CODEN: LUTFD2/(TFRT-5208)/1-080/(1978)

DECENTRALISERAD REGLERING AV STORA SYSTEM MED SJÄLVINSTÄLLANDE REGULATORER

STEFAN LINDGREN

Institutionen för Reglerteknik
Lunds Tekniska Högskola
Maj 1978

DECENTRALISERAD REGLERING AV STORA SYSTEM MED SJÄLVINSTÄLLANDE REGULATORER

Examensarbete utfört 1977-78 vid Institutionen
för Reglerteknik vid Lunds Universitet av
Stefan Lindgren
Handledare: Ivar Gustavsson, Björn Wittenmark

SAMMANFATTNING

Vid reglering av stora system finns det ibland skäl att tillgripa decentraliserad reglering. Här behandlas sådan reglering med självinställande regulatorer av en typ som baseras på minstakvadratanpassning och minimalvariansstyrning. Undersökningen, som omfattar såväl teoretiska beräkningar som simuleringar, indikerar att regulatorerna i allmänhet konvergerar så att styrningen blir "optimal" under givna förutsättningar.

ABSTRACT

The problem of controlling a large-scale system is sometimes solved by using decentralized control. Decentralized control using self tuning regulators based on least squares estimation and minimum variance control is considered. Theory and simulations indicate that the regulators generally converge in such a way that the control will be "optimal" under actual conditions.

INNEHÅLLSFÖRTECKNING

INLEDNING

Detta arbete behandlar decentraliserad reglering av "stora" system, dvs. system som naturligen kan betraktas som sammansatta av flera delsystem i samverkan. Teoretiskt sett borde ett sådant system styras centralt för att uppnå optimal reglering. Det kan emellertid finnas flera praktiska skäl att decentralisera styrningen. Central styrning kan t.ex. kräva ett orimligt stort flöde av information, vilket kan bli kostsamt och tidsödande. Det kan vara billigare med många regulatorer som löser små lokala problem än med en enda regulatorer som löser ett komplext globalt problem. Sådana frågor behandlas i [1].

För att slippa problem med identifieringen och eventuell tidsvariabilitet ligger det nära till hands att försöka reglera med självinställande regulatorer, t.ex. av den typ som beskrivs i [2].

Här behandlas uteslutande ett system bestående av två samverkande enkla stokastiska delsystem. Varje delsystem regleras med en självinställande regulator baserad på minstakvadratanpassning och minimalvariansstyrning. För att kunna simulera detta system har jag skrivit program anpassade till simuleringspaket utarbetade vid institutionen.

Undersökningen, vilken omfattar såväl teoretiska beräkningar som simuleringar, indikerar att regulatorerna i allmänhet konvergerar så att styrningen blir "optimal" under givna förutsättningar. Den decentraliserade regleringen medför dock som väntat oftast större förluster än vad som skulle varit fallet vid central styrning.

1. DECENTRALISERAD REGLERING OCH SJÄLVINSTÄLLANDE REGULATORER

Stora system, dvs. system med många mät- och styrsignaler, har ofta en sådan struktur att det faller sig naturligt att betrakta dem som sammansatta av flera delsystem i samverkan. Exempel på sådana system är kraftnäten med sina kraftverk och abonnenter.

Decentraliserad reglering

Utifrån synsättet ovan är steget till decentraliserad reglering inte långt. - Man förser varje delsystem med en egen regulator. I fig. 1.1 visas ett sådant system som dessutom har en central för samordning.

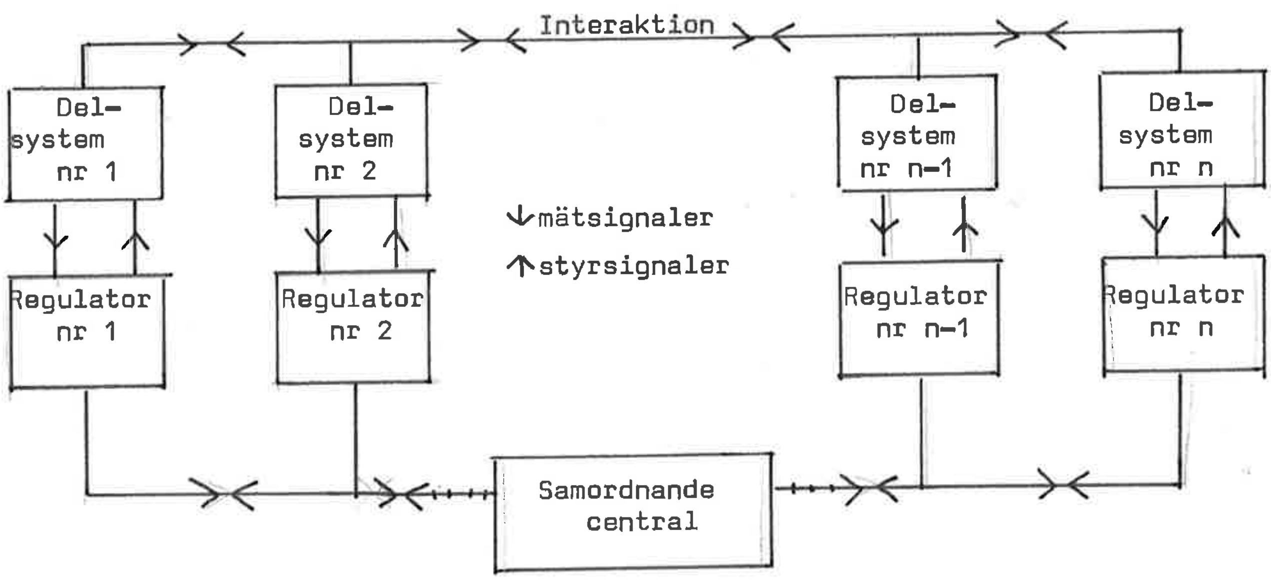


Fig. 1.1 "Stort" system med decentraliserad reglering.

De skäl som kan motivera decentraliserad reglering gäller även för stora system utan uppenbart uppdelad struktur och är följande:

1. Central reglering kan kräva ett orimligt stort flöde av information, vilket kan bli för kostsamt och tidsödande.
2. Det kan vara mindre krävande och därigenom billigare att låta många (decentraliserade) regulatorer lösa enkla lokala reglerproblem än att med en central regulator lösa ett komplicerat problem för hela systemet.

Självinställande regulatorer

Vi skall uteslutande behandla det fall att varje regulator är en så kallad självinställande regulator (Self Tuning Regulator; STURE). Härigenom hoppas vi framför allt att slippa arbete med modellbyggande men även att kunna följa, åtminstone långsamma, förändringar i systemet. Regulatorerna bygger på den typ av självinställande regulatorer, som beskrivs i [2].

Betrakta ett linjärt, tidsinvariant system med en insignal u och en utsignal y, vilket kan beskrivas med följande modell:

$$\begin{aligned} y(t)+a_1 y(t-1)+ \ldots\ldots +a_n y(t-n) &= b_1 u(t-k-1)+ \ldots\ldots +b_n u(t-k-n)+ \\ &\quad +e(t)+c_1 e(t-1)+\ldots\ldots +c_n e(t-n) \end{aligned} \tag{1.1}$$

där $b_1 \neq 0$ och $e(t)$ är ett vitt brus.
Det är välkänt att under vissa förutsättningar finns en strategi som minimerar utsignalens varians.

Om $c_1=c_2= \ldots\ldots =c_n=0$ får minimal varians-strategin en särskilt enkel form. Systemet kan då skrivas:

$$y(t)+\alpha_1 y(t-k-1)+ \ldots\ldots +\alpha_n y(t-k-n)= \\ =\beta_0 [u(t-k-1)+\beta_1 u(t-k-2)+ \ldots\ldots +\beta_l u(t-k-l-1) \quad +\varepsilon(t) \tag{1.2}$$

där $l=n+k-1$ och $\varepsilon(t)$ är ett glidande medelvärde av $e(t)$. Minimal varians-strategin blir helt enkelt:

$$\hat{u}(t)=\frac{1}{\beta_0}[\alpha_1 y(t)+ \ldots\ldots +\alpha_n y(t-n-1)] -\beta_1 u(t-1)- \ldots\ldots -\beta_l u(t-l) \tag{1.3}$$

I modell (1.2) var parametrarna konstanta och kunde beräknas med hjälp av (1.1). Vi förutsätter nu att parametrarnas värden i (1.1) är okända (ev. med undantag för β_0) och ansätter modellen (1.2) med okända parametervärden. Idén bakom regulatorn är att i varje tidssteg skatta parametrarna i (1.2) och beräkna styrsignalen ur (1.3) med dessa skattade värden på parametrarna. Mera precist kan regulatorns arbetssätt beskrivas så här: Vid varje tidssteg utföres två moment.

1. Parameterskattning.

Med hjälp av en rekursiv algoritm baserad på minsta kvadratmetoden uppdateras skattningen av parametervärden i (1.2) med hänsyn till nya mätvätden

2. Beräkning av styrsignalen

Styrsignalen beräknas ur (1.3) med de parametervärden som beräknats i moment 1.

Parametern β_0 i modellen (1.2) ges vanligen ett konstant värde men kan även ingå bland de skattade parametrarna.

Minsta kvadratanpassning i modellen (1.2) innebär att algoritmen minimerar $\sum_{i=0}^{t} \varepsilon(i)^2$. Med en liten förändring av algoritmen minimeras i stället $\sum_{i=0}^{t} \lambda^{t-i} \varepsilon(i)^2$ där $\lambda \leq 1$. Vi ser att $\lambda < 1$ innebär att regulatorn fäster störst vikt vid färska data och att äldre data "glöms bort" exponentiellt. På detta sätt kan parameterskattningarnas ändringshastighet ökas. Den rekursiva algoritmen kräver att parametrarna α_i och β_i tilldelas startvärden. Dessutom innehåller algoritmen en "kovariansmatris" P(t), som också skall ges ett startvärde P(0). Matrisen P(t) speglar osäkerheten i parameterskattningarna och P(0) bör således ange en uppskattning av osäkerheten hos parametrarnas startvärden.

Självinställande regulatorer behandlas utförligare i [2]-[5].

2. DET BEHANDLADE PROBLEMET

Den fortsatta behandlingen ägnas uteslutande åt system som kan simuleras med de i arbetet ingående programmen. Det synes därför enklast att fastställa beteckningar o. dyl. i samband med beskrivningen av programmens möjligheter.

Strukturen för de system vi skall syssla med ges i fig. 2.1

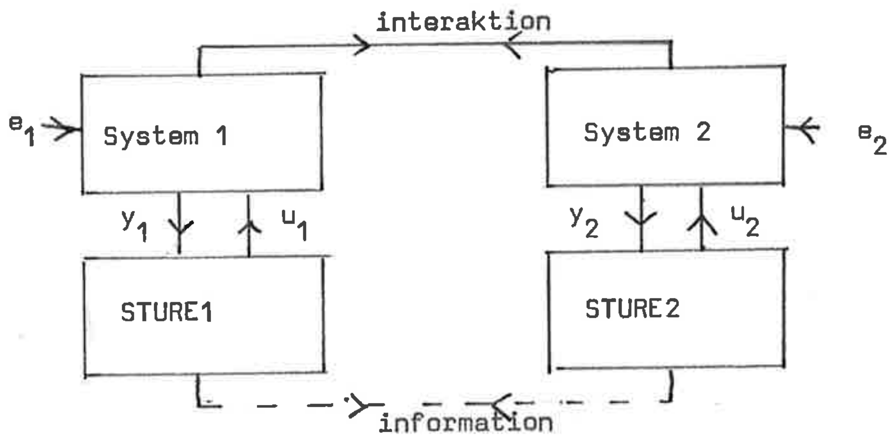


Fig. 2.1 Ett "stort" system med decentraliserad struktur.

Inför följande beteckningar:

$y(t)=\begin{pmatrix} y_1(t) \\ y_2(t) \end{pmatrix}$ systemets utsignaler

$u(t)=\begin{pmatrix} u_1(t) \\ u_2(t) \end{pmatrix}$ stysignaler

$e(t)=\begin{pmatrix} e_1(t) \\ e_2(t) \end{pmatrix}$ störningar

System 1 o. 2 beskrivs med följande modell:

$$y(t)+A(1)y(t-1)+\ldots\ldots+A(n_A)y(t-n_A)=$$

$$=B(1)u(t-k_B-1)+\ldots\ldots+B(n_B)u(t-k_B-n_B)+$$

$$+e(t)+C(1)e(t-1)+\ldots\ldots+C(n_C)e(t-n_C) \qquad (2.1)$$

där

$$A(k)=\begin{pmatrix} a_{11}(k) & a_{12}(k) \\ a_{21}(k) & a_{22}(k) \end{pmatrix} \qquad \text{etc.}$$

Vektorn e(t) är vitt, normalfördelat brus med medelvärde 0 och:

$$\mathrm{var}(e_1(t)) = r_{11}$$
$$\mathrm{var}(e_2(t)) = r_{22}$$
$$\mathrm{kov}(e_1(t),e_2(t)) = r_{12}$$

STURE1 och STURE2 är konstruerade med den självinställande regulatorn i kapitel 1 som ledstjärna. De är dock något mer komplicerade för att kunna utnyttja den information de eventuellt tillåts utbyta (jfr. fig. 2.1).
Om parametrarna β'_0 och β''_0 <u>inte</u> skattas gäller att följande modell ansättes för STURE1:

$$\begin{aligned} y_1(t) &+ \alpha'_1 y_1(t-k'_1-kdel-1)+\ldots\ldots+\alpha'_{n'_1} y_1(t-k'_1-kdel-n'_1) \\ &+ \gamma'_1 y_2(t-k'_3-kdel-1)+\ldots\ldots+\gamma'_{n'_3} y_2(t-k'_3-kdel-n'_3) \\ = \beta'_0\{u_1(t-k'_2-1) &+ \beta'_1 u_1(t-k'_2-2)+\ldots\ldots+\beta'_{n'_2} u_1(t-k'_2-n'_2-1) \\ &+ \delta'_1 u_2(t-k'_4-kdel-1)+\ldots\ldots+\delta'_{n'_4} u_2(t-k'_4-kdel-n'_4) + \varepsilon_1(t) \end{aligned} \qquad (2.2)$$

För STURE2 gäller:

$$y_2(t)+\alpha_1''y_2(t-k_2''-kdel-1)+\ldots\ldots+\alpha'_{n_1''}y_2(t-k_1''-kdel-n_1'')$$

$$+\gamma_1''y_1(t-k_3''-kdel-1)+\ldots\ldots+\gamma''_{n_3''}y_1(t-k_3''-kdel-n_3'') \tag{2.3}$$

$$= \beta_0''\{u_2(t-k_2''-1) \quad +\beta_1''u_2(t-k_2''-2)+\ldots+\beta_{n_2''}u_2(t-k_2''- \quad n_2''-1)$$

$$+\delta_1''u_1(t-k_4''-kdel-1)+\ldots\ldots+\delta''_{n_4''}u_1(t-k_4''-kdel-n_4'')\} +\varepsilon_2(t)$$

Om t.ex. β_0'' skall ingå bland de skattade parametrarna ändras modellen i STURE2 till:

$$\text{V.L.} =\beta_0''u_2(t-k_2''-1)+\ldots\ldots+\beta_{n_2''-1}u_2(t-k_2''-n_2'')$$

$$+\delta_1''u_1(t-k_4''-kdel-1)+\ldots\ldots+\delta''_{n_4''}u_1(t-k_4''-kdel-n_4'') + \varepsilon_2(t) \tag{2.4}$$

I varje tidssteg utför STUREi de två momenten: 1. Skattning av parametrarna α, β, γ, δ i (2.2) o. (2.3) 2. Beräkning av motsvarande styrsignal $\hat{u}_i(t)$.

För att förklara förekomsten av kdel i modell (2.2) antar vi att $k_2'=0$. Styrlagen i STURE1 blir:

$$\hat{u}_1(t)=\frac{1}{\beta_0'}\{\alpha_1'y_1(t-k_1'-kdel)+ \ldots\ldots +\gamma_1'y_2(t-k_3'-kdel)+ \ldots\ldots\} -$$

$$-\beta_1'u_1(t-1)- \ldots\ldots -\delta_1'u_2(t-k_4'-kdel)- \ldots\ldots$$

och vi ser att kdel kan tolkas som en generell fördröjning av information. (Jämför k_2' som är en uppskattning av tidsfördröjningen i systemet.) Observera att man måste tänka sig för när man ansätter modellen (2.2) så att styrlagen blir realiserbar och inte t.ex $\hat{u}_1(t) =\alpha y_1(t+1)$ eller $\hat{u}_1(t) =\alpha y_1(t-1)+\delta' u_2(t)$. (Vi förutsätter alltså att $u_1(t)$ och $u_2(t)$ beräknas "samtidigt".)

Parametrarna α_1 ..... tilldelas startvärden α'_{10} etc. och kovariansmatriserna $P_1(t)$ och $P_2(t)$ ges startvärden, vilka är diagonalmatriser. Vanligen sättes alla diagonalelement lika så att $P_1(0) = P_2(0) = p_0 \cdot I$, där p_0 är ett positivt tal och I är enhetsmatrisen. Vi drar oss till minnes att p_0 bör spegla osäkerheten hos parametrarnas startvärden. Detta kan förklaras med att om p_0 är stort tillåts stora förändringar av parameterskattningarna i början och vice versa.

Regulatorernas minsta kvadratalgoritmer innehåller en gemensam viktfaktor λ så att $\lambda < 1$ innebär att anpassningen av parametrarna sker med "exponentiell glömska". Det finns även möjlighet att variera λ på ett sådant sätt att λ konvergerar från ett startvärde < 1 mot 1 när $t \to \infty$.
Det är ibland fördelaktigt att begränsa styrsignalerna. I programmen kan detta göras så att $|\hat{u}_i| \leq ulim$, där ulim kan väljas godtyckligt.

