

Ole-Kristian Lie

Prosessering og presentasjon av Sensorfisk-data

Masteroppgave i Industriell kybernetikk
Veileder: Martin Føre og Walter Caharija (SINTEF)
Juni 2019



Foto: SINTEF Ocean

Ole-Kristian Lie

Prosessering og presentasjon av Sensorfisk-data

Masteroppgave i Industriell kybernetikk
Veileder: Martin Føre og Walter Caharija (SINTEF)
Juni 2019

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

Forord

Denne masteroppgaven ble skrevet som en videreføring av en prosjektoppgave (Lie, 2018) utført på oppdrag fra SINTEF Ocean. Oppgaven er skrevet ved institutt for teknisk kybernetikk (ITK) som ligger under fakultetet for informasjonsteknologi og elektronikk (IE) ved Norges teknisk-naturvitenskaplige universitet (NTNU). Masteroppgaven tilsvarer 30 studiepoeng og har en varighet på 21 uker.

Problemstillingen er relatert til FHF-prosjektet KVALISYS. KVALISYS har som hovedmål å kartlegge forholdene for laks i mekaniske avlusningssystemer gjennom utvikling og anvendelse av objektiv målemetodikk, -utstyr og informasjonspresentasjon (Svendsen, 2017). Fra prosjektoppgaven var implementasjon av estimatorstrukturer og utvikling av datafremstillingsmetoder fremhevet som neste mål. Ved oppstart av masteren kom det frem at behovet for dataprosessering var større enn først antatt og dette ble derfor gitt et større fokus i oppgaven enn først planlagt.

For gjennomførelsen av masteroppgaven har SINTEF Ocean stilt reell testdata tilgjengelig. Testene ble gjennomført på flere forskjellige avlusningssystemer og jeg var så heldig at jeg fikk delta på en ekskursjon hvor vi gjennomførte 3 felttester. Denne ekskursjonen ga meg en enorm innsikt i belastningene for laks under mekanisk avlusning, jeg var ikke i nærheten av å kunne forestille meg forholdene før jeg fikk sett systemene med egne øyne. SINTEF Ocean har også stilt en kontorplass til rådighet i deres lokaler som tilrettela for nært samarbeid med forskningsgruppen og tilgang til måleinstrumentene (Sensorfisk). NTNU har stilt en kontorplass med datamaskin til rådighet. Utvikling av algoritmer og kode ble utført i Python 3.7.1, et gratis programmeringsspråk med åpen kildekode.

Jeg ønsker å takke Eirik Svendsen som ledet KVALISYS og engasjerte meg i en sommerjobb hos SINTEF Ocean, som førte til en prosjektoppgave og videre denne masteroppgaven. Takk til veilederne under oppgaven, Walter Caharija som overtok prosjektledelsen av KVALISYS og tok på seg rollen som ekstern veileder, og Martin Føre som intern veileder. En stor takk til familien som alltid har trodd på meg, dere har vært en enorm kilde for motivasjon, støtte, positivitet og oppmuntring. Til slutt ønsker jeg å takke meg selv etter et langt og lærerikt, men ikke minst krevende utdanningsløp.

I want to thank me.

I want to thank me for believing in me.

I want to thank me for doing all this hard work.

I want to thank me for having no days off.

I want to thank me for never quitting.

I want to thank me for always being a giver and trying to give more than I receive.

I want to thank me for trying to do more right than wrong.

I want to thank me for just being me at all times.

—Snoop Dogg

Sammendrag

Mekanisk avlusning av laks er en type ikke-medikamentell avlusning som utsetter fisken for mekaniske påkjenninger. SINTEF Ocean har utviklet en sensorpakke, Sensorfisk, som måler bevegelsesrelatert data. Sensorfisk muliggjør kartlegging og kvantifisering forholdet for laks under mekanisk avlusning.

Målet med denne oppgaven er å lag en kinematisk modell for sensorfiskens posisjon, analysere data for å detektere hendelser av interesse og utvikle og teste forskjellige estimatorstrukturer. Samt utvikle metoder for å visualisere data fra Sensorfisk i 2D og 3D, og til slutt utføre tester basert på resultatene til de foreliggende målene. Datasettene benyttet er hentet inn gjennom felttester på seks forskjellige avlusningssystemer med to forskjellige Sensorfisk.

Resultatet ble flere algoritmer for både prosessing og analysering av Sensorfisk data. Automatisk deteksjon av innløp og utløpt, og automatisk deteksjon av støt ble realisert med utmerket funksjonalitet. Den optimale estimatorstrukturen for estimering av posisjon blir presentert, det gjenstår å implementere estimatorstrukturer.

Abstract

Mechanical delousing of salmon is a type of non-medicinal treatment for lice that exposes the salmon to mechanical strain. SINTEF Ocean has developed a sensor package, Sensorfisk (sensor fish), that measures motion related data. Sensorfisk makes it possible to map and quantify the conditions under mechanical delousing.

This thesis presents a kinematic model for the position of the sensor fish, analyses data to detect events that are of interest and develops and tests different methods of estimation. As well as eveloping methods to visualize data from sensor fish in 2D and 3D, and ultimately perform test based on the results of the tasks. The data used is collected from tests performed on six different delousers with two different Sensorfisk.

The result was several algorithms designed to both process and analyze data from Sensorfisk. Automatic detection of inlet and outlet, as well as automatic detection of impacts was implemented with outstanding functionality. Optimal methods for estimation of position is presented, the implementation of these methods remains.

Innhold

Forord	iii
Sammendrag	iv
Abstract	iv
1 Introduksjon	1
1.1 Motivasjon for oppgaven	1
1.2 Problemstilling	1
1.3 Sensorfisk	2
1.4 Avlusningsenheter	5
1.5 Tidligere forskning	6
1.6 Teori	8
1.6.1 Rotasjon	9
1.6.2 Modell	10
1.6.3 Tilstandsestimering	11
2 Metode	13
2.1 Interpolering	13
2.2 Autodeteksjon	15
2.2.1 Oppdage hendelser basert på trykk data	15
2.2.2 Oppdage hendelser basert på magnetometer data	19
2.2.3 Oppdage hendelser baser på gyroskop data	27
2.3 Estimering	33
2.3.1 Kalmanfilter	33
2.3.2 Variabel estimering	34
3 Resultat	35
3.1 Interpolering	35
3.2 Autodeteksjon	36
3.2.1 Automatisk deteksjon av innløp og utløp	36
3.2.2 Automatisk deteksjon av passering av magnetbånd	39
3.2.3 Automatisk støtdeteksjon	41
4 Diskusjon og konklusjon	49
5 Videre arbeid	51
5.1 Passering av magnetbånd	51
5.2 Støtdeteksjon	51

5.3	Utvikling	51
5.4	Estimering	51
	Vedlegg	53
	A Oppgavebeskrivelse	55
	B Teknisk spesifikasjon for Fisk1 (liten)	57
	Litteraturliste	59

Tabeller

1.1	Sammenlikning av sensorfisker og deres instrumentering.	3
1.2	Detaljert oversikt over instrumenteringen til Fisk1 og Fisk2.	4
1.3	Tabelloversikt over avlusere brukt til datainnsamling.	6
1.4	Notasjon for marine fartøy (SNAME, 1950).	9
2.1	Antall innsamlede datasett fra Fisk1 og Fisk2 brukt i denne rapporten. . .	13
2.2	Testtabell for <code>numpy.correlate</code>	26
2.3	Testtabell for <code>pearsonr</code>	27
3.1	Klassifiseringen av støt brukt til generering av resultater.	41

Figurer

1.1	Foto av Sensorfisk. Foto: (SINTEF Ocean)	2
1.2	Tidssteg på samplingene til akselerometerdata.	5
1.3	Siste iterasjon av Sensor Fish Device. Foto: (PNNL)	7
1.4	Foto av tidligere iterasjoner av Sensorfisk. Foto: (SINTEF Ocean)	8
1.5	Koordinatsystemet til Sensorfisk, orienteringen er representert i BODY.	8
1.6	Blokkdiagram av et lineært, diskret og dynamisk system (Haykin, 1996).	12
2.1	Sammenligning av interpolasjonsmetoder med tidsskritt på 1 ms.	14
2.2	Trykkprofil for Hydrolicer 1.	15
2.3	Trykkprofil med start og slutt markert med røde, vertikale linjer.	17
2.4	Trykkprofil med start og slutt markert med røde, vertikale linjer.	18
2.5	Tidspunkt for start og slutt med forskjellige grensesettingsmetoder.	18
2.6	Trykkprofiler testet med forskjellige grensesettingsmetoder.	19
2.7	Tidspunkt for start og slutt med forskjellige grensesettingsmetoder for intervallet [90:320].	19
2.8	Montering og data fra passering av magnetbånd.	20
2.9	Rådata fra Hydrolicer 1 med 11 magnetbånd montert.	21
2.10	Rådata fra Skamik med 7 magnetbånd montert.	21
2.11	Kvadratisk gjennomsnitt av rådata.	22
2.12	Resultat fra z-skår algoritmen med like innstillinger for to datasett.	24
2.13	Karakteristiske profiler brukt til korrelering, datautvalg markert i rødt.	25
2.14	Rådata fra Skamik.	28
2.15	Prosessert data fra gyroskop.	29
2.16	Sammenlikning av støt og eksponentiell nedgang for et lineært tidsinvariant system.	30
2.17	Sammenlikning av varigheten for støt.	31
2.18	Resultat fra <code>argrelextrema</code> , lokale toppunkt markert med grønne punkt og grenseverdier basert på maks utslag markert med røde, horisontale linjer.	32
3.1	Resultat fra lineærinterpolasjon med tidsskritt på 1 ms.	35
3.2	Resultat av automatisk deteksjon av innløp og utløp for Optilicer.	36
3.3	Resultat av automatisk deteksjon av innløp og utløp for Thermolicer.	36
3.4	Resultat av automatisk deteksjon av innløp og utløp for Skamik.	37
3.5	Resultat av automatisk deteksjon av innløp og utløp for FLS.	37
3.6	Resultat av automatisk deteksjon av innløp og utløp for Hydrolicer 1.	38
3.7	Resultat av automatisk deteksjon av innløp og utløp for Hydrolicer 2 test 1.	38
3.8	Resultat fra <code>numpy.correlate</code> med data fra Skamik, området innfor de røde merkene er datautvalget.	39

3.9	Resultat fra <code>pearsonr</code> med data fra Hydrolicer 1, området innfor de røde merkene er datautvalget.	40
3.10	Resultat av automatisk støtdeleksjon for Optilicer test 1.	41
3.11	Resultat av automatisk støtdeleksjon for Optilicer test 2.	42
3.12	Resultat av automatisk støtdeleksjon for Thermolicer test 1.	43
3.13	Resultat av automatisk støtdeleksjon for Thermolicer test 2.	43
3.14	Resultat av automatisk støtdeleksjon for Skamik test 1.	44
3.15	Resultat av automatisk støtdeleksjon for Skamik test 2.	45
3.16	Resultat av automatisk støtdeleksjon for FLS test 1.	45
3.17	Resultat av automatisk støtdeleksjon for FLS test 2.	46
3.18	Resultat av automatisk støtdeleksjon for Hydrolicer 1 test 1.	47
3.19	Resultat av automatisk støtdeleksjon for Hydrolicer 1 test 2.	47
3.20	Resultat av automatisk støtdeleksjon for Hydrolicer 2 test 1.	48

Kapittel 1. Introduksjon

1.1 Motivasjon for oppgaven

Lakselus er den største utfordringen for havbruksnæringen (Holan, 2017). Lusen er i seg selv en plage for oppdrettslaks som fører til redusert velferd, i tillegg til at behandlingen mot lakselus introduserer påkjenninger for laks og nærmiljøet. Det er to former for behandlinger mot lakselus, medikamentell avlusning og ikke-medikamentell avlusning (IMA). Med økende resistensutvikling og problemer med miljøutslipp har det vært et mål fra næringen å redusere bruk av medikamenter og øke andelen av IMA for behandling av lakselus.

Det er flere forskjellige behandlingsformer innenfor IMA. Bruk av renseskum som spiser lusen av laksen, termisk avlusning som benytter oppvarmet vann, og mekanisk avlusning som benytter seg av spyling eller børster. Disse metodene, utenom bruk av renseskum, utsetter laksen for deloperasjoner som trenging, pumping og selve behandlingen (Lie, 2018). SINTEF Ocean har utviklet en sensorpakke (Sensorfisk) som måler bevegelsesrelatert data under avlusning som muliggjør kartlegging av de mekaniske belastningene. Det vil også være mulig å kvantifisere forholdene i mekaniske avlusere og skape et sammenligningsgrunnlag for ulike maskiner og metoder.

1.2 Problemstilling

Innsamlet bevegelsesrelatert data inneholder mye informasjon om forholdene under ikke-medikamentell avlusning (IMA). Prosessering av denne informasjonen kan fortelle oss om hendelser som oppstår under IMA og gjøre det mulig å detektere slike hendelser automatisk, som eksempelvis støt. Det er også mulig å estimere posisjonen til sensorfisken gjennom avlusningssystemene og koble dette sammen med oppdagede hendelser.

Et viktig mål er informasjonrepresentasjon. Det må visualiseres hvor hendelser oppstår og hvilket omfang de har. Resultatene skal formidles til forskjellige aktører med ulik akademisk bakgrunn og gjennom populærvitenskap, det er derfor nødvendig å utforme enstydige og beskrivende illustrasjoner.

Ut fra dette og oppgavebeskrivelsen (vedlegg A) ble følgende punkter beskrevet:

1. Lage en kinematisk modell for sensorfiskens posisjon.
2. Analysere data for å detektere hendelser av interesse som f.eks. kraftige støt.

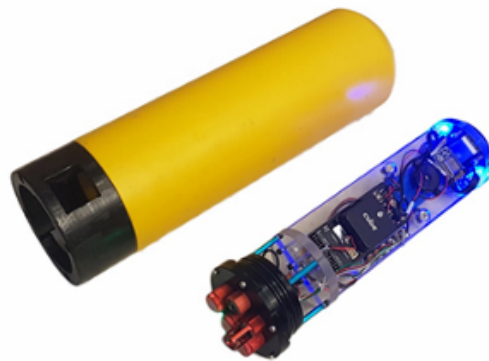
3. Utvikle og teste forskjellige estimatorstrukturer. Hovedsakelig Kalmanfilter og Utvidet Kalmanfilter, og eventuelt andre løsninger som Fossens ikke-lineære observer.
4. Utvikle metoder for å visualisere data fra sensorfisk i 2D og 3D som illustrerer målte verdien på forskjellige posisjoner i avluseren.
5. Utføre tester basert på resultatene i punkt 1) - 4) med reelle data.

1.3 Sensorfisk

SINTEF Ocean har utviklet tre forskjellige Sensorfisk (Fisk1, Fisk2 og Fisk3), to som logger bevegelsesrelatert data og en som er utstyrt med en trykkfølsom matte (fig. 1.1). De to som logger bevegelsesrelatert data har lik instrumentering men forskjellig vekt og størrelse, en på 3 kg (Fisk1) og en på 6 kg (Fisk2). Det er for å kunne måle forskjellene i påkjenninger til laks av ulik størrelse og siden enkelte avlusere er dimensjonert for mindre, eller større fisk. Den fisken med trykkfølsom matte (Fisk3) har en vekt på 6 kg og er lik Fisk2 i størrelsen. En oversikt over instrumenteringen til sensorfiskene er å finne i tabell 1.1. En mer detaljert oppsummering av instrumenteringen til Fisk1 og Fisk2 er å finne i tabell 1.2, oppsummert fra den tekniske spesifikasjonen for Fisk1 (vedlegg B).



(a) Henholdsvis Fisk1, Fisk2 og Fisk3.



(b) Fisk1, kapsel og instrumentering

Figur 1.1: Foto av Sensorfisk. Foto: (SINTEF Ocean)

For analysering av de mekaniske påkjenningene og prosessering av innsamlet data er det bevegelsesrelatert data fra Fisk 1 og Fisk2 som er benyttet. I tillegg det er flere datasett tilgjengelig fra flere avlusningsenheter innhentet på Fisk2 som gir et bedre

sammenlikningsgrunnlag. Datasettene innsamlet fra Fisk1 er lik de fra Fisk2 og konsepter som er vist til å fungere på data fra Fisk2 vil også fungere på data fra Fisk 1. Fisk3 samler ikke inn bevegelsesrelatert data, derfor er ikke data fra Fisk3 benyttet.

Tabell 1.1: Sammenlikning av sensorfisker og deres instrumentering.

	Fisk1	Fisk2	Fisk3
Vekt [kg]	3	6	6
Instrumentering	Pixhawk mikrokontroller 3 stk IMU, hver med: <ul style="list-style-type: none"> • 3 akse akselerometer • 3 akse gyroskop • 3 akse magnetometer Trykkmåler Temperaturmåler	Likt som Fisk1 bare med høyere vekt	Arduino mega mikrokontroller Trykkløslig matte med Snowboard 2 mikrokontroller Magnetometer skjold RTC - real time clock skjold

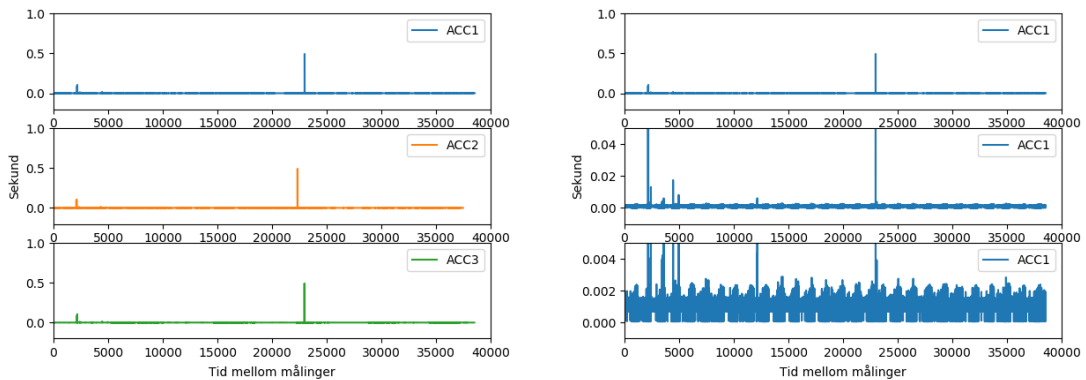
Tabell 1.2: Detaljert oversikt over instrumenteringen til Fisk1 og Fisk2.

Instrumentering for Fisk1 og Fisk2	
Mikrokontroller	
Navn	Pixhawk 2.1
IMU	
Navn	Nine-Axis (Gyro + Accelerometer + Compass) MEMS MotionTracking™ Device
Sensor	MPU9250
Antall IMU	3
Toleranse akselerometer	$\pm 3\% g$
Toleranse gyroskop	$\pm 3\% ^\circ/s$
Kombinert trykk og temperatur sensor	
Navn	Bar30 High-Resolution 300m Depth/Pressure Sensor
Sensor	MS5837
Nøyaktighet ved 25°C, 0-6 Bar	± 50 mBar
Oppløsning	± 0.20 mBar
Temperatur sensor	
Navn	Celsius Fast-Response, $\pm 0.1^\circ\text{C}$ Temperature Sensor (I2C)
Sensor	TSYS01
Nøyaktighet ved $-5^\circ\text{C} < T < 50^\circ\text{C}$	$\pm 0.1^\circ\text{C}$
Oppløsning	$\pm 0.01^\circ\text{C}$
Logger innstillinger	
Akselerometer	1000 Hz
Gyroskop	1000 Hz
Magnetometer	50 Hz
Kombinert trykk og temperatur	50 Hz
Temperatur (Fast-response)	20 Hz

Sensorfisk logger med en brukersatt samplingsfrekvens som styres av prosessoren og lagrer data på et SD-kort. Den valgte samplingsfrekvensen for hver enkelt sensor står oppført nederst i tabell 1.2. Som en følge av at loggingen er styrt av prosessoren oppstår det variasjoner i tidssteget mellom målingene. Dersom det ikke er prosesseringskraft tilgjengelig for å loggføre målingene blir samplingen utsatt til når prosessoren har tilgjengelig kapasitet. Dette påvirker de tre forskjellige IMUene forskjellig, med en liten variasjon i når- og hvor kraftig samlingen påvirkes, noe som kan ses i fig. 1.2a.

