

INFrysNING AV LIVSMEDEL - SIMULERING AV INDUSTRIELL PROCESS

ULF ADAMSSON

INSTITUTIONEN FÖR REGLERTEKNIK
LUNDS TEKNISKA HÖGSKOLA
MAJ 1985

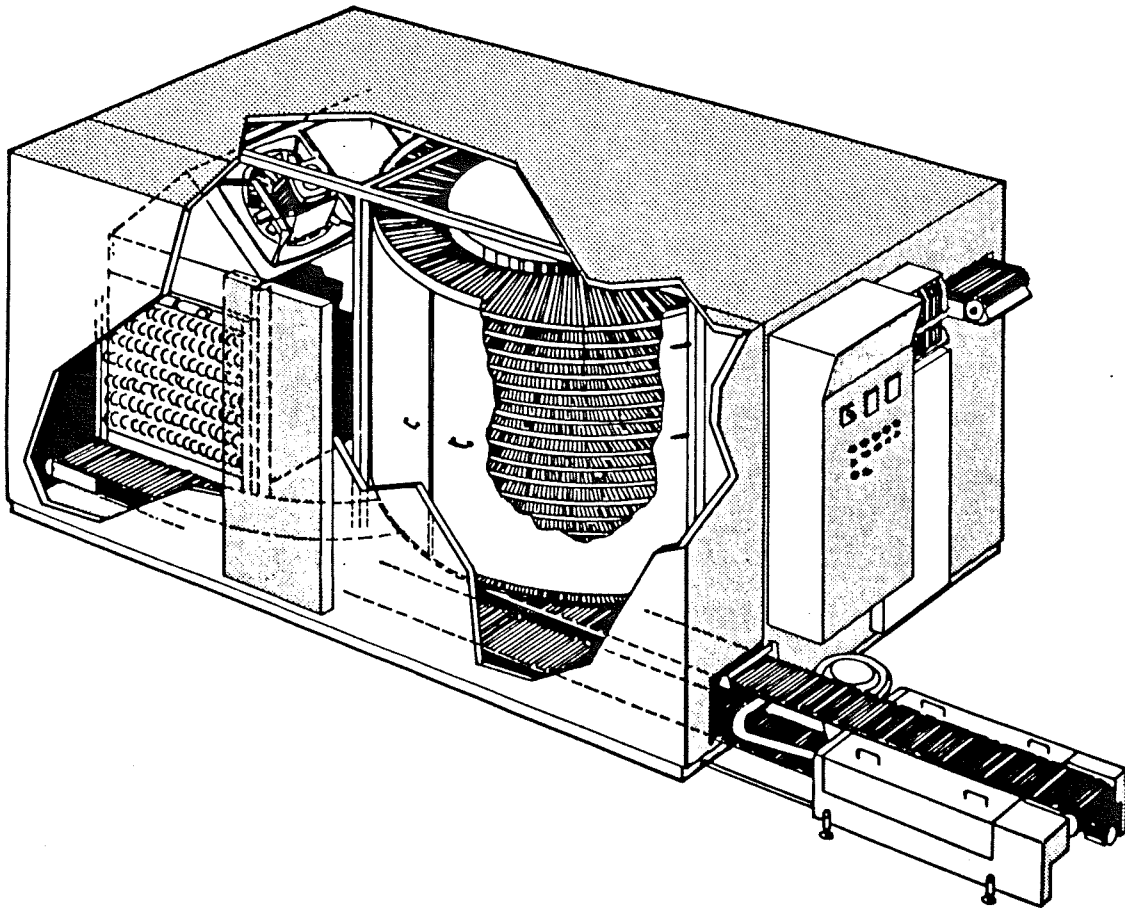
LUND INSTITUTE OF TECHNOLOGY DEPARTMENT OF AUTOMATIC CONTROL Box 118 S 221 00 Lund Sweden		Document name Master thesis	
		Date of issue May	
		Document number CODEN:LUTFD2/(TFRT-5322)/1-096/(1985)	
Author(s) Ulf Adamsson		Supervisor Björn Wittenmark	
		Sponsoring organization	
Title and subtitle Infrysning av livsmedel - Simulering av en industriell process. (Freezing of foods - Simulation of an industrial process.)			
Abstract Design and operation of industrial freezers contains the crucial problem of how to make the frozen foods attain a prescribed temperature. Methods for prediction of freezing times under given constant conditions are commonly used in order to calculate the optimum product flow through continuous freezers. However, in reality the processes are time variable, due to disturbances and parameter deviations. Thus, introduction of real time temperature measurements combined with manual or automatic control would be a feasible way to obtain improved reliability, with regard to product quality. Freezing under time-varying conditions is studied, using a numerical solution method for the heat conduction equation with temperature-dependent coefficients. A mathematical model of the freezing process is described, and a Pascal program for simulation of the Frigoscandia Gyrofreeze (spiral freezer) is presented. Finally, the process dynamics are discussed and a regulator structure, based on product temperature feedback combined with feedforward from measurable disturbances (mainly air temperatures), is proposed.			
Key words			
Classification system and/or index terms (if any)			
Supplementary bibliographical information			
ISSN and key title			ISBN
Language Swedish	Number of pages 96	Recipient's notes	
Security classification			

DOKUMENTATABLAD RT 3/81

Distribution: The report may be ordered from the Department of Automatic Control or borrowed through the University Library 2, Box 1010, S-221 03 Lund, Sweden, Telex: 33248 Lubbis Lund.

**INFrysNING AV LIVSMEDEL
— SIMULERING AV EN
INDUSTRIELL PROCESS**

EXAMENSARBETE



**AV ULF ADAMSSON
UTFÖRT FÖR FRIGOSCANDIA
VID INST FÖR REGLERTEKNIK
LUNDS TEKNISKA HÖGSKOLA**

ABSTRACTS

Design and operation of industrial freezers contains the crucial problem of how to make the frozen foods attain a prescribed temperature. Methods for prediction of freezing times under given constant conditions are commonly used in order to calculate the optimum product flow through continuous freezers. However, in reality the processes are time variable, due to disturbances and parameter deviations. Thus, introduction of real time temperature measurements combined with manual or automatic control would be a feasible way to obtain improved reliability, with regard to product quality.

Freezing under time-varying conditions is studied, using a numerical solution method for the heat conduction equation with temperature-dependent coefficients. A mathematical model of the freezing process is described, and a Pascal program for simulation of the Frigoscandia Gyrofreeze (spiral freezer) is presented. Finally, the process dynamics are discussed and a regulator structure, based on product temperature feedback combined with feedforward from measurable disturbances (mainly air temperatures), is proposed.

Konstruktion och drift av industriella frysare innefattar det avgörande problemet att få det infryssta livsmedlet att anta en föreskriven temperatur. Metoder för att förutsäga infrysningstider under givna konstanta förhållanden är de som vanligen används för att beräkna det optimala produktflödet genom kontinuerliga frysare. Emellertid är processerna i verkligheten tidsvariabla, på grund av störningar och parameteravvikelser. Därför skulle ett införande av kontinuerliga mätningar kombinerat med manuell eller automatisk reglering vara en möjlighet att uppnå förbättrad tillförlitlighet, med avseende på produktkvaliteten.

Infrysning under tidsvariabla förhållanden studeras, med hjälp av en numerisk lösningsmetod för värmeledningsekvationen med temperaturberoende koefficienter. En matematisk modell av infrysningsprocessen beskrivs, och ett Pascal-program för simulering av Frigoscandias Gyrofreeze (spiralfrysare) presenteras. Till sist diskuteras processdynamiken, och en regulatorstruktur, baserad på återkoppling från produkttemperaturen kombinerat med framkoppling från mätbara störningar (främst lufttemperaturer), föreslås.

INNEHÅLL

1	INLEDNING	1
1.1	Modell av infrysningsprocess	2
1.1.1	Frysarens funktion	2
1.1.2	Produkten	2
1.1.3	Modellbygge	3
1.2	Frysaren i drift	5
1.2.1	Förutsättningar och kriterier	5
1.2.2	Störningar	5
1.2.3	Reglering	6
1.2.4	Mätning	7
2	VARMELEDNINGSEKVATIONEN	9
2.1	Numerisk lösning av VLE	10
2.1.1	Differensapproximation	10
2.1.2	Fasövergång	12
2.1.3	Randvillkor	13
2.1.4	Algoritm	14
2.2	Val av koefficienter och parametrar	17
2.2.1	Bestämning av $c_p(T)$ och $\lambda(T)$	17
2.2.2	Val av Δx och Δt	20
2.3	Alternativa metoder	22
2.3.1	Byte av koordinatsystem	22
2.3.2	Crank-Nicolson	24
3	MODELL AV FRYSARE	27
3.1	Luften i frysaren	28
3.1.1	Luftflödet	28
3.1.2	Värmeövergång produkt→luft	30
3.1.3	Lufttemperaturer i stacken	31
3.1.4	Andra värmekällor	32
3.1.5	Kylbatteri	33
3.2	Diskret modell av infrysningsprocessen	34
3.2.1	Beskrivning av principerna	34
3.2.2	Lufttemperaturer	35

3.2.3	Konvergenskontroll	36
3.3	Allmän beskrivning av program GYROFREEZE	38
3.3.1	Principer	38
3.3.2	Val av programspråk	38
3.3.3	Parametrar, tillstånd och hjälpvariabler	39
3.3.4	Tid	39
3.3.5	Filer	40
3.3.6	Användning av programmet	40
4	PROCESSDYNAMIK OCH REGLERING	42
4.1	Modellens egenskaper	43
4.1.1	Stegsvar	43
4.1.2	Utökning/reduktion av modellen	43
4.2	Reglering	50
4.2.1	Kriterier för reglering	50
4.2.2	Regulator	50
4.2.3	Avancerade regulatorer	52
4.2.4	Sampling	53
4.2.5	Åtgärder vid temperaturökning - väljare	53
4.2.6	Framkoppling	54
4.2.7	Sammanfattning	55
5	SAMMANFATTNING	64
6	LITTERATUR	67
7	BILAGOR	68

1 INLEDNING

Hur får man en nyskjuten älg att säkert och smärtfritt (för jägaren) anta en temperatur på under -18°C ? Man styckar den och stoppar delarna i en frysbox. Med lång tid mellan jaktsäsongerna och med hälsovårdsnämnden utom skotthåll, är den lycklige jägaren befriad från den industriella livsmedelsinfrysningens effektivitets- och kvalitetskrav.

Av de två huvudtyperna av infrysningsprocesser, kontinuerlig och 'batch'-infrysning, är det den senare som ligger närmast jägarens frysboxmetod. Ett internationellt exempel kan hämtas från de Nya Zeeländska fårslakterierna, där fårkroppar stuvas in i en frysare vid slutet av arbetsdagen, för att morgonen därpå lastas ut i stelt skick. För att de största baggarna skall få tillräckligt låg temperatur, samtidigt som alltför snabb frysning av de minsta lammen undviks (kvalitetskrav), sekvensstyrts temperaturen i frysaren under natten (temperaturen följer ett givet tidsschema) .

Vid kontinuerlig infrysning lastas produkten på t ex ett band, transporteras in i frysaren, och kommer ut efter en tid (s k hålltiden) som bestäms av transporthastigheten. Här eftersträvas så små variationer som möjligt hos lufttemperatur och, framför allt, livsmedlets utgångstemperatur. Hålltiden måste optimeras: för hård infrysning ger uttorkning av produkten, för lätt infrysning bryter mot de hygieniska kraven. För låg belastning innebär låg effektivitet, för hög medför överbelastning av kylaggregat samt igenfrostning och allmän temperaturökning i frysaren.

Att testa störkänslighet och dynamik hos en frysare under drift, är problematiskt. Experimenten blir tidsödande, det är svårt eller omöjligt att mäta alla variabler, och produktionen störs (testas en frysare utanför produktionslinjen, måste 'testlivsmedel' tillföras...). I stället utförs de grundläggande undersökningarna på produkten för sig, i laboratorium, och på frysaren för sig, t ex test av kylbatteriets kapacitet. Rent teoretiskt kan sedan modeller för produkt och frysare kombineras och processen simuleras på dator.

1.1 Modell av infrysningsprocess

Infrysningsprocessen kan indelas i maskinvara (= frysaren) och produkt (= livsmedlet). Dessa kombineras åter i den teoretiska modellen.

1.1.1 Frysarens funktion

I detta examensarbete görs modellbygge och beräkningar för en kontinuerlig infrysningsprocess med en s k spiralfrysare modell GYROFREEZE Midget 118-406 (se datablad i appendix).

Funktionen hos denna är i princip:

Livsmedel lastas på ett band, vilket transporteras in i ett isolerat hus. Bandet går i spiralform uppåt, samtidigt som det passerar av en nedåtgående, kall luftström. Efter ett visst antal varv i höjled matas bandet ut ur det isolerade huset, avlastas och leds tillbaka till inmatningen.

Luftströmningen runt produkten åstadkommes med en eller flera fläktar. Luften passerar först över produkten och bandet i spiralen, och därefter till ett kylbatteri, där den änyo kyls ned.

Kylbatteriet i frysaren är kopplat till ett maskinrum, som förser batteriet med köldmedium (oftast ammoniak). Trycket hålls så lågt att köldmediet kokar vid mycket låg temperatur. För kokningen erforderlig värme tas via kylflänsarna från den genom batteriet passerande luften, varvid dennas temperatur sänks.

1.1.2 Produkten

Produkten utgörs av livsmedel, som skall frysas ned till -18° från en viss ingångstemperatur. Exempel på produkt är: hamburgare, köttbullar, kycklingdelar, glass, lasagne, pajer, osv. Dessa olika sorters produkter skiljer sig åt i fråga om bl a storlek, form, ingångstemperatur (t ex rå hamburgare / stekt köttbulle) och eventuell förpackning.

Storlek och form påverkar dels hur stor del av bandytan som täcks av produkt, dels den effektiva yta över vilken värmeavgivning till luften sker. Förpackningen inverkar på värmeövergången. Både själva förpackningsmaterialet och eventuell luft mellan detta och produkten fungerar värmeisolerande.

Alla produktparametrar inverkar direkt eller indirekt på frysareffekten och vilken hålltid som erfordras.

Eftersom ingångstemperaturen ligger över eller omkring fryspunkten (t ex rå hamburgare: 0°, stekt köttbulle: 60°), måste produkten både kylas (temperaturen sänks genom värmeledning) och frysas (fasövergång) under avgivande av värme. Produkten i sig är i allmänhet fast men innehåller en stor del vatten. En del av kylningen kommer även av den avdunstning av vatten som förekommer.

1.1.3 Modellbygge

En modell av infrysningsprocessen ska främst beskriva:

*) Värmetransporten: värmeledning i produkten; värmeövergång från produkt till luft och från luft till kylbatteriet.

*) Fasövergång: i ett skikt i produkten, med temperatur strax under 0°C, avges smältentalpin, vilken leds ut mot ytan, medan infrysningsskiktet förskjuts i motsatt riktning.

*) Transport: produkt och luft transporteras resp strömmar i (i princip) motsatta riktningar (motströms värmeväxlare). Produkt med given temperatur matas in; luften cirkuleras med hjälp av fläktar från kylbatteri till produktstack och tillbaka.

Den mer komplicerade matematiska överläggningen gäller kombinationen av värmetransport/fasövergång i produkten. Värmeledningsekvationen har, i en dimension, följande utseende:

$$\rho c_p \frac{\partial T}{\partial t} = \lambda \frac{\partial^2 T}{\partial x^2}$$

$$\left(\begin{array}{l} \rho: \text{densitet} \\ c_p: \text{värmekapacitivet} \\ \lambda: \text{värmeledningskoefficient} \end{array} \right)$$

Vid infrysning måste hänsyn tas till: att värme frigörs i infrysningsskiktet och att värmeledningstal och värmekapacitivet är olika för fryst/icke fryst ämne. Analytisk lösning tarvar följande grova uppdelning: en ekvation för

'frysta' delen ($T < 0$), en ekvation för ickefrysta delen ($T > 0$), och en fasövergångsekvation, som bestämmer frysskiktets förflyttning. Ytterligare approximationer och förutsättningar leder fram till Planck's ekvation, en lösningsmetod för infrysningstider, med några modifikationer tillämpbar både på Nya Zeeländska får och svenska älgbiffar. Under förutsättning att omgivande luftens temperatur under infrysningen är känd, kan hålltiden således beräknas.

Den fullständiga modellen (enligt ovanstående tre punkter) där hänsyn tas till bl a växelverkan produkt/luft, och där t ex temperaturer ses som funktioner av tid och läge i frysaren, kräver emellertid simulering på dator. En numerisk metod måste tillgripas. Värmeledningstal och värmekapacitivet ansätts som kontinuerliga funktioner av temperaturen. så att värmeledningsekvationen kan skrivas på följande form:

$$\rho c_p(T) \frac{\partial T}{\partial t} = \frac{\partial}{\partial x} \left[\lambda(T) \frac{\partial T}{\partial x} \right]$$

Derivatorna överförs till differensapproximationer. Lufttemperaturerna bestäms på grundval av värmeövergång från produkt och till kylbatteri, samt av lufthastighet, belastning osv.

1.2 Frysaren i drift

Den teoretiska modellen kompletteras med en betraktelse av de praktiska förutsättningarna för styrning av infrysningsprocessen.

1.2.1 Förutsättningar och kriterier

En frysare i drift är vanligen inkopplad i en produktionskedja, typ löpande band. Kapaciteten måste vara anpassad efter det önskade produktflödet genom produktionskedjan. I denna kan ingå bl a enheter för framställning av produkten, förpackning (före eller efter infrysning), nedkylning (om ej frysarens kapacitet i sig själv är tillräcklig) och lagring.

Produktionstakten kan variera; ibland kan frysaren gå på tomgång på grund av arbetspauser eller oförutsedda avbrott. Vid uppstart/avstängning t ex i samband med början/slut på en arbetsdag råder också speciella förhållanden.

Under dessa förutsättningar ställer vi främst följande krav på frysaren:

- 1) Produkten skall vid utgången ha en temperatur ej över -18° ('utjämnad' temperatur: produktens entalpi skall vara lika med den hos produkt med homogen temperatur = -18°). Även alltför låg temperatur bör undvikas, för att minimera produktens uttorkning.
- 2) Frysaren skall ha en viss produktivitet, d v s låta tillräckligt mycket produkt passera per tidsenhet.
- 3) Eventuell reglering eller omställning av frysaren under drift bör göras så att störningar på den omgivande produktionskedjan och slitage på frysare och maskinrum undviks.

1.2.2 Störningar

Processen utsätts för ett antal olika störningar, som ökar/minskar produktens utgångstemperatur. De väsentligaste innebär en förändring hos någon av följande variabler:

- *) Lufthastigheten: Värmeövergångstalet varierar med lufthastigheten. En ändring av lufthastigheten påverkar således direkt kyleffekten på produkten. Dessutom medför ett t ex minskat luftflöde att (luft-)temperaturskillnaderna i

frysaren ökar, då luften 'hinner' ta upp/avge mer värme i produkt-stack/kylbatteri. Lufthastigheten bestäms av frysarens resistans, vilken är en funktion av graden av igenfrostning av kylbatteriet, och av bandtäckningen, hur stor yta av bandet som är täckt av produkt.

*) Bandtäckningen: Denna påverkar, förutom via lufthastigheten, även direkt den totala kyleffekten. För mycket produkt på bandet ger en uppvärmning av luften, som ju ska transportera bort värmets. Produkten blir därför otillräckligt fryst, fastän frysarens totala kyleffekt ökat.

*) Köldmediets temperatur: Temperaturen på den luft som passerat kylbatteriet, bestäms i hög grad av köldmediets temperatur, den s k förångningstemperaturen (ca -40°C). Eftersom ofta flera frysare är kopplade till samma maskinrum, kan en belastningsändring hos en av frysarna påverka förångningstemperaturen, och därmed övriga frysare i systemet.

*) Produktparametrar: T ex vikt och ingångstemperatur hos produkten kan variera, orsakat av störningar på de enheter som ligger före frysaren i produktionslinjen.

*) Lufttemperaturen: De störningar som nämnts, påverkar lufttemperaturen, vilken ju är avgörande för produktens infrysning. Även direkta störningar på lufttemperaturen kan uppkomma, t ex om en dörr till frysaren öppnas.

1.2.3 Reglering

De olika störningar som har nämnts, medför att produktens utgångstemperatur och därmed kvalitet fluktuerar. Önskar man trots detta hålla temperaturen strikt under en viss nivå (t ex -18°C) kommer en stor del av produkterna att få en otillfredsställande låg temperatur och bandet kan inte köras med optimal hastighet. Genom att motarbeta orsakerna till fluktuationer och parameterdrift, t ex med tryckluft kontinuerligt frosta av kylbatteriet, kan man minska de nödvändiga marginalerna gentemot gränstemperaturen. För ytterligare förbättring kan diskuteras någon form av reglering, varvid någon av följande styrvariabler är tänkbara:

*) Lufthastigheten: Om kontinuerligt reglerbara fläktar installeras, förefaller detta vara den naturligaste styrvariabeln. En enklare variant är att ha ett

par stycken fläktar, som kan slås till/från oberoende av varandra. En mer förfinad metod är att placera fläktarna på olika platser i frysaren.

*) Förångningstemperaturen: Varje frysare kan förses med eget maskinrum. Förångningstemperaturen, styrvariabeln, kan då i sin tur regleras med hjälp av en regulator, kopplad till maskinrummet, s k kaskadreglering.

*) Bandhastigheten: Reglering genom variation av hålltiden är relativt okomplicerad att realisera, ur den synpunkten att ingen om/tillbyggnad av frysutrustningen behövs. Om produkten är för varm, minskas bandhastigheten, och vice versa. En förutsättning för regleringen är dock, att omgivande delar av produktionslinjen kan följa med i ändringarna av produktionstakten, så att inte bandtäckningen ökar med minskad bandhastighet, eller vissa 'flaskhalsar' översvämmas av produkt.

*) Bandtäckningen: I stället för att styra med bandhastigheten, kan vi hålla denna konstant, och låta bandtäckningen variera. Således styr vi inte själva frysaren, utan produktflödet från de enheter, som ligger före i produktionskedjan. En nackdel med denna styrning är att det tar tid att ändra bandtäckningen inne i frysaren.

1.2.4 Mätning

Om reglering av frysaren ska införas, måste mätvärden kontinuerligt, eller med regelbundna intervall, kunna tillföras regulatorn. Mätning av lufttemperaturer och förångningstemperatur möter inga större hinder: givarnas placering trimmas in, och någon form av filtrering av signalerna till regulatorn får införas.

Produkttemperaturens mätning är en mer komplicerad procedur. De kontrollmätningar som för närvarande utförs på frysare i drift görs off-line, genom stickprovstagning, en stunds förvaring av provet i 'termos', och mätning efter att temperaturen jämnats ut. En framtida mätning on-line skulle kunna implementeras med hjälp av IR-kamera. Temperaturen hos inkommande produkt kan i vissa fall mätas i samband med produktens framställning tidigare i produktionskedjan.

Bandlasten skulle kunna beräknas genom kontinuerlig bandvägning vid ingången till frysaren, eller genom mätning av föregående enhets produktions-takt.

2 VÄRMELEDNINGSEKVATIONEN

Ett dynamiskt system kan i allmänhet beskrivas med ett system av differentialekvationer. Dessa ekvationer kan t ex vara uppställda så att tidsderivatorna för var och en av systemets variabler uttrycks som funktioner av alla variablerna. Med hjälp av något simuleringsprogram eller -språk, t ex SIMNON, kan man studera egenskaperna hos det på standardiserad form skrivna ekvationssystemet.

Temperaturens dynamik i produkt under infrysning, beskrivs med värmeledningsekvationen (hädanefter förkortat: VLE). VLE är en partiell differentialekvation, dvs den innehåller derivator med avseende på flera variabler (tiden och lägeskoordinater). Därför är det lämpligt om vi först provar fram en fungerande lösningsmetod för VLE. Algoritmen för lösning av ekvationen för produkten kan sedan kombineras med ekvationer för bl a luftströmning, värmeövergång i kylbatteri osv, dvs den miljö som omger och påverkar produkten i frysaren. Detta leder då till den fullständiga frysarmodell som beskrivs och studeras i kap 3.

2.1 Numerisk lösning av VLE

En algoritm framtas för lösning av värmeledningsekvationen med, på grund av fasövergång, temperaturberoende koefficienter.

2.1.1 Differensapproximation

Vid den numeriska lösningen av VLE ersätts differentialekvationen med ett ekvationssystem för temperaturerna i ett ändligt antal punkter i produkten och ett uppräknligt antal tidpunkter. Ekvationssystemet löses rekursivt med utgångspunkt från kända begynnelsevärden. På så vis simuleras tidsförloppet -- man kan gå in vid valfri tidpunkt och läsa ut värden och/eller ändra randvärden.

Den numeriska metoden grundar sig på att derivatorna i differentialekvationen ersätts med differensapproximationer. Låt oss behandla VLE i det endimensionella fallet. Detta motsvarar produkt av typ t ex hamburgare eller pannkakor, vilka med avseende på värmeledningen approximativt kan betraktas som skivor av bestämd tjocklek och oändlig utsträckning (värmeövergång genom kanterna försummas).

VLE ska lösas i intervallet

$$0 \leq x \leq D$$

(D: tjocklek)

med givna randvärden i $x = 0$ och $x = D$. Anta symmetri kring $x = D/2$, och gör fortsättningsvis beräkningarna för intervallet

$$0 \leq x \leq D/2$$

Detta intervall delas i n st delar med längd

$$\Delta x = (D/2)/n$$

och fortsättningsvis sätts temperatur (och därav beroende variabler) som funktion av den diskreta variabeln j :

$$x = j\Delta x$$

Således har vi vid en viss tidpunkt temperaturerna

$$T[j], \quad j = 0, 1, \dots, n$$

där $T[0]$ är yttemperatur och $T[n]$ mittemperatur.

Tiden delas in i tidssteg Δt . I stället för att beteckna tiden med en variabel, skriver vi ekvationerna på rekursiv form, och låter T beteckna temperatur vid aktuell tidpunkt, nT ('new T ') temperatur vid påföljande tidpunkt, och oT ('old T ') temperatur vid föregående tidpunkt. Motsvarande beteckningar införs för andra tidsberoende variabler.

Nu kan vi övergå till differensapproximation av

$$\text{VLE: } \rho c_p \frac{d}{dt} T = \lambda \frac{d^2}{dx^2} T$$

Tidsderivatan ersätts med en enkel differens:

$$\frac{d}{dt} T[j] \approx \frac{1}{\Delta t} (nT[j] - T[j])$$

Lägesderivatan kan approximeras på samma sätt. För att minska felet i approximationen gör vi den dock symmetrisk:

$$\frac{d}{dx} T[j] \approx \frac{1}{2\Delta x} (T[j+1] - T[j-1])$$

Vi behöver en andraderivata och gör således en 'dubbel' differens:

$$\begin{aligned} \frac{d^2}{dx^2} T[j] &\approx \frac{1}{2\Delta x} \left[\frac{d}{dx} T[j+1] - \frac{d}{dx} T[j-1] \right] \approx \\ &\approx \frac{1}{2\Delta x} \left[\frac{1}{2\Delta x} (T[j+2] - T[j]) \right. \\ &\quad \left. - \frac{1}{2\Delta x} (T[j] - T[j-2]) \right] = \\ &= \frac{1}{4\Delta x^2} (T[j+2] - 2T[j] + T[j-2]) \end{aligned}$$

Denna ekvation har nackdelen att den behandlar punkter på två stegs avstånd från varandra, vilket ger problem vid randen $j = 0$ ($x = 0$). Skriv om ekvationen till 'enstegsform' (byt $2\Delta x \rightarrow \Delta x$):

$$\frac{d^2}{dx^2} T[j] \approx \frac{1}{\Delta x^2} (T[j+1] - 2T[j] + T[j-1])$$

Differensapproximationen av VLE blir då

$$\begin{aligned} \rho c_p \frac{1}{\Delta t} (nT[j+1] - T[j]) &\approx \\ &\approx \lambda \frac{1}{\Delta x^2} (T[j+1] - 2T[j] + T[j-1]) \end{aligned}$$

2.1.2 Fasövergång

Om lösningen innehåller temperaturändringar som innebär fasövergång, måste värmekapacitivet och värmeledningstal sättas som funktioner av T :

$$\rho c_p(T) \frac{d}{dt} T = \frac{d}{dx} \left[\lambda(T) \frac{d}{dx} T \right]$$

Högerledet blir då:

$$\begin{aligned} & \frac{d}{dx} \left[\lambda[j] \frac{d}{dx} T[j] \right] \approx \\ & \approx \frac{1}{2\Delta x} \left[\lambda[j+1] \frac{d}{dx} T[j+1] - \lambda[j-1] \frac{d}{dx} T[j-1] \right] \approx \\ & \approx \frac{1}{2\Delta x} \left[\lambda[j+1] \frac{1}{2\Delta x} (T[j+2] - T[j]) \right. \\ & \quad \left. - \lambda[j-1] \frac{1}{2\Delta x} (T[j] - T[j-2]) \right] \end{aligned}$$

där t ex $\lambda[j]$ definieras

$$\lambda[j] \equiv \lambda(T[j])$$

Tvåstegsformen överförs till ekvivalent enstegsform:

$$\begin{aligned} & \frac{d}{dx} \left[\lambda[j] \frac{d}{dx} T[j] \right] \approx \\ & \frac{1}{\Delta x} \left[\lambda[j + \frac{1}{2}] \frac{1}{\Delta x} (T[j+1] - T[j]) \right. \\ & \quad \left. - \lambda[j - \frac{1}{2}] \frac{1}{\Delta x} (T[j] - T[j-1]) \right] \approx \\ & \left[\lambda[j \pm \frac{1}{2}] \approx \frac{1}{2} (\lambda[j] + \lambda[j \pm 1]) \right] \\ & \approx \frac{1}{\Delta x} \left[\frac{1}{2} (\lambda[j+1] + \lambda[j]) \frac{1}{\Delta x} (T[j+1] - T[j]) \right. \\ & \quad \left. - \frac{1}{2} (\lambda[j] + \lambda[j-1]) \frac{1}{\Delta x} (T[j] - T[j-1]) \right] \end{aligned}$$

Ett alternativt sätt är

$$\begin{aligned} & \frac{d}{dx} \left[\lambda[j] \frac{d}{dx} T[j] \right] = \\ & = \left[\frac{d}{dx} \lambda[j] \right] \left[\frac{d}{dx} T[j] \right] + \lambda[j] \frac{d^2}{dx^2} T[j] \approx \\ & \approx \frac{1}{2\Delta x} (\lambda[j+1] - \lambda[j-1]) \frac{1}{2\Delta x} (T[j+1] - T[j-1]) \\ & + \lambda[j] \frac{1}{\Delta^2 x} (T[j+1] - 2T[j] + T[j-1]) = \\ & = \frac{1}{4\Delta^2 x} \left[(\lambda[j+1] - \lambda[j-1] + 4\lambda[j]) T[j+1] - 8\lambda[j] T[j] \right. \\ & \quad \left. + (\lambda[j-1] - \lambda[j+1] + 4\lambda[j]) T[j-1] \right] \end{aligned}$$

2.1.3 Randvillkor

VLE gäller i intervallet $0 \leq x \leq D/2$. Vid randen, $x = 0$ resp $x = D/2$, måste speciella villkor införas.

