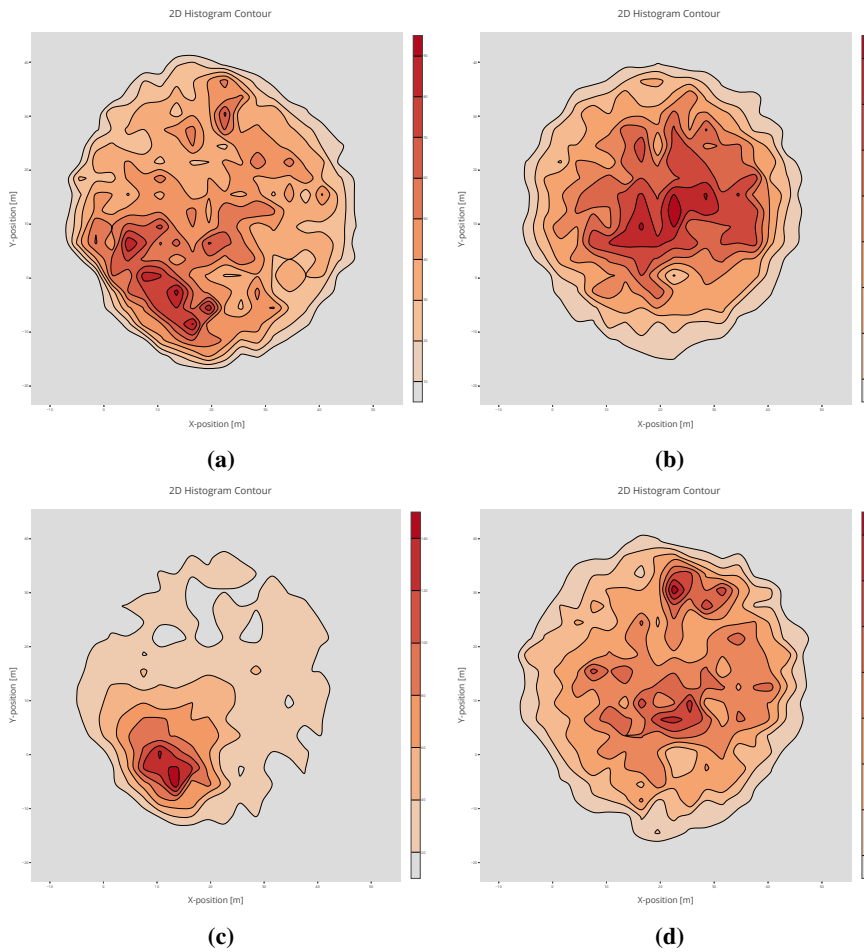
I [6] ble telemetrisk data fra noen titalls merkede fisk i en reell oppdrettsituasjon med føring sentralt analysert. Akkumulert data over omtrent tre måneder viste svært individuelle lokasjonspreferanser, og kan sees i figur 2.10. Selv om dette er et meget snevert utvalg av fisk, så stemmer disse mønstrene lite overens med simulerte scenarier som vist i [26] der all fisk trakk langt ut mot kanten, og er indikasjoner på at modellen har forbedringspotensiale på xy -fordelingen. I [6] ble det også sett på dybdemønsteret til de merkete fiskene, og resultatet kan sees i figur 2.11. Det er tydelig at fisken følger de samme dybdepreferansene som tidligere vist. Dette er nok et resultat som viser at fiskens respons i z -retning er presis.



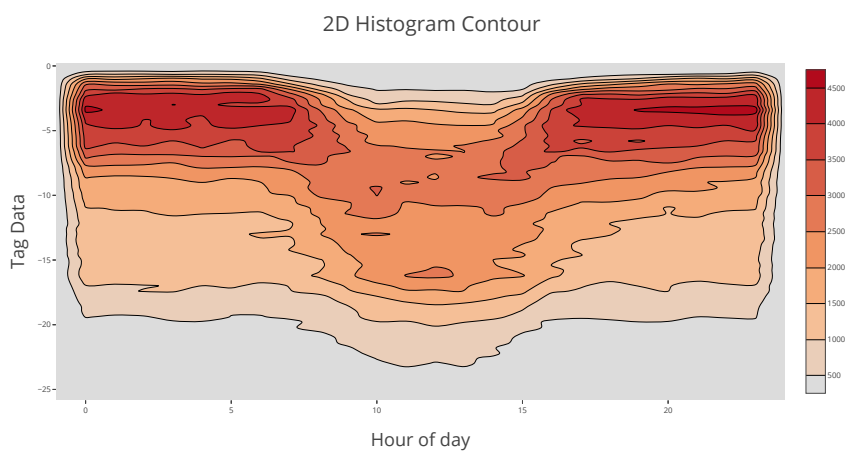
Figur 2.8: Sammenlikning med måledata og modellerte data for pelletfordeling ved forskjellige scenarier. Måledata er de heltrukne linjene, mens modellert data er punktene. Tittelen på underfigurene forteller om konfigurasjonen ved den enkelte simuleringen, altså om det var tilt opp eller ned, hvilken fôrtype som ble brukt og luftfashigheten. Figuren er hentet fra [2].



Figur 2.9: Simulert xy -fordeling av før med type 1-spreder. Sprederen er plassert i origo med retning forover, eller *dense direction*, i positiv x -retning. Figur hentet fra [2].



Figur 2.10: Akkumulert telemetrisk data fra fire merkede lakseindivider over omtrent tre måneder i et kommersielt anlegg. Figurer hentet fra [6].



Figur 2.11: Dybdemønster fra teletrisk merket fisk. Hentet fra [6].

Kapittel 3

Metode

Assimilering av måledata fra ekkolodd i reelle oppdrettssituasjoner inn i en matematisk modell kommer ikke av seg selv, og det er mye som må på plass for å få til dette. Både fra reelle og simulerte ekkolodd, og fra den matematiske atferdsmodellen genereres det enorme mengder data som skal prosesseres til noe fornuftig før det da kan utnyttes til den faktiske assimileringprosessen. I tillegg må modellen valideres slik at assimileringprosessen mellom reelle data og simulerte data skal kunne gi resultater som gir mening. Dette kapittelet vil ta for seg utviklingen av mye av rammeverket som må på plass for at dette skal kunne bli muligjort.

3.1 Matematiske målemodeller

Det første som må på plass er en matematisk modell av et ekkolodd. Output fra et ekkolodd er ofte, slike som skal vises i figurer fra reelle ekkolodd i senere avsnitt, representert av en tidsserie der det vises tetthet av fisk over varierende avstand fra ekkoloddet i én dimensjon. For å kunne sette sammen en slik tidsserie må det to ting på plass; en modell for å detektere fisk innenfor en ekkoloddmodell og en metode for å slå dette sammen til én dimensjon. De følgende underkapitlene forklarer hvordan dette gjøres i modellen som er utviklet.

Ekkoloddmodell

En enkel matematisk modell for å observere om et simulert fiskeobjekt med posisjon $[x_f \ y_f \ z_f]$ befinner seg på innsiden av en kjegleformet sonarbeam fra et ekkolodd plassert i posisjonen $[x_s \ y_s \ 0]$ ble utviklet i [26]. For å kunne utnytte denne modellen også når ekkoloddet er plassert andre steder enn i overflaten, f.eks. på 30 m dyp slik som det er på Ocean Farm 1, utvides modellen med parametreren

$$r(z) = |z_s - z| \tan \frac{\alpha}{2}$$

hvor z_s er dybden ekkoloddet er plassert, z er dybden til fiskeobjektet og α er den totale

åpningsvinkelen på ekkoloddets sender/mottaker, slik at modellen for et fiskeobjekt med posisjon $[x_f \ y_f \ z_f]$ blir gitt ved

$$\|(x_f, y_f) - (x_s, y_s)\|_2 \leq r(z_f) = |z_s - z_f| \tan \frac{\alpha}{2} \quad (3.1)$$

Denne justeringen vil gi muligheten til å observere om et fiskeobjekt befinner seg innenfor et simulert ekkolodd både plassert under vann med sonarbeamen pekende oppover, i tillegg til i overflaten med sonarbeamen pekende nedover. I vedlegg A.1.1 er det vist hvordan denne funksjonen er implementert

Fisketetthetsmodell

Et ekkogram fra et EK80-ekkolodd er vist i figur 3.1. I stedet for å se en kjegle eller trekantformet figur for hvert tidssteg, slik som for eksempel en ultralydabildning, samles det her opp data for mange tidssteg som gir en tidsserie. Siden x-aksen her er delt opp i tidssteg må dataene for hvert tidssteg konkateneres til én dimensjon. For å kunne lage sammenlignbare figurer og data fra den simulerte modellen må da disse dataene nødvendigvis også konkateneres til én dimensjon. Dette gjøres ved at z-aksen til det simulerte ekkoloddet diskretiseres i Δz lange intervaller, for så å telle opp antall fiskeobjekter som befinner seg innenfor sonarkjeglen fra et punkt til det neste, før antallet fisk deles på volumet av kjegleskiven. Dette gir da en tetthetsmodell som er representativ både der kjeglen dekker et lite volum og der den dekker et stort volum. Volumet av disse kjegleskivene er gitt ved

$$\begin{aligned} V_{z_0, z_0+\Delta z} &= \int_{z_0}^{z_0+\Delta z} A(z) dz = \int_{z_0}^{z_0+\Delta z} \pi r^2(z) dz \\ &= \int_{z_0}^{z_0+\Delta z} \pi \tan^2 \frac{\alpha}{2} z^2 dz \\ &= \frac{\pi}{3} \tan^2 \frac{\alpha}{2} [(z_0 + \Delta z)^3 - z_0^3] \end{aligned}$$

hvor z_0 representerer en tilfeldig avstand fra ekkoloddets utgangsposisjon. Tettheten av fiskeobjekter innenfor hvert av disse volumene er da gitt ved

$$\text{density}_{z_0, z_0+\Delta z} = \frac{\#\text{fish}_{z_0, z_0+\Delta z}}{V_{z_0, z_0+\Delta z}} \quad (3.2)$$

Tettheten av fisk vil da representere hvor fisk står på forskjellige dyp og tar i tillegg høyde for at ekkoloddet vil dekke et større område lenger unna utgangspunktet, noe som er viktig for at den dataen som observeres gjengir det som faktisk er tilstanden i enten en ekte merd eller en simulering.

3.2 Simuleringer

Et viktig punkt i steget mot simulerte ekkogrammer er nødvendigvis atferdssimuleringer av fisk. Gjennom FhSim simuleres her forskjellige scenarier for å ha et godt datagrunnlag for videre analyse. Parametrene som varieres er i hovedsak antall fisk, miljøparametre og

enkelte komponenter i fiskens atferd. Dette for å kunne validere modellen slik den er og utvikle den videre, i tillegg til å redusere kompleksiteten til et nivå som gjør det lettere å validere og utvikle.

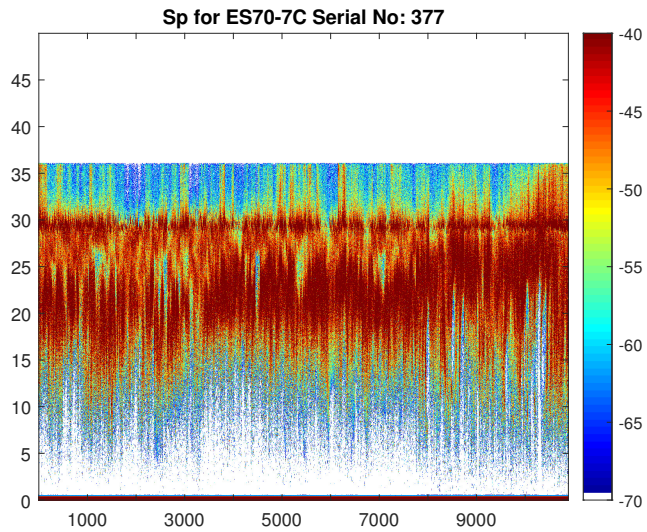
Simuleringene i FhSim genererer for hvert tidssteg en .xml-fil der hver linje representerer ett fiskeobjekt på formen $[x \ y \ z \ \dot{x} \ \dot{y} \ \dot{z}]$. I denne omgangen er det kun posisjonen til fisken som er interessant, men hastighetsdata for hvert enkelt individ er tilgjengelig for analyse der dette skulle finnes interessant. Størstedelen av simuleringene ble gjort med 5 s lagringsintervall.

Første simulering som ble gjort var med de initielle parametrene FhSim blir ”levert” med. Det vil si med laksemodellparametre beskrevet i [1], fôrdistribusjonsmodellen beskrevet i [32] og [2], en vanlig merd med diameter på 50 m, 100 000 fisk og varierende lys- og temperaturforhold gjennom dagen. Bølger, strøm og vind er ikke tatt med. For å senke kompleksiteten ble miljøparametrene ”skrudd av”. Det som da ble gjort var å sette lys- og temperaturverdien i hele merden konstant og innefor lakseindividenes foretrukne område. Dette vil gjøre at fiskens atferd blir en funksjon av noen færre parametre. Videre ble fiskenes foretrukne avstand til hverandre endret fra 0.66 BL til 2 BL der BL betyr kroppslengder. Antallet fisk ble også variert mellom 10 000 og 100 000 fisk. De aller fleste simuleringer ble gjennomført med to føringsperioder, første fra 0900 til 1200 og andre fra 1500 til 1800. Merden som simuleres er helt ordinær, med form som en sylinder med dybde 17.5 m satt sammen med en kjegle med dybde 12.5 m. Den øverste sylinderen har diameter på 50 m.