Ved å zoome inn på akselerometerdata i fig. 1.2b ser vi at tidssteget ligger rundt 1 ms, som

tilsvarende en samplingsfrekvens på 1000 Hz. Men tidssteget varierer kraftig mellom 0-2 ms samtidig som det kan oppstå større avvik med et tidssteg på hele 500 ms. Dette må tas hensyn til i dataprosesseringen.



(a) Akselerometerdata fra akselerometer 1-3. (b) Zoomet bilde av akselerometerdata fra akselerometer 1.

Figur 1.2: Tidssteg på samplingene til akselerometerdata.

1.4 Avlusningsenheter

For å forstå de mekaniske belastningene som laks utsettes for under IMA trengs det kunnskap om de forskjellige avlusningssystemene. For å kunne sammenlikne og kvantifisere belastninger må deloperasjoner gjøres rede for, og ses på adskilt fra de resterende operasjonene. Noen av deloperasjonene gjentar seg for alle IMA metodene, som trenging og pumping, mens noen forekommer kun i enkelte systemer, som f.eks. børsting eller kontakt med skovlhjul.

Disse systemene blir spesialbygget på bestilling og det gjøres kontinuerlige endringer og utskiftninger av utstyr på de eksisterende systemene. Informasjonen om de enkelte systemene kan derfor være noe upresis eller utdatert. To systemer av samme prinsipp er ikke nødvendigvis lik, f.eks. ved at rørganger flyttes. Det skal heller ikke være mulig å gjenkjenne spesifikke systemer av hensyn til rederiene og anonymisering testresultater. Konseptene og deloperasjonene er i hovedsak lik for de systemene som deler prinsipp.

Det er fem prinsipper av IMA og seks avlusere som det skal ses nærmere på, Optilicer og Thermolicer som behandler med oppvarmet vann, to forskjellige Hydrolicere og FLS som behandler med spyling, og Skamik som behandler med en kombinasjon av spyling og børsting. Optilicer sin behandlingsenhet er et åpent vannbad med oppvarmet sjøvann

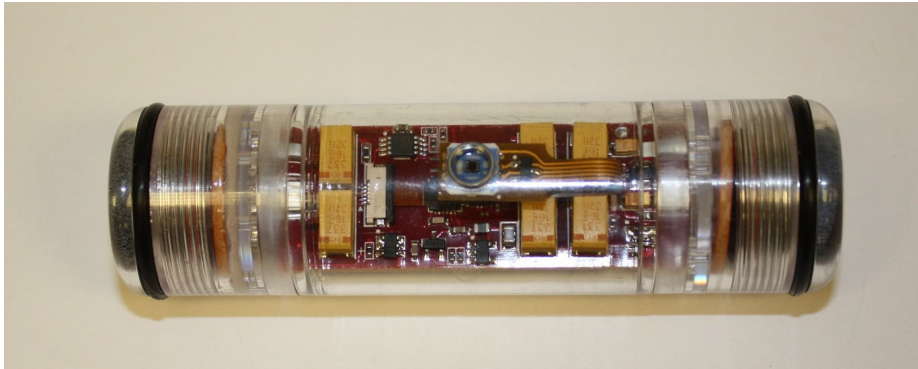
som laksen blir ført gjennom av skovlhjul. Thermolicer har en lukket behandlingsenhet med oppvarmet sjøvann som laksen bli pumpet gjennom. Hydrolicer spyler laks med en vertikal stråle som skaper turbulens. FLS spyler laksen med en motstrøms lavtrykkavluser. Skamik børster laksen ren for lus med myke børster. Tabell 1.3 viser en oversikt over hvilke deloperasjoner som inngår i avlusningssystemene.

Tabell 1.3: Tabelloversikt over avlusere brukt til datainnsamling.

	Optilicer	Thermolicer	Hydrolicer 1	Hydrolicer 2	Skamik	FLS
Pumpesystem						
Trykk-vakuumpumpe	X	X			X	
Ejektorpumpe			X			X
Impellerpumpe				X		
Avlusningssystem						
Spyling			X	X	X	X
Kontakt med varmt vann	X	X				
Børsting					X	
Design						
Kontakt med skovlhjul	X					
Avsiling	X	X		X	X	X
Delvis vannfylte rør	X	X	X	X	X	X
Helt vannfylte rør	X	X	X	X	X	X
Bøyning i rørgang på over 90 GRADER	X	X			X	
Løftehøyde	~10m	~4m	~5.5m	~6m	~5m	<1m

1.5 Tidligere forskning

Når lakseyngel gjennomgår smoltifisering trekker den vekk fra ferskvannet i elvene og ut til saltvannet havet. På denne reisen passerer smolten (ung laks) ofte demninger og vannkraftverk. Energi departementet i USA har ved sitt laboratorium, Pacific Northwest National Laboratory (PNNL), utviklet en sensorfisk (Sensor Fish Device, fig. 1.3) for kartletting av forhold gjennom vantturbiner. Denne utviklingen startet i 1998 (Carlson og Duncan, 2003) og deres siste iterasjon var utviklet i 2014 (Deng mfl., 2014). Målet til PNNL er å bidra til utviklingen av mer fiskevennlige demninger og redusere miljøinnvirkningene fra vannkraft.

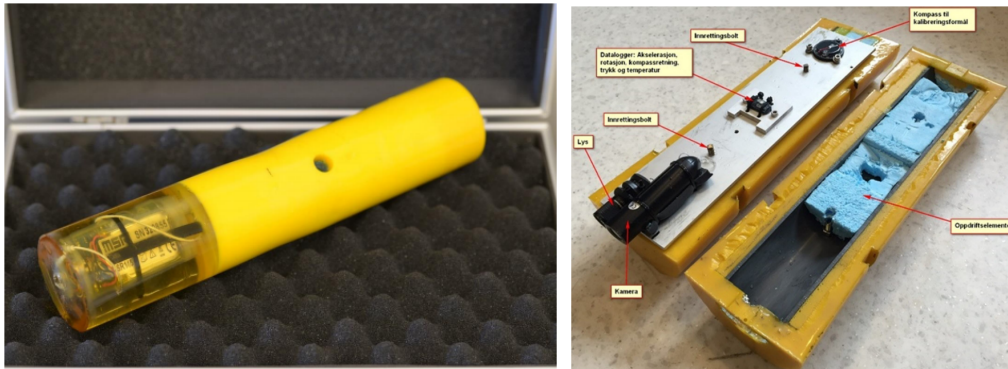


Figur 1.3: Siste iterasjon av Sensor Fish Device. Foto: (PNNL)

PNNLs Sensor Fish Device (SFD) har mange likheter med SINTEF sitt design. Den består av et akselerometer, gyroskop, kompass, trykksensor og temperatursensor som henholdsvis måler lineærakselerasjon, rotasjonshastighet, orientering, trykk og temperatur. Dette logges med en frekvens på 2048 Hz hvor innsamlet data blir skrevet til et SD-kort. Forskjellen fra Sensorfisk og SFD er størrelsen. Sensorfisk har en størrelse og vekt tilsvarende for lask på 3 og 6 kg, mens SFD har en størrelse og vekt tilsvarende smolt på ca. 45 gram. Som en følge av størrelsen og vektbegrensningen har SFD en begrenset loggetid på kun 4 minutter mot Sensorfisk sine 12 timer. Utvikling og testing er fullført og Sensor Fish Device er i bruk for å evaluere forhold til fisk under ferdsel gjennom turbiner og andre hydrauliske systemer i flere land (Deng mfl., 2014).

SINTEF Ocean begynte forskning rundt problematikken angående kvalitetsutfordringene ved pumping av pelagisk fisk tidlig på 2000-tallet og videreutviklet dette konseptet i 2017 (Tveit mfl., 2017). SINTEF utviklet en sylinderformet loggerenhet på størrelsen til en makrell, se fig. 1.4a. Denne enheten logget akselerasjon, lys og trykk, der akselerasjons- og trykk-målingene ble brukt for å determinere påkjenningene under pumping, og lysmålinger sammen med trykkmålinger ble brukt for å identifisere posisjonen til loggerenheten. SINTEF Ocean klarte å gjenkjenne punkter i pumpeforløpet som påførte fisken mer krefter enn andre og hvilke pumpeenheter som resulterte i best fiskekvalitet (Tveit mfl., 2017).

SINTEF Ocean utviklet også en tidligere iterasjon av Sensorfisk (fig. 1.4b) ved starten av KVALISYS i 2017. Dette designet var tilstrekkelig for å bevise konseptet, men det var fremdeles behov for forbedringer, derav ble Fisk1, Fisk2 og Fisk3 utviklet.



(a) Loggerenhet med ytre diameter på 50mm og lengde ca. 225mm.

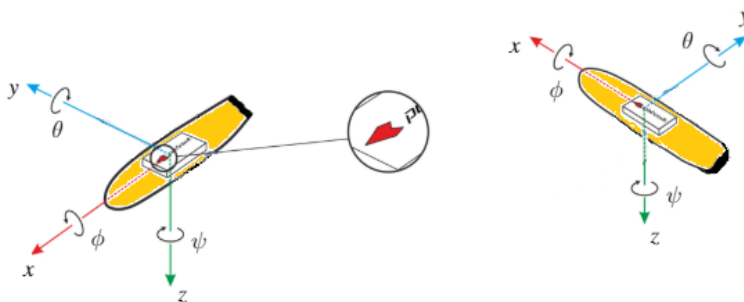
(b) Første iterasjon av Sensorfisk

Figur 1.4: Foto av tidligere iterasjoner av Sensorfisk. Foto: (SINTEF Ocean)

1.6 Teori

Det meste av relevant litteratur og teori ble innhentet i den forliggende prosjektoppgaven (Lie, 2018) hvor det ble gjennomført et litteraturstudie. Den relevante teorien stammer hovedsakelig fra pensumet til kybernetikk-fagene ved NTNU og artikler fra IEEE.org.

Notasjon brukt for å representere bevegelsen og posisjon til Sensorfisk samsvarer med standard notasjon for marine fartøy, oppsummert i tabell 1.4. Koordinatsystemet til Sensorfisk er vist i fig. 1.5 og følger et høyrehånds-system. Dette koordinatsystemet som følger Sensorfisk sin orientering er kalt BODY.



Figur 1.5: Koordinatsystemet til Sensorfisk, orienteringen er representert i BODY.

Tabell 1.4: Notasjon for marine fartøy (SNAME, 1950).

DOF		Forces and moments	Linear and angular velocities	Position and Euler angels
1	motions in the x direction (surge)	X	u	x
2	motions in the y direction (sway)	Y	v	y
3	motions in the z direction (heave)	Z	w	z
4	rotation about the x axis (roll, heel)	K	p	ϕ
5	rotation about the y axis (pitch, trim)	M	q	θ
6	rotation about the z axis (yaw)	N	r	ψ

1.6.1 Rotasjon

Akselerasjonsmålingene til Sensorfisk er påvirket av tyngdekraften. Effekten av tyngdeakselerasjonen vil variere ettersom orienteringen endres. Dette kommer av at akselerometermålingene er relativt til sensorfiskens orientering mens tyngdeakselerasjonen virker i en konstant retning. For å kunne determinere den faktiske innvirkning på målingene vil det være nødvendig å rotere tyngdeakselerasjonen til sensorfiskens orientering før den kan kompenseres for.

Det vil være mer hensiktsmessig å se på akselerasjonsmålingene relativt til avluseren, der avluseren representerer et lokalt plan. Når man skal estimere hastigheten og videre posisjonen ønsker man å se dette relativt til avlusningssystemet og fremstille posisjonen til Sensorfisk i henhold til avluseren. Det vil igjen være nødvendig å foreta rotasjoner for å realisere denne fremstillingen.

Når man skal rotere koordinater fra et koordinatsystem som er relativt til sensorfiskens orientering, kalt BODY {b}, til et lokalt tangentplan, NED {n} kan man multiplisere hver rotasjonsmatrise for å utføre rotasjon om alle aksene. Matrisene for rotasjon rund hver akse er

$$\mathbf{R}_{x,\phi} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\phi & -s\phi \\ 0 & s\phi & c\phi \end{bmatrix}, \quad \mathbf{R}_{y,\theta} = \begin{bmatrix} c\theta & 0 & s\theta \\ 0 & 1 & 0 \\ -s\theta & 0 & c\theta \end{bmatrix}, \quad \mathbf{R}_{z,\psi} = \begin{bmatrix} c\psi & -s\psi & 0 \\ s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1.1)$$

der $s \cdot = \sin(\cdot)$ og $c \cdot = \cos(\cdot)$ (Thor I. Fossen, 2011). Rotasjonen fra {b} til {n} kan skrives som $\mathbf{R}_b^n(\Theta_{nb})$, hvor argumentet $\Theta_{nb} = [\phi, \theta, \psi]^T$, rotasjonsmatrisen er definert som

$$\mathbf{R}_b^n(\Theta_{nb}) := \mathbf{R}_{z,\psi} \mathbf{R}_{y,\theta} \mathbf{R}_{x,\phi} = \begin{bmatrix} c\psi c\phi & -s\psi c\phi + c\psi s\theta s\phi & s\psi s\phi + c\psi c\phi s\theta \\ s\psi c\phi & c\psi c\phi + s\phi s\theta s\psi & -c\psi s\phi + s\theta s\psi c\phi \\ -s\theta & c\theta s\phi & c\theta s\phi \end{bmatrix} \quad (1.2)$$

1.6.2 Modell

For å kunne estimere posisjonen er det nødvendig å lage en matematisk modell som kan brukes som utgangspunkt for estimeringen (Egeland og Gravdal, 2002). Fra definisjonene i tabell 1.4 har vi følgende sammenheng mellom lineærakselerasjon a , de lineære hastighetene og posisjon:

$$\begin{aligned} a_u &= \dot{u} = \ddot{x} \\ a_v &= \dot{v} = \ddot{y} \\ a_w &= \dot{w} = \ddot{z} \end{aligned} \quad (1.3)$$

Sammenhengen mellom rotasjonsakselerasjon α , rotasjonshastighet og Euler-vinkler er som følger:

$$\begin{aligned} \alpha_p &= \dot{p} = \ddot{\phi} \\ \alpha_q &= \dot{q} = \ddot{\theta} \\ \alpha_r &= \dot{r} = \ddot{\psi} \end{aligned} \quad (1.4)$$

Det er ønskelig å representere Sensorfiskens posisjon $\{\mathbf{b}\}$ relativt til avluseren $\{\mathbf{n}\}$. Til denne representasjonen brukes notasjon introdusert av Perez og T. I. Fossen, 2007. Først defineres en vektor \vec{r}_{nb}^n som uttrykker koordinatene til Sensorfisk i $\{\mathbf{n}\}$.

$$\mathbf{r}_{nb}^n \triangleq \begin{bmatrix} N \\ E \\ D \end{bmatrix} \quad (1.5)$$

Deretter defineres orienteringen som gis av Euler-vinklene, roll, pitch og yaw, $\{\mathbf{n}\}$ relativt til $\{\mathbf{b}\}$.

$$\Theta_{nb} \triangleq \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} \quad (1.6)$$

Enderinger i posisjon og orientering er henholdsvis den lineære hastigheten og vinkelhastigheten. Disse hastighetene beskrives relativt til Sensorfisk i $\{\mathbf{b}\}$. Den lineære hastigheten er derivatet av \mathbf{r}_{nb}^n rotert fra $\{\mathbf{n}\}$ til $\{\mathbf{b}\}$:

$$\mathbf{v}_{nb}^b \triangleq \mathbf{R}_n^b \dot{\mathbf{r}}_{nb}^n = \mathbf{R}_n^b \begin{bmatrix} \dot{N} \\ \dot{E} \\ \dot{D} \end{bmatrix} = \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (1.7)$$

Vinkelhastigheten relativt til $\{b\}$ er gitt ved:

$$\omega_{nb}^b = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (1.8)$$

Komponentene i den lineære hastigheten, u, v og w , kalles *surge, sway* og *heave velocity*, og komponentene i vinkelhastigheten, p, q og r , kalles *roll, pitch* og *yaw rate* (SNAME, 1950). Derivasjon av Euler-vinklene i likning (1.6) gir følgende forhold til ω_{nb}^b :

$$\dot{\Theta}_{nb} = \mathbf{T}_b(\Theta_{nb})\omega_{nb}^b \quad (1.9)$$

der

$$\mathbf{T}_b(\Theta_{nb}) = \begin{bmatrix} 1 & s\phi t\theta & c\phi t\theta \\ 0 & c\phi & -s\phi \\ 0 & \frac{s\phi}{c\theta} & \frac{c\phi}{c\theta} \end{bmatrix} \quad (1.10)$$

Så slås uttrykkene sammen til en generalisert posisjonsvektor:

$$\eta \triangleq \begin{bmatrix} \mathbf{r}_{nb}^n \\ \Theta_{nb} \end{bmatrix} = [N, E, D, \phi, \theta, \psi]^T \quad (1.11)$$

og en generalisert hastighetsvektor:

$$\nu \triangleq \begin{bmatrix} \mathbf{v}_{nb}^b \\ \omega_{nb}^b \end{bmatrix} = [u, v, w, p, q, r]^T \quad (1.12)$$

Ved å kombinere definisjonen fra de to vektorene fra likning (1.7) og likning (1.9) blir den følgende kinematiske modell resultatet (Perez og T. I. Fossen, 2007):

$$\dot{\eta} \triangleq \mathbf{J}_b^n(\eta)\nu \quad (1.13)$$

der

$$\mathbf{J}_b^n(\eta) \triangleq \begin{bmatrix} \mathbf{R}_b^n(\Theta_{nb}) & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{T}^{-1}(\Theta_{nb}) \end{bmatrix} \quad (1.14)$$

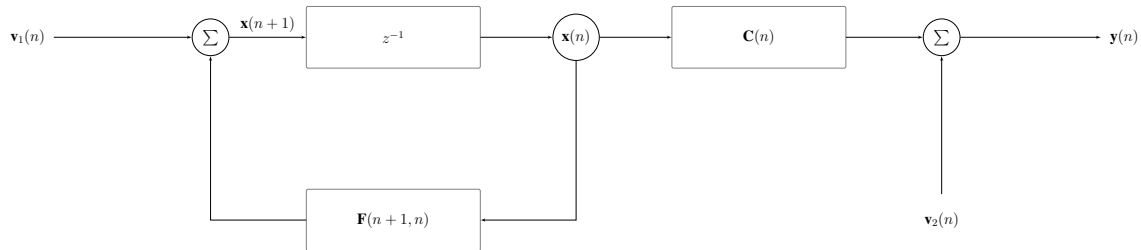
1.6.3 Tilstandsestimering

Betrakt blokkdiagrammet i fig. 1.6 hvor systemet er beskrevet. En slik representasjon i tidsdomenet gir følgende fordeler (Haykin, 1996):

- Praktisk matematisk fremstilling og notasjon

- Nært forhold til virkeligheten
- Nyttig grunnlag for redegjørelse av den statistiske oppførselen til systemet

I denne tilstandsromrepresentasjonen er $\mathbf{x}(n)$ tilstandsvektoren. For å estimere tilstandsvektoren benyttes målt data representert ved observasjonsvektoren $\mathbf{y}(n)$



Figur 1.6: Blokkdiagram av et lineært, diskret og dynamisk system (Haykin, 1996).

Fra blokkdiagrammet i fig. 1.6 kan prosesslikningen og målelikningen observeres, henholdsvis likning (1.15) og (1.16).

$$\mathbf{x}(n+1) = \mathbf{F}(n+1)\mathbf{x}(n) + \mathbf{v}_1(n) \quad (1.15)$$

$$\mathbf{y}(n) = \mathbf{C}(n)\mathbf{x}(n) + \mathbf{v}_2(n) \quad (1.16)$$

For prosesslikningen er $\mathbf{F}(n+1)$ tilstandsoppdaterings-matrisen som er forholdet mellom tilstanden fra n til $n+1$. $\mathbf{v}_1(n)$ er prosessstøyen som modelleres som gaussisk støy. For målelikningen er $\mathbf{C}(n)$ målematrisen som viser hvilke variabler som er kjent. $\mathbf{v}_2(n)$ er målestøyet som modelleres som gaussisk støy (Haykin, 1996).