Värmeövergången vid ytan, $x = 0$, ger randvillkoret

$$\lambda[0] \frac{d}{dx} T[0] = \alpha (T[0] - T_{\text{air}})$$

$$\left(\begin{array}{l} \alpha: \text{värmeövergångstal} \\ T_{\text{air}}: \text{lufttemperatur} \end{array} \right)$$

där vänsterledet står för värmeströmningen vid $x = 0$, och högerledet är värmeövergången till luften.

Derivatans approximeras t ex

$$\frac{d}{dx} T[0] \approx \frac{1}{\Delta x} (T[1] - T[0])$$

Felet i approximationen är av storleksordningen Δx . Asymmetrin medför att entalpi-balansen i ekvationen rubbas, så att felet ackumuleras under simuleringen. För att få resultatet mindre beroende av de lokala felet i varje tidssteg, ansätter vi en symmetrisk approximation:

$$\frac{d}{dx} T[0] \approx \frac{1}{2\Delta x} (T[1] - T[-1])$$

Approximationsfelet är här av storleksordningen Δx^2 . $T[-1]$ är produkttemperaturen i en punkt utanför ytan, och motsvarar således ingen verklig temperatur. Dock kan vi i våra beräkningar låta $T(x)$ vara definierad och satisfiera VLE även för $x < 0$. I och med införandet av $T[-1]$ i beräkningarna, måste en extra ekvation tillföras ekvationssystemet. Detta görs genom att låta den differensapproximerade VLE gälla för $j = 0$.

Ett alternativt sätt att införa randvillkor vid ytan är att ställa upp en entalpi-balans-ekvation

$$\alpha (T[0] - T_{\text{air}}) = - \int_0^{D/2} c_p (T(x)) \frac{dT(x)}{dt} dx$$

där vänsterledet är det vid ytan bortförda värmets, och högerledet är lika med entalpi-minskningen (Dimension: effekt/ytenhet). Tidsderivatan av T

erhålls ur VLE och integrationen görs med någon numerisk metod (t ex Simpsons formel ger ett fel av storleksordning $\Delta^2 x$). Detta sätt fungerar bra, men kan ge något mer komplicerade (och tidsödande) beräkningar än för den konventionella metoden).

Randvillkor för $x = D/2$ fås antingen genom att sätta $dT/dx = 0$ i den ursprungliga VLE och sedan differensapproximera denna, eller genom att applicera symmetrikravet på den redan genomförda approximationen.

Den senare varianten innebär att vi sätter

$$\begin{aligned}\lambda[n+1] &= \lambda[n-1] \\ T[n+1] &= T[n-1]\end{aligned}$$

Ekvationen för $j = n$ blir då

$$\begin{aligned}e c_p [n] \frac{1}{\Delta t} (nT[n] - T[n]) &\approx \\ \approx \frac{1}{2\Delta^2 x} ((\lambda[n+1] + \lambda[n]) (T[n+1] - T[n]) & \\ - (\lambda[n] + \lambda[n-1]) (T[n] - T[n-1])) &= \\ = \frac{1}{2\Delta^2 x} 2(\lambda[n-1] + \lambda[n]) (T[n-1] - T[n]) &\end{aligned}$$

2.1.4 Algoritm

För att erhålla algoritmen för lösning av VLE, skriver vi om ekvationerna, så att nya temperaturvärden kan beräknas ur värdena från föregående tidssteg:

$$\begin{aligned}e c_p \frac{1}{\Delta t} (nT[j] - T[j]) &= \\ = \lambda \frac{1}{\Delta^2 x} (T[j+1] - 2T[j] + T[j-1]) & \\ \Leftrightarrow & \\ nT[j] = T[j] + \frac{\beta}{c_p} \lambda (T[j+1] - 2T[j] + T[j-1]) , &\end{aligned}$$

($j = 1, \dots, n-1$)

$$\left[\beta = \frac{\Delta t}{e \Delta^2 x} \right]$$

Randvillkoret i mitten av produkten blir

$$e c_p \frac{1}{\Delta t} (nT[n] - T[n]) = \lambda \frac{1}{\Delta^2 x} 2(T[n-1] - T[n])$$

\Leftrightarrow

$$nT[n] = T[n] + \frac{\beta}{c_p} 2\lambda (T[n-1] - T[n])$$

och vid ytan:

$$\lambda \frac{1}{2\Delta x} (T[1] - T[-1]) = \alpha (T[0] - T_{\text{air}})$$

$$\Leftrightarrow T[-1] = T[1] - \frac{\gamma}{\lambda} (T[0] - T_{\text{air}})$$

$$\Leftrightarrow nT[-1] = nT[1] - \frac{\gamma}{\lambda} (nT[0] - nT_{\text{air}})$$

$$(\gamma = 2\Delta x \cdot \alpha)$$

och så antar vi att VLE gäller även för $j=0$.

Således blir algoritmen:

$$\left\{ \begin{array}{l} nT[j] = T[j] + \frac{\beta}{c_p} \lambda (T[j+1] - 2T[j] + T[j-1]) , \\ (j = 0, \dots, n-1) \\ nT[n] = T[n] + \frac{\beta}{c_p} 2\lambda (T[n-1] - T[n]) \\ nT[-1] = nT[1] - \frac{\gamma}{\lambda} (nT[0] - nT_{\text{air}}) \end{array} \right.$$

$$\left[\begin{array}{l} \beta = \frac{\Delta t}{e\Delta^2 x} \\ \gamma = 2\Delta x \cdot \alpha \end{array} \right]$$

Obs: att vi i sista ekvationen utnyttjar 'nya' värden av $T[1]$ och $T[0]$.

På samma sätt för temperaturberoende c_p och λ :

$$\left\{ \begin{array}{l} nT[j] = T[j] + \frac{\beta}{c_p[j]} \frac{1}{4} ((\lambda[j+1] - \lambda[j-1] + 4\lambda[j]) T[j+1] \\ - 8\lambda[j] T[j] + (\lambda[j-1] - \lambda[j+1] + 4\lambda[j]) T[j-1]) , \\ (j = 0, \dots, n-1) \\ nT[n] = T[n] + \frac{\beta}{c_p[n]} 2\lambda[n] (T[n-1] - T[n]) \\ nT[-1] = nT[1] - \frac{\gamma}{\lambda[0]} (nT[0] - T_{\text{air}}) \end{array} \right.$$

$$\left[\begin{array}{l} \beta = \frac{\Delta t}{e\Delta^2 x} \\ \gamma = 2\Delta x \cdot \alpha \end{array} \right]$$

eller

$$\left[\begin{array}{l} nT[j] = T[j] + \frac{\beta}{c_p[j]} \frac{1}{2} ((\lambda[j+1] + \lambda[j]) (T[j+1] - T[j])) \\ - (\lambda[j] + \lambda[j-1]) (T[j] - T[j-1])) \\ osv... \end{array} \right.$$

2.2 Val av koefficienter och parametrar

Temperaturberoendet för koefficienterna i VLE bestäms ur empiriska data. Val av intervallängder i de gjorda differensapproximationerna diskuteras.

2.2.1 Bestämning av $c_p(T)$ och $\lambda(T)$

Från Frigoscandia har erhållits empiriska värden på entalpi och värmeövergångstal vid olika temperaturer, se tabell i figur 2.1. Entalpin sätts definitionsmässigt till 0 vid -20° .

$\lambda(T)$ ansätter vi som en linjär funktion i infrysningsintervallet $-10 < T < 0$, se fig 2.2.

Värmekapacitivitet som funktion av temperaturen ska bestämmas ur entalpivärdena. Först anpassar vi en kurva till de experimentellt framtagna punkterna i entalpi/temperatur-diagrammet, sedan utnyttjar vi att

$$c_p(T) = \frac{d}{dT} h(T)$$

Anpassning av kurva:

I) Interpolation: detta är den enklaste metoden att ansätta en funktion $H(T)$; om vi interpolerar med rätta linjer mellan punkterna blir derivatan $c_p(T)$ en styckvis konstant funktion enligt fig 2.3.

II) Ansättande av exponentialfunktion: vi ansätter en funktion i infrysningsintervallet $-10 < T < 0$, sådan att $H(T)$ blir kontinuerlig i $T = 0$ och

T/°C	h/(kJ/kg)	λ /(W/mK)
0	261	0,50
-1	205	0,70
-3	88	1,10
-5	63	1,45
-10	33	1,45
-20	0	1,45

Fig 2.1. Entalpi (H) och värmeövergångstal (λ) vid olika temperaturer.

$T = -10$. Låt

$$c_p(T) = \begin{cases} A_0 & T \leq -10 \\ A_0 e^{\beta(T+10)} & -10 < T < 0 \\ A_1 & T \geq 0 \end{cases}$$

$c_p(T)$ blir kontinuerlig i $T = -10$. β anpassas så att

$$\int_{-10}^0 c_p(T) dT = H(0) - H(-10) \quad *)$$

med entalpivärden ur tabellen. Kurva enligt figur 2.3.

Med tanke på algoritmens stabilitet väljer vi den kurva, som har minst diskontinuiteter, dvs exponentialkurvan.

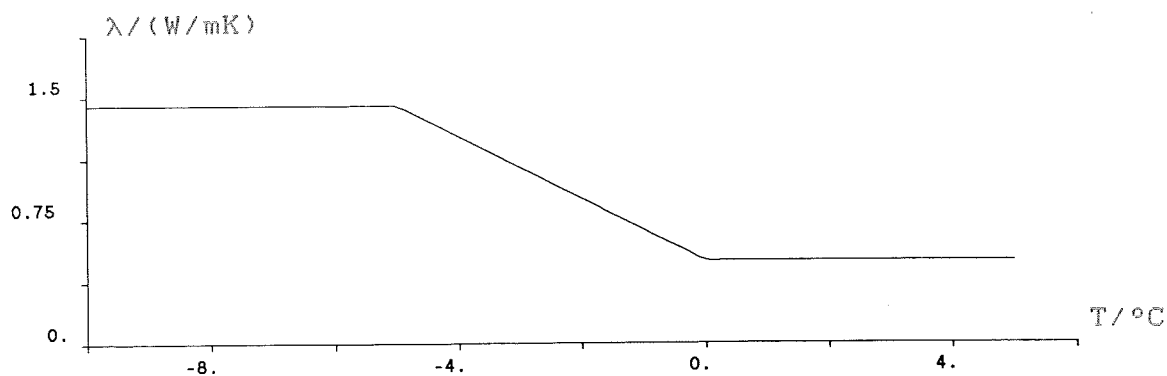


Fig 2.2. Värmeövergångstal (λ) ansatt som funktion av temperaturen.

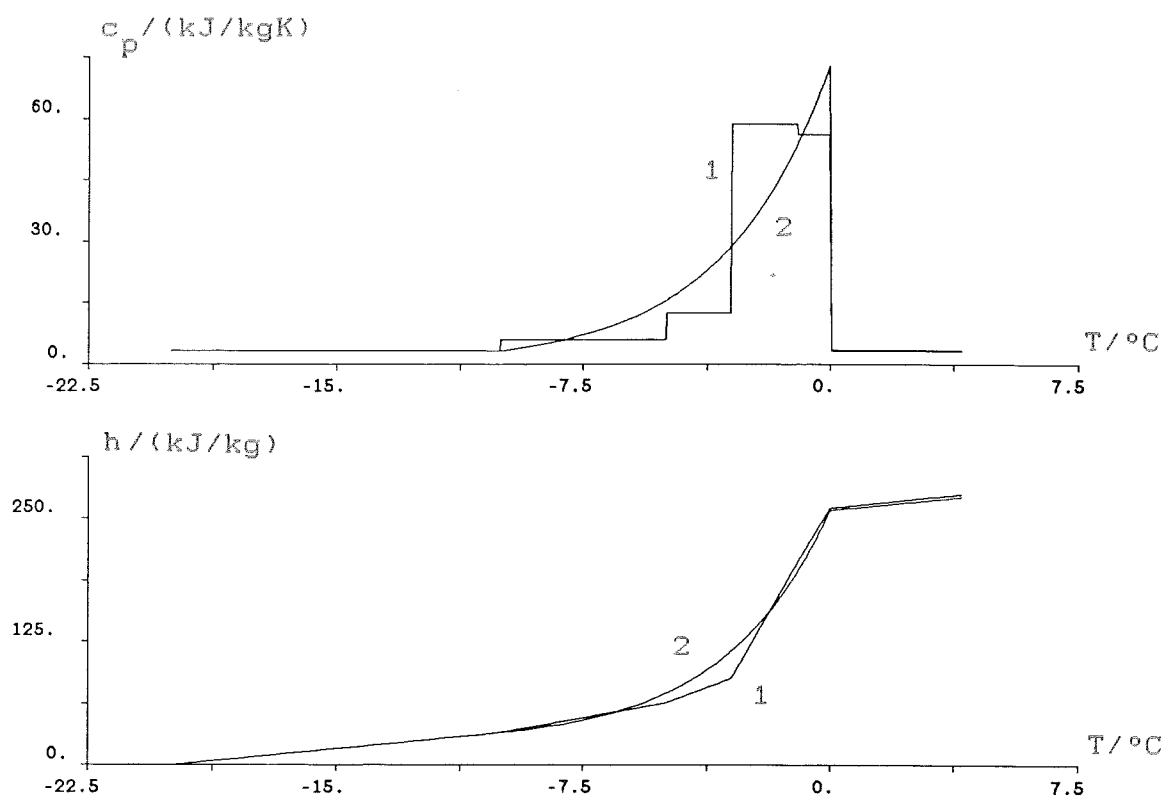


Fig 2.3. Värmekapacitivet (c_p) (övre diagrammet) respektive entalpi (h) (undre diagrammet) som funktion av temperaturen.

Två ansatser: linjär interpolation (1) respektive exponentialkurva (2).

2.2.2 Val av Δx och Δt

Vilka intervallängder Δx och Δt vi ska välja är ett inte helt trivialt problem. Hänsyn måste tas till några motsägelsefulla krav:

*) Stabilitet

Algoritmen är garanterat stabil för

$$\frac{\Delta t}{\Delta x^2} \leq \frac{e c_p}{\lambda(2 + \alpha \Delta x)}$$

(se: 'Numerical Solution of Partial Differential Equations').

I det temperaturberoende fallet kan vi sätta

$$c_p = c_{p, \min}$$

$$\lambda = \lambda_{\max}$$

$$\left[\begin{array}{l} c_{p, \min} : \text{det minsta värde, } c_p \text{ antar} \\ \lambda_{\max} : \text{det största värde, } \lambda \text{ antar} \end{array} \right]$$

Instabilitet illustreras i figur 2.4. Stabilitetskravet medför att vi eftersträvar litet Δt , stort Δx .

*) Noggrannhet

Felet i lösningen beror till stor del av felet i approximationen av VLE. Kravet på litet fel i differensapproximationen innebär att ett litet Δt och ett litet Δx bör eftersträvas.

*) Snabbhet

Beräkningstiden är omvänt proportionell mot tidssteget och proportionell mot antalet ekvationer. Det senare är i sin tur approximativt omvänt proportionellt mot Δx . Således

$$\text{tid prop mot } \frac{1}{\Delta x \Delta t}$$

Ett stort Δt och ett stort Δx ger en snabb algoritm.

Till sist kan nämnas att alltför små Δt och Δx ger problem med avrundningsfel, p g a förlust av signifikanta siffror i differensekvationerna (subtraktion av tal med flera gemensamma siffror).

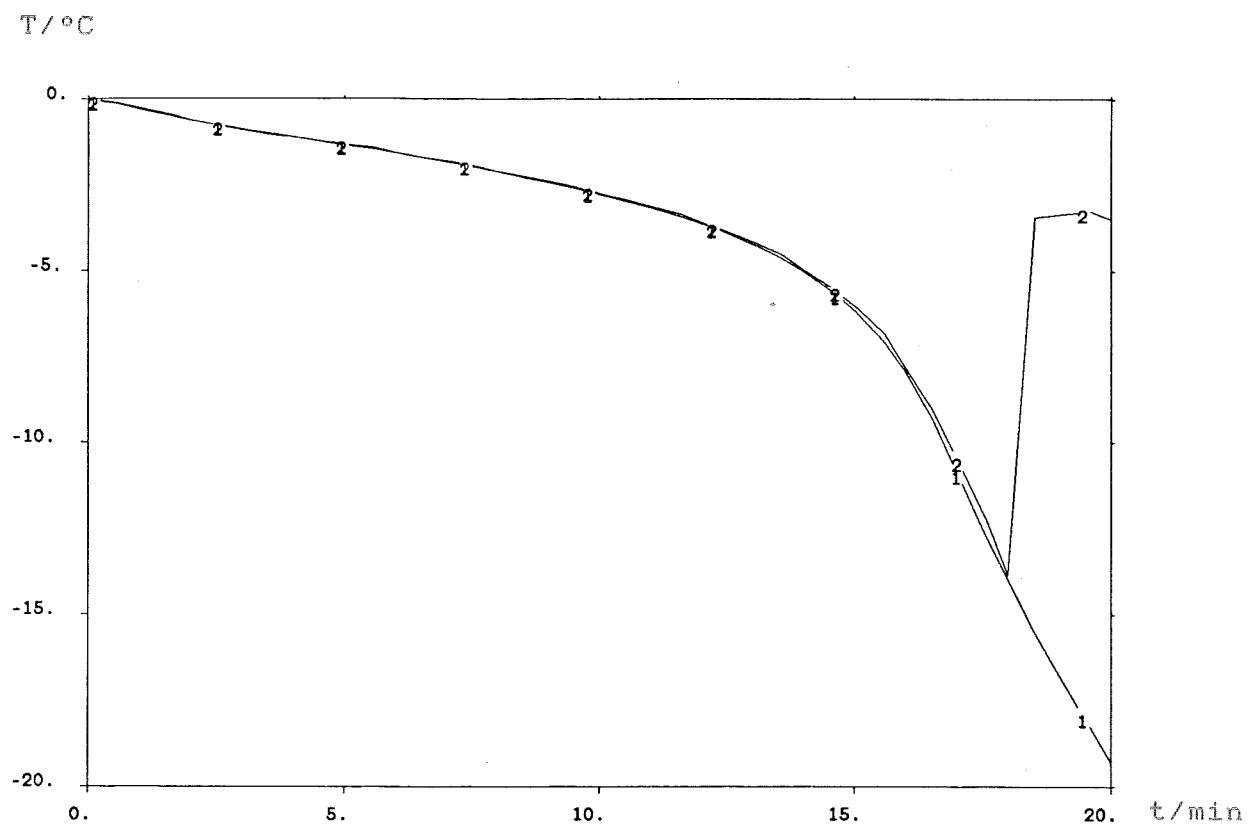


Fig 2.4. Infrysningsskurva: stabil algoritm (1), instabil algoritm (2).

2.3 Alternativa metoder

Metoden med temperaturberoende koefficienter kan generaliseras. Principerna vid byte av koordinatsystem (för studium av sfäriska/cylindriska produkter), respektive vid utnyttjande av effektivare algoritmer, visas.

2.3.1 Byte av koordinatsystem

Vi behandlar här VLE endast i det endimensionella fallet. Den förutsättning vi utgått från är att produkten är 'platt', dvs värmeövergången till luften sker över två varandra motstående sidor av produkten. I ett cartesiskt koordinatsystem beskrivs dessa sidor som de punkter (x,y,z) , sådana att $x = 0$ resp $x = D/2$.

Meddelst byte av koordinatsystem, från cartesiskt till sfäriskt eller cylindriskt, kan vi på motsvarande sätt enkelt uttrycka vissa randytor. Kombinerat med symmetriförutsättningar, medför koordinatbytet att vi kan behandla VLE endimensionellt för sfäriska resp cylindriska produkter.

Beräkningsprincipen visas här för den ena av dessa produkttyper: den sfäriska (motsvarande t ex köttbullar eller ärtor).

En övergång till sfäriska koordinater medger att randytan kan beskrivas som de punkter (r,φ,θ) , sådana att

$$r = D/2$$

$$\left\{ \begin{array}{l} r: \text{avstånd från produktens mittpunkt} \\ D: \text{produktens diameter} \end{array} \right\}$$

Om det, på samma sätt som i tidigare beräkningar, förutsätts att värmeövergången är homogen över produktytan, kan vi tack vare symmetrin skriva VLE som

$$\epsilon c_p \frac{d}{dt} T = \frac{1}{r^2} \frac{d}{dr} \left(r^2 \lambda \frac{d}{dr} T \right) = \frac{d}{dr} \left(\lambda \frac{d}{dr} T \right) + \frac{2}{r} \lambda \frac{d}{dr} T$$

Vi sätter

$$r = (n-j) \Delta r$$

så att, såsom tidigare, $T[0]$ kommer att beteckna yttemperaturen, och $T[n]$ mittemperaturen. Högerledet i VLE approximeras:

$$\frac{d}{dr} \left(\lambda[j] \frac{d}{dr} T[j] \right) + \frac{2}{r} \lambda[j] \frac{d}{dr} T[j] \approx$$

$$\begin{aligned} &\approx \frac{1}{\Delta r} \left[\frac{1}{2} (\lambda[j+1] + \lambda[j]) \frac{1}{\Delta r} (T[j+1] - T[j]) \right. \\ &- \left. \frac{1}{2} (\lambda[j] + \lambda[j-1]) \frac{1}{\Delta r} (T[j] - T[j-1]) \right] \\ &+ \frac{2}{(n-j)\Delta x} \lambda[j] \frac{1}{2\Delta x} (T[j-1] - T[j+1]), \quad j \neq n \end{aligned}$$

(obs: tecknet i differensapproximationen av dT/dr)

För att erhålla ekvationen för $j = n$ måste vi göra en gränsövergång. I mittpunkten ($j = n$) gäller

$$\frac{d}{dr} T[n] = 0$$

så

$$\begin{aligned} \frac{2}{r} \lambda \frac{d}{dr} T &= \frac{2}{r} \lambda \left[\frac{d}{dr} T[n] + r \frac{d^2}{dr^2} T[n] + o_1(r^2) \right] = \\ &= 2\lambda \left[\frac{d^2}{dr^2} T[n] + o(r) \right] \rightarrow 2\lambda[n] \frac{d^2}{dr^2} T[n], \quad r \rightarrow 0 \end{aligned}$$

Hela högerledet blir

$$\begin{aligned} &\frac{d}{dr} \left[\lambda \frac{d}{dr} T \right] + \frac{2}{r} \lambda \frac{d}{dr} T = \\ &= \frac{d}{dr} \lambda \frac{d}{dr} T + \lambda \frac{d^2}{dr^2} T + \frac{2}{r} \lambda \frac{d}{dr} T \rightarrow \\ &\rightarrow 0 + \lambda[n] \frac{d^2}{dr^2} T[n] + 2\lambda[n] \frac{d^2}{dr^2} T[n] = \\ &= 3\lambda[n] \frac{d^2}{dr^2} T[n], \quad r \rightarrow 0 \end{aligned}$$

vilket approximeras

$$\begin{aligned} 3\lambda[n] \frac{d^2}{dr^2} T[n] &\approx 3\lambda[n] \frac{1}{\Delta^2 x} (T[n-1] - 2T[n] + T[n+1]) = \\ &(\text{symmetri: } T[n+1] = T[n-1]) \\ &= \frac{6\lambda[n]}{\Delta^2 x} (T[n-1] - T[n]) \end{aligned}$$

Randvärdet för $r = D/2$ (ytan, $j = 0$) blir:

$$\frac{\lambda[0]}{2\Delta r} (T[1] - T[-1]) = \alpha (T[0] - T_{\text{air}})$$

2.3.2 Crank-Nicolson

Den algoritm som framtagits, medför att valet av intervallängder Δx och Δt inte kan göras efter något entydigt kriterium. Exekveringstiden blir relativt lång p g a stabilitetsproblem vid långa Δt .

Ett sätt att minska felet i approximationen av tidsderivatan (för att möjliggöra längre Δt) är att införa en symmetrisk differens:

$$\frac{d}{dt} T \approx \frac{1}{\Delta t} (nT - oT)$$

Här måste T -värden från ett tidigare tidssteg (oT) sparats och användas i algoritmen. Tyvärr blir denna instabil.

En metod som alltid medför stabilitet (kan påvisas med matris-metoder; se litteraturen) är den s k Crank-Nicolsonmetoden, vilken innebär att VLE approximeras

$$\begin{aligned} \rho c_p \frac{1}{\Delta x} (nT[j] - T[j]) &= \\ &= \lambda \frac{1}{\Delta^2 x} \left[\frac{1}{2} (T[j+1] - 2T[j] + T[j-1]) \right. \\ &\quad \left. + \frac{1}{2} (nT[j+1] - 2nT[j] + nT[j-1]) \right] \end{aligned}$$

Principen är, att differensekvationen görs symmetrisk i förhållande till tidpunkten mitt emellan aktuell tidpunkt och nästkommande tidpunkt (d v s tidpunkterna för T resp nT). Varje ekvation innehåller tre stycken $nT[.]$, vilket innebär att vi måste lösa hela ekvationssystemet för att få ut $nT[j]$, $j = -1, 0, 1, \dots, n$. En sådan metod kallas indirekt, till skillnad från den som tidigare beskrivits (där $nT[j]$ erhöles direkt ur ekvation nr j).

Randvärden erhålls på samma sätt som tidigare.

