

Modellbasert styring av mikrobølge tinetunnel for frosne kjøttblokker

Erlend Brandvik

Master i teknisk kybernetikk
Oppgaven levert: Juni 2009
Hovedveileder: Steinar Sælid, ITK

Oppgavetekst

Modellbasert styring av mikrobølge tinetunnel for frosne kjøttblokker.

Industriell tining av frosne kjøttblokker kan foregå i en mikrobølge tinetunnel hvor blokkene passerer gjennom styrte mikrobølgefelter. Effekten fra mikrobølgeelementene i tunnelen kan styres automatisk. Denne oppgaven omhandler metoder for automatisk, modellbasert styring av en tinetunnel.

1. Beskriv tineprosessen for tining av frosset kjøtt i tinetunnel og oppsummer problemstillingen rundt automatisk styring av denne. Beskriv spesielt mikrobølgeovnen MIP12 fra Ferrite.
2. Søk i litteraturen etter modeller og fysikalske forhold som beskriver tineprosessen, inklusive dielektrisitetsforhold, varmekapasitet, termisk konduktivitet. Summer opp dette og velg en passende modell for videre bruk i oppgaven.
3. Lag en modell som beskriver totalsystemet med tunnel, mikrobølgepådrag og blokktransport. Realiser modellen i Matlab og foreta simuleringer med valgte/tilpassede parametre. Tilpasning av parametrene baseres på erfaringer fra operasjon av virkelige tinetunneler. Fokuser spesielt på å redusere modellens dimensjon og kompleksitet, slik at denne kan benyttes for online styring.
4. Anta at blokkene har lik dimensjon, men at de kan ankomme tunnelen med ujevne mellomrom. Anta også at fettinnhold for hver blokk er kjent på forhånd. Vurder bruk av modellbasert styring av denne prosessen. Det er få eller ingen målinger som kan benyttes for tilbakekoplet styring, så en skal spesielt se på bruk av åpen sløyfe styring; dvs. at en modell benyttes for å optimalisere mikrobølge pådragene, gitt kjent fettinnhold og ankomsttid for blokker inn i tunnelen. Programmer og test ut et slikt optimaliserende system i Matlab.
5. Vurder mulighetene for tilbakekoplet styring fra målte temperaturer og/eller fra målinger relatert til mikrobølgeelementene. Beskriv hvordan slike målinger kunne benyttes for bruk i en ulineær MPC regulator. Hvis tiden tillater det skal også dette teste vha simulering i Matlab.

Oppgaven gitt: 12. januar 2009
Hovedveileder: Steinar Sælid, ITK

Sammendrag

En matematisk modell for oppvarming av frosne kjøttblokker ved bruk av en mikrobølge tine-tunnel er utviklet i denne oppgaven. Den matematiske modellen stemmer ikke helt overens med virkeligheten, og det er mange faktorer og parametre som tilpasses for at modellen skal konvergere mot det som skjer ved en virkelig tineprosess. Stoffet som presenteres her gir likevel et godt overblikk over prinsippene og varmedelingen ved oppvarming av frosne kjøttblokker. Resultatene er tilstrekkelige til å kunne utvikle et optimaliserende system for automatisk styring av tinetunnelen. Dette er også hovedmålet med denne oppgaven.

De viktigste resultatene er at fysiske egenskaper, som varmekapasitet og termisk konduktivitet vil påvirke tineprosessen i stor grad. Fettinnholdet i kjøttet har størst betydning på egenskapene. Det vises at egenskapene til kjøttet endres mest i temperaturer rundt frysepunktet til kjøttet. Frysepunktet er funnet til å være rundt -1.5°C . Under tilpassingen av parametrene er det i hovedsak penetrasjonsdybden, varmekapasiteten og konduktiviteten som endres. Det blir benyttet en tinetunnel bestående av to doble mikrobølgeelementer. Effekten fra mikrobølgeelementene bør kontrolleres for å oppnå optimal tining for hver enkelt kjøttblokk. Siden sammensetningen til kjøttblokkene er forskjellige, må effekten til mikrobølgeelementene tilpasses hver enkelt blokk. Tre metoder for optimal styring analyseres og det viser seg at alle tre kan benyttes til forskjellige formål. Alle metodene er modellbaserte så det fokuseres på åpen sløyfe styring. Metoden med likhetsbetingelser gjør at kjøttblokkene når ønsket temperatur, men her bør kjøttblokkene være veldig like og ikke variere i fettprosent. Metoden med ulikhetsbetingelser eller objektfunksjon kan gjøre at temperaturen kan bli for høy, men her er det større rom for variasjon i blokkene. Metoden med bruk av likhetsbetingelser benytter mest pådrag og vil være minst effektiv med tanke på kg per tid. De to andre metodene er mer effektive, men mindre nøyaktige. Det konkluderes med at ved tining av kjøtt bør man tine kjøttblokker med relativt lik fettprosent samtidig, uansett metode. Man bør heller ikke forvente eksakte temperaturer i kjøttblokkene, men at temperaturen ligger i et spesifikt sjikt kan forventes. I noen tilfeller bør ikke blokkene som tines ligge tett inntil hverandre.

Denne formen for automatisk styring vises å være følsom for modellfeil og støy. Det bør derfor, uansett metode for styring, være en form for tilbakekobling slik at det kan korrigeres for modellfeil og støy. Det bør helst være muligheter for temperaturmålinger inne i tunnelen, men det vil også fungere med temperaturmåling utenfor tunnelen. MPC(Model Predictive Control) anbefales som den beste metoden for å styre effekten til mikrobølgeelementene.

Forord

Denne masteroppgaven presenterer mitt arbeid med modellbasert styring av mikrobølge tinetunnel for frosne kjøttblokker. Oppgaven er utført ved institutt for teknisk kybernetikk på NTNU, våren 2009, med professor II Steinar Sælid som veileder. Oppgaveteksten var spesifisert av og skrevet for Prediktor AS. Oppgavens hensikt er å se på metoder for automatisk, modellbasert styring av en tinetunnel. Prosessen og metodene er implementert i Matlab for testing og simulering.

Det har vært svært inspirerende og lærerikt å arbeide med en slik praktisk oppgave. Det at resultatene i denne oppgaven kan brukes videre i industriell sammenheng har gjort det spennende å skrive denne masteroppgaven.

Jeg vil gjerne takke veileder Steinar Sælid for tilbakemelding og råd. Hans innspill, i form av prosessforståelse og erfaring har vært helt nødvendig for gjennomføring av denne oppgaven. Han har vært svært inspirerende og har gitt meg motivasjon til å jobbe med oppgaven.

Innhold

1	Innledning	1
2	Tineprosessen	3
2.1	Automatisk styring av tinetunnelen	3
2.2	MIP12	3
3	Modeller og fysikalske forhold	5
3.1	Fysiske betraktninger	5
3.1.1	Dielektriske egenskaper (ϵ' og ϵ'')	5
3.1.2	Termiske egenskaper	5
3.1.3	Vannets fysiske tilstand	6
3.1.4	Fuktighetsinnhold	6
3.1.5	Isinnhold	6
3.1.6	Varmekapasitet	6
3.1.7	Termisk konduktivitet	8
3.1.8	Tetthet	9
3.1.9	Fettinnhold	9
3.1.10	Penetrasjonsdybde	9
3.2	Modeller	10
3.2.1	Maxwell's ligninger	11
3.2.2	Lambert's lov	12
4	Totalsystemet	15
4.1	Diskretisering	15
4.1.1	Stabilitet	17
4.2	Simulering av tinetunnelen og tilpassing av parametre	17
4.2.1	Pådrag	18
4.2.2	Tilpasset penetrasjonsdybde	18

4.2.3	Tilpasset termisk konduktivitet	19
4.2.4	Tilpasset varmekapasitet	20
4.2.5	Temperatur i kjøttblokkene	21
5	Optimaliserende system	27
5.1	Pseudokode	27
5.2	Forenklet optimaliserende system	28
5.2.1	Likhetsbetingelse for overflatetemperaturen	32
5.3	Fullt optimaliserende system	35
5.3.1	Programmering og testing	37
5.3.2	Ulikhetsbetingelser for overflatetemperatur	39
5.3.3	Gjeninnføring av objektfunksjon	42
5.3.4	Sammendrag av metoder	47
6	Tilbakekoblet styring	49
6.1	Målinger	49
6.2	Modellfeil	49
6.2.1	Regulering	49
6.2.2	Modifisert regulator	54
6.3	Ulineær MPC	57
6.3.1	Generelt	57
6.3.2	Forslag til bruk av ulineær MPC	57
7	Konklusjon	59
A	Symbolliste	63
B	Forklaring av kode	65
B.1	Forklaring av totalE.m	65
B.1.1	Kode totalE.m	67
B.2	Forklaring av funksjon.m, constr.m og ulin.m	69
B.2.1	funksjon.m	69
B.2.2	Kode funksjon.m	70
B.2.3	constr.m	72
B.2.4	Kode constr.m	72
B.2.5	ulin.m	73

B.2.6	Kode ulin.m	73
B.3	Forklaring av modellogprosess.m	76
B.3.1	Kode modellogprosess.m	76

Figurer

3.1	Andel frossent vann som funksjon av temperatur.	7
3.2	Termisk konduktivitet for magert kjøtt	8
3.3	Forklaring av penetrasjonsdybde	10
3.4	Kobling mellom elektromagnetiske mønstre og masse og varmeoverføring	11
4.1	Diskretiserings illustrasjon	16
4.2	Mikrobølge spredning	19
4.3	Termisk konduktivitet for kjøtt med ulikt fettinnhold	20
4.4	Varmekapasitet for kjøtt med 5% og 25% fett	21
4.5	Temperaturutvikling for kjøtt med 5% og 25% fett	22
4.6	Overflatetemperatur for en hel kjøttblokk. 5% fett.	24
4.7	Overflatetemperatur for kjøttslider med ulikt fettinnhold	25
5.1	Temperaturutvikling for kjøttskiver med hhv. 5% og 25% fett	30
5.2	Pådrag fra mikrobølgeelementer for kjøttskiver med hhv. 5% og 25% fett	31
5.3	Temperaturutvikling for kjøttskiver med hhv. 5% og 25% fett og likhetsbetingelse innført	33
5.4	Pådrag fra mikrobølgeelementer for kjøttskiver med hhv. 5% og 25% fett og likhetsbetingelse innført	34
5.5	Optimalt pådrag og overflatetemperatur for kjøttblokker med hhv 5%, 15% og 25% fett. Med 2 opphold mellom hver blokk.	38
5.6	Optimalt pådrag og overflatetemperatur for kjøttblokk med 5% fett. Ikke mellomrom mellom blokker	40
5.7	Overflatetemperatur for kjøttblokker med 15% og 25% fett. Ikke mellomrom mellom blokker	41
5.8	Optimalt pådrag og overflatetemperatur for kjøttblokk med 5% fett ved bruk av objektfunksjon.	43
5.9	Overflatetemperatur for kjøttblokker med 15% og 25% fett ved bruk av objektfunksjon.	44

5.10	Optimalt pådrag og overflatetemperatur for kjøttblokk med 5% fett ved bruk av tunet objektfunksjon	46
5.11	Overflatetemperatur for kjøttblokker med 15% og 25% fett ved bruk av tunet objektfunksjon	47
6.1	P-Regulator	50
6.2	Overflatetemperatur for en kjøttblokk med 25% fett, når det er modellfeil og blokk i modell har 5% fett	51
6.3	Overflatetemperatur for en kjøttblokk med 5% fett, når det er modellfeil og blokk i modell har 25% fett	53
6.4	Overflatetemperatur og optimalt pådrag	56
6.5	MPC og Kalmanfilter	58

Tabeller

4.1	Faste parametre ved simulering	18
6.1	Overflatetemperatur til kjøttskiver når modifisert regulator benyttes	55
A.1	Symbolliste	63
B.1	Forklaring av variable i kode	66

Kapittel 1

Innledning

Oppgaven omhandler bruk av en mikrobølge tinetunnel for å tine frosne kjøttblokker. Oppgaven er en videreføring av en prosjektoppgave som ble skrevet på etterspørsel fra Stabburet, som skal installere tinetunneler av typen som blir beskrevet i denne oppgaven. Målet med oppgaven er å undersøke metoder for automatisk, modellbasert styring av en tinetunnel. Kapittel 2 er en oppsummering av tineprosessen og problemer rundt automatisk styring av tinetunnelen. Tinetunnelen som skal benyttes blir også kort beskrevet. I kapittel 3 presenteres forskjellige fysikalske forhold som påvirker tineprosessen. Ulike måter å modellere tineprosessen matematisk blir også diskutert. Kapittel 4 tar for seg det totale systemet inkludert tunnel, mikrobølgepådrag og blokktransport. Den matematiske modellen diskretiseres og systemet implementeres her i Matlab. Det kjøres simuleringer for å tilpasse ulike parametre i modellen. Parametrene blir tilpasset slik at systemet blir mest mulig likt en virkelig tineprosess. Deretter vil kapittel 5 omhandle det optimaliserende systemet for å styre tinetunnelen. Her testes forskjellige metoder for optimal styring av mikrobølgepådrag. Det benyttes til å begynne med et forenklet system, som siden utvides. Begge implementeres i Matlab. Kapittel 6 diskuterer så muligheter for tilbakekoblet styring av tinetunnelen. Det utvikles regulatorer for å ta hensyn til modellfeil og støy. Videre diskuteres muligheter for bruk av MPC(Model predictive control) for styring av tinetunnelen.

Kapittel 2

Tineprosessen

2.1 Automatisk styring av tinetunnelen

I kjøttindustrien er frysing av kjøtt meget viktig for å forlenge holdbarheten til kjøttet. Det frosne kjøttet kan dermed lagres og benyttes ved behov. Før det skal benyttes må det tines. Tining ved bruk av mikrobølger er den mest effektive tinemetoden for øyeblikket. Selve tiningen av de frosne kjøttblokkene foregår inne i tinetunnelen. Kjøttblokkene legges på et transportbånd som transporterer blokkene inn til tunnelen. Fettinnholdet i hver blokk er kjent på forhånd ved at blokkene passerer en røntgen fettmåler som måler fettinnhold i blokkene og som også detekterer metall og bein. Dersom det detekteres metall eller bein blir blokken skjøvet av båndet. Selve tinetunnelen består av flere mikrobølgeelementer. Disse sender ut mikrobølger mot kjøttblokkene og kjøttet blir tilført varme. Både før og etter tunnelen kan det være mulighet for å måle overflatetemperatur på blokkene. Temperaturen blokkene har før de entrer tunnelen måles for å vite initialtemperaturen, mens målingen etter tunnelen kan brukes til å kontrollere temperaturen når de kommer ut. I denne oppgaven er det antatt at temperaturen ikke måles ved utgangen av tunnelen. Problemet med denne måten å tine kjøtt på er å styre tinetunnelen. Styreenheten må vite ganske eksakt hvor mye energi mikrobølgeelementene skal sende ut til hver enkelt kjøttblokk. Det er ingen sensorer inne i tunnelen som kan måle temperaturen på blokkene. Den kan kun måles før og etter tunnelen. Det er dessuten kun overflatetemperaturen som kan måles. Hva temperaturen inne i blokkene er, kan ikke måles. Dette gjør at effekten fra mikrobølgeelementene ikke kan tilpasses veldig nøyaktig ut fra temperaturmålinger. Kjøttblokkene har dessuten ulikt fettinnhold og vil derfor ikke kreve like mye energi for å tines. Dette må styreenheten ta hensyn til. Problemet med automatisk styring av tinetunnelen blir derfor å vite hvor mye energi som skal sendes ut til hver enkelt kjøttblokk, basert på initialtemperatur og fettinnhold. Hastigheten på transportbåndet må også bestemmes. Den har også betydning for hvor mye energi som blir tilført kjøttblokkene. For å kunne styre tinetunnelen uten mange målinger, vil det derfor være naturlig å lage en matematisk modell av tineprosessen. Dermed kan modellbasert styring av tinetunnelen benyttes.

2.2 MIP12

Tinetunnelen som Stabburet skal benytte er en MIP12 fra Ferrite. Det er et kontinuerlig tinetunnel system hvor blokkene blir fraktet gjennom tunnelen på et transportbånd. Transportbåndet er laget av mikrobølge transparent materiale. Selve tinetunnelen er ca 2.5m lang. Det er fire stk mikrobølgeelementer i tunnelen. Disse er montert to og to, slik at det ene sender mikrobølger mot oversiden og det andre mot undersiden av kjøttblokkene. Dermed er det to mikrobølgeelementer

på oversiden av transportbåndet, og to på undersiden. På dette viset oppnår man hurtigere tining siden kjøttblokkene blir varmet opp fra over og undersiden. Alle mikrobølgeelementene har en effekt på 75 kW. MIP12 benytter mikrobølger med en frekvens på 915MHz. Tiden en blokk bruker gjennom tunnelen er ca 5 minutter. Dette tilsvarer en hastighet på transportbåndet på litt under 1cm/s. MIP12 kan tine opp til 7700kg kjøtt med initialtemperatur på -18°C per time. Se ellers [1] for mer informasjon om MIP12.

Kapittel 3

Modeller og fysikalske forhold

Denne delen av oppgaven dreier seg om et litteratur søk etter modeller og fysikalske forhold som beskriver tineprosessen. Det har i ettertid vist seg at det finnes begrenset med informasjon om dette emnet, så mye av stoffet som presenteres her kommer fra de samme kildene.

3.1 Fysiske betraktninger

Det er flere fysiske aspekter å ta hensyn til når man skal modellere tineprosessen. Under tining vil mange av egenskapene til kjøttet forandre seg ettersom temperaturen øker. Dette vil det være nødvendig å ta med i den matematiske modellen.

3.1.1 Dielektriske egenskaper (ϵ' og ϵ'')

De dielektriske egenskapene til kjøtt kan hovedsaklig oppsummeres som den dielektriske konstanten (ϵ') og dielektrisk tapfaktor (ϵ''). Kjøttets evne til å lagre elektrisk feltenergi representeres ved den dielektriske konstanten. Mens den dielektriske tapfaktoren forteller om kjøttets evne til å konvertere elektrisk energi til varme. Til sammen utgjør disse to egenskapene permitiviteten til kjøttet. Mikrobølger påvirker kjøttet basert på kjøttets dielektriske egenskaper. Varme genereres i kjøttet primært gjennom absorbering av mikrobølger. De dielektriske egenskapene er dynamiske og forandrer seg som en funksjon av frekvensen til mikrobølgene, sammensetting av kjøttet, temperaturen i kjøttet og tettheten til kjøttet [6]. De dielektriske egenskapene har en direkte påvirkning på penetrasjonsdybden til mikrobølgene. Det meldes i [5] at ved mikrobølge frekvenser (0.3 - 300 GHz), vil de dielektriske egenskapene til kjøtt ved temperaturer fra -20°C til -5°C ikke forandre seg mye. Mens de øker betraktelig i temperatur sjiktet -3°C til -1°C . Både den dielektriske konstanten og tapfaktoren varierer mye med frekvensen til mikrobølgene. Dersom frekvensen senkes, medfører det en økning i verdiene til ϵ' og ϵ'' over hele temperatur skalaen. Det er dessuten slik at både ϵ' og ϵ'' vil være større over hele temperatur skalaen dersom fettinnholdet minker. Mindre fett gir høyere verdier på de dielektriske egenskapene.

3.1.2 Termiske egenskaper

Spesifikk varmekapasitet og termisk konduktivitet utgjør kjøttets termiske egenskaper. Spesifikk varmekapasitet sier noe om hvor raskt kjøttet kan bli varmet opp. Altså hvor mye energi man trenger for å varme opp en gitt masse kjøtt, en grad. Den termiske konduktiviteten til kjøtt spiller også en viktig rolle ved oppvarming. Ved høy termisk konduktivitet avledes varme raskere enn

ved lav termisk konduktivitet. Det betyr at dersom den termiske konduktiviteten er høy, vil det ta kortere tid å oppnå uniform temperatur i kjøttet. Altså at mer varme går fra steder med høy temperatur til steder med lavere temperatur og utligner temperaturen. Dersom den termiske konduktiviteten minker vil mindre varme overføres [6].

3.1.3 Vannets fysiske tilstand

Den fysiske tilstanden til vannet i kjøttet påvirker mikrobølgeoppvarmingen. Når frossent kjøtt skal tines, er det isen i kjøttet som må smeltes. Den dielektriske konstanten til vann er større enn den til is. Når tiningen påvirkes av mikrobølger, blir deler av kjøttet som har begynt å tine, hurtigere varme enn deler som fortsatt er frosne. Dette er på grunn av forskjell i den dielektriske konstanten til is og vann. Det vil si at de delene som har begynt å tine absorberer mer varme enn de frosne. Derfor er overoppvarming ved tinte områder og underoppvarming ved frosne områder et problem ved bruk av mikrobølger [6].

3.1.4 Fuktighetsinnhold

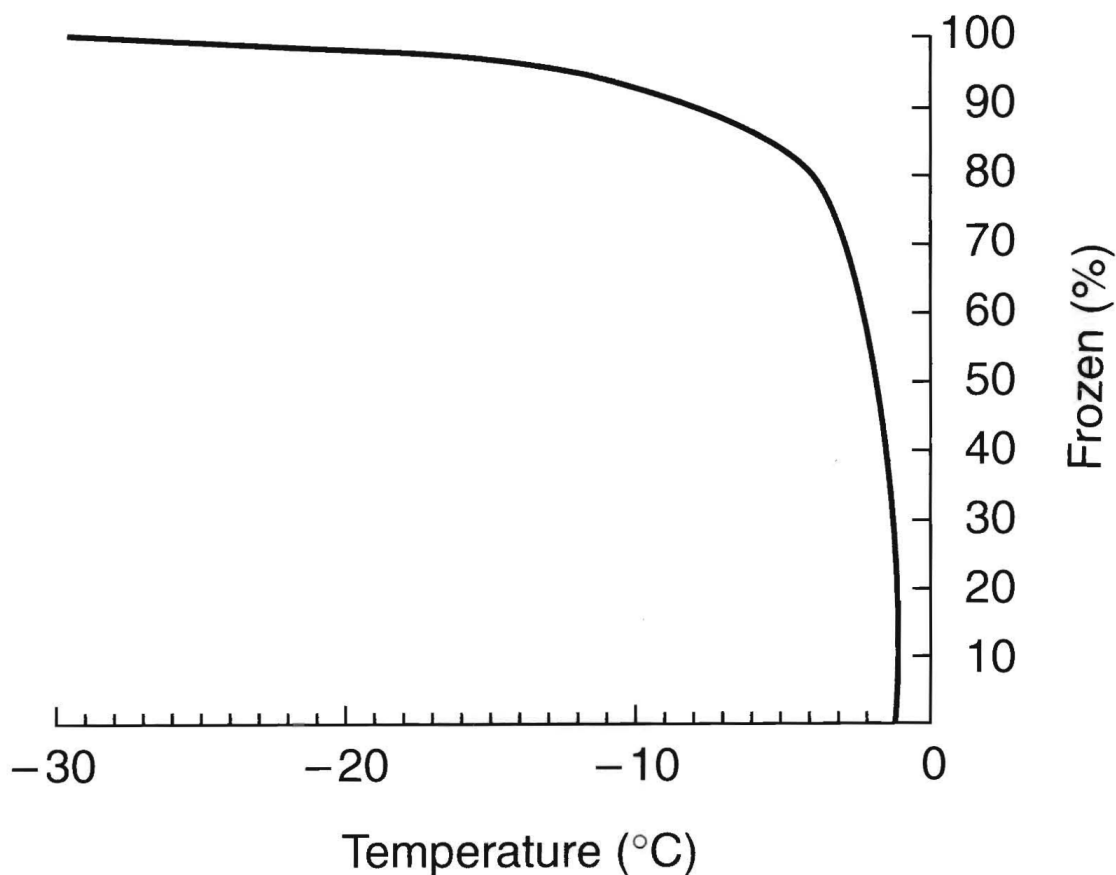
Fuktighetsinnholdet i kjøttet har en betydelig påvirkning på de dielektriske egenskapene, og følgelig penetrasjonsdybden til mikrobølgene. Ujevn oppvarmingshastighet er observert ved høyt fuktighetsinnhold pga. liten penetreringsdybde. Ved lavt fuktighetsinnhold er oppvarmingen jevnere pga. dypere penetreringsdybde. I tillegg kan varmetap gjennom overflaten som følge av fordampning oppstå ved høyt fuktighetsinnhold [6].

3.1.5 Isinnhold

Kjøttet begynner først å fryse ved temperaturer rundt -1.5°C . For temperaturer lavere enn frysepunktet vil kjøttet bli enda mer frossent. Dette skyldes at frysing fører til en økning i konsentrasjonen av vev væske, og derfor er lavere temperatur nødvendig for at frysingen skal fortsette. Figur 3.1 viser andelen frossent vann som funksjon av temperaturen i magert kjøtt (ca 5% fett). Figuren viser at ca halvparten av vannet har fryst ved -2°C , og hele 80% av vannet har fryst før temperaturen når -3°C . Dette viser at det meste av frysingen foregår i temperatursjiktet -1.5°C til -3°C [7].

3.1.6 Varmekapasitet

Den effektive varmekapasiteten til frossent kjøtt forandrer seg under tining. Det meste av tiningen foregår i temperatursjiktet -3° til -1° . I denne fasen smelter isen og det gir forandringer i den effektive varmekapasiteten [10]. Det er gjort mange forsøk på å finne en eksakt beskrivelse av varmekapasiteten for kjøtt. De forskjellige beskrivelsene kan ta hensyn til både fettinnhold, vanninnhold og temperatur. Disse er ofte basert på eksperimenter og viser hvordan varmekapasiteten varierer med temperaturen. Det som går igjen i alle beskrivelsene, er at varmekapasiteten øker veldig mye når temperaturen nærmer seg frysepunktet. Det kommer av at all isen som smeltes til vann krever mye energi. Ved temperaturer større enn frysepunktet, vil varmekapasiteten forholde seg konstant og være mye mindre enn for lavere temperaturer. I forprosjektet til denne oppgaven ble følgende beskrivelse av varmekapasiteten benyttet. Denne er basert på eksperimenter og viser hvordan varmekapasiteten varierer med temperaturen. Den er hentet fra [10].



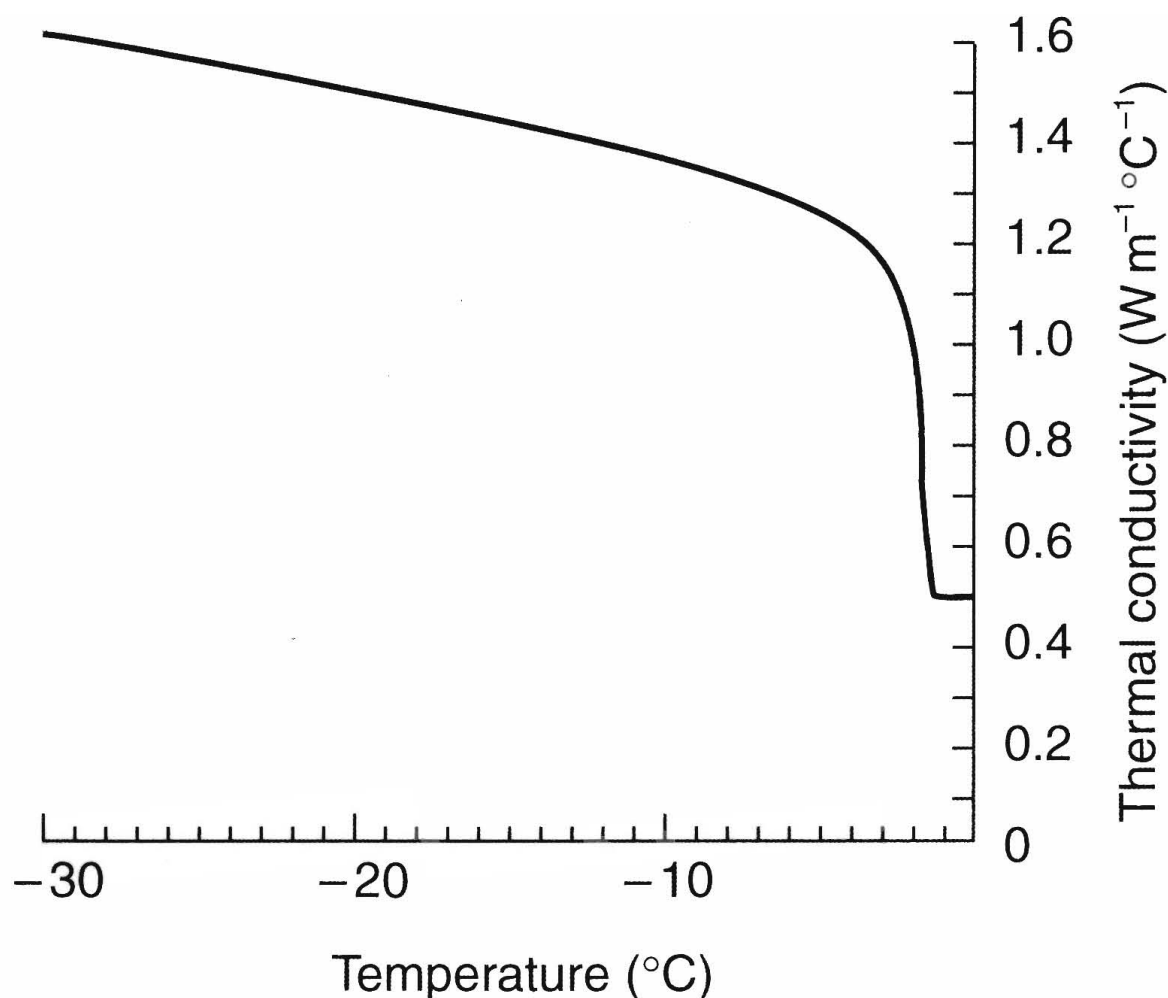
Figur 3.1: Andel frossent vann som funksjon av temperatur.

$$\begin{aligned}
 C_p &= 0.00007T^5 + 0.0061T^4 + 0.192T^3 + 2.9T^2 + 21.35T + 66.72, \text{ for } -25^\circ\text{C} \leq T \leq -4^\circ\text{C} \\
 C_p &= 5.8T^3 + 62.83T^2 + 236.6T + 329.4, \text{ for } -4^\circ\text{C} \leq T \leq -2^\circ\text{C} \\
 C_p &= 153.46T + 386 \text{ for } -2^\circ\text{C} \leq T \leq -1.4^\circ\text{C} \\
 C_p &= -374T - 370.6 \text{ for } -1.4^\circ\text{C} \leq T \leq -1^\circ\text{C} \\
 C_p &= 3510 \text{ for } -1^\circ\text{C} \leq T
 \end{aligned}
 \tag{3.1}$$

Denne beskrivelsen viste seg å gi en litt for høy varmekapasitet, dersom man sammenligner med tining av kjøttblokker i en virkelig tinetunnel. Dette ble vist i forprosjektet. Den tar heller ikke hensyn til fettinnhold eller vanninnhold. I [11] presenteres flere modeller for varmekapasiteten. Følgende formel fra denne artikkelen vil bli benyttet videre i modellen:

$$\begin{aligned}
 C_p &= -1.4842W - 0.0191F + \frac{-94.8594 + 87.8803W}{T} + \frac{2.3441}{T^2}, \text{ for } -25^\circ\text{C} \leq T \leq -1.5^\circ\text{C} \\
 C_p &= 0.00281T + 5.171132W + 2.87307F - 0.4615, \text{ for } -1.5^\circ\text{C} \leq T
 \end{aligned}
 \tag{3.2}$$

Her er W vanninnhold, F er fettinnhold og frysepunktet er antatt til -1.5°C . Denne beskrivelsen av varmekapasiteten tar hensyn til fett og vanninnhold. Usikkerhetsmomenter ved beskrivelsen



Figur 3.2: Termisk konduktivitet for magert kjøtt

av varmekapasitet er frysepunktet og toppen som oppstår rundt frysepunktet. Man kan ikke vite at frysepunktet er eksakt -1.5° . Frysepunktet kan variere med type kjøtt og fettinnhold, men for å forenkle modellen velges det for videre arbeid til -1.5°C . Denne antagelsen begrunnes ut fra figur 3.1, som viser at vannet i kjøttet ikke fryser før -1.5°C . Det samme gjelder toppen som oppstår i varmekapasiteten ved temperaturer rundt frysepunktet. Den kan også variere og er vanskelig å få helt eksakt. Det finnes et utall formler for varmekapasiteten og ingen er helt like. Det som er viktig er at det faktisk oppstår en topp på varmekapasiteten ved frysepunktet, og at varmekapasiteten er adskillig mye større der enn ved andre temperaturer.

