

CODEN: LUTFD2/(TFRT-5341)/1-51/(1985)

System med svagt dämpade poler

Peter Marbe
Håkan Möller

Institutionen för Reglerteknik
Lunds Tekniska Högskola
December 1985

EXAMENSARBETE

SYSTEM MED
SVAGT DÄMPADE POLER

utfört av:
Peter Marbe 601017-3950 MBI
och
Håkan Möller 600830-4336 MBI

godkänt certifikat 3712-85

4 ex var.

Department of Automatic Control Lund Institute of Technology P.O. Box 118 S-221 00 Lund Sweden	<i>Document name</i> Final report	
	<i>Date of issue</i> December 1985	
	<i>Document Number</i> CODEN: LUTFD2/(TFRT-5341)/1-051/(1985)	
<i>Author(s)</i> Peter Marbe , Håkan Möller	<i>Supervisor</i> Björn Wittenmark	
	<i>Sponsoring organisation</i>	
<i>Title and subtitle</i> System med svagt dämpade poler. (Systems with weakle damped poles.)		
<i>Abstract</i> <p>This report treats how to control systems with weakly damped poles. Two masses coupled with an elastic shaft is cotrolled. One of the masses is affected by a motor and the output signal is the angular velocity of the other mass. First the open loop system is studied. Two different regulators is designed by using the polynomial design approach. One with a notchfilter to prevent the oscillary processpoles from beeing excited. The other regulator damps the oscillating modes. When simulating the closed loop system, we find that the inputsignal is too high during the first few samples. To prevent this, three of the modell-poles is moved from the origin. The best result is obtained when the poles are placed in 0.6. To speed up the system, a filter is placed after the controlsignal. We then get a softer signal in to the process. When we use this filter it is found that it is suitable to place three of the modell-poles in the origin.</p>		
<i>Key words</i>		
<i>Classification system and/or index terms (if any)</i>		
<i>Supplementary bibliographical information</i>		
<i>ISSN and key title</i>		<i>ISBN</i>
<i>Language</i> Swedish	<i>Number of pages</i> 51	<i>Recipient's notes</i>
<i>Security classification</i>		

förord

Vi vill härmed tacka vår handledare Björn Wittenmark, för hjälp med material och sitt kunnande. Vi vill även tacka de andra på institutionen som ställt upp när det behövt.

Lund 1985

A handwritten signature in cursive script, appearing to read "Peter Wadley".A handwritten signature in cursive script, appearing to read "Kalle Wadley".

INNEHÅLL

KAP	RUBRIK	SID
1.	Inledning	1
2.	Sammanfattning	2
3.	Analys av processen	3
3.1	Beskrivning av processen	3
3.2	Öppna systemets egenskaper	6
3.3	Sampling av processen	11
4.	Design av församlingsfilter	12
5.	Återkoppling med filter	13
5.1	Design av notchfilter	13
5.1.1	Beskrivning av notchfilter	13
5.1.2	Framtagning av gradtal	14
5.1.3	Framtagning av polynom	16
5.2	Design av dämpningsfilter	17
5.2.1	Beskrivning av dämpningsfilter	17
5.2.2	Framtagning av gradtal	17
5.2.3	Framtagning av polynom	17
6.	Simuleringsprogram med CTRL-C	18
6.1	Presentation av CTRL-C	18
6.2	Presentation av använda macron	18
7.	Analys av system	30
7.1	System med notchfilter	30
7.2	System med dämpningsfilter	32

KAP	RUBRIK	SID
8.	Metoder att undvika spikar i styrsignalen	34
8.1	A_m :s återstående poler längs reela axeln	34
8.1.1	Notchfilter	34
8.1.2	Dämpningsfilter	35
8.1.3	Störning av A_n -polynomet	36
8.1.4	Störning av A_d -polynomet	37
8.1.5	Störning av T-polynomet	37
8.2	A_m :s återstående poler placeras i båge	38
8.2.1	Notchfilter	39
8.2.2	Dämpningsfilter	39
8.2.3	Störning av A_n -, A_d - och T-polynomen	40
8.3	Filter lägges efter kontrollsignalen	40
8.3.1	Bestämning av filtrets konstanter	41
8.3.2	Störning av A_n - och A_d -polynomen	42
8.3.3	Störning av T-polynomet	42
8.4	Slutsats	43
9.	Slutlig sammanfattning	44
	Litteraturförteckning	47

1. INLEDNING

Denna rapport är gjord till institutionen för reglerteknik, Lunds Tekniska Högskola. Önskemål om att studera system med svagt dämpade poler har framlagts. Med system med svagt dämpade poler, menas att systemet har svårt att svänga in sig snabbt kring ett speciellt värde. Systemet är på gränsen till instabilt.

En praktisk aspekt på detta är veka robotarmar, det vill säga att robotarmarna är dimensionerade på ett sådant sätt, att böjstyvhet och/eller vridstyvhet har blivit lidande. Då en vek robotarm ska gå till bestämda koordinater, kommer den att svänga kring dessa koordinater på grund av dess vekhet (se fig 1.1).

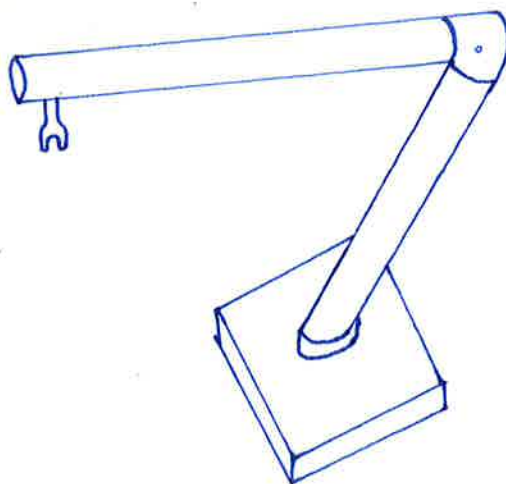


fig 1.1.

Ämnet har tidigare behandlats av Karl J. Åström och Björn Wittenmark i boken "Computer Controlled Systems" framför allt i kapitel 10. Man har där presenterat olika sätt att komma till rätta med "obekväma" poler. Ett problem man har, är att styrsignalen blir väldigt hög och orolig vid stegsvar. Syftet med denna rapport är att modifiera systemet som presenterats i boken så att styrsignalen får ett lugnare utseende i början.

2. SAMMANFATTNING

Denna rapport behandlar reglering av system med svagt dämpade poler.

I rapportens 3:e kapitel analyseras processen. Processen som behandlas, är en motor som påverkar en massa. Denna massa är i sin tur förbunden med en annan massa via en axel med fjäderkonstanten k . Jämviktsekvationer för processen ställs upp. Vidare studerar man det öppna systemets egenskaper, då olika materialkonstanter ändras. Till slut framtages lämplig samplingtid.

4:e kapitlet tar upp design av församlingsfilter. Detta för att undvika vikningseffekter vid sampling.

Kapitel 5 behandlar det återkopplade systemet med olika filter. Det tar också upp hur man tar fram önskade polynom genom polynomdesignmetoden.

I det 6:e kapitlet finns använda underprogram för simuleringar presenterade.

Det 7:e kapitlet analyserar det återkopplade systemet med såväl notchfilter som dämpningsfilter.

Kapitel 8 behandlar olika metoder att undvika "spikar" i inledningsskedet av styrsignalen. Vidare behandlas även hur systemet uppför sig vid störningar av de olika polynomen.

I det 9:e och sista kapitlet sammanfattas rapporten med slutsatser och kommentarer.

3. ANALYS AV PROCESSEN

3.1 Beskrivning av processen

En förenklad beskrivning av systemet med veka robotarmar, det vill säga system med svagt dämpade poler kan tänkas på följande vis (se fig 3.1).

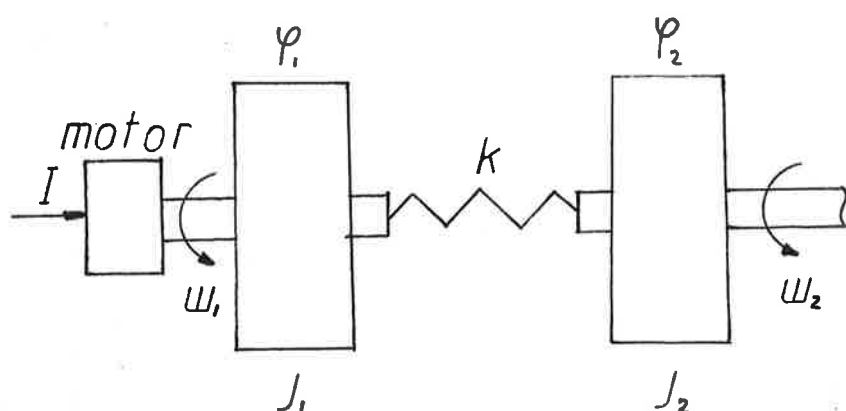


fig 3.1

En motor driver en last bestående av två massor. Dessa massor med tröghetsmoment J_1 och J_2 , är förbundna med varandra via en axel med fjäderkonstanten k . Vidare har man en viss dämpning d i processen. Massornas vinklar är φ_1 och φ_2 , deras vinkelhastigheter är ω_1 och ω_2 . Processens insignal är strömmen I till motorn. Den första massan kan störas med momentet v . Utsignalen från processen är ω_2 .

Sättes moment jämvikt upp för systemet upp fås:

$$J_1 \ddot{\varphi}_1 = k(\varphi_2 - \varphi_1) + d(\dot{\varphi}_2 - \dot{\varphi}_1) + k_1 u + v$$

$$J_2 \ddot{\varphi}_2 = k(\varphi_1 - \varphi_2) + d(\dot{\varphi}_1 - \dot{\varphi}_2)$$

För att lättare kunna simulera systemet skrives ekvationerna över på tillståndsform, där:

$$x_1 = \varphi_1 - \varphi_2$$

$$x_2 = \frac{\omega_1}{\omega_0}$$

$$x_3 = \frac{\omega_2}{\omega_0}$$

$$\omega_0 = \sqrt{k(J_1 + J_2) / (J_1 + J_2)}$$

På så sätt fås tillståndsekvationerna på matrisform:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & -1 \\ \alpha - 1 & -\beta_1 & \beta_1 \\ \alpha & \beta_2 & -\beta_2 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ \gamma \\ 0 \end{bmatrix} \cdot u + \begin{bmatrix} 0 \\ \delta \\ 0 \end{bmatrix} \cdot v$$

$$y = [0 \quad 0 \quad \omega_0] \cdot x$$

$$\alpha = \frac{J_1}{J_1 + J_2}$$

$$\beta_1 = \frac{d}{J_1 \cdot \omega_0}$$

$$\beta_2 = \frac{d}{J_2 \cdot \omega_0}$$

$$\gamma = \frac{k_i}{J_1 \cdot \omega_0}$$

$$\delta = \frac{1}{J_2 \cdot \omega_0}$$

Till att börja med sättes:

$$J_1 = 10/9$$

$$J_2 = 10$$

$$k = 1$$

$$d = 0.1$$

$$k_i = 1$$

Vilket medför $\omega_0 = 1$.

3.2 Öppna systemets egenskaper

Det givna systemet får följande överföringsfunktion i s-planet:

$$G(s) = \frac{\gamma \omega_0^2 (s\beta_2 + \alpha_2 \omega_0)}{s^3 + s^2 \omega_0 (\beta_1 + \beta_2) + s \omega_0^2}$$

Denna överföringsfunktion har en pol i 0, två svagt dämpade poler i $-0.05 \pm 0.999i$, samt ett nollställe i -10. Skrivs de komplexa polerna på formen

$$s^2 + 2\zeta_p \omega_p s + \omega_p^2$$

ser man att dessa har en dämpning, ζ_p på 0.05 och en resonans frekvens, ω_p på 1 rad/s.

