

Eline Malde

# Modellprediktiv regulering av oksygenkonsentrasjon hos KOLS- pasienter

Masteroppgave i Kybernetikk og robotikk

Veileder: Øyvind Stavdahl, Maria Vatshaug Ottermo

Juni 2019



Eline Malde

# Modellprediktiv regulering av oksygenkonsentrasjon hos KOLS- pasienter

Masteroppgave i Kybernetikk og robotikk  
Veileder: Øyvind Stavadahl, Maria Vatshaug Ottermo  
Juni 2019

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





---

---



## Masteroppgave

Studentens navn: Eline Malde  
Fag: Teknisk kybernetikk  
Tittel (norsk): Modellprediktiv regulering av oksygenkonsentrasjon hos KOLS-pasienter  
Tittel (English): Model predictive control of oxygen concentration for COPD patients  
Beskrivelse:

Menneskekroppens celler forbruker  $O_2$  og danner  $CO_2$ , og er derfor avhengig av stadig å få tilført oksygen ( $O_2$ ) og få transportert bort karbondioksid ( $CO_2$ ) for å kunne opprettholde livet.

Hos pasienter med nedsatt lungefunksjon, f.eks. KOLS-pasienter, er det vanlig å få ekstra oksygenrik pustegass for å avhjelpe respirasjonen. I dag stilles oksygenmengden inn manuelt etter legens vurdering og pasientens opplevde behov. Imidlertid kan for mye oksygen gi pasienten redusert behov for å trekke pusten og dermed opphopning av  $CO_2$ . En ønsker derfor å utvikle et system for automatisk regulering av tilført oksygen som kan øke pasientens livskvalitet og samtidig hjelpe leger til bedre å vurdere pasientens tilstand. Tidligere arbeider har etablert dynamiske modeller, såkalte kompartimentmodeller, med ulike detaljeringsgrader for systemet, og disse vil ligge til grunn for arbeidet i denne oppgaven.

En kan se for seg flere ulike reguleringsstrategier for å opprettholde ønsket nivå av de to gassene i blodet, hvorav PID-regulering ble testet i en tidligere oppgave. Modellprediktiv regulering (MPC) er en annen mulig strategi, og dette skal testes ut i praksis i denne oppgaven:

1. Sett deg inn i teori på MPC og design en reguleringsstruktur for å opprettholde ønsket nivå av de to gassene i blodet. Vurder bruk av lineær versus ulineær MPC for en eller flere av kompartimentmodellene
2. Implementer og evaluer reguleringsstrukturen i valgt programvare.
3. Sammenlign resultatene med erfaringene fra PID-regulering.

Veileder(e): Øyvind Stavdahl, Maria Vatshaug Ottermo

Trondheim, 11.01.2019

Øyvind Stavdahl  
Faglærer

---

---

# Sammendrag

Noen KOLS-pasienter får ekstra oksygenrik pustegass for å avhjelpe respirasjonen, hvor mengden oksygen i dag styres manuelt. For mye oksygen kan gi pasienten et redusert behov for å trekke pusten og dermed føre til opphopning av  $\text{CO}_2$  – en farlig og potensielt dødelig situasjon. Mange velger derfor naturlignok et for lavt nivå av oksygen i pustegassen, som ivaretar sikkerheten men gir dårligere komfort. Ved å regulere oksygenkonsentrasjonen automatisk kan livskvalitet hos pasienten forbedres, ved å bedre ivareta både komfort og sikkerhet.

Arbeidet som presenteres i denne avhandlingen er modellprediktiv regulering av oksygenkonsentrasjonen ved hjelp av programvare fra Cybernetica. En eksisterende kompartimentmodell er tilpasset til målinger fra forsøk på gris, ved offline parameterestimering i programmet ModelFit fra Cybernetica. Modellen ble ytterligere tilpasset i ModelFit ved online parameterestimering i form av et Kalmanfilter.

En reguleringsstruktur er foreslått og implementert i Cybernetica sine programmer for ulineær modellprediktiv regulering, CENIT. Programmet CENIT og RealSim fra Cybernetica ble brukt for å simulere regulering av systemet i lukket sløyfe. Systemet er representert av en prosessmodell med gaussisk varierte parameterverdier i forhold til optimale parametre, som valgt for estimatormodellen i ModelFit. Ingen av modellene inneholder tidsforsinkelser.

Fordi pustefrekvensen i modellen og forsøkene er styrt manuelt, er reguleringsstrukturen testet ved å påtrykke ulike pustefrekvenser og observere oksygenmengden bestemt av regulatoren. Simuleringstesten viser at både komfort og sikkerhet i prosessmodellen er ivaretatt med valgt MPC, for alle påsatte pustefrekvenser.

Ved sammenligning av resultatet med erfaringer fra PID-regulering fra tidligere arbeid, er det klart at MPC er best egnet for regulering av oksygenkonsentrasjon hos KOLS-pasienter. Det er også konkludert med at ulineær MPC er best egnet for regulering av oksygenkonsentrasjon i forhold til lineær MPC. Programmene til Cybernetica er evaluert og vurdert som velegnet for modelltilpasning og regulering av oksygenkonsentrasjon hos både forsøksgris og KOLS-pasienter.

Sikkerhet og komfort i prosessmodellen er ivaretatt med valgt reguleringsstruktur og fungerer tilfredstillende. Det gjenstår å se om tidsforsinkelser, naturlig variasjon av pustefrekvensen og test av reguleringsstrukturen på et virkelig system vil gi tilfredstillende resultater.

Det er foreslått å legge til tidsforsinkelser i prosessmodellen for å bedre simulere regulering av et virkelig system, før reguleringsystemet testes på en ny gris. For fremtidig regulering er det foreslått å gjøre pustefrekvensen i modellen om til en måling eller målbar forstyrrelse ved hjelp av en enkel ligning.

---

# Abstract

Some COPD patients receive extra oxygen-rich breathing gas to help with respiration, where the amount of oxygen today is controlled manually. Too much oxygen can give the patient a reduced need to breathe and thus lead to accumulation of  $\text{CO}_2$  – a dangerous and potentially fatal situation. Many therefore choose a too low level of oxygen in the breathing gas, which guarantees safety but gives poorer comfort. By regulating the oxygen concentration automatically, the quality of life of the patient can be improved by guaranteeing both comfort and safety.

The work presented in this thesis is model-predictive control of oxygen concentration using software from Cybernetica. An existing compartment model is adapted to measurements from a pig experiment, with offline parameter estimation in the program ModelFit from Cybernetica, using SQP. The model was further adapted in ModelFit with online parameter estimation using a Kalman filter.

A regulatory structure has been proposed and implemented in Cybernetica's program for nonlinear model predictive control, CENIT. The program CENIT and RealSim from Cybernetica were used to simulate control of the system in closed loop. The system is represented by a model with gaussian varied parameter values relative to optimal parameters chosen for the estimator model in ModelFit. The models does not contain time delays.

Because the breathing rate in the model and experiments is controlled manually, the control structure is tested by applying different breathing frequencies and observing the amount of oxygen determined by the controller. The simulation test shows that both comfort and safety in the model are taken care of with the chosen MPC, for all applied breathing frequencies.

When comparing the result with experiences from PID control from previous work, it is clear that MPC is best suited for controlling oxygen concentration in COPD patients. It has also been concluded that nonlinear MPC is best suited for control of oxygen concentration over linear MPC. The software from Cybernetica have been evaluated as suitable for model adaptation and control of oxygen concentration in both pigs and COPD patients.

Safety and comfort are met with the chosen regulatory structure and works satisfactorily. It remains to be seen whether time delays, natural variation of the breathing rate and testing of the regulatory structure of a real system will yield good results.

It is suggested to add time delays in the process model to better simulate the regulation of a real system before the control system is tested on a new pig. For future regulation, it is proposed to convert the breathing rate in the model into a measurement or measured disturbance using a simple equation.

---

# Forord

Arbeidet som presenteres i denne avhandlingen er blitt utført ved Institutt for teknisk kybernetikk hos Norges teknisk-naturvitenskapelige universitet (NTNU), under veiledning av professor Øyvind Stavdahl og Maria Vatshaug Ottermo. Oppgaven er skrevet som en 30 studiepoengs, avsluttende oppgave på sivilingeniørstudiet Kybernetikk og robotikk.

Jeg har alltid interessert meg både for biologi og teknologi, så det å få skrive en oppgave i skjæringspunktet mellom medisin og kybernetikk har vært midt i blinken for meg. Arbeidet med oppgaven har vært en lærerik erfaring, både i forhold til oppfriskning av gammel kunnskap og påfyll av ny.

Modellen som er brukt i oppgaven er utviklet av Maria V. Ottermo. Kasper Linnestad fra Cybernetica holdt kurs for oss i februar om aktuelle programmer fra Cybernetica og har vært til stor hjelp ved å svare på spørsmål om programmene som brukes i oppgaven. Veiledere, Linnestad og førsteamanuensis Leira har svart på spørsmål og kommet med innspill og veiledning underveis. Deres bidrag er kreditert i teksten.

Jeg har selv implementert modellen i programmene til Cybernetica, tilpasset modellen til forsøk på gris, foreslått reguleringsstruktur og implementert og tunet MPCen i programvare fra Cybernetica. Ved implementasjonsproblemer var Ottermo og Linnestad til stor hjelp. Bidragene fra arbeidet i avhandlingen er listet i 1.4.

## Takk

Først og fremst ønsker jeg å takke min biveileder, Maria Vatshaug Ottermo, som har beina på jorda, alltid har tid, er motiverende og for hennes hyggelige og behagelige vesen. Etter veiledning med Ottermo er jeg alltid litt klokere og mer motivert. Takk for gode diskusjoner, for at du har hørt på når jeg har trengt å si ting høyt, for at du har brukt tid på å sammenligne kode og for raske svar på mail. Ikke minst, takk for modell, for tips underveis og gjennomlesning og kommentarer på deler av oppgaven.

Tusen takk til Kasper Linnestad hos Cybernetica for god hjelp med de ulike programmene til Cybernetica som er brukt i oppgaven. Det har vært til stor hjelp og vært utslagsgivende for å komme i mål, da særlig siden det ikke finnes informasjon om bruk av programmene på nettet.

Jeg ønsker også å takke min inspirerende, engasjerte og smarte veileder, Øyvind Stavdahl, for hans innspill, forslag og for svar på spørsmål jeg ikke visste jeg lurte på. Selv med sykemelding nesten hele semesteret har veiledningstimene med deg vært til god hjelp.

Takk til Maria og Miriam, jentene jeg bor med, for et avslappende og godt hjem å komme hjem til. Takk for hyggelige frokoster, kjekke kvelder og helger, for gode samtaler og for

---

tøys og tull. Trivsel gjør at en presterer bedre.

Min gode, omsorgsfulle, morsomme familie, som alltid stiller opp og hjelper til, som er mine støttespillere og heiagjeng. Takk til far som elsker realfag, for ditt engasjement på mine vegne og gode råd underveis i studiet. Takk til mor, som støtter og heier på meg og som oppmuntrer og gleder. En ekstra takk til mor for korrekturlesning av bakgrunnsdelen.

Studietiden hadde ikke vært den samme uten deg, Elise Søliland. Takk for hyggelige lun-sjer, ditt gode humør, avslappende forhold til lekser og skole og for din tålmodighet i gruppearbeid med meg. Du er uvurderlig!

Eline Malde

*Trondheim, 3.juni 2019*

---

# Innhold

<b>Tabeller</b>	<b>ix</b>
<b>Figurer</b>	<b>xii</b>
<b>Nomenklatur</b>	<b>xiii</b>
<b>I Bakgrunn</b>	<b>1</b>
<b>1 Innledning</b>	<b>3</b>
1.1 Motivasjon . . . . .	3
1.2 O <sub>2</sub> og CO <sub>2</sub> . . . . .	4
1.2.1 Transport i lungene . . . . .	5
1.2.2 Transport i blodet . . . . .	5
1.3 Mål med oppgaven . . . . .	6
1.4 Bidrag . . . . .	7
1.5 Disposisjon . . . . .	7
<b>2 Kompartimentmodeller</b>	<b>9</b>
2.1 Eldhuset sin dynamiske modell av blodgassvariasjoner . . . . .	9
2.2 Ottermo sin forenklede, dynamiske modell . . . . .	12
2.2.1 Forenklinger . . . . .	13
2.2.2 Tilstandsrommodell . . . . .	15
<b>3 Programmer fra Cybernetica</b>	<b>17</b>
3.1 Programmer . . . . .	17
3.1.1 ModelFit . . . . .	17
3.1.2 RealSim . . . . .	18
3.1.3 CENIT . . . . .	18
3.2 Arbeidsflyt og implementering . . . . .	19



---

<b>II</b>	<b>Parameterestimering</b>	<b>21</b>
<b>4</b>	<b>Offline parameterestimering</b>	<b>23</b>
4.1	Teori . . . . .	23
4.1.1	SQP . . . . .	24
4.2	Metode . . . . .	24
4.2.1	Implementering og initielle verdier . . . . .	24
4.2.2	Parameterestimering . . . . .	25
4.2.3	Estimering av $MBCO_2$ , $MBO_2$ og $D$ . . . . .	27
4.2.4	Estimering av $MBCO_2$ og $MBO_2$ . . . . .	28
4.2.5	Estimering av $MBCO_2$ og $D$ . . . . .	29
4.2.6	Estimering av $MBCO_2$ , $D$ og $V_B$ . . . . .	29
4.3	Resultat . . . . .	31
4.4	Diskusjon . . . . .	31
4.4.1	Estimering av to parametere . . . . .	31
4.4.2	Estimering av flere parametere . . . . .	32
4.4.3	Ikke realistiske parameterverdier . . . . .	32
4.4.4	Resultatet . . . . .	33
<b>5</b>	<b>Online parameterestimering</b>	<b>35</b>
5.1	Teori . . . . .	35
5.1.1	Kalmanfilter . . . . .	36
5.1.2	Ulineære Kalmanfilter . . . . .	37
5.1.3	Tuning . . . . .	39
5.2	Metode . . . . .	39
5.2.1	Endring av std.avvik for parameter $MBCO_2$ . . . . .	40
5.2.2	Endring av std.avvik for parameter $D$ . . . . .	40
5.2.3	Online parameterestimering med initiell modell . . . . .	40
5.2.4	Online parameterestimering for griseforsøk B . . . . .	40
5.3	Resultat . . . . .	45
5.4	Diskusjon . . . . .	46
5.4.1	Standardavvik . . . . .	46
5.4.2	Initiell modell . . . . .	47
5.4.3	Griseforsøk B . . . . .	48
<b>III</b>	<b>MPC</b>	<b>51</b>
<b>6</b>	<b>Modellprediktiv regulering</b>	<b>53</b>
6.1	Teori . . . . .	53
6.1.1	Optimering av dynamiske system . . . . .	53
6.1.2	Modellprediktiv regulering (MPC) . . . . .	56
6.1.3	Lineær MPC . . . . .	59
6.1.4	Ulineær MPC . . . . .	59
6.1.5	MPC i CENIT og RealSim . . . . .	60
6.2	Design av reguleringsstruktur . . . . .	61

---

6.3	Modellprediktiv regulering i CENIT . . . . .	63
6.3.1	Sikkerhet . . . . .	63
6.3.2	Restriksjoner . . . . .	64
6.3.3	Tuning av regulatoren . . . . .	65
6.3.4	Prediksjonshorisont . . . . .	67
6.3.5	Kontrollhorisont og parametrisering av MV . . . . .	67
6.3.6	Funksjonstest . . . . .	67
6.4	Resultat . . . . .	70
6.5	Diskusjon . . . . .	70
6.5.1	Reguleringsstruktur . . . . .	70
6.5.2	Lineær vs. ulineær MPC . . . . .	71
6.5.3	Implementering og tuning . . . . .	72
6.5.4	Komfort og sikkerhet . . . . .	73
6.5.5	Kostfunksjon . . . . .	74
<b>7</b>	<b>Sammenligning av PID og MPC</b>	<b>75</b>
7.1	Resultat fra PI-regulering . . . . .	75
7.2	Metode . . . . .	75
7.3	Resultat . . . . .	77
7.4	Diskusjon . . . . .	77
<b>IV</b>	<b>Oppsummering og videre arbeid</b>	<b>79</b>
<b>8</b>	<b>Diskusjon</b>	<b>81</b>
8.1	Modelltilpasning i ModelFit . . . . .	81
8.2	Regulering i CENIT og RealSim . . . . .	83
8.3	Fremtidige forsøk på gris . . . . .	84
<b>9</b>	<b>Konklusjon</b>	<b>87</b>
9.1	Parameterestimering . . . . .	87
9.2	MPC . . . . .	88
9.3	Programvare fra Cybernetica . . . . .	88
<b>10</b>	<b>Videre arbeid</b>	<b>89</b>
10.1	Før nytt griseforsøk . . . . .	89
10.2	Fremtidige griseforsøk . . . . .	90
10.3	Regulering av $[O_2]$ hos mennesker og automatisk variasjon av $F_T$ . . . . .	91
	<b>Bibliografi</b>	<b>95</b>
	<b>Appendiks</b>	<b>99</b>
A	MATLAB-kode . . . . .	99
A.1	Generering av nye parameterverdier . . . . .	99
B	Implementering av Ottermo sin modell . . . . .	100
C	Data fra griseforsøk . . . . .	101
C.1	Data fra griseforsøk A . . . . .	101

---

---

C.2	Data fra griseforsøk B . . . . .	106
D	Tabell med aktuelle verdier fra [1] . . . . .	109
E	Vektmatrisene <b>Q</b> og <b>S</b> i CENIT . . . . .	109
F	Målte og estimerte utgangsverdier . . . . .	109

# Tabeller

4.1	Initielle parameterverdier . . . . .	25
4.2	Verdier fra offline parameterestimering . . . . .	28
5.1	Valgt standardavvik for prosesstøy . . . . .	46
5.2	Valgt standardavvik for målestøy . . . . .	46
6.1	Nye parameterverdier generert fra gaussisk fordeling . . . . .	63
6.2	Begrensninger på CV . . . . .	66
6.3	Begrensninger på MV/DV . . . . .	66
D.1	Utvalgte verdier fra artikkel om gris . . . . .	109

---

# Figurer

1.1	Eksempel på avvikende scenarier som krever ulike tiltak for reguleringen av oksygenkonsentrasjon . . . . .	6
2.1	Eldhuset sin dynamiske modell . . . . .	10
2.2	Ottermo sin dynamiske modell . . . . .	12
3.1	Oversikt over programmer fra Cybernetica og hvordan de kommuniserer .	18
3.2	Oversikt over arbeidsflyten ved implementering av modell i programmene til Cybernetica . . . . .	19
4.1	Inngangsverdier og utgangsverdier i ModelFit, uten modelltilpasning eller estimering . . . . .	26
4.2	Utgangsverdier fra modell og prosess der parameterne $MBCO_2$ , $MBO_2$ og $D$ er offline parameterestimert . . . . .	27
4.3	Graf som viser avhengighet mellom parameterene $MBO_2$ og $D$ . . . . .	28
4.4	Utgangsverdier fra modell og prosess der parameterne $MBCO_2$ og $MBO_2$ er offline parameterestimert . . . . .	29
4.5	Utgangsverdier fra modell og prosess der parameterne $MBCO_2$ og $D$ er offline parameterestimert . . . . .	30
4.6	Utgangsverdier fra modell og prosess der parameterne $MBCO_2$ , $D$ og $V_B$ er offline parameterestimert . . . . .	30
5.2	Utgangs- og parameterverdier før tuning av Kalmanfilter . . . . .	41
5.3	Utgangs- og parameterverdier ved endring av $std\_v(MBCO_2)$ . . . . .	42
5.4	Utgangs- og parameterverdier ved endring av $std\_v(D)$ og $std\_w(O_2)$ . . .	42
5.5	Utgangs- og parameterverdier for initiell modell med tunet Kalmanfilter for offline parameterestimert modell . . . . .	43
5.7	Tilpasning til griseforsøk B med offline parameterestimert modell tilpasset forsøk B og online parameterestimert modell fra tidligere . . . . .	45

---

6.1	Illustrasjon av prinsipp for MPC . . . . .	57
6.2	Blokkdiagram av MPC med følging av referanse . . . . .	58
6.3	Designet reguleringsstruktur . . . . .	62
6.4	Test av reguleringsystem med $F_T = 21$ . . . . .	68
6.5	Test av reguleringsystem med $F_T = 15$ . . . . .	68
6.6	Test av reguleringsystem med $F_T = 14$ . . . . .	69
6.7	Test av reguleringsystem med $F_T = 10$ . . . . .	69
7.1	Resultat fra PI-regulering . . . . .	76
F.1	Estimerte og målte utgangsverdier ved $F_T = 21$ . . . . .	110
F.2	Estimerte og målte utgangsverdier ved $F_T = 15$ . . . . .	111
F.3	Estimerte og målte utgangsverdier ved $F_T = 14$ . . . . .	112
F.4	Estimerte og målte utgangsverdier ved $F_T = 10$ . . . . .	113

---

# Nomenklatur

## Forkortelser:

CV	Controlled Variables	// z-ene i modellen
DD1	first-order divided difference	
DD2	second-order divided difference	
DV	Disturbance Variables	// forstyrrelsene i modellen
EKF	Extended Kalman Filter	
KF	Kalman Filter	
MPC	Model Predictive Controller	
MV	Manipulated Variables	// inngangsverdi som kan endres av regulator
OPC	Open Platform Communications	
SQP	Sequential Quadratic Programming	
UKF	Unscented Kalman Filter	

## Greske symboler:

$\alpha_1$	en konstant
$\alpha_2$	en konstant
$\alpha_3$	en konstant
$\alpha_4$	en konstant
$\delta$	binær variabel
$\Delta$	endring
$\varepsilon$	slakkvariabel
$\theta$	parametere
$\mu$	forventningsverdi
$\sigma$	varians
$\tau$	tidsforsinkelse
$\omega_1$	prosesstøy
$\omega_2$	prosesstøy
$\omega_3$	prosesstøy
$\omega_4$	prosesstøy
$\Omega$	løsningsområde

## Medisinske forkortelser:

KOLS	Kronisk obstruktiv lungesykdom
LTOT	Langtids oksygenterapi
SV	Slagvolum



---

**Medisinske og kjemiske symbol:**

$\text{CO}_2$	karbondioksid
$\text{O}_2$	oksygen
$[\text{CO}_2]$	konsentrasjon av karbondioksid
$[\text{O}_2]$	konsentrasjon av oksygen

**Symboler:**

<b>A</b>	kontrollmatrise
<b>B</b>	inngangsmatrise
<b>C</b>	utgangsmatrise
<b>c</b>	restriksjoner
<b>D</b>	diffusjonskoeffisient
<b>e</b>	feil, differanse mellom estimert og faktisk tilstand
$\mathcal{E}$	likehetsrestriksjoner
$\mathcal{I}$	ulikhetsrestriksjoner
<b>J</b>	kostfunksjon
$J(\theta)$	kostfunksjon
<b>K</b>	kalman-forsterkning
<b>k</b>	samplingsnummer
<b>M</b>	metabolsk produksjon eller forbruk av hhv. $\text{CO}_2$ og $\text{O}_2$
<b>M</b>	kontrollhorisont
<b>N</b>	prediksjonshorisont
<b>n</b>	totalt antall samples
$\mathcal{N}$	normalfordeling
<b>P</b>	kovariansmatrise til feil i Kalmanfilter-algoritmen
<b>P</b>	prediksjonshorisont
$\dot{Q}$	blodstrøm
<b>Q</b>	vektmatrise i kostfunksjon
<b>Q</b>	varians til prosessstøy
$\mathbb{R}$	reelle tall
<b>R</b>	vektmatrise i kostfunksjon
<b>R</b>	varians til målestøy
<b>r</b>	vekt-vektor i kostfunksjon
<b>S</b>	vektmatrise i kostfunksjon
<b>U</b>	matrise med inngangsverdier
<b>u</b>	inngangsverdi
<b>V</b>	volum
<b>v</b>	prosessstøy
$\dot{V}_A$	alveolært minuttvolum
<b>w</b>	målestøy
<b>x</b>	tilstandsvariabler
$\mathbf{x}_0$	initialtilstander
<b>y</b>	målte verdier
<b>Z</b>	matrise med utgangsverdier som tilbakekobles
<b>z</b>	beslutningsvariabler
<b>z</b>	målte verdier som tilbakekobles

---

$\Delta U$	matrise med endring av nåværende inngangsverdi i forhold til forrige
$\square$	konsentrasjon
$\hat{y}$	estimerte verdier, gjelder også andre bokstaver enn y
$\bar{y}$	estimerte verdier, gjelder også andre bokstaver enn y

**Senket og hevet skrift:**

<i>A</i>	alveoler
<i>a</i>	alveoler
<i>Aa</i>	Alveole-arterie
<i>Aa<sub>O<sub>2</sub></sub></i>	O <sub>2</sub> i alveole-arterie kompartment
<i>ACO<sub>2</sub></i>	CO <sub>2</sub> i alveoler
<i>AO<sub>2</sub></i>	O <sub>2</sub> i alveoler
<i>art</i>	arterier
<i>B</i>	kropp
<i>B,CO<sub>2</sub></i>	CO <sub>2</sub> i kropp
<i>B,O<sub>2</sub></i>	O <sub>2</sub> i kropp
<i>Bv</i>	kropp-ener
<i>D</i>	død
<i>eff</i>	effektiv
<i>H</i>	hjerne
<i>HCO<sub>2</sub></i>	CO <sub>2</sub> i hjerne
<i>HO<sub>2</sub></i>	O <sub>2</sub> i hjerne
<i>H-ven</i>	fra hjerne til vener
<i>in</i>	inn
<i>K</i>	kropp
<i>KCO<sub>2</sub></i>	CO <sub>2</sub> i kropp
<i>KO<sub>2</sub></i>	O <sub>2</sub> i kropp
<i>K-ven</i>	fra kropp til vener
<i>maks</i>	maksimum
<i>min</i>	minimum
<i>puls</i>	puls
<i>T</i>	tidal
<i>T</i>	pust ( $F_T$ er pustefrekvens)
<i>T</i>	transponert
<i>v</i>	vener
<i>v</i>	prosessstøy
<i>v</i>	parameterstøy
<i>ven</i>	vener
$\tau_{AB}$	tidsforsinkelse fra alveoler til kropp
$\tau_{AH}$	tidsforsinkelse fra alveoler til hjerne
$\tau_{AK}$	tidsforsinkelse fra alveoler til kropp

---

**Del I**

# **Bakgrunn**



# Kapittel 1

## Innledning

I dette kapittelet presenteres bakgrunn og motivasjon for den aktuelle problemstillingen med en oppsummerende beskrivelse av samlebegrepet KOLS, samt ulemper ved dagens manuelle regulering av oksygen. Det gis en liten innføring i transporten av  $O_2$  og  $CO_2$  fra luften som pustes inn til cellene i kroppen, før innledningen avsluttes med en beskrives av målet med oppgaven, bidrag og disposisjon.

Motivasjonen og bakgrunnen er generelt ganske lik som for [2], og derfor er mye av det som er skrevet i disse innledende kapitlene inspirert av tidligere arbeid [2]. Generelt vil det være slik at de avsnittene der «[kilde]» står etter siste punktum i avsnittet, gjelder kilden for hele avsnittet.

### 1.1 Motivasjon

Ta et dypt åndedrag, slipp så luften sakte ut gjennom nesen. Det føles godt og beroligende, ikke sant? Et voksent menneske i hvile puster omtrent 20 000 ganger hver dag [3], og bortsett fra ved fysiske anstrengelser ofres det sjeldent en tanke. Mellomgulvet trekkes sammen og skaper et vakuum i lungene, som gjør at oksygenrik og livsnødvendig luft strømmer med inspirasjonsluften til lungealveolene [4]. I alveolene går oksygenet over i blodet og blir transportert ut til alle cellene i kroppen, der det brukes til dannelsen av energi til kroppen. Den automatiske respirasjonsreguleringen til kroppen tilpasser oksygenbehovet svært nøyaktig [5]. Respirasjonsfrekvensen vil for eksempel øke ved fysiske anstrengelser, som følge av at musklene jobber og dermed trenger mer oksygen.

For noen er pusting en kamp, noe de tenker på hele tiden. I 2004 var, i følge WHO [6], hele 64 millioner mennesker globalt rammet av kronisk obstruktiv lungesykdom – bedre kjent som KOLS. WHO har predikert at KOLS vil være den femte hyppigste dødsårsaken

i 2020. De neste ti årene er det antatt en økning på 30%, som gjør at det i 2030 forventes at KOLS vil være den tredje hyppigste dødsårsaken. Med andre ord – et økende problem.

For de mest alvorlige KOLS-pasientene vil de ikke få nok oksygen, selv om de får legemidler som utvider bronkiene<sup>1</sup> og bremser betennelse. Disse pasientene lider av hypoksi, en kronisk tilstand med for lav tilførsel av oksygen, noe som er svært ubehagelig for pasienten da man føler åndenød og kvelningsfølelse. Behandlingen for disse KOLS-pasientene er langtids oksygenterapi (LTOT), som vil si at de får ekstra oksygenrik pustegass for å avhjelpe respirasjonen. Oksygenmengden reguleres i dag manuelt etter legens vurdering og pasientens opplevde behov, noe som ikke er optimalt verken med tanke på komfort eller sikkerhet. [2, 8]

LTOT behandlingens fryktede bivirkning er opphopning av karbondioksid i blodet, kalt hyperkapni. Dette skyldes en for høy dose oksygen, slik at pasientens pustefrekvens blir lavere, grunnet et oppfattet redusert respirasjonsbehov. Kroppen, som vanligvis kvitter seg med CO<sub>2</sub> gjennom ekspirasjonsluften, får dermed dårligere utlufting av CO<sub>2</sub>, som igjen fører til en forhøyet mengde karbondioksid i blodet. Dette er en alvorlig og potensielt dødelig tilstand. Denne fryktede bivirkningen er grunnen til at oksygentilførselen nesten alltid doseres for lavt i forhold til pasientens behov. For pasienten vil dette innebære et stort ubehag og betydelig redusert livskvalitet. [2, 8]

Dagens LTOT behandling byr på ytterligere utfordringer. Om en pasient får tilstrekkelig med oksygen i hvile, vil pasienten kunne oppleve åndenød ved økt aktivitet, om ikke doseringen av oksygen endres. Ved følelse av åndenød må pasienten selv oppjustere dosen, men dette velger mange å ikke gjøre da de er utrygge på om de overdoserer oksygen. Dette gjør at pasienten ofte har problemer med å utføre daglige gjøremål. For å øke pasientens livskvalitet er derfor behovet stort for en løsning som regulerer oksygentilførselen på en god og sikker måte. [2, 8]

Som nevnt i et tidligere avsnitt er KOLS et økende problem, men allikevel øker ikke behandlingsformer i takt med den negative utviklingen av sykdommen. Et verdifullt steg i utbedringsprosessen av behandlingsoalternativene for LTOT-pasienter kan være å utvikle et godt fungerende reguleringsssystem, i form av modellprediktiv regulering, slik at pasienten kan slippe å justere oksygentilførselen selv. Dette vil forbedre både komfort og sikkerhet, og øke pasientens livskvalitet.

OxyAid er et prosjekt ved NTNU som har til hensikt å lage en «sensor med tilhørende reguleringsystem som måler blodets gassinhold gjennom huden og benytter disse signalene til å automatisk dosere oksygen basert på behov og aktivitetsnivå hos KOLS-pasienter med LTOT behandling» [8]. Denne oppgaven er skrevet som et bidrag til prosjektet.

## 1.2 O<sub>2</sub> og CO<sub>2</sub>

O<sub>2</sub> spiller en essensiell rolle i kroppens rammeverk og generell funksjonalitet. O<sub>2</sub> er livsnødvendig for at cellene i kroppen skal kunne gjennomføre aerob forbrenning av druesuk-

---

<sup>1</sup>Luftveiene fra steder der luftrøret deler seg, og til de minste endeforgreiningene i lungene [7]

ker til ATP, som gir kroppen energi. Avfallsstoffene i denne forbrenningen er CO<sub>2</sub> og H<sub>2</sub>O. Kroppen må dermed transportere O<sub>2</sub> til cellene og avfallsstoff bort fra kroppen. [2, 9]

Gjennom nerveimpulser mottar respirasjonssentrene i hjernen informasjon om behovet for endret åndedrett fra arbeidende muskler. Lungene og kjemoreseptorer sender beskjedene om behovet for endret åndedrett. Waaler og Hauge [10] beskriver at «Kjemoreseptorer er spesialiserte strukturer som reagerer på endringer i partialtrykkene av oksygen og karbondioksid i arterieblod og på endringer i surhetsgraden (pH) i mellomcellevæsken i hjernen. Sentrene stimuleres (og åndedrettet øker) når for eksempel surhetsgraden og karbondioksidinnholdet øker i det omgivende hjernemiljøet eller i arterieblodet.» [10]

Ettersom pustefunksjonen automatisk reguleres av hvor mye karbondioksid som finnes i blodet til en hver tid, er også karbondioksid viktig for kroppen. Fordi CO<sub>2</sub> har høyere affinitet enn O<sub>2</sub>, vil gassen i midlertid ha en indirekte giftig virkning dersom det inntas for mye av den, eller dersom ikke kroppen kvitter seg med nok gjennom ekspirasjonen. [2, 9]

### 1.2.1 Transport i lungene

Luften vi puster inn består av nesten 21% oksygen og bare 0,04% CO<sub>2</sub>. Ved innånding trekkes mellomgulvet sammen slik at det oppstår et undertrykk i lungene, og oksygenrik luft fra omgivelsene strømmer inn [4]. Luften går fra øvre luftveier<sup>2</sup> til nedre luftveier, der luftrøret deler seg i høyre og venstre hovedbronkie som igjen leder til hver sin lungehalvdel. I hver lungehalvdel forgreines bronkiene, som blir tynnere og tynnere for hver forgreining. Forgreiningene ender i små lungeblærer som kalles alveoler. Rundt alveolene ligger kapillærnett, og alveolene er ordnet slik at overflaten for gassutveksling blir størst mulig. [2, 11]

Den høye konsentrasjonen av O<sub>2</sub> i alveolene, sammenlignet med konsentrasjonen i det oksygenfattige blodet i kapillærene, gjør at O<sub>2</sub> diffunderer inn i kapillærene. For CO<sub>2</sub> er det motsatt: den lave konsentrasjonen av CO<sub>2</sub> i alveolene, sammenlignet med den høye konsentrasjonen i kapillærene, gjør at CO<sub>2</sub> diffunderer fra kapillærene til alveolene. Kroppen kvitter seg med CO<sub>2</sub> via ekspirasjonsluften. [5]

### 1.2.2 Transport i blodet

Når O<sub>2</sub> er kommet over fra alveolene til kapillærene, vil hemoglobin feste seg til oksygen ved hjelp av jernioner før det blir med blodstrømmen videre. Bindeevnen til hemoglobin varierer med omgivelsene – der det er mye oksygen er bindeevnen sterk, og i de mer oksygenfattige områdene er bindeevnen svakere, slik at oksygen frigis. Slik transporterer hemoglobinet med seg store mengder oksygen fra blodårene nær lungene til cellene. Fra cellene blir CO<sub>2</sub> fraktet med hemoglobin tilbake til lungene. [2, 9]

Oksygeninnholdet i blodet endrer seg i stor grad proporsjonalt med mengden hemoglobin som finnes i blodet og dets oksygenmetning. Oksygenmetningen i blodet måles i %, og

---

<sup>2</sup>består av nese med neseganger, munnhulen, svelget og strupehodet



angir hvor stor andel av kapasiteten de røde blodlegemene har til å binde seg til oksygen, som til en hver tid utnyttes. [2, 12]

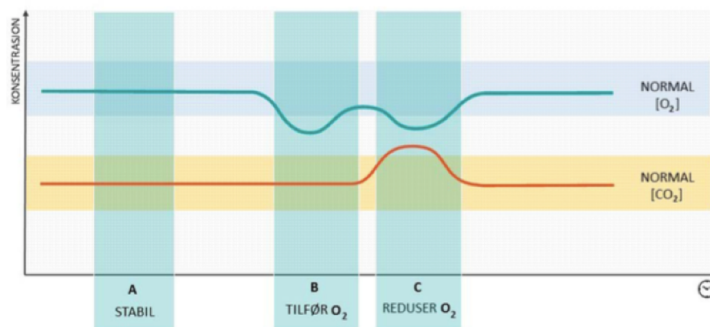
Til tross for hemoglobinetts gode og raske evne til å binde oksygen, er det visse andre gasser som har enda sterkere affinitet, f.eks.  $\text{CO}_2$ . Denne egenskapen kan i uheldige tilfeller føre til gassforgiftning, fordi hemoglobinet ikke lenger har kapasitet til å frakte nødvendig mengde oksygen til cellene i kroppen. Dermed vil organene og cellene i kroppen få et potensielt skadelig underskudd av oksygen<sup>3</sup>. [2, 13]

### 1.3 Mål med oppgaven

Det overordnede målet med denne masteroppgaven er å designe og implementere en modellprediktiv regulator for å opprettholde ønsket nivå av  $\text{O}_2$  og  $\text{CO}_2$  i kroppen, som vil gi bedre komfort og sikkerhet til KOLS-pasienter på LTOT-behandling. Figur 1.1 viser eksempler på avvikende scenarier det er nødvendig at regulatoren håndterer på en god måte.

Nødvendige delmål vil være å sette seg godt inn i teori for MPC, og kortfattet formidle teorien på en strukturert og forståelig måte. Et annet delmål vil være å vurdere bruk av ulineær og lineær MPC for en eller flere av modellene som er gitt. I tillegg må valgt programvare fra Cybernetica evalueres.

Tidligere arbeid [2] og erfaringer gjort ved implementering av PID-regulator kan brukes for å vurdere valgt MPC i forhold til PID. Videre må også figurer og grafer legges grunnlag for vurdering og evaluering av den implementerte reguleringsstrukturen. Det må vurderes om reguleringsstrukturen tilfredsstillende opprettholdelsen av ønskede nivå av  $\text{O}_2$  og  $\text{CO}_2$  i kroppen, om reguleringsstrukturen er god nok til å testes i et nytt forsøk på gris og hva som eventuelt gjenstår å gjøre.