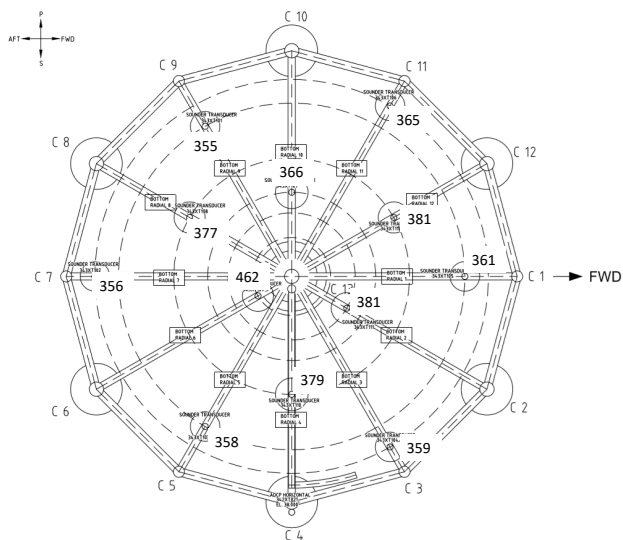
3.3 Ekkogrammer

I Ocean Farm 1 er det plassert ut 12 ekkolodd av typen Simrad EK80 med transducer Simrad ES70-7C. Denne varianten er av typen split beam, og disse produserer enorme mengder rådata. For å prosessere dette har Kongsberg Maritime laget MATLAB-programmer som leser inn rådata for så å sette disse sammen til data som kan brukes. Et eksempel på slike ekkogrammer kan sees i figur 3.1. Fra rådataen som produserte dette ekkogrammet kan det med MATLAB-funksjonen `datevec` leses at det første pinget kom 1. februar 2018 klokken 09:14:28 og det siste 20:34:01 samme dato. Ekkogrammene viser altså dybdefordelingen til fisk innenfor kjeglesnittet av merden som dette ekkoloddet dekker. Ekkoloddet er her plassert 30 m dyp pekende oppover, og overflaten kan sees relativt tydelig i ekkogrammet. I de reelle ekkogrammene som er vist er oppløsningen i tid i underkant av 1 s og oppløsning i z -retning 1 cm.

Ekkogrammene som kommer fra prosessert EK80-data fra OF1 er alle merket med *Serial No: xxx*. Dette serienummeret er hvert ekkolodd sitt ”navn”. I 2.4 vises det kun hvor ekkoloddene står, men dette spesifiseres med serienummeret i figur 3.2.



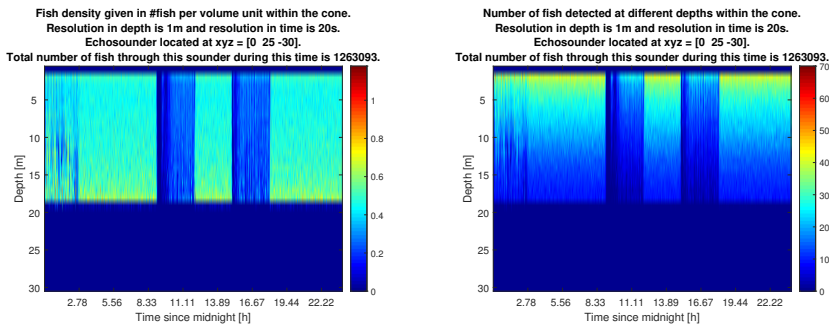
Figur 3.1: Reelt ekkogram fra OF1. Fargekodene antyder reflektert signalstyrke. Dette ekkogrammet viser et gitt punkt i merden fra tidlig morgen til sen ettermiddag/tidlig kveld. Serial No: 377 er ”navnet” på dette ekkoloddet.



Figur 3.2: Igjen kart over plasseringene til ekkoloddene i Ocean Farm 1, men merket med hvilket serienummer som hører til hvilket ekkolodd.

3.3.1 Simulerte ekkogrammer

For å kunne produsere noe som likner på de reelle ekkogrammene ble det først utviklet en metode for et simulert ”sonarping” som utnytter (3.1) og (3.2). MATLAB-koden for funksjonen kalt `sonarPing` finnes i vedlegg A.1.2, og er en funksjon som returnerer antallet fisk observert innenfor hvert intervall Δz , tettheten av fisk innefor hvert intervall og det totale antallet fisk dette ekkoloddet ser hvor hvert simulerte ping. Denne funksjonen brukes da for alle simulerte ekkolodd hvert tidssteg, før det repeteres for neste tidssteg. Ved å til slutt bruke den innebygde MATLAB-funksjonen `imagesc` sammenstilles dataene som er lest til et syntetisk ekkogram. I figur 3.3 illustreres det hvordan slike simulerte eller syntetiske ekkolodd kan se ut både med normalisering slik som i (3.2) og uten normalisering, altså at alle fisk i kjeglen telles og tas med i visualiseringen. Dette vil naturlig nok gi unaturlig høye observasjoner lengre unna der ekkoloddet er plassert. Med bakgrunn i dette vil metoden som kun teller antall fisk skrinlegges, og metoden som normaliserer med tanke på avstand bli benyttet.



(a) Eksempel på simulert ekkogram der observasjon av fisk normaliseres over observasjonsvolumet. Her fra et tilfeldig simulert scenario over et helt døgn.

(b) Eksempel på simulert ekkogram der antall fisk ikke er normalisert til volumet som observeres. Naturlig nok observeres flere fisk mot overflaten. Fra samme scenario som figuren til venstre.

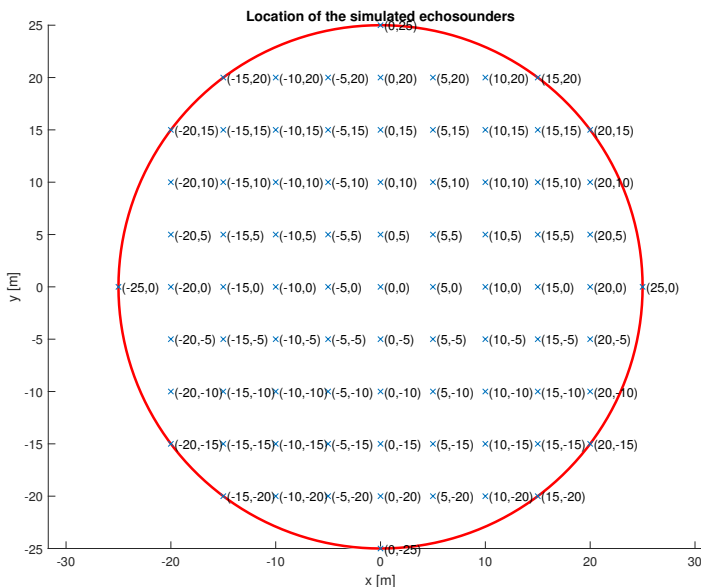
Figur 3.3: Samme simulerte scenario, men to forskjellige observasjonsmodeller. Til venstre brukes likning (3.2) til å gi en normalisert observasjon, mens til høyre brukes kun antall observert fisk innenfor kjeglen.

Når det gjelder plasseringen av ekkoloddene i bearbeidelsen av simuleringen ble de først plassert noe vilkårlig, før en mer systematisk tilnærming ble brukt. Det endte med at et ekkolodd ble plassert i en grid som vist i figur 3.4. Med åpningsvinkelen som er brukt i simuleringene og dybden ekkoloddene er plassert gir det en radius på overflaten på $30 \tan 10^\circ \approx 5.3$ m, og dermed dekker hele overflaten og store deler av den øverste halvdel av merden. Dette gjør også at det finnes et simulert ekkolodd for stort sett hele merden, og dermed gir et stort utvalg av simulerte målinger til sammenlikninger av reelle simuleringer, i tillegg til at dette gir et godt grunnlag for analyse av xy -fordelingen av fisk

i merden. Med den valgte åpningsvinkelen dekker da hvert ekkolodd omtrent

$$\begin{aligned} \frac{V_{\text{kjegle}}}{V_{\text{merd}}} &= \frac{\frac{1}{3}\pi r_{\text{kjegle}}^2 h}{\pi r_{\text{merd}}^2 h} \\ &= \frac{(30 \tan 10) ^2}{3 \cdot 25^2} \approx 0.015 \end{aligned}$$

av merden med antakelsen merd som en rett sylinder. Volumet er naturligvis noe mindre da bunnen er formet som en kjegle, men tar ikke med dette for enkelhetens skyld.



Figur 3.4: Plassering av simulerte ekkolodd.

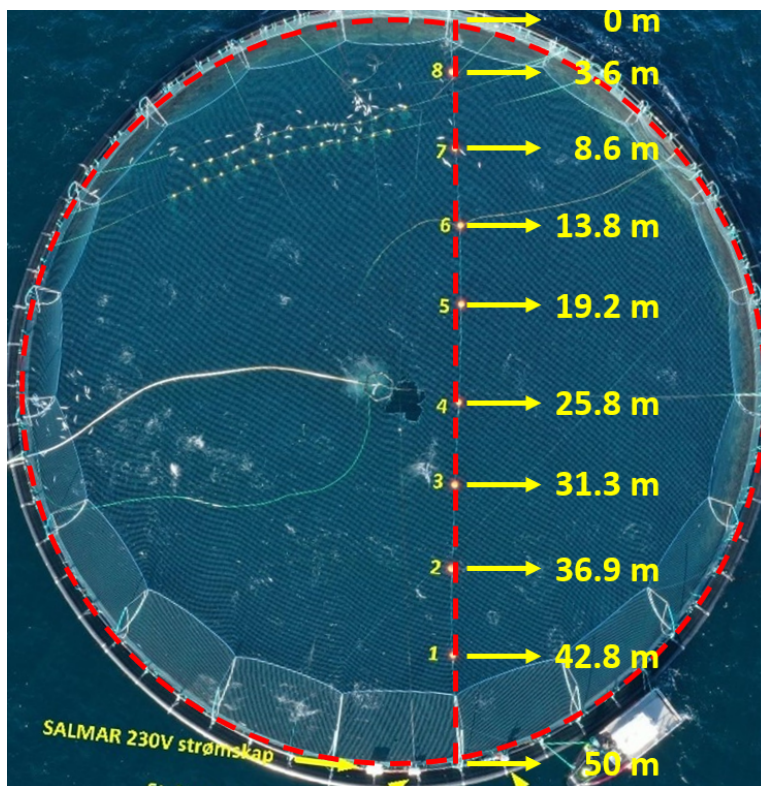
3.3.2 Tverrsnitt av merd

Et annet metode for sammenlikning med reelle data er å legge en linje med ekkolodd på tvers over en merd og sette sammen de observerte dataene til et tverrsnitt over hele merden. Dette er noe som det gjøres forsøk med hos SINTEF, men som det ikke er noe publisert forskning på enda. Med griden av ekkolodd som tidligere er definert kan man velge ut en linje og ha det som utgangspunkt, eller alternativt definere nye punkter. Fra disse ekkoloddene må dataene sammenstilles og interpoleres for å dekke hele tverrsnittet som skal observeres. Dette kan gjøres med den innebygde MATLAB-funksjonen `interp2(V, k)`. Med et grid av punkter som definerer en lavt oppløselig tetthetsfordeling i merden gir denne funksjonen et grid med $2^k - 1$ verdier mellom punktene i V . Koden for å generere

dette tverrsnittet finnes i vedlegg A.1.4.

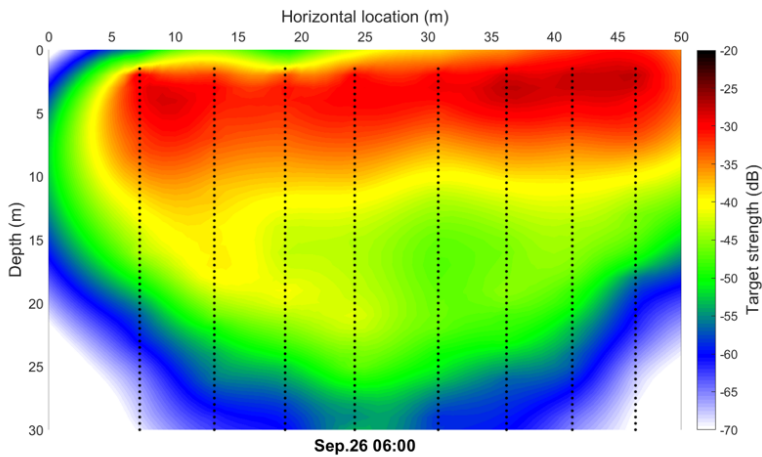
Oppsettet på hvordan dette er gjort kan sees i figur 3.5, og et eksempel på hvordan fordelingen kan se ut fra reelle ekkolodd i figur 3.6. Disse figurene er ikke hentet fra noe publisert forskning, men er et slags første resultat som det forskes videre på. I denne situasjonen ble det brukt EK15-ekkolodd, altså en enklere variant enn de avanserte EK80 som er brukt for eksempel i Ocean Farm 1. I dette forsøket ble også ekkoloddene plassert på overflaten med observasjonsretning nedover, i stedet for slik det er gjort på Ocean Farm 1, der de er plassert på undersiden med observasjonsretning oppover.

