



LUND UNIVERSITY

Processidentifiering - Projektarbeten hösten 1990

Johansson, Rolf; Bernhardsson, Bo; Dahl, Ola

1990

Document Version:
Förlagets slutgiltiga version

[Link to publication](#)

Citation for published version (APA):

Johansson, R. (Red.), Bernhardsson, B., & Dahl, O. (1990). *Processidentifiering - Projektarbeten hösten 1990*. (Technical Reports TFRT-7473). Department of Automatic Control, Lund Institute of Technology (LTH).

Total number of authors:

3

General rights

Unless other specific re-use rights are stated the following general rights apply:

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

Read more about Creative commons licenses: <https://creativecommons.org/licenses/>

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

LUND UNIVERSITY

PO Box 117
221 00 Lund
+46 46-222 00 00

CODEN: LUTFD2/(TFRT-7473)/1-128/(1990)

Processidentifiering Projektarbeten hösten 1990

Rolf Johansson (Red.)
Bo Bernhardsson
Ola Dahl

Institutionen för Reglerteknik
Lunds Tekniska Högskola
December 1990

Department of Automatic Control Lund Institute of Technology P.O. Box 118 S-221 00 Lund Sweden		<i>Document name</i> REPORT	
		<i>Date of issue</i> December 1990	
		<i>Document Number</i> CODEN:LUTFD2/(TFRT-7473)/1-128/(1990)	
<i>Author(s)</i> Rolf Johansson, Bo Bernhardsson, Ola Dahl (Eds.)		<i>Supervisor</i>	
		<i>Sponsoring organisation</i>	
<i>Title and subtitle</i> Processidentifiering — Projektarbeten hösten 1990 (Term papers in Process identification — Fall 1990)			
<i>Abstract</i> <p>This volume contains some term papers of the students taking the graduate course in process identification. The following identification subjects have been studied: Core stability of a boiler water reactor; Parametric robot identification; A salt mixing process with recirculation; Modeling and control of the ball and beam process; Dynamics of the "ball and hoop" apparatus (TecQuipment); Identification of a robot link; Identification of a servo with a flexible coupling;</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, English	<i>Number of pages</i> 128	<i>Recipient's notes</i>	
<i>Security classification</i>			

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

FÖRORD

Föreliggande arbete innehåller några rapporter från projektarbeten inom en kurs (ht 90) i processidentifiering vid Institutionen för Reglerteknik, LTH. Kursen *Processidentifiering* gavs första gången under höstterminen 1987 såsom reguljär fortsättningskurs och forskarutbildningskurs vid Reglerteknik, LTH. Kursdeltagarna förutsättes ha goda förkunskaper i reglerteori och stokastiska processer svarande mot institutionens kurs "Digital Reglering". Under kursens gång har kursdeltagarna förberetts för sina uppgifter genom fyra laborationer: Frekvensanalys, Interaktiv identifiering, Realtidsidentifiering, Syntesorienterad identifiering. Vidare har kursen innehållit övningar med Matlab-uppgifter i Processidentifiering.

Uppgifterna har alla omfattat identifiering med jämförelse mellan några olika metoder. Följande identifieringsproblem har studerats:

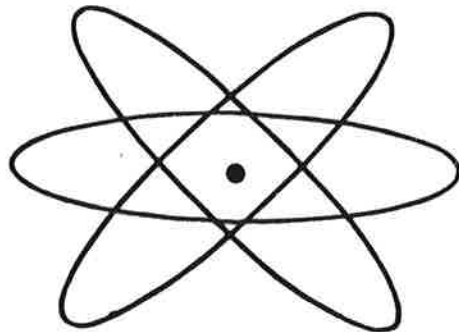
- Härddynamik hos kokarvattenreaktor (P. Abrahamsson, P. Hallgren)
- Parametrisk robotidentifiering (K. Fredriksson, M. Larsson)
- Dynamik hos saltkoncentrationsreglering med återflöde (F. Hermodsson, H. Svegerud)
- Modeling and control of the ball and beam process (U. Jönsson, H. Olsson)
- Ekorrhjul (K. Nilsson, I. Tamras)
- Identifiering av robotaxel (P. Nilsson)
- Identifiering av servo (A. Ringdahl)

Det är vår förhoppning, att dessa projektarbeten kan tjäna som exempel på tillämpningar av identifieringsmetodik.