Tilstandsoppdaterings-matrisen er avhengig av tidsskrittet og må kalkuleres for hver iterasjon, men med et konstant tidsskritt vil den være konstant (Haykin, 1996). Det er mulig å oppnå med å interpolere datasettet, mer om dette i kapittel 2.1.

Kapittel 2. Metode

Til programmering og utvikling av algoritmer ble nyeste versjon av Python benyttet, versjon 3.7.1. Python er et gratis programmeringsspråk med åpen kildekode som er mye brukt til utdanning og forskning (Python Software Foundation, 2018). Python har et bibliotek kalt *pandas* som legger til rette for rask og effektiv dataanalyse samtidig som det gir en oversiktlig håndtering av større datasett ved bruk av DataFrames. DataFrames er todimensjonale tabeller som kan inneholde forskjellig type data samtidig, mens metadata lagres “utenfor” datasettet. Det vil si at kolonnenavn og radnummerering ikke inngår i datasettet, men lagres sammen med datasettet og vises ved visualisering. *pandas* tilrettelegger for en meget intuitiv behandling av data og gjør det enkelt å visualisere data underveis som man jobber.

Datasettene brukt til utvikling, testing og generering av resultater er innhentet gjennom flere felttester med seks forskjellige avlusere. Det er brukt 10 datasett fra fem forskjellige avlusere samlet inn med Fisk2 og ett datasett samlet inn med Fisk 1, se tabell 2.1 for en detaljert oversikt.

Tabell 2.1: Antall innsamlede datasett fra Fisk1 og Fisk2 brukt i denne rapporten.

	Optilicer	Thermolicer	Hydrolicer 1	Hydrolicer 2	Skamik	FLS
Fisk 1	0	0	0	1	0	0
Fisk 2	2	2	2	0	2	2

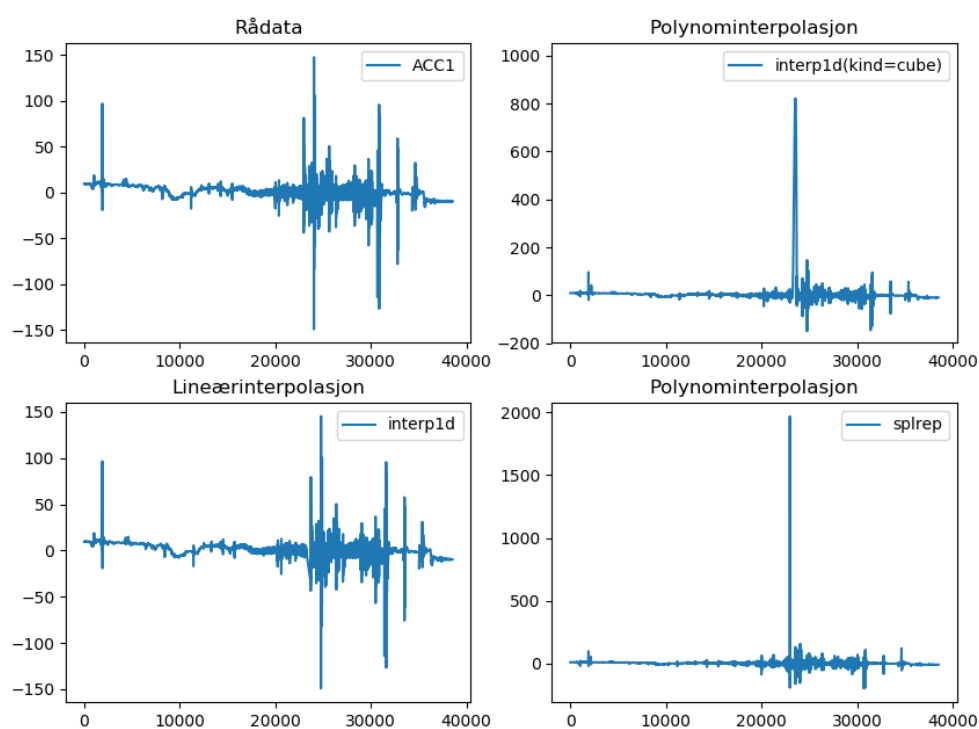
2.1 Interpolering

Siden sensorfisken ikke har et konstant tidssteg mellom samplingene er det nødvendig å utføre en interpolering for å oppnå dette. Deretter er det mulig å sample fra interpoleringen med et konstant tidssteg slik at det oppnås et tidsinvariant system. Dette vil forenkle implementasjon av estimatoren, eksempelvis ved at tilstandsoppdaterings-matrisen for modellen, F -matrisen, blir konstant (Haykin, 1996).

Ved interpolering i Python har man to valg, man kan benytte seg av lineærinterpolasjon eller splineinterpolasjon. Ved lineærinterpolasjon antas det at funksjonen er konstant mellom to kjente punkt og det regnes ut en rett linje mellom dem. Til dette brukes funksjonen `interp1d(x, y) (ny_x)` som tar inn matriser med data som det interpoleres over (x, y) og returnerer en matrise med interpolert data med det definerte tidssteget (ny_x) .

Splineinterpolasjon er en type polynominterpolasjon, polynominterpolasjon finner et polynom av høy grad som passer datapunktene, men for store datasett blir dette alt

for krevende og unøyaktig (fig. 2.1). Splineinterpolasjon benytter seg av intervaller fra datasettet og polynomer med lav grad, det kalkuleres nye polynomer for hvert intervall samtidig som det passes på at overgangene mellom polynomene er glatte. Det ble testet to funksjoner som utfører splineinterpolasjon, `interp1d` hvor argumentet `kind = 'cubic'` bestemmer at det skal utføres en splineinterpolasjon med et polynom av grad 3, og `interpolate.splrep` som utfører splineinterpolasjon med et polynom av grad 3. Resultatet av interpolasjonene med tidsskritt på 1 ms kan ses i fig. 2.1.



Figur 2.1: Sammenligning av interpolasjonsmetoder med tidsskritt på 1 ms.

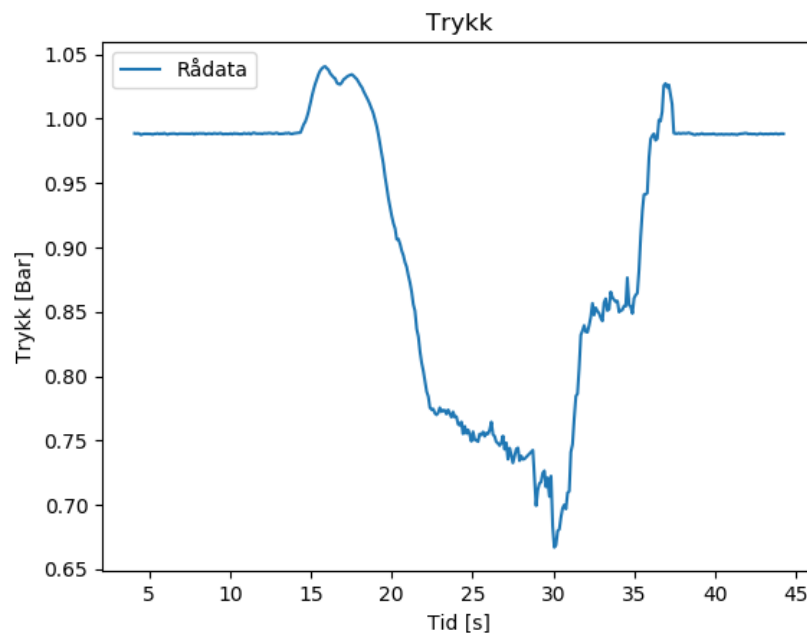
Fra resultatene av interpoleringen er det klart at polynominterpolasjon i form av spline-metoden er for ustabil og gir ikke tilstrekkelig tilnærming til rådata, men lineærinterpolasjon ser ut til å resultere i en ganske god tilnærming. I kapittel 3.1 ses det nærmere på resultatet fra lineærinterpolasjon.

2.2 Autodeteksjon

Det er flere hendelser som gjentar seg under avlusning, disse kan detekteres automatisk gjennom dataanalyse ved bruk av algoritmer. Sensorfisken passerer flere magnetbånd som er montert langs rørgangen hvor utslaget av magnetfeltet måles av magnetometeret, sensorfisken har overganger fra luft til vann hvor trykkendringen måles av trykksensoren, og Sensorfisk kolliderer med rørganger hvor rotasjon måles av gyrokopene. Hvordan kan disse hendelsene detekteres automatisk og hvilke metoder er det som gir best resultat?

2.2.1 Oppdage hendelser basert på trykk data

Trykkprofilen er meget lett gjenkjennelig for de forskjellige systemene og man ser tydelig når fisken går inn i behandlingseenheten, og når den kommer ut (fig. 2.2). Siden sensorfisken begynner datainnsamlingen ved havoverflaten før den går inn i innsuget er det en markant trykkforskjell som fremkommer i overganger fra luft til vann. Likt for utgangen av avluseren hvor den går fra å være senket i vann til å bli hentet opp til overflaten og slås av.



Figur 2.2: Trykkprofil for Hydrolicer 1.

Dette gir en tydelig overgang fra atmosfærisk trykk ved havoverflaten til høyere trykk

under havoverflaten, som indikerer innløpet. Ved slutten av avlusningen er det en overgang fra et høyere trykk under havoverflaten til et lavere atmosfærisk trykk, som indikerer utløpet. Disse overgangene gir en nøyaktig indikasjon på når avlusningen stater og når den avsluttes, noe som kan brukes videre med dataanalyse og estimeringen.

Ved å sette en øvre grense like over trykket ved havoverflaten og se på når trykket blir større enn denne grensen, er det mulig å oppdage når sensorfisken befinner seg i vann. Når trykket overgår grensen settes et signal til verdien en og indeksen lagres i en tabell, dersom trykket ikke overgår grensen er verdien til signalet null. Den første gangen trykket overgår denne grensen vil være når sensorfisken går inn i avluseren og den siste gangen vil være når den kommer ut av avluseren før den føres til overflaten. Ved å hente ut indeksene til disse punktene kan tidspunktet med tilsvarende indeks hentes, og slik finne tidspunktet når sensorfisken går inn i avluseren og når den kommer. Pseudokoden for algoritmen finnes under i algoritme 1.

Algoritme 1 Pseudokode for å oppdage start og slutt fra trykkdata.

```

1: Lager en tabell med nuller med lik lengde som trykkdata for å lagre signaler
2: for i in range( lengde av trykkdata ) do
3:   if trykk[i] > grense then
4:     Sett signal[i] = 1
5:   else
6:     Sett signal[i] = 0
7: Lagrer indeksene til signalene som ikke er null
8: Henter ut den første indeksen som indikerer start
9: Henter ut den siste indeksen som indikerer slutt
10: Skriver tiden for start og slutt i en DataFrame

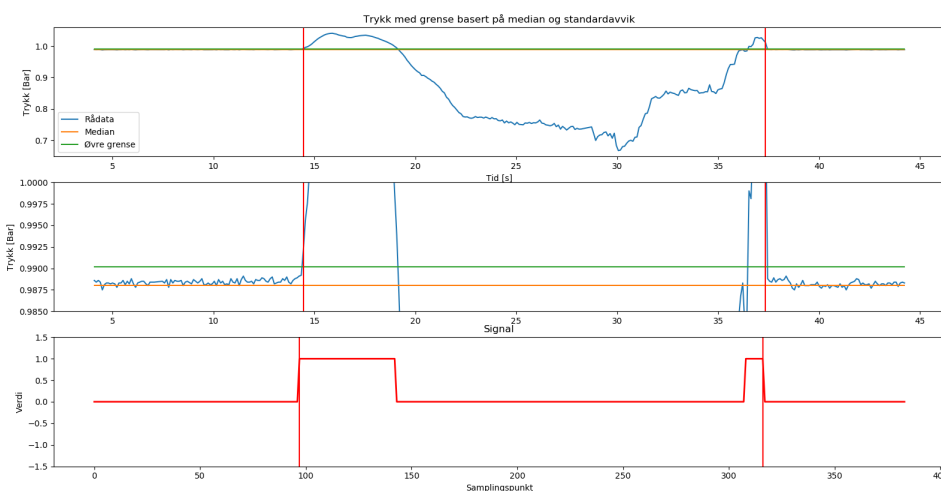
```

For å bestemme grensen som trykket må overstige ble det først testet med en marginal økning på middelveiden. Siden sensorfisken ligger ved overflaten i 20 til 40 sekunder etter den er slått på før den føres inn i avluseren (for å koble seg til GPS), er det logisk å anta at middelveiden vil ligge ved atmosfærisk trykk. Med de store variasjonene på trykket blir standardavviket mye større enn den marginale økningen som er nødvendig, derfor ble grensen satt til å være middelveiden pluss en femtiendedel av standardavviket (likning 2.1).

$$\text{grense} = \text{middelveidi} + \frac{\text{standardavvik}}{50} \quad (2.1)$$

Dette ga gode resultater og algoritmen oppdaget start og slutt uten problemer. I fig. 2.3 vises resultatet av algoritmen med trykkprofilen til Hydrolicer 1 (fig. 2.2) hvor grensen

er basert på middelværdien og standardavviket. De røde, vertikale linjene markerer hvor algoritmen registrerer start og slutt tid.



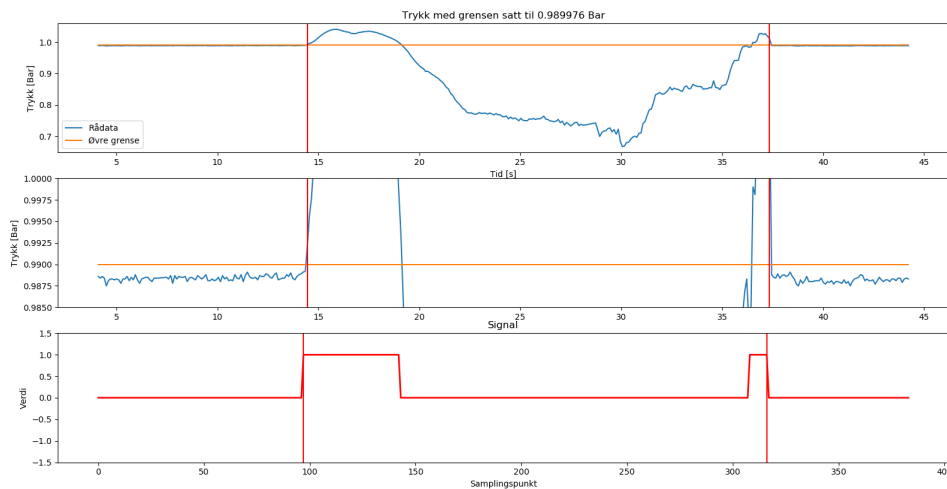
Figur 2.3: Trykkprofil med start og slutt markert med røde, vertikale linjer.

Denne grensesettingen er nøyaktig, men bare for loggfiler med et spesielt format, og den har tydelige svakheter. Dersom loggfilen er avkappet, slik at den lange perioden hvor Sensorfisk ligger ved havoverflaten ikke inngår i datasettet, vil ikke middelværdien ligge ved atmosfærisk trykk. Standardavviket kan variere kraftig og føre til at grensen blir flyttet. Denne grensesettingsmetoden er ikke være pålitelig nokk når har så liten integritet og en mer robust og pålitelig metode er nødvendig.

Den neste, og antageligvis beste, løsningen er å hardkode grensen til å ha en konstant verdi, like over atmosfærisk trykk. Dette vil forenkle implementasjonen og det vil gjøre grensen mye mer robust. Grensom som var mest robust og gav tilstrekkelige resultater var å øke grensen til 0.2% over atmosfærisk trykk, se likning 2.2.

$$\text{grense} = 1.002 \times \text{atomsfærisk trykk} \quad (2.2)$$

Dette kompliserer dataanalysen og krever manuell bestemmelse av grensen, hvor hvert datasett må grensesettes individuelt. I fig. 2.3 vises resultatet av algoritmen med trykkprofilen til Hydrolicer 1 (fig. 2.2), hvor grensen er hardkodet.



Figur 2.4: Trykkprofil med start og slutt markert med røde, vertikale linjer.

Med trykkprofilen til Hydrolicer 1 blir resultatet likt for begge metodene for grensesetting (fig. 2.5). Under vises tiden som algoritmen registrerer start og slutt for avlusningen, den interne klokken måler tiden i TimeUS (tid i mikrosekund).

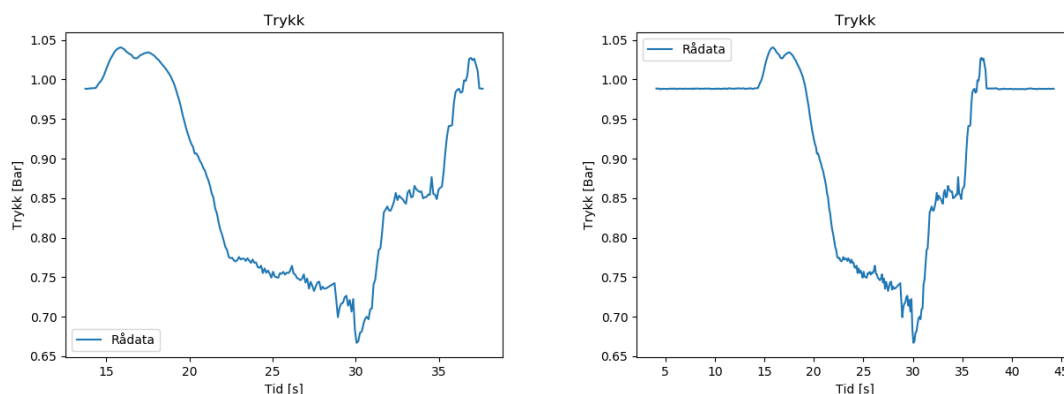
Hendelse	TimeUS	Hendelse	TimeUS
Innløp	14452373.0	Innløp	14452373.0
Utløp	37340883.0	Utløp	37340883.0

(a) Grense basert på middelvei og standardavvik.

(b) Grense satt til en konstant.

Figur 2.5: Tidspunkt for start og slutt med forskjellige grensesettingsmetoder.

Dersom samme loggfil blir brukt, men dødtiden ved havoverflaten klippes vekk oppnås det forskjellige resultater. I fig. 2.6a vises endringene i trykkprofilen, til venstre er data i intervallet [90:320], til høyre er data i intervaller [0:383] som var det originale datasettet.



(a) Trykkprofil for intervallet [90:320].

(b) Trykkprofil for intervallet [0:383].

Figur 2.6: Trykkprofiler testet med forskjellige grensesettingsmetoder.

Resultatene i fig. 2.7 viser at grensesettingsmetoden som baserte seg på middelværdi og standardavvik har nytt resultat, mens metoden med en hardkodet grenseverdi har samme resultat.

Hendelse		TimeUS	Hendelse		TimeUS
Innløp		13730374.0	Innløp		14452373.0
Utløp		37648969.0	Utløp		37340883.0

(a) Grense basert på middelværdi og standardavvik.

(b) Grense satt til en konstant.

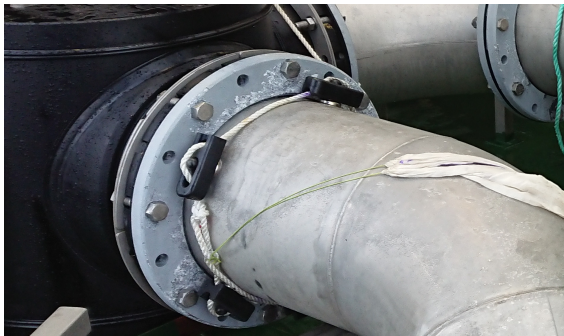
Figur 2.7: Tidspunkt for start og slutt med forskjellige grensesettingsmetoder for intervallet [90:320].

Siden metoden med en hardkodet grense hvor grenseverdien var satt til 0.2% gav de beste og mest pålitelige resultatene, ble denne metoden brukt videre. Resultatene fra deteksjon av innløp og utløp er presentert i kapittel 3.2.1.