Det interpolerte ekkogrammet som er vist i figur 3.6 er i motsetning til ekkogrammene som er illustrert over ett enkelt øyeblikk i stedet for en tidsserie. Det vil si at tverrsnittet som er vist er et øyeblikksbilde av tilstanden til volumet som ekkoloddene dekker opp i stedet for utviklingen over tid. I figuren er merdens form tydelig, med bunnoten som spesielt tydelig.



Figur 3.5: Illustrasjonsbilde på hvordan oppsettet en snor med ekkolodd er satt ut på tvers av en merd. Figur er gjengitt fra[7].

For i dette tilfellet å kunne lage noe som er sammenliknbart med de reelle situasjonene brukes et oppsett med en linje med ni ekkolodd med lik avstand på tvers av merden. I



Figur 3.6: Interpolert ekkogram over et tverrsnitt av merden. Prikkete linjer angir hvor ekkoloddene har stått. Gjengitt fra [7].

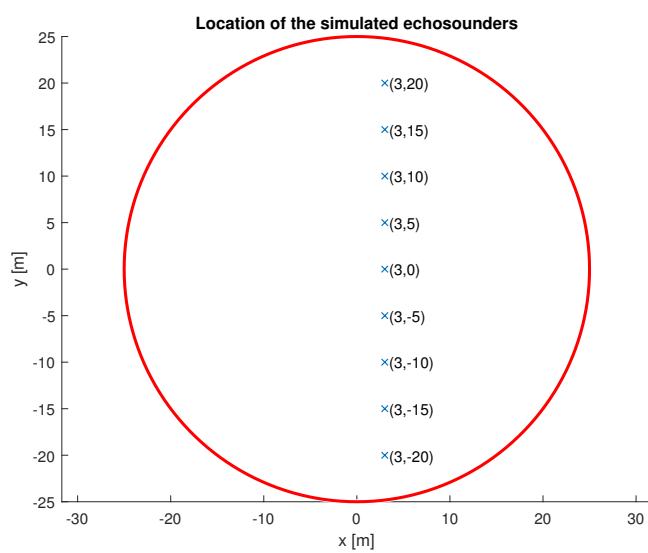
figur 3.5 kan det sees at linja ikke strekker seg over sentrum av merden, men er forskjøvet noe, derfor blir også dette tatt inn i simuleringen av tverrsnittet. De simulerte ekkoloddene er, som i det reelle scenariet, plassert på overflaten med observasjonsretning nedover. En illustrasjon av oppsettet kan sees i figur 3.7.

3.4 Heatmap

En annen metode for å kunne analysere xy -posisjonering for simulerte objekter over tid er å lage et "heatmap" av hvor det befinner seg fisk over tid. Ved hjelp av MATLAB-funksjonen `heatscatter` som er hentet fra [33]. Denne funksjonen er en utvidelse av den innebygde funksjonen `scatter` som lager et spredningsplott, der `heatscatter` i tillegg til å plote punkter farger punktene med bakgrunn i tetthet. Dette kan utnyttes både i et gitt øyeblikk eller akkumuleres over tid for å se hvor fisk plasserer seg i forskjellige scenarier.

3.5 Videogenerering

Til slutt ble det utviklet en metode for å lage video av fordelingen til fisk over et gitt spesifisert tidsintervall basert på enkeltfigurer. Koden som ligger til grunn for denne videogenereringen kan sees i vedlegg A.1.3.



Figur 3.7: Plassering på simulerte ekkolodd for observasjon av tverrsnitt.

Kapittel 4

Resultater

Dette kapittelet vil som navnet tilsier vise resultater fra metodene som er brukt. Det vil naturlig nok bli mange figurer, både av reelle ekkogrammer fra Ocean Farm 1, simulerte ekkogrammer og andre relevante figurer fra både reelle og simulerte scenarier.

4.1 Reelle ekkogrammer fra Ocean Farm 1

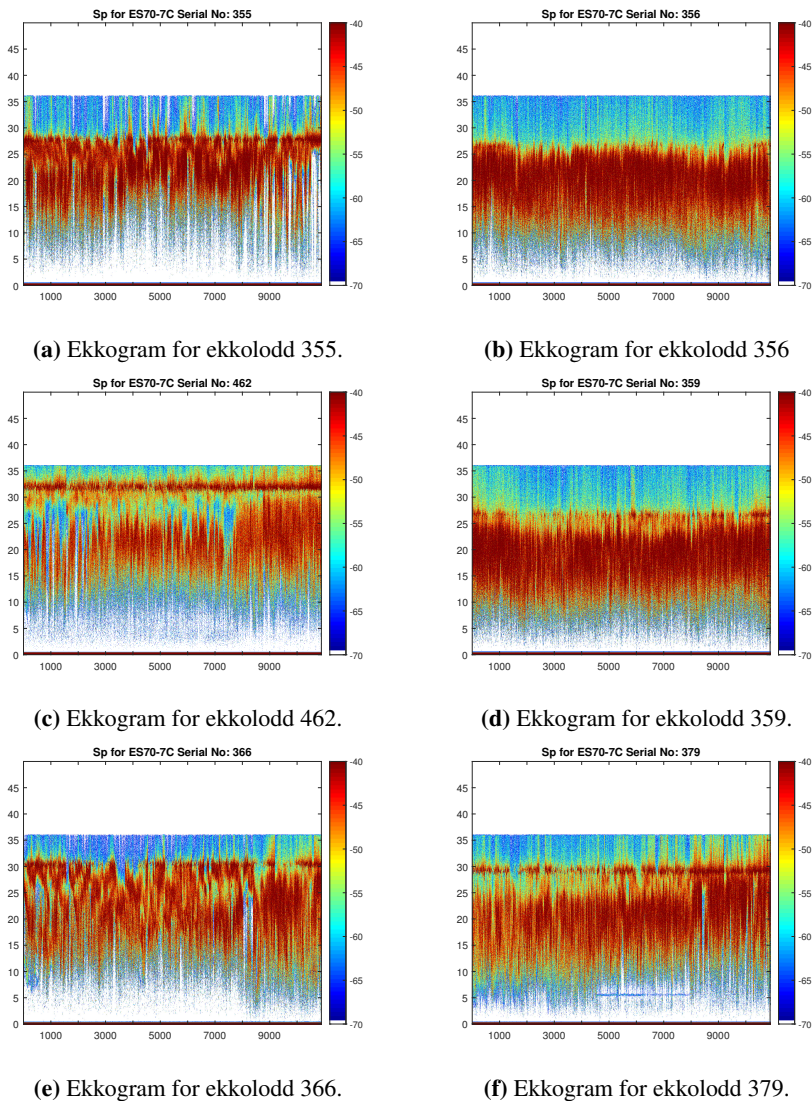
Et naturlig sted å starte med resultater er å presentere de reelle ekkogrammene fra Ocean Farm 1, slik at det kan dannes et sammenlikningsgrunnlag for de simulerte ekkogrammene som kommer senere. I figur 4.1 vises ekkogrammene til seks av de tolv ekkoloddene gjennom en hel dag. Hvor i Ocean Farm 1 disse ekkoloddene har stått kan leses fra figur 3.2.

4.2 Simulerte ekkogrammer

Fortsettelsen av det som er vist over blir å se på simulerte ekkogrammer og sammenlikne med de reelle ekkogrammene. I avsnittene som følger vil de resultater fra de forskjellige simuleringene presenteres og beskrives, for å kunne se hva som endrer seg i modellen når forskjellige parametre justeres.

4.2.1 Dybdefordeling

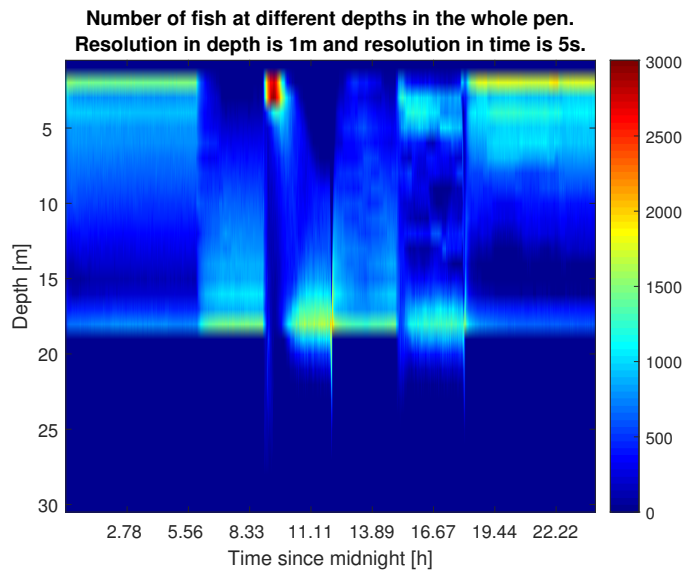
Aller først skal dybdefordelingen i hele merden sees på. I dette scenariet var alle standardparametrene initialisert, i tillegg til at miljøparametrene ikke var konstante, men lys- og temperaturforhold var en funksjon av tid og rom. I tillegg var det to fôringsperioder i løpet av dagen. Resultatet av dybdefordelingen for et døgn kan sees i figur 4.2. Her er det altså dybdefordelingen av fisk som en funksjon av tid som er beskrevet. Resultatet ligner, ikke helt overraskende, på figur 2.5, der vi kan se tydelig at fisken søker vekk fra lys på dagtid og mot overflaten nattetid. Det er også tydelig at fiskens respons på fôr er sterk, da dette



Figur 4.1: Ekkogrammer for forskjellige steder i Ocean Farm 1. Disse viser omtrent tolv timer en dag i februar 2018.

overstyrer lys- og temperaturresponsen. Dette er helt i tråd med hva som er beskrevet i (2.14), der responsen til fôr er vektet høyere enn lys og temperatur.

Med nøytrale lys- og temperaturforhold innenfor fiskenes foretrukne område ble fordelingen annerledes. Fiskene fordeler seg over en større del av vannmassen når det ikke skjer noe spesielt, men trekkes naturlig nok mot fôret når det er fôringstid. Resultatet kan sees



Figur 4.2: Dybdefordeling av all fisken i merden over en dag. Det er to fôringsperioder, første fra 0900 til 1200, siste fra 1500 til 1800.

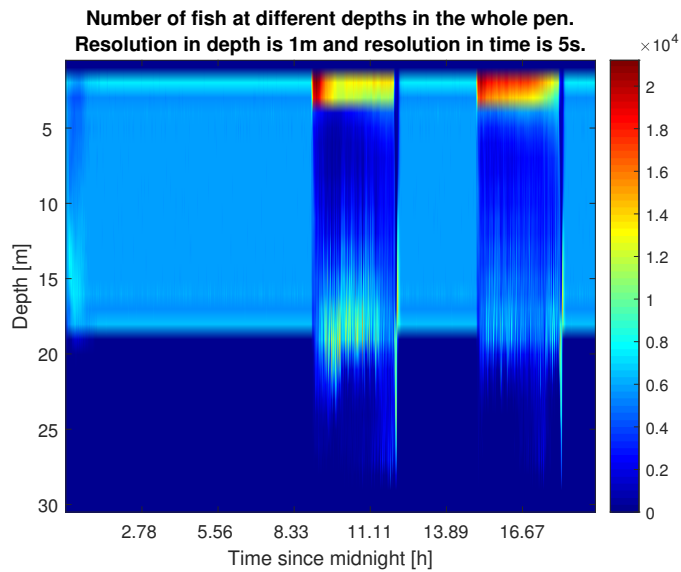
i figur 4.3. Selv om fisken trekker noe mer mot topp og bunn samles ikke alle på samme dybde, og det er heller ingen variasjon gjennom dagen. Dette er et godt tegn på at simulert fisk reagerer noenlunde slik en kan forvente fra et levende vesen. Her ble det simulert med 100 000 fisk, mens i figur 4.2 ble simuleringen kjørt med 10 000 fisk. Med 100 000 fisk og foretrukket avstand til nærmeste nabo justert opp fra 0.66 til 2 kroppslengder fordeler fisken seg som vist i figur 4.4, og utnytter større del av dybden. Siden det er respons til merdvegger som, naturlig nok, gir størst bidrag til atferdsmodellen, så vil fisken som er nær kanten av merden kun utnytte der merden er sylindrerformet, mens fisk nærmere sentrum har et større dybdespekter å bevege seg på. Med da avstandspreferanse justert opp vil fisken ikke søke like mye mot kanten, noe som vil vises i figurer senere.

