

# Adaptiv temperaturreglering av bostadshus

---

Gustav Hedberg



UPPSALA  
UNIVERSITET

Teknisk- naturvetenskaplig fakultet  
UTH-enheten

Besöksadress:  
Ångströmlaboratoriet  
Lägerhyddsvägen 1  
Hus 4, Plan 0

Postadress:  
Box 536  
751 21 Uppsala

Telefon:  
018 – 471 30 03

Telefax:  
018 – 471 30 00

Hemsida:  
<http://www.teknat.uu.se/student>

## Abstract

Adaptiv temperaturreglering av bostadshus

### **Adaptive Temperature Control in Residential Buildings**

---

*Gustav Hedberg*

The control of indoor temperature must be able to handle large time constants and both measurable disturbances like outdoor temperature, and non-measurable disturbances, like waste heat from appliances. Due to cheaper microcontrollers (small computers with designated tasks) with better performance, more advanced and computation-demanding methods for control can be used.

In this thesis, Generalized Predictive Control (GPC) has been implemented and tested on a theoretically derived model of a house. GPC states that a model of the system is estimated in every sample, and the control is carried out simultaneously, based on predictions of the model. Disturbances like solar radiation can be estimated and used in the control as well. The control signals are determined by quadratic programming, which makes it possible to consider conditions, for example limitations on the control signals.

GPC requires a number of parameters to be defined, like model order and control signal penalty, but is relatively robust for variations in them. Feed-forward by measuring or predicting the outdoor temperature did not give any better control due to the fact that the feedback is relatively fast, but better control can be achieved by predicting the solar radiation.

Handledare: Stefan Larsson  
Ämnesgranskare: Bengt Carlsson  
Examinator: Elisabet Andrésdóttir  
ISSN: 1650-8319, UPTEC STS08 020

## Populärvetenskaplig beskrivning

Reglering av inomhustemperatur måste bland annat hantera stora tidskonstanter och mätbara störningar som utetemperatur och icke-mätbara störningar som och spillvärme från apparater. Tack vare sjunkande priser och bättre prestanda på mikrokontroller, små datorer med en specifik uppgift, kan mer avancerade och beräkningskrävande reglermetoder användas.

I detta examensarbete har Generalized Predictive Control (GPC) implementerats och testats på en teoretiskt framtagen modell av ett hus. GPC går ut på att en modell av systemet skattas i varje sampel, samtidigt som regleringen utgår från prediktioner av modellen. Även störningar som till exempel solinstrålning kan prediceras och användas i regleringen. Genom att styrsignalerna bestäms med hjälp av kvadratisk programmering, kan hänsyn tas till bivillkor som till exempel begränsningar på styrsignalen.

GPC kräver att ett antal parametrar, såsom modellordning och straff på styrsignalinkrementen, definieras, men är relativt robust för variationer på dessa. Varken framkoppling genom att mäta eller predicera utetemperaturen ger bättre reglering då återkopplingen är relativt snabb, däremot fås en bättre reglering då solinstrålningen prediceras.

# Innehåll

<b>1</b>	<b>Inledning</b>	<b>4</b>
1.1	Bakgrund . . . . .	4
1.2	Syfte . . . . .	5
1.3	Avgränsningar . . . . .	5
1.4	Rapportens uppdelning . . . . .	5
1.5	Notation . . . . .	6
<b>2</b>	<b>Teori</b>	<b>7</b>
2.1	Adaptiv reglering . . . . .	7
2.2	Systemidentifiering . . . . .	7
2.2.1	Black box . . . . .	7
2.2.2	Identifiering: rekursiv prediktionsfelsmetod . . . . .	12
2.3	Generalized Predictive Control . . . . .	19
2.3.1	Styrsignals- och prediktionshorisonter . . . . .	20
2.3.2	Prediktion på tillståndsform . . . . .	21
2.3.3	Optimering med hänsyn till bivillkor . . . . .	23
2.3.4	Active Set-metoden . . . . .	25
2.3.5	Sammanfattning GPC-algoritmen . . . . .	27
2.4	Simulering . . . . .	28
<b>3</b>	<b>Metod</b>	<b>29</b>
3.1	Implementering . . . . .	29
3.1.1	Programstruktur . . . . .	29
3.1.2	Hjälpfunktioner och datastrukturer . . . . .	30
3.1.3	Minnesläckor i programmet . . . . .	30
3.2	Test av regulatorn . . . . .	31
3.2.1	Husmodell . . . . .	31
3.3	Kvalitetsmått . . . . .	35
<b>4</b>	<b>Resultat</b>	<b>36</b>

4.1	Identifiering . . . . .	36
4.1.1	Initialvärden . . . . .	36
4.1.2	Skattningsfel . . . . .	37
4.1.3	Parameterskattning . . . . .	39
4.2	Reglering . . . . .	40
4.2.1	Inverkan av designparametrar . . . . .	40
4.2.2	Prediktion av solinstrålning . . . . .	43
4.2.3	Varierande referenssignal . . . . .	47
4.2.4	Brusiga signaler . . . . .	48
<b>5</b>	<b>Diskussion</b>	<b>50</b>
<b>A</b>	<b>Matrisinverteringslemmat</b>	<b>52</b>
<b>B</b>	<b>Rekursivt lågpassfilter</b>	<b>53</b>
<b>C</b>	<b>Härledning av husmodell</b>	<b>54</b>

# Figurer

2.1	ARMAX-modellen på blockdiagramform. . . . .	9
2.2	Blockdiagram GPC-reglering. . . . .	20
2.3	Prediktion av framtida utsignaler ger optimala styrsignaler. . .	21
4.1	Skattningsfelet de 25 första stegen, utetemperatur mäts, $na = nb_1 = nb_2 = 1, nc = 0$ . . . . .	37
4.2	Skattningsfelet tidssteg 4000 till 5000, samma modellparametrar som ovan. . . . .	38
4.3	Koefficienterna i $\hat{\theta}(t)$ de första 25 tidsstegen, samma modellparametrar som ovan med tillägget att även solinstrålning mäts. . . . .	39
4.4	Koefficienterna i $\hat{\theta}(t)$ för 10 000 tidssteg, samma modellparametrar som ovan. . . . .	40
4.5	Reglering där endast innetemperaturen mäts, $na = 1, nb = 1, nc = 0$ . . . . .	42
4.6	Femstegsprediktion av solinstrålning. . . . .	44
4.7	Simulering då solinstrålningen mäts och predikteras två steg framåt, $na = 4, nb_1 = nb_2 = nb_3 = 3, nc = 0$ . . . . .	45
4.8	Simulering då solinstrålningen mäts och predikteras 15 steg framåt, samma modellparametrar som ovan. . . . .	46
4.9	Simulering då solinstrålningen mäts och prediceras 15 steg framåt, varierande referenssignal, i övrigt samma modellparametrar som ovan. . . . .	47
4.10	Simulering då solinstrålningen mäts och prediceras 15 steg framåt, varierande referenssignal, brus på stör- och utsignaler med varians 0.25, i övrigt samma modellparametrar som ovan. . . . .	48

# Kapitel 1

## Inledning

### 1.1 Bakgrund

Intresset för att minska energiåtgången i hus har ökat allteftersom energipriser gått upp och miljöhänsyn blivit en naturlig del av vardagen. Effektiva reglersystem är en viktig faktor för att spara energi i byggnader. Mikrokontrollers, alltså små datorer med en eller ett fåtal arbetsuppgifter, vanligen med integrerad processor och inbyggt minne, blir allt billigare och får högre prestanda. Det gör det möjligt att ha en avancerad reglering som kräver mer beräkningar och data.

Så kallade lågenergihus kännetecknas av att de i större utsträckning än vanliga huskonstruktioner är beroende av sådana värmekällor som boende, spillvärme från apparater och solinstrålning genom fönster. Detta medför att värmeregleringen för att fungera effektivt måste kunna hantera både värmekällor som mäts och värmekällor som inte kan mätas.

Tidskonstanten för hus är relativt stor på grund av husets massa. Husets egenskaper är ofta beroende av till exempel årstid. Detta gör det svårt för en enkel regulator att fungera bra. Reglering som bygger på en modell av regleobjektet har bättre förutsättningar att klara sådana förhållanden jämfört med en vanlig PID-regulator.

Ett sätt att bygga upp en modell på är att skatta den från uppmätta ut- och insignaler. Om skattningen görs rekursivt, det vill säga att den uppdateras i varje sampel, behöver endast små datamängder hanteras, vilket är en

förutsättning för att en billig mikrokontroller ska kunna användas.

Om regulatorn skulle kunna ta hänsyn till skattningar av framtida störsignaler skulle regleringen kunna bli ännu bättre. Till exempel skulle regulatorn kunna minska värmeeffekten innan solen gått upp, och därmed minimera reglerfelet samtidigt som energi sparas.

Idag finns början till ett reglersystem framtaget av företaget Solarus i form av ett mikrokontrollerbaserat logg- och styrsystem som kan utföra beräkningar med ungefär samma prestanda som en pc. Det befintliga reglersystemet har dock inte använts i detta examensarbete.

## 1.2 Syfte

Syftet med examensarbetet är att skapa en prototyp till ett reglersystem som förberedelse till implementering i en mikrokontroller. Reglersystemet ska med hjälp av en kontinuerligt uppdaterad modell styra inomhusklimatet i ett hus där inne-, utetemperatur och solinstrålning kan mätas. Rutiner för systemidentifiering och reglering ska implementeras. Reglering med hänsyn till skattning av framtida störsignaler ska också prövas.

Regulatorn ska testas genom att reglera en modell av ett hus med vanligt förekommande signaler och störningar.

## 1.3 Avgränsningar

Den färdiga regulatorn har av praktiska skäl inte kunnat prövas i verkligheten, utan endast på en modell. På grund av detta har aspekter som samlingsintervall och filtrering av signaler inte behandlats särskilt utförligt.

## 1.4 Rapportens uppdelning

Rapporten är förutom denna inledning uppdelad i fem delar, teori, metod, resultat, slutsats samt appendix. I teoridelen går den teori som används ige-



nom. Metodkapitlet beskriver översiktligt hur teorin implementerats i programkod, samt modellen som regulatorn testas på. I resultatkapitlet presenteras resultaten från testkörningarna. Diskussionskapitlet gör en sammanfattning av resultaten och diskuterar hur regulatorn kan utvecklas. Ett appendix beskriver mer ingående husmodellen som regulatorn testats med, där finns också bevis för matrisinverteringslemmat samt en beskrivning av ett rekursivt låpassfilter.

## 1.5 Notation

För tydlighets skull skrivs matriser och vektorer med fet stil,  $\mathbf{x}$ , medan skalärer skrivs med kursiv stil,  $x$ .

# Kapitel 2

## Teori

### 2.1 Adaptiv reglering

Adaptiv reglering betyder att regulatoren kan anpassa sig till systemet som ska styras. Systemidentifiering används för att under systemets körning kontinuerligt skatta en modell. Prediktioner utifrån modellen används sedan för att styra systemet mot ett börvärde. Systemidentifiering och reglering utifrån prediktioner med en modell kommer därför att behandlas i detta kapitel.

### 2.2 Systemidentifiering

Systemidentifiering används för att skapa en modell av ett system genom att mäta in- och utsignaler. Under adaptiv reglering genomförs den i realtid under systemets drift vilket innebär att rekursiva metoder, som kan uppdatera modellen för varje nytt sampel, måste användas.

#### 2.2.1 Black box

Om det inte finns någon kunskap om systemet eller om det anses för komplicerat att bygga upp en fysikalisk modell av det som ska modelleras är black box-metoden lämplig. Den bygger på att black box-modellens koefficienter

anpassas till den data som samlats in. Ett antal designparametrar kan anges för att skapa en så bra prediktion som möjligt. En allmän, linjär tidsdiskret modell ges av

$$y(t) = \eta(t) + w(t) \quad (2.1)$$

där  $w(t)$  representerar störning och  $\eta(t)$  den störningsfria utsignalen som kan skrivas

$$\eta(t) = G(q^{-1}, \boldsymbol{\theta})u(t)$$

där

$$G(q^{-1}, \boldsymbol{\theta}) = \frac{B(q^{-1})}{F(q^{-1})} = \frac{b_1q^{-nk} + b_2q^{-nk-1} + \dots + b_{nb}q^{-nk-nb+1}}{1 + f_1q^{-1} + \dots + f_{nf}q^{-nf}} \quad (2.2)$$

Störningen  $w(t) = H(q^{-1}, \boldsymbol{\theta})e(t)$  där  $e(t)$  representerar vitt brus kan skrivas på samma sätt, alltså

$$H(q^{-1}, \boldsymbol{\theta}) = \frac{C(q^{-1})}{D(q^{-1})} = \frac{1 + c_1q^{-1} + \dots + c_{nc}q^{-nc}}{1 + d_1q^{-1} + \dots + d_{nd}q^{-nd}} \quad (2.3)$$

Sammanfattningsvis kan 2.1 skrivas

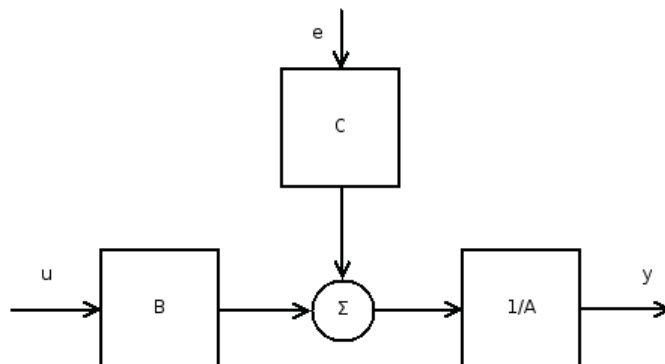
$$y(t) = G(q^{-1}, \boldsymbol{\theta})u(t) + H(q^{-1}, \boldsymbol{\theta})e(t) \quad (2.4)$$

Parametrarna  $nb$ ,  $nc$ ,  $nd$ ,  $nf$  och  $nk$  i 2.2 och 2.3 är strukturparametrar som väljs. Sedan anpassas  $b_i$ ,  $c_i$ ,  $d_i$  och  $f_i$  till insamlad data. Modellen kallas Box-Jenkins efter sina upphovsmän, statistikerna Box och Jenkins.

Det är vanligt att förenkla (2.4) genom att anta att  $F(q^{-1}) = D(q^{-1}) = A(q^{-1})$ . (2.4) förlängs sedan med  $A(q^{-1})$ :

$$A(q^{-1})y(t) = B(q^{-1})u(t) + C(q^{-1})e(t) \quad (2.5)$$

Resultatet blir ARMAX-modellen som fått sitt namn från att  $A(q^{-1})y(t)$  motsvarar en AutoRegression,  $C(q^{-1})e(t)$  ett glidande medelvärde (Moving average) och  $B(q^{-1})u(t)$  representerar en eXtra insignal eller eXogen variabel. I ARMAX-modellen förutsätts, till skillnad från Box-Jenkins-modellen, störningar gå genom samma nämnardynamik som insignalerna.



Figur 2.1: ARMAX-modellen på blockdiagramform.

Om  $C(q^{-1})e(t)$  i (2.5) sätts till 1 fås ARX-modellen som fått sitt namn genom att den till skillnad från ARMAX-modellen inte har något glidande medelvärde (Moving Average):

$$A(q^{-1})y(t) = B(q^{-1})u(t) + e(t) \quad (2.6)$$

Fördelen med ARX-modellen är att den är enkel att skatta men nackdelen är att  $A$ -polynomet måste beskriva både insignal och störningar vilket kan göra gradtalet onödigt högt när modellen anpassas till data (Ljung & Glad 2004).

