

EN SJÄLVINSTÄLLANDE PID-REGULATOR FÖR SYSTEM MED
VARIABEL TIDSFÖRDRÖJNING

MARTIN PALMGREN

INSTITUTIONEN FÖR REGLERTEKNIK
LUNDS TEKNISKA HÖGSKOLA
SEPTEMBER 1982

TILLHÖR REFERENSBIBLIOTEKET
UTLANAS EJ

LUND INSTITUTE OF TECHNOLOGY DEPARTMENT OF AUTOMATIC CONTROL Box 725 S 220 07 Lund 7 Sweden		Document name Report	
		Date of issue September 1982	
		Document number CODEN:LUTFD2/(TFRT-5283)/1-058/(1982)	
Author(s) Martin Palmgren		Supervisor Björn Wittenmark	
		Sponsoring organization	
Title and subtitle A Selftuning PID-Controller For Systems with Timevarying Timedelay (En självinställande Pid-regulator för system med variabel tidsfördröjning)			
Abstract A selftuning PID-controller is investigated. It is assumed that the process order is two. A pole-placement-algorithm is used to get the desired closed loop poles. Two methods is studied for the calculation of the controlparameters. The first method is based on solving a system of equations. The second method makes a two-steps-estimation of the controlparameters. The only a-priori knowledge of the process needed is, except the order, the maximum timedelay. The two methods are compared using simulated systems. It is shown that the discussed controller can adapt to changes in the timedelay.			
Key words			
Classification system and/or index terms (if any)			
Supplementary bibliographical information			
ISSN and key title			ISBN
Language Swedish	Number of pages 058	Recipient's notes	
Security classification			

SJÄLVINSTÄLLANDE PID-REGULATOR
FÖR SYSTEM MED
VARIABEL TIDSFÖRDRÖJNING

Examensarbete i Reglerteknik
Oktober 1982

Författare: Martin Palmgren
Handledare: Björn Wittenmark

INNEHALLSFÖRTECKNING

	SIDAN
1. INLEDNING	1
2. PROCESSIDENTIFIERING	3
3. REGLERING	6
3.1 Metod 1: Lösning av ekvationssystem	7
3.2 Realisering av Metod 1	10
3.3 Metod 2: Tvåstegs skattning av regulatorkoeff. .	11
3.4 Realisering av Metod 2	12
4. SIMULERINGAR	13
4.1 Diskret varierande tidsfördröjning på system av 2:a ordningen	14
4.2 Kontinuerligt varierande tidsfördröjning på system av 2:a ordningen	23
4.3 Fast och kontinuerligt varierande tids- fördröjning på system av 4:e ordningen	30
5. SAMMANFATTNING	34
Referenser	35
Appendix A (programmet till Metod 1)	A.1
Appendix B (programmet till Metod 2)	B.1
Appendix C (simulerade system)	C.1
Appendix D (uträkning av exakta koefficienter)	D.1

1. INLEDNING

Detta arbete avser att undersöka en självinställande PID-regulator (SPID) för styrning av system med variabel tidsfördröjning. Examensarbetet är i stort sett en fortsättning på Lars-Åke Andells rapport, ref[1], och simuleringarna är ämnade att jämföras med Andells.

Förutsättningarna är:

att process dynamiken, förutom tidsfördröjningen, är av andra ordningen och att man kan fixera den maximala tidsfördröjningen hos processen. Tidsfördröjningen får för övrigt vara variabel (både diskret och kontinuerligt).

att regulatorn skall, med måttliga krav på bandbredd, kunna reglera system av högre ordning.

Exempel på sådana processer är de där produktionen beror på något flöde. Ett fall är en pappersmaskin, där tjockleken på pappert mätes på ett ställe och påverkas på ett annat. Kopplingen mellan utsignalen (tjockleken) och styrsignalen (massaflöde) rymmer en variabel tidsfördröjning enär papperet transporteras och tillverkas med olika hastighet.

Skillnaden mellan detta examensarbete och L-Å Andells ligger i det sätt på vilket styrlagarna beräknas samt att generaliseringar av de Fortransystem Andell föreslagit, gjorts. De bör emellertid kunna generaliseras ytterligare: exempelvis för system med högre ordning.

Kapitel Processidentifiering är en kortfattad beskrivning om hur SPID i detta arbete uppfattar processpolynomen.

I kapitel Reglering beskrivs de två metoder som utnyttjats för att beräkna styrlagarna:

Metod 1. Med hjälp av processpolynomen ställa upp och lösa det ekvationssystem som är följden av en viss identitet.

Metod 2. Med hjälp av processpolynomen filtrera ut- och in-signaler för att sedan skatta styrlagarna med minsta-kvadrat-metoden.

Det av Institutionen för Reglerteknik vid Lunds Tekniska Högskola utvecklade simuleringsspråket Simnon och dess Fortransystem, finns beskrivna i ref[2].

Teori för samplade system, härledning av minsta-kvadrat-metoden (m.k.-metoden), och här förekommande

nomenklatur återfinns i ref[3].

Appendix A, B och C innehåller programlistor över regulatorer och simulerade system. Appendix D innehåller härledning och programlistning till beräkning av exakta koefficienter hos vissa system.

En tidigare rapport på området är ref[4].

2. PROCESSIDENTIFIERING

Utgå från den kontinuerliga representationen av överföringsfunktionen

$$G(s) = \exp(-t_{\text{del}} s) \frac{B_c'(s)}{A_c(s)} \quad . \quad B_c / A_c \text{ är av andra ordningen.}$$

Där t_{del} är tidsfördröjningen.

En av SPID:s uppgifter är att identifiera (skatta) den diskreta motsvarigheten till A_c och B_c -polynomens koefficienter.

Den tidskontinuerliga processens tidsfördröjning, t_{del} , kan uttryckas som:

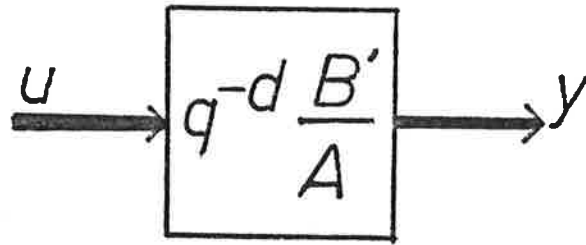
$$\left. \begin{aligned} t_{\text{del}} &= (d-1)h + \tau \\ h &= \text{samplingsintervall} \\ d &= \text{heltal} \\ 0 \leq \tau < h \end{aligned} \right\} \text{ref[3], kap. 3}$$

Den diskreta motsvarigheten till $G(s)$, $H(q^{-1})$ får utseendet

$$H(q^{-1}) = q^{-d} \frac{B'(q^{-1})}{A(q^{-1})} \quad (\text{se fig. 2.1})$$

$$(2.2) \quad \left\{ \begin{aligned} B' &= b'_0 q^{-1} + b'_1 q^{-2} + b'_2 q^{-3} & b'_i &= b'_i(\tau, h) \\ A &= 1 + a_1 q^{-1} + a_2 q^{-2} & a_i &= a_i(h) \end{aligned} \right.$$

För detta examensarbete aktuella processer antas vara av andra ordningen och med variabel tidsfördröjning. Det vill säga att A-polynomet antas vara av ordning två (2) och B-polynomet antas vara så stort att det "rymmer" dels en viss maximal tidsfördröjning, dels process dynamik av andra ordningen.



Figur 2.1. Processen. u = styrsignal. y = utsignal.

Polynomen i $H(q^{-1})$ uttryckas som

$$(2.3) \quad \begin{cases} B = b_1 q^{-1} + b_2 q^{-2} + \dots + b_{\max d+2} q^{-(\max d+2)} \\ A = 1 + a_1 q^{-1} + a_2 q^{-2} \end{cases}$$

med

$\max d$ = (heltalsdelen av maximala tidsfördröjningen) + 1
 $b'_0 = b_d$, $b'_1 = b_{d+1}$ och $b'_2 = b_{d+2}$. $b_i = 0$, annars.

Det är representationen (2.3) som SPID använder vid skattningen av processpolynomen. Skattningen görs med minsta-kvadrat-metoden.

$$\theta = [a_1 \ a_2 \ b_1 \ b_2 \ \dots \ b_{\max d+2}]^T$$

$$\varphi(N) = [-y(N-1) \ -y(N-2) \ u(N-1) \ u(N-2) \ \dots \ u(N-\max d-2)]$$

Skattningen ges då av

$$\hat{\theta}(N+1) = \hat{\theta}(N) + K(N+1)[y(N+1) - \varphi(N+1)\hat{\theta}(N)]$$

med

$$K(N+1) = \frac{P(N)}{\lambda + \varphi(N+1)^T P(N) \varphi(N+1)}$$

(estimator förstärkning)

$$P(N+1) = P(N) - K(N+1) \varphi(N+1)^T P(N)$$

(estimatorfelets varians)

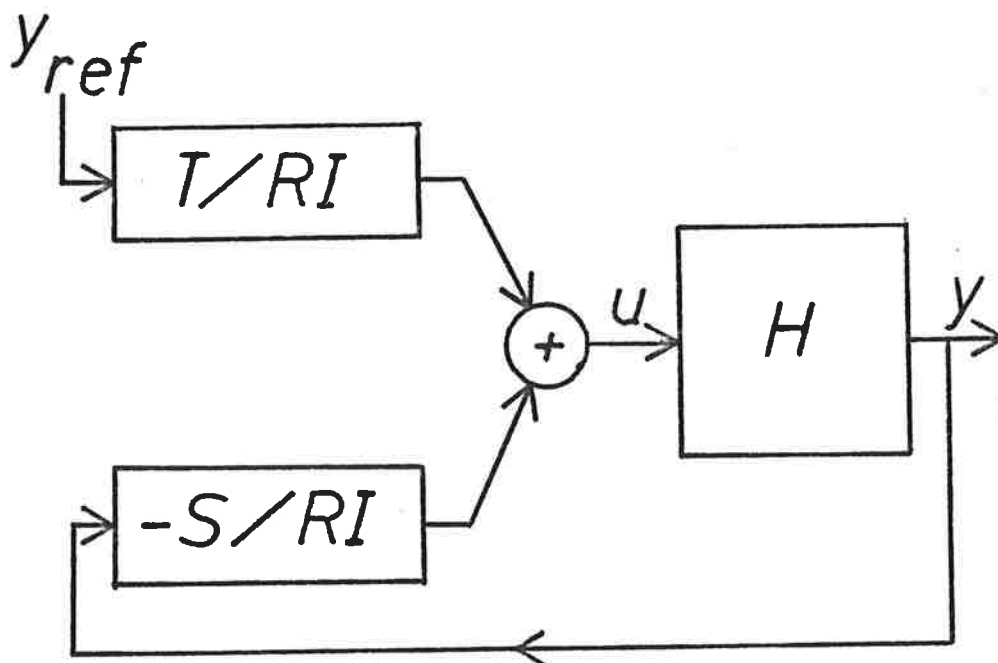
$$0 < \lambda \leq 1$$

(glömskefaktor)

(se ref [3], kap.13).

Eftersom det här är, i princip, samma skattningsalgoritm som i ref [1], nyttjas här ungefär samma parameterintervall. Glömskefaktorn, λ , väljes här till 0.96 och osäkerheten i initialvärden hos polynomkoefficienterna, diagonalelementen i P-matrisen (initialvärden), väljs till 1000. Initialvärden hos nämnda koefficienter väljs i simuleringarna till antingen 0 eller 0.1. Att insvängningsförloppet då blir häftigt saknar betydelse för detta arbete. Avsikten är ej att studera insvängningsförlopp.

3. REGLERING



Figur 3.1 . Det slutna systemet.

Antag att regulatorn kan skrivas som (se fig.3.1):

$$RIu = Ty_{ref} - Sy \quad (3.2)$$

Låt:

$$R = 1 + r_1 q^{-1} + r_2 q^{-2} + \dots + r_{nb+1} q^{-(nb+1)}$$

$$I = 1 - q^{-1} \quad (\text{"integrator"})$$

$T =$ någon konstant (se nedan)

$$S = s_0 + s_1 q^{-1} + s_2 q^{-2}$$

$H =$ samplade processens överföringsfunktion

$$B = b_1 q^{-1} + b_2 q^{-2} + \dots + b_{nb+2} q^{-(nb+2)}$$

$$A_m = 1 + a_{m1} q^{-1} + a_{m2} q^{-2}$$

$$B_m = \text{se nedan}$$

med

$d = (\text{heltalsdelen av verklig tidsfördröjning}) + 1$
 B estimatorns täljarpolynom

$nb = \text{heltalsdelen av antagen maximal tidsfördröjning}$

A_m, B_m slutna systemets nämnar- respektive täljar-
 polynom

Slutna systemet ges av:

$$y = \frac{B T}{A_m R + B_m S} y_{\text{ref}}$$

Polplacering realiserar regulatorn (se ref [3], kap.10):

$$\frac{B T}{A_m R + B_m S} \equiv \frac{B_m}{A_m} \quad (\text{index } m = \text{modell, önskat system})$$

3.1 . Metod 1: Lösning av ekvationssystem.

Låt $B \equiv B_m$ för att undvika instabila inverser (se ref[3], kap.10) och

$$T \equiv \frac{A_m(1)}{B_m(1)} = \frac{A_m(1)}{B(1)} \quad (3.3)$$

så att statiska förstärkningen för det slutna systemet blir ett (1).

Kvarstår gör beräkningen av R- och S-polynomen för att uppfylla

$$A_m R + B_m S \equiv A_m \quad (3.4a)$$

Val av koefficienter till A_m -polynomet och dessa

koefficienters koppling till bandbredd och dämpning beskrivs i ref[3] kap.3.4 . Lösning av identiteten (3.4a) kan göras på ett otal olika sätt. Ett, är att skatta tidsfördröjningen, d (i B' -polynomet), explicit (se ref[1]). Ett annat, är att, som B -polynomet antyder, ansätta tillräckligt stor ordning på processens täljarpolynom. Tillräckligt innebär att nämnda polynom i estimatorn kan rymma den verkliga processens tidsfördröjning. Gemensamt för dessa två metoder att det blir frågan om att lösa ett ekvationssystem för att lösa identiteten (3.4a).

Då

$$AI = 1 + a'_1 q^{-1} + a'_2 q^{-2} + a'_3 q^{-3}$$

$$\text{med } a'_1 = a_1 - 1, a'_2 = a_2 - a_1, a'_3 = -a_2$$

fås identiteten (3.4a) explicit:

$$(1 + a'_1 q^{-1} + a'_2 q^{-2} + a'_3 q^{-3})(1 + r_1 q^{-1} + r_2 q^{-2} + \dots$$

$$\dots + r_d q^{-d}) + B(s_0 + s_1 q^{-1} + s_2 q^{-2}) \quad (3.4b)$$

med B som antingen (2.2) eller (2.3).

Inför matriserna och vektorerna

$$\underline{A} = \left[\begin{array}{cccc|cccc} 1 & 0 & 0 & 0 & \dots & 0 & b_1 & 0 & 0 \\ a'_1 & 1 & 0 & 0 & \dots & 0 & b_2 & b_1 & 0 \\ a'_2 & a'_1 & 1 & 0 & \dots & 0 & b_3 & b_2 & b_1 \\ a'_3 & a'_2 & a'_1 & 1 & & 0 & b_4 & b_3 & b_2 \\ 0 & a'_3 & a'_2 & & & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & a'_3 & & & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & & & & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & & & & 0 & b_{nb} & b_{nb-1} & b_{nb-2} \\ \hline \dots & 0 & a'_3 & a'_2 & a'_1 & 1 & b_{nb+1} & b_{nb} & b_{nb-1} \\ \dots & 0 & 0 & a'_3 & a'_2 & a'_1 & b_{nb+2} & b_{nb+1} & b_{nb} \\ \dots & 0 & 0 & 0 & a'_3 & a'_2 & 0 & b_{nb+2} & b_{nb+1} \\ \dots & 0 & 0 & 0 & 0 & a'_3 & 0 & 0 & b_{nb+2} \end{array} \right]$$

$$\Lambda = \left[\begin{array}{cccc|cccc}
 1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 \\
 a'_1 & 1 & 0 & 0 & \dots & 0 & 0 & 0 & 0 \\
 a'_2 & a'_1 & 1 & 0 & \dots & 0 & 0 & 0 & 0 \\
 a'_3 & a'_2 & a'_1 & 1 & \dots & 0 & 0 & 0 & 0 \\
 0 & a'_3 & a'_2 & \cdot & \dots & \vdots & \vdots & \vdots & \vdots \\
 0 & 0 & a'_3 & \cdot & \dots & \vdots & \vdots & \vdots & \vdots \\
 \vdots & \vdots & \vdots & \vdots & \dots & 0 & 0 & 0 & 0 \\
 \hline
 \dots & 0 & a'_3 & a'_2 & a'_1 & 1 & b_1 & 0 & 0 \\
 \dots & 0 & 0 & a'_3 & a'_2 & a'_1 & b_2 & b_1 & 0 \\
 \dots & 0 & 0 & 0 & a'_3 & a'_2 & 0 & b_2 & b_1 \\
 \dots & 0 & 0 & 0 & 0 & a'_3 & 0 & 0 & b_2
 \end{array} \right]$$

$$x = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \\ \vdots \\ \vdots \\ \vdots \\ r_{nb+1} \\ s_0 \\ s_1 \\ s_2 \end{bmatrix}$$

$$b = \begin{bmatrix} a_{m1} - a'_1 \\ a_{m2} - a'_2 \\ -a'_3 \\ 0 \\ \vdots \\ \vdots \\ \vdots \\ 0 \end{bmatrix}$$

Dessa vektorer och matriserna \underline{A} eller $\underline{\Delta}$ bildar ekvationssystemen:

$$\underline{A}\bar{x} = \underline{b} \quad (3.5a)$$

$$\underline{\Delta}\bar{x} = \underline{b} \quad (3.5b)$$

beroende på vilken representation av B-polynomet som väljs.

Vektorn \bar{x} representerar regulatorparametrarna och \underline{b} skillnaden mellan koefficienterna hos det slutna systemets nämnarpolynom och den estimerade processens nämnarpolynom.