Processen simuleras kontinuerligt i CTRL-C (vilket beskrivs i kapitel 6). Ger man systemet en kort impuls vid tiden 0, får impuls-svaret följande svängande utseende (se fig 3.2).

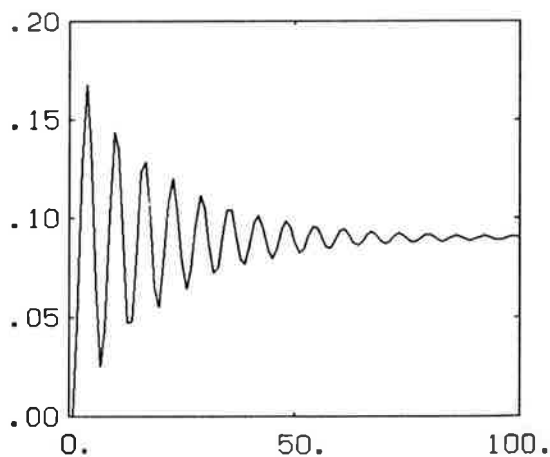


fig 3.2
impulssvar för
processen

Då man ser i BODE-diagrammet upptäcker man en kraftig nock vid resonans frekvensen $\omega_p = 1$ (se fig 3.3).

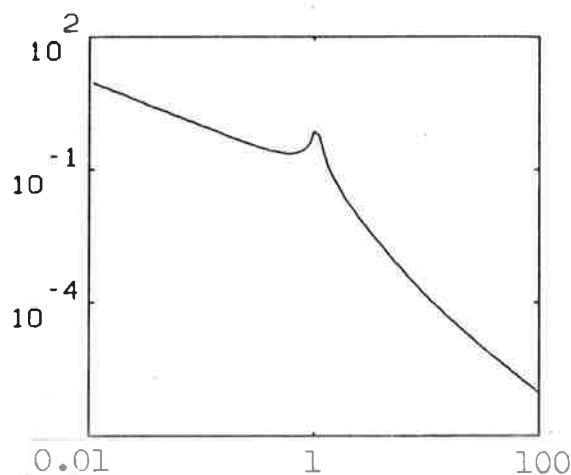


fig 3.3
BODE-diagram för
processen

Vi kommer här att ta upp vad som händer då man bland annat ändrar på tröghetsmomenten, fjäderkonstanten, dämpningskonstanten, samt då man observerar systemet från ω_1 respektive ω_2 . Först kommer vi att studera det senare fallet.

Från början har man låtit ω_2 varit utsignal och då fått följande överföringsfunktion:

$$G(s) = \frac{\gamma \omega_0^2 (s \beta_2 + \alpha_2 \omega_0)}{s^3 + s^2 \omega_0 (\beta_1 + \beta_2) + s \omega_0^2}$$

Man har då fått impuls-svar och BODE-diagram enligt figur 3.2 och 3.3. Låter man då istället ω_1 vara utsignal, kommer överföringsfunktionen få ett utseende enligt nedanstående.

$$G(s) = \frac{\gamma \omega_0 (s^2 + \beta_2 + \alpha_2 \omega_0)}{s^3 + s^2 \omega_0 (\beta_1 + \beta_2) + s \omega_0^2}$$

Detta resulterar med kraftiga svängningar med betydligt längre insvängnings tid (se fig 3.4). Vidare blir resonanstoppet avsevärt kraftigare i BODE-diagrammet (se fig 3.5).

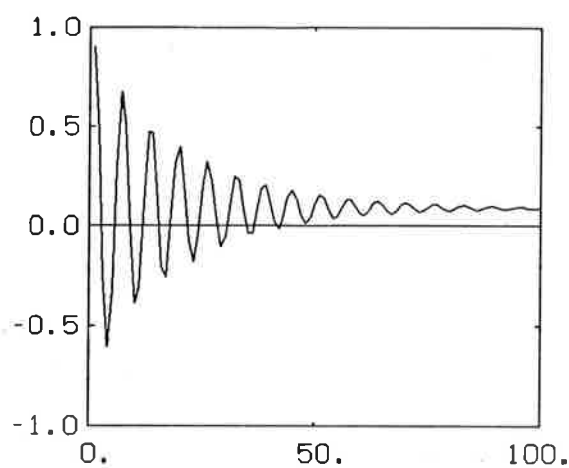


fig 3.4
impulssvar för
processen då ω_1
är utsignal

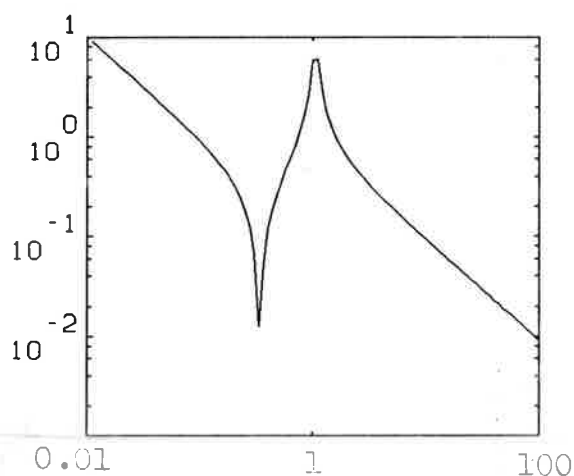


fig 3.5
BODE-diagram för
processen då ω_1
är utsignal

Studerar man ändringar av fjäderkonstanten k , upptäcker man att, vid högre värden på k får systemet svängningar med samma amplitud som tidigare. Dock får svängningarna en större frekvens då man ger en impuls (se fig 3.6). En ökad fjäderkonstant innebär en styvare axel.

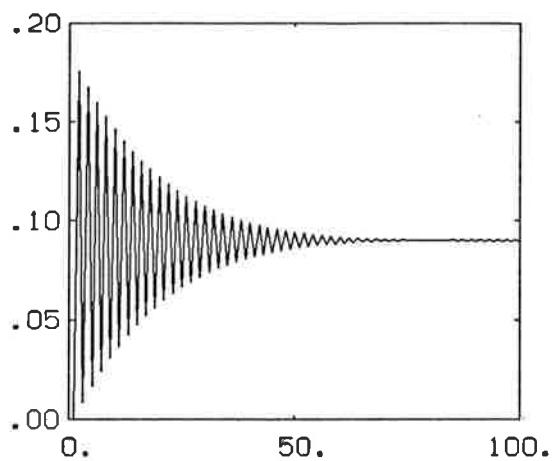


fig 3.6
impulsvar då
 $k=10$

Vidare om man ökar dämpningen d från 0,1 till 1, märkes att svängningarna stabiliserar sig mycket snabbare vid ett impuls svar (se fig 3.7).

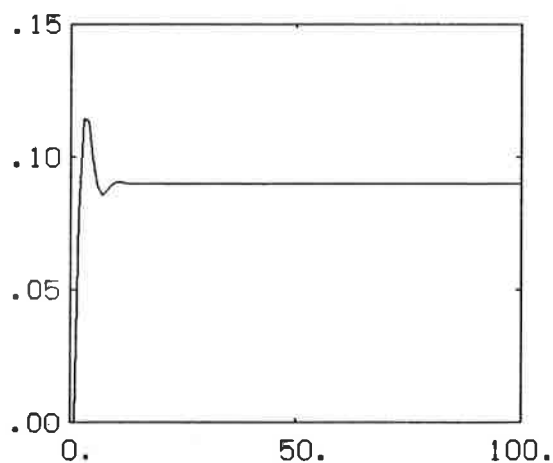


fig 3.7
impulsvar då
då $d=1$

Till sist kommer vi att se på vad som händer då man ändrar tröghetsmomenten hos de båda massorna. I ursprungsfallet är $J_1=10/9$ och $J_2=10$. Sätter man till exempel $J_1=J_2=10$, kommer insvängningen ta mycket längre tid vid ett impulssvar (se fig 3.8).

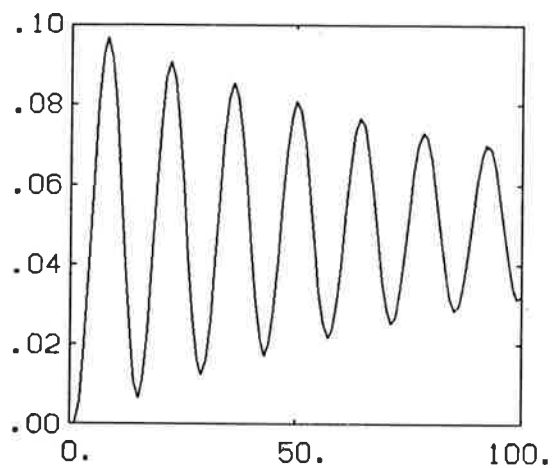


fig 3.8
impulssvar då
 $J_1=J_2=10$

3.3 Sampling av processen

Själva samplingen sker i CTRL-C (se kapitel 6). Det är viktigt att man väljer en bra samplingstid beroende av hur det önskvärda systemet ser ut. Samplingstiden påverkar hur snabbt störningar blir upptäckta. En för lång samplingstid medför att störningar upptäcker sent. En för kort samplingstid upptar för stor datortid.

Man vill ha ett system som har de dominerande polerna så att dämpningen ζ_m blir 0.7 och resonansfrekvens ω_m blir 0.5 rad/s.

Shannons samplingssteorem för andra gradens system säger att:

$$\frac{\frac{\varphi}{\tan\varphi}}{\omega_m h} \approx 2-4$$

där

$$\varphi = \cos\zeta$$

Detta medför en lämplig samplingstid på 0,5 sekunder.

4. DESIGN AV FÖRSAMPLINGSFILTER

För att undvika vinkningseffekter vid samplingen, kopplar man samman ett filter av andra ordningen med processen. Detta filter är av formen

$$G = \frac{\omega_f^2}{s^2 + 1,4\omega_f s + \omega_f^2}$$

i s-planet. Resonansfrekvensen ω_f väljes till 2. Filtret ges en dämpning på 0,7. Vidare samplas filtret och kopplas i serie med den övriga processen (se kapitel 6.2).

Den tidsdiskreta modellen ser ut på följande vis

$$y(k) = \frac{B(q)}{A(q)} \cdot u_C(k)$$

där

$$A(q) = \underbrace{(q-1)(q^2 - 1,7125q + 0,9513)}_{\text{process}} \underbrace{(q^2 - 0,7497q + 0,2432)}_{\text{filter}}$$

och

$$B(q) = 0,1417(q+12,1127)(q+1,3395)(q+0,2227)(q-0,0024)$$

Systemet är av icke minfastyp. Poler och nollställen till processen med församlingsfilter i det tidsdiskreta planet finns inritade i figur 4.1 nedan

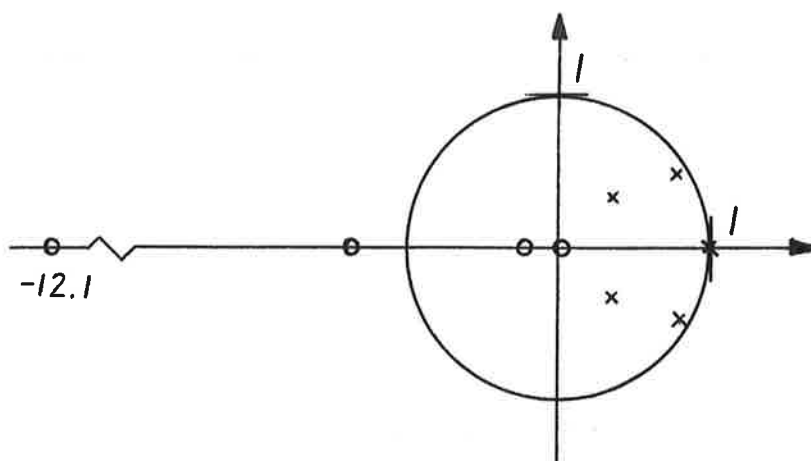


fig 4.1.