## Stationäritet

Stationäritet innebär att en process sannolikhetsfördelning är densamma vid alla tidpunkter. Detta medför att egenskaper som medelvärden och varianser inte förändras över tid. Det finns två definitioner av stationäritet, stark och svag. För en svagt stationär process räcker det om  $\text{Cov}\{X(s), X(s+t)\}$  och  $E\{X(s)\}$  samt  $E\{X^2(s)\}$  är oberoende av  $s$ . Denna definition är betydligt enklare att använda än definitionen på stark stationäritet, och tillräcklig vid systemidentifiering. Ett exempel på en icke-stationär process är slumpvandring:

$$x(t+1) = x(t) + v(t) \quad (2.7)$$

$$y(t) = x(t) + e(t) \quad (2.8)$$

där  $\text{Cov}\{v(t), v(s)\} = \lambda_v^2 \delta_{t,s}$  och  $\text{Cov}\{e(t), e(s)\} = \lambda_e^2 \delta_{t,s}$ . Processen (2.7) har en pol på enhetscirkeln (det kan ses om den skrivs som  $x(t) = \frac{v(t)}{1-q^{-1}}$ ) vilket

bryter mot definitionen för svag stationaritet. En differentiering av (2.7) ger

$$z(t) = \Delta y(t) = v(t-1) + e(t) - e(t-1) \quad (2.9)$$

där  $\Delta$  betecknar differensoperatoren  $(1 - q^{-1})$ . Ekvation (2.9) är en stationär process eftersom  $\text{Cov}\{v(t-1) + e(t) - e(t-1), v(s-1) + e(s) - e(s-1)\}$  försvinner då  $s < t-1$ . Derivering är ett viktigt verktyg för att komma runt problemet med icke-stationära processer.

Ett vanligt specialfall av icke-stationära processer kan beskrivas som ARMAX-processer med integrerat brus:

$$A(q^{-1})y(t) = B(q^{-1})u(t) + \frac{C(q^{-1})}{\Delta}$$

vilket är ekvivalent med

$$A(q^{-1})\Delta y(t) = B(q^{-1})\Delta u(t) + C(q^{-1}) \quad (2.10)$$

Denna modell benämns ARIMAX, där I står för Integrated (Söderström & Stoica 1989).

## Säsongeffekter

Det är vanligt med säsongeffekter i processer. Solinstrålning och utetemperatur är två exempel på processer med perioden, eller "säsongen" 24 timmar. För att hantera detta kan en Seasonal ARIMA eller SARIMA-process användas. Den ges av:

$$A(q^{-1})A_s(q^{-s})(1 - q^{-1})^d(1 - q^{-s})^D y(t) = C(q^{-1})C_s(q^{-s})e(t) \quad (2.11)$$

där polynomen  $A_s(q^{-1})$  och  $C_s(q^{-1})$  beskriver utsignalens beroende av tidigare perioders utsignaler och brus. Variabeln  $s$  betecknar periodlängden. Operatorerna  $(1 - q^{-1})^d$  och  $(1 - q^{-s})^D$  betecknar differentiering  $d$ :te respektive  $D$ :te differensen med första respektive  $s$ :te elementet. En SARIMA-modell skrivs vanligen  $(p, q, d) \times (P, Q, D)$  där  $p, q, P$  och  $Q$  står för antalet element i  $A(q^{-1}), C(q^{-1}), A_s(q^{-s})$  respektive  $C_s(q^{-s})$  (Chatfield 2000).

## Prediktion

Givet polynomen  $A(q^{-1})$ ,  $B(q^{-1})$  och  $C(q^{-1})$  samt gamla in- och utsignaler och prediktionsfel kan enstegsprediktionen  $y(t+1|t)$  beräknas:

$$\begin{aligned}\hat{y}(t+1|t) = & -a_1y(t) - \dots - a_nay(t-na) + b_{1,1}u_1(t) + \\ & \dots + b_{1,nb_1}u_1(t-nb_1) + \dots + b_{nu,nb_{nu}}u_{nu}(t-nb_{nu}) \\ & + c_1\varepsilon(t) + \dots + c_{nc}\varepsilon(t-nc)\end{aligned}\quad (2.12)$$

Prediktionsfelet ges av  $\varepsilon(t) = y(t) - \hat{y}(t|t-1)$ .

För  $k$ -stegsprediktion kan följande rekursiva formel användas:

$$\begin{aligned}\hat{y}(t+k|t) + a_1\hat{y}(t+k-1|t) \dots + a_n\hat{y}(t+k-na|t) \\ = b_{1,1}u_1(t) + \dots + b_{1,nb_1}u_1(t-nb_1) + \dots + \\ b_{nu,nb_{nu}}u_{nu}(t-nb_{nu}) + c_k\varepsilon(t) + \dots + c_{nc}\varepsilon(t+k-nc)\end{aligned}$$

där

$$\hat{y}(t+i|t) = \begin{cases} \hat{y}(t+i|t) & \text{för } i > 0 \\ y(t+i) & \text{för } i \leq 0 \end{cases}$$

(Söderström & Stoica 1989)

För en process med säsongeffekter av typen (2.11) blir enstegsprediktionen följande:

$$\begin{aligned}\hat{y}'(t+1|t) = & -a_1y(t) - \dots - a_nay(t-na) - a_{1_s}y(t-s) - \\ & a_{1_s}a_1y(t-s-1) - \dots - a_{1_s}a_nay(t-s-na) - \\ & \dots - a_{na_s}a_nay(t-na_s-na) + \\ & c_1\varepsilon(t-1) + \dots + c_{nc}\varepsilon(t-nc) + \\ & c_{1_s}\varepsilon(t-s) + c_{1_s}c_1\varepsilon(t-s-1) + \dots + \\ & c_{1_s}c_{nc}\varepsilon(t-s-nc) + \dots + c_{nc_s}c_{nc}\varepsilon(t-nc_s-nc)\end{aligned}\quad (2.13)$$

där

$$y'(t) = (1-q^{-1})(1-q^{-s})y(t)\quad (2.14)$$

Den rekursiva formeln för  $k$ -stegsprediktion ovan kan modifieras:

$$\begin{aligned}\hat{y}'(t+k|t) = & -a_1\hat{y}'(t+k-1|t) \dots - a_n\hat{y}'(t+k-na|t) - \\ & a_{1_s}\hat{y}'(t+k-s|t) - a_{1_s}a_1\hat{y}'(t+k-s-1|t) - \dots - \\ & a_{1_s}a_n\hat{y}'(t+k-s-na|t) - \dots - a_{na_s}a_n\hat{y}'(t+k-na_s-na|t) - \\ & c_k\varepsilon(t) + \dots + c_{nc}\varepsilon(t+k-nc) + c_{1_s}\varepsilon(t+k-s) + \\ & c_{1_s}c_1\varepsilon(t-s-1) + \dots + \\ & c_{1_s}c_{nc}\varepsilon(t-s-nc) + \dots + c_{nc_s}c_{nc}\varepsilon(t-nc_s-nc)\end{aligned}$$

där

$$\hat{y}(t+i|t) = \begin{cases} \hat{y}(t+i|t) & \text{för } i > 0 \\ y(t+i) & \text{för } i \leq 0 \end{cases}$$

(Chatfield 2000)

## 2.2.2 Identifiering: rekursiv prediktionsfelsesmetod

Koefficienterna i  $A(q^{-1})$ -,  $B(q^{-1})$ - och  $C(q^{-1})$ -polynomen kan skattas på olika sätt. I detta fall används en rekursiv metod då identifieringen ska uppdateras vid varje sampel. En mycket vanlig metod vid identifiering är minsta kvadratmetoden som går att modifiera så att den blir rekursiv. Problemet med minsta kvadratmetoden är att den inte ger konsistenta  $\hat{\boldsymbol{\theta}}(t)$  då brusets inte är vitt, alltså då

$$E\boldsymbol{\phi}(t)v(t) \neq 0$$

där  $\boldsymbol{\phi}(t)$  betecknar regressionsvektorn och  $v(t)$  brus. För att hantera detta har bland annat den rekursiva prediktionsfelsesmetoden utvecklats. Den beskrivs och härleds i Söderström & Stoica (1989). Metoden går ut på att minimera ett kriterium genom att välja lämpliga parametervärden. Kriteriet har vanligen formen

$$V_t(\boldsymbol{\theta}) = \frac{1}{2} \sum_{s=1}^t \lambda^{t-s} \boldsymbol{\varepsilon}^T(s, \boldsymbol{\theta}) \mathbf{Q} \boldsymbol{\varepsilon}(s, \boldsymbol{\theta}) \quad (2.15)$$

där  $\mathbf{Q}$  är en positivt definit viktmatris. Om en process med endast en utsignal ska identifieras, blir prediktionsfelet skalärt (och kvadratisk) och  $\mathbf{Q}$  blir en skalär konstant. Kriteriet är i så fall oberoende av  $\mathbf{Q}$ .

Genom att anta att  $\hat{\boldsymbol{\theta}}(t-1)$  minimerar  $V_{t-1}(\boldsymbol{\theta})$  och att minimum av  $V_t(\boldsymbol{\theta})$  är nära  $\hat{\boldsymbol{\theta}}(t-1)$  är det rimligt att approximera  $V_t(\boldsymbol{\theta})$  med en andra ordningens Taylorutveckling runt  $\hat{\boldsymbol{\theta}}(t-1)$ :

$$V_t(\boldsymbol{\theta}) \approx V_t(\hat{\boldsymbol{\theta}}(t-1)) + V_t'(\hat{\boldsymbol{\theta}}(t-1))(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}(t-1)) \quad (2.16)$$

$$+ \frac{1}{2}(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}(t-1))^T V_t''(\hat{\boldsymbol{\theta}}(t-1))(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}(t-1)) \quad (2.17)$$

vilket är en kvadratisk funktion av  $\boldsymbol{\theta}(t)$ . Det nya parameterestimatet  $\hat{\boldsymbol{\theta}}(t)$  fås då av en minimering av (2.17) med avseende på  $\hat{\boldsymbol{\theta}}(t)$ :

$$\hat{\boldsymbol{\theta}}(t) = \hat{\boldsymbol{\theta}}(t-1) - (V_t''(\hat{\boldsymbol{\theta}}(t-1)))^{-1} (V_t'(\hat{\boldsymbol{\theta}}(t-1)))^T$$

vilket motsvarar ett steg med Newton-Raphsons algoritm. Från (2.15) fås följande samband:

$$\begin{aligned} V_t(\boldsymbol{\theta}) &= \lambda V_{t-1}(\boldsymbol{\theta}) + \frac{1}{2} \varepsilon^T(t, \boldsymbol{\theta}) \mathbf{Q} \varepsilon(t, \boldsymbol{\theta}) \\ V'_t(\boldsymbol{\theta}) &= \lambda V'_{t-1}(\boldsymbol{\theta}) + \varepsilon^T(t, \boldsymbol{\theta}) \mathbf{Q} \varepsilon'(t, \boldsymbol{\theta}) \\ V''_t(\boldsymbol{\theta}) &= \lambda V''_{t-1}(\boldsymbol{\theta}) + (\varepsilon'(t, \boldsymbol{\theta}))^T \mathbf{Q} \varepsilon'(t, \boldsymbol{\theta}) + \varepsilon^T(t, \boldsymbol{\theta}) \mathbf{Q} \varepsilon''(t, \boldsymbol{\theta}) \end{aligned} \quad (2.18)$$

Sambandet (2.18) gäller om  $\varepsilon$  är skalär. Eftersom  $\hat{\boldsymbol{\theta}}(t-1)$  antas vara minimum för  $V_{t-1}(\boldsymbol{\theta})$  görs följande approximation:

$$V'_{t-1}(\hat{\boldsymbol{\theta}}(t-1)) = 0$$

Vidare kan  $V''_{t-1}(\boldsymbol{\theta})$  antas variera långsamt med  $\boldsymbol{\theta}$ :

$$V''_{t-1}(\hat{\boldsymbol{\theta}}(t-1)) = V''_{t-1}(\hat{\boldsymbol{\theta}}(t-2))$$

Om  $\varepsilon(t, \boldsymbol{\theta})$  är vitt brus för den sanna  $\boldsymbol{\theta}$ -vektorn gäller:

$$\varepsilon^T(t, \boldsymbol{\theta}) \mathbf{Q} \varepsilon''(t, \boldsymbol{\theta}) \approx 0$$

Följande rekursiva algoritm kan sammanställas:

$$\begin{aligned} \hat{\boldsymbol{\theta}}(t) &= \hat{\boldsymbol{\theta}}(t-1) - (V''_t(\hat{\boldsymbol{\theta}}(t-1)))^{-1} (\varepsilon'(t, \hat{\boldsymbol{\theta}}))^T \\ &\quad \times \mathbf{Q} \varepsilon(t, \hat{\boldsymbol{\theta}}(t-1)) \end{aligned} \quad (2.19)$$

$$V''_t(\hat{\boldsymbol{\theta}}(t-1)) = \lambda V''_{t-1}(\hat{\boldsymbol{\theta}}(t-2)) + (\varepsilon'(t, \hat{\boldsymbol{\theta}}(t-1)))^T \mathbf{Q} \varepsilon'(t, \hat{\boldsymbol{\theta}}(t-1)) \quad (2.20)$$

Här finns två problem: Inversen av  $V''_t$  behövs i (2.19), men det är själva matrisen som uppdateras i (2.20). Vidare krävs beräkningar av  $\varepsilon(t, \hat{\boldsymbol{\theta}}(t-1))$  och dess derivata med data fram till  $t$  för varje  $t$ . Först definieras följande approximationer:

$$\begin{aligned} \varepsilon(t) &\approx \varepsilon(t, \hat{\boldsymbol{\theta}}(t-1)) \\ \boldsymbol{\psi}(t) &\approx -(\varepsilon'(t, \hat{\boldsymbol{\theta}}(t-1)))^T \end{aligned} \quad (2.21)$$

En smidigare notation för inversen av  $V''_t$ :

$$\mathbf{P}(t) = (V''_t(\hat{\boldsymbol{\theta}}(t-1)))^{-1} \quad (2.22)$$

Ekvationerna (2.22) med (2.19) och (2.20) ger

$$\mathbf{P}^{-1}(t) = \lambda \mathbf{P}^{-1}(t-1) + \boldsymbol{\psi}(t) \mathbf{Q} \boldsymbol{\psi}^T(t) \quad (2.23)$$



Genom att använda matrisinverteringslemmat (se appendix A kan (2.23) skrivas om:

$$\mathbf{P}(t) = \tag{2.24}$$

$$\frac{\mathbf{P}(t-1) - \mathbf{P}(t-1)\boldsymbol{\psi}(t)(\lambda\mathbf{Q}^{-1} + \boldsymbol{\psi}^T(t)\mathbf{P}(t-1)\boldsymbol{\psi}(t))^{-1}\boldsymbol{\psi}^T(t)\mathbf{P}(t-1)}{\lambda} \tag{2.25}$$

Sammanfattningsvis ser en generell rekursiv prediktionsfelsalgoritm alltså ut som följer:

$$\hat{\boldsymbol{\theta}}(t) = \hat{\boldsymbol{\theta}}(t-1) + \mathbf{K}(t)\varepsilon(t)$$

$$\mathbf{K}(t) = \mathbf{P}(t)\boldsymbol{\psi}(t)\mathbf{Q}$$

$$\mathbf{P}(t) =$$

$$\frac{\mathbf{P}(t-1) - \mathbf{P}(t-1)\boldsymbol{\psi}(t)(\lambda\mathbf{Q}^{-1} + \boldsymbol{\psi}^T(t)\mathbf{P}(t-1)\boldsymbol{\psi}(t))^{-1}\boldsymbol{\psi}^T(t)\mathbf{P}(t-1)}{\lambda}$$

Modellstrukturen bestämmer sedan hur  $\varepsilon(t)$  och  $\boldsymbol{\psi}(t)$  ska beräknas.