Algoritmen erhåller vi efter att ha skrivit om ekvationen:

$$\begin{aligned}
& \frac{2c}{\beta} p (nT[j] - T[j]) = \\
& = \lambda (T[j+1] - 2T[j] + T[j-1] + nT[j+1] - 2nT[j] + nT[j-1]) \\
& \quad \Leftrightarrow \\
& - \lambda nT[j-1] + \left[\frac{2c}{\beta} p + 2\lambda \right] nT[j] - \lambda nT[j+1] = \\
& = \lambda T[j-1] + \left[\frac{2c}{\beta} p + 2\lambda \right] T[j] + \lambda T[j+1] \quad , \quad j = 0, 1, \dots, n
\end{aligned}$$

och får på så vis alla nya T-värden i vänstra ledet. Skriv randvärdet för $j = 0$ på formen

$$\begin{aligned}
T[-1] &= T[1] - \frac{\gamma}{\lambda} (T[0] - T_{\text{air}}) \\
&\quad \Leftrightarrow \\
nT[-1] &= nT[1] - \frac{\gamma}{\lambda} (nT[0] - nT_{\text{air}})
\end{aligned}$$

Sätt in i ekvationen för $j = 0$:

$$\begin{aligned}
& -\lambda \left(nT[1] - \frac{\gamma}{\lambda} (nT[0] - nT_{\text{air}}) \right) \\
& + \left[\frac{2c}{\beta} p + 2\lambda \right] nT[0] - \lambda nT[1] = \\
& = \left[\gamma + \frac{2c}{\lambda} p + 2\lambda \right] nT[0] - 2\lambda nT[1] - \gamma nT_{\text{air}} = \\
& = \lambda T[j-1] + \left[\frac{2c}{\beta} p + 2\lambda \right] T[0] + \lambda T[1]
\end{aligned}$$

Ekvationssystemet kan då skrivas

$$\left\{ \begin{aligned}
& \left[\gamma + \frac{2c}{\lambda} p + 2\lambda \right] nT[0] - 2\lambda nT[1] = \\
& = \lambda T[j-1] + \left[\frac{2c}{\beta} p + 2\lambda \right] T[0] + \lambda T[1] + \gamma nT_{\text{air}} ; \\
& - \lambda nT[j-1] + \left[\frac{2c}{\beta} p + 2\lambda \right] nT[j] - \lambda nT[j+1] = \\
& = \lambda T[j-1] + \left[\frac{2c}{\beta} p + 2\lambda \right] T[j] + \lambda T[j+1] \quad , \quad j = 1, \dots, n-1 ; \\
& - \lambda nT[n-1] + \left[\frac{c}{\beta} p + \lambda \right] nT[n] = \lambda T[n-1] + \left[\frac{c}{\beta} p - \lambda \right] T[n] ; \\
& nT[-1] = nT[1] - \frac{\gamma}{\lambda} (nT[0] - nT_{\text{air}})
\end{aligned} \right.$$

Undantaget den sista ekvationen utgör detta ett tridiagonalt ekvationssystem, som måste lösas för varje tidssteg i simuleringen. Ekvationssystemet och dess härledning är nu skrivet på en form, som gör det enkelt att införa temperaturberoende koefficienter. λ och c_p substitueras mot $\lambda[.]$ resp $c_p[.]$, enligt tidigare mönster.

3 MODELL AV FRYSARE

I föregående kapitel tog vi fram en algoritm för att kunna studera infrysningsförloppet för själva produkten, med givna värden på lufttemperatur och värmeövergångstal insatta i ekvationerna. Vi ska nu ta fram en enkel modell av infrysningsprocessen som helhet:

Produkten, med viss ingångstemperatur, matas in i nederdelen av frysaren, och transporteras uppåt i produktstacken. Här antar vi att produkten utsätts för en lufttemperatur som varierar med tiden och längs stacken, samt ett värmeövergångstal som varierar med tiden. Denna miljö påverkas av ett antal faktorer, t ex luftflödet, kylbatteriets kyleffekt och diverse 'oönskade' värmekällor. Maskinvarudelen av den matematiska modellen utgörs av de ekvationer vi väljer för att bestämma frysarmiljön.

Till sist överför vi ekvationerna till diskret form, och sammanför algoritmerna för produkt och frysare i ett Pascalprogram.

3.1 Luften i frysaren

De samband, som bestämmer värmeövergångstal (produkt - luft) och lufttemperaturer, tas fram.

3.1.1 Luftflödet

Luften i frysaren cirkulerar från kylbatteri till stack och tillbaka till kylbatteriet. I denna schematiska beskrivning antar vi ett slutet flöde, så att

$$\text{luftflödet} = W_{\text{stack}} A_{\text{stack}} = W_{\text{bat}} A_{\text{bat}}$$

$$\left[\begin{array}{l} W_{\text{bat}}: \text{luftens hastighet genom kylbatteriet} \\ W_{\text{stack}}: \text{luftens hastighet genom produktstacken} \\ A_{\text{bat}}: \text{luftströmmens tvärsnittsarea i kylbatteriet} \\ A_{\text{stack}}: \text{luftströmmens tvärsnittsarea i produktstacken} \end{array} \right]$$

W_{stack} och W_{bat} påverkar värmeövergången i stack respektive kylbatteri samt bestämmer temperaturfördelningen i frysaren.

Luftflödet bestäms av fläktens/fläktarnas effekt (på luften) och frysarens resistans mot luftströmmen. I figur 3.1 visas fläktkaraktistik.

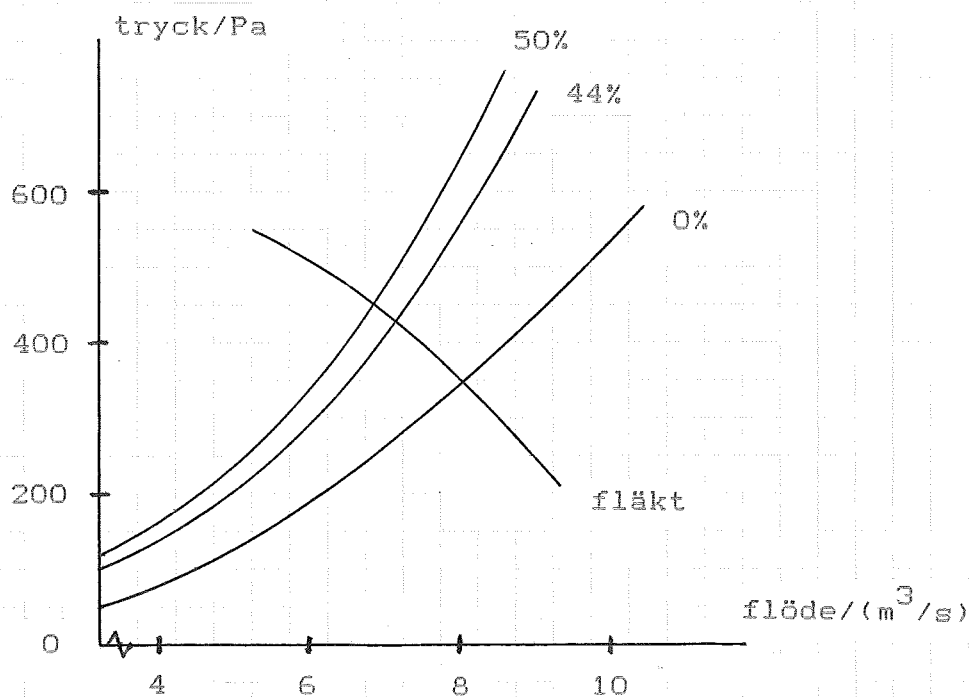


Fig 3.1. Fläktkaraktistik samt frysarresistans vid tre olika bandtäckningar.

Övertrycket Δp som skapas av fläkten ses som funktion av luftflödet (Effekten [Nm/s] = tryck [N/m²] flöde [m³/s]). Resistansen i frysaren har av Frigoscandia uppmätts vid bandtäckning 0%, 44% och 50%, och åskådliggörs genom motsvarande graf tryck/luftflöde.

Vid stationära förhållanden är det av fläktarna skapade övertrycket lika med tryckfallet i frysaren. Ur tryck/flöde-diagrammet erhålls de stationära punkterna för respektive bandtäckning (0%, 44%, 50%).

Luftflödet som funktion av bandtäckningen uppskattas genom att vi interpolerar mellan de tre mätpunkterna. Resultatet (en andragsgradskurva) visas i fig 3.2. Den extrapolerade kurvan (> 50%) verkar rimlig, emedan luftflödet gått ner till ett lågt värde vid bandtäckning 100%. Försiktighet måste emellertid iaktas vid användande av denna del av grafen, eftersom

- vi har inga mätvärden för bandtäckning > 50%
- känsligheten för fel i de uppmätta värdena är här stor
- relativa felet blir större för små värden på luftflödet
- vår fullständiga frysarmodell kan förväntas stämma dåligt för extrema värden på bandtäckningen

Då bandtäckningen inte är konstant längs med bandet, kan ett medelvärde användas i beräkningarna.

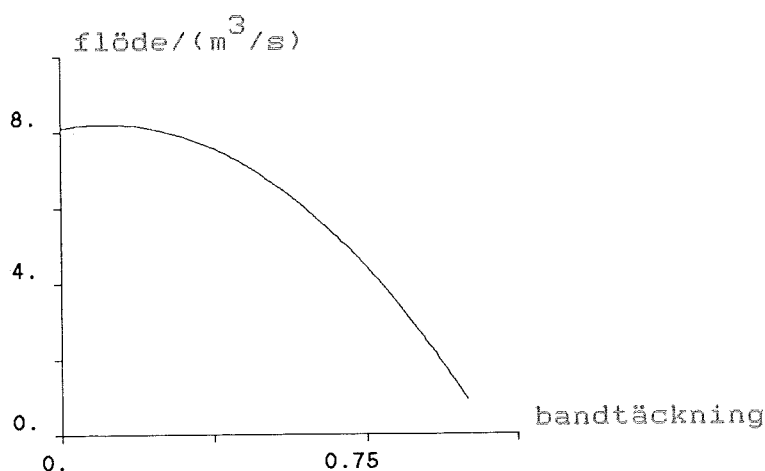


Fig 3.2. Luftflöde som funktion av bandtäckningen (grov approximation).

3.1.2 Värmeövergång produkt→luft

Värmeövergång per areaenhet produkt antas

$$\dot{q} = \alpha \{ T_{[O]} - T_{\text{air}} \}$$

(α : värmeövergångstal)

Förhållandet mellan värmeövergång och luftströmning ges av relationen

$$Nu = f(Re)$$

För en cirkulär skiva i en luftström definieras Nu, Nussels tal, som

$$Nu = \frac{\alpha d}{\lambda}$$

(d: skivans diameter
 λ : luftens värmeledningstal)

och Re, Reynolds tal, definieras:

$$Re = \frac{Wd}{\nu}$$

(W: luftens hastighet
 ν : luftens viskositet)

På Frigoscandias lab har experiment utförts med aluminiumplattor. Plattor av olika storlek har placerats i en luftström, och genom mätning av plattornas temperaturvariation har Nu kunnat bestämmas för olika värden på Re. Anpassning har gjorts till relationen

$$Nu = \text{prop mot } Re^n$$

varvid n har bestämts till

$$n \approx 0.8$$

genom att lösa ut α erhålls

$$\alpha \text{ prop mot } W^{0.8} d^{-0.2}$$

Vid infrysning sker ca 1% av värmeövergången genom fuktiförlust (ökad produktmängd ger ökad luftfuktighet och därmed ökad nedfrostning av kylbatteriet). Denna värmeövergång försummas i modellen.

3.1.3 Lufttemperaturer i stacken

För en luftström som upptar effekt från omgivningen blir entalpi-differensen

$$\Delta h = \frac{\text{effekt}}{\text{luftflöde}} \left(\frac{\text{J/s}}{\text{kg/s}} \right)$$

(h: entalpi/viktenhet)

och temperaturdifferensen

$$\Delta T_{\text{air}} = \frac{\Delta h}{c_{p, \text{air}}}$$

($c_{p, \text{air}}$: luftens värmekapacitivitet)

Luften antas strömma nedåt med homogen hastighet

$$w_{\text{stack}} \approx 3 \text{ m/s}$$

Produktens hålltid är minst ca 10 min, så

$$\begin{aligned} & (\text{produktens hastighetskomponent i vertikal led}) = \\ & = \frac{\text{stackhöjd}}{\text{hålltid}} \approx \frac{1.6}{10 \cdot 60} \approx 0.003 \text{ m/s} \ll 3 \text{ m/s} \end{aligned}$$

Produktens rörelse är således försumbar vid beräkning av lufttemperaturer. Dessutom försummar vi eventuell dynamik hos lufttemperaturerna i stacken (tidskonstant för lufttemp \ll produktens hålltid), så att vid varje tidpunkt kan lufttemperaturerna beräknas ur rådande produkttemperaturer, luftflöde, osv.

Vid luftströmmens passage av ett skikt av stacken [z , z+ Δz] ändras lufttemperaturen med

$$\Delta T_{\text{air}} \approx \frac{\dot{q}_z(z) \Delta z}{\text{airflow} \cdot c_{p, \text{air}}}$$

$$\left[\begin{array}{l} \dot{q}_z(z): \text{effekt per längdenhet} \\ \quad \quad \quad \text{i z-led} \\ \text{airflow: luftflöde} \end{array} \right]$$

där luftflödet är

$$\text{airflow} = w_{\text{stack}} A_{\text{stack}} \rho_{\text{air}}$$

Temperaturen på höjd z i stacken beräknas från temperaturen på till stacken inkommande luft:

$$T_{\text{air}}(z) = T_{\text{air}}(h) + \frac{1}{\text{airflow} \cdot c_{p, \text{air}}} \int_z^h \dot{q}_z(z') dz'$$

Integranden ges av

$$\dot{q}_z(z) = \dot{q}(z) A_z(z)$$

$$\left[\begin{array}{l} \dot{q}(z): \text{effekt/produktareaenhet} \\ A_z(z): \text{produktarea/längdenhet} \\ \quad \quad \quad \text{i z-led} \end{array} \right]$$

där effekten ges av värmeövergången:

$$\dot{q}(z) = \alpha (T(z) - T_{\text{air}}(z))$$

och produktarean av

$$A_z(z) = \text{areafactor} \cdot \text{bandcover}(z) \cdot A_{\text{belt}}/h$$

$$\left[\begin{array}{l} \text{areafactor: produktarea/täckt bandyta} \\ \text{bandcover : bandtäckning = täckt bandyta/bandyta} \\ A_{\text{belt}} \quad : \text{totala ytan av bandet i stacken} \\ h \quad \quad : \text{stackens höjd} \end{array} \right]$$

Här multipliceras den täckta bandytan med en konstant: areafactor. Denna bestäms av produktens geometri, och är lika med förhållandet mellan den yta av produkten, över vilken värmeövergång sker, och den yta, produkten upptar på bandet.

Till sist får vi

$$T_{\text{air}}(z) = T_{\text{air}}(h) + \frac{1}{\text{airflow} \cdot c_{p, \text{air}}} \int_z^h \alpha (T(z') - T_{\text{air}}(z')) \cdot (\text{areafactor} \cdot \text{bandcover}(z') \cdot A_{\text{belt}}/h) dz'$$

3.1.4 Andra värmekällor

Förutom från produkten upptar luften värme från fläktarnas motorer och från frysarens omgivning (t ex genom läckströmning vid produktbandets in- och utgång). Värmeutbyte med metallen i frysaren sker vid icke-stationära förhållanden, t ex uppstart.

Temperaturökningen blir

$$\Delta T_{\text{air}} = (\text{baseload} + \dot{Q}_{\text{met}}) / \text{airflow}$$

(baseload: baslast, konstant effekt,
 \dot{Q} : effekt från metallen i frysaren)

3.1.5 Kylbatteri

Kyleffekten på luften ges av

$$\dot{Q} = k(T_{\text{air}} - T_{\text{met}}) = k(T_{\text{air}} - T_{\text{evap}} - (T_{\text{met}} - T_{\text{evap}}))$$

(T_{air} : inströmmande luftens temperatur
 T_{met} : temperatur hos metallen i kylbatteriet
 T_{evap} : köldmediets förångningstemperatur)

Här är $T_{\text{met}} - T_{\text{evap}}$ (temperaturdifferensen på köldmediesidan) endast svagt effektberoende, och kan sättas konstant = $\Delta T_{\text{m,e}}$ förutsatt att

$$T_{\text{air}} - T_{\text{evap}} \gg \Delta T_{\text{m,e}}$$

Värmeövergångstalet har bestämts empiriskt till

$$k = k_1 W_{\text{bat}}^{A_{\text{bat}}} \left(\frac{T_{\text{air}}}{W_{\text{bat}}} \right)^{k_2}$$

(W_{bat} : luftens hastighet i kylbatteriet
 A_{bat} : luftströmmens tvärsnittsyta i kylbatteriet
 T_{air} : inströmmande luftens absoluta temp.
 k_1, k_2 : konstanter)

Lufttemperaturdifferensen över kylbatteriet blir

$$\begin{aligned} \Delta T_{\text{air}} &= \frac{\dot{Q}}{\text{airflow} \cdot c_{p, \text{air}}} = \\ &= k_1 \frac{W_{\text{bat}}^{A_{\text{bat}}}}{(W_{\text{bat}}^{A_{\text{bat}}} e_{\text{air}}) c_{p, \text{air}}} \left(\frac{T_{\text{air}}}{W_{\text{bat}}} \right)^{k_2} \\ &\cdot (T_{\text{air}} - T_{\text{evap}} - \Delta T_{\text{m,e}}) = \\ &= \frac{k_1}{e_{\text{air}} c_{p, \text{air}}} \left(\frac{T_{\text{air}}}{W_{\text{bat}}} \right)^{k_2} (T_{\text{air}} - T_{\text{evap}} - \Delta T_{\text{m,e}}) \end{aligned}$$

3.2 Diskret modell av infrysningsprocessen

Modeller för frysare och produkt samordnas. Ekvationerna för bestämning av lufttemperaturer kombineras till en iterativ algoritm.

3.2.1 Beskrivning av principerna

De samband och ekvationer, som ställts upp, kommer att utgöra stommen i det datorprogram som skall användas för simulering av frysaren. Först måste vi fastställa de principer, enligt vilka produktstacken delas in i ett antal skikt och vilka bestämmer simuleringen av produkttransporten.

Temperaturerna beräknas i ett antal punkter:

- *) T_{air} i $m+1$ punkter, från ingång till utgång, med konstanta lägen
- *) T i $m+2$ punkter - ingång, utgång samt m st punkter, medföljande bandet

Med tidsintervall $holdtime/m$ skiftas produktvärdena framåt mellan närliggande punkter, motsvarande att punkterna flyttas tillbaka till sina utgångslägen.

Således: punkt $m+1$ får utgångsvärdena från punkt m , som nått utgången; punkt m hoppar tillbaka och får värdena från punkt $m-1$;.....; punkt 1 hoppar tillbaka till ingången och får värdena från punkt 0 .

Punkt längs bandet markeras med index 'i':

$$T_{\text{air}}[i], \quad i=0, 1, \dots, m$$

$$T[i, j], \quad i=0, 1, \dots, m+1; \quad j=-1, 0, 1, \dots, n$$

Index 'j' markerar läge i produkten: $T[i,0]$: ytemperatur; $T[i,n]$: mittemperatur

T_{air} hålls konstanta mellan skift. VLE för de glidande punkterna beräknas med lufttemperaturen linjärt interpolerad mellan $T_{\text{air}}[i]$. Punkternas lägen markeras med tillståndet 's': $0 \leq s \leq 1$, där $s=1$ motsvarar utgångsläge, dvs punkt 1 sammanfaller med punkt 0 , och $s=0$ motsvarar läget vid skift, dvs punkt m sammanfaller med punkt $m+1$. s beräknas då:

$$s = \frac{\text{tid för nästa skift} - \text{aktuell tid}}{\text{tidsintervall mellan skift}}$$

Den interpolerade lufttemperaturen vid produktunkt 'i' blir då

$$s \cdot T_{\text{air}}[i-1] + (1-s) \cdot T_{\text{air}}[i]$$

Samtidigt med skift, korrigeras T_{air} . Produktpunkterna och luftpunkterna sammanfaller då, så att värmeövergången vid luftpunkterna kan bestämmas, och därmed lufttemperaturerna.

3.2.2 Lufttemperaturer

Temperaturerna i stacken ska beräknas i vissa punkter. Vår kontinuerliga modell överförs till diskret form.

Vi utgår från ekvationen (se kap 3.1.3)

$$T_{\text{air}}(z) = T_{\text{air}}(h) + \frac{1}{\text{airflow} \cdot c_{p, \text{air}}} \int_z^h \dot{q}_z(z') dz'$$

Definiera

$$z[i] = \frac{i}{m} h$$

och byt

$$T_{\text{air}}(z[i]) \rightarrow T_{\text{air}}[i]$$

Antag att bandtäckningen är styckvis konstant, dvs

$$\text{bandcover}(z) = \text{bandcover}(z[i+1]), \quad z[i] < z \leq z[i+1]$$

Byt sedan

$$\text{bandcover}(z[i]) \rightarrow \text{bandcover}[i]$$

$$A_z(z[i]) \rightarrow A_z[i]$$

Vi får

$$T_{\text{air}}[i] = T_{\text{air}}[m] + \frac{1}{\text{airflow} \cdot c_{p, \text{air}}} \int_{z[i]}^{h=z[m]} \dot{q}_z(z') dz'$$

⇒

$$T_{\text{air}}[i] - T_{\text{air}}[i+1] = \frac{1}{\text{airflow} \cdot c_{p, \text{air}}} \int_{z[i]}^{z[i+1]} \dot{q}_z(z') dz' \approx$$

$$\approx \frac{1}{\text{airflow} \cdot c_{p, \text{air}}}$$

$$\begin{aligned}
 & \cdot (z[i+1] - z[i]) \frac{\dot{q}_z(z[i+1]) + \dot{q}_z(z[i])}{2} = \\
 & = \frac{1}{\text{airflow} \cdot c_{p, \text{air}}} \frac{h}{m} \frac{\dot{q}(z[i+1]) + \dot{q}(z[i])}{2} A_z[i+1] = \\
 & = \frac{1}{\text{airflow} \cdot c_{p, \text{air}}} \\
 & \frac{\alpha}{2} (T[i+1, 0] - T_{\text{air}}[i+1] + T[i, 0] - T_{\text{air}}[i]) \cdot \\
 & \text{areafactor} \cdot \text{bandcover}[i+1] A_{\text{belt}} / m
 \end{aligned}$$

vilket kan skrivas som

$$\left\{ \begin{aligned}
 \dot{q}_m[i+1] &= \frac{\alpha}{2} (T[i+1, 0] - T_{\text{air}}[i+1] + T[i, 0] - T_{\text{air}}[i]) \\
 A[i+1] &= \text{areafactor} \cdot \text{bandcover}[i+1] A_{\text{belt}} / m \\
 T_{\text{air}}[i] &= T_{\text{air}}[i+1] + \frac{1}{\text{airflow} \cdot c_{p, \text{air}}} \dot{q}_m[i+1] A_z[i+1]
 \end{aligned} \right.$$

$$\left[\begin{aligned}
 \dot{q}_m[i+1]: & \text{approximativt medelvärde av } \dot{q} \\
 & \text{i intervallet } z[i] < z \leq z[i+1] \\
 A[i+1]: & \text{sammanlagda produktytan i intervallet } z[i] < z \leq z[i+1]
 \end{aligned} \right]$$

för $i = m-1, \dots, 0$

3.2.3 Konvergenstkroll

De ekvationer, som bestämmer lufttemperaturerna, bildar sammanlagda ett ekvationssystem, som lämpligen löses iterativt. Tilldelningarna görs upprepade gånger uppifrån och ned, och med insättande av nya värden i högerleden efterhand som värdena räknats fram. En grov kontroll av konvergensten görs genom insättande av rimliga värden:

$$\left\{ \begin{aligned}
 \dot{q}_m[i+1] &= \frac{35}{2} (T[i+1, 0] - T_{\text{air}}[i+1] + T[i, 0] - T_{\text{air}}[i]) \\
 A[i+1] &= 2 \cdot 0.58 \cdot 43 \approx 50 \\
 T_{\text{air}}[i] &= T_{\text{air}}[i+1] + \frac{1}{(3.2.5 \cdot 1.293) \cdot 1000} \dot{q}_m[i+1] A_z[i+1]
 \end{aligned} \right.$$

$$\Rightarrow \Delta \dot{q}[i+1] = -35 (\Delta T_{\text{air}}[i] + \Delta T_{\text{air}}[i+1])$$

$$\Rightarrow \Delta T_{\text{air}}[i] \approx \Delta T_{\text{air}}[i+1] + 5 \cdot 10^{-4} \Delta \dot{q}_m[i+1] \approx \Delta T_{\text{air}}[i+1]$$

$$\left(\begin{array}{l} \Delta T_{\text{air}}: \text{ändring sedan föregående iteration} \\ \Delta' T_{\text{air}}: \text{ändring i föregående iteration} \end{array} \right)$$

I princip fortplantas ΔT_{air} oförändrad genom stacken.

För kylbatteriet gäller (se kap 3.1.5)

$$T_{\text{air}} [\text{m}] \approx T_{\text{air}} [0] - 255.0 / (1.293 \cdot 1000) \\ \cdot ((-30 + 273.15) / 3.5)^{0.34} \cdot (T_{\text{air}} [0] - T_{\text{evap}} - 1.3)$$

($T_{\text{air}} [0] \approx -30$ insatt i värmeövergångstalet)

⇒

$$\Delta T_{\text{air}} [\text{m}] \approx \Delta T_{\text{air}} [0] - 0.83 \Delta T_{\text{air}} [0] = 0.17 \Delta T_{\text{air}} [0]$$

Felet i T_{air} bör således minska till under en femtedel vid varje iteration. Då t ex $\Delta T_{\text{air}} [0]$ blir tillräckligt liten avbryter vi rundgången.

Hur noggrant resultat som önskas beror dels på resultatets inverkan på övriga delar av modellen, dels på hur många signifikanta siffror som är relevanta vid jämförelse med praktiskt uppmätta temperaturer.

Inverkan på produkten i modellen bestäms av värmeövergången, vilken beror av temperaturdifferensen mellan produkt och luft. Om denna ligger i storleksordningen $>10^\circ$ och vi önskar ett fel på högst 1% i den beräknade värmeövergången pga felet i T_{air} , bör vi begränsa detta fel till $<0,1^\circ$.

Avvikelser mindre än denna är ej heller relevanta vid praktiska mätningar.

3.3 Allmän beskrivning av program GYRoFREEZE

Programmet och dess principer beskrivs kortfattat.

3.3.1 Principer

Simuleringsprogrammet för GYRoFREEZE har gjorts 'användarvänligt' enligt följande principer:

- (*) Användaren styr programförloppet genom val från menyer
- (*) Vissa variabler i programmet kan visas/ändras mellan simuleringar
- (*) Simulering kan göras etappvis, med eventuella ändringar av variabler mellan 'etapperna'
- (*) Simulering kan avbrytas (genom avbrytande av exekvering) och omstartas från början efter eventuella variabeländringar
- (*) Resultatdata lagras på extern fil, för efterbehandling. Utgångstemperatur plottas på outputfil (= bildskärm)
- (*) Med givna fasta värden på lufttemperaturerna i frysaren, kan infrysningsskurvor tas fram (lösning av VLE med givna randvärden)

3.3.2 Val av programspråk

Försök har gjorts att skriva programmet i SIMNON. Den slutliga versionen är dock skriven i Pascal, detta på grund av följande problem med SIMNON-versionen:

Antalet variabler i systemet är stort (i storleksordning: flera hundra) och måste anpassas efter SIMNONS övre gränser för antal tillstånd, hjälpvariabler osv. Ett SIMNON-program av denna storlek kräver långa exekveringstider, och blir därför aningen tålamodsprövande för användaren. Dessutom blir även ändringar av intervallängden Δx (t ex i samband med ändring av tjockleken D) tidskrävande, eftersom antalet variabler måste ändras, genom editering av själva programmet.

Övergången till Pascal innebar följande fördelar: Ett godtyckligt antal variabler kan införas. Genom användning av fält ('array') kan simuleringsdelen av programmet göras kompaktare och därmed överskådligare och lättare att göra ändringar i. Intervallängden Δx kan läggas in som en parameter i programmet, och ändras under exekveringens gång. Justering av intervallängder kan göras optimal, med avseende på exekveringstid och önskad beräkningsnoggrannhet.

Nackdelen med Pascal är att det krävs ett stort antal programrader för

ändring och utskrift av variabler och parametrar. Plottning måste göras med hjälp av specialprogram för detta (t ex med programpaketet IDPAC).

3.3.3 Parametrar, tillstånd och hjälpvariabler

Programvariablerna har indelats i parametrar ('parameters') och tillstånd ('states').

Parametrarna bestäms före en simulering och ändras inte under denna.

Tillstånden ges initialvärden före simulering.

Vissa parametrar beror av andra, och måste beräknas i programmet, andra parametrar korrigeras (t ex delta ($\equiv \Delta x$), som avpassas så att antalet skikt tjocklek/delta blir ett jämnt heltal).

Variabler som beräknas under själva simuleringen, och som inte är 'intressanta' att dokumentera, definieras i underprogrammet 'Run'. Hit hör vissa beroende parametrar och tillstånd, samt hjälpvariabler som förenklar programmet eller styr simuleringen. Övriga variabler är globalt definierade.

3.3.4 Tid

Bland tillstånden finns inkluderade variabler som styr programflödet genom att bestämma när vissa rutiner ska genomlöpas, och när simuleringen ska avbrytas. Dessa tillstånd är: storetime, shifttime, stoptime. Själva tiden representeras av time. Time räknas upp med deltat ($\equiv \Delta t$) efter varje loop i 'process'-delen av Run.

Varje gång time uppnått storetime, shifttime eller stoptime, räknas ifrågevarande variabel upp med värdet av parametern storeint, shiftint resp stopint, samtidigt som motsvarande del av Run exekveras. I det senare fallet, time = stoptime, avbryts simuleringen.

Tillståndet shifttime representeras även av s, som är den variabel som sparas mellan simuleringar. Detta eftersom s direkt motsvarar produktpunkternas lägen. Vid ändring av hålltiden, dvs av bandhastigheten, ändras shifttime men inte s.

Tider representeras i hundradels sekunder. Alla tider är av heltalstyp, för att undvika förskjutningar p g a avrundningsfel. Vid operatörskommunikation, dvs vid utskrift eller vid inläsning från inputfil sker omvandling till resp från minuter (eller sekunder).