5. ATERKOPPLING MED FILTER

5.1 Design av notchfilter

5.1.1 Beskrivning av notchfilter

Man måste ta hänsyn till den mekaniska resonansen hos processen, när man designar det slutna systemet. Detta hänger samman med att den frekvens $\omega_p = 1$ som är förknippad med den mekaniska resonansen ligger väldigt nära det slutna systemets önskvärda frekvens $\omega_m = 0,5$. Ett sätt att ta hänsyn till den mekaniska resonansen, är att introducera ett kompenserande nät som undviker att excitera de oscillerande process-polerna. Detta filter som användes, har en resonanstopp vid de oönskade moderna och kallas notch-filter. Denna metod garanterar att de oscillernde moderna inte exciteras av kommando signaler. Emellertid medför det ingen dämpning av de oscillerande moderna vid en momentstörning. Detta betyder att det slutna systemet kommer att reagera på samma sätt vid excitation av de oscillerande moderna som det öppna systemet.

Själva notchfilter kompensatorn erhålles genom att använda polynomdesign metoden. Vi tänker oss ett enkelt återkopplat system enligt figur 5.1 nedan.

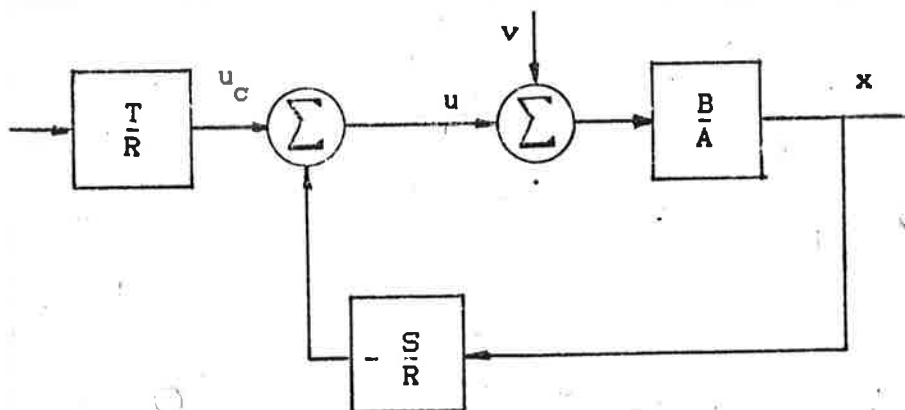


fig 5.1
återkopplat system

med överföringsfunktionen:

$$H_m = \frac{BT}{AR+BS} = \frac{B_m}{A_m}$$

Tittar man på polerna i överföringsfunktionen, så måste den innehålla polynomet A_m . Det vill säga det polynom som innehåller de önskvärda egenskaper man vill ha. Vidare måste ett polynom A_n som syftar på de oscillerande polerna ingå. Detta eftersom återkopplingen inte påverkar dessa poler. Utöver detta måste även ett observerarpolynom A_o ingå. Då man inte dividerar bort några process nollställen får man:

$$AR+BS=A_m A_n A_o$$

5.1.2 Framtagning av gradtal

Framtagning av gradtal hos de olika polynomen fås genom att se på ekvationerna. Till att börja med krävs det att

$$\text{grad } R \geq \text{grad } T$$

$$\text{grad } R \geq \text{grad } S$$

för att göra bakåtkopplings- och framåtkopplings-överföringsfunktioner realiserbara. Då uträkningarna i datorn endast tar en bråkdel av samplingstiden kan grad R sättas lika med grad T och grad S.

$$\text{grad } R = \text{grad } S = \text{grad } T$$

Därmed blir där ingen tidsfördröjning i regulatorn.

Eftersom det slutna systemet måste vara stabilt, kan man bara ta bort stabila nollställen. B-polynomet från det öppna systemet kan delas upp i ett B^+ -polynom och ett B^- -polynom, där B^+ -polynomet har sina nollställen innanför enhetscirkeln.

$$B = B^+ B^-$$

B^- -polynomet ligger utanför enhetscirkeln och kan ej vara en faktor av $AR+BS$, utan en faktor av B_m .

$$B = B^- B'_m$$

Eftersom B^+ är en faktor av $AR+BS$, måste det även vara en faktor av R.

$$R = BR'$$

Sätter man in detta i överföringsfunktionen

$$\frac{BT}{AR+BS} = \frac{B}{A_m}$$

får man

$$\frac{T}{AR'+B^-S} = \frac{B'}{A_m}$$

Det vill säga A_m -polynomet är en faktor av $AR'+B^-S$. Det finns en lösning till polplaceringsdesignen om

$$\text{grad} \frac{A_m}{B_m} \geq \text{grad} \frac{A}{B}$$

och

$$\text{grad} A_o \geq 2\text{grad} A - \text{grad} A_m - \text{grad} B^+ - 1$$

Ansättes $\text{grad} A_m$ lika med $\text{grad} A$ och $\text{grad} B_m$ lika med $\text{grad} B$ fås $\text{grad} A_m = 5$ och $\text{grad} B_m = 4$. Vidare fås $\text{grad} A_o = 2$ då $\text{grad} B^+ = 2$. Vidare är $\text{grad}(AR'+B^-) = \text{grad} A_o A_m$. Det vill säga

$$\text{grad}(AR/B^+ + BS/B^+) = \text{grad} A_o A_m$$

Eftersom $\text{grad} S \leq \text{grad} R$ och $\text{grad} B < \text{grad} A$ fås att $\text{grad} AR' = \text{grad} A_o A_m$, härav

$$\text{grad} R' = \text{grad} A_o A_m$$

Vilket medför $\text{grad} R' = 2$ och $\text{grad} R = 4$. Dessutom är $\text{grad}(AR+BS) = \text{grad} A_n A_m A_o$, vilket ger $\text{grad} A_n = 2$.

Sammanfattningsvis är

$$\text{grad } A = 5$$

$$\text{grad } B = 4$$

$$\text{grad } A_m = 5$$

$$\text{grad } B_m = 4$$

$$\text{grad } A_n = 2$$

$$\text{grad } A_o = 2$$

$$\text{grad } R = 4$$

$$\text{grad } S = 4$$

$$\text{grad } T = 4$$

5.1.3. Framtagning av polynom

I A_n polynomet måste sedan tidigare de oscillerande processpolerna ingå.

$$A_n = (q^2 - 1, 7125q + 0, 9513)$$

Polynomet A_m väljes utgående från de önskvärda specifikationerna med dämpning 0,7 och en resonans frekvens 0,5 rad/s. De tre övriga polerna lägges i noll.

$$A_m = q^3 (q^2 - 1, 6522q + 0, 7047)$$

I polynomet A_o , lägges till en början de båda observerarpolerna i noll. R- och S- polynomen fås genom att

$$AR + BS = A_m A_n A_o$$

vilket räknas ut i CTRL-C (se kapitel 6.2). T-polynomet sättes till

$$T = t_o A_n A_o$$

där

$$t_o = \frac{A_m(1)}{B(1)}$$

5.2. Design av dämpningsfilter

5.2.1 Beskrivning av dämpningsfilter

I stället för att ha ett notchfilter, använder man här i stället ett dämpningsfilter som även dämpar de oscillerande moderna. Detta märkes på att svängningarna försvinner vid en momentstörning, (se kapitel 7). Dämpningsfiltret erhålles på samma sätt som tidigare, det vill säga genom polynomdesignmetoden.

Samma polynom som tidigare måste ingå, förutom det att A_n polynomet bytes ut mot ett A_d polynom. Härav fås

$$AR+BS=A_m A_d A_o$$

5.2.2 Framtagning av gradtal

Gradtalen hos de olika polynomen blir av samma storlek som i kapitel 5.1.2. Det vill säga

$$\text{grad } A_d = 2$$

5.2.3 Framtagning av polynom

För att dämpa de oscillerande moderna, ansätter man ett polynom av andra graden. I detta polynom har man jämfört med notchfilterpolynomet ökat dämpningen från 0,05 till 0,707. Vidare behåller man samma resonansfrekvens 1 rad/s. Detta medför

$$A_d = (q^2 - 1, 3176q + 0, 4971)$$

då samplingstiden är 0,5 sekunder.

6. SIMULERINGSPROGRAM MED CTRL-C

6.1 Presentation av CTRL-C

Detta data-system arbetar i huvudsak med matriser. Det simulerar såväl tidskontinuerliga såsom tidsdiskreta system. Stegsvär, impulssvar, BODE-diagram o.s.v kan ritas upp. För att lättare kunna följa de i avsnitt 6.2 beskrivna underprogrammen, presenterar vi här några av de mest använda kommandonen.

TF2SS: överför överföringsfunktion till matriser på tillståndsform.

SS2TF: överför matriser på tillståndsform till överföringsfunktion.

C2D: tar över system från kontinuerlig till tidsdiskret form.

SERI: kopplar samman olika system med varandra.

För ytterligare information se användarmanual för CTRL-C.

För att simulera processerna har flera macron används. De olika underprogrammen finns presenterade med förklaringar i kommande kapitel.

6.2 Presentation av använda macron

PLT1

Här plottas utsignalen x ut då man ger systemet ett steg vid tiden 0 och ger det ett störande moment v vid tiden $t=25$ sekunder (=50 samplingsintervall).

$$x = \frac{A_n A_o B t_o}{A_m A_n A_o} \cdot u_c + \frac{BR}{A_m A_n A_o} \cdot v$$

Först simuleras den delen som är beroende av referenssignalen u_c , därefter den delen som är beroende av det störande momentet v . Efter detta lägges signalerna tillsammans och plottas ut. Indata är de ändringsbara modellpolerna $mp1$ - $mp3$ som skrives in manuellt, samt programmet ANNA eller ADDA.

```
//PLT1
//PLOTTAR UTSIGNAL
DO MULT;
FOR K=1:200,UC(K)=1;
Y=DSIM(A,B,C,D,UC');
FOR I=1:200,IF I=50,V(I)=5;ELSE V(I)=0;
Z=DSIM(AA,BB,CC,DD,V');
X=Y+Z;
PLOT(X)
```

PLT2

Styrsignalen för det slutna systemet plottas ut här.

$$u = \frac{TA}{A_m A_n A_o} \cdot u_c - \frac{BS}{A_m A_n A_o} \cdot v$$

Utrycken för u_c och v tages fram, simuleras var för sig, för att sedan summeras och plottas ut. Programmen ANNA eller ADDA används som indata.

```
//PLT2
//PLOTTAR STYRSIGNAL
BP=NA;
SP=[ S1 S2 S3 S4 S5];
TP=T0*ANO;
TA=CONV(TP,QA);
[ CU,BU,CU,DU]=TF2SS(TA,AMNO);
BS=CONV(SP,NA);
[ AX,BX,CX,DX]=TF2SS(BS,AMNO);
FOR J=1:200,UL(J)=1;
UCS=DSIM(AU,BU,CU,DU,UL');
FOR I=1:200;IF I=50,V(I)=5;ELSE V(I)=0;
XCS=DSIM(AX,BX,CX,DX,V');
US=UCS-XCS;
US(199)=9.5;
US(200)=-9.5;
PLOT(US)
```

ANNA

Programmet tar fram matriserna A, B, C, D och AA, BB, CC, DD, som användes i PLT1 och PLT2. Modellpolynomen mp1-mp3 som skrives in manuellt, samt underprogrammen nedan användes som indata.

```
//ANNA
//BERAKNINGAR MED NOTCHFILTER
DO TD;
DO KOFBEST;
DO AN;
DO G;
DO FIA;
```