**Figur 1.1:** Eksempel på avvikende scenarier som krever ulik regulering ved regulering av oksygenkonsentrasjon hos KOLS-pasienter. Figuren er hentet fra [8]

<sup>3</sup>Det er dette som kalles hyperkapni, som også er beskrevet under avsnitt 1.1

## 1.4 Bidrag

Oppgaven bidrar mot målet om å kunne regulere oksygenkonsentrasjon hos KOLS-pasienter på en komfortabel og sikker måte. Nyttige arbeider og resultater fra oppgaven, som bidrar til prosjektet, er gjengitt i teksten under. Filer og det som er gjort i programmene til Cybernetica og implementering av modellen i Cyberneticas template i Visual Studios, er tilgjengelige for OxyAid for videre bruk.

- Tilpasning av parametere i modell, ved hjelp av Cybernetica ModelFit, slik at utgangsverdier til modellen ligner mest mulig på målinger fra griseforsøk. Tuning av Kalmanfilter ved online parameterestimering i ModelFit. Kalmanfilteret estimerer målinger, uten å replikere støy, på en god måte. Utgangsverdiene estimeres tilfredstillende, selv for en prosessmodell med gaussisk varierte parameterverdier som avviker fra optimale parameterverdier brukt i estimeringsmodellen.
- Forslag til en reguleringsstruktur som ivaretar både komfort og sikkerhet er foreslått. Det er også foreslått verdier for tuning av regulatoren, som gjør at regulatoren tilfredstiller kravene beskrevet i mål med oppgaven i 1.3.
- Vurdering av MPC i forhold til PID-regulering gjort i tidligere arbeid, hvor MPC er vurdert som mest egnet ved regulering av oksygenkonsentrasjon hos KOLS-pasienter. Det er også vurdert valg av lineær MPC vs. ulineær MPC ut i fra teori, hvor ulineær MPC har vist seg som best egnet for problemet i denne oppgaven. Også programvaren fra Cybernetica, som er benyttet i oppgaven, er vurdert i forhold til arbeidet gjort i oppgaven og videre bruk.
- Oppgaven bidrar også med tanker og ideer for videre arbeid, som kan brukes for å ytterligere evaluere reguleringsstrukturen for videre bruk, forbedre eksisterende løsning og utvidelser for fremtidig regulering. Det er foreslått hva som bør gjøres før et nytt griseforsøk, ved et fremtidig griseforsøk, samt utvidelser av modellen ved regulering av oksygenkonsentrasjon hos mennesker.

## 1.5 Disposisjon

Oppgaven er delt inn i fire overordnede deler, I Bakgrunn, II Parameterestimering, III MPC og IV Oppsummering og videre arbeid, hvor hver av disse inneholder kapitler. Kapitlene er organisert slik at de har en teoridel, etterfulgt av metode, resultat og diskusjon. Unntaket er del IV, som inneholder de tre kapitlene Diskusjon, Konklusjon og Videre arbeid.

På neste side følger en oversikt over hvordan oppgaven er strukturert og innholdet hvert enkelt kapittel presenterer.

## I Bakgrunn

**Kapittel 1** gir en introduksjon til bakgrunn og motivasjon for oppgaven. Deretter følger en kortfattet introduksjon av viktigheten og transport av  $O_2$  og  $CO_2$  i kroppen. Kapitlet avsluttes med en beskrivelse av målene med oppgaven, oppgavens bidrag og til slutt disposisjon.

**Kapittel 2** består av en beskrivelse av kompartmentmodellene som er gitt.

**Kapittel 3** gir en oversikt og beskrivelse av programvare som er brukt.

## II Parameterestimering

**Kapittel 4** presenterer utvalgt teori, metode og resultater fra offline parameterestimering i programmet ModelFit, der resultatene diskuteres i slutten av kapitlet.

**Kapittel 5** presenterer relevant teori, metode og resultater fra online parameterestimering i programmet ModelFit. En diskusjon av resultatene er gitt i slutten av kapitlet.

## III MPC

**Kapittel 6** omhandler modellprediktiv regulering. Kapitlet omfavner en god del teori, både om generelle optimeringsproblemer, MPC, lineær og ulineær MPC, samt modellprediktiv regulering i CENIT. Et forslag til design av MPC regulator for prosessen er også gitt, samt en vurdering av om lineær eller ulineær MPC er mest passende for regulering av modellene som er gitt. Kapitlet avsluttes med implementasjon av regulatoren i CENIT, der fremgangsmåte og resultater blir gitt. Til slutt diskuteres resultatene.

**Kapittel 7** gir en sammenligning av PI-regulering av prosessen, gjort i et tidligere arbeid, og MPC-reguleringen som ble gjort i kapittel 6.

## IV Oppsummering og videre arbeid

**Kapittel 8** innholdet i foregående kapitler oppsummeres med en overordnet diskusjon, i tillegg diskuteres fremtidige forsøk på gris.

**Kapittel 9** presenterer en konklusjon av arbeidet som er gjort og diskusjoner i foregående kapitler.

**Kapittel 10** gir forslag til videre arbeid.

# Kapittel 2

## Kompartimentmodeller

Kroppen er kompleks, og forenklinger må gjøres for å kunne beskrive den matematisk. I en kompartimentmodell deles modellen inn i ulike deler, kalt kompartments, der det antas at konsentrasjoner eller tilstander er like innenfor disse delene. Det gjøres altså en antagelse om at konsentrasjoner innenfor en kompartiment, ikke varierer som funksjon av rom, bare av tid. På denne måten går en fra et sett med differensialligninger, som varierer både med tid og rom, til ligninger som bare varierer med tid – altså et betydelig mindre komplekst system. [2]

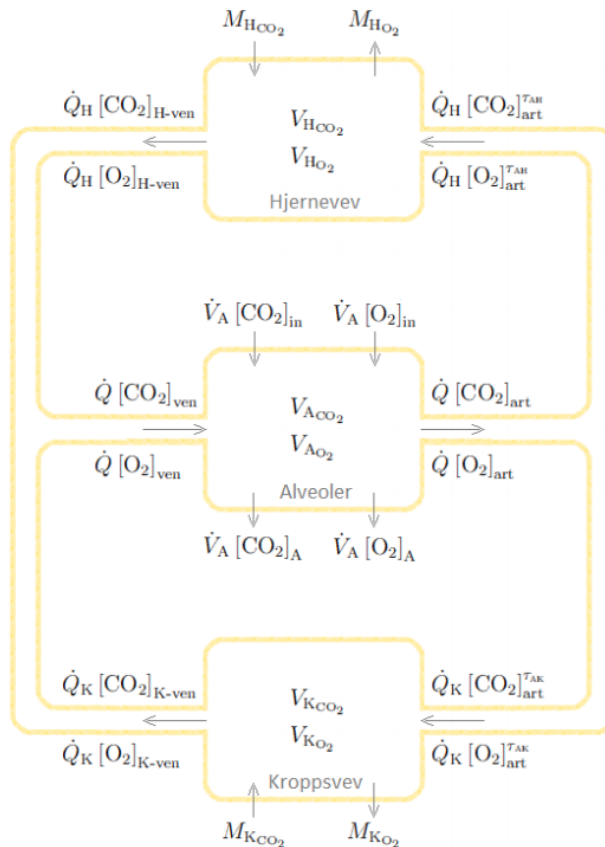
Den dynamiske modellen, som brukes videre i denne oppgaven, er en slik kompartimentmodell. Modellen er en forenkling av modellen til Eldhuset [14], som er inspirert av arbeidet til H. T. Milhorn et al. [15]. I arbeidet til H. T. Milhorn et al. [15] foreslås det en dynamisk modell der kroppen deles inn i tre overordnede kompartiment, som antas å ha sin spesifikke reguleringsmekanisme. På denne måten blir ulike deler av kroppens blod-sirkulasjon ulike kompartments i en kompartimentmodell. Blod strømmer fritt i kroppen, og dermed vil en endring i tilstand i en del av kroppen, påvirke en annen del etter en gitt tidsforsinkelse. [2]

### 2.1 Eldhuset sin dynamiske modell av blodgassvariasjoner

Det dynamiske ligningssettet til Eldhuset er et resultat av modellering av endring i oksygen- og karbondioksidkonsentrasjon i henholdsvis alveolene, hjernevevet og kroppsvevet. Eldhuset [14] beskriver systemet slik: «Basisligningene som beskriver systemet modellerer endringen i gassvolum av karbondioksid og oksygen i de ulike kompartimentene basert på den unike reguleringsmekanismen i hver avdeling. Disse ligningene defineres basert på utgangspunktet for modellen. Et utvalg blodgasser er antatt å bevege seg rundt i systemet

med uendret verdi. Dette medfører at flere variabler i modellen er angitt med en tidsforsinkelse. Tidsforsinkelsene angir endringen i tid fra et blodvolum har beveget seg fra ett kompartiment til et annet. De spesifikke tidsverdiene og utgangspunktet for basisligningene er funnet i litteraturen.» [2]

Ligning (2.1) - (2.6) beskriver autoreguleringen av systemet i figur 2.1, og utgjør basisligningene for den dynamiske modellen. Det henvises til Eldhuset [14] s.52 for antagelsene og kriterier Eldhuset har tatt hensyn til, samt utledning av den dynamiske modellen. [2]



**Figur 2.1:** Illustrasjon av respirasjonssystemet. Hver blokk representerer et kompartiment, og forbindelsene representerer blodløpet mellom dem. Det kontinuerlig varierende volumet ( $V$ ) av karbondioksid og oksygen er angitt for hvert kompartiment. Forbruk og produksjon i metabolismen ( $M$ ), blodflyt ( $\dot{Q}$ ) og alveolær minuttventilasjon ( $\dot{V}_A$ ) sørger for regulering og transport av oksygen og karbondioksid gjennom hele systemet. Figur og figurtekst hentet fra [14].

Ligninger for endring av karbondioksidvolum i henholdsvis hjerne, kropp og lunger er beskrevet i ligning (2.1), (2.2) og (2.3),

$$\frac{d}{dt}V_{HCO_2} = [M_{HCO_2} + \dot{Q}_H([CO_2]_{art}^{\tau AH} - [CO_2]_{H-ven})] \quad (2.1)$$

$$\frac{d}{dt}V_{KCO_2} = [M_{KCO_2} + \dot{Q}_K([CO_2]_{art}^{\tau AK} - [CO_2]_{K-ven})] \quad (2.2)$$

$$\frac{d}{dt}V_{ACO_2} = [\dot{Q}([CO_2]_{ven} - [CO_2]_{art}) + \dot{V}_A([CO_2]_{in} - [CO_2]_A)] \quad (2.3)$$

der  $V$  beskriver volum av karbondioksid i henholdsvis hjerne, kropp og alveoler.  $M$  er produksjon av karbondioksid fra metabolismen i respektive kompartment, mens  $\dot{Q}$  er blodflyt i hjerne og kropp.  $\dot{V}_A$  er alveolær minuttventilasjon,  $art$  er arterier og  $ven$  er vener.  $\tau$  beskriver en tidsforsinkelelse mellom ulike kompartment, mens  $in$  er konsentrasjon i innpustet luft.

Ligning (2.4), (2.5) og (2.6) beskriver endring av oksygenvolum i henholdsvis hjerne, kropp og alveoler.

$$\frac{d}{dt}V_{HO_2} = [-M_{HO_2} + \dot{Q}_H([O_2]_{art}^{\tau AH} - [O_2]_{H-ven})] \quad (2.4)$$

$$\frac{d}{dt}V_{KO_2} = [-M_{KO_2} + \dot{Q}_K([O_2]_{art}^{\tau AK} - [O_2]_{K-ven})] \quad (2.5)$$

$$\frac{d}{dt}V_{AO_2} = [\dot{Q}([O_2]_{ven} - [O_2]_{art}) + \dot{V}_A([O_2]_{in} - [O_2]_A)] \quad (2.6)$$

Variablene er de samme som i (2.1)-(2.3), bare at de her beskriver endring av  $O_2$ . I tillegg kommer ligningen for alveolær minuttventilasjon,  $\dot{V}_A$ .

$$\dot{V}_A = (P_{ACO_2} - \alpha_1) \cdot \left( \alpha_2 \cdot 10^{\frac{\alpha_3}{P_{AO_2} - \alpha_4}} \right) \quad (2.7)$$

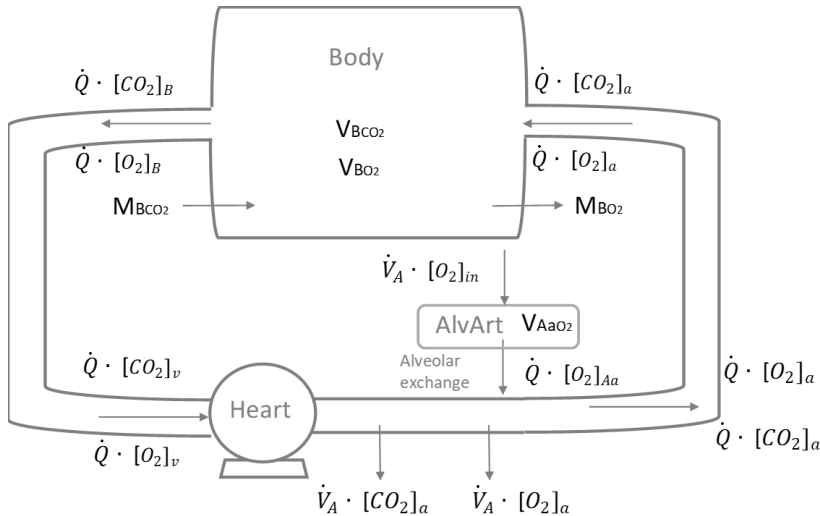
der  $P_{ACO_2}$  og  $P_{AO_2}$  er partialtrykk av henholdsvis  $CO_2$  og  $O_2$  i alveolene og  $\alpha_1, \alpha_2, \alpha_3$  og  $\alpha_4$  er konstanter<sup>1</sup>.

Ved regulering av oksygenkonsentrasjon og måling av  $[O_2]$  i kroppen, kan modellen til Eldhuset med fordel forenkles en del. Kroppens naturlige regulering av oksygentilførsel til hjernen er veldig god, derfor vil en reduksjon av oksygennivået påvirke andre deler av kroppen lenge før hjernen rammes. En god regulering av oksygennivået i kroppen vil derfor også medføre at hjernen har nok oksygen.

<sup>1</sup>I originalteksten [14] er  $\alpha_i$  skrevet som  $\omega_i$

## 2.2 Ottermo sin forenklete, dynamiske modell

Ottermo sin dynamiske modell er en forenkling av modellen til Eldhuset, der blodomløpet bare deles i et kompartiment, kroppen, i tillegg til at det er lagt til dynamikk mellom lunger og blod for å forhindre direkte feedthrough av inngangsverdi til utgangsverdi. Dette er modellen som brukes videre i oppgaven, og er illustrert i figur 2.2. Modellen er først beskrevet av ligning (2.9) - (2.11), før det gjøres en del antagelser som forenkler modellen til (2.22) - (2.24).



**Figur 2.2:** Illustrasjon av en forenkling av respirasjonssystemet som en kompartimentmodell. De to blokkene representerer hvert sitt kompartiment, og forbindelsene representerer blodomløpet mellom dem. Det kontinuerlig varierende volumet ( $V$ ) av karbondioksid og oksygen er angitt for hvert kompartiment. Forbruk og produksjon i metabolismen ( $M$ ), blodflyt ( $\dot{Q}$ ) og alveolær minuttventilasjon ( $\dot{V}_A$ ) sørger for regulering og transport av oksygen og karbondioksid gjennom hele systemet. For å få inn en blodstrøm  $\dot{Q}$ , er også hjertet med i figuren. Figuren er laget av Maria Vatshaug Ottermo og figurtekst er inspirert av Eldhuset [14] s.63.

Den alveolære minuttventilasjonen er beskrevet av:

$$\begin{aligned}\dot{V}_A &= F_T \cdot (V_T - V_D) \\ &= F_T \cdot V_{eff}\end{aligned}\tag{2.8}$$

hvor  $F_T$  er respirasjonsfrekvensen,  $V_T$  er tidalvolumet og  $V_D$  er dødvolumet.  $\dot{V}_A$  her er en annen enn den som er definert i ligning (2.7). Hos pasienter som får ekstra oksygenrik pustegass, vil også oksygenpartialtrykk ha noe å si for den alveolære minuttventilasjonen, som tatt hensyn til i (2.7), men det er ikke tatt hensyn til i denne ligningen.

Endring av karbondioksid- og oksygenvolum i kroppen er beskrevet av ligning (2.9) og (2.10), og endring av oksygenvolum i alveole-arterie-kompartimentet er beskrevet av ligning (2.11).

$$\frac{d}{dt}V_{B,CO_2} = M_{B,CO_2} + \dot{Q} \cdot ([CO_2]_a^{\tau AB} - [CO_2]_{Bv}) \quad (2.9)$$

$$\frac{d}{dt}V_{B,O_2} = -M_{B,O_2} + \dot{Q} \cdot ([O_2]_a^{\tau AB} - [O_2]_{Bv}) \quad (2.10)$$

$$\frac{d}{dt}V_{AaO_2} = -D \cdot \dot{Q} \cdot [O_2]_{Aa} + \dot{V}_A \cdot [O_2]_{in}^{\tau in Aa} \quad (2.11)$$

Variablene i ligningene (2.9)-(2.11), med unntak av subscriptene B for *Body*, Aa for *Alveolearterie*, a for arterie og v for vene, er som i Eldhuset sine ligninger, der V er det kontinuerlig varierende volumet av karbondioksid og oksygen, M beskriver forbruk og produksjon av henholdsvis O<sub>2</sub> og CO<sub>2</sub> i metabolismen, F<sub>T</sub> respirasjonsfrekvens, V<sub>eff</sub> differansen mellom tidalvolum<sup>2</sup> og dødvolum<sup>3</sup> og  $\dot{Q}$  er blodstrømmen, gitt av puls·SV. I tillegg beskriver D diffusjonskoeffisienten mellom alveolene og arteriene.

## 2.2.1 Forenklinger

Ottermo gjør en rekke forenklinger [16], som er gjengitt i listen under:

1. Alveolær CO<sub>2</sub> = arteriell CO<sub>2</sub>:

$$[CO_2]_A = [CO_2]_a \quad (2.12)$$

Fordi CO<sub>2</sub> diffunderer så raskt

2. Alveolær O<sub>2</sub> = arteriell O<sub>2</sub>:

$$[O_2]_A = [O_2]_a \quad (2.13)$$

3. Innåndet CO<sub>2</sub> lik 0

$$[CO_2]_{in} = 0 \quad (2.14)$$

Siden [CO<sub>2</sub>] i innpust er så liten antas den lik 0

4. Tidsforsinkelsen fra luft til kompartimentet mellom lunge og blod, Aa, er neglisjerbar

$$[O_2]_{in}^{\tau in Aa} = [O_2]_{in} \quad (2.15)$$

5. Tidsforsinkelsene fra alveoler til blod er neglisjerbar.

$$[CO_2]_a^{\tau AB} = [CO_2]_a \quad (2.16)$$

$$[O_2]_a^{\tau AB} = [O_2]_a \quad (2.17)$$

<sup>2</sup>Luftmengden som pustes inn eller ut i en normal respirasjonssyklus, normalt 0,5 L

<sup>3</sup>Innåndet luft som ikke er en del av gassutvekslingen i lungene



6. Kropp og venøse blodgasser er i ekvilibrum

$$[CO_2]_{Bv} = [CO_2]_B \quad (2.18)$$

$$[O_2]_{Bv} = [O_2]_B \quad (2.19)$$

som gjør at

$$[CO_2]_v = [CO_2]_B \quad (2.20)$$

$$[O_2]_v = [O_2]_B \quad (2.21)$$

2., 4. og 5. er relativt store forenklinger, som gjøres for å forenkle modellen, mens forenklingene i 1. og 3. er ganske greie antagelser som i praksis ikke gir store endringer. Forenklingene som er gjort gjelder for resten av oppgaven.

Forenklingene gjør at ligning (2.9)-(2.11) forenkles til:

$$\frac{d}{dt}V_{BCO_2} = M_{B,CO_2} + \dot{Q} \cdot ([CO_2]_a - [CO_2]_B) \quad (2.22)$$

$$\frac{d}{dt}V_{BO_2} = -M_{B,O_2} + \dot{Q} \cdot ([O_2]_a - [O_2]_B) \quad (2.23)$$

$$\frac{d}{dt}V_{AaO_2} = -D \cdot \dot{Q} \cdot [O_2]_{Aa} + \dot{V}_A \cdot [O_2]_{in} \quad (2.24)$$

Konsentrasjon av  $CO_2$  og  $O_2$  i arteriene er gitt av:

$$[CO_2]_a = \frac{\dot{Q} \cdot [CO_2]_B}{\dot{Q} + \dot{V}_A} \quad (2.25)$$

$$[O_2]_a = \frac{\dot{Q} \cdot ([O_2]_B + [O_2]_{Aa})}{\dot{Q} + \dot{V}_A} \quad (2.26)$$

Ligning (2.25) og (2.26) kan settes inn i basisligningene (2.22) og (2.23). Fordi volumet i kompartimentene er konstant, kan generelt  $\frac{d}{dt}V_{B,x}$  skrives som  $V_B \cdot \frac{d}{dt}[x]_a$ . Det blir skrevet  $F_T \cdot V_{eff}$  i stede for  $\dot{V}_A$  for at neste steg skal bli mer intuitivt.

$$\frac{d}{dt}[CO_2]_B = \frac{1}{V_B} \cdot \left( M_{B,CO_2} + \dot{Q} \cdot \left( [CO_2]_B - \frac{\dot{Q} \cdot [CO_2]_B}{\dot{Q} + F_T \cdot V_{eff}} \right) \right) \quad (2.27)$$

$$\frac{d}{dt}[O_2]_B = -\frac{1}{V_B} \cdot \left( M_{B,O_2} + \dot{Q} \cdot \left( [O_2]_B - \frac{\dot{Q} \cdot ([O_2]_B + [O_2]_{Aa})}{\dot{Q} + F_T \cdot V_{eff}} \right) \right) \quad (2.28)$$

$$\frac{d}{dt}[O_2]_{Aa} = -\frac{1}{V_{Aa}} \cdot \left( D \cdot \dot{Q} \cdot [O_2]_{Aa} - [O_2]_{in} \cdot F_T \cdot V_{eff} \right) \quad (2.29)$$

## 2.2.2 Tilstandsrommodell

Definerer:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} [CO_2]_B \\ [O_2]_B \\ V_{eff} \\ [O_2]_{Aa} \end{bmatrix}$$

$$\mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} [O_2]_{in} \\ F_T \\ F_{puls} \end{bmatrix}$$

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} [CO_2]_a \\ [O_2]_a \end{bmatrix}$$

som gjør at (2.27)-(2.29) kan omformuleres til tilstandsrommodellen:

$$\dot{x}_1 = \frac{1}{V_B} \cdot \left( M_{B,CO_2} + \dot{Q} \cdot \left( x_1 - \frac{\dot{Q} \cdot x_1}{\dot{Q} + u_2 \cdot x_3} \right) \right) + \omega_1 \quad (2.30)$$

$$\dot{x}_2 = -\frac{1}{V_B} \cdot \left( M_{B,O_2} + \dot{Q} \cdot \left( x_2 - \frac{\dot{Q} \cdot (x_2 + x_4)}{\dot{Q} + u_2 \cdot x_3} \right) \right) + \omega_2 \quad (2.31)$$

$$\dot{x}_3 = \omega_3 \quad (2.32)$$

$$\dot{x}_4 = -\frac{1}{V_{Aa}} \left( D \cdot \dot{Q} \cdot x_4 - u_1 \cdot u_2 \cdot x_3 \right) + \omega_4 \quad (2.33)$$

der  $\omega_1, \omega_2, \omega_3$  og  $\omega_4$  beskriver prosessstøy.



# Kapittel 3

## Programmer fra Cybernetica

På Cybernetica sine hjemmesider [17] står det at Cybernetica utvikler, implementerer og vedlikeholder løsninger for modellbasert regulering, tilsyn og optimering av industrielle prosesser. OxyAid har kjøpt programmer fra Cybernetica for å utvikle modellprediktiv regulering av oksygenkonsentrasjon hos KOLS-pasienter.

I dette kapitlet gis det bakgrunnsinformasjon om de ulike programmene fra Cybernetica som brukes i oppgaven, for å gi leseren oversikt og en enkel innføring i de ulike programmene. Dette vil gjøre det lettere å forstå det som blir gjort i kommende kapittel.

### 3.1 Programmer

Programvare som er kjøpt av Cybernetica er programmene ModelFit, RealSim og CENIT, som hver utfører ulike oppgaver. Ved hjelp av disse programmene tilpasses modellen til målingene, reguleres og simuleres.

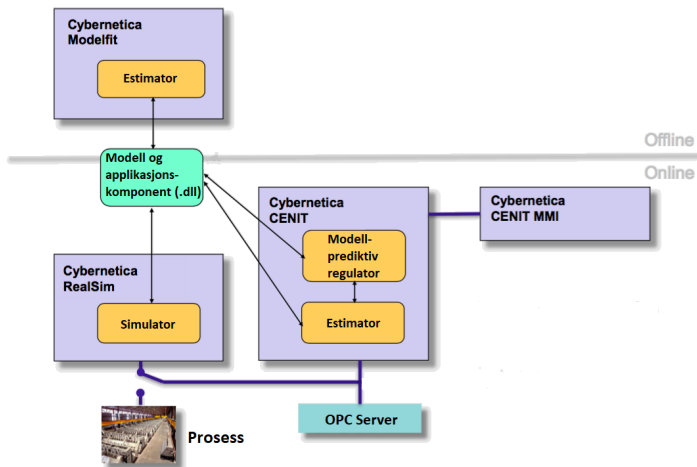
Figur 3.1 gir en oversikt over programmene som brukes og kommunikasjonen dem imellom. RealSim og CENIT kommuniserer gjennom en åpen plattform kommunikasjon (OPC) server og er online, mens ModelFit er offline.

#### 3.1.1 ModelFit

ModelFit er et program som brukes for å tilpasse parameterne i modellen til målinger fra prosessen. Parameterne tilpasses i to steg, først modelltilpasning (offline parameterestimering) og deretter estimering (online parameterestimering). Online parameterestimering er estimering av parametere med Kalmanfilter eller MHE<sup>1</sup>, mens offline parameterestimering

---

<sup>1</sup>Moving Horizon Estimation



**Figur 3.1:** Figuren viser en oversikt over programmene til Cybernetica og kommunikasjonen mellom dem. CENIT MMI er brukergrensesnittet til CENIT. Figuren er hentet fra dokument gitt under kurs hos Cybernetica [18].

er tilpasning av parametere basert på målinger som er gjort av prosessen og tilpasses ved hjelp av SQP. De samme inngangsverdiene påtrykkes modell og prosess for å kunne sammenligne utgangsverdier. Parameterne endres slik at utgangsverdier fra modellen tilpasses utgangsverdiene fra prosessen.

Det er mulig å velge hvilke parametere som skal optimeres, og det er også mulig å velge hvilke målinger som skal være gjeldende for tilpasningen. En modell som er bedre tilpasset prosessen gir bedre prediksjoner og bedre regulering.

### 3.1.2 RealSim

RealSim er et simuleringsverktøy og brukes for test og tuning av regulatoren før regulatoren implementeres på den virkelige prosessen [17]. Logget data fra prosessen kan brukes for å kjøre simuleringene og for debugging og tuning av regulatoren.

### 3.1.3 CENIT

CENIT består av CENIT Kernel, hvor utregninger og kommunikasjon foregår, og CENIT MMI, som er brukergrensesnittet. CENIT regulerer en prosess med ulineær modellprediktiv regulering og har integrert estimator, både Kalmanfilter og MHE kan velges. Verktøyet kommuniserer med prosessen eller RealSim via OPC, som illustrert i figur 3.1.

I CENIT MMI tunes MPCen ved simulering i RealSim, ved å endre på tuningsparameterne, som forklares nærmere i teoridelen i kapittel 6.

## 3.2 Arbeidsflyt og implementering

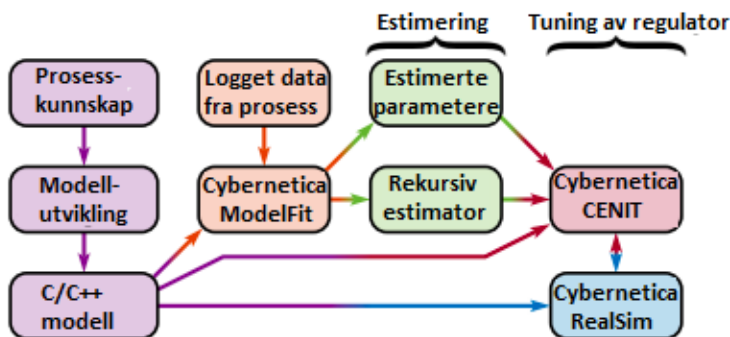
Ved implementering av modellen i Cybernetica sine programmer vil arbeidsflyten være som vist i figur 3.2, hvor modellen først legges inn i Dymola, Visual Studios eller andre programmer. Det er mulig å bruke en ferdig template fra Cybernetica for implementering av modellen i Visual Studio som kompileres til en .dll-fil. Modellen legges inn i programmene fra Cybernetica via .dll-filen. En optimert modell med estimator fra ModelFit brukes i RealSim. Ved simulering i RealSim og CENIT tunes regulatoren som deretter kan brukes i CENIT for online regulering av prosessen.

Ved implementering av modellen i Visual Studios, vil modellen være definert av ulike variabler og parametere:

x	–	dynamiske tilstander
y	–	målinger
z	–	kontrollvariabler (CV)
u	–	inngangsverdier (MV/DV)
par	–	parametere
con	–	konstanter

der

CV	–	kontrollvariabler
MV	–	manipulerbare variabler
DV	–	forstyrrelser



**Figur 3.2:** Figuren viser en typisk arbeidsflyt ved utvikling av en MPC. Modellen tilpasses til data fra prosessen ved å estimere parametere, så tunes en rekursiv estimator – ofte et Kalmanfilter. Regulatoren tunes før den implementeres. ModelFit, CENIT og RealSim er Cybernetica sine verktøy for henholdsvis parameterestimering, MPC og simulering. Figuren er inspirert av [19].



**Del II**

# **Parameterestimering**





# Kapittel 4

## Offline parameterestimering

For mest mulig likhet mellom prosess og modell, ble et utvalg parametere i prosessmodellen tilpasset til data fra prosessen. Tilpasningen av parametere mellom modell og prosess ble gjort med Cybernetica sitt verktøy for offline parameterestimering, ModelFit, hvor parameterverdiene optimeres til de verdiene som passer best for dataene fra prosessen.

I førstkommende delkapittelet gis det en kort innføring i teori som ligger til grunn for optimering av parameterverdier, også kalt offline parameterestimering, i ModelFit. Videre i kapittelet fremlegges fremgangsmåte og resultater for parameterestimeringen, som til slutt diskuteres i et eget delkapittel i slutten av kapittelet.

### 4.1 Teori

I MPC er behovet stort for å kunne gjøre gode prediksjoner, derfor er modellkvaliteten det mest kritiske i MPC. Formålet med modellverifisering er å verifisere modellkvaliteten. Verifiseringen gjøres ved å sammenligne historiske målinger fra modellen med målinger fra prosessen. Dersom disse er så like som mulig, er modellen en veldig god representasjon av virkeligheten og vil gi gode prediksjoner. Gode prediksjoner gjør det mulig å styre prosessen nøyaktig. Dersom det er store avvik mellom modell og prosess, kan parameterne i modellen tilpasses, ved hjelp av offline parameterestimering, slik at modellen passer bedre til prosessen. [18]

I Cybernetica ModelFit blir både modell og målinger fra prosessen lagt inn, som gjør det mulig å sammenligne modell og prosess med samme påsatte inngangsverdier. Med nok variasjon i datasettet, vil det være mulig å optimere flere parametre enn det er målinger. Ved å velge parametere som ModelFit skal optimere, sammenligner programmet utgangsverdier fra modell og prosess og foreslår en endring i parametere som tilpasser modellen bedre til prosessen. [18]

### 4.1.1 SQP

Parameterne som blir utvalgt for estimering i ModelFit, blir utregnet ved å løse følgende optimeringsproblem [18]:

$$\min_{\theta} \quad J(\theta) = \frac{1}{2}(\mathbf{y} - \mathbf{y}_m)^T \mathbf{Q}(\mathbf{y} - \mathbf{y}_m) \quad (4.1a)$$

$$\text{slik at} \quad \mathbf{x}_k = \mathbf{x}_{k+1} + \int_{t_{k-1}}^{t_k} \mathbf{f}(\mathbf{x}(\tau), \mathbf{u}(\tau); \theta) d\tau, \quad \forall k \in \{1, \dots, n\} \quad (4.1b)$$

$$\mathbf{y}_k = \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k; \theta) \quad (4.1c)$$

$$\theta_{min} \leq \theta \leq \theta_{max} \quad (4.1d)$$

der  $\theta$  er parametere,  $\mathbf{f}()$  er prosessmodellen og  $\mathbf{g}()$  gir målingene,  $k$  er samplingsnummeret, mens  $n$  er antall samples til sammen. Vektorene  $\mathbf{y}$  og  $\mathbf{y}_m$  inneholder henholdsvis alle modellverdier og målinger for hver sample. Matrisen  $\mathbf{Q}$  inneholder vekten av hver variabel, som brukes for å gjøre kostfunksjonen dimensjonsløs og kan brukes til å vekte mellom ulike målinger og samples.

Kostfunksjonen  $J(\theta)$  er summen av kvadratet av differansen mellom estimert måling og faktisk måling, altså kvadratavviket. En ting å legge merke til, er at kostfunksjonen bare er avhengig av parameterne. Målingene er implisitt gitt av parameterne gjennom modelligningene.

Optimeringsproblemet løses ved bruk av SQP-algoritmen, som beskrives av Nocedal og Wright[20] i kapittel 18. SQP-algoritmen approksimerer kostfunksjonen med en kvadratisk funksjon og linearisering av restriksjonene. Det kvadratiske subproblemet løses for hver iterasjon til konvergenzkriteriet er tilfredsstillt.

## 4.2 Metode

Ulike parametere ble forsøkt estimert for å finne den beste kombinasjonen av parametere som gav den beste tilpasningen mellom modell- og prosessutganger. Målet var å få modellen til å ligne mest mulig på målingene, uten å få med støy. I begynnelsen ble det antatt at  $V_B$  og  $V_{Aa}$  var konstanter, men i 4.2.6 endres konstantene til parametere for å gi en enda bedre tilpasning mellom modell og måling.

### 4.2.1 Implementering og initiale verdier

Modellen til Ottermo ble først lagt inn i Visual Studios ved å fylle inn i en template<sup>1</sup> fra Cybernetica, kompilert til en .dll-fil og deretter lagt inn i ModelFit. For å fylle inn i templateen var det nødvendig med en definisjon av parametere, konstanter, inngangsverdier,

---

<sup>1</sup>Ferdilagede filer, der modelligninger og spesifikk informasjon om modellen legges inn

tilstander og målinger for modellen til Ottermo, som er gitt i appendiks B. Målte data og inngangsverdier fra griseforsøk, gitt i appendiks C.1, ble også lagt inn i ModelFit.

De initielle, gjattede parameterverdiene ble anslått ved informasjon fra artikkel [1]. I tabell 11 i artikkelen [1], som forøvrig er gjengitt i appendiks D, står det at forbruk av  $O_2$  og produksjon av  $CO_2$  hos gris er henholdsvis 4,4-9,2 ml/min/kg og 3,4-7,6 ml/min/kg. Ved å regne om til verdier for grisen som ble brukt i forsøk A<sup>2</sup>, gav det verdiene:

$$MBO_2 = 4,4 \cdot 10^{-3} \cdot 35\text{kg} - 9,2 \cdot 10^{-3} \cdot 35\text{kg} = 0,154 - 0,322 \text{ L/min}$$

$$MBCO_2 = 3,4 \cdot 10^{-3} \cdot 35\text{kg} - 7,6 \cdot 10^{-3} \cdot 35\text{kg} = 0,119 - 0,266 \text{ L/min}$$

Gjennomsnittsverdien av disse intervallene ble valgt som initielle verdier for modellen. Også verdier for diffusjonskoeffisienten  $D$  og slagvolumet  $SV$  ble inspirert fra denne artikkelen. I begynnelsen ble det antatt at  $V_B$  og  $V_{Aa}$  var konstanter, men ble endret til parametere etter hvert. Initielle parameterverdier er gitt i tabell 4.1.

Initielle parameterverdier		
Parameter	Initiell, gjettet parameterverdi	Enhet
$MBCO_2$	0.193	L/min
$MBO_2$	0.238	L/min
$D$	0.9	-
$SV$	0.05	L/slag
$V_B$	36	L
$V_{Aa}$	4	L

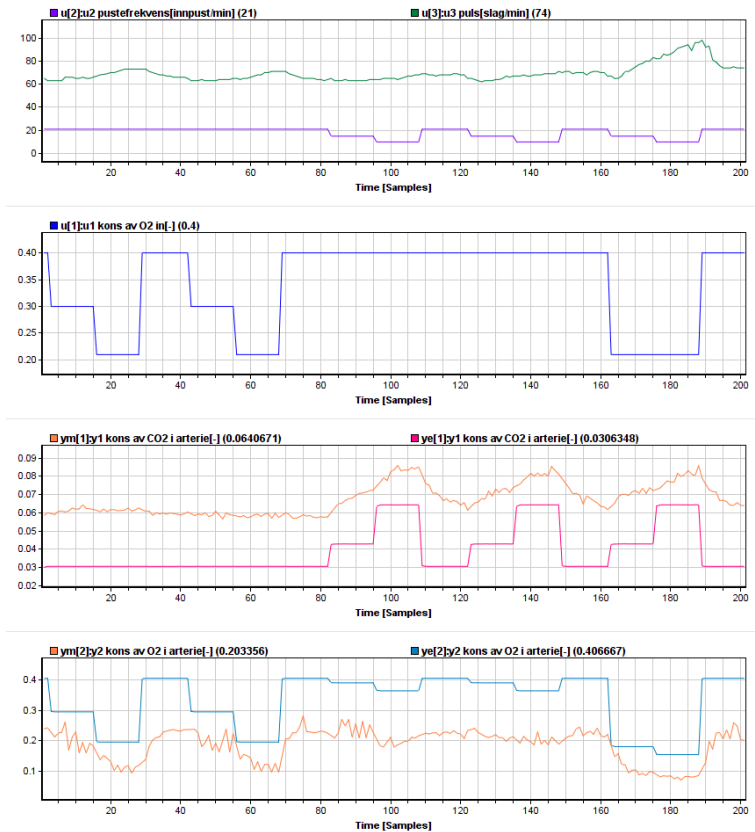
**Tabell 4.1:** Tabellen viser initielle parameterverdier. De fire øverste verdiene er hentet fra [1].  $V_B$  og  $V_{Aa}$  er hentet fra [15]. I begynnelsen ble  $V_B$  og  $V_{Aa}$  antatt som konstanter, før de etter hvert ble endret til parametere.