2.2.2 Oppdage hendelser basert på magnetometer data

Ved å plassere magnetbånd rundt rørgangene i avlusningssystemet introduseres det et magnetisk felt som er målbart med magnetometer. Når sensorfisken passerer magnetbåndene blir det målt et utslag med karakteristisk profil, lignende en sinuskurve eller invertert sinuskurve (fig. 2.8b). Ved å detektere disse utslagene vil det være mulig å kunne bestemme, med stor nøyaktighet, når sensorfisken passerer gitte punkter

i avlusningsløpet. Denne metoden kan brukes for å isolere tidsrom for når Sensorfisk befinner seg i en rørgang eller en komponent, ved å plassere bånd før og etter komponenten, se fig. 2.8a



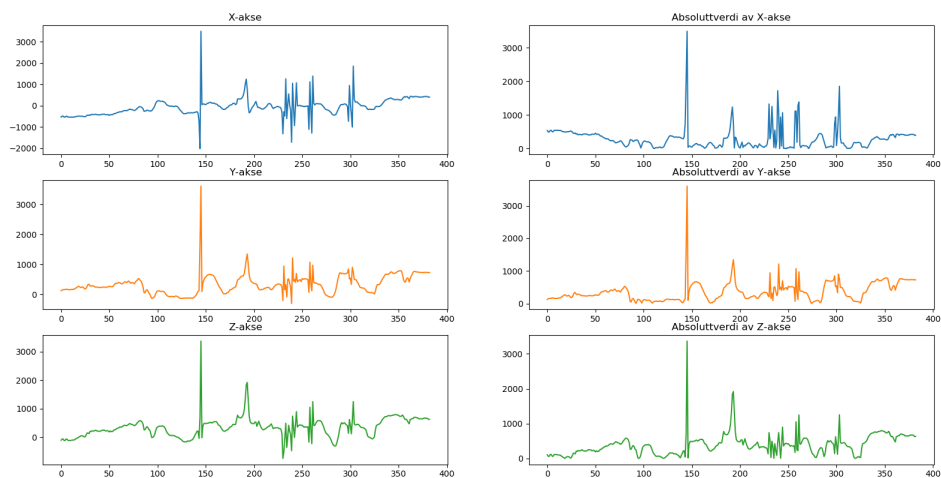
(a) Magnetbånd plassert ved bend i rørgang. (b) Karakteristisk profil for passering av magnetbånd, invertert sinus.

Figur 2.8: Montering og data fra passering av magnetbånd.

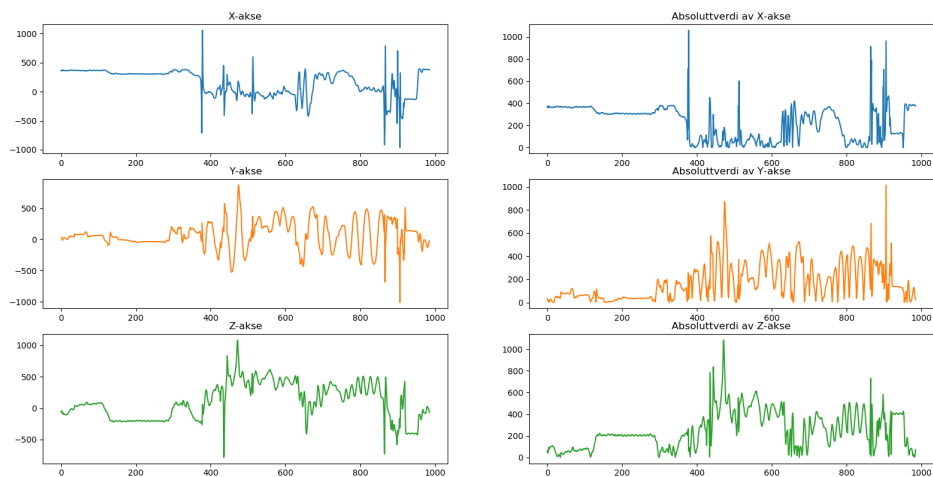
To metoder for deteksjon er testet. Den første metoden benyttet seg av en z-skår algoritme (Brakel, 2014) som leter etter topper i datasettet ved å se på hvor mange standardavvik datapunktet er fra gjennomsnittet. Den andre metoden baserer seg på korrelasjon, hvor det letes etter trender gjennom datasettet. Da letes det etter hvor mange ganger den karakteristiske profilen i fig. 2.8b, eller andre tegn på passeringer oppstår.

Ved utvikling og testing av metoder for for automatisk deteksjon av passering av magnetbånd ble både rådata og prosessert data benyttet. Rådata fra magnetometerets tre akser, X, Y og Z, absoluttverdien av hver enkelt akse, og det kvadratiske gjennomsnittet (likning 2.3) av de tre aksene ble brukt. Dataen brukt er illustrert i fig. 2.9, fig. 2.10, fig. 2.11a og fig. 2.11b, her vises analysert data fra Hydrolicer 1 og Skamik.

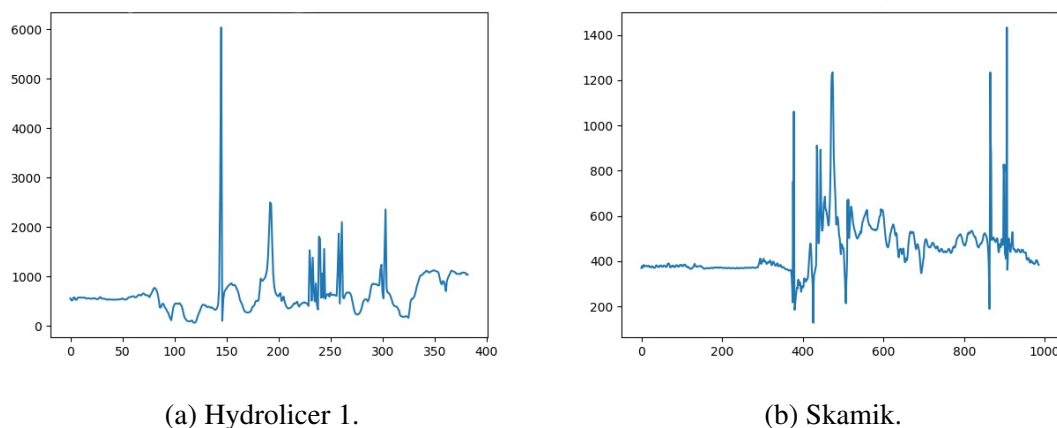
$$\text{kvadratisk gjennomsnitt} = \sqrt{X^2 + Y^2 + Z^2} \quad (2.3)$$



Figur 2.9: Rådata fra Hydrolicer 1 med 11 magnetbånd montert.



Figur 2.10: Rådata fra Skamik med 7 magnetbånd montert.



Figur 2.11: Kvadratisk gjennomsnitt av rådata.

Z-skår algoritmen er en robust og tilpasningsdyktig måte å oppdage topper i sanntid (Brakel, 2014). Selv om algoritmen ble utviklet for å analysere data i sanntid er den like pålitelig til å analysere innsmalede datasett. Algoritmen bruker et sett med datapunkter som skyves gjennom datasettet for å kalkulere et bevegelig gjennomsnitt (\bar{x}) og standardavvik (σ_x). Dersom verdien (z-skåren) er over en bestemt terskel vil datapunktet (x) bli registrert som en topp og defineres som et signal. Likningen for z-skår er oppført i likning 2.4.

$$Z = \frac{x - \bar{x}}{\sigma_x} \quad (2.4)$$

Algoritmen lar brukeren bestemme innflytelsen til signaler (registrerte topper) i utregningen av standardavviket og gjennomsnittet via en variabel kalt *influence* (i_{inf}). Denne variabelen har en verdi mellom 0 og 1, hvor 0 ignorerer signaler i utregningen av den nye grensen, 0.5 gir signaler halve innflytelsen til det normale datapunkt har og en verdi på 1 gir signaler like stor innflytelse som normale datapunkt.

Det er to andre variabler som er brukerdefinert, *lag* og *threshold*. *lag* (l) bestemmer hvor mange datapunkt som inngår i settet som skyves gjennom datasettet. *threshold* (thr) bestemmer z-skåren for at algoritmen skal definere et signal, med andre ord: hvor mange standardavvik datapunktet befinner seg fra gjennomsnittet. I likning 2.5, 2.6 og 2.7 vises beregningene for komponentene til algoritmen, der x_i er verdien til et datapunkt og s_i er verdien til et signal etter det er gjort justeringer for innflytelsen (Perkins og Heber, 2018). Pseudokode for z-skår algoritmen finnes i algoritme 2.

$$\bar{s}_i = \frac{1}{l} \sum_i^{i+l} s_i, \quad s_i = i_{inf} x_i + (1 - i_{inf}) s_{i-1} \quad (2.5)$$

$$\sigma_{s_i} = \sqrt{\frac{\sum_i^{i+l} (s_i - \bar{s}_i)^2}{l-1}}, \quad z_i = \frac{x_i - \bar{s}_{i-1}}{\sigma_{s_{i-1}}} \quad (2.6)$$

$$thr = \text{konstant}, \quad l = \text{konstant}, \quad 0 \leq i_{inf} \leq 1 \quad (2.7)$$

For pseudokoden er y en vektor med datapunkt, $\text{gns}()$ er en funksjon som regner ut gjennomsnittet, $\text{sd}()$ er en funksjon som regner ut standardavviket og $\text{abs}()$ regner ut absoluttverdien til argumentet (Brakel, 2014).

Algoritme 2 Pseudokode for z-skår algoritme.

```

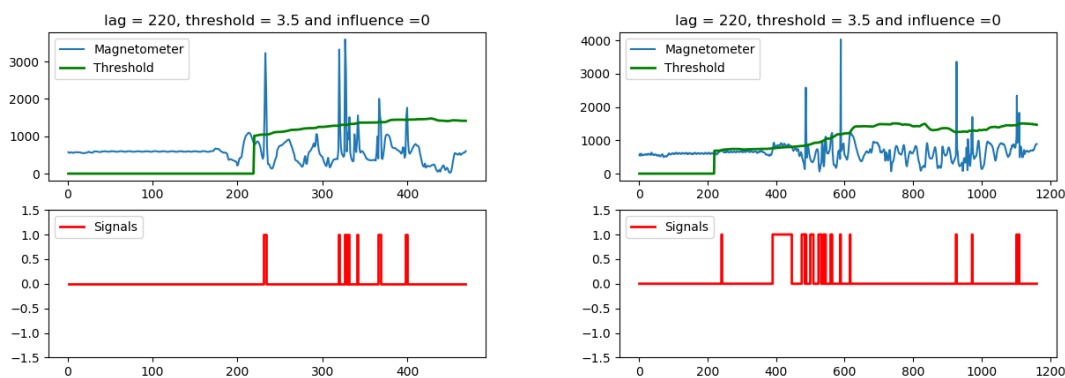
1: Definerer brukersatt variabler: lag, threshold og influence
2: Setter signaler til vektor 0,...,0 med lengden til y
3: Setter filtrertY til å være lik y
4: Setter gnsFilter til vektor 0,...,0 med lengden til y
5: Setter sdFilter til vektor 0,...,0 med lengden til y
6: Setter gnsFilter(lag) til gns(y[0:lag])
7: Setter sdFilter(lag) til sd(y[0:lag])
8: for i in range(lag, len(y)) do
9:   if abs(y[i] - gnsFilter [i-1]) > threshold * sdFilter [i-1] then
10:    if y[i] > gnsFilter [i-1] then
11:      Setter signal [i] = 1
12:    else
13:      Setter signal [i] = 0
14:      Setter filtrertY [i] to influence*y [i] + (1-influence)*filtrertY [i-1]
      % Oppdater gnsFilter og sdFilter
15:    else
16:      Setter signal [i] = 0
17:      Setter filtrertY [i] = y [i]
      % Oppdater gnsFilter og sdFilter

```

Selv om initielle tester med data generert under laboratorieforhold gav meget gode resultater, gav z-skår algoritmen ingen brukbare resultater for reell data. Det er ikke mulig å pålitelig gjenkjenne passeringen fra rådata, absoluttverdi-data eller fra det kvadratiske gjennomsnittet. Det kan være flere faktorer som fører til dette resultatet. Det kan hende at dataen er for påvirket av støy eller at toppene som skal detekteres ikke fremgår tydelig nok i datasettet.

Det er i tillegg meget tidkrevende å justere algoritmen for hvert enkelt datasett. Selv på data fra samme avlusningssystem kan det fremkomme store variasjoner i utslagene og spesielt

for data fra forskjellige systemer. Innstillinger som gir lovende resultater for et datasett er ikke overførbart til andre datasett, se fig. 2.12. Tiden brukt til justering vil overgå tiden det ville tatt å manuelt analysere datasettet for å finne passeringene. Derfor er z-skår algoritmen ikke egnet til automatisk deteksjon av passeringer av magnetbånd.



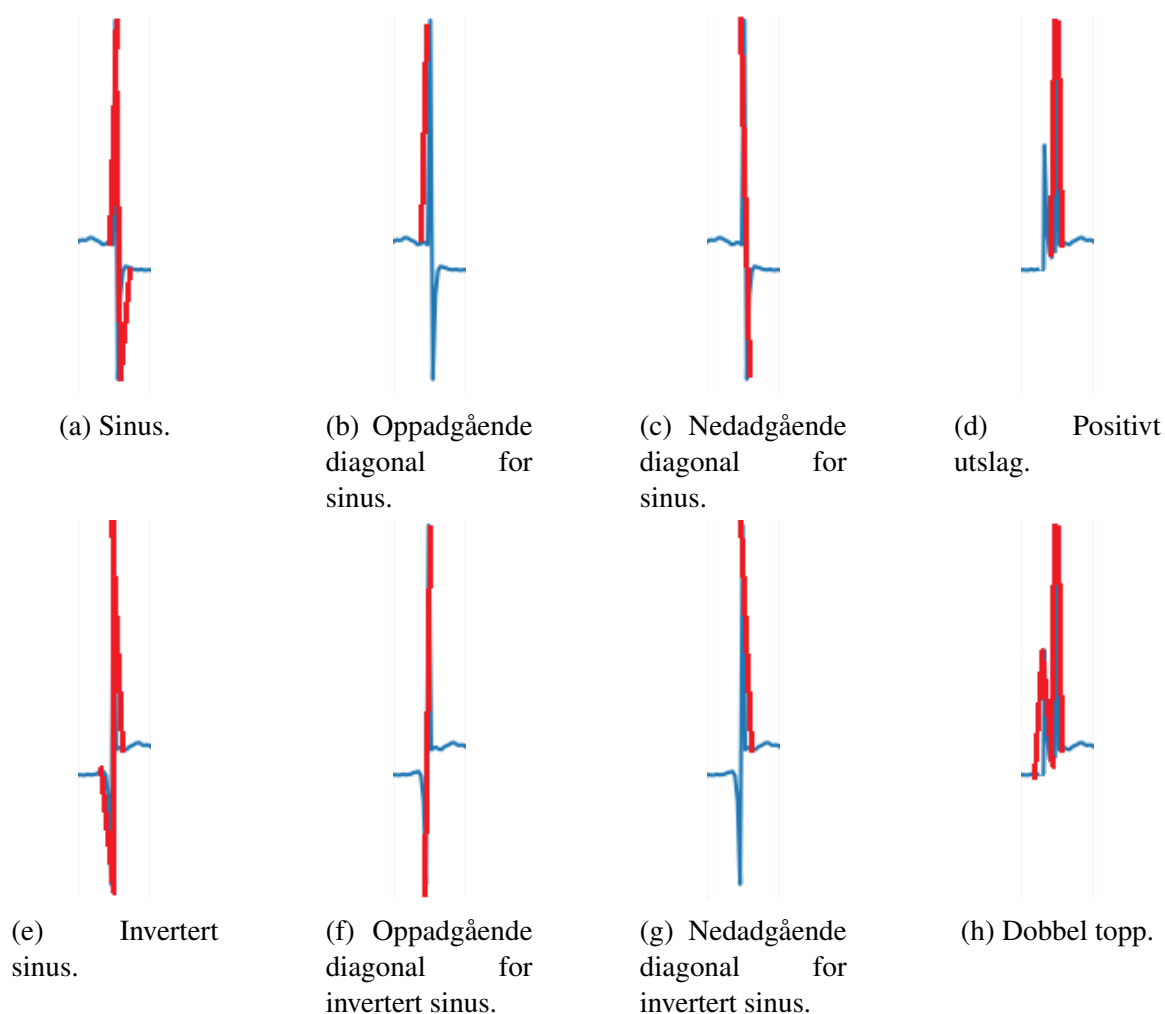
(a) Kvadratisk gjennomsnitt Hydrolicer 1.

(b) Kvadratisk gjennomsnitt Skamik.

Figur 2.12: Resultat fra z-skår algoritmen med like innstillinger for to datasett.

Siden utslagene i datasettet ikke var distinktive nokk til å oppdages basert på z-skår algoritmen blir andre metoder testet. Passeringene forbi magnetbåndene har fremdeles karakteristiske profiler som gjentar seg. Det er mulig å lete etter disse profilene ved bruk av korrelasjon. Det velges ut et datautvalg som representerer profilen til en kjent passering, deretter korreleres dette datautvalget med hele datasettet for å finne intervaller med lik profil. Når man korrelerer får man et statistisk mål på hvor mye to målbare hendelser henger sammen (Frøslie, 2018). Siden de karakteristiske profilene er så unik, er tanken at korrelasjonsverdien vil være så høy når like hendelser fremtrer i datasettet at man kan gjenkjenne en passering fra korrelasjonsverdien.

De karakteristiske profilene som ble brukt som datautvalg for korreleringen er vist i fig. 2.13. I testtabellene (tabell 2.3 og tabell 2.2) er det søkt etter profiler lik både sinus og invertert sinus for kolonnen markert "Sinus". For begge sinusprofilene er det søkt etter både oppadgående diagonaler og nedadgående diagonaler, ettersom lengden vil variere ut fra hvilken sinusprofil utvalget stammer fra, se fig. 2.13 for forklaring. Disse profilene ble valg på grunnlag av magnetometerdata fra kjente passeringer, hvor profilene var tydelige indikasjoner på passering av magnetbånd.



Figur 2.13: Karakteristiske profiler brukt til korrelering, datautvalg markert i rødt.

Korreleringen ble testet med to funksjoner, `numpy.correlate` og `scipy.stats.pearsonr`, som tar inn datautvalget og datasettet som to vektorer. Datautvalget velges manuelt og iterertes over hele datasettet.

`numpy.correlate` gir et mål på både korrelasjon til profilen og utslaget, ved å regne ut kovariansen mellom vektorene, se likning 2.8. Det betyr at dersom to passeringer har lik profil, men ulik verdi, vil korrelasjonskoeffisienten ha lavere verdi enn hvis profilene og verdiene var lik (SciPy, 2019).

$$Cov[X, Y] = E[(X - \bar{X})(Y - \bar{Y})] \quad (2.8)$$

Tabell 2.2: Testtabell for `numpy.correlate`.