Rolf Johansson, Bo Bernhardsson, Ola Dahl, (Red.)

ProcessIdentifiering

Projektarbete hösten 1990



Peter Hallgren

Per Abrahamsson

Handledare: Rolf Johansson

Institutionen för Reglerteknik

Lunds Tekniska Högskola

November 1990

INNEHÅLLSFÖRTECKNING

1	Inledning	
2	Processen	
	2.1 Bakgrund	3
	2.2 Systemet	6
	2.3 Mätsignalen	8
3	Identifiering	
	3.1 Allmänt	12
	3.2 AR-modell	14
	3.3 ARMA-modell	15
	3.4 Validering	16
4	Sammanfattning	17

REFERENSER

APPENDIX

1. INLEDNING

Avsikten med projektarbetet i Processidentifiering är att tillämpa teorin på någon av skolans processer. Då vi har valt att göra examensarbete inom process-identifiering vid samma tidpunkt, föll det sig naturligt att låta projektarbetet ingå som en del av detta examensarbete. Det medför att vår rapport blir en redovisning av problemet vi ställts inför, samt de försök och resultat som vi uppnått hittills.

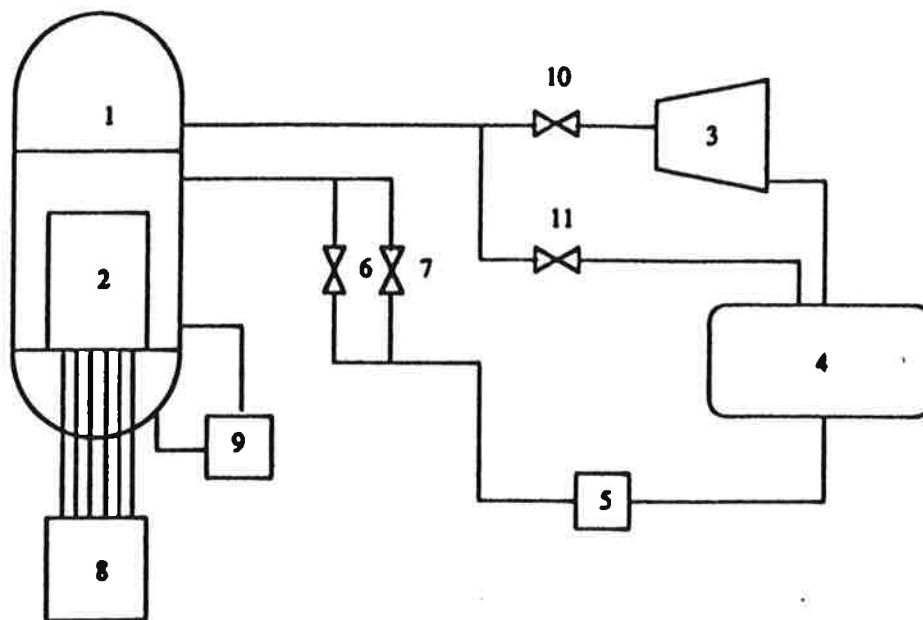
I vårt examensarbete ska vi göra en rekursiv skattning av processen och arbetet hittills har varit "off-line" för att skaffa oss så mycket förkunskaper om processen som möjligt. Det är dessa off-line identifieringar som vi redovisar som projekt-arbete. Målsättningen i allmänhet för projektarbetet är att försöka anpassa en modell så bra som möjligt, mha. två olika skattningsmetoder. Våra förutsättningar gör det svårt att kontrollera om modellen är "bra", se kap. 2, utan vi redovisar dels våra kriterier samt ett försök till uppfattning om "bra" modell.