Ekvationssystemet (3.5b) har sådan symmetri att det kan lösas med handräkning medan (3.5a) är mindre lämpat för detta. Det är (3.5a) som är aktuellt i detta examensarbete. I ref[1] används representationen (3.5b)

3.2. Realisering av Metod 1.

Det Fortransystem som skall motsvara Metod 1-regulatorn finns listat i appendix A.

För lösning av ekvationssystemet (3.5a) användes biblioteksrutinerna Decom och Solve.

Beräkning av R- och S-polynomen kan ske genom invertering av \underline{A} . Det kan emellertid hända att \underline{A} är singular. Detta inträffar då A- och B-polynomen har gemensamma faktorer. I programmet behandlas detta fenomen så, att när Decom meddelar att \underline{A}^{-1} är en illa konditionerad matris (detta genom att sätta IERR \neq 0), görs ej heller någon lösning av ekvationssystemet. Istället används de från föregående samplingsintervall, beräknade regulatorparametrarna - regulatorn uppdateras ej.

En test görs också på T (=TP i programmet), (3.3). Om $\sum b_i$ är liten, begränsas T till 10000. Under exempelvis insvängningsförloppet är det möjligt att $\sum b_i$ någon gång blir liten.

Osäkerheten i skattningarna, matrisen P, är symmetrisk och lagras därför som triangulär i Simnon. För utförligare information om överföring av variabler mellan Simnon och Fortran, se ref[2].

3.3. Metod 2: Tvåstegs skattning av regulatorkoeff.

För Metod 2 har följande strategi använts:

Atergå till identiteten (3.4a).
Multiplitera höger- och vänster-led med utsignalen y . Och utnyttja att $Ay = Bu$.

$$AIRy + BSy \equiv A y_m \Leftrightarrow RIBu + BSy \equiv A y_m$$

$$\text{Låt } IBu(k) = \Delta u_f(k)$$

$$By(k) = y'_f(k)$$

$$A y_m(k) = y_f(k)$$

där

f = filtrering

k = tid

Med nya beteckningar:

$$R\Delta u_f + S y'_f \equiv y_f \quad (3.6)$$

Identiteten (3.6) kan tolkas som R- och S-polynomen, verkande på filtrerade ut- och insignaler.

Enligt ref [3], kap. 13.5, kan nu R- och S-polynomens koefficienter skattas med minsta-kvadrat-metoden. Med samma nomenklatur som tidigare, kan detta uttryckas som:

$$\theta = [r_1 \quad r_2 \quad \dots \quad r_{nb+1} \quad s_0 \quad s_1 \quad s_2]^T$$

$$\varphi(k) = [\Delta u_f(k-1) \quad \Delta u_f(k-2) \quad \dots \quad \Delta u_f(k-nb-1) \\ y'_f(k) \quad y'_f(k-1) \quad y'_f(k-2)]$$

$$\text{och tidigare } [y - \varphi\theta] \text{ byts mot } [y_f(k) - \Delta u_f(k) - \varphi\theta]$$

Insvängningen mot stationära koefficienter, både process- och regulator-polynomens, är mer komplex och tar längre tid relativt Metod 1. Se kapitel Simuleringar.

3.4 Realisering av Metod 2.

Fortransystemet som motsvarar regulatorn i Metod 2 finns listat i appendix B.

Samma algoritm som skåttar processpolynomen, används vid skattning av regulatorpolynomen. Då behövs ytterligare en glömskefaktor - här döpt till λ_c . Simuleringar visar

att $\lambda_c = 0.94$ är lämplig. Denna intrimning redovisas ej.

4.1. SIMULERINGAR

Följande simuleringar är gjorda på en VAX 11/780. Simuleringspråket är Simnon. Vissa av Simnonsystemen är skrivna i Fortran. Fortranprogrammen är tidigare beskrivna och Simnonprogrammen är listade i appendix C.

De simulerade processernas överföringsfunktioner är

$$H_1 = \exp(-t_{del} s) \frac{b^2}{(s + b)^2}$$

och

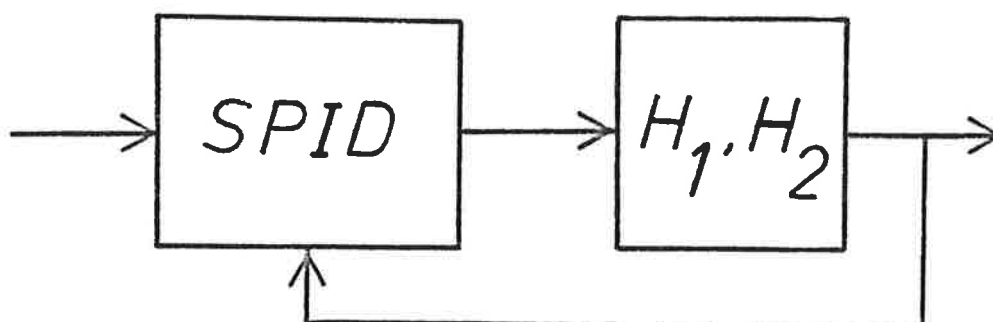
$$H_2 = \exp(-t_{del} s) \frac{b^4}{(s + b)^4}$$

Tidsfördröjningen t_{del} varieras på olika sätt.

Slutna systemet kan beskrivas enligt fig 4.1

Alla tidsangivelser är angivna i antal samplingsintervall. Simuleringarna är här gjorda med tidsfördröjningen varierande i steg, från 4 till 1 och vice versa, samt kontinuerligt varierande, från 3 till 1 och vice versa. Den kontinuerligt varierande tidsfördröjningen följer en rampfunktion med lutningen 0.004.

Vidare har simuleringar av det samplade systemet H_1 med $b = 0.3$ och för ovan nämnda utseende på t_{del} , gjorts.



Figur 4.1 . Slutna systemet

Härledning av de exakta koefficient-beräkningarna och listning av program för dessa återfinns i appendix D.

H har simulerats med $b = 1.0$ och för fast ($= 2$) respektive

kontinuerligt varierande tidsfördröjning.

Här kommer nu att redovisas:

Fall 1 = Metod 1:s användning på beskriven process.

Fall 2 = Metod 2:s användning på beskriven process.

4.1 . . Diskret varierande tidsfördröjning på system av 2:a ordningen

Tidsfördröjningen varierar enligt figur 4.2. Av fig. 4.3a&b framgår att då tidsfördröjningen minskar, klarar regulatören lättare av att ta hand om förändringar i B-polynomet, än då tidsfördröjningen ökar. Av utseendet hos utsignalerna kan redan här dras slutsatsen att Metod 1 är bättre än Metod 2. Styrsignalen hos Metod 1-regulatören visar (fig.4.4a) snabbare adaptation för att fylla kraven på slutna systemet, än Metod 2 (fig.4.4b). Slutna systemets bandbredd, $\omega_0 = 0.5$,

är vald så att utsignalen ej skall nå begränsningarna (± 5.0) då systemet stabiliserat sig.

Figurerna 4.5a&b visar skattningarna av A- och B-polynomens koefficienter. De "raka och okrusade" linjerna i figurerna är de exakta koefficienterna, beräknade enligt appendix D. Det är Fall 1 som avbildats, enär Fall 2 ej avviker nämnvärt i processskattningen. För att förse estimatorn med tillräcklig information om processen, krävs ca 5 - 6 transienter (referensvärdes ändringar).

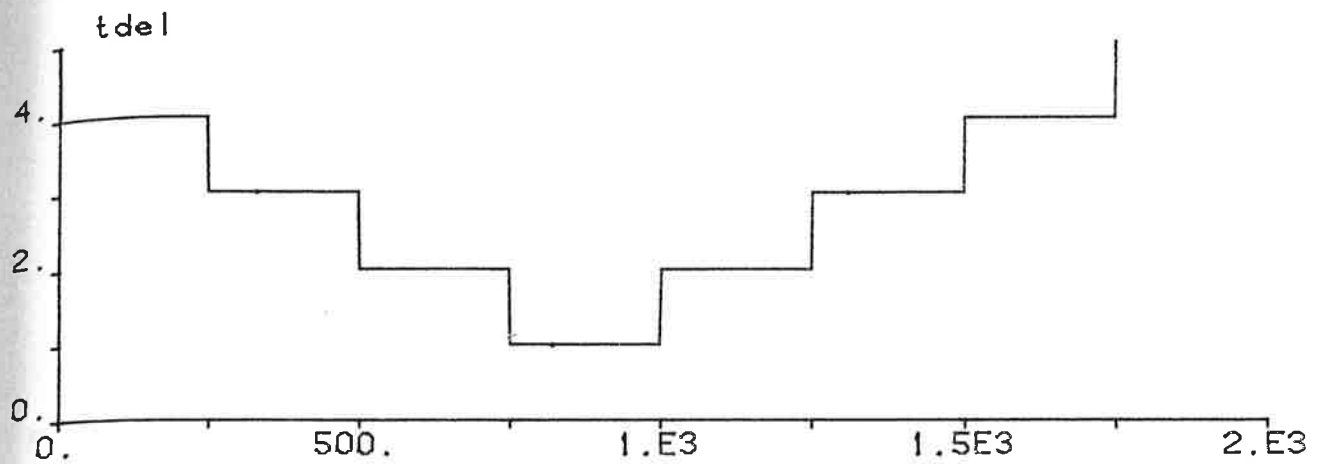
I figurerna 4.6a&b samt 4.7a,b&c ses tydligt skillnaden mellan Metod 1 & 2. Det tar ungefär dubbelt så lång tid, ibland mer, för Metod 2 att skatta värdena på regulatorkoefficienterna, som det gör för Metod 1. Figur 4.7c kan betraktas som ett "worst case". Att Metod 2 är sämre än Metod 1 kan förklaras av den dubbla skattning som görs. Den mer komplexa insvängningen ger en långsammare konvergens.

Simuleringar, vilka ej redovisas, har också gjorts för $b = 1.0$. Resultatet av dem är de samma som ovan, frånsett att ω_0 kan ökas utan att styrsignalen blir för stor samt att

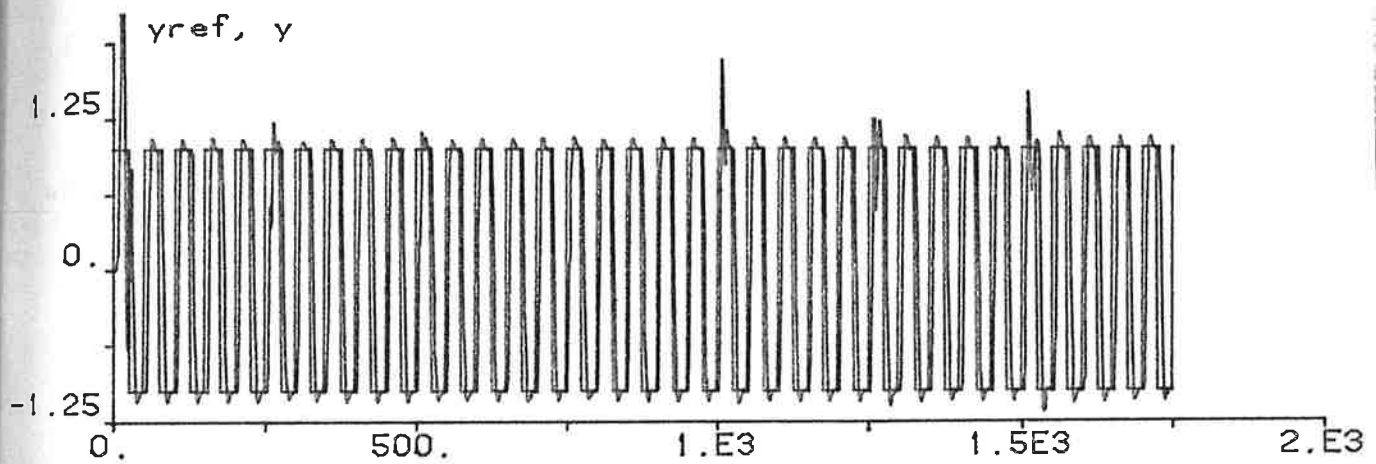
utsignalen från Metod 2 blev betydligt "slängigare".

Figurerna 4.2 till 4.7c kan jämföras med figurerna 5.4a&b i ref[1]. En sådan jämförelse visar att både Metod 1&2 är lika bra som den i ref[1] använda metoden. Det bör observeras att det är olika bandbredder för slutna systemet, här och i ref[1].

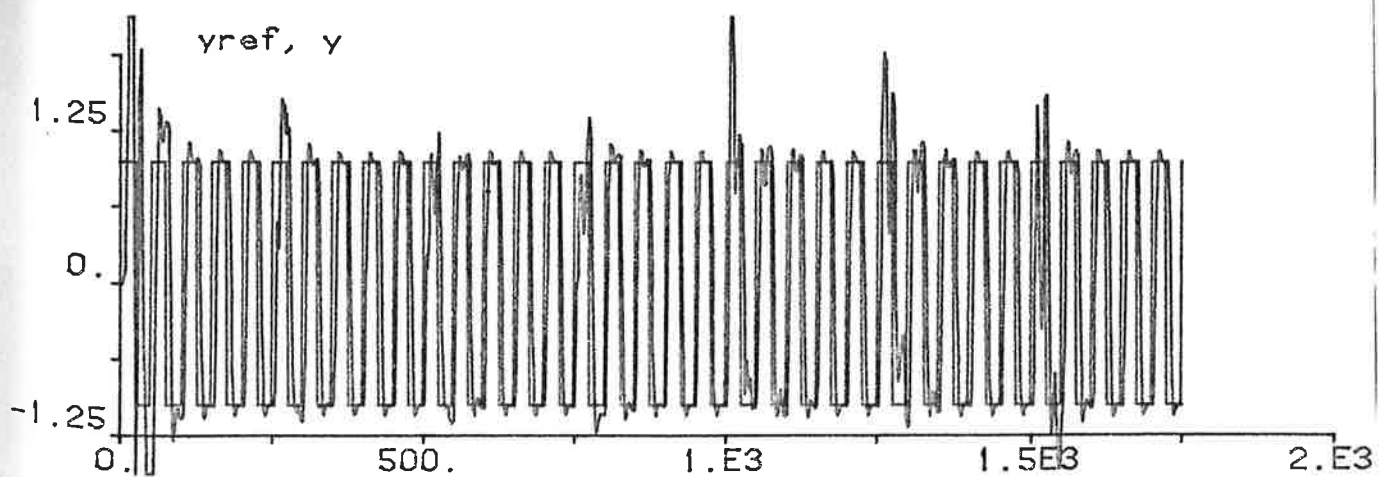
simuleringar på process H 1



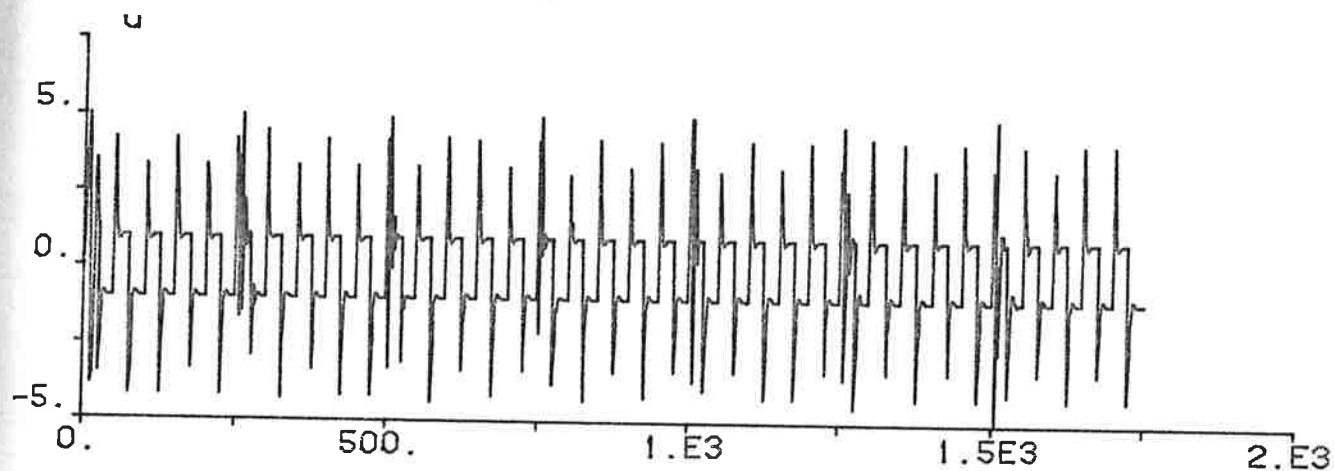
Figur 4.2 . Tidsfördröjningens variation i tiden.



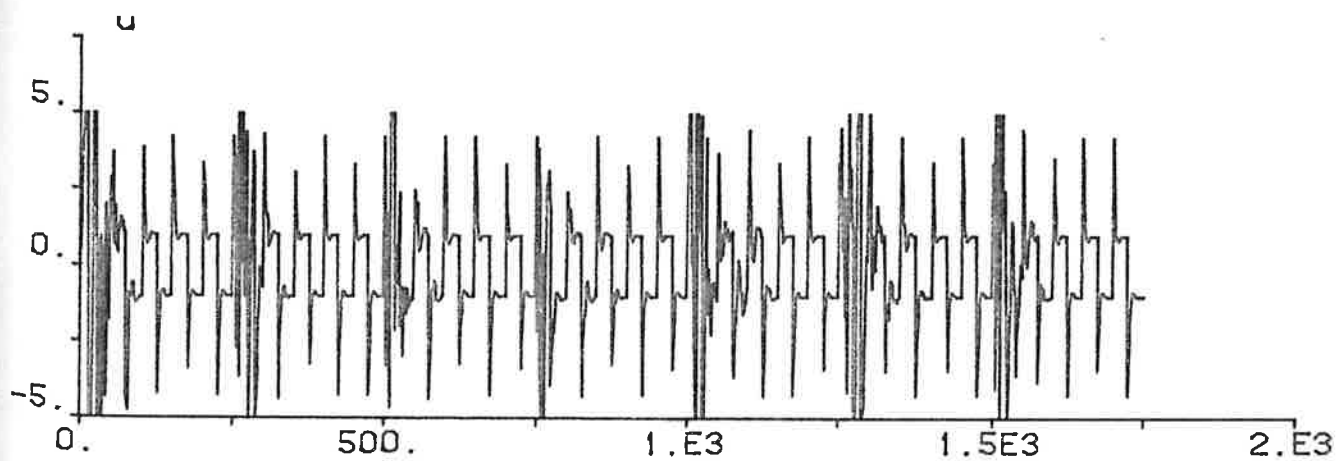
Figur 4.3a. Utsignal och referenssignal, y och y_{ref} , i
Fall 1. Slutna systemets bandbredd, $\omega_0 = 0.5$.