3.1.7 Termisk konduktivitet

Den termiske konduktiviteten til kjøtt er fra tidligere forskning funnet å variere med temperatur, vanninnhold og fettinnhold [10]. Den termiske konduktiviteten til is er ca fire ganger større enn den til vann, derfor vil den termiske konduktiviteten til kjøtt øke ved økende isinnhold. For magert kjøtt, ca 5% fett, er den termiske konduktiviteten målt til 0.49W/mC for temperaturer over 0°C . For rent fett er den målt til ca 0.25W/mC . Figur 3.2 viser den termiske konduktiviteten til magert kjøtt for temperaturer lavere enn 0°C [7]. Grunnen til at den varierer mye for temperaturer mindre enn ca -1.5°C er at isinnholdet i kjøttet forandres. Konduktiviteten til is er større enn konduktiviteten til vann. For temperaturer større enn ca -1.5°C vil det ikke være is i kjøttet, kun vann, og derfor vil konduktiviteten være konstant.

3.1.8 Tetthet

For å kunne lage en matematisk modell for tining av kjøtt er det viktig å vite noe om tettheten til kjøttet. Tettheten til kjøtt varierer ikke betydelig med temperaturen. For magert kjøtt er tettheten målt til å være $1070\text{kg}/\text{m}^3$. For rent fett er den målt til $920\text{kg}/\text{m}^3$ [7]. Siden variasjonen i tetthet er lav med hensyn på fettinnhold og temperatur, trenger man ikke en veldig nøyaktig verdi for tettheten. Det vil være godt nok med en verdi i området $920 - 1070\text{kg}/\text{m}^3$.

3.1.9 Fettinnhold

Innholdet av fett i kjøttet påvirker oppvarmingen ved bruk av mikrobølger. Fettinnholdet påvirker de termiske og dielektriske egenskapene til kjøttet og fett absorberer energi raskere enn rent kjøtt [2]. Det er derfor viktig at man har kontroll over fettinnholdet. Ofte kan fettene i blokkene være ujevnt fordelt. En del av blokken kan bestå av veldig mye fett, mens en annen del kan bestå av rent kjøtt. Det er kun samlet fettinnhold i blokken man vet noe om. Dette kan være problematisk for tineprosessen. Dersom en del av kjøttblokken består av veldig lite fett, mens en annen del består av større mengder fett, kan den delen av blokken som består av mye fett bli veldig varm, mens den som består av lite fett fortsatt kan være frossen. Fettet påvirker altså varmekapasiteten til kjøttet. Dette er det vanskelig å gjøre noe med og er noe som bør taes i betraktning.

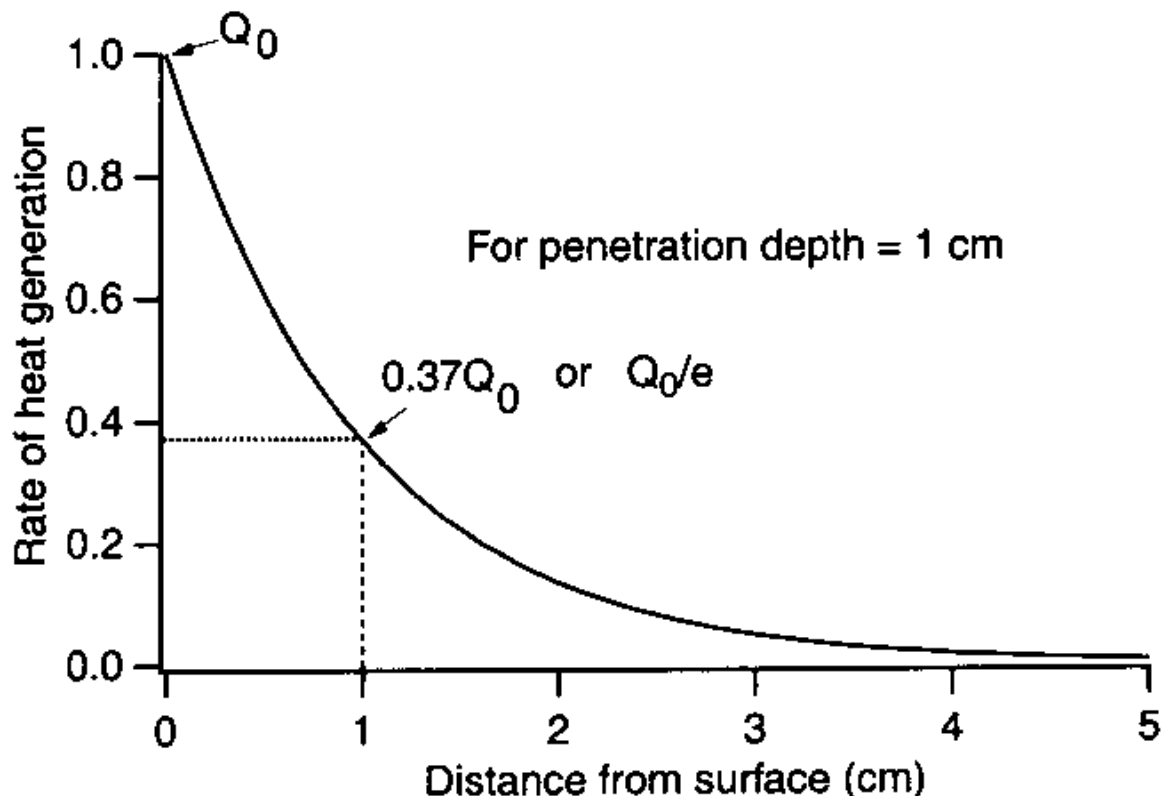
3.1.10 Penetrasjonsdybde

Definisjonen på penetrasjonsdybde er distansen fra overflaten og inn i materialet hvor 63% av innfallende mikrobølge energi er gått tapt gjennom absorbasjon i materialet. Dette er illustrert i figur 3.3

Penetrasjonsdybden til mikrobølgene varierer med temperatur og fettinnhold. Penetrasjonsdybden vil variere som følge av at de dielektriske egenskapene forandrer seg. Sammenhengen mellom penetrasjonsdybden og de dielektriske egenskapene er følgende:

$$\delta_p = \frac{c}{2\pi f \sqrt{2\varepsilon'(\sqrt{1 + (\frac{\varepsilon''}{\varepsilon'})^2} - 1)^{\frac{1}{2}}}} \quad (3.3)$$

hvor ε' er den dielektriske konstanten og ε'' er den dielektriske tapfaktoren [4]. Det er funnet forskjellige uttrykk for permittiviteten til kjøtt. Noen tar hensyn til fettinnhold og temperatur [6], mens andre kun tar hensyn til temperaturen i kjøttet. Disse uttrykkene er funnet ved eksperimenter. I forprosjektet vises det hovedsaklig at når temperaturen nærmer seg frysepunktet (-1.5°C) begynner både den dielektriske konstanten og tapfaktoren å stige betraktelig. Det betyr at kjøttet vil ta opp mer energi fra mikrobølgene når temperaturen stiger, og samtidig gjøre om mer av denne energien til varme. Når temperaturen har passert frysepunktet vil den dielektriske konstanten og tapfaktoren være svakt avtagende. Dette gjør at penetrasjonsdybden blir vesentlig større når kjøttet er i frossen tilstand, enn når det er tint. Andre eksperimenter, [10], melder om en relativt høy og konstant penetrasjonsdybde for kjøtt i frossen tilstand, gitt at temperaturen er lavere enn -5°C . For temperaturer fra -5°C og opp til frysepunktet vil penetrasjonsdybden være avtagende, og for temperaturer høyere enn 0°C og opp til 20°C melder [6] at penetrasjonsdybden er nokså konstant uansett fettprosent. Dette gjelder når de ulike kjøtttypene blir utsatt for mikrobølger med lik frekvens. Penetrasjonsdybden vil variere kraftigere med frekvensen til mikrobølgene enn den vil variere med fettinnhold i kjøttet. Lavere frekvens vil gi større penetrasjonsdybde. For mikrobølger på 915MHz vil penetrasjonsdybden være ca 2.2cm



Figur 3.3: Forklaring av penetrasjonsdybde

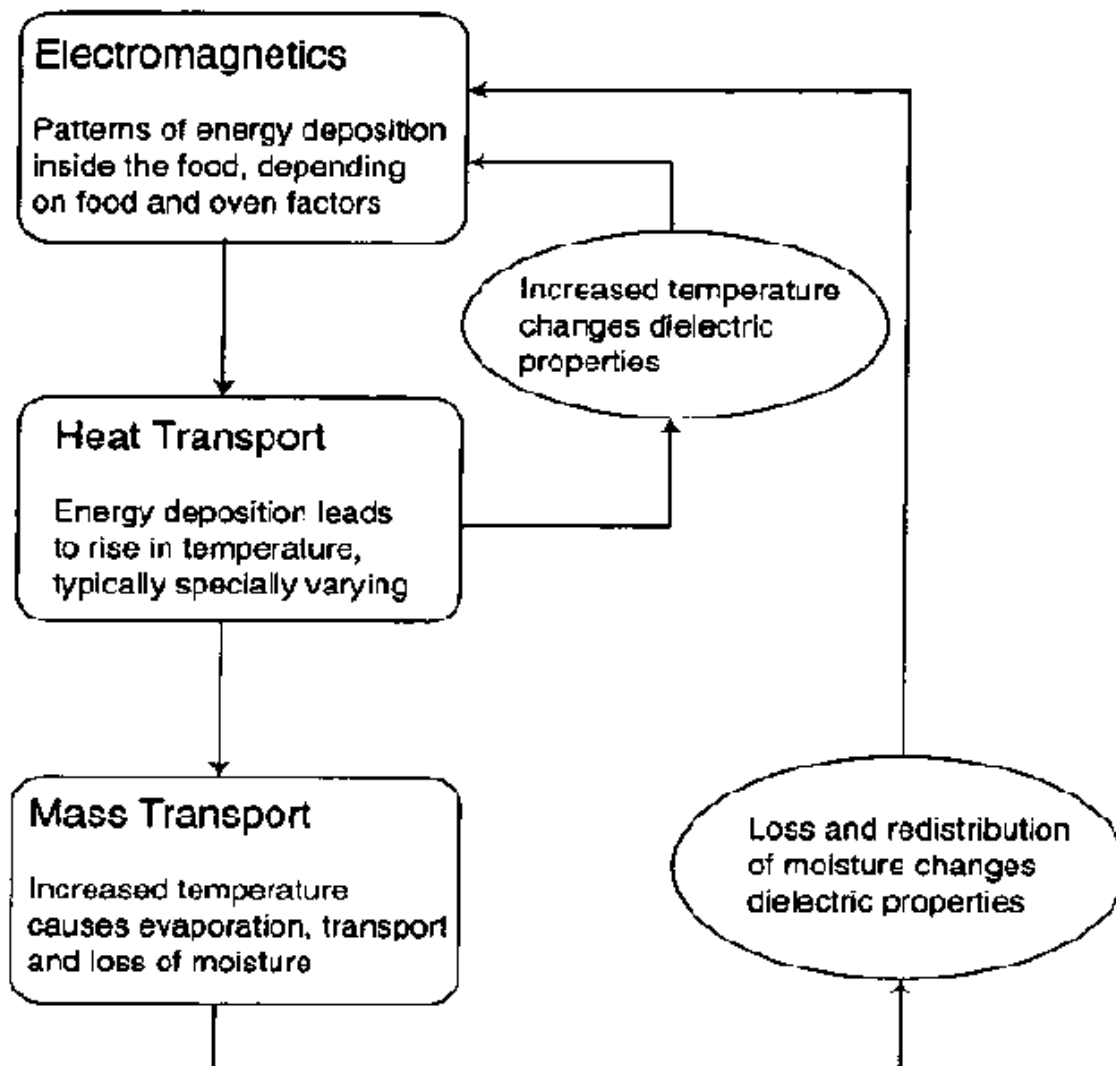
for temperaturer over 0°C og opp til 20°C . For temperaturer lavere enn -5°C vil penetrasjonsdybden være større enn 15 cm. I sjiktet -5°C opp til frysepunktet minker penetrasjonsdybden lineært fra 15 cm til ca 2.2 cm [6]. Dette kommer av forandringer i de dielektriske egenskapene til kjøttet i dette sjiktet. Det er i dette sjiktet mesteparten av isen smelter.

3.2 Modeller

Når temperaturen forandrer seg, forandrer de dielektriske egenskapene til kjøttet seg. Dette, pluss at de elektromagnetiske energimønstrene endres, gjør at beskrivelsen av energidistribusjonen for tining av kjøtt i mikrobølgeovner blir svært kompleks. Det er stor sammenheng mellom de elektromagnetiske mønstrene, og varme og masseoverganger. Figur 3.4 fra [4] viser koblingen mellom disse.

I [4] presenteres tre måter å modellere mikrobølger for mat prosessering. Disse tre rangeres her etter kompleksitet og nøyaktighet.

1. Koblet modell hvor elektromagnetikk og varmeoverføring(eller fuktighet) er inkludert, og egenskaper er temperatur avhengige. Dette krever gjentatt løsning av Maxwell's ligning under oppvarming. Denne er mest nøyaktig og mest kompleks.
2. Elektromagnetikk og fuktighets overføring er inkludert. Den elektromagnetiske modellen er løst en gang og det resulterende mikrobølgemønsteret blir brukt som varmekilde for å beregne transient varme og/eller fuktighets overføring. Kompleksiteten av den elektromagnetiske modelleringen er her fjernet, og under noen betingelser, f.eks. korte perioder med



Figur 3.4: Kobling mellom elektromagnetiske mønstre og masse og varmeoverføring

oppvarming, er dette en rimelig god beskrivelse av prosessen.

3. Elektromagnetikken er forenklet til Lambert's lov, ligning 3.15, og brukt som en varmekilde i en varme eller fuktighets overføringsmodell. Dette er den enkleste og minst nøyaktige beskrivelsen av prosessen. Denne er den mest vanlige for modellering av matprosessering.

Ellers i litteraturen presenteres det i hovedsak to måter å modellere tining med mikrobølger. Disse to er Maxwell's ligninger og Lambert's lov. Disse er to litt forskjellige metoder når det gjelder kompleksitet. Løsning av Maxwell's ligninger er mer kompleks enn løsning av Lambert's lov. Det er disse to som er de vanligste modellene, derfor blir begge de to metodene presentert her.

3.2.1 Maxwell's ligninger

I [3] blir Maxwell's ligninger presentert. Maxwell's ligninger gir en beskrivelse av det elektromagnetiske feltet som skapes av mikrobølgene. De gjeldende ligningene er følgende:

$$\nabla \times H = \frac{\partial D}{\partial t} + J = \frac{\partial \varepsilon E}{\partial t} + \sigma E \quad (3.4)$$

$$\nabla \times E = -\frac{\partial B}{\partial t} = -\frac{\partial \mu H}{\partial t} \quad (3.5)$$

$$\nabla \cdot D = \nabla \cdot \varepsilon E = \rho_e \quad (3.6)$$

$$\nabla \cdot B = \nabla \cdot \mu H = 0 \quad (3.7)$$

Sammenhengen mellom J, D og B til E og H er følgende:

$$J = \sigma(\omega)E(t) \quad (3.8)$$

$$D = \varepsilon(\omega)E(t) \quad (3.9)$$

$$B = \mu(\omega)H(t) \quad (3.10)$$

ε er kompleks permitivitet og er definert som følgende:

$$\varepsilon = \varepsilon' - j\varepsilon'' \quad (3.11)$$

Her er $\omega = 2\pi f$ hvor f er frekvensen til mikrobølgene. Ved å løse disse ligningene får man et uttrykk for det elektriske feltet.

$$E = E_0 \sin\left(\frac{\pi x}{a}\right) \quad (3.12)$$

Deretter finner man vektoren $P(t)$.

$$P(t) = E(t) \times H(t) \quad (3.13)$$

Fra denne kan man finne ut hvor mye energi som er absorbert av materialet utsatt for mikrobølger, ut fra det lokale elektriske feltet.

$$Q_{abs} = \omega E_{rms}^2 \varepsilon_0 \varepsilon'' \quad (3.14)$$

Ut fra dette kan man finne temperaturen i kjøttet.

3.2.2 Lambert's lov

Lambert's lov er en forenklet beskrivelse av energiabsorbasjonen og varmeoverføring i mat, ved bruk av mikrobølger. Den er nøyaktig i begrensede situasjoner, hvor mikrobølgene kan tilnærmes som plane bølger, gitt at de er langt fra kilden og over et lite område [4].

$$Q = Q_0 e^{-y/\delta_p} \quad (3.15)$$

Q_0 er varmen på overflaten.

y er distanse inn i materialet, fra overflaten.

δ_p er penetrasjonsdybden.

Det er viktig å notere seg at ligning 3.15 kun er en kvalitativ beskrivelse av energiabsorbasjonen ved bruk av mikrobølger og kun er gyldig i begrensede situasjoner. I materialet som inneholder mye vann med salt, viser en detaljert beskrivelse et hurtig fall i energiabsorbasjonen fra overflaten og inn i materialet. Ganske likt en eksponentiell avtagende funksjon. Når materialet ikke inneholder så mye vann med salt, kan ikke ligning 3.15 beskrive energiabsorbasjonen veldig nøyaktig. Videre blir Lambert's lov benyttet for å modellere temperaturen og energiopptaket i kjøttet. Den ledende energibevaringsligningen for mikrobølgeoppvarming benytter raten av varme generering, Q fra ligning 3.15, som kilde. Dette er vist i følgende en-dimensjonale energibevarings ligning for mikrobølgeoppvarming:

$$\rho c_p \frac{\delta T}{\delta t} = k \frac{\delta^2 T}{\delta y^2} + Q \quad (3.16)$$

Likning 3.15 og 3.16 gir da følgende partielle differensialligninger

$$\frac{dQ}{dy} = -\frac{1}{\delta_p(T)} Q \quad (3.17)$$

$$\frac{\delta T}{\delta t} = \frac{k(T)}{\rho c_p(T)} \frac{\delta^2 T}{\delta y^2} + \frac{Q}{\rho c_p(T)} \quad (3.18)$$

Som ligning 3.17 og 3.18 viser, har nå varmekapasiteten c_p , konduktiviteten k og penetreringsdybden δ_p blitt avhengige av temperaturen i kjøttblokken. Som nevnt tidligere vil de dielektriske og termiske egenskapene variere med temperaturen og dermed vil også varmekapasiteten, konduktiviteten og penetreringsdybden variere med temperaturen. Ved å ha varierende varmekapasitet, konduktiviteten og penetreringsdybde, blir det tatt hensyn til fase endringer i kjøttet, f.eks is til vann.

Videre i oppgaven vil Lambert's lov bli benyttet i simuleringer, da denne er minst kompleks [3].

Kapittel 4

Totalsystemet

Fram til nå er det kun fokusert på å modellere varmeutviklingen i en kjøttblokk som blir utsatt for mikrobølger. I denne seksjonen blir hele systemet beskrevet. Det blir tatt hensyn til tunnelen, mikrobølgepådragene og transport av blokker. Videre blir også de forskjellige fysiske parametrene til kjøtt tilpasset slik at det hele korrelerer med operasjon av en virkelig tinetunnel. Ligning 3.16 er kun en beskrivelse av varmeutviklingen i en kjøttblokk som utsettes for mikrobølger. For å ta hensyn til bevegelse av blokkene gjennom mikrobølgetunnelen må modellen utvides. Følgende ligning vil derfor bli benyttet:

$$\rho c_p \frac{\delta T}{\delta t} + v \frac{\delta T}{\delta x} = k \frac{\delta^2 T}{\delta y^2} + Q \quad (4.1)$$

Her er leddet $v \frac{\delta T}{\delta x}$ et transportledd, og v er hastigheten på transportbåndet i m/s. Dermed får man følgende partielle differensialligning som erstatter ligning 3.18:

$$\frac{\delta T}{\delta t} = \frac{k(T)}{\rho c_p(T)} \frac{\delta^2 T}{\delta y^2} + \frac{Q}{\rho c_p(T)} - \frac{v}{\rho c_p(T)} \frac{\delta T}{\delta x} \quad (4.2)$$

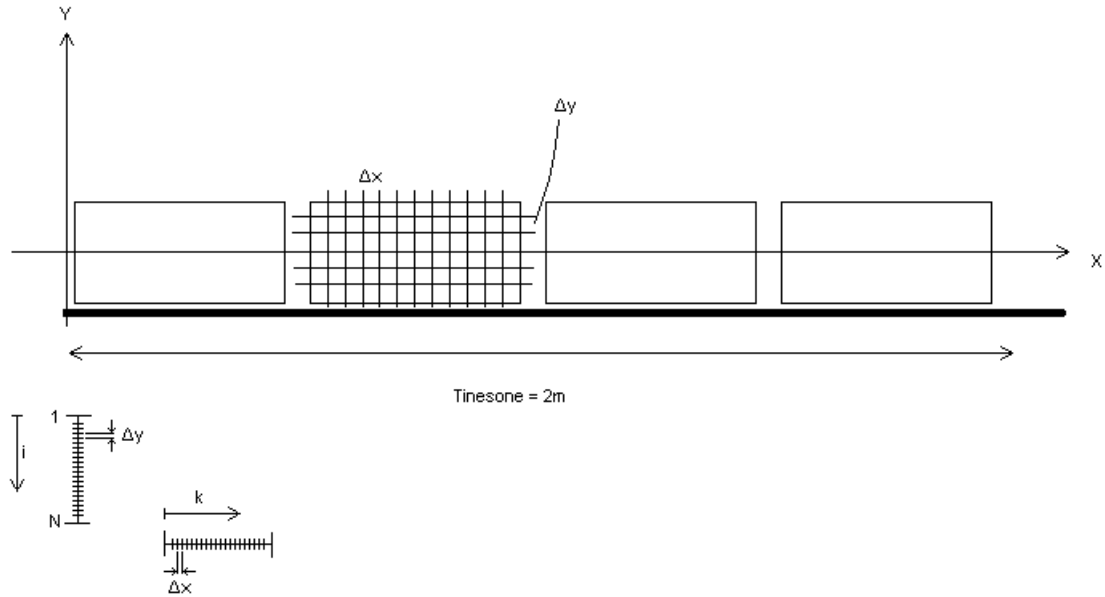
Ligning 3.17 og 4.2 utgjør nå den totale matematiske modellen for systemet. For å kunne realisere den valgte modellen i Matlab, må ligningene for modellen diskretiseres.

4.1 Diskretisering

Det benyttes en Finite-Difference approksimasjon, fra f.eks [9], for å diskretisere ligningene 3.17 og 4.2. Det antas at kjøttblokken kan deles horisontalt i to, slik at tykkelsen blir halvert, som vist på figur 4.1. Dette antas fordi ovnen sender mikrobølger fra oversiden og undersiden av blokken. Det antas også at sideveggene til blokken er isolerte, altså at det ikke er noe varmeutveksling gjennom sideveggene. Man får så følgende approksimasjoner for de deriverte i tid og rom:

$$\frac{\delta T}{\delta t} = \frac{T_{i,j}^{k+1} - T_{i,j}^k}{\Delta T} \quad (4.3)$$

$$\frac{\delta^2 T}{\delta t^2} = \frac{T_{i,j+1}^k - 2T_{i,j}^k + T_{i,j-1}^k}{\Delta y^2} \quad (4.4)$$



Figur 4.1: Diskretiserings illustrasjon

$$\frac{dT}{dx} = \frac{T_{i,j}^k - T_{i-1,j}^k}{\Delta x} \quad (4.5)$$

$$\frac{dQ}{dy} = \frac{Q_{i,j+1}^k - Q_{i,j}^k}{\Delta y} \quad (4.6)$$

Dette gir følgende uttrykk for temperaturen og energien.

$$T_{i,j}^{k+1} = T_{i,j}^k - \frac{v\Delta T}{\Delta x} T_{i,j}^k + \frac{v\Delta T}{\Delta x} T_{i-1,j}^k + \Delta T \frac{k}{\rho c_p} \frac{T_{i,j+1}^k - 2T_{i,j}^k + T_{i,j-1}^k}{\Delta y^2} + \Delta T \frac{Q_{i,j}^k}{\rho c_p} \quad (4.7)$$

$$Q_{i,j}^{k+1} = Q_{i,j}^k - \frac{\Delta y}{\delta_p} Q_{i,j}^k \quad (4.8)$$

Velger så $\frac{v\Delta T}{\Delta x} = 1$ dvs: $\Delta x = v\Delta T$

Får da at

$$T_{i,j}^{k+1} = \frac{v\Delta T}{\Delta x} T_{i-1,j}^k + \Delta T \frac{k}{\rho c_p} \frac{T_{i,j+1}^k - 2T_{i,j}^k + T_{i,j-1}^k}{\Delta y^2} + \Delta T \frac{Q_{i,j}^k}{\rho c_p} \quad (4.9)$$

Δx velges da slik at blokkene beveger seg Δx meter på et tidsskritt. Det er viktig å velge ΔT slik at Δx ikke blir for liten, slik at man unngår stabilitetsproblemer. Hastigheten v er ca 2 meter på 5 minutter, $\frac{0,4}{60}$ m/s. Dersom Δx ønskes til 0.1m blir $\Delta T = \frac{\Delta x}{v} = \frac{0,1}{0,4/60} = 15$ s

Grensebetingelsene ved overflaten og bunnen er følgende:

Energi ved overflaten:

$$Q_{i,0}^k = \text{konstant} \quad \forall i \quad (4.10)$$

Temperatur ved overflaten:

$$\begin{aligned}
 k \frac{\delta T}{\delta y} &= h(T_{omgivelser}^k - T_{i,1}^k) \quad \forall i \\
 \Downarrow \\
 T_{i,1}^k &= \frac{T_{i,2}^k + \frac{T_{omgivelser}^k \Delta y h}{k}}{1 + \frac{\Delta y h}{k}}
 \end{aligned} \tag{4.11}$$

Temperatur i bunnen:

$$\frac{\delta T}{\delta y} = 0, \text{ for } y_j = y_N \Rightarrow \frac{T_{i,N}^k - T_{i,N-1}^k}{\Delta y} = 0 \Rightarrow T_{i,N}^k = T_{i,N-1}^k \tag{4.12}$$

Disse ligningene utgjør modellen for temperaturen og varmen i kjøttblokkene som funksjon av tid. Det er også slik at varmekapasitet, termisk konduktivitet og penetrasjonsdybde vil variere med temperatur og fettinnhold. De vil derfor bli følgende:

$$\begin{aligned}
 k &= k(T_{i,j}^k, F) \\
 c_p &= c_p(T_{i,j}^k, F) \\
 \delta_p &= \delta_p(T_{i,j}^k, F)
 \end{aligned} \tag{4.13}$$

4.1.1 Stabilitet

Et problem med Finite Difference metoden er at den kan bli ustabil dersom de fysiske egenskapene til kjøttet varierer, noe de i dette tilfellet gjør. Dersom man ser på den paraboliske delen av ligning 4.2, nemlig leddet $\frac{\delta T}{\delta t} = \frac{k(T)}{\rho c_p(T)} \frac{\delta^2 T}{\delta y^2}$ kan man definere følgende tall:

$$\alpha = \frac{k \Delta T}{\rho c_p (\Delta y)^2} \tag{4.14}$$

Ut fra kjente stabilitetskrav vet man at α må være mindre enn $\frac{1}{2}$ for at metoden skal være stabil i y-retning.

Den hyperbolske delen av ligningen representeres ved leddet $\frac{\delta T}{\delta t} = -\frac{v}{\rho c_p(T)} \frac{\delta T}{\delta x}$. At en likning er hyperbolsk betyr at det er transport eller bølgebevegelser involvert. I dette tilfellet er det transport av kjøtt gjennom tunnelen. På samme måte som den paraboliske delen har også den hyperbolske delen et eget stabilitetskriterium som må tilfredsstilles. Det er kjent fra før og sier følgende: $v \frac{\delta T}{\delta x} \leq 1$. Når dette er oppfylt blir metoden stabil i x-retningen.

I tillegg ser man også fra ligning 4.8 at $\frac{\Delta y}{\delta_p}$ må være mindre enn 1 for at metoden skal være stabil. Dersom dette tallet er større enn 1, vil energi fordelingen nedover lagene i kjøttet veksle mellom positive og negative verdier. For å unngå dette problemet vil ligning 4.8 bli skiftet ut med ligning 4.15 ved simulering:

$$Q_{i,j}^{k+1} = Q_{i,j}^k e^{-\frac{\Delta y}{\delta_p}} \tag{4.15}$$

Her antas det at penetrasjonsdybden er konstant over integrasjonsintervallet [9].

4.2 Simulering av tinetunnelen og tilpassing av parametre

For å kunne simulere tinetunnelen må de diskretiserte ligningene implementeres i Matlab. Så må de forskjellige parametrene tilpasses, slik at modellen er korrekt i forhold til virkeligheten. Det er

	$T > -1.5^{\circ}\text{C}$	$T \leq -1.5^{\circ}\text{C}$
δ_p	0.022 m	0.15 m
ρ	1057 kg/m ³	1057 kg/m ³
h	5 W/m ² K	5 W/m ² K
ΔT	15	15
Δx	0.1	0.1
Δy	0.01	0.01
v	$\frac{0.4}{60}$ m/s	$\frac{0.4}{60}$ m/s
$T_{\text{Omgivelser}}$	10°C	10°C

Tabell 4.1: Faste parametre ved simulering

i hovedsak varmekapasiteten som tilpasses, da det er mest usikkerhet knyttet til denne. Det blir også utviklet ligninger for den termiske konduktiviteten. Det vil videre bli presentert plott som viser pådraget som ble benyttet, hvordan varmekapasiteten og den termiske konduktiviteten endrer seg med temperaturen i kjøttet, og hvordan temperaturen endrer seg når kjøttblokkene beveger seg gjennom tunnelen. Det vil være simuleringer av kjøttblokker med ulikt fettinnhold så man kan se hvordan fettinnholdet påvirker tiningen. I alle simuleringer ble det benyttet et tidsskritt på 15 sekunder. Dette ble gjort for å øke hastigheten på simuleringene. Nøyaktigheten ble litt lavere enn om tidsskrittet hadde vært mindre, men sluttemperaturen på kjøttet ble den samme. Andre parametre som ble benyttet vises i tabell 4.1. Se ellers appendiks B for Matlab kode som ble benyttet i denne seksjonen. Det er også vedlagt forklaring til kode som ble benyttet.

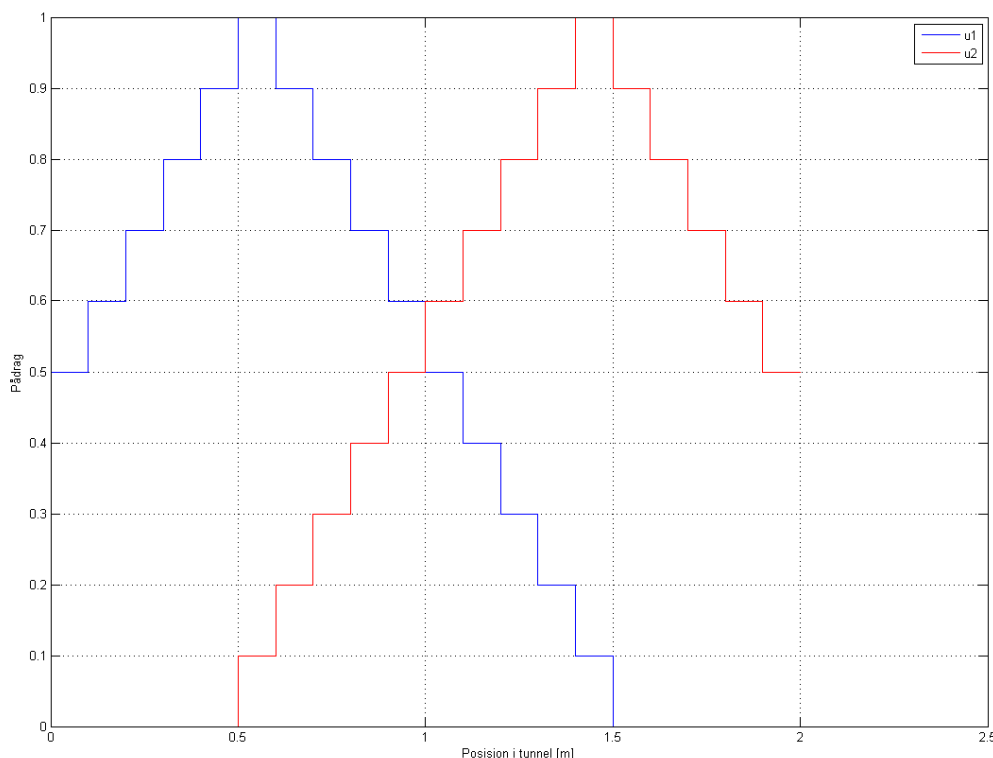
4.2.1 Pådrag

Tinetunnelen har to mikrobølgepådrag, u_1 og u_2 . Disse kan varieres. Mikrobølgeeffekten fra hvert element antas å være størst rett under elementet og bli mindre når man beveger seg vekk fra elementet. Mikrobølgene fra de to elementene vil spre seg utover i tunnelen. Det må derfor antas en spredning på mikrobølgene, slik at man vet hvor stor del av mikrobølgene som treffer kjøttblokkene på en gitt posisjon i tunnelen. Den antatte spredningen er vist i figur 4.2.