Vår uppgift i examensarbetet är att göra en rekursiv identifiering av härdynamiken hos kokvattenreaktorer (BWR) hos ABB-ATOM i Västerås. Resultatet ska användas till en förbättrad stabilitetsmonitering som idag görs med en icke-parametrisk metod. Efter val av algoritm o.dyl. ska det implementeras i C (VENIX) och uttestas i realtidsmiljö.

2. PROCESSEN

2.1 BAKGRUND

För att bättre förstå problematiken, ska vi försöka att, kortfattat, beskriva en kokarreaktors uppbyggnad och dess funktion med avseende på stabilitetsegenskaper. Se figur 2.1



Figur 2.1 Principschema för kokvattenreaktor (BWR).

I reaktortanken (1) finns reaktorns härd i form av bränsleelement (2) av uran. Värmeutvecklingen i bränslet kan regleras med styrstavar (8) och huvudcirkulations-pumpar(9). Bränslet kyls med vatten som strömmar förbi bränsleelementen. Pga den stora värmeutvecklingen förångas kylmediat (vattnet) , och ångan som bildas leds ut genom ledningar i reaktortankens övre del .

Flödet av den heta ångan (ca280 grader), når turbinanläggningen (3) och avger sin energi till turbinerna. Dessa driver elgeneratorer som avger elenergi med en spänning på ca 2000 Volt. När ångan har passerat turbinen, strömmar den in i kondensorn (4), där den kyls och övergår i vatten.

Vattnet kallas nu matarvatten och en pump (5) pumpar in det i reaktorn igen. Innan vattnet når reaktorn passerar det två ventiler, avstängningsventilen (6) och reglerventilen (7). Det är viktigt att den mängd ånga som lämnar reaktorn ersätts med lika mycket vatten, för att bibehålla en, i stort sett, konstant nivå.

Trycket i reaktorn hålls konstant genom att ångflödet ändras med pådragsventiler (10) och dumpventiler (11). Pådragsventilerna styr flödet till turbinerna och dumpventilerna gör det möjligt att dumpa ånga till kondensorn, utan att ångan passerar turbinen.

Regleringen av effekt sker genom ändring av neutronflödet i härden. Det kan regleras dels mha styrstavar och dels med hjälp av vattenflödet genom reaktorn. När styrstavarna (8) förs in i härden ökar absorberingen av neutroner och effekten minskar. Med nytt bränsle är styrstavarna inskjutna ca 60% och de dras kontinuerligt ut, i takt med att bränslet förbränns, för att hålla mängden aktivt bränsle konstant. Detta, tillsammans med nödstopp, är styrstavarnas huvudsakliga funktion.