## 4.2.2 Parameterestimering

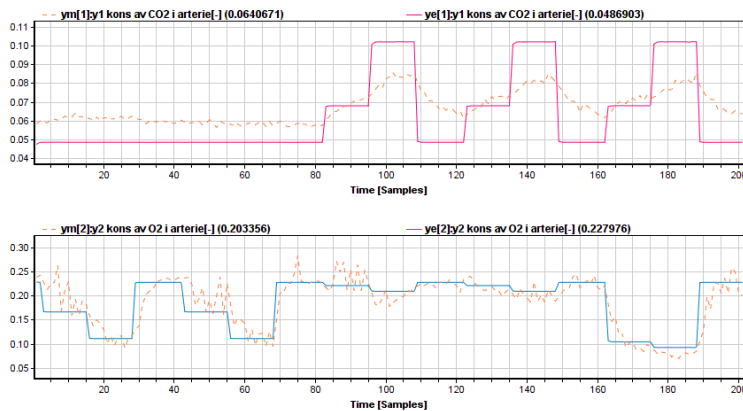
Ved offline parameterestimering ble verdier på diagonalen til  $Q$  i SQP-problemet satt til 1, slik at begge målingene ble vektet like mye. Ulike parametere ble valgt for parameterestimering, og ModelFit fant optimal verdi som minimerte  $J(\theta)$ .

Med bare to målinger og ikke så mye variasjon i datasettet, var det naturlig at det ikke var mulig å estimere alle parameterne. ModelFit krasjet ved forsøk på å optimere fire parametere eller mer, men også generelt ved optimering av  $SV$ .  $SV$  ble derfor ikke estimert, men antatt som en konstant parameter.

<sup>2</sup>Grisen som ble brukt i forsøket var 35 kg. Mer info om forsøket og verdier er gitt i appendiks C.1



**Figur 4.1:** Grafene viser inngangsverdier og utgangsverdier i ModelFit, som gir en sammenligning av modell og måling *før* parametere er optimalisert. Modellen har altså initielt gjettede parameterverdier. De to øverste grafene viser inngangsverdiene som påsettes både modell og prosess. I de to nederste grafene er utgangsverdier fra prosess og modell i samme graf,  $y_1 = [\text{CO}_2]_a$  øverst og  $y_2 = [\text{O}_2]_a$  nederst. Fra grafene er det tydelig at det er en del avvik mellom utgangsverdien fra modell og målingene fra prosessen. Verdiene som initielt var gjettet for parameterne er gitt i tabell 4.1. Målingene er verdier fra griseforsøk, gitt i appendiks C.1.



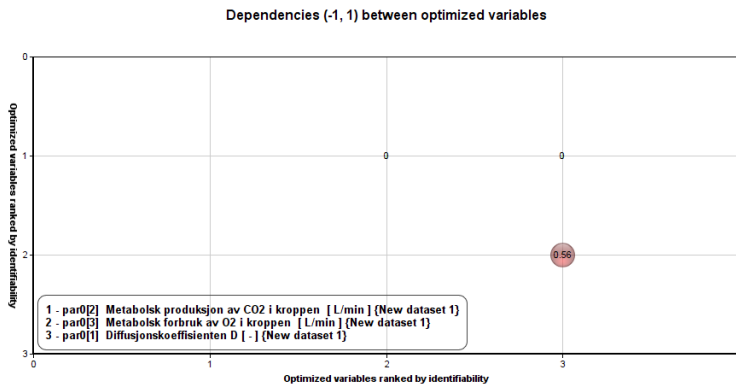
**Figur 4.2:** Her er modellen tilpasset til målingene ved offline parameterestimering av parametrene  $MBCO_2$ ,  $MBO_2$  og  $D$  i ModelFit. De nye verdiene på parametrene  $MBCO_2$ ,  $MBO_2$  og  $D$  er gitt i tabell 4.2, under kolonnen "Verdi fra ModelFit, forsøk 1".

### 4.2.3 Estimering av $MBCO_2$ , $MBO_2$ og $D$

Offline parameterestimering av  $MBCO_2$ ,  $MBO_2$  og  $D$ , gav resultatet gitt i kolonnen «Verdi fra ModelFit, forsøk 1» i tabell 4.2. Figur 4.2 viser målinger og utgangsverdi for modell med nye, tilpassede parametre, og viser en betydelig forbedring sammenlignet med figur 4.1. Modellen var tilpasset bra til målingene, men gav avhengighet mellom to av parametrene, indikert som en rød sirkel i «Dependencies»-grafene i ModelFit. «Dependencies»-grafene er gitt i figur 4.3. Linnestad skrev i mail:

«Rød sirkel betyr at parametrene som optimeres er lineært avhengig av hverandre og at det derfor er vanskelig å bestemme korrekt verdi for dem uavhengig. Det kommer av at man optimerer for mange parametre med de målingene man har. Man vil kunne få omtrent like godt resultat om man fjerner en av parametrene fra de valgte parametrene.»

På dette tidspunktet var  $V_B$  og  $V_{Aa}$  antatt som konstanter, og ikke vurdert som mulige kandidater for parameterestimering. Det var derfor bare fire aktuelle parametre som var aktuelle for estimering. På grunn av avhengighet mellom parametre ved estimering og at  $SV$  ikke kunne optimeres, ble det nødvendig å velge ut bare to parametre til offline parameterestimering. Fordi både  $MBO_2$  og  $D$  hadde med oksygen å gjøre og ville gi avhengighet, falt valget på en av disse i kombinasjon med  $MBCO_2$ .



**Figur 4.3:** Figuren viser at ved offline parameterestimering av parametrene  $D$ ,  $MBCO_2$  og  $MBO_2$  vil det være en avhengighet mellom parametrene  $MBCO_2$  og  $MBO_2$ . Avhengigheten kan skyldes at for mange parametere estimeres i forhold til målinger, men her er nok grunnen at begge parametrene har med  $O_2$  å gjøre.

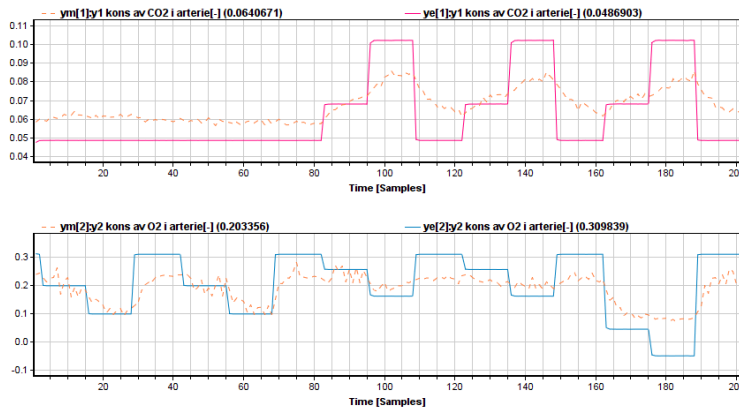
Parameterverdier					
Parameter	Initiell, gjetted verdi (-)	Verdi fra ModelFit, forsøk 1	Verdi fra ModelFit, forsøk 2	Verdi fra ModelFit, forsøk 3	Verdi fra ModelFit, forsøk 4
$MBCO_2$	0.193	0.3067	0.3067	0.3067	0.3231
$MBO_2$	0.238	0.1053	0.8480	–	–
$D$	0.9	1.6348	–	1.4069	1.4083
$SV$	0.05	–	–	–	–
$V_B$	36	–	–	–	1126
$V_{Aa}$	4	–	–	–	–

**Tabell 4.2:** Tabellen viser resultatene av offline parameterestimering og initielle, gjettede verdier på parametrene. Bindestreken angir at parameteren ikke er valgt for optimering og har samme verdi som initielt gjetted. Før forsøk 4 ble det antatt at  $V_B$  og  $V_{Aa}$  var konstanter.

#### 4.2.4 Estimering av $MBCO_2$ og $MBO_2$

Ved å benytte offline parameterestimering i ModelFit, for parametrene  $MBCO_2$  og  $MBO_2$ , ble parametrene endret til verdier gitt i kolonnen «Verdi fra ModelFit, forsøk 2» i tabell 4.2. Figur 4.4 viser utgangsverdiene fra modell og prosess med tilpassede parametere for modellen.

Verdiene som ble estimert av ModelFit for  $MBCO_2$  og  $MBO_2$  var ikke realistiske verdier, men gjorde at modellen ble tilpasset ganske greit til målingene. Tilstandene til modellen med disse parameterverdiene tok også verdier som ikke var helt realistiske. Særlig  $[O_2]_B$ , som ofte tok verdien 0, var ikke særlig realistisk. Dette gjorde det nødvendig å forsøke å estimere andre parameterpar.



**Figur 4.4:** Her er modellen tilpasset til målingene ved offline parameterestimering i ModelFit. De nye verdiene på parameterne  $MBCO_2$  og  $MBO_2$  er gitt i tabell 4.2, under kolonnen "Verdi fra ModelFit, forsøk 2".

I forsøket ble grisen presset både på oksygenmengde  $[O_2]_{in}$  og pustefrekvens  $F_T$ . I et forsøk for å se om de høye parameterverdiene skyldtes at grisen ble simulert til å ha KOLS, ble bare målinger der pustefrekvensen  $F_T = 21$  og  $[O_2]_{in} = 0,21$ , som er normalverdier for gris [1], brukt. Ved offline parameterestimering med disse målingene, gav ModelFit  $MBCO_2 = 0,377$  og  $MBO_2 = 0,6817$ . Disse verdiene var også høyere enn det som angis som normalverdier for  $MBCO_2$  og  $MBO_2$  i artikkelen [1].

#### 4.2.5 Estimering av $MBCO_2$ og D

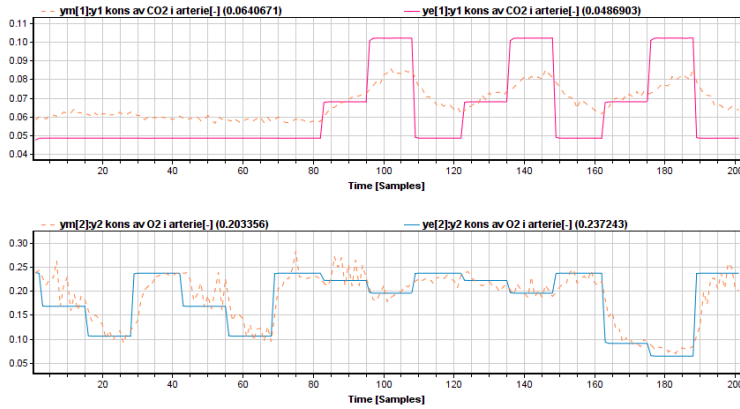
Ved å benytte offline parameterestimering i ModelFit, for parameterne  $MBCO_2$  og D, ble parameterne endret til verdier gitt i kolonnen «Verdi fra ModelFit, forsøk 3» i tabell 4.2. Figur 4.5 viser utgangsverdiene fra modell og prosess med tilpassede parametere for modellen. For disse parameterverdiene er også tilstandene realistiske verdier, og holder seg innenfor intervallet 0-1.

#### 4.2.6 Estimering av $MBCO_2$ , D og $V_B$

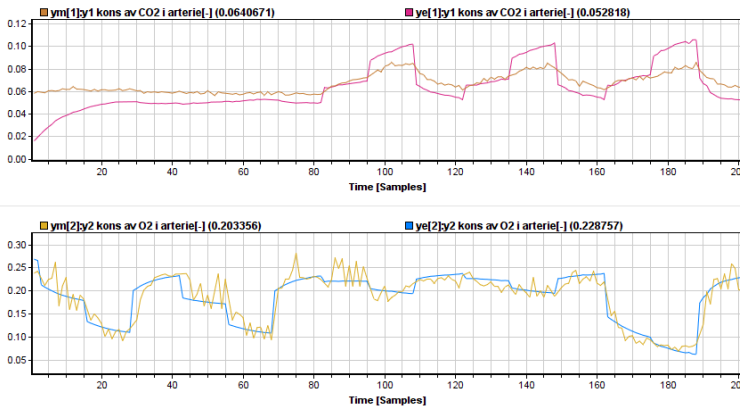
I figur 4.5 er det tydelig at dynamikken til modellen var raskere enn dynamikken til prosessen og at  $[CO_2]_a$  fra modellen avvok en del fra målingene. Etter samtale med veileder Stavadahl ble det foreslått å prøve å også estimere tidligere antatte konstant  $V_B$ , i et forsøk på å bedre tilpasse  $y_1 = [CO_2]_a$  mellom modell og prosess. Dette fordi en bedre modell gir en bedre regulering. Estimering av parameteren  $V_B$  ble foreslått fordi dynamikken til  $[CO_2]_B$  er avhengig av størrelsen på  $V_B$ , som kan ses ved å studere ligning 2.27.

Ved å endre konstantene til parametere, og estimere  $MBCO_2$ , D og  $V_B$  med Cybernetica ModelFit, ble parameterne  $MBCO_2$ , D og  $V_B$  endret til verdier gitt i kolonnen «Verdi fra





**Figur 4.5:** Her er modellen tilpasset til målingene ved offline parameterestimering i ModelFit. De nye verdiene på parameterne  $MBCO_2$  og  $D$  er gitt i tabell 4.2, under kolonnen "Verdi fra ModelFit, forsøk 3".



**Figur 4.6:** Her er modellen tilpasset til målingene ved offline parameterestimering i ModelFit. De nye verdiene på parameterne  $MBCO_2$ ,  $D$  og  $V_B$  er gitt i tabell 4.2, under kolonnen "Verdi fra ModelFit, forsøk 4". Her er det tydelig at grafen for  $[O_2]_a$  var bedre tilpasset enn i tidligere tilfeller. Legg merke til at avviket mellom  $[CO_2]_a$  fra prosess og modell for høye verdier er omtrent lik som tidligere.

ModelFit, forsøk 4» i tabell 4.2. Figur 4.6 viser utgangsverdiene fra modell og prosess med tilpassede parametere for modellen.

## 4.3 Resultat

Offline parameterestimering av  $MBCO_2$ ,  $D$  og  $V_B$  gav den beste tilpasningen av modellen til målingene, og ble derfor valgt som resultat. Verdien på parameterne  $MBCO_2$ ,  $D$  og  $V_B$  er gitt i tabell 4.2, under «Verdi fra ModelFit, forsøk 4». Utgangsverdier fra modell og prosess med disse parameterne er gitt i figur 4.6.

## 4.4 Diskusjon

Ved å studere figur 4.1, som viser inngangsverdier, målte utgangsverdier og utgangsverdier fra modellen uten optimerte parametere, var det tydelig at det var en del avvik mellom utgangsverdiene fra modell og prosess. Alle forsøk på offline parameterestimering gav bedre tilpasning mellom modell og prosess enn med parameterverdier initielt gjetet.

I dette delkapittelet diskuteres de ulike parameterestimeringene, betydningen av urealistiske parameterverdier og resultatet.

### 4.4.1 Estimering av to parametere

Estimering av to parametere gav ingen avhengighet for valgte parametere, men urealistiske verdier for parameterne. Ved offline parameterestimering ble parameteren  $MBCO_2$  estimert til 0,3067, for de første estimeringsforsøkene. Denne verdien var høyere enn det som er angitt i [1], men allikevel innenfor intervallet 0-1.

Ved estimering av  $MBO_2$ , ble parameteren estimert til en høyere verdi enn det som er indikert som vanlige verdier for gris av [1]. En høy verdi på  $MBO_2$  indikerer at veldig mye av  $O_2$  blir brukt opp i kroppen, og fører til at tilstanden  $[O_2]_B$  ofte blir 0. Dette er ikke realistisk, men fører til en ganske bra tilpasning mellom utgangsverdier fra modell og prosess.

Estimering av diffusjonskoeffisienten  $D$  gav parameterverdien som var mest urealistisk. Artikkelen om gris [1] sier at verdien skal ligge på mellom 0,67-0,96, mens ModelFit estimerte verdien til omtrent 1,5.  $D$  angir mengden  $O_2$  som diffunderer over fra lungene til blodet. Med en diffusjonskoeffisient på over 1, indikerer det at mer enn 100% av oksygenen i lungene går over i blodet. Modellen med  $D \approx 1,5$  gav derimot en veldig bra tilpasning av modellen til målingene og alle tilstandene hadde realistiske verdier.

## 4.4.2 Estimering av flere parametere

I teoridelen i 4.1 står det at med nok variasjon i datasettet skulle være mulig å estimere flere parametere enn det var målinger. Det ble bare brukt to målinger fra forsøket på gris, der ingen av dem var veldig varierende. Det var derfor ikke rart at det ikke var mulig å estimere fire parametere. Estimering av parameterne  $MBO_2$  og  $D$ , som begge hadde med oksygen å gjøre, gav naturlig nok en avhengighet mellom de to parameterne.

Parameterne som ikke ble estimert var som konstanter å regne. Ved å inkludere konstantene  $V_B$  og  $V_{Aa}$  som parametere, ble det mulig å estimere tre parametere – en volumparameter, en parameter som hadde med oksygen å gjøre og en som hadde med  $CO_2$  å gjøre.

Ved å bruke flere målinger fra griseforsøket hadde det kanskje vært mulig å estimert en ekstra parameter, f.eks.  $SV$ . I følge [1] skulle  $SV$  ha verdi på mellom 0,036-0,065 L/slag. I ligningene for modellen var det bare blodstrømmen  $\dot{Q}$  som var avhengig av  $SV$ .  $\dot{Q}$  var også avhengig av pulsen. Pulsen varierte, som gjorde at  $\dot{Q}$  varierte selv om  $SV$  var antatt konstant. Ved å øke  $SV$  til 0,5 eller minke til 0,005 gav det ingen synlige endringer i utgangsverdiene til modellen. Siden små, realistiske variasjoner i  $SV$  ikke ville gitt store utslag på utgangsverdien, ville nok ikke estimering av parameteren  $SV$  gitt et betydelig forbedring av modell i forhold til prosess.

For at utgangsverdiene skulle være merkbart annerledes måtte  $SV < 0,0001$  eller  $> 0,6$ . Å sette  $SV = 0,05$  for alle individ var en forenkling, men fungerte.  $SV$  hadde altså ikke så mye å si for utgangsverdiene, og var derfor ok å anta konstant.

## 4.4.3 Ikke realistiske parameterverdier

Ved offline parameterestimering ble alle parameterne estimert til verdier som ikke var helt realistiske, men som gjorde at modellen ble bedre tilpasset til målingene enn uten offline parameterestimering.

Realistiske parameterverdier ville vært viktig i situasjoner der det er ønskelig å lære mer om prosessen, så om parameterne bør ha realistiske verdier kommer litt an på formålet med modellen. Er formålet å oppnå økt forståelse i prosessen, gjennom f.eks. simuleringer, vil det være nyttig med realistiske parameterverdier. I reguleringstekniske problemer, slik som i denne oppgaven, vil det være mye viktigere at modellen stemmer så godt som mulig – selv på bekostning av urealistiske parameterverdier.

Ved regulering av systemet er det utgangsverdiene som er av interesse. Parameterne i seg selv har ingenting å si for reguleringen, men påvirker utgangsverdien som blir brukt i reguleringen. Dersom verdien på utgangen følger målingene bedre med parametere som ikke er helt slik de burde, er det bedre enn parametere som er realistiske, men som gir stor differanse mellom utgang fra prosess og modell.

Det kan være flere grunner til at parameterne, som ble estimert ved offline parameterestimering i ModelFit, ble estimert til urealistiske verdier. Verdiene som er gitt for  $MBCO_2$ ,  $MBO_2$  og  $D$  i artikkel [1], er gitt for en bevisst gris. Grisen som ble brukt i forsøket, som

dataene i appendiks C.1 er fra, var en bevisstløs gris som ble simulert til å ha KOLS. Grisen var naturlig nok under stress og det var derfor ikke uventet om verdier fra denne grisen ville avvike fra normalverdiene til en frisk, bevisst gris. Dette ville forklart små avvik fra verdiene som er foreslått i [1], men ville ikke forklart de store avvikene. Parameterverdiene avviker mye, selv for målinger der normale inngangsverdier for gris, dvs.  $F_T = 21$  og  $[O_2]_{in} = 0,21$ , ble brukt. Ingen gris, verken bevisst eller ubevisst, ville f.eks. hatt en diffusjonskoeffisient på over én. Avvikene kan nok derfor ikke skyldes grisens ubevissthet.

Målestøy kan ha påvirket målingene. Biveileder Ottermo sa at målingene skulle være ganske nøyaktige, men at ved måling av  $O_2$  var det viktig at det var vakuum i sprøyten før prøven ble tatt. Ved å ufrivillig suge inn litt luft fra omgivelsene ville det påvirke prøven slik at målingen av  $[O_2]_a$  avvek fra korrekt verdi. Det er ikke umulig at dette kan ha forekommet under forsøket. Likevel ville nok selv ikke en perfekt måling uten målestøy, gitt verdier for målingene som hadde passet perfekt med modellen.

En annen, og sannsynligvis den mest realistiske grunnen til at estimerte parameterverdier var urealistiske, kan være at modellen som ble brukt er en forenkling av modellen til Eldhuset, som igjen er en forenkling av virkeligheten. Det er umulig å få med seg all dynamikk til en prosess i en modell. I modellen til Ottermo er det gjort en rekke antagelser og forenklinger som garantert har ført til at modellen har mangler eller feil i forhold til virkeligheten. Parameterverdier som ikke er helt realistiske kan kompensere for en modell som kanskje er litt for enkel. Men *litt for høye* parameterverdier så fungerer jo modellen veldig bra. En modell med gode utgangsverdier, selv med urealistiske parametere, var det viktigste.

#### 4.4.4 Resultatet

Som beskrevet tidligere i diskusjonen var formålet med parameterestimeringen i denne oppgaven av reguleringsteknisk art. Interessen var ikke å lære mer om systemet som ble regulert, men tilpasse modellen til målingene for å kunne regulere prosessen på en god måte. Det var derfor ikke uheldig at estimerte parameterverdier tok urealistiske verdier. En godt tilpasset modell gir bedre prediksjon som igjen gir bedre regulering.

Resultatet, med  $D = 1,408$ ,  $MBCO_2 = 0,323$  og  $V_B = 1126$ , hadde urealistisk høye estimerte parameterverdier, men gav en veldig god tilpasning mellom modellutganger og målinger. Med bakgrunn i det som er diskutert tidligere i diskusjonen, anses resultatet som et godt resultat. I tillegg gav denne parameterestimeringen mer realistiske tilstander enn andre parameterestimeringer, som f.eks. i 4.2.4 hvor tilstandene ofte ble 0.

Ved første øyekast kunne det se ut som resultatet gav en mye bedre estimering av  $[CO_2]_a$  enn ved offline parameterestimering av parameterne  $MBCO_2$  og  $D$ . Ved å sammenligne figur 4.5 og 4.6 ble det klart at avviket mellom måling og utgangsverdi fra modellen for de høyeste verdiene av  $[CO_2]_a$  var ganske like. Forbedringen med estimering av  $MBCO_2$ ,  $D$  og  $V_B$  var altså ikke veldig mye bedre enn ved estimeringen av  $MBCO_2$  og  $D$ . Forskjellen var forbedringen av dynamikk i modell i forhold til måling. Ved offline parameterestimering av parameterne  $MBCO_2$ ,  $D$  og  $V_B$  passet dynamikken til modellen bedre til målingene enn for de andre forsøkene av offline parameterestimering.

Når det gjelder estimering av parameteren  $V_B$  hadde det nok vært mer realistisk om  $V_{Aa}$  hadde vært estimert i stedet for. En veldig liten  $V_{Aa}$  ville antageligvis gitt samme effekt som å øke  $V_B$ , da øking av  $V_B$  ikke gav veldig mye bedre resultater for  $[\text{CO}_2]_a$ . Estimering av parameteren  $V_{Aa}$  ville ikke la seg gjøre i ModelFit. Siden det ikke hadde noe å si at parameterverdiene ikke var realistiske, hadde det ikke så mye å bety for denne oppgaven. Ved andre formål enn regulering, hadde det kanskje vært nyttig å undersøkt estimering av  $V_{Aa}$  nærmere. Det er ikke sikkert det ville gitt bedre tilpasning mellom modell og prosess, men det ville antageligvis gitt mer realistiske parameterverdier. I alle fall for volumene  $V_B$  og  $V_{Aa}$ , hvor  $V_{Aa}$  antageligvis er ganske liten og  $V_B$  garantert ikke like stor som estimert av ModelFit.

Resultatet gir en god tilpasning mellom utgangsverdier for modell og målinger fra prosess, som gir et godt utgangspunkt for online parameterestimering i neste kapittel. Det er ikke nødvendig å estimere flere parametere da resultatet gir en god nok tilpasning mellom modell og prosess, men det hadde nok vært mulig å gjort endringer som kanskje hadde gitt et like godt, eller bedre, resultat. En endring av modellen der  $[\text{CO}_2]_a$  ikke er like avhengig av  $F_T$  som i nåværende modell, særlig for  $F_T$  langt fra  $F_T = 15$ , er et slikt eksempel.

# Kapittel 5

## Online parameterestimering

Modeller er en forenkling av virkeligheten, og vil generelt ikke kunne reprodusere de faktiske målte utgangsverdiene fra prosessen perfekt. Det vil derfor være forskjeller mellom målinger og målinger som er prediktert av modellen, selv med offline parameterestimering som ble gjort i forrige kapittel. Dette kan til en viss grad rettes opp av rekursive tilstandsestimeringsteknikker, hvor tilstanden oppdateres basert på differansen mellom prediktert måling og faktisk måling. Det finnes flere slike teknikker, som blant annet Kalmanfilter eller MHE<sup>1</sup>. [18]

Kalmanfilter er det som brukes videre i denne oppgaven, og i dette kapitlet tunes Kalmanfilteret i åpen sløyfe. Filteret ble brukt sammen med modellen, slik at det var mulig å følge målingene fra prosessen mer presist. I etterfølgende delkapittel gis det en kort innføring i teori knyttet til denne typen estimator, for oversikt, innføring og at leseren enklere skal kunne skjønne resultatene og diskusjonen i slutten av kapitlet.

### 5.1 Teori

Ulusoy [21] beskriver et generalisert system der  $\mathbf{u}$  er inngangsverdiene,  $\mathbf{y}$  er utgangsverdiene og  $\mathbf{x}$  representerer de tilstandene det er ønskelig å estimere. Målet er å drive den estimerte tilstanden  $\hat{\mathbf{x}}$  til faktisk tilstand  $\mathbf{x}$ , der den matematiske feilen er definert som:

$$\mathbf{e} = \mathbf{x} - \hat{\mathbf{x}} \quad (5.1)$$

Systemet beskrives av ligning (5.2), mens observatoren beskrives av ligning (5.3).

---

<sup>1</sup>Moving Horizon Estimator/Estimation

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \qquad \mathbf{y} = \mathbf{C}\mathbf{x} \qquad (5.2)$$

$$\dot{\hat{\mathbf{x}}} = \mathbf{A}\hat{\mathbf{x}} + \mathbf{B}\mathbf{u} + \mathbf{K}(\mathbf{y} - \hat{\mathbf{y}}) \qquad \hat{\mathbf{y}} = \mathbf{C}\hat{\mathbf{x}} \qquad (5.3)$$

Feildynamikken,  $\dot{\mathbf{e}}$ , er gitt av differansen mellom disse ligningene, som gir:

$$\underbrace{\dot{\mathbf{x}} - \dot{\hat{\mathbf{x}}}}_{\dot{\mathbf{e}}} = \underbrace{\mathbf{A}\mathbf{x} - \mathbf{A}\hat{\mathbf{x}}}_{\mathbf{A}\mathbf{e}} + \underbrace{\mathbf{B}\mathbf{u} - \mathbf{B}\mathbf{u}}_0 - \underbrace{\mathbf{K}(\mathbf{y} - \hat{\mathbf{y}})}_{\mathbf{K}(\mathbf{C}(\mathbf{x} - \hat{\mathbf{x}}))} \qquad \mathbf{y} - \hat{\mathbf{y}} = \mathbf{C}(\mathbf{x} - \hat{\mathbf{x}}) \quad (5.4)$$

$$\Rightarrow \dot{\mathbf{e}} = (\mathbf{A} - \mathbf{K}\mathbf{C}) \cdot \mathbf{e} \qquad \mathbf{y} - \hat{\mathbf{y}} = \mathbf{C}\mathbf{e} \quad (5.5)$$

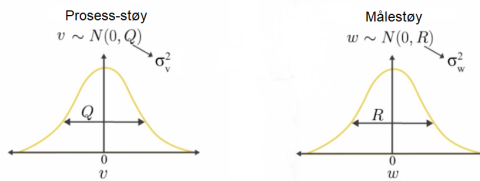
$$\Rightarrow \mathbf{e}(t) = e^{(\mathbf{A} - \mathbf{K}\mathbf{C})t} \cdot \mathbf{e}(0) \quad (5.6)$$

Løsningen på differensialligningen i (5.5) er en eksponensialfunksjon (5.6), som vil gå mot 0 dersom eksponenten er  $< 0$ .  $\mathbf{K}$  velges derfor slik at  $(\mathbf{A} - \mathbf{K}\mathbf{C}) < 0$ , som gjør at  $\mathbf{e} \rightarrow 0$  når  $t \rightarrow \infty$ , og dermed vil  $\hat{\mathbf{x}} \rightarrow \mathbf{x}$  når  $t \rightarrow \infty$ . Fordi  $\mathbf{y} = \mathbf{C}\mathbf{x}$  vil også  $\hat{\mathbf{y}} \rightarrow \mathbf{y}$  når  $t \rightarrow \infty$ .

### 5.1.1 Kalmanfilter

En optimal måte å velge  $\mathbf{K}$  på, gjøres ved hjelp av et Kalmanfilter – en optimal tilstandsestimator for stokastiske prosesser. Det konvensjonelle Kalmanfilteret er gitt for lineære systemer, med additivt støy for både tilstander og målinger slik som i ligning (5.7), hvor  $\mathbf{v}_k$  er prosessstøy mens  $\mathbf{w}_k$  er målestøy [22]. Støy kan representeres som gaussisk varierende signaler, slik som i figur 5.1.

$$\begin{aligned} \mathbf{x}_k &= \mathbf{A}\mathbf{x}_{k-1} + \mathbf{B}\mathbf{u}_k + \mathbf{v}_k \\ \mathbf{y}_k &= \mathbf{C}\mathbf{x}_k + \mathbf{w}_k \end{aligned} \quad (5.7)$$



**Figur 5.1:** Grafene i figuren viser normalfordelingen til støy for henholdsvis prosess og måling. Variansen til prosess-støyet er gitt av  $Q$ , mens variansen til målestøyet er gitt av  $R$ . Begge fordelingene har en forventningsverdi på 0. 68,3% av verdiene faller innenfor intervallet  $[\mu - \sigma, \mu + \sigma]$ , hvor  $\mu = 0$  er forventningsverdien [22]. Figuren er hentet fra video [22].

En matematisk beskrivelse av Kalmanfilteret er gitt av

$$\begin{aligned}\hat{\mathbf{x}}_k &= \mathbf{A}\hat{\mathbf{x}}_{k-1} + \mathbf{B}\mathbf{u}_k + \mathbf{K}_k(\mathbf{y}_k - \mathbf{C}(\mathbf{A}\hat{\mathbf{x}}_{k-1} + \mathbf{B}\mathbf{u}_k)) \\ &= \underbrace{\hat{\mathbf{x}}_k^-}_{\text{Prediksjon}} + \underbrace{\mathbf{K}_k(\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_k^-)}_{\text{Oppdatering}}\end{aligned}\quad (5.8)$$

der  $\hat{\mathbf{x}}_k^-$  kalles *a priori* estimat, mens  $\hat{\mathbf{x}}_k$  kalles *a posteriori* estimat. (5.8) kan minne mye om estimatoren i (5.3), men er for stokastiske system.

Algoritmen for et Kalmanfilter består av to steg. Steg en er prediksjon,

$$\hat{\mathbf{x}}_k^- = \mathbf{A}\hat{\mathbf{x}}_{k-1} + \mathbf{B}\mathbf{u}_k \quad (5.9a)$$

$$\mathbf{P}_k^- = \mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^T + \mathbf{Q} \quad (5.9b)$$

mens steg to er oppdatering av predikterte tilstander  $\mathbf{x}$ , Kalman-forsterkning  $\mathbf{K}$  og kovariansen til feilen  $\mathbf{P}$ , alle for tidsskritt  $k$

$$\mathbf{K}_k = \frac{\mathbf{P}_k^- \mathbf{C}^T}{\mathbf{C}\mathbf{P}_k^- \mathbf{C}^T + \mathbf{R}} \quad (5.10a)$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k(\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_k^-) \quad (5.10b)$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k\mathbf{C})\mathbf{P}_k^- \quad (5.10c)$$

Kalmanfilter-algoritmen er en rekursiv algoritme. For å estimere nåværende tilstand trenger ikke algoritmen all tidligere informasjon – det eneste som trengs er de estimerte tilstandene og kovariansmatrisen for feil fra det forrige tidssteget og nåværende måling [23]. Dette gjør algoritmen rekursiv.

## 5.1.2 Ulineære Kalmanfilter

Utvidelser av Kalmanfilteret er utviklet for ulineære modeller, slik som utvidet (eng.: *extended*) Kalmanfilter (EKF), uparfymert (eng.: *unscented*) Kalmanfilter (UKF) og divided difference filter fra Nørsgaard et al. [24].

I EKF lineariseres modellen ved å Taylorutvide nåværende tilstandsestimat, mens UKF propagerer en samling med punkter, kjent som sigmapunkter, gjennom de ulineære modelligningene [25]. Filtrene til Nørsgaard et al. [24] er basert på approksimasjoner ved polynomer i stedet for Taylorutvikling. I masteren til Linnestad [19], er det referert til [26], som hevder at disse filterne er ganske like som UKF, og kan anses som et spesialtilfelle av disse.

I *Cybernetica* sine verktøy er det Nørsgaard et al. [24] sine førsteordens divided difference (DD1) filter og andreordens divided difference (DD2) filter som brukes. I mail har Kasper Linnestad fra *Cybernetica* skrevet:



«I teorien skal DD2 være litt mer nøyaktig, men det er også mindre robust og forskjellen i nøyaktighet er ikke stor nok til at det er verdt å bruke DD2. Av den grunn brukes det i flest tilfeller DD1 i Cybernetica sine verktøy.»

Kolås et al. [27] foreslår en metode for hvordan støy kan legges til i en modell. Metode 3 fra Kolås et al. [27] er ganske lik som metoden som brukes av Cybernetica i ModelFit [19, 18]. Støyet er antatt å legges til modellen gjennom parametere, som gjør at balanseligningne<sup>2</sup> ikke brytes. I tillegg vil parameterne selv være estimert og ha lov til å variere med tiden. Tilstandsvektoren vil derfor være tilpasset de estimerte parametere, som gir et tilpasset Kalmanfilter. Parametere er derfor modellert som integrert hvitt støy, og modelligningene kan skrives som [19, 26, 27, 28]:

$$\mathbf{x}_k^- = \mathbf{F}(\mathbf{x}_{k-1}^+, \mathbf{u}_{k-1}, \tilde{\boldsymbol{\theta}}_{k-1}) \quad a \text{ priori tilstandsestimat} \quad (5.11a)$$

$$\tilde{\boldsymbol{\theta}}_{k-1} = \boldsymbol{\theta}_{k-1}^+ (1 + \mathbf{v}_{k-1}) \quad \text{prosessstøy via parameter} \quad (5.11b)$$

$$\boldsymbol{\theta}_k^- = \boldsymbol{\theta}_{k-1}^+ (1 + \mathbf{v}_{k-1}) \quad a \text{ priori parameterestimert} \quad (5.11c)$$

$$\mathbf{y}_k^- = \mathbf{g}(\mathbf{x}_k^-, \mathbf{u}_k, \boldsymbol{\theta}_k^-) + \boldsymbol{\omega}_k \quad \text{måleestimat} \quad (5.11d)$$

hvor + og - betegner henholdsvis *a posteriori* og *a priori* estimat. Slik som i det lineære Kalmanfilteret, vil ligningene i (5.11) evalueres etter at målinger fra forrige sample,  $k-1$ , er prosessert og gir *a posteriori* estimater for nåværende tidssteg,  $k$ . Når målingene fra nåværende sample er prosessert, evalueres ligning (5.12) som gir *a posteriori* estimatene.

$$\begin{bmatrix} \mathbf{x}_k^+ \\ \boldsymbol{\theta}_k^+ \end{bmatrix} = \begin{bmatrix} \mathbf{x}_k^- \\ \boldsymbol{\theta}_k^- \end{bmatrix} + \mathbf{K}_k (\mathbf{y}_k - \mathbf{y}_k^-) \quad \begin{array}{l} a \text{ posteriori tilstandsestimat} \\ a \text{ posteriori parameterestimert} \end{array} \quad (5.12)$$

$\mathbf{K}$  er Kalmanfilter-forsterkningen, som multipliseres med differansen mellom måling og estimert måling. Kalmanfilter-forsterkningen er avhengig av kovariansen av de utvidede tilstandene (eng: *augmented state*), som er estimert både *a priori* og *a posteriori* av filterne foreslått av Nørsgaard et al. [24] og Schei [29].

Støyen er antatt å være ukorrelert og normalfordelt [26, 27]:

$$\begin{array}{ll} \mathbf{v}_k \sim \mathcal{N}(0, \mathbf{R}_v) & \text{Gaussisk prosessstøy} \\ \mathbf{v}_k \sim \mathcal{N}(0, \mathbf{R}_v) & \text{Gaussisk parameterstøy} \\ \boldsymbol{\omega}_k \sim \mathcal{N}(0, \mathbf{Q}) & \text{Gaussisk målestøy} \end{array} \quad (5.13)$$

hvor  $\mathcal{N}(0, \sigma^2)$  indikerer en normalfordelt variabel med forventningsverdi 0 og varians  $\sigma^2$ . Kalmanfilteret vil vekte målingen med minst støy, så dersom variansen i prosess- og

---

<sup>2</sup>feks. massebalanse, energibalanse osv.

parameterstøy er høy, vil tilstand og parameter oppdateres mye og målingen ha mye å si. Stor varians i målestøy vil gi en lav oppdatering, fordi målingen da ikke kan stoles like mye på. I følge Cybernetica [18] kan standardavviket  $\sigma$  kan ses på som en parameter som kan brukes til tuning av filteret.