ADDA

Samma som ANNA förutom att dämpningspolynomet AD användes istället för AN.

```
//ADDA
//BERAKNINGAR MED DAMPNINGSFILTER
DO TD;
DO KOFBEST;
DO AD;
DO G;
DO FIA;
```

TD

Detta program tar fram överföringsfunktionen na/qa för det öppna systemet med församlingsfilter i samplad form.

```
//TD
//TAR FRAM OVERFORINGSFUNKTIONEN I SAMPLAD FORM
TERM='4006';
P1=0;
P2=P1;
TS=0.5;
HARD='PEN';
j1=10/9;
j2=10;
k=1;
d=0.1;
ki=1;
w0=sqrt(k*(j1+j2)/(j1*j2));
a1=j1/(j1+j2);
be1=d/(j1*w0);
be2=d/(j2*w0);
ga=ki/(j1*w0);
a=[0 w0 -w0;w0*(a1-1) -w0*be1 w0*be1;w0*a1 w0*be2 -w0*be2];
b=[0;ga;0];
c=[0 0 w0];
d=0;
wf=2;
n0=wf*wf;
n=[n0];
q0=wf*wf;
q1=1.4*wf;
q2=1;
q=[q2 q1 q0];
[a1,b1,c1,d1]=tf2ss(n,q);
[a2,b2,c2,d2]=seri(a,b,c,d,a1,b1,c1,d1);
[p,g]=c2d(a2,b2,ts);
[na,qa]=ss2tf(p,g,c2,d2,1);
```

Till en början sättes observerarpoler till noll. Därefter sättes de olika konstanterna. Systemekvationerna skrives in i matrisform (se kapitel 3.1). Överföringsfunktionen för församlingsfiltret skrives in och omvandlas till matrisform. Slutligen seriekopplas detta med processen samplas och tages över på överföringsfunktionsform.

KOFBEST

Detta program tar fram en bandmatris BM utgående från överföringsfunktionen för det slutna systemet na/qa. Således krävs det att man använder programmet TD som indata till detta program.

```
//KOFBEST
//TAR FRAM BANDMATRISEN BM
M1=[ 1;0;0;0;0;0];
M2=[ 0;1;0;0;0;0];
M3=[ 0;0;1;0;0;0];
M4=[ 0;0;0;1;0;0];
M5=[ 0;0;0;0;1;0];
M6=[ 0;0;0;0;0;1];
A1=QA*M1;
A2=QA*M2;
A3=QA*M3;
A4=QA*M4;
A5=QA*M5;
A6=QA*M6;
B1=NA*M2;
B2=NA*M3;
B3=NA*M4;
B4=NA*M5;
B5=NA*M6;
V1=[ A1 0 0 0 0 0 0 0 0];
V2=[ A2 A1 0 0 0 B1 0 0 0];
V3=[ A3 A2 A1 0 0 B2 B1 0 0];
V4=[ A4 A3 A2 A1 0 B3 B2 B1 0];
V5=[ A5 A4 A3 A2 A1 B4 B3 B2 B1];
V6=[ A6 A5 A4 A3 A2 B5 B4 B3 B2 B1];
V7=[ 0 A6 A5 A4 A3 0 B5 B4 B3 B2];
V8=[ 0 0 A6 A5 A4 0 0 B5 B4 B3];
V9=[ 0 0 0 A6 A5 0 0 0 B5 B4];
V10=[ 0 0 0 0 A6 0 0 0 0 B5];
BM=[ V1;V2;V3;V4;V5;V6;V7;V8;V9;V10];
```

AN

Här tages notchfilterpolynomet AND fram. Man har utgått från tillståndsmatriserna A, B, C, D i programmet TD och samplat dessa. Därefter har man tagit över det på överföringsform. Slutligen dividerar man bort den pol som ligger i +1, med hjälp av kommandot DECONV.

```
//AN
//TAR FRAM AN-POLYNOM
[P,G]=C2D(A,B,TS);
[N,Q]=SS2TF(P,G,C,D,1);
[AN,R]=DECONV(Q,[1 -1]);
AND=AN;
```

AD

Dämpningspolynomet AND tages fram utgående från givet ω_p och ζ_p i s-planet. Filtret som är på överföringsfunktionsform, tages över på matrisform och samplas. Därefter överföres det på överföringsfunktionsform igen.

```
//AD
//TAR FRAM DAMPNINGSPOLYNOM
ZP=0.707;
WP=1;
NP=[WP*WP];
QP=[1 2*ZP*WP WP*WP];
[AP,BP,CP,DP]=TF2SS(NP,QP);
[PP,GP]=C2D(AP,BP,TS);
[NP5,AND]=SS2TF(PP,GP,CP,DP,1);
```

AM

Programmet tar fram modellpolynomet AM. Indata till programmet är de tre ändringsbara modellpolerna mp1, mp2 och mp3, som skrives in manuellt. Polynomet QM är det önskade polynomet i tidskontinuerlig form. Detta polynom göres om till tidsdiskret form. Polynomet multipliceras med de tre ändringsbara modellpolerna mp1-mp3.

```
//AM
//BERAKNAR AM
NM=[0.25];
QM=[1 0.7 0.25];
[AMM,BMM,CMM,DMM]=TF2SS(NM,QM);
[PMM,GMM]=C2D(AMM,BMM,TS);
[NM,AM]=SS2TF(PMM,GMM,CMM,DMM,1);
AM=CONV([1 -MP1],AM);
AM=CONV([1 -MP2],AM);
AM=CONV([1 -MP3],AM);
```

TIA

Här tages de sex koefficienterna AM1-AM6 fram en efter en ur modellpolynomet A_m . Indata till programmet är AM.

```
//TIA
//TAR FRAM AM:S KOEFFICIENTER
AM1=AM*[1;0;0;0;0;0];
AM2=AM*[0;1;0;0;0;0];
AM3=AM*[0;0;1;0;0;0];
AM4=AM*[0;0;0;1;0;0];
AM5=AM*[0;0;0;0;1;0];
AM6=AM*[0;0;0;0;0;1];
```


G

Det slutna systemets överföringsfunktion på tillståndsform i matriserna A, B, C, D tages fram här. Observerarpolynomet A_o beräknas med hjälp av kommandot CONV. Observerarpolerna är givna i TD. Indata till programmet är TD, KOFBEST, AND, AM, och TIA.

```
//G
//BERAKNAR B*T DER B=NA (FRAN TD) OCH T=AN*AO*T0
AO=CONV([1 -P1],[1 -P2])
DO AM;
DO TIA;
T0=(AM1+AM2+AM3+AM4+AM5+AM6)/(B1+B2+B3+B4+B5);
ANO=CONV(AND,AO);
TP=ANO*T0;
AMNO=CONV(AM,ANO);
BT=CONV(TP,NA);
[A,B,C,D]=TF2SS(BT,AMNO);
```

FIA

Programmet tar fram de fem koefficienterna R1-R5 ur R-polynomet, samt S1-S5 ur S-polynomet. Bandmatrisen BM från programmet KOFBEST användes som indata.

```
//FIA
//R OCH S-KOEFFICIENTER FRAMTAGES UR BANDMATRISEN
C1=AMNO*[ 1;0;0;0;0;0;0;0;0;0];
C2=AMNO*[ 0;1;0;0;0;0;0;0;0;0];
C3=AMNO*[ 0;0;1;0;0;0;0;0;0;0];
C4=AMNO*[ 0;0;0;1;0;0;0;0;0;0];
C5=AMNO*[ 0;0;0;0;1;0;0;0;0;0];
C6=AMNO*[ 0;0;0;0;0;1;0;0;0;0];
C7=AMNO*[ 0;0;0;0;0;0;1;0;0;0];
C8=AMNO*[ 0;0;0;0;0;0;0;1;0;0];
C9=AMNO*[ 0;0;0;0;0;0;0;0;1;0];
C10=AMNO*[ 0;0;0;0;0;0;0;0;0;1];
CQ=[ C1;C2;C3;C4;C5;C6;C7;C8;C9;C10];
RS=BM\CQ;
R1=[ 1 0 0 0 0 0 0 0 0 0]*RS;
R2=[ 0 1 0 0 0 0 0 0 0 0]*RS;
R3=[ 0 0 1 0 0 0 0 0 0 0]*RS;
R4=[ 0 0 0 1 0 0 0 0 0 0]*RS;
R5=[ 0 0 0 0 1 0 0 0 0 0]*RS;
S1=[ 0 0 0 0 0 1 0 0 0 0]*RS;
S2=[ 0 0 0 0 0 0 1 0 0 0]*RS;
S3=[ 0 0 0 0 0 0 0 1 0 0]*RS;
S4=[ 0 0 0 0 0 0 0 0 1 0]*RS;
S5=[ 0 0 0 0 0 0 0 0 0 1]*RS;
```

MULT

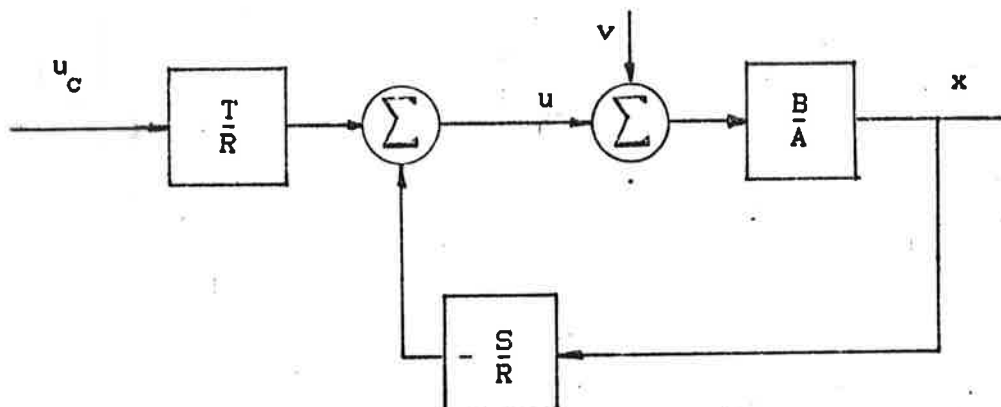
Programmet använder resultaten från programmen TD, KOFBEST, AN, G, och FIA för att få fram $BR/(A_{m \ n \ o})$ på tillståndsform i matriserna AA, BB, CC, DD.

```
//MULT  
//BERAKNAR B*R/(AM*AN*AO)  
R=[R1 R2 R3 R4 R5];  
BR=CONV(NA,R);  
[AA,BB,CC,DD]=TF2SS(BR,AMNO);
```

7. ANALYS AV SYSTEM

7.1 System med notchfilter

Vid simulering av systemet använder vi de macron i CTRL-C som redan presenterats i kapitel 6. Modellen ser i princip ut enligt nedan.