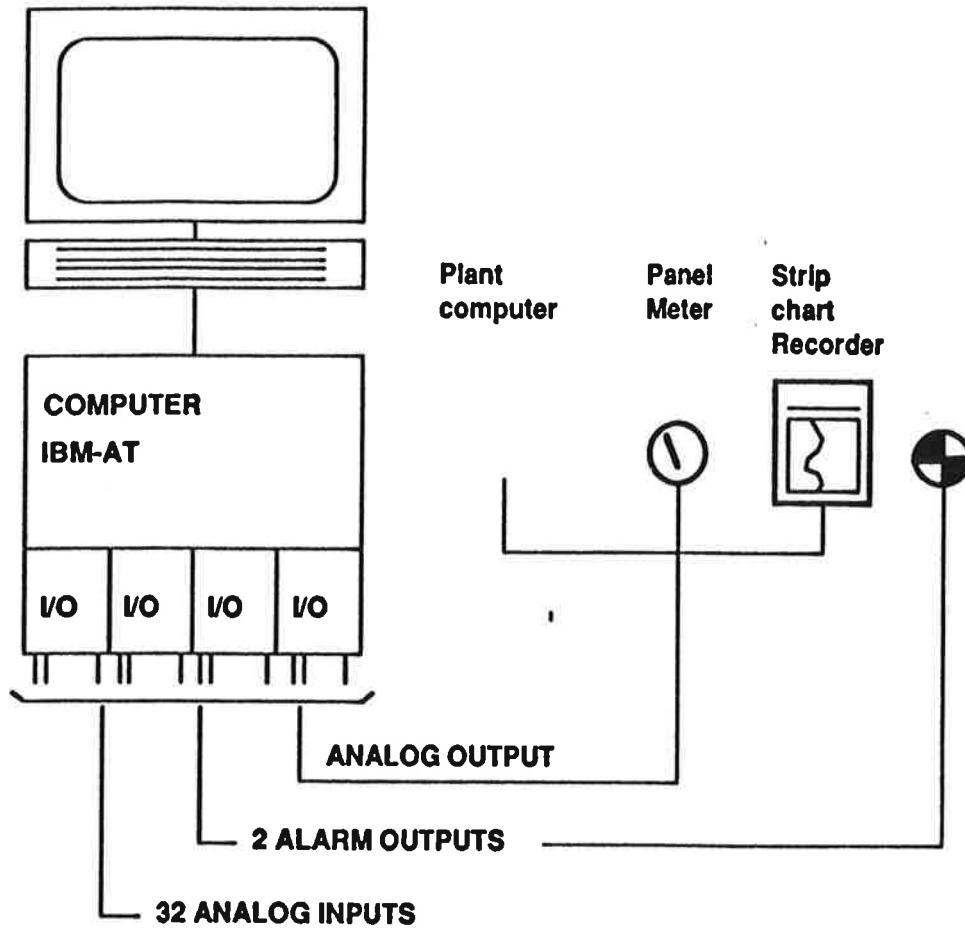
Vid normal drift regleras neutronflödet främst genom ändring av vattenflödet i reaktorn i samband med effektuttags-förändring. Med huvudcirkulationspumparna (9) cirkuleras en blandning av vattnet i reaktorn och inkommande matarvatten. På så sätt ändras ånghalten i härden. När effekten minskar, minskas cirkulationen, vilket medför att ånghalten ökar och neutronernas hastighet ökar. Det är främst termiska neutroner som ger fission i denna typ av reaktor. Genom ändring av pumparnas varvtal, kan man reglera ca 40% av effekten, medan resterande 60% regleras med styrstavarna.

När det gäller instabilitetsmekanismer, finns det ett antal som är intressanta i reaktorsammanhang. Av dem kan Xenon-stabilitet, flödesexkursion, "täthetsvåg"-instabilitet, parallellkanal-instabilitet, samt reaktordynamisk instabilitet nämnas. Den sistnämnda är den vanligaste instabilitetsmekanismen i BWR-reaktorer.

On-line-identifiering av dynamiska systemegenskaper tillämpas vid s.k stabilitetsmonitoring i kokarreaktorer. COSMOS (Core Stability Monitoring System) är en utrustning med snabb respons, som kontinuerligt övervakar härdstabiliteten i härden (se figur 2.2). Systemet samlar in mätdata från 32 kanaler, analyserar dessa och bestämmer dämpfaktor, brusnivå och resonans-frekvenser.