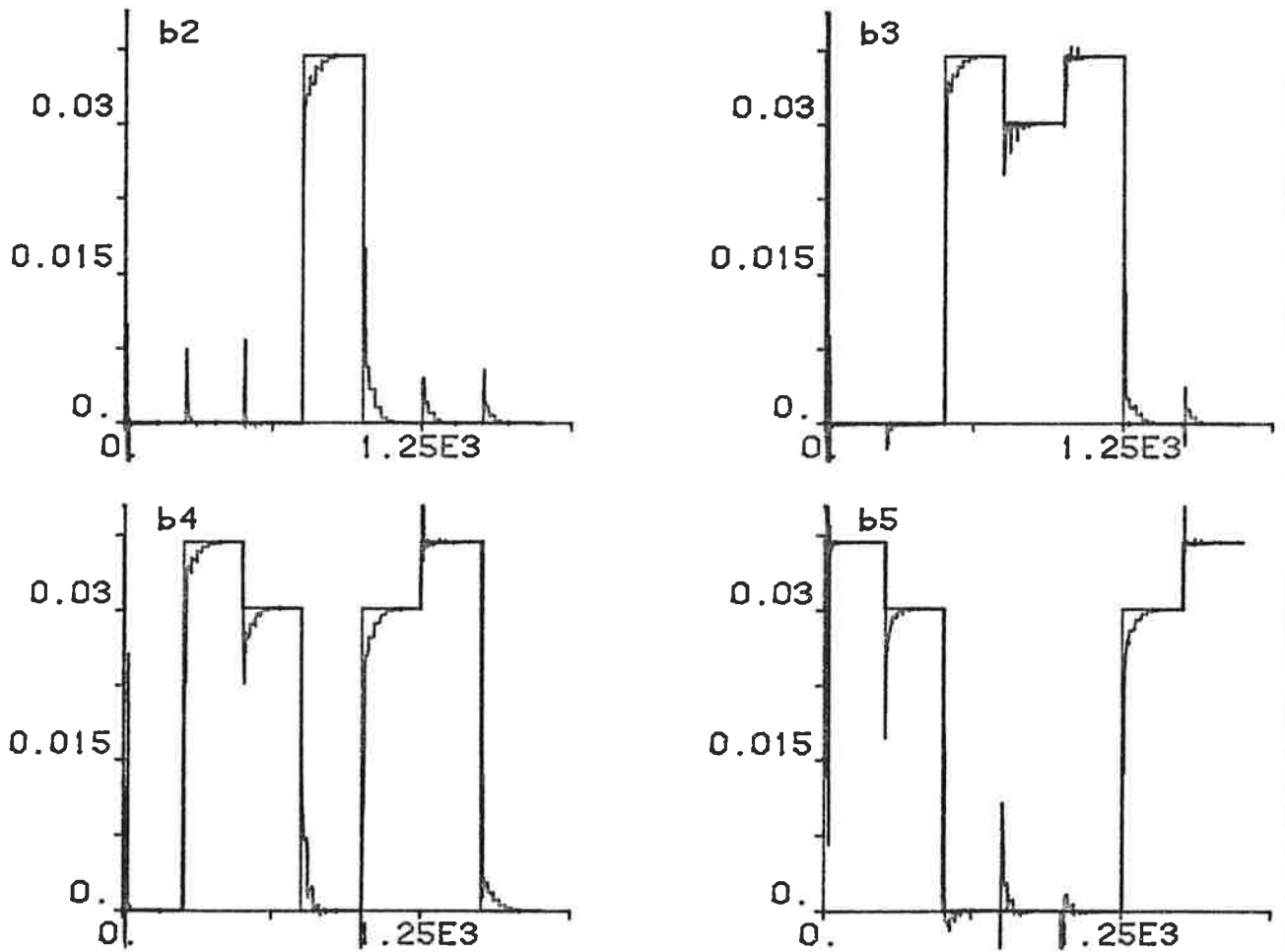
Figur 4.3b. Utsignal och referenssignal, y och y_{ref} , i
Fall 2. $\omega_0 = 0.5$.



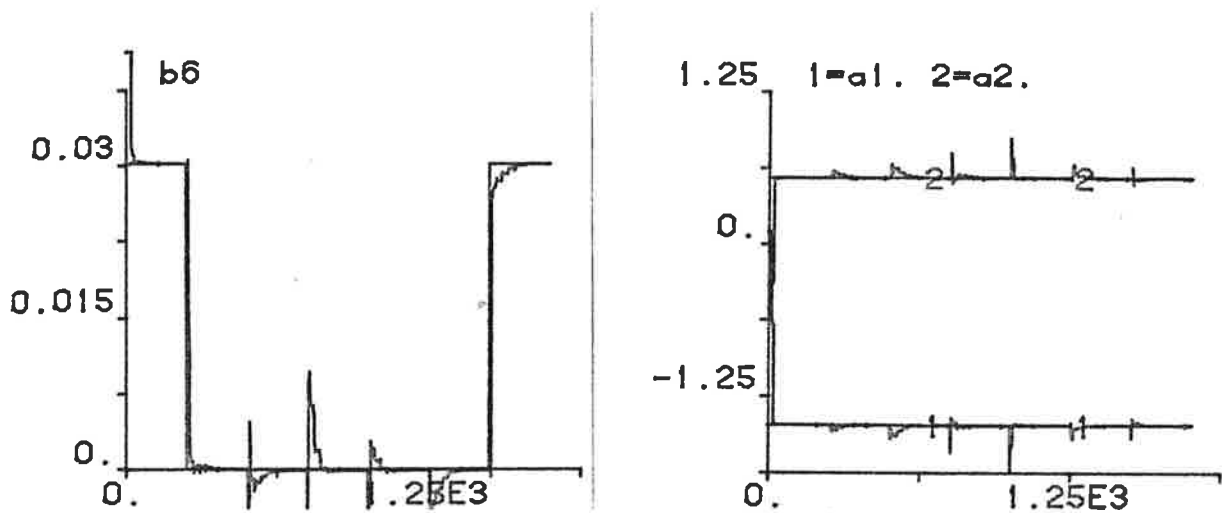
Figur 4.4a. Styrsignal till processen i Fall 1.



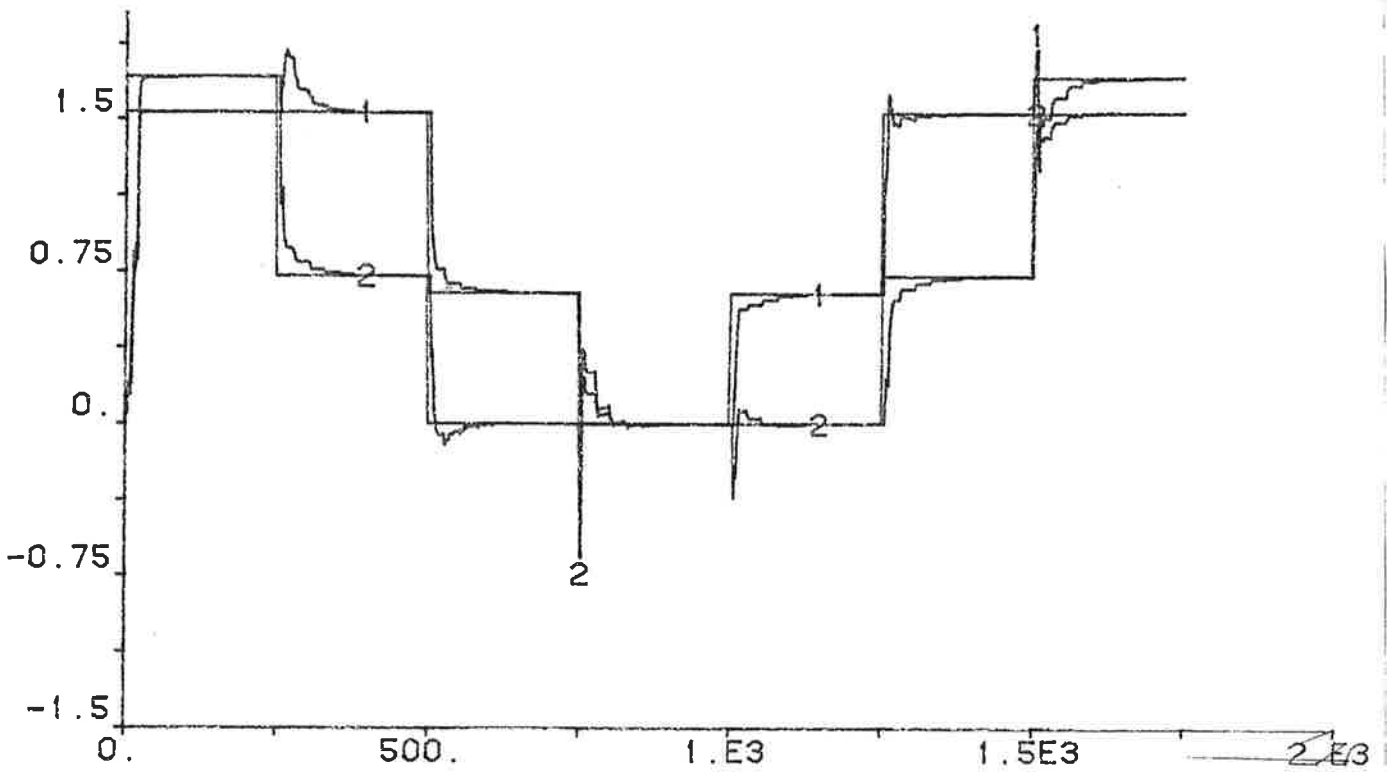
Figur 4.4b. Styrsignal till processen i Fall 2.



Figur 4.5a. Estimerade och verkliga koefficienter hos B-polynomet, $b_2 - b_5$ i Fall 1&2. $b_1 \approx 0$, plottas ej.

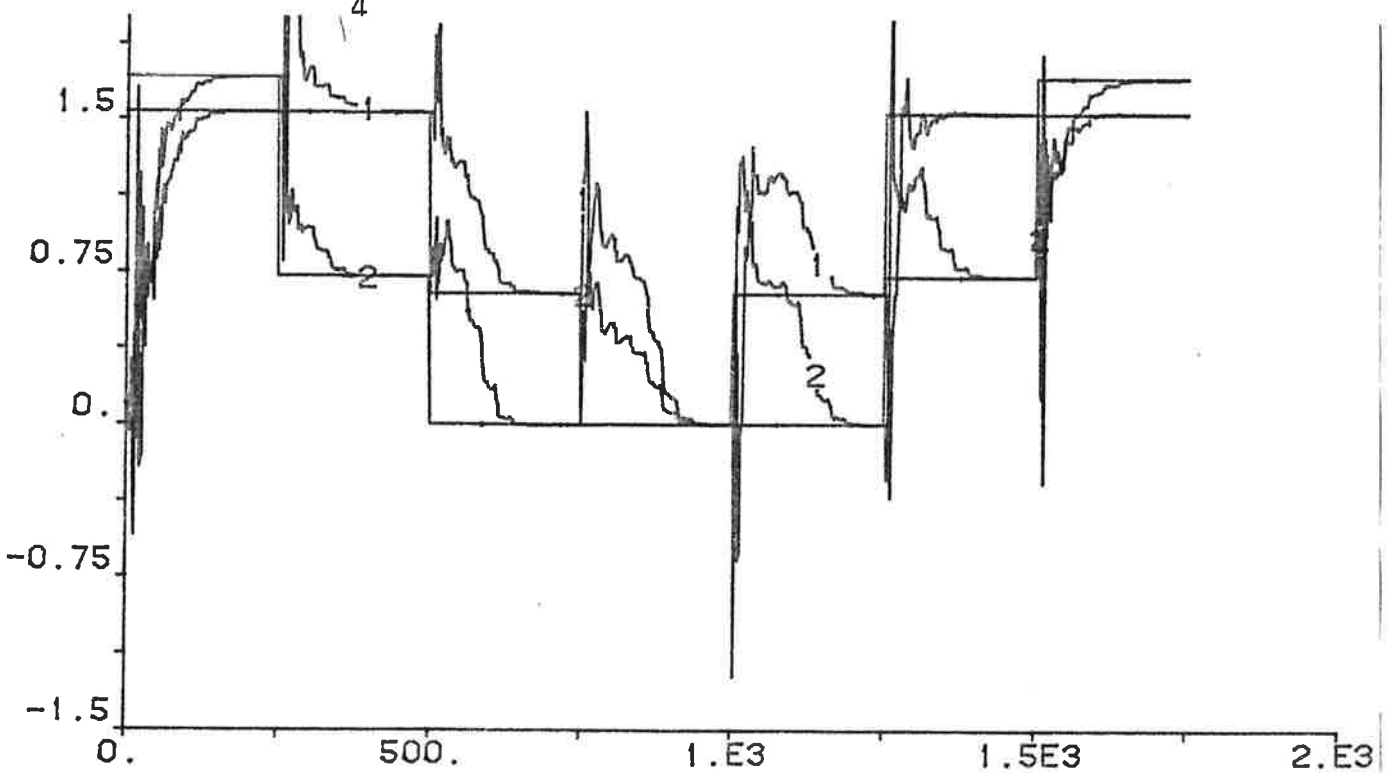


Figur 4.5b. Estimerade och verkliga koefficienter i B- och A-polynomet, b_6 , a_1 och a_2 : Fall 1&2



Figur 4.6a. Några koefficienter i R-polynomet: $1 \equiv r_3$.

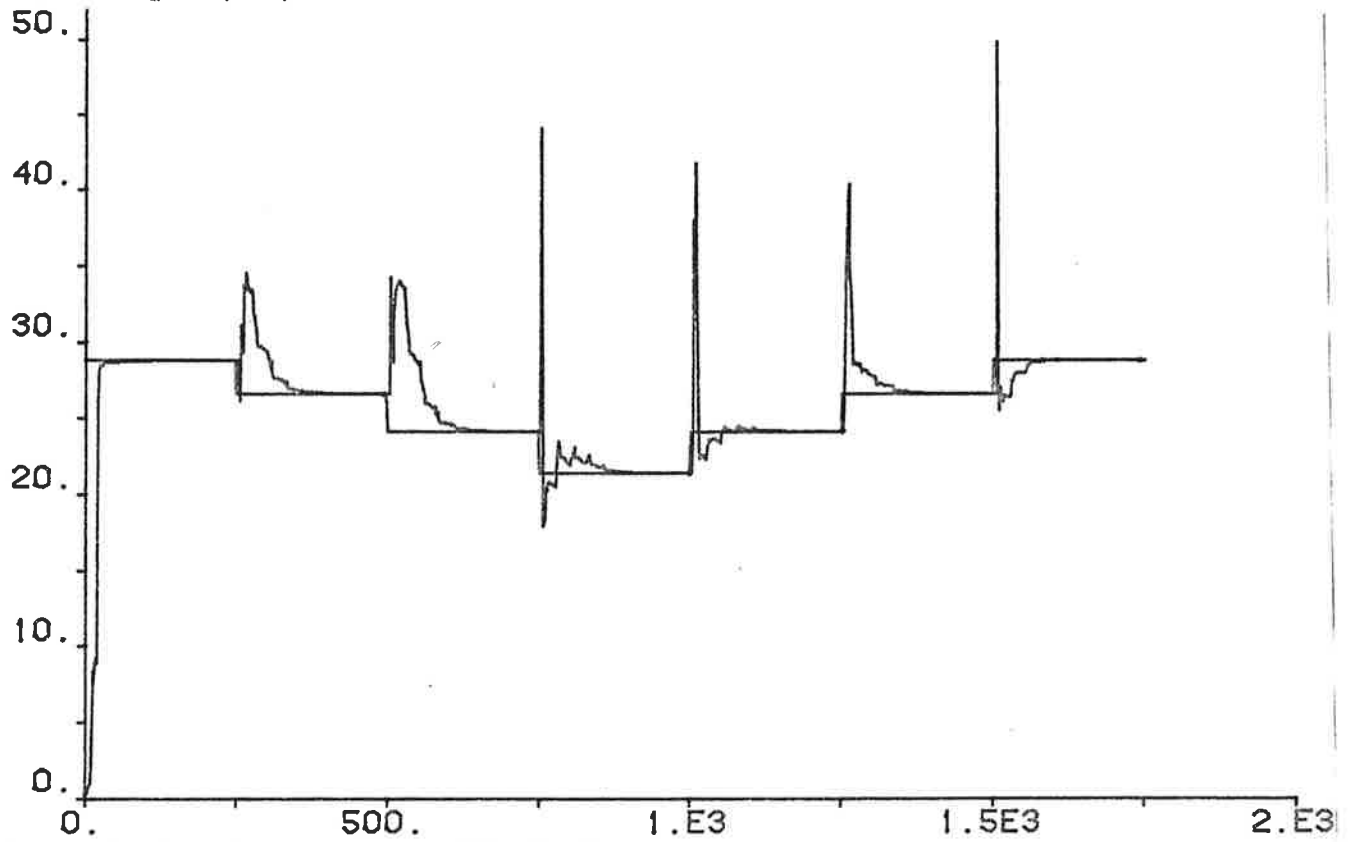
$2 \equiv r_4$, Fall 1.



Figur 4.6b. Några koefficienter i R-polynomet. $1 \equiv r_3$.