Figur 7.1

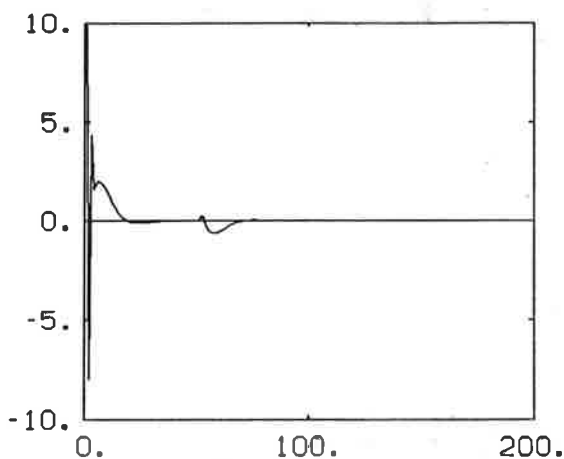
Det är B/A som är själva systemet och skall regleras. R -, S - och T -polynomen framräknas och blir olika beroende på vad A_m , A_n och A_o bestäms till. Vid första simuleringen sättes:

$$A_m = (q^2 - 1.6522q + 0.7047)q^3$$

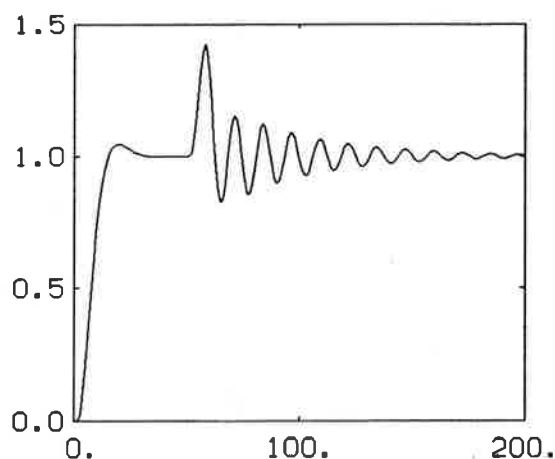
$$A_n = (q^2 - 1.7125q + 0.9513)$$

$$A_o = (q - p_1)(q - p_2)$$

Observerarpolerna p_1 och p_2 sättes till noll i de första simuleringarna. Samplingstiden är enligt tidigare 0,5 sekunder. Figur 7.2 visar styr och utsignal då vi simulerar ett stegsvar vid $t=0$ och en momentstörning vid $t=25$ sekunder (=50 samplingsintervall).



Figur 7.2 Styrsignal



Utsignal

Som synes blir styrsignalen väldigt hög i initialskedet. I vårt fortsatta arbete kommer vi bland annat försöka finna metoder att få bort dessa "spikar".

Ur figur 7.1 erhålles:

$$\begin{aligned} u &= \frac{T}{R} \cdot u_c - \frac{S \cdot x}{R} \\ x &= \frac{B}{A} \cdot (v + u) \end{aligned} \quad (7.1)$$

Styrsignalen u blir då

$$u = \frac{T}{R} \cdot u_c - \frac{SB}{RA} \cdot u - \frac{SB}{RA} \cdot v \quad (7.2)$$

$$u \cdot (AR + BS) = TAu_c - BSv$$

$$u = \frac{TA}{AR + BS} \cdot u_c - \frac{BS}{AR + BS} \cdot v \quad (7.3)$$

där $T = t_0 A_n A_o$ och $AR + BS = A_m A_n A_o$

$$\Rightarrow u = \frac{t_0 A_n A_o A}{A_m A_n A_o} \cdot u_c - \frac{BS}{A_m A_n A_o} \cdot v$$

$$u = \frac{t_0 A}{A_m} \cdot u_c - \frac{BS}{A_m A_n A_o} \cdot v \quad (7.4)$$

På liknande sätt fås för utsignalen x

$$x = \frac{Bt_0}{A_m} \cdot u_C + \frac{BR}{A_m A_n A_o} \cdot v \quad (7.5)$$

Det är termen framför referenssignalen u_C i ekvation 7.4 som är intressant vid stegsvar. Varken t_0 eller A kan vi ändra på. Vår enda möjlighet att förändra styrsignalens uppträdande vid steg är att ändra i A_m . Varken A_n eller A_o har någon betydelse vid ändring i referenssignal, däremot vid störning enligt ekvation 7.4.

7.2 System med dämpningsfilter

Vid simulering av systemet med dämpningsfilterdesign ser modellen likadan ut som vid notchfilterdesign. Det enda som skiljer dem åt är att vi har ett dämpningspolynom $A_d = q^2 - 1.3176q + 0.4971$ istället för A_n .

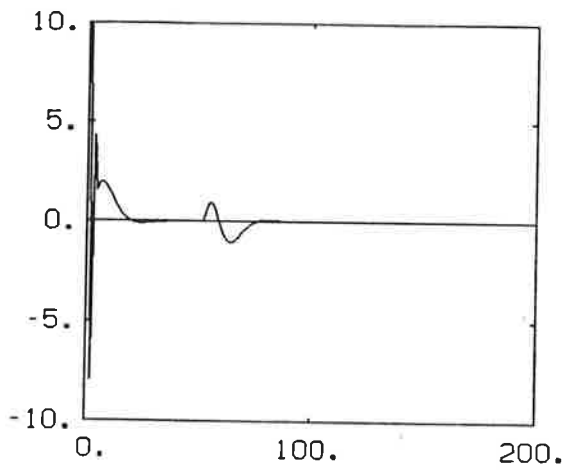
Altså blir

$$u = \frac{t_0 A}{A_m} \cdot u_C - \frac{BS}{A_m A_d A_o} \cdot v \quad (7.6)$$

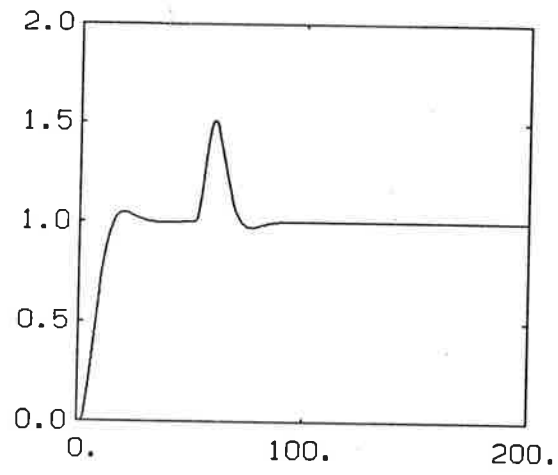
Och utsignalen x blir

$$x = \frac{Bt_0}{A_m} \cdot u_C + \frac{BR}{A_m A_d A_o} \cdot v \quad (7.7)$$

Termen framför referenssignalen u_C blir densamma som i notchfilterfallet. Alltså beter sig systemen lika vid referenssignalsändring. Vid störning skiljer sig de olika systemen åt. Dämpningsfiltret dämpar ut svängningarna efter störning mycket bättre. Figur 7.3 på nästa sida visar stegsvar och styrsignal vid dämpningsfallet.



Figur 7.3 Styrsignal



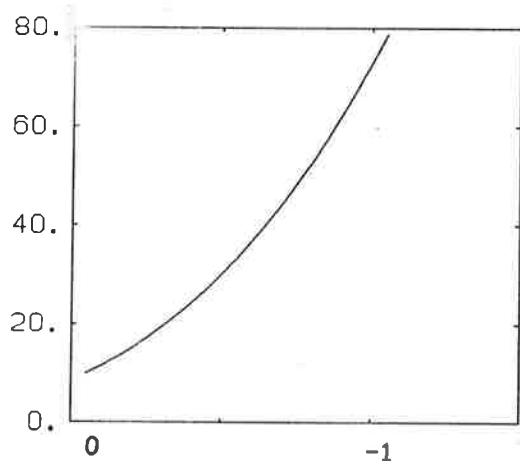
Utsignal

8 METODER ATT UNDVIKA SPIKAR I STYRSIGNALEN

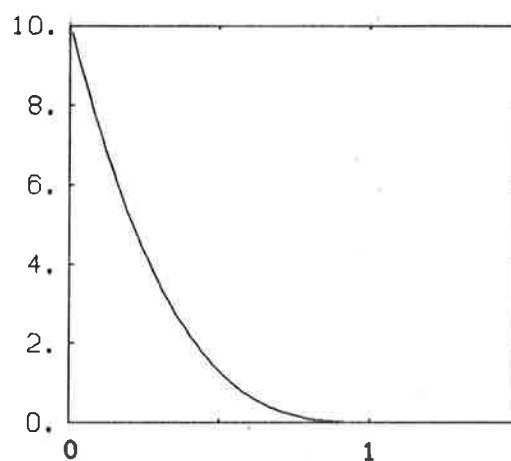
8.1 Am:s återstående poler placeras längs reela axeln

8.1.1 Notchfilter

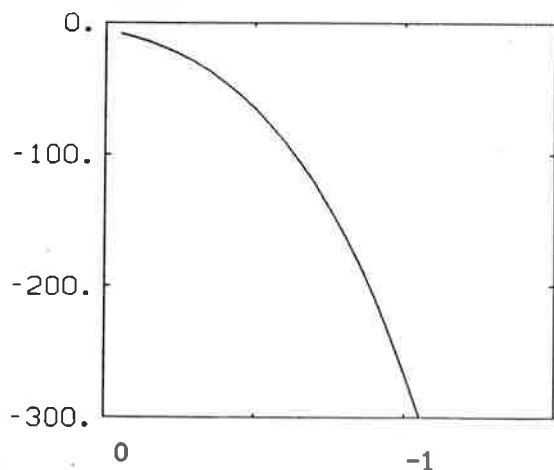
Vi sätter $A_m = (q^2 - 1.6522q + 0.7047)(q - \alpha)^3$ och simulerar systemet för olika α .



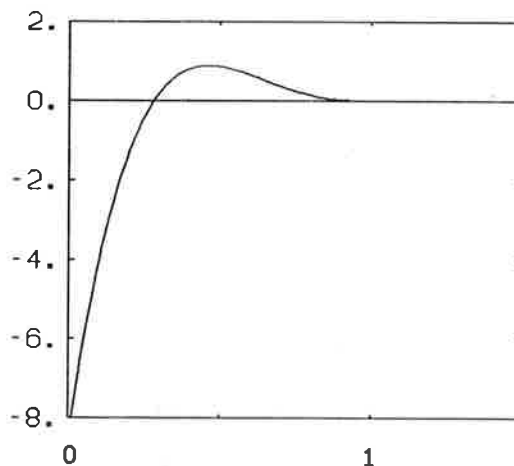
Figur 8.1 styrsignalen under 1:a samplingen sfa α ($-1 < \alpha < 0$)



Figur 8.3 styrsignalen under 1:a samplingen sfa α ($0 < \alpha < 1$)

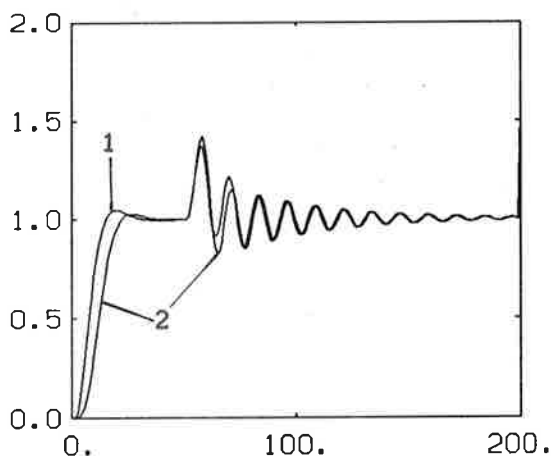


Figur 8.2 styrsignalen under 2:a samplingen sfa α ($-1 < \alpha < 0$)

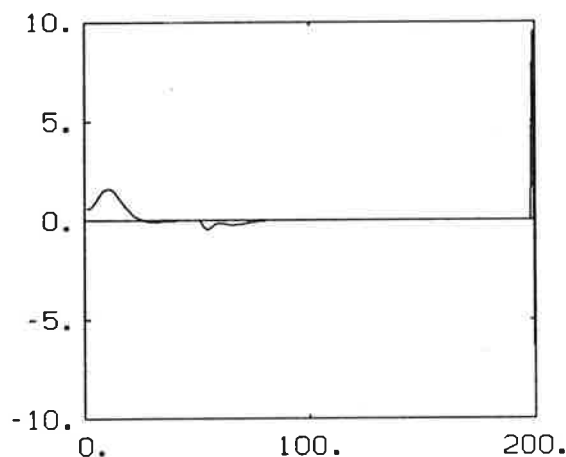


Figur 8.4 styrsignalen under 2:a samplingen sfa α ($0 < \alpha < 1$)

Figurerna på föregående sida visar att styrsignalens absolutvärde under de första samplingarna blir större med mindre α . Det visar sig emellertid att styrsignalen vid störningen blir större med ökande α . Om α sättes större än 0.70 blir styrsignalen större vid störningen än vid referenssignalsändringen. Efter många simuleringar kom vi fram till att 0.60 är det α -värde som uppfyller våra krav bäst. $\alpha=0.60$ är tillräckligt stor för att ej ge spikar i styrsignalen i början. Om man valt ett större α hade man också sluppit ifrån spikarna i början, men dels hade systemet blivit långsammare, och så hade styrsignalen vid störning blivit onödigt stor. Vi har också provat att lägga en eller två av de tre polerna i origo eller på negativa reela axeln och samtidigt placerat de övriga längs positiva reela axeln. Det bästa blir dock att placera alla tre polerna i 0.60. Att detta system med $\alpha=0.60$ är långsammare än det ursprungliga syns i figur 8.5 nedan.