COSMOS

COre Stability MOnitoring System



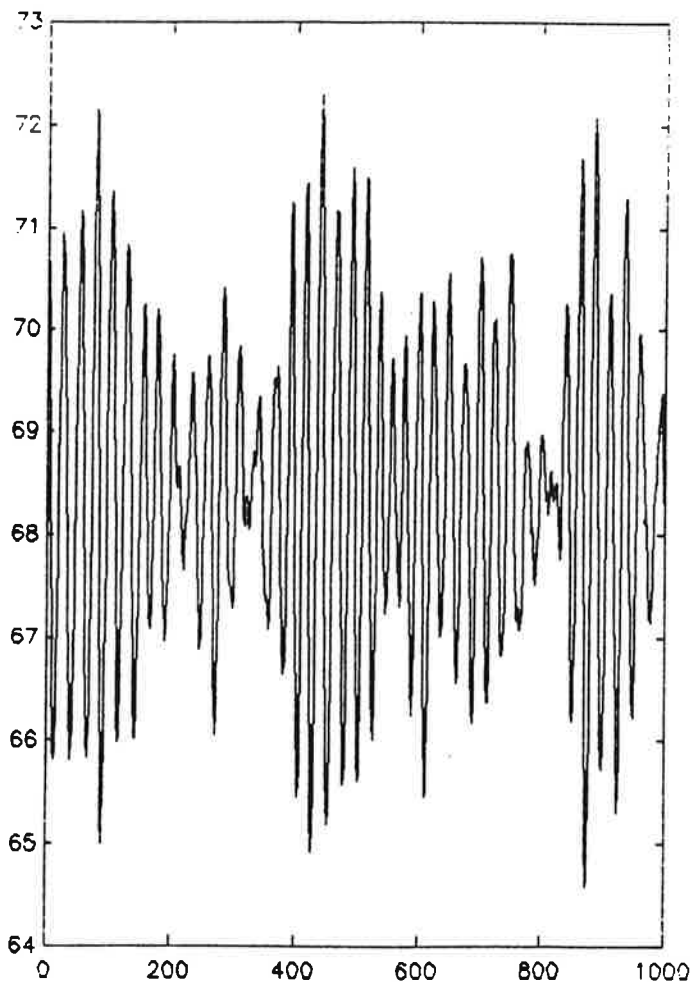
Figur 2.2 principskiss över moniterningsystemet, COSMOS

2.2 SYSTEMET

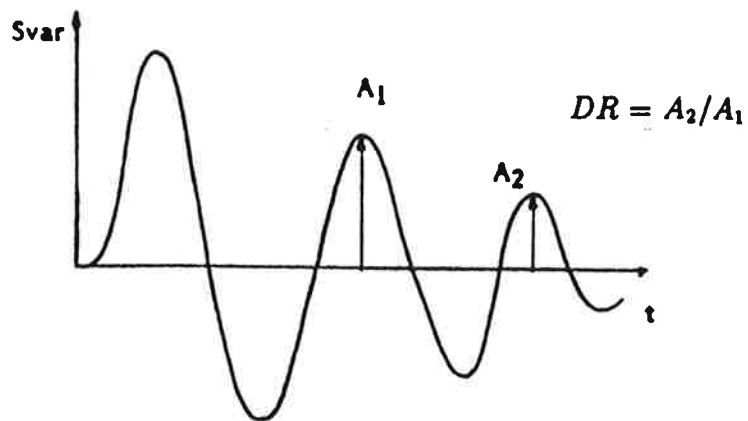
Stabilitetsmonitorn mäter på reaktordynamisk instabilitet. Signalen som erhålls ur en sådan mätning är en "brussignal" (Se figur 2.3), vilket innebär att modellen får väljas som en AR eller ARMA-modell. Det innebär svårigheter med validering då ingen insignal finns. Processen är dessutom återkopplat och välreglerad, vilket också det innebär problem med identifieringen (Se R.Johansson, kap 9.2) .

Av tradition i kärnkraftssammanhang är man intresserad av att beräkna dämpkvoten, DR (Decay Ratio). DR definieras som amplitudförhållandet mellan två på varandra följande toppar i impulsvaret .

Det finns olika sätt att räkna ut dämpkvoten. Ett sätt är att ur systemets utsignal skatta dess autokorrelationsfunktion (impulssvar), då det exiteras av vitt brus och därur beräkna dämpkvoten (ger "momentan" DR, om de två första används), se figur 2.4.