3. PROGRAMBESKRIVNING

Simuleringarna har gjorts på en PDP 15 med hjälp av ett interaktivt program - SIMNON, se [6]. Interaktionen försiggår via skrivmaskinsterminal, bildskärm och radskrivare.
De subrutiner som tillsammans beskriver hela systemet heter DIST, SYSS, REGS ("discrete systems") och CONN ("connecting system"). DIST, SYSS och REGS är skrivna i FORTRAN och har länkats samman med SIMNON. CONN är skrivet i SIMNONS speciella simuleringsspråk. I ett appendix återges dessa och andra subrutiner som skrivits för att kunna generera SIMNON-element i sin helhet. Dessutom ges en länklista för en sådan generering.

Låt oss nu titta lite närmare på programmen.

DIST

Subrutinen DIST genererar bruset som driver våra system.

För att beräkna brussignalerna ($e_1$och e_2) använder DIST två sviter av oberoende, N(0,1) variabler. Med parametrarna NOD1 ochNOD2 kan man välja olika sådana sviter, ty dessa parametrar, som skall vara udda heltal, utgör startvärden till en slumptalsgenerator.

I tabell 3.1 sammanfattas utsignaler och parametrar. Med standardvärde avses det värde en parameter tilldelas då SIMNON-kommandot SYST ges.

DIST	Standardvärde	Kommentar
utsignaler		
ED1	-	$=e_1$
ED2	-	$=e_2$
parametrar		
R011	1	$=r_{11}=\text{var}(e_1)$
R022	1	$=r_{22}=\text{var}(e_2)$
R012	0	$=r_{12}=\text{cov}(e_1,e_2)$
NOD1	19	
NOD2	27	

Tabell 3.1 DIST-utsignaler och parametrar

SYSS

Denna subrutin beräknar utsignalerna från ett linjärt stokastiskt system, med två utsignaler och fyra insignaler, som kan beskrivas med modell (2.1).
Matriserna A, B, C har formen (med självklara beteckningar)

$$A(k) = \begin{pmatrix} A11(k) & A12(k) \\ A21(k) & A22(k) \end{pmatrix} \qquad \text{etc.}$$

och deras element lagras i parametervektorerna A11, A12, ..., B11, ..., C22. Dessa vektorer är deklarerade med dimensionen 5, men deras aktuella dimension, IACTV, fastställs med kommandot LET IACTV. = j, där $1 \leq j \leq 5$, givet för SYST.
Vi måste uppenbarligen kräva:

$$\left.\begin{matrix} NA \\ NB \\ NC \end{matrix}\right\} \leq IACTV$$

Dessutom krävs NB + KB $\leq$ 10
En sammanfattning återfinnes i tabell 3.2

SYSS	Standardvärde	Kommentar
<u>utsignaler</u>		
YS1, YS2	-	$=y_1, y_2$
<u>insignaler</u>		
US1, US2	-	styrsignaler, $= u_1, u_2$
ES1, ES2	-	brus , $= e_1, e_2$
<u>parametrar</u>		
NA, NB, NC	0	$\leq$ IACTV
KB	0	NB+KB$\leq$10
<u>parametervektorer</u>		
A11.........A22	0	aktuell dimension: IACTV
B11.........B22	0	
C11.........C22	0	

<u>Tabell 3.2</u> SYSS-utsignaler och parametrar

<u>REGS</u>

REGS innehåller STURE1 och STURE2 som de beskrivits i kapitel 2.

Vi ser i modellerna (2.2) - (2.4) att n_1' anger antalet α'-parametrar, n_2' antalet β'-parametrar och så vidare.
Dessa antal lagras i vektorerna N1 och N2.

$N1 = (n'_1, n'_2, n'_3, n'_4)$

$N2 = (n''_1, n''_2, n''_3, n''_4)$

Vi har tidigare kallat kdel för generell informationsfördröjning. Låt oss kalla k'_1, k'_2 .... för lokala signalfördröjningar. De lagras i vektorerna K1 och K2.

$K1 = (k'_1, k'_2, k'_3, k'_4)$

$K2 = (k''_1, k''_2, k''_3, k''_4)$

Parametrarna α'_1, α'_2 .... lagras i vektorerna TH1 och TH2.

$$TH1 = (\alpha'_1, \ldots, \alpha'_{n'_1}, \beta'_1, \ldots, \beta'_{n'_2}, \gamma'_1, \ldots, \gamma'_{n'_3}, \delta'_1, \ldots, \delta'_{n'_4})$$

$$TH2 = (\alpha''_1, \ldots, \alpha''_{n''_1}, \beta''_1, \ldots, \beta''_{n''_2}, \gamma''_1, \ldots, \gamma''_{n''_3}, \delta''_1, \ldots, \delta''_{n''_4})$$

Dessa vektorers aktuella dimension bestäms med kommandona (före SYST):

```
LET  IACT1. = j
LET  IACT2. = k
```

där $1 \le j, k \le 5$ (i mån av plats kan man höja den övre gränsen genom att ändra fältvidder och IP i SYST och SYSTSC).

TH1 och TH2 tilldelas startvärden med vektorerna TH01 och TH02.

Kovariansmatriserna i minsta kvadratalgoritmen heter P1 och P2. Som nämnts tilldelas endast diagonalelementen i dessa matriser startvärden $\neq 0$, vilket sker med vektorerna P01 och P02.

Startvärde för t.ex. P1 blir alltså matrisen

$$\begin{pmatrix} P01_1 & 0 & \cdots & 0 \\ 0 & P01_2 & & \\ \vdots & & \ddots & \\ 0 & & & P01_j \end{pmatrix}$$

där j är totala antalet parametrar som skattas i STURE1.

Viktfaktorn, λ, i minsta kvadratalgoritmen återfinns under namnet WT. Den ges startvärdet WTI och uppdateras i varje tidssteg enligt formeln:

WT: = WTM · WT + (1-WTM),

vilket innebär följande:

1. WTM=1 : WT= konstant = WTI
2. WTM<1 : WT konvergerar från WTI mot 1 med tiden.

Även för att beskriva parametern ULIM:s funktion skiljer vi på två fall:

1. ULIM ≥0 : styrsignalerna begränsas så att $|\hat{u}_i| \leq$ ULIM
2. ULIM <0 : Ingen begränsning.

Förluster summeras i variablerna V1, V2 och V.

$$V1 = \sum_{i=1}^{t} y_1(i)^2$$

$$V2 = \sum_{i=1}^{t} y_2(i)^2$$

V = V1 + V2

Huruvida β_0' och β_0'' skall skattas avgörs med parametern IBO.

1. IBO = 0 : β_0'=konstant=B01, β_0''=konstant=B02
2. IBO = 1 : β_0' och β_0'' skattas.

I matriserna S1 och S2 lagras gamla värden på y_1, y_2, u_1, u_2. De har aktuell dimension (4,j) efter kommandot:

LET ISB. = j, $2 \leq j \leq 10$

Tabell 3.3 visar de viktigaste komponenterna i REGS. Observera att indices ges utan parentes i SIMNON! Man skriver således ej B21(3), TH01(1) och P1(2,3) utan B213, TH011 och P123.

REGS	Standardvärde	Aktuell dimension	Kommentar
utsignaler			
UR1,UR2	-	-	beräknade styrsignaler; $\hat{u}_1,\hat{u}_2$
utsignalvektorer			
TH1	-	IACT1	innehåller parameterskattningar
TH2	-	IACT2	
insignaler			
Y1, Y2	-	-	verkliga utsignaler; y_1,y_2
U1, U2	-	-	verkliga styrsignaler; u_1,u_2
parametrar			
KDEL	0	-	generell informationsfördröjning
IBO	0	-	=0: β_0 skattas ej; =1 β_0 skattas
B01, B02	1	-	konstantvärden på β_0
WTI	1	-	initialvärden på λ
WTM	1	-	modulering av λ (om WTM<1)
ULIM	-1	-	begränsar $\hat{u}_1$, $\hat{u}_2$
parametervektorer			
N1, N2	0	4	antal parametrar
K1, K2	0	4	lokala signalfördröjningar
TH01	0	IACT1	initialvärden
TH02	0	IACT2	
P01	(100,100,....)	IACT1	initialdiagonaler till P1, P2
P02	(100,100,....)	IACT2	
variabler			
V1, V2, V	-	-	förluster
variabelmatriser			
P1	-	IACT1	kovariansmatriser
P2	-	IACT2	
S1, S2	-	(4,ISB)	lagrar y_1, y_2, u_1, u_2
parametermatris			
S0	0	(4,ISB)	initialvärden till S1, (S2)

Tabell 3.3 REGS - viktiga komponenter

CONN

Vi har nu tre program, som beskriver var sin isolerad bit av vårt stora system. För att kunna simulera hela systemet måste vi också deklarera samband mellan de tre programmens in- och utsignaler. Detta görs i CONN.

I fig. 3.1 återges CONN i sin helhet.

```
CONNECTING SYSTEM CONN
"
TIME T
"
ES1[SYSS]=ED1[DIST]
ES2[SYSS]=ED2[DIST]
Y1[REGS]=YS1[SYSS]
Y2[REGS]=YS2[SYSS]
U1[REGS]=UR1[REGS]
U2[REGS]=UR2[REGS]
US1[SYSS]=UR1[REGS]
US2[SYSS]=UR2[REGS]
"
END
```

Fig. 3.1 CONN

Exempel

Som demonstration av användningen av programmen avslutas detta kapitel med ett exempel.

Antag att ett system beskrivs av:

$$y(t)+\begin{pmatrix}-0.9 & 0.5\\ 0 & -0.8\end{pmatrix}y(t-1)+\begin{pmatrix}0 & 0.1\\ 0 & 0\end{pmatrix}y(t-2)=\begin{pmatrix}1 & 0\\ 0.2 & 1\end{pmatrix}u(t-1)+e(t)+\begin{pmatrix}-0.5 & 0\\ 0 & 0\end{pmatrix}e(t-1)$$

med $r_{11}=\mathrm{var}(e_1)=1$, $r_{22}=\mathrm{var}(e_2)=0.8$, $r_{12}=\mathrm{cov}(e_1,e_2)=0$

Låt regulatorn ha följande struktur:

$$\begin{cases} y_1(t)+\alpha_1' y_1(t-1)+\alpha_2' y_1(t-2)+\gamma' y_2(t-2)=u_1(t-1)+\beta' u_1(t-2)+\varepsilon_1(t) \\ y_2(t)+\alpha'' y_2(t-1) \qquad\qquad +\gamma'' y_1(t-1)=1.5u_2(t-1) \qquad\qquad +\varepsilon_2(t) \end{cases}$$

I fig. 3.2 visas en dialog som hölls med datorn för att simulera detta system. På skivan RK fanns filerna SIMNON XCT, SIMNON XXX och CONN SRC. Bildskärmens utseende efter simuleringen visas i fig. 3.3 och utskriften från radskrivaren, föranledd av kommandot DISP(LP) i fig. 3.4. Observera att

TH1 = $(\alpha_1', \alpha_2', \beta_1', \gamma')$
TH2 = $(\alpha'', \gamma'', -, -)$

```
DOS-15 UV3 A002
$A RK <SIM> 4

$A RK 3/NON 5,7,15,16

$BUFFS 5

$E SIMNON


>                                "KOMMENTARER
>
>LET IACTV.=2                     "AKTUELLA MATRISDIMENSIONER
>,IACT1.=4
>,IACT2.=4
>,ISB.=3
>SYST DIST SYSS REGS CONN         "SYSTEMBESKRIVNING
   WARNING: US1,US2 ARE UNDEFINED IN OUTPUT-SECTION OF SYSS
   WARNING: U1,U2 ARE UNDEFINED IN OUTPUT-SECTION OF REGS
>
>PAR NA:2                         "PARAMETRAR I SYSS


>,NB:1
>,NC:1
>,A111:-.9
>,A121:.5
>,A122:.1
>,A221:-.8
>,B111:1
>,B211:.2
>,B221:1
>,C111:-.5
>
>PAR RO22:.8                      "PARAMETER I DIST
>
>PAR N11:2                        "PARAMETRAR I REGS
>,N12:1
>,N13:1
>,K13:1
>,N21:1
>,N23:1
>,B02:1.5
>
>PLOT V1 V2
>AXES H 0 2000 V 0 4000
>SIMU 0 2000 -MARK                "SE FIG. 3.3
>
>DISP(LP)                         "SE FIG. 3.4
>STOP

STOP  000000


DOS-15 UV3 A002
$
```

Fig. 3.2   Exempel på simulering med SIMNON

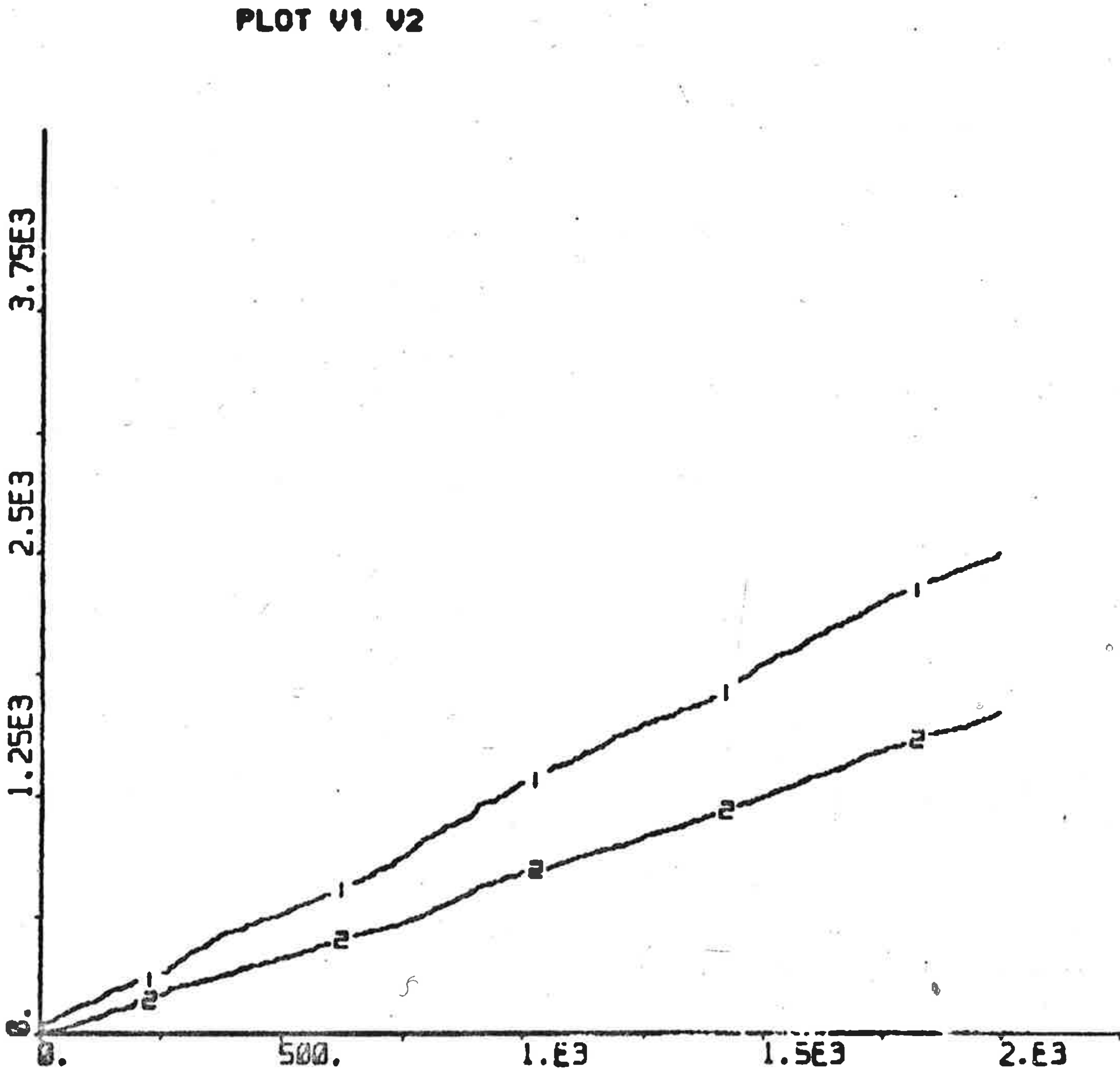


Fig. 3.3 Kopia av bildskärmen

```
                   DISCRETE SYSTEM DIST
OUTPUT: ED1      1.31230          ED2     8.935264E-02
TSAMP : TS       2001.00
PAR   : R011     1.00000          R012    0.000000         R022     0.800000
        NOD1     19.0000          NOD2     27.0000         DT       1.00000

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

                   DISCRETE SYSTEM SYSS
INPUT : US1    -0.469667          US2     0.156197         ES1      1.31230
        ES2    8.935264E-02
OUTPUT: YS1      1.30337          YS2     5.107574E-03
TSAMP : TS       2001.00
PAR   : NA       2.00000          NB       1.00000         NC       1.00000
        KB     0.000000           A111   -0.900000         A112     0.000000
        A121   0.500000           A122    0.100000         A211     0.000000
        A212   0.000000           A221   -0.800000         A222     0.000000
        B111     1.00000          B112    0.000000         B121     0.000000
        B122   0.000000           B211    0.200000         B212     0.000000
        B221     1.00000          B222    0.000000         C111    -0.500000
        C112   0.000000           C121    0.000000         C122     0.000000
        C211   0.000000           C212    0.000000         C221     0.000000
        C222   0.000000           DT       1.00000

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

                   DISCRETE SYSTEM REGS
INPUT : U1     -0.469667          U2      0.156197         Y1       1.30337
        Y2     5.107574E-03
OUTPUT: UR1    -0.469667          UR2     0.156197         TH11    -0.389117
        TH12   -7.698941E-02      TH13    0.110130         TH14     0.361886
        TH21    -1.15986          TH22    0.184306         TH23     0.000000
        TH24   0.000000           RES1     1.30337         RES2     3.405049E-03
TSAMP : TS       2001.00
PAR   : N11      2.00000          N12      1.00000         N13      1.00000
        N14    0.000000           N21      1.00000         N22      0.000000
        N23      1.00000          N24     0.000000         K11      0.000000
        K12    0.000000           K13      1.00000         K14      0.000000
        K21    0.000000           K22     0.000000         K23      0.000000
        K24    0.000000           KDEL    0.000000         B01      1.00000
        B02      1.50000          IBO     0.000000         TH011    0.000000
        TH012  0.000000           TH013   0.000000         TH014    0.000000
        TH021  0.000000           TH022   0.000000         TH023    0.000000
        TH024  0.000000           P011     100.000         P012     100.000
        P013     100.000          P014     100.000         P021     100.000
        P022     100.000          P023     100.000         P024     100.000
        S011   0.000000           S012    0.000000         S013     0.000000
        S021   0.000000           S022    0.000000         S023     0.000000
        S031   0.000000           S032    0.000000         S033     0.000000
        S041   0.000000           S042    0.000000         S043     0.000000
        R11    0.000000           R12     0.000000         R13      0.000000
        R14    0.000000           R21     0.000000         R22      0.000000
        R23    0.000000           R24     0.000000         ULIM    -1.00000
        WTI      1.00000          WTM      1.00000         DELT     0.000000
        REG      1.00000          DT       1.00000
VAR   : V1       2503.82          V2       1670.92         V        4174.73
        P111   4.847368E-04       P112   -6.613618E-05     P113     1.078367E-04
        P114   2.435514E-04       P121   -6.613618E-05     P122     1.465410E-03
        P123  -2.142503E-03       P124   -1.289082E-05     P131     1.078367E-04
        P132  -2.142503E-03       P133    4.312716E-03     P134    -2.570524E-06
        P141   2.435514E-04       P142   -1.289082E-05     P143    -2.570524E-06
        P144   7.217468E-04       P211    1.346896E-03     P212    -1.765882E-05
        P213   0.000000           P214    0.000000         P221    -1.765882E-05
```

Fig. 3.4 Utskrift från radskrivaren

P222	8.994618E-04	P223	0.000000	P224	0.000000
P231	0.000000	P232	0.000000	P233	0.000000
P234	0.000000	P241	0.000000	P242	0.000000
P243	0.000000	P244	0.000000	S111	-1.30337
S112	2.43819	S113	1.46468	S121	-0.469667
S122	1.08789	S123	1.24374	S131	-5.107574E-03
S132	8.402981E-02	S133	-0.434441	S141	0.156197
S142	-0.234600	S143	-0.515942	S211	-3.405049E-03
S212	5.601987E-02	S213	-0.289627	S221	0.156197
S222	-0.234600	S223	-0.515942	S231	-0.868913
S232	1.62546	S233	0.976450	S241	-0.313112
S242	0.725262	S243	0.829162		

CONNECTING SYSTEM CONN

TIME : T 2000.00

Fig. 3.4 (forts.)