I følge Linnestad fra Cybernetica er det kun hensiktsmessig å estimere like mange parametre som målinger i en rekursiv estimator.

### 5.1.3 Tuning

Kalmanfilteret tunes ved å endre på standardavvikene til støy gitt i (5.13). Dersom støy på tilstandene velges for høyt vil målingene kunne gjenskapes perfekt, fordi tilstandene oppdateres slik at måling og modell matcher. Det vil også være mulig å gjenskape målingene perfekt dersom støy på parameterne er for høy, men da vil parameterne endres kraftig fra sample til sample, som igjen gir veldig dårlig prediksjon ved bruk av MPC. Ved prediksjon i MPC vil siste parameterverdi brukes for hele prediksjonshorisonen. Når parameterverdien varierer kraftig fra sample til sample vil det være helt tilfeldig hvilken verdi den tar og dermed gi en helt vilkårlig og urealistisk prediksjon. Dette gjør at optimeringen vil få forskjellig resultat fra sample til sample og det optimerte pådraget vil bli kraftig påvirket av støyen i parameterne. Dessuten er det i utgangspunktet ganske urealistisk at Kalmanfilteret følger målingene perfekt. Det tyder enten på en ekstremt god modell eller at Kalmanfilteret oppdaterer parameterne og tilstandene for mye. [18]

Det er ønskelig at parameterne varierer relativt glatt og jevnt fordi det gir bedre prediksjonsegenskaper som MPCen avhenger av. Hvor mye prosessstøy som skal velges avhenger av applikasjonen. Dersom prosessen har mange forstyrrelser som modellen ikke klarer å fange opp kan det være lurt med litt høy prosessstøy for at tilstandene skal endre seg raskt slik at man får inn effekten av forstyrrelsene.

I ModelFit vil det være mulig å endre både på målestøy, prosessstøy og parameterstøy.

## 5.2 Metode

I dette delkapittelet ble to parametere,  $MBCO_2$  og  $D$ , valgt og standardavvikene for respektive parametere ble endret for å tilpasse modellen best mulig til målingene. Kalmanfilteret ble testet på modeller med andre parameterverdier for å kunne si noe om betydningen av offline parameterestimering fra forrige kapittel.

Ved tuning av Kalmanfilteret ble en og en verdi endret til parameterne holdt seg ganske stabil og utgang fra modellen fulgte målt utgang tilfredsstillende. Det var ønskelig at utgangsverdiene fulgte målingene, uten å replikere for mye av målestøyen. Som beskrevet under avsnittet 5.1.3, var det også ønskelig at parameterne varierte relativt glatt og jevnt for å gi bedre prediksjonsegenskaper.

En figur av online parameterestimering før tuning av filteret, som vil si at alle standardavvik var  $10^{-4}$ , er gitt i figur 5.2.

To parametere,  $MBCO_2$  og  $D$ , ble valgt til online parameterestimering, fordi det bare var to målinger fra prosessen. Parameterparet ble valgt fordi parameterne har med ulike konsentrasjoner å gjøre,  $MBCO_2$  med  $[CO_2]$  og  $D$  med  $[O_2]$ . Det hadde også vært aktuelt å brukt kombinasjonen  $MBCO_2$  og  $V_B$  eller  $D$  og  $V_B$ , men uten å gjøre endringer på standardavviket så det ut som  $MBCO_2$  og  $D$  gav den beste estimeringen. Filteret ble tunet manuelt i ModelFit ved å justere på standardavvikene til prosess- og parameterstøy for respektive parametere, samt målestøy.

For standardavviket til målestøyet ble default-verdier  $= 10^{-4}$  brukt for måling av  $[CO_2]_a$  og 0,0003 for måling av  $[O_2]_a$ . For enkelhetsskyld ble det antatt at disse standardavvikene var ganske små, som dermed vil si at målingene ble antatt ganske gode. For ikke-estimerte parametere ble default-verdier for parameterstøy valgt, dvs.  $std\_v = 10^{-4}$ .

### 5.2.1 Endring av std.avvik for parameter $MBCO_2$

Ulike standardavvik-verdier for parameterstøy til  $MBCO_2$  ble testet. Default-verdien til  $std\_v(MBCO_2)$  var satt til  $10^{-4}$ , og ble satt gradvis høyere til tilfredstillende resultater for utgangsverdier og parametere, henholdsvis  $y_1 = [CO_2]_a$  og  $MBCO_2$ , ble oppnådd. Figur 5.3 viser grafene ved valgt standardavvik,  $std\_v(MBCO_2) = 0,0004$  og målestøy som beskrevet.

### 5.2.2 Endring av std.avvik for parameter $D$

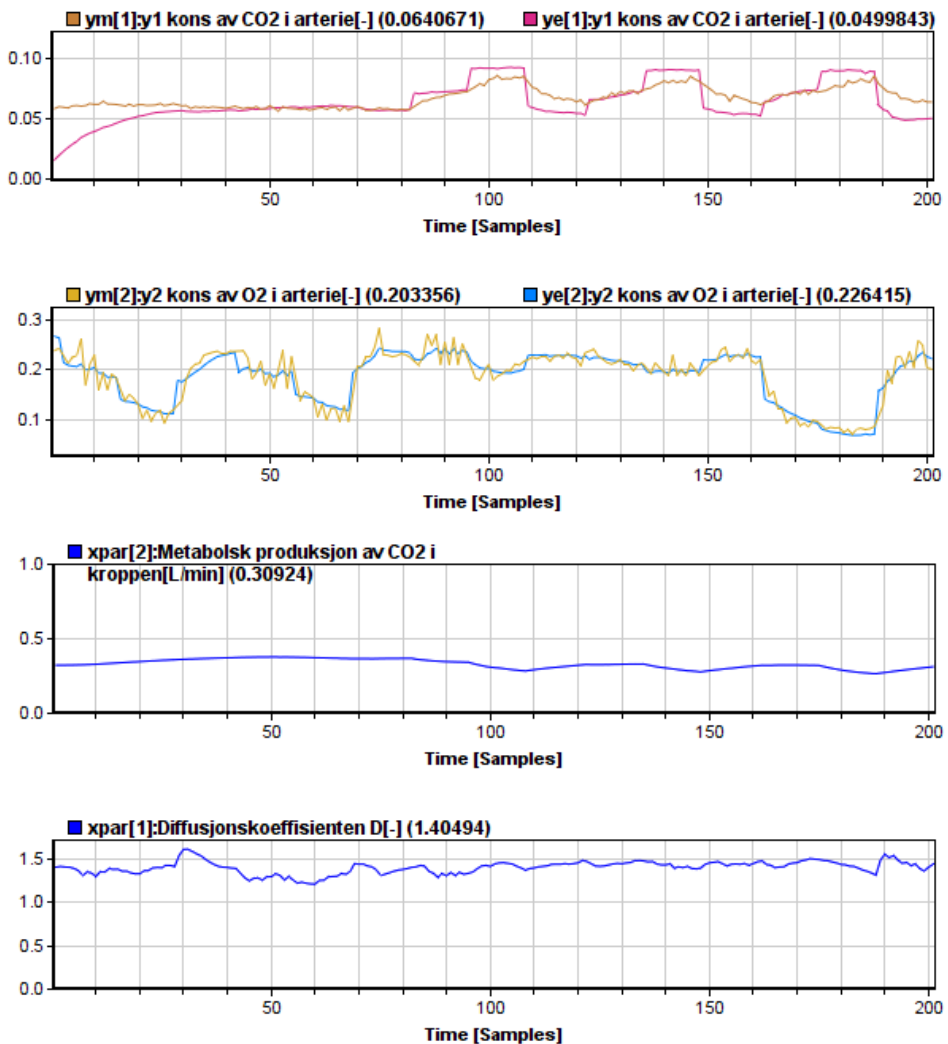
Estimert utgangsverdi for  $[O_2]_a$  var tilfredstillende med default-verdi for standardavviket og standardavvik for målestøy på 0,0003. Figur 5.4 viser grafene for valgt standardavvik,  $std\_v(MBCO_2) = 0,0001$  og  $std\_w(O_2a) = 0,0003$ . Fra figuren kan en se at støy ikke replikeres og at estimert parameterverdi varierer relativt glatt.  $std\_v(D)$  ble derfor ikke endret.

### 5.2.3 Online parameterestimering med initiell modell

For å se verdien av offline parameterestimering, ble initiell modell – altså modell som var gitt før offline parameterestimering – lagt inn i ModelFit med samme standardavvik som valgt over. Utganger og parametere fra modell og måling er vist i figur 5.5. Standardavvikene er gitt i tabell 5.1 og 5.2.

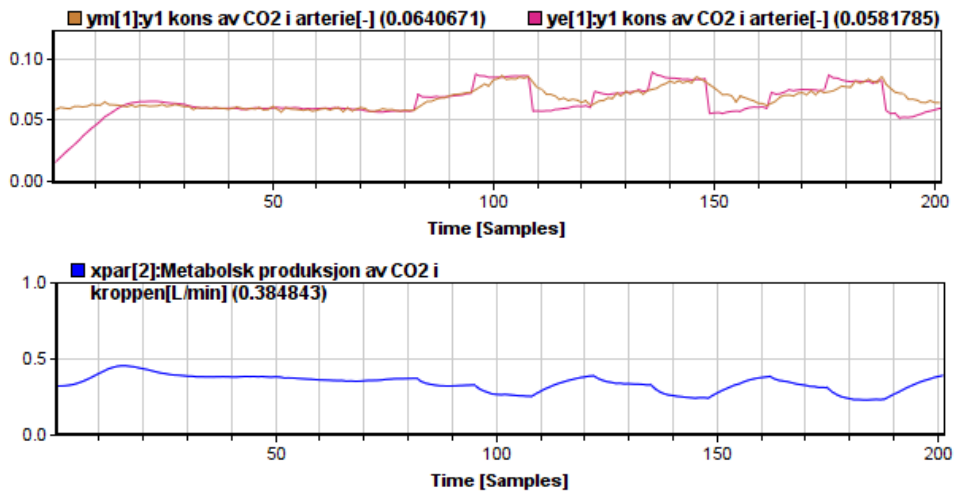
### 5.2.4 Online parameterestimering for griseforsøk B

For å sjekke om online parameterestimering bare var god for målingene som Kalmanfilteret var tunet etter, ble data fra et annet griseforsøk, griseforsøk B, også testet.

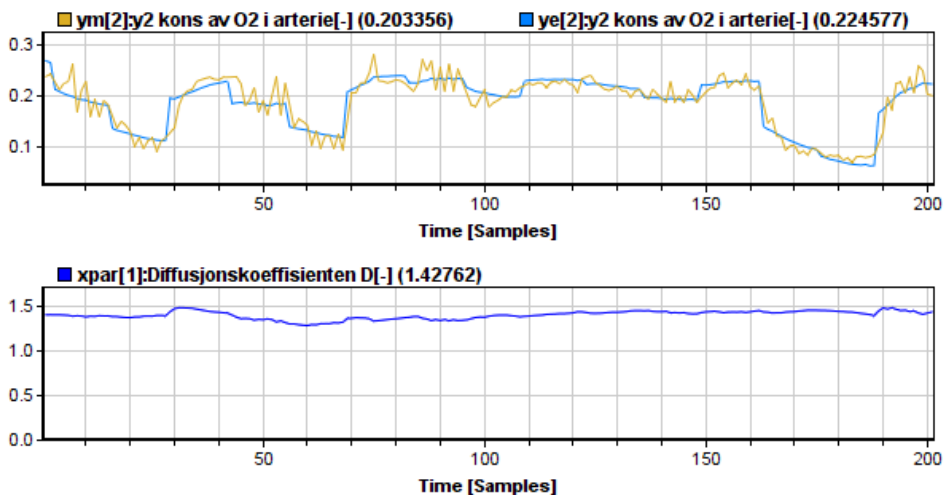


**Figur 5.2:** Figuren viser utgangsverdier fra måling og estimerte utgangsverdier fra Kalmanfilteret øverst, mens de nederste grafene viser parameterne som er estimert av Kalmanfilteret. Her er ikke Kalmanfilteret tunet, dvs. det er ikke gjort noen endringer på standardavvik for prosess- eller målestøy som alle har verdi  $10^{-4}$ .

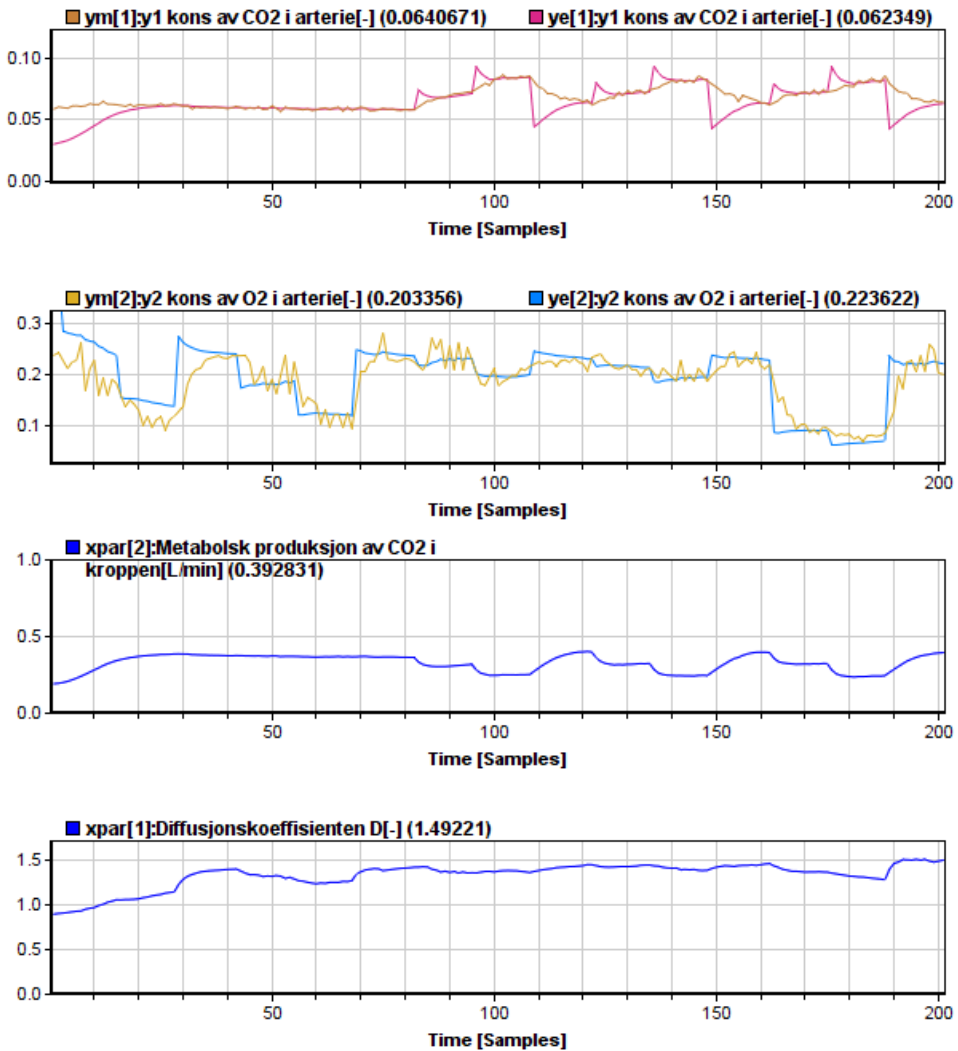
Ved å legge målingene og inngangsverdiene fra griseforsøk B i appendiks C.2 inn i ModelFit, med parameterverdier bestemt av resultatene fra offline parameterestimering fra forrige kapittel og standardavvik gitt i tabell 5.1 og 5.2, ble utgangsverdier fra prosess og modell som i figur 5.6.



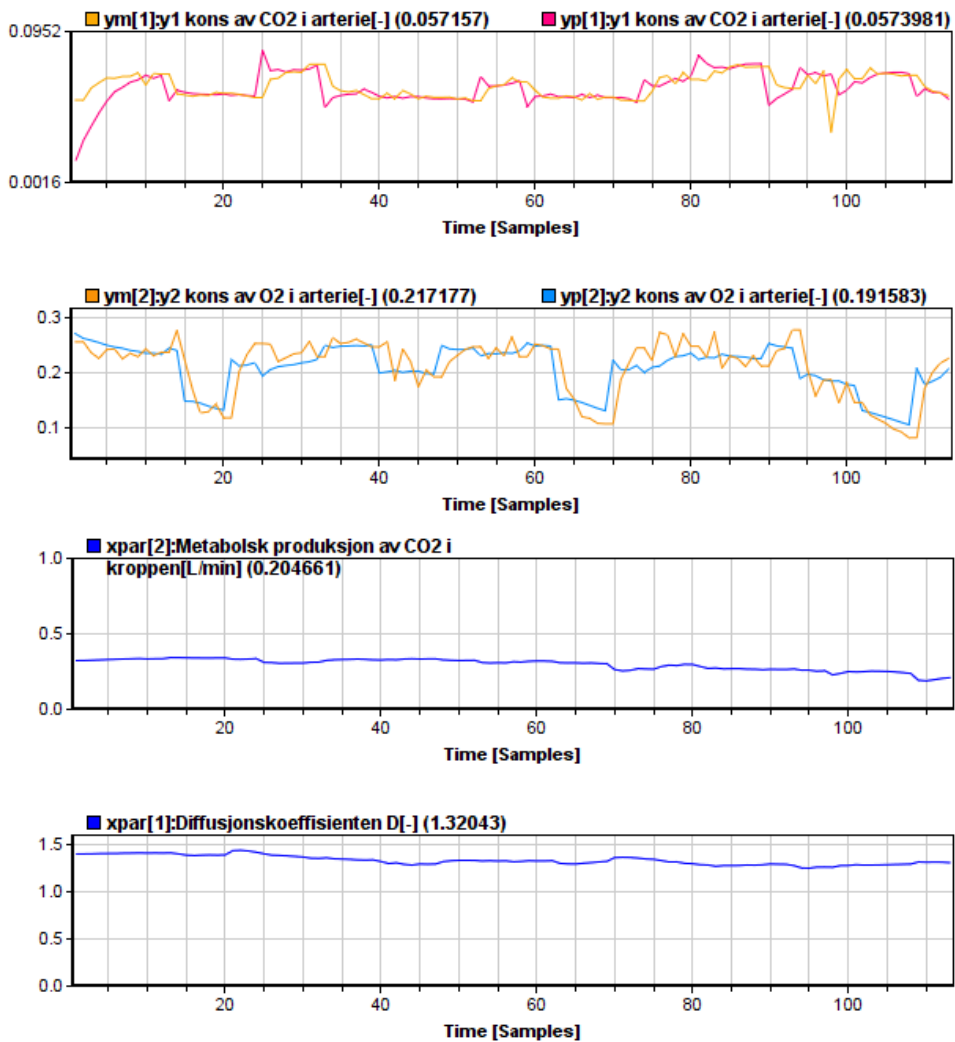
**Figur 5.3:** Figuren viser utgangsverdier fra måling og estimerte utgangsverdier fra Kalmanfilteret øverst, mens den nederste grafen viser parameterne som er estimert av Kalmanfilteret. Her er standardavviket til  $\text{MBCO}_2$  endret fra  $10^{-4}$  til 0,0004.



**Figur 5.4:** Figuren viser utgangsverdier fra måling og estimerte utgangsverdier fra Kalmanfilteret øverst, mens den nederste grafen viser parameterne som er estimert av Kalmanfilteret. Her er standardavviket til  $D$  ikke endret, dvs.  $\text{std}_v(D) = 0,0001$ . Mens standardavviket til målestøyet for  $[\text{O}_2]_a$  er satt til 0,0003.

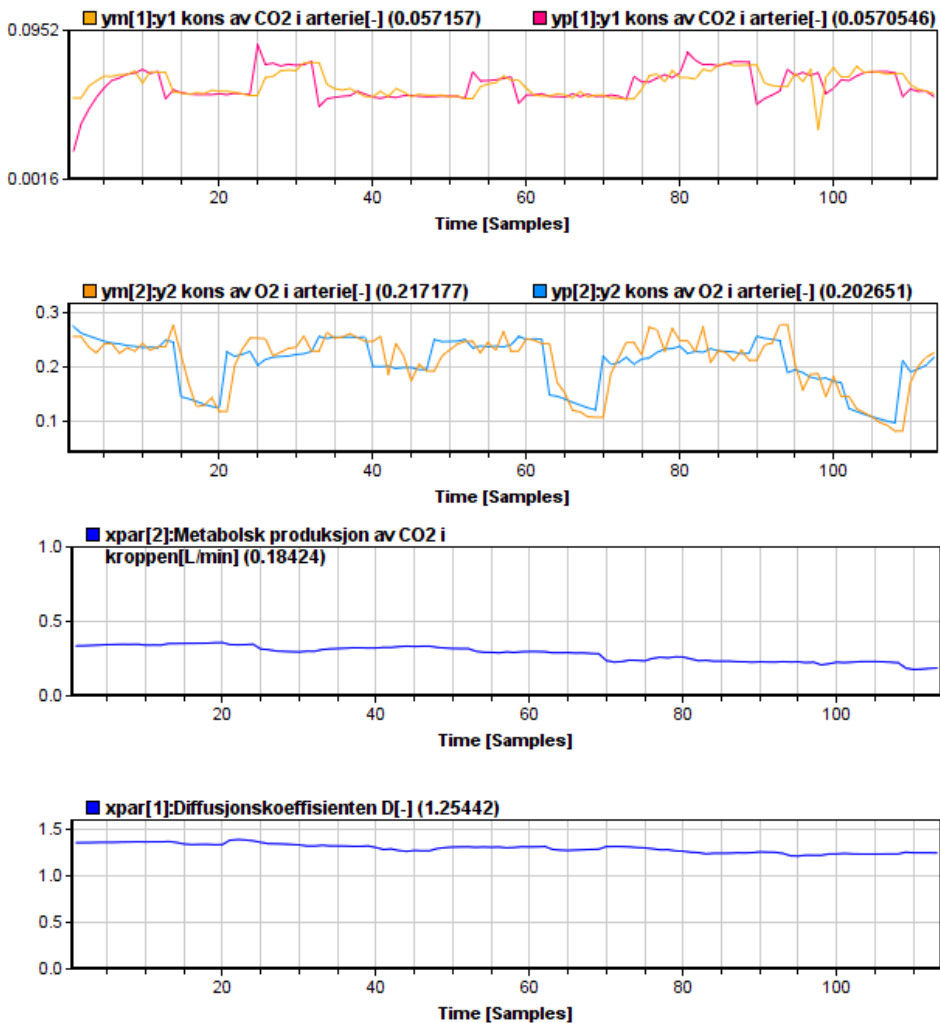


**Figur 5.5:** Figuren viser utgangsverdier fra måling og estimerte utgangsverdier fra Kalmanfilteret øverst, mens de nederste grafene viser parameterne.  $MBCO_2$  og  $D$  er estimert av Kalmanfilteret. I denne figuren er den initielle modellen, dvs. modell uten offline parameterestimering, brukt. Parameterverdier for denne modellen er gitt i tabell 4.1. Kalmanfilteret er tunet for en modell med andre parameterverdier, men figuren er med for å vise hvordan Kalmanfilteret fungerer for en modell med litt andre verdier som kan representere en prosess/gris som er litt annerledes.



**Figur 5.6:** Figuren viser utgangsverdiene til modell og prosess fra et annet griseforsøk, gitt i appendiks C.2. Offline parameterestimerte verdier fra forrige kapittel,  $MBCO_2 = 0,3231$ ,  $D = 1,408$  og  $VB = 1125,9$ , og resten som initielt gjettet, ble brukt for denne modellen også. Standardavvik fra tabell 5.1 og 5.2 ble brukt.

For å si noe om verdien av offline parameterestimering for hver individuelle prosess, ble det gjort offline parameterestimering for griseforsøk B, før online parameterestimering av prosessen ble gjort. Resultatet av dette er vist i figur 5.7, og gav parameterverdiene  $MBCO_2 = 0,336$ ,  $D = 1,362$  og  $V_B = 532,9$ . De samme standardavvikene som ble bestemt tidligere i kapittelet, gitt i tabell 5.1 og 5.2, ble brukt her også.



**Figur 5.7:** Figuren viser utgangsverdiene til prosess og utgangsverdier estimert med Kalmanfilter øverst og estimerte parameterverdier i de to nederste plotene. De optimaliserte parameterverdiene fra offline parameterestimering tilpasset forsøk B var  $MBCO_2 = 0,336$ ,  $D = 1,363$  og  $V_B = 532,9$ , og standardavvikene for online parameterestimering er gitt i tabell 5.1 og 5.2.

## 5.3 Resultat

Figur 5.3 og 5.4 viser resultatet av tuningen av Kalmanfilteret, ved plot av målt utgangsverdi sammen med estimerte utgangsverdier og estimerte parameterverdier. Valgte standardavvik er gitt i tabell 5.1 og 5.2. Verdiene for prosessstøy var  $10^{-4}$ .



Standardavvik for parameterstøy, std_v	
Diffusjonskoeffisienten, D	0,0001
Metabolsk produksjon av CO <sub>2</sub> i kroppen, MBCO <sub>2</sub>	0,0004
Metabolsk forbruk av O <sub>2</sub> i kroppen, MBO <sub>2</sub>	0,0001
Slagvolum, SV	0,0001
Volum i kroppen, V <sub>B</sub>	0,0001
Volum i alveole-arterie, V <sub>Aa</sub>	0,0001

**Tabell 5.1:** Tabellen viser resultatene av valgt standardavvik for parameterstøy. Det er bare MBCO<sub>2</sub> som er endret i forhold til default-verdiene = 10<sup>-4</sup>.

Standardavvik for målestøy, std_w	
y <sub>1</sub> = [CO <sub>2</sub> ] <sub>a</sub>	0,0001
y <sub>2</sub> = [O <sub>2</sub> ] <sub>a</sub>	0,0003

**Tabell 5.2:** Tabellen viser resultatene av valgt standardavvik for målestøy. Det er bare y<sub>2</sub> som er endret i forhold til default-verdiene = 10<sup>-4</sup>.

## 5.4 Diskusjon

Modellen med offline parameterestimerte verdier fra forrige kapittel og med valgt standardavvik for Kalmanfilteret estimeres utgangsverdiene fra prosessen svært bra til målingene fra griseforsøk A. Ved å studere figur 5.3 og 5.4 er det tydelig at estimatet av både [CO<sub>2</sub>]<sub>a</sub> og [O<sub>2</sub>]<sub>a</sub> er godt tilpasset til målingene fra prosessen.

Det største avviket mellom utgangsverdiene fra modell og prosess er for [CO<sub>2</sub>]<sub>a</sub> ved øking av F<sub>T</sub> fra 10 til 21, som gir et bratt reduksjon i [CO<sub>2</sub>]<sub>a</sub>. Dynamikken i modellen er mye raskere enn målingene. Å øke standardavviket for MBCO<sub>2</sub> vil gi en mer varierende parameterverdi. Tilpasning mellom modell og prosess og nokså konstante parameterverdier er en avveining.

Ved regulering av systemet kan dette avviket ha betydning. Filteret er tunet i åpen sløyfe. Ved regulering av systemet vil det kanskje være nødvendig å tune filteret ytterligere, men det gjenstår å vurdere ved regulering av prosessen.

### 5.4.1 Standardavvik

Lave verdier både for prosess- og målestøy gav gode estimeringer som fulgte utgangsverdiene tilfredsstillende uten å få med for mye av støyet, samtidig som parameterverdiene

varierte relativt glatt. Som beskrevet i teoridelen er det gunstig for MPC med tanke på prediksjon med parameterverdier som ikke varierer så mye.

### Lavt målestøy

Lave standardavvik for målestøy indikerer at målingene er ansett som gode. I følge biveileder Ottermo var det rimelig å anta at standardavviket for målt  $O_2$  i arteriene var høyere enn standardavviket for målt  $CO_2$  i arteriene. Grunnen til det er at luften inneholder en del  $O_2$ . Om det ikke er helt vakuu i sprøyten som prøven tas med, kunne oksygenet fra luften påvirke prøven. Fordi  $CO_2$ -nivået i luften er såpass lav, vil påvirkning av  $CO_2$  fra luft kunne neglisjeres i målingene for  $[CO_2]_a$ . Med bakgrunn i feilkilder for målingene, var det ok at målestøyet for  $[O_2]_a$  er satt høyere enn for  $[CO_2]_a$ . Begge standardavvikene er i midlertid ganske lave.

### En tuningssak

I følge artikkel om gris [1] er standardavvik for  $MBCO_2 = 0,0375^3$ , som avviker en del i fra estimert standardavvik for  $MBCO_2$  i ModelFit. Fra offline parameterestimering er parameterverdiene urealistiske verdier, så det ville vært rart om ikke også standardavvikene for parameterne skulle vært gitt verdier som tilpasset modellen til prosessen, uten å tenke på at verdiene skal reflektere den virkelige usikkerheten i parameterne.

I følge Kasper Linnestad fra Cybernetica, vil det være «rimelig med høye standardavvik dersom det er forventet at parameterne skal variere mye over tid for en gitt gris, eller dersom de skal gjelde for flere griser hvor det ikke er mulig å gjøre offline parameterestimering til hver enkelt gris på forhånd.» Han legger til at standardavviket burde være ganske lavt dersom parameterne varierer lite for en gitt gris over tid, og man har mulighet til å gjøre offline parameterestimering for hver ny gris.

Med bakgrunn i dette vil det være rimelig å anta at valgte standardavvik er gode, siden parameterne varierer lite for en gitt gris over tid og det vil være mulig å gjøre offline parameterestimering for hver ny gris. Standardavvikene kunne derfor først og fremst bli sett på som en tuningssak.

Som beskrevet i teoridelen, vil en måling med lite standardavvik gjøre at Kalmanfilteret venter målingene mye og vil med andre ord bety at det antas at målingen er veldig god. Det er ikke sikkert at det er realistisk at målingen har mindre standardavvik enn parameterne, men også her antas det som er diskutert tidligere – om tuningen gir god tilpasning er det viktigere enn realistiske verdier.

## 5.4.2 Initiell modell

Ved å endre parameterverdiene fra verdier gitt i siste kolonne i tabell 4.2 til verdier initielt gjettet, gitt i første kolonne i samme tabell, ble estimatet av utgangsverdiene endret noe.

<sup>3</sup>Verdien er gitt for en 30kg tung gris, en mellomverdi mellom de to grisene i forsøk A og B.

Tilpasningen mellom estimerte utgangsverdier og målinger var ikke like bra som med off-line parameterestimerte parameterverdier, i tillegg varierte estimatet av parameterne også mer. Dynamikken til modellen var raskere enn dynamikken til prosessen, og  $[\text{CO}_2]_a$  hadde «topper» som avvek mye fra målinger ved endring i pustefrekvens  $F_T$ .

Tilpasningen mellom modell og prosess ved online parameterestimering er likevel ganske god, som viser at modeller med parameterverdier som avviker mye fra optimale parameterverdier også gir en god estimering med valgt Kalmanfilter.

### 5.4.3 Griseforsøk B

Tilpasset modell og tunet Kalmanfilter for griseforsøk A, gav en ok estimering av utgangsverdier for griseforsøk B. Parameterverdiene varierte glatt og støy ble ikke replikert i utgangsverdiene. Dynamikken var derimot for rask for de estimerte utgangsverdiene i forhold til målingene. Det kan være flere grunner til at estimeringen ikke passet like bra som for griseforsøk A. Forsøkene ble gjort litt forskjellig, og det kan tenkes at forsøk A var bedre gjennomført enn forsøk B. I tillegg kan det hende det var mer målestøy i det ene forsøket i forhold til det andre. Parameterverdier og tuning av filteret er tilpasset forsøk A, og vil nok være hovedgrunnen til at estimeringen ikke er like bra for forsøk B som for forsøk A.

#### $\text{CO}_2$

For  $y_1 = [\text{CO}_2]_a$  følger de estimerte utgangsverdiene målingene ganske greit, men dynamikken er for rask og de estimerte verdiene har «topper» ved de fleste overganger fra høye til lave verdier eller motsatt. Ved regulering av prosessen vil det bare være de høye verdiene av  $\text{CO}_2$  som vil ha noe å si for reguleringen –  $[\text{O}_2]_{in}$  stoppes dersom  $[\text{CO}_2]_a > \text{grense}$ . Dersom «toppene» går over grensen for farlig  $[\text{CO}_2]_a$  vil det gjøre at  $[\text{O}_2]_{in}$  stoppes. Estimert av  $[\text{CO}_2]_a$ -verdien nærmer seg raskt målt verdi som vil gi en normal regulering av prosessen.  $[\text{O}_2]_{in}$  stoppes bare dersom  $[\text{CO}_2]_a > \text{grense}$  og ellers reguleres  $[\text{O}_2]_{in}$  etter mengden  $[\text{O}_2]_a$ . Så om  $[\text{O}_2]_{in}$  stoppes et lite øyeblikk, mens toppen er høy, antas det at det ikke er lenge nok til at det blir ukomfortabelt for pasienten. Leira har skrevet at alle pasienter som får  $\text{O}_2$ -tilskudd klarer seg noen sekunder uten økt tilskudd.

#### $\text{O}_2$

De estimerte utgangsverdiene til  $y_2 = [\text{O}_2]_a$  følger målingene ganske bra, men også her er dynamikken litt for rask. De estimerte verdiene for  $[\text{O}_2]_a$  får ikke med seg de laveste målte verdiene, som for eksempel ved  $\text{Time}[\text{Samples}]=70$ . Det positive er at parameteren  $D$  varierer relativt glatt og at støy i målingene ikke replikeres i de estimerte utgangsverdiene.

Ved regulering av prosessen vil forhåpentligvis ytterpunktene sjelden nås – verken veldig lav  $[\text{O}_2]_a$  eller for høy  $[\text{CO}_2]_a$  – og konsentrasjonene holde seg ganske stabile rundt referanseverdiene sine. Et nivå av  $[\text{CO}_2]_a$  der estimatet krysser farlig grense vil i alle fall ivareta sikkerheten ved at oksygenmengden i pustegassen reduseres før kritisk grense nås.

Kanskje det da ikke er så farlig om de laveste  $[O_2]_a$ -verdiene ikke estimeres og at  $[CO_2]_a$  har «topper» som kanskje bikker  $[CO_2]_a$  over grensen ved lave pustefrekvenser  $F_T$ . Da vil offline parameterestimering og tuningen av Kalmanfilteret som er gjort for det ene grise-forsøket fungere greit for alle andre griser også. Da vil det ikke være nødvendig å gjøre offline parameterestimering for hver enkelt gris.

### **Med offline parameterestimering**

Offline parameterestimering av modellen for griseforsøk B gav ikke nevneverdig forbedring ved online parameterestimering av utgangsverdiene.  $[CO_2]_a$  var som før, mens  $[O_2]_a$  var en anelse forbedret i forhold til online parameterestimering med parameterverdier fra kapittel 4. Forbedringene var så små at det ikke ville gi forbedringer i forhold til regulering av prosessen. Det at forbedringene var så små er gode resultater – det betyr at det ikke vil være nødvendig å gjøre offline parameterestimering for hver enkelt gris. Det bør imidlertid ikke brukes initielt gjettede verdier, men estimerte parameterverdier fra kapittel 4. Forbedringen av tilpasningen mellom modell og prosess ved offline parameterestimerte parameterverdier i forhold til initielt gjettede verdier var betydelig.

Det vil alltid være avvik mellom modell og prosess. Det kan virke som om modellen er en god nok representasjon av det virkelige systemet ved online parameterestimering, både tilpasset ved offline parameterestimering og ikke. Tilpasningen er best til griseforsøk A, men de estimerte utgangsverdiene avviker lite fra målingene i begge griseforsøkene som er gjort.

Det hadde nok vært mulig og gjort tilpasninger i modellen eller tunet Kalmanfilteret nøyere, for at avviket mellom estimert utgangsverdi og faktisk utgangsverdi skulle vært mindre, for alle griseforsøk. Om det er nødvendig å gjøre gjenstår å se ved modellprediktiv regulering av systemet. Fungerer simulert reguleringen bra for alle scenarier, beskrevet i 1.3, kan det bety at modell og tuning fungerer tilfredsstillende. For et nytt griseforsøk vil det gjøres både offline og online parameterestimering, som tilpasser modellen til spesifikk gris, før systemet reguleres.



**Del III**

**MPC**



# Kapittel 6

## Modellprediktiv regulering

Ved modellprediktiv regulering av oksygenkonsentrasjon hos KOLS-pasienter og modellprediktiv regulering generelt, er det flere ting som må tenkes på – restriksjoner, slakkvariabler, prediksjonshorisont og kontrollhorisont – bare for å nevne noen.

Kapitlet starter med en forholdsvis lang teoridel. I etterfølgende delkapittel designes en reguleringsstruktur i form av et blokkdiagram for problemet. Den designede MPC-strukturen implementeres i Cybernetica sitt program for (ulinear) MPC, CENIT. Tuning av de ulike parameterne og resultat fremlegges i egne delavsnitt. Metoden for kapitlet består av 6.2 og 6.3. Design, ulinear vs. linear MPC, implementasjon, tuning og resultat diskuteres i et delavsnitt til slutt.

### 6.1 Teori

Delkapitlet består av en innholdsrik teoridel som starter med en generell beskrivelse av optimering av dynamiske system, som deretter glir over i teori om MPC. Det blir gitt en innføring i MPC generelt, samt en beskrivelse av linear og ulinear MPC. Delkapitlet avsluttes med MPC i CENIT og Realsim.