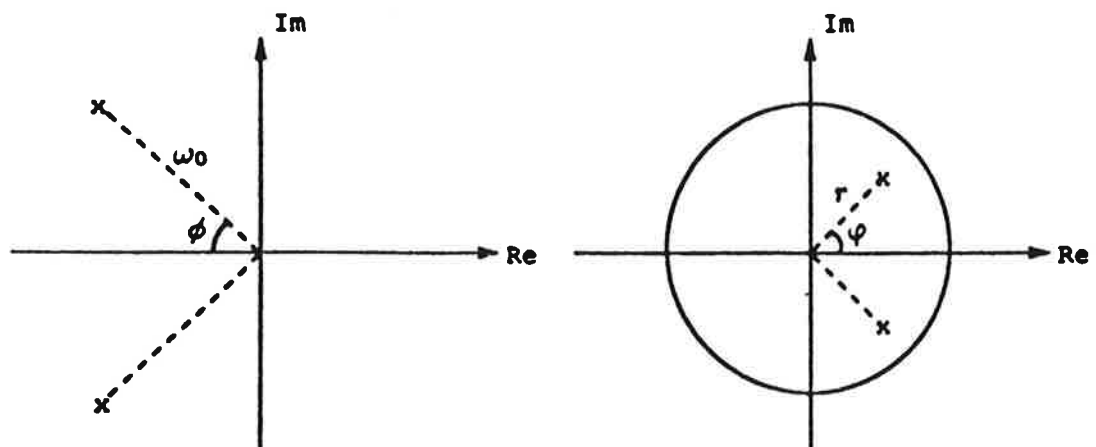
Figur 2.3 Neutronflödet i härden vid normal drift



Figur 2.4 Dämpkvoten (Decay Ratio)

Ett annat sätt, som vi ska använda oss av, är att ur systemets poler beräkna dämpkvoten analytiskt. Det ger en "asymptotisk" DR, om det dominerande polparet används. Se figur 2.5. Man kan visa att dämpkvoten för ett andra ordningens system fås ur:

$$DR = e^{2\pi \ln r/\phi}$$



Figur 2.5 Tidskontinuerliga och tidsdiskreta poler för andra ordningens system

I den nuvarande utrustningen beräknas dämpkvoten ur auto-korrelationen, dvs icke-parametriskt. Det ger bra noggrannhet för den momentana dämpkvoten, men sämre för den asymptotiska. Syftet är att se om man kan anpassa en modell med en parametrisk metod för att uppnå bättre noggrannhet. Målet för examensarbetet är att implementera en rekursiv algoritm, som ska klara ett antal punkter :

- Skatta medelvärde rekursivt
- Bestämma modellordning on-line
- Identifiera systemets parametrar
- Beräkna dämpkvoten (Decay Ratio)