4. SIMULERINGAR OCH BERÄKNINGAR

I detta kapitel skall vi studera några exempel. Vårt intresse inriktas därvid främst mot parametrarnas konvergensegenskaper och förlusternas storlek. Som förlustfunktioner används:

$$V_1 = Ey_1^2, \quad V_2 = Ey_2^2 \quad \text{och} \quad V = V_1 + V_2.$$

Om annat ej anges har simuleringarna omfattat 2100 steg och förlusterna beräknats för de sista 1000 stegen, dvs.

$$V_1 = \frac{1}{1000} \sum_{t=1101}^{2100} y_1(t)^2 \quad \text{etc.}$$

Parameterskattningarna har i allmänhet konvergerat efter 1000 steg och viktfaktorn λ har därefter knappast någon betydelse för resultatet. Om inget annat anges är $\lambda=1$.
Vad som nu följer är fem exempel. Exempel på enkelriktad interaktion, på dubbelriktad interaktion och på interaktion via styrsignalen samt slutligen två speciella,"svåra" exempel.

Exempel 4.1 - Enkelriktad interaktion

Betrakta ett system med modellen

$$y(t) + \begin{pmatrix} a_{11} & a_{12} \\ 0 & a_{22} \end{pmatrix} y(t-1) = u(t-1) + e(t) + \begin{pmatrix} c_{11} & 0 \\ 0 & c_{22} \end{pmatrix} e(t-1) \qquad (4.1)$$

och låt dem styras av två regulatorer med följande strukturer.

$$\begin{cases} y_1(t)+\alpha y_1(t-1)=u_1(t-1)+\varepsilon_1(t) & \text{(STURE1)} \\ y_2(t)+\alpha y_2(t-1)=u_2(t-1)+\varepsilon_2(t) & \text{(STURE2)} \end{cases}$$

Vi noterar att (del-)system 2 (undre raden i 4.1) i själva verket är oberoende av (del-)system 1 och under vissa förutsättningar finns en minimalvariansstrategi sådan att y_2 blir ett vitt brus.

Om STURE2 konvergerar mot en minimal variansregulator så kommer y_2 att uppträda som brus i system 1, som då också får frikopplad karaktär, vilket förenklar analysen.

För de system som simulerats gäller att:

$a_{11}= -0.9, \quad a_{22}= -0.8 , \quad r_{11}=r_{22}=1$

medan a_{12}, c_{11}, c_{22} och r_{12} varierar.
Vi skiljer mellan <u>tre fall</u> med olika karaktär på bruset.

A. Vitt okorrelerat brus

dvs. $c_{11}=c_{22}=0$, $r_{12}=0$.
Att $r_{12}=0$ innebär att e_1 och e_2 är okorrelerade.
<u>System 2</u> beskrivs alltså av modellen

$$y_2(t)+a_{22}y_2(t-1)=u_2(t-1)+e_2(t)$$

och en optimal strategi för STURE2 ges av minimalvariansstrategin:
$u_2(t)=a_{22}y_2(t)$, vilken medför $y_2(t)=e_2(t)$ och således
$V_2=\mathrm{var}(y_2)=r_{22}$.
<u>System 1</u> beskrivs i så fall av modellen

$$y_1(t)+a_{11}y_1(t-1)=u_1(t-1)+e_1(t)-a_{12}e_2(t-1).$$

"Minimalvariansstrategin" $u_1(t)=a_{11}y_1(t)$ ger:
$V_1=\mathrm{var}(y_1)=r_{11}+a_{12}^2 r_{22}$.

Under förutsättning att system 2 styrs med minimal varians blir således effekten av interaktionen att "extra" brus uppträder i system 1.

Det är nu intressant att se om STURE1 och 2 konvergerar mot minimal variansregulatorerna ovan.

Simuleringar

Figur 4.1 visar parameterskattningar och ackumulerad förlust i fallet $a_{12}=0.5$ $(r_{11}=r_{22}=1)$. Styrlagarna i regulatorerna är $u_1(t) = \alpha' y_1(t)$ resp. $u_2(t) = \alpha'' y_2(t)$ och vi ser att parametrarna konvergerar mot "rätt"värden (streckade i fig. 4.1). Förlusterna blev $V_1= 1.24$ och $V_2= 1.07$ per steg i god överensstämmelse med teorins $V_1= 1.25$, $V_2= 1.00$. (V_1 införes i tabell 4.1) Kanske kan överensstämmelsen mellan teori och praktik synas mindre god. Detta är dock endast en stokastisk effekt. Bruset genereras ju med hjälp av två sviter av stokastiska variabler och används andra sviter än ovan erhålls något annorlunda, ofta "snyggare", resultat.
Som exempel på vad som händer med andra brussviter kan nämnas $V_1= 1.26$, $V_2= 0.93$ samt parameterskattningar enligt fig. 4.2.

B. Färgat okorrelerat brus

dvs. $(c_{11}, c_{22}) \neq (0,0)$, $r_{12}= 0$.
På samma sätt som i föregående avsnitt erhålles:

System 2 : Minimalvariansstrategin är $u_2(t) = (a_{22}-c_{22})y_2(t)$ och ger $y_2(t) = e_2(t)$ och alltså $V_2= r_{22}$.

System 1: $y_1(t)+a_{11}y_1(t-1) = u_1(t-1)+e_1(t)+c_1e_1(t-1)-a_{12}e_2(t-1)$
Vi kan skriva:
$e_1(t)+c_1e_1(t-1)-a_{12}e_2(t-1) = e_0(t)+ce_0(t-1)$
där $e_0(t)$ är "ekvivalent", vitt brus med varians r_{00}.
Med $u_1(t) = (a_{11}-c)y_1(t)$ får vi minimal varians: $V_1= r_{00}$.

Simuleringar

Betrakta samma exempel osm ovan med tillägget $c_{11}= -0.5$, $c_{22}= -0.25$. Med dessa värden blir $c = -0.38$ och $r_{00}= 1.31$ och regulatorerna konvergerade mot motsvarande struktur.
(Se tabell 4.1)

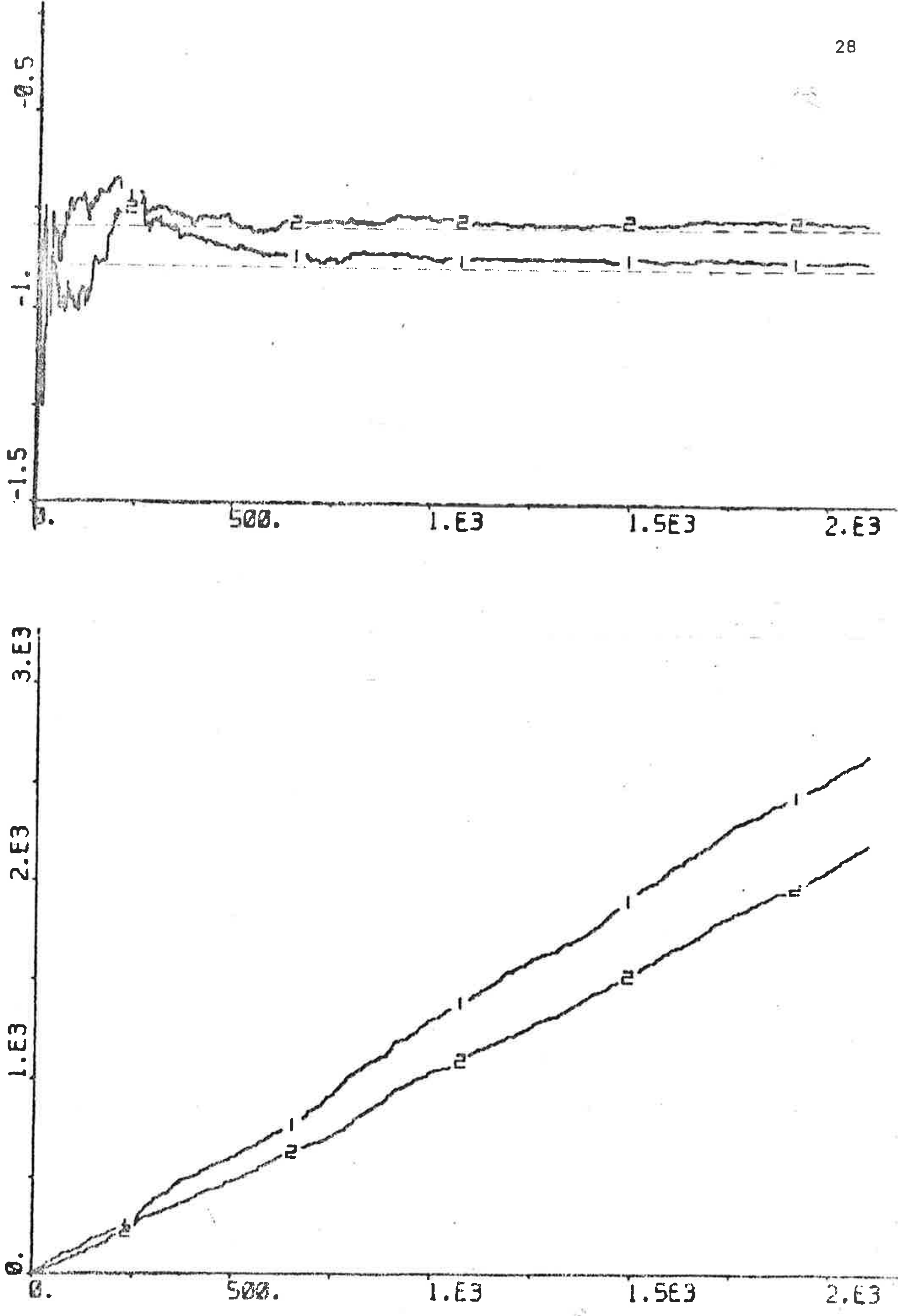
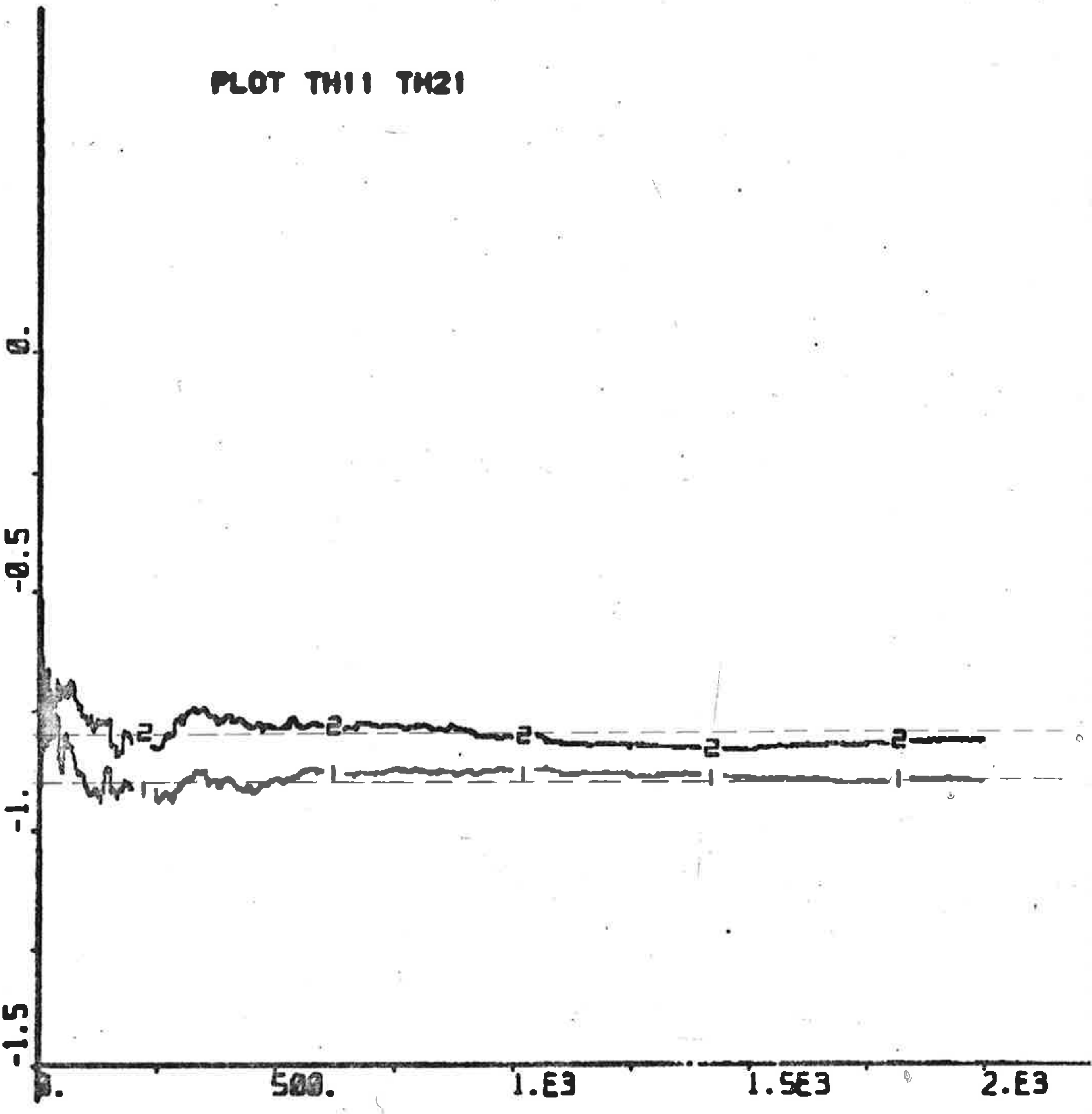


Fig. 4.1 Parameterskattningar och ackumulerad förlust
$a_{12}= 0.5$. ($r_{11}= r_{22}= 1$)



<u>Fig. 4.2</u> Parameterskattningar med andra brussviter än i fig. 4.1

C. Vitt, korrelerat brus

dvs. $c_{11}=c_{22}=0$, $r_{12}\neq 0$

System 2: $u_2(t) = a_{22}y_2(t)$ ger $y_2(t) = e_2(t)$ och alltså $V_2= r_{22}$.

System 1: $y_1(t)+a_{11}y_1(t-1)=u_1(t-1)+e_1(t)-a_{12}e_2(t-1)$

Vi skriver $e_1(t)-a_{12}e_2(t-1)=e_0(t)+ce_0(t-1)$ och får åter minimal varians, $V_1= r_{00}$ för $u_1(t)=(a_{11}-c)y_1(t)$.

Simuleringar

I dessa exempel är $c_{11}=c_{22}=0$, $r_{12}= 0.3, 0.7, 1.0$ och $a_{12}= 0.5, 0.9, -0.99, 3.0$.
Även här konvergerade regulatorerna mot minimal varians.
(Se tabell 4.1)

D. Några modifieringar

Om STURE1 får tillgång till $y_2(t)$, dvs. ges strukturen:

$y_1(t)+\alpha' y_1(t-1)+\gamma' y_2(t-1)=u_1(t-1)+\varepsilon_1(t)$

med styrlag:

$u_1(t)=\alpha' y_1(t)+\gamma' y_2(t)$ kan de förluster som interaktionen hittills förorsakats elimineras.

Strategin $u_1(t)=(a_{11}-c_{11})y_1(t)+a_{12}y_2(t)$ ($c_{11}=0$ i fall A och C) ger $V_1=r_{11}$. Simuleringar har också givit detta resultat.

Med tidsfördröjning i informationsflödet skulle STURE1 ej veta $y_2(t)$, men kanske $y_2(t-1)$.
Regulatorstrukturen blir då:

$y_1(t)+\alpha' y_1(t-1)+\gamma' y_2(t-2)=u_1(t-1)+\varepsilon_1(t)$

Styrlag: $u_1(t)=\alpha' y_1(t)+\gamma' y_2(t-1)$.