3.3.5 Filer

Externa filer används dels för lagring av parametrar/tillstånd mellan simuleringar, dels för lagring av resultatdata.

- *) parameterfile: fil för parametervärden
- *) statefile: fil för tillståndsvärden
- *) reservefile: reservfil för parameter- eller tillståndsvärden; utnyttjas t ex då värden från en fil ska flyttas till en annan fil
- *) backup: används för lagring av en uppsättning parameter- och tillståndsvärden; lagringen utförs vid start av simulering; en simulering som avbrutits genom att programmets exekvering avbrutits, kan återupptas efter val av backup-restart efter det att programmet återuppstartats
- *) storefile: här lagras värden, som samplats under simulering; vidare behandling av värdena, t ex plottning av kurvor, kan sedan göras med utnyttjande av särskilda program för detta
- *) printfile: aktuella parametrar/tillstånd kan skrivas in på printfile, för senare utskrift

En intern fil ('buffertfile') utnyttjas i programmet för tillfällig lagring av parametrar/tillstånd; detta eftersom Pascal ej tillåter omväxlande skrivning/läsning på/från samma fil.

3.3.6 Användning av programmet

Efter start av programmet presenteras en meny:

- (*) 'p: parameters' - visa/ändra/spara parametrar, se nedan.
- (*) 's: states' - visa/ändra/spara tillstånd, se nedan.
- (*) 'r: run' - starta simulering med aktuella parameter- & tillståndsvärden.
- (*) 'i: initialize' - uppstartsproceduren upprepas. Kan t ex följas av 'backup-restart'.
- (*) 'b: backup-restart' - inläsning av de parametrar och begynnelsevärden på tillstånden, som gällde vid senaste simuleringen. Kan användas då exekveringen av programmet av någon anledning avbrutits (t ex under simulering) och återuppstartats.
- (*) 'c: choose' - val av vilka variabler som skall lagras, samt val av 'stopcheck', en enkel konvergenstest, eller av 'oneproduct', infrysningsskurva med fasta lufttemperaturer.
- (*) 'e: exit' - avsluta exekvering.

'p: parameters'

Aktuella parametervärden och en meny skrivs ut:

- (*) 'g: get' - inläsning av parametervärden från parameterfile. Har filslut nåtts, läses från början av filen. Med upprepade 'get' kan filen genomsökas efter önskade värden.
- (*) 'x: get(reserve)' - inläsning av parametervärden från reservefile.
- (*) 'p: print' - utskrift av aktuella parametrar på printfile, för senare utskrift.
- (*) 'c: change' - ändring av parametrar. Undernivå till 'parameters'. Välj parameter. Ge det nya värdet. När önskade ändringar utförts, återgå till 'parameters' med 'exit'. Härvid beräknas och korrigeras vissa parametrar.
- (*) 's: save' - spara parametervärden. Värdena skrivs på buffertfile, och skrivs sedan över till parameterfile vid utgång ur 'parameters'.
- (*) 'd: delete buffert' - buffertfile töms. Före utgång ur 'parameters' kan de gamla parameteruppsättningar som önskas sparade, tas fram med 'get' och åter skrivs på buffertfile med 'save'.
- (*) 'e: exit' - utgång ur 'parameters'. Återgång till huvudprogrammet, efter det att innehållet i buffertfile skrivits över till parameterfile.

's: states'

analogt med 'parameters'.

4 PROCESSDYNAMIK & REGLERING

Vi ska nu, med hjälp av det framtagna datorprogrammet, undersöka dynamiken hos infrysningsprocessen. De stegsvar, som erhålls vid ändring av vissa parametrar och tillstånd, ligger sedan till grund för det regleringsförslag, som sammanfattas i slutet av kapitlet.

4.1 Modellens egenskaper

Processdynamiken beskrivs grafiskt, med stegsvar för olika variabeländringar. Modellens komplexitet diskuteras.

4.1.1 Stegsvär

I figurer 4.1-5 visas de grafer, som erhållits vid exekvering av program GYROFREEZE (plottning har utförts med IDPAC). Dessa variabler har stegändrats:

Fig 4.1) holdtime - hålltiden minskas, dvs bandhastigheten ökas. Produkten blir varmare, vilket i sin tur medför en höjning av lufttemperaturen.

Fig 4.2) T_{evap} - förångningstemperaturen höjs. Lufttemperaturen ökar - produkttemperaturen ökar, vilket medför en viss ytterligare höjning av lufttemperaturen.

Fig 4.3) W_{frost} - minskning av lufthastigheten (t ex p g a nedfrostning). Minskat luftflöde ger en höjning av lufttemperaturen. Minskad värmeövergång medför varmare produkt, vilket i sin tur höjer lufttemperaturen ytterligare.

Fig 4.4) bandcover - bandtäckningen ökar. Gradvis ökande belastning medför höjning av lufttemperaturen. Ökande lufttemperatur ger efter hand varmare produkt. Insvängningstiden blir dubbelt så lång (\approx två hålltider) i förhållande till vid övriga störningar.

Fig 4.5) $T_{[0,j]}$ - ingångstemperaturen höjs. Temperaturhöjningen återfinns (men har reducerats) vid utgången efter en hålltid. Lufttemperaturen höjs, vilket ger en viss ytterligare produkttemperaturhöjning.

Av dessa, och andra tänkbara ändringar, kan två huvudtyper urskiljas. Den ena innebär i första hand påverkan på kylningen av produkt; den andra innebär i första hand förändrade förhållanden vid inmatningen. Ändringar av holdtime, T_{evap} och W_{frost} kan räknas till den första kategorin, medan alla typer av förändringar hos den inmatade produkten, t ex ökad ingångstemperatur, räknas till den senare kategorin. Bandtäckningen ger en kombination av de två typerna: mer produkt vid inmatningen, medför efter hand sämre kylning. Denna påverkan ger således det mest komplexa stegsvaret.

4.1.2 Utökning/reduktion av modellen

Till det relativa fåtal ekvationer, som ingår i modellen, kan efterhand läggas andra kända samband, som inverkar på processens tillstånd. Lufttemperaturerna, t ex, påverkas av en mängd faktorer, vilka det är lämpligt att genom mätningar uppskatta kvantitativt. I denna första version av modellen har antalet ekvationer medvetet valts lågt, för att den grundläggande dynamiken ska vara enkel att studera.

De stegsvar som framtagits antyder att processdynamiken är tillräckligt okomplicerad för att modellen bör kunna uttryckas med ett färre antal variabler och ekvationer. Om ett stort antal simuleringar önskas utförda, med variablerna varierande kring samma stationära läge, kan en approximativ, linjäriserad modell utnyttjas.

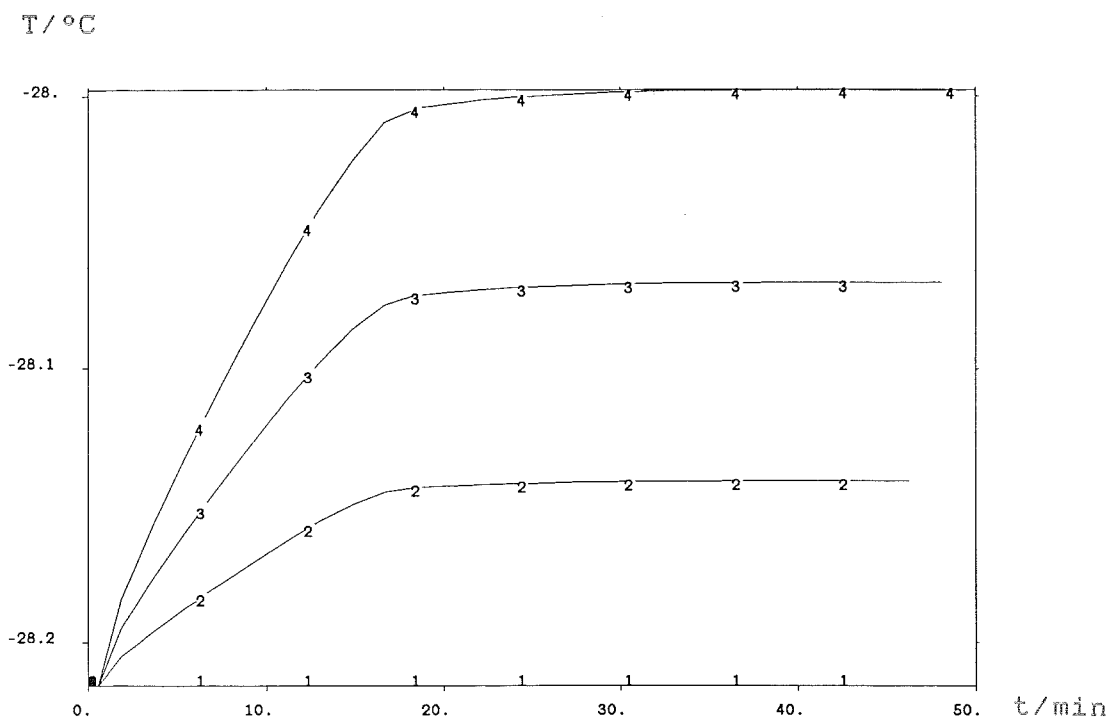
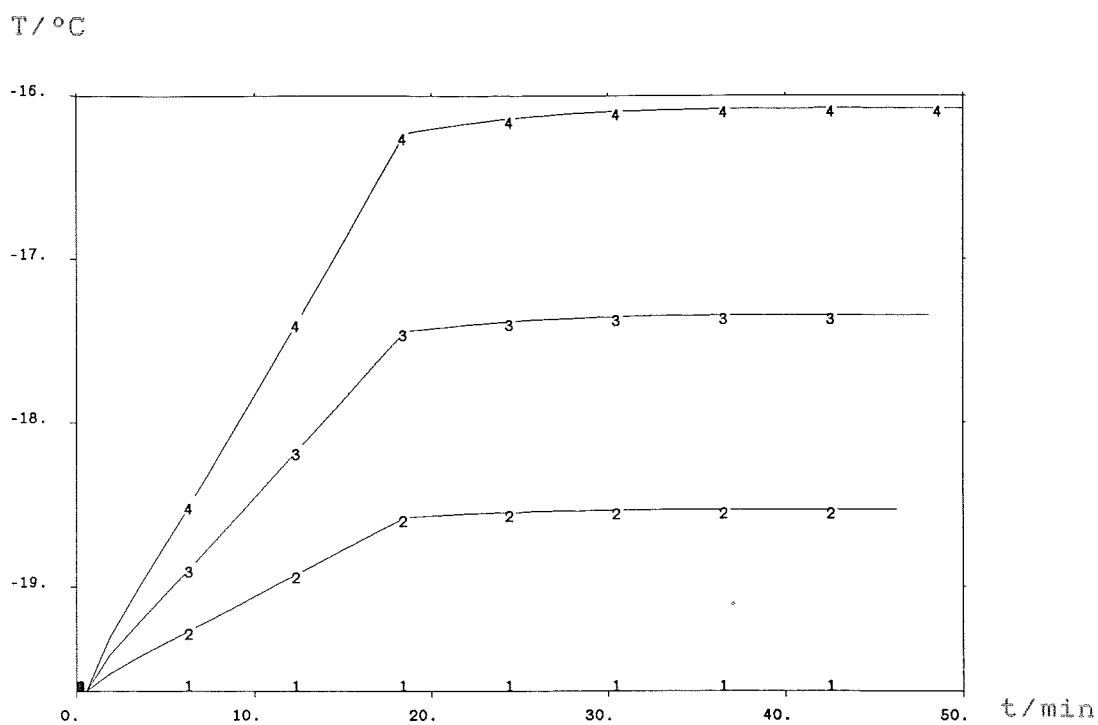


Fig 4.1. Produkttemperatur vid utgången (övre diagr.) och lufttemperatur efter stacken (undre diagr.) efter stegändring av holdtime från 20 min till:
 (1) ingen ändr. (2) 19.5 min (3) 19.0 min (4) 18.5 min .

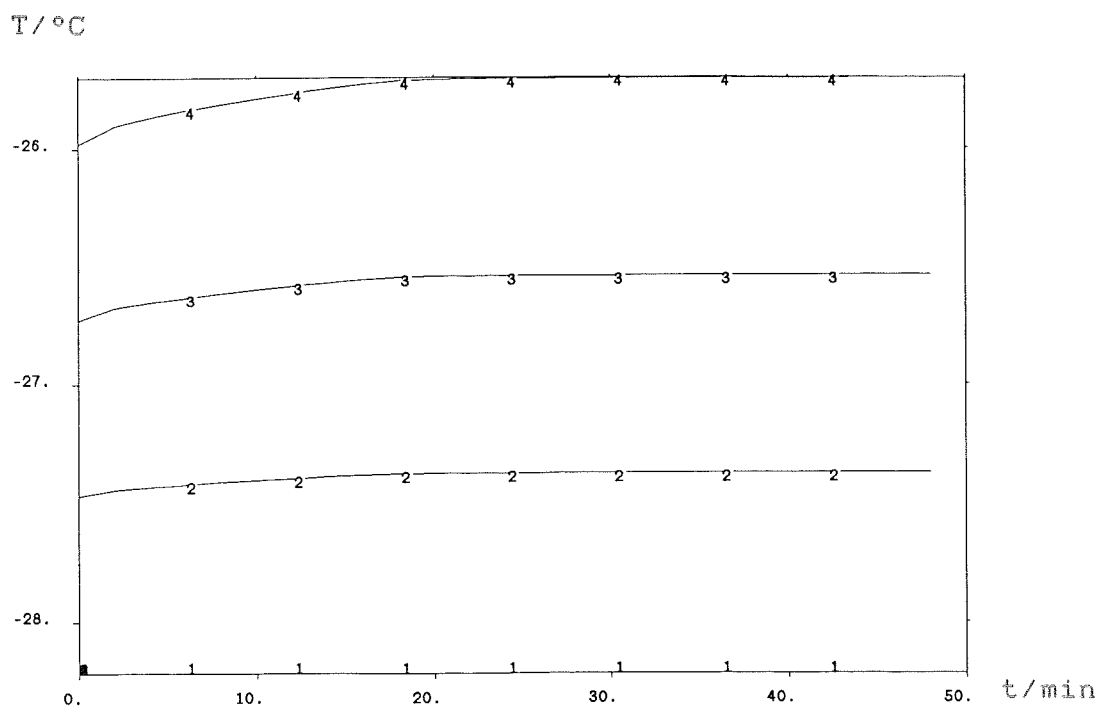
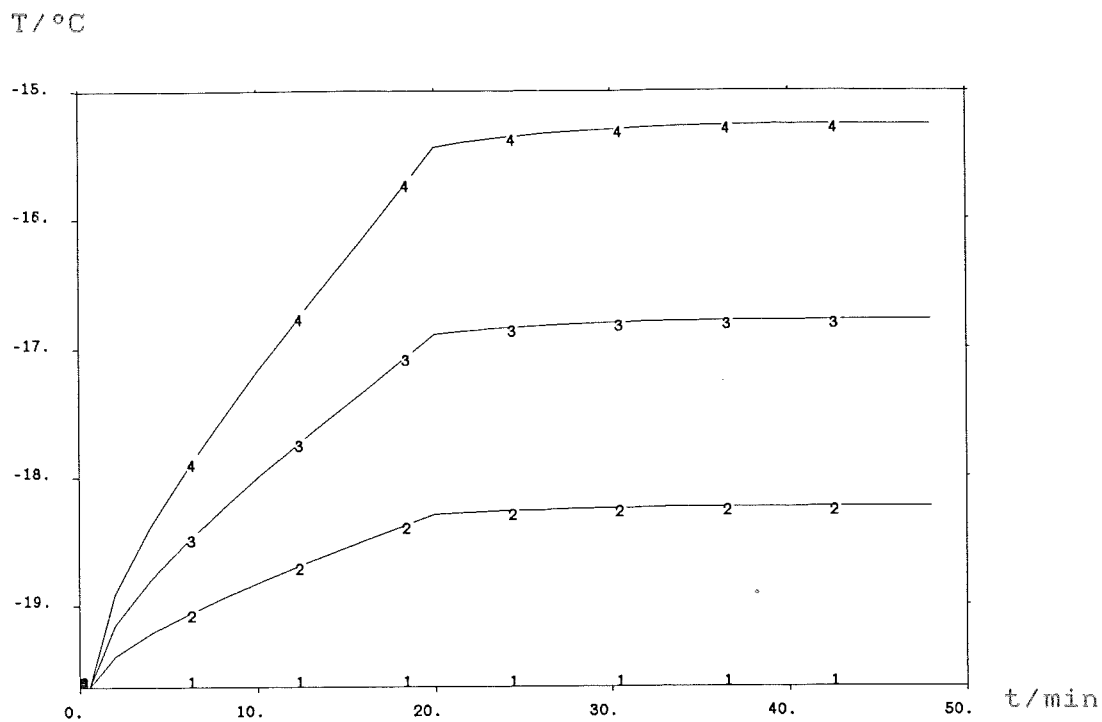
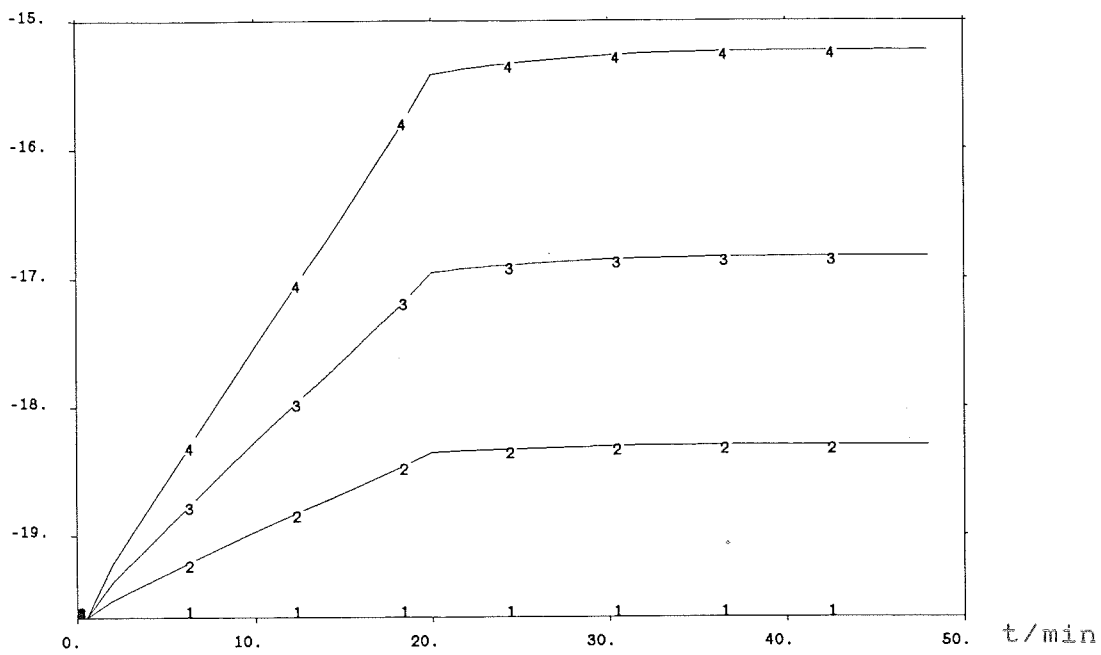


Fig 4.2. Produkttemperatur vid utgången (övre diagr.) och lufttemperatur efter stacken (undre diagr.) efter stegändring av T_{evap} från -40° till:
 (1) ingen ändr. (2) 39° (3) 38° (4) 37° .

T/°C



T/°C

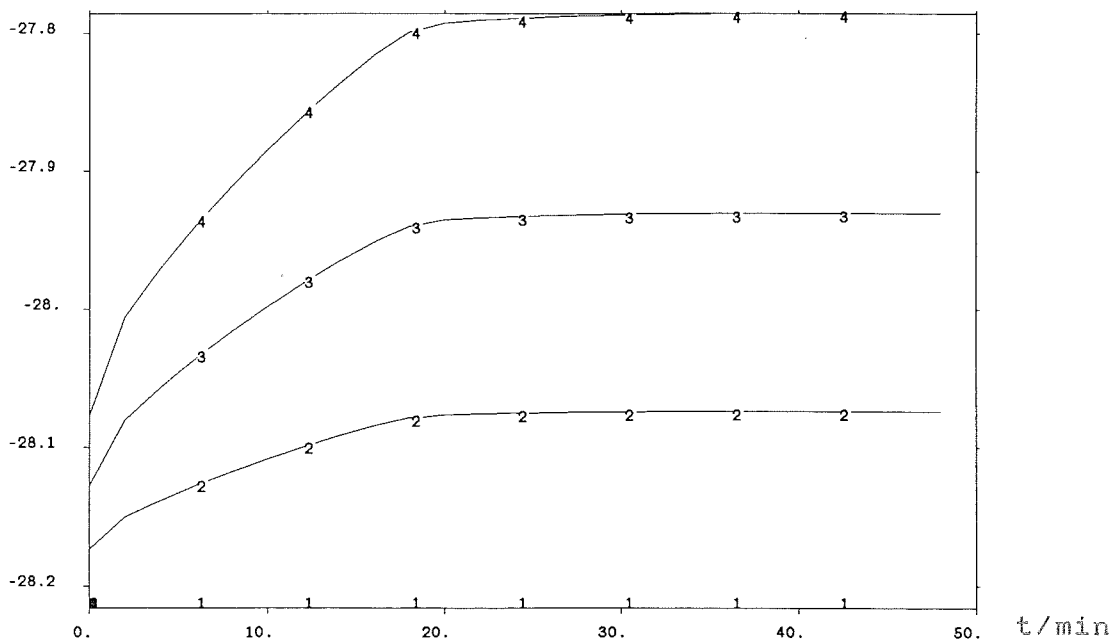
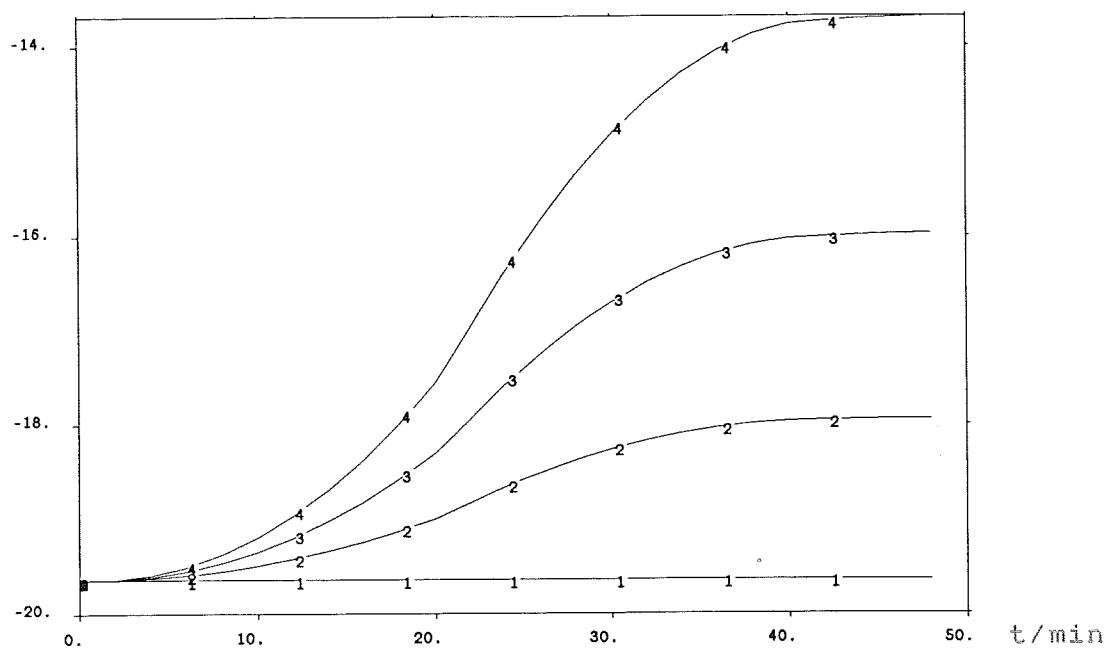


Fig 4.3. Produkttemperatur vid utgången (övre diagr.) och lufttemperatur efter stacken (undre diagr.) efter stegändring av W_{frost} (minskning av lufthastigheten) från 0 m/s till:

(1) ingen ändr. (2) 0.1 m/s (3) 0.2 m/s (4) 0.3 m/s .

T/°C



T/°C

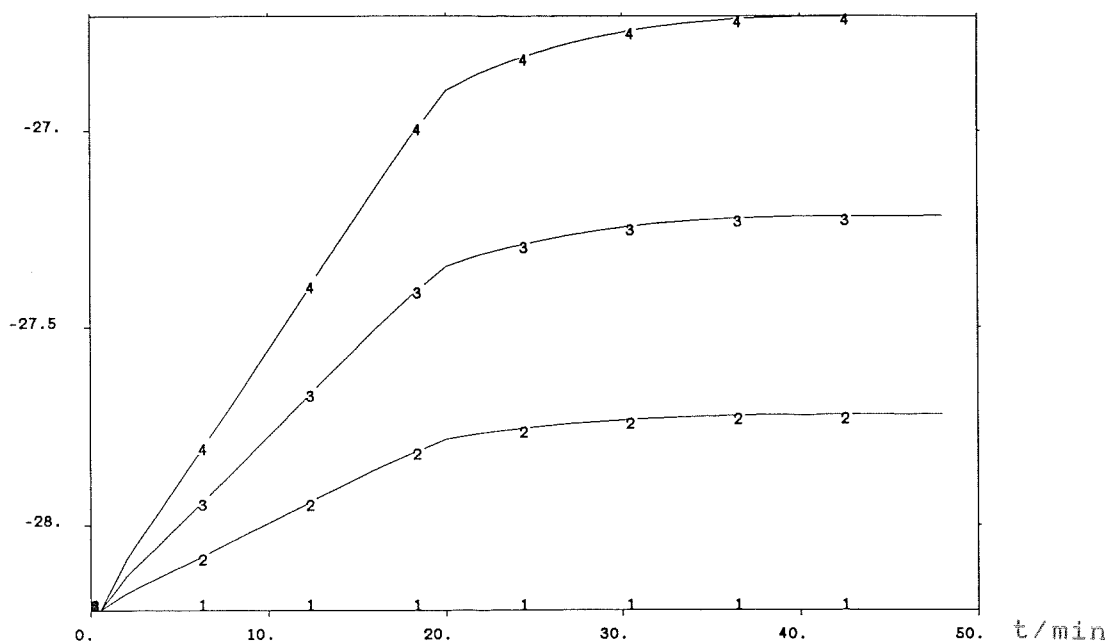
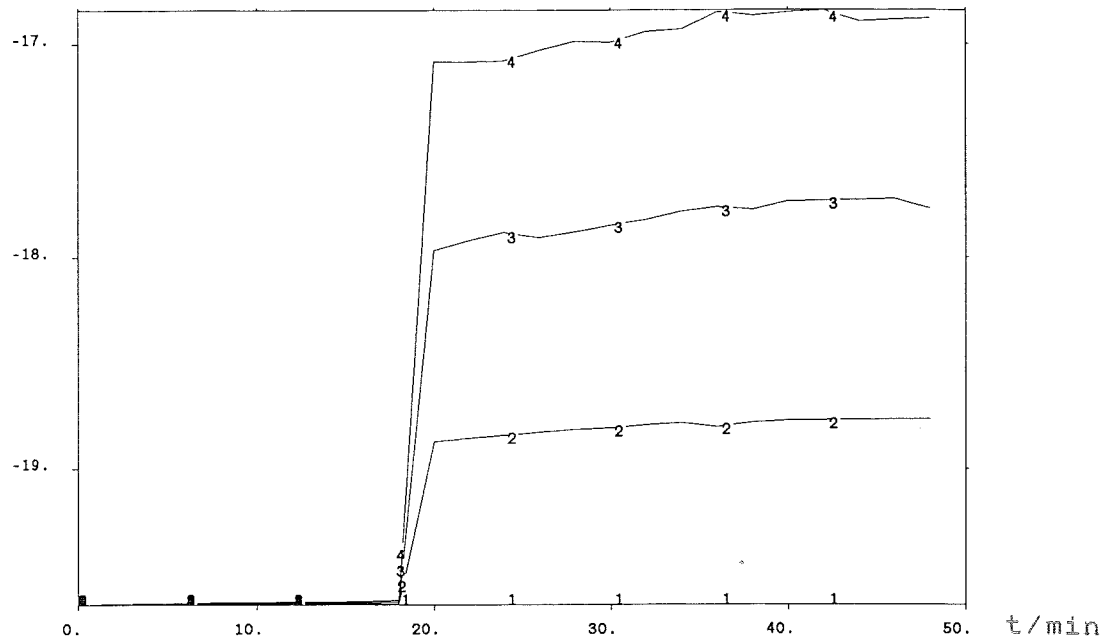


Fig 4.4. Produkttemperatur vid utgången (övre diagr.) och lufttemperatur efter stacken (undre diagr.) efter stegändring av bandcover från 0.58 till: (1) ingen ändr. (2) 0.61 (3) 0.64 (4) 0.67 .