En modell med strukturen  $A(q^{-1})y(t) = B(q^{-1})u(t) + C(q^{-1})e(t)$  har prediktionsfelet  $\varepsilon(t, \boldsymbol{\theta})$  av

$$C(q^{-1})\varepsilon(t, \boldsymbol{\theta}) = A(q^{-1})y(t) - B(q^{-1})u(t) \tag{2.26}$$

Derivatans av (2.26) med avseende på elementen i  $A(q^{-1})$ -,  $B(q^{-1})$  och  $C(q^{-1})$ -polynomen ger

$$\begin{aligned} \frac{\partial \varepsilon(t, \boldsymbol{\theta})}{\partial a_i} &= \frac{1}{C(q^{-1})} y(t-i) \\ \frac{\partial \varepsilon(t, \boldsymbol{\theta})}{\partial b_i} &= -\frac{1}{C(q^{-1})} u(t-i) \\ \frac{\partial \varepsilon(t, \boldsymbol{\theta})}{\partial c_i} &= -\frac{1}{C(q^{-1})} \varepsilon(t-i, \boldsymbol{\theta}) \end{aligned}$$

Enligt (2.21) blir

$$\begin{aligned} \boldsymbol{\psi}(t) &= (-y^F(t-1, \boldsymbol{\theta}) \dots - y^F(t-n, \boldsymbol{\theta}) u^F(t-1, \boldsymbol{\theta}) \\ &\dots u^F(t-n, \boldsymbol{\theta}) \varepsilon^F(t-1, \boldsymbol{\theta}) \dots \varepsilon^F(t-n, \boldsymbol{\theta}))^T \end{aligned}$$

där

$$\begin{aligned}y^F(t, \boldsymbol{\theta}) &= \frac{1}{C(q^{-1})}y(t) \\u^F(t, \boldsymbol{\theta}) &= \frac{1}{C(q^{-1})}u(t) \\ \varepsilon^F(t, \boldsymbol{\theta}) &= \frac{1}{C(q^{-1})}\varepsilon(t)\end{aligned}$$

och enligt (2.26) blir

$$\varepsilon(t, \boldsymbol{\theta}) = \frac{A(q^{-1})y(t) - B(q^{-1})u(t)}{C(q^{-1})}$$

vilket medför att

$$\begin{aligned}\varepsilon(t) &= y(t) + \hat{a}_1(t-1)y(t-1) + \dots + \hat{a}_n(t-1)y(t-n) \\ &\quad - \hat{b}_1(t-1)u(t-1) - \dots - \hat{b}_n(t-1)u(t-n) \\ &\quad - \hat{c}_1(t-1)\bar{\varepsilon}(t-1) - \dots - \hat{c}_n(t-1)\bar{\varepsilon}(t-n)\end{aligned}$$

vilket är i enlighet med definitionen av prediktorn (2.12).

I ekvationerna ovan har prediktionsfelet beräknats från det senaste datat genom att använda det framräknade  $\hat{\boldsymbol{\theta}}(t)$  istället för att använda  $\hat{\boldsymbol{\theta}}(t-1)$ . Detta kan utnyttjas i beräkningen av  $\boldsymbol{\psi}(t)$ . Prediktionsfelet som beräknas utifrån  $\hat{\boldsymbol{\theta}}(t)$  benämns  $\bar{\varepsilon}(t)$ . Dock måste  $\hat{\boldsymbol{\theta}}(t)$  beräknas med  $\varepsilon(t)$ .

En rekursiv prediktionsfalsalgoritm för en ARMAX-modell med strukturen  $A(q^{-1})y(t) = B(q^{-1})u(t) + C(q^{-1})e(t)$  blir således:

$$\begin{aligned}
\boldsymbol{\psi}(t) &= (-y^F(t-1, \boldsymbol{\theta}) \dots - y^F(t-n, \boldsymbol{\theta}) \ u^F(t-1, \boldsymbol{\theta}) \dots u^F(t-n, \boldsymbol{\theta}) \\
&\quad \bar{\varepsilon}^F(t-1, \boldsymbol{\theta}) \dots \bar{\varepsilon}^F(t-n, \boldsymbol{\theta}))^T \\
\varepsilon(t) &= y(t) + \hat{a}_1(t-1)y(t-1) + \dots + \hat{a}_n(t-1)y(t-n) \\
&\quad - \hat{b}_1(t-1)u(t-1) - \dots - \hat{b}_n(t-1)u(t-n) \\
&\quad - \hat{c}_1(t-1)\bar{\varepsilon}(t-1) - \dots - \hat{c}_n(t-1)\bar{\varepsilon}(t-n) \\
\mathbf{P}(t) &= \mathbf{P}(t-1)\lambda^{-1} - \\
&\quad \mathbf{P}(t-1)\boldsymbol{\psi}(t)(\lambda\mathbf{Q}^{-1} + \boldsymbol{\psi}^T(t)\mathbf{P}(t-1)\boldsymbol{\psi}(t))^{-1}\boldsymbol{\psi}^T(t)\mathbf{P}(t-1)\lambda^{-1} \\
\hat{\boldsymbol{\theta}}(t) &= \hat{\boldsymbol{\theta}}(t-1) + \mathbf{P}(t)\boldsymbol{\psi}(t)\mathbf{Q}\varepsilon(t) \\
\bar{\varepsilon}(t) &= y(t) + \hat{a}_1(t)y(t-1) + \dots + \hat{a}_n(t)y(t-n) \\
&\quad - \hat{b}_1(t)u(t-1) - \dots - \hat{b}_n(t)u(t-n) \\
&\quad - \hat{c}_1(t)\bar{\varepsilon}(t-1) - \dots - \hat{c}_n(t)\bar{\varepsilon}(t-n) \\
y^F(t) &= y(t) - \hat{c}_1(t)y^F(t-1) - \dots - \hat{c}_n(t)y^F(t-n) \\
u^F(t) &= u(t) - \hat{c}_1(t)u^F(t-1) - \dots - \hat{c}_n(t)u^F(t-n) \\
\bar{\varepsilon}^F(t) &= \bar{\varepsilon}(t) - \hat{c}_1(t)\bar{\varepsilon}^F(t-1) - \dots - \hat{c}_n(t)\bar{\varepsilon}^F(t-n)
\end{aligned}$$

En ARIMAX-modell identifieras på samma sätt, fast signalerna  $\Delta u(t)$  och  $\Delta y(t)$  används istället för  $u(t)$  och  $y(t)$  som i (2.10).

Vid identifiering av en SARIMA-modell (se (2.11)) kan prediktionsfelet bestämmas på samma sätt som för en AR(I)MAX-modell ovan, det vill säga med hjälp av enstegsprediktionen (2.13):

$$\varepsilon(t) = y(t) - \hat{y}(t|t-1)$$

## Initialvärden

Parametervektorn  $\hat{\boldsymbol{\theta}}(0)$  och dess kovariansmatris  $\mathbf{P}(0)$  måste initieras innan identifieringen börjar. Om det finns kunskap om systemet är det lämpligt att använda den kunskapen för initialgissningen  $\hat{\boldsymbol{\theta}}(0)$  och låta  $\mathbf{P}(0)$  vara mindre ju bättre estimaten av  $\hat{\boldsymbol{\theta}}(0)$  anses vara. Ju mindre  $\mathbf{P}(0)$  är, desto mindre förändrar identifieringsalgoritmen  $\hat{\boldsymbol{\theta}}(t)$  i varje steg vid relativt små  $t$ , vilket leder till lägre brus känslighet, men också långsammare anpassning till verkliga förändringar av systemet. Då  $t \gg 0$  bestämmer  $\lambda$  och  $\boldsymbol{\psi}(t)$  storleken

på  $\mathbf{P}(t)$ . Utan någon information om systemet kan följande ansättas:

$$\begin{aligned}\hat{\boldsymbol{\theta}}(0) &= 0 \\ \mathbf{P}(0) &= \rho \mathbf{I}\end{aligned}\tag{2.27}$$

där  $\rho$  är ett tillräckligt stort tal och  $\mathbf{I}$  enhetsmatrisen (Söderström & Stoica 1989).

### Uppdatering av $\mathbf{P}(t)$

Algoritmen ovan medför att  $\mathbf{P}(t)$  alltid blir positivt definit. Om identifieringen genomförs med dator kan dock avrundningsproblem uppstå, och om algoritmen körs många gånger ackumuleras dessa vilket kan göra  $\mathbf{P}(t)$  icke positivt definit, vilket i sin tur leder till att parameterestimaten divergerar. Följande metod kan användas för att i praktiken eliminera denna risk.

Definiera  $\mathbf{S}(t)$ :

$$\mathbf{P}(t) = \mathbf{S}(t)\mathbf{S}^T(t)$$

och uppdatera  $\mathbf{S}(t)$  istället för  $\mathbf{P}(t)$ .  $\mathbf{P}(t)$  blir då positivt definit.  $\mathbf{S}(t)$  uppdateras på följande sätt:

$$\begin{aligned}\mathbf{f}(t) &= \mathbf{S}^T(t-1)\boldsymbol{\psi}(t) \\ \beta(t) &= \lambda \mathbf{Q}^{-1} + \mathbf{f}^T(t)\mathbf{f}(t) \\ \alpha(t) &= \frac{1}{\beta(t) + \sqrt{\beta(t)}} \\ \mathbf{L}(t) &= \mathbf{S}(t-1)\mathbf{f}(t) \\ \mathbf{S}(t) &= \mathbf{S}(t-1) - \alpha(t)\mathbf{L}(t)\mathbf{f}^T(t)\end{aligned}$$

vilket leder till att  $\hat{\boldsymbol{\theta}}(t)$  uppdateras som följer:

$$\hat{\boldsymbol{\theta}}(t) = \hat{\boldsymbol{\theta}}(t-1) + \mathbf{L}(t)\frac{\varepsilon(t)}{\beta(t)}$$

Denna metod kräver endast en division, istället för  $n\boldsymbol{\theta}$  divisioner vilket leder till färre avrundningar och därmed minskad risk för numeriska problem (Söderström & Stoica 1989).

## Glömskefaktorn

Glömskefaktorn  $\lambda$  i (2.15) kan användas för att bestämma hur känslig identifieringen ska vara för nya data. I (2.15) syns att då  $\lambda = 1$  viktas alla observationer lika, och ju mindre  $\lambda$  är, desto mindre betydelse får gamla observationer då  $t$  ökar. Ändringarna av koefficienterna i  $\hat{\theta}$  blir större i varje steg och de går mot sina sanna värden snabbare, men brus känsligheten ökar också då  $\lambda < 1$ .

Det är vanligt att låta  $\lambda$  gå exponentiellt mot 1 då det kan vara fördelaktigt att låta  $\hat{\theta}$  ändra sig mycket i början för att minimera transienteffekter som beror på initialvärden som ligger långt ifrån de verkliga värdena. Det kan göras enligt följande:

$$\lambda(t) = \lambda_0 \lambda(t-1) + (1 - \lambda_0)$$

Vanliga värden är  $\lambda_0 = 0.99$  och  $\lambda(0) = 0.95$  (Söderström & Stoica 1989).

## Krav på insignalen

Det är lätt att inse att insignalen är mycket viktig för möjligheten att identifiera ett system. Om  $u(t) \equiv 0$  inses lätt att ingen information om systemets svar på insignalen kan fås. Insignalen måste excitera systemet så att dess viktiga egenskaper kommer fram och kan bygga upp modellen (Ljung & Glad 2004).

Termen ”persistent excitation” (pe) av ordningen  $n$  används för att beskriva hur väl en insignal exciterar ett system. En insignal  $u(t)$  är pe av ordningen  $n$  om dess kovariansmatris av dimension  $n \times n$  är positivt definit. Det är ofta nödvändigt att insignalen  $u(t)$  är pe av ordning  $2n$  då ett  $n$ :te ordningens system ska identifieras (Söderström & Stoica 1989).

## Identifiering i återkopplat system

I återkopplade system är utsignalen beroende av insignalen vilket kan ställa till problem vid identifiering. Om ingen hänsyn tas till återkopplingen kan systemet dock ofta identifieras ändå om återkopplingen är av tillräckligt hög ordning. Ytterligare insignaler som inte är beroende av utsignalen ökar också möjligheten att identifiera systemet. (Söderström & Stoica 1989)

## Estimatoruppvridning

Ett fenomen med nära koppling till excitering av systemet och användning av glömskefaktor är estimatoruppvridning. Anta extremfallet att systemet inte exciteras alls, det vill säga att  $\boldsymbol{\psi}(t) = 0$ . Uppdateringen av  $\mathbf{P}$  blir då enligt (2.25):

$$\mathbf{P}(t+1) = \frac{1}{\lambda}\mathbf{P}(t)$$

Om glömskefaktorn  $\lambda < 1$  kommer  $\mathbf{P}$  att växa exponentiellt. Uppdateringen av  $\hat{\boldsymbol{\theta}}$  är  $\mathbf{P}\boldsymbol{\psi}_\varepsilon$  vilket innebär att  $\hat{\boldsymbol{\theta}}(t)$  kommer att ändras väldigt kraftigt då  $\boldsymbol{\psi}(t) \neq 0$ . Motsvarande resonemang gäller till exempel om  $\boldsymbol{\psi}(t)$  endast innehåller brus eller signaler som inte exciterar systemet ordentligt, till exempel gamla insignaler.

För att motverka detta finns flera metoder. Villkorlig uppdatering kan användas för att endast uppdatera  $\hat{\boldsymbol{\theta}}(t)$  om  $\boldsymbol{\psi}(t)$  är tillräckligt exciterad, för att på så sätt begränsa  $\mathbf{P}(t)$ . En korrekt analys av tillräcklig excitation ska ta hänsyn till spektra och kovarianser, men enklare villkor är vanligt förekommande, till exempel:

$$\boldsymbol{\psi}^T(t)\mathbf{P}(t)\boldsymbol{\psi}(t) > 2(1 - \lambda)$$

(Åström & Wittenmark 1995)

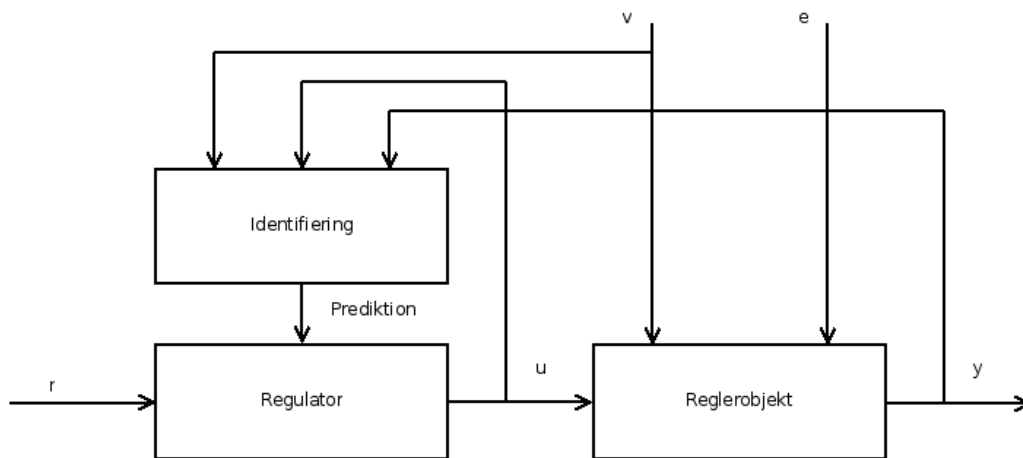
## 2.3 Generalized Predictive Control

Modeller av system är lämpliga att använda för att förutsäga systemets framtida beteende. Det faller sig då naturligt att med hjälp av modellen beräkna förväntade framtida värden på reglerstorheterna som funktion av de styr-signaler som är möjliga. Sedan väljs den styrsignal som minimerar en kriteriefunktion som kan bestå av exempelvis reglerfelet och insignalens storlek. Resultatet blir en form av tillståndsåterkoppling i och med att regleringen använder tillstånden (eller skattningar av dem) vid återkopplingen. Metoden kallas Model Predictive Control (MPC) då modellen är statisk, men om modellen skattas samtidigt som regleringen sker, benämns den Generalized Predictive Control (GPC), se figur 2.2 för blockschema (Boucher et al. 1996).

En stor fördel med GPC är att det är relativt lätt att hantera bivillkor med lämpliga optimeringsmetoder. Vanliga bivillkor är mättning av någon signal eller något tillstånd, till exempel att en radiator endast kan leverera effekt

i intervallet 0-1000 W eller att en ventil endast kan vara 0 %-100 % öppen. Bivillkor leder till att regulatorn blir olinjär även om systemet är linjärt (Ljung & Glad 2003).

GPC-metodiken är robust vid både under- och överparametrisering, det vill säga då för låg respektive för hög modellordning valts. Även tidsfördröjningar kan hanteras givet att  $nb$  och samplingsperioden väljs så att en förändring i signalen har en chans att registreras av regulatorn (Clarke et al. 1987).