Vi får $y_1(t)+a_{11}y_1(t-1)+a_{12}y_2(t-1)=\alpha' y_1(t-1)+\gamma' y_2(t-2)+e_1(t)+c_{11}e_1(t-1)$ (*)

I fall A med vitt, okorrelerat brus kan vi inte uppnå några förbättringar med denna modell, men i fall B med färgat brus ger strategin

$u_1(t)=(a_{11}-c_{11})y_1(t)-a_{12}c_{11}y_2(t-1)$

variansen $V_1=r_{11}+a_{12}^2r_{22}$, dvs vi har eliminerat inverkan av "brusets färg".

Beträffande fall C ser vi att $y_2(t-2)$ inte är korrelerad med någon annan term i (*), varför vi inte väntar någon förbättring. Simuleringarna tyder dock på att vi kan uppnå små förbättringar jämfört med den enklare regulatorn.
I stället för till $y_2(t-1)$ kan vi ge STURE1 tillgång till $u_2(t-1)$. Detta ger likvärdiga resultat, ty styrlagen i STURE2 är $u_2(t) = \alpha'' y_2(t)$, där förhoppningsvis $\alpha'' \approx a_{22}-c_{22}$.
Vi har alltså $u_2(t) \approx$ konstant$\cdot y_2(t)$ och modellen $y_1(t)+\alpha' y_1(t-1)=u_1(t-1)+\delta' u_2(t-2)+\varepsilon_1(t)$ är ekvivalent med den tidigare $y_1(t)+\alpha' y_1(t-1)+\gamma' y_2(t-2)=u_1(t-1)+\varepsilon_1(t)$.

I tabell 4.1 visas V_1 i några simuleringar.
Vi konstaterar först att V_1 mycket sällan skiljer sig från den teoretiskt beräknade minimala variansen (given inom parentes).
Fallet att STURE1 vet $y_2(t)$ förutsätter att information kan överföras utan tidsfördröjning och resultatet blir det samma som skulle uppnåtts med central styrning.
I horisontell led kan vi nu se vad det kostar med "sant" decentraliserad reglering. Förlusterna ökar alltså avsevärt jämfört med "idealfallet" (med $y_2(t)$ i STURE1).
Däremot är skillnaderna mellan övriga fall avsevärt mindre.

brus	"inter- aktion" a_{12}	V_1 per steg om STURE1 har tillgång till: endast $y_1(t), u_1(t-1)$	dessutom $y_2(t)$	$y_2(t-1)$	$u_2(t-1)$
A. vitt, okorrelerat brus	0.5	1.24 (1.25)	1.02(1.00)	1.24(1.25)	1.24(1.25
B. färgat, okorrelerat brus $c_1 = -0.5$, $c_2 = -0.25$	0.5	1.31 (1.31)	1.02(1.00)	1.24(1.25)	1.24(1.25
C. vitt, korrelerat brus					
$r_{12} = 1.0$	0.5	1.01 (1.00)	1.01(1.00)	1.01	1.01
	0.9	1.02 (1.00)	" "	1.03	1.02
	0.99	1.11 (1.00)	" "	1.07	1.05
	3.0	9.29 (9.00)	" "	7.14	7.22
$r_{12} = 0.7$	0.5	1.14 (1.14)	1.01 "	1.13	1.13
	0.9	1.57 (1.55)	1.02 "	1.49	1.48
	0.99	1.74 (1.70)	1.02 "	1.60	1.60
$r_{12} = 0.3$	0.5	1.23 (1.23)	1.01 "	1.22	1.22
	0.9	1.95 (1.77)	1.02 "	1.75	1.75
	0.99	1.95 (1.94)	1.02 "	1.91	1.91

Tabell 4.1 Förluster vid enkelriktad interaktion.
Teoretiskt beräknade värden ges inom parentes.

Exempel 4.2 - Dubbelriktad interaktion.

Betrakta ett system givet av modellen

$$y(t)+\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} y(t-1)=u(t-1)+e(t)$$

Den optimala styrlagen för detta system är

$$u(t)=\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} y(t)$$

och ger $V_1=r_{11}$, $V_2=r_{22}$.
Vi vill emellertid inte tillåta informationsutbyte mellan regulatorerna utan skall uteslutande använda regulatorstrukturer vars styrlagar får formen

$$\begin{cases} u_1(t)=f_1(y_1(t),y_1(t-1),\ldots\ldots\ldots\ldots) \\ u_2(t)=f_2(y_2(t),y_2(t-1),\ldots\ldots\ldots\ldots) \end{cases}$$

I simuleringarna gäller: $a_{11}= -0.9$, $a_{22}= -0.8$, $r_{22}= 1$, $r_{12}= 0$ och om annat ej anges: $a_{12}= 0.5$, $r_{11}= 1$

1 + 1 parametrar

Vi försöker först med den "gamla" strukturen med en parameter i varje STURE:

$$\begin{cases} y_1(t)+\alpha' y_1(t-1)=u_1(t-1)+\varepsilon_1(t) \\ y_2(t)+\alpha'' y_2(t-1)=u_2(t-1)+\varepsilon_2(t) \end{cases}$$

Styrlag: $\begin{cases} u_1(t)=\alpha' y_1(t) \\ u_2(t)=\alpha'' y_2(t) \end{cases}$

För det slutna systemet gäller alltså:

$$y(t)+\begin{pmatrix} a_{11}-\alpha' & a_{12} \\ a_{21} & a_{22}-\alpha'' \end{pmatrix} y(t-1)=e(t) \qquad \Leftrightarrow$$

$$(I+Aq^{-1})y(t)=e(t) \qquad \text{där } A=\begin{pmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{pmatrix} = \begin{pmatrix} a_{11}-\alpha_1 & a_{12} \\ a_{21} & a_{22}-\alpha_2 \end{pmatrix} \qquad \Leftrightarrow$$

$$y(t)=(I+Aq^{-1})^{-1}e(t) \qquad \Leftrightarrow$$

$$\begin{cases} y_1(t)=\{(1+k_{22}q^{-1})e_1(t)-k_{12}q^{-1}e_2(t)\} / \{1+(k_{11}+k_{22})q^{-1}+(k_{11}k_{22}-k_{12}k_{21})q^{-2}\} \\ y_2(t)=\{-k_{21}q^{-1}e_1(t)+(1+k_{11}q^{-1})e_2(t)\} / \{1+(k_{11}+k_{22})q^{-1}+(k_{11}k_{22}-k_{12}k_{21})q^{-2}\} \end{cases}$$

Vi kan nu beräkna V_1 och V_2 på formen

$$\begin{cases} V_1= Ey_1^2=f_{11}\cdot r_{11}+f_{12}\cdot r_{22} \\ V_2= Ey_2^2=f_{21}\cdot r_{11}+f_{22}\cdot r_{22} \end{cases}$$

och sedan $V = V_1+V_2$.

Att direkt minimera V med avseende på (α_1,α_2) dvs. (k_{11},k_{22}) är tyvärr ganska knepigt.
Låt oss försöka en omväg!
Enligt [2] gäller att om parameterskattningarna konvergerar så gäller:

$$\begin{cases} r_{y_1}(1)=Ey_1(t)y_1(t-1)=0 \\ r_{y_2}(1)=Ey_2(t)y_2(t-1)=0 \end{cases}$$

En lösning till detta ekvationssystem är $k_{11}=k_{22}=0$ och åtminstone om $r_{11}=r_{22}$ är detta också den enda stabila lösningen.

Nu är det naturligt att fråga sig om V har minimum i origo. Som svar får vi nöja oss med att origo åtminstone är en stationär punkt $\left(\frac{\delta V}{\delta k_{11}} = \frac{\partial V}{\partial k_{22}} = 0\right)$.

Observera att $\begin{cases} k_{11} = 0 \\ k_{22} = 0 \end{cases}$ motsvarar $\begin{cases} \alpha' = a_{11} \\ \alpha'' = a_{22} \end{cases}$

Vid simulering konvergerade regulatorparametrarna mot dessa värden och förlusterna mot motsvarande värden, se tabell 4.2.

	central styrning			decentraliserad styrning					
	teoretiskt optimalt			teoretiskt för $k_{11}=k_{22}=0$			simulerat		
a_{21}	V_1	V_2	V	V_1	V_2	V	V_1	V_2	V
0.1	1.00	1.00	2.00	1.25	1.01	2.26	1.25	1.09	2.34
0.5	1.00	1.00	2.00	1.33	1.33	2.67	1.33	1.44	2.77
1.0	1.00	1.00	2.00	1.67	2.67	4.34	1.65	2.80	4.45

Tabell 4.2 Förluster vid central resp. decentraliserad styrning.

Med hjälp av program utarbetade vid institutionen beräknades kovarianserna $r_{y_1}(\tau)$ och $r_{y_2}(\tau)$.

I tabell 4.3 visas autokorrelationsfunktionerna $\rho_{y_1}(\tau)$ och $\rho_{y_2}(\tau)$ $\left(\rho_y(\tau) = \frac{r_y(\tau)}{r_y(0)}\right)$ för simuleringen med $a_{21} = 0.5$.

-Mycket riktigt är $r_{y_1}(1)$ och $r_{y_2}(1)$ mycket små.

τ	$\rho_{y_1}(\tau)$	$\rho_{y_2}(\tau)$
1	$9{,}7\ 10^{-3}$	$-1{,}7\ 10^{-2}$
2	0,26	0,23
3	$2{,}8\ 10^{-2}$	$2{,}1\ 10^{-2}$
4	$-7{,}5\ 10^{-4}$	$3{,}9\ 10^{-2}$
5	$1{,}1\ 10^{-2}$	$-3{,}4\ 10^{-2}$
6	$-7{,}2\ 10^{-2}$	$1{,}7\ 10^{-2}$
7	$1{,}5\ 10^{-2}$	$1{,}7\ 10^{-4}$
8	$1{,}2\ 10^{-2}$	$4{,}9\ 10^{-2}$
9	$6{,}2\ 10^{-2}$	$9{,}5\ 10^{-3}$
10	$1{,}5\ 10^{-2}$	$5{,}5\ 10^{-2}$

Tabell 4,3 Autokorrelation för $a_{12}=a_{21}=0{,}5$

2 + 1 (1+2) parametrar

Som rubriken antyder ändrar vi modellen till

$$\begin{cases} y_1(t)+\alpha_1' y_1(t-1)+\alpha_2' y_1(t-2)=u_1(t-1)+\varepsilon_1(t) \\ y_2(t)+\alpha_1'' y_2(t-1) \qquad\qquad =u_2(t-1)+\varepsilon_2(t) \end{cases}$$

Styrlag: $\begin{cases} u_1(t)=\alpha_1' y_1(t)+\alpha_2' y_1(t-1) \\ u_2(t)=\alpha_1'' y_2(t) \end{cases}$

Vi får:

$$\begin{pmatrix} 1+(a_{11}-\alpha_1')q^{-1}-\alpha_2' q^{-2} & a_{12}q^{-1} \\ a_{21}q^{-1} & 1+(a_{22}-\alpha_1'')q^{-1} \end{pmatrix} y(t) = e(t)$$

$$\Longleftrightarrow$$

$$y(t) = \frac{1}{1+\{(a_{11}-\alpha_1')+(a_{22}-\alpha_1'')\}q^{-1}+\{(a_{11}-\alpha_1')(a_{22}-\alpha_1'')-(\alpha_2'+a_{12}a_{21})\}q^{-2}-\alpha_2'(a_{22}-\alpha_1'')q^{-3}} \cdot$$

$$\cdot \begin{pmatrix} 1+(a_{22}-\alpha_1'')q^{-1} & -a_{12}q^{-1} \\ -a_{21}q^{-1} & 1+(a_{11}-\alpha_1')q^{-1}-\alpha_2'q^{-2} \end{pmatrix} e(t)$$

Om $\alpha_1'=a_{11}$, $\alpha_1''=a_{22}$ och $\alpha_2'=-a_{12}a_{21}$ erhålles

$$\begin{cases} y_1(t)=e_1(t)-a_{12}e_2(t-1) \\ y_2(t)=-a_{21}e_1(t-1)+e_2(t)+a_{12}a_{21}e_2(t-2) \end{cases}$$

dvs. glidande medelvärde och $r_{y_1}(1)=r_{y_1}(2)=r_{y_2}(1)=0$ som sig bör.

$$\begin{cases} V_1=r_{11}+a_{12}^2r_{22} \\ V_2=a_{21}^2r_{11}+(1+a_{12}^2a_{21}^2)r_{22} \end{cases}$$

$$\Rightarrow \quad V^1=(1+a_{21}^2)r_{11}+(a_{12}^2a_{21}^2+1+a_{12}^2)r_{22}$$

Om vi i stället låter STURE2 ha 2 parametrar (alltså 1+2) får vi på samma sätt (med $\alpha_1'=a_{11}$, $\alpha_1''=a_{22}$ och $\alpha_2''=-a_{12}a_{21}$)

$$\begin{cases} V_1=(1+a_{12}^2a_{21}^2)r_{11}+a_{12}^2r_{22} \\ V_2=a_{21}^2r_{11} + r_{22} \end{cases}$$

$$\Rightarrow V^2=(1+a_{21}^2+a_{12}^2a_{21}^2)r_{11}+(1+a_{12}^2)r_{22}$$

I simuleringarna har regulatorparametrarna konvergerat mot de värden vi "gissat" ovan, och i tabell 4.4 visas förlusterna i några fall.

Av $V^1-V^2=a_{12}^2a_{21}^2(r_{22}-r_{11})$ framgår att man bör ha 2 parametrar i STURE1 om $r_{11}>r_{22}$ och vice versa.

Detta förhållande illustreras nederst i tabell 4.4 där $r_{11}-r_{22}$. V minskar om vi ökar parameteruppsättningen från 1+1 till 1+2 och blir ännu mindre för 2+1.

a_{21}	V_1	V_2	V	
1 + 1 parametrar				
0.1	1.25 (1.25)	1.09 (1.01)	2.34 (2.26)	
0.5	1.33 (1.33)	1.44 (1.33)	2.77 (2.67)	
1.0	1.65 (1.67)	2.80 (2.67)	4.45 (4.34)	
1.0	5.57 (5.67)	6.80 (6.67)	12.37(12.33)	$r_{11}=4$
2 + 1 parameter				
a_{21}	V_1	V_2	V	
0.1	1.24 (1.25)	1.09 (1.01)	2.33 (2.26)	
0.5	" "	1.42 (1.31)	2.66 (2.56)	
1.0	" "	2.40 (2.25)	3.64 (3.50)	
1.0	4.25 (4.25)	5.49 (5.25)	9.74 (9.50)	$r_{11}=4$
1 + 2 parametrar				
a_{21}	V_1	V_2	V	
0.1	1.25 (1.25)	1.09 (1.01)	2.34 (2.26)	
0.5	1.33 (1.31)	1.37 (1.25)	2.70 (2.56)	
1.0	1.54 (1.50)	2.22 (2.00)	3.76 (3.50)	
1.0	5.38 (5.25)	5.32 (5.00)	10.70(10.24)	$r_{11}=4$

Tabell 4.4 Förluster vid olika parameteruppsättningar. Beräknade värden är givna inom parentes.

2 + 2 parametrar

Det är nu naturligt att studera vad som händer om både STURE1 och STURE2 har två parametrar, dvs. ges modellerna:

$$\begin{cases} y_1(t)+\alpha_1' y_1(t-1)+\alpha_2' y_1(t-2)=u_1(t-1)+\varepsilon_1(t) \\ y_2(t)+\alpha_1'' y_2(t-1)+\alpha_2'' y_2(t-2)=u_2(t-1)+\varepsilon_2(t) \end{cases}$$

För det slutna systemet gäller då

$$Y(t)=\frac{1}{1+\{(a_{11}-\alpha_1')+(a_{22}-\alpha_1'')\}q^{-1}+\{(a_{11}-\alpha_1')(a_{22}-\alpha_1'')-(\alpha_2'+\alpha_2''+a_{12}a_{21})\}q^{-2}-\{\alpha_2'(a_{22}-\alpha_1'')+\alpha_2''(a_{11}-\alpha_1')\}q^{-3}+\alpha_2'\alpha_2''q^{-4}} \cdot \begin{pmatrix} 1+(a_{22}-\alpha_1'')q^{-1}-\alpha_2''q^{-2} & -a_{12}q^{-1} \\ -a_{21}q^{-1} & 1+(a_{11}-\alpha_1')q^{-1}-\alpha_2'q^{-2} \end{pmatrix} \cdot e(t)$$

Kommer parametrarna att konvergera så att $\alpha_2'=0$ eller $\alpha_2''=0$? Det skulle betyda att vi kom tillbaka till situationen i förra avsnittet. – Eller finns det några parametervärden som ger bättre reglering?

I tabell 4.5 a) och fig. 4.3 visas resultatet för systemet

$$y(t)+\begin{pmatrix} -0.9 & 0.5 \\ 1.0 & -0.8 \end{pmatrix}\cdot y(t-1)=u(t-1)+e(t)$$

dvs. $a_{12}=0.5$, $a_{21}=1.0$, $-a_{12}a_{21}=-0,5$ med $r_{11}=4$, $r_{22}=1$.

Ingen av parametrarna konvergerar således mot noll, men i varje fall blir $|\alpha_2'|>|\alpha_2''|$ och förlusten inte mycket större än i idealfallet $\alpha_2''=0$.

Del b) av tabell 4.5 redovisar en undersökning av huruvida extremfallen $\alpha_2'=0$ respektive $\alpha_2''=0$ är stabila. I dessa simuleringar har parametrarna fått startvärdena:

$\alpha_{10}'= -0.9$, $\alpha_{10}''= -0.8$ samt

$$\begin{cases} \alpha_{20}'= -0.5 \\ \alpha_{20}''= 0 \end{cases} \quad \text{resp.} \begin{cases} \alpha_{20}'=0 \\ \alpha_{20}''=-0.5 \end{cases}$$

dessutom diagonalelementen i minstakvadratalgoritmens kovariansmatriser $p_0=0.001$, vilket innebär att regulatorerna sätter stor tilltro till startvärdena. Konvergensen blir då något långsammare beroende på att elementen i kovariansmatriserna blir mycket små fortare än tidigare.