4.2.2 Ekkogrammer uten miljøpåvirkning

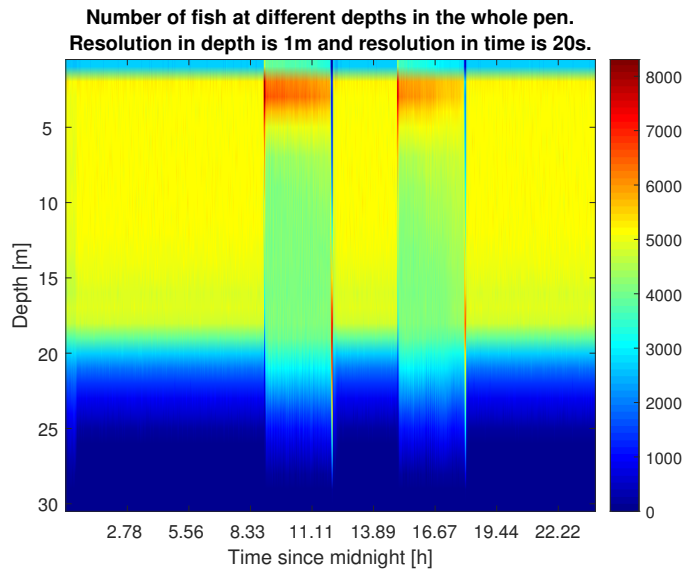
Med simuleringer der miljøpåvirkningen er ”skrudd av” og simulerte ekkolodd plassert i hele merden slik det er vist i figur 3.4. 81 ekkogrammer gir nok av figurer for hvert scenario, derfor følger her et utvalg av disse.

100 000 fisk og 0.66 kroppslengder minimum foretrukket avstand

Med 100 000 fisk, 0.66 BL minimum foretrukket avstand og to fôrperioder blir et utvalg av ekkogrammene vist i figur 4.5 - 4.11. I alle figurer står det i figurtittelen at ”total number of fish” er null, men det er bare fordi det ikke ble implementert fisketelling før ved senere simuleringer. Det området som ekkoloddet som gir figur 4.5 er det tydelig at den simulerte fisken synes er spennende, med enorme mengder fisk under fôring. Her ser det

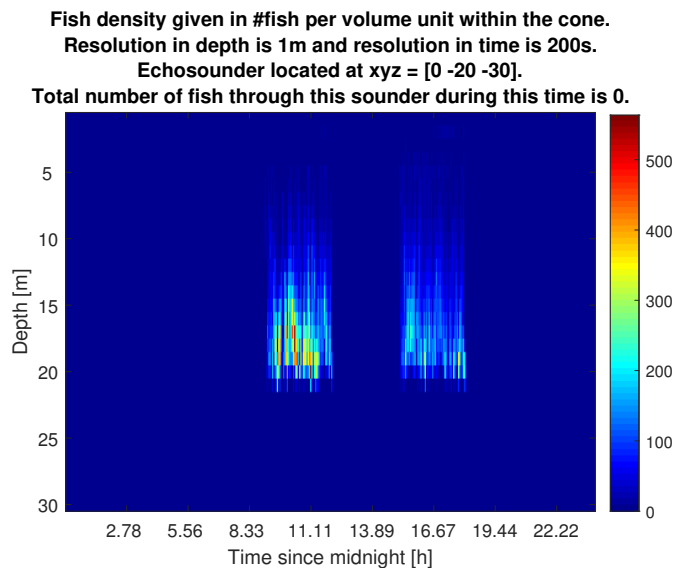


Figur 4.3: Dybdefordeling av fisk når miljøet er helt nøytralt. Fisken fordeler seg jevnere utover vannmassen. Simuleringen ble stoppet noe tidligere enn i figuren over, derfor viser den ikke fordelingen over et helt døgn.

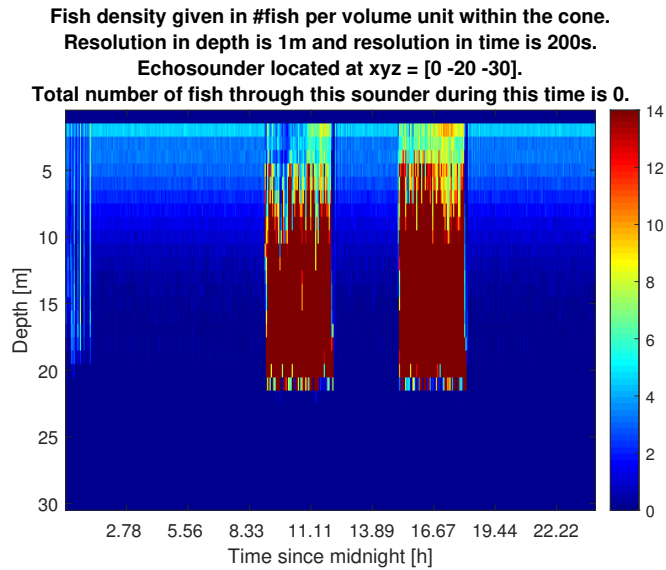


Figur 4.4: Dybdefordeling av fisk når miljøet er helt nøytralt og foretrukket avstand til naboene er justert opp til 2 BL.

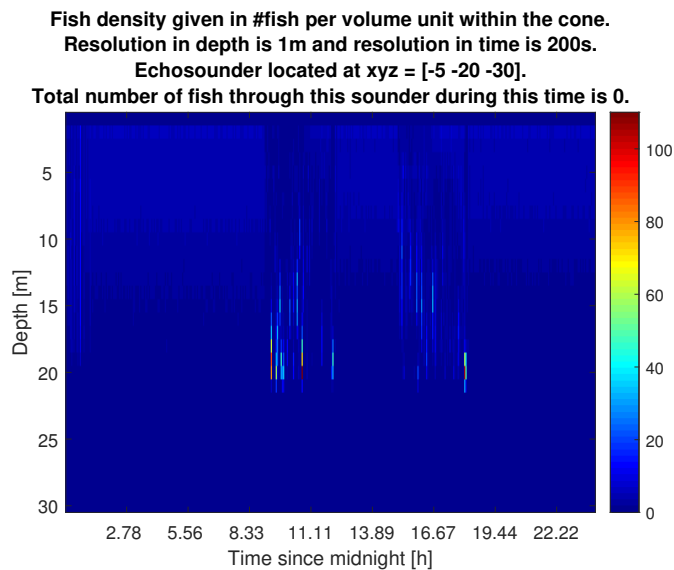
ut som det ikke finnes fisk i det hele tatt utenom fôrperiodene. I figur 4.6 vises derimot nøyaktig samme scenariet, men her er fargene for tetthet justert slik at alt over 14 fisk per observasjonsvolum gir samme farge. Det viser at fisken stimer rundt dette området også mens det ikke fôres, i motsetning til hva det ser ut som i figur 4.5. I figur 4.7 vises også naboekkoloddet, som viser veldig mye fisk korte perioder med fôring, men også jevnlig stiming når det ikke fôres. Uansett viser dette en voldsom samling av fisk i området rundt $x = 0$ m, $y = -20$ m. Speilet om x-aksen, altså rundt $y = 20$ m finnes ekkoloddet som gir 4.8. Her stimes det jevnt når det ikke fôres, men fisken forsvinner helt når det er tid for mat. Det samme sees også i figur 4.9 og 4.10, som henholdsvis viser ekkogrammer for områdene $x = 20$ m, $y = 0$ m og $x = -15$ m, $y = 20$ m. Ser man derimot til området rundt $x = 10$ m, $y = 5$ m er resultatet nærmest motsatt. Her er det en god del fisk i tiden det fôres, men helt tomt tiden det ikke fôres. Dette kan sees i figur 4.11.



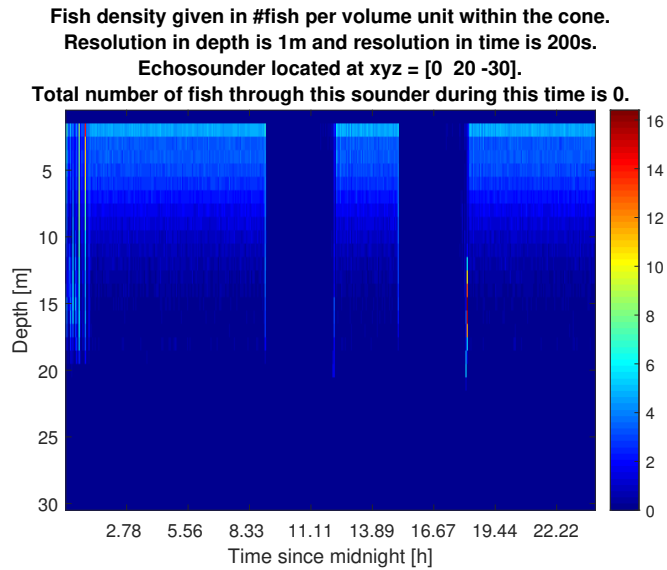
Figur 4.5: Simulert ekkogram med tidssteg på 200 s over et døgn. Under fôrperiodene er det ekstremt tett mellom fisken, men det kan se ut som fisken kun oppholder seg her når det fôres.



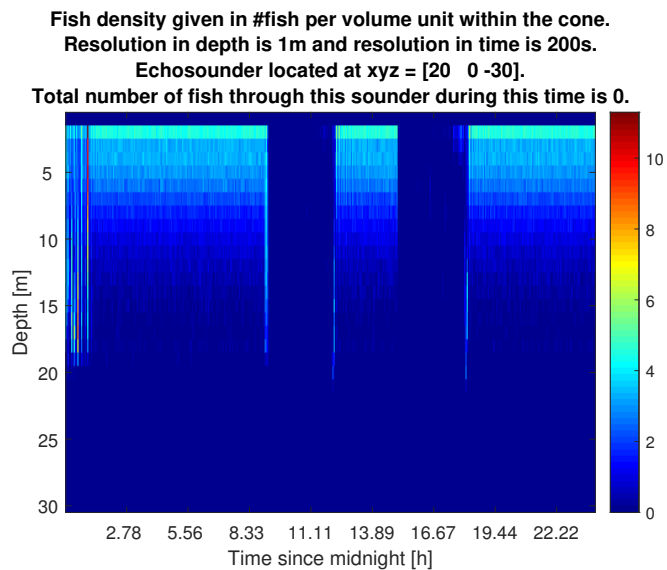
Figur 4.6: Samme scenario som i figur 4.5, men her er fargekodingen begrenset til 14 individer per rute. Altså vil alt over 14 gi samme farge. Dette viser at det er fisk her utenom fôringsperioder også.



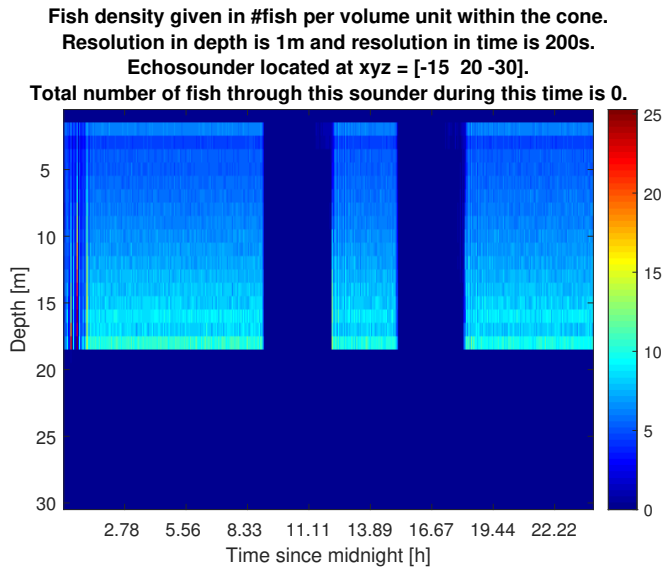
Figur 4.7: Simulert ekkogram med tidssteg på 200 s over et døgn. Her kan det også sees stor ansamling av fisk under fôring, med tidspunkter med ekstrem tetthet.



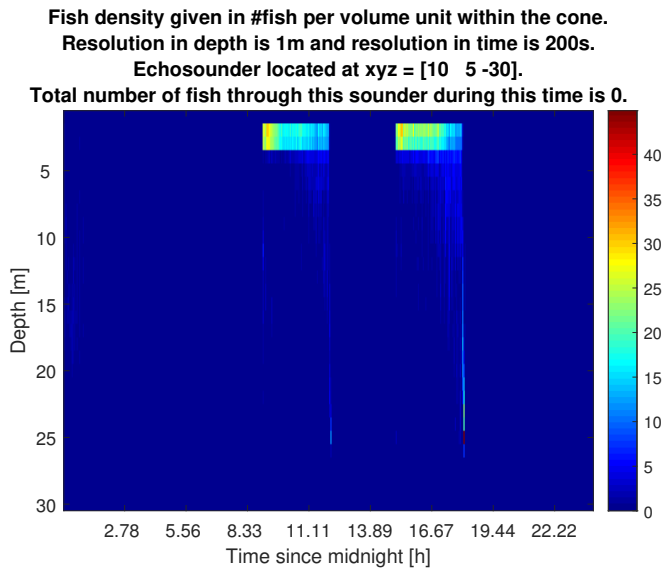
Figur 4.8: Simulert ekkogram med tidssteg på 200 s over et døgn. Fra dette området forsvinner fisken totalt under förperiodene, men holder seg jevnt utenom föring.