T/°C



T/°C

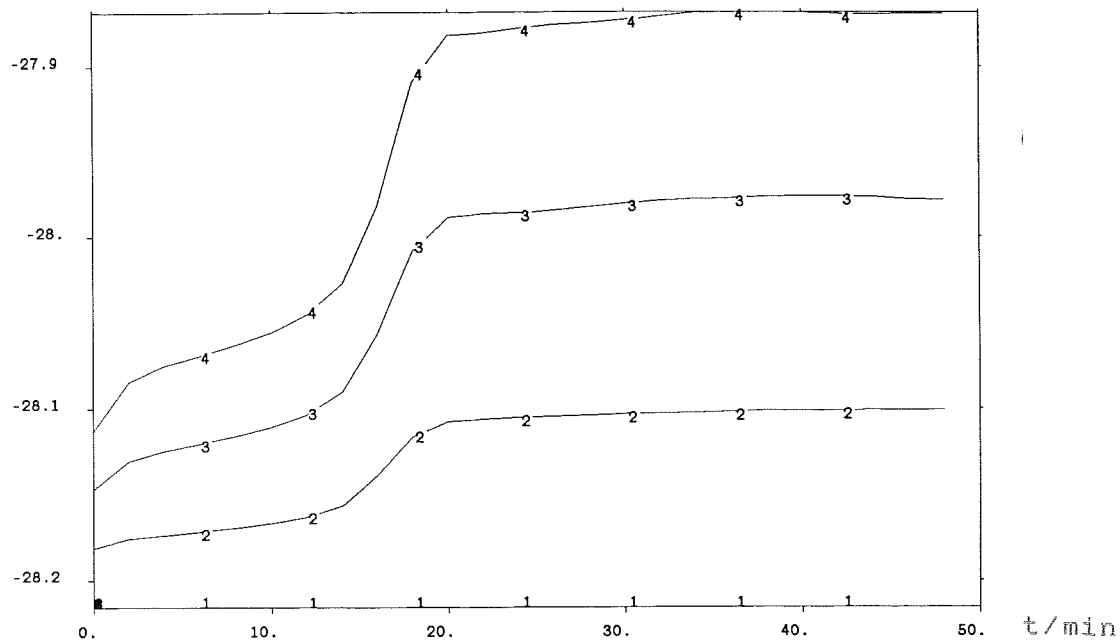


Fig 4.5. Produkttemperatur vid utgången (övre diagr.) och lufttemperatur efter stacken (undre diagr.) efter stegändring av $T[0,1]$ (ingångstemperatur) från 0° till:

(1) ingen ändr. (2) 2.5° (3) 5° (4) 7.5° .

4.2 Reglering

PI-reglering med/utan framkoppling simuleras, och praktiska åtgärder föreslås.

4.2.1 Kriterier för reglering

De krav vi i kap 1 ställde på frysaren i drift var:

- 1) Utgångstemperatur $\leq -18^\circ$
- 2) Produktivitet
- 3) Jämn gång

Vid normal drift är förångningstemperatur, bandhastighet, osv inställda på konstanta, beräknade värden. En kompromiss måste göras mellan krav 1 och 2: om frysaren ställs in för en utgångstemperatur på -18° (vid ideala förhållanden), kommer omkring hälften av produkten att få för hög temperatur, p g a fluktuationer kring medelvärdet. Genom förlängning av hålltiden minskas andelen 'varm' produkt, men samtidigt minskas produktflödet (Dessutom utsätts en större andel produkt för oönskade fuktförluster).

För att minska både andelen otillräckligt fryst och andelen alltför hårt infrost produkt kan vi införa någon form av reglering. Ju snabbare denna reglering reagerar på störningar hos processen, desto mindre blir produktens temperaturfluktuationer.

Dock måste de vinster som görs med regleringen vägas mot de kostnader som tillförs i form av dels de investeringar som krävs vad gäller anpassning och intrimning av utrustningen, dels slitage och störningar som tillkommer beroende på vilken styrprincip som används (se krav nr 3 !).

Vid en diskussion av variabla förhållanden hos infrysningsprocessen, kan det vara värt att lägga märke till att även frysarens totala kyleffekt kan varieras, oberoende av t ex produktflödet.

4.2.2 Regulator

P-regulator: För att undersöka möjligheterna till reglering kopplar vi först en P-regulator till modellen. P-regulatorns funktion är att styrsignalen görs proportionell mot utsignalens avvikelse från ett referensvärde ('P':

proportionell). Som styrsignal till processen väljs bandhastigheten, dvs bandet körs med en hastighet lika med den vi beräknat för stationära förhållanden plus den beräknade styrsignalen.

Sålunda beräknas först ett reglerfel

$$e = T_{\text{ref}} - T_{\text{level}}$$

$$\left(\begin{array}{l} T_{\text{ref}}: \text{referenstemperatur} \\ T_{\text{level}}: \text{utjämnad temperatur hos produkt vid utgången} \end{array} \right)$$

varpå styrsignalen bestäms till

$$u = K \cdot e$$

I programmet bestäms bandhastigheten av skiftintervallet.

I fig 4.6 visas resultatet, då T_{ref} ändras. Stegsvaret ger en illustration av systemets dynamik (Vid normal drift hålls referensvärdet konstant). Styrsignalen representeras med den hålltid, vilken motsvarar bandhastigheten vid aktuell tidpunkt (Transienten vid uppstart av regulatorn kan bortses från - beror av svaghet i programmet). Som synes i figuren medför ett högt värde på förstärkningen K att regleringen blir snabb, och reglerfelet minskar. Dock erhåller vi 'knyckar' med en hålltids mellanrum (Efter en hålltid börjar de produkter matas ut, som inte var inne i frysaren då T_{ref} ändrades).

Regulatorn fungerar approximativt som en kontinuerlig P-regulator, fastän styrsignalen hålls konstant mellan skift i modellen. Förutsättningen för detta är att styrsignalen ställs om tillräckligt många gånger per hålltid (här 10 ggr). Eftersom vi reglerar med själva skiftintervallet, förutsätts även att styrsignalen hålls inom rimliga gränser.

PI-regulator: För att eliminera det stationära felet, utan att behöva ta till en extrem förstärkning, och samtidigt minska effekten av de periodiska knyckarna, förser vi regulatorn med integration.

En s k PI-regulator ger en styrsignal

$$u(t) = K \left[e(t) + \frac{1}{T_i} \int_0^t e(t') dt' \right]$$

(T_i : 'integraltid')

Integraltiden T_i är en parameter som bestämmer inverkan av integraldelen.

Vi skriver regleralgoritmen t ex som

$$\text{Int} = \text{Int} + K \cdot \text{shiftint} \cdot e / T_i$$

$$u = K \cdot e + \text{Int}$$

Integraldelen Int räknas upp vid varje skift. Integrationen är approximativt kontinuerlig om

$$\text{shiftint} \ll \text{holdtime}$$

(i det här fallet är $\text{shiftint} = \text{holdtime}/10$).

Figur 4.7 visar resultatet vid ändring av T_{ref} och olika värden på K och T_i . I fig 4.7a varieras T_i , i fig 4.7b varieras K, och i fig 4.7c visas tre lämpliga kombinationer av K och T_i . Man kan som synes välja att låta stegsvaret bli en långsam ramp upp till börvärdet, eller att göra regulatorn snabbare, med vissa svängningar kring börvärdet och stökigare styrsignal.

Hur regulatorn motverkar störningar, visas i fig 4.8. Temperaturen svänger efter hand in mot det rätta värdet, oavsett vilka regulatorparametrar som valts.

4.2.3 Avancerade regulatorer

Ett sätt att förse regulatorn med mer information om processens tillstånd är att använda tillståndsåterkoppling. Genom att mäta produkttemperaturen på flera ställen längs bandet, och låta dessa mätvärden påverka styrsignalen, ökar vi möjligheterna att specificera regulatorns funktion.

Direkt mätning av produkttemperaturen i stacken är knappast genomförbar. Via lufttemperaturdifferenserna (främst före-efter stacken) kan temperaturändringar hos produkten indirekt detekteras. Mätningarna blir tyvärr små och känsligheten för störningar stor (t ex lufthastigheten påverkar förutsättningarna för värmeövergången).

En ren utsignalåterkoppling (kombinerad med framkoppling från mätbara störningar) är mest realistisk. Genom mätningar identifieras parametrar och koefficienter i en förenklad modell av frysaren, varpå parametrarna för en regulator med önskade egenskaper kan beräknas.

Arbetet med mätningar på processen och, framför allt, införande av givare för tillräckligt snabbt insamlande av mätvärden till den presumtiva regulatorn, ligger dock utanför ramen för detta examensarbete. Funktionen hos P- och PI-reglering av frysaren bör därför tills vidare vara tillräcklig illustration av möjligheterna till reglering.

4.2.4 Sampling

De mätmetoder, som för närvarande föreligger, medger inte kontinuerlig mätning av produkttemperaturen. Om värden samplas med längre intervaller än i exemplen med P- och PI-reglering, dvs ett fåtal gånger per hålltid, kan regulatorn beräknas från en tidsdiskret modell. En smärre nackdel med denna reglering är att vi inför en tidsfördröjning på upp till ett samplingsintervall från det att en störning uppträder på utgången, till det att regulatorn börjar motverka störningen. Denna fördröjning skall läggas till den tid som krävs för själva mätningen (Båda dessa tidsfördröjningar bör minimeras).

Om samplingsintervallet är så stort att det är i samma storleksordning som hålltiden, får vi en urartad reglering, i den meningen att regulatorn inte kan motverka störningar med full snabbhet. Regleringen får då inriktas på att eliminera inverkan av parameteravvikelse och långsamt varierande störningar.

4.2.5 Åtgärder vid temperaturökning - väljare

Det väsentligaste kravet på frysaren är att produktens utgångstemperatur inte får vara för hög. Regulatorn bör därför på något sätt prioritera detta krav. Ett enkelt exempel på sådan reglering kan vara, att en alternativ styrlag införs då reglerfelet överskrider en viss nivå (reglering med 'väljare').

I fig 4.9 visas resultatet om insignalen till PI-regulatorn fördubblas då produkttemperaturen överskrider referensvärdet med en halv grad. För jämförelsens skull visas resultatet då processen utsätts för samma störning, men med omvänt tecken. Exemplet är grovt tillyxat, men visar principen: det

maximala felet blir mindre vid temperaturhöjning, och felet går snabbare ner till en acceptabel nivå.

4.2.6 Framkoppling

De störningar som är mätbara kan motverkas genom framkoppling. Om dynamiken för störningens respektive styrsignalens inverkan på utsignalen är känd, kan det med en dynamisk framkoppling från störning till styrsignal i det ideala fallet uppnås att störningen överhuvudtaget inte slår igenom på utsignalen. I praktiken kan t ex parameteravvikelser, olinjäriteter samt begränsningar hos styrsignalen medföra avvikelser, vilka måste korrigeras med hjälp av återkoppling.

Om vi använder bandhastigheten som styrsignal, och antar att det system vi skall reglera utgörs av produkten på bandet, kommer de väsentliga störningarna att innebära variationer hos:

- *) Lufttemperaturen
- *) Värmeövergångstalet
- *) Produktens ingångstemperatur

Lufttemperaturerna i frysaren är relativt enkla att mäta. Med givare före och efter stacken (och lämpliga åtgärder för att eliminera mätstörningar p g a läckage m m) kan medeltemperaturen i stacken uppskattas, och framkoppling göras från denna.

Värmeövergångstalet varierar med lufthastigheten, vilken är svår att mäta. Däremot kan tillfälliga bandtäckningsstörningar, vilka påverkar värmeövergången, mätas och effekterna på produkttemperaturen därmed motverkas. Detta förutsätter att störningen inte längre påverkar själva styrningen om denna förutsätter reglering även av bandtäckningen.

Förändringar av produktens ingångstemperatur bör om möjligt kompenseras med motsvarande stegändring på utgången, efter en hålltids fördröjning. Endast störningar, som innebär en ökning av temperaturen, bör motverkas med framkoppling, eftersom en perfekt reproduktion av stegändringen är svår att uppnå med reglering.

Hur en enkel proportionell framkoppling från lufttemperaturerna vid frysarens in- och utgång kan motverka störningar på lufttemperaturen visas i fig 4.10a.

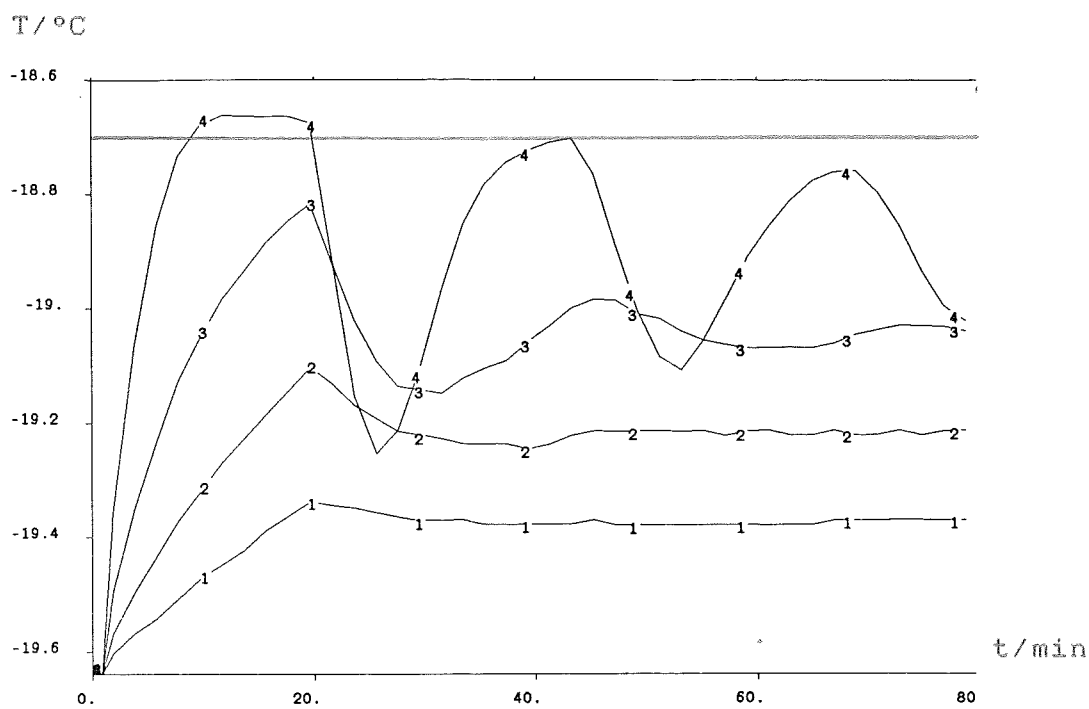
Framkoppling från mätbara störningar är ett effektivt sätt att reglera infrysningsprocessen. Dock bör regulatorn innehålla en återkopplingsdel, för att garantera att utgångstemperaturen inte får ett stationärt fel på grund av t ex parameteravvikelser. Känsligheten för avvikelser mellan modell och verklighet är nämligen avsevärt mindre vid återkoppling än vid framkoppling, som illustreras i fig 4.10b. Med PI-reglering konvergerar alltid temperaturen mot det önskade värdet.

4.2.7 Sammanfattning

Med utgångspunkt från den modell som framtagits kan följande åtgärder för styrning av frysaren föreslås:

*) Utan regulator: Genom mätningar och beräkningar optimera frysarens stationära inställning. Mätningar av variationer hos produktens utgångstemperatur - infrysning med marginal, så att temperaturen i medeltal blir kallare än -18° , och som varmest antar denna temperatur.

*) Med regulator: Genom mätningar kartlägga temperaturvariationer och deras snabbhet, och därefter koppla in tillräckligt snabb regulator (i synnerhet med avseende på motverkande av större temperaturökningar). Mätbara störningar motverkas med framkoppling. Om vissa störningar är för snabba för regulatorn, läggs referensvärdet med marginal i förhållande till -18° . Eventuellt kan optimering av regulatorinställningen med avseende på kyleffekten byggas in i regulatorn (t ex optimalsökande reglering, parameterstyrning).



holdtime/min

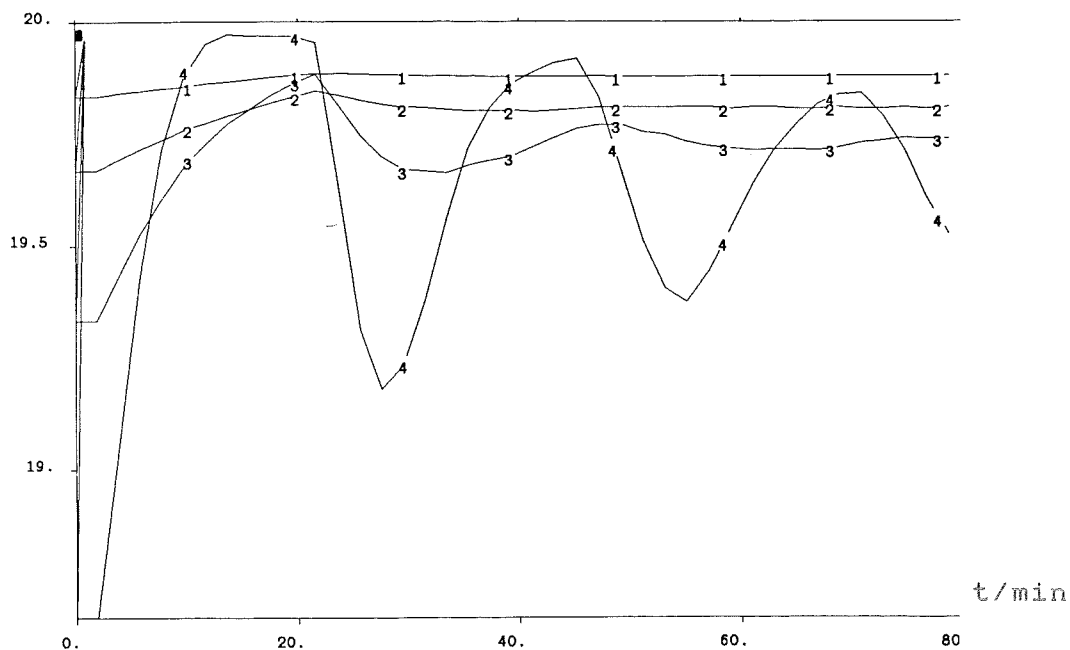
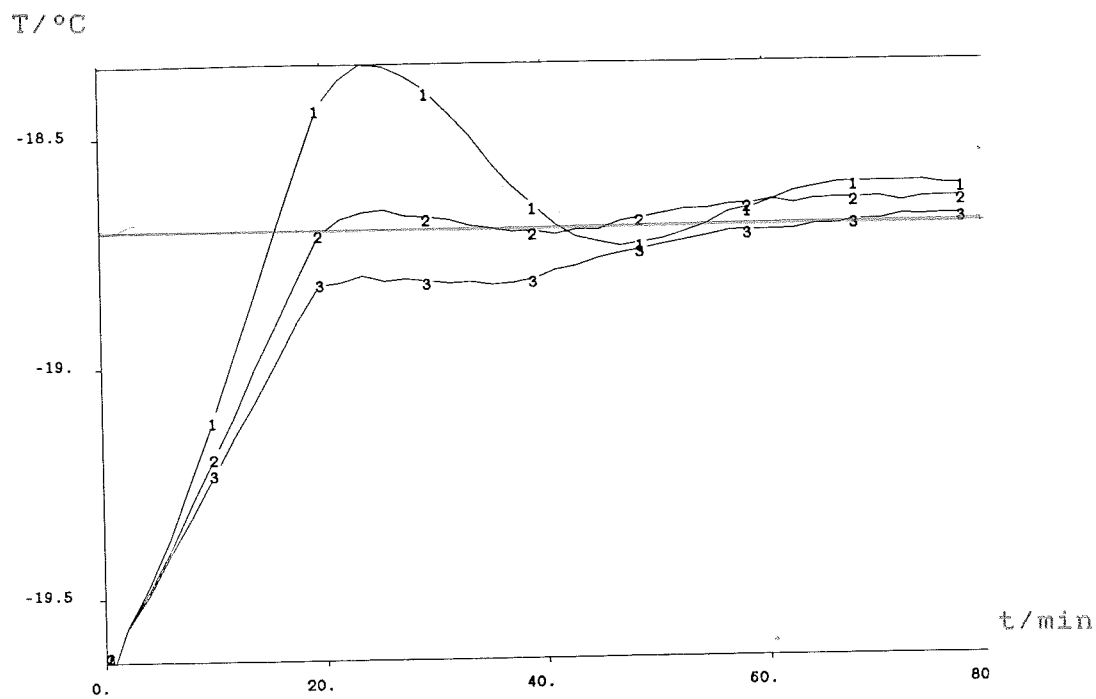


Fig 4.6. P-regulator: utsignal = produkttemperatur (övre diagr.) och styrsignal = 'hålltid' (se texten) (undre diagr.)

efter stegändring av referenstemperaturen från -19.7° till -18.7° .

$K = (1) 100 (2) 200 (3) 400 (4) 800$



holdtime/min

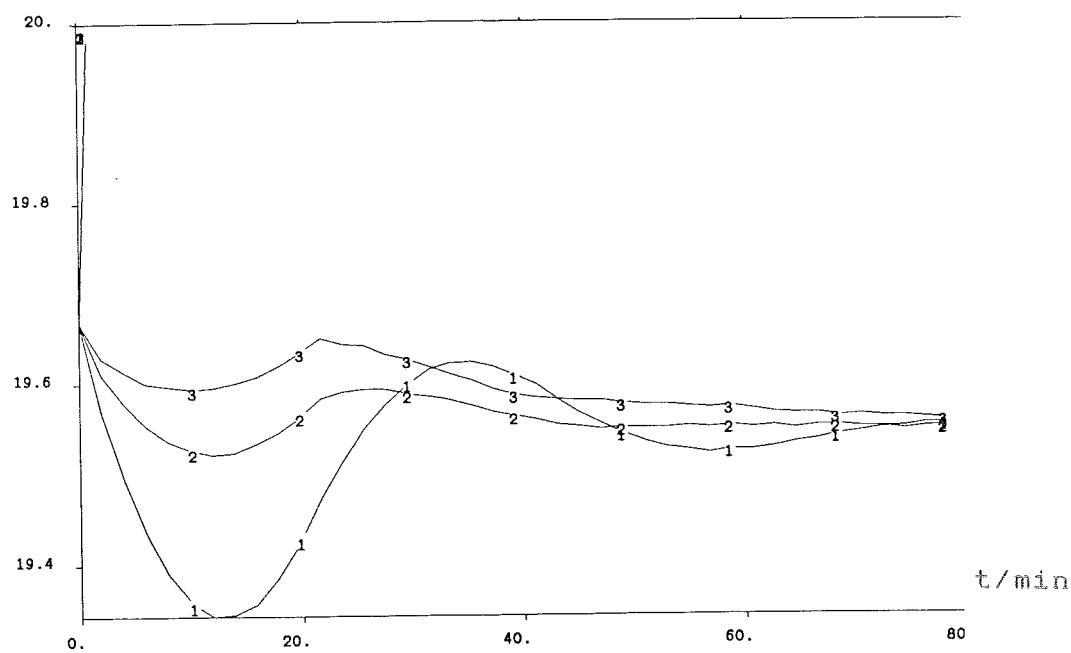
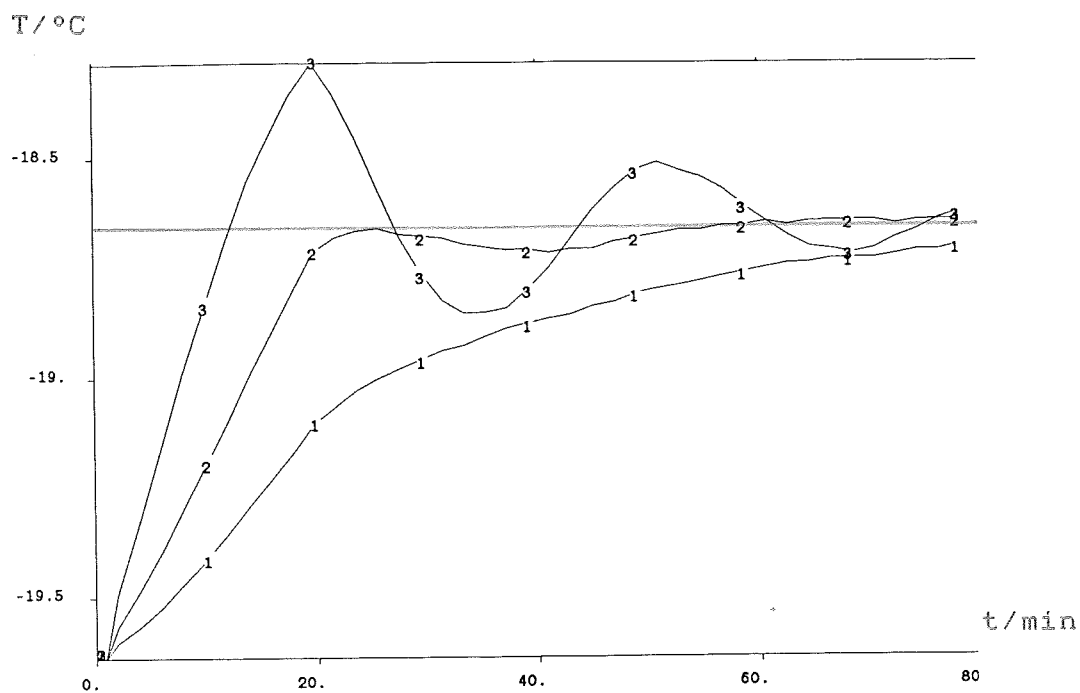


Fig 4.7a. PI-regulator: utsignal = produkttemperatur (övre diagr.) och styrsignal = 'hålltid' (se texten) (undre diagr.) efter stegändring av referenstemperaturen från -19.7° till -18.7° .
 $K = 200$, $T_i = (1) 400 \text{ s}$ (2) 700 s (3) 1000 s



holdtime/min

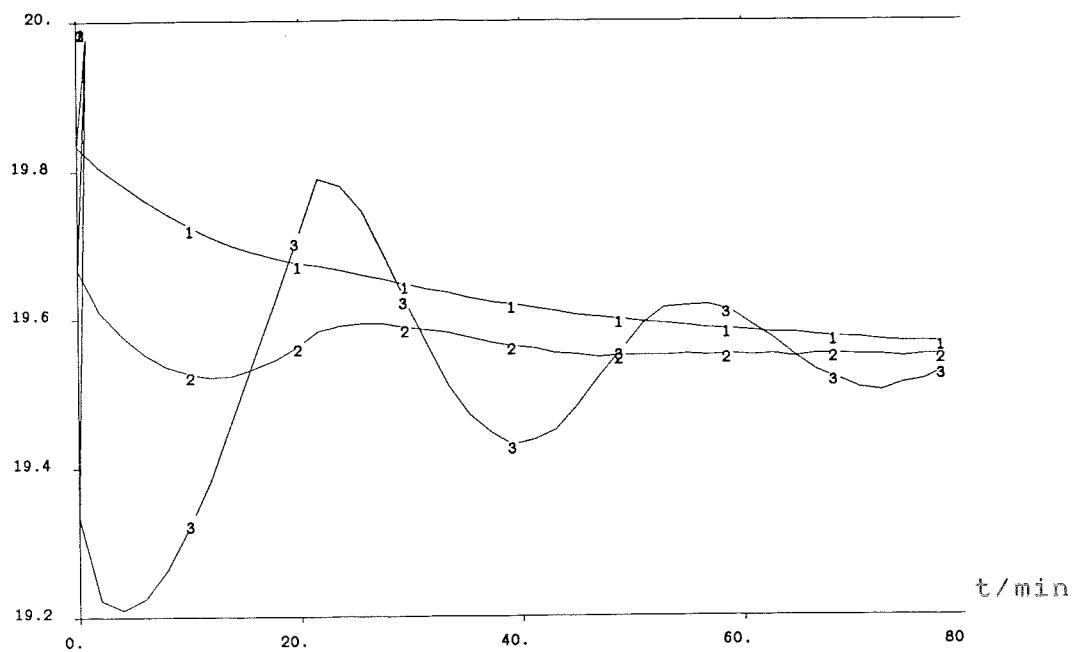
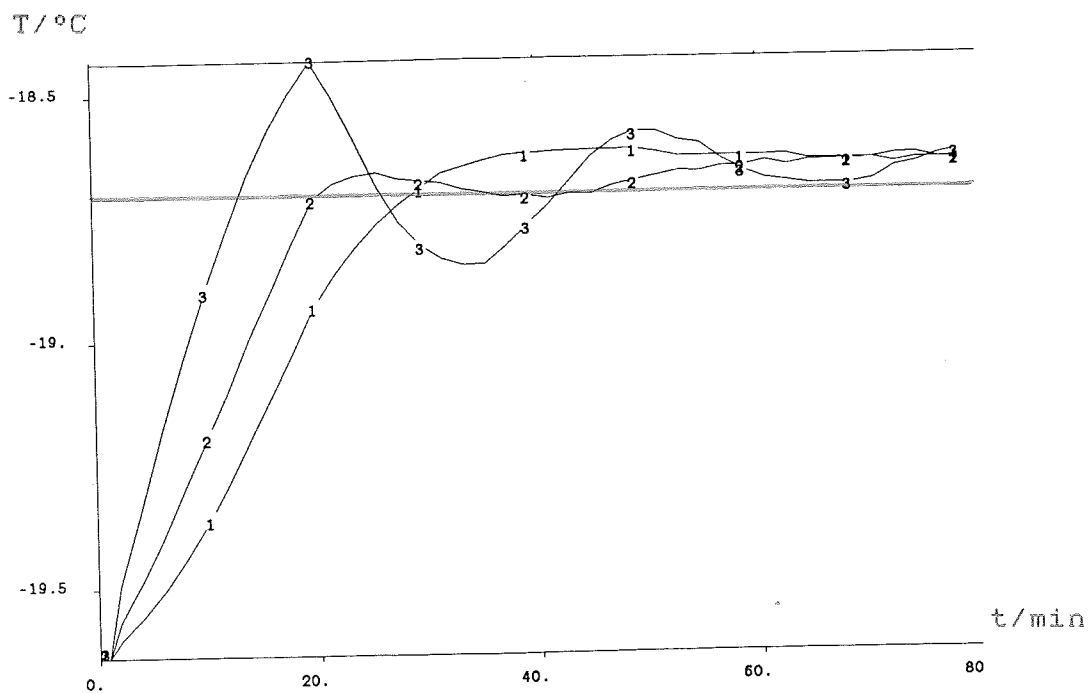


Fig 4.7b. PI-regulator: utsignal = produkttemperatur (övre diagr.) och styrsignal = 'hålltid' (se texten) (undre diagr.)

efter stegändring av referenstemperaturen från -19.7° till -18.7° .