Efter 4100 steg var

$\begin{cases} \alpha_2' = -0.49 \\ \alpha_2'' = -0.096 \end{cases}$ (se fig, 4.4) resp. $\begin{cases} \alpha_2' = -0.44 \\ \alpha_2'' = -0.23 \end{cases}$

Extremfallen $\alpha_2'=0$ respektive $\alpha_2''=0$ tycks således inte vara stabila.

	parametrar	startvärden	V_1	V_2	V	α_2'	α_2''
a.)	1+1	$\alpha_{j0}=0$, $p_0=100$	5.57	6.80	12.37	-	-
	2+1		4.25	5.49	9.73	-0.51	-
	1+2		5.38	5.32	10.70	-	-0.50
	2+2		4.32	5.50	9.81	-0.49	-0.13
b.)	2+2	$\alpha_{10}'=-0.9$, $\alpha_{10}''=-0.8$, $p_0=0.001$	4.29	5.48	9.77	-0.51	-0.091
	2+2	$\alpha_{20}'=-0.5, \alpha_{20}''=0$ / $\alpha_{20}'=0, \alpha_{20}''=-0.5$	4.50	5.55	10.05	-0.41	-0.27

<u>Tabell 4.5</u> Parameterskattningar efter 2100 steg och förluster.
$r_{11}=4$, $r_{22}=1$, $a_{12}=0.5$, $a_{21}=1.0$

Enligt tidigare avsnitt gäller för systemet ovan att kvoten mellan förlusterna för 1+2 resp. 2+1 parametrar är:

$\frac{V^2}{V^1} = 1.10$. Teorin ovan ger $\frac{V^2}{V^1} = 1.08$.

Vi skall nu se ett exempel där denna kvot är större. För $a_{12}= a_{21}=1.0$, $r_{11}=4$, $r_{22}=1$ ger teorin $\frac{V^2}{V^1} = 1.27$ och praktiken

$\frac{V^2}{V^1} = 1.24$ (se nedan)

Regleringen av detta system är förenad med vissa svårigheter. Således visade det sig omöjligt att reglera med 1+1 parameter, och man kan visa att det slutna systemet är instabilt - åtminstone för $\alpha_1'=a_{11}$, $\alpha_1''=a_{22}$ dvs. $k_{11}=k_{22}=0$.

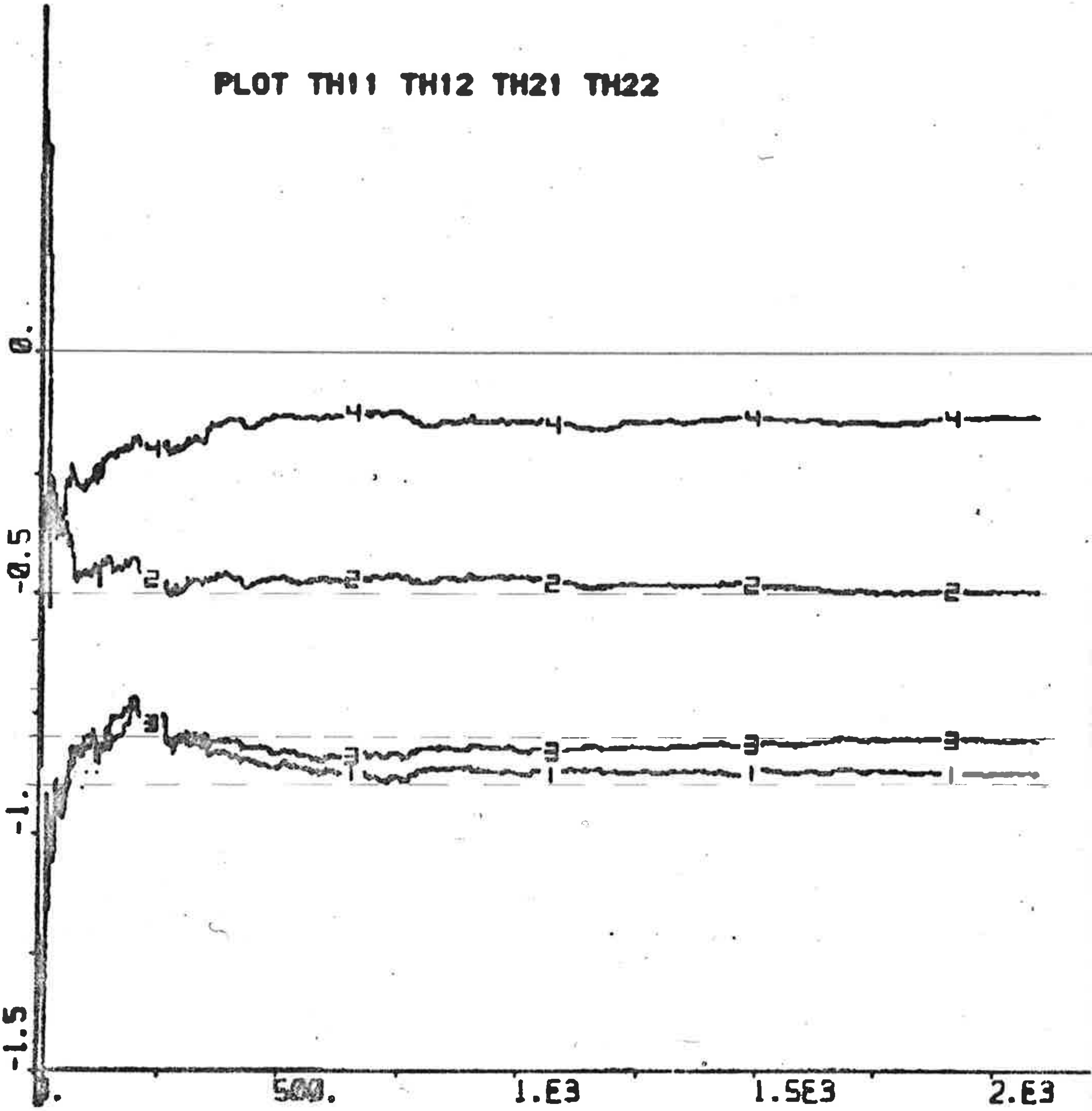


Fig. 4.3 Parameterskattningar för $a_{12}=0.5$, $a_{21}=1.0$, $r_{11}=4$, $r_{22}=1$
Skattningarna startar i origo och $p_0=100$.

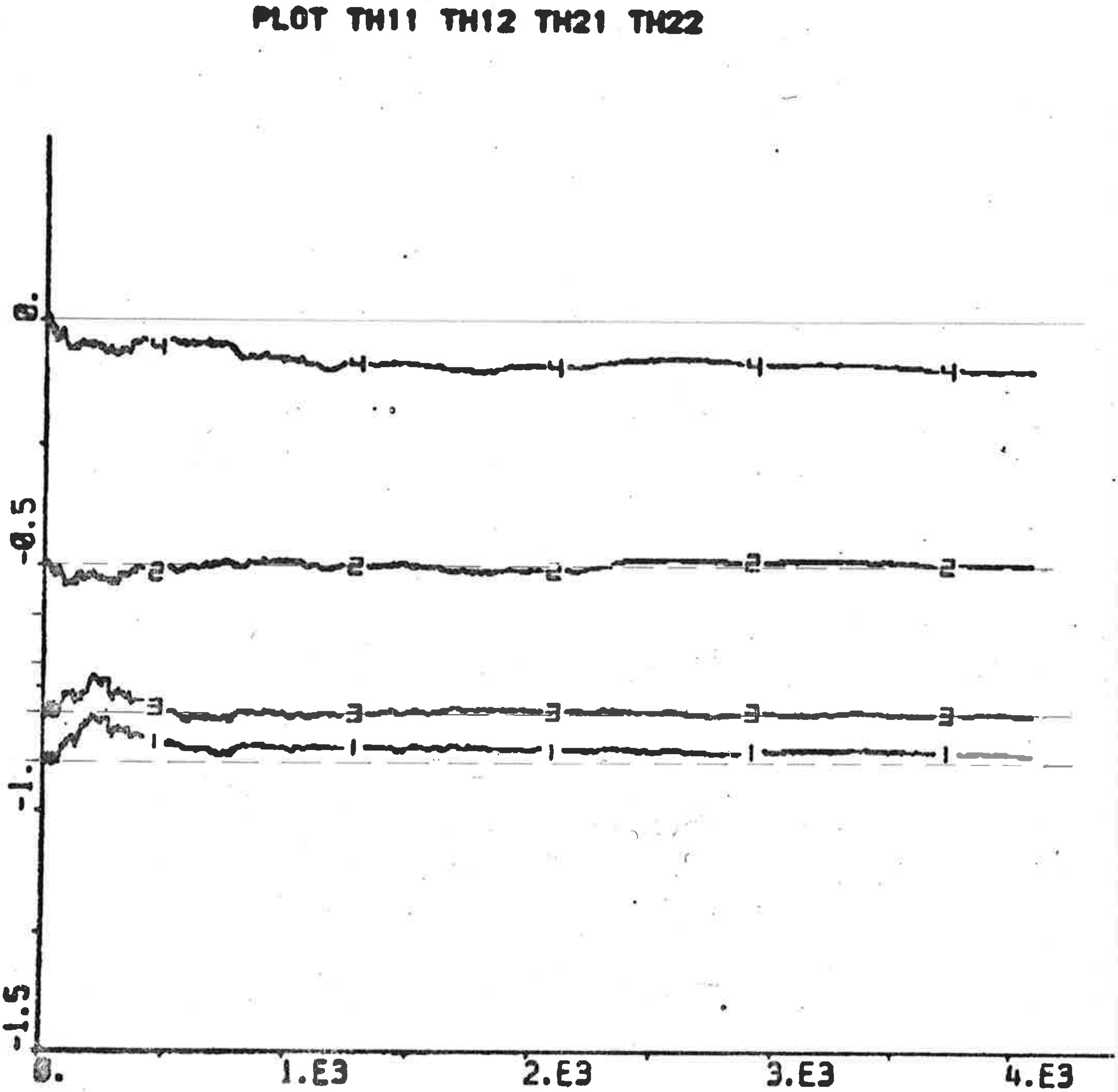


Fig. 4.4 Parameterskattningar som i fig. 4,3, men skattningarna startar i 2+1 -konfiguration och p_0=0,001.

Med flera parametrar konvergerade emellertid regulatorerna, ehuru långsamt. För att råda bot på denna slöhet användes "exponentiell glömska" (se avsnitt 2) med glömskefaktorn $\lambda = 0,995$. Parametrarna svängde då in sig avsevärt snabbare men oscillerade kraftigare även vid jämvikt, varför den andra decimalen i parameterskattningarna bör tas med en nypa salt. (se fig. 4.5) I tabell 4.6 visas parameterskattningarna efter 5000 steg samt förlusterna mellan T = 5000 och T = 6000. Även i detta exempel ger 2+2-konfigurationen förluster av samma storlek som den bättre av de enklare konfigurationerna.

parametrar	startvärden	V_1	V_2	V	α_2'	α_2''
2+1	$\alpha_{J0}=0$	4.72	5,65	10,37	-1,00	-
1+2	$p_0=100$	8,15	4,69	12,84	-	-0,99
2+2		4,79	5,55	10,34	-0,96	-0,20

Tabell 4.6 Parameterskattningar efter 5000 steg och förluster per steg mellan T=5000 och T=6000.
$r_{11}=4$, $r_{22}=1$, $a_{12}=1.0$, $a_{21}=1.0$, $\lambda = 0,995$

Extremfallen $\begin{cases}\alpha_{20}' = -1.0\\ \alpha_{20}'' = 0.0\end{cases}$ resp. $\begin{cases}\alpha_{20}'=0.0\\ \alpha_{20}''=-1,0\end{cases}$ undersöktes

med $p_0=0.0001$, men tycks inte vara stabila.

Färgat och korrelerat brus

Många av de exempel, av vilka några relateras ovan, som simulerats med vitt, okorrelerat brus har testats dels med färgat och dels med korrelerat brus.

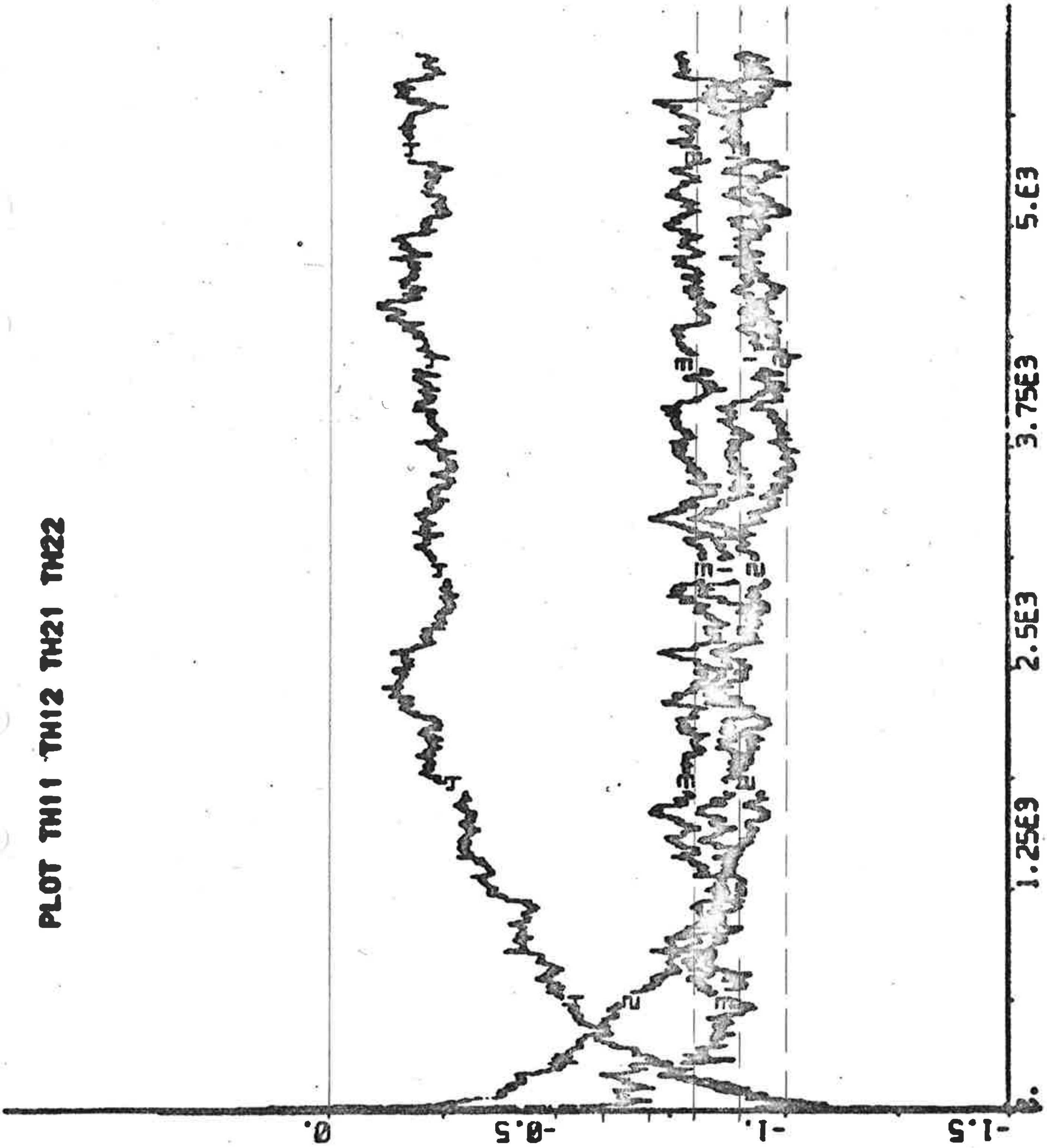


Fig. 4.5 Parameterskattningar för $a_{12}=a_{21}=1$, $r_{11}=4$, $r_{22}=1$
Skattningarna startar i origo och p=100, $\doteq$ 0,995

Resultaten från dessa simuleringar ligger helt i linje med vad som redan beskrivits, dvs. man kan så att säga superponera resultaten från exempel 4.1 och exempel 4.2.

Anmärkning

I några av simuleringarna i detta avsnitt har styrsignalerna begränsats (ulim=15) för att minska förlusterna under insvängningsförloppet. Är dessa mycket större än förlusterna i senare skeden kan nämligen den ackumulerade förlusten bli så stor under insvängningen att signifikansen sedan inte räcker för att med nöjaktig noggrannhet beräkna de förändringar som förlusterna åstadkommer (dvs, förlusterna per steg).
Någon icke önskvärd effekt torde denna åtgärd inte ha haft, ty utsignalerna nådde gränsen endast i ett fåtal steg.

Exempel 4.3 - Koppling genom styrsignalen

Vi studerar systemen

$$y(t)+\begin{pmatrix} a_{11} & 0 \\ 0 & a_{22} \end{pmatrix} y(t-1)=\begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix} u(t-1)+e(t)$$

Den optimala styrlagen är

$$u(t)=\begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix}^{-1} \begin{pmatrix} a_{11} & 0 \\ 0 & a_{22} \end{pmatrix} y(t)$$

och ger $V_1=r_{11}$, $V_2=r_{22}$.

Låt oss reglera med STURE1 och STURE2 med samma strukturer som förut, dvs. exempelvis (1+2)

$$\begin{cases} y_1(t)+\alpha_1' y_1(t-1) = u_1(t-1)+\varepsilon_1(t) \\ y_2(t)+\alpha_1'' y_2(t-1)+\alpha_2'' y_2(t-2)=u_2(t-1)+\varepsilon_2(t) \end{cases}$$

Den teoretiska behandlingen blir något mer komplicerad än den i exempel 4,2, om än likartad. Vi nöjer oss med simuleringen.