Som man kan se av figuren er det u_1 som gir effekt i første del av tunnelen, mens u_2 gir effekt i siste del. Ved inngangen tilfører u_1 kjøttet halvparten av mikrobølgene som den tilfører når kjøttet er rett under elementet. Det samme gjelder for u_2 ved utgangen av tunnelen. Det er også et område hvor u_1 og u_2 overlapper hverandre. Spredningen viser hvor stor fraksjon av mikrobølgene som vil treffe overflaten til kjøttet på den gitte posisjonen i tunnelen, i forhold til rett under elementene. Rett under elementene antas det at 100% av mikrobølgene fra elementene når overflaten til kjøttet. Dette er kun en antagelse av hvordan mikrobølgene fordeler seg i tunnelen og vites ikke eksakt. Denne spredningen vil være lik for over og undersiden av blokkene, siden det er to mikrobølgeelementer på undersiden og på oversiden. Det vil være vanskelig å påvirke spredningen, så det er mikrobølgeeffekten fra de to elementene som blir kontrollvariable for styring av pådraget.

4.2.2 Tilpasset penetrasjonsdybde

Penetrasjonsdybden som benyttes i simuleringene vil ikke variere mye med fettinnholdet i kjøttet, siden fettinnholdet ikke påvirker penetrasjonsdybden i stor grad. Derfor forenkles penetrasjonsdybden til kun å variere med temperatur. Som nevnt tidligere vil penetrasjonsdybden være ca konstant for temperaturer over frysepunktet, derfor velges den til 2,2 cm for temperaturer over -1.5°C . For å forenkle enda mer velges den til 15 cm for temperaturer mindre enn -1.5°C .



Figur 4.2: Mikrobølge spredning

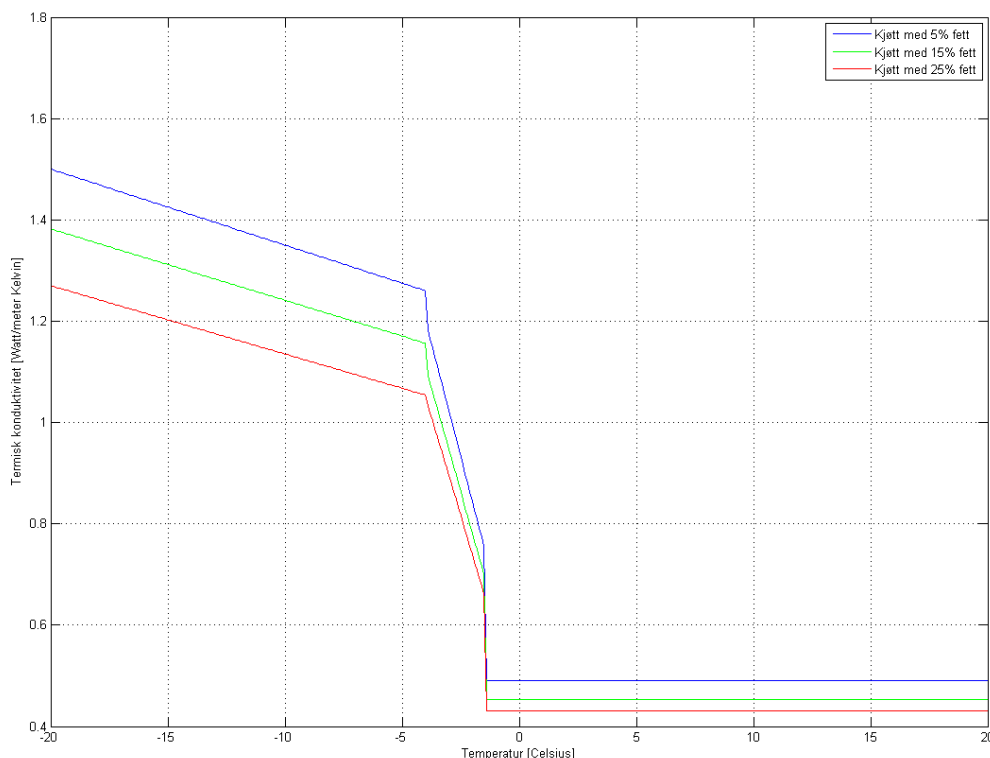
Dermed vil mikrobølgene penetrere hele kjøttblokken, som er ca 14 cm tykk, for temperaturer mindre enn -1.5°C .

4.2.3 Tilpasset termisk konduktivitet

Fra figur 3.2 vet man at konduktiviteten øker med økende isinnhold og minkende temperatur. Siden den termiske konduktiviteten er lavere hos fett enn hos magert kjøtt, er det naturlig å anta at den også vil være lavere for kjøtt med høyere fettprosent. Med dette som grunnlag er derfor følgende funksjon for den termiske konduktiviteten utviklet, med temperatur og fettprosent som variable. Denne er basert på figur 3.2, som gjelder for magert kjøtt, og tilpasset videre for kjøtt med ulikt fettinnhold.

$$\begin{aligned}
 k &= 0.49\left(1 - \frac{F}{100}\right) + 0.25\frac{F}{100}, \text{ for } T \geq -1.5^{\circ}\text{C} \\
 k &= 0.49\left(1 - \frac{F}{100}\right) + 0.25\frac{F}{100} - T\left(0.1775 - 0.09\frac{F}{100}\right), \text{ for } -4^{\circ}\text{C} \leq T \leq -1.5^{\circ}\text{C} \\
 k &= 1.25\left(1 - \frac{F}{100}\right) + 0.25\frac{F}{100} - T\left(0.015 - 0.06\frac{F}{100}\right), \text{ for } T \leq -4^{\circ}\text{C}
 \end{aligned} \tag{4.16}$$

Figur 4.3 viser plotet av funksjonen i ligning 4.16 og hvordan den er tilpasset figur 3.2. Denne beskrivelsen kan ikke garanteres å være helt nøyaktig, men fungerer godt til å vise at den termiske konduktiviteten varierer med temperatur og fettinnhold.



Figur 4.3: Termisk konduktivitet for kjøtt med ulikt fettinnhold

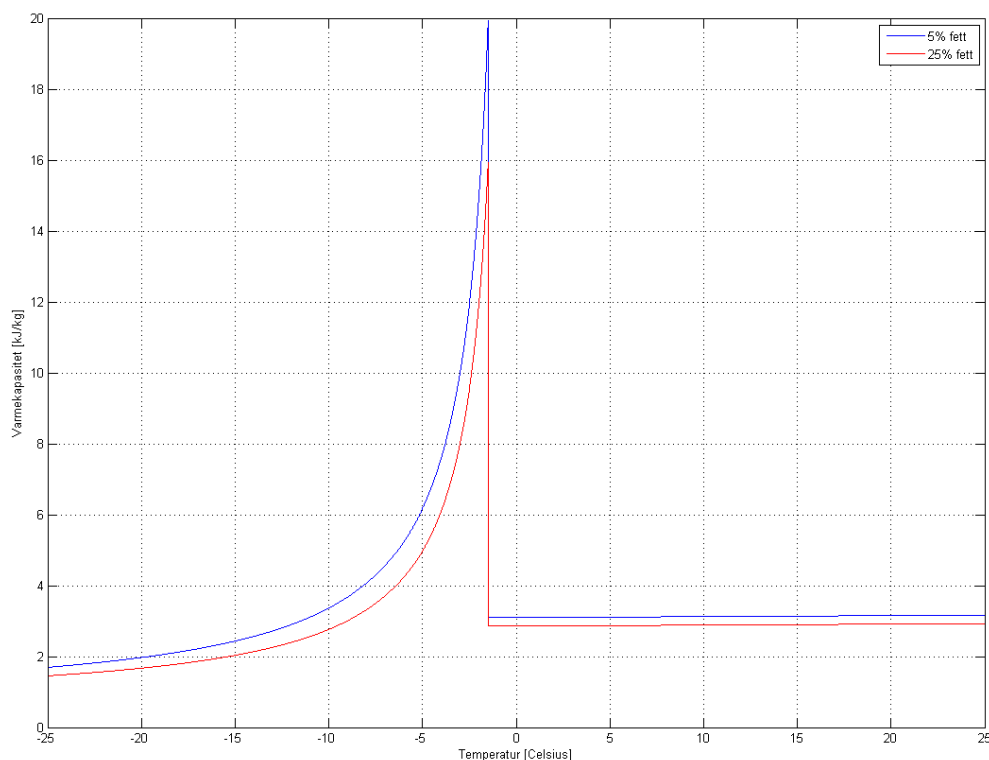
4.2.4 Tilpasset varmekapasitet

For at kjøttblokkene skal ha riktig temperatur når de kommer ut av tunnelen, tilpasses varmekapasiteten. Fra virkeligheten vet man at kjøttet har en temperatur på ca -2°C . Pådraget fra mikrobølgeelementene var på halv styrke, 375000 W , og spredningen av mikrobølgene presentert i forrige seksjon ble benyttet for å finne riktige verdier på varmekapasiteten. Grunnen til at pådraget ble satt til halv styrke var for å ha noe pådrag å gå på når det skal lages et optimaliserende system for tinetunnelen. Dersom varmekapasiteten ble tilpasset med maks pådrag ville ikke en optimaliserende funksjon for pådraget kunne hjelpet. Under justeringen av varmekapasiteten ble det tatt utgangspunkt i ligning 3.2. Varmekapasiteten for temperaturer over frysepunktet er temmelig godt kjent og ble derfor holdt som før, så det ble i hovedsak toppen som oppstår ved frysepunktet som ble senket. Også for temperaturer lavere enn frysepunktet ble varmekapasiteten senket litt. Verdien på varmekapasiteten ble også tilpasset slik at større fettinnhold ga lavere varmekapasitet, da fett har lavere varmekapasitet enn rent kjøtt. Verdiene for varmekapasiteten ble funnet ved prøving og feiling, ved å se på temperaturen i kjøttblokkene når de kom ut av tunnelen. Dette resulterte i følgende ligninger for varmekapasiteten.

$$\begin{aligned}
 C_p &= 0.59368 + \frac{30F}{T} + \frac{-94.8594 + 65.9102}{T} + \frac{2.3441}{T^2} \text{ kJ/kg, for } -25^{\circ}\text{C} \leq T \leq -1.5^{\circ}\text{C} \\
 C_p &= 0.00281T + 3.6198 - 1.2F - 0.4615 \text{ kJ/kg for } -1.5^{\circ}\text{C} < T
 \end{aligned}
 \tag{4.17}$$

Til forskjell fra ligning 3.2 tar ikke disse hensyn til vanninnholdet i kjøttet, men kun fett og

temperatur. Figur 4.4 viser varmekapasiteten til kjøtt med ulikt fettinnhold, som beskrevet av ligning 4.17.

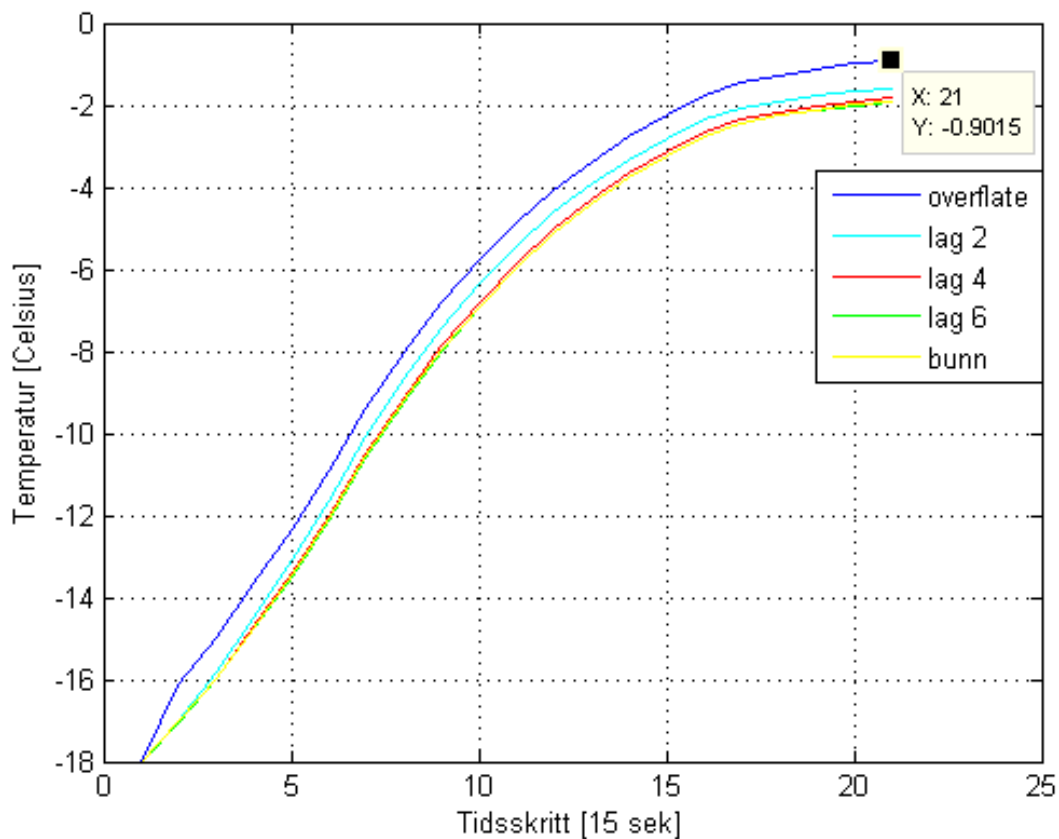


Figur 4.4: Varmekapasitet for kjøtt med 5% og 25% fett

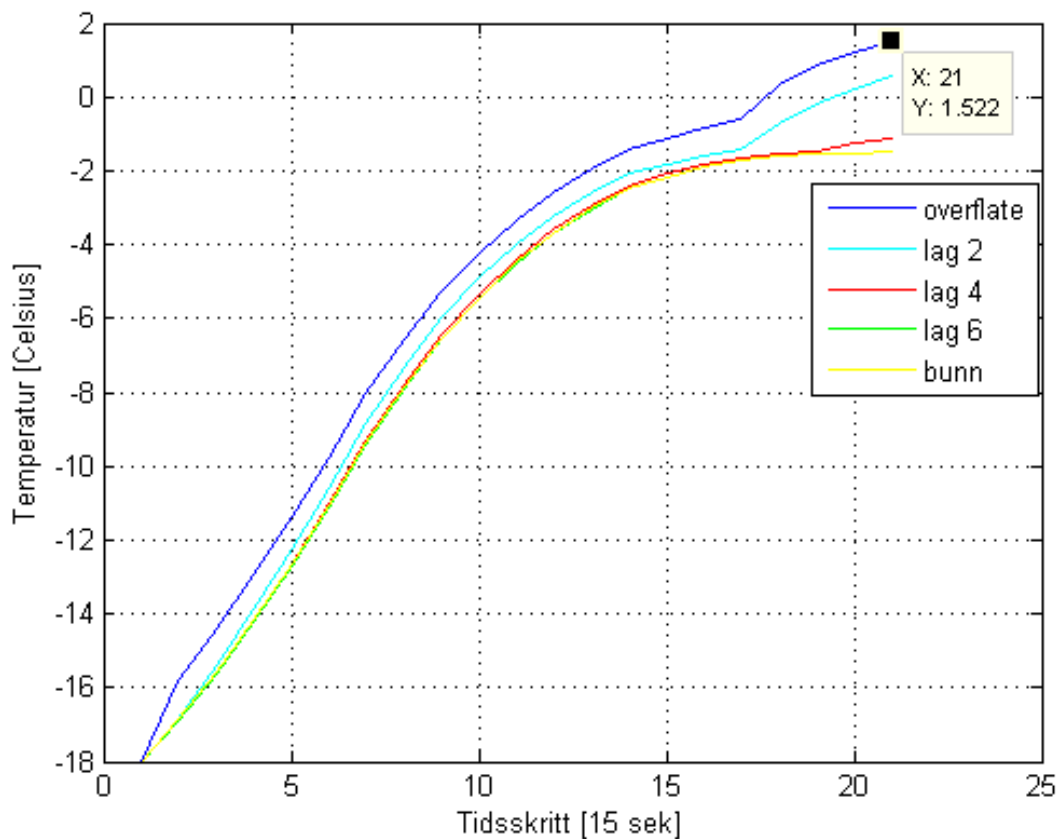
Det må påpekes at denne beskrivelsen av varmekapasiteten ikke nødvendigvis vil være korrekt i forhold til virkeligheten. Den er tilpasset slik at modellen skal passe til virkeligheten. Det som kan gjøres for å finne eksakt varmekapasitet, er å kjøre eksperimenter med tinetunnelen. I eksperimentene kan man måle tilført effekt og temperatur i kjøttblokkene, og så finne ut varmekapasiteten og eventuelt andre parametre.

4.2.5 Temperatur i kjøttblokkene

Varmekapasiteten ble tilpasset ut fra temperaturen i kjøttblokkene. Temperaturutviklingen i blokkene når de beveger seg gjennom tunnelen er vist i figur 4.5. Her vises kun temperaturen for halvparten av blokken, men den er antatt å være lik i begge halvdelene, siden det er mikrobølgeelementer på oversiden og undersiden. Pådraget var på halv styrke på begge elementene. Ut fra simulering av temperaturen ble varmekapasiteten tilpasset slik at kjøttet fikk ønsket slutttemperatur. Simuleringene viser temperaturen i en slice av en kjøttblokk som transporteres gjennom tunnelen. En slice vil typisk være 10 cm bred og en kjøttblokk vil bli delt opp i 5 slicer. Som man kan se av figur 4.5, ser man at høyere fettinnhold i kjøttet gjør at man trenger mindre energi for tining. Man ser at når kjøttet kommer ut av tunnelen har overflaten til kjøttsslicen med 25% fett en temperatur på ca 1.5°C. Overflaten til kjøttsslicen med 5% fett har derimot en temperatur på ca -0.9°C. For begge fettprosentene ser man også her at det nest nederste laget (lag 6) holder en lavere temperatur enn bunnen. Dette kommer som sagt av grensebetingelsen og har ingen stor betydning for modellen.



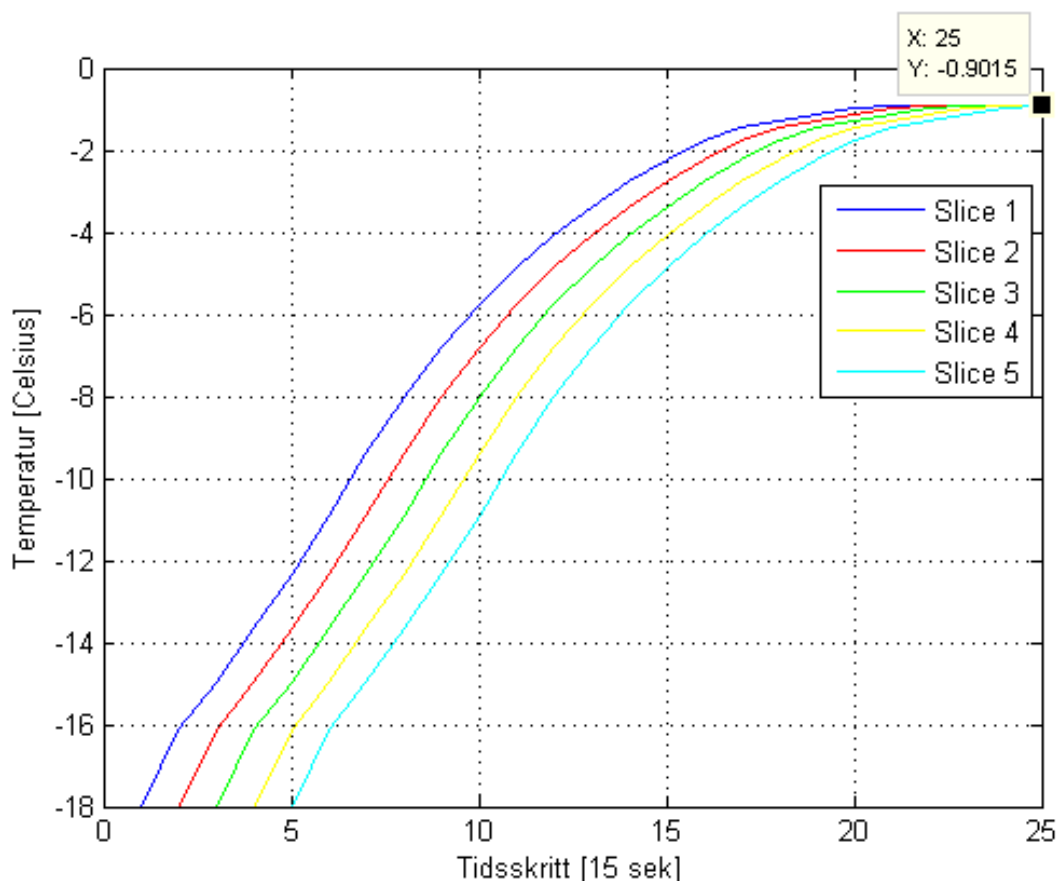
(a) 5% fett



(b) 25% fett

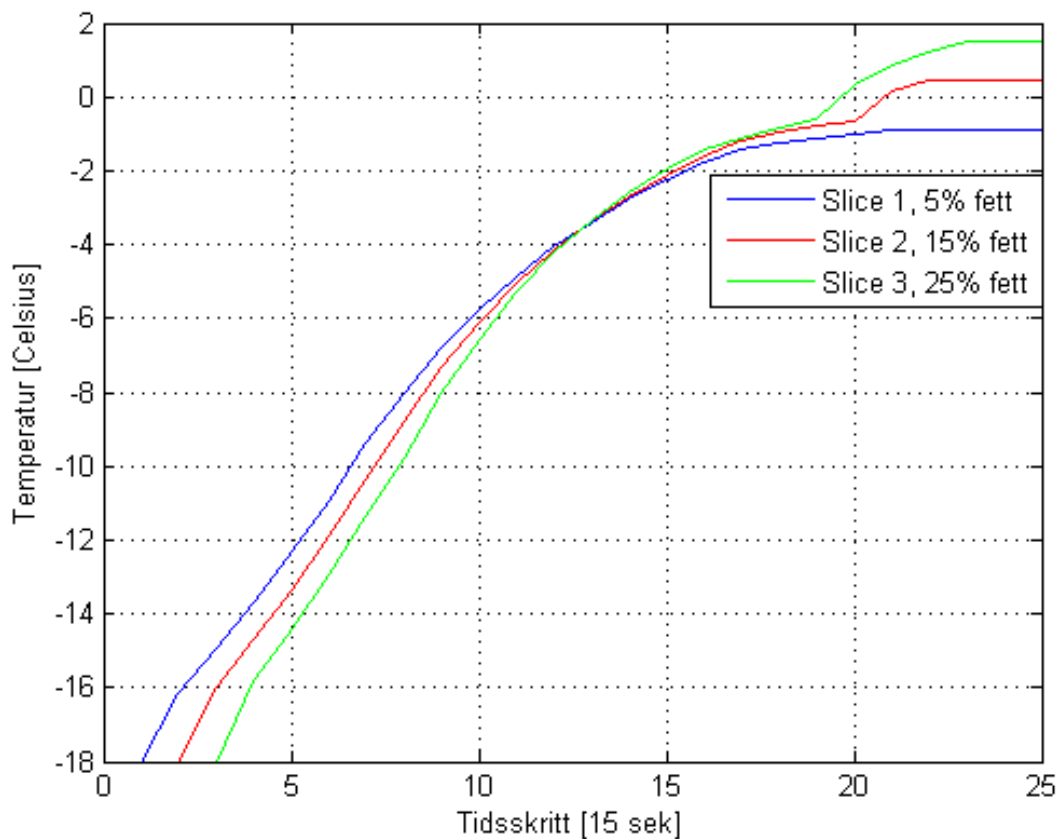
Figur 4.5: Temperaturutvikling for kjøtt med 5% og 25% fett

Man kan også se på hvordan temperaturen endrer seg i en hel kjøttblokk. Hver kjøttblokk består av 5 slicer. Disse vil entre tunnelen rett etter hverandre, og for å forenkle modellen antas det at det ikke vil være noe varmeutveksling mellom slicene. Figur 4.6 viser overflatetemperaturen i en kjøttblokk. Man ser at hver slice oppnår lik temperatur til slutt, men naturlig nok vil første del av blokken stige i temperatur før siste del. Man kan også se at de forskjellige slicene entrer tunnelen ved forskjellige tidssteg, som var en del av hensikten med modellen.



Figur 4.6: Overflatetemperatur for en hel kjøttblokk. 5% fett.

Det vil være naturlig å se på scenarioer hvor det kommer inn kjøttblokker med forskjellig fettinnhold. Det er dette som skjer i praksis, kjøttblokker med ulikt fettinnhold skal tines og kommer inn i tunnelen i tilfeldig rekkefølge. Figur 4.7 viser hvordan tining av kjøttblokker med ulikt fettinnhold vil skje med ulik hastighet. Figuren viser 3 slicer kjøtt, med henholdsvis 5%, 15% og 25% fett, som transporteres gjennom tunnelen. Som man ser har slicene med høyest fettinnhold hurtigere temperaturøkning enn de med lavere fettinnhold. Dette kommer naturlig nok av forskjell i varmekapasiteten og termisk konduktivitet til kjøtt med ulikt fettinnhold. Ved denne simuleringen ble det også brukt halv effekt på mikrobølgene.



Figur 4.7: Overflatetemperatur for kjøttstlicer med ulikt fettinnhold

De foregående plottene hadde til intensjon å vise at modellen for tinetunnelen fungerer, og tar hensyn til naturlig variasjon i kjøttblokkene. Det ble vist at fettinnholdet har betydning for hvor mye energi som må tilføres kjøttblokkene for at de skal få ønsket temperatur. Denne modellen vil ikke være helt eksakt i forhold til virkeligheten, men fungerer til sitt formål, modellbasert styring av tinetunnelen.

Kapittel 5

Optimaliserende system

Mikrobølgepådragene i tinetunnelen skal optimaliseres. I denne seksjonen utvikles det et system som optimaliserer pådragene. Det er få eller ingen målinger som kan benyttes for tilbakekoblet styring av pådragene. Det fokuseres derfor på åpen sløyfe styring, altså at en modell av prosessen benyttes for å optimalisere mikrobølge pådragene. Modellen som benyttes er den som ble utviklet i forrige seksjon for totalsystemet. Det antas at fettinnhold til hver blokk og ankomsttiden til hver blokk inn til tunnelen er kjent.

5.1 Pseudokode

For å vise ved enkelhet hvordan optimaliseringen av pådragene skjer, settes det opp en pseudokode for det optimaliserende systemet.

1. Velg sekvens av pådrag $Q : [q_{1,k}, q_{1,k+1} \dots q_{1,k+N}, q_{2,k}, q_{2,k+1} \dots q_{2,k+N}]$
2. Kall `fmincon`: $X = \text{FMINCON}(\text{FUN}, X_0, A, b)$

Her er `FUN` en funksjon som skal minimeres. `FUN` tar inn en vektor x , simulerer temperaturen i kjøttet gjennom tunnelen og returnerer en skalar F . X_0 vil være startverdier til pådraget q_1 og q_2 ved alle tidssteg. Altså $2 \times N$ pådrag hvor N er antall tidssteg. A og b er matriser som uttrykker lineære ulikheter, $Ax \leq b$. Det vil i denne betydningen være begrensninger på pådragene ved hvert tidssteg. Pådragene kan kun være en fraksjon mellom 0 og 1. Når `fmincon` har funnet den minste verdien for skalar F , som returneres fra `FUN`, er de optimale pådragene også funnet. `fmincon` returnerer så pådragene i vektoren X .

`fmincon` vil kjøre simuleringer av funksjonen `FUN` mange ganger før den finner det optimale pådraget. Det betyr at temperatursimuleringen for kjøtt som transporteres gjennom tunnelen vil gjøres mange ganger for å finne det optimale pådraget. Dette systemet realiseres så i Matlab.

For å bli kjent med systemet og se hvordan det fungerer, velges det å først benytte kun en kjøttskive som skal tines. Etter at systemet er utviklet for en skive, vil det utvides til å ta hensyn til flere kjøttblokker etter hverandre.

5.2 Forenklet optimaliserende system

Når det kun er en skive som skal tines vil antall tidssteg N være lik 20, siden det tar 20 tidssteg å transportere en skive gjennom tunnelen. Dermed vil det være $2 \times 20 = 40$ pådrag som må bestemmes. Sekvensen av pådrag velges til følgende: $Q : [q_{1,k}, q_{1,k+1} \dots q_{1,k+20}, q_{2,k}, q_{2,k+1} \dots q_{2,k+20}]$ Begrensningene til pådraget er følgende:

$$\begin{bmatrix} q_{1,1} \\ q_{1,2} \\ \vdots \\ q_{1,20} \\ -q_{1,1} \\ -q_{1,2} \\ \vdots \\ -q_{1,20} \\ q_{2,1} \\ q_{2,2} \\ \vdots \\ q_{2,20} \\ -q_{2,1} \\ -q_{2,2} \\ \vdots \\ -q_{2,20} \end{bmatrix} \leq \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 1 \\ \vdots \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (5.1)$$

Grunnen til at pådraget er mellom 0 og 1 og ikke mellom 0 og 75000W er at f_{\min} fungerer best når begrensningene er definert som mellom 0 og 1. Pådraget vil dermed representere en fraksjon av 75000W. Dette er implementert i Matlab koden. Begrensningene kan uttrykkes som følgende matriser:

$$AQ \leq b \quad (5.2)$$

Dimensjonene til A matrisen og b matrisen er hhv. 80×40 og 80×1

$$A = \begin{bmatrix} 1 & 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 & 0 & 0 & \dots & 0 \\ -1 & 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ 0 & -1 & \dots & 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & -1 & 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & 0 & 0 & \dots & 1 \\ 0 & 0 & \dots & 0 & -1 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & 0 & -1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & 0 & 0 & \dots & -1 \end{bmatrix}, b = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 1 \\ \vdots \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (5.3)$$

Funksjonen FUN som benyttes av fmincon regner ut en objektfunksjon hver gang den blir kjørt. Det er objektfunksjonen som minimeres av fmincon for å finne optimale pådrag. Objektfunksjonen vil typisk være økende med økende pådrag, avvik fra ønskede tilstander osv. Man kan se på den som en kostfunksjon hvor det koster å bruke pådrag og avvike fra ønsket resultat. fmincon finner så den optimale bruken av pådrag ved å minimere objektfunksjonen. I objektfunksjonen kan man bestemme hva som skal tas mest hensyn til, bruken av pådrag eller å oppnå ønsket tilstandsverdi. Dersom det er viktigst å bruke minst mulig pådrag, kan man straffe bruk av pådrag mer enn man straffer avvik fra referanseverdi. For det forenklete systemet velges følgende objektfunksjon:

$$F = \alpha(T_{20,1}^{20} - T_{20,1}^{Set})^2 + \beta[(Q_1^k)^T(Q_1^k) + (Q_2^k)^T(Q_2^k)] \quad (5.4)$$

$T_{20,1}^{20} - T_{20,1}^{Set}$ er her overflatetemperaturen til kjøttskiven i det den forlater tunnelen, minus ønsket overflatetemperatur. $T_{20,1}^{Set}$ settes til -2°C siden kjøttet antas å være ferdig tint ved denne overflatetemperaturen. Når overflaten har nådd denne temperaturen vil resten av blokken også ha en tilfredsstillende temperatur. Q_1^k og Q_2^k er vektorer som inneholder de forskjellige pådragene ved hvert tidssteg. De er arrangert på samme måte som Q :

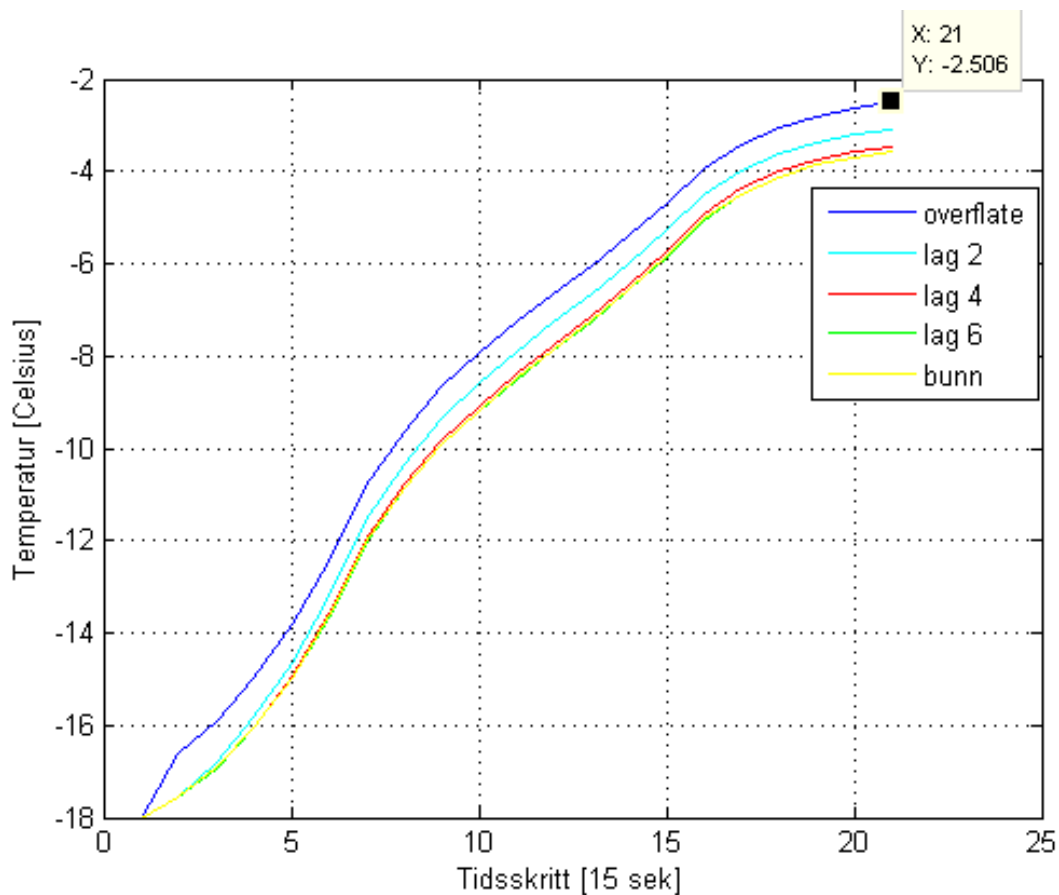
$$Q_1^k = \begin{bmatrix} q_{1,1} \\ q_{1,2} \\ \vdots \\ q_{1,20} \end{bmatrix} \quad Q_2^k = \begin{bmatrix} q_{2,1} \\ q_{2,2} \\ \vdots \\ q_{2,20} \end{bmatrix} \quad (5.5)$$

$q_{1,k}$ og $q_{2,k}$ er pådragene ved hvert tidssteg for de to mikrobølgeelementene. α og β er parametrene som bestemmer hvor mye bruk av pådrag og avvik fra referanse skal straffes. α her er ikke den samme som ble definert under stabilitetsanalysen. Det er viktig å vite at det er forholdet mellom α og β som er avgjørende, ikke selve verdien på de. Dersom de har lik verdi, straffes bruk av pådrag og avvik like mye. Dersom α er dobbelt så stor som β , straffes avvik fra referanse dobbelt så mye som bruk av pådrag. Man kan dermed bestemme hva fmincon skal legge størst vekt på under optimaliseringen.