Figur 4.9: Simulert ekkogram med tidssteg på 200 s over et døgn. Her forsvinner fisken nærmest helt gjennom förperiodene, men holder seg stabilt utenom.



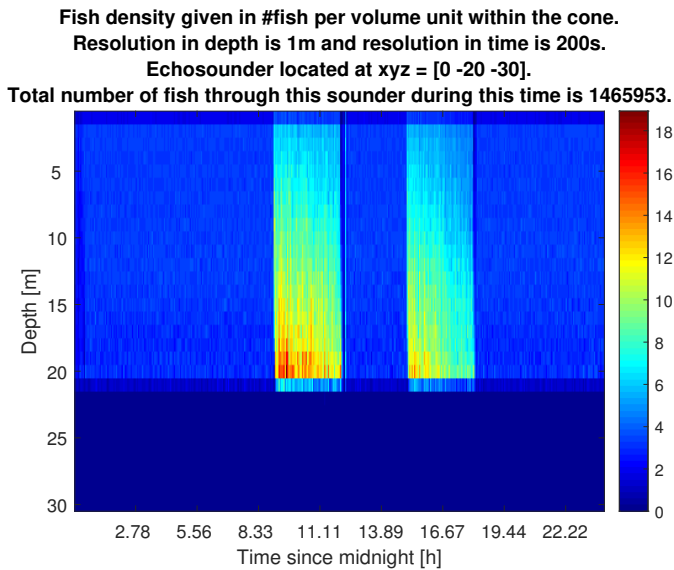
Figur 4.10: Simulert ekkogram med tidssteg på 200 s over et døgn. Det samme som i tidligere figurer kan observeres her, fiskemengden holder seg stabilt utenom førperiodene der mesteparten av fisken forsvinner fra området.



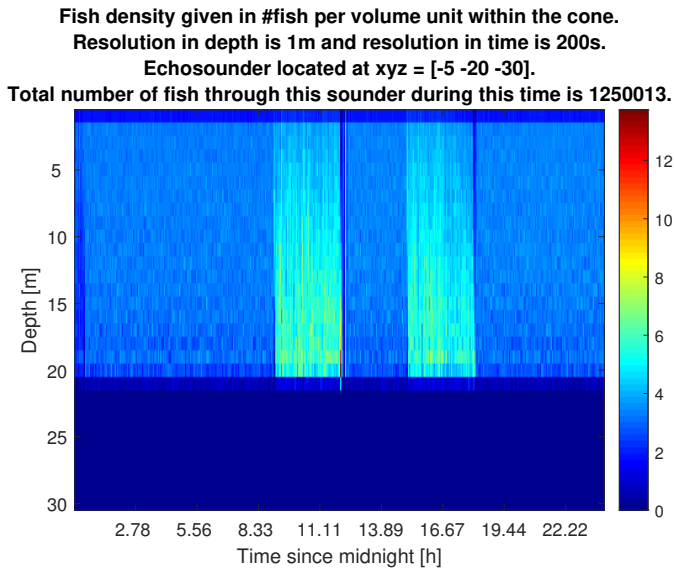
Figur 4.11: Simulert ekkogram med tidssteg på 200 s over et døgn. Her er ekkoloddet plassert noe nærmere sentrum av merden og det kan sees at utenom førperiodene er det null aktivitet i dette området.

100 000 fisk og 2 kroppslengder minimum foretrukket avstand

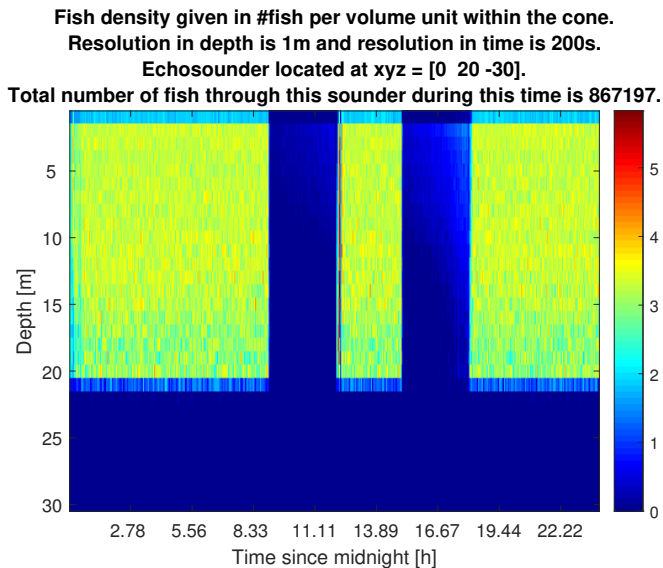
Med parameteren for minimum foretrukket avstand justert opp til 2 BL blir ekkogrammene annerledes. Her simuleres det nøyaktig det samme scenariet, altså like mange fisk, samme fôrregime, amme miljø og samme tidssteg mellom hvert simulerte sonarping. Det er langt ifra like tett mellom fisken, både når det ikke skjer noe spesielt og fisken bare stimer, men også når det føres. Der figur 4.5 hadde ekstreme mengder fisk på små områder under føring kan det i figur 4.12 sees at det er noe mer realistisk antall fisk observert når det er høy tetthet. Denne figuren viser nøyaktig samme sted i merden, $x = 0$ m, $y = -20$ m, akkurat like mange fisk, 100 000, og samme mengde fôr til samme tid. Det eneste som er endret er den minste foretrukne avstanden til andre fisk. Alle figurene som følger her, altså figur 4.12 til 4.17, er dermed direkte sammenliknbare med de i forrige avsnitt. De viser samme steder og samme scenarier, men forskjellig resultat kun på grunn av den ene justeringen som er gjort i simuleringen. Det er tydelig fra figurene at fiskene sprer seg over større område av merden, og ikke samler seg over et minimalt område helt mot ytterkanten av merden.



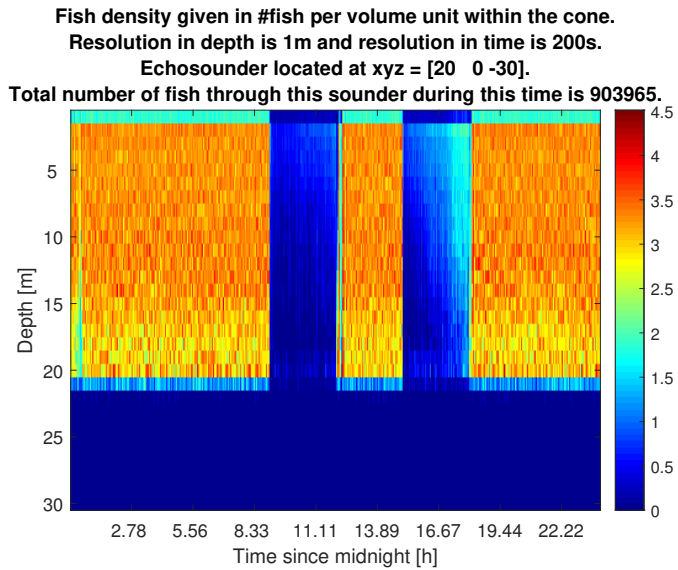
Figur 4.12: Simulert ekkogram med tidssteg på 200 s over et døgn. Under fôrperiodene er det tettere mellom fisken, og dette området er sammenliknet med andre steder i merden veldig populært under føring.



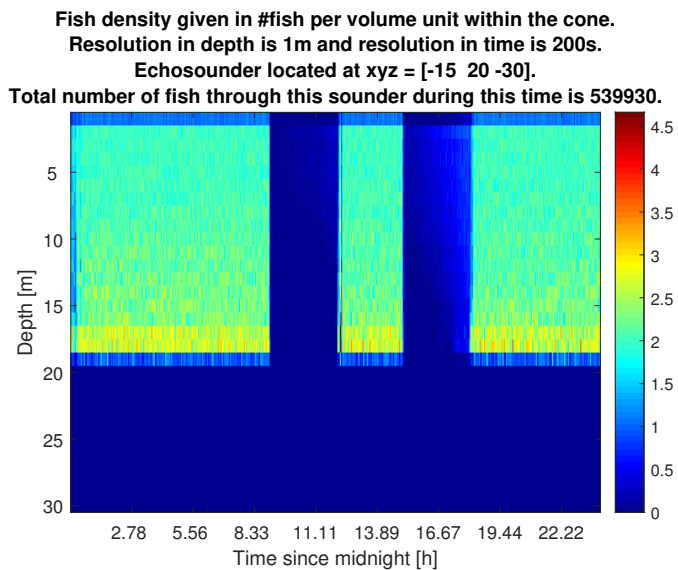
Figur 4.13: Simulert ekkogram med tidssteg på 200 s over et døgn. Her kan det også sees stor ansamling av fisk under føring.



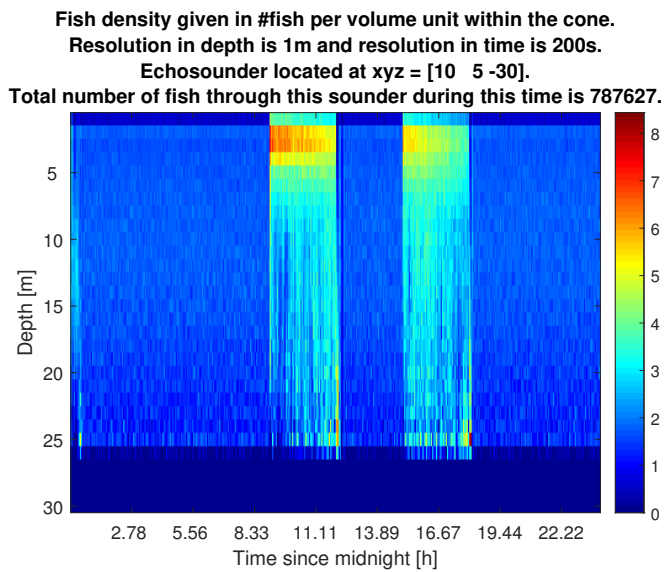
Figur 4.14: Simulert ekkogram med tidssteg på 200 s over et døgn. I dette området forsvinner meste-parten av fisken mens det føres, men det kan sees at noe fisk dukker opp mot slutten av begge førperiodene.



Figur 4.15: Simulert ekkogram med tidssteg på 200 s over et døgn. Her er det noe fisk gjennom føring, spesielt i andre førperiode.



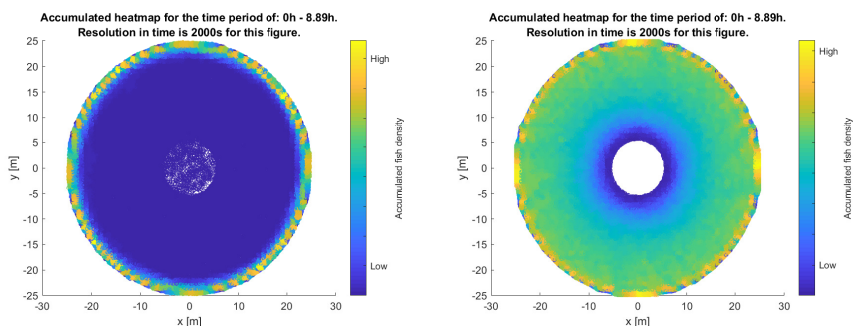
Figur 4.16: Simulert ekkogram med tidssteg på 200 s over et døgn. Her forsvinner også mye fisk mens det føres.



Figur 4.17: Simulert ekkogram med tidssteg på 200 s over et døgn. Her er ekkoloddet plassert noe nærmere sentrum av merden, men i motsetning til figur 4.11 som viser nøyaktig samme plassering er det aktivitet her også utenom fôrperiodene.

4.3 Horisontalfordeling med 100 000 fisk

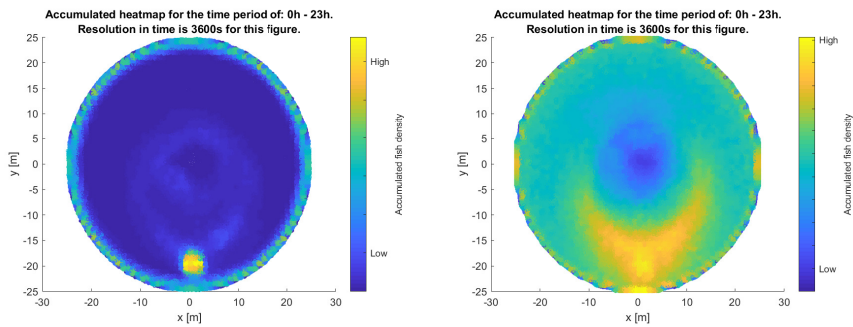
Med ekkogrammene fra forrige delkapittel friskt i minne er det naturlig å se på horisontalfordelingen av fisk i merden. Med utgangspunkt i de samme simuleringene som over, er det nå hvor fisken befinner seg i xy -planet som er interessant. For scenariet der fisken har en minimum foretrukket avstand på 0.66 BL blir den akkumulerte horisontalfordelingen over de første ni timene som vist i figur 4.18a. Her er det ingen fôrperioder og tidsintervallet mellom datasamlingen er 2000 s, altså omtrent 33 min. Figuren består altså av dataene fra hvert 2000 s i nesten ni timer. Dette gir totalt 1 600 000 punkter som danner grunnlaget for figuren. Det er tydelig at fisken trekker unaturlig mye ut mot kanten, og at den stimer rundt der om det ikke er noe annet som innvirker. At det i det hele tatt er fisk som befinner seg andre steder enn helt ut mot merdkanten er fordi ved initialisering av simuleringen er fisken fordelt jevnt rundt i merden, og det tar litt tid før simuleringen finner sin ”steady state”. Med minimum foretrukket avstand justert opp til to kroppslengder og resterende likt som over blir fordelingen de første ni timene som vist i figur 4.18b. Her utnytter fisken mye større del av arealet når simuleringen finner steady state, men det er fremdeles tydelig at fisken ønsker seg utover mot kanten og ikke inn mot midten.