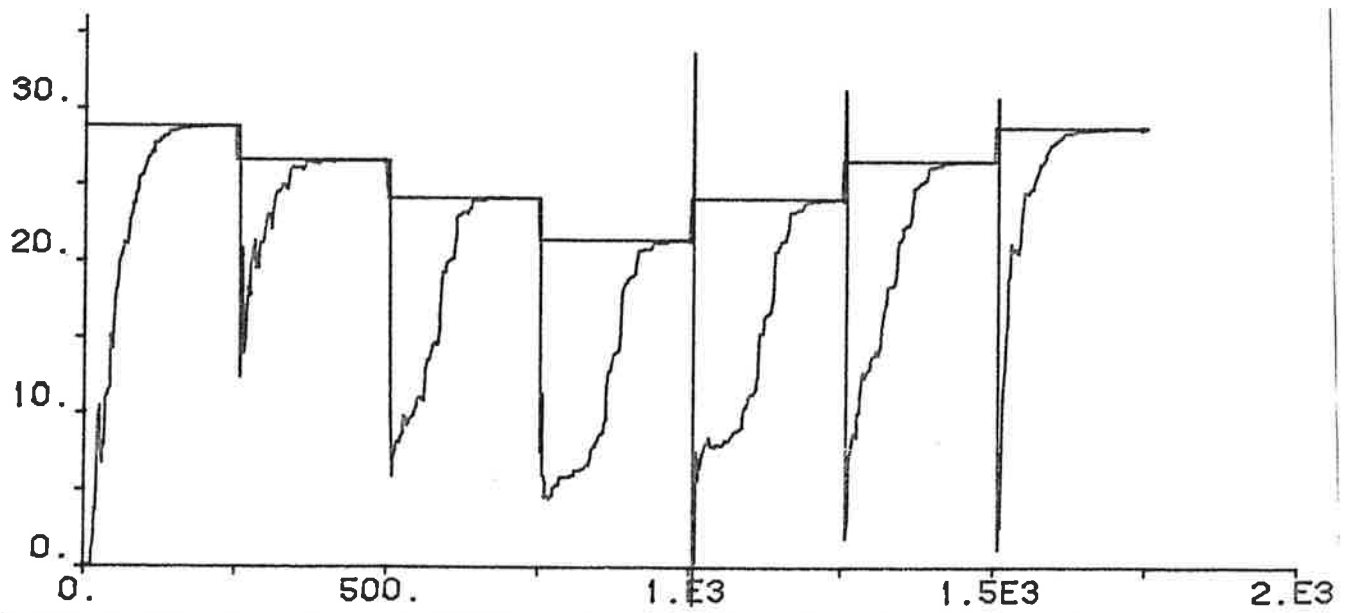
$2 \equiv r_4$, Fall 2

Simuleringar på process H 1



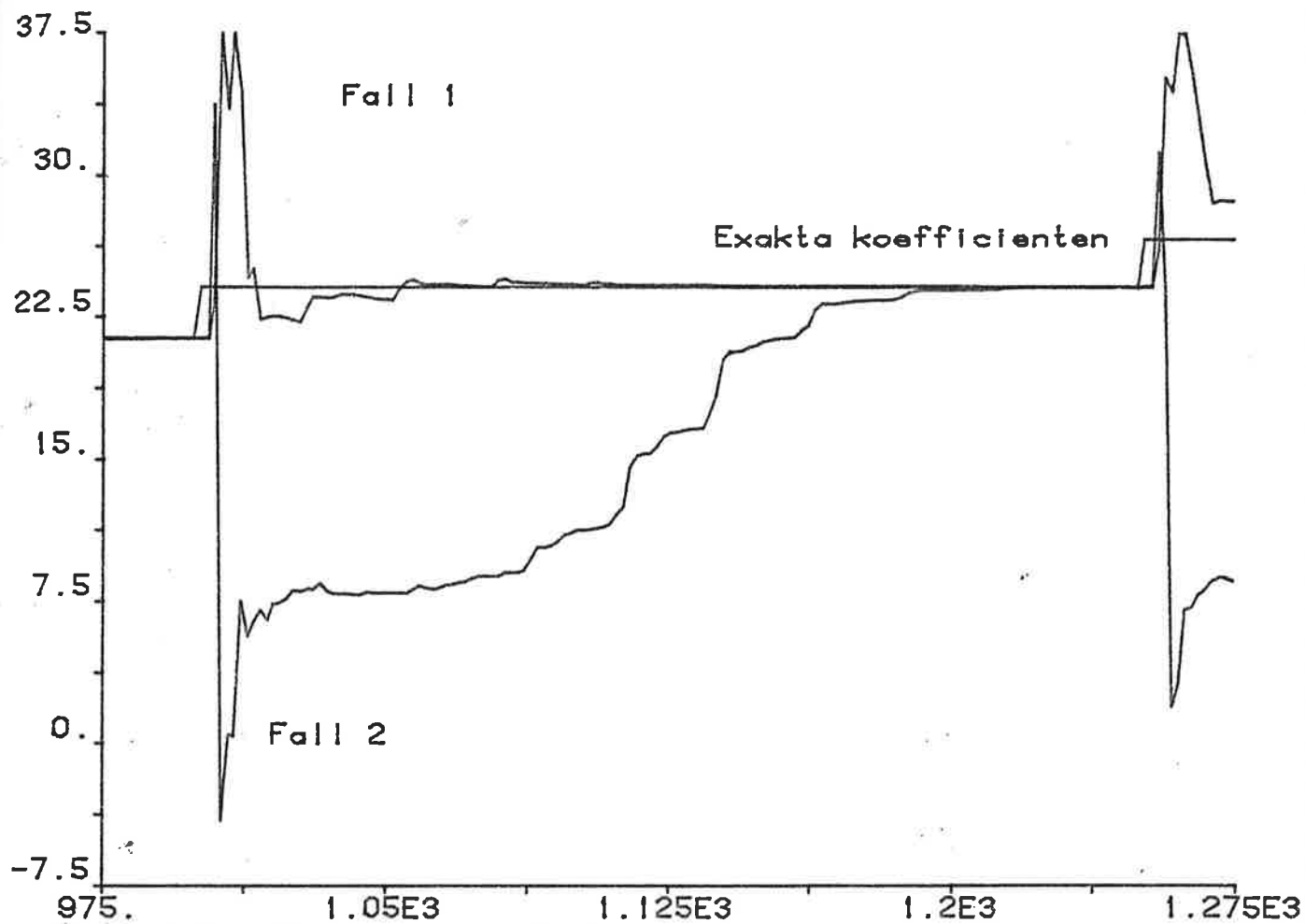
Figur 4.7a. En av koefficienterna i S-polynomet, s_0 .

Fall 1.



Figur 4.7b. En av koefficienterna i S-polynomet, s_0 .

Fall 2.



Figur 4.7c. Utdrag och uppförstoring, ur 4.7a och 4.7b.

4.2 . Kontinuerligt varierande tidsfördröjning på system av 2:a ordningen

Tidsfördröjningen varierar enligt figur 4.8. Den mindre drastiskt varierande tidsfördröjningen gör att regulatorerna hinner med bättre än tidigare. Jämför figurer 4.9a&b med 4.3a&b. Styrsignalen blir också lugnare, figurer 4.10a&b. Med avseende på utsignal från det slutna systemet, är Metod 1 bättre än Metod 2.

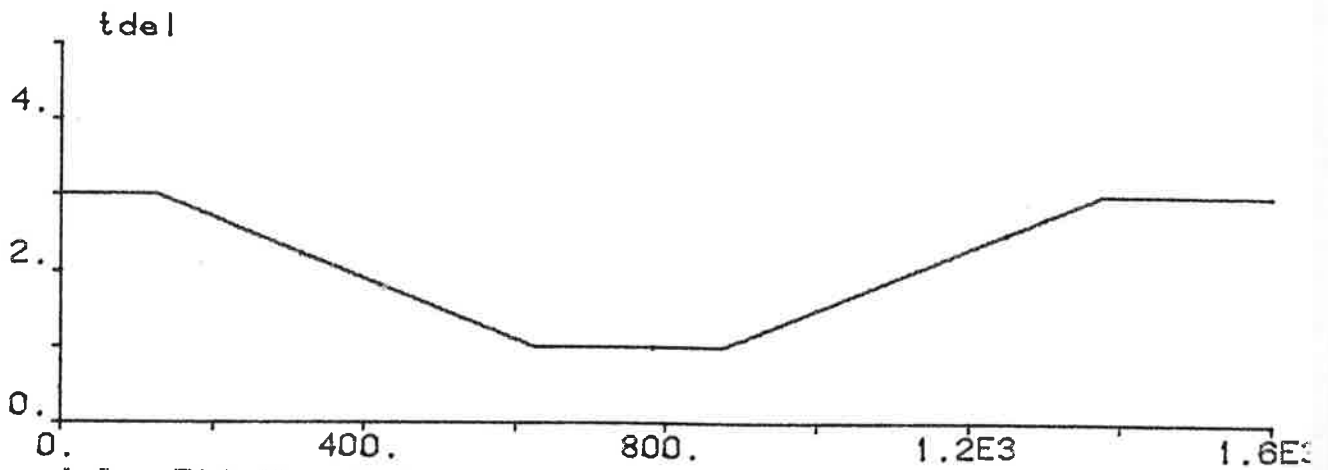
Som figur 4.11 visar sker uppdatering vid varje referensvärdesändring då tidsfördröjningen ändrar sig. Uppdatering sker endast då processen exciteras. Frekvensinnehållet hos in- och ut-signal räcker ej för att "informera" estimatorn annat än vid ändringen i y_{ref} . Det är

Metod 1:s estimering som plottats: Metod 2 mycket lik. Skattningen av A-polynomets koefficienter är i stort sett den samma som 4.5.b.

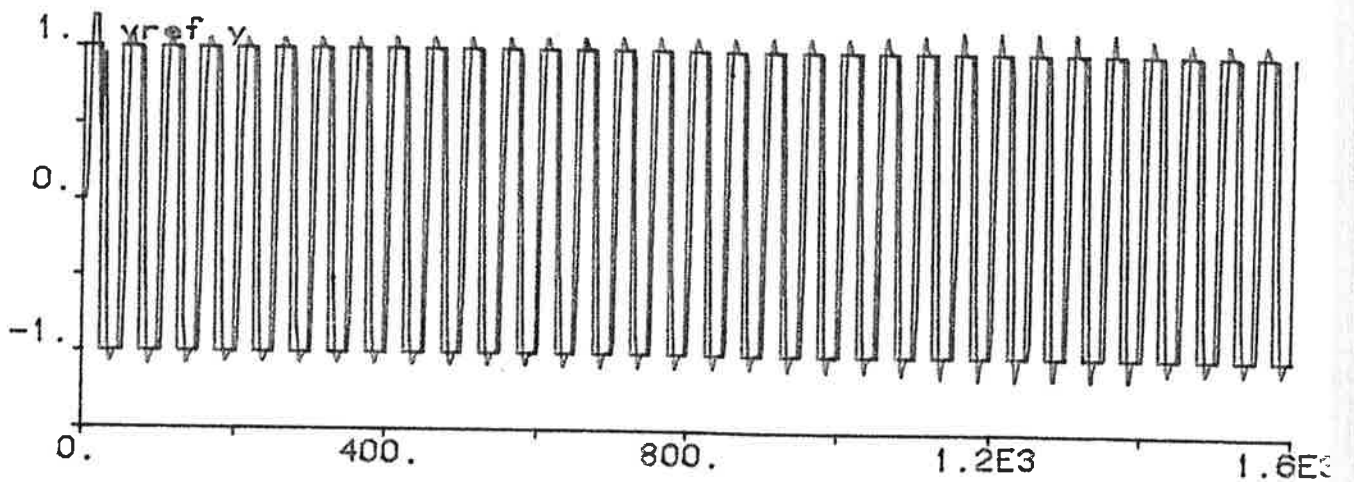
Jämförelse av figurer 4.12a och 4.12b pekar på skillnaden mellan exakt och skattad styrlag. Vid tidpunkterna 375 och 1120 sker övergång på helt antal samplingsintervall i tidsfördröjningen. Här har Metod 2 svårt att finna de rätta styrlagskoefficienterna (figurer 4.12b och 4.13b). Av figurutseendet att dömma, verkar det som om Metod 2 "tappar tråden" då tidsfördröjningen ≤ 2.0 , men stabiliserar sig då tidsfördröjningen varit konstant ($= 1.0$) ett tag.

Figurer 4.12b.i&ii antyder att estimatorn i Metod 2 inte riktigt kan skilja på koefficienterna. Vad denna instabilitet beror på utreds ej i detta examensarbete.

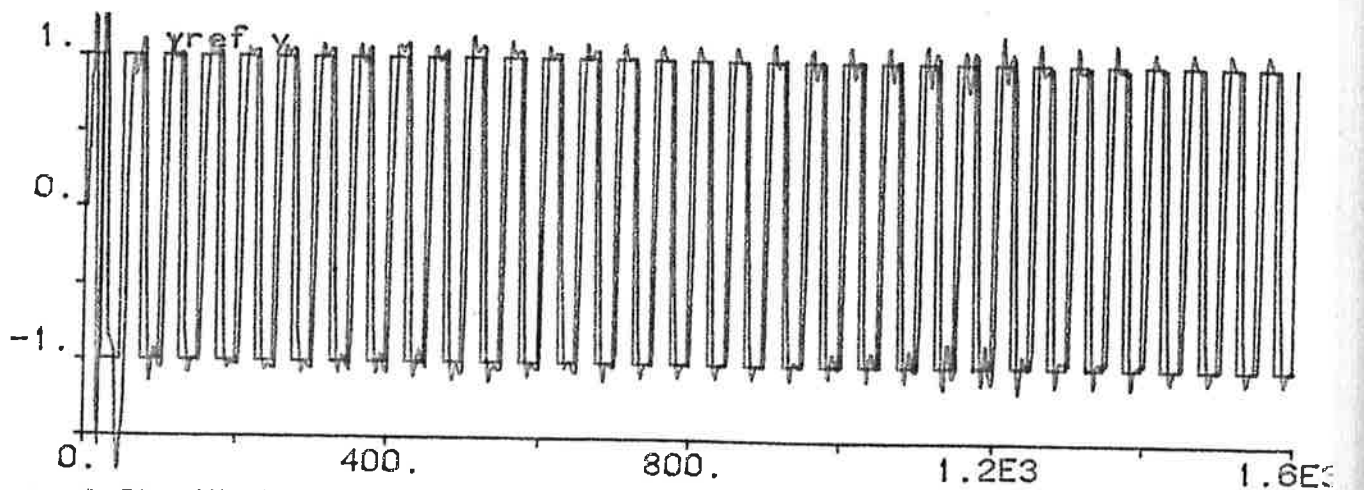
Jämförelser kan göras med figur 5.6 i ref[1]. Denna blir ej rättvis då Andell använt en betydligt brantare ramp vid ändringen av tidsfördröjningen. De här redovisade metoderna skulle med tvekan klara av denna snabba förändring, om ej antalet transienter (referensvärdesändringar) också ökades. För att estimatorn skall hinna med en snabbare ändring i tidsfördröjningen, måste den få mer information om processen. Hur kopplingen mellan rampens branthet, antalet transienter och regulatorns adaptation förhåller sig har ej utretts här.



Figur 4.8 . Tidsfördröjningens variation i tiden.

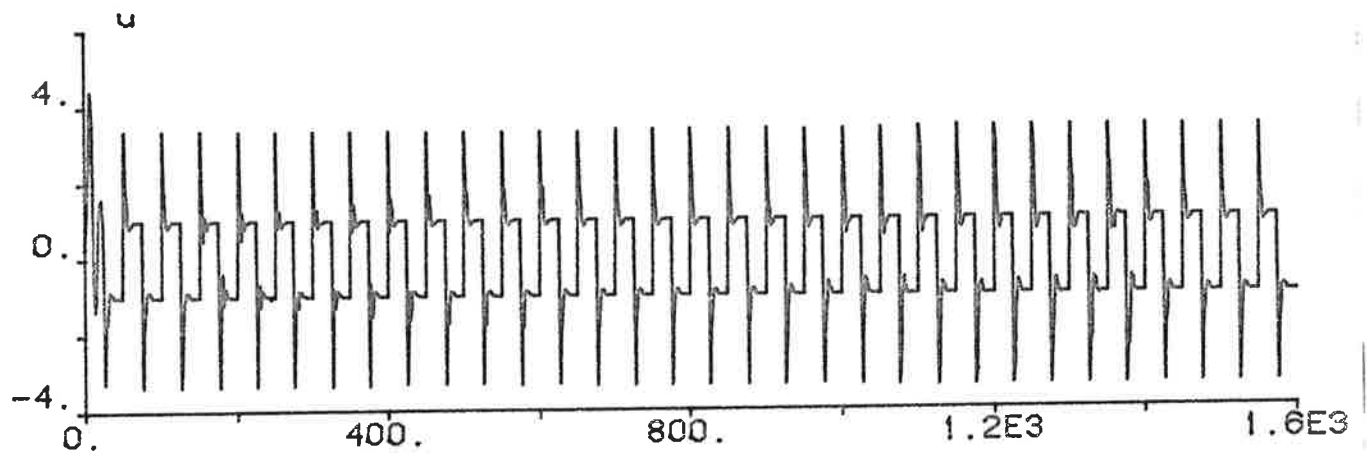


Figur 4.9a. Utsignal och referenssignal, y och y_{ref} , i
Fall 1. Slutna systemets bandbredd, $\omega_0 = 0.5$.