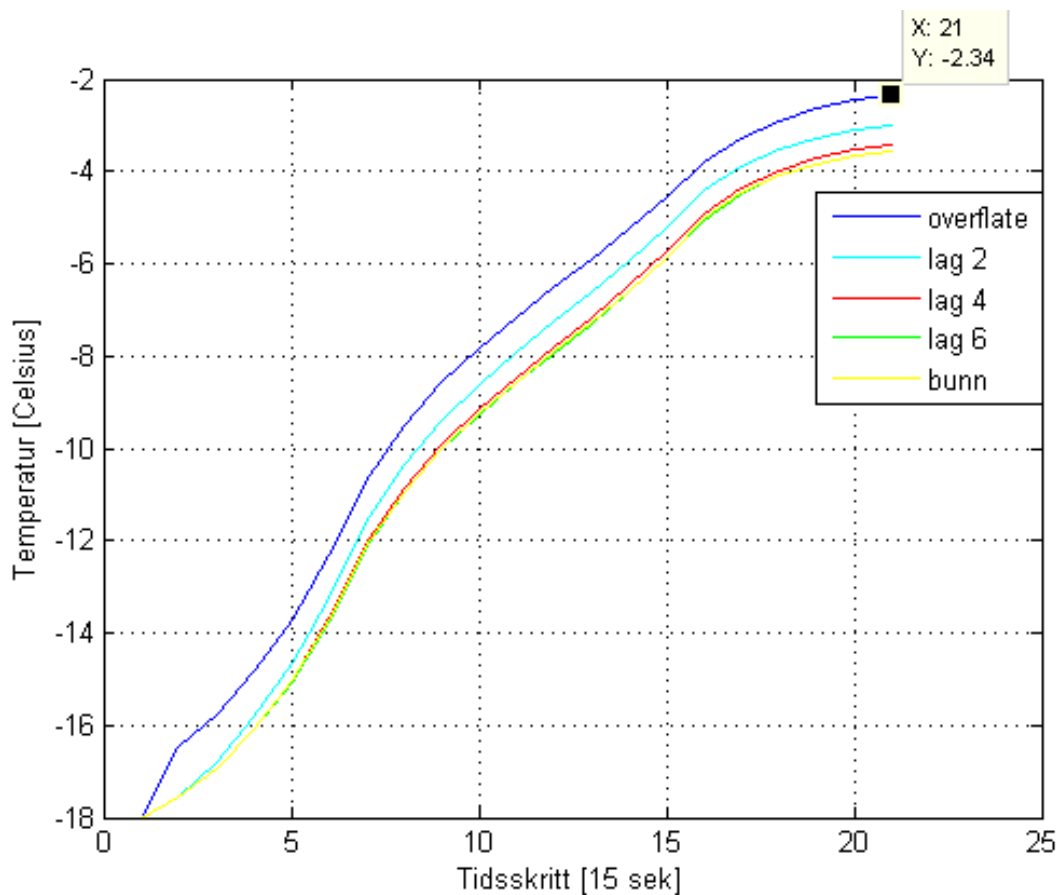
Matlab koden som ble benyttet for denne simuleringen er vedlagt. Se vedlegg A.1 for kode og forklaring. Det ble gjort to simuleringer med denne modellen hvor det ved første tidssteg kom en kjøttskive med hhv. 5% og 25% fett. Figur 5.1 viser temperaturutviklingen i kjøttskivene, og figur 5.2 viser de optimale pådragene som ble funnet ved hver av simuleringene.

Det man kan legge merke til ved temperaturene i de to kjøttskivene er at overflaten aldri når settpunktet på -2°C . Overflatetemperaturen når nesten settpunktet, men ikke helt nøyaktig. Dette kommer av at slik problemet er formulert til nå, vil det være optimalt å ikke nå settpunktet helt. Dette er pga av at det kun er definert ulikhetsbetingelser. Alle disse betingelsene begrenser pådraget, men det er ingen betingelser for temperaturen. Det er kun i objektfunksjonen at avvik fra settpunktet blir straffet. Det vil derfor være optimalt å havne litt unna settpunktet. For å treffe settpunktet må det innføres integralvirkning slik at man unngår dette avviket. Bortsett fra dette er temperaturutviklingen som tidligere.

Når det kommer til pådraget, ser man for begge kjøttskivene at u_1 er større enn 0 i de første 15 tidsstegene. Etter 15 tidssteg er kjøttskiven så langt vekk at u_1 ikke kan påvirke temperaturen i den. Derfor settes u_1 da lik 0. Det samme gjelder for u_2 , bare i de fem første tidsstegene. Både u_1 og u_2 er størst i det tidssteget hvor kjøttskiven er rett under elementene. Dette er naturlig, da u_1 og u_2 påvirker temperaturen mest når kjøttskiven er på denne posisjonen. Derfor får man også den karakteristiske formen på de to pådragene. Av pådraget ser man også at det brukes mindre pådrag når fettprosenten er 25% i motsetning til 5%. Dette er forklart tidligere og kommer hovedsakelig av forskjell i varmekapasiteten ved forskjellig fettprosent. Det som er litt forskjellig mellom u_1 og u_2 er at u_2 har en høyere topp, altså at u_2 er høyere på det høyeste enn u_1 . Samtidig er u_2 også lavere på det laveste. Dette gjelder uansett fettprosent. Grunnen

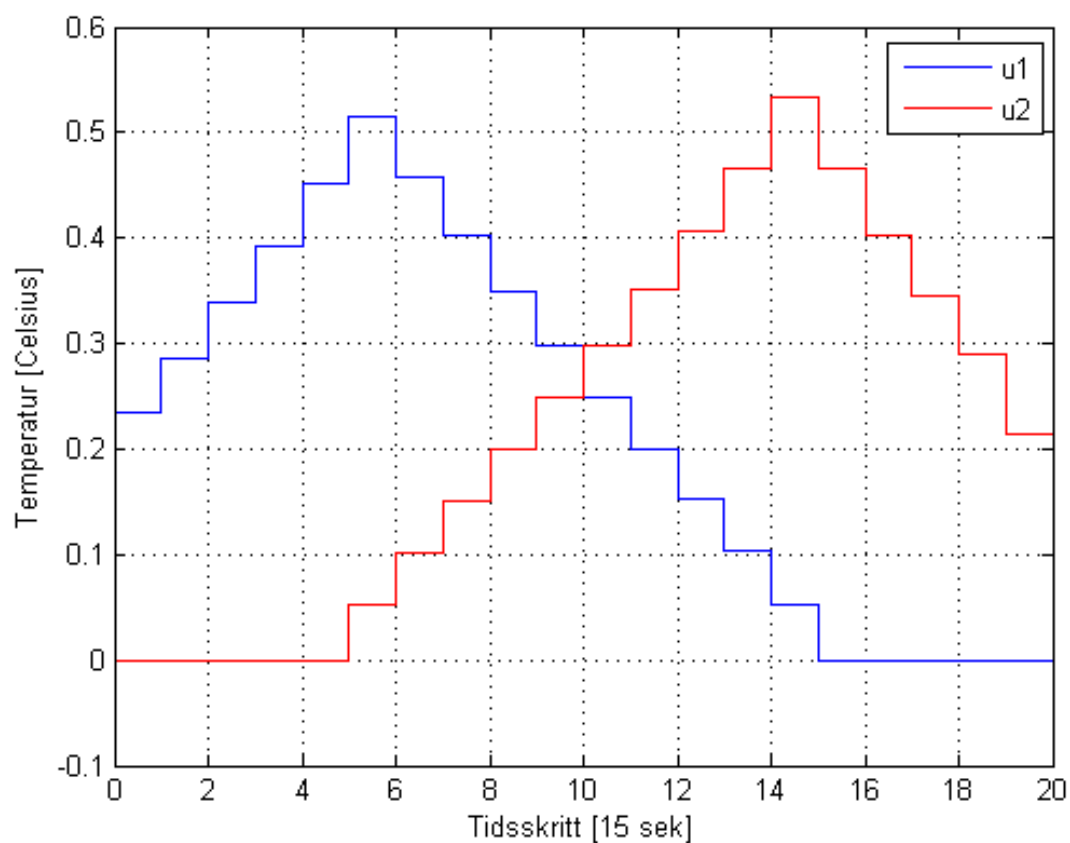


(a) 5% fett

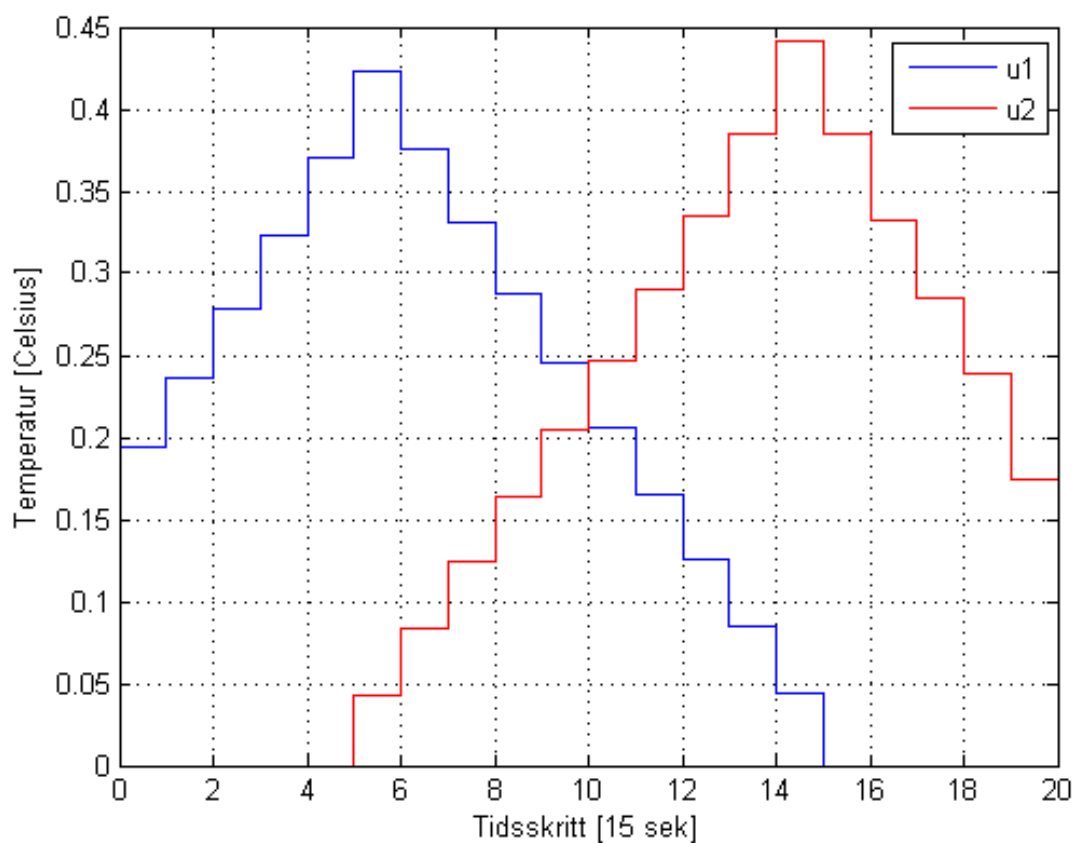


(b) 25% fett

Figur 5.1: Temperaturutvikling for kjøttskiver med hhv. 5% og 25% fett



(a) 5% fett



(b) 25% fett

Figur 5.2: Pådrag fra mikrobølgeelementer for kjøttskiver med hhv. 5% og 25% fett

til at u_2 varierer mer enn u_1 kan være at det er lettere å påvirke temperaturen når kjøttet nærmer seg enden av tunnelen. Med lettere menes det her at man trenger mindre pådrag for å øke temperaturen mot slutten av tunnelen enn ved starten, pga endrede fysiske forhold. Derfor benyttes kanskje u_2 mer enn u_1 for å regulere temperaturen.

5.2.1 Likhetsbetingelse for overflatetemperaturen

For at overflatetemperaturen skal nå settpunktet innføres det integralvirkning. `fmincon` har en egen funksjon for å gjøre akkurat dette. Ved å innføre en likhetsbetingelse som sier at overflatetemperaturen skal være lik settpunktet får man dermed den samme effekten som ved integralvirkning, og stasjonært avvik fjernes. I og med at det innføres likhetsbetingelse, må objektfunksjonen F også forandres. Den blir nå følgende:

$$F = [(Q_1^k)^T(Q_1^k) + (Q_2^k)^T(Q_2^k)] \quad (5.6)$$

Nå vil kun bruk av pådrag straffes i objektfunksjonen. Det er ingen grunn til å ta med avvik fra ønsket temperatur i objektfunksjonen, siden dette nå tas hånd om av likhetsbetingelsene.

Likhetsbetingelsen som innføres har følgende form: $ceq(x) = 0$ Det gir følgende likning for $ceq(x)$:

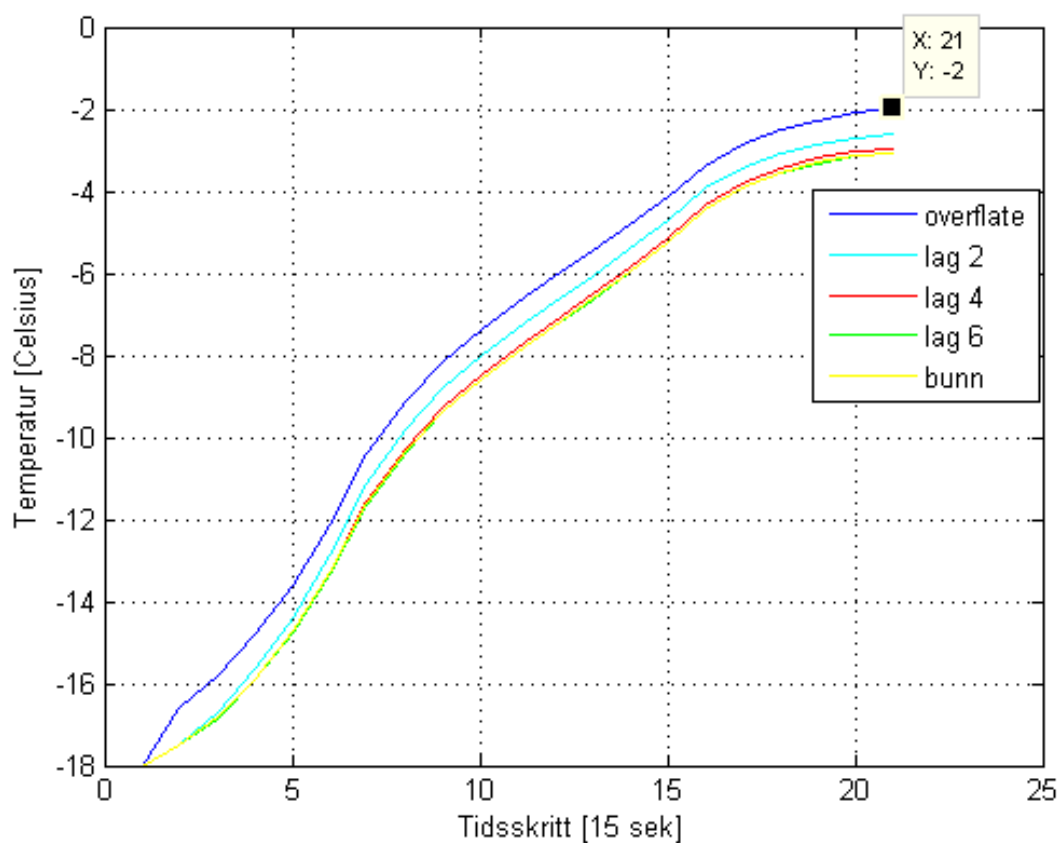
$$ceq(x) = T_{20,1}^{20} - T^{Set} \quad (5.7)$$

Denne ulikheten er ulineær mhp. temperaturen. Betingelsen gjør at overflatetemperaturen idet kjøttskiven går ut av tunnelen må være lik settpunktet. På samme måte som i `objektfunksjon.m` finner objektfunksjonen F , lages det en fil for å finne $ceq(x)$. Denne filen heter `ulin.m` og er forklart og lagt ved i vedlegg.

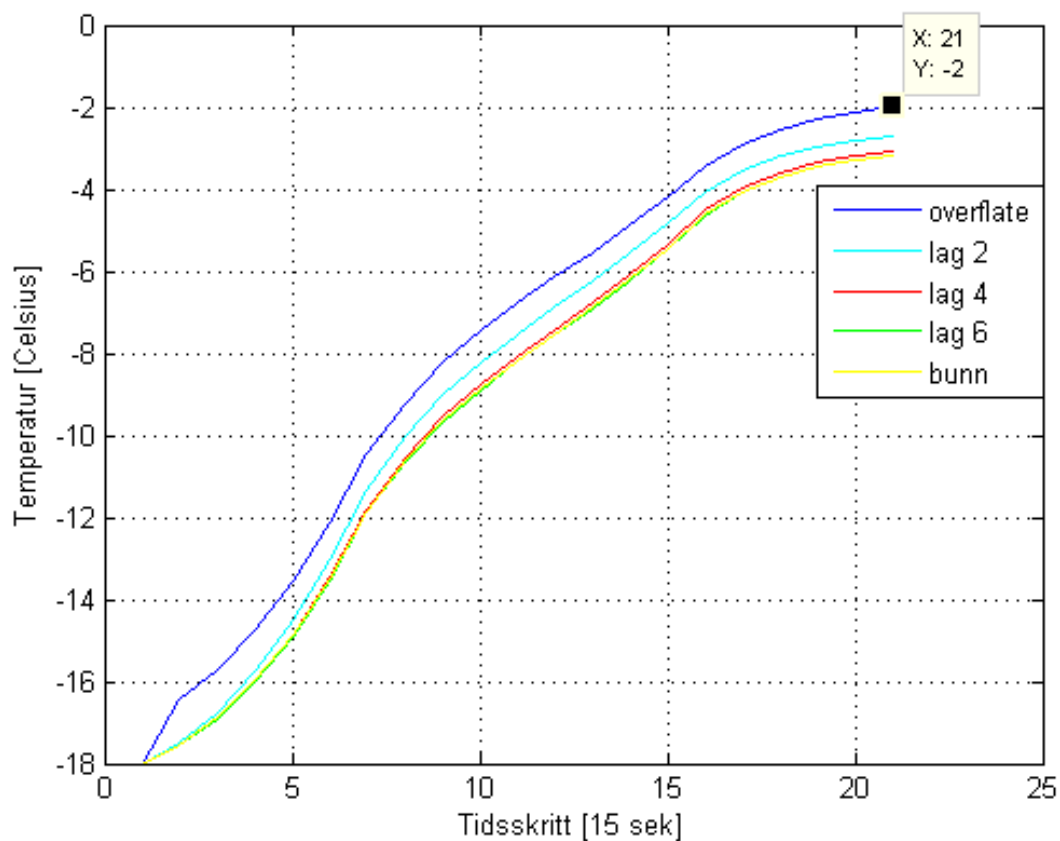
Dersom man nå simulerer en kjøttskive som transporteres gjennom tunnelen skal overflatetemperaturen til skiven nå settpunktet ved utgangen av tunnelen. Figur 5.3 og 5.4 viser hhv. temperaturutviklingen og pådrag når kjøttskiver med 5% og 25% fett transporteres gjennom tunnelen.

Som man ser i figur 5.3 blir nå overflatetemperaturen til kjøttskiven akkurat -2°C idet kjøttskiven kommer ut av tunnelen. Dette betyr at innføring av likhetsbetingelse har gjort det optimaliserende systemet bedre, med tanke på at ønsket effekt er oppnådd. Det viste seg derfor at man ikke kun kan belage seg på objektfunksjonen for å nå ønsket temperatur, men må innføre likhetsbetingelse ved bruk av `fmincon`. Ellers er temperaturutviklingen lik de foregående plottene og akkurat som forventet.

Pådraget som vises i figur 5.4 er litt større enn det som var før likhetsbetingelse ble innført (figur 5.2). Dette er veldig naturlig siden temperaturen generelt i blokken nå er noe høyere. Ellers ser man den samme karakteristiske formen på pådraget, og også nå er maksimum og minimum verdien til q_2 hhv. større og mindre enn q_1 .

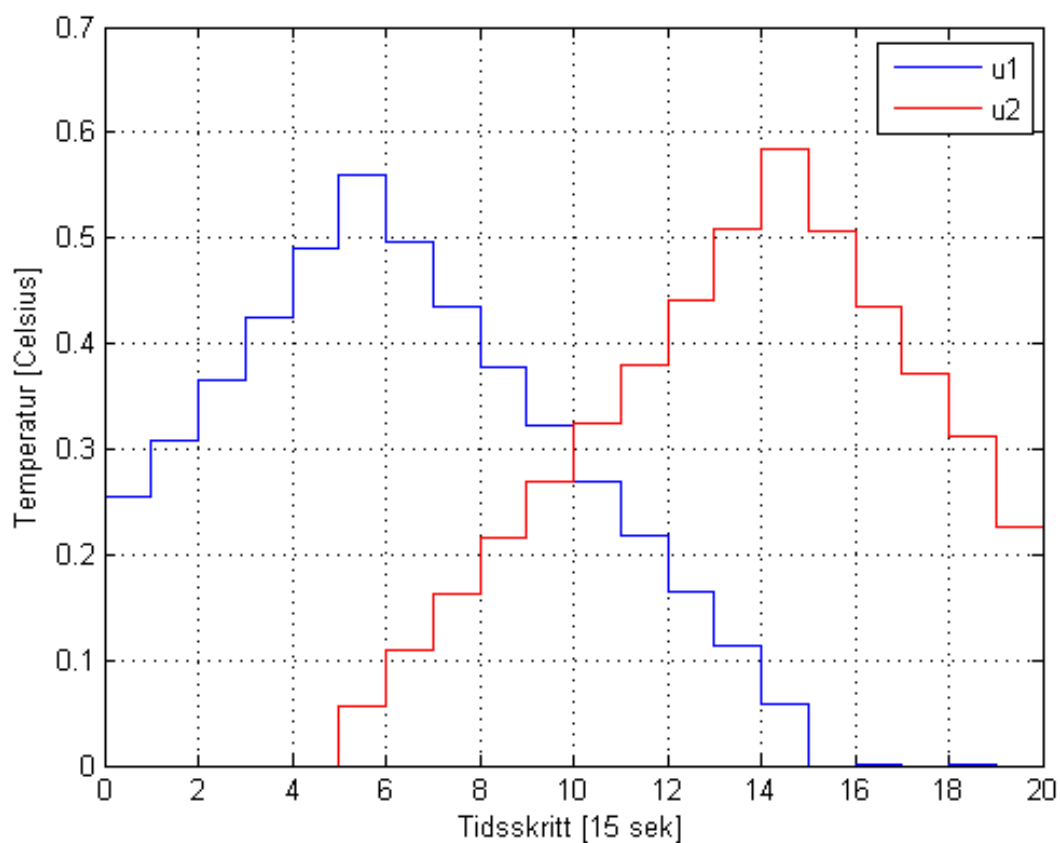


(a) 5% fett

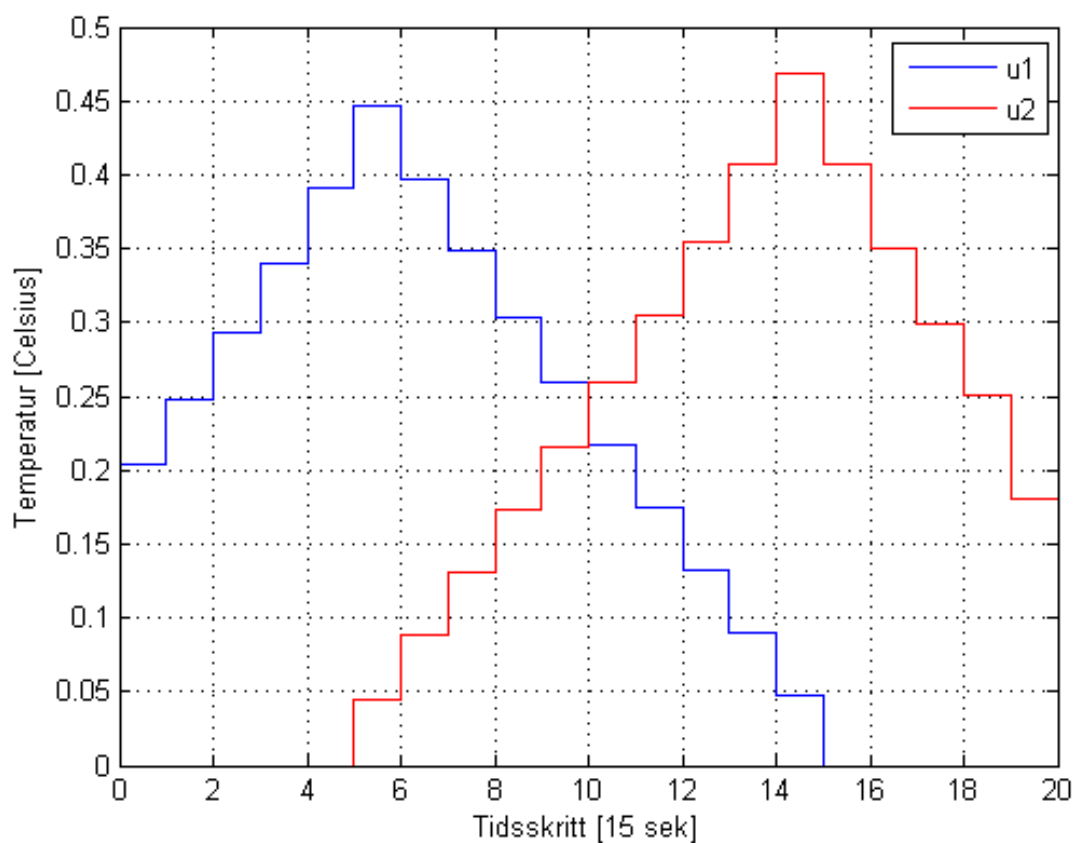


(b) 25% fett

Figur 5.3: Temperaturutvikling for kjøttskiver med hhv. 5% og 25% fett og likhetsbetingelse innført



(a) 5% fett



(b) 25% fett

Figur 5.4: Pådrag fra mikrobølgeelementer for kjøttskiver med hhv. 5% og 25% fett og likhetsbetingelse innført

5.3 Fullt optimaliserende system

Det har fram til nå handlet om simulering av en kjøttskive. I praksis vil det være mange kjøttblokker som skal tines. Som nevnt tidligere vil en kjøttblokk bestå av fem kjøttskiver etter hverandre. Det vil derfor være naturlig å se hvordan systemet oppfører seg når det kommer flere kjøttblokker inn i tunnelen. Dette vil være fokus i denne delen. Det vil bli sett på hvordan systemet fungerer når kjøttblokker med forskjellig fettprosent skal tines. Forskjellige metoder for optimalisering av pådraget vil også vurderes og sammenlignes.

Dersom det skal simuleres at flere kjøttblokker tines, må systemet utvides. Dimensjonen til de forskjellige matrisene må endres og pådraget må utvides. Sekvensen av pådrag vil ha samme form som for det forenklete systemet. Forskjellen er nå at det vil være flere pådrag siden det skal simuleres over lengre tid. Pådraget blir derfor slik $Q : [q_{1,k}, q_{1,k+1} \dots q_{1,k+N}, q_{2,k}, q_{2,k+1} \dots q_{2,k+N}]$

Begrensningene til pådraget er følgende:

$$\begin{bmatrix} q_{1,1} \\ q_{1,2} \\ \vdots \\ q_{1,N} \\ -q_{1,1} \\ -q_{1,2} \\ \vdots \\ -q_{1,N} \\ q_{2,1} \\ q_{2,2} \\ \vdots \\ q_{2,N} \\ -q_{2,1} \\ -q_{2,2} \\ \vdots \\ -q_{2,N} \end{bmatrix} \leq \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 1 \\ \vdots \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (5.8)$$

Dimensjonene til A matrisen og b matrisen er hhv. $4N \times 2N$ og $4N \times 1$

$$A = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & 0 & 0 & \cdots & 0 \\ -1 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ 0 & -1 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & -1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & \cdots & 0 & -1 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & -1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & -1 \end{bmatrix}, b = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 1 \\ \vdots \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (5.9)$$

Dette er veldig likt det forenklete systemet, bortsett fra at N nå er større. I det forenklete systemet var $N = 20$. Nå vil N variere med antall kjøttskiver som skal tines. N vil derfor være forskjellig i de ulike simuleringene, og matrisene vil også endre dimensjon ved forskjellige simuleringer. Det innføres også her likhetsbetingelser, slik at alle skivene skal oppnå ønsket overflatetemperatur. I tillegg til at N endres mellom de forskjellige simuleringene, vil også likhetsbetingelsene forandres. Likhetsbetingelsene skal kun være oppfylt på de tidspunktene det faktisk kommer kjøttskiver ut av tunnelen. Disse tidspunktene kan variere pga. av varierende mellomrom mellom kjøttblokkene. Likhetsbetingelsene vil fortsatt ha samme form som før: $ceq(x) = 0$, men det vil nå være flere likhetsbetingelser på ulike tidspunkt. Det gir følgende liking for $ceq(x)$:

$$ceq(x) = \begin{bmatrix} ceq(1) \\ ceq(2) \\ ceq(3) \\ \vdots \\ ceq(k) \end{bmatrix} = \begin{bmatrix} T_{20,1}^{20} - T^{Set} \\ T_{20,1}^{20+1} - T^{Set} \\ T_{20,1}^{20+2} - T^{Set} \\ \vdots \\ T_{20,1}^{20+k} - T^{Set} \end{bmatrix} \quad (5.10)$$

Det er her gitt at det ved tidspunkt k faktisk kommer en kjøttskive ut av tunnelen. Dersom det ikke er slik, settes $ceq(k) = 0$. Dette kan endres mellom de forskjellige simuleringene og er avhengig av når kjøttskivene går ut av tunnelen. Dette utdypes ikke mer her, men i vedlagt kode ser man hvordan det er implementert. Det som er hovedpoenget er at det vil være likhetsbetingelser for å oppnå ønsket temperatur i alle kjøttblokkene.