#### 6.1.1 Optimering av dynamiske system

I følge Foss og Heirung [30] er et optimeringsproblem et problem der en kostfunksjon minimeres eller maksimeres, med hensyn til visse beslutningsvariabler og restriksjoner. Generelt kan et optimeringsproblem formuleres som i ligning (6.1), der  $f(\mathbf{z})$  er kostfunksjon som skal minimeres (eller maksimeres),  $\mathbf{z}$  er beslutningsvariablene og  $c_i(\mathbf{z})$  er restriksjonene. Den optimale løsningen til problemet vil dermed være den minimale (eller maksimale)



verdien av  $f(\mathbf{z})$  innenfor disse restriksjonene. Det er to typer restriksjoner; likhetsrestriksjoner,  $i \in \mathcal{E}$ , og ulikhetsrestriksjoner,  $i \in \mathcal{I}$ .

$$\begin{aligned} & \min_{\mathbf{z} \in \mathbb{R}^n} && f(\mathbf{z}) \\ & \text{slik at} && c_i(\mathbf{z}) = 0, \quad i \in \mathcal{E} \\ & && c_i(\mathbf{z}) \leq 0, \quad i \in \mathcal{I} \end{aligned} \quad (6.1)$$

På grunn av restriksjonene vil beslutningsvariablene bestemmes fra en delmengde  $\Omega$  av  $\mathbb{R}^n$ . Denne delmengden er kalt løsningsområde eller løsningsmengden, og er definert av restriksjonene  $c_i$ . Et punkt  $\mathbf{z}^* \in \Omega$  er løsningen på minimeringsproblemet dersom [30]

$$f(\mathbf{z}^*) \leq f(\mathbf{z}), \text{ for alle } \mathbf{z} \in \Omega$$

I et dynamisk, ulineært system, vil systemet beskrives av et sett med ulineære differensialligninger som avhenger av tid, tidligere tilstander  $\mathbf{x}$  og inngangsverdier  $\mathbf{u}$ . Ligning (6.2) er et eksempel på dette, der både  $\mathbf{x}$  og  $\mathbf{u}$  er beslutningsvariabler og  $\mathbf{x}_0$  er initialtilstand for tilstandsvariablene [30].

$$\dot{\mathbf{x}} = \mathbf{g}(t, \mathbf{x}, \mathbf{u}), \quad \mathbf{x}(0) = \mathbf{x}_0 \quad (6.2)$$

Det er også mulig å inkludere ulikhetsrestriksjoner,  $\phi \geq 0$ , som er vanlig dersom beslutningsvariablene representerer ikke-negative variabler, som f.eks. konsentrasjon av visse gasser i blodet. For å formulere et optimeringsproblem for et slikt system, må systemet diskretiseres og løses for et tidsintervall  $k \in [0, N]$ .  $N$  er prediksjonshorizonten, fordi systemet optimeres og predikteres over  $N$  diskrete tidsskritt inn i fremtiden. I følge Seborg et al. [31] er det vanlig å velge prediksjonshorizonten  $N$  slik at effekten av den siste inngangsverdien er dødd ut, sagt med andre ord - dynamikken til tilstandsvariablen  $\mathbf{x}$  har stabilisert seg etter en endring i inngangsverdi. Foss og Heirung [30] understreker at prediksjonshorizonten påvirker stabiliteten til løsningen, og at en lengre horisont gjerne gir bedre stabilitet. Vanligvis og for enkelthets skyld er *tiden mellom*  $k$  og  $k+1$ ,  $\Delta k$ , konstant gjennom hele prediksjonshorizonten, men kan også variere. Differensialligningene for det diskrete systemet kan beskrives av:

$$\mathbf{x}_{k+1} = \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k), \quad \mathbf{x}(0) = \mathbf{x}_0, \quad k = 0, \dots, N - 1 \quad (6.3)$$

Optimeringsproblemet for et dynamisk, ulineært system, slik som i ligningene i (6.3), vil ha samme form som det generelle optimeringsproblemet i (6.1), med både likhets- og ulikhetsrestriksjoner, og er gitt i ligning (6.4) [30].  $\Delta u$  er endringsraten til inngangsverdien, og er gitt av forskjellen mellom  $u_k$  og  $u_{k-1}$ .

$$\begin{aligned}
\min_{\phi \in \mathbb{R}^n} \quad & f(\phi) = \sum_{k=0}^{N-1} f_k(\mathbf{x}_{k+1}, \mathbf{u}_k) \\
\text{slik at} \quad & \mathbf{x}_{k+1} = \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k), \quad k = 0, \dots, N-1 \\
& \mathbf{x}_0, \mathbf{u}_{-1} \text{ er gitt} \\
& \mathbf{x}_k^{\text{nedre}} \leq \mathbf{x}_k \leq \mathbf{x}_k^{\text{øvre}}, \quad k = 1, \dots, N \\
& \mathbf{u}_k^{\text{nedre}} \leq \mathbf{u}_k \leq \mathbf{u}_k^{\text{øvre}}, \quad k = 0, \dots, N-1 \\
& \Delta \mathbf{u}^{\text{nedre}} \leq \Delta \mathbf{u}_k \leq \Delta \mathbf{u}_k^{\text{øvre}}, \quad k = 0, \dots, N-1 \\
\text{der} \quad & \mathbf{x} \in \mathbb{R}^{n_x} \\
& \mathbf{u} \in \mathbb{R}^{n_u} \\
& \Delta \mathbf{u}_k = \mathbf{u}_k - \mathbf{u}_{k-1} \\
& \phi^T = (\mathbf{x}_1^T, \dots, \mathbf{x}_N^T, \mathbf{u}_0^T, \dots, \mathbf{u}_{N-1}^T) \\
& \phi \in \mathbb{R}^n, \quad n = N \cdot (n_x + n_u)
\end{aligned} \tag{6.4}$$

For tiden  $t_0$  vil algoritmen som løser optimeringsproblemet finne et sett av inngangsverdier  $\mathbf{u}_k$ ,  $k = 0, \dots, N-1$ , som minimerer funksjonen  $f(\phi)$  over prediksjonshorisonten  $N$ , samtidig som restriksjonene overholdes. Designet av kostfunksjonen spiller en vesentlig rolle for løsningen til optimeringsproblemet. Ulineære optimeringsproblemer er ikke-konvekse, på grunn av de ulineære restriksjonene, som vanligvis vil gi flere regioner med flere lokale løsningspunkter. Feil valg av  $f(\phi)$  kan resultere i en veldig høy løsningsstid for problemet, og i verste fall et problem der optimal løsning ikke blir funnet [30]. En generell måte å representere kostfunksjonen  $f(\phi)$  i ligning (6.4) på, er gitt i ligning (6.5), der matrisene  $\mathbf{Q}$  og  $\mathbf{R}$  for hvert tidskritt er positive semidefinitte matriser,  $\mathbf{Q}_k, \mathbf{R}_k \geq 0$ .

$$f(\phi) = \sum_{k=0}^{N-1} \frac{1}{2} \mathbf{x}_{k+1}^T \mathbf{Q}_{k+1} \mathbf{x}_{k+1} + d_{x,k+1}^T \mathbf{x}_{k+1} + \frac{1}{2} \mathbf{u}_k^T \mathbf{R}_k \mathbf{u}_k + d_{u,k}^T \mathbf{u}_k \tag{6.5}$$

Designet av kostfunksjonen i ligning (6.5), gjør det enkelt å straffe eller minimere noen tilstander eller inngangsverdier mer enn andre. Hvis for eksempel uttrykket  $\frac{1}{2} \mathbf{x}_{k+1}^T \mathbf{Q}_{k+1} \mathbf{x}_{k+1}$  blir større enn andre ledd, vil optimeringsalgoritmen prøve å minke tilstandene tilstrekkelig slik at  $f(\phi)$  også minker tilstrekkelig. Designet gjør det også mulig å straffe tilstander relativt til andre tilstander og inngangsverdier relativt til andre inngangsverdier. Et eksempel kan være at en høy verdi i siste element i  $\mathbf{R}$ , vil gjøre at algoritmen vil fokusere på å minimere bruk av den siste inngangsverdien relativt til de andre inngangsverdiene, for å minimere hele kostfunksjonen.

En ulempe med formuleringen i (6.5), er at restriksjonene noen ganger gjøre det umulig å finne en løsning. Ved å utvide grensene, slik at restriksjonene blir mindre strenge, vil det være mulig å finne en løsning. Utvidelsen av grensene gjøres ved å legge til og trekke fra  $\epsilon$ , og gir en ny, utvidet formulering av optimeringsproblemet. En slik formulering er gitt i

ligning (6.6). Endringer i formuleringen fra tidligere er merket med rødt. Fordi løsningen som ligger utenfor de egentlige grensene ikke er ønskelig, settes vanligvis vekten av  $\varepsilon$  høy i kostfunksjonen.

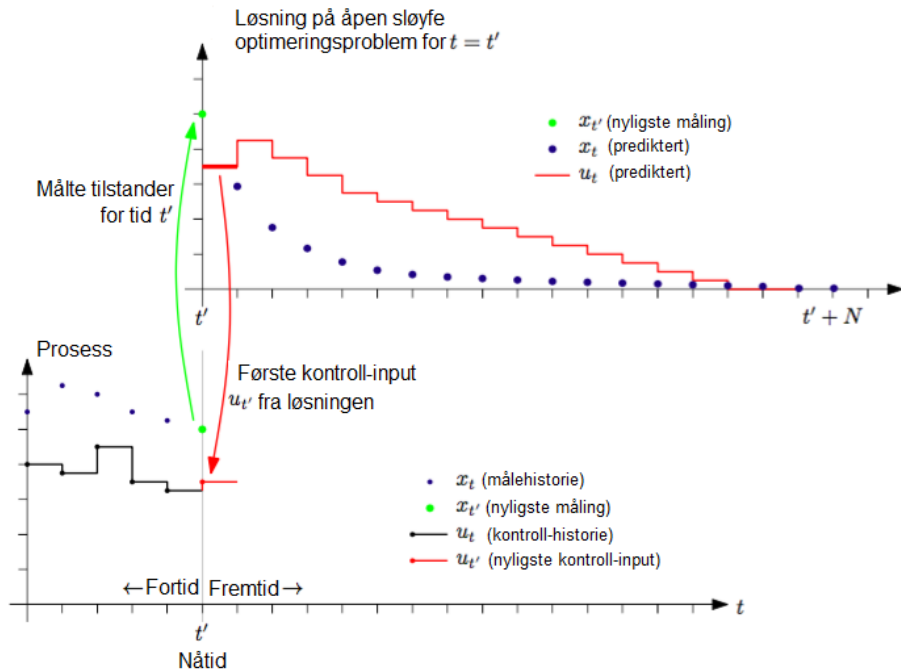
$$\begin{aligned}
 \min_{\phi \in \mathbb{R}^n} \quad & f(\phi) = \sum_{k=0}^{N-1} \frac{1}{2} \mathbf{x}_{k+1}^T Q_{k+1} \mathbf{x}_{k+1} + d_{x,k+1}^T \mathbf{x}_{k+1} + \frac{1}{2} \mathbf{u}_k^T R_k \mathbf{u}_k \\
 & \quad + d_{u,k}^T \mathbf{u}_k + \frac{1}{2} \Delta \mathbf{u}_k^T R_{\Delta,k} \Delta \mathbf{u}_k + \rho^T \varepsilon + \frac{1}{2} \varepsilon^T S \varepsilon \\
 \text{slik at} \quad & \mathbf{x}_{k+1} = \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k), \quad k = 0, \dots, N-1 \\
 & \mathbf{x}_0, \mathbf{u}_{-1} \text{ er gitt} \\
 & \mathbf{x}^{\text{nedre}} - \varepsilon \leq \mathbf{x}_k \leq \mathbf{x}^{\text{øvre}} + \varepsilon, \quad k = 1, \dots, N \\
 & \mathbf{u}^{\text{nedre}} \leq \mathbf{u}_k \leq \mathbf{u}^{\text{øvre}}, \quad k = 0, \dots, N-1 \\
 & \Delta \mathbf{u}^{\text{nedre}} \leq \Delta \mathbf{u}_k \leq \Delta \mathbf{u}^{\text{øvre}}, \quad k = 0, \dots, N-1 \\
 \text{der} \\
 & \varepsilon \in \mathbb{R}^{n_x} \leq 0 \\
 & \rho \in \mathbb{R}^{n_x} \leq 0 \\
 & \mathbf{S} \in \text{diag}\{s_1, \dots, s_{n_x}\}, \quad s_i \leq 0, \quad i = \{1, \dots, n_x\} \\
 & \mathbf{x} \in \mathbb{R}^{n_x} \\
 & \mathbf{u} \in \mathbb{R}^{n_u} \\
 & \Delta \mathbf{u}_k = \mathbf{u}_k - \mathbf{u}_{k-1} \\
 & \phi^T = (\mathbf{x}_1^T, \dots, \mathbf{x}_N^T, \mathbf{u}_0^T, \dots, \mathbf{u}_{N-1}^T) \\
 & \phi \in \mathbb{R}^n, \quad n = N \cdot (n_x + n_u)
 \end{aligned} \tag{6.6}$$

### 6.1.2 Modellprediktiv regulering (MPC)

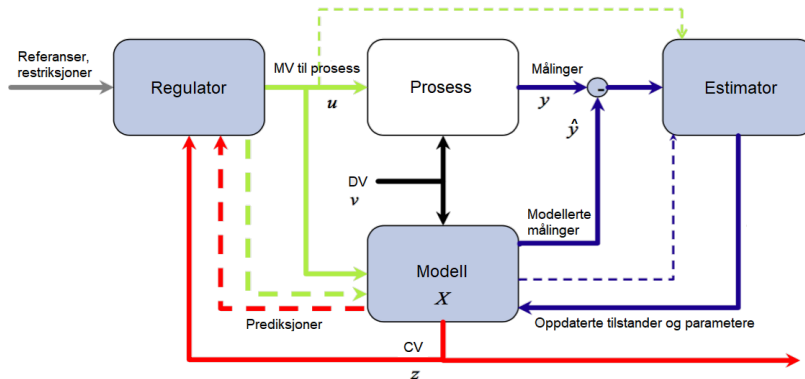
Det generelle optimeringsproblemet, som er beskrevet tidligere i kapittelet, er i følge Foss og Heirung [30] et åpent sløyfe optimeringsproblem. Dette fordi den optimale inngangsverdien  $\mathbf{u}_k$  blir utregnet ved tiden  $t_0$  for hele prediksjonshorizonten, uten tilbakekobling fra tilstander under simuleringen. Foss og Heirung [30] forklarer videre at modellprediktiv regulering (MPC) er et lukket sløyfe optimeringsproblem, der tilbakekobling utnyttes. En illustrasjon av prinsippet for MPC er vist i figur 6.1.

Mayne et al. [32] beskriver modellprediktiv regulering som en regulator der kommende inngangsverdi finnes ved å, for hvert tidssteg, løse et optimaliseringsproblem i åpen sløyfe med endelig horisont, ved å bruke nåværende tilstander til systemet som initielle tilstander. Optimaliseringen gir en optimal sekvens av inngangsverdier, der den første inngangsverdien i denne sekvensen gir en ny inngangsverdi til systemet.

For å beskrive MPC ytterligere er pseudo-algoritmen fra Foss og Heirung [30] gjengitt



**Figur 6.1:** Illustrasjon av prinsippet for MPC, hvor prediksjonshorizonten er  $N$ . MPC regner ut optimal inngangsverdi  $u$  for at faktisk tilstand skal nærme seg ønsket tilstand. MPC regner ut optimal sekvens av inngangsverdier i forhold til målingen som er gjort i nåtid, for at fremtidige tilstander skal oppnå ønsket verdier. I figuren kan en se hvordan  $z_k$  og  $u_k$ , varierte i fortid og hvordan prediksjonen av variablene fra nåtid og til  $N$  tidsskritt frem i tid ser ut. Prediksjonen viser hvordan sekvensen av  $z$ -verdier endres gitt at den optimale sekvensen av  $u$  anvendes. Det vil bare være den første optimale inngangsverdien  $u_k$  som velges, før en ny løsning regnes ut for neste tidsskritt. Figur er hentet fra [30], s.46.



**Figur 6.2:** Figuren viser en regulator, MPC, som passer på at prosessen følger en referanse. Figuren er hentet fra [18] og oversatt til norsk.

nedenfor. Denne algoritmen antar at den faktiske tilstanden er tilgjengelig for tilbakekobling.

Algoritme: (N)MPC prosedyre

**for** k = 0,1,2,... **do**  
 Finn nåværende tilstand  $x_k$ ;  
 Løs det dynamiske optimeringsproblemet for prediksjonshorizonten fra k til k+N med  $x_k$  som initialtilstand;  
 Bruk den første inngangsverdien  $u_k$  fra løsningen over;  
**end for**

Dersom tilstandene til systemet ikke er tilgjengelige for tilbakekobling, vil det være nødvendig med en estimator som estimerer tilstandene som ikke kan måles direkte. Den mest vanlige estimatoren er Kalmanfilteret, som ble beskrevet i forrige kapittel.

I reguleringssammenheng er det ofte ønskelig at systemet følger en referanse. I disse tilfellene vil målingene fra systemet tilbakekobles og sammenlignes med ønsket utgangssignal. Forskjellen mellom signalene sendes inn til «MPC-blokken» som regner ut optimale inngangsverdier. En MPC som følger en referanse er det som blir brukt videre i denne oppgaven, og et slikt optimeringsproblem vil generelt ha følgende kostfunksjon:

$$f(\phi) = \sum_{k=0}^{N-1} \frac{1}{2} (\mathbf{y}_k - \mathbf{y}^{ref})^T \mathbf{Q} (\mathbf{y}_k - \mathbf{y}^{ref}) + \frac{1}{2} (\Delta \mathbf{u}_k^T \mathbf{R} \Delta \mathbf{u}_k)$$

der  $\mathbf{y}_k$  kan være målt eller estimert. Estimerte parametere vil ofte være definert med en hatt over, slik at estimert utgangsverdi er gitt som  $\hat{\mathbf{y}}_k$ . En grafisk fremstilling i form av et blokkdiagram er vist i figur 6.2.

### 6.1.3 Lineær MPC

I lineær MPC er det kvadratiske kostfunksjoner og lineære restriksjoner, som betyr at et konvekst QP problem løses for hvert tidsskritt. En generell formulering av lineær optimeringsproblem er oppnådd ved å erstatte  $\mathbf{g}(\mathbf{x}_k, \mathbf{u}_k)$  med en lineær funksjon  $\mathbf{A}_k \mathbf{x}_k + \mathbf{B}_k \mathbf{u}_k$ .

Lineære modeller gir gode resultater når prosessen opererer i områder nær lineariseringspunktene. Der målet er å holde prosessen rundt en stasjonær tilstand vil derfor ofte en lineær modell være nok. En fordel med lineær MPC i forhold til ulineær, er at utregninger for lineær er mye raskere. Dessuten vil en lineær modell sammen med en kvadratisk kostfunksjon gi et konvekst problem. Konvekse problemer har løsninger som er nøye studert og med mange kommersielle produkter ledige [33]. Det kan også legges til at for et konvekst problem vil et lokalt optimum også være et globalt optimum [30].

### 6.1.4 Ulineær MPC

Camacho og Bordons [33] beskriver at noen prosesser kanskje aldri vil være i steady-state eller ha mye forstyrrelser som gjør at systemet ofte er utenfor «operating region». For slike prosesser vil ikke en lineær regulator være veldig effektiv, så ulineære regulatorer vil være essensielle for å forbedre ytelse eller stabilitet.

Uttrykket ulineær modellprediktiv regulering (NMPC) er vanligvis reservert for applikasjoner der prosessmodellen er ulineær, ikke optimeringsproblemet [34]. Det betyr at i ulineær MPC vil ulineære likhetsrestriksjoner innføres, fordi  $\mathbf{g}(\mathbf{x}_k, \mathbf{u}_k)$  i ligning (6.6) er ulineær [30]. Videre beskriver Foss og Heirung [30] at dette gjør at problemet ikke lenger er konvekst, og vil gjøre problemet mye mer komplisert. Algoritmene som løser problemet krever mye mer kjøretid og løsningstiden er mindre robust i den forstand at en løsning, hvis den finnes, kanskje ikke blir funnet. Det er også få måter å bestemme kvaliteten på løsningen.

En av fordelene med MPC med ulineær modell i forhold til en linearisert modell, er at med den ulineære modellen vil det bli gitt en mer presis prediksjon, uavhengig av driftsforhold. Med en bedre prediksjon vil reguleringen av systemet også bli bedre. Det er derimot ikke sikkert at den estimerte parameteren vil gi en tilstrekkelig nøyaktig prediksjon for alle driftsforhold. For å rette opp i dette kan en rekursiv estimeringsteknikk innføres, f.eks. et Kalmanfilter som beskrevet i kapittel 5. På en annen side vil bruk av Kalmanfilter komplisere tuningen av MPC siden Kalmanfilteret selv må tunes. En tommelfingerregel er å gjøre estimatorens dynamikk betydelig raskere enn den lineære MPC tilbakekoblingssløyfen for å begrense interaksjon mellom estimatoren og reguleringssløyfen [30].

Foss og Heirung [30] beskriver at en viktig grunn til den industrielle suksessen til MPC er forbedret ytelse gjennom økt undertrykking av forstyrrelser, samt bedre tracking. MPC hjelper dermed med å overholde strammere ytelsespesifikasjoner og strammere restriksjoner fra miljø- og sikkerhetshensyn, som dermed forbedrer økonomi.

### 6.1.5 MPC i CENIT og RealSim

For CENIT og RealSim er det nødvendig at modellen som brukes er definert slik at programmene vet hva som er hva. Modellen defineres med MV, DV og CV, som forklart i bakgrunnsdelen i kapittel 3.

Optimeringsproblemet i CENIT og RealSim vil være definert som følger:

$$\min \quad J = \frac{1}{2}(\mathbf{Z} - \mathbf{Z}_{ref})^T \mathbf{Q} (\mathbf{Z} - \mathbf{Z}_{ref}) + \frac{1}{2}(\Delta \mathbf{U}^T \mathbf{S} \Delta \mathbf{U}) + \mathbf{r}_1^T \boldsymbol{\varepsilon} + \frac{1}{2} \boldsymbol{\varepsilon}^T \text{diag}(\mathbf{r}_2) \boldsymbol{\varepsilon}$$

$$\begin{aligned} \text{slik at} \quad & \mathbf{x}_{k+j} = \mathbf{f}(\mathbf{x}_{k+j-1}, \mathbf{u}_{k+j-1}, \mathbf{v}_k) \\ & \mathbf{z}_{k+j} = \mathbf{h}(\mathbf{x}_{k+j}, \mathbf{u}_{k+j}) \\ & \mathbf{Z}_{min} - \boldsymbol{\varepsilon} \leq \mathbf{Z} \leq \mathbf{Z}_{maks} + \boldsymbol{\varepsilon} \\ & 0 \leq \boldsymbol{\varepsilon} \leq \boldsymbol{\varepsilon}_{maks} \\ & \mathbf{U}_{min} \leq \mathbf{U} \leq \mathbf{U}_{maks} \\ & \Delta \mathbf{U}_{min} \leq \Delta \mathbf{U} \leq \Delta \mathbf{U}_{maks} \end{aligned} \quad (6.7)$$

der

$$\mathbf{Z} = \begin{bmatrix} z_{k+1} \\ z_{k+2} \\ \vdots \\ z_{k+P} \end{bmatrix}, \quad \mathbf{U} = \begin{bmatrix} u_k \\ u_{k+1} \\ \vdots \\ u_{k+M-1} \end{bmatrix}, \quad \Delta \mathbf{U} = \begin{bmatrix} \Delta u_k \\ \Delta u_{k+1} \\ \vdots \\ \Delta u_{k+M-1} \end{bmatrix} = \begin{bmatrix} u_k - u_{k-1} \\ u_{k+1} - u_k \\ \vdots \\ u_{k+M-1} - u_{k+M-2} \end{bmatrix}$$

$\mathbf{Z}$  er utgangsverdier som tilbakekobles,  $\mathbf{U}$  er inngangsverdier og  $\Delta \mathbf{U}$  er endring fra forrige inngangsverdi til nåværende.  $P$  er prediksjonshorizonten,  $k$  er tidskritt og  $M$  er kontrollhorizonten.

Ved tuning av MPC i CENIT er det variablene i  $\mathbf{Q}$ ,  $\mathbf{S}$ ,  $\mathbf{r}_1$ ,  $\mathbf{r}_2$ ,  $\mathbf{Z}_{min}$ ,  $\mathbf{Z}_{maks}$ ,  $\boldsymbol{\varepsilon}_{maks}$ ,  $\mathbf{U}_{min}$ ,  $\mathbf{U}_{maks}$ ,  $\Delta \mathbf{U}_{min}$  og  $\Delta \mathbf{U}_{maks}$ , alle merket med grønne bokstaver i formuleringen over, som må tilpasses for å få ønsket oppførsel på regulatoren. Maks- og minverdier for  $\mathbf{U}$  og  $\mathbf{Z}$  er enkle å bestemme ut fra kunnskap om prosessen og verdiene er vanligvis de samme for hele prediksjonshorizonten.  $\boldsymbol{\varepsilon}_{maks}$  er vanligvis uendelig for å unngå tilstander der problemet ikke har noen løsning.  $\mathbf{r}$  er straff for å krysse grensene til CV, der  $\mathbf{r}_1$  brukes til å prioritere mellom CV-grenser.  $\Delta \mathbf{U}$  settes for å garantere at maksimalt akseptabel endring i inngangsverdi ikke brytes av regulatoren og er de eneste harde restriksjonene i optimeringsproblemet. [18]

Når det kommer til  $\mathbf{Q}$  og  $\mathbf{S}$  er disse alltid diagonale matriser og har vanligvis samme verdi for hele prediksjonshorizonten. Elementene i  $\mathbf{Q}$  straffer avvik fra ønsket verdi på CV, mens elementene i  $\mathbf{S}$  straffer endring i MV-verdier. I Cybernetica sine programmer er vekten av  $\mathbf{Q}$  og  $\mathbf{S}$  gitt av en vekt dividert med kvadratet av spennet til variabelen som skal straffes.

Det gjør at alle variablene har samme størrelsesorden, slik at matrisene  $\mathbf{Q}$  og  $\mathbf{S}$  bedre reflekterer den individuelle vekten av hver variabel. [18]

For hver variabel  $z$  og  $u$  er  $\mathbf{Q}$  og  $\mathbf{S}$  gitt av ligning (6.8) [18].

$$Q = \frac{vekt}{(z\text{-spenn})^2}, \quad S = \frac{vekt}{(u\text{-spenn})^2} \quad (6.8)$$

Prediksjonshorizonten er antall tidsskritt MPCen predikterer inn i fremtiden og er i programmene til Cybernetica definert ut fra den høyeste verdiene som ligger i parametriseringen av CVene ( $z$ -variablene). Jo lenger prediksjonshorizont, jo flere variabler og lenger kjøretid.

Det finnes to metoder for å eliminere variabler for å oppnå et enklere problem med færre frihetsgrader, nemlig parametrisering av MV (eng: *MV blocking*) og kontrollvariabel evalueringpunkter (eng: *CV evaluation points*). Parametrisering av MV er i følge Foss og Heirung [30] en måte å strukturere inngangsvariablene til en annen struktur, slik at antallet optimale, predikterte inngangsverdier blir færre enn i det originale problemet. For det originale problemet vil antallet beslutningsvariabler være bestemt av  $N \cdot n_u$ . Det betyr at det for hvert tidsskritt vil være en variabel for hver  $u$ . Med parametrisering av MV er det mulig å spesifisere antallet tidsskritt som er ønskelig før neste endring i inngangsverdi, og holde inngangsverdien konstant mellom de to tidspunktene. På den måten vil antallet inngangsverdier reduseres og dermed også føre til færre frihetsgrader.

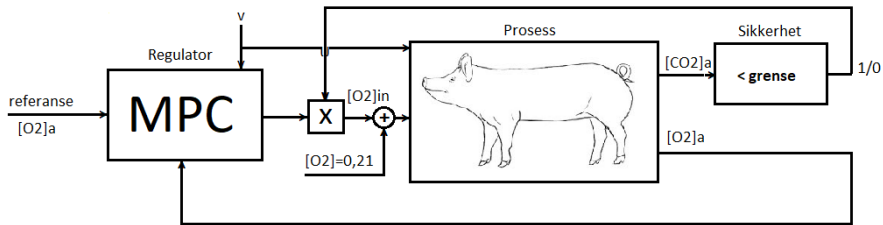
Kontrollvariabel evalueringpunkter er også en teknikk som blir brukt for å redusere antall frihetsgrader til et optimeringsproblem ved å redusere antall restriksjoner [35]. I stede for at objektivfunksjonen evalueres for hvert tidsskritt  $k$ , vil et mindre antall av tidspunktene mellom nåtid og  $N$  velges slik at objektivfunksjonen bare evalueres for disse punktene. Hvor mange punkter som er nødvendig og hvor de skal plasseres er avhengig av systemet og den ønskede oppførselen til regulatoren. I tillegg kan ulike CV ha ulikt antall evalueringpunkter, og de kan også være spesifisert uavhengig av hverandre. Strand og Sageli [35] sier at 5 til 20 evalueringpunkter er vanlig for standard MPC algoritmer. For store ulineære systemer vil parametrisering av MV og kontrollvariabel evalueringpunkter være viktige teknikker for å redusere kjøretiden. Med disse teknikkene vil beslutningsvariablene i  $\mathbf{Z}$ ,  $\mathbf{U}$  og  $\Delta\mathbf{U}$  reduseres i optimeringsproblemet over. Resten av formuleringen vil forbli lik.

## 6.2 Design av reguleringsstruktur

MPCens evne til å optimere og prediktere fremtidige hendelser, gjør den til et godt redskap ved regulering av oksygenmengden. I dette delkapittelet presenteres et mulig design av en reguleringsstruktur.

Målet med oppgaven, som beskrevet i innledende kapittel, var å designe en reguleringsstruktur som regulerer mengden  $O_2$  i pustegassen slik at både komfort og sikkerhet blir





**Figur 6.3:** Figuren viser et design av en reguleringsstruktur for regulering av oksygenkonsentrasjon hos gris. «Regulator»-blokken inneholder en MPC, prosessen kan være en modell av gris eller en levende gris og blokken med «X» er en multiplikator-blokk. Sikkerhetsblokken er en binær, logisk blokk som gir 0 dersom  $[CO_2]_a$  er over en gitt grense og 1 ellers. Sikkerhetsblokken gjør at grisen ikke får noe ekstra oksygen i pustegassen så lenge  $[CO_2]_a > grense$ . Bilde av grisen er hentet fra [36]

ivaretatt. For å opprettholde komfort var det viktig at  $[O_2]_a$  ikke blir for lav og for å opprettholde sikkerhet var det viktig at  $[CO_2]_a$  ikke ble for høy.

Grisen fra forsøk A ble, etter å ha blitt lagt i narkose, stabilisert før forsøket startet. Grisen var stabil med  $[O_2]_{in} = 0,4$  og  $F_T = 21$ . Verdiene der grisen var stabil,  $[CO_2]_a = 0,058 \approx 0,6$  og  $[O_2]_a = 0,238 \approx 0,24$ , ble videre i oppgaven regnet som referanseverdier for grisen. Disse verdiene er annerledes enn referanseverdiene for en frisk, våken gris [1].

I figur 6.3 er det designet en reguleringsstruktur med MPC for regulering av oksygenkonsentrasjon ved måling av  $[CO_2]$  og  $[O_2]$  i arteriene. Utgangsverdien fra «Regulator»-blokken er mengden ekstra oksygen i pustegassen til grisen. Dersom ekstra oksygenmengde i pustegassen er 0, vil pasienten bare puste inn oksygen fra luften – som er 21%. Forstyrrelsene  $v$  er pustefrekvens  $F_T$  og puls  $F_{puls}$ , som begge varierer med blant annet fysisk anstrengelse og stress. Pustefrekvensen vil også være avhengig av blant annet  $[O_2]$  i blodet<sup>1</sup>, men er i griseforsøket styrt manuelt.

«Sikkerhet»-blokken i figur 6.3 er en blokk som ivaretar sikkerheten til grisen. For enkelthetsskyld ble den designet som en binær, logisk enhet - nevnt av Stavdahl på veiledningsmøte, men også brukt i tidligere arbeid [2]. Den logiske enheten tar verdien 1 dersom  $[CO_2]_a < grense$  og verdien 0 dersom  $[CO_2]_a > grense$ . Den binære verdien multipliseres med optimal verdi for  $[O_2]_{in}$ , beregnet av «Regulator»-blokken. På den måten vil ekstra mengde oksygen i pustegassen være 0 dersom nivået av  $CO_2$  i blodet er for høyt og ellers reguleres på vanlig vis.

«MPC»-blokken i figur 6.3 inneholder regulatoren og Kalmanfilteret. Ligningene for regulatoren er gitt i ligning (6.7), mens Kalmanfilteret er forklart i kapittel 4.1.

<sup>1</sup>Hos friske personer spiller  $[CO_2]$  i blodet en stor rolle for pustefrekvensen.

## 6.3 Modellprediktiv regulering i CENIT

Ved estimering av modellprediktiv regulering av prosessen i CENIT og RealSim, måtte prosessen også være representert av en modell. Modellen for prosess og estimering kunne ikke være den samme, da det ville gitt perfekt prediksjon og en urealistisk god regulering. Det var derfor nødvendig å lage en ny modell. Den enkleste måten å lage en annenledes modell på, var å endre på parameterverdiene til den allerede eksisterende modellen til Ottermo med optimerte parameterverdier fra kapittel 4.

Etter forslag fra veileder Stavdahl, ble en random-operator fra MATLAB brukt, for å plukke ut tilfeldige verdier for hver av parameterne. Ideen var å velge en tilfeldig parameterverdi fra et gaussisk-fordelt tall med forventningsverdi lik parameterverdiene fra offline parameterestimering i kapittel 4. Standardavviket til de ulike parameterne ble valgt med bakgrunn i [1] for en 30kg<sup>2</sup> tung gris. Forventningsverdi, standardavvik og nye parameterverdier er gitt i tabell 6.1. Kode for utregning av nye parameterverdier er gitt i appendiks A.1.

Parameterverdier til ny modell			
Parameter	Forventningsverdi	Standardavvik	Ny parameterverdi
D	1,4083	0,067	1,257
MBCO <sub>2</sub>	0,3231	0,0375	0,3433
MBO <sub>2</sub>	0,238	0,0381	0,33079
SV	0,05	0,0077	0,0567
V <sub>B</sub>	1126	100*	1157,9
V <sub>Aa</sub>	4	1*	2,6923

**Tabell 6.1:** Tabellen viser nye parameterverdier generert fra gaussisk fordeling med gitt forventningsverdi og standardavvik. Forventningsverdien er de offline parameterestimerte verdiene fra kapittel 4, mens standardavviket er verdier fra [1]. Verdiene med \* er tippede verdier, og ble tippet siden verdiene ikke er kjent og forventningsverdiene avviker mye fra realistiske verdier.

### 6.3.1 Sikkerhet

Det viktigste med reguleringen av oksygenkonsentrasjon hos KOLS-pasienter er sikkerhetsaspektet - å tvinge pasienten til å puste ut overflødig CO<sub>2</sub> fra kroppen dersom verdien av [CO<sub>2</sub>]<sub>a</sub> blir for høy. For å tvinge u<sub>1</sub>=[O<sub>2</sub>]<sub>in</sub> til å ta verdien 0,21 dersom z<sub>1</sub>=[CO<sub>2</sub>]<sub>a</sub> > grense, var en mulig løsning å legge til en binær variabel, δ, og tre andre restriksjoner til restriksjonene i (6.7):

<sup>2</sup>En mellomverdi mellom de to grisene som er brukt i forsøkene, som det kan antas at vil være ca. vekt for en ny gris

$$1. \quad \delta \geq u_1 - 0,21 \quad (6.9)$$

$$2. \quad z_1 - \text{grense} \leq M \cdot (1 - \delta) \quad (6.10)$$

$$3. \quad z_1 - \text{grense} \geq -M \cdot \delta \quad (6.11)$$

$$4. \quad \delta \in \{0, 1\} \quad (6.12)$$

M er her et *stort nok* tall og  $z_1 = [\text{CO}_2]_a$ .  $\delta$  kan tenkes på som «er  $[\text{CO}_2]_a < \text{grense}$ », og dermed gi 1 for ja og 0 for nei. Restriksjon 2. tvinger  $\delta$  til å ta verdi 0 for  $z_1$ -verdier  $>$  grense, mens restriksjon 3. tvinger  $\delta$  til å ta verdi 1 for  $z_1$ -verdier  $<$  grense. Fordi maks- og min-restriksjonene til  $u$ -ene er harde, vil ikke  $u_1$  kunne velges til en mindre verdi enn  $u_{1,\text{min}} = 0,21$  eller større enn  $u_{1,\text{maks}} = 0,96$ . For  $\delta = 1$ , som vil si at  $z_1$ -verdier  $<$  grense, så kan  $u_1$  ta alle verdier, mens  $\delta = 0$  vil tvinge  $u_1$  til 0,21.

Etter elegant hint fra Linnestad fra Cybernetica om hvordan det var vanlig å legge inn ekstra begrensninger i Cybernetica sine programmer, ble den endelige løsningen å gi en kostnad på å ha  $u_1 = [\text{O}_2]_{\text{in}}$  ulik 0,21 dersom  $[\text{CO}_2]_a >$  grense. For å få kostnaden inn i kostfunksjonen, var det nødvendig å definere en ny  $z$ -variabel,  $z_3$ , definert som følger:

```

1      if ( z1 > grense ) {
2          z3 = u1 - 0.21;
3      }
4      else {
5          z3 = 0.0;
6      }

```

Dette gjorde at  $z_3$  kunne være større enn 0 dersom  $z_1 >$  grense og  $u_1 >$  0,21. Det var ønskelig at  $z_3$  alltid var null, og derfor ble  $z_{3,\text{maks}}$  satt til 0.

### 6.3.2 Restriksjoner

For optimeringsproblemet var det ønskelig at de målte verdiene  $\mathbf{Z}$ , inngangsverdiene  $\mathbf{U}$  og endringen av inngangsverdier  $\Delta\mathbf{U}$  holdt seg innenfor grenser gitt av systemet og i henhold til modelligningene, som beskrevet i (6.7).

Modelligningene var det ikke mulig å gjøre noe med – de var gitt av modelligningene til Ottermo, beskrevet i kapittel 2. Minimum- og maksimumverdier for  $\mathbf{Z}$ -,  $\mathbf{U}$ - og  $\Delta\mathbf{U}$ -variabelene var gitt av fysiske begrensninger til systemet og tilpasning av systemet til forsøksgris A. Verdiene er gitt i tabellene 6.2 og 6.3, og forklaring til valgene følger i påfølgende avsnitt.

Grensene til  $z_1 = [\text{CO}_2]_a$  ble satt til 0 og 0,08. Disse verdiene ble valgt med bakgrunn i griseforsøk A, gitt i appendiks C.1. Fordi det ikke er farlig om nivået av  $\text{CO}_2$  i blodet blir lavere enn referanseverdien, ble nedre grense satt til 0. På den måten ville lave verdier for  $[\text{CO}_2]_a$  aldri bli straffet. Øvre grense ble satt til 0,08 fordi det var nivået av  $[\text{CO}_2]_a$  hos

grisen i forsøket ved pustefrekvens  $F_T = 10$ . I samråd med biveileder Ottermo ble  $F_T = 10$  valgt fordi det førte til et nivå av  $\text{CO}_2$  som begynte å bli farlig høyt.