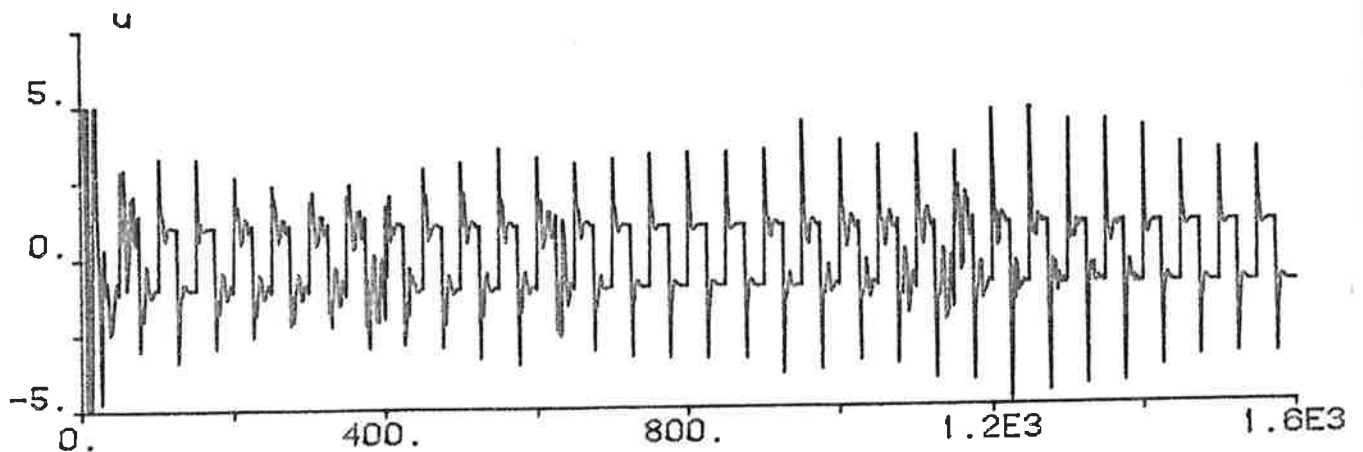


Figur 4.9b. Utsignal och referenssignal, y och y_{ref} , i
Fall 2. $\omega_0 = 0.5$.

Simuleringar på process H 1

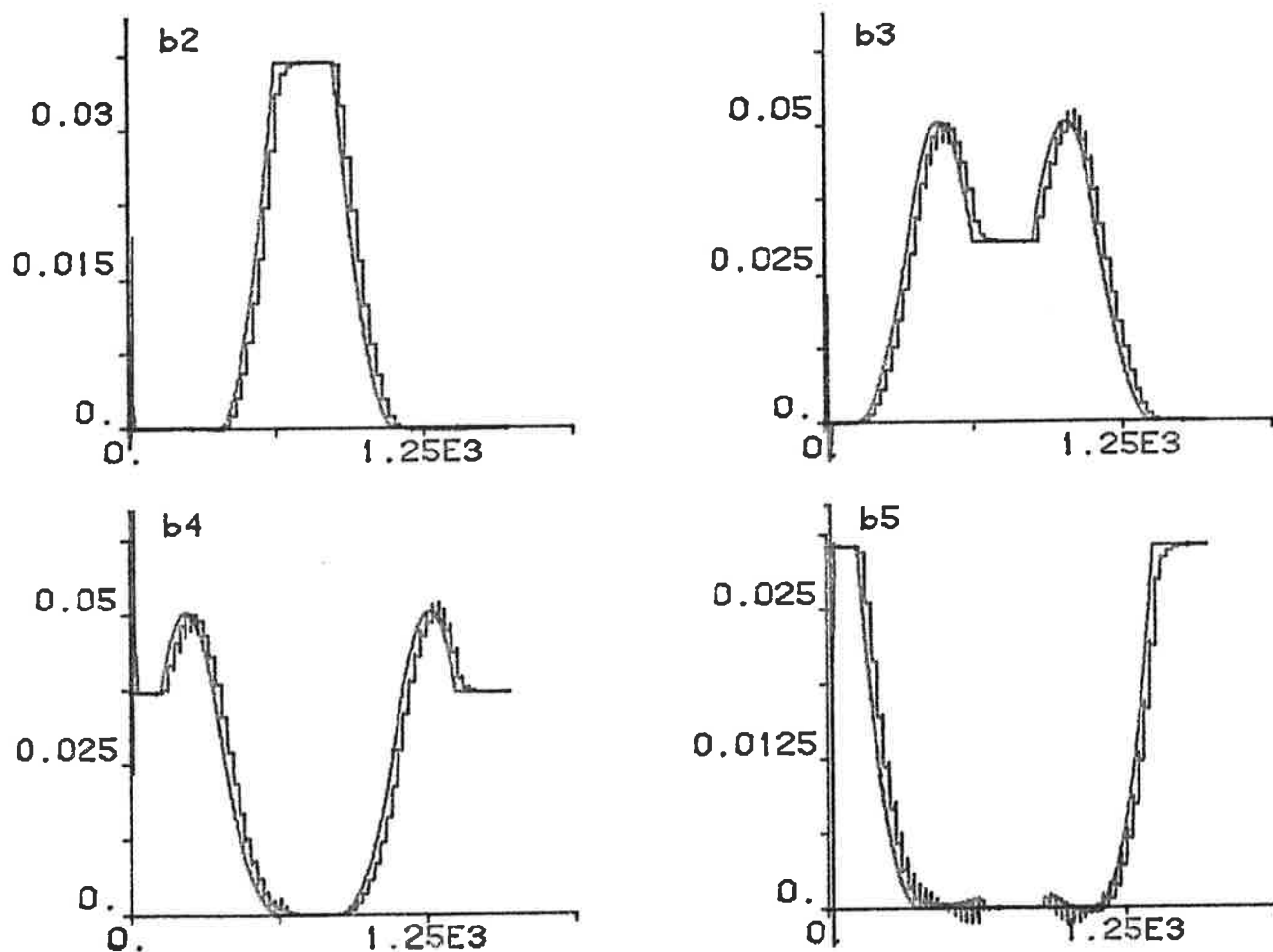


Figur 4.10a. Styrsignal till processen i Fall 1.



Figur 4.10b. Styrsignal till processen i Fall 2.

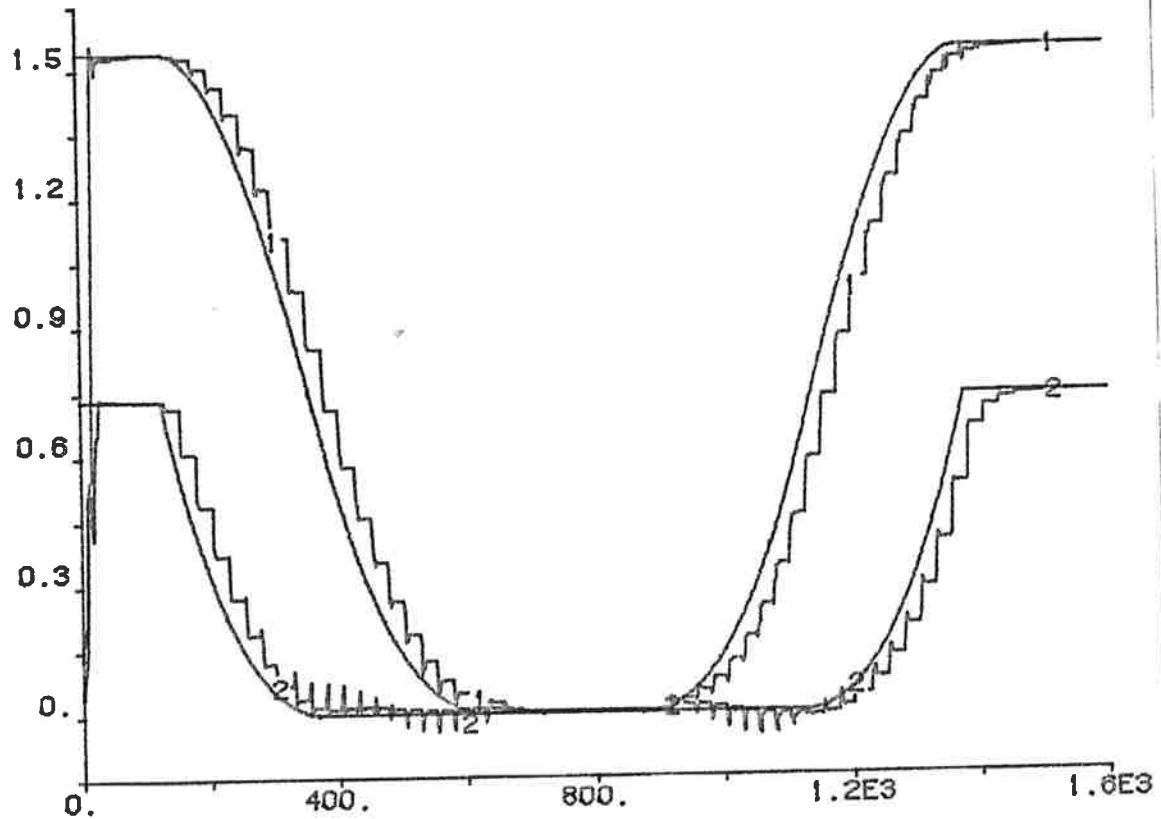
Simuleringar på process H 1



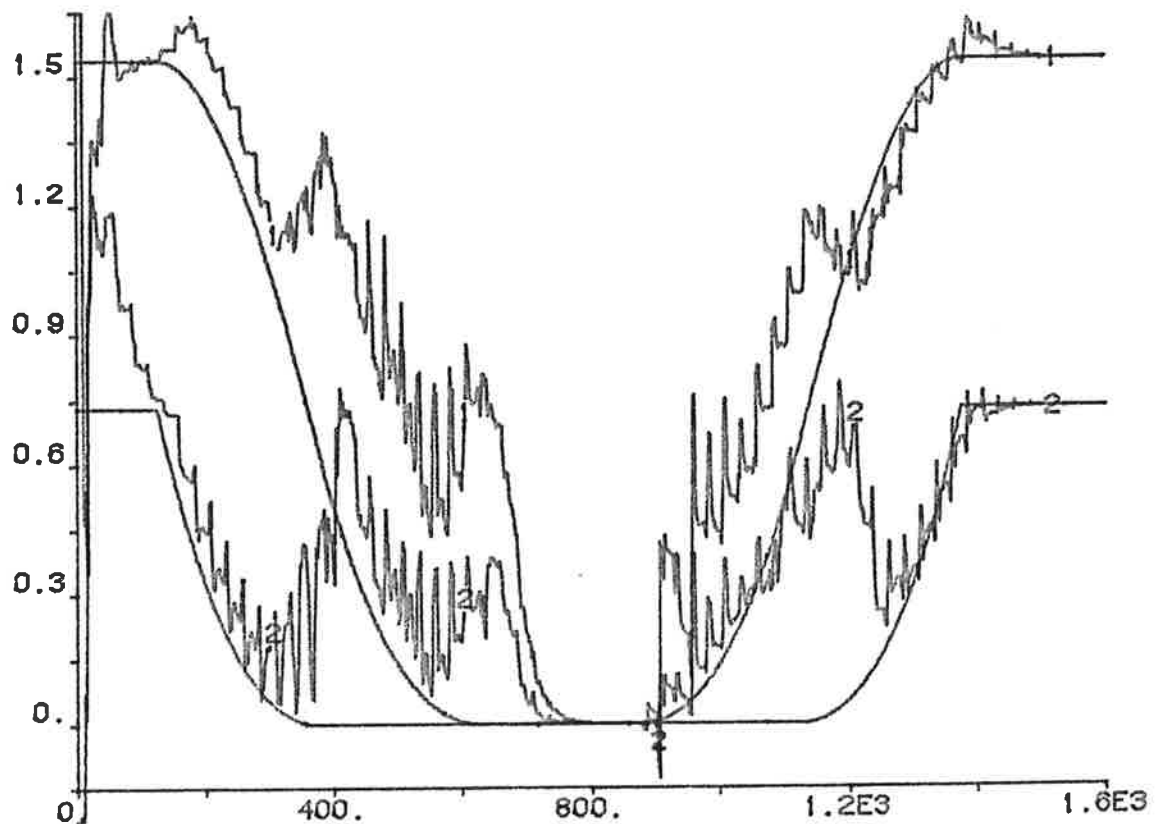
Figur 4.11. Estimerade och verkliga koefficienter hos B-polynomet i Fall 1 och Fall 2. $b_1 \approx 0$,

$b_6 \approx 0$, plottas ej.

Simuleringar på process H 1

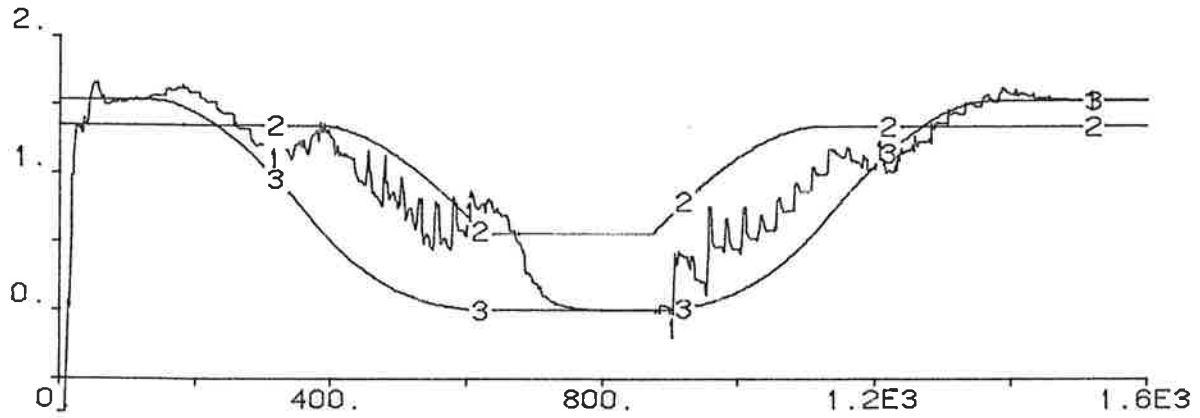


Figur 4.12a. Några koefficienter i R-polynomet: $1 \equiv r_3$.
 $2 \equiv r_4$. Fall 1.

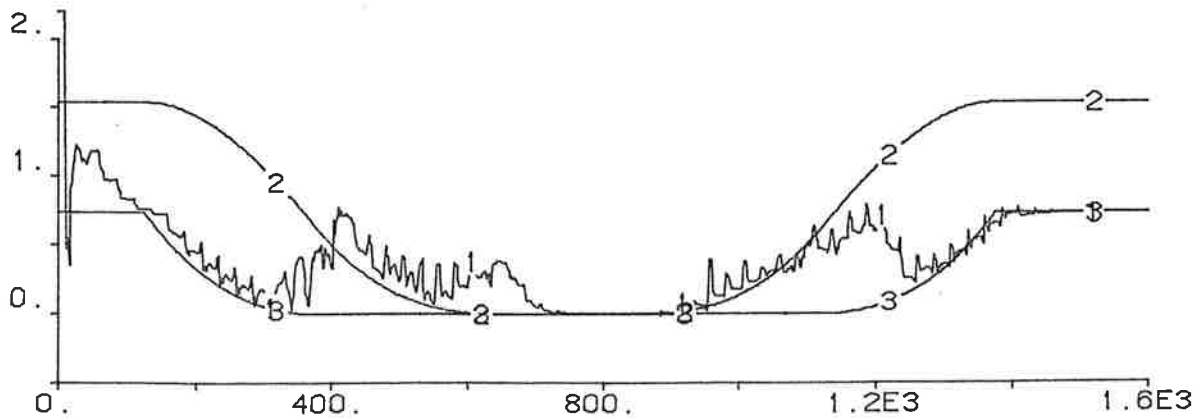


Figur 4.12b. Några koefficienter i R-polynomet. $1 \equiv r_3$.
 $2 \equiv r_4$. Fall 2

Simuleringar på process H 1

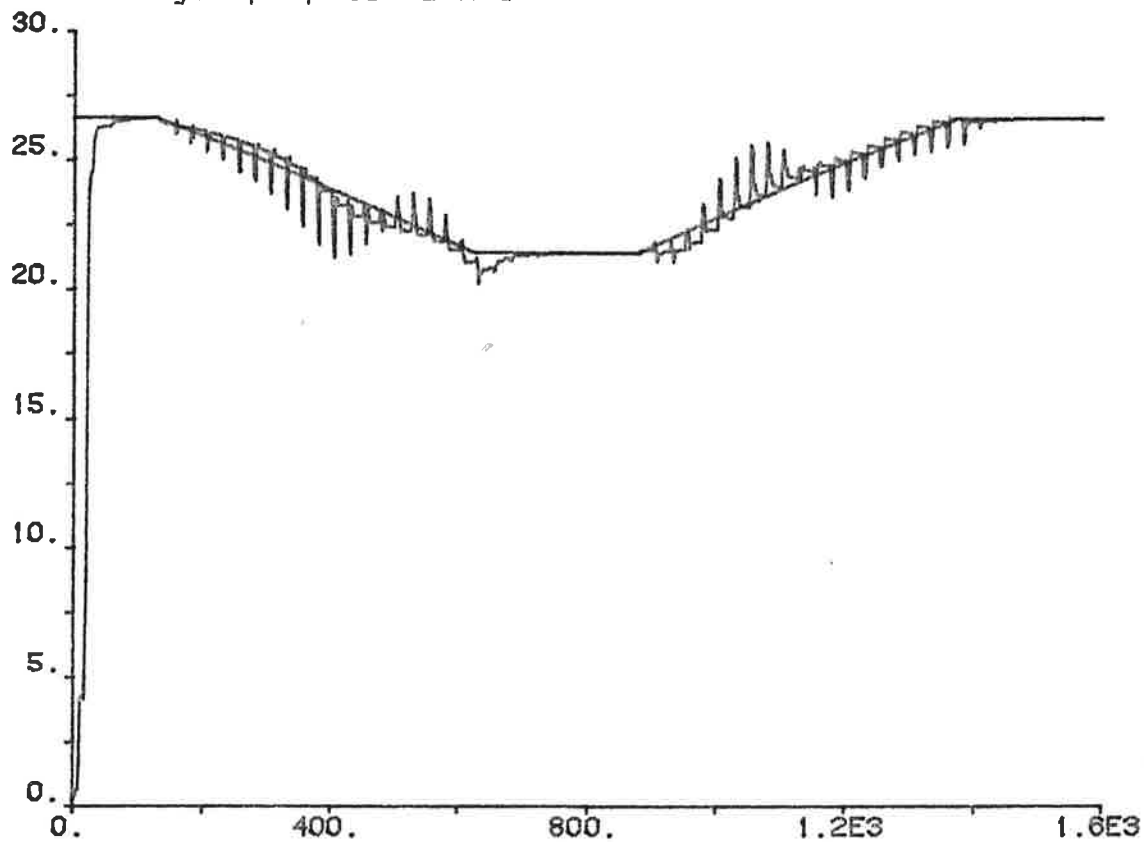


Figur 4.12b.i. 1 = skattat r_3 . 2 = exakt r_2 . 3 = exakt r_3 .