Objektfunksjonen som skal minimaliseres er den samme som tidligere:

$$F = [(Q_1^k)^T (Q_1^k) + (Q_2^k)^T (Q_2^k)] \quad (5.11)$$

Nå inneholder derimot Q_1 og Q_2 flere pådrag, siden N ikke lengre er 20.

$$Q_1^k = \begin{bmatrix} q_{1,1} \\ q_{1,2} \\ \vdots \\ q_{1,N} \end{bmatrix}, Q_2^k = \begin{bmatrix} q_{2,1} \\ q_{2,2} \\ \vdots \\ q_{2,N} \end{bmatrix} \quad (5.12)$$

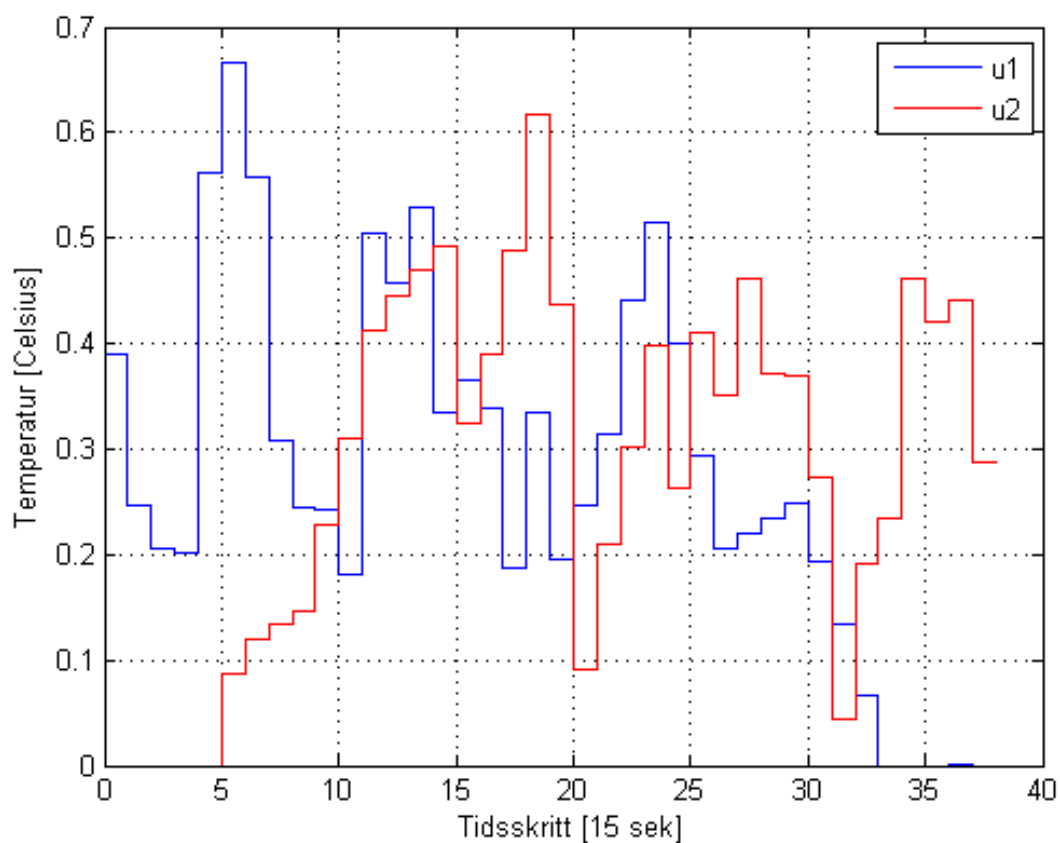
Dette systemet ble implementert i Matlab. Mye av forarbeidet ble gjort når det forenklete systemet ble implementert, men det måtte likevel utvides litt.

5.3.1 Programmering og testing

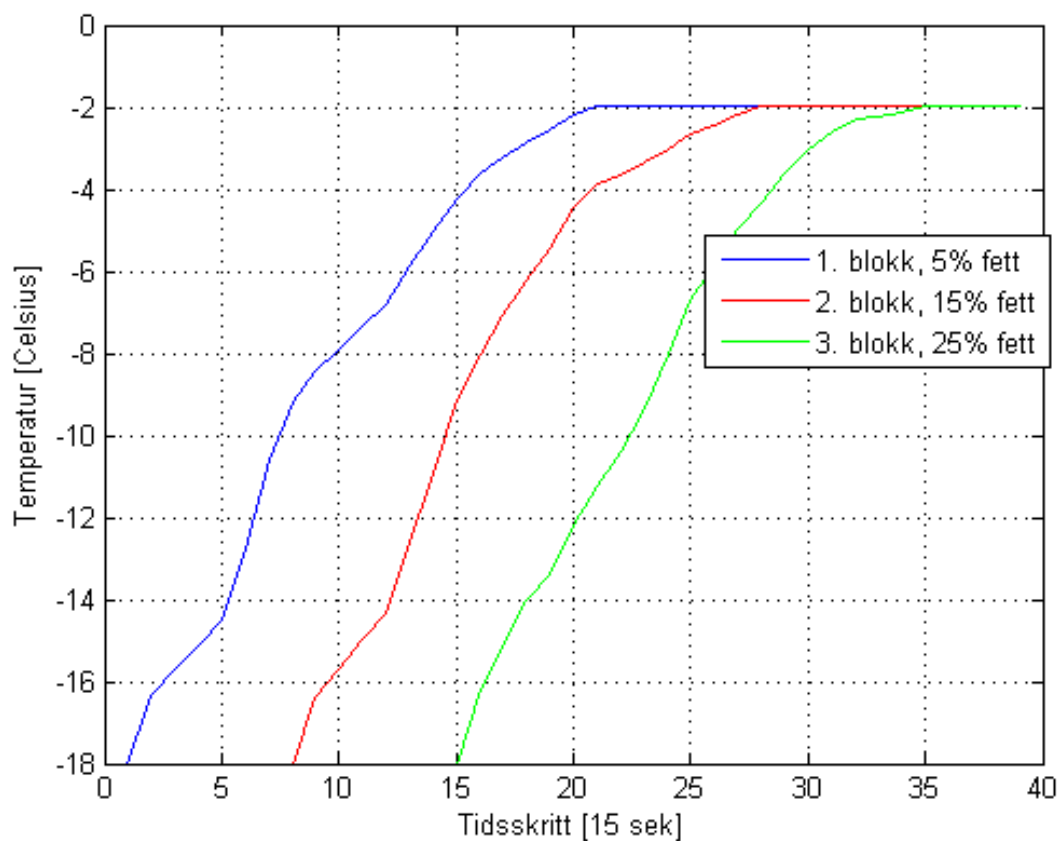
Under programmeringen av det fullt optimaliserende systemet ble det gjort en interessant oppdagelse. Ved en testsimulering, hvor det var 4 kjøttskiver på rad som skulle tines, ble følgende oppdaget: Dersom fettprosenten til de 4 kjøttskivene var veldig forskjellige, kunne ikke likhetsbetingelsen nås for alle kjøttskivene. Simuleringen stoppet derfor opp, uten å finne et optimalt pådrag. Det antas at skjedde fordi den store forskjellen i fettprosenten gjorde at noen kjøttskiver enten måtte ha lavere temperatur enn ønsket, eller at andre måtte ha høyere temperatur, med det tilgjengelige pådraget. Siden likhetsbetingelsene ikke kunne bli nådd, terminerte programmet. De 4 skivene hadde følgende fettprosent: 5%, 25%, 15%, 5%. Dette var ikke et realistisk scenario, fordi en kjøttblokk består av 5 skiver med lik fettprosent. Likevel kunne dette tyde på at dersom det er mange blokker i tunnelen med forskjellig fettprosent, kan det være vanskelig, om ikke umulig, å nå referanse temperaturen for hver kjøttblokk, spesielt om blokkene ligger tett inntil hverandre. Denne hypotesen ble så testet ut. Følgende testsett ble benyttet: 3 kjøttblokker på hhv. 5, 15 og 25 prosent fett. Dette tilsvarer 15 kjøttskiver. Alle skivene entret tunnelen rett etter hverandre slik at det ikke var noe mellomrom mellom blokkene. Siden det var 15 kjøttskiver, måtte det simuleres fram til at den siste skiven kom ut av tunnelen, altså $N = 35$. Utfallet av denne simuleringen ble følgende: Maximum number of function evaluations exceeded; increase OPTIONS.MaxFunEvals. Det betyr at det ikke var mulig å oppnå ønsket overflate temperatur i de tre blokkene. Altså stemte hypotesen om det kunne bli vanskelig å oppnå ønsket temperatur dersom fettprosenten i blokkene endret seg mye. Det ble derfor interessant å finne ut hva som måtte til for at de tre blokkene skulle få ønsket temperatur. Det ble så testet ut hvilken avstand det måtte være mellom blokkene for at det skulle fungere. Etter mye prøving og feiling ble det funnet ut at dersom det var 2 mellomrom mellom blokkene kunne referansen nås. Med et mellomrom menes her den lengden en kjøttskive har, altså ca 10 cm. Da det ble simulert med to mellomrom, 20 cm, mellom alle blokkene, ble følgende optimale pådrag oppnådd. Se figur 5.5 for pådrag og overflatetemperatur for blokkene. Det måtte nå simuleres 39 tidssteg så $N = 39$.

I figur 5.5 ser man at alle tre kjøttblokkene når ønsket overflatetemperatur. Her vises overflatetemperaturen til den første skiven i hver blokk. Siden det er likhetsbetingelser, vil også de andre skivene i blokken ha den samme temperaturen selv om disse ikke vises i figuren. Pådraget varierer veldig mye for hvert tidssteg og det er vanskelig å forklare hvorfor det akkurat blir slik. Man kan likevel se at u_2 ikke slås på før etter 5 tidssteg, som kommer av at u_2 ikke kan påvirke temperaturen før den første blokken har beveget seg i 5 tidssteg. u_1 slås helt av etter 33 tidssteg, for da kan ikke u_1 påvirke temperaturen i den siste blokken lengre.

I tillegg til kjøttblokker med ulik fettprosent ble det også gjort simuleringer på kjøttblokker med lik fettprosent. Disse kjøttblokkene kunne sendes inn uten noe mellomrom. Dette pga at de trenger det samme pådraget for å oppnå en gitt temperatur. Kjøttblokker med ulik fettprosent trenger derimot ulikt pådrag. Konklusjonen ut fra dette er derfor at dersom fettprosenten til kjøttblokkene varierer mye, bør de ikke sendes inn i tunnelen rett etter hverandre dersom overflatetemperaturen skal oppnå en bestemt verdi. Mellomrommet det bør være mellom blokkene vil variere med fettprosenten, og det er vanskelig å gi en eksakt avstand kun basert på disse eksperimentene. Ved tining av kjøtt kan det derfor være fornuftig å tine kjøttblokker med relativt lik fettprosent samtidig, og ikke blande blokker med høy og lav fettprosent.



(a) Optimalt pådrag



(b) Overflatetemperatur til første kjøttskive i hver blokk

Figur 5.5: Optimalt pådrag og overflatetemperatur for kjøttblokker med hhv 5%, 15% og 25% fett. Med 2 opphold mellom hver blokk.

5.3.2 Ulikhetsbetingelser for overflatetemperatur

I metoden brukt fram til nå har det handlet om at overflatetemperaturen skulle bli eksakt -2°C . Denne likhetsbetingelsen gjør hele problemet vanskeligere enn om det kun er ulikhetsbetingelser som skal oppfylles. Dersom det gjelder å få tinet så mye kjøtt som mulig på kortest mulig tid er det kanskje ikke ønskelig med mye ubrukt plass i tunnelen. Da finnes det derimot andre muligheter. Det vil derfor innføres ulikhetsbetingelser for overflatetemperaturen. Da kan man kreve at overflatetemperaturen skal være større enn en gitt verdi. Problemet som før var umulig å løse kan da forhåpentligvis løses, og man slipper mye mellomrom mellom blokkene.

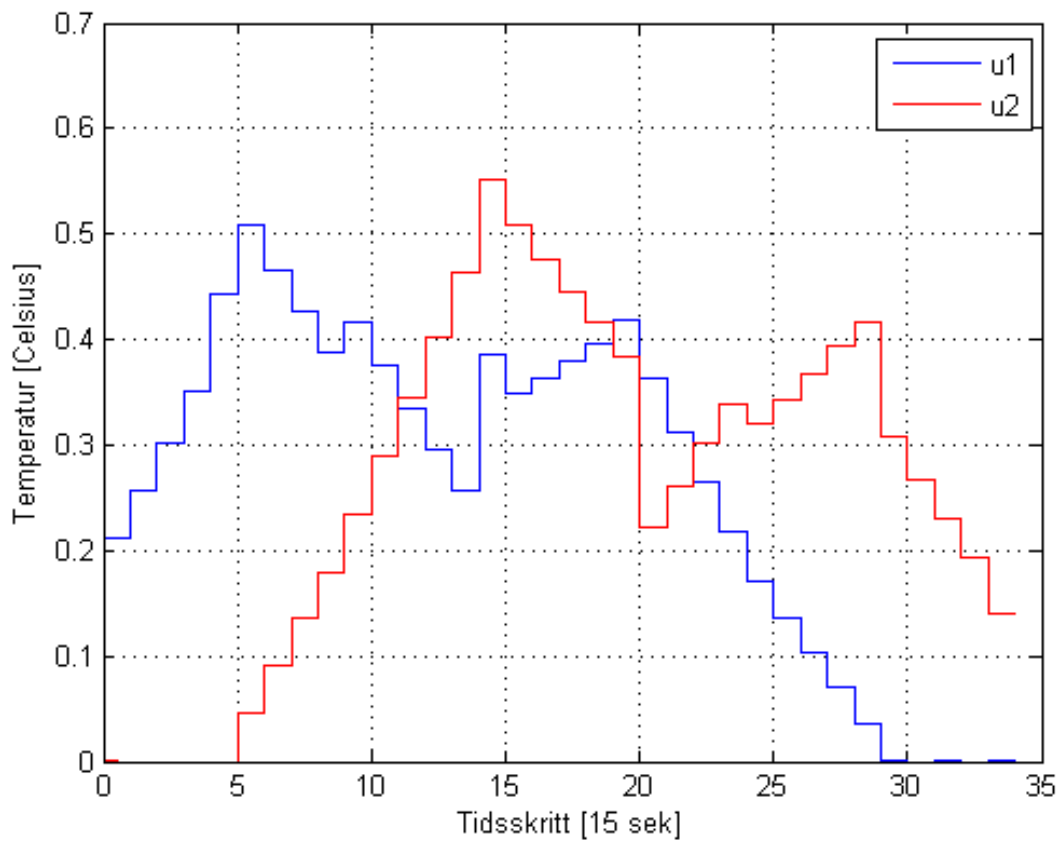
Ulikhetsbetingelsene vil være de samme som likhetsbetingelsene, bortsett fra at temperaturen nå skal være større eller lik ønsket settpunkt, $c(x) \leq 0$

$$c(x) = \begin{bmatrix} c(1) \\ c(2) \\ c(3) \\ \vdots \\ c(k) \end{bmatrix} = \begin{bmatrix} T^{Set} - T_{20,1}^{20} \\ T^{Set} - T_{20,2}^{20} \\ T^{Set} - T_{20,3}^{20} \\ \vdots \\ T^{Set} - T_{20,k}^{20} \end{bmatrix} \quad (5.13)$$

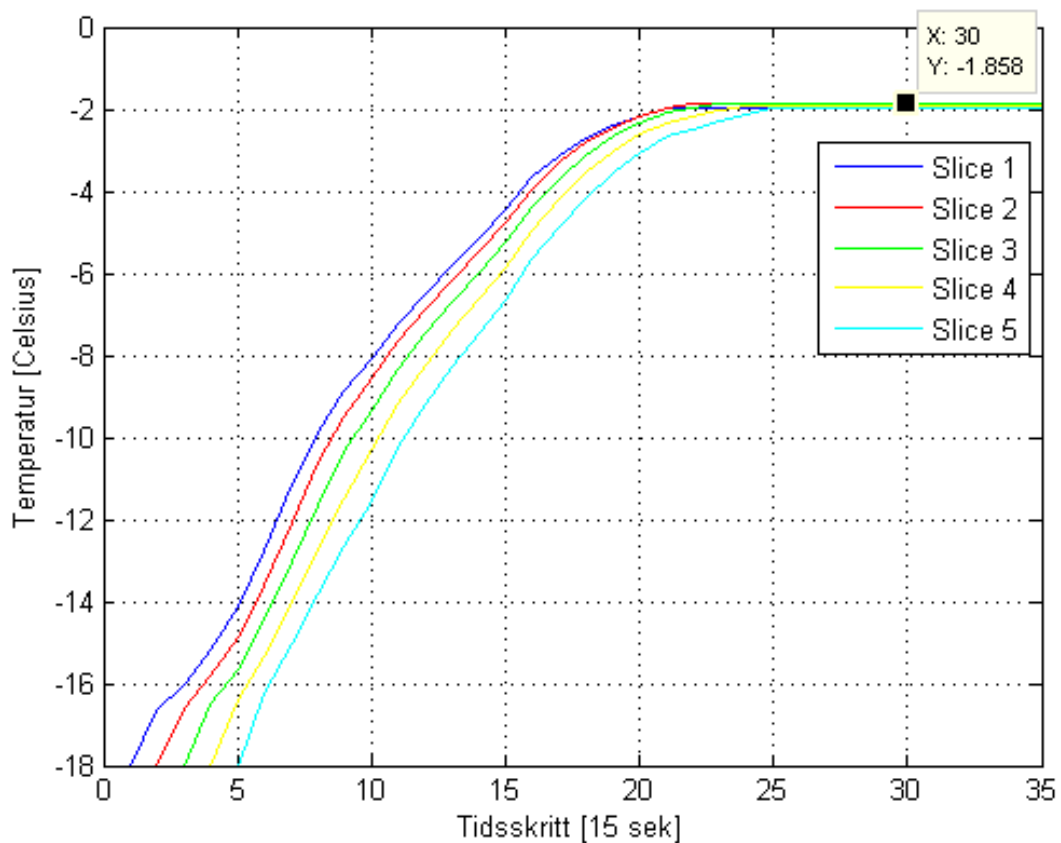
Også her er det gitt at det ved tidspunkt k faktisk kommer en kjøttskive ut av tunnelen. Dersom det ikke er slik, settes $c(k) = 0$. Som før kan dette endres mellom forskjellige simuleringene og er avhengig av når kjøttskivene går ut av tunnelen.

Siden overflatetemperaturen nå skal være større enn en gitt verdi vet man ikke hvor mye større den vil bli. Det som kan være faren ved en slik metode er at overflaten blir for varm. Dersom det er stor forskjell i fettprosenten til kjøttblokkene kan det være at noen blokker får for høy overflatetemperatur. Dette kommer fram i førstkommende simulering. En løsning kan da være å senke settpunktet. Man kan da for eksempel si at overflatetemperaturen skal være større enn $-2,5^{\circ}\text{C}$, dersom det er ønskelig. Det naturlig ville vært å sette settpunktet lik den laveste temperaturen overflatene kan ha for at blokken kan kvalifiseres som tint.

Systemet med ulikhetsbetingelser implementeres så i Matlab, se vedlegg B for kode. For å teste hvordan dette nå fungerer og om ønsket mål blir oppnådd velges det å simulere at tre kjøttblokker skal tines. Som før har de en fettprosent på hhv 5%, 15% og 25% og de entrer tunnelen i denne rekkefølgen. Siden hensikten skal være å unngå stort mellomrom mellom blokkene, velges det å simulere at blokkene kommer rett etter hverandre slik at det ikke blir noe mellomrom. Det er 15 skiver og N blir derfor lik 35. Etter denne simuleringen oppnår man følgende optimale pådrag og overflatetemperatur til de tre blokkene. Se figur 5.6 og 5.7. Ved å bruke ulikhetsbetingelser slipper man altså å ha mellomrom mellom blokkene. Figur 5.6 viser pådrag og overflatetemperatur til de fem skivene som utgjør den første blokken. Figur 5.7 viser overflatetemperaturen til de to siste blokkene. Også disse to består av fem kjøttskiver hver.

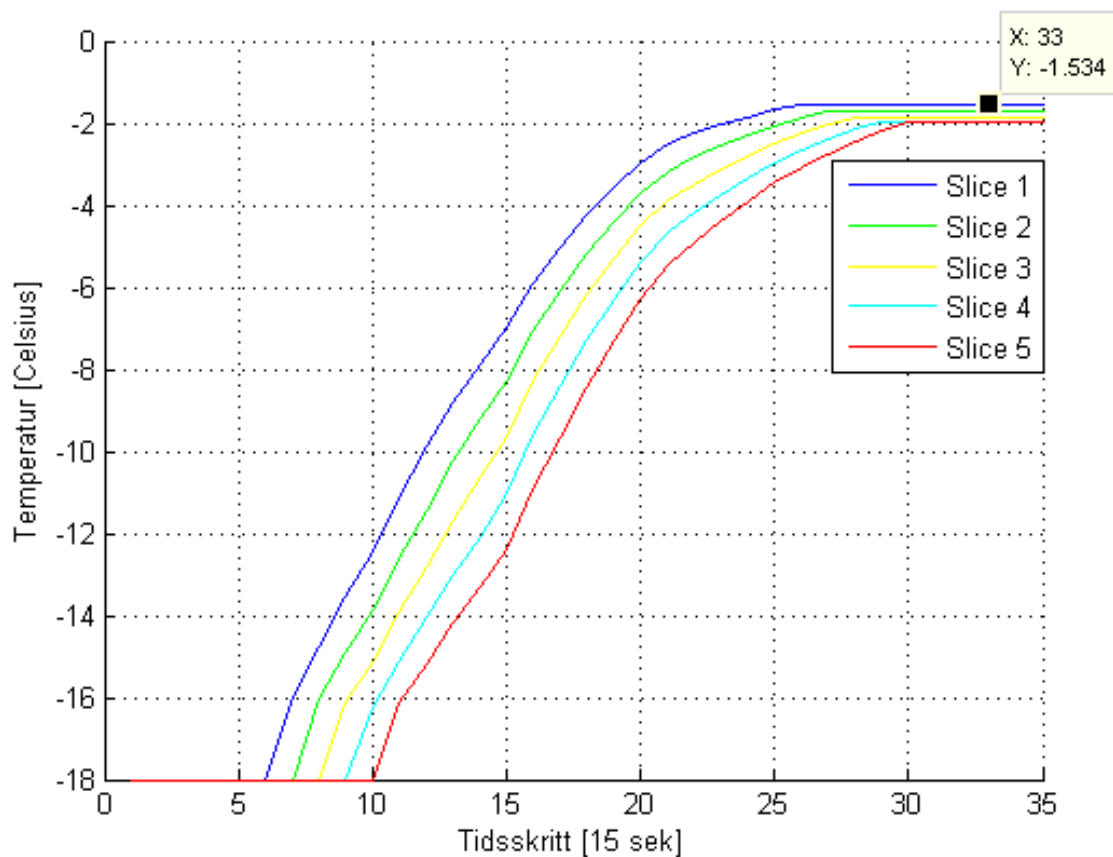


(a) Optimalt pådrag

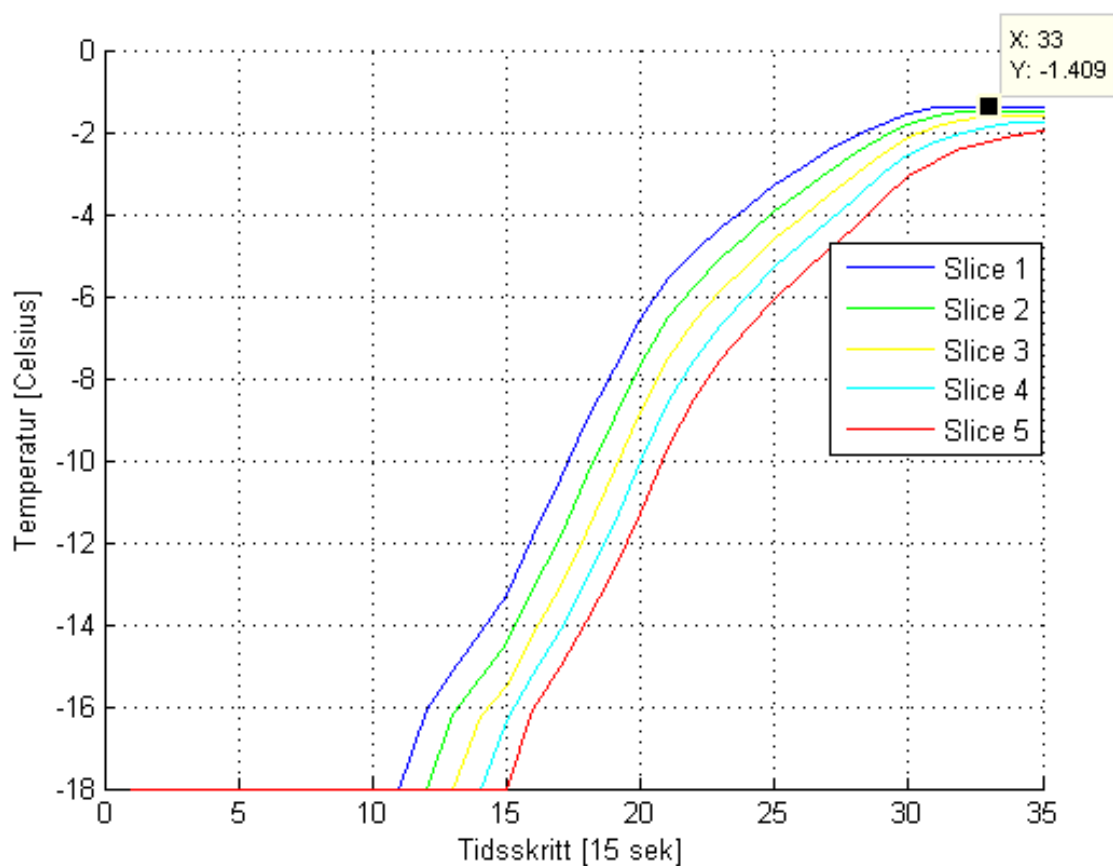


(b) Overflatetemperatur til alle skiver i første kjøttblokk, 5% fett

Figur 5.6: Optimalt pådrag og overflatetemperatur for kjøttblokk med 5% fett. Ikke mellomrom mellom blokker



(a) Overflatetemperatur til alle skiver i andre kjøttblokk, 15% fett



(b) Overflatetemperatur til alle skiver i tredje kjøttblokk, 25% fett

Figur 5.7: Overflatetemperatur for kjøttblokker med 15% og 25% fett. Ikke mellomrom mellom blokker

Som man kan se hos alle de tre blokkene, vil overflatetemperaturen til skivene som utgjør en blokk, være forskjellig. Overflaten til alle skivene har en temperatur høyere eller lik -2°C , som spesifisert i ulikhetsbetingelsene. På hver figur er den høyeste overflatetemperaturen i blokken spesifisert. Som man ser vil derfor temperaturen på overflaten variere mellom -2°C og -1.4°C . Grunnen til at hver skive i kjøttblokkene har forskjellig temperatur er at de blir utsatt for forskjellig pådrag. Pådraget forandrer seg ved hvert tidssteg og siden skivene også beveger seg opplever de aldri det samme pådraget.

Dersom man sammenligner det optimale pådraget ved denne metoden med det optimale pådraget som oppnås ved bruk av likhetsbetingelser, ser man at det er mye jevnere og mer systematisk enn når likhetsbetingelser benyttes. Ved likhetsbetingelser variere det kraftig ved nesten hvert tidssteg og virker uryddigere enn ved denne metoden. Den totale fraksjonen av pådraget ved bruk av likhetsbetingelser ble 21.2495, mens den ved bruk av ulikhetsbetingelser bare ble 18.0726. Dette viser at det vil være mer effektivt å benytte ulikhetsbetingelser framfor likhetsbetingelser for overflatetemperaturen. Da lar man kjøttblokkene få litt høyere temperatur slik at man slipper mye ubrukt plass i tunnelen.

Som nevnt kan overflatetemperaturen bli høyere enn ønsket ved bruk av denne metoden. Faren øker dersom fettprosenten til blokkene varierer kraftig. Dersom det er slik at fettprosenten varierer mye bør denne metoden muligens ikke benyttes. Det som da kan være en ide' er å gå tilbake til bruk av objektfunksjonen som straffer bruk av pådrag og avvik fra referanse, ligning 5.4. Når denne objektfunksjonen benyttes vil det ikke alltid være optimalt at settpunktet blir nådd. Man havner likevel i nærheten av referansen. Problemet med denne metoden er at overflaten kan få for lav temperatur. Dette kan forhåpentligvis motvirkes ved å justere på tuningparametrene α og β . Metoden med bruk av en slik objektfunksjon vil videre prøves ut.

5.3.3 Gjeninnføring av objektfunksjon

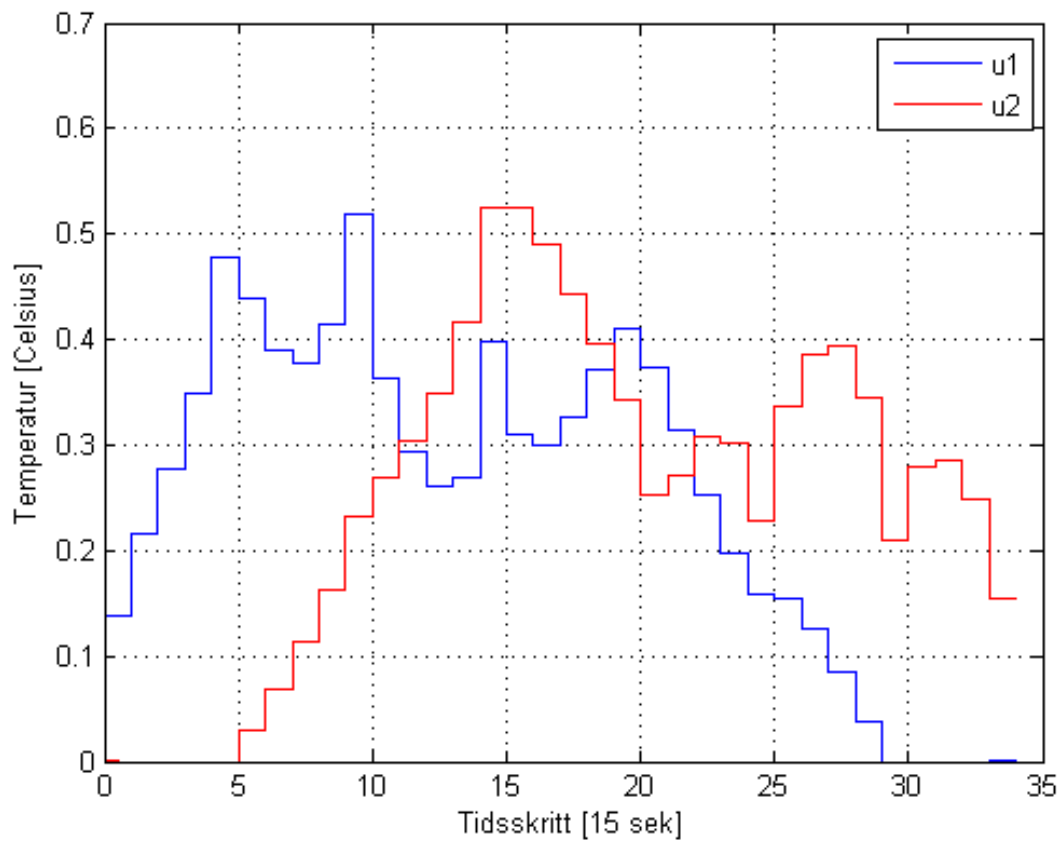
Her innføres objektfunksjonen som ble benyttet i det forenklede systemet for å se hvordan dette fungerer når det er mye kjøtt i tunnelen. Det ble tidligere fokusert på å nå eksakt temperatur, men dersom dette ikke har høyeste prioritet kan det være lurt å benytte denne metoden igjen. Dette undersøkes videre i denne delen av oppgaven.