Datasett	<code>numpy.correlate</code>				
	Mønster				
<i>Hydrolicer 1</i>	Sinus	Oppadgående diagonal	Nedadgående diagonal	Positivt utslag	Dobbel topp
X-akse	x	x	x	x	
Y-akse	x	x	x	x	
Z-akse	x	x	x	x	
Absoluttverdi av X-akse		x	x	x	x
Absoluttverdi av Y-akse		x	x	x	x
Absoluttverdi av Z-akse		x	x	x	x
Kvadratisk gjennomsnitt		x	x	x	x
<i>Skamik</i>					
X-akse	x	x	x	x	
Y-akse	x	x	x	x	
Z-akse	x	x	x	x	
Absoluttverdi av X-akse		x	x	x	x
Absoluttverdi av Y-akse		x	x	x	x
Absoluttverdi av Z-akse		x	x	x	x
Kvadratisk gjennomsnitt		x	x	x	x

`pearsonr` normaliserer korrelasjonskoeffisienten til å ha en verdi mellom -1 og +1 ved å dividere kovariansen på produktet til standardavviket for vektorene, se likning 2.9. En verdi på null vil indikere ingen korrelasjon og en på 1 indikerer full korrelasjon. Positiv korrelasjon indikerer at når det ene vektorer øker, så øker den andre vektoren også. Negativ korrelasjon indikerer at når den ene vektoren øker, så synker den andre, altså en negativ sammenheng mellom vektorene.

$$\rho_{X,Y} = \frac{E[(X - \bar{X})(Y - \bar{Y})]}{\sigma_X \sigma_Y} \quad (2.9)$$

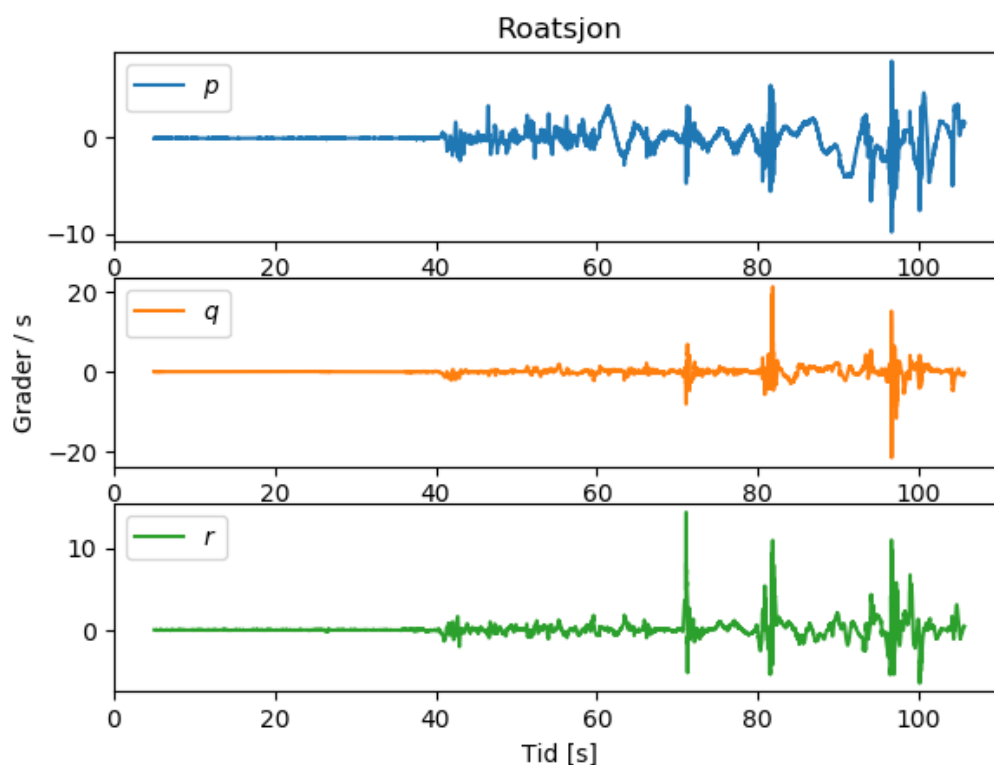
Tabell 2.3: Testtabell for `pearsonr`.

<u>Datasett</u>	<u>pearsonr</u>				
	Mønster				
<i>Hydrolicer 1</i>	Sinus	Oppadgående diagonal	Nedadgående diagonal	Positivt utslag	Dobbel topp
x	x	x	x	x	
y	x	x	x	x	
z	x	x	x	x	
Absoluttverdi av x		x	x	x	x
Absoluttverdi av y		x	x	x	x
Absoluttverdi av z		x	x	x	x
Kvadratisk gjennomsnitt		x	x	x	x
<i>Skamik</i>					
x	x	x	x	x	
y	x	x	x	x	
z	x	x	x	x	
Absoluttverdi av x		x	x	x	x
Absoluttverdi av y		x	x	x	x
Absoluttverdi av z		x	x	x	x
Kvadratisk gjennomsnitt		x	x	x	x

Resultatene fra korreleringen er presentert i kapittel 3.2.2.

2.2.3 Oppdage hendelser baser på gyroskop data

Gyroskopet måler rotasjonshastigheten til Sensorfisk i tre akser, rotasjon om x-aksen kalt roll p , rotasjon om y-aksen kalt pitch q og rotasjon om z-aksen kalt yaw r (SNAME, 1950). I fig. 2.14 er rotasjonsdata fra Skamik presentert, dette datasettet er representativt for hvordan rotasjonsdata forspeiles gjennom avlusning for alle systemene. Sensorfisk føres inn ved innsuget til avluserenheter uten noen rotasjon. Når Sensorfisk pumpes gjennom vannstrømmen oppstår rotasjon når den kommer i kontakt med komponenter eller endringer i strømningsretning. Den rotasjonen som oppstår av fysisk kontakt mellom avluserenheten og Sensorfisk er en indikasjon for støt.

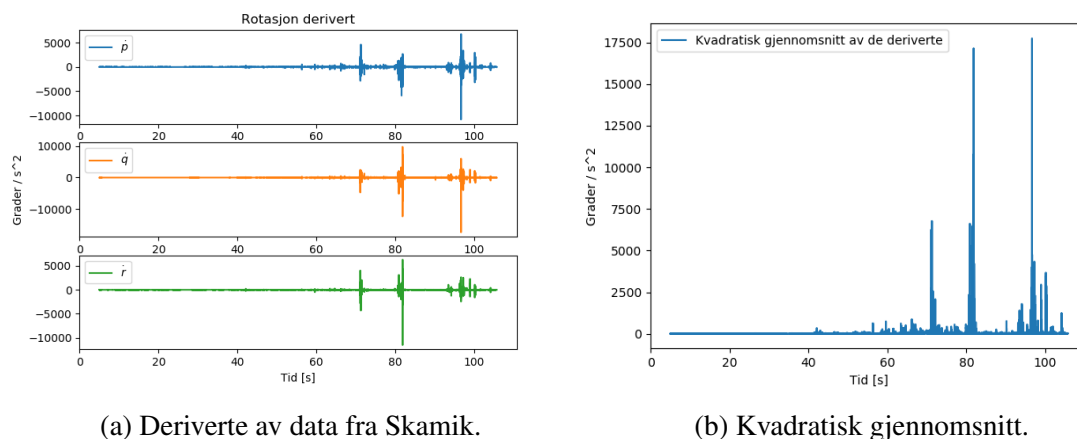


Figur 2.14: Rådata fra Skamik.

For alle støt gjelder loven om at summen av legemenes bevegelsesmengde er den samme etter støtet, som før støtet (Ormestad, 2017). Siden avlusningsenheten er å se på som et stasjonært objekt, som betyr at det ikke blir flyttet på av Sensorfisk, vil all bevegelsesmengden bli bevart i Sensorfisk. Da vil støtet resultere i en ending i rotasjonshastigheten, f.eks. ved at retningen endres eller ved at hastigheten øker. Ved å derivere rotasjonshastigheten for hver akse finnes denne endringen som er en direkte indikasjon for støt, resultatet blir da rotasjonsakselerasjonen (fig. 2.15a).

Det er ønskelig å se på utslaget av rotasjonsakselerasjonen. Både for å få en indikasjon for hvor stor endring som forårsakes av støtene til bruk i grensesetting og definisjon av støtene, men også for å ha ett datasett å analysere istedenfor for tre. Den euklidske normen gir lengden for akselerasjonsvektoren, $\alpha_{rot} = [\dot{p}, \dot{q}, \dot{r}]^T$, i form av en skalar (likning 2.10). Resultatet av den euklidske normen (omtalt som det kvadratiske gjennomsnittet i figurene) for det nevnte datasettet kan ses i fig. 2.15b.

$$\|\alpha_{rot}\| = \sqrt{\dot{p}^2 + \dot{q}^2 + \dot{r}^2} \quad (2.10)$$



Figur 2.15: Prosessert data fra gyroskop.

Normen har varierende utslag, selv på samme system. De høyeste utslagene varierer fra $40.000^\circ/s^2$ til $8.000^\circ/s^2$ for forskjellige avlusningssystemer. For samme type system var den største variasjon fra $35.000^\circ/s^2$ til $20.000^\circ/s^2$, de nøyaktige verdiene kan ses i kapittel 3.2.3. Denne variasjonen skaper noen utfordringer ved grensesetting som omtales senere.

Først er det ønskelig å finne varigheten til støtene. Når Sensorfisk befinner seg i avlusningssystemet vil vannet dempe rotasjonen, dette kan modelleres ved bruk av Newtons 2. lov: $\sum F = ma$ (Mason, 2001). De eneste kreftene som virker på systemet er dempningen fra vannet, beskrevet av en dempningsfaktor D multiplisert med hastigheten, pluss et pådrag u (likning 2.11).

$$\begin{aligned} m\alpha_p &= -Dp + u \\ m\alpha_q &= -Dq + u \\ m\alpha_r &= -Dr + u \end{aligned} \quad (2.11)$$

Fra definisjonen av rotasjonsakselerasjonen (kapittel 1.6.2, likning 1.4) kan det settes opp differensiallikninger for modellen. Resultatet blir en første ordens lineære differensiallikninger (likning 2.12) (Chen, 2014).

$$\begin{aligned} m\dot{p} &= -Dp + u \\ m\dot{q} &= -Dq + u \\ m\dot{r} &= -Dr + u \end{aligned} \quad (2.12)$$

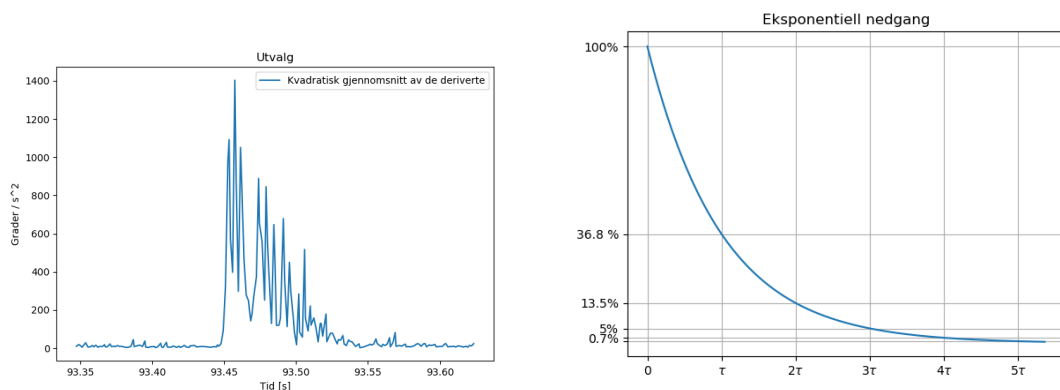
Ved å utføre Laplace transformasjon og løse for den transformerte fremkommer transferfunksjonen som bestemmer modellen (Egeland og Gravdal, 2002).

Transferfunksjonene er oppført på standardform i likning 2.13. Anntar at starthastigheten er null: $p(0) = 0$.

$$\begin{aligned} s\hat{p} &= -\frac{D}{m}\hat{p} + \hat{u} & \Rightarrow & \frac{\hat{p}}{\hat{u}} = \frac{1}{s + \frac{D}{m}} \\ s\hat{q} &= -\frac{D}{m}\hat{q} + \hat{u} & \Rightarrow & \frac{\hat{q}}{\hat{u}} = \frac{1}{s + \frac{D}{m}} \\ s\hat{r} &= -\frac{D}{m}\hat{r} + \hat{u} & \Rightarrow & \frac{\hat{r}}{\hat{u}} = \frac{1}{s + \frac{D}{m}} \end{aligned} \quad (2.13)$$

For et støt vil pådraget u karakteriseres som en impuls, der impulspådraget er $u = 1$ (Kumar og Arya, 2017). Gjennom invers Laplace transformasjon finner vi løsningen til transferfunksjonene (likning 2.14).

$$\begin{aligned} p &= e^{-\frac{D}{m}t} \\ q &= e^{-\frac{D}{m}t} \\ r &= e^{-\frac{D}{m}t} \end{aligned} \quad (2.14)$$



(a) Utvalg av datasettet som viser et støt.

(b) Impulsrespons for et første ordens lineært tidsinvariant system der $\tau = \frac{D}{m}$.

Figur 2.16: Sammenlikning av støt og eksponentiell nedgang for et lineært tidsinvariant system.

I fig. 2.16 vises likheten mellom et støt i datasettet og impulsresponsen for transferfunksjonene (likning 2.13). Dette bekrefter at beregningene og modellen stemmer.

Fra fig. 2.16b kommer det frem at etter fem tidskonstanter (τ) har støtet blitt redusert til 0.7% av utslaget, og støter er å anse som over. Dette kan vi lete etter i datasettet for å finne når støtet er over og hvor lang varighet støtet hadde.

Funksjonen `scipy.signal.argrelextrema` brukes til å finne lokale toppunkt med 50 datapunkt på hver side. Grunnen til at det skal være 50 datapunkt på hver side er at samplingsfrekvens er på 1ms, og med støt som har en varighet på ca 0.1 sekund vil det tilsvare 100 datapunkter for hvert støt. Når toppunktene er lokalisert søkes det etter neste datapunkt som har et utslag på under 0.7% av toppen, dette vil markere slutten på støtet. Det kommer frem i fig. 2.16a at et støt begynner før det lokale toppunktet inntreffer. Det søkes derfor etter første datapunkt før toppunktet som har et utslag på under 0.7% av toppen, dette vil markere starten på støtet. Varigheten for støtet kan enkelt regnes ut nå som differansen mellom slutt- og starttiden.



(a) Utvalg av datasettet med intervallet for støtet markert.

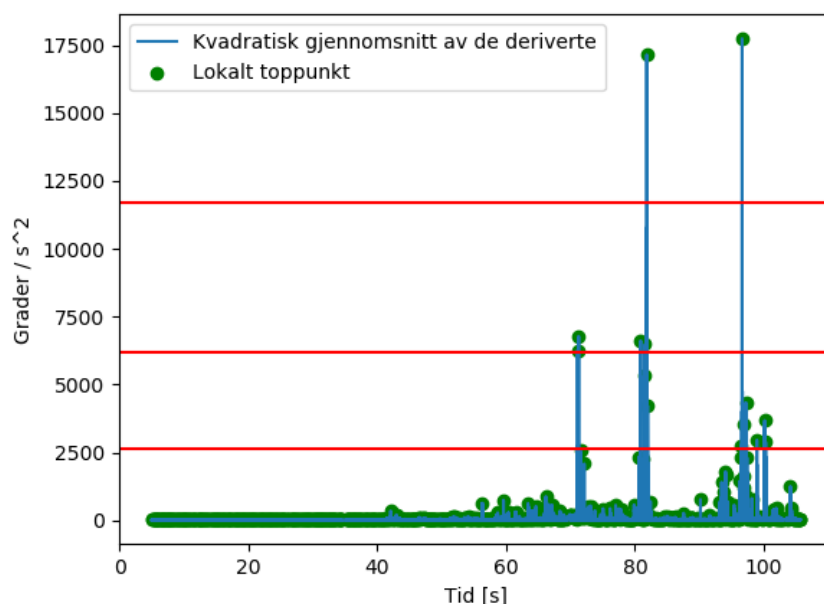
(b) To støt, ett med lengre og ett med kortere varighet.

Figur 2.17: Sammenlikning av varigheten for støt.

I fig. 2.17a vises resultatet av dataanalysen for å finne startpunkt og sluttunkt for støt. Fra fig. 2.17b er det tydelig at støtene har ulik varighet selv med likt utslag, derfor er det nødvendig å utføre analyse av hvert enkelt støt. Starten detekteres når akselerasjonen begynner å øke og slutten markeres når impulsresponsen har dødd ut. Denne metoden var ikke robust nok dersom to støt inntraff innfor såpass kort tid at det første støtet ikke rakk å dø ut. Dette ble forbedret med en ny betingelse; dersom det ikke ble oppdaget et punkt med utslag under 0.7% av toppen innen 100 datapunkt, ble det søkt etter et datapunkt med utslag på under 5% av toppen. Dette tilsvarer at støtet har en varighet på tre tidskonstanter i stedet for fem. Med dette tillegg var det ingen overlapp av støtene som ble oppdaget av algoritmen og både start- og sluttunkt ble oppdaget.

Det ble testet flere metoder for karakterisering av støtene. Først ble det testet med grenser som baserte seg på utslaget til støtet. Da ble grensene satt til 66%, 35% og 15% av utslaget, hvor alt over 66% ble definert som et stort støt. Støt med utslag mellom 66%

og 35% ble definert som middels støt og støt med utslag mellom 35% og 15% ble definert som små støt. Alt under 15% var å anse som små perturbasjoner på Sensorfisk uten noen betydelig form for mekanisk belastning (fig. 2.18). Denne grensesettingen ga et godt bilde av hvordan fordelingen av støtene var for de enkelte datasettene, men den var ikke egnet til sammenlikning mellom forskjellige datasett. Derfor ble det besluttet å bruke de samme grensene for alle datasettene. Grenseverdier som ble valg for genereringen av resultatene var $8.000^{\circ}/s^2$, $4.000^{\circ}/s^2$ og $2.000^{\circ}/s^2$, klassifiseringen av disse grensene er presentert i tabell 3.1.



Figur 2.18: Resultat fra `argrelextrema`, lokale toppunkt markert med grønne punkt og grenseverdier basert på maks utslag markert med røde, horisontale linjer.

Resultater som starttiden, slutt-tiden og varigheten for støtene blir lagret i en database. Slik vil de være lett tilgjengelig for videre prosessering og analyse. Fremstillingen av resultatene er presentert i kapittel 3.2.3, der fremkommer trykkprofilen sammen med støtprofilen og normen (det kvadratiske gjennomsnittet) for rotasjonsakselerasjonen. Det er nødvendig å presentere trykkprofilen sammen med støtprofilen for å kunne gi et tydelig bilde av når støt oppstår. Ved å se dette i sammenheng med normen skapes det et mer tydelig bilde av utslaget til støtene under avlusningsløpet.

2.3 Estimering

2.3.1 Kalmanfilter

Et Kalmanfilter er en optimal estimator for tilstandsestimering av lineære systemer (Kalman, 1960). For å kunne håndtere ulineære systemer må et utvidet Kalmanfilter benyttes. For et utvidet Kalman filter er fremgangsmåten lik den for et Kalmanfilter, men det ulineære systemet må lineariseres om det nåværende estimatet ved bruk av en første ordens Taylorrekke (Lie, 2018). For et generelt system kan følgende utvidet Kalmanfilter defineres (Van der Merwe, 2004; Lie, 2018):

Filteret initialiseres ved

$$\hat{\mathbf{x}}_0 = E[\mathbf{x}_0] \quad (2.15)$$

$$\mathbf{P}_{\mathbf{x}_0} = E[(\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)^\top] \quad (2.16)$$

$$\mathbf{R}_v = E[(\mathbf{v} - \bar{\mathbf{v}})(\mathbf{v} - \bar{\mathbf{v}})^\top] \quad (2.17)$$

$$\mathbf{R}_n = E[(\mathbf{n} - \bar{\mathbf{n}})(\mathbf{n} - \bar{\mathbf{n}})^\top] \quad (2.18)$$

1. Første steg er å estimere tilstanden:

- Beregner Jacobimatrissene for modellen

$$\mathbf{F}_{\mathbf{x}_k} = \nabla_{\mathbf{x}} \mathbf{f}(\mathbf{x}, \bar{\mathbf{v}}, \mathbf{u}_k) |_{\mathbf{x}=\hat{\mathbf{x}}_{k-1}} \quad (2.19)$$

$$\mathbf{G}_v = \nabla_v \mathbf{f}(\hat{\mathbf{x}}_{k-1}, \mathbf{v}, \mathbf{u}_k) |_{\mathbf{v}=\bar{\mathbf{v}}} \quad (2.20)$$

- Oppdaterer estimatet og kovariansmatrisen

$$\hat{\mathbf{x}}_k^- = \mathbf{f}(\hat{\mathbf{x}}_{k-1}^-, \bar{\mathbf{v}}, \mathbf{u}_k) \quad (2.21)$$

$$\mathbf{P}_{\mathbf{x}_k}^- = \mathbf{F}_{\mathbf{x}_k} \mathbf{P}_{\mathbf{x}_k} \mathbf{F}_{\mathbf{x}_k}^\top + \mathbf{G}_v \mathbf{R}_v \mathbf{G}_v^\top \quad (2.22)$$

2. Andre steg er å oppdatere estimatet med målingene:

- Beregner Jacobimatrissen for målingene

$$\mathbf{H}_{\mathbf{x}_k} = \nabla_{\mathbf{x}} \mathbf{h}(\mathbf{x}, \bar{\mathbf{n}}) |_{\mathbf{x}=\hat{\mathbf{x}}_k^-} \quad (2.23)$$

$$\mathbf{D}_n = \nabla_n \mathbf{h}(\hat{\mathbf{x}}_k^-, \mathbf{n}) |_{\mathbf{n}=\bar{\mathbf{n}}} \quad (2.24)$$

- Oppdaterer estimatet med de siste målingene

$$\mathbf{K}_x = \mathbf{P}_{\mathbf{x}_k}^- \mathbf{H}_{\mathbf{x}_k}^\top (\mathbf{H}_{\mathbf{x}_k} \mathbf{P}_{\mathbf{x}_k}^- \mathbf{H}_{\mathbf{x}_k}^\top + \mathbf{D}_n \mathbf{R}_n \mathbf{D}_n^\top)^{-1} \quad (2.25)$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k [\mathbf{y}_k - \mathbf{h}(\hat{\mathbf{x}}_k^-, \mathbf{n})] \quad (2.26)$$

$$\mathbf{P}_{\mathbf{x}_k}^- = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_{\mathbf{x}_k}) \mathbf{P}_{\mathbf{x}_k}^- \quad (2.27)$$

2.3.2 Variabel estimering

Akselerometrene måler lineærakselerasjonen $a_{m,i}$. Denne målingen består av følgende komponenter:

$$a_{m,i} = a_i + a_{g,i} + a_{n,i}, \quad i \in \{u, v, w\} \quad (2.28)$$

Der a_i er den faktiske lineærakselerasjonen, $a_{g,i}$ er innvirkningen tyngdeakselerasjonen har på målingen, og $a_{n,i}$ er målestøyen sin innvirkning på målingen. For å kunne estimere lineærhastigheten og videre posisjonen er det ønskelig å fjerne innvirkningen fra tyngdeakselerasjonen og støy på målingene.