- (a) Fordeling av fisk over ni timer med foretrukket avstand til nærmeste nabo minst 0.66 kroppslengder. Tidssteg her er 2000 s og det er ingen fôring under denne simuleringen.
- (b) Fordeling av fisk over nesten ni timer med foretrukket avstand til nærmeste nabo minst to kroppslengder. Tidssteg her er 2000 s og det er ingen fôring under denne simuleringen.

Figur 4.18: Heatmap av fiskens fordeling i xy -planet med to forskjellige konfigurasjoner. Begge figurer viser de første ni timene med tidssteg på 2000 s.

Når akkumuleringen strekker seg gjennom et helt døgn med to fôrperioder og datainnsamling hver time blir den akkumulerte horisontalfordelingen av fisken som vist i figur 4.19a. Det er tydelig at fisken samler seg tett i et spesifikt område når det føres, som det kunne sees i de foregående delkapitlene med ekkogrammer. I dette scenariet var avstandsprefransene skrudd ned til 0.66 BL. Den samme ansamlingen kan også sees når avstandsprefransen skrues opp til 2 BL. Her samles mye fisk i et lite område, men ikke like ekstremt som når avstandsprefransen er lavere. Dette kan sees i figur 4.19b. Fisken samler seg, men det er mindre forskjell fra resten av merden og ansamlingen er heller ikke like konsentrert som tidligere. Det er altså ikke like tett mellom fisken, men de dekker et større område.



(a) Heatmap av xy -fordelingen av fisk gjennom 23 timer med foretrukket avstand til nærmeste nabo minst 0.66 kroppslengder. Her er det tatt et øyeblikksbilde hver time og sammenstilt dette til en fordeling gjennom dagen.

(b) Heatmap av xy -fordelingen av fisk gjennom 23 timer med foretrukket avstand til nærmeste nabo minst 2 kroppslengder. Her er det tatt et øyeblikksbilde hver time og sammenstilt dette til en fordeling gjennom dagen.

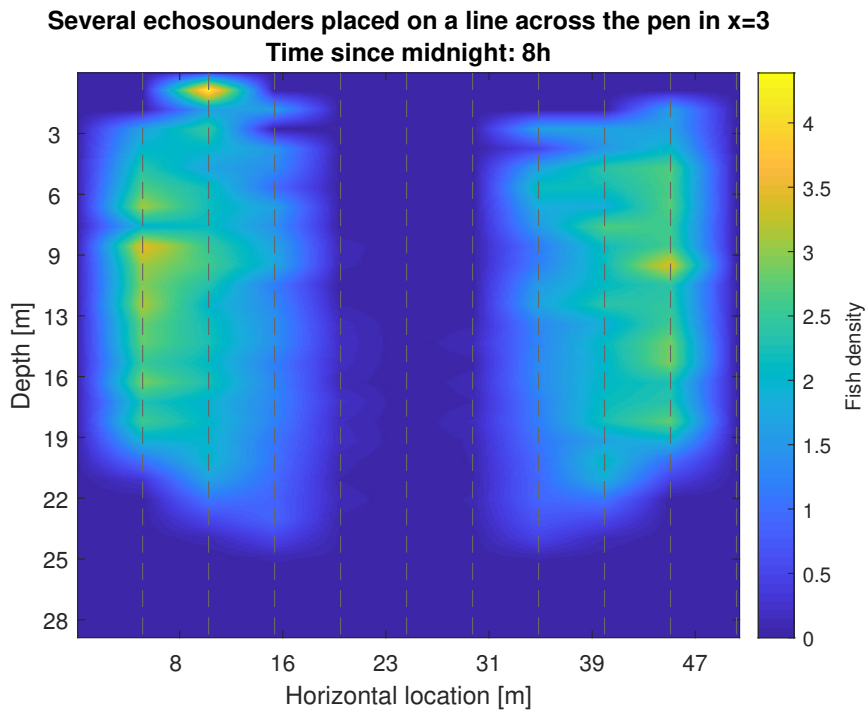
Figur 4.19: Heatmap av fiskens fordeling i xy -planet med to forskjellige konfigurasjoner. Begge figurer viser 23 timer med tidssteg på 3600 s og to førperioder.

4.3.1 Tverrsnitt

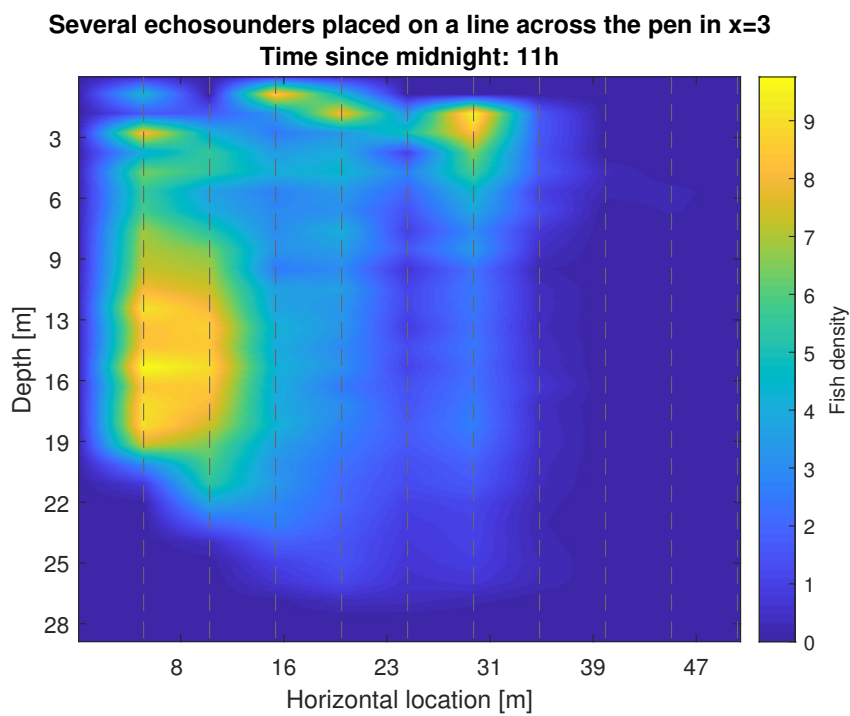
En annen måte å se på hvordan fisken romlig fordeler seg er å se på et tverrsnitt av merden. Slik et oppsett ble vist i kapittel 3.3.2 gir dette et interpolert tverrsnitt av merden som vist i figur 4.20 og 4.21. Figurene viser henholdsvis tilstanden i merden etter 8 og 11 timer fra midnatt, altså når tilstanden i merden er i steady state og fisken stimer og når det har blitt føret i 2 timer. I figuren som viser tilstanden ved steady state, er formen på merden tydelig med kjegleformet bunnot under en sylinder, slik en også kan se i 3.5. I dette simulerte tilfellet er det som vist i ekkogrammer og heatmap lite fisk i sentrum av merden til forskjell fra det som ble observert i 3.5. Etter elleve timer er det tydelig at fisken trekker mot områder med mye fôr, et resultat som er helt som forventet med tanke på hva andre simulerte ekkogrammer og heatmaps har vist.

4.3.2 Video

Som et supplement til alle ekkogrammer og heatmaps av fordelingen til fisken er det laget videoer som viser fordelingen til fisken gjennom et døgn, både med lav og høy foretrukket avstand til naboene. Disse videoene har et tidssteg på 30 s og kjører 10 tidssteg hvert sekund. Det resulterer i videoer på 4 min 46 s som viser hvordan fisken beveger seg i merden gjennom et døgn med to førperioder. Videoene legges ved rapporten.



Figur 4.20: Tversnitt av merden åtte timer etter midnatt.



Figur 4.21: Tverrsnitt av merden etter elleve timer, altså to timer inn i første fôrperiode.

Diskusjon og konklusjon

5.1 Diskusjon

5.1.1 Ekkogrammer og fiskens atferd

Ekkogrammene fra Ocean Farm 1 har dannet grunnlaget for de simulerte ekkogrammene som har blitt vist gjennom denne oppgaven. For å kunne lage disse var det en del som måtte på plass, blant annet transformasjonen fra kjegleform til én dimensjon. Hvordan dette blir gjort i et ekkolodd slik som EK80 er vanskelig å oppdrive, men i denne rapporten ble det antatt at styrken på det reflekterte signalet må skaleres proporsjonalt med volumet som ekkoloddbeamen dekker. Altså for lange avstander dekker ekkoloddbeamen et større område enn for korte avstander. Siden metoden de simulerte ekkogrammene fremstilles ikke bruker reflektert signal, men heller teller antall simulerte fisk nøyaktig, ble metoden for å gjøre om fra kjeglesnitt til én dimensjon å telle alle fisk innenfor et gitt volum for så å dele på størrelsen av det volumet. Dette er metoden som virker mest logisk når ingen ting annet er oppgitt. Videre ble det også illustrert forskjellen på simulerte ekkogrammer med og uten denne skaleringen. Om en bare teller antall fisk for så å kollapse det ned til én dimensjon vil det i ekkogrammet se ut som det er veldig mye fisk høyt i vannmassen, mens nedover er veldig lite, noe som ved kvalitative observasjoner av merden viser seg å ikke stemme. Derfor kan det argumenteres for at metoden som er brukt for skalering gir et godt estimat av hvor mange fisk som befinner seg på forskjellige dyp i ekkoloddbeamen.

En annen ting som skiller de simulerte ekkogrammene fra de reelle er støy. Det som vises i et simulert ekkogram er en presis måling av tilstanden innenfor observasjonsvolumet. Det er ikke lagt inn noen feilkilder, slik en ser i reelle anlegg. Der kan det være støy i målingene, fremmedlegemer i veien, bølger som gjør overflaten ujevn eller andre kilder til upresise målinger. Dette er i alle fall noe som må tas høyde for dersom de reelle målingene skal kunne sammenliknes med målingene fra simuleringen. Videre må det også finnes en metode for sammenlikning av reflektert signalstyrke og tetthet av fisk på forskjellige avstander dersom simulerte og reelle ekkogrammer skal kunne sammenliknes.

En annen ting som er verdt å tenke på er hva slags informasjon en egentlig får fra et ekkogram. Et ekkogram er et plot som viser hvor mye som befinner seg innenfor en sonarbeam som en funksjon av tid, altså forteller et enkeltplott bare om det befinner seg fisk innenfor utsnittet et ekkolodd dekker og eventuelt hvor mye. Et ekkolodd som er tomt vil altså gi ingen annen nyttig informasjon enn at fisken ikke befinner seg der ekkoloddet er. Med tanke på hvor lite ett enkelt ekkolodd dekker av merden vil dette praktisk talt gi null informasjon. I ekkogrammene som er presentert tidligere er det perioder av dagen der det ikke er spesielt tett med fisk, som da vil si perioder der dette ekkoloddet gir lite informasjon. På andre tidspunkt eller andre steder i merden er det ekkolodd som gir mye informasjon.

Om en ser på hvor det er fisken samler seg antyder alle simuleringene at fisken trekker mot samme sted, det er bare tettheten som varierer. Dette kan settes i sammenheng med distribusjonen av fôr i modellen. Det er tydelig modellert slik at i en retning vil det samles mer fôr enn i den andre retningen, noe det er naturlig at fisken følger. Responsen mot fôr er veldig sterk, og det er naturlig at fisken tar sikte mot områdene der er mest fôr. Om en sammenlikner figur 2.7, 2.8 og 2.9 med figur 4.19a og 4.19b er mønstrene til å kjenne igjen. Aller mest gjelder dette for figur 2.9 og 4.19b.