Figur 8.5 Utsignal då
(1) $\alpha=0$
(2) $\alpha=0.60$

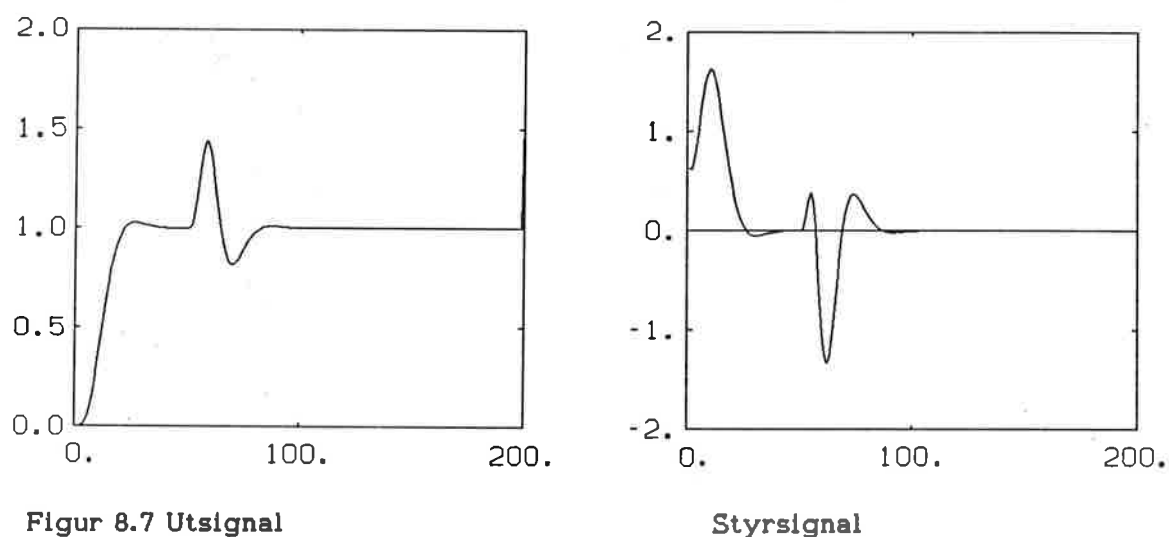


Figur 8.6 Styrsignal
för $\alpha=0.60$

8.1.2 Dämpningsfilter

Som redan nämnts i kapitel 7 reagerar styrsignalen likadant vid referenssignalsändring oberoende om vi har notch- eller dämpningsfilter enligt ekvation 7.4 och 7.6. Signalen i början ser alltså likadan ut i de båda fallen, och figurerna 8.1-8.4 gäller också för detta filtret. Efter ett antal simuleringar kom vi fram till att $\alpha=0.60$ är det bästa värdet även för dämpningsfiltret. Det som skiljer utsignalen då vi lagt tre av A_m 's poler i 0.6 istället för i 0 är dels att systemet blir långsammare och "underslängen" efter

störningen blir större. Om α sättes större blir "underslängen" allt kraftigare och styrsignalen går allt längre ner på negativa sidan vid störningen. Vi har även försökt få ner amplituden på utsignalen vid störningen genom att flytta observerarpolerna A_o , men någon större skillnad kunde ej märkas. Förmodligen kompenseras ändringen i A_o av S-polynomet i ekvation 7.7. S-polynomet framräknas ju bland annat med A_o . I figurerna nedan visas styr- och utsignal för systemet med dämpningsfilter och med $A_m = (q^2 - 1.6522q + 0.7047)(q - 0.6)^3$.



Figur 8.7 Utsignal

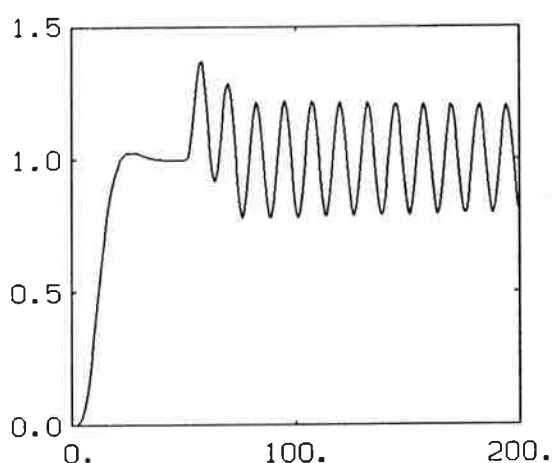
Styrsignal

8.1.3 Störning i A_n -polynomet

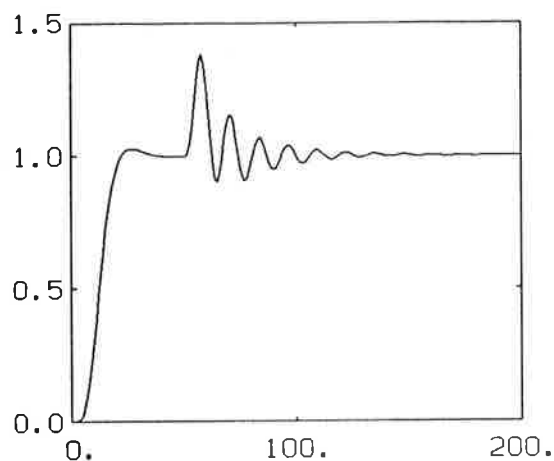
Normalt har A_n polerna $0.8562 \pm 0.4687i$. Om polerna flyttas ut så långt som $0.87 \pm 0.48i$ blir systemet nästan ostabilt (se figur 8.8). Detta beror på att

$$\sqrt{0.87 \cdot 0.87 + 0.48 \cdot 0.48} = 0.9936$$

Alltså ligger polerna väldigt nära enhetscirkeln. Om vi nöjer oss med att flytta ut polerna till $0.86 \pm 0.47i$ uppträder systemet ungefär som då polerna är $0.8562 \pm 0.4687i$. Om vi flyttar polerna inåt till $0.84 \pm 0.45i$ uppträder utsignalen efter störningen mer dämpad än den ursprungliga, och ju längre in vi flyttar polerna desto mer dämpad blir den efter störningen. A_n tål en störning på 0.005 utåt är ganska okänslig för störningar inåt.



Figur 8.8 A_n :s poler utflyttade till $0.87 \pm 0.48i$



Figur 8.9 A_n :s poler inflyttade till $0.84 \pm 0.45i$

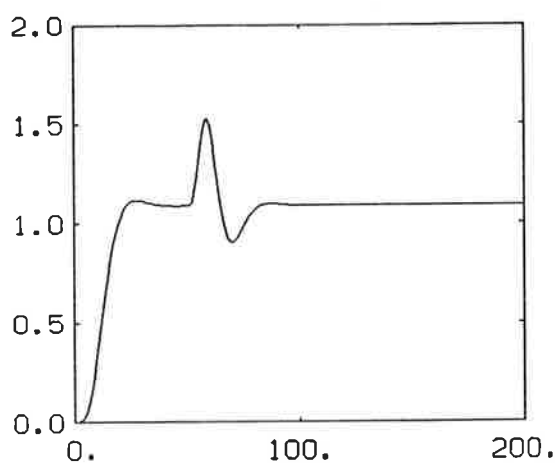
8.1.4 Störning av A_d -polynomet

Normalt har A_d polerna $0.6588 \pm 0.2432i$ altså på behörigt avstånd från enhetscirkeln. Det märks därför ingen större skillnad varken vi flyttar ut A_d :s poler till $0.67 \pm 0.26i$ eller vi flyttar in dem till $0.64 \pm 0.23i$

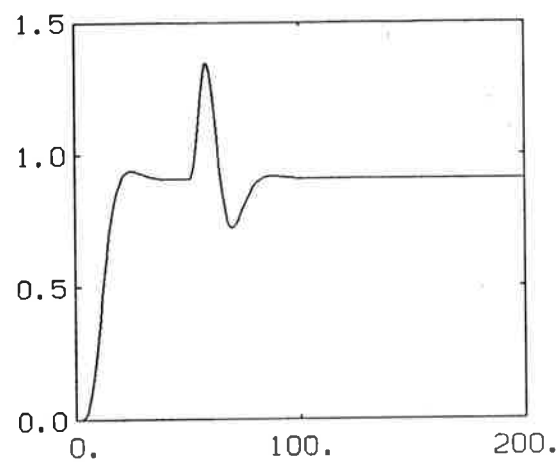
8.1.5 Störning av T-polynomet

Vid störning av T-polynomets sista koefficienten ger detta ett kvarstående fel i utsignalen. Felet blir lika stort antingen man har notch eller dämpningsfilter. Om T-polynomets sista koefficient ökas med 0.01 blir utsignalen cirka 0.1 för hög (ökar från 1.0 till ca 1.1). Om T-polynomets sista koefficient istället minskas med 0.01 blir utsignalen logiskt nog cirka 0.1 för låg. Att avvikelsen blir som beskrivits kan man se i ekvation 7.7 som något omskriven lyder:

$$x = \frac{BT}{AR+BS} \cdot u_c + \frac{BR}{AR+BS} \cdot v$$



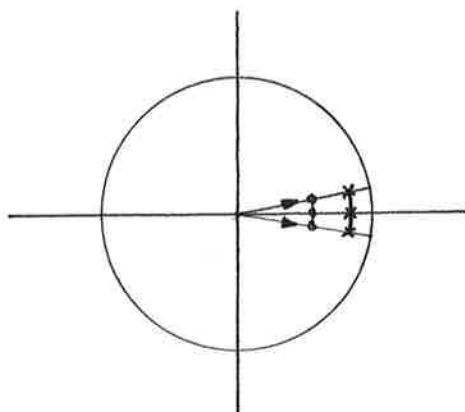
Figur 8.10 Utsignal då T-polynomet störs med +0.01



Figur 8.11 Utsignal då T-polynomet störs med -0.01

8.2 A_m 's återstående poler placeras i båg

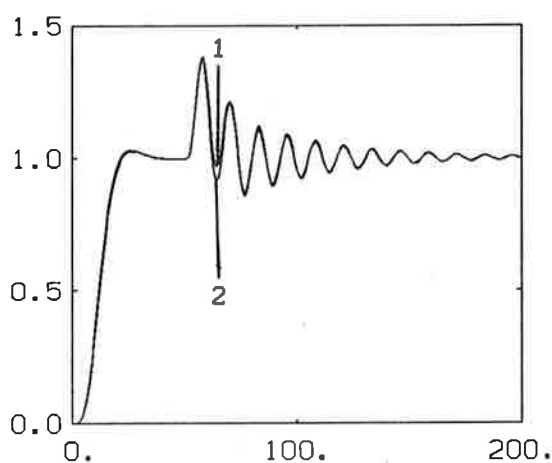
Istället för att placera A_m 's återstående poler längs reella axeln provar vi i detta kapitlet att placera poler utanför denna. Vi låter två av polerna gå från origo mot A_m 's ursprungliga båda poler dvs $0.8261 \pm 0.1492i$. Den tredje polen placeras på reella axeln på samma avstånd från origo de båda andra. Se figur 8.12 nedan.