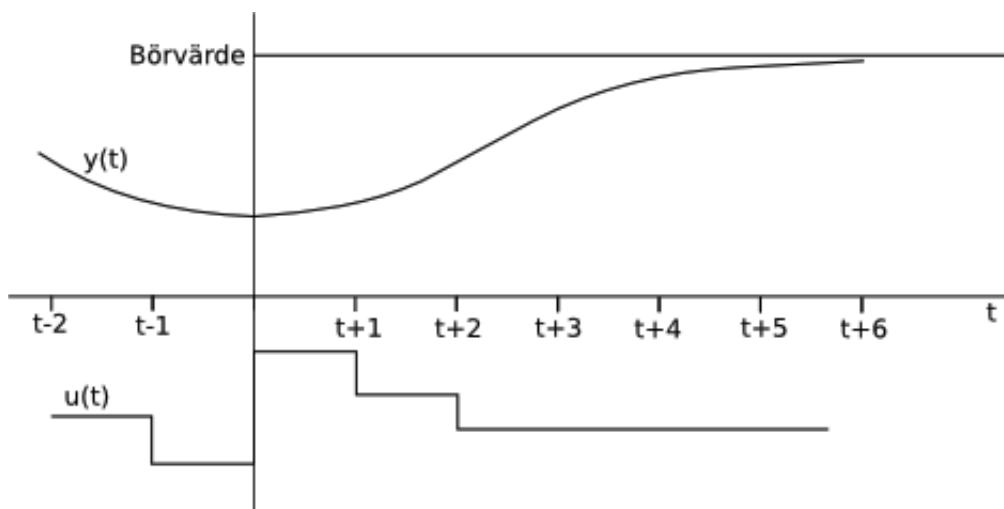


Figur 2.2: Blockdiagram GPC-reglering.

### 2.3.1 Styrsignals- och prediktionshorisonter

Beräkningen av de optimala insignalerna är krävande, och därför är det vanligt att beräkna dessa för relativt få steg framåt, medan prediktionen av framtida utsignaler sträcker sig längre bort i tiden. Dessa olika steglängder benämns styrsignalhorisont med längden  $N_u$  respektive prediktionshorisont med längden  $N_y$ . För  $N_u < j \leq N_y$  är det vanligt att anta att  $\Delta u(t) = 0$ . Se figur 2.3. Prediktionshorisonten väljs vanligen så att den täcker in systemets insvängningstid.

Återkoppling med GPC sker enligt receding horizon-principen, vilket betyder att även om beräknade optimala insignaler som täcker in hela styrsignalhorisonten tas fram, används endast den första. I nästa tidssteg beräknas nya styrsignaler utifrån nya prediktioner (Ljung & Glad 2003).



Figur 2.3: Prediktion av framtida utsignaler ger optimala styrsignaler.

### 2.3.2 Prediktion på tillståndsform

Det skattade systemet kan med hjälp av polynomen  $A(q^{-1})$ ,  $B(q^{-1})$  och  $C(q^{-1})$  formuleras på tillståndsform för att skapa en prediktor. Då prediktorn ska förutsäga de faktiska utsignalerna  $y(t+j)$  och inte  $\Delta y(t+j)$  används polynomet  $\tilde{A}(q^{-1}) = \Delta A(q^{-1})$  istället. Tillståndsmodellen ges av:

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{A}\mathbf{x}(k) + \mathbf{B}\Delta u(k) + \mathbf{B}_\varepsilon \varepsilon(k) \\ y(k) &= \mathbf{C}\mathbf{x}(k) + \varepsilon(k) \end{aligned} \quad (2.28)$$

vilket på observerbar kanonisk form ger följande matriser:

$$\mathbf{A} = \begin{pmatrix} -\tilde{a}_1 & 1 & 0 & \cdots & 0 \\ -\tilde{a}_2 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -\tilde{a}_{n-1} & 0 & 0 & \cdots & 1 \\ -\tilde{a}_n & 0 & 0 & \cdots & 0 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

$$\mathbf{C} = [1 \quad 0 \quad \cdots \quad 0], \quad \mathbf{B}_\varepsilon = \begin{pmatrix} c_1 - \tilde{a}_1 \\ c_2 - \tilde{a}_2 \\ \vdots \\ c_n - \tilde{a}_n \end{pmatrix}$$



Tillståndsvektorn har dimensionen  $n = \max(n_a + 1, n_b + 1, n_c)$ . En  $j$ -stegsprediktor blir

$$\hat{\mathbf{x}}(k + j + 1) = \mathbf{A}\hat{\mathbf{x}}(k + j) + \mathbf{B}\Delta u(k + j) \quad (2.29)$$

$$\hat{y}(k + j) = \mathbf{C}\hat{\mathbf{x}}(k + j) \quad (2.30)$$

om väntevärdet av framtida brussignaler är 0. Alla prediktioner fram till  $t + j|t$  kan skrivas

$$\hat{\mathbf{Y}} = \mathbf{S}\mathbf{U} + \mathbf{F} \quad (2.31)$$

där

$$\hat{\mathbf{Y}} = \begin{pmatrix} \hat{y}(t + N_y|t) \\ \hat{y}(t + N_y - 1|t) \\ \vdots \\ \hat{y}(t + 1|t) \end{pmatrix}$$

$$\mathbf{U} = \begin{pmatrix} \Delta u(t + N_u) \\ \Delta u(t + N_u - 1) \\ \vdots \\ \Delta u(t) \end{pmatrix}$$

där  $\mathbf{S}\mathbf{U}$  motsvarar den del av de framtida utsignalerna som kan påverkas genom att välja  $\mathbf{U}$ . Genom att använda (2.29) och (2.30) fås

$$\mathbf{S} = \begin{pmatrix} \mathbf{C}\mathbf{A}^{N_y - N_u}\mathbf{B} & \mathbf{C}\mathbf{A}^{N_y - N_u + 1}\mathbf{B} & \dots & \mathbf{C}\mathbf{A}^{N_y - 1}\mathbf{B} \\ \mathbf{C}\mathbf{A}^{N_y - N_u - 1}\mathbf{B} & \mathbf{C}\mathbf{A}^{N_y - N_u}\mathbf{B} & \dots & \mathbf{C}\mathbf{A}^{N_y - 2}\mathbf{B} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{C}\mathbf{B} & \mathbf{C}\mathbf{A}\mathbf{B} & \dots & \mathbf{C}\mathbf{A}^{N_u - 1}\mathbf{B} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & \mathbf{C}\mathbf{B} \end{pmatrix}$$

och

$$\mathbf{F} = \mathbf{H}\mathbf{x}(t) = \begin{pmatrix} \mathbf{C}\mathbf{A}^{N_y} \\ \mathbf{C}\mathbf{A}^{N_y - 1} \\ \vdots \\ \mathbf{C}\mathbf{A} \end{pmatrix}$$

(Park et al. 2007)

Om det finns skattade framtida störningar kan  $j$ -stegsprediktorn på tillståndsform (2.28) skrivas som

$$\hat{\mathbf{x}}(k + j + 1) = \mathbf{A}\mathbf{x}(k + j) + \mathbf{B}\Delta u(k + j) + \mathbf{N}\Delta \mathbf{v}(k + j) + \mathbf{B}_\varepsilon \varepsilon(k + j)$$

Väntevärdet av framtida prediktionsfel är 0, så  $\varepsilon(k+j) = 0$  där  $j > 0$ .

Ekvationen (2.31) kan då skrivas

$$\hat{\mathbf{Y}} = \mathbf{S}_u \mathbf{U} + \mathbf{S}_v \mathbf{V} + \mathbf{F}$$

där  $\mathbf{V}$  är uppbyggd som  $\mathbf{U}$  men innehåller differenser av framtida skattade störningar.  $\mathbf{S}_v$  beräknas då på liknande sätt som  $\mathbf{S}_u$ , fast störningsmatrisen  $\mathbf{N}$ , som innehåller de parametrar i  $\hat{\boldsymbol{\theta}}(t)$  som beskriver mätta störningars påverkan på systemet, används istället för  $\mathbf{B}$ .

$$\mathbf{N} = \begin{pmatrix} b_{1,v1} & b_{1,v2} & \cdots & b_{1,n_v} \\ b_{2,v1} & \cdots & & \\ \vdots & \vdots & \vdots & \vdots \\ b_{n_b,v1} & & \cdots & b_{n_b,n_v} \end{pmatrix}$$

$$\mathbf{V} = \begin{pmatrix} \Delta v_1(t+N_v) & \Delta v_2(t+N_v) & \cdots & \Delta v_{n_v}(t+N_v) \\ \Delta v_1(t+N_v-1) & \Delta v_2(t+N_v-1) & \cdots & \Delta v_{n_v}(t+N_v-1) \\ \vdots & \vdots & \vdots & \vdots \\ \Delta v_1(t) & \Delta v_2(t) & \cdots & \Delta v_{n_v}(t) \end{pmatrix}$$

### 2.3.3 Optimering med hänsyn till bivillkor

För att få framtida utsignaler att gå mot ett börvärde (betecknat  $r(t)$ ) kan utsignalerna formuleras som en konvex funktion som medför att utsignalerna går mot börvärdet om funktionen minimeras:

$$J = \sum_{j=1}^{N_y} (\hat{y}(t+1) - r(t+j))^2 + q_u \sum_{j=1}^{N_u} \Delta \mathbf{u}^2(t+j) + \sum_{j=1}^{N_v} \Delta \mathbf{v}^2(t+j)$$

där

$$\Delta u(t+j) = 0 \quad \text{för } j \geq N_u$$

och

$$\Delta \mathbf{v}(t+j) = 0 \quad \text{för } j \geq N_v$$

där  $N_v$  är prediktionshorisonten för störningsprediktionen. Observera att  $N_v \leq N_y$ . Funktionen kan formuleras på matrisform med referenssignalvektorn  $\mathbf{R}$  för att hantera nollskilda referenssignaler ( $J$  blir skalär):

$$J = (\mathbf{S}_u \mathbf{U} + \mathbf{S}_v \mathbf{V} + \mathbf{Hx}(t) - \mathbf{R})^T (\mathbf{S}_u \mathbf{U} + \mathbf{S}_v \mathbf{V} + \mathbf{Hx}(t) - \mathbf{R}) + \mathbf{Q}_u \mathbf{U}^T \mathbf{U} \quad (2.32)$$

där

$$\mathbf{Q}_u = \begin{pmatrix} q_u & 0 & \cdots & 0 \\ 0 & \ddots & & \vdots \\ \vdots & & & \\ 0 & \cdots & & q_u \end{pmatrix}$$

Ofta finns det bivillkor i  $\Delta u$ . Hänsyn till bivillkor i  $u$  och  $y$  kan också tas genom att formulera dem i  $\Delta u$ . Vanliga exempel är maximum- och minimum-niåver i styrsignaler på grund av effektbegränsningar i till exempel motorer eller begränsningar i ventillägen. En mycket använd och effektiv metod för att minimera (2.32) med hänsyn till bivillkor är kvadratisk programmering. Den metoden kan minimera en funktion på formen

$$J = \frac{1}{2} \mathbf{U}^T \mathbf{Q} \mathbf{U} + \mathbf{c}^T \mathbf{U} \quad (2.33)$$

$$\text{med hänsyn till: } \mathbf{A}_u \mathbf{U} \leq \mathbf{b} \quad (2.34)$$

(Boucher et al. 1996)

Omskrivning av (2.32) på formen (2.33) ger (termer utan  $\mathbf{U}$  försvinner):

$$\begin{aligned} & (\mathbf{H}\mathbf{x}(t) + \mathbf{S}_u \mathbf{U} + \mathbf{S}_v \mathbf{V} - \mathbf{R})^T (\mathbf{H}\mathbf{x}(t) + \mathbf{S}_u \mathbf{U} + \mathbf{S}_v \mathbf{V} - \mathbf{R}) + \mathbf{Q}_u \mathbf{U}^T \mathbf{U} = \\ & \mathbf{U}^T (\mathbf{S}_u^T \mathbf{S}_u + \mathbf{Q}_u) \mathbf{U} + (\mathbf{S}_u^T \mathbf{H}\mathbf{x}(t) + \mathbf{S}_u^T \mathbf{S}_v \mathbf{V} - \mathbf{S}_u^T \mathbf{R})^T \mathbf{U} \end{aligned} \quad (2.35)$$

Insättning av (2.35) i (2.33) ger

$$\min_{\mathbf{U}} \frac{1}{2} \mathbf{U}^T (\mathbf{S}_u^T \mathbf{S}_u + \mathbf{Q}_u) \mathbf{U} + (\mathbf{S}_u^T \mathbf{H}\mathbf{x}(t) + \mathbf{S}_u^T \mathbf{S}_v \mathbf{V} - \mathbf{S}_u^T \mathbf{R})^T \mathbf{U} \quad (2.36)$$

## Viktmatriser

Det är vanligt att inte bara straffa reglerfelet  $y - r$ , utan också storleken på insignalinkrementen. Det görs genom att sätta  $q_u > 0$  i ekvationerna ovan. Att straffa insignalinkrementen ökar också stabiliteten eftersom systemet blir långsammare.

## Bivillkor

I kvadratisk programmering skrivs bivillkoren som  $\mathbf{A}_u \mathbf{U} \leq \mathbf{b}$ . För att begränsa  $\Delta u(t)$  till  $\Delta u_{\min}(t+j) \leq \Delta u(t+j) \leq \Delta u_{\max}(t+j)$  väljs  $\mathbf{A}_u$  i (2.34)

till

$$\begin{pmatrix} \mathbf{I} \\ -\mathbf{I} \end{pmatrix}$$

där  $\mathbf{I}$  betecknar enhetsmatrisen. Det är dock vanligare med begränsningar på  $u(t+j)$  snarare än dess derivata. Detta görs enligt följande:

$$\mathbf{A}_u = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 0 & 1 & \ddots & 1 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & 1 \\ -1 & -1 & \cdots & -1 \\ 0 & -1 & \ddots & -1 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & -1 \end{pmatrix}$$

$$\mathbf{b} = \begin{pmatrix} u(t) - u_{\max}(t + N_u) \\ \vdots \\ u(t) - u_{\max}(t + 1) \\ -u(t) + u_{\min}(t + N_u) \\ \vdots \\ -u(t) + u_{\min}(t + 1) \end{pmatrix}$$

(Boucher et al. 1996)

### 2.3.4 Active Set-metoden

En effektiv metod för kvadratisk programmering är Active Set-metoden som beskrivs i Lundgren et al. (2001). Principen för den bygger på att bestämma vilka bivillkor som ska vara uppfyllda med likhet, det vill säga vara aktiva, i en iteration. Sedan kontrolleras om lösningen uppfyller de så kallade Karush-Kuhn-Tucker-villkoren (KKT-villkoren). De måste vara uppfyllda för att en lösning till en icke-linjär funktion ska vara optimal. Är de inte uppfyllda bestäms en ny sökdirkning, steglängd och uppsättning aktiva bivillkor.