Når det gjelder fiskens atferd er det interessant å sammenlikne simulert fisk med data fra fisk merket med hydroakustiske sendere. I figur 2.10 viser data fra fire enkeltindivider i et stort anlegg, så datagrunnlaget er snevert, men likevel er sammenlikningen interessant. De simulerte fiskene ser ut til å unngå sentrum av merden med mindre det skjer noe spennende der, slik som fôring, mens flere av de merkede fiskene tilbringer mye tid i sentrum. Dette er selvsagt veldig individuelt for enkeltindividene, slik det kan sees i figur 2.10c, der en fisk ser ut til å trives veldig godt på et lite område. Uansett viser ingen av disse fiskene den samme tendensen til å stime i ytterkant når det ikke skjer noe, men heller å oppholde seg litt rundt omkring i merden. Dette kan selvsagt ha noe med fôringsregime å gjøre, en parameter fra den reelle casen som ikke er gitt. Det som imidlertid er gitt er at fôring har foregått sentralt slik som i simuleringene. Likevel kan det i figur 4.1a, 4.1d og 4.1c til en viss grad observeres at fisken trekker mot kanten i Ocean Farm 1. Figur 4.1a og 4.1d viser begge ekkolodd som er plassert mot kanten på hver sin side av den enorme merden, mens 4.1c viser et ekkolodd mot sentrum. Fra ekkogrammene kan det her se ut som det er mer fisk ut mot kanten, og at når de forsvinner fra kanten, slik som mot slutten av figur 4.1a så dukker de opp i figur 4.1c. Ekkogrammene fra Ocean Farm 1 viser riktignok bare en halv dag, slik at det er vanskelig å si noe sikkert, men tendenser som observeres i modellen kan også til en viss grad observeres i et reelt anlegg.

Med sammenlikningen av fiskens atferd i reelle anlegg og simulert i bakhodet er det naturlig å se på om modellen gir en tilstrekkelig realistisk romlig distribusjon av fisk til å kunne brukes i en observerstruktur. Med de initiale parametrene som ble gitt er svaret kategorisk nei. Fisken trekkes for tett sammen ut mot kanten, noe som er helt urealistisk. Med en enkel justering av den minimale foretrukne avstanden til nærmeste nabo justert fra 0.66 til 2 BL kan ikke dette avvises like kategorisk, men det virker fremdeles som fisken unngår sentrum urealistisk mye. Nå ble disse simuleringene utført med 100 000 fisk, og

en simulering med 200 000 ville kanskje gitt en annen fordeling, men med det utleverte utstyret ville både simuleringen av scenariet og analysen krevet mer regnekraft enn det som var tilgjengelig ved utførelsen av arbeidet. Likevel kan det argumenteres for at andre parametre må justeres slik at fiskens atferd i simuleringen ikke blir like deterministisk. Med enkle visuelle simuleringer kunne det observeres at fisken begynner å stime ”kun” som funksjon av å ikke kolliderer med hverandre. Dette gjør at i simuleringen finner fisken sirkelmønstrene de følger, så blir de der. En simulering er nødvendigvis en forenkling av virkeligheten, men kanskje er den for enkel i dette tilfellet. Biologi er et komplekst fagfelt, og enkeltindivider kan ha større forskjeller enn antatt. I modellen er det også et svært høyt antall parametre som kan justeres, og justere disse og se på forskjeller for å finne en optimal konfigurasjon kunne kanskje vært en doktorgrad alene. Men om modellen gir en riktig romlig fordeling eller ei, kan det uansett konkluderes med at responsen i z -retning mot lys og temperatur er god.

Om man ser på tverrsnittet av en reell merd slik det er vist i figur 3.6 og sammenlikner med figur 4.20, er det også her tydelig å se at den romlige fordelingen i modellen ikke stemmer helt, da det ikke befinner seg noe fisk mot sentrum av merden. Her er ekkolodden plassert i overflaten, og dermed blir det også gjort i det simulerte ekkogrammet. I figuren fra den reelle merden kan det observeres at mye fisk holder seg høyt i vannmassen, men at noe trekker nedover. Dette resulterer i at formen på merdens bunnot er tydelig, noe som også kan sees i det simulerte tverrsnittet.

5.1.2 Ekkolodd som observer

Å fange dynamikken i til en merd, både reelt og simulert, gjennom å observere ekkogrammer er en utfordrende oppgave. For en simulert modell som til syvende og sist er ganske deterministisk virker dette overkommelig. Med utgangspunkt i ekkogrammene som er presentert i resultatkapittelet, da spesielt de som gir den mest sannsynlige fordelingen, er det tydelig å se at når fisken forsvinner fra et område dukker den opp igjen et annet sted, slik som under forperiodene. Utenom forperioden er det i modellen enda enklere å fange dynamikken, da fisken stort sett ender opp med å stime rundt, og en kan observere stort sett det samme i alle ekkolodd, men noe høyere tetthet ut mot kanten. Ekkogrammene viser riktignok fordeling over en hel dag, og ekkogrammer for kortere perioder vil gi større usikkerhet i den helhetlige tilstanden i merden.

Som en sammenlikning til det forrige avsnittet kan en se på ekkogrammene fra Ocean Farm 1. Det er ikke enkelt å si noe om hvordan tilstanden i merden endrer seg ut ifra de seks ekkogrammene som er vist i figur 4.1. Det kan argumenteres for at det forsvinner fisk fra området til ekkolodd 355 som kanskje dukker opp igjen i området til ekkolodd 462, men fra resten av ekkogrammene er det vanskelig å si noe mer. Det befinner seg fisk i områdene ekkoloddene dekker, men kanskje er dette fisk som står stille. Kanskje er dette fisk som flytter seg. Innledningsvis ble det også beregnet at 12 ekkolodd dekker opp omlag 0.5% av det totale volumet i Ocean Farm 1, noe som betyr at hvert enkelt ekkolodd dekker mindre enn en halv promille av det totale volumet. Sammenliknet med de simulerte ekkoloddene som dekker omlag 1.5% av en vanlig merd alene, selv om disse kanskje har

en kunstig høy åpningsvinkel, dekker ekkoloddene i Ocean Farm 1 et veldig lite område totalt sett. Naturlig nok klarer de simulerte ekkoloddene å fange opp dynamikken i merden ganske bra med få observasjonsenheter, mens en merd på størrelse med Ocean Farm 1 vil trenge vesentlig flere. I tillegg er det mer usikkert hvordan fisken oppfører seg når områdene de kan bevege seg på blir så store som i OF1. Dette er et interessant område å både observere i reelle caser og i simulering.

Plasseringen av ekkoloddene er også en viktig faktor for hvordan en klarer å fange opp dynamikken. Som det kan sees xy -fordelingen i figur 4.18b og i telemetridataen fra figur 2.10 er det ikke symmetrisk hvor fisken beveger seg. Dette kan for eksempel være fordi føret sprer seg ujevnt eller at fisken har foretrukne oppholdssteder. Likefullt betyr det at posisjonen til et ekkolodd er viktig for observerbarheten i systemet. Et nøkkelpunkt kan være der fisken samles under føring. Her vil man kunne observere apetitten til fisken for eksempel gjennom hvor mange fisk som befinner seg i områdene med mye før. Videre vil plassering med varierende avstand til sentrum være fordelaktig for å kunne ta høyde for å kunne observere fisken også når den ikke stimer langs kanten. For simuleringen gjorde all fisk dette om det ikke skjedde noe annet, men det er usannsynlig at det er en presis gjengivelse av virkeligheten. I simuleringen der avsatnspreferansene var justert kunne det også observeres fisk med ekkolodd nærmere sentrum. Slik oppsettet er i Ocean Farm 1 virker som en fornuftig variasjon, selv om denne studien ikke har grunnlag for å si dette sikkert. Dataene fra ekkoloddene i Ocean Farm 1 er kun benyttet som en referanse, og er ikke analysert for å se på dynamikken i anlegget. Det som hadde gitt fordelaktige resultater var om det kunne vises at atferdsmønstrene til grupper av fisk hadde gitt noen slags form for symmetri, da dette hadde gjort det lettere å interpolere opp eller aggregere tilstander. Resultatene fra telemetriske observasjoner viser imidlertid at det er svært varierende fra individ til individ. For å kunne si noe mer om dette må det gjøres forsøk i større skala, at flere fisk merkes, for å få et bredere datagrunnlag for hvordan grupper av fisk beveger seg. Dette krever store arbeidsmengder, men kunne gitt viktig læring.

5.2 Oppsummering og konklusjon

I løpet av denne oppgaven ble det først sett på hva dataassimilering er, hvilke metoder som kan byttes for dette og hvordan det er utført innenfor andre fagfelt. Det viste seg at for punktmålinger med distribuerte modeller var Ensemble Kalmanfilter en metode som gir gode resultater og er relativt gjennomførbar å implementere for slike type problemer. Det ble videre sett på ekkoloddet Simrad EK80 med tilhørende Simrad ES70-7C 70 kHz transducer, egenskapene til disse, i tillegg til et kjapt blick på hvordan disse er montert på Ocean Farm 1. Litteraturstudiet ble avsluttet med et dykk inn i hvordan fiskeatferd er modellert i FhSim, modellen som ble brukt i videre arbeid, i tillegg til å se på observert fiskeatferd.

Videre ble det utviklet målemodeller for å kunne simulere ekkolodd fra fiskeatferdsmodellen. Disse ble satt sammen til forskjellige ekkogrammer som viste fordeling av fisk på flere forskjellige måter for å gi et grunnlag for videre analyse. Deretter ble det simulert

mange scenarier, og dataene herfra ble analysert. Det ble vist at målemodellen gir gode ekkogrammer som observerer en liten simulert merd på en grei måte, men at dataene fra den reelle merden er vanskelig å analysere presist uten flere målepunkter eller forståelse for fiskens atferd.

Om modellen kan det konkluderes med at responsen i z -retning stemmer godt overens med fiskens naturlige respons. Dette henger i hovedsak sammen med lys- og temperaturrespons, noe som tyder på at denne biten av atferdsmodelleringen er presis og god. Der modellen skiller seg mer fra observert fisk er i xy -planet, der fisken i for stor grad søker ut mot kanten. Modellen ble en god del bedre med å justere paramteren for avstand til naboen, men likevel stemmer det ikke helt med observerte data. En bedre modell må på plass før en i det hele tatt kan tenke på å bruke dette i en assimilasjonsprosess.

Med ekkolodd som observator ble det vist at i simulering med standard merd, men noe høy åpningsvinkel i ekkoloddet, fungerte ekkoloddet godt som observer. Det var intuitivt å forstå hvor fisken og hvordan den beveget seg under forholdene gjennom å se på ekkogrammene. Dette var på langt nær like tydelig i ekkogrammene fra Ocean Farm 1. Disse dekker hver et mye mindre område av merden, og det er dermed vanskeligere å få nyttig informasjon fra disse. Konklusjonen får være at ekkolodd som observer er spennende og kan gi nyttig informasjon, men modellen er enda ikke god nok til å gjengi virkeligheten. Når dette er på plass kan et Ensemble Kalmanfilter være veien å gå.

5.3 Videre arbeid

Neste steg i denne prosessen vil være å få mer data på hvordan fisk faktisk beveger seg, for så å justere flere parametre i modellen slik at dette stemmer bedre med virkeligheten. Uten å ha dette på plass vil ikke modellen klare å gjengi virkeligheten og er dermed ubrukelig å ha med i en observerstruktur. Videre trengs det også en matematisk modell som "oversetter" et ekkogram fra virkeligheten, altså reflektert signalstyrke, til antall fisk som i modellen. Uten denne konverteringen blir det litt som å sammenlikne epler og pærer.

Ved siden av dette går det an å implementere en tvillingsimulering som bruker måledata fra en "simulert virkelighet" og assimilerer dette inn i en annen simulering med perturberte parametre ved hjelp av for eksempel et Ensemble Kalmanfilter. På denne måten kan strukturen med ekkolodd som observer testes for å se om det har noe for seg, og det vil gi svar på hvor mange ekkolodd som trengs for å gi et godt estimat gjennom filteret.

En siste ting som kunne vært interessant å se på, men som ble begrenset av regnekraft i denne oppgaven, er å simulere en merd i på størrelse med Ocean Farm 1 og realistisk miljø og antall fisk, for så å plassere ekkolodd i de faktiske posisjonene det er ekkolodd i det reelle anlegget, for så å sammenlikne direkte med de reelle ekkogrammene fra anlegget. Dette vil kunne gi et nyttig hjelpemiddel for å validere modellen i scenarier modellen ikke er testet, og se om atferden til de simulerte fiskene endrer seg når merden blir større. Her finnes observasjonsdataene, men simuleringen med ekkoloddmodellen må utføres. Dette krever regnekraft.