For  $z_2 = [\text{O}_2]_a$  ble nedre og øvre grense satt til henholdsvis 0,12 og 1. Den nedre grensen ble valgt som verdien av  $[\text{O}_2]_a$  der det ikke var tilført ekstra oksygen i pustegassen, som vil si  $[\text{O}_2]_{in} = 0,21$ , men der pustefrekvensen var holdt normal,  $F_T = 21$ . Fordi grisen var stabil med  $[\text{O}_2]_{in} = 0,4$ , ble det antatt at ingen ekstra oksygen i pustegassen ville gå ut over komforten til grisen. Den øvre grensen ble valgt som 1 fordi det i seg selv ikke er farlig med et høyt nivå av  $\text{O}_2$  i blodet.

Etter diskusjon med biveileder Ottermo ble øvre og nedre grense for  $\Delta U$  bestemt. Fordi det måtte være mulig å gå fra  $F_T = 21$  til 10 på kort tid ble  $\Delta u_2$  satt til 15. Pulsen ble ikke endret på i simuleringen, men  $\Delta u_3$  ble satt til 150 for at pulsen også kunne tatt store sprang i løpet av et tidsskritt.  $\Delta u_1$  ble satt til 1 fordi det ikke var ønskelig å straffe endringer i  $[\text{O}_2]_{in}$ .  $\Delta = 1$  for  $[\text{O}_2]_{in}$  vil si at  $[\text{O}_2]_{in}$  kan gå til alle verdier i løpet av 1 tidsskritt.

De fysiske begrensningene til oksygenkonsentratoren fra griseforsøket gav øvre og nedre begrensninger for  $u_1 = [\text{O}_2]_{in}$ . Grensene for forstyrrelsene ble satt som fysiske, men romslige, grenser for at optimeringsproblemet ikke skulle være begrenset av disse grenseverdiene.

### 6.3.3 Tuning av regulatoren

Ved tuning av regulatoren var det viktig å tenke på at sikkerhet og komfort skulle være bevart. Tuningsparameterne, som er markert med grønn i ligning (6.7), var parameterne som kunne endres for å tune regulatoren. Maks- og minverdier for  $\mathbf{Z}$ ,  $\mathbf{U}$  og  $\Delta \mathbf{U}$  var bestemt av prosessen og implementør, gitt i tabell 6.2 og 6.3 og beskrevet i 6.3.2.  $\epsilon$  ble valgt til uendelig, for å unngå uløselige optimeringsproblem. Det gjestod å endre på  $\mathbf{Q}$ ,  $\mathbf{S}$  og  $\mathbf{r}_1$  og  $\mathbf{r}_2$  for å tune optimeringsproblemet.

$\mathbf{Q}$  ble tunet for å ivareta komforten til grisen. For å ivareta komforten var det viktig at avvik fra referanseverdien til  $[\text{O}_2]_a$  ble straffet. Elementet i  $\mathbf{Q}$  tilhørende avviket mellom referanse og målt verdi for  $[\text{O}_2]_a$  ble satt til 10. Avvik fra referanseverdien til  $[\text{CO}_2]_a$  ble ikke ansett som viktig og derfor satt til 0,5. Sikkerheten var ivaretatt av andre funksjoner, så om  $[\text{CO}_2]_a$  avvek noe fra referanseverdien var ikke det så farlig.  $\mathbf{Q}$ -verdien som straffet avvik mellom referanse- og faktisk verdi til  $z_3$ ,  $Q_{z_3}$ , ble satt til 10 for å gi kostnad på å ikke velge  $[\text{O}_2]_{in} = 0,21$  ved for høye verdier av  $[\text{CO}_2]_a$ .

For  $\mathbf{S}$  ble alle vektene valgt til 1, fordi det ble ansett som uviktig å straffe endring av inngangsverdier. Av sikkerhetsmessige årsaker var det nødvendig å kunne endre  $[\text{O}_2]_{in}$  fra en høy verdi til 0,21 på kort tid. Det var heller ikke nødvendig å straffe  $DV$ , som uansett vil variere. Verdier av  $\mathbf{Q}$  og  $\mathbf{S}$  slik de er definert i CENIT er gitt i appendiks E.

$$\mathbf{Q} = \begin{bmatrix} 0,5 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 10 \end{bmatrix}, \quad \mathbf{S} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (6.13)$$

Fordi høyt nivå av  $[\text{CO}_2]_a$  potensielt kan være dødelig var det viktig at sikkerheten var ivaretatt. Det ville være dumt å ikke utnytte egenskapene til MPC for å holde  $[\text{CO}_2]_a$  på et forsvarlig nivå. For å kunne ivareta sikkerheten hos pasientene var det viktig at kostnaden for å nærme seg høyt  $[\text{CO}_2]_a$  ble satt høyt. Når  $[\text{CO}_2]_a > \text{grense}$  er det ønskelig at  $[\text{O}_2]_{in} = 0,21$ . I følge simuleringen i CenitMMI førte  $[\text{O}_2]_{in} = 0,21$  til at  $[\text{O}_2]_a$  kunne ta verdier helt ned i 0,1. Dette vil gi et avvik på 0,14 i forhold til  $[\text{O}_2]_{a,ref}$ . Det var viktig at straffen for å gå over kritisk grense for  $[\text{CO}_2]_a$  var større enn straffen for avvik mellom  $[\text{O}_2]_a$  og  $[\text{O}_2]_{a,ref}$ . I simuleringen var  $[\text{CO}_2]_a = 0,081$  ved  $F_T = 14$ , som betyr at  $\varepsilon_{[\text{CO}_2]_a} = 0,001$ . For å ivareta sikkerheten var følgende nødt å holde:

$$0,001 \cdot r_{z_1} > Q_{z_2} \cdot 0,14 \quad (6.14)$$

Hvor  $0,001 = \varepsilon_{[\text{CO}_2]_a}$  og 0,14 er avvik mellom  $[\text{O}_2]_a$  og  $[\text{O}_2]_{a,ref}$  ved  $F_T = 14$ .  $Q_{z_2}$  var valgt til 10, som gjorde at  $r_{z_1} > 1400$ .  $r_{z_1}$  ble valgt til 2000. Ved å gi stor verdi til  $r_{z_1}$  ville  $r_{z_3}$ -leddet bli neglisjert, derfor holdt det å ha stor verdi på  $r_{z_1}$ . Verdien på  $r_{z_3}$  ble satt til 10, for å gi en ekstra straff på å ikke velge  $[\text{O}_2]_{in} = 0,21$  ved  $[\text{CO}_2]_a > \text{grense}$ .  $r_{z_2}$  ble satt til 1.

CV begrensninger					
CV	$z_{min}$	$z_{maks}$	$r_1$	$r_2$	$\varepsilon$
$z_1 = [\text{CO}_2]_a$	0	0,08	2000	2000	$\infty$
$z_2 = [\text{O}_2]_a$	0,12	1	1	1	$\infty$
$z_3 = \text{sikkerhet}$	0	0	10	10	$\infty$

**Tabell 6.2:** Tabellen viser utvalgte verdier for maks- og minimumsverdier for z-ene, straff på brudd av disse grenseverdiene, gitt av r og mulighet for overskridelse av grensene gitt av  $\varepsilon$ .

MV og DV begrensninger			
MV/DV	$u_{min}$	$u_{maks}$	$\Delta u_{maks}$
$u_1 = [\text{O}_2]_{in}$	0,21	0,96	1
$u_2 = F_T$	4	75	15
$u_3 = F_{puls}$	30	220	150

**Tabell 6.3:** Tabellen viser valgte verdier for maks- og minimumsverdier for u-ene. Begrensning av endringsraten til de ulike u-ene er gitt av  $\Delta u_{maks}$ , hvor  $\Delta u_{min} = -\Delta u_{maks}$ .

Det var ikke nødvendig å tune Kalmanfilteret på nytt da verdiene som ble valgt i kapittel 5 fungerte tilfredsstillende også for prosessmodell med nye parameterverdier. Figur F.1 - F.4 i appendiks F viser estimerte og målte utgangsverdier,  $[\text{O}_2]_a$  og  $[\text{CO}_2]_a$ , ved ulike pustefrekvenser  $F_T$ .

### 6.3.4 Prediksjonshorison

Som beskrevet i teori-delen var en lang prediksjonshorison  $N$  å foretrekke fordi det gav alle CV-ene tid til å stabilisere seg i løpet av  $N$ , selv med store endringer i inngangsverdier. En lang prediksjonshorison øker imidlertid kjøretiden, som dermed kan føre til et uløselig problem for realtime-systemer. På grunn av grisen sin raske dynamikk og forenklingen med neglisjering av tidsforsinkelser, var det ikke nødvendig med en kjempe lang prediksjonshorison for simuleringen. Et sprang i  $F_T$  fra 21 til 10, gjorde at  $[O_2]_{in}$  gikk fra  $\approx 0,37$  til  $0,21$ . Det tok omtrent 120 tidssteg før alle verdiene var stabilisert. Selv om systemet nok ikke vil ha større endringer enn dette, ble prediksjonshorisonen valgt til 150. CV evalueringpunkter ble valgt som default verdiene til CENIT, et punkt for hver 5. tidssteg.

### 6.3.5 Kontrollhorison og parametrisering av MV

Etter å ha testet med ulike verdier, ble kontrollhorisonen valgt til 120. Parametrisering av inngangsverdiene ble valgt til 5-10-20-40-60-80-120 for kontrollhorisonen på 120.

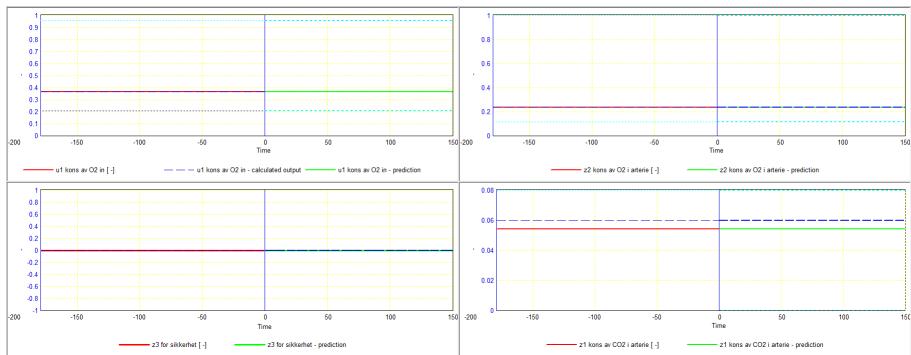
### 6.3.6 Funksjonstest

I modellen til Ottermo og i griseforsøkene er  $F_T$  en manuelt styrt inngangsverdi. Vanligvis ville denne verdien vært en funksjon av blant annet  $[O_2]_a$ . Fordi denne tilbakekobling mellom  $[O_2]_a$  og  $F_T$  mangler, var det litt vanskelig å teste MPCen slik den ville fungert på en gris der pustefrekvensen ikke er styrt manuelt. Sikkerhet og komfort for forsøksgrisen kunne imidlertid sjekkes ved å påtrykke ulike  $F_T$ .

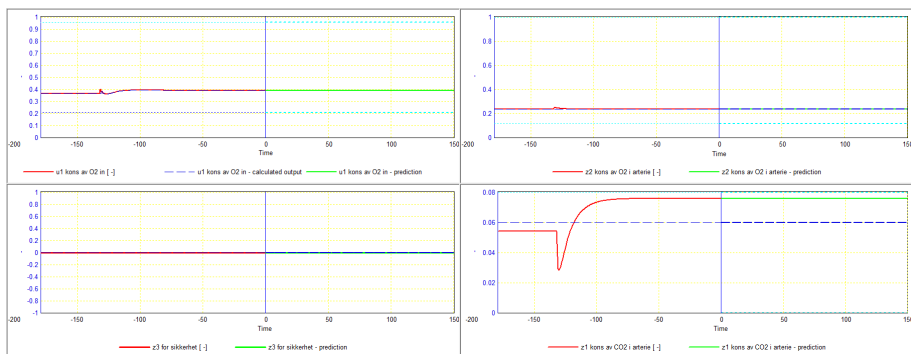
I testen av reguleringssystemet ble pustefrekvensen  $F_T$  endret for å se at sikkerheten var ivaretatt ved lave verdier av  $F_T$  som gav  $[CO_2]_a >$  grense, og at verdien av  $[O_2]_a$  holdt seg nær  $[O_2]_{a,ref}$  for alle andre verdier av  $F_T$  der  $[CO_2]_a <$  grense. Fordi pulsen ikke kan endres manuelt, ble det valgt å bare variere pustefrekvensen for å teste reguleringssystemet.

Figurene 6.4 - 6.7 viser skjermbilder av  $[O_2]_{in}$ ,  $[CO_2]_a$ ,  $[O_2]_a$  og sikkerhetsbryteren  $z_3$ . I figur 6.4 er  $F_T = 21$ , mens  $F_T$  i figur 6.5 er redusert til 15. I figur 6.6, hvor  $F_T$  er satt til 14, går  $[CO_2]_a$  så vidt over grensen, mens  $[CO_2]_a$  går langt over grensen i figur 6.7 hvor  $F_T$  er 10. Testen viser at komfort er ivaretatt når  $[CO_2]_a <$  grense og at sikkerhet er ivaretatt når  $[CO_2]_a >$  grense.

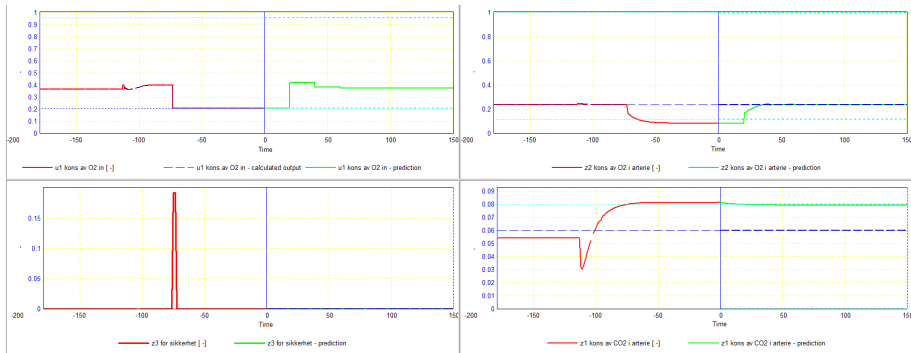
Fordi RealSim måtte resettes før hver endring, er hver endring gjort med utgangspunkt i  $[O_2]_{in} = 0,4$ ,  $F_T = 21$  og  $F_{puls} = 65$ .



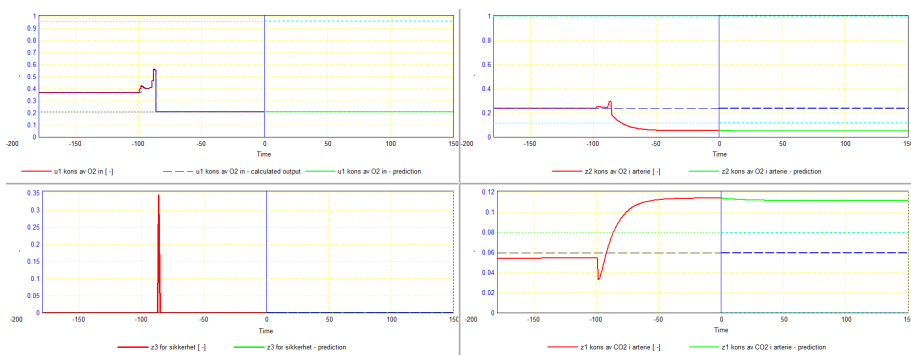
**Figur 6.4:** Figuren viser  $[O_2]_{in}$ ,  $[CO_2]_a$ ,  $[O_2]_a$  og sikkerhetsvariabelen for  $F_T = 21$ . Komforten er ivarettatt ved at  $[O_2]_a \approx [O_2]_{a,ref}$ , så lenge  $[CO_2]_a < 0,08$ . Stiplet blå linje representerer referanseverdier for  $[CO_2]_a$  og  $[O_2]_a$ . Røde verdier er målte verdier, mens grønne er predikterte. De lyseblå verdiene er maks og min verdier for variabelen i plotet.



**Figur 6.5:** Figuren viser et sprang i  $F_T$  fra 21 til 15, hvor  $[O_2]_a$  raskt oppnår  $[O_2]_{a,ref}$  ved å korrigere  $[O_2]_{in}$ . Verdien av  $[CO_2]_a$  øker, men ikke over grensen. Stiplet blå linje representerer referanseverdier for  $[CO_2]_a$  og  $[O_2]_a$ . Røde verdier er målte verdier, mens grønne er predikterte.



**Figur 6.6:** Figuren viser at sikkerheten er ivarettatt dersom  $[\text{CO}_2]_a$  overskrider farlig nivå,  $[\text{CO}_2]_a > 0,08$ , ved at  $[\text{O}_2]_{in}$  settes til 0,21. Siden kostnaden for å ha  $[\text{CO}_2]_a > \text{grense}$  er satt høy, ønsker systemet at  $[\text{CO}_2]_a$  skal ta en verdi  $< \text{grense}$  og regulere  $[\text{O}_2]_a$  til  $[\text{O}_2]_{a,ref}$  igjen. Siden  $F_T$  er satt som en konstant vil ikke  $[\text{CO}_2]_a < \text{grense}$ , slik regulatoren predikerer, så lenge  $F_T$  er 14. Stiplet blå linje representerer referanseverdier for  $[\text{CO}_2]_a$  og  $[\text{O}_2]_a$ . Røde verdier er målte verdier, mens grønne er predikerte.



**Figur 6.7:** Figuren viser et sprang i  $F_T$  fra 21 til 10 og at sikkerheten er ivarettatt når  $[\text{CO}_2]_a > 0,08$ , ved å redusere  $[\text{O}_2]_{in}$  til 0,21. Stiplet blå linje representerer referanseverdier for  $[\text{CO}_2]_a$  og  $[\text{O}_2]_a$ . Røde verdier er målte verdier, mens grønne er predikerte.



## 6.4 Resultat

Den designede reguleringsstrukturen er vist i figur 6.3. Ved implementering av reguleringsstrukturen i CENIT er sikkerheten og mengden  $[O_2]_{in}$  inkludert i regulatoren.

Med valgte verdier på variablene som kunne tunes, vist i tabell 6.2, 6.3 og i ligning (6.8), prediksjonshorisont = 150 og kontrollhorisont = 120, ble både komfort og sikkerhet ivare tatt i henhold til målsetningen og figur 1.1.

## 6.5 Diskusjon

I avsnittene som følger diskuteres hele MPC-delen – reguleringsstrukturen, implementering og tuning, om mål er nådd og kostfunksjon. Diskusjonen er delt inn i delkapittel for å gjøre det mer oversiktlig og lesbart.

### 6.5.1 Reguleringsstruktur

Det valgte designet er enkelt og intuitivt, men har flere forbedringsområder. Når det kommer til sikkerhet, er det flere måter å løse dette på. En lignende metode av valgt løsning, som foreslått av Stavdahl, ville vært å integrere ned  $[O_2]_{in}$  dersom  $[CO_2]_a >$  grense. Da ville pasienten fått mindre og mindre  $[O_2]_{in}$  helt til  $[CO_2]_a$  var under forsvarlig nivå. I mailkontakt med førsteamanuensis Leira skrev han at forhøyet  $[CO_2]_a$  i blodet er såpass farlig, at dersom det skjer må oksygentilførsel slås av og hjelp alarmeres. Denne løsningen, med gradvis reduksjon av oksygen i pustegassen, ble derfor vurdert som dårligere enn valgt løsning.

I designet er regulatoren en MPC. Sikkerhetsblokken implementeres som en del av MPC-en, slik at kostnaden for å krysse farlig  $[CO_2]$ -grense kunne bli satt veldig høy. Regulatoren vil regulere mengden tilført oksygen, slik at verdien av  $[CO_2]_a$  holder seg innenfor referanseområdet sitt. Ved implementering er altså alt utenom «Prosess»-blokken inkludert i MPC-en. Ved implementering av reguleringsstrukturen i programmene til Cybernetica, er både «sikkerhetsbryter» for  $[O_2]_{in}$  ved  $[CO_2]_a >$  grense og høy kostnad for å overskride grense for  $[CO_2]_a$ , innebygd i regulatoren.

Ved et forhøyet nivå av  $CO_2$  i blodet hos en pasient, er det viktig at regulatoren reduserer mengden  $[O_2]_{in}$  slik at  $[O_2]_a$  minker, som igjen fører til at pustefrekvensen øker og lufter ut overflødig  $CO_2$ . I mailkontakt med førsteamanuensis Leira har han forklart at det for KOLS-pasienter vil være store variasjoner i hvor lite oksygen det bør gis i pustegassen, dersom  $[CO_2]_a$  er over referanseverdien sin. Referanseverdien for  $[CO_2]_a$  vil også variere fra pasient til pasient. Det finnes ulike pasienter – de som alltid har et forhøyet  $CO_2$ -nivå fordi de har vendt seg til det og kroppen kompenserer, og de som ikke har et forhøyet  $CO_2$ -nivå fra før. Noen er så følsomme at nesten alt over romluft gir stigning i  $[CO_2]_a$ . Uansett vil en verdi på  $[CO_2]_a$  over referanseverdi til en gitt pasient, indikere at pasienten dekompenierer, er sliten og puster for lite. Pasienten må tvinges til å puste mer ved å øke

pustefrekvensen. For noen pasienter vil det si at mengden  $O_2$  i pustegassen kuttet helt, slik at pasienten bare puster inn oksygen fra luften i rommet. For andre kan ikke  $O_2$  i pustegassen skrues helt av, men må reduseres betydelig til  $CO_2$  igjen er tilbake i referanseområdet.

Ulike pasienter og behov gjør at sikkerhetsbryteren, som reduserer  $[O_2]_{in}$  til et minimum så lenge  $[CO_2]_a >$  grense, må tilpasses hver individuelle pasient. I griseforsøket ble det antatt at grisen tålte at ekstra oksygen i pustegassen ble kuttet helt, det vil si  $[O_2]_{in} = 0,21$ .

For de pasientene som ikke tåler at  $[O_2]_{in}$  kuttet helt, var ikke valgt løsning optimal. I valgt løsning kunne dette problemet vært løst ved å legge til en «saturation»-blokk før inngangen til prosessen. Nedre grense for den individuelle pasienten kunne dermed vært satt som nedre grense i «saturation»-blokken. Ved å benytte MPCens egenskaper og innebygge «sikkerhetsbryteren» i regulatoren, kunne løsningen på problemet vært å sette grensen for  $u_1 = [O_2]_{in}$  til den verdien aktuell pasient tåler.

I og med at det ble antatt at gris, som reguleres i denne oppgaven, tålte at  $[O_2]_{in}$  ble satt til 0, ble individuelle tilpasninger og begrensninger av minimumsverdien til  $[O_2]$  i pustegassen ikke implementert.

Forhåpentligvis fungerer reguleringen med MPC såpass bra, at pasienten sjeldent vil komme i den situasjonen der  $[CO_2]_a$  blir for høy. Noen KOLS-pasienter er så følsomme at nesten alt over romluft gir stigning i  $[CO_2]_a$ . Ved tuning av estimator og regulator til den enkelte vil MPCen kunne prediktere dette og gjøre optimale valg for mengde  $[O_2]_{in}$ .

## 6.5.2 Lineær vs. ulineær MPC

Eldhuset sin modell er ikke observerbar, som gjør at systemets tilstander ikke kan forut-sies med de målingene som gjøres. Modellen er derfor uegnet for både lineær og ulineær MPC. Modellen er i tillegg litt for omfattende og må skrives om, slik at den er observerbar og mindre omfattende, for eventuelt bruk i regulering med MPC. For Ottermo sin gamle modell, beskrevet i [2], har modellen «direct feedthrough»<sup>3</sup>, som gjør modellen uegnet for MPC.

Som beskrevet i avsnittet om ulineær MPC i teoridelen i 6.1.4, er en av fordelene med ulineær MPC er at det gir en mer presis prediksjon som igjen gir bedre regulering. Bedre regulering gir bedre komfort og sikkerhet. Det i seg selv vil være grunn nok til å velge ulineær MPC foran lineær MPC. Programmene til Cybernetica, som brukes i oppgaven, er laget for ulineær MPC som også taler for å bruke ulineær MPC. Ved valg av lineær MPC må modellen lineariseres, som i følge Foss og Heirung [30] kan gi en god nok modell bare dersom modelligningene ikke er for ulineære. En lineære approksimasjonen vil være enda bedre dersom tilstander, inngangsverdier og forstyrrelser ikke varierer så mye – da er det mulig å bruke en enda grovere approksimasjon [30].

Modellen til Ottermo er ikke veldig ulineær, noe som gjør den egnet for linearisering. Det er derimot ikke gitt at forstyrrelsene ikke varierer så mye, som kan sette systemet utenfor det lineariserte området. Kjøretid for lineær MPC er generelt mye bedre enn ulineær MPC,

<sup>3</sup>Den styrbare inngangsverdien inngår som en variabel i ligningssettet for y

som taler for lineær MPC. Det kan derimot argumenteres for at det vil være dumt å bruke tid på linearisering som gir et dårligere resultat, enn å bare bruke modellen som den er. Det at modellen ikke er så ulineær gjør at utregninger blir lettere og kjøretiden grei.

For noe som kan stå om liv og død og gi bedre livskvalitet til et menneske, er det viktig at alt er best mulig. Ulineær MPC gir en mer presis prediksjon og vil dermed gi bedre regulering. Selv om det potensielt kan bety høyere kjøretid, er modellen såpass enkel at utregningene ikke vil være veldig harde. Det er i tillegg mulig å redusere kjøretiden ved å benytte parametrisering av MV og kontrollvariabel evalueringpunkter.

Argumentene gitt i de foregående avsnittene er grunnen til at ulineær MPC ble vurdert som mest egnet ved regulering av oksygenkonsentrasjon hos KOLS-pasienter med Ottermo sin modell.

### 6.5.3 Implementering og tuning

Systemet er tunet for å opprettholde komfort og sikkerhet, som beskrevet i 1.3 og vist i figur 1.1.

Den øvre grensen for  $[O_2]_a$  ble valgt som 1, fordi det i seg selv ikke er farlig med et høyt nivå av  $O_2$  i blodet.  $[O_2]_a$  vil uansett være begrenset av straffen for høyt nivå av  $[CO_2]_a$ . Et for høyt nivå av  $[O_2]_a$ , fører til en lav pustefrekvens, som igjen fører til et forhøyet nivå av  $[CO_2]_a$ , som vil bli straffet høyt i kostfunksjonen. Dermed vil høye verdier av  $[O_2]_a$  være begrenset av kostfunksjonen.

Verdiene i  $\mathbf{r}$  og  $\mathbf{Q}$  er tilpasset slik at kostnaden ved å overskride grense og ikke velge  $[O_2]_{in} = 0,21$ , er større enn kostnaden for å avvike fra referanseverdien til  $[O_2]_a$  – for alle tilfeller. Verdiene som tilhører sikkerheten er satt høye, men så lenge  $[CO_2]_a < \text{grense}$  har ikke disse verdiene noe å si – da vil det bare være avvik fra  $[O_2]_a$  i forhold til  $[O_2]_{a,ref}$  som vil gi verdi til J. Det hadde gått an å satt verdien for  $r_{z_1}$  høyere enn det som valgt, for å være helt sikker på at sikkerheten er høyest vektet for alle modeller.

Det er ok å velge  $r_{z_2} = 1$  fordi det ikke må straffes at  $[O_2]_a$  går under grensen sin, som bare vil skje når  $[CO_2]_a > \text{grense}$ . Da settes  $[O_2]_{in}$  til 0,21 som gjør at  $[O_2]_a$  blir lav.  $\mathbf{Q}_{z_2}$  passer på at det straffer seg å ha  $[O_2]_a$  ulik  $[O_2]_{a,ref}$ . Verdien her er imidlertid mindre enn verdien for å gå over grensen for  $[CO_2]_a$ , slik at sikkerheten kommer før komfort, som beskrevet i forrige avsnitt. Ved verdier av  $[CO_2]_a < \text{grense}$  vil det være avvik mellom  $[O_2]_a$  og  $[O_2]_{a,ref}$  som gir verdi til J. Da vil det beste for optimeringsproblemet være å holde  $[O_2]_a$  så nær referansen som mulig. Det vil gi den laveste verdien.

Verdiene for  $\mathbf{S}$  ble valgt til 1, fordi det ikke var ønskelig å straffe endringer i inngangsverdi og forstyrrelser. Forstyrrelsene kan ta mange ulike verdier, og i forsøket er gjerne  $F_T$  et sprang fra en verdi til en annen. Av sikkerhetshensyn er det nødvendig at  $[O_2]_{in}$  kan gå fra en verdi til 0,21 på kort tid – det er ikke ønskelig å straffe det.

$\Delta U$  ble valgt til verdier som gjorde at systemet kunne hoppe fra en verdi til en helt annen i løpet av et tidssteg. Dette er en ok antagelse for estimeringen, der det er blitt gjort antagelser som blant annet å neglisjere tidsforsinkelser. I et virkelige system vil antageligvis

ikke oksygenkonsentratoren kunne endre  $[O_2]_a$  fra lavt til høyt på veldig kort tid. Det tar tid å bygge opp konsentrasjon, og tid fra endringen gjøres i konsentratoren til ny oksygenkonsentrasjon når arteriene. For et virkelig system vil det heller ikke være mulig å kunne måle endring i pustefrekvens i løpet av et sekund – et innpust kan ta flere sekunder.

Romslige grenser for  $U$  gjør at problemet ikke begrenses og settes utenfor løsningsområdet for forstyrrelser av alle verdier. Grensene for  $u_1 = [O_2]_{in}$  er ikke mer romslige enn de må være og er reelle. Ulike verdier innenfor grensen til  $u_1$  vil sjekkes for hvert tidskritt, og optimeringsalgoritmen velger inngangsverdien som gir lavest verdi til kostfunksjonen.

Prediksjons- og kontrollhorisont er valgt til henholdsvis 150 og 120. Det kunne vært brukt tid på å tilpasse disse verdiene enda bedre, men valgte verdier gir tilfredstillende resultater i simuleringen. Horisontene er lange nok til at verdier får tid til å stabilisere seg ved store sprang i forstyrrelsene. I et virkelig system vil det kanskje være slik at pasienten vanligvis sitter mye i ro, som vil gjøre at  $F_T$  og  $F_{puls}$  holder seg nokså konstant. Her vil det også være tidsforsinkelser, som gjør at prediksjons- og kontrollhorisont burde vært litt lenger. Det kan ta tid før verdier stabiliserer seg, men samtidig vil det nok ikke være like store variasjoner og sprang som det er i griseforsøket og simuleringen.

Det ble antatt at det ikke var nødvendig å tune Kalmanfilteret, siden det fungerte såpass bra på initiell modell – som har mye mer annerledes parameterverdier enn modell for prosess brukt i dette kapitlet. Det viste seg å stemme. Likevel kunne det nok vært brukt tid på å tilpasse estimeringsmodellen enda bedre til prosessmodellen ved å endre på standardavvikene. I figur F.1 - F.4 viser målt og estimert utgangsverdi ved  $F_T$  lik henholdsvis 21, 15, 14 og 10. For de ulike scenariene estimeres utgangsverdien tilfredstillende med valgte verdier for Kalmanfilteret og prosessmodell med avvikende parameterverdier gitt av gaussisk fordeling.

#### 6.5.4 Komfort og sikkerhet

Det var viktig at sikkerhet hadde høyere prioritet enn komfort, og det er oppnådd i denne oppgaven. Ved farlig høyt nivå av  $CO_2$  i blodet, måtte komforten vike for å tvinge pasienten til å puste hyppigere. Regulatoren er tunet slik at sikkerheten hos pasienten er ivaretatt, selv for tilfeller der  $[CO_2]_a$  bare så vidt overskrider grensen. Dersom verdien av  $[CO_2]_a >$  grense, vil variablene i kostfunksjonen straffe dette avviket mer enn avvik mellom  $[O_2]_a$  og  $[O_2]_{a,ref}$  – som er leddet som gir mest straff når  $[CO_2]_a <$  grense og leddet som ivaretar komforten.

Fra testene gjort i 6.3.6 var det tydelig at regulatoren holder nivået av  $[O_2]_a$  nær referanseverdien for alle tilfeller av  $F_T$ , utenom der  $F_T$  blir så lav at  $[CO_2]_a >$  grense. For verdier der  $[CO_2]_a$  var over grense viser testen at  $[O_2]_{in}$  settes til 0,21, slik at komforten blir nedprioritert i forhold til sikkerheten.

I testen ble  $F_T = 14$  valgt fordi det gav  $[CO_2]_a$  så vidt over 0,08. I griseforsøket var det  $F_T = 10$  som gav kritisk høy  $[CO_2]_a$ , men fordi parameterne var endret i modellen for prosessen, ble den nye, farlige grensen for  $F_T = 14$  i simuleringen. På grunn av den høye verdien til  $r_{z_1}$  ville de fleste verdier av  $[CO_2]_a >$  grense gi høye verdier til  $J$ .  $[CO_2]_a$  så vidt

over grensen gav bare en liten økning i  $J$ . Det var viktig å passe på at verdien for sikkerhet var høyere enn vekten av avviket mellom  $[O_2]_a$  og  $[O_2]_{a,ref}$ , gitt av  $Q$ , for å være sikker på at sikkerheten ikke ble hindret.

Det hadde kanskje gitt mening å hatt en test der  $F_T$  fra griseforsøket ble programmert inn, for å sjekke at komfort og sikkerhet var ivaretatt for pustefrekvensene som ble påført grisen i forsøket. Imidlertid viser ingen av testene hva som ville skjedd i virkeligheten: Lav  $[O_2]_{in}$  ville ført til hyppigere pustefrekvens for å opprettholde nivå av  $[O_2]_a$  og dermed luftet ut  $CO_2$ , og høy  $[O_2]_{in}$  ville ført til en redusert pustefrekvens.

Modellene mangler tilbakekobling som regulerer  $F_T$ , derfor forble nivået av  $CO_2$  høyt. Fordi kostnaden av å ha  $[CO_2] > \text{grense}$  er stor er det ønskelig for optimeringsproblemet å redusere den. Det blir predikert  $[CO_2] < \text{grense}$  for  $F_T = 14$ .

For de pasientene som ikke tåler at  $[O_2]_{in}$  settes til 0,21, kan verdien endres ved å endre tallet i definisjonen av ny  $z$ -variabel i 6.3.1, der det står 0,21, til den minimale verdien pasienten tåler. På den måten kan «sikkerhetsbryteren» tilpasses den enkelte.

### 6.5.5 Kostfunksjon

Kostfunksjonen er gitt som en standard fra Cybernetica sin side, men det hadde nok vært mulig å programmere den om til en annen om ønskelig. Stavdahl har påpekt at det ideelt ville det vært ønskelig å straffe øvre og nedre grense for  $z_1 = [CO_2]_a$  og  $z_2 = [O_2]_a$  ulikt. Det burde for eksempel vært høyere straff for å nærme seg øvre, farlige grense for  $[CO_2]_a$  enn nedre grense. Det burde også vært høyere straff på å gå mot nedre grense til  $[O_2]_a$  enn øvre.

Selv om ikke grensene er straffet ulikt, er samme effekt oppnådd i oppgaven. Kostfunksjonen fungerer bra og gir et godt resultat med de testene som er gjort. Ved å velge nedre grense for  $[CO_2]_a = 0$  og øvre grense til  $[O_2]_a = 1$  oppnår en effekten av å straffe grensene ulikt.  $[CO_2]_a$  vil aldri bli 0 og  $[O_2]_a$  vil aldri bli 1. Systemet vil alltid prøve å oppnå minimum av  $J$ , som vil være å holde seg rundt referanseverdien for  $[O_2]_a$  og under grense for  $[CO_2]_a$ .

Å lage en kostfunksjon som straffer øvre og nedre grense ulikt ville krevd mye mer tuning. I tillegg ville det vært jobb å endre eksisterende kostfunksjon, hvis det i det hele tatt er mulig. Om eksisterende kostfunksjon gjør jobben, vil det ikke være grunn til å endre den.

# Sammenligning av PID og MPC

I tidligere arbeid [2], ble en tidligere versjon av modellen til Ottermo forsøkt regulert med en PI-regulator. Elementer av reguleringen og resultatene kan sammenlignes, for å gi en indikasjon på fordeler, ulemper og foretrukket reguleringsstruktur.

Førstkommende delkapittel viser en figur av resultatet av reguleringen med PI-regulator [2]. I metodedelen beskrives det hvordan ulike deler av regulering med PI og MPC ble sammenlignet for å kunne bestemme fordeler og ulemper med respektive reguleringsstrukturer. Sammenligningen og resultatet diskuteres i et eget delkapittel til slutt.

## 7.1 Resultat fra PI-regulering

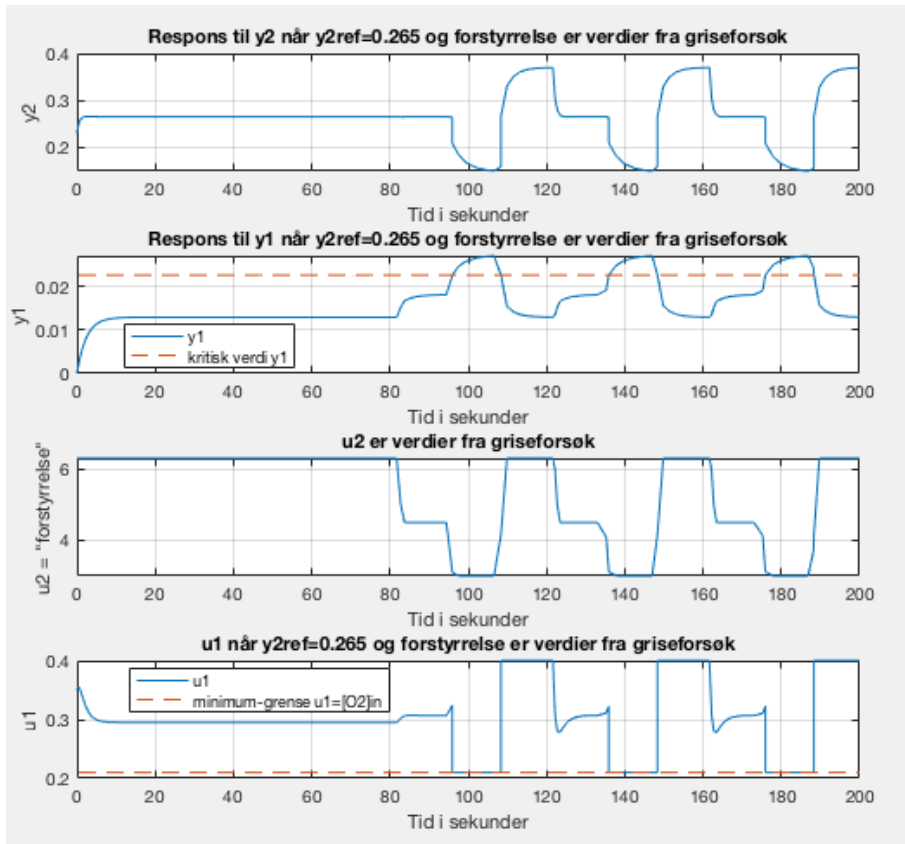
Resultatet fra reguleringen gjort i [2] for den ulineære modellen er gjengitt under, i figur 7.1.