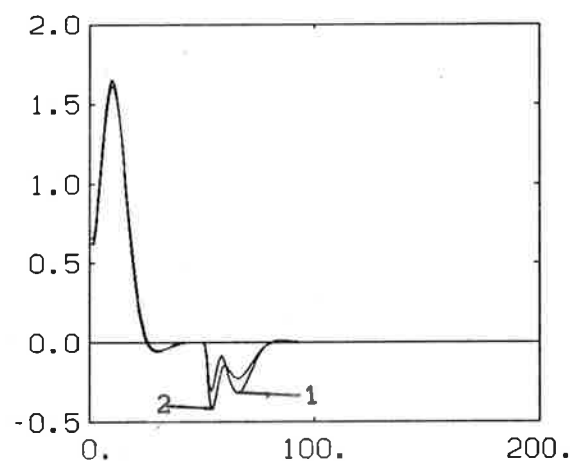
Figur 8.12

8.2.1 Notchfilter

Det visar sig vid simulering att kurvorna för såväl styr- som referenssignal i detta fallet ser nästan exakt likadana ut som kurvorna i avsnitt 8.1 med motsvarande reela värde på α . Bästa kurva för styr- och referenssignal erhålls sålunda för polerna i $0.60 \pm 0.1083i$ samt tredje polen i 0.61. Figurerna 8.13 och 8.14 visar styr- respektive utsignal då polerna placerats enligt ovan. (kurva 1). Som jämförelse har vi lagt in kurva 2 som visar samma då de tre polerna placerats i 0.60.



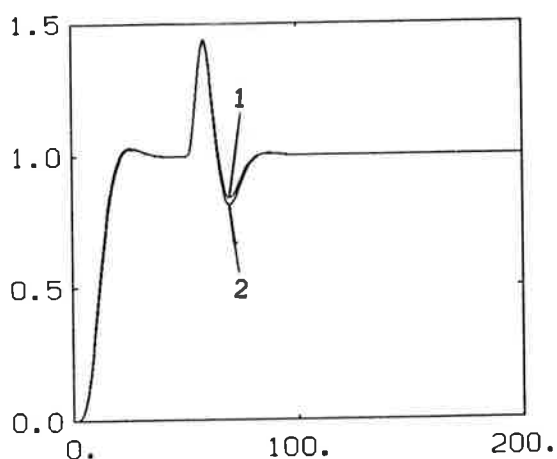
Figur 8.13 Utsignal då polerna placerats i
(1) 0.60 ± 0.1083 och 0.61
(2) alla tre i 0.60



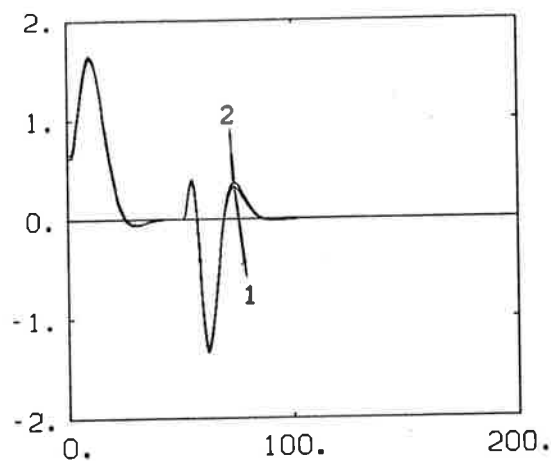
Figur 8.14 Styrsignal då polerna placerats i
(1) 0.60 ± 0.1083 och 0.61
(2) alla tre i 0.60

8.2.2 Dämpningsfilter

Liksom i avsnitt 8.2.1 blir bästa polplaceringen i $0.60 \pm 0.1083i$, en jämförelse med då alla poler placerats i 0.60 ges i figur 8.15 och 8.16. Som synes är det inte mycket som skiljer.



Figur 8.15 Utsignal då polerna placerats i
(1) 0.60 ± 0.1083 och 0.61
(2) alla tre i 0.60

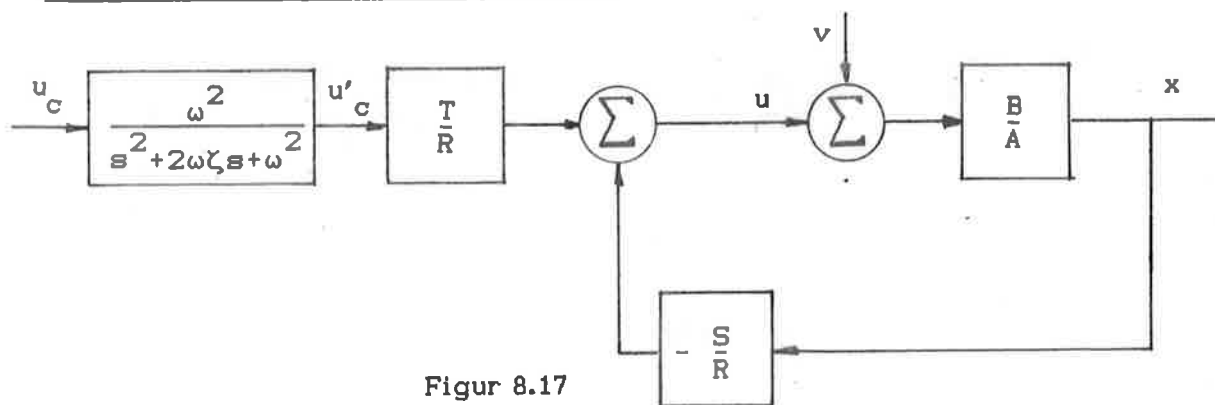


Figur 8.16 Styrsignal då polerna placerats i
(1) 0.60 ± 0.1083 och 0.61
(2) alla tre i 0.60

8.2.3 Störning i An-, Ad- och T-polynomen

Det blir precis samma resultat som i avsnitt 8.1. Se avsnitt 8.1.3, 8.1.4 och 8.1.5.

8.3 Filter lägges efter referenssignalen



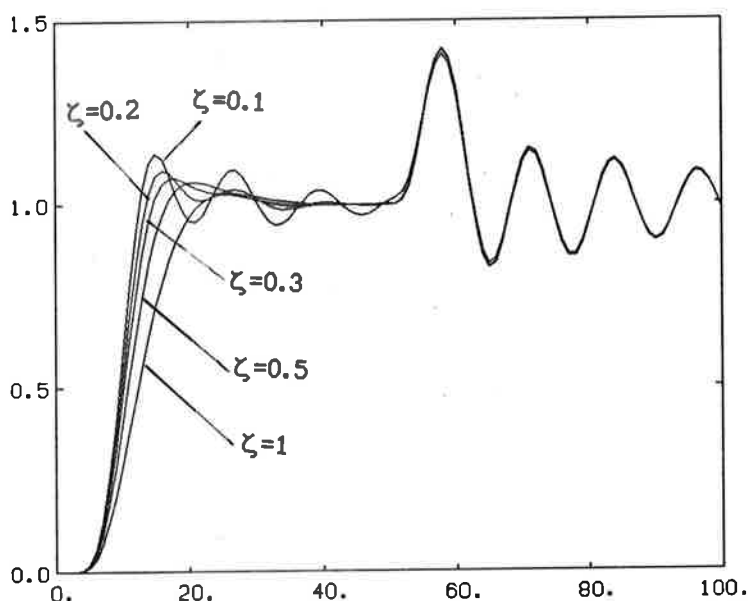
Figur 8.17

Ett annat sätt att få bort "spikarna" i början är att lägga ett filter efter referenssignalen (se figur 8.17 ovan). På så sätt blir "systemets referenssignal" inte så snabb utan förändras lugnare. Hur snabb signalen är bestäms av ω och ζ . Poängen med denna figuration är att man kan tillåta sig att göra H_m snabbare utan att erhålla spikar i styrsignalen vid stegändring i styrsignalen. Vi gör H_m snabb genom att sätta A_m till det vi hade från början

dvs placera tre av polerna i origo.

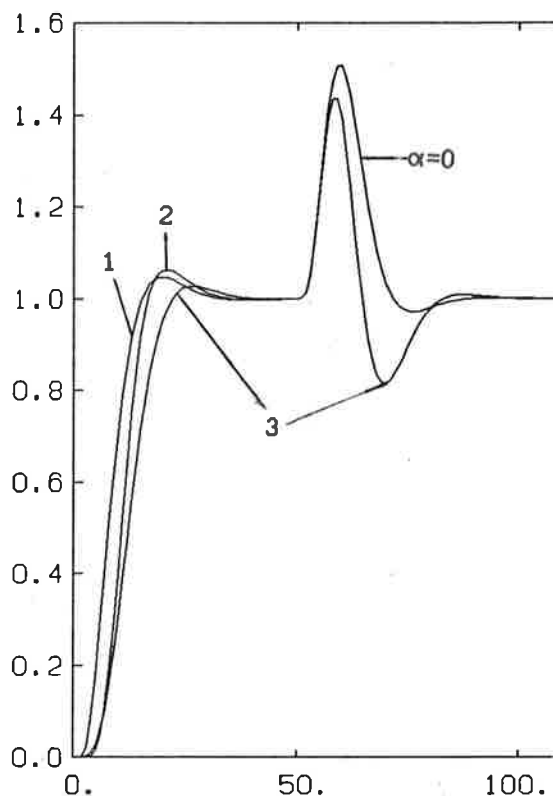
8.3.1 Bestämning av filtrets konstanter

Dämpnings- och notchdesignen uppför sig precis lika vid referenssignalsändring. Det visat sig att det filter som är bäst för det ena fallet också är bäst för det andra fallet. Vi bestämmer därför filtrets utseende i gemensamt kapitel. Stegsvaret för filtret blir snabbare med ökande ω . Vi fann dock vid simulering att om ω sattes större än 1 uppträdde spikar i början i styrsignalen. Vi sätter alltså $\omega=1$. Figur 8.18 nedan visar utsignalen då $\omega=1$ för olika ζ . Vid låga alltför låga ζ blir utsignalen slängig vid stegsvar. Detta beror på att u'_c är slängig vid låga ζ . Vi bedömer att $\zeta=0.5$ är bästa alternativet.

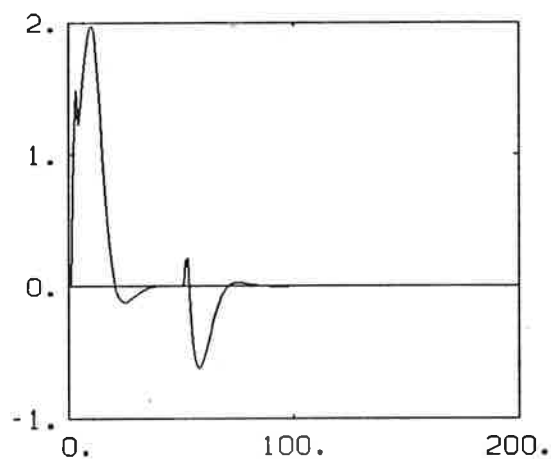


Figur 8.18 Utsignal för olika ζ vid $\omega=1$

Styrsignalen för ovanstående modell ($\omega=1$ och $\zeta=.5$) visas nedan i figur 8.20. Som synes har vi eliminerat spikarna i styrsignalen. Figur 8.20 visar filterfallet jämfört med andra tidigare presenterade förslag. Metoden att ha filter efter referenssignalen visar sig vara effektiv. Systemet blir mycket snabbare än de tidigare presenterade fallen, nästan lika snabbt som det ursprungliga systemet. Styrsignalen håller sig inom rimliga gränser och inga spikar uppträder i styrsignalen. Vid dämpningsdesign blir dessutom "underslängen" mindre i detta fall än då man flyttar A_m 's poler från origo



Figur 8.19 Utsignal då $A_m = (q^2 - 1.6522q + 0.7047)(q - \alpha)^3$
 (1) $\alpha=0$ utan filter
 (2) $\alpha=0$ med filter
 (3) $\alpha=0.60$ utan filter



Figur 8.20 Styrsignal då filter inlagt efter u_c

8.3.2 Störning av An- och Ad-polynomen

Det blir samma resultat som i avsnitt 8.1. Se avsnitt 8.1.3 och 8.1.4.