<u>Exempel</u> : $y(t)+\begin{pmatrix} -0{,}9 & 0 \\ 0 & -0{,}8 \end{pmatrix} y(t-1) = \begin{pmatrix} 1 & 0{,}5 \\ -0{,}7 & 1 \end{pmatrix} u(t-1)+e(t)$

Resultaten följer samma mönster som tidigare,: jämför tabell 4,7 där $r_{11}=1$, $r_{22}=4$.

parametrar	startvärden	V_1	V_2	V
1+1	$\alpha_{j0}=0$	1,69	4,76	6,45
2+1	$P_0=100$	1,55	4,78	6,34
1+2		1,67	4,47	6,13
2+2		1,61	4,52	6,13

<u>Tabell 4,7</u> Förluster vid koppling genom styrsignalen. $r_{11}=1$, $r_{22}=4$

<u>Exempel 4,4 – Tillämpningsexempel 1</u>

Detta exempel omfattar ett system som visade sig svårt att styra.

Vi utgår från ett kontinuerligt system med överföringsfuktionen:

$$G(s) = \frac{1}{1{,}25(s+1)(s+2)} \begin{pmatrix} s-1 & s \\ -6 & s-2 \end{pmatrix} \qquad (4{,}2)$$

Den diskreta motsvarigheten med samplingstiden 0,5 s blir:

$$y(t)+\begin{pmatrix} -0{,}9744 & 0 \\ 0 & -0{,}9744 \end{pmatrix} y(t-1) \begin{pmatrix} 0{,}2231 & 0 \\ 0 & 0{,}2231 \end{pmatrix} y(t-2) = \begin{pmatrix} 0{,}1290 & 0{,}1909 \\ -0{,}3716 & 0{,}0671 \end{pmatrix} u(t-1)+$$

$$+\begin{pmatrix} -0{,}2285 & -0{,}1909 \\ -0{,}2254 & -0{,}2660 \end{pmatrix} u(t-2)+e(t) \qquad (4{,}3)$$

$\Leftrightarrow$

$$\overset{\text{def.}}{\Longleftrightarrow}$$

$$y(t)+A_1y(t-1)+A_2y(t-2) = B_1u(t-1)+B_2u(t-2)+e(t)$$

En minimalvariansregulator ges av

$$u(t)=B_1^{-1}\{A_1y(t)+A_2y(t-1)-B_2u(t-1)\} \quad \Longleftrightarrow$$

$$u(t)=\begin{pmatrix}-0.821 & 2.337\\ -4.549 & -1.579\end{pmatrix}y(t)+\begin{pmatrix}0.188 & -0.535\\ 1.042 & 0.362\end{pmatrix}y(t-1)-\begin{pmatrix}0.348 & 0.477\\ -1.432 & -1.322\end{pmatrix}u(t-1)$$

Med motsvarande startvärden på parametrarna i regulatorerna

$$\begin{cases} y_1(t)+\alpha_1'y_1(t-1)+\alpha_2'y_1(t-2)+\gamma'y_2(t-1)=u_1(t-1)+\beta'u_1(t-2)+\delta'u_2(t-2)+\varepsilon_1(t)\\ y_2(t)+\alpha_1''y_2(t-1)+\alpha_2''y_2(t-2)+\gamma''y_1(t-1)=u_2(t-1)+\beta''u_2(t-2)+\delta''u_1(t-2)+\varepsilon_2(t)\end{cases}$$

dvs. styrlagarna

$$u(t)=\begin{pmatrix}\alpha_1' & \gamma'\\ \gamma'' & \alpha_1''\end{pmatrix}y(t)+\begin{pmatrix}\alpha_2' & 0\\ 0 & \alpha_2''\end{pmatrix}y(t-1)-\begin{pmatrix}\beta' & \delta'\\ \delta'' & \beta''\end{pmatrix}u(t-1)$$

och $p_0=0.005$ erhölls vid simulering ($r_{11}=r_{22}=1$):

$V_1=1.05$, $V_2=1.08$. Det gick lika bra med $p_0=100$ och dessutom med $\lambda=0.995$, men om parameterskattningarna fick starta i origo "havererade" systemet efter några tusen steg. Parameterskattningarna ändras plötsligt radikalt och förlusterna blev "oändliga". Med färre parametrar i regulatorerna tycks endast de strukturer som användes i exempel 4.2 (1+1, ...,2+2) ge stabila system. Förlusterna blir dock mycket stora, för 1+1 c:a 40 per steg och för 2+2 c:a 90 per steg. (Styrsignalen uppfyllde ungefär $|\hat{u}|\leq 100$)

Avslutningsvis följer en möjlig förklaring till svårigheten i detta exempel. Enligt [4] är det en nödvändig förutsättning för minimalvariansreglering att systemet i fråga är minimum-fas.

Systemet är visserligen minimum-fas, men försummas kopplingarna (dvs. alla matriser i (4.2) och (4.3) är diagonalmatriser) återstår delsystem som är icke minimum-fas.

Exempel 4.5 - Tillämpningsexempel 2: Rosenbrock's exempel.

Detta sista exempel visar att STURE's liv inte enbart är en dans på rosor.

Betrakta ett kontinuerligt system med överföringsfunktion

$$G(s) = \begin{pmatrix} \frac{1}{s+1} & \frac{2}{s+3} \\ \frac{1}{s+1} & \frac{1}{s+1} \end{pmatrix}$$

Den diskreta motsvarigheten med T=0.1s blir:

$$y(t)+\begin{pmatrix} -1.6457 & 0 \\ 0 & -0.9048 \end{pmatrix} y(t-1)+\begin{pmatrix} 0.6703 & 0 \\ 0 & 0 \end{pmatrix} y(t-2) =$$

$$=\begin{pmatrix} 0.0952 & 0.1728 \\ 0.0952 & 0.0952 \end{pmatrix} u(t-1)+\begin{pmatrix} -0.0705 & -0.1563 \\ 0 & 0 \end{pmatrix} u(t-2)+e(t) \qquad (4.4)$$

$$\Longleftrightarrow \quad (\text{def.})$$

$$y(t)+A1y(t-1)+A2y(t-2)=B1u(t-1)+B2u(t-2)+e(t)$$

Ekvationen $\det(B1+B2q^{-1})=0$ har här lösningar utanför enhetscirkeln (gäller även om $T<0.1s$). Systemet är alltså icke minimum-fas och enligt [4] skall återkoppling med en minimalvariansregulator ge instabilitet. Jag försökte i alla fall med

$$u(t)=B1^{-1}\{A1y(t)+A2y(t-1)-B2u(t-1)\}$$

$$u(t)=\begin{pmatrix} 21.21 & -21.16 \\ -21.21 & 11.66 \end{pmatrix} y(t)+\begin{pmatrix} -8.638 & 0 \\ 8.638 & 0 \end{pmatrix} y(t-1)-\begin{pmatrix} 0.9085 & 2.014 \\ -0.9085 & -2.014 \end{pmatrix} u(t-1)$$

dvs. modellerna

$$\begin{cases} y_1(t)+\alpha_1' y_1(t-1)+\alpha_2' y_1(t-2)+\gamma' y_2(t-1)=u_1(t-1)+\beta' u_1(t-2)+\delta' u_2(t-2)+\varepsilon_1(t) \\ y_2(t)+\alpha'' y_2(t-1)+\gamma_1'' Y_1(t-1)+\gamma_2'' y_1(t-2)=u_2(t-1)+\beta'' u_2(t-2)+\delta'' u_1(t-2)+\varepsilon_2(t) \end{cases}$$

med motsvarande startvärden för parametrarna. Detta försök krön-tes dock inte med framgång.

Inte heller med en parameter i varje regulator var systemet stabilt. Som illustration visas i fig. 4.6 exempel på parametrar-nas trajektorier (p_0=0.005). Det kan vara intressant att jäm-föra med två figurer från [7] . I fig. 4.7 visas stabilitets-området för k_1, k_2 om systemet 4.4 regleras med P-regulatorer-na.

$$\begin{cases} u_1(t)=k_1 y_1(t) \\ u_2(t)=k_2 y_2(t) \end{cases}$$

där k_1 och k_2 äe konstanta (sätt p_0=0 i STURE1 o. 2).

Detta område är även inritat i fig. 4.6.

Parameterskattningarna divergerar alltså mot eller förbi sta-bilitetsgränsen - förlusterna blir stora och skattningarna "kastas" åter in i stabilitetsområdet. osv......

Fig. 4.8 visar trajektorierna för parametrarna som ovan, men beräknade med "Ljungs differentialekvation", [8].

PLOT TH21(TH11)

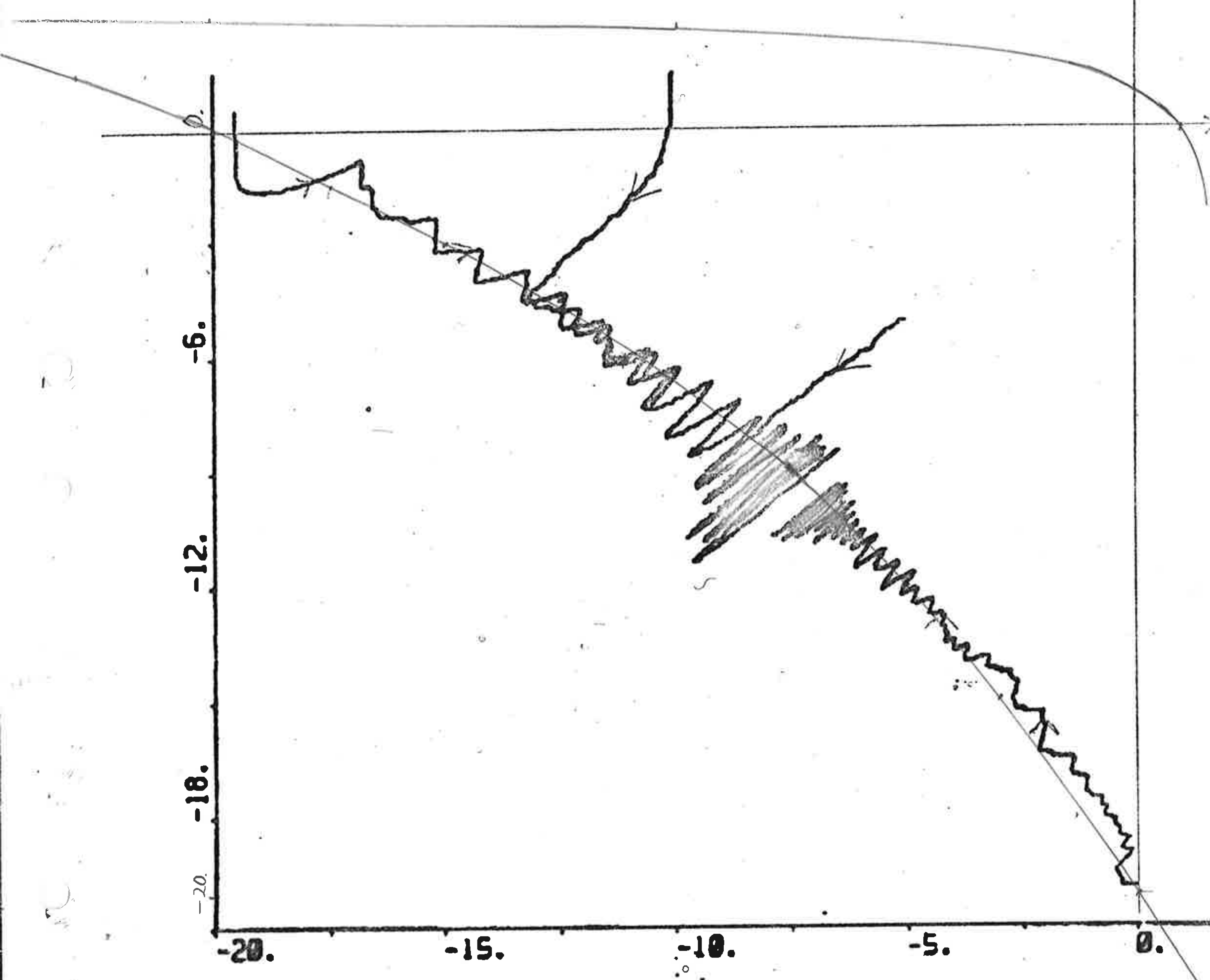


fig. 4.6 Parametertrajektorier

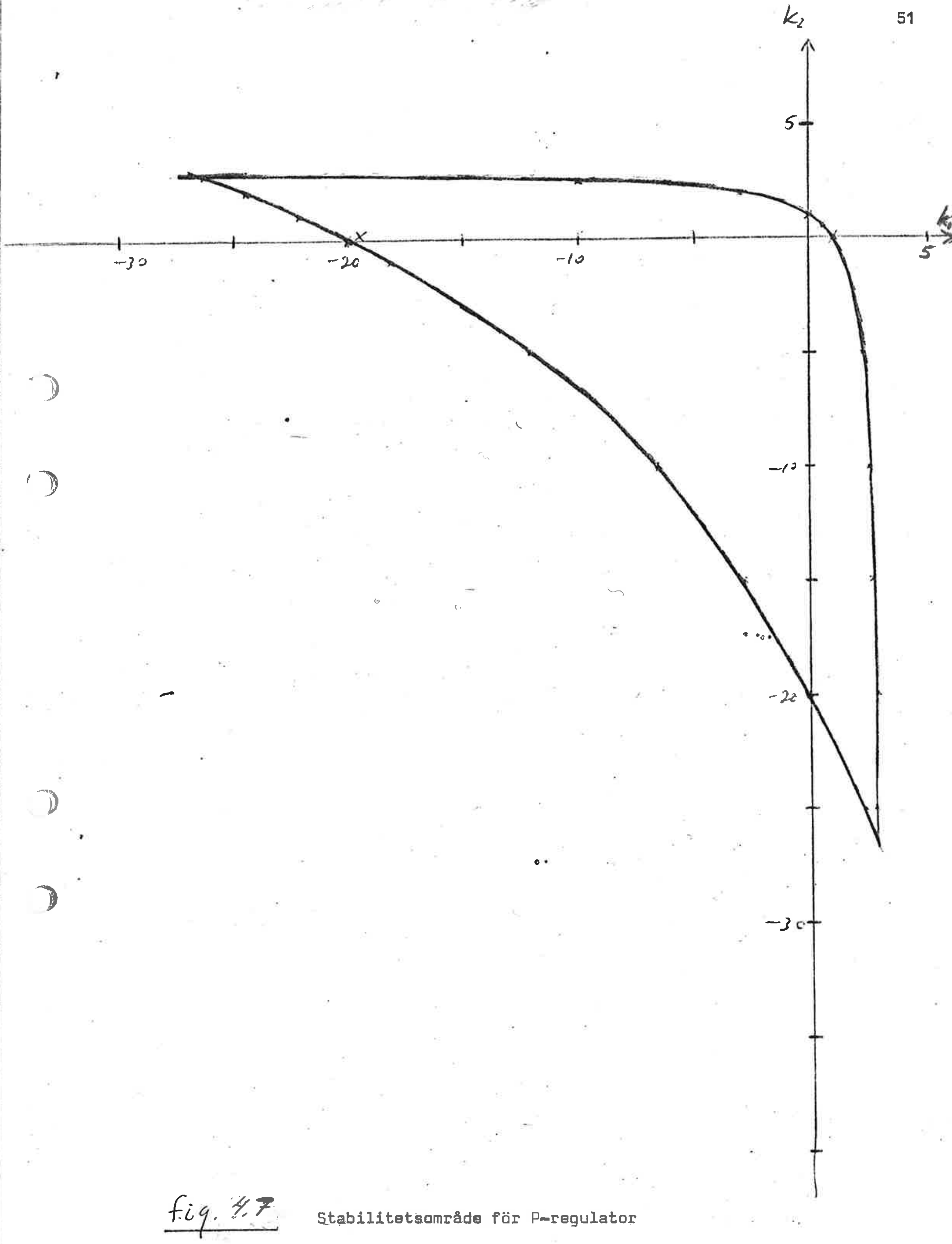


fig. 4.7 Stabilitetsområde för P-regulator

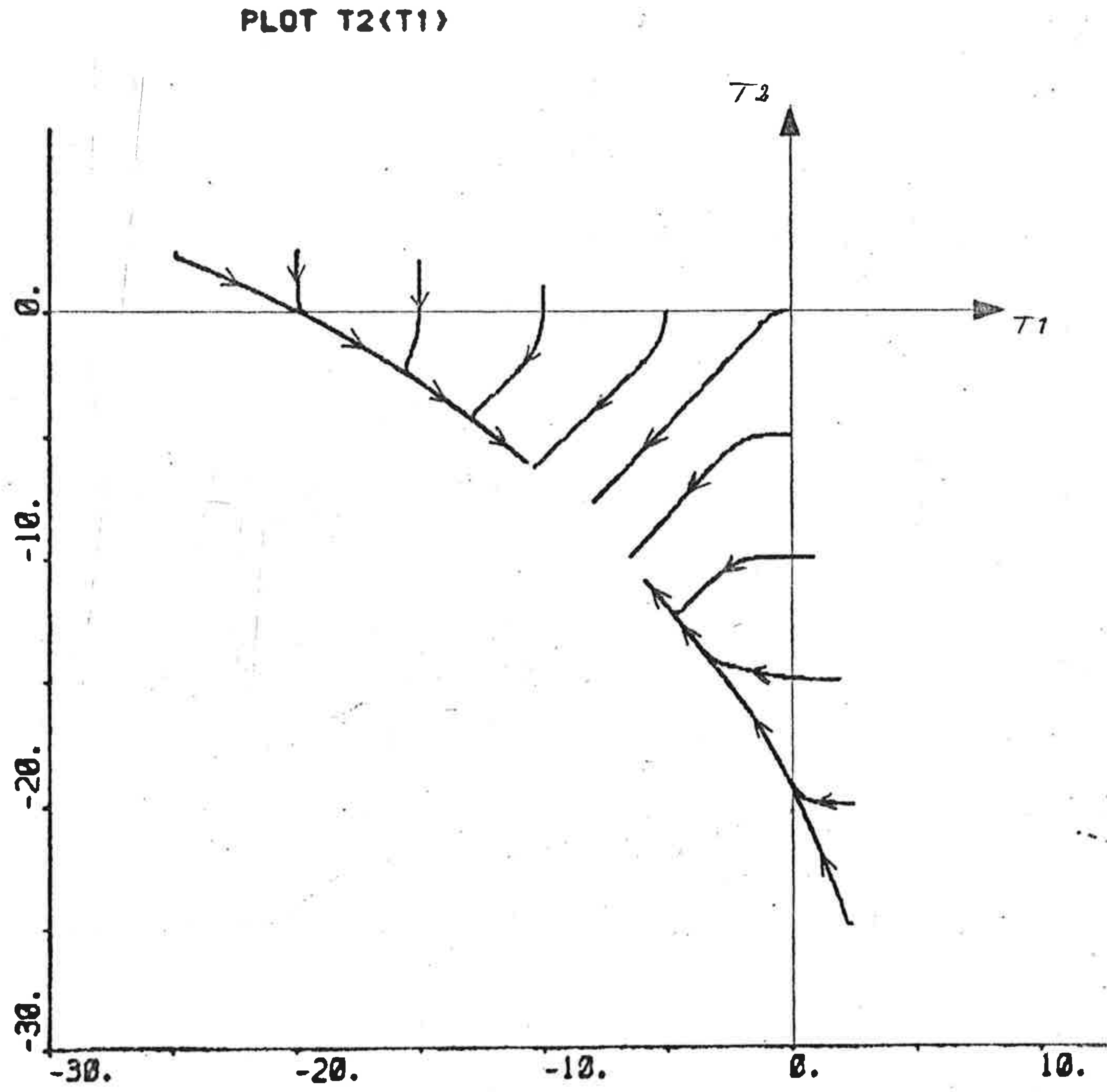


fig 4.8 Beräknade parametertrajektorier

SLUTLEDNINGAR

Exemplen i kapitel 4 visar tydligt att decentraliserad reglering bör användas endast om det verkligen finns skäl att inte styra centralt. I allmänhet krävs en central regulator för optimal styrning medan en decentraliserad regulatorstruktur automatiskt innebär större förluster.
De gånger decentraliserad reglering är önskvärd kan de ovan beskrivna självinställande regulatorerna vara användbara. Vi har sett många exempel på att sådana konvergerar mot de regulatorer som skulle använts med ingående kännedom om systemet som regleras. Självinställarnas styrka är ju att sådan kännedom inte är nödvändig. Valet av regulatorstruktur är däremot väsentligt. I några exempel blir visserligen skillnaderna i förluster för olika strukturer relativt små men i andra åter ger en regulator med för få parametrar upphov till instabilitet. Att använda onödigt många parametrar verkar däremot inte farligt.