## 7.2 Metode

Målet med regulering av oksygen i pustegassen hos KOLS-pasienter er å ivareta komfort og sikkerhet. Regulatoren som best ivaretar dette er den beste. Ved å sammenligne resultatene fra MPC- og PID-reguleringen, ble det vurdert hvilken regulator som best ivaretok både komfort og sikkerhet.

Det ble stilt ulike spørsmål for å avgjøre om regulatoren ivaretok sikkerhet og komfort på en god måte:

- 1) Blir  $[O_2]_{in}$  satt til 0,21 dersom  $[CO_2]_a > \text{grense}$ ?



**Figur 7.1:** Figuren viser PI-regulering av  $y_2 = [\text{O}_2]_a$ , ved å endre på  $u_1 = [\text{O}_2]_{in}$ , i den forenklete, ulinære modellen til Ottermo (annen modell enn den som er brukt i denne oppgaven).  $u_2$  er her verdier fra griseforsøk A. Legg merke til at farlig grense for  $[\text{CO}_2]_a$  her er satt til 0,0227. Denne grensen ble valgt fordi modellen gav  $[\text{CO}_2]_a = 0,0227$  ved  $F_T = 12$  og  $[\text{O}_2]_{in} = 0,21$ . I tillegg er grensen for  $[\text{O}_2]_{in,max}$  satt til 0,4. Dette er litt andre tall enn det som er valgt i denne oppgaven, men prinsippet for sikkerhetsbryteren er den samme.

- 2) Unngår regulatoren situasjoner der det nærmer seg farlig grense for  $[\text{CO}_2]_a$  for pasienten?
- 3) Holdes verdien av  $[\text{O}_2]_a$  nokså konstant selv om  $F_T$  og  $F_{puls}$  varierer?

I tillegg ble fordeler og ulemper med de ulike regulatorene tatt i betraktning.

## 7.3 Resultat

MPC er best egnet for regulering av oksygenkonsentrasjon hos KOLS-pasienter.

## 7.4 Diskusjon

### Sikkerhet

Figur 7.1 viser at sikkerhetsbryteren for PI-regulatoren virker – dersom  $[\text{CO}_2]_a >$  grense, som her er satt til 0,0227, settes  $[\text{O}_2]_{in}$  til 0,21. I tidligere arbeid [2] er det skrevet «Logikken som er implementert ivaretar sikkerheten, men kunne nok vært gjort smartere for å bedre ivareta komforten til pasienten.». I MPCen er det lagt inn en høy kostnad på å overskride grensen til  $[\text{CO}_2]_a$  - en såkalt smartere måte for å bedre ivareta komforten og sikkerhet til pasienten.

For MPC fungerer også sikkerhetsbryteren. I tillegg vil MPC regulere  $[\text{O}_2]_{in}$  slik at systemet unngår å nærme seg farlig grense, noe som vil være mye mer komfortabelt for pasienten. PI-regulatoren er mye «dummere» og vil bare regulere oksygenmengden slik at  $[\text{O}_2]_a$  er så lik  $[\text{O}_2]_{a,ref}$  som mulig. PI-regulatoren vil ikke redusere mengden  $[\text{O}_2]_{in}$  selv om  $[\text{CO}_2]_a$  nærmer seg farlig grense. Det er først når  $[\text{CO}_2]_a$  bikker grensen at  $[\text{O}_2]_{in}$  settes til 0,21.

### Komfort

Ved ulike verdier av  $F_T$ , som ikke er så lave at  $[\text{CO}_2]_a >$  grense, vil MPC holde nivået av  $[\text{O}_2]_a$  stabilt og nær referanseverdien for  $[\text{O}_2]_a$ . Dette er vist både i figur 6.4 og 6.5, hvor  $F_T$  enten ikke er endret eller bare endret fra en verdi til en annen.

Sikkerheten er ivaretatt for PI-reguleringen i [2], men komforten kunne vært bedre. Regulatoren «ser» ikke at farlig grense for  $[\text{CO}_2]_a$  nærmer seg, og at det vil føre til at  $[\text{O}_2]_{in}$  settes til 0,21 og vil gjøre det ubehagelig for pasienten.

For PI-regulatoren er endringene i  $F_T$  litt for raske eller regulatoren litt for treg, slik at regulatoren ikke rekker å få  $[\text{O}_2]_a = [\text{O}_2]_{a,ref}$  før neste sprang i  $F_T$ . Figur 7.1 viser at regulatoren holder  $[\text{O}_2]_a$  på et stabilt og ønsket nivå fra  $t = 0$  til  $t \approx 95$ , tiden før  $[\text{CO}_2]_a >$  grense. Etter at  $[\text{CO}_2]_a >$  grense blir endringene i  $[\text{O}_2]_{in}$  så store at regulatoren ikke rekker å stabilisere  $[\text{O}_2]_a$  før neste sprang i  $F_T$ .



De hyppigste endringene i  $F_T$  i griseforsøk A er bare etter 10 tidsskritt. I forsøk A er hvert tidsskritt  $\geq 30$  sek, så det at tidsskrittene i PI-reguleringen er i sekunder er ikke helt representativt.  $10 \cdot 30 \text{ sek} = 300 \text{ sek} = 5$  minutter, gjør at grisen i forsøk A har god tid på å stabilisere seg før et nytt sprang i  $F_T$  blir påsatt. I [2] burde ikke sprangene i  $F_T$  vært så hyppige. De kommer mye raskere enn det er gjort i forsøk A, og det igjen er hyppige endringer i forhold til virkeligheten. Ved mindre hyppige endringer i  $F_T$ , ville nok PI-regulatoren klart å regulere  $[O_2]_a$  til referanseverdien igjen. I [2] er det gjort tester av regulatoren uten sikkerhetsbryteren. Her holdes nivået av  $[O_2]_a \approx [O_2]_{a,ref}$  for alle verdier av  $F_T$  gitt i griseforsøk A.

For et virkelig system, der  $F_T$  ikke er styrt manuelt, vil nok ikke  $F_T$  variere like mye som i griseforsøk A. PI-regulatoren vil nok i mange tilfeller klare å holde  $[O_2]_a$  nær referansen, men vil ikke unngå situasjoner der  $[CO_2]_a$  nærmer seg farlig grense – som vil gå ut over komforten til pasienten.

For MPCen er det ikke testet med  $u_2 = F_T =$  verdier fra griseforsøk A, men regulatoren er rask og det er antatt at verdiene raskt ville stabilisert seg. Som vist i figur 6.5 vil en endring i  $F_T$  fra 21 til 15 omtrent ikke ta noe tid før  $[O_2]_a = [O_2]_{a,ref}$ . Det hadde vært interessant om pustefrekvensene  $F_T$  fra griseforsøk A hadde vært implementert i CENIT for å se hvor godt MPCen hadde taklet situasjonen.

### Annet

En klar fordel med reguleringsstrukturen og det som er gjort i denne oppgaven, er modelltilpasning med ModelFit. En bedre modell gir helt klart bedre regulering. Også estimatoren, som tilpasser modellen underveis, er med på å gjøre reguleringen med MPC veldig god.

Andre fordeler med MPC, som PID mangler, er evnen til å prediktere fremtidige hendelser og ta beslutninger basert på det som er best nå og i fremtiden. PID er «dummere», og velger bare det beste alternativet her og nå. Dette gjør MPCen unik og verdien som velges vil være optimal i forhold til de kriteriene som settes.

Det er også bedre muligheter for å skreddersy en løsning til systemet med MPC. Ved å straffe avvik fra referanseverdier og straffe brudd på grenser, vil optimeringsproblemet i MPC prøve å unngå situasjoner der det er fare for at kostfunksjonen tar høye verdier. PID har ikke mulighet til å forutse slike hendelser.

En fordel med PID-regulering er at den er veldig enkelt og intuitivt og har få variabler å tune.

## **Del IV**

# **Oppsummering og videre arbeid**



## Diskusjon

I dette kapitlet gis det en overordnet og oppsummerende diskusjon til det som er gjort i de foregående kapitlene og hvordan det kan brukes videre. Fremtidige forsøk på gris diskuteres, etter ønske fra biveileder Ottermo.

### 8.1 Modelltilpasning i ModelFit

Offline parameterestimering i ModelFit gjorde at modellen ble tilpasset bra til målingene fra griseforsøkene. De estimerte parameterverdiene var ikke realistiske verdier, men kompenserte for mangler i modellen. Som diskutert i tidligere diskusjon har det ingenting å si om parameterne ikke har realistiske verdier, når formålet med parameterestimeringen er regulering av systemet.

Valget av hvilke parametere som skulle offline parameterestimeres var bra – en volumparameter, en  $[O_2]$ -parameter og en  $[CO_2]$ -parameter. Det hadde vært mulig å teste om flere målinger hadde bidratt til at flere parametere kunne vært estimert. Da kunne det også vært mulig å teste ut flere kombinasjoner av parametere for å potensielt tilpasse modellen enda bedre til målingene.

Ved offline parameterestimering i kapittel 4 er de største avvikene mellom modell og målinger for  $[CO_2]_a$  når  $F_T$  avviker fra  $F_T = 15$ . Dette er tydelig både i figur 4.5, hvor  $V_B$  ikke estimeres, og 4.6, hvor  $V_B$  estimeres. Estimering av parameteren  $V_B$  gjør at dynamikken tilpasses bedre, men avvikene for  $F_T = 21$  og  $F_T = 10$  er omtrent like store som ved offline parameterestimering uten  $V_B$ . Det hadde vært mulig å gjøre endringer i ligningen for  $[CO_2]_a$  slik at  $[CO_2]_a$  ikke er like avhengig av  $F_T$  når  $F_T$  avviker mye fra  $F_T = 15$ .

Standardavvikene for både prosess-, parameter- og målestøy var forholdsvis lave. Dette representere heller ikke nødvendigvis virkeligheten – noe som er uviktig når formålet med

tuningen er estimering for å regulere prosessen på en god måte. Med valgt standardavvik for Kalmanfilteret, varierte estimatene av parameterne lite over tid, og forbedret likheten mellom modell og måling fra prosess. Det var fremdeles noen uoverensstemmelser. Disse avvikene kunne mest sannsynlig vært dempet ved å fintune estimatoren, men fordi estimatoren fungerte tilfredsstillende ble det ikke vurdert som nødvendig.

Kalmanfilteret var tunet i åpen sløyfe i ModelFit. Ved online parameterestimering ble både modell med offline parameterestimerte parameterverdier og modell med initielt gjattede parameterverdier tilpasset veldig bra til målingene fra griseforsøkene. Av den grunn kan det spekuleres i om det er nødvendig å gjøre offline parameterestimering for hver individuelle gris. Optimale parameterverdier bestemt ved offline parameterestimering for griseforsøk A, tilpasset modellen bedre til griseforsøk B enn ved initielt gjattede parameterverdier. Ved å benytte offline parameterestimerte verdier fra griseforsøk A i stede for offline parameterestimering for hver individuelle prosess, kan det hende Kalmanfilteret tilpasser modellen til prosessen tilfredsstillende.

Linnestad fra Cybernetica har skrevet at det vil være rimelig med høye standardavvik dersom det er forventet at parameterne skal variere mye over tid for en gitt gris eller dersom de skal gjelde for flere griser der det ikke er mulig å gjøre offline parameterestimering til hver enkelt gris på forhånd. På en annen side har han også skrevet at om det er mulig å gjøre offline parameterestimering for hver individuelle gris vil det være en fordel. Modellen vil være bedre tilpasset målingene fra prosessen med offline parameterestimering. Ved å øke standardavvikene på Kalmanfilteret kan det også hende at estimatene av parameterne varierer mer over tid, noe som vil gå ut over reguleringen.

### **Fordeler og ulemper**

Fordelene med tilpasning av modell i ModelFit er at tilpasningen mellom modell og målinger fra prosessen blir bra. Programmet er enkelt og intuitivt og det er mulig å sette grenser slik at parametere ikke estimeres utenfor gitte rammer. Som diskutert, går det fint at parametere som estimeres med offline parameterestimering i ModelFit ikke tar realistiske verdier når tilpasningen mellom modell og prosess gjøres for å regulere prosessen.

Ved bruk av ModelFit i forsøk kan mye av jobben gjøres på forhånd, slik at det eneste som gjenstår under forsøket er å legge inn verdier fra målingene og estimere parameterne. Optimerte parameterverdier fra griseforsøk A kan legges inn som startverdier. Da vil det være mulig å se om tilpasningen mellom modell og målinger fra nytt forsøk er gode, før offline parameterestimering for den nye grisen gjøres.

Ut fra målingene som allerede er gjort i to forsøkene A og B, ser det ut som parameterverdiene er ganske like fra gris til gris. Det bør være nokså like standardavvik også. I et nytt forsøk vil det da egentlig bare være å offline parameterestimere de tre valgte parametere, og så eventuelt tune Kalmanfilteret litt, dersom det vurderes som nødvendig. Offline parameterestimering er fort gjort, og antageligvis trenger ikke standardavvikene for Kalmanfilteret å endres så mye.

Ulemper med å bruke ModelFit til modelltilpasning kan være at modellen må defineres ut

fra Cybernetica sin definisjon og legges inn i Visual Studios. Det er litt tidkrevende og tar litt tid å sette seg inn i hvordan programmet fungerer.

I ModelFit-interfacet er det flere ting å holde styr på, f.eks. å sette 1-tall på riktige steder som indikerer hvilke målinger som skal brukes i optimeringen og riktig perturbasjonssteg. Som diskutert i tidligere avsnitt er det mulig å gjøre alt dette på forhånd, slik at en slipper å tenke på dette under forsøket. Når en først skjønner ModelFit, som ikke tar veldig lang tid, er det et veldig bra og nyttig verktøy.

## 8.2 Regulering i CENIT og RealSim

Den designede reguleringsstrukturen implementeres ikke direkte slik den er designet i 6.2. Sikkerhetsblokken som er designet i figur 6.3, tas i programvaren til Cybernetica med i både restriksjoner og kostfunksjon. På den måten velger regulatoren optimal løsning, der både sikkerhet og komfort er tatt i betraktning. Sikkerhet har høyest prioritet, som gjør at sikkerhet kommer foran komforten.

Optimeringsproblemet straffer avvik mellom  $[O_2]_a$  og  $[O_2]_{a,ref}$ , som ivaretar komfort. Ved  $[CO_2]_a$ -verdier over en gitt grense gis det en stor verdi til kostfunksjonen. Denne verdien er større enn verdien som legges til kostfunksjonen for avviket mellom  $[O_2]_a$  og  $[O_2]_{a,ref}$ , som gjør at det straffes mer å gå over farlig  $[CO_2]$ -grense enn å opprettholde komforten. Dersom  $[CO_2]_a$  mot formodning skulle gå over grensen, tvinges  $[O_2]_{in}$  til en minimumverdi for å tvinge pasienten til å puste hyppigere.

Ved simulering av regulering av en prosess, der prosessen er representert av en prosessmodell med gaussisk varierte parameterverdier, var det ikke nødvendig å tune Kalmanfilteret mer enn det som allerede var gjort i åpen sløyfe i ModelFit. Estimeringen av utgangsverdier med valgt Kalmanfilter gav tilfredstillende resultater, som vist i figurene i appendiks F.

Prosesen som reguleres i oppgaven – en prosessmodell med gaussisk varierende parameterverdier – har ingen tidsforsinkelser, og vil ikke være en fullgod representasjon av regulering av et virkelig system. For å sjekke om reguleringsstrukturen hadde håndtert tidsforsinkelser vil det være nødvendig å legge til realistiske tidsforsinkelser i prosessmodellen.

Test av robustheten til MPCen, vil gi et bilde av hvor mye regulatoren tåler. Test av robustheten sammen med simulering av regulering av prosess med tidsforsinkelser vil gi et mer realistisk bilde av hvordan regulatoren vil regulere en virkelig prosess og om eventuelt Kalmanfilteret må tunes på nytt for å passe til forsøk på en virkelig prosess med tidsforsinkelser.

I simuleringen av reguleringen av prosessen er både kontrollvariabel evalueringpunkter og parametrisering av MV benyttet. Dette reduserer kjøretiden til det ulineære optimeringsproblemet. For videre bruk kan det være nødvendig å endre disse punktene dersom kjøretiden blir for lang og går ut over reguleringen.

### **Fordeler og ulemper**

Fordelene med CENIT og RealSim er at en kan simulere reguleringen av prosessen med ulike modeller og tune regulator og Kalmanfilter deretter. Det er mulig å legge inn ulike modeller for prosess og estimering. I CENIT MMI, brukergrensesnittet til CENIT, kan det legges inn grafer som viser estimerte, predikerte og målte verdier. Tuningsvariablene til regulatoren kan enkelt endres og det er også enkelt å skru av og på Kalmanfilteret og endre standardavvik for å tune estimatoren.

Som med ModelFit gjelder det at når en først skjønner programvaren er det veldig gode og nyttige verktøy. Det tar forøvrig litt lenger tid å gjøre seg kjent med CENIT og RealSim, og litt flere ting å passe på at gjøres på riktig måte, som f.eks. å endre på en del konfigureringsfiler o.l. og passe på at filer ligger i riktig mappe. Også her kan mye settes opp på forhånd, slik at det vil være enkelt å bruke under forsøk. Ved å følge instruksjer gitt fra Cybernetica under kurs [18], bør det være mulig å få til både CENIT og RealSim.

Ved regulering av en prosess må det settes opp OPC-tilkobling, som kobler sammen prosess med regulator. Det krever nok litt tid å sette seg inn i dette og, og er ikke noe som er gjort i denne avhandlingen. OPC-tilkoblingen i simuleringen ble bare satt opp lokalt på samme PC som simuleringen kjørte på.

## **8.3 Fremtidige forsøk på gris**

Biveileder Ottermo har ytret ønske om at et fremtidig forsøk på gris blir vurdert ut i fra det som er gjort i oppgaven, både med tanke på parameterestimering i ModelFit og regulering i CENIT.

I neste forsøk som skal gjøres på gris, har biveileder Ottermo fortalt at vil det blir gjort på lignende måte som tidligere. De første timene vil bli brukt på å teste grisen, for å gi verdier som modellen kan tilpasses etter med offline og online parameterestimering i ModelFit. Målingene av  $O_2$  og  $CO_2$  vil bli gjort med ikke-invasive målemetoder – pustoksymetri for  $[O_2]_a$  og kapnografi for  $[CO_2]_a$ .

I følge resultatene fra del II kan det virke som det kanskje ikke er nødvendig å gjøre parameterestimering for hver gris – tilpasningen mellom modell og måling fungerer tilfredstillende med optimale parameterverdier fra griseforsøk A. Med valgt Kalmanfilter tilpasses modellen veldig bra til målingene, selv modell med initielle parameterverdier. Det er derfor grunn til å tro at parameterestimeringen i ModelFit kunne vært droppet og heller fintunet Kalmanfilteret i CENIT.

Det ville vært interessant å sett om hypotesen om at optimale parameterverdier fra resultatene i kapittel 4 og tunet Kalmanfilter fra kapittel 5 ville gjort nytten ved regulering av systemet. Det kan gjøres en test av det i ModelFit – ved å legge inn målingene på den nye grisen og bruke tilpasningen som ble gjort tidligere i denne oppgaven, for å se om det vil være en vesentlig forbedring i reguleringen dersom parametere i modellen til den nye grisen estimeres med offline og online parameterestimering først. Som argumentert

av Linnestad, vil det være nyttig å gjøre parameterestimering for hver gris, dersom det er mulig. En bedre tilpasning gir en bedre regulering.

Fordi prosessen inneholder tidsforsinkelser og andre uoverensstemmelser med modellen, vil det være nødvendig å gjøre tester i CENIT og RealSim med en prosessmodell med tidsforsinkelser og ikke bare gaussisk varierte parametereverdier, som i kapittel 6. Om filteret ikke fungerer like bra for prosessen med tidsforsinkelse, kan Kalmanfilteret tunes slik at verdier fra modell tilpasses til prosess med tidsforsinkelser, ved å endre på standardavvikene til parameter-, prosess- og målestøy. Fordi modellen ikke inneholder tidsforsinkelser vil det nok være nødvendig å øke standardavvikene for at Kalmanfilteret skal ta mer hensyn til målingene ved oppdatering av estimerte parametereverdier. Om det skulle bli vanskelig å tilpasse modellen uten tidsforsinkelser til en prosessmodell med tidsforsinkelser, vil det også være mulig å legge til tidsforsinkelser i modellen til Ottermo.

Fordi testingen gjøres på samme måte som i de andre forsøkene, vil ikke tidsforsinkelsene fra prosessen være synlige i målingene som legges inn i ModelFit. Unntaket er hvis ikke målingene fra det nye forsøket gjøres mye mer kontinuerlig, slik at forsinkelsene er synlige i målingene – noe det kanskje vil gjøre med de ikke-invasive målemetodene. Men dersom det tas utgangspunkt i at målingene gjøres på samme vis som i de tidligere forsøkene, må tilpasningen i forhold til tidsforsinkelser kompenseres for ved tuning av Kalmanfilter og regulator i lukket sløye. Ved å på forhånd ha gjort simuleringer med prosessmodell med tidsforsinkelser, vil det forhåpentligvis være enkelt å tune reguleringssystemet slik at oksygenkonsentrasjonen hos grisen reguleres på en god og trygg måte, som ivaretar både komfort og sikkerhet.

Når det kommer til selve forsøket, styres pustefrekvensen manuelt som i de andre forsøkene. Dette er nødvendig for å «tvinge grisen til å ha KOLS». Ved lave pustefrekvenser vil nivået av  $[CO_2]_a$  stige. Ved å gjøre sammen funksjonstest som i 6.3.6, vil det være mulig å sjekke om komfort og sikkerhet er ivare tatt for grisen.





# Kapittel 9

## Konklusjon

I dette kapittelet gis det en konklusjon av offline og online parameterestimering i ModelFit, MPC og regulering i CENIT og RealSim, samt en oppsummering av evaluering av programvare som er brukt i oppgaven.

### 9.1 Parameterestimering

Den forholdsvis enkle modellen til Ottermo tilpasses bra til målingene fra griseforsøket med offline parameterestimering i Cybernetica ModelFit. De offline parameterestimerte parameterverdiene, som tilpasser utgangene fra modellen til målingene, er ikke realistiske verdier – men det er heller ikke nødvendig når formålet med modelltilpasningen er regulering av systemet. En modell som er bedre tilpasset til systemet gir bedre regulering enn en modell med realistiske parameterverdier som er dårligere tilpasset til målingene fra prosessen. Parameterestimeringen vurderes derfor som god med Cybernetica ModelFit, og nyttig til fremtidige forsøk.

Tuning av Kalmanfilter og online parameterestimering, både i ModelFit og CENIT, tilpasser modellen svært bra til målingene fra begge griseforsøkene. Også modell uten offline parameterestimerte parameterverdier tilpasses bra til målingene med valgt Kalmanfilter.

I et nytt griseforsøk vil det være nødvendig med både offline og online parameterestimering, men verdier funnet ved parameterestimering for forsøk A kan brukes som utgangspunkt.

## 9.2 MPC

Ved å sammenligne resultatet fra MPC-reguleringen med erfaringer fra PI-regulering i tidligere arbeid [2], var det tydelig at MPC-regulering var å foretrekke ved regulering av oksygenkonsentrasjon hos KOLS-pasienter. I tillegg var det klart at ulineær MPC var bedre egnet enn lineær MPC. Tuning av MPCen gjør at både komfort og sikkerhet er ivare tatt i henhold til målsetning i 1.3 og figur 1.1.

Kalmanfilteret fungerer tilfredsstillende ved simulering av regulering av systemet, der systemet er representert av en prosessmodell med gaussisk varierte parameterverdier, og det var ikke nødvendig å tune filteret mer i lukket sløyfe. Det gjenstår å se om Kalmanfilteret estimerer parameterverdiene like bra når prosessen er et virkelig system, der prosessen har tidsforsinkelser og kanskje mer forskjell fra modell enn det prosessmodellen har. Ingen av modellene i oppgaven inneholder tidsforsinkelse, noe som ville ført til mer unøyaktighet mellom modell og prosessmodell og gjort det nødvendig med ytterligere tuning av Kalmanfilteret i lukket sløyfe.

I et nytt griseforsøk er det foreslått å gjøre funksjonstest ved å endre pustefrekvens, på samme måte som i oppgaven, for å sjekke at komfort og sikkerhet tilfredsstillende krav ved valgt tuning av MPC og Kalmanfilter. På grunn av tidsforsinkelser og andre uoverenstemmelser mellom modell og prosess ved regulering av en virkelig prosess vil det nok være nødvendig å tune estimator og regulator på nytt for å ivareta komfort og sikkerhet. Det er foreslått å legge til tidsforsinkelser i prosessmodellen for å bedre simulere et virkelig system, før regulering av et virkelig system.

## 9.3 Programvare fra Cybernetica

I oppgaven er oksygenkonsentrasjon hos gris regulert på en måte som ivaretar både komfort og sikkerhet med programvaren fra Cybernetica.

ModelFit er et nyttig og intuitivt program, som egner seg godt for tilpasning av modell til målinger, både med offline og online parameterestimering. Om mye gjøres på forhånd, vil det bare gjenstå å kopiere inn verdier fra målingene til programmet og kjøre offline parameterestimering for nye forsøk. Optimerte parameterverdier og valgte standardavvik må legges inn i Visual Studios uansett hvilket program som brukes for tilpasningen mellom modell og prosess dersom CENIT skal brukes for å regulere prosessen.

CENIT og RealSim er også nyttige verktøy, som egner seg godt til regulering og simulering. Også her kan mye gjøres på forhånd, slik at brukt av programmet under forsøket er enkelt. Ulemper med CENIT og RealSim er at det er litt å sette seg inn i og finne ut av. Programmet er likevel vurdert som godt egnet for regulering av oksygenkonsentrasjon, både hos forsøksgris og hos KOLS-pasienter.

# Kapittel 10

## Videre arbeid

I dette kapittelet presenteres forslag for hva som kan og bør gjøres videre. Det er foreslått hva som bør gjøres før det gjøres et nytt forsøk på gris, hva som bør gjøres i et fremtidig griseforsøk og implementering av automatisk variasjon av pustefrekvensen i modellen for fremtidig regulering av oksygenkonsentrasjon.

### 10.1 Før nytt griseforsøk

#### Tidsforsinkelse og bedre tilpasning

Før det gjøres et nytt forsøk på gris, er det ting som bør gjøres. En virkelig prosess vil inneholde tidsforsinkelser, og det vil derfor være naturlig å legge til tidsforsinkelser i prosessmodellen for å se hva som skjer dersom prosessen har tidsforsinkelser men ikke modellen. Observer hvordan regulatoren håndterer tidsforsinkelser i prosessmodellen og tilpass regulatoren slik at både komfort og sikkerhet er ivaretatt for alle situasjoner.

Parametere kan tunes for å tilpasse modellen til prosessen. Kalmanfilteret vil rette på noe av avviket mellom modell og prosessmodell med forsinkelser. Ved å øke standardavvikene til prosess og parametere, tas det mer hensyn til målinger enn verdier fra modell. Om det likevel ikke skulle fungere vil det være nødvendig å legge til tidsforsinkelser også i modellen.

For å tilpasse modellen bedre til målingene er det også mulig å gjøre  $[\text{CO}_2]_a$  mindre avhengig av  $F_T$  når  $F_T$  er langt i fra 15. Ved å bytte ut  $F_T$  i ligningen for  $[\text{CO}_2]_a$  med en ligning som øker  $F_T$  ved lave  $F_T$  og minker ved høye, som f.eks.  $F_T - \frac{1}{2}(F_T - 15)$ , vil antageligvis avvikene mellom modell og måling bli mindre.

### **Test av robusthet**

Det er også mulig å teste robustheten til regulatoren. Det ble foreslått av Stavdahl å endre på parameterverdiene i modellen for prosessen, som er endret statistisk ut fra en gaussisk fordeling. Ved å øke standardavvikene ville de nye parameterverdiene vært litt lenger i fra forventningsverdien i forhold til tidligere modell. Koden som ble brukt ved endring av parameterverdier, gitt i appendiks A.1, kunne vært brukt videre. Det er lagt inn en default, som gjør at random-operatoren har samme utgangspunkt hver gang.

Kasper Linnestad fra Cybernetica nevnte at i Cybernetica sine programmer var det vanlig å sjekke robustheten til MPCen ved å gjøre diverse endringer i prosessmodellen for å simulere modellavvik. Dette blir gjort ved å legge til støy, bias eller svingninger på enten inngangsverdier, parametere, tilstander eller målinger. Da er det viktig at endringene ikke skrives til OPC slik at CENIT får vite om endringene.

Forslagene fra Stavdahl og Linnestad kan brukes for å danne et bilde av robustheten til MPCen. På den måten vil det være mulig å si noe om hvor mye variasjoner MPCen kan håndtere og likevel ivareta både komfort og sikkerhet. Robustheten bør sjekkes for prosessmodell med tidsforsinkelse og med variasjon i parameterverdier som er realistisk for en ny gris. Ved en lite robust regulator vil det være nødvendig å gjøre endringer som sikrer en god og sikker regulering av ulike griser/prosesser.

### **Reduksjon av kjøretid**

Dersom utregninger med tidsforsinkelse gjør at kjøretiden blir for lang, kan teknikkene beskrevet i 6.1.5 benyttes for å redusere kjøretiden. Det går an å bruke kontrollvariabel evalueringspunkter for å redusere kjøretiden, dersom den er for lang. Parametrisering av MV kan også gjøres annerledes enn i kapittel 6, og tilpasses til prosessen som skal reguleres.

## **10.2 Fremtidige griseforsøk**

### **Nye målemetoder, ny parameterestimering**

Ottermo har fortalt at i et nytt griseforsøk vil måling av  $[\text{CO}_2]_a$  skje med kapnografi, mens  $[\text{O}_2]_a$  vil måles med pulsoksymetri. Begge disse ikke-invasive målemetodene har stor korrelasjon til konsentrasjonen i blodet målt med invasive metoder, som i de tidligere forsøkene. Det vil nok likevel være noen forskjeller fra tidligere, på grunn av blant annet tidsforsinkelser og en ny gris. Offline og online parameterestimering vil dermed bidra til å tilpasse modellen til grisen. Dersom målingene blir gjort med samme intervall som i tidligere forsøk, vil tilpasningen være til prosess uten tidsforsinkelser, fordi tidsforsinkelsene ikke viser i målinger med så store tidssteg mellom hver måling. Dersom regulatoren er veldig robust og estimatoren fungerer tilfredsstillende, vil det kanskje ikke være nødvendig å gjøre parameterestimering for alle griser, men som påpekt av Linnestad kan det være lurt

å gjøre dersom en har muligheten. Med tidsforsinkelser i prosessen må nok Kalmanfilteret tunes på nytt i lukket sløyfe i CENIT.

### Funksjonstest

Ved å regulere oksygenkonsentrasjonen med valgt MPC og variere pustefrekvensen  $F_T$  som i funksjonstesten i 6.3.6 med tilpassede pustefrekvenser til aktuell gris, kan det sjekkes om valgt MPC opprettholder komfort og sikkerhet. For  $[CO_2]_a < \text{grense}$  bør regulatoren holde  $[O_2]_a$  nær  $[O_2]_{a,ref}$  og for  $[CO_2]_a > \text{grense}$  bør  $[O_2]_{in}$  settes til 0,21. Det må gjøres valg om hva som er grensen for et for høyt nivå av  $[CO_2]_a$ , og verdien må implementeres i Visual Studios (eller CENIT).

Det må vurderes om MPCen opprettholder komfort og sikkerhet på en tilfredstillende måte, som beskrevet i avsnitt 1.3. Om ikke alle avvikende scenarier håndteres på en god måte, må regulatoren tunes slik at komfort og sikkerhet er ivaretatt.

## 10.3 Regulering av $[O_2]$ hos mennesker og automatisk variasjon av $F_T$

Ved regulering av oksygenmengden i pustegassen til et menneske, vil ikke pustefrekvensen være styrt manuelt som på en forsøksgris. Pustefrekvensen er en automatisk prosess som en del av det autonome nervesystemet – og vil, som beskrevet i innledningen, være påvirket av kjemoreseptorer i blodet som blant annet  $O_2$ ,  $CO_2$  og surhetsgrad.

For KOLS-pasienter vil mengden oksygen i pustegassen ha stor påvirkning på pustefrekvensen. Pustefrekvensen reguleres generelt av hvor mye  $CO_2$  som finnes i blodet til enhver tid, som beskrevet i innledende kapitler, men for KOLS-pasienter vil ikke sensorene for måling av  $CO_2$  fungere normalt. Leira har skrevet at mange pasienter går rundt med en konstant forhøyet  $[CO_2]$ , som de har vent seg til og kroppen har kompensert for.

For å få inn en automatisk variasjon av pustefrekvensen i modellen, og som en del av systemet, vil det enkleste være å gjøre  $F_T$  om til en måling eller målbar forstyrrelse. Pustefrekvensen kan enkelt måles hos pasienten, og verdien kan regnes ut i modellen. For modellen må da  $F_T$  være gitt av en ligning, som for enkelhetsskyld kan antas å bare være avhengig av  $[O_2]_a$ , fordi pustefrekvensen hos KOLS-pasienter i stor grad avhenger av oksygenkonsentrasjon i pustegassen. Et forslag til et uttrykk for  $F_T$  er:

$$F_T = F_{T,normal} + \text{faktor} \cdot ([O_2]_{a,ref} - [O_2]_a) \quad (10.1)$$

$F_{T,normal}$  vil være en normal pustefrekvens for pasienten. I følge SNL [3] er normal pustefrekvens for et menneske 10 - 18 innpust/min i hvile, og normale verdier for  $[O_2]_a$  vil, i følge Eldhuset [14], ligge på mellom 0,104 - 0,128. Den predikerte verdien for  $[O_2]_a$  bør brukes for utregning av  $F_T$ .

I ligningen er normal pustefrekvens addert med et skalert avvik mellom  $[O_2]_{a,ref}$  og  $[O_2]_a$ . På den måten vil pustefrekvensen øke dersom  $[O_2]_a$  er lavere enn normalverdi og minke dersom  $[O_2]_a$  er høyere enn normalverdi.

### Realistisk $F_T$ og tilpasning til individet

Ved å ta utgangspunkt i verdier for en normal gris [1],  $F_{T,normal} = 20$ ,  $[O_2]_{a,ref} = 0,124$  og velge faktor = 100, vil  $[O_2]_a = 0,08$  gi  $F_T \approx 24$  og  $[O_2]_a = 0,24$  gi  $F_T \approx 8$ . Det er vanskelig å si om verdiene er helt representative for et virkelig system, men det gir en endring i  $F_T$  som ikke er helt urealistisk. Reguleringsystemet vil også ha en estimator som vil tilpasse modellen til målingene. For forsøksgrisen og for KOLS-pasienter må nok verdiene tilpasses – og det kan, om ønskelig, tas hensyn til flere faktorer.

Om uttrykket som er foreslått gir en naturlig variasjon i pustefrekvens som stemmer overens med virkeligheten, kan testes ved å implementere uttrykket i modellen i Visual Studios og gjøre simuleringer i RealSim og CENIT ved å endre  $[O_2]_{in}$  manuelt. Ved høyere verdier av  $[O_2]_{in}$  bør pustefrekvensen senkes og ved lavere verdier bør pustefrekvensen øke.

For grisen på forsøksbenken vil nok kanskje ikke pulsen ha så mye å si for  $F_T$ , men for et menneske som plutselig reiser seg opp for å gå vil pulsen øke, som gir lavere  $[O_2]_a$  og høyere  $[CO_2]_a$ . Dette er en av faktorene det hadde gått an å tatt hensyn til i ligning (10.1). Det enkle er ofte det beste, så om (10.1) eller en lignende ligning gjør nytten, vil nok det være det enkleste og beste.

En ligning vil aldri kunne gjengi virkeligheten perfekt. Det bør undersøkes om uttrykket, der  $F_T$  bare er funksjon av  $[O_2]_a$ , er tilstrekkelig og gir en god nok estimering av  $F_T$  eller om det må bli tatt hensyn til flere faktorer, f.eks. puls og  $[CO_2]_a$ . Det kan gjøres med simuleringstesten allerede beskrevet eller det kan sjekkes ved å gi KOLS-pasienten ulike konsentrasjoner av oksygen i pustegassen og måle pustefrekvens og oksygeninnhold i blodet. Tilpasningen av ligningen til målingene kan gjøres med ModelFit. Kalmanfilteret vil også bidra til å tilpasse modellen til målingene.

Eldhuset [14] skriver: «Ved å inkludere persondata kan eventuelt fremtidig integrert justering av tilført oksygen reguleres basert på medisinske normalverdier som er tilpasset individet. Behovet til en gjennomsnittlig person vil ikke representere en god referanseverdi for hver pasient, gitt de store forskjellene de ulike sykdomsgruppene og individene presenterer.»

### Måling eller forstyrrelse

Ved å velge  $F_T$  som en måling, kan det settes begrensninger også på  $F_T$  og på den måten ivareta komfort og sikkerhet. Om en vet at en pustefrekvens på 10 gir farlig mengde  $CO_2$  i blodet uansett aktivitetsnivå, og at det er ønskelig å holde pustefrekvensen under 15 for å gjøre det behagelig for pasienten - så kan også overskridelse av disse grensene straffes gjennom  $r_{F_T}$ . Dette vil imidlertid gi flere variabler å tune og «farlig eller ubehagelig pustefrekvens» vil nok variere fra person til person og for ulike situasjoner, som beskrevet

av Eldhuset [14].