Mängden  $\mathbf{S}$  definierar vilka bivillkor som är aktiva i punkten  $\mathbf{U}^{(k)}$  där  $k$  står för den aktuella iterationen:

$$\mathbf{S} = \{i | a_i \mathbf{U}^{(k)} = \mathbf{b}_i\}$$

där  $a_i$  betecknar rad  $i$  i bivillkorsmatrisen  $\mathbf{A}_u$ . Matrisen  $\mathbf{A}_S$  får sedan bestå av de rader i  $\mathbf{A}_u$  som motsvarar de aktiva bivillkoren. Multiplikatorerna för dessa villkor betecknas  $\mathbf{v}_S$  och högerledet som motsvarar de aktiva bivillkoren betecknas  $\mathbf{b}_S$  vilket innebär att följande samband gäller i den tillåtna punkten  $\mathbf{U}_t^{(k)}$ :

$$\mathbf{A}_S \mathbf{U}^{(k)} = \mathbf{b}_S$$

Differensen  $\mathbf{d} = \mathbf{U} - \mathbf{U}^{(k)}$  betecknar en sökriktning från punkten  $\mathbf{U}^{(k)}$  och varje punkt som fås genom en förflyttning i den riktningen måste uppfylla villkoren

$$\mathbf{A}_S(\mathbf{U}^{(k)} + \mathbf{d}) = \mathbf{b}_S$$

vilket kan skrivas  $\mathbf{A}_S \mathbf{d} = 0$  eftersom  $\mathbf{A}_S \mathbf{U}^{(k)} = \mathbf{b}_S$ . Vid en förflyttning i riktningen  $\mathbf{d}$  måste följande gälla:

$$-\mathbf{Q}(\mathbf{U}^{(k)} + \mathbf{d}) + \mathbf{A}_S^T \mathbf{v}_S = \mathbf{c} \Leftrightarrow -\mathbf{Q}\mathbf{d} + \mathbf{A}_S^T \mathbf{v}_S = \mathbf{Q}\mathbf{U}^{(k)} + \mathbf{c}$$

KKT-villkoren uttryckt i de obekanta vektorerna  $\mathbf{d}$  och  $\mathbf{v}_S$  kan skrivas som ett linjärt ekvationssystem:

$$\begin{pmatrix} -\mathbf{Q} & \mathbf{A}_S^T \\ \mathbf{A}_S & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{d} \\ \mathbf{v}_S \end{pmatrix} = \begin{pmatrix} \mathbf{Q}\mathbf{U}^{(k)} + \mathbf{c} \\ \mathbf{0} \end{pmatrix} \quad (2.37)$$

där

$$\begin{aligned} \mathbf{Q} &= \mathbf{S}_u^T \mathbf{S}_u + \mathbf{Q}_u \\ \mathbf{c}^T &= \mathbf{S}_u^T \mathbf{H}\mathbf{x}(t) + \mathbf{S}_u^T \mathbf{S}_v \mathbf{V} - \mathbf{S}_u^T \mathbf{R} \end{aligned}$$

enligt (2.33) och (2.36) (Lundgren et al. 2001).

---

**Algorithm 2.1** Sammanfattning Active Set-algoritmen

---

Välj en noggrannhet  $\epsilon$

Utgå från en tillåten startpunkt  $\mathbf{U}^0$

Initiera mängden  $\mathbf{S}$  med de aktiva bivillkoren i  $\mathbf{U}^0$

**loop**

Bestäm en lösning som uppfyller KKT-villkoren genom att lösa ut  $\mathbf{d}^{(k)}$  och  $\mathbf{v}_S^{(k)}$  i ekvation (2.37)

Avbrottskriterium:

**if**  $\|\mathbf{d}^{(k)}\| \leq \epsilon$  och, om det finns aktiva bivillkor,  $\max \mathbf{v}_S^{(k)} \leq 0$  **then**

$\mathbf{U}^{(k)}$  är optimal, avbryt loopen

**end if**

**if**  $\|\mathbf{d}^{(k)}\| \leq \epsilon$  och någon komponent i  $\mathbf{v}_S^{(k)} > 0$  **then**

Välj ett bivillkor i  $i \in \mathbf{S}$  där  $v_i > 0$  och tag bort det bivillkoret från den aktiva mängden  $\mathbf{S}$ .

**else if**  $\|\mathbf{d}^{(k)}\| > \epsilon$  **then**

Bestäm maximal steglängd  $t$  i riktningen  $\mathbf{d}^{(k)}$ , beteckna den  $t^{(k)}$ .

Beräkna en ny punkt enligt  $\mathbf{U}^{(k+1)} = \mathbf{U}^{(k)} + t^{(k)}\mathbf{d}^{(k)}$  och uppdatera mängden av aktiva bivillkor i punkten  $\mathbf{U}^{(k+1)}$ .

**end if**

**end loop**

---

### 2.3.5 Sammanfattning GPC-algoritmen

GPC kan sammanfattas i en algoritm där varje del innehåller en rad avvägningar och bedömningar som på olika sätt påverkar regleringen av systemet, vilket kommer att visa sig i resultatdelen.

---

**Algorithm 2.2** Sammanfattning GPC-algoritmen

---

Identifiera eventuellt det öppna systemet

**loop**

Samla in data och förbehandla

Uppdatera skattningen av systemet med nya data och parametrisera på tillståndsform

Predicera genom att använda (2.31)

Hitta optimala styrsignaler genom att använda algoritm 2.1

Applicera styrsignaler

**end loop**

---

## 2.4 Simulering

Istället för att testa en regulator i verkligheten, kan den testas på en modell av verkligheten. Tillståndsmodeller i kontinuerlig tid används ofta för att beskriva system och ser ut enligt följande:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) \\ y(t) &= \mathbf{C}\mathbf{x}(t)\end{aligned}$$

Fysikaliska samband och mätningar kan ligga till grund för modellen. Detta betyder att den oftast är formulerad i kontinuerlig tid. En regulator som är implementerad på dator arbetar dock i diskret tid vilket innebär att den kontinuerliga modellen måste samplas om den ska testas mot en regulator. Modellen ger då endast information vid samplingsögonblicken. Systemets insignaler antas vidare vara styckvis konstanta mellan samplingarna. Det sampelade systemet ges av:

$$\begin{aligned}\mathbf{x}(kT + T) &= \mathbf{F}\mathbf{x}(kT) + \mathbf{G}u(kT) \\ y(kT) &= \mathbf{H}\mathbf{x}(kT)\end{aligned}$$

där

$$\begin{aligned}\mathbf{F} &= e^{\mathbf{A}T}, \quad \mathbf{G} = \int_0^T e^{\mathbf{A}t}\mathbf{B}dt, \quad \mathbf{H} = \mathbf{C} \\ e^{\mathbf{A}t} &= L^{-1}(s\mathbf{I} - \mathbf{A})^{-1}\end{aligned}$$

där  $L^{-1}$  betecknar invers laplacetransform. Det finns många numeriskt effektiva metoder för ovanstående procedur, men rent praktiskt är det enklaste att använda kommandot `c2d` i Matlab.

# Kapitel 3

## Metod

### 3.1 Implementering

Implementering har gjorts i C så att en framtida implementering i en mikrokontroller kan genomföras utan att behöva skriva om koden alltför mycket. Ett alternativ till C hade varit Assembler, men C är ett bättre val för mer omfattande applikationer samt lättare att programmera i. Java och andra högnivåspråk har inte varit något alternativ då en mikrokontroller inte har tillräckligt med prestanda för det.

Vid implementering i mikrokontroller hämtar regulatorprogrammet insignaler från sensorer och ger styrsignaler till styrenheter via någon typ av anslutning. I detta examensarbete har regulatorprogrammet istället använt textfiler för att kunna läsa in till exempel utetemperaturer och spara styr- och utsignaler för senare analys.

#### 3.1.1 Programstruktur

Huvudprogrammet är en implementering av algoritm 2.2. För att göra källkoden mer överskådlig har algoritmen delats upp i olika funktioner. Detta underlättar även tester av delar av programmet, till exempel kan funktionen som utför kvadratisk programmering prövas för sig för att kontrollera att den ger korrekt resultat, innan den implementeras.



I den prototypversion som skapats i examensarbetet, har även en samplad modell av ett hus implementerats, vilket diskuteras längre fram. Modellen simulerar ett verkligt system och styrs genom att ta in styrsignaler från regulatorn och regulatorn har även tillgång till dess genererade utsignaler.

### 3.1.2 Hjälpfunktioner och datastrukturer

Till skillnad från till exempel Matlab har C inget stöd för bland annat matrisberäkningar utan sådant måste implementeras. För att enkelt kunna räkna med matriser har en struktur (C-term: struct) som beskriver en matris implementerats. Strukturen har ett fält med flyttal för att lagra matriselement samt två heltalsvariabler för att lagra antalet rader och kolumner. Detta möjliggör adressering av ett matriselement genom att ange dess rad- och kolumnindex med hjälp av funktionen

```
double  
getIndex(int row,int column,Matrix *A_ptr)
```

som returnerar det önskade elementet. Andra viktiga funktioner som implementerats som underlättar vid matrisberäkningar är matrismultiplikation, LUP-faktorisering för lösning av linjära ekvationssystem (se till exempel Heath (2005)), elementvis addition och subtraktion, oändlighetsnormen av en matris med mera. Dessa hjälpfunktioner kan användas av alla funktioner. Kvadratisk programmering, identifiering av ARIMAX-processer är också uppdelade i funktioner och anropas av huvudprogrammet.

---

**Program 1** Definition av matrisstrukturen.

---

```
typedef struct MATRIX {  
int rows; // Antal rader  
int cols; // Antal kolumner  
double *matr_ptr;  
} MATRIX;
```

---

### 3.1.3 Minnesläckor i programmet

C har till skillnad från bland annat Java och andra högnivåspråk ingen automatisk återställning av allokerat minne efter att en variabel eller en data-

struktur slutat användas. Om inte minnet avallokeras uppstår en minnesläcka vilket gör att minnet tar slut förr eller senare. Det finns inte heller någon mekanism som uppmärksammar försök att skriva till icke-allokerat minne och liknande.

Eftersom mjukvaran ska implementeras i en mikrokontroller med små minnesresurser är det extra viktigt att minneshanteringen fungerar väl. Den fria mjukvaran Valgrind ([www.valgrind.org](http://www.valgrind.org)) har använts för att kontrollera detta. Testkörningar har kunnat avslöja minnesläckor och skrivning och läsning till och från icke allokerat minne så att sådant kunnat åtgärdas.

## 3.2 Test av regulatorn

För att undersöka om en regulator har önskade egenskaper kan den testas i verkligheten. Detta kan vara svårgenomförbart av olika skäl, och därför testas regulatorer ofta i modeller av verkligheten. Då det inom ramen för detta examensarbete inte funnits praktisk möjlighet att implementera regulatorn i en mikrokontroller har det senare alternativet valts.

### 3.2.1 Husmodell

Modellen som regulatorn testas med behöver inte vara en perfekt modell av ett specifikt hus. Modellens syfte är att ge rimliga svar på verklighetstrogna styr- och stör signaler så att regulatorn kan testas på ett relativt realistiskt sätt. Utetemperatur och solinstrålning som används i simuleringarna är uppmätta, verkliga värden som kommer från ett tidigare projekt.

#### Lågenergihus

Lågenergihus har ofta en öppen planlösning där värmekällan sitter. Andra rum, till exempel sovrum, håller då en lägre temperatur då de endast värms av inomhusluften och eventuell kroppsvärme. Huset har den mesta fönsterytan åt söder för att maximera instrålningen av solenergi och undvika värmeförlust genom fönstren åt mindre soliga väderstreck. Ett lågenergihus kan därför approximativt modelleras som ett rum med ett fönster åt söder.

En sådan rummodell ges i Soleimani-Mohseni (2005) och har tagits fram genom teoretisk modellering med hjälp av energibalanskvationer.

### Beskrivning av modellen

I modellen är rummets dimensioner  $4 \times 3 \times 3$  m vilket ger en golvyta på  $12 \text{ m}^2$ . Volymen är  $36 \text{ m}^3$ . Väggarna är 20 cm tjocka, taket 10 cm och golvet 30 cm. Innerväggarna är av betong, mellanväggen har isolering av mineralull och ytterväggen består av tegel. Taket och golvet antas i modellen för enkelhets skull vara perfekt isolerade, inget värmeutbyte sker alltså där (Soleimani-Mohseni 2005).

### Modifiering av modellen

Modellen kan lätt modifieras så att även solinstrålning är insignal till modellen. En del av solinstrålningen reflekteras av fönstret och en del av energin stannar i fönstret och värmer upp det, därför kan solinstrålningens effekt skrivas enligt följande:

$$\phi_s(t) = I(t)A_e$$

där  $I(t)$  är uppmätt solinstrålning och  $A_e$  kan ses som den del av fönstret som släpper in solenergin.  $A_e$  kan till exempel vara 60% av den totala fönsterarean. Solinstrålningen antas påverka både rumstemperaturen och en del av ytterväggen (Andersen et al. 2000).

I den modifierade modellen antas rummet ha en effektiv fönsterarea på  $1 \text{ m}^2$ .

### Andra störningar

Andra viktiga störningar i hus där människor vistas är värme från boende och apparater. Varje människa avger värmeenergi motsvarande cirka 100 W om han eller hon inte utför hårt fysiskt arbete. En dator bidrar också med cirka 100 W till uppvärmning. Denna värme antas komma enbart luften till del. Denna typ av störningar är ofta svåra att mäta. Dessa benämns i fortsättningen stegstörningar.

## Slutlig modell

Genom att använda energibalansen för varje del av byggnaden kan systemet skrivas på tillståndsform med rumstemperatur och temperaturer i inner-, mellan- och yttervägg som tillstånd. Radiatoreffekt utgör styrsignal, utetemperatur och solinstrålning är stör signaler som kan mätas och värmeeffekt på grund av aktivitet är en stör signal som i detta fall inte kan mätas. Temperaturen på inkommande ventilationsluft antas vara konstant oavsett utetemperatur. Numeriskt blir resultatet:

$$\begin{pmatrix} \dot{T}_r \\ \dot{T}_c \\ \dot{T}_{mw} \\ \dot{T}_b \end{pmatrix} = \begin{pmatrix} -6.65 \times 10^{-3} & 6.32 \times 10^{-3} & 0 & 0 \\ 2.99 \times 10^{-5} & -3.23 \times 10^{-5} & 2.43 \times 10^{-6} & 0 \\ 0 & 9.15 \times 10^{-5} & -1.90 \times 10^{-4} & 9.89 \times 10^{-5} \\ 0 & 0 & 2.98 \times 10^{-6} & -5.35 \times 10^{-5} \end{pmatrix} \begin{pmatrix} T_r(t) \\ T_c(t) \\ T_{mw}(t) \\ T_b(t) \end{pmatrix} + \begin{pmatrix} 2.31 \times 10^{-5} & 0 & 3.33 \times 10^{-4} & 2.31 \times 10^{-5} & 2.31 \times 10^{-5} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 5.05 \times 10^{-5} & 0 & 0 & 4.422 \times 10^{-6} \end{pmatrix} \begin{pmatrix} \dot{Q} \\ T_o \\ T_i \\ \dot{Q}_a \\ \phi_s \end{pmatrix}$$

$$T_r = [1 \quad 0 \quad 0 \quad 0] \begin{pmatrix} T_r \\ T_c \\ T_{mw} \\ T_b \end{pmatrix}$$

Se appendix C för variabler och förklaringar.

Enligt det tidigare resonemanget om sampling så diskretiseras systemet med en samplingsperiod på 10 minuter innan det implementeras som en rutin regulatorprogrammet. Regulatorn kan ta emot rumstemperaturen, utetemperaturen och solinstrålningen som insignaler, och identifierar och bestämmer utifrån detta en lämplig utsignal utifrån den skattade modellen av systemet. Regulatorn tar inte emot information om temperaturerna i väggarna.