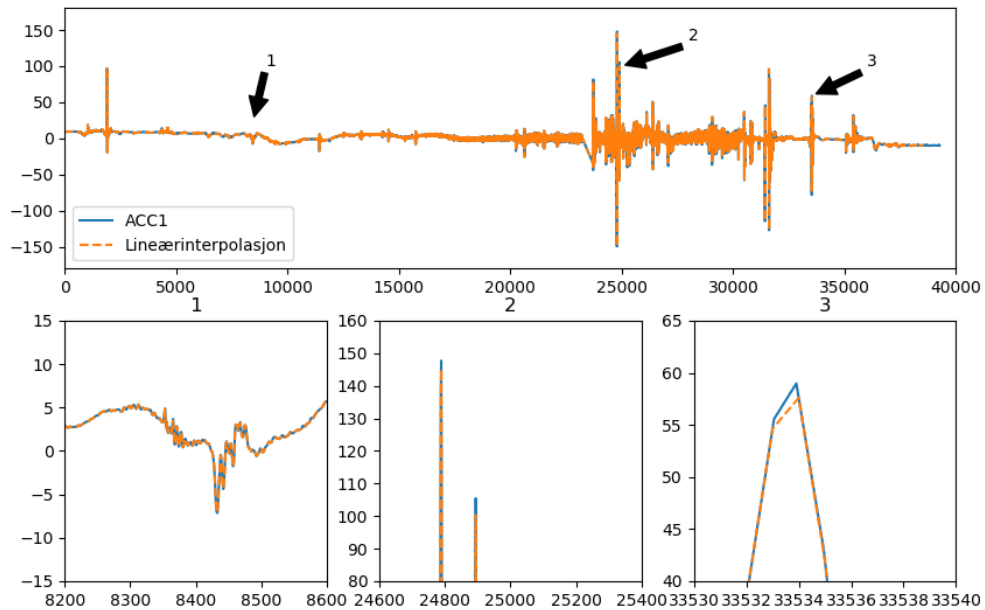
Tyngdeakselerasjonen $a_{g,i}$, er et produkt av tyngdeakselerasjonen og dens virkning i de tre retningene Sensorfisk måler i. Denne virkningen vil variere etter orienteringen til Sensorfisk, med full virkingen dersom aksene som måles er vertikal iht. jordoverflaten og null virkning dersom den målte aksene er horisontal iht. jordoverflaten. Tyngdeakselerasjonen i forhold til avlyserens koordinatsystem $\{n\}$ er $g_n = [0, 0, 9.81]^T$ (Lovren og Pieper, 1997). For å representere tyngdeakselerasjonen relativt til aksene for Sensorfisk utføres det koordinatrotasjoner, presentert i kapittel 1.6.1.

$$a_{g,i} = \mathbf{R}_n^b(\Theta_{nb})g_n, \quad i \in \{u, v, w\} \quad (2.29)$$

Resultatet fra likning 2.29 kan trekkes fra $a_{m,i}$ for å fjerne innvirkning. Det som gjenstår å eliminere er målestøyen som kan antas til å være hvit støy (Jiang, He og Han, 2015) Ved å bruke et lavpass filter blir innvirkningen av målestøyet redusert og den faktiske lineærakselerasjonen a_i er funnet (Rahni og Yahya, 2007). Fra dette kan hastigheten estimeres gjennom integrasjon, og posisjonen estimeres gjennom dobbelintegrering av akselerasjonen.

Kapittel 3. Resultat

3.1 Interpolering



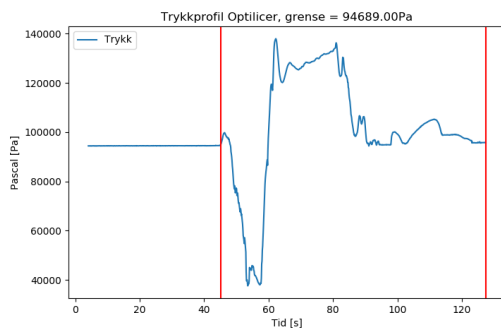
Figur 3.1: Resultat fra lineærinterpolasjon med tidsskritt på 1 ms.

Interpoleringen gir et godt estimat for akselerasjonen, kun ved store utslag oppstår det mindre avvik (fig. 3.1).

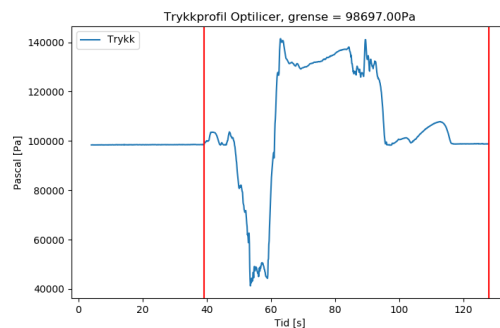
3.2 Autodeteksjon

3.2.1 Automatisk deteksjon av innløp og utløp

Tidspunktene for innløp og utløp er automatisk oppdaget og markert med røde vertikale streker. Grensen som brukes til detekteringen er notert over plottet. Denne grensen er satt til 1.002 ganger trykket ved havoverflaten.

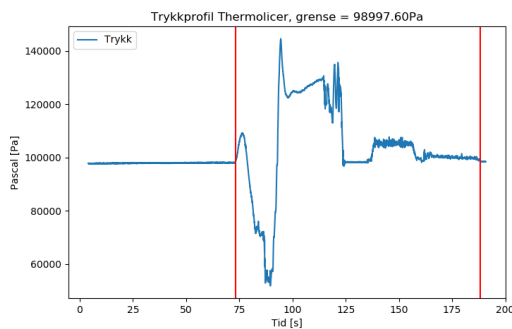


(a) Test 1.

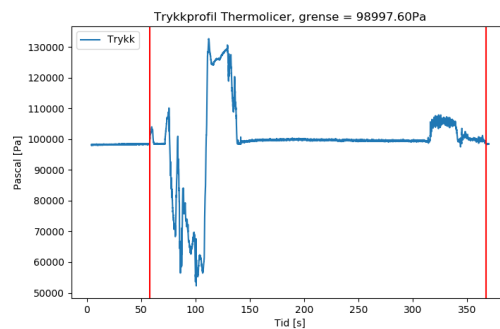


(b) Test 2.

Figur 3.2: Resultat av automatisk deteksjon av innløp og utløp for Optilicer.

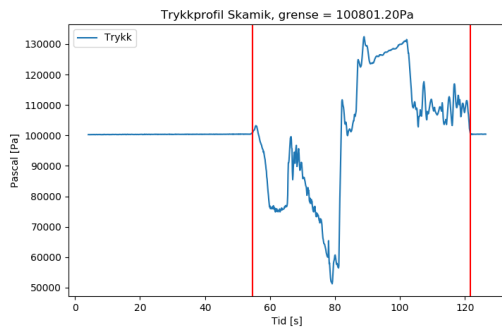


(a) Test 1.

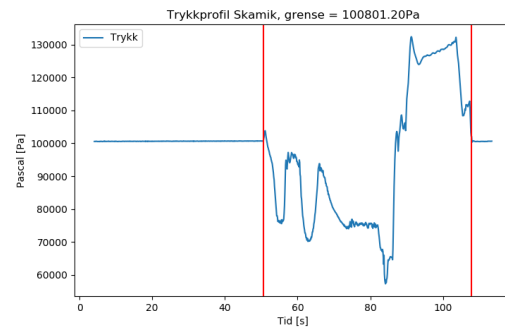


(b) Test 2.

Figur 3.3: Resultat av automatisk deteksjon av innløp og utløp for Thermolicer.

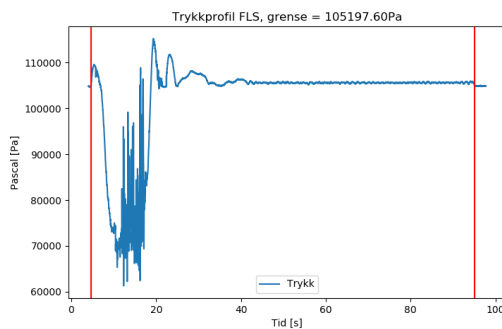


(a) Test 1.

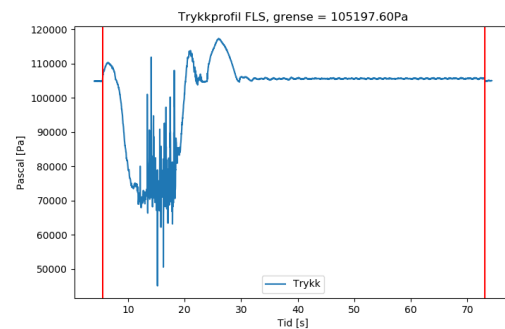


(b) Test 2.

Figur 3.4: Resultat av automatisk deteksjon av innløp og utløp for Skamik.



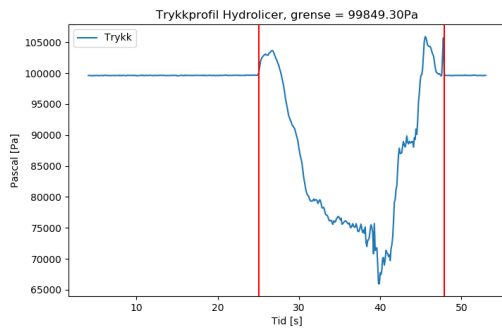
(a) Test 1.



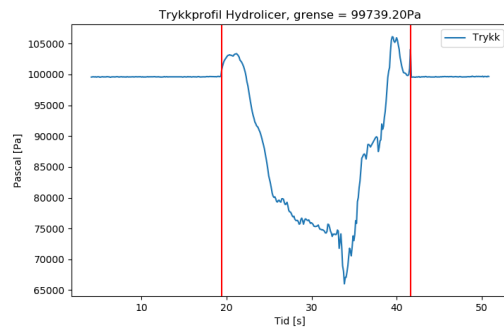
(b) Test 2.

Figur 3.5: Resultat av automatisk deteksjon av innløp og utløp for FLS.

Under avlusning med FLS ble fisken liggende i vannet etter endt avlusning (fig. 3.5). Dette datasettet ble innhentet før rutinene hvor Sensorfisk skulle ligge ved overflaten i ca. 60 sekunder før innløp og hentes opp å slås av rett etter utløpet ble innført.

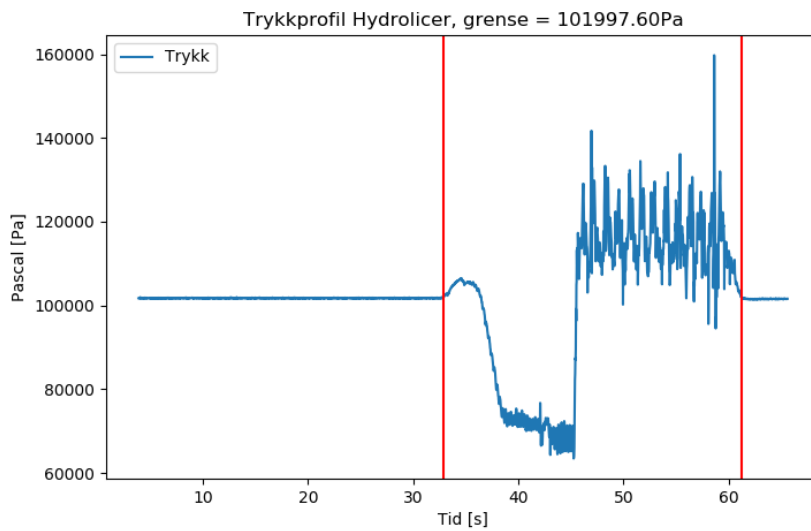


(a) Test 1.



(b) Test 2.

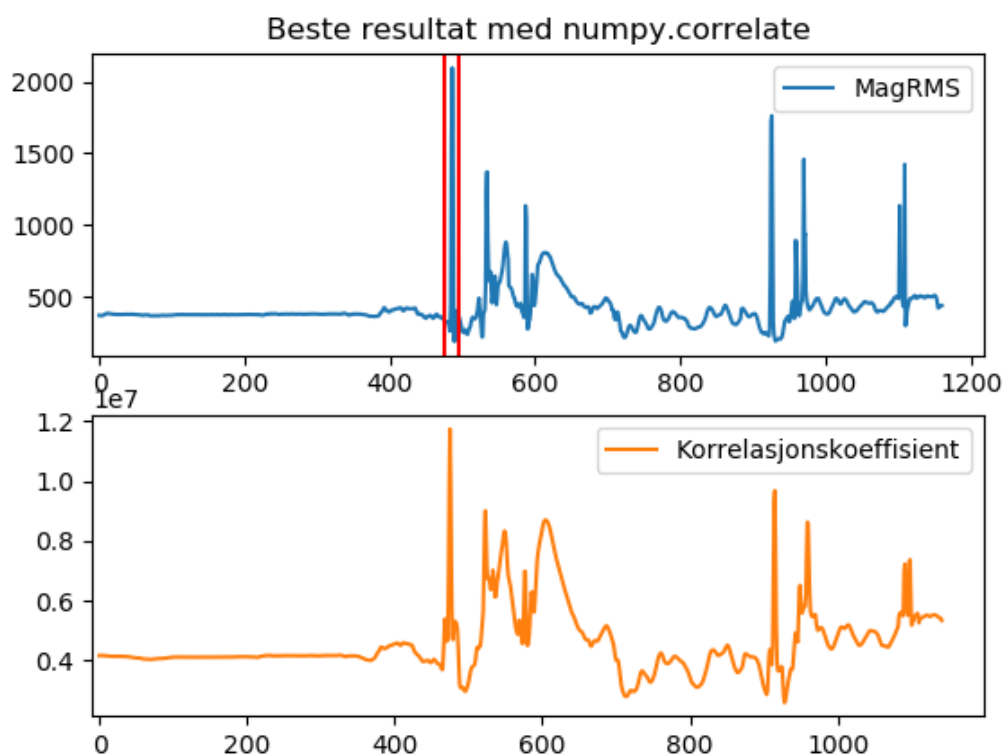
Figur 3.6: Resultat av automatisk deteksjon av innløp og utløp for Hydrolicer 1.



Figur 3.7: Resultat av automatisk deteksjon av innløp og utløp for Hydrolicer 2 test 1.

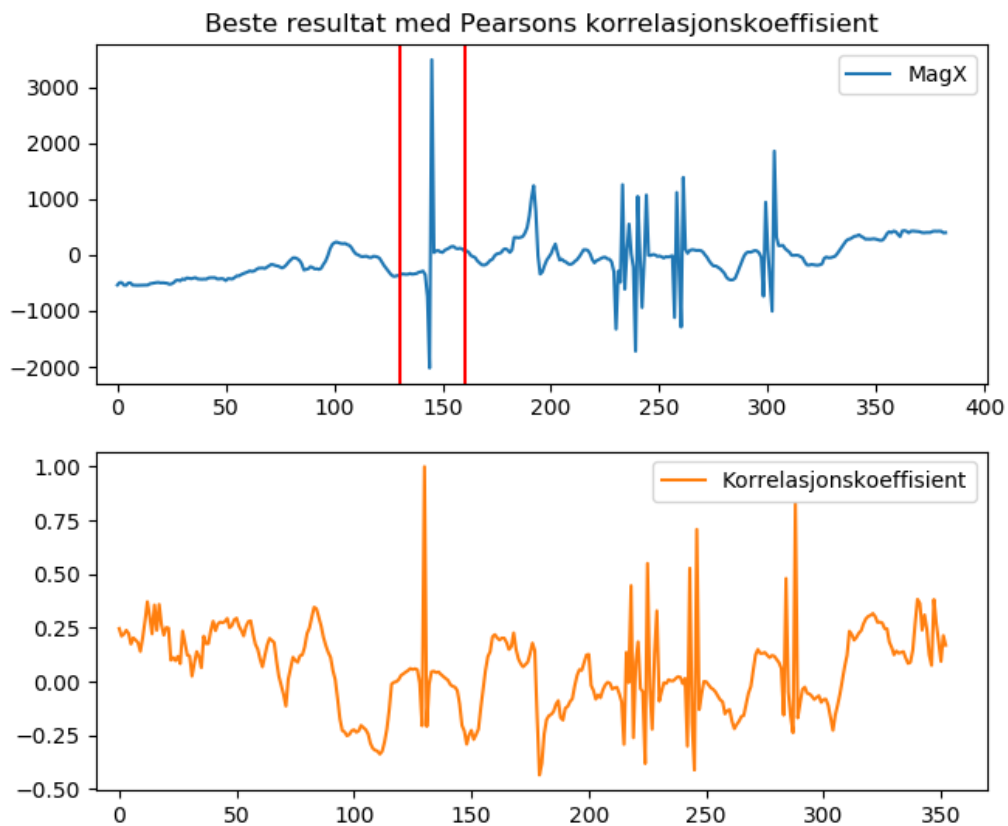
3.2.2 Automatisk deteksjon av passering av magnetbånd

Det ble utført mange tester i et forsøk på å oppdage passeringer av magnetbånd, se testtabellene tabell 2.2 og tabell 2.3. Kun de beste resultatene fra de to forskjellige korrelasjonsmetodene blir fremstilt.



Figur 3.8: Resultat fra `numpy.correlate` med data fra Skamik, området innfor de røde merkene er datautvalget.

Det beste resultatet ved bruk av `numpy.correlate` (fig. 3.8) kom fra å søke etter positivt utslag (fig. 2.13) i prosessert data fra Skamik, det kvadratiske gjennomsnittet ble benyttet. Det var montert 7 magnetbånd på avlusningssystemet, men fra resultatet ble det oppdaget 8 utslag som indikerer passeringer.



Figur 3.9: Resultat fra `pearsonr` med data fra Hydrolicer 1, området innfor de røde merkene er datautvalget.

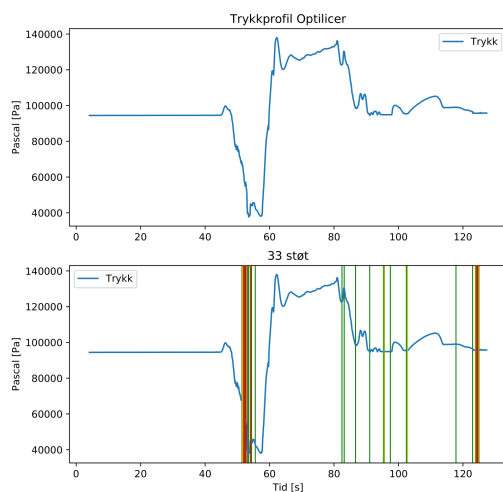
Det beste resultatet ved bruk av `pearsonr` (fig. 3.9) kom fra å søke etter invertert sinus (fig. 2.13) i rådata fra Hydrolicer 1, x-retning ble benyttet. Det var montert 11 magnetbånd på avlusningssystemet, men til tross for dette kommer det frem 8 tydelige utslag på magnetometeret. Resultatet fremhever 8 utslag som indikerer passeringer.