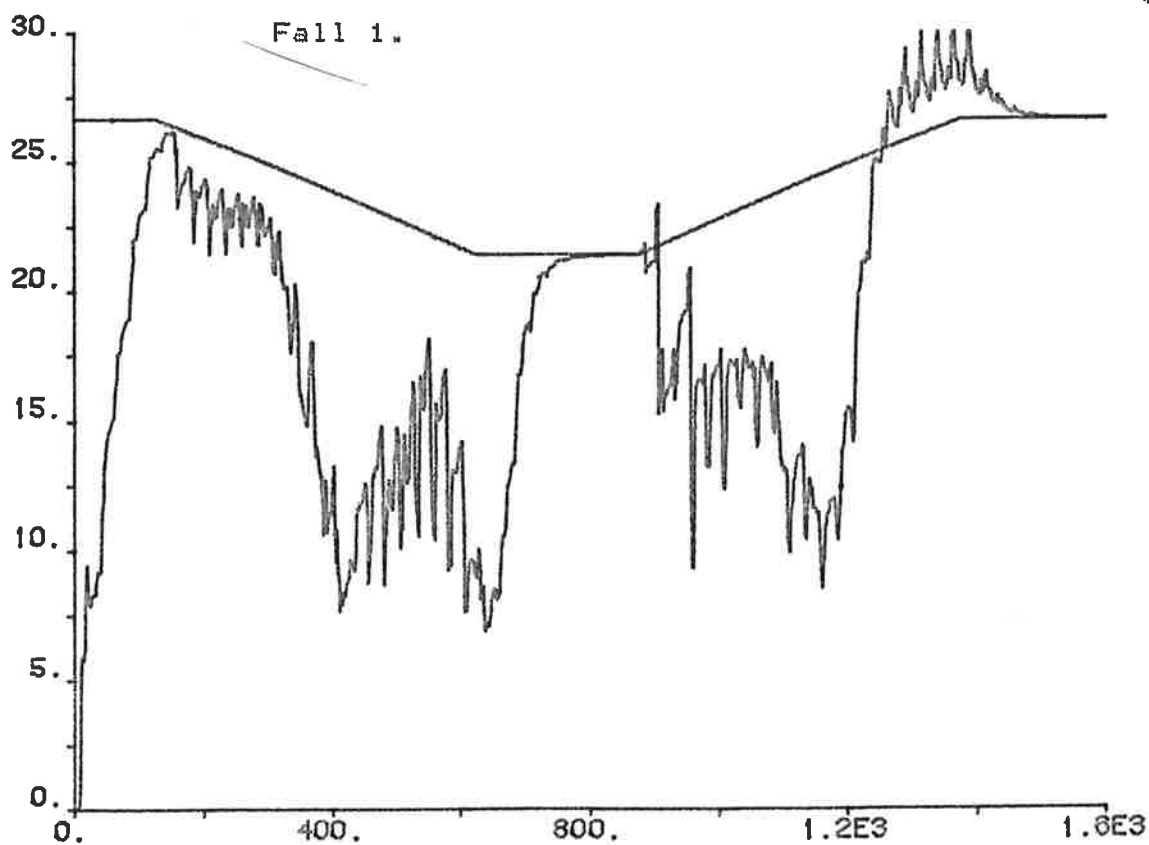


Figur 4.12b.ii. 1 = skattat r_4 . 2 = exakt r_3 . 3 = exakt r_4 .

Simuleringar på process H 1



Figur 4.13a. En av koefficienterna i S-polynomet, s_0 .



Figur 4.13b. En av koefficienterna i S-polynomet, s_0 .

Fall 2.

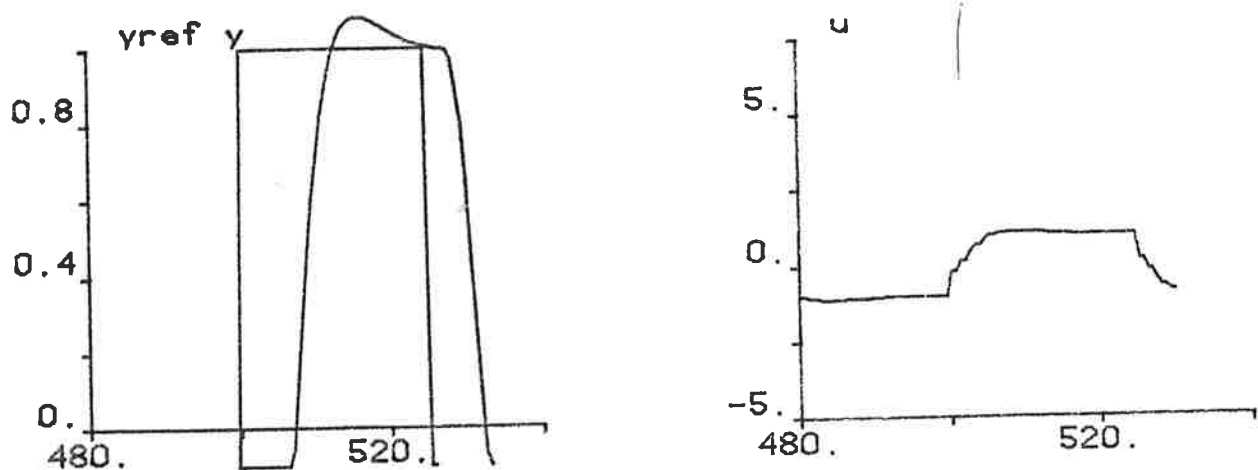
4.3 . Fast och kontinuerligt varierande tidsfördröjning på system av 4:e ordningen

Figurerna 4.14a - e visar på Metod 1:s förmåga att reglera en process av fjärde ordningen med fast tidsfördröjning ($= 2$), då systemet stabiliserat sig. Överslängen svarar mot ett system av andra ordningen med dämpningen $\zeta = 0.707$ då steget är två (från -1.0 till 1.0). När bandbredden för det slutna systemet, ω_0 , ökas, blir systemet snabbare till en viss gräns. Gränsen är här ungefär ett ($\omega_0 \approx 1.0$), dvs då ω_0

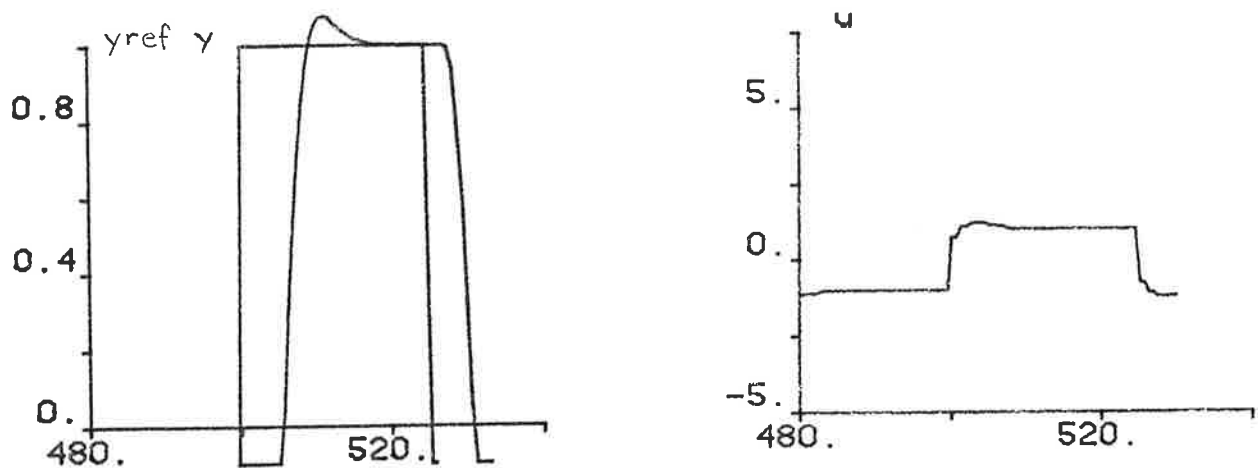
ökas ytterligare (exempelvis figur 4.14e) sker ingen större förändring av utsignalen i fråga om snabbhet. Detta förklaras av att regulatorn har begränsningar på styrsignalen. Eftersom regulatorn då ej tillåts "ta i" kraftigare kan ej heller slutna systemet fås snabbare. Denna "klippning" av styrsignalen orsakar svängningar i excitationen. Metod 2 ger här i stort sett samma resultat ty processutseendet ej tidsberoende. Simuleringarna kan jämföras med 3.11a&b i ref[1]. Skillnaden är att Andell ej haft någon tidsfördröjning i sina simuleringar. De här redovisade simuleringarna visar att Metod 1&2 är bättre på att klara ett fel i antagandet av processordning.

Simulering på fjärde ordningens process med kontinuerligt varierande tidsfördröjning återges i figurer 4.15b&c. Dessa figurer vill visa på att regulatorerna hjälpligt klarar av detta. Det är Metod 1 som återges: Metod 2 visar, som tidigare, sämre resultat. Någon motsvarande simulering finns ej i ref[1].

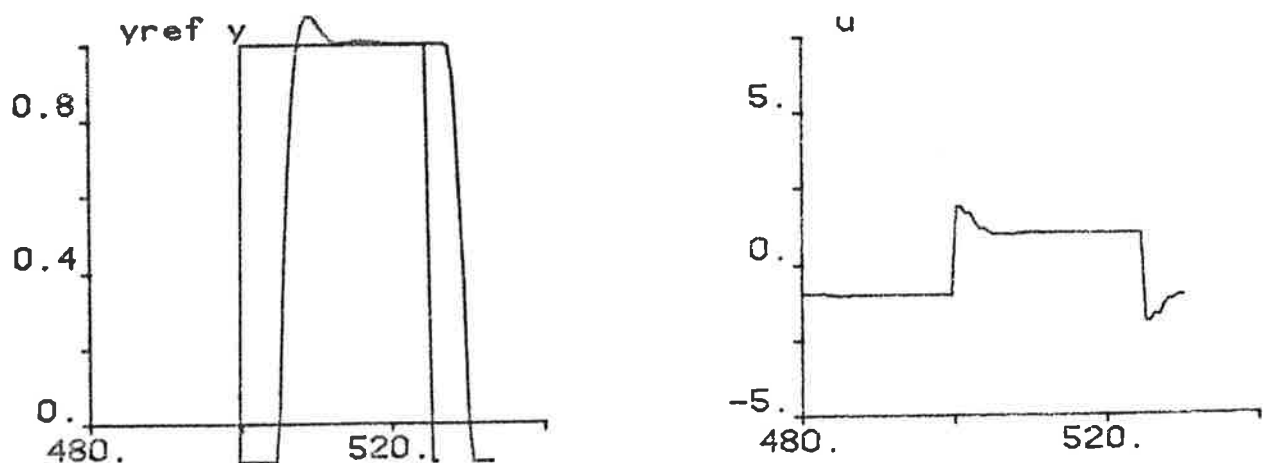
Simuleringar på process H 2



Figur 4.14a. Då systemet ställt in sig. Referenssignal, y_{ref} , utsignal, y och styrsignal, u . Slutna systemets bandbredd, $\omega_0 = 0.35$.

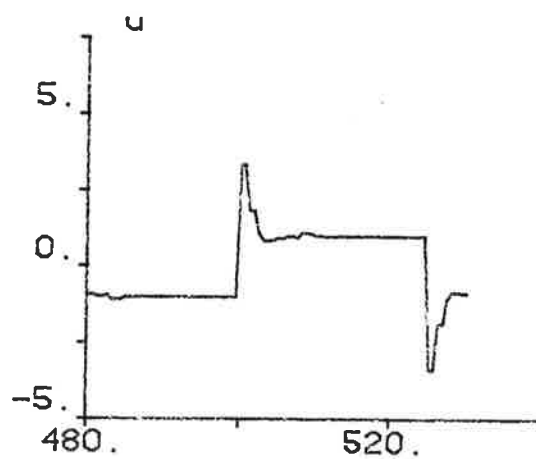
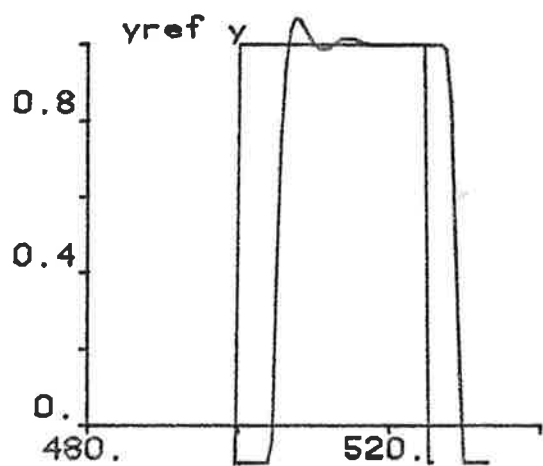


Figur 4.14b. Samma som Figur 4.14a då $\omega_0 = 0.55$.

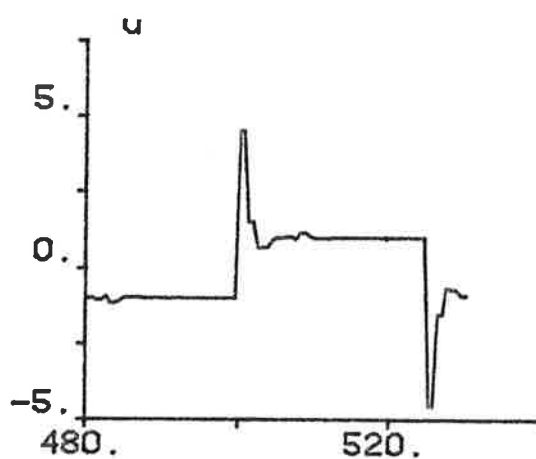
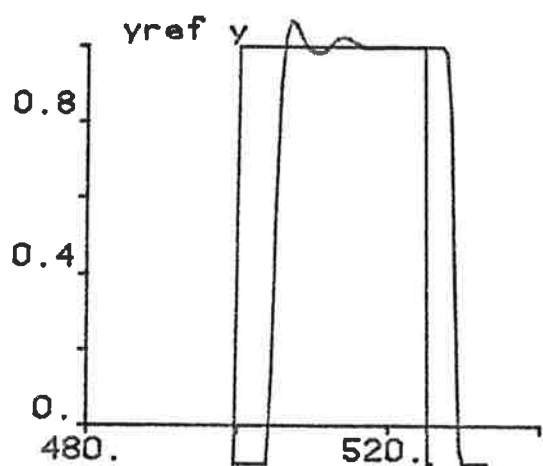


Figur 4.14c. Samma som Figur 4.14a då $\omega_0 = 0.75$.

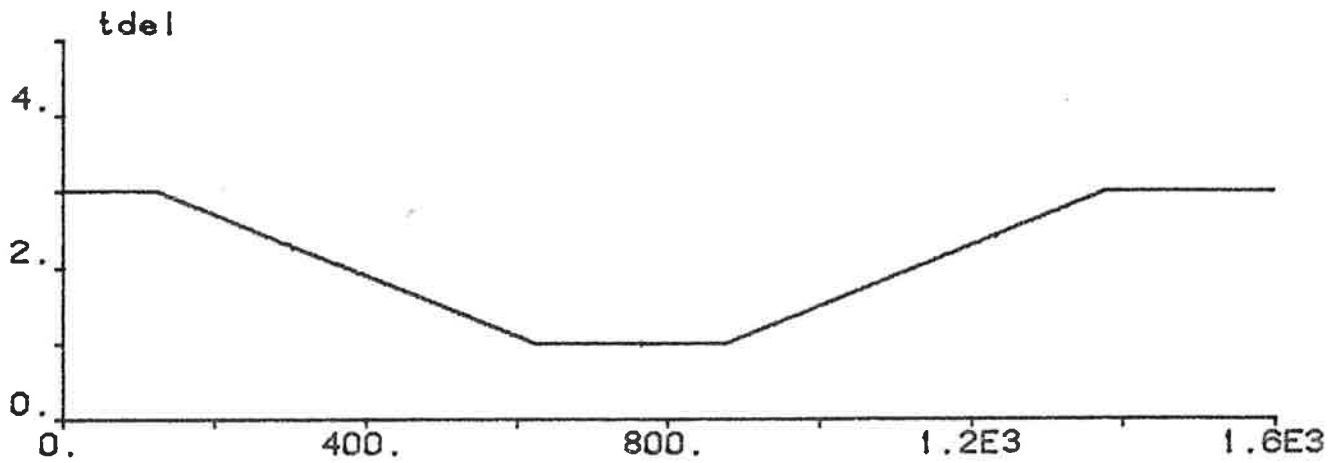
Simuleringar på process H 2



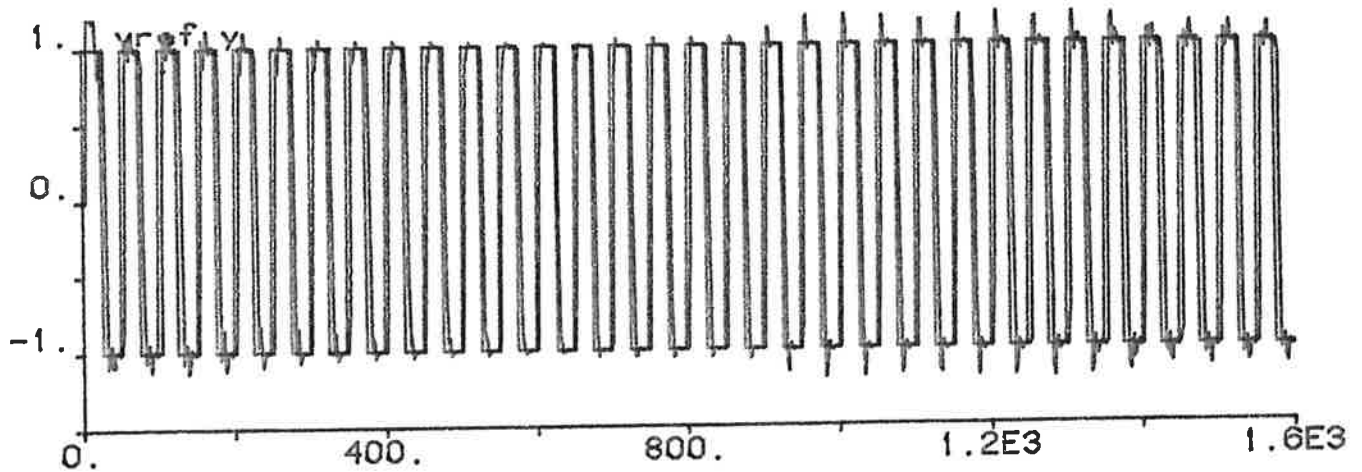
Figur 4.14d. Samma som Figur 4.14a då $\omega_0 = 1.0$.