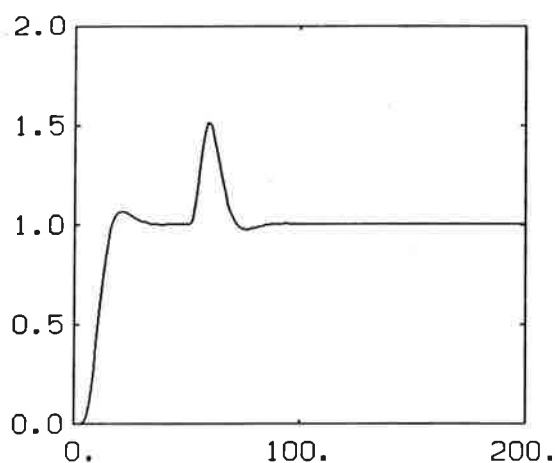
8.3.3 Störning av T-polynomet

Vid störning av T-polynomets sista koefficient ger detta ett kvarstående fel i utsignalen. Felet blir lika stort antingen man har notch- eller dämpningsfilter. Om T-polynomets sista koefficient ökas med 0.01 blir utsignalen cirka 0.01 för hög. Om T-polynomets sista koefficient istället minskas med 0.01 blir utsignalen cirka 0.01 för låg. Att avvikelserna blir som beskrivits kan man se i ekvation 7.7 som något omskriven lyder:

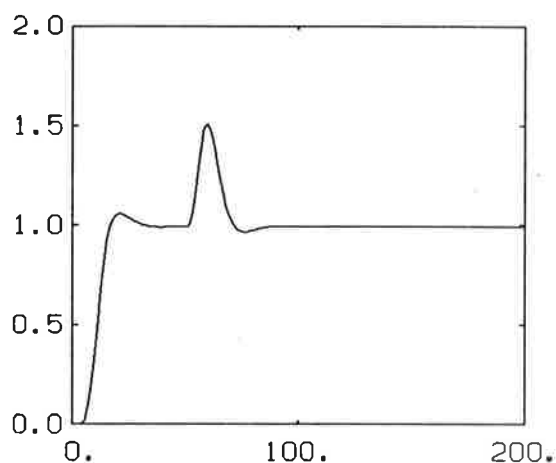
$$x = \frac{BT}{AR+BS} \cdot u_c + \frac{BR}{AR+BS} \cdot v$$

$$\text{där } AR+BS = A_m A_n A_o$$

Om man jämför avvikelsen i detta avsnitt med avvikelsen i avsnitt 8.1 är avvikelsen här cirka 1/10 av avvikelsen i 8.1. Det enda som skiljer är A_m och det måste vara den som gör att utsignalen blir mindre känslig för störningar i detta fallet.



Figur 8.21 Utsignal då T-polynomet störs med +0.01



Figur 8.22 Utsignal då T-polynomet störs med -0.01

8.4 Slutsats

Det bästa sättet som vi kommit på att ta bort spikarna i styrsignalen är att lägga ett filter efter styrsignalen enligt avsnitt 8.3. Att den metoden är överlägsen de andra syns tydligt i figur 8.19. Den är snabbare än de båda andra fallen utan att man får spikar i styrsignalen. Dessutom är denna metod mindre känslig för störningar i T-polynomet.

9. SLUTLIG SAMMANFATTNING

Då vi använder notchfilter uppträder systemet väldigt svängigt vid momentstörningar. Anledningen till detta är att A_n -polynomet har sina poler väldigt nära enhetscirkeln. Det visar sig också i avsnitt 7.1.3 att A_n -polynomet är väldigt känsligt för avvikelser då polerna flyttas ut mot enhetscirkeln. Om polerna flyttas ut i storleksordningen 0.02 hamnar polerna utanför enhetscirkeln och systemet blir instabilt. Att utsignalen inte oscillerar vid referenssignalsändring syns i ekvation 7.5. Den visar att varken A_n eller A_o påverkar utsignalen vid referenssignalsändring.

Då vi i stället använder dämpningsfilter uppträder systemet mycket lugnare vid momentstörning. Dämpningspolynomets poler ligger på behörigt avstånd från enhetscirkeln därför är polynomet ej känsligt för störningar.

Då vi i kapitel 7 simulerar ut- och styrsignal för notch- och dämpningsfilter får vi oacceptabla spikar i initialskedet på styrsignalen. Ekvationerna 7.4 och 7.6 lyder:

$$u = \frac{t_0 A}{A_m} \cdot u_c - \frac{BS}{A_m A_n A_o} \cdot v \quad (7.4)$$

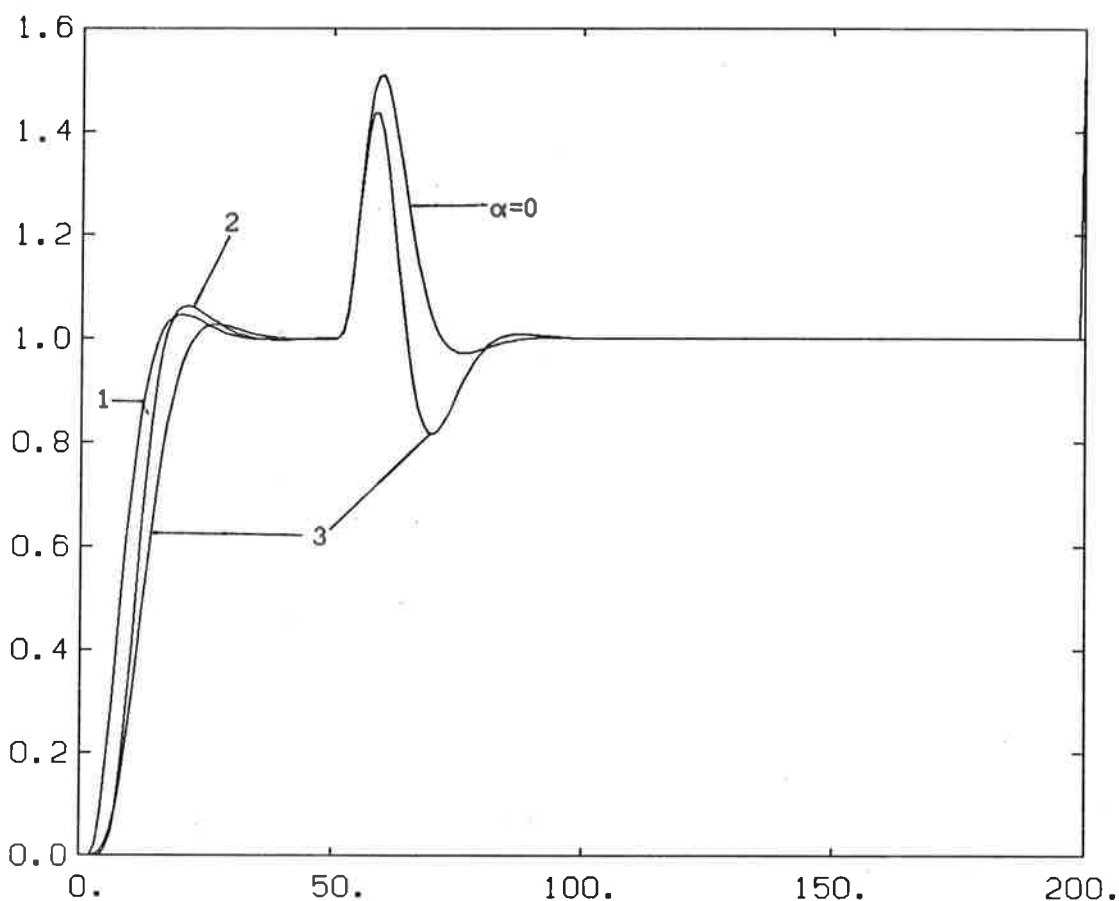
$$u = \frac{t_0 B}{A_m} \cdot u_c + \frac{BR}{A_m A_d A_o} \cdot v \quad (7.6)$$

Här är det termen framför u_c som är intressant. Vi kan inte ändra på t_0 eller A , alltså var vår enda chans att ändra på A_m . A_m var ursprungligen $(q^2 - 1.6522 + 0.7047)q^3$. Vi provar först att flytta ut A_m 's poler från origo till någonstans på den reela axeln. Vi kommer fram till att $(q^2 - 1.6522 + 0.7047)(q - 0.6)^3$ är bästa lösningen i detta fall. Nackdelen jämfört med ursprungliga fallet är att stegsvaret blir långsammare och att underslängen efter störningen blir större.

Vi provar också att placera A_m 's resterande poler i en "båge" enligt avsnitt 8.2. Bästa alternativet i detta fall blir då polerna placeras i $0.60 \pm 0.1083i$ och 0.61. Uppträdandet blir nästan identiskt med då alla polerna placerats i 0.60. Enda skillnaden är att underslängen efter störningen blir något mindre.

Till sist kom vi på den bästa lösningen: att lägga ett filter mellan referenssignalen och systemet. Vi gjorde systemet snabbare genom att placera A_m 's poler i origo. Vi fick inga spikar i styrsignalen vid

referenssignalsändring tack vare att referenssignalen uppträder lugnare efter filtret. Denna figuration med filter är snabbare än om man placerar alla de tre polerna i 0.60. Den blir nästan lika snabb som i det ursprungliga fallet (se figur 9.1). Underslängen blir i "filterfallet" mindre än om man flyttar A_m 's poler från origo. Figur 9.1 visar utsignalen i några olika fall.



Figur 9.1 Utsignal då $A_m = (q^2 - 1.6522q + 0.7047)(q - \alpha)$
 (1) $\alpha=0$ utan filter
 (2) $\alpha=0$ med filter
 (3) $\alpha=0.60$ utan filter

Det visar sig också att "filtermodellen" är mindre känslig för störningar i T-polynomet.

Vi har även försökt få ner utsignalens avvikelse vid momentstörning. I analogi med att vi varierade A_m för att förändra styrsignalen vid stegsvar borde det enligt ekvation 7.5 och 7.6 gå att förändra utsignalens utseende vid momentstörning genom att variera A_o . Våra simuleringar visade emellertid att det knappast inte hade någon som helst betydelse var vi placerade observerarpolerna. Detta beror antagligen på att R- och S-polynomen

framräknas med hjälp av bl.a. A_o .

Som redan nämnts i avsnitt 8.4 anser vi att det bästa sättet att reglera den beskrivna processen att ha notchfilter, placera tre av A_m 's poler i origo och ha ett filter efter referenssignalen.

LITTERATURFÖRTECKNING

Wittenmark, B & Aström, K J: Computer Controlled Systems.

Wittenmark, B: Analysis and design of control systems.

Aström, K J: Reglerteori.

CTRL-C manual.