Samplat system:

$$\begin{pmatrix} T_r(t+1) \\ T_c(t+1) \\ T_{mw}(t+1) \\ T_b(t+1) \end{pmatrix} = \begin{pmatrix} 2.2359 \times 10^{-2} & 9.2749 \times 10^{-1} & 9.9328 \times 10^{-4} & 2.4459 \times 10^{-5} \\ 4.3879 \times 10^{-3} & 9.9355 \times 10^{-1} & 1.3719 \times 10^{-3} & 4.1073 \times 10^{-5} \\ 1.7695 \times 10^{-4} & 5.1658 \times 10^{-2} & 8.9234 \times 10^{-1} & 5.5177 \times 10^{-2} \\ 1.3129 \times 10^{-7} & 4.6600 \times 10^{-5} & 1.6626 \times 10^{-3} & 9.6846 \times 10^{-1} \end{pmatrix} \begin{pmatrix} T_r(t) \\ T_c(t) \\ T_{mw}(t) \\ T_b(t) \end{pmatrix} + \begin{pmatrix} 3.4405 \times 10^{-3} & 2.1352 \times 10^{-7} & 4.9597 \times 10^{-2} & 3.4405 \times 10^{-3} & 3.4405 \times 10^{-3} \\ 4.6794 \times 10^{-5} & 4.2008 \times 10^{-7} & 6.7456 \times 10^{-4} & 4.6794 \times 10^{-5} & 4.6830 \times 10^{-5} \\ 1.0270 \times 10^{-6} & 8.5653 \times 10^{-4} & 1.4804 \times 10^{-5} & 1.0270 \times 10^{-6} & 7.6028 \times 10^{-5} \\ 5.2425 \times 10^{-10} & 2.9819 \times 10^{-2} & 7.5574 \times 10^{-9} & 5.2425 \times 10^{-10} & 2.6111 \times 10^{-3} \end{pmatrix} \begin{pmatrix} \dot{Q} \\ T_o(t) \\ T_i(t) \\ \dot{Q}_a(t) \\ \phi_s(t) \end{pmatrix}$$

$$T_r(t) = [1 \quad 0 \quad 0 \quad 0] \begin{pmatrix} T_r(t) \\ T_c(t) \\ T_{mw}(t) \\ T_b(t) \end{pmatrix}$$

## Förvärmning

Det är ofta önskvärt att ha olika temperaturer i en byggnad olika tider på dygnet för att spara energi. Detta innebär att den framtida referenssignalen vanligen är känd. Denna typ av styrning är mycket praktisk att implementera i GPC-metoden. Om en temperaturändring önskas vid tidpunkten  $t + j$  får de  $N_y - j$  översta elementen i referenssignalvektorn  $\mathbf{R}$  helt enkelt anta detta värde. Optimeringen av kriteriefunktionen (2.32) tar då hänsyn till detta och regulatorn kan minimera reglerfelet med hänsyn till framtida referenssignaler.

### 3.3 Kvalitetsmått

För att kunna jämföra simuleringar med olika reglerparametrar behövs ett kvantitativt mått på regleringens kvalitet. Ett vanligt sådant är genomsnittet av kvadraten av felet, alltså skillnaden mellan referenssignal och utsignal, som benämns MSE (Mean Squared Error).

$$\text{MSE} = \frac{1}{N} \sum_{t=1}^N (y(t) - r(t))^2$$

För identifieringens kvalitet kan samma princip användas, men signalerna som subtraheras är  $y(t)$  och  $\hat{y}(t|t-1)$ .

# Kapitel 4

## Resultat

### 4.1 Identifiering

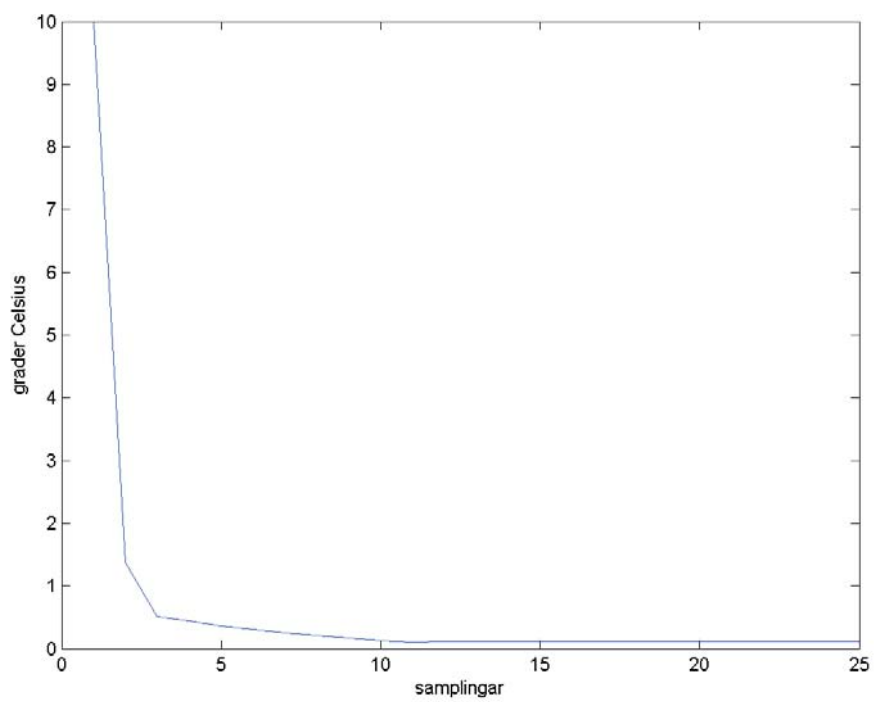
Regulatorns förmåga att skatta en modell som korrekt kan predicera systemets utsignaler är viktig för att regulatorn ska fungera bra. I beskrivningarna av simuleringarna nedan betecknar  $nb_1$  radiatoreffektens modellordning,  $nb_2$  utetemperaturens modellordning och  $nb_3$  solinstrålningens modellordning.

#### 4.1.1 Initialvärden

##### Kovariansmatris

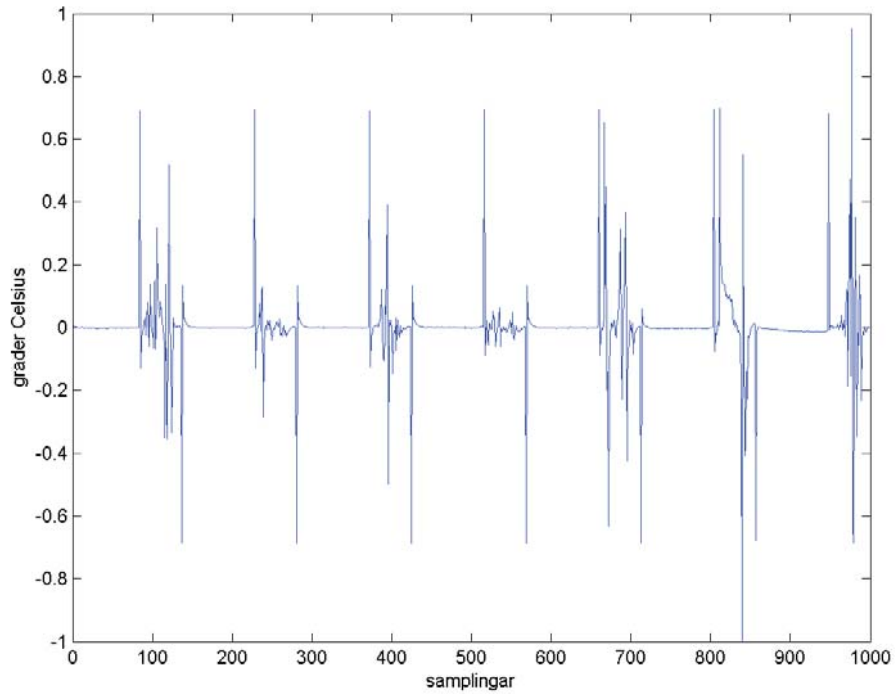
De sanna parametervärdena i  $\hat{\theta}$  ligger relativt nära noll. Om kovariansmatri-sens initialvärden,  $\mathbf{P}(0)$ , då väljs till ett relativt högt värde, finns det risk att vissa parametrar i  $\hat{\theta}$  får fel tecken. Detta gäller särskilt då solinstrålningen mäts. Om den eller de parametrar som beskriver insignalen får fel tecken, ”tror” regulatorn att mer effekt sänker temperaturen och vice versa. Detta leder till att den kommer ”fastna” vid full eller ingen effekt. Simuleringar har gett vid handen att ett  $\rho$  i (2.27) i detta fall bör ligga på ungefär en tiondel av  $\hat{\theta}(0)$  då solinstrålningen mäts för att vara på den säkra sidan.

## 4.1.2 Skattningsfel



Figur 4.1: Skattningsfelet de 25 första stegen, utetemperatur mäts,  $na = nb_1 = nb_2 = 1$ ,  $nc = 0$ .

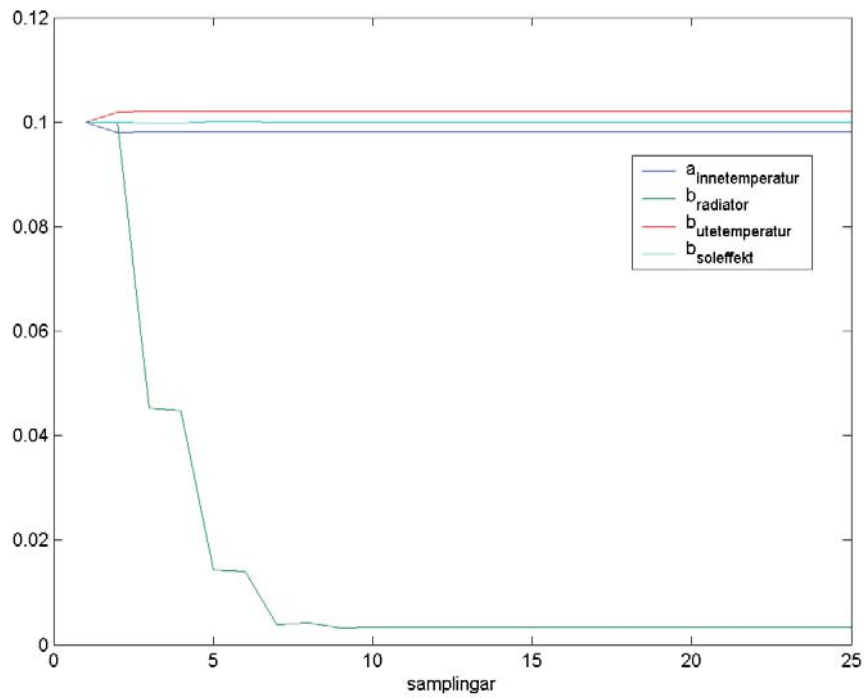




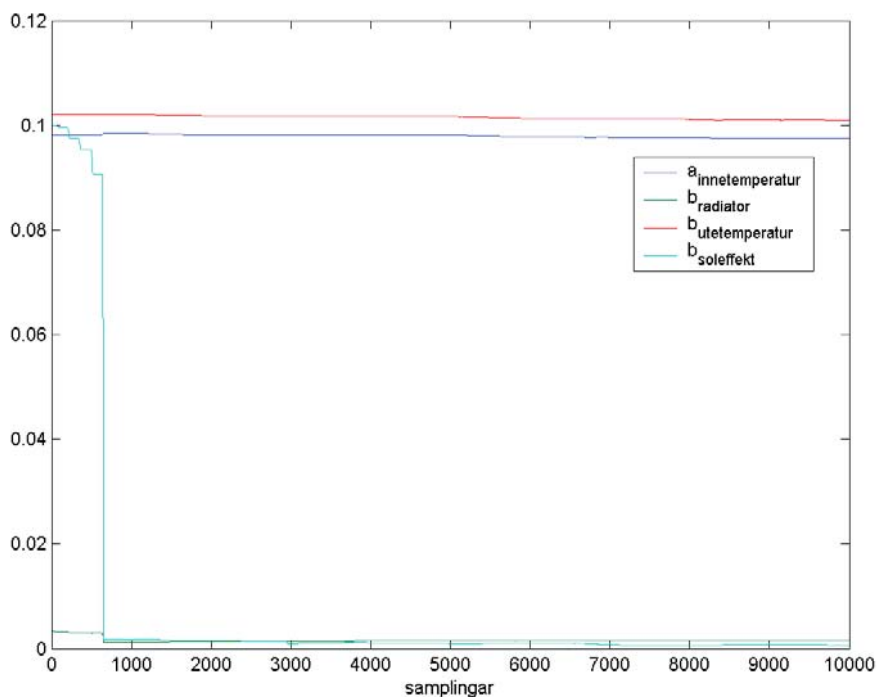
Figur 4.2: Skattningsfelet tidssteg 4000 till 5000, samma modellparametrar som ovan.

I figur 4.1 ses hur skattningsfelet vid enstegsprediktion avtar allteftersom modellen uppdateras. Skattningsfelen i figur 4.2 beror på stegstörningar och solinstrålning. MSE: 0.029516. En modellordning där  $na = 4$ ,  $nb_1 = nb_2 = 3$  ger ett något högre MSE: 0.029792. Båda dessa MSE anses vara acceptabla för reglering.

### 4.1.3 Parameterskattning



Figur 4.3: Koefficienterna i  $\hat{\theta}(t)$  de första 25 tidsstegen, samma modellparametrar som ovan med tillägget att även solinstrålning mäts.



Figur 4.4: Koefficienterna i  $\hat{\theta}(t)$  för 10 000 tidssteg, samma modellparametrar som ovan.

Figurerna 4.3 och 4.4 visar hur skattningen av koefficienterna i  $\hat{\theta}(t)$  varierar. Alla koefficienter utom den som representerar solinstrålningens påverkan på inomhustemperaturen svänger in snabbt i början i figur 4.3 vilket kan förklaras med liten solinstrålning och därmed liten excitation under den perioden. Sett över hela perioden i figur 4.4 är det tydligt när solinstrålningen uppdaterar skattningen.

## 4.2 Reglering

### 4.2.1 Inverkan av designparametrar

GPC-metoden är beroende av en mängd parametrar, som påverkar prestanda, stabilitet och robusthet. Simuleringarna som följer är genomförda med

en stegstörning som motsvarar 200 W om inte annat anges.

## Styr- och prediktionshorisonter

Modellparametrarna som används i följande fall är  $na = 4$ ,  $nb_1 = nb_2 = 3$ ,  $nc = 2$ . Ingen stegstörning användes och solinstrålningen mättes inte av regulatorn.

En prediktionshorisont på  $N_y = 6$  har jämförts med en på  $N_y = 15$  där styrsignalhorisonten i båda fall var  $N_u = 6$ .  $N_y = 6$  gav ett MSE på 0.15972 och  $N_y = 15$  gav ett MSE på 0.15787. Detta är rimligt då regulatorn kan "planera längre i förväg" givet att modellen är tillräckligt bra.

Även två olika styrsignalshorisonter provades,  $N_u = 2$  och  $N_y = 8$  då  $N_y = 15$  i båda fall. Resultatet blev, något överraskande, ett MSE på 0.15441 för  $N_u = 2$  och respektive 0.15819 för  $N_u = 8$ . För att förstå detta gjordes ett försök där modellen inte hade solinstrålning som insignal, och det visade sig då att en längre styrsignalhorisont gav ett lägre MSE. Detta tyder på att regulatorn vid en längre styrsignalhorisont "planerar fel" då (relativt kraftiga) ej uppmätta eller dåligt predicerade störningar förekommer, och reglerfelet blir större vid längre styrsignalhorisont.

## Straff på styrsignalinkrementen

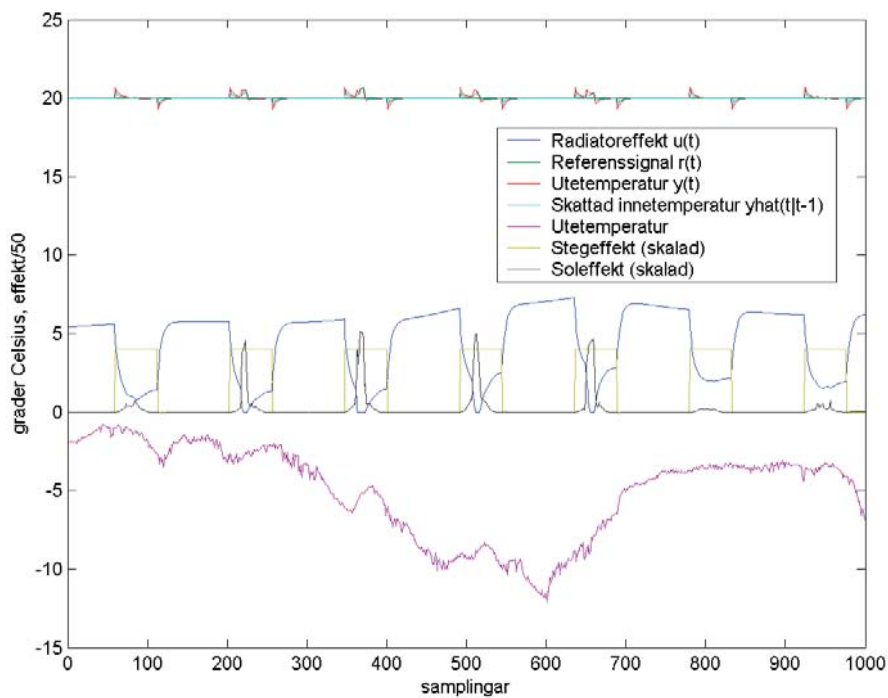
Straffkonstanten  $q_u$  i (2.32) på styrsignalinkrementen har ett nära samband med robusthet. Om denna parameter sätts till ett för lågt värde visar det sig att regulatorn lätt överkompenserar vid felaktig modellordning stabiliteten går förlorad. Om  $q_u$  sätts till ett högt värde blir regulatorn långsam eftersom förändringarna i varje tidssteg blir små. I simuleringarna har  $q_u = 10^{-4}$  fungerat som en bra kompromiss.