Det vil verken være likhetsbetingelser eller ulikhetsbetingelser for overflatetemperaturen til kjøttblokkene. Det vil kun være følgende objektfunksjon:

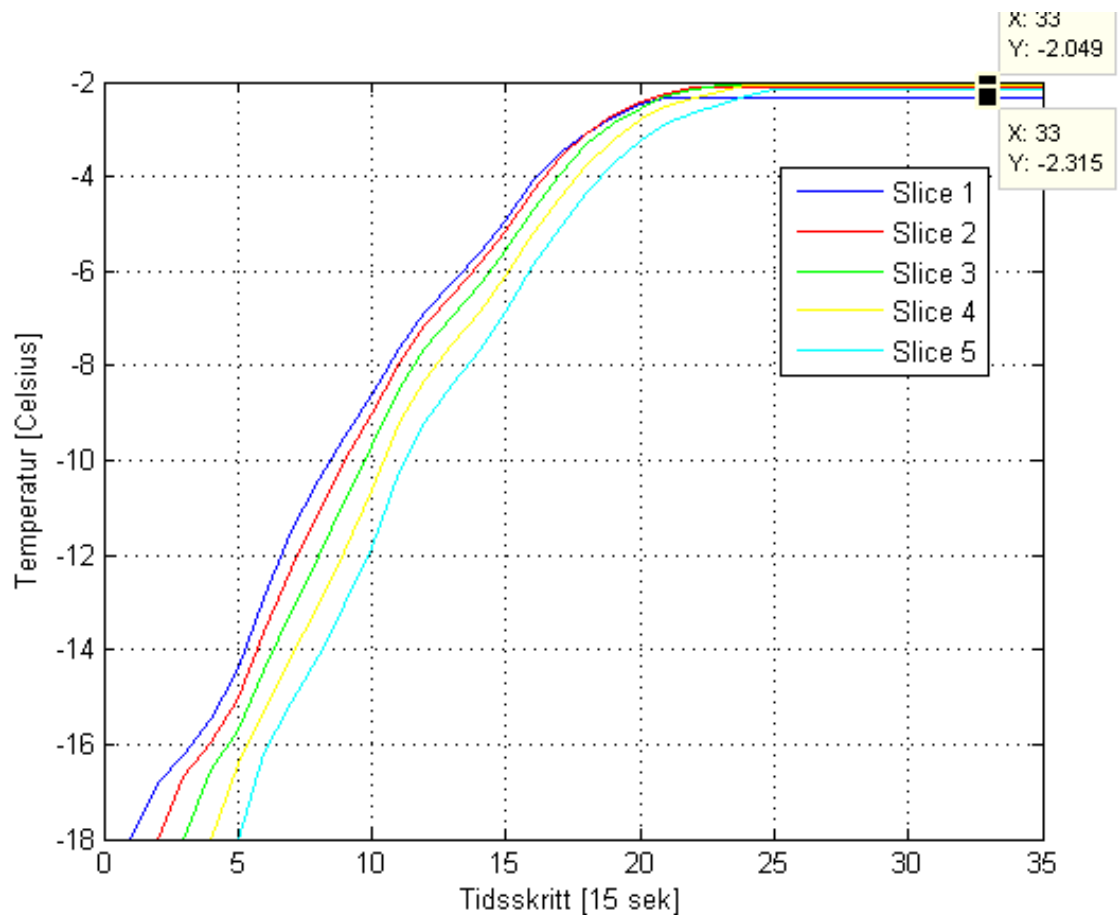
$$F = \alpha[(T_{20,1}^1 - T_{20,1}^{Set})^2 + (T_{20,1}^2 - T_{20,1}^{Set})^2 + (T_{20,1}^3 - T_{20,1}^{Set})^2 + \dots + (T_{20,1}^k - T_{20,1}^{Set})^2] + \beta[(Q_1^k)^T(Q_1^k) + (Q_2^k)^T(Q_2^k)] \quad (5.14)$$

Her har de forskjellige k samme betydning som tidligere. k representerer de tidspunktene hvor det kommer en kjøttskive ut av tunnelen. Disse tidspunktene kan variere for hver simulering.

Denne metoden prøves nå ut med den samme rekken av kjøttblokker som tidligere: 3 blokker med hhv. 5%, 15% og 25% fett. Det er heller ikke noe mellomrom mellom blokkene. N ble da 35 og $\alpha = 1$ og $\beta = 1$

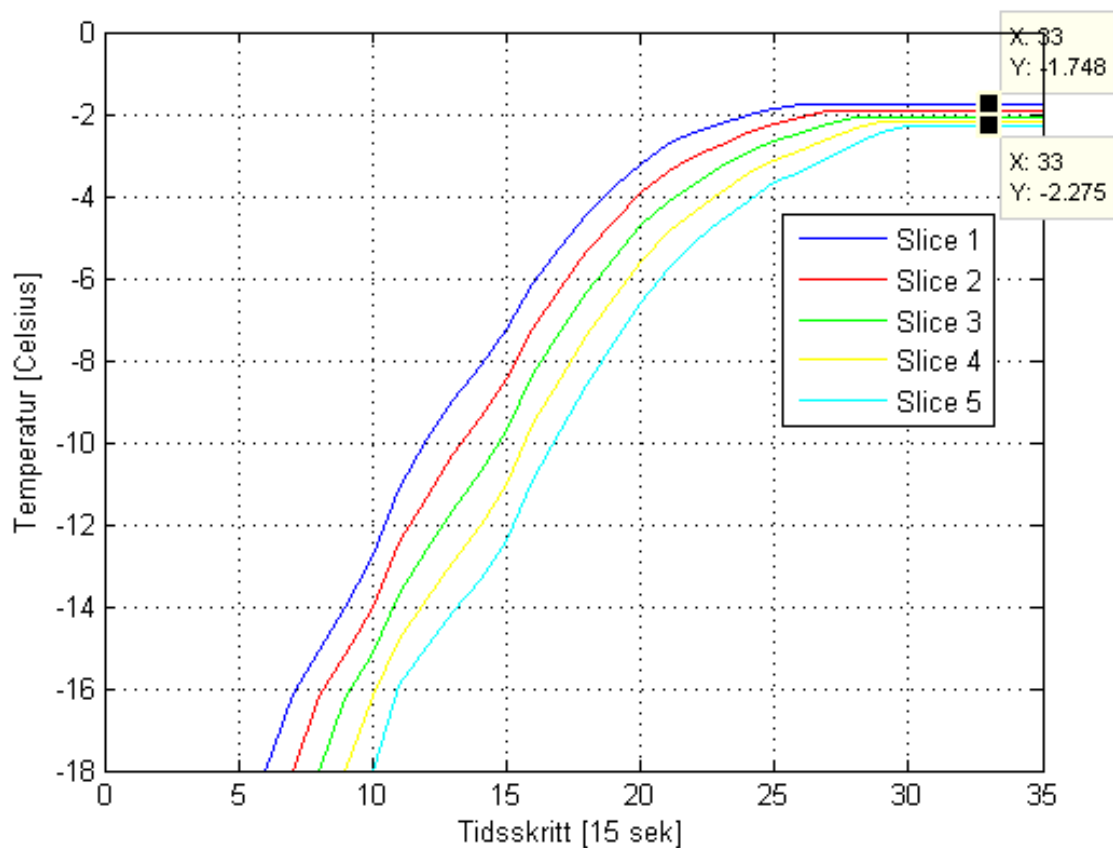


(a) Optimalt pådrag

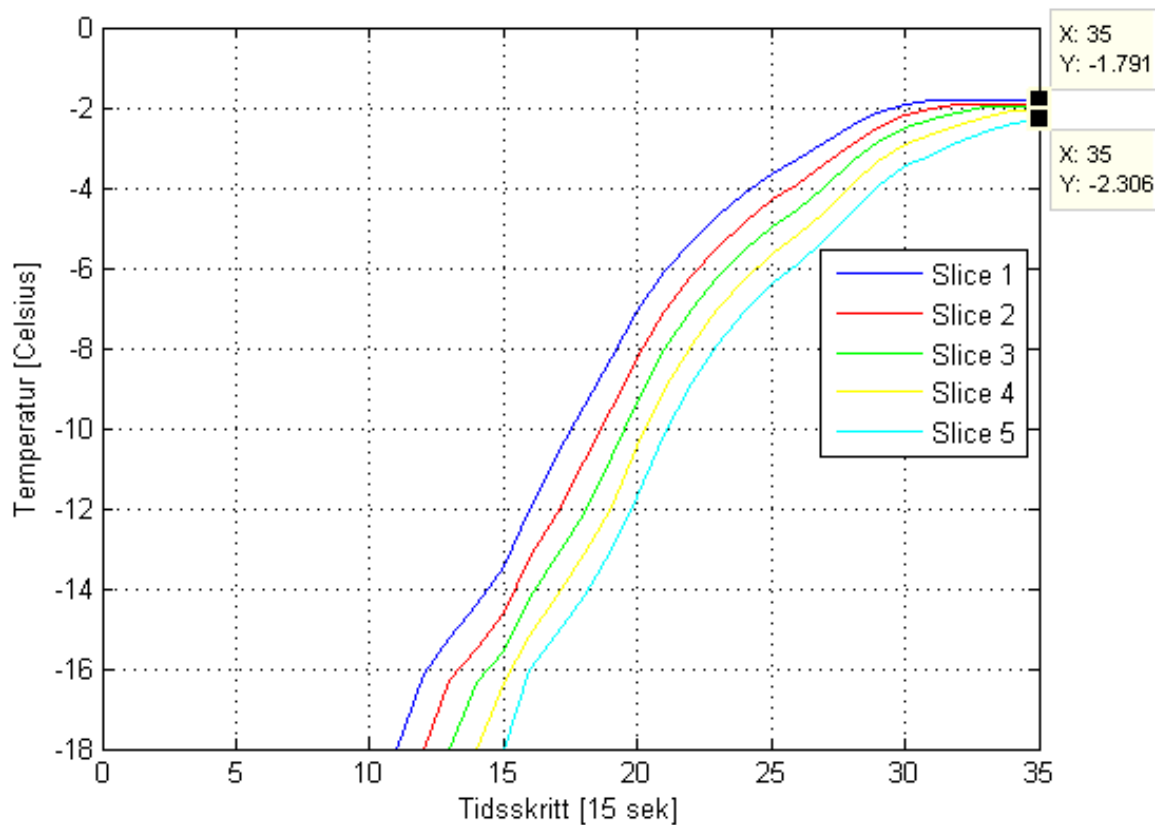


(b) Overflatetemperatur til alle skiver i første kjøttblokk, 5% fett

Figur 5.8: Optimalt pådrag og overflatetemperatur for kjøttblokk med 5% fett ved bruk av objektfunksjon.



(a) Overflatetemperatur til alle skiver i andre kjøttblokk, 15% fett

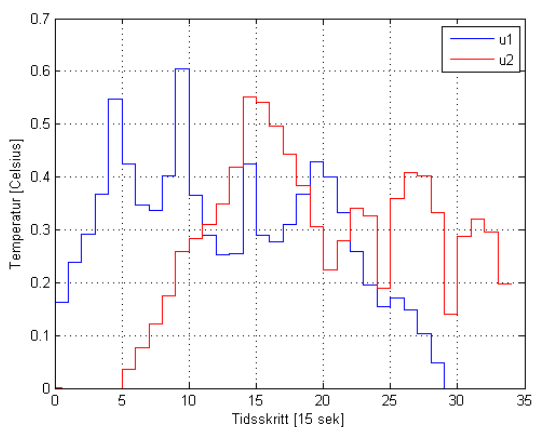


(b) Overflatetemperatur til alle skiver i tredje kjøttblokk, 25% fett

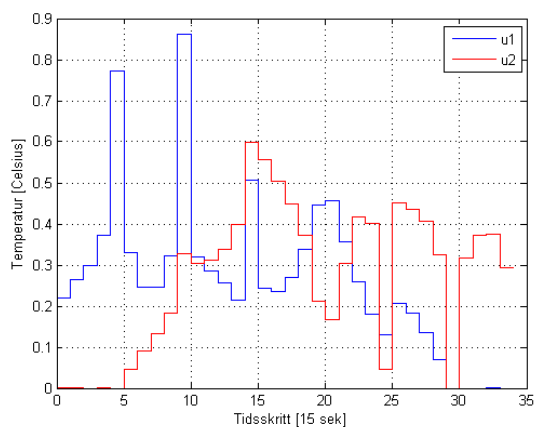
Figur 5.9: Overflatetemperatur for kjøttblokker med 15% og 25% fett ved bruk av objektfunksjon.

I figur 5.8 ser man pådraget ved alle tidssteg. Det er ikke veldig ulikt pådraget man får ved bruk av ulikhetsbetingelser, figur 5.6. Man ser at den samme formen går igjen i begge plottene. Summen av pådraget er 17.2690, noe som er mindre enn noen av de foregående pådragene. Overflatetemperaturen til kjøttskivene i blokkene vises også i figur 5.8 og 5.9. På hver figur er laveste og høyeste temperatur for skivene i hver blokk markert. For blokken med 5% fett ser man at ingen av skivene når settpunktet. Spranget mellom den kaldeste og varmeste skiven for denne blokken er litt i underkant av 0.3°C . Ingen av skivene er veldig langt unna å nå ønsket temperatur. For den andre blokken med 15% fett ser man at et par av skivene faktisk oppnår høyere temperatur enn ønsket, mens de andre får litt lavere temperatur. Her er spranget mellom den kaldeste og varmeste skiven litt i overkant av 0.5°C . Også her er ingen av skivene spesielt langt ifra settpunktet. Overflatetemperaturen til den tredje og siste blokken med 25% fett er relativt lik den andre blokken. Her er forskjellen mellom varmeste og kaldeste skive også i overkant av 0.5°C .

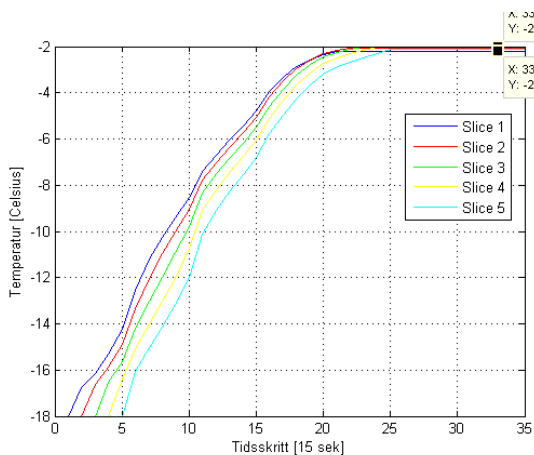
For å unngå dette avviket i temperatur mellom skivene vil nå parameteren α tunes. Det prøves med $\alpha = 2$ og $\alpha = 10$ for tydeliggjøre forskjeller. Ved å øke α vil avvik fra referansen straffes hardere og temperaturen vil bli jevnere mellom skivene. Dette gjør sannsynligvis at pådraget også øker. I figur 5.10 og 5.11 er pådrag og overflatetemperaturer sammenlignet for $\alpha = 2$ og $\alpha = 10$



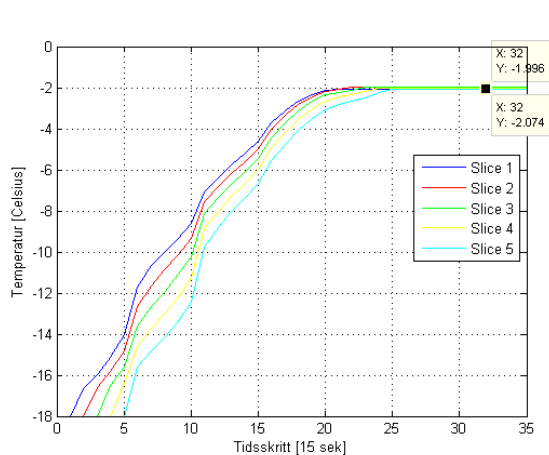
(a) Optimalt pådrag, $\alpha = 2$



(b) Optimalt pådrag, $\alpha = 10$

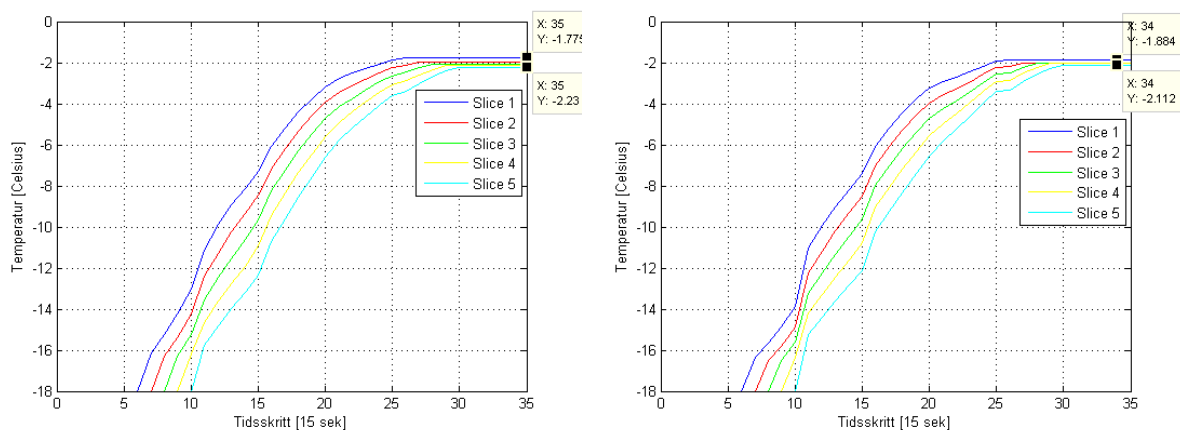


(c) Overflatetemperatur til alle skiver i første kjøttblokk, 5% fett, $\alpha = 2$

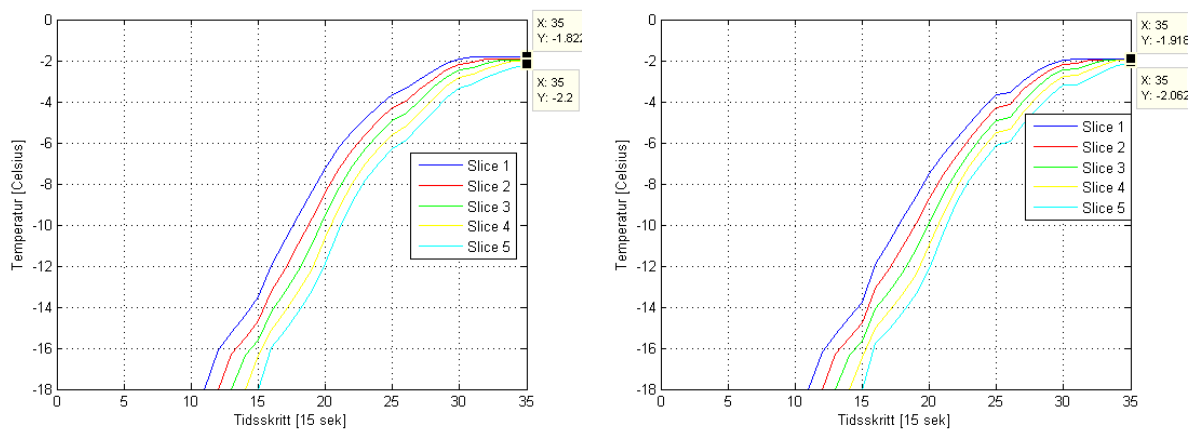


(d) Overflatetemperatur til alle skiver i første kjøttblokk, 5% fett, $\alpha = 10$

Figur 5.10: Optimalt pådrag og overflatetemperatur for kjøttblokk med 5% fett ved bruk av tunet objektfunksjon



(a) Overflatetemperatur til alle skiver i andre kjøttblokk, 15% fett, $\alpha = 2$ (b) Overflatetemperatur til alle slicer i andre kjøttblokk, 15% fett, $\alpha = 10$



(c) Overflatetemperatur til alle skiver i tredje kjøttblokk, 25% fett, $\alpha = 2$ (d) Overflatetemperatur til alle skiver i tredje kjøttblokk, 25% fett, $\alpha = 10$

Figur 5.11: Overflatetemperatur for kjøttblokker med 15% og 25% fett ved bruk av tunet objektfunksjon

I figurene ser man at pådraget har en tendens til å variere mer når α øker. Avvik fra referanse straffes hardere og det må brukes mer pådrag for å oppnå ønsket temperatur. For $\alpha = 2$ er summen av pådraget 17.6503, mens det for $\alpha = 10$ er 18.1766. Begge disse er større enn pådraget når $\alpha = 1$, noe som ikke var uventet. Det vil altså bli benyttet mer pådrag når α øker. Når $\alpha = 1$ er summen av pådrag lavere enn metodene med bruk av både likhetesbetingelser og ulikhetsbetingelser, mens $\alpha = 10$ gir et pådrag på nivå med bruk av ulikhetsbetingelser. Når man sammenligner overflatetemperaturen til blokkene ved ulik α , ser man at når α øker får man større sprang i temperaturen og ikke den jevne utviklingen man er vant med fra før. Dette kommer av at pådraget også varierer kraftigere. Det positive er at man får mindre forskjell i temperaturen til de forskjellige skivene som utgjør en kjøttblokk ved økende α . Dette gjelder uansett fettprosent.

5.3.4 Sammendrag av metoder

Det er tre metoder for å finne optimalt pådrag ved tining av kjøtt som er undersøkt her. Bruk av likhetesbetingelser for å oppnå ønsket overflatetemperatur; Denne metoden ga en eksakt temperatur på kjøttblokkene. Den benyttet mest pådrag av alle metodene og dersom fettprosenten i

kjøttblokkene varierte mye, måtte det være mellomrom mellom blokkene. Dette førte til at man ikke kunne utnytte tinekapasiteten til tunnelen til det fulle og gjorde metoden minst effektiv. Dersom fettprosenten var lik for alle blokkene kunne man utnytte hele tunnelen til enhver tid.

Bruk av ulikhetsbetingelser for overflatetemperatur; Denne metoden gjorde at overflatetemperaturen i kjøttet ble større eller lik en gitt verdi. Man hadde ikke kontroll på hvor høy temperaturen kunne bli og metoden kunne derfor være litt risikabel, spesielt dersom fettprosenten til blokkene varierte. Fordelen framfor bruk av likhetsbetingelser var at man kunne utnytte hele tunnelen og trengte ikke mellomrom mellom blokkene. Dersom man godtar unøyaktig temperatur vil det være mer effektivt å bruke denne metoden enn likhetsbetingelser.

Bruk av objektfunksjon; Ved å benytte en objektfunksjon fikk man mye av den samme effekten som bruk av ulikhetsbetingelser gav. Man hadde ikke full kontroll på hva temperaturen endte opp på, bare at den ble i nærheten av en gitt verdi. Det kunne se ut som det var tryggere å benytte denne metoden, med tanke på for høy temperatur i blokkene. I forhold til bruk av ulikhetsbetingelser havnete man nå i nærheten av ønsket temperatur, og ikke garantert over. Med denne metoden kunne man selv bestemme hvor mye man ville straffe avvik fra settpunktet, og dermed ha bedre kontroll over hvor høy temperaturen ble. Man slipper også mellomrom mellom blokkene, dersom man godtar litt avvik i temperaturen. Denne metoden er kanskje den mest energi effektive, dersom man ikke er for streng på temperatur nøyaktigheten.

Det kan konkluderes med at ved tining av kjøtt bør man tine kjøttblokker med relativt lik fettprosent samtidig, uansett metode. Man bør heller ikke forvente eksakte temperaturer i kjøttblokkene, men at temperaturen ligger i et spesifikt sjikt kan forventes. Metodene bør testes ut i virkeligheten for å se hvilke metoder som passer best til forskjellige scenarier og hva som vil være mest økonomisk effektivt.

Kapittel 6

Tilbakekoblet styring

Hittil er det optimaliserende systemet bygget på modellbasert styring av prosessen. Det har ikke eksistert noen målinger som kunne vært benyttet for tilbakekoblet styring. Derfor har det fram til nå vært åpen sløyfe styring. I denne seksjonen vil mulighetene for tilbakekoblet styring, fra for eksempel målte temperaturer, vurderes. Det beskrives også hvordan disse kan benyttes i en ulineær MPC regulator.

6.1 Målinger

Hensikten med tiningen er at kjøttet skal oppnå en gitt temperatur. Derfor bør man ha mulighet til å sjekke om den faktiske temperaturen er lik ønsket temperatur. Det er derfor naturlig å forsøke å måle temperaturen i kjøttet. Man kan da for eksempel måle overflatetemperaturen til kjøttet. Siden det er vanskelig å ha måleutstyr inne i tunnelen foreslås det derfor at måling av temperatur skjer idet kjøttet kommer ut av tunnelen. Ved å måle temperaturen blir man istand til å korrigere pådraget, basert på et avvik mellom ønsket temperatur og virkelig temperatur.

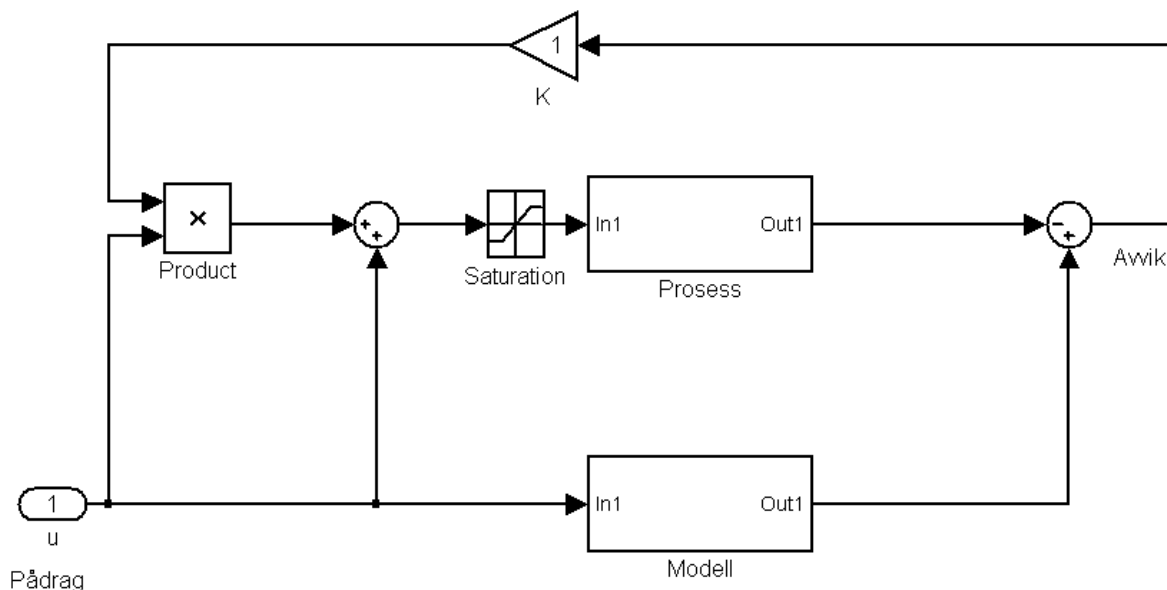
I tillegg til å måle temperaturen kan man også tenke seg at måling av pådrag kan være fornuftig. Det er ikke sikkert det pådraget man setter faktisk er den effekten man får ut av mikrobølgeelementene. Det kan da være en ide å måle effekten man får ut. Slik kan man korrigere for feil og justere pådraget slik at det blir nøyaktig.

6.2 Modellfeil

En av antagelsene for at det optimale systemet skal fungere er at alle parametrene i modellen for kjøttet er korrekte. I virkeligheten vil kanskje kjøttet ha andre egenskaper enn antatt. Dette kan føre til at temperaturen i de virkelige kjøttblokkene vil avvike fra de estimerte verdiene man får ved optimaliseringen av pådragene. modellfeil er veldig vanlig og det bør derfor tas hensyn til at dette kan forekomme.

6.2.1 Regulering

En måte å ta hensyn til dette er ved å benytte en regulator for å justere pådraget, dersom virkelig temperatur ikke stemmer med de estimerte verdiene man får fra simuleringen. Dersom man kan måle overflatetemperaturen til blokker som kommer ut av tunnelen, kan denne sammenlignes med den estimerte overflatetemperaturen man får fra modellen. Et eventuelt avvik mellom disse

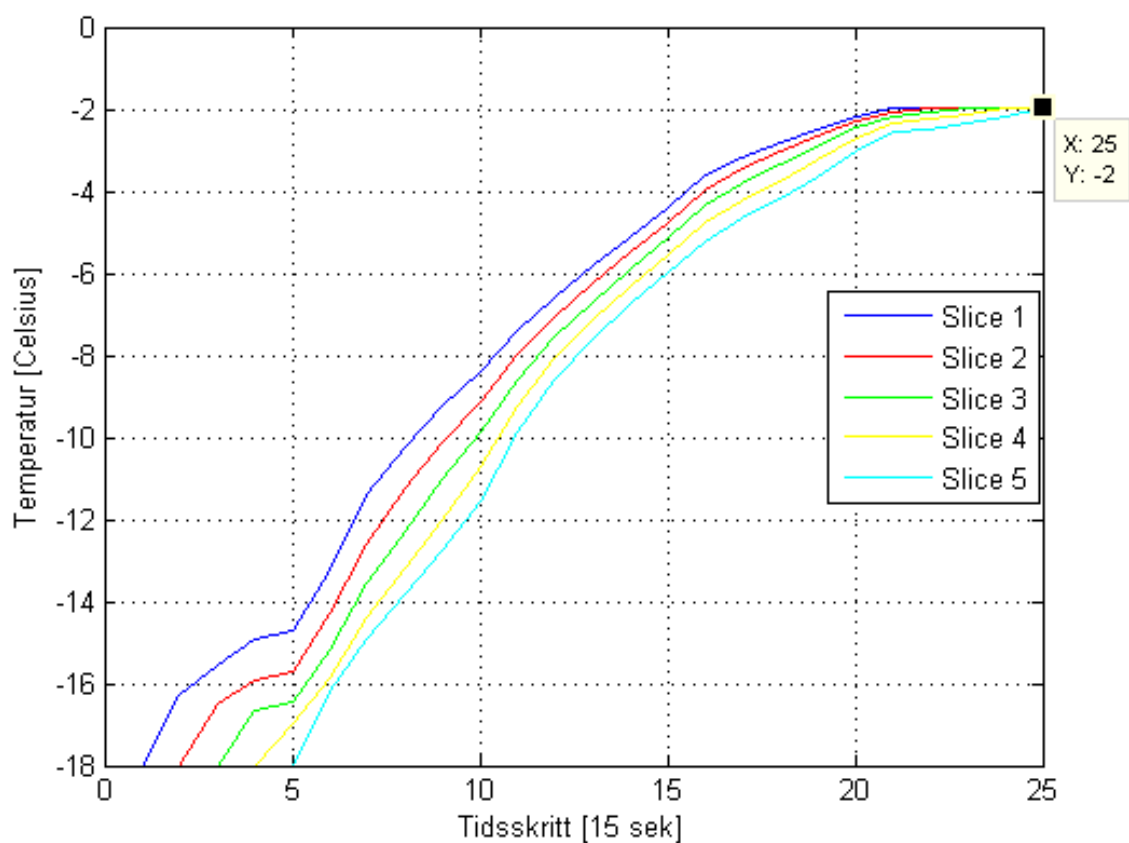


Figur 6.1: P-Regulator

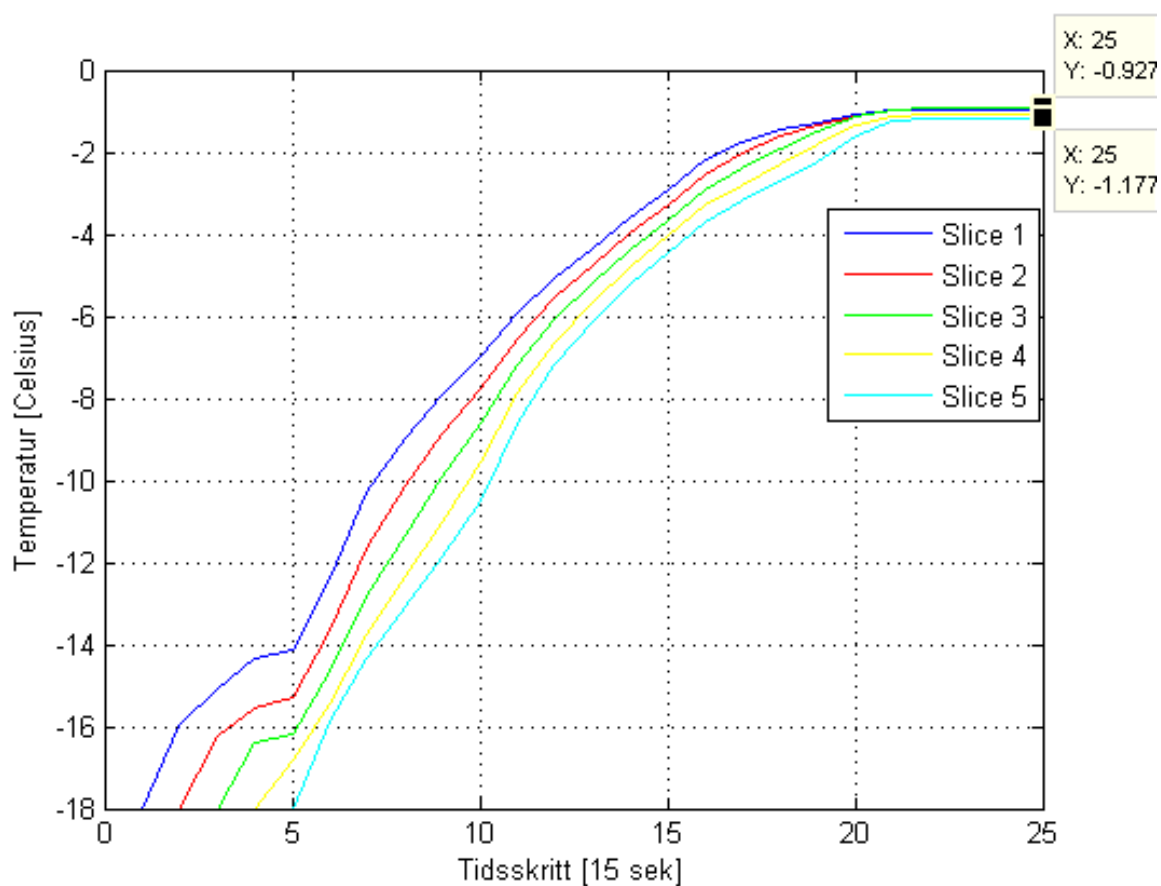
to vil benyttes i en regulator for å endre pådraget ved neste tidssteg. Figur 6.1 viser hvordan en slik regulator kan være.