Referenser

[1] Sandell N et al.: A Survey of Decentralized Control Methods for Large Scale Systems
Department of Electrical Engineering and Computor Sciences, Massachusetts Institute of Technologi Cambridge, Massachusetts 02139

[2] Åström KJ- Wittenmark B: On self Tuning Regulators
Automatica, Vol. 9 pp 185-199,1973

[3] Wittenmark B: A Self-tuning Regulator. Report 7311, LTH 1973

[4] Åström KJ -Borisson U,-Ljung C,-Wittenmark B:
Theory and Applications of Self-tuning Regulators
Automatica, vol 13, pp 451-476, 1977

[5] Borisson U: Self- Tuning Regulators - Industrial Applications and Multivariable Theory.
Report 7513, LTH 1975

[6] Elmqvist H: SIMNON - An Interactive Simulation Program for Nonlinear Systems
Report 7502, LTH 1975

[7] Gustavsson, I: Personlig kommunikation.

[8] Ljung, L - Wittenmark, B: Asymptotic properties of Self-tuning Regulators
Report 7404, LTH 1974

APPENDIX

Programlistor

innehåll:

SIMBAT - Länklista för generering av SIMNON.
De element som tillförts SIMBAT är understrukna. De
finns även återgivna i detta appendix.

REGLOM,SYSCOM, DISTC, SYSTSC - Commonblock

HAMPC - "Dummy"-rutin

SYSTS - Förbinder SIMNON med systemprogrammen

REGS, SYSS, DIST - Systemprogram

MESS, STURE, LS, RTS, SHIF - Subrutiner till systemprogrammen

DUM1 - "Dummy"-rutin

CONN- "Connecting system"

```
ÄJOB
A RK <SIM> -5/RK <NEW> -1/LP -12
CHAIN
SIMNON
SZ
SIMNON,BLDATA,¤DISHDL,¤SDSHDL,¤BINIO,¤BCDIO,¤FILE,SCAPRO,RMOVE,
¤MOD,¤PARTWD,¤RMNMX,¤FETCH,¤IBREAK,REGCOM,SYSCOM,DISTC,SYSTSC
LSIM=¤SIMU,¤OUTP
LSIM2=¤PSREAD,DERIV¤,CALCUL¤,INIVAR¤,EXTCAL¤
LI=¤ESIMN,¤SUBST
LCOMP=¤SIMNSY/.COMP,.EDIT
L1=¤PARINT,¤CODISP,¤COPLOT,¤COLIST/¤DEV,¤LOGARG
L2=¤COSIMU,¤WMESS,¤COALG/¤LOGARG
L3=¤COSAVE,¤COGET/¤LOGARG
L4=¤COEDIT/.EDIT,¤LOGARG
L5=¤COSTOR,¤COAXES,¤COSYST/¤LOKFOR,¤VARIB,
¤OFNEF,¤PFLOAT,¤LOGARG
L6=¤COTURN,¤TEST,¤INTERR,¤ISIMN,¤EXTSYS/¤LOGARG
L7=¤COSHOW,¤COERR/¤LOGARG
LI1=¤WRITE,¤LPCOM
LI2=¤IFGOLA
LI3=¤FORNXT
LI4=¤FREE,¤EXEC,¤SWTCH
LI5=¤LET
LI6=¤MFRE
LII=¤INTR
LDUM=¤DUM
LINT1=HAMPC
LINT2=¤RKADM,¤RKFIX
LUSER=SYSTS/¤EDECL,¤EDECLV,¤EDECLM,REGS,SYSS,DIST,MESS
LDUM1=STURE,LS,RTS,SHIF,DUM1
LSIM:LCOMP:LI,L1
L1:L2:L3:L4:L5:L6:L7:LII:LI1:LI2:LI3:LI4:LI5:LI6:
¤MAC:LDUM,LINT1,LDUM1,LSIM2,LUSER
LINT1:LINT2

ÄJOB
ABS
SIMNON
ÄEXIT
```

```
      BLOCK DATA
      COMMON/REGCOM/IACT1,IACT2,ISB,U1,U2,Y1,Y2,UR1,UR2,RES1,RES2
Ä            ,RN1(4),RN2(4),RK1(4),RK2(4),RKDEL,B01,B02,RB0
Ä            ,ULIM,WTI,WTM,DELT,REG,V1,V2,V,TS,DT
      END
```

```
      BLOCK DATA
      COMMON/SYSCOM/IACTV,US1,US2,ES1,ES2,YS1,YS2
Ä           ,RNA,RNB,RNC,RKB,A11(5),A12(5),A21(5),A22(5)
Ä           ,B11(5),B12(5),B21(5),B22(5),C11(5),C12(5)
Ä           ,C21(5),C22(5),TS,DT
      END
```

```
BLOCK DATA
COMMON/DISTC/ED1,ED2,RO11,RO12,RO22,RNOD1,RNOD2,TS,DT
END
```

```
      BLOCK DATA
      COMMON/SYSTSC/TH1(5),TH2(5),R1(5),R2(5),P1(5,5),P2(5,5)
Ä             ,S1(4,10),S2(4,10),S0(4,10),W1(5),W2(5),W3(5),NW(4)
Ä             ,P01(5),P02(5),TH01(5),TH02(5)
      END
```

```
SUBROUTINE HAMPC
RETURN
END
```

```
      SUBROUTINE SYSTS
C
      COMMON/SYSTSC/ TH1(5),TH2(5),R1(5),R2(5),P1(5,5),P2(5,5)
     Ä         ,S1(4,10),S2(4,10),S0(4,10),W1(5),W2(5),W3(5),NW(4)
     Ä         ,P01(5),P02(5),TH01(5),TH02(5)
      COMMON /DESTIN/ ISYST,IDUM
      COMMON /NSYSTS/ NSYST
      COMMON /NALLOC/ NS
C
      NSYST=3
      NS=1
C
      IS=4
      IP=5
C
      GO TO(1,2,3),ISYST
C
    1 CALL SYSS
      RETURN
    2 CALL DIST
      RETURN
    3 CALL REGS(TH1,TH2,R1,R2,P1,P2,IP,S1,S2,S0,IS,
     Ä             W1,W2,W3,NW,P01,P02,TH01,TH02)
      RETURN
      END
```

```
      SUBROUTINE REGS(TH1,TH2,R1,R2,P1,P2,IP,S1,S2,S0,
     Ä                 IS,W1,W2,W3,NW,P01,P02,TH01,TH02)
C
C
C     SIMULATES TWO SELFADJUSTING REGULATORS BASED ON
C     LEAST SQUARES ESTIMATION AND MINIMUM VARIANCE CONTROL
C
C
C
C     MODEL (IF IB0=0,I.E. B0 IS A CONSTANT)
C
C     Y1(T)+A1*Y1(T-K1(1)-KDEL-1)+....+AN1(1)*Y1(T-K1(1)-KDEL-N1(1))
C          +C1*Y2(T-K1(3)-KDEL-1)+....+CN1(3)*Y2(T-K1(3)-KDEL-N1(3))
C     =B0*[U1(T-K1(2)-1)+B1*U1(T-K1(2)-2)+....
C                                 +BN1(2)*U1(T-K1(2)-N1(2)-1)]
C          +D1*U2(T-K1(4)-KDEL-1)+....+DN1(4)*U2(T-K1(4)-KDEL-N1(4))
C
C     TH1=[A1,...,AN1(1),B1,...,BN1(2),C1,...,CN1(3),D1,...,DN1(4)]
C
C
C     ETC.
C
C
C
      LOGICAL LSTOP,LDARK
C
      DIMENSION TH1(1),TH2(1),R1(1),R2(1),P1(IP,1),P2(IP,1)
      DIMENSION S1(4,1),S2(4,1),S0(4,1),ACTY(4)
      DIMENSION NW(1),W1(1),W2(1),W3(1)
      DIMENSION P01(1),P02(1),TH01(1),TH02(1)
      DIMENSION N1(4),N2(4),K1(4),K2(4),KACT1(4),KACT2(4)
      DIMENSION RN1(4),RN2(4),RK1(4),RK2(4)
C
      COMMON/DESTIN/IDUM,IPART
      COMMON/TIME/T
      COMMON/USER/LSTOP,LDARK
C
      COMMON/REGCOM/IACT1,IACT2,ISB,U1,U2,Y1,Y2,UR1,UR2,RES1,RES2
     Ä        ,RN1,RN2,RK1,RK2,RKDEL,B01,B02,RB0
     Ä        ,ULIM,WTI,WTM,DELT,REG,V1,V2,V,TS,DT
C
      GOTO (1,2,3,4,5,6,7,8),IPART
C
C     IDENTIFICATION
C
    1 CALL IDENT(5HDISCR,4HREGS)
      RETURN
C
C     DECLARATIONS
C
C     CHECK OUTER MATRIX DIMENSION PARAMETERS
C
    2 CALL FINT(5HIACT1,4H    ,IACT1,IND1)
      IF(IND1.EQ.0.AND.IACT1.GE.1.AND.IACT1.LE.10)GOTO 200
      CALL BADVAL(5HIACT1,4H    )
      GOTO 900
C
  200 CALL FINT(5HIACT2,4H    ,IACT2,IND1)
      IF(IND1.EQ.0.AND.IACT2.GE.1.AND.IACT2.LE.10)GOTO 210
      CALL BADVAL(5HIACT2,4H    )
      GOTO 900
C
  210 CALL FINT(4HISB ,4H    ,ISB,IND1)
```

```
      IF(IND1.EQ.0.AND.ISB.GE.2.AND.ISB.LE.10)GOTO 220
      CALL BADVAL(4HISB ,4H    )
      GOTO 900
C
C
  220 CALL INPUT(U1,4HU1  )
      CALL INPUT(U2,4HU2  )
      CALL INPUT(Y1,4HY1  )
      CALL INPUT(Y2,4HY2  )
C
      CALL OUTPUT(UR1,4HUR1 )
      CALL OUTPUT(UR2,4HUR2 )
      CALL OUTPUV(TH1,IACT1,4HTH1 )
      CALL OUTPUV(TH2,IACT2,4HTH2 )
      CALL OUTPUT(RES1,4HRES1)
      CALL OUTPUT(RES2,4HRES2)
C
      CALL PARV(RN1,4,4HN1  )
      CALL PARV(RN2,4,4HN2  )
      CALL PARV(RK1,4,4HK1  )
      CALL PARV(RK2,4,4HK2  )
C
      CALL PAR(RKDEL,4HKDEL)
C
      CALL PAR(B01,4HB01 )
      CALL PAR(B02,4HB02 )
      CALL PAR(RB0,4HIB0 )
C
      CALL PARV(TH01,IACT1,4HTH01)
      CALL PARV(TH02,IACT2,4HTH02)
      CALL PARV(P01,IACT1,4HP01 )
      CALL PARV(P02,IACT2,4HP02 )
C
      CALL PARM(S0,4,ISB,IS,4HS0  )
C
      CALL PARV(R1,IACT1,4HR1  )
      CALL PARV(R2,IACT2,4HR2  )
C
      CALL PAR(ULIM,4HULIM)
C
      CALL PAR(WTI,4HWTI )
      CALL PAR(WTM,4HWTM )
C
      CALL PAR(DELT,4HDELT)
      CALL PAR(REG,4HREG )
C
      CALL VAR(V1,4HV1  )
      CALL VAR(V2,4HV2  )
      CALL VAR(V,4HV   )
C
      CALL VARM(P1,IACT1,IACT1,IP,4HP1  )
      CALL VARM(P2,IACT2,IACT2,IP,4HP2  )
      CALL VARM(S1,4,ISB,IS,4HS1  )
      CALL VARM(S2,4,ISB,IS,4HS2  )
C
C
      CALL PAR(DT,4HDT  )
C
      CALL TSAMP(TS,4HTS  )
C
      RETURN
C
C     CONSTANT ASSIGNMENTS
C
```

```
    3     WTI=1.
          WTM=1.
C
          DT=1.
          REG=1.
          RB0=0.
          B01=1.
          B02=1.
C
          ULIM=-1.
          EPS=0.1
C
          DO 300 I=1,IACT1
  300     P01(I)=100.
          DO 310 I=1,IACT2
  310     P02(I)=100.
C
          DO 320 I=1,4
          RN1(I)=0.
          RN2(I)=0.
          RK1(I)=0.
  320     RK2(I)=0.
C
          RKDEL=0.
C
          ACTY(2)=0.
          ACTY(4)=0.
C
          RETURN
C
C         INITIAL
C
    4     TS=T+DT
C
C
C         CONVERSION TO INTEGERS
C
          DO 400 I=1,4
          N1(I)=RN1(I)+EPS
          N2(I)=RN2(I)+EPS
          K1(I)=RK1(I)+EPS
  400     K2(I)=RK2(I)+EPS
C
          KDEL=RKDEL+EPS
C
          MSTOP=.FALSE.
C
          IB0=RB0+EPS
C
C         COUNT THE NUMBER OF PARAMETERS IN THE REGULATORS
          NPAR1=0
          NPAR2=0
          DO 410 I=1,4
          NPAR1=NPAR1+N1(I)
  410     NPAR2=NPAR2+N2(I)
          IF(NPAR1+NPAR2.GT.0.AND.NPAR1.LE.IACT1.AND.NPAR2.LE.IACT2)GOTO 42
          MSTOP=.TRUE.
          LSTOP=.TRUE.
          KMESS=4
          RETURN
C
C         IS THE MODEL REALISTIC?
C
  420     IF(N1(4).GT.0.AND.K1(4)+KDEL.LE.K1(2))GOTO 421
```

```
      IF(N2(4).GT.0.AND.K2(4)+KDEL.LE.K2(2))GOTO 421
      GOTO 425
  421 MSTOP=.TRUE.
      LSTOP=.TRUE.
      KMESS=6
      RETURN
C
  425 CALL RMOVE(TH01(1),1,TH1(1),1,NPAR1)
      CALL RMOVE(TH02(1),1,TH2(1),1,NPAR2)
C
      X=0.
      DO 430 I=1,NPAR1
      CALL RMOVE(X,0,P1(I,1),IP,NPAR1)
  430 P1(I,I)=P01(I)
C
      DO 440 I=1,NPAR2
      CALL RMOVE(X,0,P2(I,1),IP,NPAR2)
  440 P2(I,I)=P02(I)
C
      V1=0.
      V2=0.
      WT=WTI
C
C     GIVE THE INITIAL VALUES IN S0 TO S1 AND S2
      DO 450 I=1,4
      K=I+2-I/3*4
      DO 450 J=1,ISB
      X=S0(I,J)
      S1(I,J)=X
  450 S2(K,J)=X
C
      B1=1.
      B2=1.
      IF(IB0.EQ.0.AND.REG.GT.0.5)B1=1./B01
      IF(IB0.EQ.0.AND.REG.GT.0.5)B2=1./B02
C
      RETURN
C
C     OUTPUTS
C
C
C     PREPARATIONS FOR LS-ESTIMATION
C
    5 ACTY1=Y1
      ACTY2=Y2
C
      IF(REG.GT.0.5)GOTO 510
C
      DO 520 I=1,4
      KACT1(I)=K1(I)
  520 KACT2(I)=K2(I)
C
      GOTO 530
C
  510 IF(IB0.NE.0)GOTO 540
      KB=K1(2)
      ACTY1=Y1/B01-S1(2,KB+1)
      KB=K2(2)
      ACTY2=Y2/B02-S2(2,KB+1)
C
  540 KACT1(1)=K1(1)+KDEL
      KACT2(1)=K2(1)+KDEL
      KACT1(2)=K1(2)+1
      KACT2(2)=K2(2)+1
```

```
C
      IF(IB0.NE.0)KACT1(2)=K1(2)
      IF(IB0.NE.0)KACT2(2)=K2(2)
C
      KACT1(3)=K1(3)+KDEL
      KACT2(3)=K2(3)+KDEL
      KACT1(4)=K1(4)+KDEL
      KACT2(4)=K2(4)+KDEL
C
C     UPDATE THE WEIGHT-FACTOR (WT<1 MEANS EXPONENTIAL FORGETTING)
  530 WT=WTM*WT+(1.-WTM)
C
C     LEAST SQUARES ESTIMATION
C
      CALL LS(WT,R1,DELT,S1,IS,N1,KACT1,ACTY1,TH1,P1,W1,RES1,IP,W2,
     Ä        W3,NPAR1,4)
C
      CALL LS(WT,R2,DELT,S2,IS,N2,KACT2,ACTY2,TH2,P2,W1,RES2,IP,W2,
     Ä        W3,NPAR2,4)
C
      IF(REG.LT.0.5)RETURN
C
C
C     COMPUTE THE CONTROL SIGNALS
C
C     PREPARATIONS
      DO 550 I=1,4
      KACT1(I)=K1(I)
  550 KACT2(I)=K2(I)
C
      ACTY(1)=-Y1*B1