## Modellordningar

GPC-metoden medför som tidigare nämnts robusthet även vid felaktiga modellordningar vilket medför stor flexibilitet vid val av modellordning. En högre ordningens modell kan dock ha större möjlighet att täcka in dynamiken hos systemet vilket illustreras av att en modell med  $na = 1$ ,  $nb_1 = nb_2 = 1$

ger ett MSE på 0.56973 medan en modell med  $na = 4$ ,  $nb_1 = nb_2 = 3$  ger ett MSE på 0.56400.

## Uppmätta och ej uppmätta störsignaler



Figur 4.5: Reglering där endast innetemperaturen mäts,  $na = 1$ ,  $nb = 1$ ,  $nc = 0$ .

Att mäta en störsignal och använda den för prediktion är en form av framkoppling med potential att förbättra regleringen. Dock visar det sig, något överraskande, att regleringen inte förbättras nämnvärt av att mäta vare sig utetemperaturen eller solinstrålningen. I fallet med utetemperaturen beror det förmodligen på att dess påverkan på innetemperaturen är så långsam att så fort en liten effekt av en temperaturförändring utomhus mäts, kan den regleras bort. Det beror också på att återkopplingen är snabb, radiatorn kan sättas igång direkt och kompensera för en lägre temperatur vilket syns i figur 4.2.1 där varken utetemperatur eller solinstrålning mäts och  $na = 1$ ,

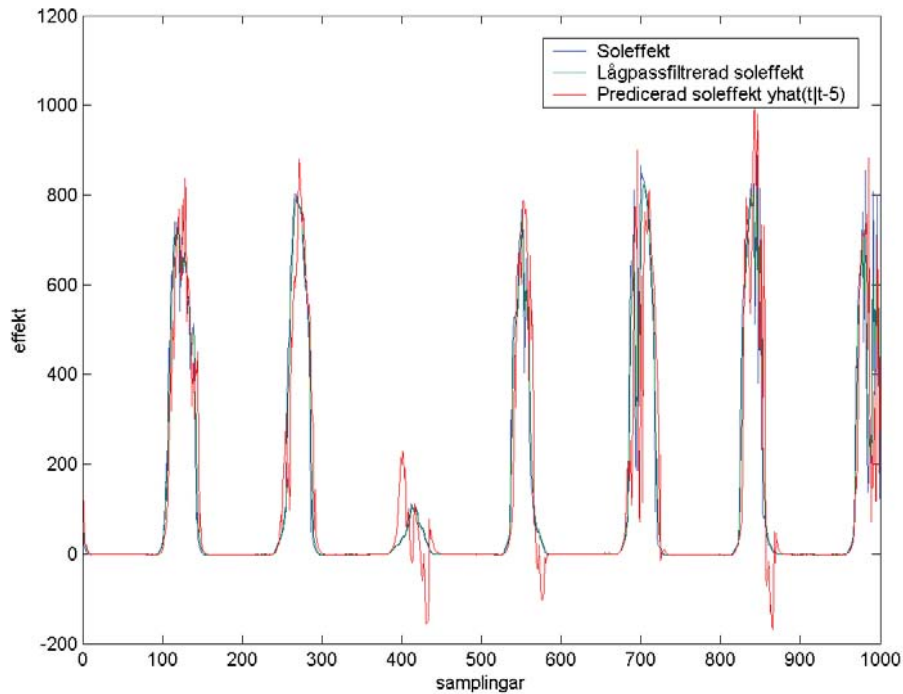
$nb = 1$ ,  $nc = 0$ . Det är möjligt att det lönar sig bättre att använda utetemperaturen i regleringen med andra typer av hus, till exempel då det finns tidsfördröjningar mellan styrsignal och värmeeffekt.

### Villkor för uppdatering av modell

Enligt det tidigare resonemanget om estimatoruppvridning prövades olika villkor för uppdatering av  $\hat{\theta}(t)$ . Dock visade det sig att något villkor inte behövdes, inte heller då solinstrålningen mättes, trots att den signalen ofta är 0.

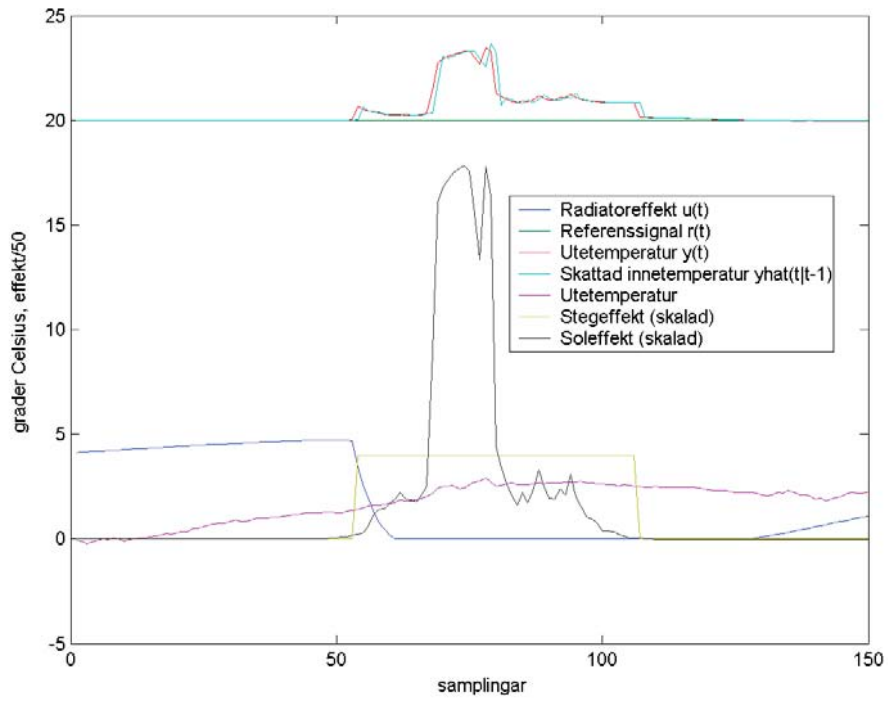
### 4.2.2 Prediktion av solinstrålning

Av de två stör signaler som kan mätas, utgör utetemperaturen inget problem i regleringen då den förändras relativt långsamt och inte behöver förutsägas i det system som undersökts. Solinstrålningen däremot, påverkar systemet relativt kraftigt och det är rimligt att en god prediktion av framtida solinstrålning skulle medföra bättre reglering och energibesparingar. Detta prövades därför genom att solinstrålningen modellerades som en SARI-MA(5, 0, 1)  $\times$  (1, 1, 1)-process med sampeltiden gånger antal samplingsperioder/dygn som period. I simuleringarna med prediktion av solinstrålning har stegstörning på 200 W använts.



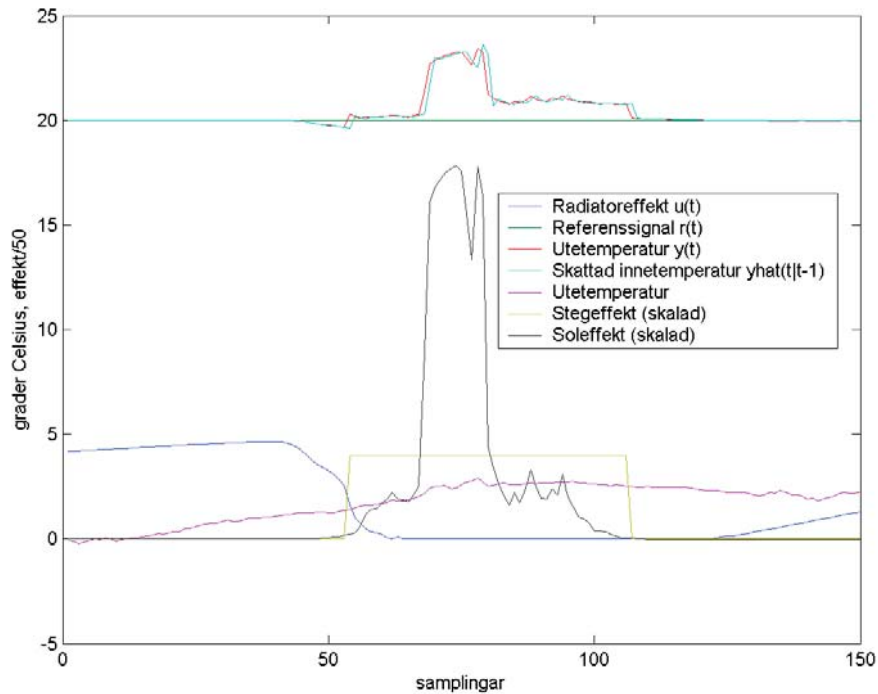
Figur 4.6: Femstegsprediktion av solinstrålning.

I figur 4.6 syns några av svårigheterna vid predicering av solinstrålning med en SARIMA-modell. Den blir mycket osäkrare ju fler steg framåt som prediktionen görs. Eftersom skattningsmetoden använder sig av gamla data för att skatta framtida värden, blir prediktionen dålig då solinstrålningen varierar mycket. En förbättring som kan göras är att ersätta negativa värden med 0, då solinstrålningen aldrig är negativ. Ett rekursivt lågpassfilter används där  $\alpha = 0.5$  (se appendix B) för att jämna ut högfrekventa komponenter i signalen.



Figur 4.7: Simulering då solinstrålningen mäts och predikteras två steg framåt,  $na = 4$ ,  $nb_1 = nb_2 = nb_3 = 3$ ,  $nc = 0$ .

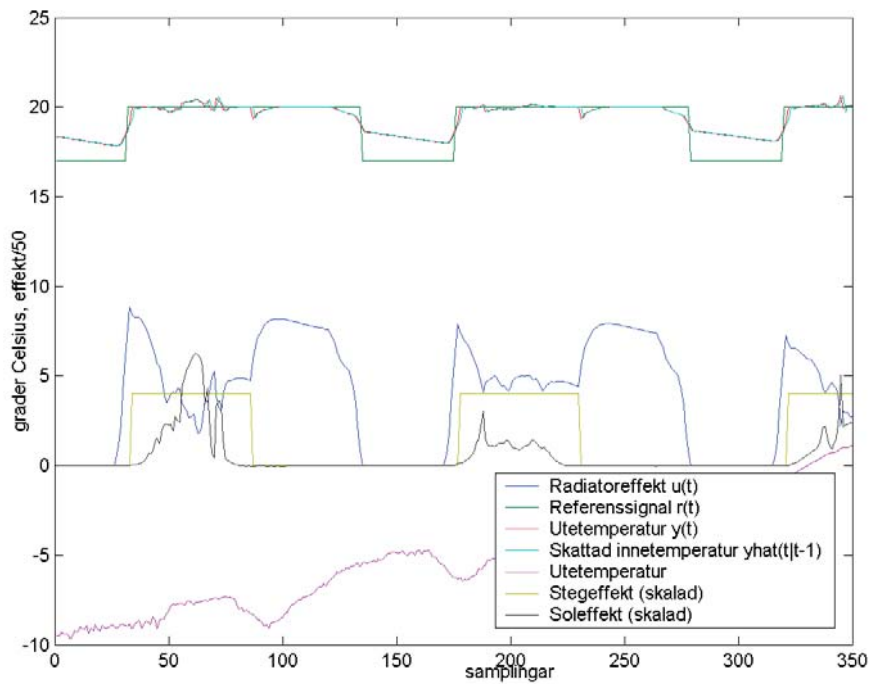




Figur 4.8: Simulering då solinstrålningen mäts och predikteras 15 steg framåt, samma modellparametrar som ovan.

En viss skillnad kan skönjas mellan två- och 15-stegsprediktion i figur 4.7 respektive 4.8, radiatoreffekten sänks tidigare vid den längre prediktionshorisonten för att kompensera för den förväntade värmeeffekten från solinstrålningen. Regleringen blir också något bättre, med ett MSE på 0.55070 för tvåstegsprediktion och 0.53869 vid 15 steg. Även om längre prediktionshorisonter ger regulatorn en chans att agera i tid, riskerar prediktionen att bli osäkrare då den enbart bygger på tidigare solinstrålningsdata.

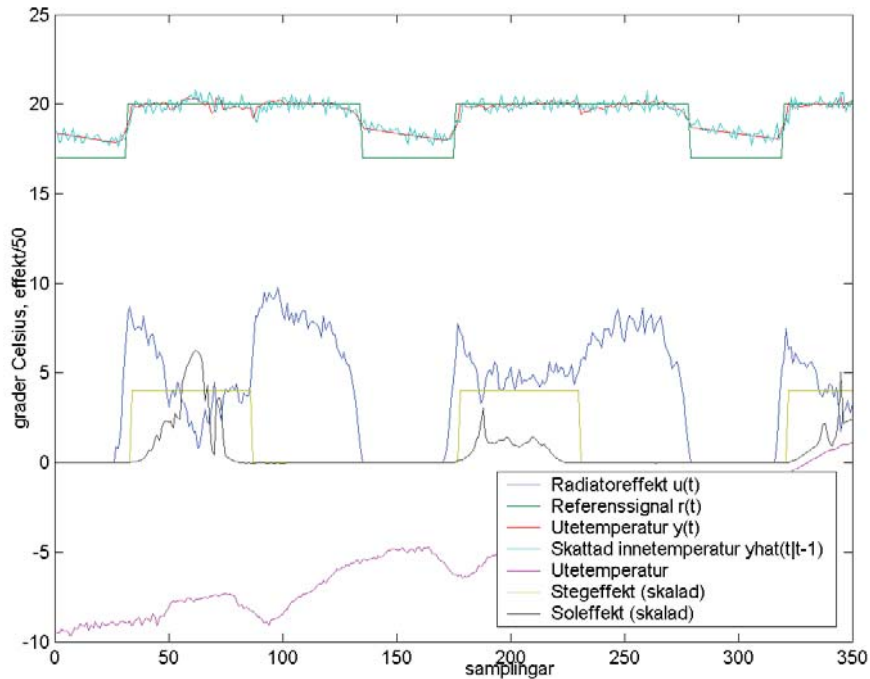
### 4.2.3 Varierande referenssignal



Figur 4.9: Simulering då solinstrålningen mäts och prediceras 15 steg framåt, varierande referenssignal, i övrigt samma modellparametrar som ovan.

Det är vanligt att ha en lägre temperatur på natten för att spara energi. Om framtida referenssignaler är kända kan övergången mellan olika referenssignaler ske på ett sätt som håller den faktiska innetemperaturen så nära den önskade som möjligt.

#### 4.2.4 Brusiga signaler



Figur 4.10: Simulering då solinstrålningen mäts och prediceras 15 steg framåt, varierande referenssignal, brus på stör- och utsignaler med varians 0.25, i övrigt samma modellparametrar som ovan.

Mätsignaler kan ofta vara brusiga, men regulatorn måste fungera ändå. Denna simulering hade samma förutsättningar som den förra, med skillnaden att vitt brus med variansen 0.25 lagts på alla mätsignaler. MSE förändras från 1.4828 till 1.5856.

#### $C(q^{-1})$ -polynomets inverkan

Vid brusiga signaler kan  $C(q^{-1})$ -polynomet ge bättre skattning av parametervektorn genom att det modellerar bruset. Det visar sig dock att skillnaden blir försumbar mellan  $nc = 0$  och  $nc = 2$  vid simuleringar med brusiga signaler då modellordningen  $na = 4$ ,  $nb_1 = nb_2 = nb_3 = 3$  används. Detta tyder

på att  $A(q^{-1})$ -polynomet klarar av att beskriva störningarna eftersom de då  $C(q^{-1}) = 1$  filtreras genom  $\frac{1}{A(q^{-1})}$ .