$T_i = 700$ s, $K = (1) 100 (2) 200 (3) 400$



holdtime/min

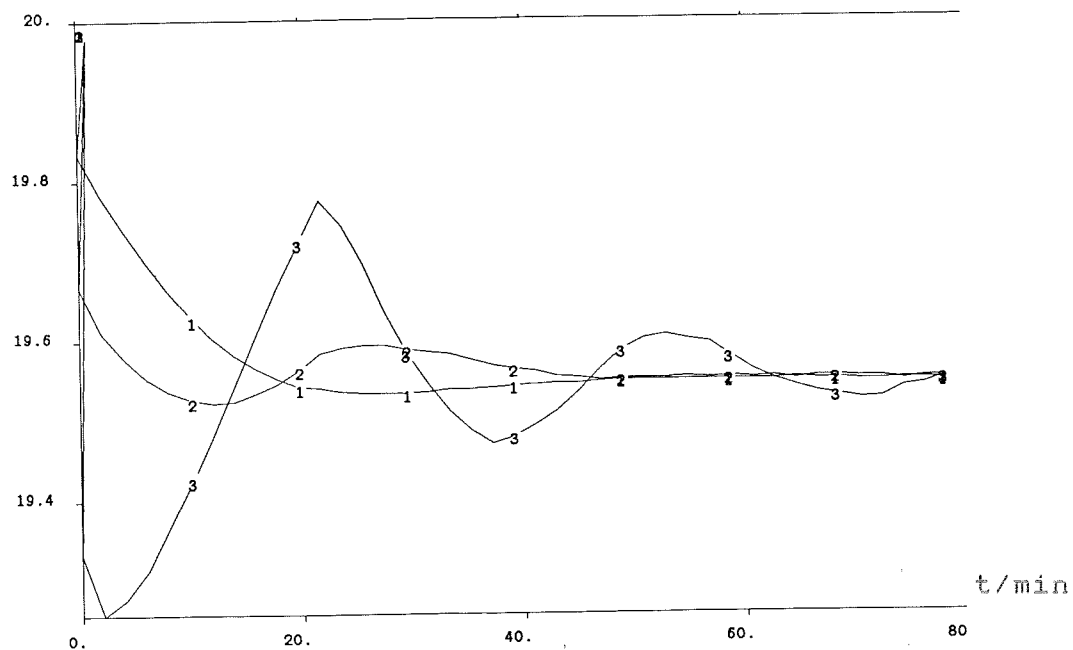


Fig 4.7c. PI-regulator: utsignal = produkttemperatur (övre diagr.) och styrsignal = 'hålltid' (se texten) (undre diagr.)
 efter stegändring av referenstemperaturen från -19.7° till -18.7° .
 (1) $K = 100$, $T_i = 400$ s (2) $K = 200$, $T_i = 700$ s
 (3) $K = 400$, $T_i = 1000$ s

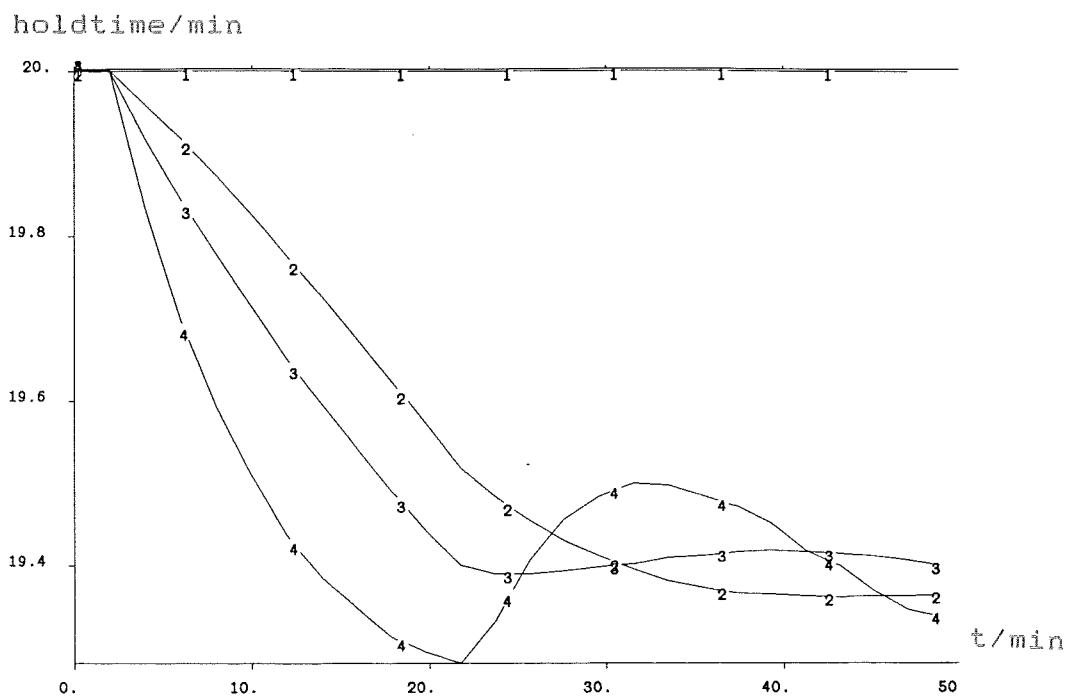
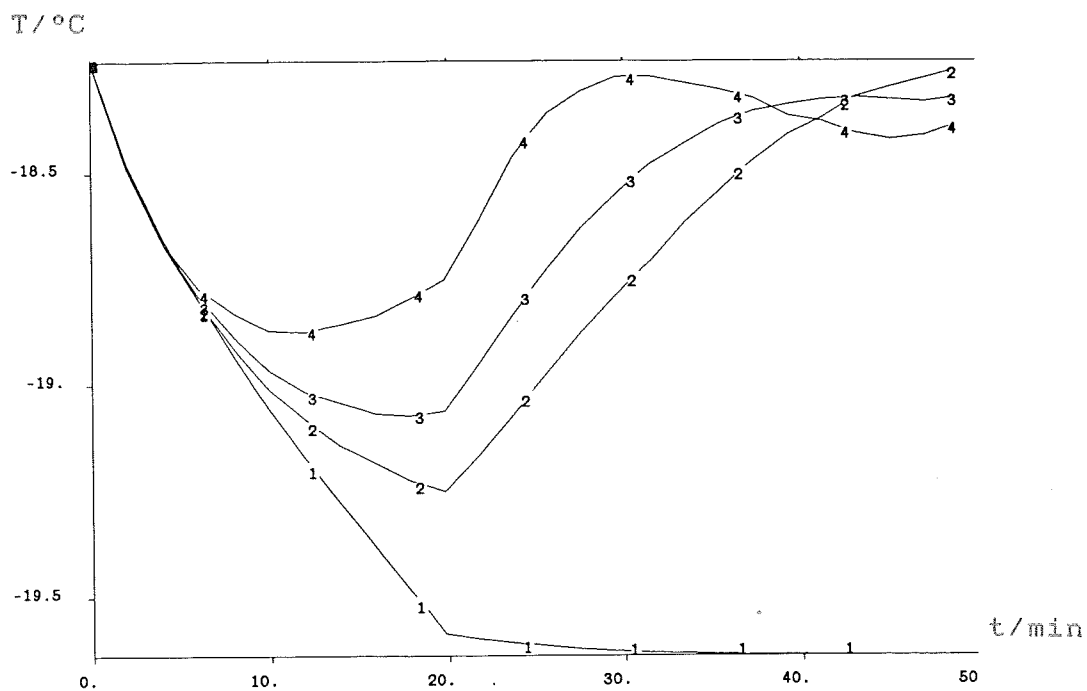


Fig 4.8. PI-regulator: utsignal = produkttemperatur (övre diagr.) och styrsignal = 'hålltid' (se texten) (undre diagr.) efter stegändring av T_{evap} från -39° till -40° .
 (1) ingen reg. (2) $K = 100, T_i = 400\text{ s}$
 (3) $K = 200, T_i = 700\text{ s}$ (4) $K = 400, T_i = 1000\text{ s}$

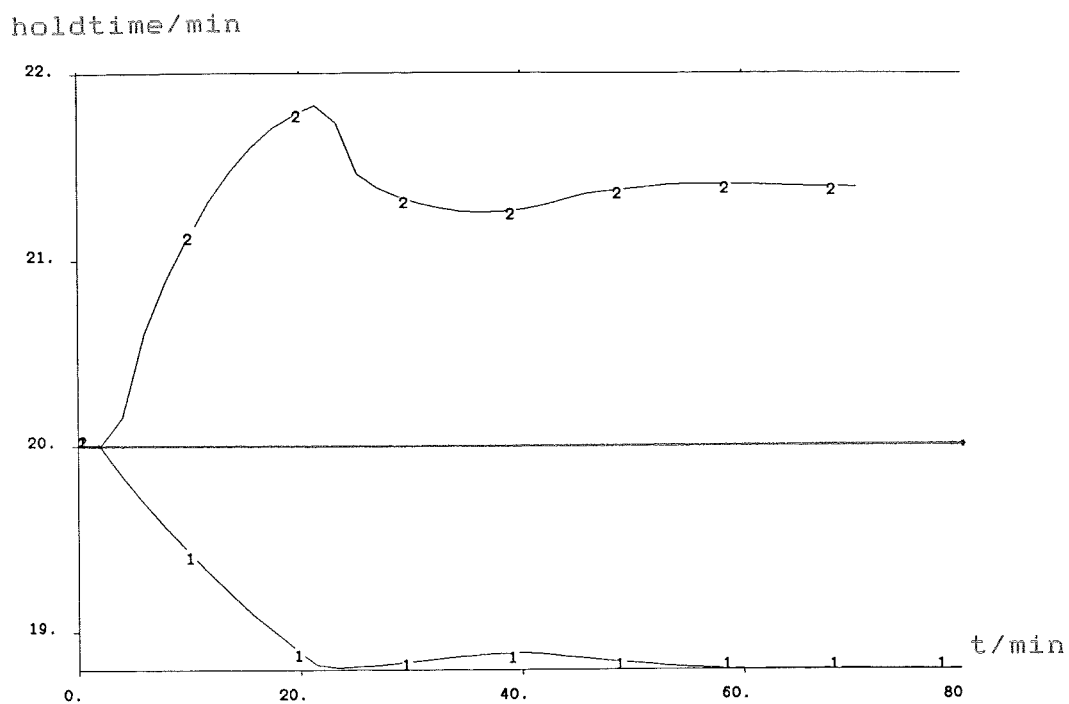
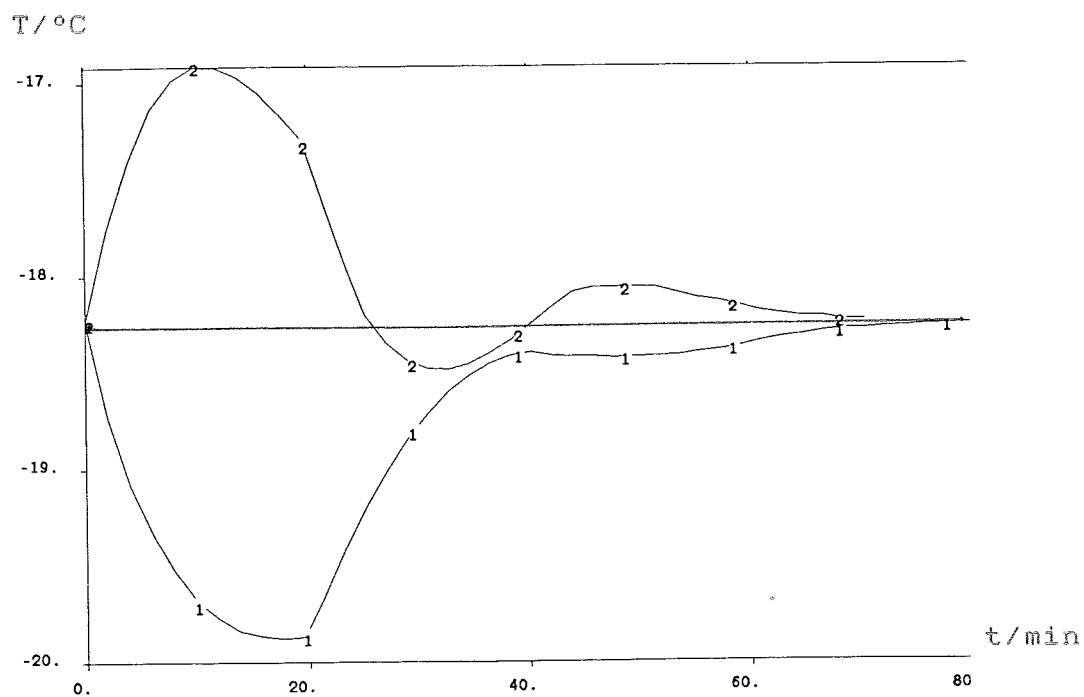


Fig 4.9. Undertryckande av temperaturhöjningar: utsignal = produkttemperatur (övre diagr.) och styrsignal = 'hålltid' (se texten) (undre diagr.)

efter stegändring av T_{evap} från -39° till

(1) -41° (2) -37° ($K = 200$, $T_i = 700$ s)

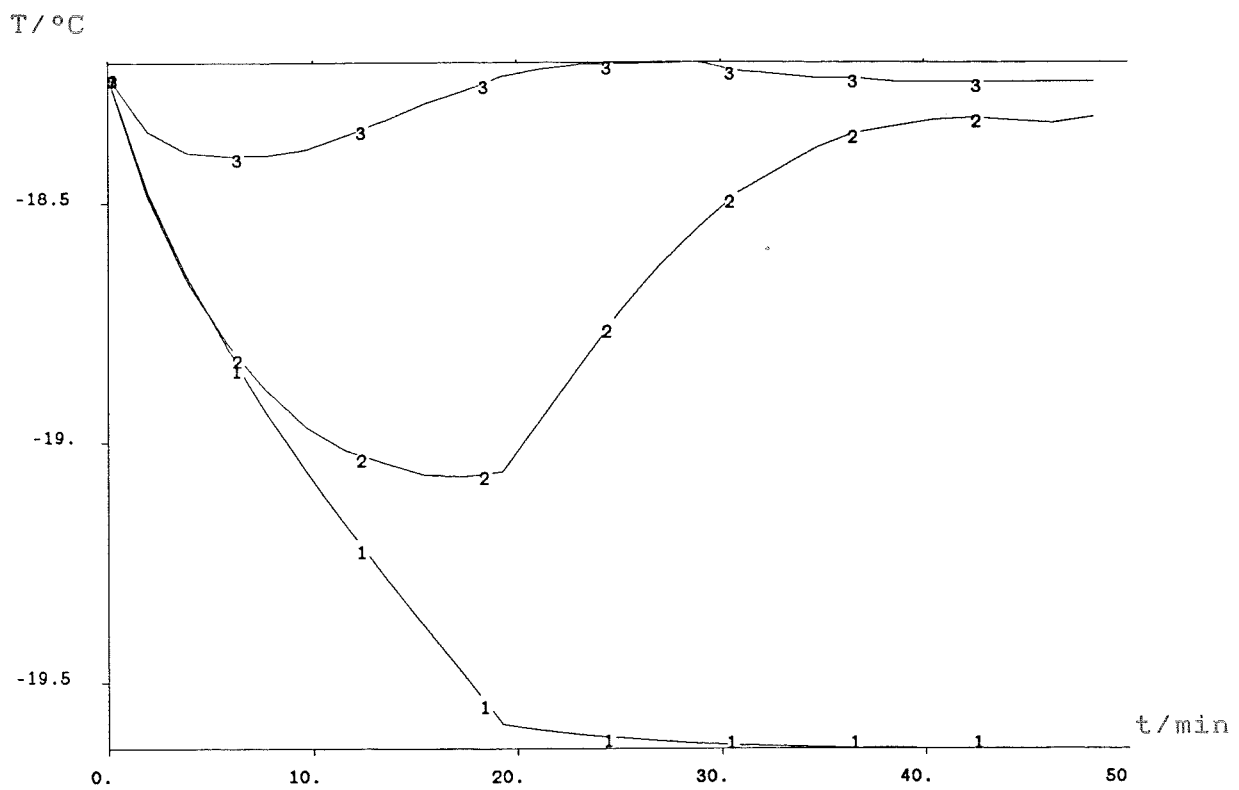


Fig 4.10a. Framkoppling: utsignal = produkttemperatur vid stegändring av T_{evap} från -39° till -40° .
 (1) ingen reg (2) PI-reg (3) PI-reg med framkoppling

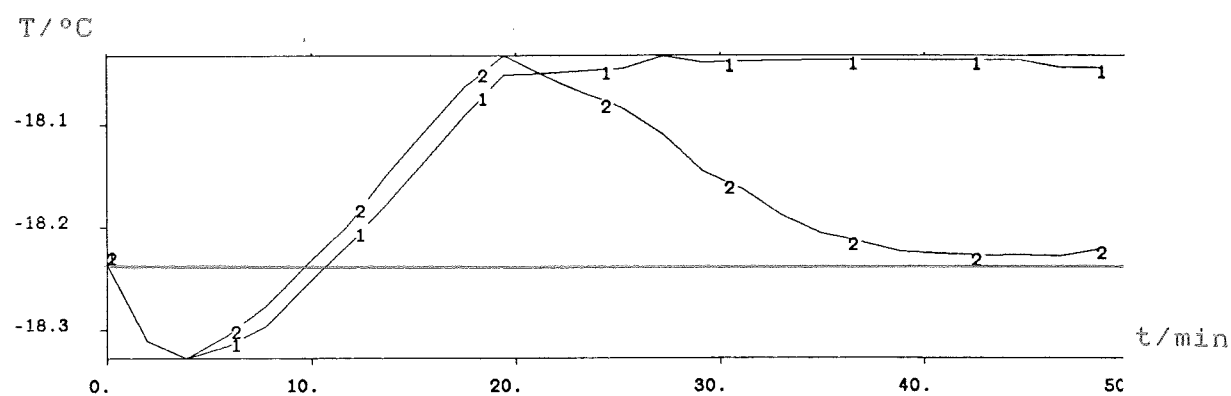
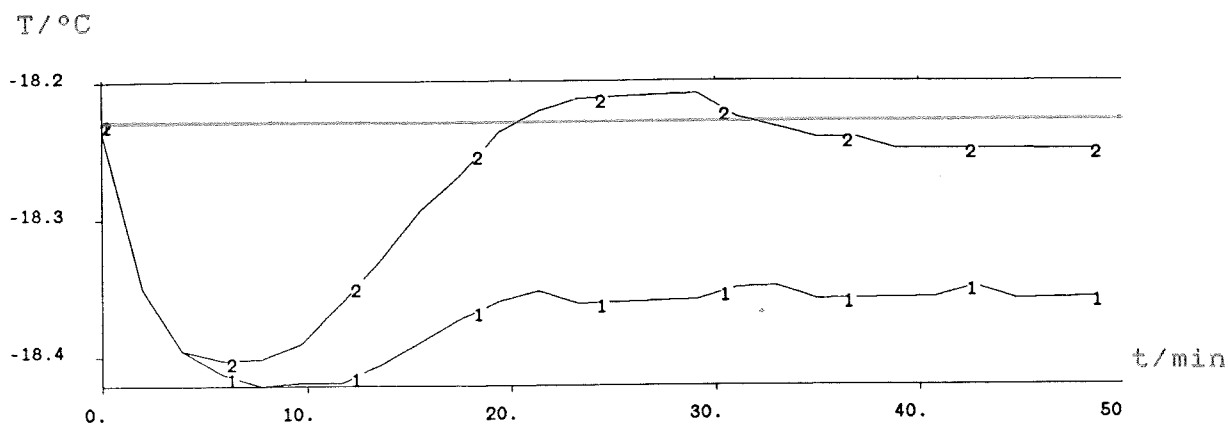


Fig 4.10b. Framkoppling: utsignal = produkttemperatur vid stegändring av

T_{evap} från -39° till -40° .

Övre diagr: för svag framkoppling ($K_{\text{airout}} = K_{\text{airin}} = 200$)

Undre diagr: för stark framkoppling ($K_{\text{airout}} = K_{\text{airin}} = 250$)

(1) endast framkoppling (2) PI-reg med framkoppling

5 SAMMANFATTNING

Vid industriell infrysning eftersträvas att det frysta livsmedlet antar en viss föreskriven temperatur (oftast -18°). Används en kontinuerlig frysare, ingående i en produktionskedja, finns dessutom kravet att frysaren ska arbeta med så hög och jämn takt som möjligt. För att bättre kunna följa dessa krav, då processen utsätts för störningar, vore det lämpligt att införa kontinuerlig temperaturmätning för reglering av frysaren under drift.

Syftet med detta examensarbete har varit att uppställa en matematisk modell för infrysningsprocessen, samt genom simulering på dator undersöka dynamik och möjligheter till reglering. Resultatet utgörs främst av det Pascalprogram som framtagits för simulering av spiralfrysaren Gyrofreeze, samt de förslag till praktiska åtgärder, som ges i kap 4.2.7.

Arbetet kan indelas i tre moment, motsvarande kap 2,3 och 4:

I kap 2 utarbetas en numerisk metod för beskrivning av infrysningsförloppet i produkten (livsmedlet). Den så kallade direktmetoden för lösning av värmeledningsekvationen har omarbetats för att inbegripa både värmeledning och fasövergång. Värmeledningstalet och värmekapacitiviteten, vilka ingår i ekvationens koefficienter, har satts som funktioner av temperaturen. Ett infrysningsdiagram, framtaget med denna metod, visas nedan i fig 5.1.

Kap 3 innehåller en sammanställning av de faktorer, som påverkar miljön i frysaren. Värmeövergångstalet (produkt - luft) och lufttemperaturerna bestäms av mängden produkt i frysaren, kylbatteriets kyleffekt, luftens strömningshastighet, osv. Valda samband ingår, tillsammans med den framtagna lösningsmetoden för värmeledningsekvationen, i program Gyrofreeze. Vid användande av detta program kan man välja, att studera produkttemperaturen vid utgången från frysaren (kap 4), eller att låta frysarmiljön vara förutbestämd och då erhålla ett fullständigt infrysningsdiagram (fig 5.1).

I kap 4 studeras dynamiken hos processen genom simulering. Stegsvvar har framtagits för olika variabeländringar. Reglering av frysaren genom variation av produktens transporthastighet (dvs varierande hålltid) har

simulerats. Som åtgärd föreslås införande av PI-reglering kombinerad med framkoppling från mätbara störningar (främst på lufttemperaturen). Exempel på hur regulatorn motverkar störningar visas i figur 4.10.

Examensarbetet har utförts i hög grad självständigt, dvs som ett konsultuppdrag men utsträckt över en 'oprofessionellt' lång tidsrymd. Jag vill tacka Jan Johansson på Frigoscandia, för hjälp med framtagande av grundmaterial till arbetet, och Björn Wittenmark på Reglerteknik, för ett oändligt tålamod.

Lund 1985

Ulf Adamsson

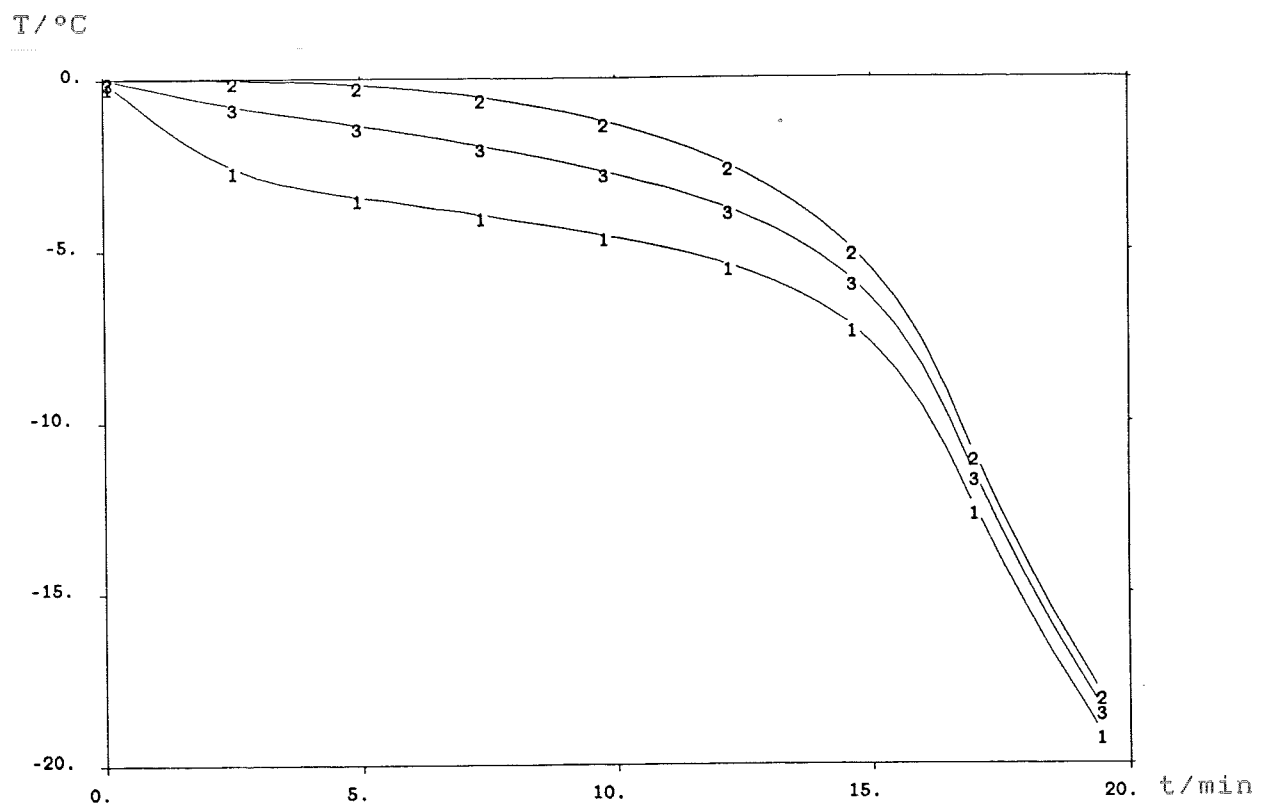


Fig 5.1 Infrysningsdiagram. Exempel: hamburgare 1 cm tjock.

(1) Yttemperatur ,(2) Kärntemperatur ,(3) Medeltemperatur.