      ACTY(3)=-Y2*B1
C
C     COMPUTE
C
      CALL STURE(S1,IS,N1,KACT1,KDEL,4,TH1,IB0,ACTY,UR1,NW)
C
      ACTY(1)=-Y2*B2
      ACTY(3)=-Y1*B2
C
      CALL STURE(S2,IS,N2,KACT2,KDEL,4,TH2,IB0,ACTY,UR2,NW)
C
C
C     COMPUTE THE LOSS
C
      V1=V1+Y1*Y1
      V2=V2+Y2*Y2
      V=V1+V2
C
C     LIMITATION OF CONTROL SIGNALS (IF ULIM>0)
C
      IF(ULIM.LE.0.)RETURN
C
      UR1=SIGN(AMIN1(ABS(UR1),ULIM),UR1)
      UR2=SIGN(AMIN1(ABS(UR2),ULIM),UR2)
C
      RETURN
C
C     DYNAMICS
C
C     UPDATE THE "INFORMATION MATRICES"
C
    6 CALL SHIF(S1,IS,4,ISB)
      CALL SHIF(S2,IS,4,ISB)
```

```
C
      S1(1,1)=-Y1*B1
      S1(2,1)=U1
      S1(3,1)=-Y2*B1
      S1(4,1)=U2*B1
C
      S2(1,1)=-Y2*B2
      S2(2,1)=U2
      S2(3,1)=-Y1*B2
      S2(4,1)=U1*B2
C
C
      TS=T+DT
C
      RETURN
C
    7 RETURN
C
C
C     WRITE ERROR MESSAGE (IF ANY)
C
    8 IF(MSTOP)CALL MESS(KMESS)
      RETURN
C
  900 LSTOP=.TRUE.
      RETURN
C
      END
```

```
      SUBROUTINE SYSS
C
C     COMPUTES THE OUTPUT FROM THE SYSTEM
C     I*Y(T)+A(1)*Y(T-1)+......+A(NA)*Y(T-NA)=
C     =B(1)*U(T-KB-1)+....+B(NB)*U(T-KB-NB)+
C     +I*E(T)+C(1)*E(T-1)+.....+C(NC)*E(T-NC)
C
C     WHERE THE MATRICES A(K),B(K),C(K)
C
C             ( A11(K)   A12(K) )
C     A(K)=(                    )         ETC.
C             ( A21(K)   A22(K) )
C
C     AND THE VECTORS Y(T-K),U(T-K),E(T-K)
C
C              ( Y1(K) )
C     Y(T-K)=(         ) (T)              ETC.
C              ( Y2(K) )
C
C     ARE REPRESENTED BY VECTORS A11,A12,...B11,...C22,Y1,Y2,...E2
C
C     INPUTS   U:(US1,US2)
C              E:(ES1,ES2)  (NOISE)
C     OUTPUT   Y:(YS1,YS2)
C
C     SUBROUTINES REQUIRED
C     RMOVE,SCAPRO,MESS
C
C
      LOGICAL LSTOP,LDARK,MSTOP
      DIMENSION Y1(5),Y2(5),E1(5),E2(5),U1(10),U2(10)
      DIMENSION A11(5),A12(5),A21(5),A22(5),B11(5),B12(5),B21(5),B22(5),
     Ä          C11(5),C12(5),C21(5),C22(5)
C
      COMMON/DESTIN/IDUM,IPART
      COMMON/TIME/T
      COMMON/USER/LSTOP,LDARK
C
      COMMON/SYSCOM/IACTV,US1,US2,ES1,ES2,YS1,YS2
     Ä      ,RNA,RNB,RNC,RKB,A11,A12,A21,A22
     Ä      ,B11,B12,B21,B22,C11,C12,C21,C22,TS,DT
C
      GOTO(1,2,3,4,5,6,7,8),IPART
C
C     IDENTIFICATION
C
    1 CALL IDENT(5HDISCR,4HSYSS)
      RETURN
C
C     DECLARATIONS
C
    2 CALL INPUT(US1,4HUS1 )
      CALL INPUT(US2,4HUS2 )
      CALL INPUT(ES1,4HES1 )
      CALL INPUT(ES2,4HES2 )
      CALL OUTPUT(YS1,4HYS1 )
      CALL OUTPUT(YS2,4HYS2 )
C
      CALL PAR(RNA,4HNA  )
      CALL PAR(RNB,4HNB  )
      CALL PAR(RNC,4HNC  )
      CALL PAR(RKB,4HKB  )
C
```

```
      MSTOP=.FALSE.
C
C     CHECK OUTER MATRIX DIMENSION PARAMETER
C
      CALL FINT(5HIACTV,4H    ,IACTV,IND1)
      IF(IND1.EQ.0.AND.IACTV.GE.1.AND.IACTV.LE.5)GOTO 200
      CALL BADVAL(5HIACTV,4H    )
      LSTOP=.TRUE.
      MSTOP=.TRUE.
      KMESS=1
      RETURN
C
  200 CALL PARV(A11,IACTV,4HA11 )
      CALL PARV(A12,IACTV,4HA12 )
      CALL PARV(A21,IACTV,4HA21 )
      CALL PARV(A22,IACTV,4HA22 )
      CALL PARV(B11,IACTV,4HB11 )
      CALL PARV(B12,IACTV,4HB12 )
      CALL PARV(B21,IACTV,4HB21 )
      CALL PARV(B22,IACTV,4HB22 )
      CALL PARV(C11,IACTV,4HC11 )
      CALL PARV(C12,IACTV,4HC12 )
      CALL PARV(C21,IACTV,4HC21 )
      CALL PARV(C22,IACTV,4HC22 )
C
      CALL PAR(DT,4HDT  )
C
      CALL TSAMP(TS,4HTS  )
C
      RETURN
C
C     CONSTANT ASSIGNMENTS
C
    3 DT=1.
      EPS=0.1
C
      RETURN
C
C     INITIAL
C
    4 TS=T+DT
C
C     CONVERSION TO INTEGERS
C
      NA=RNA+EPS
      NB=RNB+EPS
      NC=RNC+EPS
      KB=RKB+EPS
C
      MSTOP=.FALSE.
C
C     IS THE SYSTEM TOO LARGE?
C
      IF(NA.GT.5.OR.NB.GT.5)GOTO 400
      IF(NC.LE.5.AND.(NB+KB).LE.10)GOTO 410
  400 LSTOP=.TRUE.
      MSTOP=.TRUE.
      KMESS=2
      RETURN
C
  410 IF(NA+NB+NC.GT.0)GOTO 415
      MSTOP=.TRUE.
      KMESS=5
C
```

```
  415   DO 420 I=1,5
        Y1(I)=0.
        Y2(I)=0.
        E1(I)=0.
  420   E2(I)=0.
        DO 430 I=1,10
        U1(I)=0.
  430   U2(I)=0.
C
        RETURN
C
C       OUTPUTS
C
C
C       COMPUTE THE OUTPUTS
C
    5   YS1=ES1
        YS2=ES2
        IF(NA.GT.0)YS1=YS1-SCAPRO(A11(1),1,Y1(1),1,NA)
     Ä                    -SCAPRO(A12(1),1,Y2(1),1,NA)
        IF(NA.GT.0)YS2=YS2-SCAPRO(A21(1),1,Y1(1),1,NA)
     Ä                    -SCAPRO(A22(1),1,Y2(1),1,NA)
C
        IF(NB.GT.0)YS1=YS1+SCAPRO(B11(1),1,U1(KB+1),1,NB)
     Ä                    +SCAPRO(B12(1),1,U2(KB+1),1,NB)
        IF(NB.GT.0)YS2=YS2+SCAPRO(B21(1),1,U1(KB+1),1,NB)
     Ä                    +SCAPRO(B22(1),1,U2(KB+1),1,NB)
        IF(NC.GT.0)YS1=YS1+SCAPRO(C11(1),1,E1(1),1,NC)
     Ä                    +SCAPRO(C12(1),1,E2(1),1,NC)
        IF(NC.GT.0)YS2=YS2+SCAPRO(C21(1),1,E1(1),1,NC)
     Ä                    +SCAPRO(C22(1),1,E2(1),1,NC)
C
        RETURN
C
C       DYNAMICS
C
C
C       UPDATE THE "INFORMATION VECTORS"
C
    6   CALL RMOVE(Y1(4),-1,Y1(5),-1,4)
        CALL RMOVE(Y2(4),-1,Y2(5),-1,4)
        Y1(1)=YS1
        Y2(1)=YS2
        CALL RMOVE(U1(9),-1,U1(10),-1,9)
        CALL RMOVE(U2(9),-1,U2(10),-1,9)
        U1(1)=US1
        U2(1)=US2
        CALL RMOVE(E1(4),-1,E1(5),-1,4)
        CALL RMOVE(E2(4),-1,E2(5),-1,4)
        E1(1)=ES1
        E2(1)=ES2
C
        TS=T+DT
C
        RETURN
C
    7   RETURN
C
    8   IF(MSTOP)CALL MESS(KMESS)
        RETURN
C
        END
```

```
      SUBROUTINE DIST
C
C     GENERATES OUTPUTS ED1,ED2,
C     WHICH ARE BOTH WHITE NOISE,WITH
C     VAR(ED1)=RO11
C     VAR(ED2)=RO22
C     COV(ED1,ED2)=RO12
C
C     SUBROUTINES REQUIRED:MCNODI,MESS
C
      LOGICAL LSTOP,LDARK,MSTOP
C
      COMMON/DESTIN/IDUM,IPART
      COMMON/TIME/T
      COMMON/USER/LSTOP,LDARK
C
      COMMON/DISTC/ED1,ED2,RO11,RO12,RO22,RNOD1,RNOD2,TS,DT
C
      GOTO(1,2,3,4,5,6,7,8),IPART
C
C     IDENTIFICATION
C
    1 CALL IDENT(5HDISCR,4HDIST)
      RETURN
C
C     DECLARATIONS
C
    2 CALL OUTPUT(ED1,4HED1 )
      CALL OUTPUT(ED2,4HED2 )
C
      CALL PAR(RO11,4HRO11)
      CALL PAR(RO12,4HRO12)
      CALL PAR(RO22,4HRO22)
C
      CALL PAR(RNOD1,4HNOD1)
      CALL PAR(RNOD2,4HNOD2)
C
      CALL PAR(DT,4HDT  )
C
      CALL TSAMP(TS,4HTS  )
C
      RETURN
C
C     CONSTANT ASSIGNMENTS
C
    3 DT=1.
      RNOD1=19.
      RNOD2=27.
C
      RO11=1.
      RO22=1.
C
      EPS=0.1
C
      RETURN
C
C     INITIAL
C
    4 TS=T+DT
C
      MSTOP=.FALSE.
C
      IF(RO11*RO22.GE.RO12*RO12)GOTO 400
```

```
      LSTOP=.TRUE.
      MSTOP=.TRUE.
      KMESS=3
      RETURN
C
  400 NOD1=RNOD1+EPS
      NOD2=RNOD2+EPS
C
C     AVOID DIVIDING WITH 0
      IF(RO11.LT.1E-30)GOTO 410
C
      A=SQRT(RO11)
      C=RO12/RO11
      D=SQRT(RO22-RO12*RO12/RO11)
C
      RETURN
C
  410 A=0
      C=0
      D=SQRT(RO22)
      RETURN
C
C     OUTPUT
C
    5 CALL MCNODI(NOD1,EH)
      ED1=A*EH
      CALL MCNODI(NOD2,EH)
      ED2=C*ED1+D*EH
C
      RETURN
C
C     DYNAMICS
C
    6 TS=T+DT
C
      RETURN
C
C
    7 RETURN
C
    8 IF(MSTOP)CALL MESS(KMESS)
      RETURN
C
      END
```

```
      SUBROUTINE MESS(KMESS)
C
      NU=10
C
      GOTO(1,2,3,4,5,6),KMESS
C
    1 WRITE(NU,11)
   11 FORMAT(1X,9HBAD IACTV)
      RETURN
    2 WRITE(NU,22)
   22 FORMAT(1X,18HBAD NA,NB,NC OR KB)
      RETURN
    3 WRITE(NU,33)
   33 FORMAT(1X,24HSORRY R11*R22.LT.R12*R12)
      RETURN
    4 WRITE(NU,44)
   44 FORMAT(1X,34HNPAR1,NPAR2,IACT1 OR IACT2 NO GOOD)
      RETURN
    5 WRITE(NU,55)
   55 FORMAT(1X,21HWARNING:NA+NB+NC.LE.0)
      RETURN
    6 WRITE(NU,66)
   66 FORMAT(1X,35HK14+KDEL.LE.K12.OR.K24+KDEL.LE.K22.)
      RETURN
C
      END
```

```
      SUBROUTINE STURE(S,IS,N,K,KDEL,ISA,TH,IBZ,ACTS,UR,NZ)
C
C     TEXT
C
      DIMENSION S(IS,1),N(1),K(1),TH(1),ACTS(1),NZ(1)
C
      NZ(1)=0
      DO 10 I=2,ISA
   10 NZ(I)=NZ(I-1)+N(I-1)
C
      UR=0.
C
      DO 20 I=1,ISA
      IF(I.EQ.2.OR.N(I).EQ.0) GO TO 20
      NB=NZ(I)+1
      IR=K(I)-K(2)+KDEL
      IF(IR.EQ.0) UR=UR-TH(NB)*ACTS(I)
      IF(IR.NE.0) UR=UR-TH(NB)*S(I,IR)
      IF(N(I).NE.1) UR=UR-SCAPRO(TH(NB+1),1,S(I,IR+1),IS,N(I)-1)
   20 CONTINUE
C
C
      IF(N(2).EQ.0) GO TO 40
      J=1
      NB=N(2)
      IF(IBZ.EQ.0) GO TO 30
      IF(N(2).EQ.1) GO TO 40
      NB=N(2)-1
      J=2
   30 UR=UR-SCAPRO(TH(N(1)+J),1,S(2,1),IS,NB)
C
   40 IF(IBZ.NE.0) UR=UR/TH(N(1)+1)
C
      RETURN
      END
```

```
      SUBROUTINE LS(WT,R1,DELTA,S,IS,N,KACT,ACT,T,P,FI,RES,
     * IP,W1,W2,NPAR,ISA)
C
C     TEXT
C
      DIMENSION T(1),FI(1),R1(1),P(IP,1),S(IS,1)
      DIMENSION W1(1),W2(1),N(1),KACT(1)
C
C     UPDATING OF FI-VECTOR
C
      J1=0
      DO 20 I=1,ISA
      N1=N(I)
      K1=KACT(I)
      IF(N1.LE.0) GO TO 20
      DO 10 J=1,N1
      J1=J1+1
   10 FI(J1)=S(I,K1+J)
   20 CONTINUE
C
C     COMPUTATION OF THE RESIDUAL
C
      RES=ACT-SCAPRO(FI(1),1,T(1),1,NPAR)
C
C     LS-ESTIMATION
C
      CALL RTS(FI,RES,WT,R1,DELTA,T,P,NPAR,IP,W1,W2)
C
C
      RETURN
      END
```

```
      SUBROUTINE RTS(FI,RES,WT,R1,DELTA,T,P,NPAR,IP,W1,W2)
C
C     TEXT
C
      DIMENSION T(1),FI(1),R1(1),P(IP,1)
      DIMENSION W1(1),W2(1)
C
C     UPDATING OF THE P-MATRIX
C
      DO 10 I=1,NPAR
      SL=SCAPRO(P(I,1),IP,FI(1),1,NPAR)
      W1(I)=SL
   10 W2(I)=SL
C
      SL=WT+SCAPRO(FI(1),1,W1(1),1,NPAR)
C
      DO 30 I=1,NPAR
      W1(I)=W1(I)/SL
      DO 20 J=1,I
      P(I,J)=(P(I,J)-W1(I)*W2(J))/WT
   20 P(J,I)=P(I,J)
      SK=P(I,I)
   30 P(I,I)=SK+(R1(I)-DELTA*SK*SK)/WT
C
C     UPDATING OF PARAMETER VECTOR
C
      DO 40 I=1,NPAR
   40 T(I)=T(I)+RES*W1(I)
C
      RETURN
      END
```

```
      SUBROUTINE SHIF(S,IS,ISA,ISB)
C
      DIMENSION S(IS,1)
C
      ISB1=ISB-1
C
      DO 10 I=1,ISA
   10 CALL RMOVE(S(I,ISB1),-IS,S(I,ISB),-IS,ISB1)
C
      RETURN
      END
```

```
      SUBROUTINE DUM1
      DIMENSION XX(400)
      END
```

```
CONNECTING SYSTEM CONN
"
TIME T
"
ES1[SYSS]=ED1[DIST]
ES2[SYSS]=ED2[DIST]
Y1[REGS]=YS1[SYSS]
Y2[REGS]=YS2[SYSS]
U1[REGS]=UR1[REGS]
U2[REGS]=UR2[REGS]
US1[SYSS]=UR1[REGS]
US2[SYSS]=UR2[REGS]
"
END
```