3.2.3 Automatisk støtdeteksjon

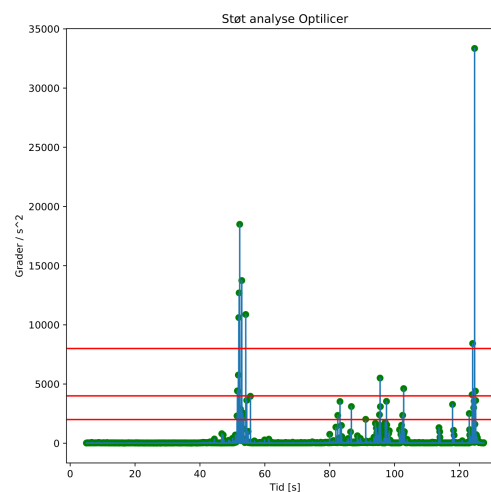
Tester på Hydrolicer 2 ble gjennomført med Fisk1 på 3kg, mens alle de øvrige testene ble gjennomført med Fisk2 på 6 kg. Grensene brukt til automatisk deteksjon og klassifisering av støtene er presentert i tabell 3.1. Identifikasjon av hvor i avlusningsystemene støtene oppstår er basert på trykkprofilen og kjennskap til systemene (Lie, 2018).

Tabell 3.1: Klassifiseringen av støt brukt til generering av resultater.

Grense [$^{\circ}/s^2$]	Klassifisering	Illustrasjon
Over 8.000	Stort støt	Rød vertikal linje
Mellom 8.000 og 4.000	Middels støt	Gul vertikal linje
Mellom 4.000 og 2.000	Lite støt	Grønn vertikal linje

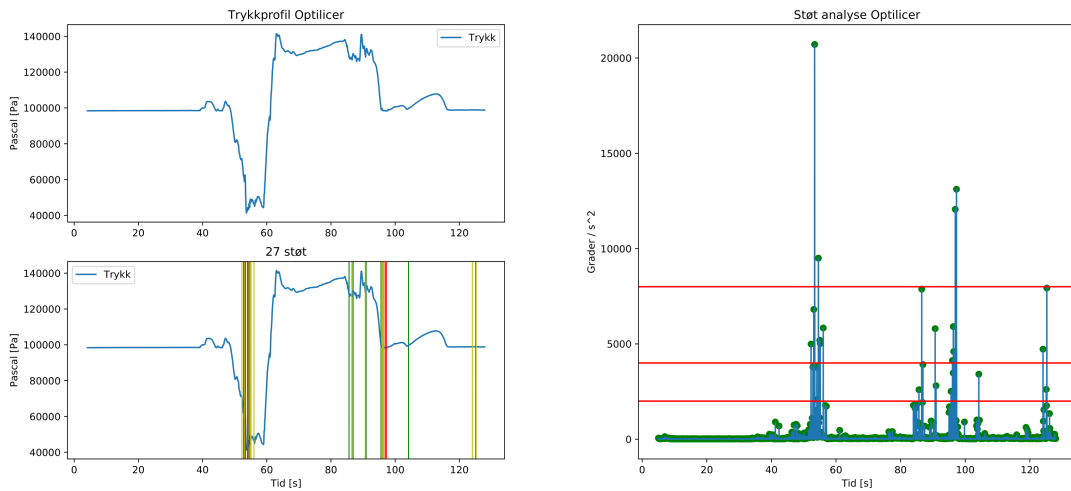


(a) Trykkprofil og støtprofil.



(b) Støt analyse.

Figur 3.10: Resultat av automatisk støtdeteksjon for Optilicer test 1.

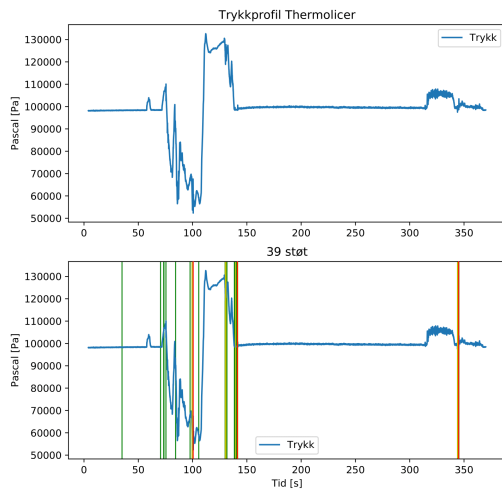


(a) Trykkprofil og støtprofil.

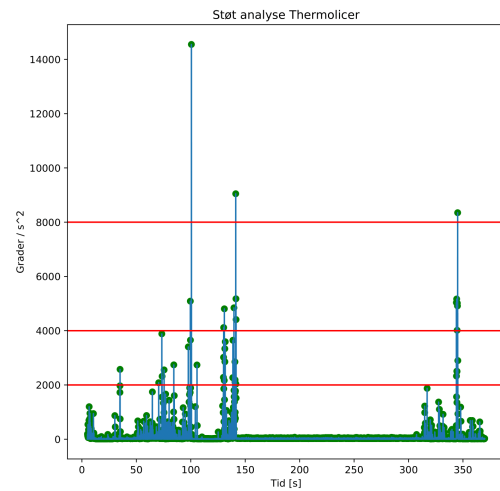
(b) Støt analyse.

Figur 3.11: Resultat av automatisk støtdeteksjon for Optilicer test 2.

Testene fra Optilicer (fig. 3.10 og fig. 3.2a) har lik trykkprofil og støtprofil, men det er en betydelig differanse i utslag. Test 1 har maksimalt utslag på ca. $34.000^\circ/s^2$ og test 2 har maksimalt utslag på ca. $21.000^\circ/s^2$. Fra trykkprofilen kommer det frem at de fleste støtene oppstår ved tre tidspunkt under avlusning. Den første samlingen av støt er i rørgangen inn til trykk-vakuumpumpen som oppstår ved tiden 55s for test 1 og test 2. Den andre samlingen støt oppstår når Sensorfisk passerer varmebehandlingen og kommer i kontakt med skovlhjul, ved tiden 80s - 100s for test 1 og test 2. Den tredje og siste samlingen av støt oppstår når Sensorfisk kommer fra avsiling etter varmebehandlingen og faller ned en høyde på ca. 40 cm til rørgangen som skal føre lasken tilbake til merd.

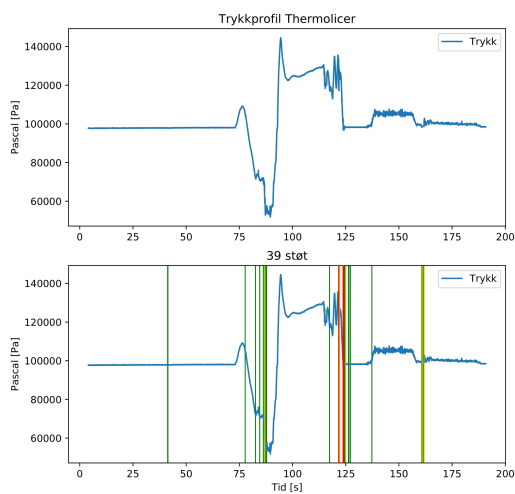


(a) Trykkprofil og støtprofil.

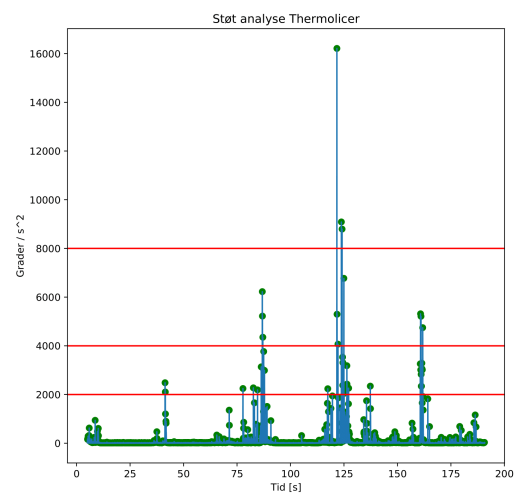


(b) Støt analyse.

Figur 3.12: Resultat av automatisk støtdeteksjon for Thermolicer test 1.



(a) Trykkprofil og støtprofil.



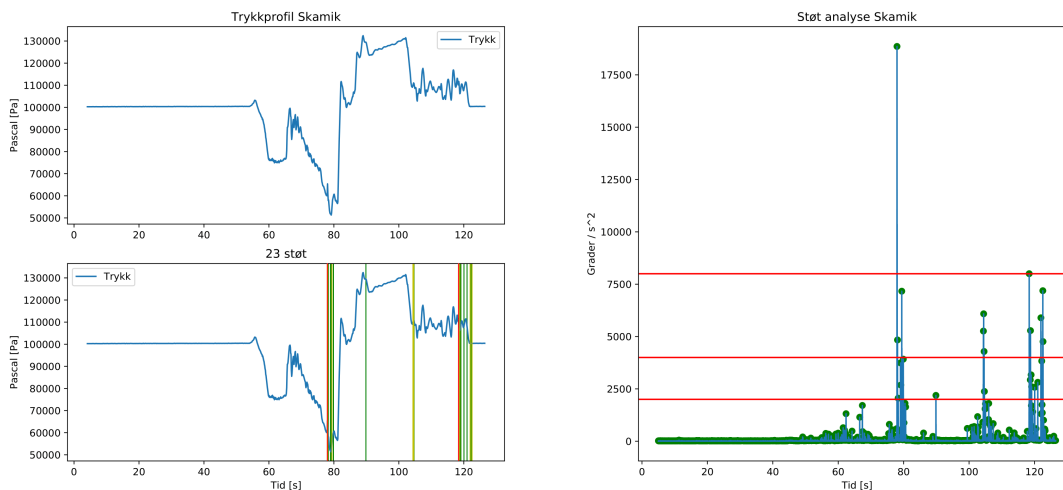
(b) Støt analyse.

Figur 3.13: Resultat av automatisk støtdeteksjon for Thermolicer test 2.

Det oppdages like mange støt for testene gjennomført i Thermolicer (fig. 3.12 og fig. 3.13) til tross for noen ulikheter i trykkprofil, støtprofil og utslaget. Det oppdages et støt før

innløp for både test 1 og for test 2, dette kommer av håndteringen når operatøren fører inn Sensorfisk. De fleste støtene oppstår på samme sted under avlusningen. Den første samlingen støt oppstår ved innsuget til trykk-vakuumpumpen, ved tiden 75s - 100s for test 1 og 80s - 90s for test 2. Den neste samlingen av støt er ved avsileren etter pumpingen, ved tiden 140s for test 1 og 125s for test 2. Den siste samlingen av støt oppstår ved avsileren etter behandlingen i oppvarmet sjøvann, ved tiden 345s for test 1 og 160s for test 2.

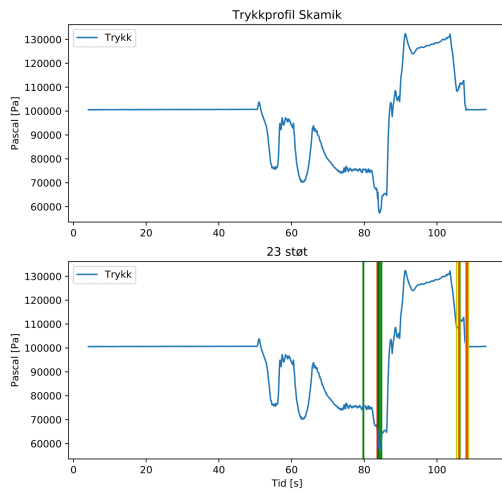
Resultatene fra Skamik (fig. 3.14 og fig. 3.15) omfatter kun pumping og avsiling, før børsting og spyling. Dette kommer av at behandlingsløpet for avluseren ble delt i to og det ble gjort separate tester for inndelingene. Trykkprofilene, støtprofilene og utslaget er ulikt for testene, men det oppdages like mange støt for testene hvor de fleste støtene oppstår på samme punkt. Den første samlingen av støt oppstår når Sensorfisk befinner seg i tanken til trykk-vakuumpumpen. Den andre samlingen oppstår i den siste bøyningen i rørgangen før avsileren.



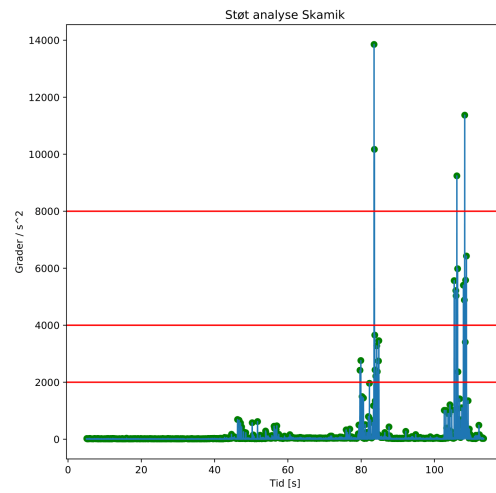
(a) Trykkprofil og støtprofil.

(b) Støt analyse.

Figur 3.14: Resultat av automatisk støtdeteksjon for Skamik test 1.

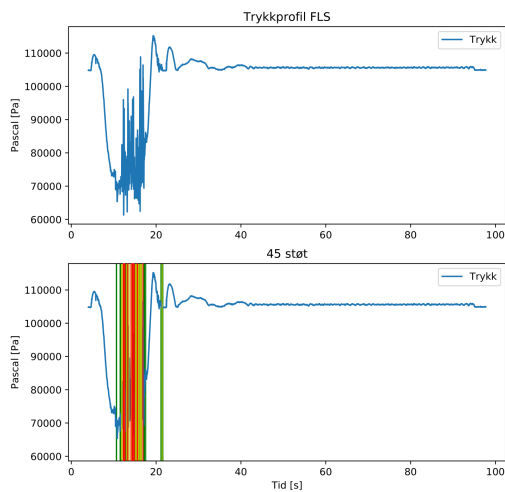


(a) Trykkprofil og støtprofil.

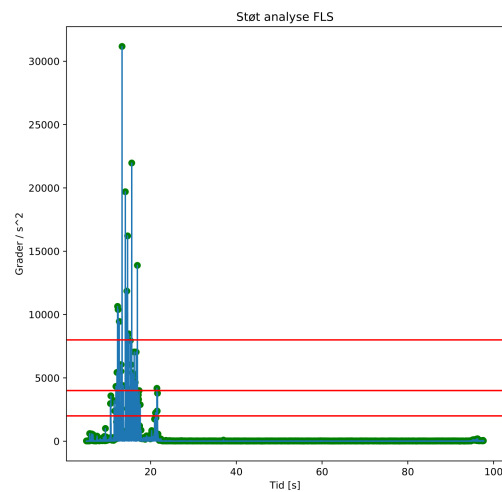


(b) Støt analyse.

Figur 3.15: Resultat av automatisk støtdeteksjon for Skamik test 2.

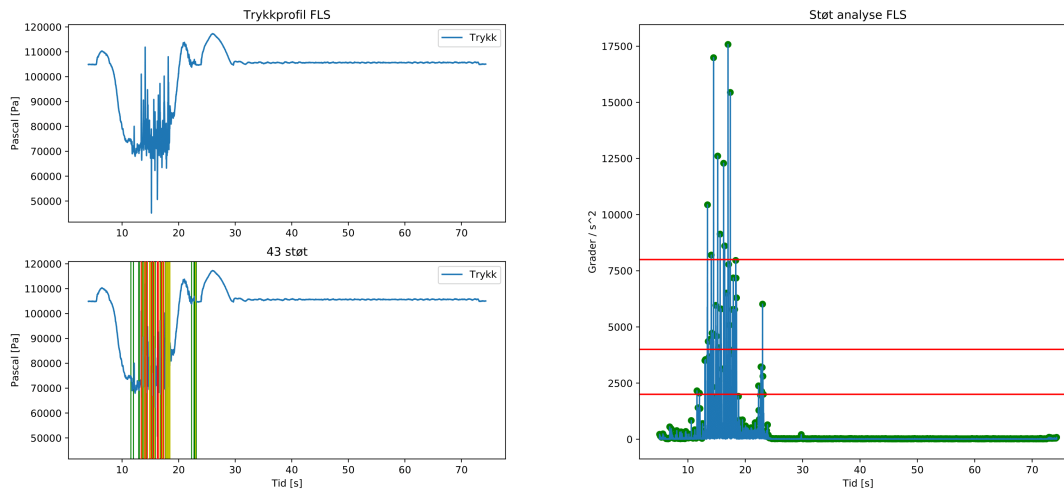


(a) Trykkprofil og støtprofil.



(b) Støt analyse.

Figur 3.16: Resultat av automatisk støtdeteksjon for FLS test 1.



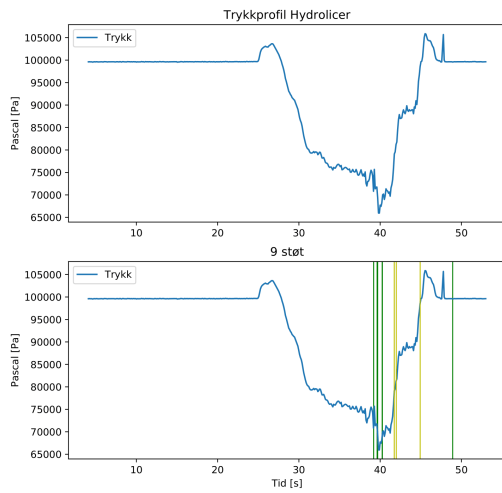
(a) Trykkprofil og støtprofil.

(b) Støt analyse.

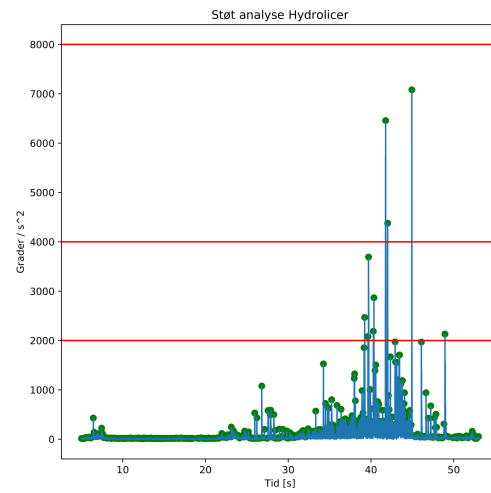
Figur 3.17: Resultat av automatisk støtdeteksjon for FLS test 2.

Støtprofilene for FLS (fig. 3.16 og fig. 3.17) er lik og det oppdages nesten like mange støt for testene, til tross for varierende utslag. Test 1 hadde et maksimalt utslag på ca. $32.000^{\circ}/s^2$ og 45 støt, mens test 2 hadde et maksimalt utslag på ca. $17.500^{\circ}/s^2$ og 43 støt. Fra trykkprofilen kommer det frem at støtene oppstår først etter Sensorfisk kommer i kontakt med lavtrykkavluseren.

Støtprofilene for Hydrolicer 1 (fig. 3.18 og fig. 3.19) er lik og det oppstår nesten like mange støt, hhv 9 og 10 støt for test 1 og test 2. Det største utslaget er likt for begge testene og ligger på rundt $7.000^{\circ}/s^2$, som tilsvarer middels støt. Støtene oppstår i overganger hvor trykket begynner å øke for begge testene. Dette er når Sensorfisk passerer ejektorpumpen og går gjennom det turbulente vannstrålen som skal spyle laskelusen av fisken.

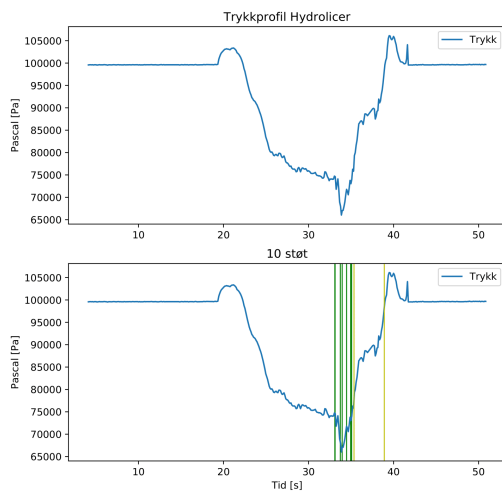


(a) Trykkprofil og støtprofil.