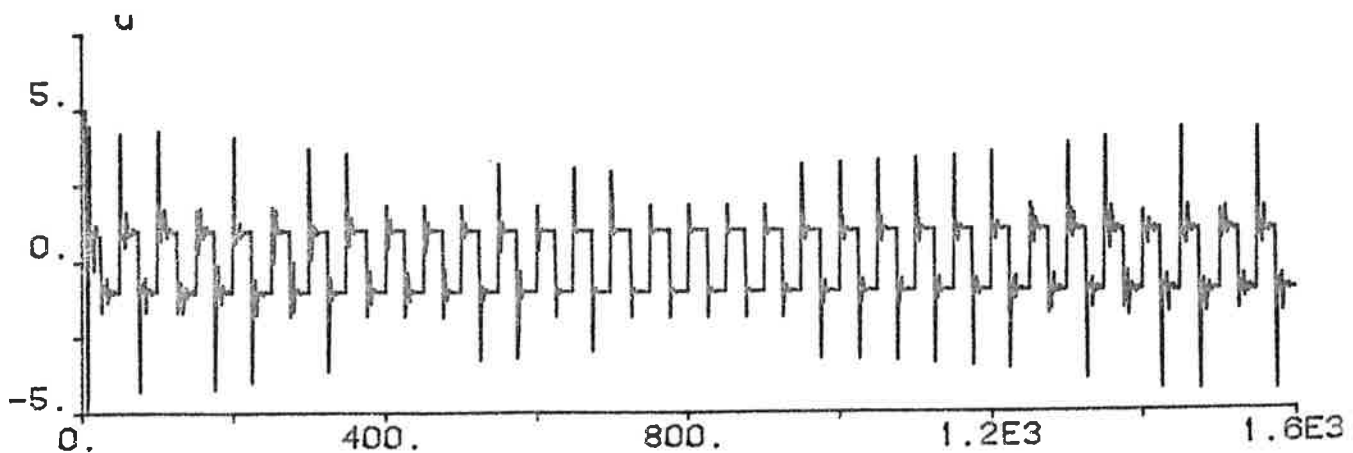
Figur 4.14e. samma som Figur 4.14a då $\omega_0 = 1.20$.



Figur 4.15a. Tidsfördröjningens variation i tiden.



Figur 4.15b. Referens- och ut-signal, y_{ref} och y , då slutna systemets bandbredd, $\omega_0 = 1.0$. Fall 1 på H_2 med $b = 1.0$.



Figur 4.15c. Styrsignal till processen, u .

5. SAMMANFATTNING

Enligt ref[1] är det känt att en realisering av en självinställande PID-regulator för system av andra ordningen med fix tidsfördröjning, är möjlig. Här har visats att, för samma system med variabel tidsfördröjning, en självinställande PID-regulator är realiserbar. Två metoder för att få styrlagsparametrarna har testats: lösning av ekvationssystem och minsta-kvadrat-skattning. Lösning av ekvationssystem är klart att föredra framför tvåstegsskattningen i det att den ger stabilare styrlagskoefficienter. Stabilitet i styrlagskoefficienter ger stabil utsignal. Processpolynomens koefficienter skattas lika bra, ty samma metod används.

På processer med högre ordning än antagen, klarar den bättre av metoderna av att reglera bra. Kraven på det slutna systemets bandbredd får ej vara för höga om rimliga styrsignaler skall genereras.

Referenser

- [1] L-A Andell Självinställande PID-regulatorer för
system med tidsfördröjning.
LUTFD2/(TFRT-5270)/1-087/(1982)
- [2] H Elmqvist Simnon - an interactive simulation
program for nonlinear systems.
Report 7502 April 1975
Dept. of Automatic Control
Lund Institute of Technology
- [3] K J Aström Computer Control Theory.1982
B Wittenmark Dept. of Automatic Control
Lund Institute of Technology
- [4] Guy A Dumont Adaptive Dead-Time Compensation.
Preprints 6th IFAC Symposium on
Identification and System Parameter
Estimation, Washington Juni 1982,
sidorna 397 - 402

Regulatorn vilken löser ekvationssystemet

```

SUBROUTINE SPID1
C
  PARAMETER (MAXNB=10, LOCLIM=MAXNB+4)
  DIMENSION ARG(1:2),EXT(1:2), A(1:LOCLIM,1:LOCLIM),
#       Z(1:LOCLIM),B(1:LOCLIM),IPVT(1:LOCLIM),
#       VNAMP(1:2),WNAMP(1:2),
#       SYSID(1:2),HLPV(1:LOCLIM),
#       PNAM(1:2),DPNAM(1:2),PONAM(1:2)
  INTEGER DMAXU,DEST
  LOGICAL TEST,LSTOP
  DATA ARG/4HNB ,4H /,EXT/4H ,4H /,SYSID/4HSPID,4H1 /,
#       PNAM/4HP ,4H /,DPNAM/4HDP ,4H /,
#       PONAM/4HPO ,4H /
C
  COMMON /DESTIN/IDUM,IPART
  COMMON /TIME/T
  COMMON /USER/LSTOP
  COMMON /DEVICE/LKB,LTP,LLP,LDIS,LTO,LPLOT,LXXX
C
  COMMON /NAME/YREF,Y,U,F(1:LOCLIM),TH(1:LOCLIM),
#P(1:LOCLIM,1:LOCLIM) ,DF(1:LOCLIM),DTH(1:LOCLIM),
#DP(1:LOCLIM,1:LOCLIM),FO(1:LOCLIM),THO(1:LOCLIM),
#PO(1:LOCLIM,1:LOCLIM),
#ZZ,W,DT,ULOW,UHIG,R(1:MAXNB+1),AM1,AM2,EPS,EPS2,
#SO,S1,S2,TS,XLAM,NB,IERR
C
  GOTO (1,2,3,4,5,6,7,8),IPART
C
  IDENTIFIERING AV SYSTEMET
C
  1  CALL IDENT(4HDISC,SYSID)
  RETURN
C
  DEKLARATION AV VARIABLER
C
  2  CALL FINT(ARG,EXT,NB,IERR)
  IF((IERR .EQ. 0) .AND. (NB .GE. 0) .AND. (NB .LE. MAXNB)) GOTO 200
  CALL BADVAL(ARG,EXT)
  LSTOP=.TRUE.
  RETURN
C
  200  OM NB FELAKTIG FORTSATTER EJ SIMULERINGEN
  CONTINUE
  CALL INPUT(YREF,4HYREF)
  CALL INPUT(Y,4HY )
C
  CALL OUTPUT(U,4HU )
C
  CALL STATEV(F,NB+4,4HF )

```


Appendix A

```

CALL STATEV(TH,NB+4,4HTH )
C
DO 210 I=1,NB+4
CALL CRENAM(PNAM,I,VNAMP)
IP=LENGTH(VNAMP,8)
CALL PAC(IP+1,VNAMP,4H )
DO 205 J=1,NB+4
CALL CRENAM(VNAMP,J,WNAMP)
CALL STATE2(P(I,J),WNAMP)
205 CONTINUE
210 CONTINUE
C
C
CALL NEWV(DF,NB+4,4HDF )
CALL NEWV(DTH,NB+4,4HDTH )
C
DO 215 I=1,NB+4
CALL CRENAM(DPNAM,I,VNAMP)
IP=LENGTH(VNAMP,8)
CALL PAC(IP+1,VNAMP,4H )
DO 213 J=1,NB+4
CALL CRENAM(VNAMP,J,WNAMP)
CALL NEW2 (DP(I,J),WNAMP)
213 CONTINUE
215 CONTINUE
C
C
CALL INITV(FO,NB+4,4HFO )
CALL INITV(THO,NB+4,4HTHO )
C
DO 220 I=1,NB+4
CALL CRENAM(PONAM,I,VNAMP)
IP=LENGTH(VNAMP,8)
CALL PAC(IP+1,VNAMP,4H )
DO 217 J=1,NB+4
CALL CRENAM(VNAMP,J,WNAMP)
CALL INIT2 (PO(I,J),WNAMP)
217 CONTINUE
220 CONTINUE
C
C
CALL PAR(EPS,4HEPS )
CALL PAR(ZZ,4HZZ )
CALL PAR(W,4HW )
CALL PAR(DT,4HDT )
CALL PAR(ULOW,4HULOW)
CALL PAR(UHIG,4HUHIG)
CALL PAR(XLAM,4HXLAM)
C
CALL VAR(EPS2,4HEPS2)
CALL VARV(R,NB+1,4HR )
CALL VAR(AM1,4HAM1 )
CALL VAR(AM2,4HAM2 )
CALL VAR(SO,4HSO )
CALL VAR(S1,4HS1 )

```

```

        CALL VAR(S2,4HS2 )
        CALL TSAMP(TS,4HTS )
        RETURN
C
C   INITIALVARDEN
C
C   3   CONTINUE
C
        DO 310 I=1,NB+4
        PO(I,I)=1000
        TH(I)=0.1
310    CONTINUE
C
        EPS=0.0
        ZZ=0.707
        W=0.9
        DT=1
        ULOW=-5
        UHIG=5
        XLAM=0.96
        RETURN
C
C   UTRAKNING AV INITIALVARDEN
C
C   4   CONTINUE
C
        ONSKADE KOEFF. I DET SLUTNA SYSTEMET
C
        AM1=-2*EXP(-ZZ*W*DT)*COS(W*DT*SQRT(1-ZZ*ZZ))
        AM2=EXP(-2*ZZ*W*DT)
        RETURN
C
C   OUTPUT-SECTION
C
C   UTRAKNING AV MATRISEN A INNEHÅLLANDE A1PRIM,...,B1,...,Bnb
C
C   5   CONTINUE
        LSIZEZ=NB+4
        LSIZER=NB+1
C
        A1PRIM=TH(1)-1
        A2PRIM=TH(2)-TH(1)
        A3PRIM=-TH(2)
C
        DO 500 I=1,NB+4
        DO 500 II=1,NB+4
        A(I,II)=0.0
500    CONTINUE
C
C
        DO 505 II=1,LSIZER
        A(II,II)=1.0
        A(II+1,II)=A1PRIM
        A(II+2,II)=A2PRIM
        A(II+3,II)=A3PRIM

```

Appendix A

```

505 CONTINUE
C
NNB=LSIZER+1
DO 507 II=0,2
DO 507 I=1,NNB
A(I+II,II+NNB)=TH(I+2)
507 CONTINUE
C
C
C EVALUERA Ax=b
C
B(1)=AM1-A1PRIM
B(2)=AM2-A2PRIM
B(3)=-A3PRIM
DO 510 I=4,LSIZET
B(I)=0.0
510 CONTINUE
C
CALL DECOM(A,A,IPVT,X,LSIZET,EPS,IERR,LOCLIM,LOCLIM,Z)
IF (IERR .NE. 0) GO TO 525
C OM MATRISEN HAR GEMENSAMMA EGENVARDE KAN INVERSEN
C EJ BERÄKNAS: INGEN UPDATERING GÖRS AV R OCH S
CALL SOLVE(A,IPVT,B,Z,LSIZET,1,LOCLIM,LOCLIM,LOCLIM)
C
C Z INNEHÄLLER STYRLAGARNA R1,...,Rnb+1,S0,S1,S2
C
DO 520 I=1,LSIZER
R(I)=Z(I)
520 CONTINUE
C
S0=Z(LSIZER+1)
S1=Z(LSIZER+2)
S2=Z(LSIZER+3)
C
525 CONTINUE
C
TP=0.0
DO 530 I=3,LSIZET
TP=TP+TH(I)
530 CONTINUE
IF (ABS(TP) .LT. 1E-4) THEN
TP=1E4*(1+AM1+AM2)
ELSE
TP=(1+AM1+AM2)/TP
ENDIF
C STATISKA FÖRSTÄRKNINGEN TP
CLAWU=(1.0-R(1))*F(3)
DO 540 I=1,LSIZER-1
CLAWU=CLAWU+(R(I)-R(I+1))*F(I+3)
540 CONTINUE
CLAWU=CLAWU+R(LSIZER)*F(LSIZET)
C GAMLA UTSIGNALERS INVERKAN CLAWU
CLAWY=-S0*Y+S1*F(1)+S2*F(2)
C GAMLA INSIGNALERS INVERKAN CLAWY
U1=TP*YREF+CLAWU+CLAWY

```

```
C
C
  IF (U1.LT.ULOW) THEN
    U=ULOW
  ELSEIF (U1.LT.UHIG) THEN
    U=U1
  ELSE
    U=UHIG
  ENDIF
C
  RETURN
C
C  DYNAMICS
C
C  SKATTNING AV A-OCH B-POLYNOMENS KOEFF MED M.K.-METODEN
C  HJÄLPVARIABLER: HLPS1=XLAM+F*P*F (SKALAR)
C                   HLPS2=F*TH (SKALAR)
C                   HLPV =P*F (VEKTOR)
C
  6  HLPS1=0.0
    HLPS2=0.0
    DO 600 I=1,NB+4
      HLPS2=HLPS2+F(I)*TH(I)
      HLPV(I)=0.0
    600 CONTINUE
    E=Y-HLPS2
C
    DO 602 I=1,NB+4
      DO 601 II=1,NB+4
        HLPV(I)=HLPV(I)+ P(I,II)*F(II)
      601 CONTINUE
      HLPS1=HLPS1+F(I)*HLPV(I)
    602 CONTINUE
    HLPS1=HLPS1+XLAM
C
    DO 603 I=1,NB+4
      DTH(I)=TH(I)+(HLPV(I)*E/HLPS1)
    603 CONTINUE
C
    DO 604 I=1,NB+4
      DO 604 II=1,I
        X=(P(I,II)-HLPV(I)*HLPV(II)/HLPS1)/XLAM
        DP(I,II)=X
        DP(II,I)=X
    604 CONTINUE
C
C  UPPDATERING AV Y-OCH U-TILLST.
C
  DF(1)=-Y
  DF(2)=F(1)
  DF(3)=U
  DO 608 I=4,NB+4
    608 DF(I)=F(I-1)
C
```

```
C   UPDAT. AV SAMPLINGSINTERVALLET
C
C   TS=T+DT
C   RETURN
C
7   RETURN
8   RETURN
   END
```

Regulatorn med tvåstegsskattning

SUBROUTINE SPID2

```

C
PARAMETER (MAXNB=10, LOCLIM=MAXNB+4)
DIMENSION ARG(1:2),EXT(1:2), A(1:LOCLIM,1:LOCLIM),
#      Z(1:LOCLIM),B(1:LOCLIM),IPVT(1:LOCLIM),
#      VNAMP(1:2),WNAMP(1:2),
#      SYSID(1:2),HLPV(1:LOCLIM),
#      PNAM(1:2),DPNAM(1:2),PONAM(1:2),
#      CTNAM(1:2),CTONAM(1:2),DCTNAM(1:2),
#      CFNAM(1:2),CFONAM(1:2),DCFNAM(1:2),
#      CPNAM(1:2),CPONAM(1:2),DCPNAM(1:2),
#      DUNAM(1:2),DUONAM(1:2),DDUNAM(1:2),
#      YRNAM(1:2),YRONAM(1:2),DYRNAM(1:2)
LOGICAL TEST,LSTOP
DATA ARG/4HNB ,4H /,EXT/4H ,4H /,SYSID/4HSPID,4H2 /,
#      PNAM/4HP ,4H /,DPNAM/4HDP ,4H /,
#      PONAM/4HPO ,4H /,
#      CTNAM/4HCT ,4H /,CTONAM/4HCTO ,4H /,
#      DCTNAM/4HDCT ,4H /,CFNAM/4HCF ,4H /,
#      CFONAM/4HCFO ,4H /,DCFNAM/4HDCF ,4H /,
#      CPNAM/4HCP ,4H /,CPONAM/4HCPO ,4H /,
#      DCPNAM/4HDCP ,4H /,DUNAM/4HOLDU,4H /,
#      DUONAM/4HOLDU,4HO /,DDUNAM/4HDOLD,4HU /,
#      YRNAM/4HYRS ,4H /,YRONAM/4HYRSO,4H /,
#      DYRNAM/4HDYRS,4H /
C
COMMON /DESTIN/IDUM,IPART
COMMON /TIME/T
COMMON /USER/LSTOP
COMMON /DEVICE/LKB,LTP,LLP,LDIS,LTO,LPLOT,LXXX
C
COMMON /NAME/YREF,Y,U,F(1:LOCLIM),TH(1:LOCLIM),
#P(1:LOCLIM,1:LOCLIM),DF(1:LOCLIM),DTH(1:LOCLIM),
#DP(1:LOCLIM,1:LOCLIM),FO(1:LOCLIM),THO(1:LOCLIM),
#PO(1:LOCLIM,1:LOCLIM),
#ZZ,W,DT,ULOW,UHIG,R(1:MAXNB+1),AM1,AM2,E1,E2,
#SO,S1,S2,TS,XLAM,NB,IERR,EPS,
#CF(1:LOCLIM),CT(1:LOCLIM),CP(1:LOCLIM,1:LOCLIM),
#DCF(1:LOCLIM),DCT(1:LOCLIM),DCP(1:LOCLIM,1:LOCLIM),
#CFO(1:LOCLIM),CTO(1:LOCLIM),CPO(1:LOCLIM,1:LOCLIM),
#OLDU(1:MAXNB+3),OLDUO(1:MAXNB+2),DOLDU(1:MAXNB+2),
#YRS(1:MAXNB+2),YRSO(1:MAXNB+2),DYRS(1:MAXNB+2),
#CLAM
C
GOTO (1,2,3,4,5,6,7,8),IPART
C
IDENTIFIERING AV SYSTEMET
C
1 CALL IDENT2(4HDISC,SYSID)