6 LITTERATUR

MODELLBYGGE

H Nicholson: Modelling of Dynamical Systems, Vol 1
(Institution of Electrical Engineers, 1980)

NUMERISK ANALYS

G Eriksson: Numerisk Analys FK
(Sigmatryck, 1979)

G D Smith: Numerical Solution
of Partial Differential Equations
(Oxford University Press, 1965)

PROGRAMMERING

T Ekman, J Karlsson: Pascal för dig som kan programmera
(Studentlitteratur, 1981)

REGLERTEKNIK

K J Aström: Reglerteori
(Almqvist & Wiksell, 1976)

7 BILAGOR

A Program Gyrofreeze (listning)

B Frysar- och produktdata (Frigoscandia)

```

(*****)
Program GyroFreeze
  (input, output, parameterfile, statefile, reservefile,
   backup, storefile, printfile);

(*****)

const

(* PLOTTING LIMIT *)
  horizontal= 72;

(* ARRAY LIMITS *)
  mmax=      10;
  nmax=      20;

(* FREEZER *)
  Astack=    2.50;
  Abat=      2.14;
  Abelst=    43;
  baseload=23000;

(* AIR *)
  cpair=     1000;
  airdens=   1.293;

(* PRODUCT *)
  dens=      980;
  areafactor= 2.0;

type

  vector= array[-1..nmax] of real;
  realfile= file of real;

var

(* PARAMETERS *)
  delta, D, Tevap, Wfrost, plotmin, plotmax: real;
  m, holdtime, storeint, deltat, stopint: integer;
  n: integer;

(* REGULATOR PARAMETERS *)
  K, Ti, Tref, Int: real;
  Kairin, Kairout: real;
  Tairinref, Tairoutref: real;

(* STATES *)
  Tlevel, Wstack, alpha: real;
  T: array[0..mmax+1] of vector;
  Tair, bandcover: array[0..mmax+1] of real;
  time, stoptime: integer;
  s: real;

(* FILES *)
  parameterfile, statefile, reservefile, backup: realfile;
  storefile, printfile: text;

(* AUXILIARY VARIABLES *)
  value: real;

```

```

storeTlevel, storeT, storeTair, storeholdtime, stopcheck: boolean;
oneproduct: boolean;
i, j: integer;
choice: char;

```

```

(*****

```

```

Procedure Copy(var infile,outfile:realfile);

```

```

begin
  reset(infile);
  rewrite(outfile);
  while not eof(infile) do
    begin
      outfile↑:= infile↑;
      put(outfile);
      get(infile);
    end;
  reset(infile);
end;

```

```

(*****

```

```

Procedure Compute;

```

```

begin
  n:=      round(D/delta) div 2;
  delta:=  D/(2*n);
end;

```

```

(*****

```

```

Procedure GetParameters(var pfile: realfile);

```

```

begin
  if eof(pfile) then begin reset(pfile); writeln('EOF') end;
  if not eof(pfile) then
    begin
      read(pfile,delta  );
      read(pfile,D);
      read(pfile,value  ); m:=      round(value);
      read(pfile,Tevap  );
      read(pfile,Wfrost);
      read(pfile,value  ); holdtime:= round(value);
      read(pfile,value  ); storeint:= round(value);
      read(pfile,value  ); deltat:=  round(value);
      read(pfile,value  ); stopint:=  round(value);
      read(pfile,plotmin );
      read(pfile,plotmax );
      read(pfile,K,Ti,Kairin,Kairout,Tref,Tairinref,Tairoutref);
    end;

```

```

      Compute;
    end;
end;

```

```

(*****

```

```

Procedure WriteParameters(var writefile: text);

```

```

begin
  writeln(writefile,'1: delta=      ',delta:8:4);

```

```

writeln(writefile,'2: D=           ',D:6:2);
writeln(writefile,'3: m=           ',m:3 );
writeln(writefile,'4: Tevap=       ',Tevap:5:1);
writeln(writefile,'5: Wfrost=      ',Wfrost:5:1);
writeln(writefile,'6: holdtime=    ',holdtime/6000:5:1);
writeln(writefile,'7: storeint=    ',storeint/100:5:1);
writeln(writefile,'8: deltat=      ',deltat/100:6:2);
writeln(writefile,'9: stopint=     ',stopint/100:4:0);
writeln(writefile,'a: plotmin=     ',plotmin:6:2);
writeln(writefile,'b: plotmax=     ',plotmax:6:2);
writeln(writefile,'c: K=           ',K      :10:2);
writeln(writefile,'d: Ti=          ',Ti      :10:2);
writeln(writefile,'f: Kairin=      ',Kairin :10:2);
writeln(writefile,'g: Kairout=     ',Kairout:10:2);
writeln(writefile,'h: Tref=        ',Tref   :10:2);
writeln(writefile,'i: Tairinref=   ',Tairinref:10:2);
writeln(writefile,'j: Tairoutref=  ',Tairoutref:10:2);
end;

(*****

Procedure ChangeParameters;

var choice: char;

begin
  repeat
    writeln('Change parameters');
    WriteParameters(output);
    readln(choice);
    case choice of
      '1': readln(delta );
      '2': readln(D);
      '3': readln(m );
      '4': readln(Tevap );
      '5': readln(Wfrost );
      '6': begin readln(value); holdtime:= round(value*6000) end;
      '7': begin readln(value); storeint:= round(value*100) end;
      '8': begin readln(value); deltat:= round(value*100) end;
      '9': begin readln(value); stopint:= round(value*100) end;
      'a': readln(plotmin);
      'b': readln(plotmax);
      'c': readln(K );
      'd': readln(Ti );
      'f': readln(Kairin );
      'g': readln(Kairout);
      'h': readln(Tref);
      'i': readln(Tairinref);
      'j': readln(Tairoutref);

      'e': (* EXIT *)
    end;
  until choice='e';

  Compute;
end;

(*****

Procedure SaveParameters(var pfile: realfile);

```

```

begin
  write(pfile,delta      );
  write(pfile,D);
  value:= m             ; write(pfile,value);
  write(pfile,Tevap,Wfrost );
  value:= holdtime ; write(pfile,value);
  value:= storeint ; write(pfile,value);
  value:= deltat      ; write(pfile,value);
  value:= stopint    ; write(pfile,value);
  write(pfile,plotmin );
  write(pfile,plotmax );
  write(pfile,K,Ti,Kairin,Kairout,Tref,Tairinref,Tairoutref);

  writeln('parameters saved ');
end;

(*****

Procedure Parameters;

var
  choice: char;
  buffertfile: realfile;

begin
  Copy(parameterfile,buffertfile);
  repeat
    writeln('Parameters:');
    WriteParameters(output);
    writeln('choice:');
    writeln('g: get; x: get(reserve); p: print; c: change;');
    writeln('s: save; d: delete buffert; e: exit');
    readln(choice);
    case choice of
      'g': GetParameters(parameterfile);
      'x': GetParameters(reservefile);
      'p': WriteParameters(printfile);
      'c': ChangeParameters;
      's': SaveParameters(buffertfile);
      'd': (* DELETE BUFFERT *) rewrite(buffertfile);
      'e': (* EXIT *) ;
    end;
  until choice='e' ;

  Copy(buffertfile,parameterfile);
end;

(*****

Procedure GetStates(var sfile: realfile);

var
  i,j: integer;

begin
  if eof(sfile) then begin reset(sfile); writeln('EOF') end;
  if not eof(sfile) then
    begin
      read(sfile,Tlevel,Wstack,alpha);
      for i:= 0 to mmax+1 do
        begin

```

```

        for j:= -1 to nmax do read(sfile,T[i,j]);
        read(sfile,bandcover[i],Tair[i]);
    endi;
    read(sfile,value); time:= round(value);
    read(sfile,value); stoptime:= round(value);
    read(sfile,s);
end;
end;

(*****)

Procedure WriteStates(var writefile: text);

var
    i,j: integer;

begin

    writeln(writefile,'Tlevel: ',Tlevel:6:2);
    writeln(writefile,'Wstck: ',Wstack:6:2);
    writeln(writefile,'alpha: ',alpha: 6:2);

    writeln(writefile,'1: a: T=');
    for j:= -1 to n do
    begin
        for i:= 0 to m+1 do write(writefile,T[i,j]:6:2);
        writeln(writefile);
    end;

    writeln(writefile,'2: bandcover=');
    for i:= 0 to m do
        write(writefile,bandcover[i]:6:2);
    writeln(writefile);

    writeln(writefile,'3: Tair=');
    for i:= 0 to m do write(writefile,Tair[i]:6:2);
    writeln(writefile);

    writeln(writefile,'4: time= ',time div 6000,' min',
        (time mod 6000)/100:6:2,' sek');

    writeln(writefile,'5: stoptime= ',stoptime div 6000,' min',
        (stoptime mod 6000)/100:6:2,' sek');

    writeln(writefile,'6: s= ',s:5:3 );

end;

(*****)

Procedure ChangeStates;

var
    choice: char;
    i,j: integer;

begin
    repeat
        writeln('Change states');
        WriteStates(output);
    until choice = 'q';
end;

```

```

    readln(choice);
  case choice of
    '1': begin
      readln(value);
      for j:= -1 to n do T[0,j]:= value;
    end;
    'a': begin
      readln(value);
      for i:= 0 to m+1 do
        for j:= -1 to n do T[i,j]:= value;
      end;
    '2': readln(bandcover[0]);
    '3': for i:= 0 to m do readln(Tair[i]);
    '4': begin readln(value); time:= round(value*6000) end;
    '5': begin readln(value); stoptime:= round(value*6000) end;
    '6': readln(s);

    'e': (* EXIT *);
  end;
until choice='e';
end;

(*****)

Procedure SaveStates(var sfile: realfile);

var
  i, j: integer;

begin
  write(sfile,Tlevel,Wstack,alpha);
  for i:= 0 to mmax+1 do
    begin
      for j:= -1 to nmax do write(sfile,T[i,j]);
      write(sfile,bandcover[i],Tair[i]);
    end;
    value:= time ; write(sfile,value);
    value:= stoptime ; write(sfile,value);
    write(sfile,s);

    writeln('states saved');
  end;

(*****)

Procedure States;

var
  buffertfile: realfile;
  choice: char;

begin
  Copy(statefile,buffertfile);
  repeat
    writeln('States');
    WriteStates(output);
    writeln('choice:');
    writeln('g: get; x: get(reserve); p: print; c: change');
    writeln('s: save; d: delete buffert; e: exit');
    readln(choice);
  case choice of

```



```

    'g': GetStates(statefile);
    'x': GetStates(reservefile);
    'p': WriteStates(printfile);
    'c': ChangeStates;
    's': SaveStates(buffertfile);
    'd': (* DELETE BUFFERT *) rewrite(buffertfile);
    'e': (* EXIT *) ;
  end;
until choice='e' ;

  Copy(buffertfile,statefile);
end;

(*****)

Procedure Init;

begin
  reset(parameterfile);
  reset(statefile);
  reset(reservefile);
  reset(backup);
  rewrite(storefile);
  rewrite(printfile);
  GetParameters(parameterfile);
  Getstates(statefile);
end;

(*****)

Procedure Choose;

var choice: char;

begin
  repeat
    writeln('Choose');

    write('1: Tlevel      ');
    if storeTlevel      then write('store'); writeln;
    write('2: T          ');
    if storeT          then write('store'); writeln;
    write('3: Tair         ');
    if storeTair         then write('store'); writeln;
    write('4: holdtime      ');
    if storeholdtime    then write('store'); writeln;
    write('5: stopcheck     ');
    if stopcheck        then write('yes  '); writeln;
    write('6: oneproduct   ');
    if oneproduct       then write('yes  '); writeln;

    readln(choice);
  case choice of
    '1': storeTlevel:= not storeTlevel;
    '2': storeT:= not storeT;
    '3': storeTair:= not storeTair;
    '4': storeholdtime:= not storeholdtime;
    '5': stopcheck:= not stopcheck;
    '6': oneproduct:=not oneproduct;
    'm': (* MENU *) ;
  end;
end;

```

```

        'e': (* EXIT *) ;
    end;
until choice='e';
end;

(*****)

Procedure Run;

var

(* AUXILIARY PARAMETERS *)
    beta, Wbat, range: real;
    products: integer;

(* AUXILIARY VARIABLES *)
    gamma: real;
    samplecount: integer; lasttime :real;
    location, storetime, shifttime, shiftint: integer;
    cp, lambda, mlambda, nT: vector;

    started, stop: boolean;

(* REGULATOR VARIABLE *)
    e:real;

(*****)

Procedure StoreAndPlot(value:real);

var
    j: integer;

begin

(* STORE *)
    writeln(storefile,time/6000);
    if storeTlevel then writeln(storefile,Tlevel);

    if oneproduct then
    begin
        if storeT then writeln(storefile,T[1,0],T[1,n]);
        if storeTair then writeln(storefile,Tair[0]*s+Tair[1]*(1-s));
    end
    else
    begin
        if storeT then writeln(storefile,T[m+1,0],T[m+1,n]);
        if storeTair then writeln(storefile,Tair[0],Tair[m]);
        if storeholdtime then writeln(storefile,shiftint*m/6000);
    end;

(* PLOT *)
    (* ADJUST LIMITS *)
    while value>plotmax do
    begin
        plotmax:= plotmax+range/2;
        plotmin:= plotmin+range/2;
        writeln('plotmax= ',plotmax,'plotmin= ',plotmin);
    end;
    while value<plotmin do
    begin

```

```

    plotmax:= plotmax-range/2;
    plotmin:= plotmin-range/2;
    writeln('plotmax= ',plotmax,'plotmin= ',plotmin);
end;

(* ADJUST VALUE *)
value:= (value-plotmin)*horizontal/range;

(* PLOT TIME AND VALUE *)
write(time/6000:6:2);
j:= 0;
while j<value do begin write(' '); j:= j+1 end;
writeln('$');
end;

(*****)

Procedure Shift;

var

(* AUXILIARY VARIABLES *)
q, A, Tcheck, cpairflow, bcsum: real;

(* REGULATOR VARIABLES *)
u, F: real;

i, j: integer;

begin
  if oneproduct then
    (* ONEPRODUCT *)
    begin
      location:= location+1;
      Tair[0]:= Tair[1];
      Tair[1]:= Tair[location];
      if location>m then stop:= true;
    end
  else
    begin

      (* AIRFLOW *)
      bcsum:= 0;
      for i:= 1 to m do
        bcsum:= bcsum+bandcover[i];
      Wstack:= ( (2-9.2*bcsum/m)*bcsum/m+8.1 )/Astack-Wfrost;
      Wbat:= Wstack*Astack/Abat;
      alpha:= 20*Wstack**0.8;

      (* AIRTEMPERATURES *)
      repeat
        Tcheck:= Tair[0];
        cpairflow:= cpair*airdens*Wstack*Astack;
        Tair[m]:= Tair[0] -255.0/(cpair*airdens)
          *((Tair[0]+273.15)/Wbat)**0.34 *(Tair[0]-Tevap-1.3)
          +baseload/cpairflow;
        for i:= m-1 downto 0 do
          begin
            q:= alpha*(T[i,0]+T[i+1,0]-Tair[i]-Tair[i+1])/2;
            A:= bandcover[i+1]*Abelt*areafactor;
            Tair[i]:= Tair[i+1] +q*A/(m*cpairflow);
          end
        until Tcheck=Tair[0];
      end;
    end;
  end;
end;

```

```

        endi;
    until abs(Tcheck-Tair[0]) < 5E-3;

(* SHIFT STATES *)
Tcheck:= T[m+1,0];
for i:= m+1 downto 1 do
begin
    for j:= -1 to n do
        T[i,j]:= T[i-1,j];
    bandcover[i]:= bandcover[i-1];
    endi;

    if stopcheck and started then
        if abs(Tcheck-T[m+1,0]) < 5E-2 then stop:= true;

(* REGULATOR *)

    Int:= Int +K*shiftint*e/Ti;
    F:= Kairin*(Tairinref-Tair[m])+Kairout*(Tairoutref-Tair[0]);
    e:= Tref-Tlevel;
    u:= K*e +Int +F;
    shiftint:= round(holdtime div m - u);

(* TLEVEL *)
    Tlevel:= ( T[m+1,0] + T[m+1,n] )/2;
    for j:=1 to n-1 do
        Tlevel:= Tlevel + T[m+1,j];
    Tlevel:= Tlevel/n;

    StoreAndPlot(Tlevel);
end;
started:= true;
end;

(******)

(* RUN *)

begin

    (* BACKUP *)
    rewrite(backup);
    SaveParameters(backup);
    SaveStates(backup);

    (* AUXILIARY PARAMETERS *)
    beta:= (deltat/100)/(dens*delta*delta);
    range:= plotmax-plotmin;
    if oneproduct then products:= 1
    else
        products:= m;

    (* INITIAL VALUES *)
    location:= 1;
    shiftint:= holdtime div m;
    shifttime:= round(s*shiftint+time);
    storetime:= 0;
    stop:= false;
    started:= false;
    e:= 0;
    Int:= 0;

```

```

(* PROCESS *)
if not oneproduct then StoreAndPlot(Tlevel);

while (time<stoptime) and not stop do
begin

  if oneproduct and (time)= storetime) then
  begin
    (* TLEVEL *)
    Tlevel:= ( T[1,0] + T[1,n] )/2;
    for j:=1 to n-1 do
      Tlevel:= Tlevel + T[1,j];
    Tlevel:= Tlevel/n;

    StoreAndPlot(Tlevel);
    storetime:= storetime+storeinti;
  end;

  if time)=shifttime then
  begin
    Shift;
    shifttime:= shifttime+shiftinti;
  end;

  for i:= 1 to products do
  begin

    (* lambda(T) *)
    j:=-1;
    if T[i,j]<-5 then lambda[j]:=1.45
    else if T[i,j]<0 then lambda[j]:=1.45-(T[i,j]+5)*0.19
    else lambda[j]:=0.50;
    for j:=0 to n do
    begin
      (* cp(T) *)
      if (T[i,j]<-10) or (T[i,j]>0) then cp[j]:=3.3E3
      else cp[j]:=(1.33*exp(0.42*(T[i,j]+10)) +1.97)*1E3;

      (* lambda(T) *)
      if T[i,j]<-5 then lambda[j]:=1.45
      else if T[i,j]<0 then lambda[j]:=1.45-(T[i,j]+5)*0.19
      else lambda[j]:=0.50;

      (* mlambda *)
      mlambda[j]:=(lambda[j-1]+lambda[j])/2;
    end;

    (* TEMPERATURES *)
    for j:=0 to n-1 do
      nT[j]:=T[i,j]+(beta/cp[j])*(mlambda[j]*(T[i,j-1]-T[i,j])
      +mlambda[j+1]*(T[i,j+1]-T[i,j]));
    T[i,n]:=T[i,n]+(beta/cp[n])*2*mlambda[n]*(T[i,n-1]-T[i,n]);
    for j:=0 to n-1 do T[i,j]:=nT[j];

    (* SURFACE HEAT TRANSFER *)
    gamma:= 2*delta*alpha;
    T[i,-1]:= T[i,1] -(gamma/lambda[0])
      *(T[i,0]-Tair[i-1]*s-Tair[i]*(1-s));

  end;
  time:= time+deltat;

```

```

    s:= (shifttime-time)/shiftint;
end;
stoptime:= stoptime+stopint;
end;

```

```

(*****)

```

```

(* GYROFREEZE *)

```

```

begin
  Init;

  repeat
    writeln('GyroFreeze');
    writeln('choice:');
    writeln('p: parameters;      s: states; r: run; i: initialize');
    writeln('b: backup-restart; c: choose; e: exit');
    readln(choice);
    case choice of
      'p': Parameters;
      's': States;
      'r': Run;
      'i': Init;
      'b': (* RESTART FROM BACKUP-FILE *)
            begin GetParameters(backup); GetStates(backup); end;
      'c': Choose;
      'e': (* EXIT *);
    end;
  until choice='e';
end.

```

```

(*****)

```

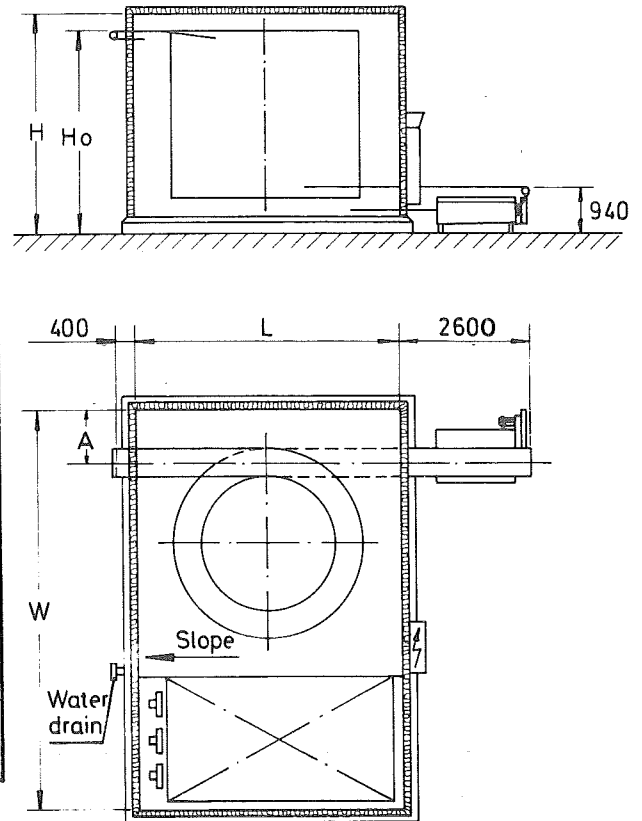
Freezer Division

GYRoFREEZE

Dimensions (Lay-out EW)

SI-System

20", 24" and 28" models (125 mm tier pitch)				
No. of tiers	Ho mm	H mm *		
		20"	24"	28"
14	2830	3300	3400	3500
16	3080	3550	3650	3750
<u>18</u>	3330	3800	3900	4000
20	3580	4050	4150	4250
22	3830	4300	4400	4500
<u>24</u>	4080	4550	4650	4750
26	4330	4800	4900	5000
28	4580	5050	5150	5250
<u>30</u>	4830	5300	5400	5500
32	5080	5550	5650	5750
34	5330	5800	5900	6000
36	5580	6050	6150	6250



Min. internal height		
Belt width inch	Coil surface m ²	H mm
20"	600	3200
	700-800	3450
	900-1000	3750
24"	800	3350
	1000-1100	3600
	1300-1400	3900
28"	1100	3500
	1300-1500	3750
	1800-2000	4050

Handwritten notes:
 3 m
 1
 5
 4
 1
 7
 5
 1
 10

Belt width	L mm	W single vertical mm	W split vertical mm	A mm
20"	5200	6400	7310	1220
24"	5800	7000	8130	1245
28"	6400	7900	9330	1275
32"	7000	9100	10990	1295
36"	7600	10300	12670	1320

H and Ho are based on 300 mm floor thickness, including insulation, sloping 150 mm towards the outfeed end.

Standard enclosure thickness approx. 100 mm.

* H = No. of tiers x tier pitch + one belt width + 1040 (Minimum internal height for certain coil sizes, see table).

Underlined no. of tiers to be first choice.

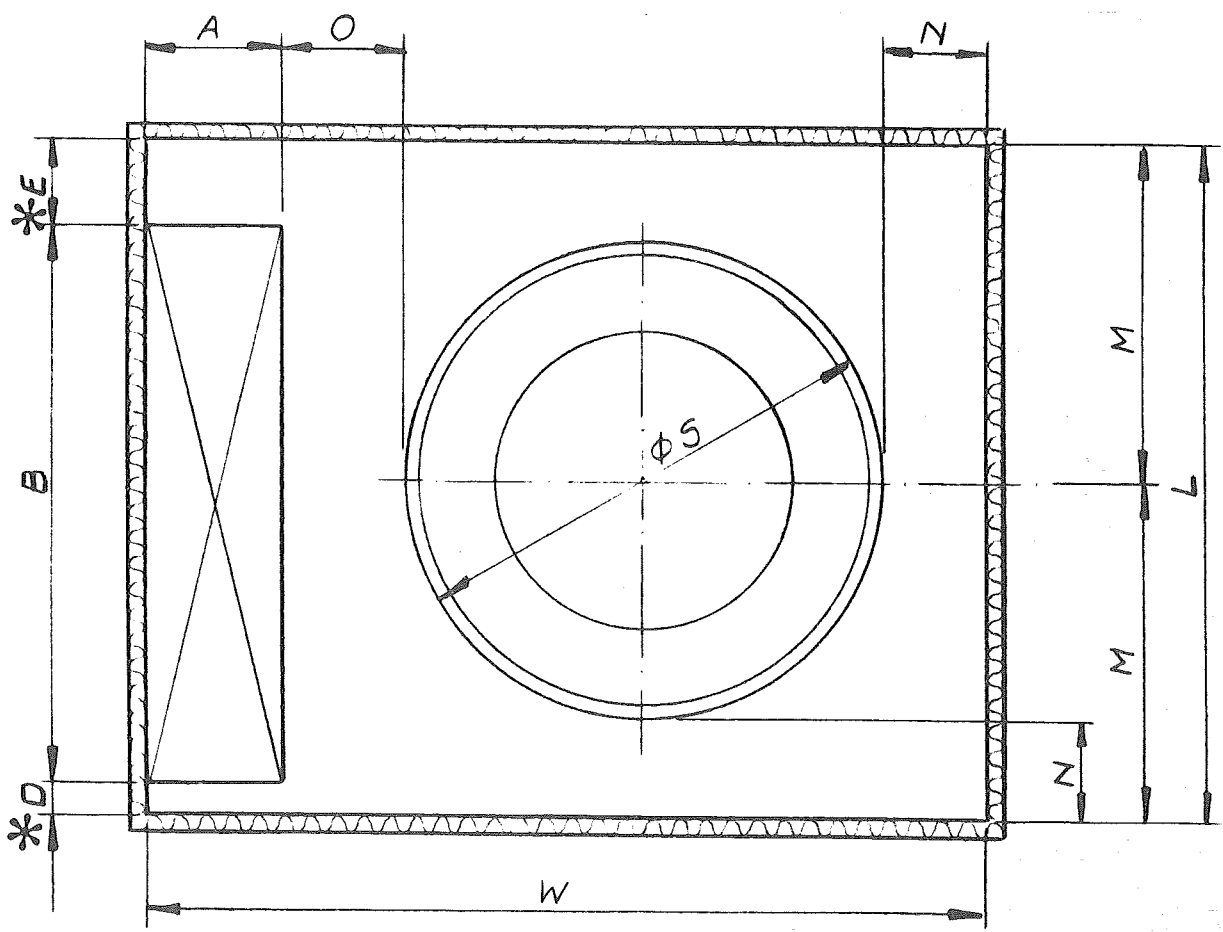
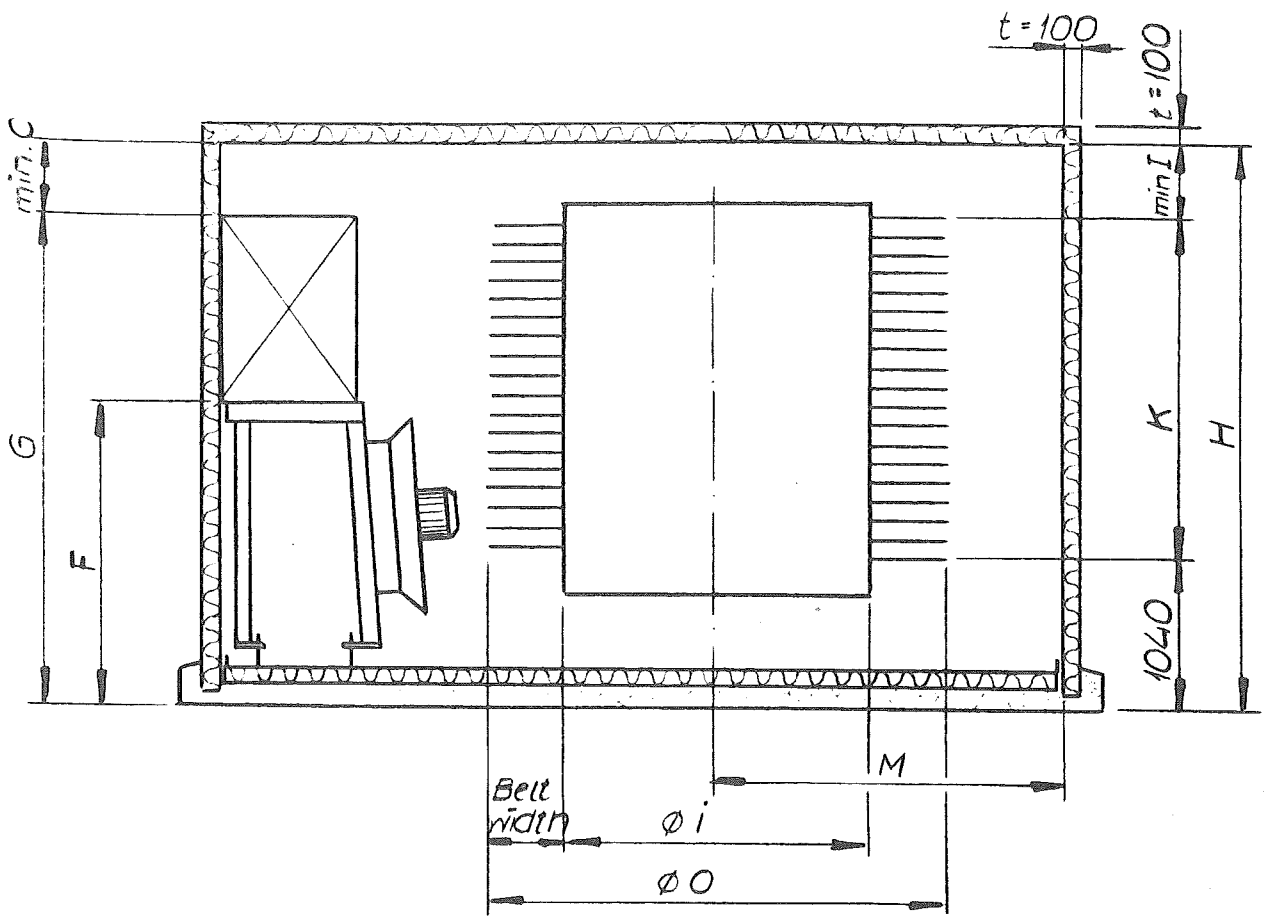
Hedau stående uppgifter gäller för
sile - byggda frysar Gyro freeze:

Belt width	\varnothing_i mm	\varnothing_o mm	\varnothing_s sheet ing mm
20"	2249	3265	3295
24"	2702	3922	3952
28"	3139	4561	4591
32"	3591	5217	5247
36"	4044	5872	5902

Front area, Air velocity, Air volume:

Freezer	Nominell Air volume M^3/h	Spiral		Evaporator	
		Area m^2	Air velocity m/s	Area m^2	Air velocity m/s
20"	49 000	4,55	3	3,87	3,5
24"	71 000	6,53	3	5,54	3,5
28"	95 000	8,81	3	7,50	3,5
32"	124 000	11,49	3	9,92	3,5
36"	157 000	14,51	3	12,44	3,5

$$\pi \left(\frac{\varnothing_s}{2}\right)^2 - \pi \left(\frac{\varnothing_i}{2}\right)^2$$



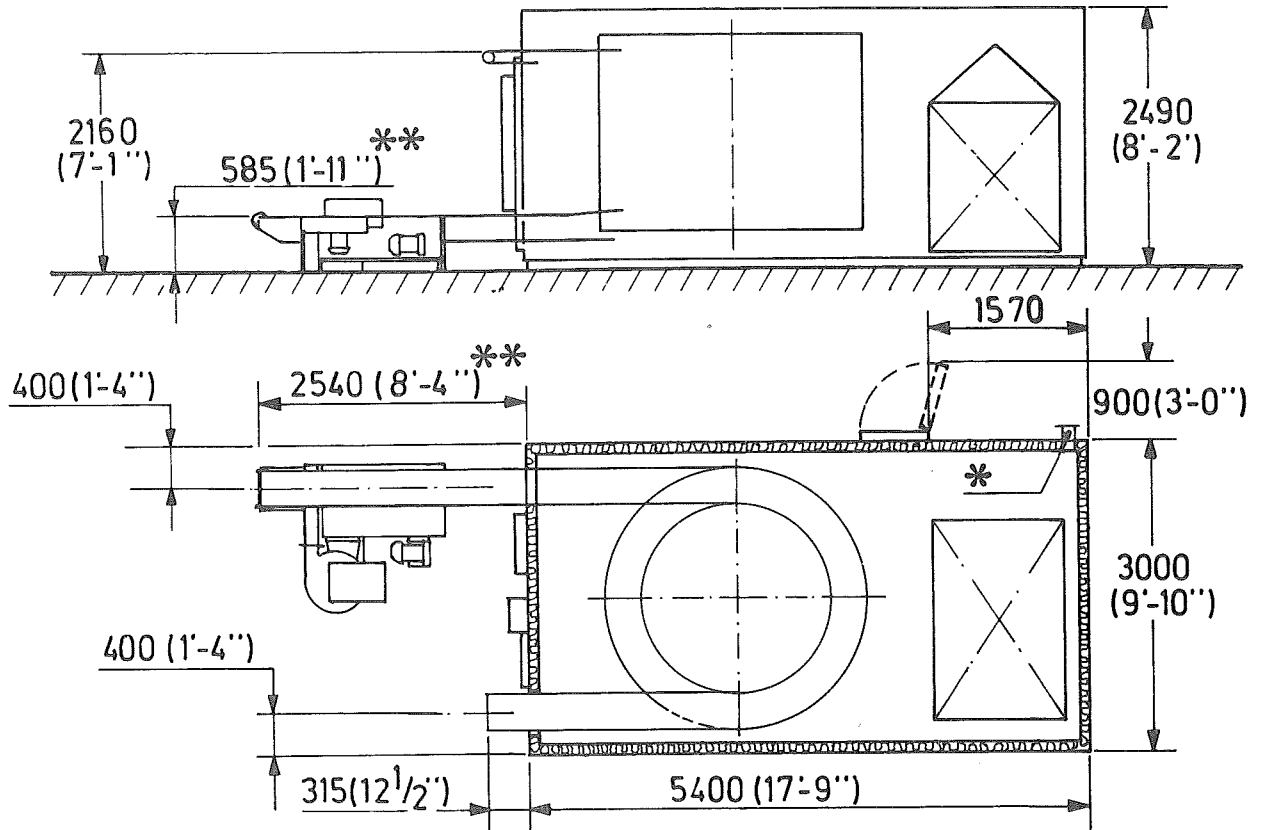
*Dimensions D and E can be interchanged.

Freezer Division

GYROFREEZE Midget

(Baubredd 16")

Dimensions /Ausmasse /Dimension /Dimensioner /Medidas



<p>* Water drain Auslass für Abtauwasser Collecteur d'eau</p>	<p>Utlopp för avfrostningsvatten Desagü</p>
---	---

<p>** Can be extended as an optional Kann wahlfrei vergrößert werden Peut être allongé en option</p>	<p>Förlängning kan levereras som extra utrustning Se puede ampliar como alternativa opcional</p>
--	--

Total operational weight 10.5 tons
Betriebsgewicht 10,5 Tonnen.
Poids total opérationel: 10,5 tonnes.

Totalvikt 10,5 ton
Peso total en funcionamiento 10,5 toneladas

Freezer Division

0042-1-006-D	1/2
Nov. 1982	

GYRoFREEZE 20", 24" and 28" models

Technical Data. Lay-out EW, WE, EE, WW, NN. Single vertical air flow.

SI-System (Old system)

MODELS		118-506	124-508	130-510	118-608	124-611	130-614	118-711	124-715	130-720
Belt types available		Omni-Grid stainless, with or without mesh. Omniflex galvanized or stainless steel.								
Belt width, total	mm	508	508	508	610	610	610	711	711	711
Belt width, net	mm	457	457	457	559	559	559	660	660	660
Number of effective tiers		18	24	30	18	24	30	18	24	30
Total belt length	m	208	270	331	247	321	395	285	371	457
Effective belt area (Omni-Grid)	m ²	70	93	116	104	138	173	144	193	241
Variable belt speed ¹	m/min.	2.5-17.0	2.5-17.0	2.5-17.0	2.7-17.0	2.7-18.0	2.9-23.0	2.9-22.0	2.9-22.0	2.9-17.0
Max. permitted product load (based on belt B24-12-17)	kg/m strength belt	5.4	5.4	5.4	6.8	5.4	5.0	4.5	5.0	6.5
Vertical clearance for product										
Omni-Grid	mm	60	60	60	60	60	60	60	60	60
Omniflex	mm	55	55	55	55	55	55	55	55	55
Electrical power Voltage		3 x 380 V or 3 x 415 V, 50 Hz, 3 x 440 V, 60 Hz								
Normal power consumption ²	kW	18	21	26	21	28	34	38	41	55
Max. power requirement ²	kVA	25	28	30	29	36	57	48	58	74
Motors of more than ² 1 kW rating		1x2.2 1x15.0	1x3 1x18.0	1x4 2x11.0	1x3 1x18.0	1x4 1x22.0	1x5.5 3x11.0	1x4 2x15.0	1x5.5 2x18.5	1x7.5 4x11.0
Refrigeration		Ammonia. Pump circulation. Circulation ratio = 4 to 6 times evaporation.								
Standard design		Ammonia. Pump circulation. Circulation ratio = 4 to 6 times evaporation.								
Standard evap. temp.	°C	-40 suction at evaporator (ammonia).								
Base load ³	kW	29	29	29	44	44	44	58	58	58
	(kcal/h)	(25,000)	(25,000)	(25,000)	(38,000)	(38,000)	(38,000)	(50,000)	(50,000)	(50,000)
Coil volume	l	300	400	500	400	550	700	550	750	1000
Defrosting system		Hot gas or water.								
Water defrost system ⁴	°C	Min. temp. +15								
Nominal required ⁵ spraying time	min.	15 - 20								
Water consumption at a pressure of 300 kPa (3 kp/cm ²) min. pressure 100 kPa (1 kg/cm ²)	l/min.	425	425	425	475	475	475	525	525	525

- 1 Drive system with other data available (optional extra).
- 2 Excluding belt washer. (Max. power requirement 5.5 kW).
- 3 Required refrigeration capacity minus product heat (excluding belt thawing).
- 4 Water defrost system (optional extra).
- 5 Provided a normal operation of the freezer.

Subject to change without notice.

Freezer Division

Manual	
0045-1-011-H	1/2
Nov. 1982	

GYROFREEZE Midget, GYROFREEZE MH

Technical Data

SI-System (Old system) (British Units overleaf)

		MIDGET				MH	MH	MH	MH
MODELS		111-406	114-406	116-406	118-406	118-406	120-406	123-406	125-406
Belt types available		Omni-Grid stainless, with or without mesh							
Belt, standard delivery		Omni-Grid stainless steel with mesh B24-12-17							
Belt width, total	mm	406	406	406	406	406	406	406	406
Belt width, net	mm	355	355	355	355	355	355	355	355
Number of effective tiers		10.5	13.5	15.5	17.5	17.5	19.5	22.5	24.5
Total belt length	m	112	135	153	168	168	185	209	226
Effective belt area (Omni-Grid)	m ²	26	33	38	43	43	48	55	60
Variable belt speed ¹	m/min	2 - 16				2 - 15			
Max. permitted product load (based on belt B24-12-17)	kg/m straight belt	11.7	8.5	7.0	6.0	10.0	9.0	8.0	7.0
Vertical clearance for product	mm	100	68	57	45	80	67	53	45
Electrical power		3 x 380 V or 3 x 415 V, 50 Hz, 3 x 440 V, 60 Hz							
Normal power consumption ²	kW	15	15	15	15	15	15	15	15
Max. power requirement ²	kVA	18	18	18	18	18	18	18	18
Motors of more than 1 kW rating ²		1 x 1.5 + 3 x 4				1 x 2.2 + 3 x 4			
Refrigeration		Ammonia. Pump circulation. Circulation ratio = 4 to 6 times evaporation.							
Standard design		R22. Pump circulation. Circulation ratio = 3 times evaporation							
Standard evap. temp.	°C	-40°C suction at evaporator (Ammonia) -43°C suction at evaporator (R22)							
Base load ³	kW	23							
	(kcal/h)	(20.000)							
Coil volume	l	275 VTA 580 M²							
Defrosting system		Hot gas or water.							
Water defrost system ⁴	°C	Min. temp. +15							
Water consumption at a pressure of 300 kPa (3 kp/cm ²) min. pressure 100 kPa (1 kg/cm ²)	l/min	330	330	330	330	330	330	330	330
Nominal required spraying time ⁵	min.	15							

- 1 Drive system with other data available (optional extra).
- 2 Excluding belt washer. (Max. power requirement 5.5 kW).
- 3 Required refrigeration capacity minus product heat (excluding belt thawing).
- 4 Water defrost system (optional extra).
- 5 Provided normal operation of the freezer.

NOTE! GYROFREEZE Midget 118-406 is the normally serial manufactured model.

Subject to change without notice.

Freezer Division

0042-2-015-A

7/42

Feb. 1982

* * *

GYROFREEZE

Product and Capacity Data

Product	Hamburger, raw, single 100% beef	Hamburger, raw, single 100% beef	Hamburger, raw, single 100% beef
Dim. mm.	Diam. 90 x 6,5	Diam. 90 x 9,5	Diam. 90 x 13
Weight, g	40	60	80
Temp. in °C	0	0	0
Enthalpy kJ/kg kcal/kg	251 (60)	251 (60)	251 (60)
Holding time, min.	12	18	23
Belt type	Omni-Grid B24-12-17	Omni-Grid B24-12-17	Omni-Grid B24-12-17

GF-Model	Capacity data	Capacity data	Capacity data
118-405	790 kg/h, 58% <i>Band läcken</i> 79 kW (68 000 kcal/h) 95 kW (82 000 kcal/h) 5	780 kg/h, 58% 78 kW (67 000 kcal/h)	805 kg/h, 58% 80 kW (69 000 kcal/h)
118-5xx	1000 kg/h, 46% 1 99 kW (85 000 kcal/h) xx = 06 119 kW (102 000 kcal/h) xx = 07 5	1000 kg/h, 46% 99 kW (85 000 kcal/h) xx = 06	1030 kg/h, 46% 101 kW (87 000 kcal/h) xx = 06
124-5xx	1340 kg/h, 46% 1 123 kW (105 000 kcal/h) xx = 07 147 kW (126 000 kcal/h) xx = 08 5	1340 kg/h, 46% 1 123 kW (105 000 kcal/h) xx = 07	1400 kg/h, 46% 1 127 kW (109 000 kcal/h) xx = 07
130-5xx	1670 kg/h, 46% 1 146 kW (126 000 kcal/h) xx = 08 176 kW (152 000 kcal/h) xx = 10 5	1670 kg/h, 46% 1 146 kW (125 000 kcal/h) xx = 08	1745 kg/h, 46% 1 151 kW (130 000 kcal/h) xx = 09
118-6xx	1810 kg/h, 55% 1 171 kW (147 000 kcal/h) xx = 10 205 kW (176 000 kcal/h) xx = 11 5	1810 kg/h, 55% 1 171 kW (147 000 kcal/h) xx = 10	1890 kg/h, 55% 1 174 kW (150 000 kcal/h) xx = 11
124-6xx	2420 kg/h, 55% 1 214 kW (184 000 kcal/h) xx = 13 250 kW (215 000 kcal/h) xx = 14 5	2420 kg/h, 55% 1 214 kW (184 000 kcal/h) xx = 13	2520 kg/h, 55% 1 220 kW (190 000 kcal/h) xx = 13
130-6xx	3020 kg/h, 55% 1,2 287 kW (247 000 kcal/h) xx = 16 330 kW (285 000 kcal/h) xx = 19 5	3020 kg/h, 55% 1,2 287 kW (247 000 kcal/h) xx = 16	3150 kg/h, 55% 1 264 kW (227 000 kcal/h) xx = 14
118-7xx	2120 kg/h, 46% 1 207 kW (178 000 kcal/h) xx = 11 238 kW (205 000 kcal/h) xx = 11 5	2120 kg/h, 46% 1 207 kW (178 000 kcal/h) xx = 11	2210 kg/h, 46% 1 213 kW (183 000 kcal/h) xx = 13
124-7xx	2830 kg/h, 46% 1 256 kW (220 000 kcal/h) xx = 15 295 kW (253 000 kcal/h) xx = 18 5	2830 kg/h, 46% 1 256 kW (220 000 kcal/h) xx = 15	2950 kg/h, 46% 1 264 kW (227 000 kcal/h) xx = 15
130-7xx	3540 kg/h, 46% 1 306 kW (263 000 kcal/h) xx = 18 352 kW (303 000 kcal/h) xx = 20 5	3540 kg/h, 46% 1 306 kW (263 000 kcal/h) xx = 18	3690 kg/h, 46% 1 316 kW (272 000 kcal/h) xx = 20

GFU

20"

24"

28"

1 Twin belt system **2** Split vertical airflow
4 Omitted due to loading problems etc.

3 Capacity adjusted to coil size
5 When continuous belt thawing is required

Freezer Division

0042-2-015-A

13/42

Feb. 1982

GYRoFREEZE

Product and Capacity Data

Product	Meat ball, raw,	Faggot, raw,	
Dim. mm.	Sphere 25	Sphere 43	
Weight, g	8	43	
Temp. in °C	0	+5	
Enthalpy kJ/kg kcal/kg	251 (60)	272 (65)	
Holding time, min.	13	28	
Belt type	Omni-Grid B36-16-17	Omni-Grid B36-16-17	

GF-Model	Capacity data	Capacity data	
118-405	1100 kg/h, 38% 100 kW (86 000 kcal/h)	980 kg/h, 38% 98 kW (84 000 kcal/h)	
118-5xx	1950 kg/h, 37% 165 kW (142 000 kcal/h) xx = 09	1720 kg/h, 40% 160 kW (137 000 kcal/h) xx = 09	
124-5xx	2600 kg/h, 37% 229 kW (197 000 kcal/h) xx = 14	2300 kg/h, 40% 222 kW (191 000 kcal/h) xx = 14	1,2
130-5xx	3250 kg/h, 37% 275 kW (236 000 kcal/h) xx = 16	2870 kg/h, 40% 265 kW (228 000 kcal/h) xx = 16	2
118-6xx	2350 kg/h, 30% 208 kW (179 000 kcal/h) xx = 13	2610 kg/h, 41% 242 kW (208 000 kcal/h) xx = 14	
124-6xx	3130 kg/h, 30% 295 kW (253 000 kcal/h) xx = 16	3480 kg/h, 41% 337 kW (290 000 kcal/h) xx = 20	1,2
130-6xx	3920 kg/h, 30% 350 kW (301 000 kcal/h) xx = 20	4350 kg/h, 41% 405 kW (348 000 kcal/h) xx = 22	1,2
118-7xx	4040 kg/h, 37% 341 kW (293 000 kcal/h) xx = 20	3650 kg/h, 41% 335 kW (288 000 kcal/h) xx = 20	1
124-7xx	5390 kg/h, 37% 474 kW (408 000 kcal/h) xx = 26	4870 kg/h, 41% 467 kW (401 000 kcal/h) xx = 26	1,2
130-7xx	6740 kg/h, 37% 569 kW (489 000 kcal/h) xx = 30	6090 kg/h, 41% 559 kW (480 000 kcal/h) xx = 30	1,2

- 1 Twin belt system 2 Split vertical airflow 3 Capacity adjusted to coil size
- 4 Omitted due to loading problems etc. 5 When continuous belt thawing is required

Manual

0042-2-015-A

14/42

Feb. 1982

* * *

Freezer Division

GYROFREEZE

Product and Capacity Data

Product	Meat ball, fried	Meat ball, fried	Chicken quarters, raw
Dim. mm.	Sphere 24	Sphere 33	145 x 105 x 55 0,036 m ² /kg
Weight, g	7	18	320
Temp. in °C	+60	+60	+10
Enthalpy kJ/kg kcal/kg	377 (90)	377 (90)	293 (70)
Holding time, min.	16	22	60
Belt type	Omni-Grid B24-12-17	Omni-Grid B24-12-17	Omniflex 1"x1/2" with special outfeed roller

GF-Model	Capacity data	Capacity data	Capacity data
118-405	1100 kg/h, 60% 139 kW (119 000 kcal/h) 3	1200 kg/h, 60% 149 kW (128 000 kcal/h) 3	4
118-5xx	2190 kg/h, 60% 278 kW (239 000 kcal/h) xx = 16 2	2410 kg/h, 60% 300 kW (258 000 kcal/h) xx = 18 2	1160 kg/h, 60% 123 kW (106 000 kcal/h) xx = 07
124-5xx	2930 kg/h, 60% 355 kW (305 000 kcal/h) xx = 20 2	3150 kg/h, 60% 378 kW (325 000 kcal/h) xx = 20 2,3	1550 kg/h, 60% 156 kW (134 000 kcal/h) xx = 09
130-5xx	3030 kg/h, 60% 365 kW (314 000 kcal/h) xx = 20 1,2	4	1940 kg/h 60% 206 kW (177 000 kcal/h) xx = 12 2
118-6xx	3270 kg/h, 60% 419 kW (360 000 kcal/h) xx = 26 2	3600 kg/h, 60% 452 kW (388 000 kcal/h) xx = 26 2	1740 kg/h, 60% 186 kW (160 000 kcal/h) xx = 11
124-6xx	4370 kg/h, 60% 483 kW (415 000 kcal/h) xx = 28 1,2	4600 kg/h, 60% 556 kW (478 000 kcal/h) xx = 28 2,3	2320 kg/h, 60% 234 kW (201 000 kcal/h) xx = 13
130-6xx	5000 kg/h, 55% 599 kW (515 000 kcal/h) xx = 28 1,2,3	4	2900 kg/h, 60% 312 kW (268 000 kcal/h) xx = 20 2
118-7xx	4570 kg/h, 60% 577 kW (496 000 kcal/h) xx = 36 2	5020 kg/h, 60% 624 kW (536 000 kcal/h) xx = 36 2	2410 kg/h, 60% 255 kW (219 000 kcal/h) xx = 15
124-7xx	5900 kg/h, 60% 723 kW (621 000 kcal/h) xx = 40 1,2,3	4	3210 kg/h 60% 320 kW (275 000 kcal/h) xx = 18
130-7xx	4	4	4010 kg/h, 60% 425 kW (365 000 kcal/h) xx = 26 2

- 1** Twin belt system **2** Split vertical airflow **3** Capacity adjusted to coil size
4 Omitted due to loading problems etc. **5** When continuous belt thawing is required

Freezer Division

Manual	
0042-2-015-A	29/42
Feb. 1982	* * *

GYROFREEZE

Product and Capacity Data

Product	Ice-cream, 12 % fat	Ice-cream cake, Sponge cake 1 Ice-cream 2 Cream 3	
Dim. mm.	190 x 95 x 55 Capsule A	Diam. 192 x 53 Tray 0	
Weight, g	550	550	
Temp. in °C	-4	1 +20 2 -4 3 +6	
Enthalpy kJ/kg kcal/kg	210 (50)	210 (50)	
Holding time, min.	80	50	
Belt type	Omni-Grid without mesh	Omni-Grid without mesh	
GF-Model	Capacity data	Capacity data	
118-405	4	4	
118-5xx	990 kg/h, 62% 87 kW (75 000 kcal/h) xx = 06	790 kg/h, 48% 76 kW (65 000 kcal/h) xx = 06	
124-5xx	1320 kg/h, 62% 106 kW (91 000 kcal/h) xx = 06	1100 kg/h, 48% 93 kW (80 000 kcal/h) xx = 06	
130-5xx	1650 kg/h, 62% 126 kW (108 000 kcal/h) xx = 07	1320 kg/h, 48% 108 kW (93 000 kcal/h) xx = 06	
118-6xx	1470 kg/h, 62% 130 kW (112 000 kcal/h) xx = 08	1460 kg/h, 58% 129 kW (111 000 kcal/h) xx = 08	
124-6xx	1960 kg/h, 62% 158 kW (136 000 kcal/h) xx = 10	1900 kg/h, 58% 155 kW (133 000 kcal/h) xx = 08	
130-6xx	2450 kg/h, 62% 187 kW (161 000 kcal/h) xx = 11	2380 kg/h, 58% 183 kW (157 000 kcal/h) xx = 11	
118-7xx	2030 kg/h, 62% 177 kW (152 000 kcal/h) xx = 11	1700 kg/h, 49% 157 kW (135 000 kcal/h) xx = 11	
124-7xx	2700 kg/h, 62% 215 kW (185 000 kcal/h) xx = 13	2210 kg/h, 49% 187 kW (161 000 kcal/h) xx = 11	
130-7xx	3380 kg/h, 62% 255 kW (219 000 kcal/h) xx = 15	2760 kg/h, 49% 219 kW (188 000 kcal/h) xx = 13	

- 1 Twin belt system 2 Split vertical airflow 3 Capacity adjusted to coil size
- 4 Omitted due to loading problems etc. 5 When continuous belt thawing is required

Freezer Division

0042-2-015-A

37/42

Feb. 1982

* * *

GYRoFREEZE

Product and Capacity Data

Product	Cheese cake	Lasagne	
Dim. mm.	Diam. 160 x 35 Tray I	220 x 130 x 35 Tray E	
Weight, g	cake 360 Jam 160 Total: 520	500	
Temp. in °C	cake +32 Jam +21	+40	
Enthalpy kJ/kg kcal/kg	272 (65)	419 (100)	
Holding time, min.	60	60	
Belt type	Omni-Grid without mesh	Omni-Grid without mesh	
GF-Model	Capacity data	Capacity data	
118-405	580 kg/h, 52% 68 kW (58 000 kcal/h)	400 kg/h, 54% 70 kW (60 000 kcal/h)	
118-5xx	1120 kg/h, 62% 114 kW (98 000 kcal/h) xx = 07	780 kg/h, 65% 120 kW (103 000 kcal/h) xx = 07	
124-5xx	1500 kg/h, 62% 143 kW (123 000 kcal/h) xx = 08	1050 kg/h, 65% 151 kW (130 000 kcal/h) xx = 09	
130-5xx	1870 kg/h, 62% 171 kW (147 000 kcal/h) xx = 10	1310 kg/h, 65% 182 kW (156 000 kcal/h) xx = 10	
118-6xx	1350 kg/h, 50% 147 kW (126 000 kcal/h) xx = 08	1260 kg/h, 69% 191 kW (164 000 kcal/h) xx = 11	
124-6xx	1810 kg/h, 50% 182 kW (156 000 kcal/h) xx = 10	1680 kg/h, 69% 240 kW (206 000 kcal/h) xx = 14	
130-6xx	2260 kg/h, 50% 215 kW (185 000 kcal/h) xx = 13	2100 kg/h, 69% 320 kW (275 000 kcal/h) xx = 19	2
118-7xx	2100 kg/h, 56% 217 kW (187 000 kcal/h) xx = 13	1460 kg/h, 58% 228 kW (196 000 kcal/h) xx = 13	
124-7xx	2800 kg/h, 56% 270 kW (232 000 kcal/h) xx = 15	1950 kg/h, 58% 285 kW (245 000 kcal/h) xx = 15	
130-7xx	3500 kg/h, 56% 327 kW (281 000 kcal/h) xx = 20	2440 kg/h, 58% 342 kW (294 000 kcal/h) xx = 20	

1 Twin belt system **2** Split vertical airflow

3 Capacity adjusted to coil size

4 Omitted due to loading problems etc.

5 When continuous belt thawing is required

0042-2-015-A

36/42

Freezer Division

Feb. 1982

* * *

GYROFREEZE

Product and Capacity Data

Product	Meat pie	Pizza pie, unbaked	Apple pie, baked
Dim. mm.	130 x 100 x 25 Tray G	Diam. 180 x 14 Tray I	Diam. 180 x 25 Tray I
Weight, g	150	190	350
Temp. in °C	+5	+35	+25
Enthalpy kJ/kg kcal/kg	272 (65)	293 (70)	335 (80)
Holding time, min.	40	28	37
Belt type	Omni-Grid without mesh	Omni-Grid without mesh	Omni-Grid without mesh

GF-Model	Capacity data	Capacity data	Capacity data
118-405	500 kg/h, 62% 62 kW (53 000 kcal/h)	490 kg/h, 68% 64 kW (55 000 kcal/h)	560 kg/h, 57% 76 kW (65 000 kcal/h)
118-5xx	800 kg/h, 66% 90 kW (77 000 kcal/h) xx = 06	640 kg/h, 54% 82 kW (70 000 kcal/h) xx = 06	720 kg/h, 45% 97 kW (83 000 kcal/h) xx = 06
124-5xx	1060 kg/h, 66% 110 kW (94 000 kcal/h) xx = 07	820 kg/h, 54% 96 kW (83 000 kcal/h) xx = 06	960 kg/h, 45% 119 kW (102 000 kcal/h) xx = 07
130-5xx	1330 kg/h, 66% 131 kW (112 000 kcal/h) xx = 07	1030 kg/h, 54% 114 kW (98 000 kcal/h) xx = 07	1210 kg/h, 45% 142 kW (122 000 kcal/h) xx = 08
118-6xx	1200 kg/h, 66% 135 kW (116 000 kcal/h) xx = 08	1100 kg/h, 65% 134 kW (115 000 kcal/h) xx = 08	1310 kg/h, 55% 167 kW (143 000 kcal/h) xx = 10
124-6xx	1600 kg/h, 66% 165 kW (142 000 kcal/h) xx = 10	1470 kg/h, 65% 164 kW (141 000 kcal/h) xx = 10	1750 kg/h, 55% 207 kW (178 000 kcal/h) xx = 11
130-6xx	2000 kg/h, 66% 196 kW (168 000 kcal/h) xx = 11	1840 kg/h, 65% 195 kW (167 000 kcal/h) xx = 11	2190 kg/h, 55% 280 kW (241 000 kcal/h) xx = 16 2
118-7xx	1630 kg/h, 65% 182 kW (156 000 kcal/h) xx = 11	1330 kg/h, 54% 167 kW (143 000 kcal/h) xx = 11	2030 kg/h, 62% 248 kW (213 000 kcal/h) xx = 13
124-7xx	2170 kg/h, 65% 223 kW (192 000 kcal/h) xx = 13	1700 kg/h, 54% 197 kW (169 000 kcal/h) xx = 11	2700 kg/h, 62% 313 kW (269 000 kcal/h) xx = 18
130-7xx	2720 kg/h, 65% 264 kW (227 000 kcal/h) xx = 15	2120 kg/h, 54% 231 kW (199 000 kcal/h) xx = 13	3380 kg/h, 62% 372 kW (320 000 kcal/h) xx = 20

1 Twin belt system 2 Split vertical airflow

3 Capacity adjusted to coil size

4 Omitted due to loading problems etc.

5 When continuous belt thawing is required