Vid en modellordning där  $na = 2$ ,  $nb_1 = nb_2 = nb_3 = 1$  ger  $nc = 2$  något bättre prediktion vilket är rimligt då ett lägre  $na$  medför att  $A(q^{-1})$ -polynomet inte kan modellera bruset lika bra. Regleringen förbättras dock endast försumbart.

# Kapitel 5

## Diskussion

Syftet med examensarbetet var att reglera inomhustemperaturen med hjälp av en kontinuerligt skattad modell och att pröva reglering med hänsyn till skattade framtida störsignaler. GPC-metoden fungerade bra för reglering av innetemperatur vid förekomst av både uppmätta och ej uppmätta störningar. Vid tillgång till goda skattningar av framtida störsignaler, kan regleringen ytterligare förbättras. Ett antal parametrar kan justeras för att trimma robusthet och snabbhet.

Kriteriefunktionen och bivillkorsmatriserna i GPC-metoden erbjuder ett smidigt sätt att realisera framkoppling och hantera begränsningar i styrsignaler. GPC kan implementeras med rekursiva identifieringsmetoder och numeriskt effektiva implementationer för att spara på datalagringskapacitet och processorkraft.

Ett annat resultat av examensarbetet är funktionsbiblioteket med matrisrutiner som förenklar framtida utveckling av regulatorprogrammet. Genom funktionsuppdelningen underlättas också tester av olika aspekter av regulatorn.

En möjlig utveckling av regulatorn är bättre skattning av framtida störsignaler. En lokal, säker solinstrålningsprognos från till exempel en väderleks-tjänst skulle öka regulatorns möjlighet att kompensera för solinstrålning.

Även en adaptiv regulator bör trimmas så att den är så effektiv som möjligt för det specifika systemet. Alternativet är att ge den en relativt konservativ inställning, vilket dock skulle medföra en ineffektiv reglering. Därför bör de

hustyper regulatorn ska reglera, modelleras så att de kan testas av regulatorprogrammet.

En möjlighet är att låta regulatorn identifiera det öppna systemet med exempelvis slumpmässiga styrsignaler under en period och inte uppdatera modellen under regleringen. Då ingen systemidentifiering utförs i det slutna systemet, försvinner kravet på excitation vilket gör regulatorn säkrare. Nackdelen blir att regulatorn inte kan anpassa sig till till exempel säsongsmässiga förändringar av systemet vilket kan leda till en sämre reglering. Identifieringsfasen måste också täcka in relevanta förhållanden, exempelvis både kalla och varma perioder, samt olika solinstrålningsintensiteter, för att modellen av systemet ska bli bra.

Det som återstår för att regulatorn ska kunna testas i riktiga hus, är att skriva funktioner som kommunicerar med sensorer och styrdon. Någon typ av användargränssnitt måste också skapas. En samplingsperiod som fungerar bra måste bestämmas.

# Bilaga A

## Matrisinverteringslemmat

Givet att inverserna existerar:

$$(\mathbf{A} + \mathbf{BCD})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{B}(\mathbf{C}^{-1} + \mathbf{DA}^{-1}\mathbf{B})^{-1}\mathbf{DA}^{-1} \quad (\text{A.1})$$

Bevis:

$$\begin{aligned} & (\mathbf{A} + \mathbf{BCD})(\mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{B}(\mathbf{C}^{-1} + \mathbf{DA}^{-1}\mathbf{B})^{-1}\mathbf{DA}^{-1}) \\ &= \mathbf{I} + \mathbf{BCDA}^{-1} - (\mathbf{B} + \mathbf{BCDA}^{-1}\mathbf{B})(\mathbf{C}^{-1} + \mathbf{DA}^{-1}\mathbf{B})^{-1}\mathbf{DA}^{-1} \\ &= \mathbf{I} + \mathbf{BCDA}^{-1} - \mathbf{BC}(\mathbf{C}^{-1} + \mathbf{DA}^{-1}\mathbf{B})(\mathbf{C}^{-1} + \mathbf{DA}^{-1}\mathbf{B})^{-1}\mathbf{DA}^{-1} \\ &= \mathbf{I} \end{aligned}$$

vilket bevisar (A.1).

# Bilaga B

## Rekursivt lågpasfilter

De signaler som används för att bygga upp en modell innehåller ibland högfrekvent brus. För att filtrera bort det kan ett rekursivt lågpasfilter användas. Ett första ordningens lågpasfilter (i kontinuerlig tid) ges av  $Y_f(s) = \frac{1}{1+s\tau}Y(s)$ .

I tidsdomänen:

$$\tau \frac{dy_f(t)}{dt} + y_f(t) = y(t)$$

Diskretisering:

$$\begin{aligned} \frac{dy_f(t)}{dt} &\approx \frac{y_f(t) - y_f(t-1)}{T} \\ T &= 1 \\ \Rightarrow y(t) &= \tau \frac{y_f(t) - y_f(t-1)}{1} + y_f(t) \end{aligned}$$

Alltså

$$\begin{aligned} y_f(t) &= \frac{\tau}{\tau+1}y_f(t-1) + \frac{1}{\tau+1}y(t) \\ \alpha &= \frac{\tau}{\tau+1} \Rightarrow (1-\alpha) = \frac{1}{1+\tau} \end{aligned}$$

Slutligen

$$y_f(t) = \alpha y_f(t-1) + (1-\alpha)y(t)$$

där  $0 < \alpha < 1$ . Ju större  $\alpha$ , desto större utjämning men också större förskjutning av signalen.



# Bilaga C

## Härledning av husmodell

Här följer förklaringar till och värden på variablerna i husmodellen som använts. Modellen har byggts upp utifrån principen om energibalans, det vill säga att förändringen i energi är lika med energi in minus energi ut.

Energibalanser för innetemperaturen, betongväggen, isoleringen och tegelväggen (indexen  $a$ ,  $c$ ,  $mw$  och  $b$  står för luft, betong, mineralull och tegel):

$$\rho_a c_{pa} V_r \frac{dT_r}{dt} = \dot{Q} - \dot{V}_r \rho c_{pa} (T_r - T_i) - \frac{1}{\frac{1}{\alpha_i} + \frac{l_c}{2\lambda_c}} A_1 (T_r - T_c) \quad (\text{C.1})$$

$$\rho_c c_c V_c \frac{dT_c}{dt} = \frac{1}{\frac{1}{\alpha_i} + \frac{l_c}{2\lambda_c}} A_1 (T_r - T_c) - \frac{2\lambda_c \lambda_{mw}}{l_c \lambda_{mw} + l_{mw} \lambda_c} (T_c - T_{mw}) A_2$$

$$\rho_{mw} c_{mw} V_{mw} \frac{dT_{mw}}{dt} = \frac{2\lambda_c \lambda_{mw}}{l_c \lambda_{mw} + l_{mw} \lambda_c} (T_c - T_{mw}) A_2 - \frac{2\lambda_b \lambda_{mw}}{l_b \lambda_{mw} + l_{mw} \lambda_b} (T_{mw} - T_b) A_3$$

$$\rho_c c_c V_c \frac{dT_c}{dt} = \frac{2\lambda_b \lambda_{mw}}{l_b \lambda_{mw} + l_{mw} \lambda_b} (T_{mw} - T_b) A_3 - \frac{1}{\frac{1}{\alpha_o} + \frac{l_b}{2\lambda_b}} A_4 (T_b - T_o) \quad (\text{C.2})$$

(Soleimani-Mohseni 2005)

Till ekvation (C.1) läggs termerna  $\dot{Q}_a$  och  $A_e \phi_s$ , effektiv fönsterarea gånger solinstrålningseffekt, för att hantera värmeeffekt på grund av aktiviteter och värmeeffekt på grund av solinstrålning. Solinstrålningen påverkar även ytterväggen, så termen  $(A_{b,s} - A_w) \phi_s$ , solbelyst area gånger solinstrålningseffekt,

läggs till ekvation (C.2). Det ger följande system på tillståndsform:

$$\begin{pmatrix} \dot{T}_r \\ \dot{T}_c \\ \dot{T}_{mw} \\ \dot{T}_b \end{pmatrix} = \begin{pmatrix} \frac{-\dot{V}_r \rho_c c_{pa} - k_1 A_1}{\rho_a c_{pa} V_r} & \frac{k_1 A_1}{\rho_a c_{pa} V_r} & 0 & 0 \\ \frac{k_1 A_1}{\rho_c c_c V_c} & \frac{-k_1 A_1 - k_2 A_2}{\rho_c c_c V_c} & \frac{k_2 A_2}{\rho_c c_c V_c} & 0 \\ 0 & \frac{k_2 A_2}{\rho_{mw} c_{mw} V_{mw}} & \frac{-k_2 A_2 - k_3 A_3}{\rho_{mw} c_{mw} V_{mw}} & \frac{k_3 A_3}{\rho_{mw} c_{mw} V_{mw}} \\ 0 & 0 & \frac{k_3 A_3}{\rho_b c_b V_b} & \frac{-k_3 A_3 - k_4 A_4}{\rho_b c_b V_b} \end{pmatrix} \begin{pmatrix} T_r \\ T_c \\ T_{mw} \\ T_b \end{pmatrix} \\
+ \begin{pmatrix} \frac{1}{\rho_a c_{pa} V_r} & 0 & \frac{\dot{V}_r}{V_r} & \frac{1}{\rho_a c_{pa} V_r} & \frac{A_e}{\rho_a c_{pa} V_r} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{k_4 A_4}{\rho_b c_b V_b} & 0 & 0 & \frac{A_{b,s} - A_w}{\rho_b c_b V_{b,s}} \end{pmatrix} \begin{pmatrix} \dot{Q} \\ T_o \\ T_i \\ \dot{Q}_a \\ \phi_s \end{pmatrix} \\
T_r = [1 \quad 0 \quad 0 \quad 0] \begin{pmatrix} T_r \\ T_c \\ T_{mw} \\ T_b \end{pmatrix}$$

där

$$\begin{aligned}
k_1 &= \frac{1}{\frac{1}{\alpha_i} + \frac{l_c}{2\lambda_c}} \\
k_2 &= \frac{2\lambda_c \lambda_{mw}}{l_c \lambda_{mw} + l_{mw} \lambda_c} \\
k_3 &= \frac{2\lambda_b \lambda_{mw}}{l_b \lambda_{mw} + l_{mw} \lambda_b} \\
k_4 &= \frac{1}{\frac{1}{\alpha_o} + \frac{l_b}{2\lambda_b}} \\
\dot{V}_r &= \frac{n_v V_r}{3600}
\end{aligned}$$

Variabler och parametrar:

- $A$  = Area för de olika lagren (1 ytterst, 4 innerst), [m<sup>2</sup>]
- $A_{b,s}$  = Area för solbelyst tegelvägg, [m<sup>2</sup>]
- $A_e$  = Effektiv fönsterarea, [m<sup>2</sup>]
- $A_w$  = Total fönsterarea, [m<sup>2</sup>]
- $T$  = Temperatur för visst lager, [K]
- $T_r$  = Rumstemperatur, [K]
- $T_o$  = Utomhustemperatur, [K]
- $T_i$  = Temperatur ventilationsluft, [K]
- $\rho$  = densitet, [kg/m<sup>3</sup>]
- $c$  = Specifik värmekapacitet, [J/kgK]
- $V_r$  = Rumsvolym, [m<sup>3</sup>]
- $V_{b,s}$  = Volym solbelyst tegelvägg, [m<sup>3</sup>]
- $l$  = Lagertjocklek, [m]
- $\dot{Q}$  = Radiatoreffekt, [W]
- $\dot{Q}_a$  = Värmeeffekt på grund av aktivitet, [W]
- $\phi_s$  = Solinstrålningseffekt, [W]
- $n_v$  = Antal luftutbyten per timme, [h<sup>-1</sup>]
- $\lambda$  = Värmekonduktivitet, [W/(mK)]
- $\alpha_i$  = Inre värmeöverföringskoefficient, [W/(m<sup>2</sup>K)]
- $\alpha_o$  = Yttre värmeöverföringskoefficient, [W/(m<sup>2</sup>K)]

Följande konstanter har använts:

$$n_v = 1.2[\text{h}^{-1}]$$

$$\text{Inre dimensioner} = 4 \times 3 \times 3[\text{m}]$$

$$V_r = 36[\text{m}^3]$$

$$l_c = 0.1[\text{m}]$$

$$l_{mw} = 0.15[\text{m}]$$

$$l_b = 0.12[\text{m}]$$

$$A_1 = 42[\text{m}^2]$$

$$A_2 = 44.4[\text{m}^2]$$

$$A_3 = 48[\text{m}^2]$$

$$A_4 = 50.88[\text{m}^2]$$

$$A_{b,s} = 10.33[\text{m}^2]$$

$$A_e = 1.0[\text{m}^2]$$

$$A_w = 1.67[\text{m}^2]$$

$$c_{pa} = 1000[\text{J}/(\text{kgK})]$$

$$\rho_a = 1.20[\text{kg}/\text{m}^3]$$

$$\alpha_i = 8[\text{W}/(\text{m}^2\text{K})]$$

$$\alpha_o = 20[\text{W}/(\text{m}^2\text{K})]$$

$$c_c = 920[\text{J}/(\text{kgK})]$$

$$\begin{aligned}\rho_c &= 2300[\text{kg/m}^3] \\ \lambda_c &= 1.7[\text{W}/(\text{mK})] \\ c_b &= 800[\text{J}/(\text{kgK})] \\ \rho_b &= 1700[\text{kg/m}^3] \\ \lambda_b &= 0.80[\text{W}/(\text{mK})] \\ c_{mw} &= 1000[\text{J}/(\text{kgK})] \\ \rho_{mw} &= 35[\text{kg/m}^3] \\ \lambda_{mw} &= 0.04[\text{W}/(\text{mK})] \\ V_c &= 4.32[\text{m}^3] \\ V_b &= 5.93[\text{m}^3] \\ V_{b,s} &= 1.44[\text{m}^3] \\ V_{mw} &= 6.93[\text{m}^3] \\ k_1 &= 6.50[\text{W/K}] \\ k_2 &= 0.50[\text{W/K}] \\ k_1 &= 0.50[\text{W/K}] \\ k_1 &= 8[\text{W/K}]\end{aligned}$$

(Soleimani-Mohseni 2005)

# Litteraturförteckning

- Andersen, K. K., Madsen, H. & Hansen, L. H. (2000), ‘Modelling the heat dynamics of a building using stochastic differential equations’, *Energy and Buildings* **31**(1), 13–24.
- Boucher, P., Dumur, D. & Ehrlinger, A. (1996), ‘Unified approach of equality and inequality constraints in g.p.c.’, *Proceedings of the 1996 IEEE International Conference on Control Applications* .
- Chatfield, C. (2000), *Time-Series Forecasting*, Chapman & Hall/CRC.
- Clarke, D. W., Mohtadi, C. & Tuffs, P. S. (1987), ‘Generalized predictive control - part ii. the basic algorithm’, *Automatica* **23**(2), 149–160.
- Heath, M. T. (2005), *Scientific Computing - An introductory Survey*, McGrawHill.
- Ljung, L. & Glad, T. (2003), *Reglerteori - Flervariabla och olinjära metoder*, Studentlitteratur.
- Ljung, L. & Glad, T. (2004), *Modellbygge och simulering*, Studentlitteratur.
- Lundgren, J., Rönnqvist, M. & Värbrand, P. (2001), *Linjär och ickelinjär optimering*, Studentlitteratur.
- Park, J. H., Han, S. & Kwon, W. H. (2007), Generalized predictive control over state space, in ‘SICE Annual Conference’, pp. 1281–1286.
- Soleimani-Mohseni, M. (2005), Modelling and Intelligent Climate Control of Buildings, PhD thesis, Chalmers University of Technology.
- Söderström, T. & Stoica, P. (1989), *System Identification*, Prentice Hall.
- Åström, K. J. & Wittenmark, B. (1995), *Adaptive Control*, Addison-Wesley Publishing Company.