Avviket multipliseres med en konstant, K , og med det opprinnelige pådraget. Dette addresseres så på det opprinnelige pådraget. Dette gir følgende effekt: dersom temperaturen er lavere enn estimert blir avviket positivt. Da vil pådraget til prosessen øke. Dersom temperaturen er høyere enn estimert blir avviket negativt. Det gjør at pådraget til prosessen minker i forhold til det opprinnelige pådraget. Man må også passe på at pådraget ikke settes til å være høyere enn maksimal effekt eller lavere enn minimal effekt. Dette sørger metningsblokken for. Regulatoren baserer seg på de pådragene som ble funnet ved optimalisering av en serie kjøttblokker med antatt riktige parametre. Når en kjøttblokk kommer ut av tunnelen og det viser seg at den ikke har riktig temperatur, er det rimelig å anta at de andre kjøttblokkene heller ikke vil oppnå riktig temperatur. Derfor justeres pådraget på bakgrunn av disse målingene. Siden målingen blir tatt ved enden av tunnelen kan man ikke gjøre noe for å endre temperaturen til blokkene som allerede har kommet ut av tunnelen. En mulighet for å gjøre noe med dette kan være å ha en temperatur måling midt i tunnelen, dersom det er mulig.

For å vise funksjonaliteten til denne regulatoren er det laget følgende to plott: Figur 6.2a viser den estimerte temperaturen i en kjøttblokk med 5% fett. Figur 6.2b viser temperaturen i kjøttblokken som faktisk kommer ut av tunnelen. Denne blokken er satt til å ha litt andre parametre for å vise hvordan regulatoren reagerer på modellfeil. I dette tilfellet er fettprosenten satt til å være 25% for å vise et ekstremt tilfelle. Forsterkningen i regulatoren er 5, og en økning i forsterkningen vil ikke gjøre den nevneverdig bedre. Av figuren ser man at alle skivene i blokken i modellen estimeres til å nå -2°C . Siden modellfeilen her er at den virkelige blokken har en fettprosent på 25, og ikke 5, vil denne blokken få større overflatetemperatur enn estimert. Regulatoren begynner å virke i det den første skiven kommer ut av tunnelen. Da blir avviket mellom modellen og prosessen tatt med i beregning av nytt pådrag, og pådraget senkes. Det ser man på de neste skivene. De oppnår alle lavere temperatur enn den første. Ingen av skivene når settpunktet. Dette er pga. at uansett hvor lavt pådraget settes, vil ikke temperaturen synke. Alle skivene har allerede for høy temperatur idet den første skiven kommer ut av tunnelen. Man kan se at de resterende skivene faktisk ikke øker noe i temperatur etter at den første har kommet ut av tunnelen. Dette betyr at pådraget er satt til null.



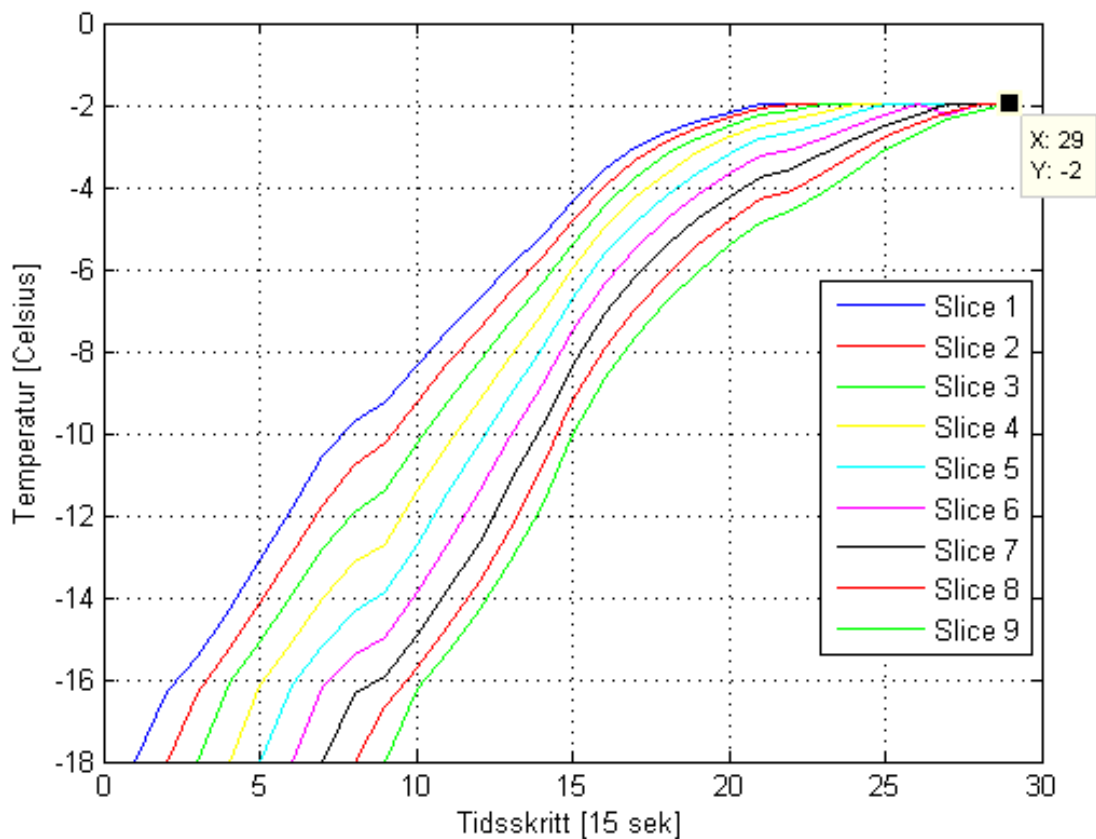
(a) Estimert overflatetemperatur



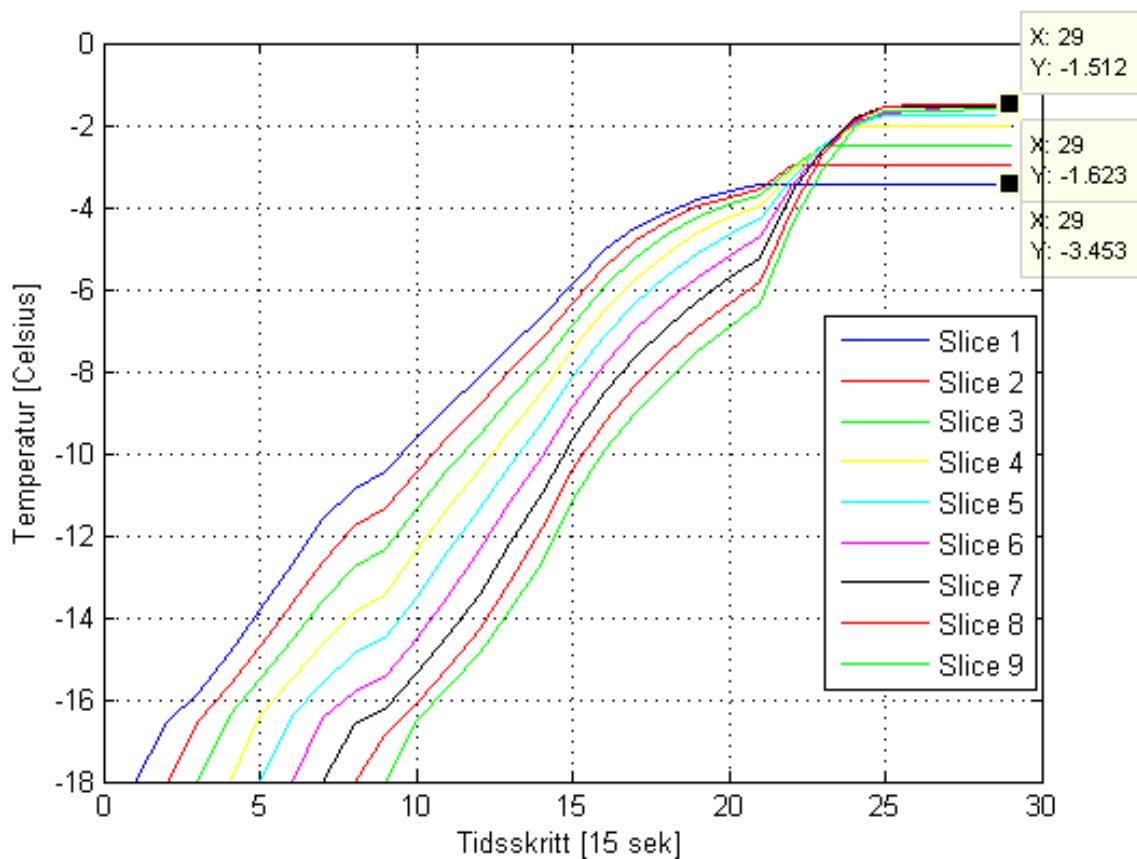
(b) Virkelig overflatetemperatur

Figur 6.2: Overflatetemperatur for en kjøttblokk med 25% fett, når det er modellfeil og blokk i modell har 5% fett

Det ble sett på hvordan regulatoren reagerte dersom en modellfeil gjorde at temperaturen ble høyere enn estimert. Nå vil det undersøkes hva som skjer om temperaturen blir for lav i forhold til estimerte verdier. Det simuleres at den virkelige verdien på fettprosenten er 5, mens den i modellen er på 25%, altså omvendt fra forrige simulering. Det simuleres 9 kjøttskiver, slik at man bedre kan se effekten av regulatoren. I figur 6.3a ser man at de estimerte verdiene for overflatetemperaturen er -2°C . Siden den virkelige blokken har høyere fettprosent, vil den trenge mer energi, og derfor får den lavere temperatur enn estimert. I figur 6.3b ser man at den første skiven i blokken har en overflatetemperatur på -3.45°C når den kommer ut. Regulatoren tar så hensyn til dette og øker pådraget. Dermed oppnår de resterende skivene gradvis høyere temperatur. Temperaturen til skive 8 havner på -1.51°C , som er over settpunktet. Den siste skiven oppnår så en temperaur på -1.62°C , siden pådraget senkes når temperaturen er for høy. Det vil nå være interessant å se hvor lang tid regulatoren bruker på å svinge inn til settpunktet. Dersom simuleringen får fortsette med flere kjøttskiver, viser det seg at temperatuen blir ustabil og svinger om settpunktet. Grunnen til det er at pådraget forandres ved hvert tidssteg. På grunn av spredningen i tunnelen blir skivene utsatt for forskjellig pådrag etter hvor de befinner seg i tunnelen. Siden man sjekker temperaturforskjell på alle skivene som kommer ut forandres også pådraget ved hvert tidssteg. Det betyr at man ikke får tid til å se hvordan det gitte pådraget faktisk påvirker hele serien av skiver. Dersom en slik regulator skal kunne svinge inn må man la alle skivene som ble utsatt for pådraget komme ut av tunnelen før man tar en ny måling og eventuelt forandrer pådraget på nytt. Bare slik vet man hvordan pådraget påvirket skivene. Regulatoren vil derfor modifieres slik at temperaturen skal kunne svinge inn til settpunktet.



(a) Estimert overflatetemperatur



(b) Virkelig overflatetemperatur

Figur 6.3: Overflatetemperatur for en kjøttblokk med 5% fett, når det er modellfeil og blokk i modell har 25% fett

6.2.2 Modifisert regulator

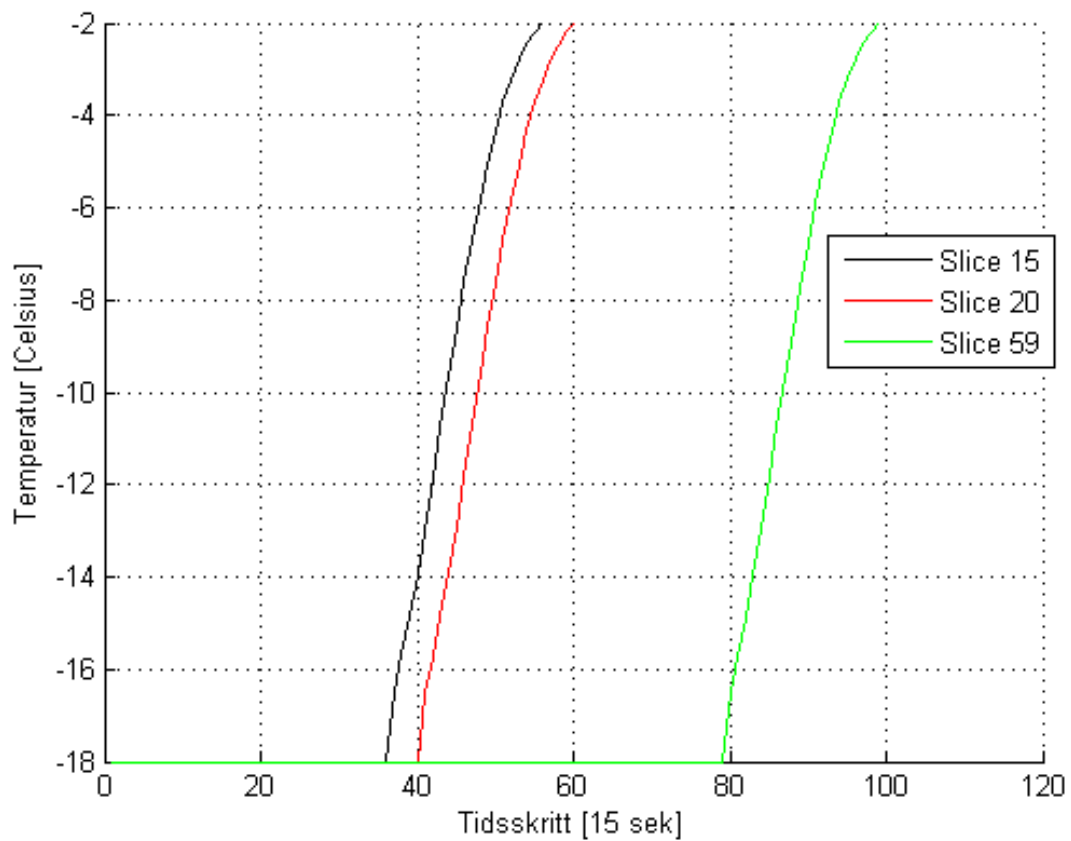
Det første som gjøres er å la regulatoren endre kun u_2 , for å unngå å forholde seg til to pådrag. Styringen blir mindre komplisert med kun et pådrag ved hvert tidssteg. Pådragene finner man, som før, ved optimalisering. Etter optimaliseringen kjører man så de optimale pådragene på den virkelige prosessen. u_1 forandres ikke underveis og vil være det som ble funnet ved optimaliseringen. Det er kun u_2 som endres av regulatoren. Neste forskjell er at man venter 15 tidsskritt før man gjør en sammenligning mellom virkelig temperatur og predikert temperatur. Grunnen til at man sjekker hvert 15. tidssteg er at u_2 påvirker de 15 siste skivene i tunnelen. Ved å sjekke hvert 15 tidssteg rekker skivene å komme ut av tunnelen og man ser hvordan pådraget har påvirket alle skivene. Man unngår da at pådraget endres hvert tidssteg. Avviket man finner ved ny måling legges til avviket ved forrige måling. Ved å huske avviket fra forrige måling unngår man at pådraget senkes/økes for mye, dersom man havner veldig nært settpunktet. Dersom første avvik for eksempel var på to grader endres det optimale pådraget ganske mye. Dersom neste avvik er lite vil endringen i det optimale pådraget bli liten. Dersom man ikke tar hensyn til hva forrige avvik var, vil man ikke endre det optimale pådraget nok, slik at temperaturen vil bevege seg bort fra settpunktet. Se vedlegg B.3 for Matlab kode av implementert regulator.

For å illustrere hvordan regulatoren nå virker, kjøres en simulering med 95 kjøttskiver. Det er simulert at kjøttblokker man tror inneholder 25% fett skal tines. De inneholder egentlig 5% fett og derfor vil pådragene funnet ved optimaliseringen gi feil temperatur. Dette skal regulatoren forsøke å rette opp. Under optimaliseringen benyttes metoden med likhetsbetingelser, slik at overflatetemperaturen skal bli eksakt -2°C . Ved å benytte den modifiserte regulatoren vil temperaturen være -2.0067°C på skive nr 20. Dette kan anses som en akseptabel temperatur. De neste skivene nærmer seg settpunktet og når skive 59 kommer ut holder den eksakt -2°C . De neste skivene variere likevel litt, men det er kun snakk om en variasjon på 0.001°C . Grunnen til dette er at det optimale pådraget, som egentlig skal gi riktig temperatur, ikke er konstant. Det variere litt ved hvert tidssteg. Derfor er det umulig for regulatoren å sette ut et pådrag som fører til eksakt overflatetemperatur, med mindre denne variasjonen tas hensyn til. Men avviket er så lite at det er mer enn godt nok. I figur 6.4a ser man temperaturforløpet til skive 15, 20 og 59. Figur 6.4b viser pådraget funnet ved optimaliseringen. Dette variere en del, derav den lille unøyaktigheten i temperaturen. Merk at dette ikke er det pådraget som settes ut av regulatoren. Tabell 6.1 viser overflatetemperaturen i alle skivene. Det faktum at skive 15 nesten når riktig temperatur tyder på at regulatoren var godt tunet. Det er små justeringer som skal til for at de neste skivene skal få helt riktig temperatur. Grunnen til at temperaturen fortsatt svinger litt er at det optimale pådraget varierer. De helt siste skivene synker en del i temperatur. Tendensen begynner ved skive 88. Grunnen til dette er at u_1 minker kraftig mot slutten og tilfører ikke skivene varme. Regulatoren påvirker bare u_2 , men den er sansynligvis for treg slik at u_2 ikke øker hurtig nok til å motvirke effekten av at u_1 senkes såpass hurtig.

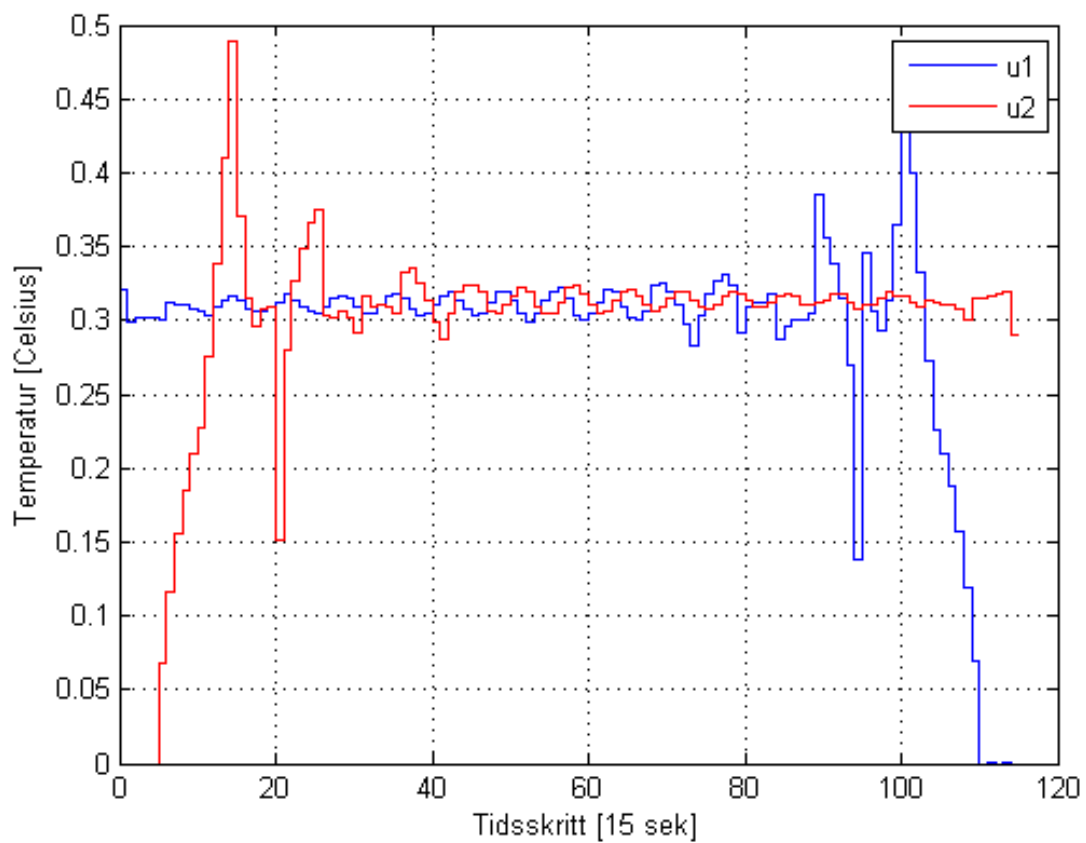
Den modifiserte regulatoren har vist seg å fungere greit. Den bruker ikke altfor lang tid på å svinge inn til riktig temperatur. Etter ca 20 skiver oppnår man en akseptabel temperatur. Dette tilsvarer å sende gjennom 4 blokker kjøtt. Dette gjelder derimot kun for den type feil som ble prøvd ut, altså for lav fettprosent. Det er ikke sikkert den vil stabilisere seg så fort dersom det er andre feil i modellen. Uansett vil en slik regulator kunne fungere bedre dersom det er muligheter for måling inne i tunnelen. Da slipper man den lange tidsforsinkelsen og har mulighet til å endre pådraget med en gang. Da vil det også være lettere å endre både u_1 og u_2 . Gjennom forsøkene med de to regulatorene har det vist seg at det er vanskelig å styre en slik prosess med lang tidsforsinkelse. Den beste måten å gjøre det på vil nok være å benytte seg av en MPC for å styre prosessen. Dette diskuteres nærmere i neste delkapittel.

Skive nr	Overflatetemperatur	Skive nr	Overflatetemperatur
1	-3.4515	2	-3.4612
3	-3.3795	4	-3.2740
5	-3.1205	6	-2.9656
7	-2.7922	8	-2.6220
9	-2.4726	10	-2.3457
11	-2.2429	12	-2.1656
13	-2.1076	14	-2.0637
15	-2.0324	16	-2.0139
17	-2.0085	18	-2.0083
19	-2.0076	20	-2.0067
21	-2.0056	22	-2.0048
23	-2.0037	24	-2.0031
25	-2.0029	26	-2.0028
27	-2.0027	28	-2.0025
29	-2.0019	30	-2.0013
31	-2.0010	32	-2.0009
33	-2.0009	34	-2.0010
35	-2.0010	36	-2.0009
37	-2.0008	38	-2.0007
39	-2.0008	40	-2.0008
41	-2.0008	42	-2.0007
43	-2.0005	44	-2.0003
45	-2.0003	46	-2.0003
47	-2.0005	48	-2.0006
49	-2.0006	50	-2.0004
51	-2.0003	52	-2.0002
53	-2.0001	54	-2.0002
55	-2.0002	56	-2.0002
57	-2.0002	58	-2.0001
59	-2.0000	60	-2.0000
61	-2.0001	62	-2.0002
63	-2.0001	64	-2.0000
65	-1.9998	66	-1.9999
67	-1.9999	68	-1.9999
69	-2.0001	70	-2.0001
71	-2.0000	72	-1.9999
73	-1.9998	74	-1.9996
75	-1.9998	76	-2.0003
77	-2.0003	78	-2.0005
79	-2.0006	80	-2.0006
81	-2.0012	82	-2.0009
83	-2.0007	84	-2.0007
85	-2.0003	86	-2.0004
87	-2.0005	88	-2.0011
89	-2.0022	90	-2.0048
91	-2.0062	92	-2.0077
93	-2.0667	94	-2.1483
95	-2.2464		

Tabell 6.1: Overflatetemperatur til kjøttskiver når modifisert regulator benyttes



(a) Overflatetemperatur til skive 15, 20 og 59



(b) Optimalt pådrag funnet ved optimalisering

Figur 6.4: Overflatetemperatur og optimalt pådrag

6.3 Ulineær MPC

Dersom man har mulighet for å ta målinger, bør disse benyttes for å korrigere for feil i modellen og støy. For å kunne ta hensyn til disse målingene bør man benytte en ulineær MPC regulator. Det er flere grunner til at denne bør benyttes. Den benytter en intern modell av prosessen for å beregne optimale pådrag. Pådragene man finner minimerer avvik fra settpunkt og total bruk av pådrag. MPC kan håndtere begrensninger både på inngangs og utgangs variable, så vel som interne tilstander. MPC er også multivariabelt så det kan ta hensyn til flere inngangs og utgangs variable [8]. Siden systemet som er utviklet i denne oppgaven både er ulineært, multivariabelt og har begrensninger på pådrag, bør en ulineær MPC regulator benyttes dersom man ønsker tilbakekobling fra eventuelle målinger.

6.3.1 Generelt

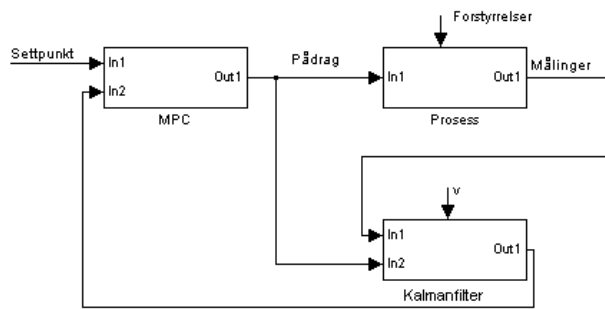
Man kan anta at man måler overflatetemperaturen til blokkene idet de kommer ut av tunnelen. Denne målingen er da den virkelige overflatetemperaturen til blokken det måles på. Denne vil ikke være lik den predikerte overflatetemperaturen fra den interne modellen, dersom det er modellfeil eller støy. Målingene brukes til å oppdatere tilstandsverdiene slik at disse ikke drifter vekk fra de virkelige verdiene. Deretter kan MPC regulatoren finne nye optimaliserende pådrag, slik at man oppnår riktig overflatetemperatur. Problemet med ulineær MPC er at en optimalisering kan ta veldig lang tid. Kompliserte og tunge utregninger kan sette begrensninger på bruk av MPC. Det er derfor viktig at tiden det tar å regne ut de nye pådragene ikke er lengre enn tiden det tar fra et pådrag settes ut til neste pådrag skal settes ut. Dersom måling underveis i tunnelen er mulig ville dette vært en fordel i forhold til måling ved utgangen. Man kunne da unngått tidsforsinkelsen som oppstår når man venter på at blokkene skal komme ut av tunnelen.

Forskjellen på å benytte MPC og regulatoren beskrevet tidligere i oppgaven blir at pådragene optimaliseres hver gang det kommer en ny måling. Før ble pådragene bestemt ved å optimalisere en gang. Målingene ble benyttet til å justere pådraget basert på avviket. Med MPC får man optimale pådrag ved hvert tidssteg, basert på målinger og estimerte tilstander. Dersom det er modellfeil vil en MPC regulator hurtigere tilpasse pådragene slik at blokkene oppnår riktig temperatur.

6.3.2 Forslag til bruk av ulineær MPC

Et forslag til hvordan man kan benytte en MPC for denne prosessen beskrives her. Man kan anta at man måler overflatetemperaturen til blokkene to til tre plasser i tunnelen, og benytter et Kalmanfilter for å estimere alle andre temperaturer til blokkene i tunnelen. Temperatur estimatene man får fra Kalmanfilteret kan så benyttes av MPC'en til å finne optimale pådrag som gir riktig temperatur i blokkene. Figur 6.5 viser en skisse av hvordan dette systemet henger sammen.

MPC'en benytter de estimerte tilstandene fra Kalmanfilteret til å beregne optimale pådrag. Pådragene settes så ut til prosessen. Kalmanfilteret benytter seg av målingene for å estimere tilstandene til de ikke-målte tilstandene. Kalmanfilteret kan også estimere en eventuell målebias eller pådragsfaktor fra temperaturmålingene. Da slipper man å måle den egentlige effekten fra mikrobølgeelementene, som er nevnt tidligere. Ved å gjøre dette slår man sammen feil i modellen av kjøttet pluss feil i modellen av pådraget. Ved å benytte MPC og Kalmanfilter vil modellfeil og støy motvirkes hurtigere enn ved bruk av de forrige regulatoren. Dersom det er en konstant modellfeil bør den etter kun få tidssteg korrekteres for. Det som kan være et problem er at MPC'en bruker for lang tid på å regne ut nye pådrag. Dersom dette er tilfellet bør modellen



Figur 6.5: MPC og Kalmanfilter

kanskje reduseres slik at den ikke blir for kompleks, for eksempel ved å ha færre lag eller færre skiver i vår kjøttblokk. Dette bør testes ut dersom man vil benytte et slikt system.

Kapittel 7

Konklusjon

Kjøttblokker har ulik fysisk sammensetning, for eksempel forskjellige termiske egenskaper. Dette gjør at de trenger ulik mengde energi for å tines. I denne oppgaven ble det funnet ut at fettprosenten til kjøttet har størst betydning for hvor mye energi tiningen krever. Fettprosenten har en direkte innvirkning på varmekapasiteten til kjøttet, og penetrasjonsdybden til mikrobølgene. Derfor må mikrobølgepådraget justeres etter disse fysiske parametrene. Lamberts lov ble valgt som matematisk modell for å beskrive tineprosessen. Den var den minst komplekse og krevde minst regnetid. Den matematiske modellen ble simulert og parametrene ble tilpasset virkeligheten. Det ble funnet ut at variasjonen i parametre er størst når kjøttet forandrer seg mellom frossent og tint tilstand. Frysepunktet til kjøtt varierer rundt -1.5°C . Det er derfor viktig at når et slikt system skal implementeres i virkeligheten, må det gjøres forsøk, slik at alle parametre og den matematiske modellen blir så riktig som mulig.

Styresystemet til tinetunnelen ble basert på en modell av kjøttet, mikrobølgene og tunnelen. Ut fra modellen ble det funnet optimale pådrag som skulle gi kjøttet riktig temperatur basert på målt inngående fettprosent. Forskjellige systemer ble testet og simulert, både med bruk av ulikhetsbetingelser, likhetsbetingelser og objektfunksjon. Konklusjonen fra testingen var at alle metodene kan benyttes til ulike formål. Ved tining av kjøtt bør man tine kjøttblokker med relativt lik fettprosent samtidig, uansett metode. Man bør heller ikke forvente eksakte temperaturer i kjøttblokkene, men at temperaturen ligger i et spesifikt sjikt kan forventes. I noen tilfeller bør ikke blokkene som tines ligge tett inntil hverandre. Metodene bør testes ut i virkeligheten for å se hvilke metoder som passer best til forskjellige scenarier og hva som vil være mest økonomisk effektivt.

Ved å benytte modellbasert åpen sløyfe styring vil systemet være følsomt for modellfeil og støy. Det blir derfor konkludert med at det bør være temperaturmålinger av kjøttet slik at dette kan motvirkes. Helst bør det være målinger inne i tunnelen. Dette vil være bedre enn om det kun er en måling etter tunnelen. Målingene bør benyttes til å regulere pådraget slik at riktig temperatur oppnås selv om det er modellfeil eller støy. For å regulere pådraget anbefales det å benytte seg av MPC og Kalmanfilter. Dette vil være den beste måten å benytte seg av målinger for å finne optimale pådrag.