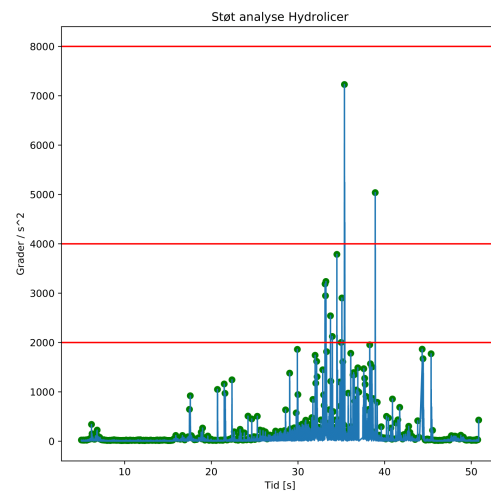


(b) Støt analyse.

Figur 3.18: Resultat av automatisk støtdeteksjon for Hydrolicer 1 test 1.

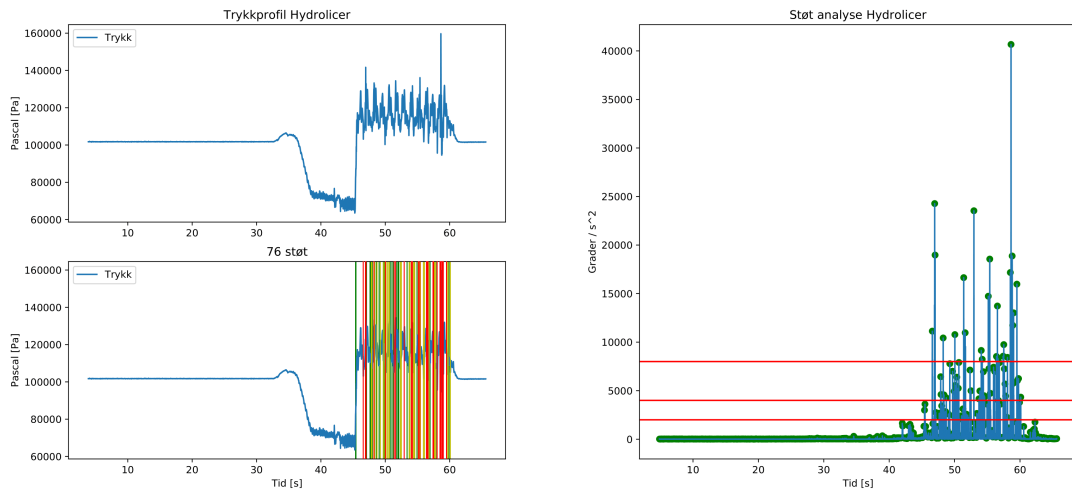


(a) Trykkprofil og støtprofil.



(b) Støt analyse.

Figur 3.19: Resultat av automatisk støtdeteksjon for Hydrolicer 1 test 2.



(a) Trykkprofil og støtprofil.

(b) Støt analyse.

Figur 3.20: Resultat av automatisk støtdeteksjon for Hydrolicer 2 test 1.

Hydrolicer 2 er dimensjonert for mindre fisk og den har en impellerpumpe. Det oppdages flest støt for dette avlusningssystemet (fig. 3.20), nesten dobbelt så mange støt enn det som ble oppdaget i FLS som hadde nest mest. Fra trykkprofilen vises det at alle støtene oppstår når Sensorfisk oppholder seg i impellerpumpen og skal pumpes videre i systemet.

Kapittel 4. Diskusjon og konklusjon

De små avvikene som oppstår under interpoleringen er et resultat av at toppen til utslaget faller mellom to interplasjonspunkt og kuttet vekk. Det er mulig å redusere denne effekten ved å interpolere med et tidsskritt lavere enn 1 ms, eksempelvis 0.5 ms, men dette avviket er så lite at det har meget liten - til ingen innvikring på datasettet. Spesielt for akselerasjonsdata siden det er tregere svingninger som inneholder mest informasjon om bevegelse, og ikke store, korte utslag. I tillegg til at et lavere tidsskritt er mer regnekrevende for datamaskinen. Metoden for å automatisk oppdage innløp og utløp er meget vellykket. Med testrutinene slik dem er nå, hvor Sensorfisk starter loggingen ca. 60 sekunder før innløp, kan det fint brukes en automatisk metode for grensesetting. Hvor grensen settes til 0.2% over medianen, som først ble testet i kapittel 2.2.1. Den manuelle grensesettingen ble brukt siden den automatiske metoden gav varierende resultater på samme datasett, og den manuelle metoden var mer robust. Med starttiden og sluttiden kjent er det mulig å forbedre de andre algoritmene. Det kan brukes til å begrense søk av hendelser til innfor dette intervallet, slik kan det spares prosesseringskraft og det vil redusere kjøretiden. I tillegg vil det føre til at hendelser ikke oppdages utenfor avlusningen, f.eks. ved at det registreres støt når operatør håndterer sensorfisk.

Ingen av metodene for automatisk detektering av passeringen av magnetbånd var vellykket. Verken z-skår algoritmen eller korrelasjon klarte å gjenkjenne passeringene. Dette var ikke overraskende når det ses på rådata fra magnetometeret. Det er passeringen som ikke kommer tydelig frem, noen passeringer opptrer som en dersom de er for nærme hverandre, og dataen var veldig påvirket av støy. Resultatene fra z-skår algoritmen var lovende med data samlet under laboratorieforhold, men der skilte utslagene seg tydelig ut fra resten av datasettet. Dersom passeringene kan komme tydeligere frem i rådata, vil denne algoritmen muligens klare å gjenkjenne passeringene. Det anbefales at det utvikles nye magnetbånd med kraftigere magneter slik at magnetfeltene blir sterkere og resulterer i kraftigere utslag ved passeringer. For så å teste z-skår algoritmen på disse datasettene.

Algoritmen for automatisk støtdeteksjon har utmerket funksjonalitet. Med tillegget av at varigheten kunne bli justert ned til tre tidskonstanter dersom det ikke fantes punkter etter fem tidskonstanter, ble også robustheten utmerket. For alle testede datasett ble både starttiden, sluttiden og varigheten for hvert enkelt støt oppdaget uten noen form for overlapp. Dette betyr at hver enkelt støt oppdages individuelt og uavhengig av hverandre, selv om det kan være mindre enn en tiendel av ett sekund som skiller dem. Med denne kunnskapen tilgjengelig åpner det for videre analysering av datasettene. Det vil blant annet være mulig å se på endringene i lineærakselerasjon under støt og kalkulere rykket (eng. jerk) som oppstår. Ved estimering er det mulig å unngå bruk av data under et støt til estimering, og heller vente til støt er over før man samler datasettet.

Det er problematisk å trekke konklusjoner fra resultatene fra støtdeteksjonen.

Grensesettingen er vilkårlig valgt og det er ikke gjort sammenlikninger mot hvilken effekt avlusningsmetodene har på levende laks. Det er derimot mulig å sammenlikne de enkelte systemene presentert her siden den samme metoden er benyttet for alle datasettene. FLS og Hydrolicer 2 skiller seg ut som avlusningssystemene med flest og kraftigest støt. Den turbulente vannstrømmen til FLS skaper store variasjoner i trykket og utsetter Sensorfisk for mange støt. Impellerpumpen på Hydrolicer 2 er verst av alle pumpesystemene, noe som er forståelig da det er bevegelige deler inne i pumpen som treffer Sensorfisk. Det kommer også frem for alle tre systemene med trykk-vakuumpumper at støtene oppstår like før inngangen til tanken for pumpen. Fra resultatene til Hydrolicer 1 og FLS kommer det frem at ved passeringen av ejetorpumpen oppstår det støt. Felles for alle systemene er at laksen utsettes for undertrykk ved innløp.

Visualiseringsmetodene brukt til presentasjon av resultatene er enkle og informative. De inneholder informasjon fremstilt på et enstydig format. Det vil være mulig å formidle resultatene gjennom vitenskaplige tidsskrifter og populærvitenskaplige artikler i samme format. Profilene er intuitive og de forteller hvor hendelser oppstår. En forbedring av visualiseringen kan være å markere på profilene hvor Sensorfisk passerer punkter i avlusningssystemet. Det kan markeres hvor avsilere, bøyninger i rørganger eller spyling inntreffer. Dette er ikke gjort her av hensyn til anonymisering av avlusningsenhetene og fordi det ikke eksisterer nøyaktige metoder for å determinere slike plasseringer.

Kapittel 5. Videre arbeid

5.1 Passering av magnetbånd

Det må utvikles magnetbånd med sterkere magneter dersom passeringene av magnetbåndene skal fremkomme tydeligere i datasettet. Deretter må det utføres tester med z-skår algoritmen på dette datasettet.

5.2 Støtdeteksjon

Det gjenstår mye jobb med grensesettingen til støtene. Hvor skal grensene settes, hva skal inngå i definisjonene og hvor stort skal toleransen for den laveste grensen være? Dette er spørsmål som må besvares gjennom et samarbeid mellom biologer, veterinærer og eventuelt andre relevante aktører. Det er heller ikke nødvendigvis rotasjonsakselerasjonen som skal benyttes til klassifiseringen av støtene. Men at et annet mål for bevegelsesmengden kan bli brukt til å klassifisere støtene, og da bruke rotasjonsakselerasjon til å kun identifisere støtene.

5.3 Utvikling

Det er mulig å forbedre ytelsen til algoritmene ved å inkorporere resultatene fra andre algoritmer. Ved å ta i bruk de kjente tidene for innløp og utløp er det mulig å forbedre støtdeteksjonen til å kun søke etter støt i dette intervallet, likt for passeringen av magnetbånd. Dette vil øke effektiviteten til algoritmene og forbedre resultatene.

5.4 Estimering

Modellene og estimatorstrukturene presentert i oppgaven legger et godt utgangspunkt for videre arbeid. Det gjenstår å implementere estimatorstrukturen og utføre simuleringer på dem. Dette kan være et utgangspunkt for fremtidige prosjektoppgaver eller masteroppgaver.

Vedlegg

A Oppgavebeskrivelse



NTNU
Norges Teknisk-Naturvitenskapelige
Universitet

Fakultet for informasjonsteknologi og
elektroteknikk
Institutt for teknisk kybernetikk

MASTEROPPGAVE (30 Stp.)

Kandidatens navn: Ole-Kristian Lie
Program: Industriell kybernetikk
Tittel: Prosessering og presentasjon av Sensorfisk-data
Engelsk tittel: Processing and presentation of Sensorfisk data

Oppgavens tekst:

Oppdrettslaks håndteres og transporteres gjennom rør, pumpesystemer og behandlingenheter i ulike sammenhenger som, for eksempel, behandling mot lus. Dette fører til redusert velferd og økt behandlingsdødelighet ved at laksen utsettes for fysiske påkjenninger og mekanisk belastning som kan medføre økt stress og fysiske skader. Det er per i dag lite kunnskap om hvilke deler av maskineriet som påfører fisken størst belastning. SINTEF Ocean har derfor utviklet "Sensorfisk", et konsept for måling av fiskens forhold under rørtransport, håndtering og behandling. Gjennom anvendelse av denne teknologien kan behovet for ny kunnskap om de rådende fysiske forhold under rørtransport, håndtering og behandling adresseres, og fiskevelferden forbedres.

Sensorfiskene samler inn bevegelsesrelaterte data som gjør det mulig å estimere posisjonen til sensorfiskene gjennom rørsystemene i en begrenset tidsperiode. Når posisjonen er kjent, kan data som krefter, hastighet, trykk, temperatur m.m. knyttes til estimert posisjon. For å kommunisere resultatene fra målingene på en god måte for beslutningstakere i havbruksnæringen, er det behov for å prosessere og presentere data med nye, kreative, grafiske fremstillingsformer. Med bakgrunn i dette omfatter oppgaven følgende punkter:

1. Lage en kinematisk modell for sensorfiskens posisjon.
2. Analysere data for å detektere hendelser av interesse som f.eks. kraftige støt.
3. Utvikle og teste forskjellige estimatorstrukturer. Hovedsakelig Kalmanfilter og Utvidet Kalmanfilter, og eventuelt andre løsninger som Fossens ikkelineære observer.
4. Utvikle metoder for å visualisere data fra sensorfisk i 2D og 3D som illustrerer målte verdien på forskjellige posisjoner i avluseren.
5. Utføre tester basert på resultatene i punkt 1) – 4) med reelle data.

Oppgaven gitt: 14. januar 2019
Besvarelsen leveres: 10. juni 2019
Utført ved: Institutt for teknisk kybernetikk
Veileder: Martin Føre, ITK, NTNU
Medveildere: Walter Caharija, SINTEF Ocean

Trondheim, 14. januar 2019
Martin Føre
Veileder

B Teknisk spesifikasjon for Fisk1 (liten)

SPECIFICATIONS	
General	
Weight	3-4 kg
Maximum Depth	150m
Microcontroller	Pikavak 2.1
Battery	Lithium-Ion Battery H2B180 2S1P Sanyo URA18650FK
Battery Voltage	7.4V /17.2Wh/2320mAh
Charge Voltage	8.4V
SD-card	SandDisk Extreme U3 32GB
Power Consumption (FMU+IO) including LED's and Piezo Buzzer	250mA@5V - 1.25W
Power Consumption NEO-MSP	35mA@3V (Maximum 67mA) - 0.2W [Max]
Total Power Consumption - Maximum	1.45W
	Estimated operating time 17.2Wh / 1.45W = 11.86h
	1.45W
Logging Setup	
IMU	1000Hz
Compass	50Hz
Pressure temp	50Hz
Temperature fast-response	20Hz

MOP

IMU	
Sensor	MPU9250
Number Of IMU	3
Full-Scale Range	±2 / ±4 / ±8 / ±16g
Axis orientation	X
Pressure/temperature sensor [BAR]	
Sensor Name	Bar30 High-Resolution 300m Depth Pressure Sensor
Sensor	MMS5837
I2C Address	0x76
Temperature sensor [TSYS]	
Sensor Name	Celsius Fast-Response, ±0.1°C Temperature Sensor (I2C)
Sensor	TSYS01
I2C Address	0x77
Power adapter	
Adapter Name	Mascot 2-Cell Li-ion battery charger
Type	2541
Output Voltage	8VDC
Output Current Max	2.7A
Input Voltage Max/Min	264VAC/90VAC

Bibliografi

- Brakel, J.P. van (2014). *Smoothed z-score algo*. Tilgjengelig: <https://stackoverflow.com/questions/22583391/peak-signal-detection-in-realtime-timeseries-data>, (Avlest: 09.07.2018).
- Carlson, T. J. og J.P. Duncan (2003). *Evolution of the Sensor Fish Device for Measuring Physical Conditions in Severe Hydraulic Environments*. Tilgjengelig: https://www.pnnl.gov/main/publications/external/technical_reports/PNNL-15708.pdf, (Avlest: 12.03.2019).
- Chen, Chi-Tsong (2014). *Linear System Theory and Design*. Oxford University Press, Incorporated. ISBN: 9780199964543.
- Deng, Z. D. mfl. (2014). «Design and implementation of a new autonomous sensor fish to support advanced hydropower development». I: *Review of Scientific Instruments* 85.11, s. 115001. DOI: [10.1063/1.4900543](https://doi.org/10.1063/1.4900543). Tilgjengelig: <https://doi.org/10.1063/1.4900543>.
- Egeland, Olav og Tommy Gravdal (2002). *Modeling and Simulation for Automatic Control*. Marine Cybernetics AS. ISBN: 9788292356012.
- Fossen, Thor I. (2011). *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons, Ltd. ISBN: 9781119994138. DOI: [10.1002/9781119994138](https://doi.org/10.1002/9781119994138).
- Frøslie, Kathrine Frey (2018). «Korrelasjon». I: *Store norske leksikon*. Tilgjengelig: <https://snl.no/korrelasjon>, (Avlest: 27.05.2019).
- Haykin, Simon (1996). *Adaptive Filter Theory (3rd Ed.)* Upper Saddle River, NJ, USA: Prentice-Hall, Inc. ISBN: 0-13-322760-X.
- Holan, Astrid Buran mfl. (2017). «Beste praksis for medikamentfrie metoder for lakseluskontroll (MEDFRI)». I: Faglig sluttrapport. Avdeling: Produksjonsbiologi.Nr 10/2017.
- Jiang, Z., Y. He og J. Han (2015). «Disturbance estimation for RUAV using UKF with acceleration measurement». I: *2015 IEEE International Conference on Mechatronics and Automation (ICMA)*, s. 500–505. DOI: [10.1109/ICMA.2015.7237536](https://doi.org/10.1109/ICMA.2015.7237536).
- Kalman, Rudolph Emil (1960). «A New Approach to Linear Filtering and Prediction Problems». I: *Transactions of the ASME–Journal of Basic Engineering* 82.Series D, s. 35–45.
- Kumar, A. og N. Arya (2017). «A study relation between energy spectral density and probability density function with impulse response first order control system». I: *2017 International conference of Electronics, Communication and Aerospace Technology (ICECA)*. Bd. 2, s. 152–154. DOI: [10.1109/ICECA.2017.8212784](https://doi.org/10.1109/ICECA.2017.8212784).
- Lie, Ole-Kristian (2018). «Prosessering og presentasjon av sensorfisk-data». I: Norges teknisk-naturvitenskapelige universitet, Institutt for teknisk kybernetikk.

- Lovren, N. og J. K. Pieper (1997). «A strapdown inertial navigation system for the flat-Earth model». I: *IEEE Transactions on Aerospace and Electronic Systems* 33.1, s. 214–224. ISSN: 0018-9251. DOI: [10.1109/7.570747](https://doi.org/10.1109/7.570747).
- Mason, M. T. (2001). «Dynamics». I: *Mechanics of Robotic Manipulation*. MITP. ISBN: 9780262256629. Tilgjengelig: <https://ieeexplore.ieee.org/document/6291091>.
- Ormestad, Helmut (2017). «Støt - fysikk». I: *Store norske leksikon*. Tilgjengelig: https://snl.no/st%C3%B8t_-_fysikk, (Avlest: 28.05.2019).
- Perez, T. og T. I. Fossen (2007). «Kinematic Models for Manoeuvring and Seakeeping of Marine Vessels». I: *Modeling, Identification and Control* 28. DOI: [10.4173/mic.2007.1.3](https://doi.org/10.4173/mic.2007.1.3).
- Perkins, P. og S. Heber (2018). «Identification of Ribosome Pause Sites Using a Z-Score Based Peak Detection Algorithm». I: *2018 IEEE 8th International Conference on Computational Advances in Bio and Medical Sciences (ICCABS)*, s. 1–6. DOI: [10.1109/ICCABS.2018.8541902](https://doi.org/10.1109/ICCABS.2018.8541902).
- Python Software Foundation (2018). *About Python*. Tilgjengelig: <https://www.python.org/about/>, (Avlest: 04.12.2018).
- Rahni, A. A. A. og I. Yahya (2007). «Obtaining translation from a 6-DOF MEMS IMU — an overview». I: *2007 Asia-Pacific Conference on Applied Electromagnetics*, s. 1–5. DOI: [10.1109/APACE.2007.4603861](https://doi.org/10.1109/APACE.2007.4603861).
- SciPy (2019). *numpy.correlate*. Tilgjengelig: <https://docs.scipy.org/doc/numpy/reference/generated/numpy.correlate.html>, (Avlest: 06.05.2018).
- SNAME (1950). «Nomenclature for treating the motion of a submerged body through a fluid». I: *Technical Report Bulletin 1-5, Society of Naval Architects and Marine Engineers* 28.
- Svendsen, Eirik (2017). «Standardisert metodikk for kvalifisering av mekaniske avlusingsystemer». I: Prosjektbeskrivelse for FHF-prosjektet: KVALISYS (FHF-901397).
- Tveit, G. M. mfl. (2017). «Utvikling av beste praksis for pumping av pelagisk fisk». I: *Faglig sluttrapport: FHF-901257*. Tilgjengelig: <https://www.fhf.no/prosjekter/prosjektbasen/901257/>.
- Van der Merwe, Rudolph (2004). «Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models». Scholar Archive. 8. Ph.d.-avh. Oregon Health & Science University. DOI: [10.6083/M4Z60KZ5](https://doi.org/10.6083/M4Z60KZ5).