$F_T$  kan også legges inn som en målbar forstyrrelse. I modellen må da ligningen for  $F_T$  legges inn slik som f.eks. (10.1).  $[O_2]_a$  er en måling, men ved å legge den til i ligningen for  $F_T$  blir den også en inngangsverdi. Det blir ikke helt konsistent å ha  $y_2 = [O_2]_a$  som både en måling og en inngangsverdi. Det beste vil nok derfor være å bruke den predikterte  $y$ -verdien for utregning av  $F_T$ , og definere denne som en ny  $u$ ,  $u_4$ . Dette er imidlertid ikke testet, og kan i verste fall føre til en loop.

### **Pasientbestemt komfort**

Et innspill fra Leira var at det kanskje burde være mulig for pasienten å initiere en økning av  $[O_2]_{in}$  på egenhånd, slik at om pasienten f.eks. er på vei opp en trapp så slipper pasienten å vente på at  $O_2$ -innhold i blodet blir lavt eller at pulsen stiger før det utløser et tilskudd. På den måten kan komforten også ivaretas av pasienten selv og bidra til å unngå situasjoner der  $[O_2]_a$  synker under behagelig nivå for pasienten. Ved å lage en bryter for ekstra oksygen i pustegassen som pasienten kan styre selv, vil pasienten kunne hjelpe regulatoren med å ivareta komforten.





# Bibliografi

- [1] J. P. Hannon, C. A. Bossone, and C. E. Wade, "Normal physiological values for conscious pigs used in biomedical research," *Letterman Army Institute of Research Presidio of San Francisco California*, May 1989.
- [2] E. Malde, "Regulering av oksygenmetning hos KOLS-pasienter," Prosjektoppgave, NTNU, Desember 2018.
- [3] O. H. Skjønberg. respirasjonsfrekvens. [Online]. Available: <https://sml.snl.no/respirasjonsfrekvens>
- [4] M. Kvam. Lungene. [Online]. Available: <https://nhi.no/kroppen-var/organer/lungene/>
- [5] A. Hauge. Respirasjon. [Online]. Available: <https://sml.snl.no/respirasjon>
- [6] Chronic obstructive pulmonary disease (copd). [Online]. Available: <https://www.who.int/respiratory/copd/en/>
- [7] P. Holck. Bronkier. [Online]. Available: <https://sml.snl.no/bronkier>
- [8] M. A. Andersen, H. Leira, S. A. Mausest, L. Randeberg, B. Skallerud, and Ø. Stavadahl, "Oxyaid," en søknad til Forny.
- [9] J. Kofstad, *Blodgasser, elektrolytter og hemoglobin - Metode og klinikk*. Tano Aschehoug, 1995.
- [10] B. A. Waaler and A. Hauge. Åndedrett. [Online]. Available: <https://snl.no/åndedrett>
- [11] P. Holck. lungene. [Online]. Available: <https://sml.snl.no/lungene>
- [12] S. W. Jacob, C. A. Francone, W. J. Lassow, P. Hurlen, and E. Dietrichs, *Anatomi og fysiologi*, 2nd ed. Universitetsforlaget, 1989.
- [13] Hemoglobin. [Online]. Available: <https://no.wikipedia.org/wiki/Hemoglobin>
- [14] A. Eldhuset, "Dynamisk modell av ph, o2-og co2-metning i blod," Master's thesis,

- 
- NTNU, 2016. [Online]. Available: [https://brage.bibsys.no/xmlui/bitstream/handle/11250/2433776/14393\\_FULLTEXT.pdf?sequence=1&isAllowed=y](https://brage.bibsys.no/xmlui/bitstream/handle/11250/2433776/14393_FULLTEXT.pdf?sequence=1&isAllowed=y)
- [15] H. T. Milhorn, J. R. Benton, R. Ross, and A. C. Guyton, "A mathematical model of the human respiratory control system," *Biophysical Journal* 5, 1965.
- [16] M. V. Ottermo, "Oxyaid simple model maria," 2018, en forenklet versjon av Eldhuset sin dynamiske modell.
- [17] Cybernetica. Company. [Online]. Available: <http://cybernetica.no/company/>
- [18] J. G. Dyrset, K. Linnestad, and P. Singstad, "Introduction to cybernetica apc technology," 2019, en powerpoint fra kurs hos Cybernetica februar 2019.
- [19] K. Linnestad, "Model predictive control of a polyolefin reactor," Masteroppgave, NTNU, Juni 2015. [Online]. Available: [https://brage.bibsys.no/xmlui/bitstream/handle/11250/2351686/13218\\_FULLTEXT.pdf?sequence=1&isAllowed=y](https://brage.bibsys.no/xmlui/bitstream/handle/11250/2351686/13218_FULLTEXT.pdf?sequence=1&isAllowed=y)
- [20] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed., ser. Springer Series in Operations Research and Financial Engineering. New York: Springer-Verlag New York, 2006. [Online]. Available: [http://www.bioinfo.org.cn/~wangchao/maai/Numerical\\_Optimization.pdf](http://www.bioinfo.org.cn/~wangchao/maai/Numerical_Optimization.pdf)
- [21] M. Ulusoy. Understanding kalman filters, part 2: State observer. [Online]. Available: <https://se.mathworks.com/videos/understanding-kalman-filters-part-2-state-observers-1487694734527.html>
- [22] ——. Understanding kalman filters, part 3: An optimal state estimator. [Online]. Available: <https://se.mathworks.com/videos/understanding-kalman-filters-part-3-optimal-state-estimator--1490710645421.html>
- [23] ——. Understanding kalman filters, part 4: An optimal state estimator algorithm. [Online]. Available: <https://se.mathworks.com/videos/understanding-kalman-filters-part-4-optimal-state-estimator-algorithm--1493129749201.html>
- [24] M. Nørgaard, N. K. Poulsen, and O. Ravn, "New developments in state estimation for nonlinear systems," *Automatica*, vol. 36, no. 11, pp. 1627–1638, 2000. [Online]. Available: [https://www.lsv.uni-saarland.de/fileadmin/teaching/seminars/SL\\_Seminar/New\\_developments\\_in\\_state\\_estimation\\_for\\_nonlinear\\_systems.pdf](https://www.lsv.uni-saarland.de/fileadmin/teaching/seminars/SL_Seminar/New_developments_in_state_estimation_for_nonlinear_systems.pdf)
- [25] M. Ulusoy. Understanding kalman filters, part 5: Nonlinear state estimators. [Online]. Available: <https://se.mathworks.com/videos/understanding-kalman-filters-part-5-nonlinear-state-estimators-1495052905460.html>
- [26] D. Simon, *Optimal State Estimation, Kalman, H, and Nonlinear Approaches*, 1st ed., Hoboken, Ed. New Jersey: John Wiley Sons, 2006.
- [27] S. Kolås, B. Foss, and T. Schei, "Noise modeling concepts in nonlinear state estimation," *Journal of Process Control*, vol. 19, no. 7, pp. 1111–1125, 2009.

- 
- [28] S. Kolås, B. A. Foss, and T. S. Schei, "Constrained nonlinear state estimation based on the UFK approach," *Computers Chemical Engineering*, vol. 33, no. 8, pp. 1386–1401, August 2009. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0098135409000209?via%3Dihub>
- [29] T. Schei, "A finite-difference method for linearization in nonlinear estimation algorithms," *Journal of Process Control*, vol. 18, no. 9, pp. 821–828, Oktober 2008.
- [30] B. Foss and T. Heirung, "Merging optimization and control," NTNU, Tech. Rep., 2016.
- [31] D. Seborg, T. Edgar, and D. Mellichamp, *Process Dynamics and Control*, 2, Ed. John Wiley Sons, Inc, 2004.
- [32] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. P. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.
- [33] E. F. Camacho and C. Bordons, "Nonlinear model predictive control," *Model Predictive control. Advanced Textbooks in Control and Signal Processing*, 2007.
- [34] E. Camacho and C. Bordons, *Nonlinear Model Predictive Control: An Introductory Review*, R. Findeisen, F. Allgöwer, and L. Biegler, Eds. Springer, Berlin, Heidelberg, 2007, vol. 358. [Online]. Available: [https://link.springer.com/chapter/10.1007/978-3-540-72699-9\\_1](https://link.springer.com/chapter/10.1007/978-3-540-72699-9_1)
- [35] S. Strand and J. Sagli, "MPC in Statoil - Advantages with In-House Technology," *Elsvier*, vol. 37, no. 1, pp. 197–103, 2004.
- [36] Hvordan tegne en gris vakkert, og viktigst, troverdig. [Online]. Available: <https://no.ajeshashok.com/iskusstvo-i-razvlecheniya/55219-kak-narisovat-svinyu-krasivo-a-glavnoe-pravdopodobno.html>

---

---

# Appendiks

## A MATLAB-kode

Verdiene for standardavvik er funnet i [1].

### A.1 Generering av nye parameterverdier

```
1 %% Nye parameterverdier for modell til prosess
2
3 %Forventningsverdi (mu) = tidligere parameterverdier
4 mu_MBCO2 = 0.3231;
5 mu_MBO2 = 0.238;
6 mu_D = 1.4083;
7 mu_SV = 0.05;
8 mu_VB = 1126;
9 mu_VAa = 4;
10
11 %Standardavvik (sigma) for 30kg tung gris
12 sigma_D = 0.067;
13 sigma_MBCO2 = 0.0375;
14 sigma_MBO2 = 0.0381;
15 sigma_SV = 0.0077;
16 sigma_VB = 100;      % Gjettet verdi
17 sigma_VAa = 1;      % Gjettet verdi
18
19 %Nye parameterverdier = tilfeldige ,gaussisk fordelte
    verdier
20 rng('default')      % For reproduserbarhet
21 MBCO2 = normrnd(mu_MBCO2, sigma_MBCO2);
22 MBO2 = normrnd(mu_MBO2, sigma_MBO2);
23 D = normrnd(mu_D, sigma_D);
24 SV = normrnd(mu_SV, sigma_SV);
25 VB = normrnd(mu_VB, sigma_VB);
26 VAa = normrnd(mu_VAa, sigma_VAa);
```

---

## B Implementering av Ottermo sin modell

Ved implementering av Ottermo sin modell i Visual Studio via Cybernetica sin template, var det nødvendig å definere modellen med antall parametere, konstanter, inngangsverdier, tilstander og målinger, som beskrevet i avsnitt 3.2.

Modellen til Ottermo består av:

$$\begin{aligned}x &- 4 \\y &- 2 \\z &- 2 \\u &- 3 \\par &- 4 \\con &- 2\end{aligned}$$

hvor

$$x = \begin{bmatrix} [CO_2]_B \\ [O_2]_B \\ V_{eff} \\ [O_2]_{Aa} \end{bmatrix}$$

$$y = z = \begin{bmatrix} [CO_2]_a \\ [O_2]_a \end{bmatrix}$$

$$u = \begin{bmatrix} [O_2]_{in} \\ F_T \\ F_{puls} \end{bmatrix}$$

$$par = \begin{bmatrix} D \\ MBCO_2 \\ MBO_2 \\ SV \end{bmatrix}$$

$$con = \begin{bmatrix} V_B \\ V_{Aa} \end{bmatrix}$$

y er målinger og z er kontrollvariablene (CV),  $u_1 = [O_2]_{in}$  er den manipulerbare inngangsvariabelen (MV), mens  $u_2 = F_T$  og  $u_3 = F_{puls}$  er sett på som forstyrrelser (DV).

---

## C Data fra griseforsøk

OxyAid har gjort forsøk på gris. Det første ble gjennomført 17.10.2017 på en 28,3kg tung gris. Data fra dette forsøket er gitt i C.2 og omtales i teksten som griseforsøk B. Det andre forsøket ble gjennomført 29.05.2018 på en 35kg tung gris. Data fra dette forsøket er gitt i C.1, og omtales i teksten som griseforsøk A.

Omregning fra partialtrykk til konsentrasjon er gjort ved å dividere partialtrykket med atmosfæretrykket på 101,3kPa [15]. I de åpne rutene er det regnet ut en gjennomsnittsverdi mellom målingen før og etter, som er brukt i simuleringene.

### C.1 Data fra griseforsøk A

<i>Sample#</i>	<i>O2(%)</i>	<i>Pustefrekvens</i>	<i>Tidalvol(ml)</i>	<i>pCO2(kPA)</i>	<i>pO2(kPA)</i>
1	40	21	0.375	5.94	24.2
2	40	21	0.375	6.08	24.6
Step	30	21	0.375	6.03	
3	30	21	0.375	5.98	21.5
4	30	21	0.375	6.15	22.8
5	30	21	0.375	6.18	23.1
6	30	21	0.375	6.13	26.6
7	30	21	0.375	6.18	17.10
8	30	21	0.375	6.34	21.4
9	30	21	0.375	6.28	23.2
10	30	21	0.375	6.29	16.2
11	30	21	0.375	6.53	19.6
12	30	21	0.375	6.32	16.1
13	30	21	0.375	6.3	19.39
14	30	21	0.375	6.27	18.5
Step	21	21	0.375	6.21	
15	21	21	0.375	6.14	13.9
16	21	21	0.375	6.28	15.3
17	21	21	0.375	6.14	14.4
18	21	21	0.375	6.25	13.2
19	21	21	0.375	6.25	10.3
20	21	21	0.375	6.19	12
21	21	21	0.375	6.2	9.83



---

22	21	21	0.375	6.23	11.4
23	21	21	0.375	6.34	11.9
24	21	21	0.375	6.18	9.33
25	21	21	0.375	6.24	11.4
26	21	21	0.375	6.34	12
Step	40	21	0.375	6.25	
27	40	21	0.375	6.16	14
28	40	21	0.375	6.17	18.39
29	40	21	0.375	5.94	20.5
30	40	21	0.375	6.07	21.3
31	40	21	0.375	6.02	21.6
32	40	21	0.375	6.06	23.2
33	40	21	0.375	6	23.5
34	40	21	0.375	6.07	23.8
35	40	21	0.375	6.03	24
36	40	21	0.375	6	23.6
37	40	21	0.375	5.95	23.5
38	40	21	0.375	6.02	24
39	40	21	0.375	6.12	24
Step	30	21	0.375	6.02	
40	30	21	0.375	5.92	24.1
41	30	21	0.375	6.01	22.7
42	30	21	0.375	5.96	18.3
43	30	21	0.375	6.08	19.6
44	30	21	0.375	5.88	22
45	30	21	0.375	5.92	17
46	30	21	0.375	6.16	19.39
47	30	21	0.375	5.96	16.5
48	30	21	0.375	5.73	20.3
49	30	21	0.375	6.07	24.1
50	30	21	0.375	5.97	16.39
51	30	21	0.375	5.94	22.8
Step	21	21	0.375	5.9	
52	21	21	0.375	5.85	14
53	21	21	0.375	5.92	15.7
54	21	21	0.375	5.8	15.2
55	21	21	0.375	5.85	14.5
56	21	21	0.375	5.96	10.6
57	21	21	0.375	6	13.3
58	21	21	0.375	5.88	9.94
59	21	21	0.375	6	12.3
60	21	21	0.375	6.05	12.4
61	21	21	0.375	5.76	9.779
62	21	21	0.375	6.06	12.8

---

---

63	21	21	0.375	5.81	9.63
Step	40	21	0.375	5.95	
64	40	21	0.375	6.08	20.9
65	40	21	0.375	5.99	20.9
66	40	21	0.375	5.77	22.9
67	40	21	0.375	5.78	22.9
68	40	21	0.375	5.86	24.9
69	40	21	0.375	5.97	28.6
70	40	21	0.375	5.9	23.3
71	40	21	0.375	5.92	23.2
72	40	21	0.375	5.83	22.9
73	40	21	0.375	5.8	23.1
74	40	21	0.375	5.86	23.5
75	40	21	0.375	5.83	23.3
76	40	21	0.375	5.85	22.8
Step	40	15	0.375	6.1	
77	40	15	0.375	6.34	21.2
78	40	15	0.375	6.58	22.8
79	40	15	0.375	6.6	27.5
80	40	15	0.375	6.76	25.2
81	40	15	0.375	6.86	27.3
82	40	15	0.375	6.89	21.4
83	40	15	0.375	7.04	25.9
84	40	15	0.375	7.15	21.3
85	40	15	0.375	7.18	26.7
86	40	15	0.375	7.23	21.3
87	40	15	0.375	7.31	25.6
88	40	15	0.375	7.34	23.1
Step	40	10	0.375	7.56	
89	40	10	0.375	7.78	18.5
90	40	10	0.375	8.02	18.2
91	40	10	0.375	7.87	19.89
92	40	10	0.375	8.36	21.4
93	40	10	0.375	8.470	18.1
94	40	10	0.375	8.69	18.89
95	40	10	0.375	8.41	19.39
96	40	10	0.375	8.46	20.1
97	40	10	0.375	8.44	20.1
98	40	10	0.375	8.59	21.4
99	40	10	0.375	8.52	21.2
100	40	10	0.375	8.63	21.9
Step	40	21	0.375	8.17	
101	40	21	0.375	7.7	22.8
102	40	21	0.375	7.61	22.5

---

103	40	21	0.375	7.17	22.9
104	40	21	0.375	7.17	22.9
105	40	21	0.375	7.03	22
106	40	21	0.375	6.76	23
107	40	21	0.375	6.85	23.2
108	40	21	0.375	6.67	22.7
109	40	21	0.375	6.76	23.6
110	40	21	0.375	6.69	22.8
111	40	21	0.375	6.5	22.5
112	40	21	0.375	6.64	21
113	40	21	0.375	6.23	23.7
Step	40	15	0.375	6.43	
114	40	15	0.375	6.62	24.4
115	40	15	0.375	6.67	23
116	40	15	0.375	6.87	22.4
117	40	15	0.375	6.83	21.7
118	40	15	0.375	7.27	21.4
119	40	15	0.375	7.02	21.7
120	40	15	0.375	7.39	22.3
121	40	15	0.375	7.21	21.3
122	40	15	0.375	7.41	21.3
123	40	15	0.375	7.42	20
124	40	15	0.375	7.21	21.2
125	40	15	0.375	7.48	21.6
Step	40	10	0.375	7.58	
126	40	10	0.375	7.67	19.5
127	40	10	0.375	7.91	21.7
128	40	10	0.375	8.05	20.6
129	40	10	0.375	8.26	19.89
130	40	10	0.375	8.119	18.89
131	40	10	0.375	8.279	23.2
132	40	10	0.375	8.13	19.1
133	40	10	0.375	8.25	20.39
134	40	10	0.375	8.13	18.89
135	40	10	0.375	8.64	21.5
136	40	10	0.375	8.42	20.39
137	40	10	0.375	8.23	19.1
Step	40	21	0.375	7.97	
138	40	21	0.375	7.72	21.1
139	40	21	0.375	7.44	21.8
140	40	21	0.375	7.14	22
141	40	21	0.375	7.16	24.3
142	40	21	0.375	7.06	24.8
143	40	21	0.375	6.56	21.9
144	40	21	0.375	7.02	22.9

145	40	21	0.375	6.89	23.4
146	40	21	0.375	6.75	22.5
147	40	21	0.375	6.63	24.6
148	40	21	0.375	6.44	22
149	40	21	0.375	6.4	21.5
150	40	21	0.375	6.26	22.3
Step	21	15	0.375	6.43	
151	21	15	0.375	6.6	14.9
152	21	15	0.375	6.97	15.9
153	21	15	0.375	7.09	12.3
154	21	15	0.375	7.11	12.2
155	21	15	0.375	7.04	9.44
156	21	15	0.375	7.25	10.4
157	21	15	0.375	7.3	10.5
158	21	15	0.375	7.15	8.86
159	21	15	0.375	7.45	9.32
160	21	15	0.375	7.16	8.58
161	21	15	0.375	7.46	9.99
162	21	15	0.375	7.31	9.64
Step	21	10	0.375	7.38	
163	21	10	0.375	7.46	8.17
164	21	10	0.375	7.64	8.5
165	21	10	0.375	7.85	8.36
166	21	10	0.375	7.79	8.46
167	21	10	0.375	7.77	7.5
168	21	10	0.375	8.27	8.01
169	21	10	0.375	8.1	7.12
170	21	10	0.375	8.19	8.19
171	21	10	0.375	8.42	8.3
172	21	10	0.375	8.23	8.06
173	21	10	0.375	8.15	8.23
174	21	10	0.375	8.69	8.64
Step	40	21	0.375	8	10.82
175	40	21	0.375	7.67	13
176	40	21	0.375	7.36	20.1
177	40	21	0.375	7.26	17.5
178	40	21	0.375	7.23	22.8
179	40	21	0.375	6.76	23
180	40	21	0.375	6.77	20.8
181	40	21	0.375	6.69	24
182	40	21	0.375	6.5	20.8
183	40	21	0.375	6.49	26.3
184	40	21	0.375	6.64	25.2
185	40	21	0.375	6.49	20.6
186	40	21	0.375	6.48	20.39

---

## C.2 Data fra griseforsøk B

Det første griseforsøket, med en gris på 28,3 kg. Tidalvolum er (antatt) 0,3L i dette forsøket.

<i>Sample#</i>	<i>O2(%)</i>	<i>Pustefrekvens</i>	<i>pCO2(kPA)</i>	<i>pO2(kPA)</i>
1	40,00	24,00	5,34	25,90
Step	40,00	16,00	5,34	25,90
2	40,00	16,00	6,10	23,90
3	40,00	16,00	6,43	22,90
4	40,00	16,00	6,73	24,50
5	40,00	16,00	6,71	24,60
6	40,00	16,00	6,82	22,80
7	40,00	16,00	6,83	23,80
8	40,00	16,00	7,05	23,20
9	40,00	16,00	6,30	24,60
10	40,00	16,00	7,01	23,40
11	40,00	16,00	6,97	24,00
Step	40,00	24,00	6,97	24,00
12	40,00	24,00	5,74	28,00
Step	21,00	24,00	5,66	22,60
13	21,00	24,00	5,58	17,20
14	21,00	24,00	5,66	12,90
15	21,00	24,00	5,63	13,10
16	21,00	24,00	5,82	14,50
17	21,00	24,00	5,77	12,00
Step	40,00	24,00	5,77	12,00
18	40,00	24,00	5,70	20,50
19	40,00	24,00	5,60	23,60
20	40,00	24,00	5,50	25,60
Step	40,00	12,00	5,50	25,60
21	40,00	12,00	6,64	25,50
22	40,00	12,00	6,72	22,30
23	40,00	12,00	7,07	23,00
24	40,00	12,00	7,12	23,70
25	40,00	12,00	7,08	23,90
26	40,00	12,00	7,59	26,00
27	40,00	12,00	7,59	23,20
Step	40,00	24,00	7,59	23,20
28	40,00	24,00	6,21	26,70
29	40,00	24,00	5,93	25,60
30	40,00	24,00	5,85	25,80
31	40,00	24,00	5,94	26,40
Temp	40,00	24,00	5,69	25,70
32	40,00	24,00	5,43	25,00

Step	30,00	24,00	5,43	25,00
33	30,00	24,00	5,76	25,90
34	30,00	24,00	5,43	18,90
35	30,00	24,00	5,94	24,60
36	30,00	24,00	5,72	22,30
37	30,00	24,00	5,42	17,70
38	30,00	24,00	5,58	20,70
39	30,00	24,00	5,51	19,50
Step	40,00	24,00	5,51	19,50
40	40,00	24,00	5,52	22,30
41	40,00	24,00	5,41	23,40
42	40,00	24,00	5,47	24,50
43	40,00	24,00	5,30	25,00
Step	40,00	16,00	5,30	25,00
44	40,00	16,00	6,06	22,90
45	40,00	16,00	6,23	24,80
46	40,00	16,00	6,31	23,30
47	40,00	16,00	6,72	26,80
48	40,00	16,00	6,48	23,20
Step	40,00	24,00	6,48	23,20
49	40,00	24,00	5,94	25,50
50	40,00	24,00	5,53	25,20
51	40,00	24,00	5,46	24,60
Step	21,00	24,00	5,46	24,60
52	21,00	24,00	5,60	17,40
53	21,00	24,00	5,54	15,40
54	21,00	24,00	5,35	12,20
55	21,00	24,00	5,74	11,90
56	21,00	24,00	5,40	11,00
57	21,00	24,00	5,52	10,90
Step	40,00	24,00	5,52	10,90
58	40,00	24,00	5,34	19,00
59	40,00	24,00	5,32	21,90
60	40,00	24,00	5,28	24,80
Step	40,00	16,00	5,28	24,80
61	40,00	16,00	5,88	22,50
62	40,00	16,00	6,76	27,60
63	40,00	16,00	6,87	27,20
64	40,00	16,00	6,36	23,20
65	40,00	16,00	7,06	27,40
66	40,00	16,00	6,64	25,10
Step	40,00	12,00	6,64	25,10
67	40,00	12,00	6,54	23,10
68	40,00	12,00	7,16	27,60
70	40,00	12,00	7,05	21,10

---

71	40,00	12,00	7,43	23,30
72	40,00	12,00	7,56	22,90
73	40,00	12,00	7,41	21,40
74	40,00	12,00	7,42	23,30
75	40,00	12,00	7,44	21,50
Step	40,00	24,00	7,44	21,50
76	40,00	24,00	6,29	24,30
77	40,00	24,00	6,13	24,70
78	40,00	24,00	6,07	28,10
Step	30,00	16,00	6,07	28,10
79	30,00	16,00	6,91	20,70
80	30,00	16,00	6,37	16,00
81	30,00	16,00	7,13	18,90
82	30,00	16,00	3,31	19,00
83	30,00	16,00	6,65	14,70
84	30,00	16,00	7,25	18,40
85	30,00	16,00	6,68	14,80
Step	21,00	16,00	6,68	14,80
86	21,00	16,00	7,35	12,50
87	21,00	16,00	6,97	11,75
88	21,00	16,00	7,03	11,00
89	21,00	16,00	6,97	9,96
90	21,00	16,00	6,86	9,41
91	21,00	16,00	6,90	8,35
Step	40,00	24,00	6,90	8,35
92	40,00	24,00	6,19	17,30
93	40,00	24,00	5,90	20,30
94	40,00	24,00	5,79	22,00
95	40,00	24,00	5,63	22,8

---

## D Tabell med aktuelle verdier fra [1]

Utvalgte verdier fra artikkel om gris [1] er gjengitt i tabellen under.

Utvalgte verdier fra [1]				
Measurement	N	Mean	S.D	Range
O <sub>2</sub> consumption (ml/min/kg)	22	6.6	1.27	4.4 - 9.2
CO <sub>2</sub> production (ml/min/kg)	22	5.5	1.25	3.4 - 7.6
Respiratory exchange ratio	22	0.81	0.067	0.67 - 0.96
Ventilatory rate (breaths/min)	22	20	2.9	16 - 25
Stroke volume (ml/beat/kg)	15	1.34	0.258	1.03 - 1.87
Heart rate (beats/min)	15	105	10.6	90 - 107

**Tabell D.1:** Tabellen viser utvalgte, normal-verdier for gris, gitt i artikkel [1]. Dette er verdier for friske griser. «S.D» er standardavvik og «Mean» er gjennomsnittsverdien. Diffusjonskoeffisienten D er inspirert av «Respiratory exchange ratio». Pustefrekvensen  $F_T$  er inspirert av «Ventilatory rate».

## E Vektmatrisene Q og S i CENIT

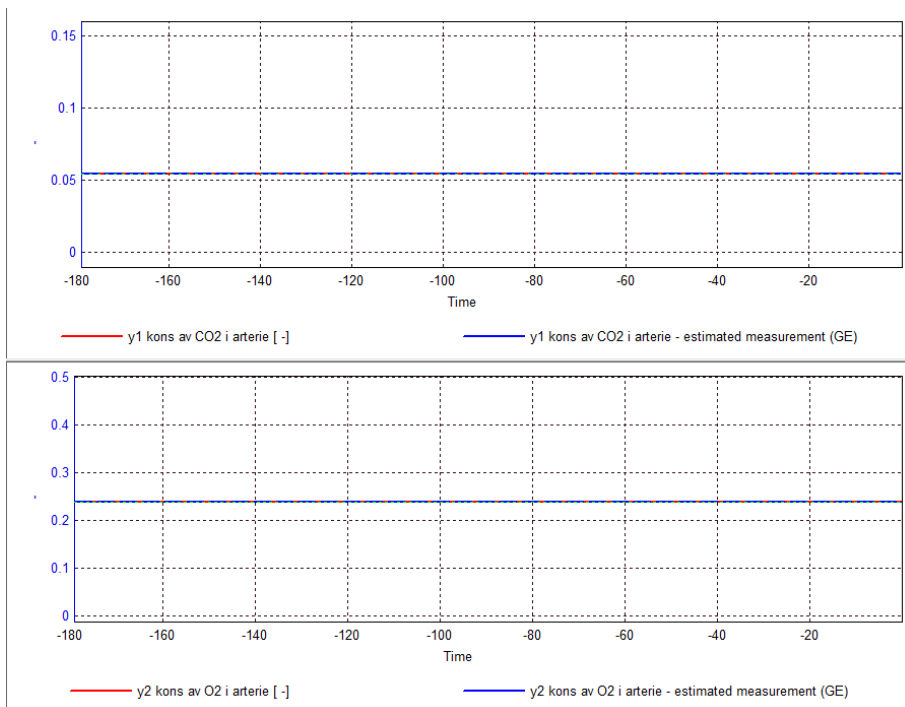
I CENIT er **Q** og **S** definert som i ligning (6.8). «spenn» er definert som akseptabel endring i forhold til settpunktet. Matrisene, slik de er definert i CENIT, er gitt under.

$$\mathbf{Q} = \begin{bmatrix} \frac{0,045}{0,3^2} & 0 & 0 \\ 0 & \frac{0,144}{0,12^2} & 0 \\ 0 & 0 & \frac{1,225}{0,35^2} \end{bmatrix}, \quad \mathbf{S} = \begin{bmatrix} \frac{0,4225}{0,65^2} & 0 & 0 \\ 0 & \frac{225}{15^2} & 0 \\ 0 & 0 & \frac{22500}{150^2} \end{bmatrix} \quad (10.2)$$

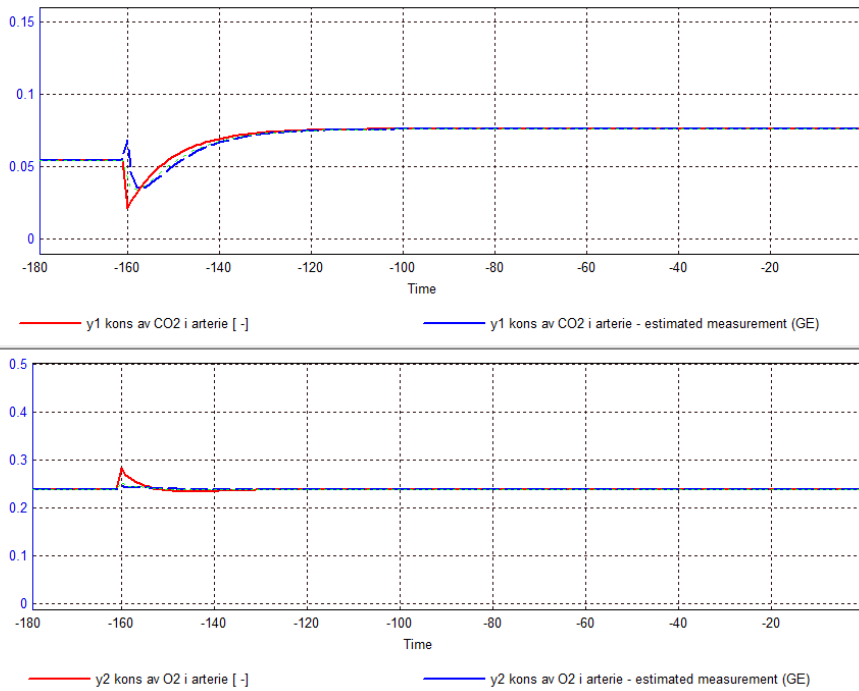
## F Målte og estimerte utgangsverdier

Med valgte standardavvik, gitt i tabell 5.1 og 5.2, fungerer Kalmanfilteret ganske bra. Figur F.1 viser målt og estimert utgangsverdi ved  $F_T = 21$ . I figur F.2-F.4 er  $F_T$  endret til henholdsvis 15, 14 og 10.

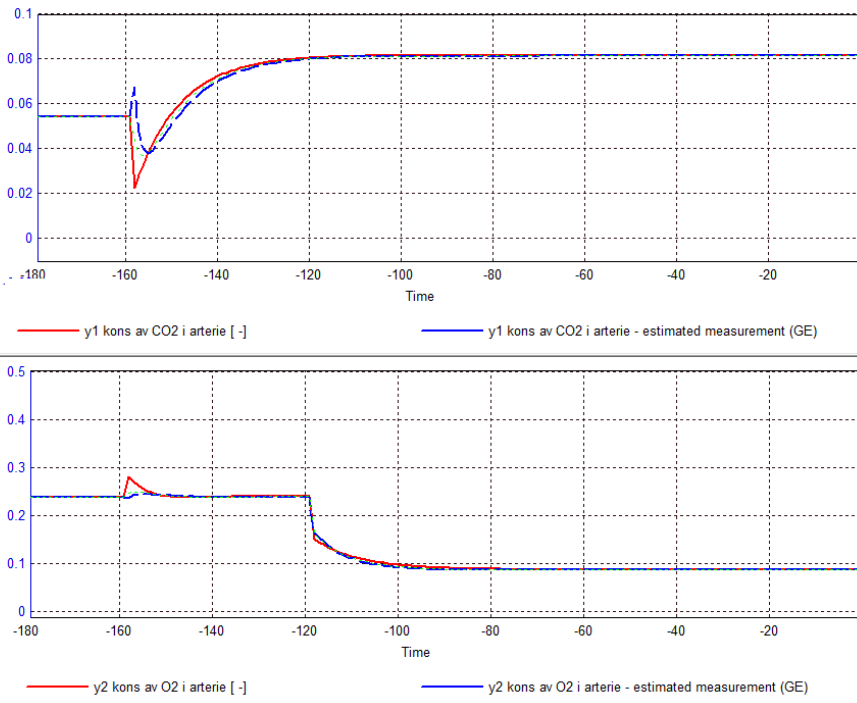




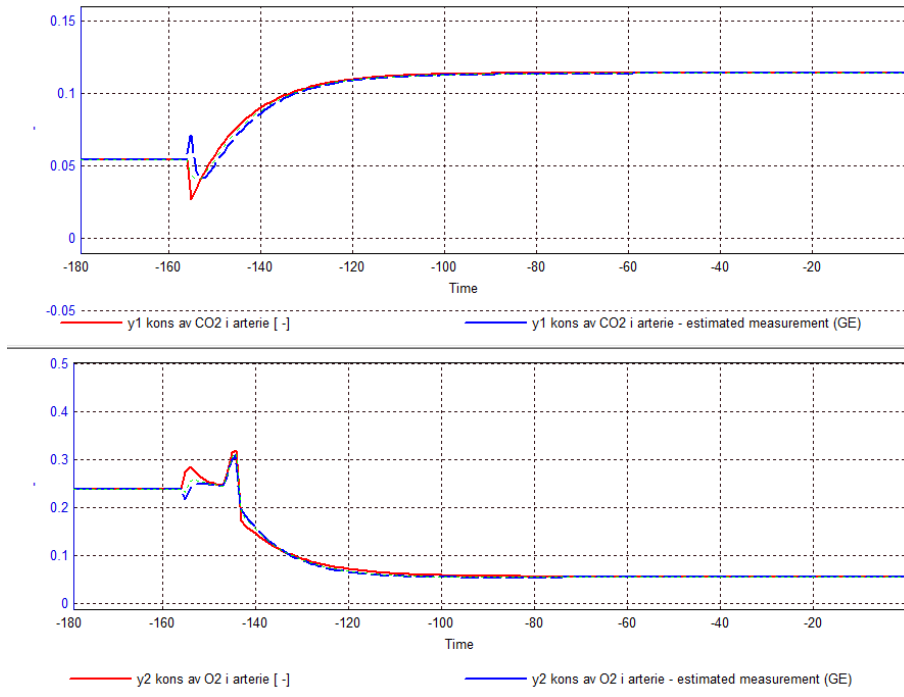
**Figur F.1:** I denne figuren viser plot av  $y_1 = [\text{CO}_2]_a$  og  $y_2 = [\text{O}_2]_a$ , både «målte» utgangsverdier fra prosessmodell og estimerte utgangsverdier, når  $F_T$  settes til 21. Estimerte og målte utgangsverdier er veldig like.



**Figur F.2:** I denne figuren viser plot av  $y_1 = [\text{CO}_2]_a$  og  $y_2 = [\text{O}_2]_a$ , både «målte» utgangsverdier (rød) fra prosessmodell og estimerte utgangsverdier (blå), når  $F_T$  settes til 15. Estimerte og målte utgangsverdier er veldig like. For endring av verdier må RealSim resettes. Resettingen gjør at også  $[\text{O}_2]_{in}$  resettes, og fører til større endringer i overgangen enn nødvendig. En glattere overgang kunne vært oppnådd ved å programmere inn endringene i pustefrekvens  $F_T$ .



**Figur F.3:** I denne figuren viser plot av  $y_1 = [\text{CO}_2]_a$  og  $y_2 = [\text{O}_2]_a$ , både «målte» utgangsverdier (rød) fra prosessmodell og estimerte utgangsverdier (blå), når  $F_T$  settes til 14. Estimerte og målte utgangsverdier er veldig like. For endring av verdier må RealSim resettes. Resettingen gjør at også  $[\text{O}_2]_{in}$  resettes, og fører til større endringer i overgangen enn nødvendig. En glattere overgang kunne vært oppnådd ved å programmere inn endringene i pustefrekvens  $F_T$ .



**Figur F.4:** I denne figuren viser plot av  $y_1 = [\text{CO}_2]_a$  og  $y_2 = [\text{O}_2]_a$ , både «målte» utgangsverdier fra prosessmodell og estimerte utgangsverdier, når  $F_T$  settes til 10. Estimerte og målte utgangsverdier er veldig like. For endring av verdier må RealSim resettes. Resettingen gjør at også  $[\text{O}_2]_{in}$  resettes, og fører til større endringer i overgangen enn nødvendig. En glattere overgang kunne vært oppnådd ved å programmere inn endringene i pustefrekvens  $F_T$ .