```

Appendix B

```

RETURN
C
C   DEKLARATION AV VARIABLER
C
2   CALL FINT(ARG,EXT,NB,IERR)
   IF((IERR .EQ. 0) .AND. (NB .GE. 0) .AND. (NB .LE. MAXNB)) GOTO 200
   CALL BADVAL(ARG,EXT)
   LSTOP=.TRUE.
   RETURN
C
C           OM NB FELAKTIG FORTSATTER EJ SIMULERINGEN
200  CONTINUE
    CALL INPUT(YREF,4HYREF)
    CALL INPUT(Y,4HY )
C
    CALL OUTPUT(U,4HU )
C
C
    CALL STATEV(F,NB+4,4HF )
    CALL STATEV(TH,NB+4,4HTH )
    CALL STATV2(CF,NB+4,CFNAM)
    CALL STATV2(CT,NB+4,CTNAM)
    CALL STATV2(OLDU,NB+3,DUNAM)
    CALL STATV2(YRS,NB+2,YRNAM)
C
    DO 210 I=1,NB+4
    CALL CRENAM(PNAM,I,VNAMP)
    IP=LENGTH(VNAMP,8)
    CALL PAC(IP+1,VNAMP,4H )
    DO 205 J=1,I
    CALL CRENAM(VNAMP,J,WNAMP)
    CALL STATE2(P(I,J),WNAMP)
205  CONTINUE
210  CONTINUE
C
    DO 212 I=1,NB+4
    CALL CRENAM(CPNAM,I,VNAMP)
    IP=LENGTH(VNAMP,8)
    CALL PAC(IP+1,VNAMP,4H )
    DO 211 J=1,I
    CALL CRENAM(VNAMP,J,WNAMP)
    CALL STATE2(CP(I,J),WNAMP)
211  CONTINUE
212  CONTINUE
C
C
    CALL NEWV(DF,NB+4,4HDF )
    CALL NEWV(DTH,NB+4,4HDTH )
    CALL NEWV2(DCF,NB+4,DCFNAM)
    CALL NEWV2(DCT,NB+4,DCTNAM)
    CALL NEWV2(DOLDU,NB+3,DDUNAM)
    CALL NEWV2(DYRS,NB+2,DYRNAM)
C
    DO 214 I=1,NB+4
    CALL CRENAM(DPNAM,I,VNAMP)
    IP=LENGTH(VNAMP,8)

```

```

        CALL PAC(IP+1,VNAMP,4H )
        DO 213 J=1,I
        CALL CRENAM(VNAMP,J,WNAMP)
        CALL NEW2 (DP(I,J),WNAMP)
213  CONTINUE
214  CONTINUE
C
        DO 217 I=1,NB+4
        CALL CRENAM(DCPNAM,I,VNAMP)
        IP=LENGTH(VNAMP,8)
        CALL PAC(IP+1,VNAMP,4H )
        DO 216 J=1,I
        CALL CRENAM(VNAMP,J,WNAMP)
        CALL NEW2(DCP(I,J),WNAMP)
216  CONTINUE
217  CONTINUE
C
C
        CALL INITV(F0,NB+4,4HF0 )
        CALL INITV(TH0,NB+4,4HTHO )
        CALL INITV2(CF0,NB+4,CFONAM)
        CALL INITV2(CT0,NB+4,CTONAM)
        CALL INITV2(OLDUO,NB+3,DUONAM)
        CALL INITV2(YRSO,NB+2,YRONAM)
C
        DO 220 I=1,NB+4
        CALL CRENAM(PONAM,I,VNAMP)
        IP=LENGTH(VNAMP,8)
        CALL PAC(IP+1,VNAMP,4H )
        DO 219 J=1,I
        CALL CRENAM(VNAMP,J,WNAMP)
        CALL INIT2 (PO(I,J),WNAMP)
219  CONTINUE
220  CONTINUE
C
        DO 230 I=1,NB+4
        CALL CRENAM(CPONAM,I,VNAMP)
        IP=LENGTH(VNAMP,8)
        CALL PAC(IP+1,VNAMP,4H )
        DO 225 J=1,I
        CALL CRENAM(VNAMP,J,WNAMP)
        CALL INIT2(CPO(I,J),WNAMP)
225  CONTINUE
230  CONTINUE
C
        CALL PAR(ZZ,4HZZ )
        CALL PAR(W,4HW )
        CALL PAR(DT,4HDT )
        CALL PAR(ULOW,4HULOW)
        CALL PAR(UHIG,4HUHIG)
        CALL PAR(XLAM,4HXLAM)
        CALL PAR(AM1,4HAM1 )
        CALL PAR(AM2,4HAM2 )
        CALL PAR(CLAM,4HCLAM)

```



```
      CALL PAR(EPS,4HEPS )
C
      CALL VARV(R,NB+1,4HR  )
      CALL VAR(E1,4HE1  )
      CALL VAR(E2,4HE2  )
      CALL VAR(S0,4HS0  )
      CALL VAR(S1,4HS1  )
      CALL VAR(S2,4HS2  )
      CALL TSAMP(TS,4HTS  )
      RETURN
C
C      INITIALVARDEN
C
C      3      CONTINUE
C
      DO 310 I=1,NB+4
      PO(I,I)=1000
      CPO(I,I)=1000
      THO(I)=0.1
      CTO(I)=0.1
310    CONTINUE
C
      ZZ=0.707
      W=0.5
      DT=1
      ULOW=-5
      UHIG=5
      XLAM=0.96
      CLAM=0.94
      RETURN
C
C      UTRAKNING AV INITIALVARDEN
C
C      4      CONTINUE
C      ÖNSKADE KOEFF. I DET SLUTNA SYSTEMET
C
      AM1=-2*EXP(-ZZ*W*DT)*COS(W*DT*SQRT(1-ZZ*ZZ))
      AM2=EXP(-2*ZZ*W*DT)
      RETURN
C
C      OUTPUT-SECTION
C
C      ANVÄNDNING AV DEN M.K.-SKATTADE STYRLAGEN R OCH S
C
C      5      CONTINUE
      DO 510 I=1,NB+1
      R(I)=CT(I)
510    CONTINUE
C
      S0=CT(NB+2)
      S1=CT(NB+3)
      S2=CT(NB+4)
C
C      525    CONTINUE
C
```



```

C
  DO 603 I=1,NB+4
  DO 602 II=1,NB+4
  HLPV(I)=HLPV(I)+ P(I,II)*F(II)
602 CONTINUE
  HLPS1=HLPS1+F(I)*HLPV(I)
603 CONTINUE
  HLPS1=HLPS1+XLAM
C
  DO 604 I=1,NB+4
  DTH(I)=TH(I)+(HLPV(I)*E1/HLPS1)
604 CONTINUE
C
  DO 605 I=1,NB+4
  DO 605 II=1,I
  DP(I,II)=(P(I,II)-HLPV(I)*HLPV(II)/HLPS1)/XLAM
605 CONTINUE
C
C
C   UPDATERING AV Y-OCH U-TILLST.
C
  DF(1)=-Y
  DF(2)=F(1)
  DF(3)=U
  DO 608 I=4,NB+4
608 DF(I)=F(I-1)
C
C
C   R- OCH S-POLYNOMENS SKATTNING
C
  DELUF=TH(3)*OLDU(1)
  DO 615 I=2,NB+2
  DELUF=DELUF+(TH(I+2)-TH(I+1))*OLDU(I)
615 CONTINUE
  DELUF=DELUF-TH(NB+4)*OLDU(NB+3)
C
  HLPS1=0.0
  HLPS2=0.0
  DO 620 I=1,NB+4
  HLPS2=HLPS2+CF(I)*CT(I)
  HLPV(I)=0.0
620 CONTINUE
  E2=Y+AM1*YRS(1)+AM2*YRS(2)-DELUF-HLPS2
C
  DO 627 I=1,NB+4
  DO 625 II=1,NB+4
  HLPV(I)=HLPV(I)+CP(I,II)*CF(II)
625 CONTINUE
  HLPS1=HLPS1+CF(I)*HLPV(I)
627 CONTINUE
  HLPS1=HLPS1+CLAM
C
  DO 635 I=1,NB+4
  DCT(I)=CT(I)+(HLPV(I)*E2/HLPS1)

```

Appendix B

```
      DO 630 II=1,I
      DCP(I,II)=(CP(I,II)-HLPV(I)*HLPV(II)/HLPS1)/CLAM
630   CONTINUE
635   CONTINUE
C
      DO 640 I=2,NB+2
      DOLDU(I)=OLDU(I-1)
      DYRS(I)=YRS(I-1)
640   CONTINUE
      DOLDU(1)=U
      DOLDU(NB+3)=OLDU(NB+2)
      DYRS(1)=Y
C
      DO 645 I=2,NB+4
      DCF(I)=CF(I-1)
645   CONTINUE
      DCF(1)=DELUF
      YPRIMF=TH(3)*Y
      DO 650 I=1,NB+1
      YPRIMF=YPRIMF+TH(3+I)*YRS(I)
650   CONTINUE
      DCF(NB+2)=YPRIMF
C
C   UPPDAT. AV SAMPLINGSINTERVALLET
C
      TS=T+DT
C
      RETURN
C
7    RETURN
8    RETURN
      END
```

Systemet_av_andra_ordningen

```
continuous system lars
```

```
input u
output y
state x1 x2
der dx1 dx2
```

```
dx1=x2
dx2=b*b*(u-x1)-2*b*x2
```

```
y=x1
```

```
b:0.3
end
```

Systemet_av_fjärde_ordningen

```
continuous system sarl
```

```
time t
input u
output y
state x1 x2 x3 x4
der dx1 dx2 dx3 dx4
```

```
dx1=x2
dx2=x3
dx3=x4
dx4=b*b*b*b*(u-x1)-4*b*b*b*x2-6*b*b*x3-4*b*x4
```

```
y=x1
```

```
b:1
end
```

Uträkning av exakta koefficienter.

Den kontinuerliga tillståndsformen

$$\dot{x} = Ax + Bu \quad y = Cx$$

med

$$A = \begin{bmatrix} -b & 1 \\ 0 & -b \end{bmatrix} \quad B = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad C = [1 \quad -1] b^2$$

överförs till

$$qx = \Phi x + [\Gamma_0 q^{1-d} + \Gamma_1 q^{-d}] u$$

med

$$\Phi = e^{Ah} = \begin{bmatrix} e^{-bh} & he^{-bh} \\ 0 & e^{-bh} \end{bmatrix} = \begin{bmatrix} \alpha & \beta \\ 0 & \alpha \end{bmatrix}$$

$$\Gamma_0 = \int_0^{h-\tau} e^{As} ds B = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$

$$\Gamma_1 = e^{A(h-\tau)} \int_0^{\tau} e^{As} ds B = \begin{bmatrix} y_3 \\ y_4 \end{bmatrix}$$

$$t_{\text{del}} = (d-1)h + \tau$$

h samplingsintervall

t_{del} tidsfördröjningen

d heltal

0 ≤ τ < h

$$\int_0^t e^{As} ds B = \begin{bmatrix} f(t) + \{f(t) - g(t)\}/b \\ f(t) \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$

$$f(t) = \frac{(1 - e^{-bt})}{b}$$

$$g(t) = te^{-bt}$$

$$e^{A(h-\tau)} = \begin{bmatrix} e^{-b(h-\tau)} & (h-\tau)e^{-b(h-\tau)} \\ 0 & e^{-b(h-\tau)} \end{bmatrix} = \begin{bmatrix} x_1 & x_2 \\ 0 & x_1 \end{bmatrix}$$

Följande beräkning leder till överföringsfunktionen:

$$(qI - \Phi)x = \begin{bmatrix} \Gamma_0 q + \Gamma_1 \\ 0 \end{bmatrix} q^{-d} u$$

$$y = Cx = C(\Gamma_0 q + \Gamma_1) q^{-d} u$$

$$H = \frac{y}{u} = \frac{b^2}{(1 - \alpha q^{-1})} \left\{ \begin{aligned} & [\gamma_1 - \gamma_3] + \\ & + [\gamma_2 - \alpha\gamma_1 + (\alpha + \beta)\gamma_3 - \gamma_4] q^{-1} + \\ & + [(\alpha + \beta)\gamma_4 - \alpha\gamma_2] q^{-2} \end{aligned} \right\} q^{-d}$$

Detta ger B-polynomet följande utseende:

$$B(q^{-1}) = [b_i + b_{ii} q^{-1} + b_{iii} q^{-2}] q^{-d}$$

$$b_i = [\gamma_1 - \gamma_3] b^2$$

$$b_{ii} = [\gamma_2 - \alpha\gamma_1 + (\alpha + \beta)\gamma_3 - \gamma_4] b^2$$

$$b_{iii} = [(\alpha + \beta)\gamma_4 - \alpha\gamma_2] b^2$$

$$\begin{aligned}
 \alpha &= e^{-bh} & \beta &= he^{-bh} \\
 x_1 &= 1 - bf(h-\tau) & x_2 &= g(h-\tau) \\
 y_1 &= f(\tau) + [f(\tau) - g(\tau)]/b & y_2 &= f(\tau) \\
 \\
 Y_1 &= f(h-\tau) + [f(h-\tau) - g(h-\tau)]/b \\
 Y_2 &= x_1 y_1 + x_2 y_2 \\
 Y_3 &= f(h-\tau) \\
 Y_4 &= x_1 y_2
 \end{aligned}$$

Realiseringen av denna beräkning i Simnon är listad på nästa sida. För beräkning av regulatorkoefficienterna används samma rutiner som i Metod 1, Decom och Solve, i ett Fortransystem.

Appendix D

```

continuous system proco
time t
input tdel
output b1 b2 b3 b4 b5 b6 a1 a2
d=int(tdel/h)+1
tau=tdel-(d-1)*h
fh=(1-exp(-b*h))/b
gh=h*exp(-b*h)
fht=(1-exp(-b*(h-tau)))/b
ght=(h-tau)*exp(-b*(h-tau))
ft=(1-exp(-b*tau))/b
gt=tau*exp(-b*tau)
x1=1-b*fht
x2=ght
y1=ft+(ft-gt)/b
y2=ft
gam1=fht+(fht-ght)/b
gam2=x1*y1+x2*y2
gam3=fht
gam4=x1*y2
alfa=exp(-b*h)
beta=h*exp(-b*h)
bi=(gam1-gam3)*b*b
bii=(gam2-alfa*gam1+(alfa+beta)*gam3-gam4)*b*b
biii=(gam4*(alfa+beta)-alfa*gam2)*b*b
b1=if d<1.5 then bi else 0
b2=if d<1.5 then bii else if d<2.5 then bi else 0
b3=if d<2.5 then bii else if d<3.5 then bi else 0
xb4=if d<2.5 then biii else if d<3.5 then bii else 0
b4=if d>3.5 then bi else xb4
b5=if d>2.5 then biii else if d>3.5 then bii else 0
b6=if d>3.5 then biii else 0
a1=-2*alfa
a2=alfa*alfa
h=1
b=0.3
end

```

Appendix D

```

SUBROUTINE SOLV

DIMENSION X(1:8,1:8),Z(1:8),IPVT(1:8),Y(1:8)
COMMON/DESTIN/IDUM,IPART
COMMON/TIME/T
COMMON/USER/LSTOP
COMMON/NAME/B(1:6),A(1:2),R(1:5),ZZ,W,H,TS,SO,S1,S2,AM1,AM2

GOTO(1,2,3,4,5,6,7,8),IPART

1 CALL IDENT(4HCONT,4HSOLV)
  RETURN

2 CALL INPUTV(B,6,4HB )
  CALL INPUTV(A,2,4HA )
  CALL OUTPUTV(R,5,4HR )
  CALL OUTPUT(S2,4HS2 )
  CALL OUTPUT(S1,4HS1 )
  CALL OUTPUT(SO,4HSO )
  CALL PAR(ZZ,4HZZ )
  CALL PAR(H,4HH )
  CALL PAR(W,4HW )
  CALL VAR(AM1,4HAM1 )
  CALL VAR(AM2,4HAM2 )
  RETURN

3 CONTINUE
  ZZ=0.707107
  W=0.5
  H=1.0
  RETURN

4 CONTINUE
  AM1=-2*EXP(-ZZ*W*H)*COS(W*H*SQRT(1-ZZ*ZZ))
  AM2=EXP(-2*ZZ*W*H)
  RETURN

5 CONTINUE
  A1PRIM=A(1)-1
  A2PRIM=A(2)-A(1)
  A3PRIM=-A(2)

  DO 505 I=1,8
  DO 504 II=1,8
  X(I,II)=0
504 CONTINUE
  Y(I)=0
  Z(I)=0
  IPVT(I)=0
505 CONTINUE

  DO 510 II=1,5
  X(II,II)=1

```

```
X(II+1,II)=A1PRIM
X(II+2,II)=A2PRIM
X(II+3,II)=A3PRIM
510 CONTINUE

DO 520 II=0,2
DO 520 I=1,6
X(I+II,II+6)=B(I)
520 CONTINUE

Y(1)=AM1-A1PRIM
Y(2)=AM2-A2PRIM
Y(3)=-A3PRIM

CALL DECOM(X,X,IPVT,Q,8,0,IERR,8,8,Z)
IF (IERR.NE.0) GOTO 599
CALL SOLVE(X,IPVT,Y,Z,8,1,8,8,8)

DO 530 I=1,5
530 R(I)=Z(I)
S0=Z(6)
S1=Z(7)
S2=Z(8)
599 CONTINUE
RETURN
6 RETURN
7 RETURN
8 RETURN
END
```