Bibliografi

- [1] www.ferriteinc.com.
- [2] P. G. Creed and S. J. James. Predicting thawing time of frozen boneless beef blocks. 1981.
- [3] S. Curet, O. Rouaud, and L. Boillereaux. Microwave tempering and heating in a single-mode cavity: Numerical and experimental investigations. Chemical Engineering and Processing, 47:1656–1665, 2007.
- [4] A.K. Datte and R.C. Anantheswaran. Handbook of microwave technology for food applications. Marcel Dekker, 2001.
- [5] K.W. Farag, J.G. Lyng, D.J. Morgan, and D.A. Cronin. Dielectric and thermophysical properties of different beef meat blends over a temperature range of -18 to +10 c. Meat Science, 79:740–747, 2008.
- [6] Nishkaran Gunasekaran. Effect of fat content and food type on heat transfer during microwave heating. Master's thesis, Faculty of Virginia Polytechnic Institute and State University, 2002.
- [7] S. J. James and C. James. Meat refrigeration. Woodhead Publishing Limited, Abington Hall, 2002.
- [8] Jan Maciejowski. Predictive control with constraints. Pearson Educational Limited, 2002.
- [9] Adam Powell. Finite difference solution of the heat equation. 2002.
- [10] B.J. Taher and M.M. Farid. Cyclic microwave thawing of frozen meat: experimental and theoretical investigation. Chemical Engineering and Processing, 40:379–389, 2001.
- [11] Tavman, Sebnem, Kumcuoglu, Seher, and Volker Gaukel. Apparent specific heat capacity of chilled and frozen meat products. International Journal of Food Properties, 10:1:103–112, 2007.

Tillegg A

Symbolliste

Symbol	Betydning	Enhet
B	Magnetisk induksjon	Ns/Cm
c	Lyshastighet	m/s
c_p	Spesifikk varmekapasitet	J/kgK
D	Elektrisk forskyvning	C/m ²
E_0	Amplitude til elektrisk felt	V/m
E	Elektrisk feltstyrke	V/m
E_{rms}	Root-mean-square elektrisk feltstyrke	V/m
f	Frekvens til mikrobølger	Hz
F	Fettinnhold i prosent	
h	Varmeoverførings koeffisient ved overflaten	W/m ² °C
H	Magnetisk felt intensitet	A/m
J	Strøm flux	A/m ²
k	Termisk konduktivitet	W/mK
P(t)	Poynting vektor	
Q	Volumetrisk varmerate	W/m ³
t	Tid	s
T	Temperatur	°C
v	Hastighet på transportbånd	m/s
x	Distanse	m
W	Vanninnhold	l/l
δ_p	Penetreringsdybde	m
ϵ_0	Vakuums permittivitet	8,854 * 10 ⁻¹² F/m
ϵ''	Dielektrisk taps faktor	
ϵ'	Dielektrisk konstant	
μ	Magnetisk permitivitet til materialet	H/m
μ_0	Magnetisk permitivitet til vakum	1.256 * 10 ⁻⁶ H/m
ρ_0	Fri elektrisk ladningstetthet	C/m ³
ρ	Tetthet	kg/m ³
σ	Elektrisk konduktivitet	S/m

Tabell A.1: Symbolliste

Tillegg B

Forklaring av kode

Begynner med forklaring av variable, se tabell B.1

B.1 Forklaring av totalE.m

Vektorene `meat` og `F` forteller om det kommer inn en ny slice kjøtt ved gitte tidspunkt. `meat` er 0 eller 1, `F` forteller hva fettprosenten til den gitte slicen er. Ved å sette `meat(1)=1` og `F(1)=25` vil det eksempelvis komme inne en kjøttslice med 25% fett ved første tidssteg. Dersom man vil sende inn en hel kjøttblokk, settes `meat(1)`, `meat(2)`... `meat(5)` til 1, med tilhørende fettprosent i `F(1)`...`F(5)`.

Det er `N` tidssteg, og `2*N` pådrag. For hver slice tar det 20 tidssteg, 300 sek, å passere tunnelen. Dersom den siste slicen som annkommer tunnelen ligger i `meat(5)`, må `N` utvides til `N+5` tidssteg, slik at simuleringen kan kjøres ferdig å den siste slicen kommer ut av tunnelen. I denne filen var pådraget til de to elementene det samme ved hvert tidssteg, og ble satt til halvparten av maks effekt. I simuleringen tas det hensyn til at det i virkeligheten er mikrobølge elementer på både under og oversiden i tunnelen. Dette er også vist i koden. Koden er bygd opp av tre for løkker, en for tid, en for `x`-retning og en for `y`-retning. `x`-retning forteller hvor i tunnelen kjøttet befinner seg, `y`-retning forteller hvor dypt i kjøttet, altså hvilket lag (overflate, midt i, bunn), man gjør beregninger på. For hvert tidssteg regnes så temperaturene i alle lag i kjøttslisen ut. Dette gjøres for alle deler av tunnelen hvor det befinner seg en slice, og lagres i `Tarr` slik at `Tarr` inneholder temperaturen til alle lag i kjøttslisene som er i tunnelen. Dersom nåværende del av tunnelen er tom for kjøtt, hopper man videre i løkken for å sjekke neste del av tunnelen. Først sjekkes første del av tunnelen for å se om det nettopp har kommet inn en kjøttslice. Dersom det kommer inn kjøtt settes temperaturen der lik -18 grader. Så sjekkes resten av tunnelen for å se om det er slicer i de resterende delene. Dersom nåværende del av tunnelen er tom for kjøtt, hopper man videre i løkken for å sjekke neste del av tunnelen Når det er kjøtt på gitte posisjon i tunnelen regnes temperaturen ut for alle lag i slicen. Først regnes lag nr to ut, så alle lag ned til det nest siste laget. Til slutt benyttes grensebetingelsene for å finne temperaturen på overflaten og bunnen. Bunnen vil være varmere enn det nest nederste laget i simuleringen. Dette strider imot virkeligheten og kommer av grensebetingelsen for bunnen. Siden det nederste laget settes lik det nest nederste, og varmen fra undersiden er større i bunnen enn det nest nederste laget, blir bunnen varmere.

l	lengde på kjøttblokk
o	overflateareal
dy	lengde på dybdesteg
Ny	antall dybdeesteg
dx	lengde på breddesteg
Nx	antall breddesteg i tunnel
v	hastighet på bånd
dk	tidssteg
Nk	antall tidssteg
Tarr	Matrise med temperaturer
T	Matrise med temperaturer
Q / Q_under	Matriser for effekt
q1 / q2	Pådrag fra hver av de to elementene
isMeat	Vektor som sier hvor i tunnelen det er kjøtt
meat	Vektor som sier hvilke tidspunkt det kommer kjøtt inn i tunnelen
fat	Vektor som sier hva fettprosenten til kjøttet i tunnelen er
F	Samme som meat, bare med fettprosent
x	Vektor med optimale pådrag
Q_vek1 / Q_vek2	Vektor for lagring av pådrag, benyttes i objektfunksjon
F1	Objektfunksjon
Andre	Oppdaterings variable

Tabell B.1: Forklaring av variable i kode

B.1.1 Kode totalE.m

```
l=0.5;
o=l*0.36; % Overflateareal til kjøttblokkene

dy=0.01; %
Ny=round(0.07/dy); % antall dybdesteg

dx=0.1;
Nx=round(2/dx); % antall breddesteg

v=0.4/60;

dk=dx/v; % tidsteg [sek/steg]
Nk=300/dk; %+ 5; % antall tidssteg

Tarr = zeros(Ny,Nx+1,Nk+1); % lager temperatur matrise
T     = zeros(Ny,Nx+1);

%% Definerer Tny for å lagre ny T i
Tnew = zeros(Ny,Nx+1);
T_a  = 10; % omgivelses temperatur

isMeat   = zeros(1,Nx+1);
meat     = zeros(1,Nk+1);
isMeatNew = zeros(1,Nx+1);
fatNew   = zeros(1,Nx+1);

Q=zeros(Nx,Ny); % energi matrise
Q_under=zeros(Nx,Ny); % energi matrise
q1=1/2;
q2=1/2;

r=dk/dy^2;

F=zeros(1,Nk+1); % Matrise for fettprosent i kjøttet
fat=zeros(1,Nx+1);

p=1057; % tetthet kg/m^3

h=5; % varmeovergangstall mellom kjøtt og luft W/m^2K

% Setter inn kjøttstripe for k=1 og k=5
meat(1) = 1;
F(1)=5;
meat(5) = 1;
F(5)=5;
```

```

for k=1:Nk

% Starter med å legge inn grensebetingelsene for i = 1 først.
% Antagelsen her er at denne første stripen får tilført materiale fra
% utenfor ved beregningstidspunktet.

isMeat(1) = meat(k);
fat(1)     = meat(k)*F(k); % Setter fettprosent til slice
if (isMeat(1)==1)
    ti = -18.0; % Initial temperatur
else
    ti = 0.0;
end

% Grensebetingelse for første del av tunnelen
for j=1:Ny
    T(j,1)     = ti;
    Tnew(j,1)  = T(j,1);
    Tarr(j,1,k)= T(j,1); % Dersom det kommer inn mer kjøtt, må temperaturen
                        % settes ved x=1 for nåværende tidsskritt.
end

% Resten av tunnelen
for i=2:Nx+1
% sjekker på forrige slice, hvorvidt det er kjøtt der
    if (isMeat(i-1)==1)

        %Effekt fra oppvarming fra oversiden
        Q(i-1,1)=q1/o*uarr1(i-1,Nx) + q2/o*uarr2(i-1,Nx);
        for j=2:Ny-1
            Q(i-1,j) = Q(i-1,j-1)*exp(-dy/dp(T(j-1,i-1)));
            % Lagrer ny T i Tnew, så en ikke benytter oppdaterte T-er
            % som en gammel T
            Tnew(j,i) = T(j,i) - v*dk/dx*T(j,i) + v*dk/dx*T(j,i-1) + ...
                r*ko(fat(i-1),T(j,i-1))/p/cp(fat(i-1),T(j,i-1))*(T(j+1,i-1)- ...
                2*T(j,i-1)+T(j-1,i-1)) + dk*Q(i-1,j)/p/cp(fat(i-1),T(j,i-1));
        end

        Tnew(1,i) = (Tnew(2,i) + T_a*dy*h/ko(fat(i-1),T(2,i-1)))/...
            (1+dy*h/ko(fat(i-1),T(2,i-1)));
        Tnew(Ny,i) = Tnew(Ny-1,i);
        Q(i-1,Ny) = Q(i-1,Ny-1)*exp(-dy/dp(T(Ny-1,i-1))); %

        %Effekt fra oppvarming fra undersiden
        Q_under(i-1,Ny) = Q(i-1,Ny);
        for j=Ny:-1:2
            Q_under(i-1,j-1) = Q_under(i-1,j)*exp(-dy/dp(T(j-1,i-1)));

            Tnew(j,i) = Tnew(j,i) + dk*Q_under(i-1,j)/p/cp(fat(i-1),T(j,i-1));
        end
    end
end

```

```

        %Undersidens effekt på overflaten:
        Tnew(1,i) = Tnew(1,i) + dk*Q_under(i-1,1)/p/cp(fat(i-1),T(j,i-1));

    %Setter temp = null der det ikke er kjøtt
    else
        for j=1:Ny
            Tnew(j,i) = 0.0;
        end
    end
    isMeatNew(i) = isMeat(i-1); % Lagrer gamle variable
    fatNew(i)    = fat(i-1); % og oppdaterer nye
end

% Oppdatering
Tarr(:, :,k+1) = Tnew;
T              = Tnew;
fat            = fatNew;
isMeat        = isMeatNew;
end

```

B.2 Forklaring av funksjon.m, constr.m og ulin.m

B.2.1 funksjon.m

funksjon.m benyttes av fmincon for å optimalisere pådragene.

funksjon er mye av det samme som totalE, med noen forandringer.

Det er N tidssteg, og $2*N$ forskjellige pådrag. x vil bestå av de $2*N$ pådragene, organisert slik at de N pådragene til $q1$ kommer først, og de N pådragene til $q2$ kommer sist.

Tiden simuleres fra 1 til N tidssteg. Ved hvert tidssteg settes $q1$ og $q2$ ved å hente ut nye verdier fra x vektoren. $q1$ og $q2$ er pådraget fra de to elementene. Q_vek1 og Q_vek2 lagrer de forskjellige pådragene til $q1$ og $q2$ og benyttes senere i F1 for å regne ut objektfunksjonen.

Ellers simuleres temperaturen til kjøttslisene på samme måte som i totalE. Det som er forskjellig mellom filen funksjon og filen totalE er at temperaturen ved gitt tid, lagres i $Tarr(:, :,k+1)$ i totalE. I funksjon lagres temperaturen ved gitt tid i $Tarr(:, :,k)$. Man slipper dermed å simulere i $N+1$ tidsskritt, slik man må i totalE for å få med effekten pådraget gir i siste tidsskritt. Dersom funksjon også måtte simuleres i $Nk + 1$ tiddsteg, måtte man innført to ekstra pådrag for å få de forskjellige vektorene til å passe, og det ble bare rot. Ellers er det liten forskjell på de to filene. Til slutt regnes objektfunksjonen ut.

Etter at de optimale pådragene er regnet ut er det hensiktsmessig med plot av temperaturen. For å lage disse plottene benyttes x vektoren med pådrag i filen totalE.

B.2.2 Kode funksjon.m

```
function F1 = funksjon(x)

l=0.5;
o=l*0.36; % Overflateareal til kjøttblokkene

dy=0.01; %
Ny=round(0.07/dy); % antall dybdesteg

dx=0.1;
Nx=round(2/dx); % antall breddesteg

v=0.4/60;

dk=dx/v; % tidsteg [sek/steg]
Nk=300/dk + 1-1; % antall tidssteg

Tarr = zeros(Ny,Nx+1,Nk+1); % temperatur matrise
T     = zeros(Ny,Nx+1);

Tnew = zeros(Ny,Nx+1);
T_a  = 10; % omgivelses temperatur

isMeat    = zeros(1,Nx+1);
meat      = zeros(1,Nk+1);
isMeatNew = zeros(1,Nx+1);
fatNew    = zeros(1,Nx+1);

Q=zeros(Nx,Ny); % energi matrise
Q_under=zeros(Nx,Ny);

r=dk/dy^2;

F=zeros(1,Nk+1); % fettprosent i kjøttet
fat=zeros(1,Nx+1);

p=1057; % tetthet kg/m^3

h=5; % varmeovergangstall mellom kjøtt og luft W/m^2K

% Setter inn kjøttstripe for k=1
meat(1) = 1;
F(1)=25;

for k=1:Nk
```

```

q1=x(k);
q2=x(Nk+k);

Q_vek1(k)=x(k);
Q_vek2(k)=x(Nk+k);

% Antagelsen her er at denne første stripen får tilført materiale fra
% utenfor ved beregningstidspunktet.

isMeat(1) = meat(k);
fat(1)     = meat(k)*F(k); %
if (isMeat(1)==1)
    ti = -18.0;
else
    ti = 0.0;
end

for j=1:Ny
    T(j,1) = ti;
end

for i=2:Nx+1 % Sjekker på forrige slice, hvorvidt det er kjøtt der
    if (isMeat(i-1)==1)

        %Effekt fra oppvarming fra oversiden
        Q(i-1,1)=q1/o*uarr1(i-1,Nx) + q2/o*uarr2(i-1,Nx);
        for j=2:Ny-1
            Q(i-1,j) = Q(i-1,j-1)*exp(-dy/dp(T(j-1,i-1)));
            % Lagrer ny T i Tnew, så en ikke benytter oppdaterte T-er
            % som en gammel T
            Tnew(j,i) = T(j,i) - v*dk/dx*T(j,i) + v*dk/dx*T(j,i-1) + ...
                r*ko(fat(i-1),T(j,i-1))/p/cp(fat(i-1),T(j,i-1))*(T(j+1,i-1)-...
                2*T(j,i-1)+T(j-1,i-1)) + dk*Q(i-1,j)/p/cp(fat(i-1),T(j,i-1));

        end

        Tnew(1,i) = (Tnew(2,i) + T_a*dy*h/ko(fat(i-1),T(2,i-1)))/...
            (1+dy*h/ko(fat(i-1),T(2,i-1)));
        Tnew(Ny,i) = Tnew(Ny-1,i);
        Q(i-1,Ny) = Q(i-1,Ny-1)*exp(-dy/dp(T(Ny-1,i-1)));

        %Effekt fra oppvarming fra undersiden
        Q_under(i-1,Ny) = Q(i-1,Ny);
        for j=Ny:-1:2
            Q_under(i-1,j-1) = Q_under(i-1,j)*exp(-dy/dp(T(j-1,i-1)));

            Tnew(j,i) = Tnew(j,i) + dk*Q_under(i-1,j)/p/cp(fat(i-1),T(j,i-1));
        end
    end
end

```

```

        %Undersidens effekt på overflaten:

        Tnew(1,i) = Tnew(1,i) + dk*Q_under(i-1,1)/p/cp(fat(i-1),T(j,i-1));
% Setter temp = null der det ikke er kjøtt
else
    for j=1:Ny
        Tnew(j,i) = 0.0;
    end
end
isMeatNew(i) = isMeat(i-1);
fatNew(i)     = fat(i-1);
end

Tarr(:, :,k)   =   Tnew;
T              =   Tnew;
fat           =   fatNew;
isMeat       =   isMeatNew;

end

beta=1;
%alfa=1;
F1=beta*(Q_vek1*Q_vek1'+Q_vek2*Q_vek2'); % Benyttes med likhetsbetingelse
% F1=alfa*(Tarr(1,Nx+1,Nk)-T_set)^2 + beta*(Q_vek1*Q_vek1'+Q_vek2*Q_vek2'); Benyttes
% ved objektfunksjon.

Tarr(1,Nx+1,Nk)

```

B.2.3 constr.m

Denne filen lager matrisene A og b og setter initialt pådrag x_0 . Deretter kaller den fmincon, hvor den sender inn funksjon.m, x_0 , A, b og ulin.m som argumenter. Det optimale pådraget man så ender opp med lagres i x vektoren. Summen av pådragsfraksjoner regnes også ut her.

B.2.4 Kode constr.m

```

clear all
ll=0;
ul=1;

% N=20;
N=14;

%Lager A og B matrise til q1
A1=[eye(20+N);- eye(20+N)];
B1=[ul*ones(20+N,1); ll*ones(20+N,1)];

```

```

%Lager A og B matrise til q2
A2=[eye(20+N);- eye(20+N)];
B2=[ul*ones(20+N,1); ll*ones(20+N,1)];

A=[A1 zeros((20+N)*2,20+N); zeros((20+N)*2,20+N) A2];
B=[B1; B2];
x_0=zeros((20+N)*2,1);

% options = optimset('MaxFunEvals',[(40+N)*10000]);

x = fmincon(@funksjon,x_0,A,B); %Uten likhetsbetingelse
% x = fmincon(@funksjon,x_0,A,B,[],[],[],[],@ulin); % Med likhetsbetingelse
% x = fmincon(@fullopt,x_0,A,B,[],[],[],[],@ulin); % Fullt system
x1=x(1:20+N);
x2=x(20+N+1:(20+N)*2);

summen=0;
for i=1:length(x)
    summen = summen + x(i);
end

```

B.2.5 ulin.m

ulin.m er akkurat det samme som funksjon.m, bortsett fra at ulin.m returnerer to parametre, c og ceq. Objektfunksjonen F1 blir heller ikke regnet ut i ulin.m, men c og ceq blir istedet regnet ut. Det er likevel den samme simuleringen og ulin.m tar også inn pådragsvektoren x. ulin.m forandres etter hvilken metode som brukes. Dersom det er likhetsbetingelser returneres ceq. Dersom det er ulikhetsbetingelser returneres c. c og ceq er nesten de samme, men har ulik betydning for fmincon.

B.2.6 Kode ulin.m

```

{}

l=0.5;
o=l*0.36; % Overflateareal til kjøttblokkene

dy=0.01; %
Ny=round(0.07/dy); % antall dybdesteg

dx=0.1;
Nx=round(2/dx); % antall breddesteg

v=0.4/60;

dk=dx/v; % tidsteg [sek/steg]
Nk=300/dk + 1-1; % antall tidssteg

```



```
Tarr = zeros(Ny,Nx+1,Nk+1); % temperatur matrise
T    = zeros(Ny,Nx+1);

Tnew = zeros(Ny,Nx+1);
T_a  = 10; % omgivelses temperatur

isMeat    = zeros(1,Nx+1);
meat      = zeros(1,Nk+1);
isMeatNew = zeros(1,Nx+1);
fatNew    = zeros(1,Nx+1);

Q=zeros(Nx,Ny); % energi matrise
Q_under=zeros(Nx,Ny);

r=dk/dy^2;

F=zeros(1,Nk+1); % fettprosent i kjøttet
fat=zeros(1,Nx+1);

p=1057; % tetthet kg/m^3

h=5; % varmeovergangstall mellom kjøtt og luft W/m^2K

% Setter inn kjøttstriper
for i =1:N+1
    meat(i)=1;
    F(i)=5;
end

for k=1:Nk

    q1=x(k);
    q2=x(Nk+k);

    Q_vek1(k)=x(k);
    Q_vek2(k)=x(Nk+k);

    % Antagelsen her er at denne første stripen får tilført materiale fra
    % utenfor ved beregningstidspunktet.

    isMeat(1) = meat(k);
    fat(1)     = meat(k)*F(k); %
    if (isMeat(1)==1)
        ti = -18.0;
    else
        ti = 0.0;
    end
end
```

```

for j=1:Ny
    T(j,1) = ti;
end

for i=2:Nx+1 % Sjekker på forrige slice, hvorvidt det er kjøtt der
    if (isMeat(i-1)==1)

        %Effekt fra oppvarming fra oversiden
        Q(i-1,1)=q1/o*uarr1(i-1,Nx) + q2/o*uarr2(i-1,Nx);
        for j=2:Ny-1
            Q(i-1,j) = Q(i-1,j-1)*exp(-dy/dp(T(j-1,i-1)));
            % Lagrer ny T i Tnew, så en ikke benytter oppdaterte T-er
            % som en gammel T
            Tnew(j,i) = T(j,i) - v*dk/dx*T(j,i) + v*dk/dx*T(j,i-1) + ...
                r*ko(fat(i-1),T(j,i-1))/p/cp(fat(i-1),T(j,i-1))*(T(j+1,i-1)-...
                2*T(j,i-1)+T(j-1,i-1)) + dk*Q(i-1,j)/p/cp(fat(i-1),T(j,i-1));

        end

        Tnew(1,i) = (Tnew(2,i) + T_a*dy*h/ko(fat(i-1),T(2,i-1)))/...
            (1+dy*h/ko(fat(i-1),T(2,i-1)));
        Tnew(Ny,i) = Tnew(Ny-1,i);
        Q(i-1,Ny) = Q(i-1,Ny-1)*exp(-dy/dp(T(Ny-1,i-1)));

        %Effekt fra oppvarming fra undersiden
        Q_under(i-1,Ny) = Q(i-1,Ny);
        for j=Ny:-1:2
            Q_under(i-1,j-1) = Q_under(i-1,j)*exp(-dy/dp(T(j-1,i-1)));

            Tnew(j,i) = Tnew(j,i) + dk*Q_under(i-1,j)/p/cp(fat(i-1),T(j,i-1));
        end

        %Undersidens effekt på overflaten:

        Tnew(1,i) = Tnew(1,i) + dk*Q_under(i-1,1)/p/cp(fat(i-1),T(j,i-1));
        % Setter temp = null der det ikke er kjøtt
        else
            for j=1:Ny
                Tnew(j,i) = 0.0;
            end
        end
        isMeatNew(i) = isMeat(i-1);
        fatNew(i) = fat(i-1);
    end
end

Tarr(:, :, k) = Tnew;
T = Tnew;
fat = fatNew;

```

```

        isMeat          =   isMeatNew;

end

% Ulikhet/likhets betingelser
T_set=-2;
c=[];
ceq=[];
for i=1:length(F)
    if meat(i)==1
        ceq(i)=Tarr(1,Nx+1,Nk-N-1+i)-T_set;
%         c(i)=Tarr(1,Nx+1,Nk-N-1+i)-T_set;

    end
end
c=c*(-1);

```

B.3 Forklaring av modellogprosess.m

Denne filen brukes for å teste ut modellfeil. Både antatt modell og virkelig prosess kjøres her. De optimale pådragene funnet av constr.m benyttes i modellen og prosessen. Det er også implementert regulator for den virkelige prosessen for å motvirke modellfeil. Den første regulatoren som er beskrevet i oppgaven og den modifiserte er implementert i denne filen.

B.3.1 Kode modellogprosess.m

```

N=95;
l=0.5;
o=l*0.36; % Overflateareal til kjøttblokkene

dy=0.01; %
Ny=round(0.07/dy); % antall dybdesteg

dx=0.1;
Nx=round(2/dx); % antall breddesteg

v=0.4/60;

dk=dx/v; % tidsteg [sek/steg]
Nk=300/dk + N ; % antall tidssteg

Tarr = zeros(Ny,Nx+1,Nk+1); % lager temperatur matrise
T     = zeros(Ny,Nx+1);
TarrP = zeros(Ny,Nx+1,Nk+1); % lager temperatur matrise
TP    = zeros(Ny,Nx+1);

%% SS Definerer Tny for å lagre ny T i

```

```
Tnew = zeros(Ny,Nx+1);
TnewP = zeros(Ny,Nx+1);

T_a = 10; % omgivelses temperatur

isMeat = zeros(1,Nx+1);
meat = zeros(1,Nk+1);
isMeatNew = zeros(1,Nx+1);
fatNew = zeros(1,Nx+1);

Q=zeros(Nx,Ny); % energi matrise
Q_under=zeros(Nx,Ny);

r=dk/dy^2;

F=zeros(1,Nk+1); % fettprosent i kjøttet
fat=zeros(1,Nx+1);

p=1057; % tetthet kg/m^3

h=5; % varmeovergangstall mellom kjøtt og luft W/m^2K

e=0;
%% Modell med antatte parametre:

for i =1:N+1
    meat(i)=1;
    F(i)=5;
end

for k=1:Nk

    q1=x(k);
    q2=x(Nk+k);
    isMeat(1) = meat(k);
    fat(1) = meat(k)*F(k);
    if (isMeat(1)==1)
        ti = -18.0;
    else
        ti = 0.0;
    end

    for j=1:Ny
        T(j,1) = ti;
        Tnew(j,1) = T(j,1);
        Tarr(j,1,k)= T(j,1); % Dersom det kommer inn mer kjøtt, må temperaturen settes ved x
    end

    for i=2:Nx+1

        if (isMeat(i-1)==1)
```

```

%Effekt fra oppvarming fra oversiden
Q(i-1,1)=q1/o*uarr1(i-1,Nx) + q2/o*uarr2(i-1,Nx);
for j=2:Ny-1
    Q(i-1,j) = Q(i-1,j-1)*exp(-dy/dp(T(j-1,i-1)));

    Tnew(j,i) = T(j,i) - v*dk/dx*T(j,i) + v*dk/dx*T(j,i-1) + ...
        r*ko(fat(i-1),T(j,i-1))/p/cp(fat(i-1),T(j,i-1))*(T(j+1,i-1)-2*T(j,i-1)+T(j-1,i-1))
        + dk*Q(i-1,j)/p/cp(fat(i-1),T(j,i-1));

end

%% SS Endret til Tnew
Tnew(1,i) = (Tnew(2,i) + T_a*dy*h/ko(fat(i-1),T(2,i-1)))/(1+dy*h/ko(fat(i-1),T(2,i-1)));
Tnew(Ny,i) = Tnew(Ny-1,i);
Q(i-1,Ny) = Q(i-1,Ny-1)*exp(-dy/dp(T(Ny-1,i-1)));

%Effekt fra oppvarming fra undersiden
Q_under(i-1,Ny) = Q(i-1,Ny);
for j=Ny:-1:2
    Q_under(i-1,j-1) = Q_under(i-1,j)*exp(-dy/dp(T(j-1,i-1)));

    Tnew(j,i) = Tnew(j,i) + dk*Q_under(i-1,j)/p/cp(fat(i-1),T(j,i-1));
end

%Undersidens effekt på overflaten:

Tnew(1,i) = Tnew(1,i) + dk*Q_under(i-1,1)/p/cp(fat(i-1),T(j,i-1));

%% Setter temp = null der det ikke er kjøtt
else
    for j=1:Ny
        Tnew(j,i) = 0.0;
    end
end
isMeatNew(i) = isMeat(i-1);
fatNew(i) = fat(i-1);
end

Tarr(:, :, k+1) = Tnew;
T = Tnew;
fat = fatNew;
isMeat = isMeatNew;
end

%% Prosess med usikre parametre

```

```
isMeat    = zeros(1,Nx+1);
meat      = zeros(1,Nk+1);
isMeatNew = zeros(1,Nx+1);
fatNew    = zeros(1,Nx+1);

for i =1:N+1
    meat(i)=1;
    F(i)=25;
end

just=ones(1,Nk+1);

for k=1:Nk

    Q_vek1(Nk-k+1)=x(Nk-k+1);
    Q_vek2(Nk-k+1)=x(2*Nk-k+1);
    if(just(k)~=1)
%       q1= x(k) + x(k)*just(k)*5;    % Første Regulator
%       q2 = x(Nk+k) + x(Nk+k)*just(k)*5;

%% Modifisert regulator
        q1 = x(k);
        q2 = x(Nk+k) + x(Nk+k)*just(k);

        if(q1>1)
            q1=1;
        elseif(q1<0)
            q1=0;
        end
        if(q2>1)
            q2=1;
        elseif(q2<0)
            q2=0;
        end
    else
        q1= x(k) ;
        q2= x(Nk+k) ;
    end

    isMeat(1) = meat(k);
    fat(1)    = meat(k)*F(k);
    if (isMeat(1)==1)
        ti = -18.0;
    else
        ti = 0.0;
    end

    for j=1:Ny
```

```

    TP(j,1)      = ti;
    TnewP (j,1) = TP(j,1);
    TarrP(j,1,k)= TP(j,1);
end

for i=2:Nx+1
    if (isMeat(i-1)==1)

        %Effekt fra oppvarming fra oversiden
        Q(i-1,1)=q1/o*uarr1(i-1,Nx) + q2/o*uarr2(i-1,Nx);    %% SS i -> i-1; også lenger
        for j=2:Ny-1
            Q(i-1,j) = Q(i-1,j-1)*exp(-dy/dp(TP(j-1,i-1)));

            TnewP(j,i) = TP(j,i) - v*dk/dx*TP(j,i) + v*dk/dx*TP(j,i-1) + ...
                r*ko(fat(i-1),TP(j,i-1))/p/cp(fat(i-1),TP(j,i-1))*(TP(j+1,i-1)-2*TP(j,i-1))
                + dk*Q(i-1,j)/p/cp(fat(i-1),TP(j,i-1));

        end

        TnewP(1,i) = (TnewP(2,i) + T_a*dy*h/ko(fat(i-1),TP(2,i-1)))/(1+dy*h/ko(fat(i-1),TP(2,i-1)));
        TnewP(Ny,i) = TnewP(Ny-1,i);
        Q(i-1,Ny)   = Q(i-1,Ny-1)*exp(-dy/dp(TP(Ny-1,i-1)));

        %Effekt fra oppvarming fra undersiden
        Q_under(i-1,Ny) = Q(i-1,Ny);
        for j=Ny:-1:2
            Q_under(i-1,j-1) = Q_under(i-1,j)*exp(-dy/dp(TP(j-1,i-1)));

            TnewP(j,i) = TnewP(j,i) + dk*Q_under(i-1,j)/p/cp(fat(i-1),TP(j,i-1));
        end

        %Undersidens effekt på overflaten:

        TnewP(1,i) = TnewP(1,i) + dk*Q_under(i-1,1)/p/cp(fat(i-1),TP(j,i-1));

        %% Setter temp = null der det ikke er kjøtt
        else
            for j=1:Ny
                TnewP(j,i) = 0.0;
            end
        end
        isMeatNew(i) = isMeat(i-1);
        fatNew(i)    = fat(i-1);
    end
end

TarrP(:, :, k+1) = TnewP;
TP               = TnewP;

```

```
fat          = fatNew;
isMeat       = isMeatNew;

%% Første regulator
%   if(TarrP(1,Nx+1,k+1)~=Tarr(1,Nx+1,k+1))
%       if Tarr(1,Nx+1,k+1)<TarrP(1,Nx+1,k+1);
%           just(k+1) = just(k) + (TarrP(1,Nx+1,k+1) - Tarr(1,Nx+1,k+1));
%       else
%           just(k+1) = (Tarr(1,Nx+1,k+1) - TarrP(1,Nx+1,k+1));
%       end
%   end

%% Modifisert regulator
if (k==21)|| (k==36)|| (k==51)|| (k==66)|| (k==81)|| (k==96)
    if(TarrP(1,Nx+1,k+1)~=Tarr(1,Nx+1,k+1))
        e = e + Tarr(1,Nx+1,k+1) - TarrP(1,Nx+1,k+1);

        for t=1:15
            just(k+t) = e;

        end

    end
end
end
end
```