Bibliografi

- [1] Martin Føre, Tim Dempster, Jo Arve Alfredsen, Vegar Johansen, and David Johansson. Modelling of Atlantic salmon (*Salmo salar* L.) behaviour in sea-cages: A Lagrangian approach. *Aquaculture*, 288(3):196 – 204, 2009.
- [2] Morten O Alver, Kristoffer Rist Skøien, Martin Føre, Turid Synnøve Aas, Maike Oehme, and Jo Arve Alfredsen. Modelling of surface and 3d pellet distribution in atlantic salmon (*salmo salar* l.) cages. *Aquacultural engineering*, 72:20–29, 2016.
- [3] NA Thacker and AJ Lacey. Tutorial: The kalman filter. *Imaging Science and Biomedical Engineering Division, Medical School, University of Manchester*, page 61, 1998.
- [4] Simrad. Simrad ES70-7C Data sheet. https://www.simrad.online/td_070/es70_7c/es70_7c_ds_english_a4.pdf, 2006. Lastet ned: 24.10.19.
- [5] Maike Oehme, Turid Synnøve Aas, Mette Sørensen, Ingolf Lygren, and Torbjørn Åsgård. Feed pellet distribution in a sea cage using pneumatic feeding system with rotor spreader. *Aquacultural engineering*, 51:44–52, 2012.
- [6] Per Arne Kjelsvik. Internet of fish-real-time monitoring of fish through lpwan and internet technologies. Master’s thesis, NTNU, 2019.
- [7] SINTEF. unpublished. unpublished.
- [8] SalMar. HAVBASERT FISKEOPPDRETT, En ny æra innen havbruk er på vei. <https://www.salmar.no/havbasert-fiskeoppdrett-en-ny-aera/>. Lastet ned: 10.12.19.
- [9] Karl-Johan Reite, Martin Føre, Karl Gunnar Aarsæther, Jørgen Jensen, Per Rundtop, Lars T Kyllingstad, Per Christian Endresen, David Kristiansen, Vegar Johansen, and Arne Fredheim. Fhsim — Time domain simulation of marine systems. In *ASME 2014 33rd International Conference on Ocean, Offshore and Arctic Engineering*, pages V08AT06A014–V08AT06A014. American Society of Mechanical Engineers, 2014.

-
- [10] A. Apte, C. K. R. T. Jones, A. M. Stuart, and J. Voss. Data assimilation: Mathematical and statistical perspectives. *International Journal for Numerical Methods in Fluids*, 56(8):1033–1046, 2008.
- [11] Kody Law, Andrew Stuart, and Kostas Zygalakis. Data assimilation. *Cham, Switzerland: Springer*, 2015.
- [12] Amit Apte. An introduction to data assimilation. In Susmita Sarkar, Uma Basu, and Soumen De, editors, *Applied Mathematics*, pages 31–42, New Delhi, 2015. Springer India.
- [13] Geir Evensen. *Data assimilation: The Ensemble Kalman Filter*. Springer Science & Business Media, 2007.
- [14] Athanasios Papoulis and S Unnikrishna Pillai. *Probability, random variables, and stochastic processes*. Tata McGraw-Hill Education, 2002.
- [15] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45, 1960.
- [16] Simon J Julier and Jeffrey K Uhlmann. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3):401–422, 2004.
- [17] Eric Wan. Sigma-point filters: an overview with applications to integrated navigation and vision assisted control. In *2006 IEEE Nonlinear Statistical Signal Processing Workshop*, pages 201–202. IEEE, 2006.
- [18] Geir Evensen. Sequential data assimilation with a nonlinear quasi-geostrophic model using monte carlo methods to forecast error statistics. *Journal of Geophysical Research: Oceans*, 99(C5):10143–10162, 1994.
- [19] W Keith Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, 04 1970.
- [20] Geir Evensen. The ensemble kalman filter: Theoretical formulation and practical implementation. *Ocean dynamics*, 53(4):343–367, 2003.
- [21] Jan Magnusson, David Gustafsson, Fabia Hüsler, and Tobias Jonas. Assimilation of point swe data into a distributed snow cover model comparing two contrasting methods. *Water resources research*, 50(10):7816–7835, 2014.
- [22] Thomas Nagler, Helmut Rott, Petra Malcher, and Florian Müller. Assimilation of meteorological and remote sensing data for snowmelt runoff forecasting. *Remote sensing of environment*, 112(4):1408–1420, 2008.
- [23] Konstantinos M Andreadis and Dennis P Lettenmaier. Assimilating remotely sensed snow observations into a macroscale hydrology model. *Advances in water resources*, 29(6):872–886, 2006.

-
- [24] Xianhong Xie and Dongxiao Zhang. Data assimilation for distributed hydrological catchment modeling via ensemble kalman filter. *Advances in Water Resources*, 33(6):678–690, 2010.
- [25] Gabriëlle JM De Lannoy, Paul R Houser, Niko EC Verhoest, Valentijn RN Pauwels, and Timothy J Gish. Upscaling of point soil moisture measurements to field averages at the ope3 test site. *Journal of Hydrology*, 343(1-2):1–11, 2007.
- [26] Sondre Myrberg. Kombinasjon mellom gruppedata og matematiske modeller av fiskeatferd i oppdrettsanlegg. Prosjektoppgave ttk4550, NTNU, 2019.
- [27] Lise Doksaeter, Nils Olav Handegard, Olav Rune Godø, Petter H Kvadsheim, and Nina Nordlund. Behavior of captive herring exposed to naval sonar transmissions (1.0–1.6 kHz) throughout a yearly cycle. *The Journal of the Acoustical Society of America*, 131(2):1632–1642, 2012.
- [28] Muhammad Zainuddin Lubis, Henry M Manik, et al. Acoustic systems (split beam echo sounder) to determine abundance of fish in marine fisheries. *Journal of Geoscience, Engineering, Environment, and Technology*, 2(1):76–83, 2017.
- [29] Simrad. Simrad EK60 Installation manual. [https://www.simrad.com/www/01/NOKBG0397.nsf/AllWeb/67F3FF5683856EA7C1257ED8002C08C5/\\$file/164696ab_ek60_installation_manual_english_lores.pdf?OpenElement](https://www.simrad.com/www/01/NOKBG0397.nsf/AllWeb/67F3FF5683856EA7C1257ED8002C08C5/$file/164696ab_ek60_installation_manual_english_lores.pdf?OpenElement), 2015. Lastet ned: 30.04.19.
- [30] David Johansson, Kari Ruohonen, Anders Kiessling, Frode Oppedal, Jan-Erik Stiansen, Mark Kelly, and Jon-Erik Juell. Effect of environmental factors on swimming depth preferences of atlantic salmon (*salmo salar* l.) and temporal and spatial variations in oxygen levels in sea cages at a fjord site. *Aquaculture*, 254(1-4):594–605, 2006.
- [31] Fiskeridirektoratet. Lønnsomhetsundersøkelse for produksjon av laks og regnbueørret 2018. <https://www.fiskeridir.no/Akvakultur/Tall-og-analyse/Statistiske-publikasjoner/Loennsomhetsundersoekelser-for-laks-og-regnbueoerret>, 2018. Lastet ned: 12.11.19.
- [32] Kristoffer Rist Skjøien, Morten Omholt Alver, and Jo Arve Alfredsen. Modelling spatial surface pellet distribution from rotary pneumatic feed spreaders. In *2015 23rd Mediterranean Conference on Control and Automation (MED)*, pages 883–888. IEEE, 2015.
- [33] Lukas. Heatscatter plot for variables X and Y (Version 1.1.0.0), 2014.

Kodesnutter

A.1 MATLAB

A.1.1 inCone

```
function in = inCone(fish_point, xyz_echosounder, phi)
    in = 0;
    if pdist([fish_point(1:2) ; xyz_echosounder(1:2)], '
        Euclidean') < tan(deg2rad(phi)) * (abs(xyz_echosounder
            (3)-fish_point(3)))
        in = 1;
    end
end
```

A.1.2 SonarPing

```
function [dens, fish_depths, nr_fish] = sonarPing(
    xyz_echosounder, depth_echosounder, dz, pos_cart, alpha,
    cone_flag)
    if depth_echosounder > 0
        depth_echosounder = -depth_echosounder;
        fish_depths = zeros(-depth_echosounder/dz,1);
    elseif depth_echosounder == 0
        fish_depths = zeros(30/dz,1);
    else
        fish_depths = zeros(-depth_echosounder/dz,1);
    end

    nr_fish = 0;
```

```

for i = 1:length(pos_cart(:,3))
    if cone_flag
        if inCone(pos_cart(i,:), xyz_echosounder, alpha
            /2)
            fish_depths(round(-pos_cart(i,3)/dz,0)+1) =
                ...
                fish_depths(round(-pos_cart(i,3)/dz,0)
                    +1) + 1;
            nr_fish = nr_fish + 1;
        end
    else
        fish_depths(round(-pos_cart(i,3)/dz,0)+1) = ...
            fish_depths(round(-pos_cart(i,3)/dz,0)+1) +
                1;
        nr_fish = nr_fish + 1;
    end
end
dens = nan(size(fish_depths));
if cone_flag
    for j = 1:length(fish_depths)
        z0 = -depth_echosounder/dz - j;
        V0 = pi/3*(tan(deg2rad(alpha/2))^2)*((z0+dz)^3
            - z0^3);
        dens(j) = fish_depths(j)/V0;
    end
else
    dens = fish_depths;
end
end
end

```

A.1.3 Video

Under følger en kodesnutt som vil generere en video sammensatt av enkeltfigurer. Konstanter og variabler er definert andre steder i koden, slik at dette er kun koden som faktisk lager figurer og setter de sammen til en video.

```

for ii = start:step:stop
    pos = fishRead(filepath + string(ii)+'.xml',nr_fish);
    %pos = fishRead(filepath + string(0)+'.xml',nr_fish);

    scale = 100;

    f1 = figure('units','normalized','outerposition',[0 0 1
        1]);

```

```

figure(f1)
    hist3([pos(:,1), pos(:,2)], 'Nbins', [50 50]);hold
    on;
    s = findobj(gca, 'Type', 'Surface');
    s.FaceAlpha = 1;
    %zticks([-2000 -1000 0 500 1000 2000 3000]);
    %    zticklabels([-2000/scale -1000/scale 0 500 1000
    2000 3000]);
    xlim([-25 25]); ylim([-25 25]); zlim([0 400]); %
        zlim 3500at066BL, 400 at 2BL
    h = get(gca, 'child');
    heights      = get(h, 'Zdata');
    mask         = ~logical(imdilate(heights,ones(3)))
        ;
    heights(mask) = NaN;
    set(h, 'ZData', heights)

%
%    hh = scatter3(pos(:,1), pos(:,2), scale*pos(:,3),
    '.');
%    set(hh, 'MarkerEdgeAlpha', 0.3, 'MarkerFaceAlpha
    ', 0.3)
    xlabel('x [m]'); ylabel('y [m]'); zlabel(['z [m] |
        #fish']);
    titleText = {'Distribution of fish', ...
        strcat('Time elapsed since midnight: ', num2str(
            round(ii/3600 , 2)), 'h')};
    title1 = title(titleText);
    view(140,50);
    drawnow
    F(jj) = getframe(gcf);
    delete(title1);
    close(f1);

end

vid = VideoWriter(video_filename, 'Motion JPEG AVI'); % '
    Uncompressed AVI', 'Motion JPEG 2000'
vid.FrameRate = 10;
open(vid)
writeVideo(vid,F)
close(vid)

```

A.1.4 Interpolert tverrsnitt

```
nrfish = 100000;
```

```

%start = 0; step = 30; stop =start+86100;%86395
depth_echosounder = 0;
dz = 1; alpha = 20; cone_flag = 1; x_line = 3;

time = 3600*11;%3600*12+1000;

if nrfish == 10000
    filepath = 'data/1day10000noenv_2BL/PosVel-';
elseif nrfish == 100000
    filepath = 'data/1day100000noenv_2BL/PosVel-';
%     filepath = 'data/1day100000noenv_066BL/PosVel-';
end

ii = 1;
for yy = -25:5:25
    xyz_echosounders(ii,:) = [x_line yy depth_echosounder];
    ii = ii + 1;
end

pos_cart = fishRead(filepath + string(time)+'.xml',nrfish);
for i = 1:size(xyz_echosounders,1)
    [dens(:, i), fish_depths(:, i) , nr_fish] ...
        = sonarPing(xyz_echosounders(i,:),
                    depth_echosounder, dz, pos_cart, alpha,
                    cone_flag);
end
dens(:,1) = zeros(size(dens(:,1))); %to ignore echosounders
    placed at y=+-25
dens(:,end) = dens(:,1);           %as this is on or
    outside the cage

%% interpolation
k = 5;
dens_q = interp2(dens,k);

%% plotting
figure
    imagesc(dens_q):
    ticks = get(gca, {'XTick', 'YTick'});
    xticks = round(ticks{1,1}/(2^k) * 5 , 0);
    yticks = round(ticks{1,2}/(2^k) , 0);
    set(gca, 'XTickLabels',xticks, 'YTickLabels', yticks);
    c = colorbar;
        c.Label.String = 'Fish density';

```

```
%         labels = string(c.TickLabels);
%         labels(1) = 'Low'; labels(end) = 'High'; labels
(2:end-1) = nan;
%         set(c, 'TickLabels', labels);
title(['Several echosounders placed on a line across
       the pen in x='+string(x_line); ...
       'Time since midnight: '+string(round(time/3600, 2))
       +'h']);
xlabel('Horizontal location [m]');
ylabel('Depth [m]');
xline(0*(2^k)/5, '--');
xline(5*(2^k)/5, '--');
xline(10*(2^k)/5, '--');
xline(15*(2^k)/5, '--');
xline(20*(2^k)/5, '--');
xline(25*(2^k)/5, '--');
xline(30*(2^k)/5, '--');
xline(35*(2^k)/5, '--');
xline(40*(2^k)/5, '--');
xline(45*(2^k)/5, '--');
xline(50*(2^k)/5, '--');
```