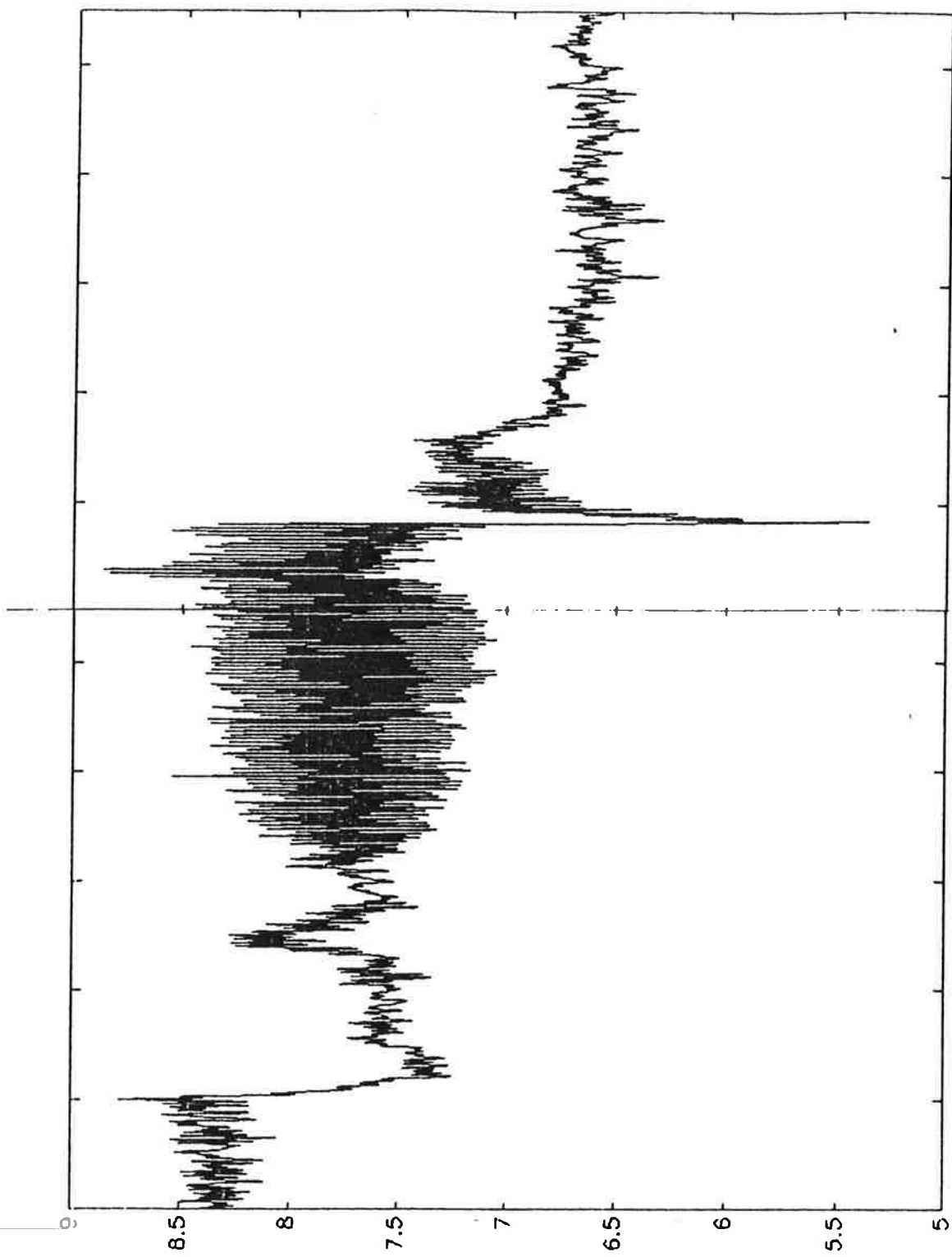
Målet för projektarbetet blir att med hjälp av matlab försöka identifiera systemet off-line och belysa svårigheterna med offset, varierande modellordning och annat som uppstått under identifieringsarbetet.

Då signalen har en offset som varierar ganska kraftigt samt att processen är tidsvariant (modellordningen varierar för olika driftsituationer, och därmed dämpkvoten), ställer det höga krav på algoritmen. Den ska dessutom vara tillräckligt snabb för att presentationen av dämpkvoten inte ska bli fördröjd.

2.3 ANALYS AV MÄTSIGNALEN

För att skaffa oss lite förkunskaper om systemet innan identifieringen började vi med att titta på signalens spektrum. Ett problem är att signalerna är så olika vad gäller offsetnivå, samplingsfrekvens och dämpkvot. De mätdata vi har tittat på är dels ett par "snälla" dvs med konstant offset-nivå och DR, dels ett par extremfall som innefattar bla ett nödstops-situation med stora variationer i offset och DR. (se figur 2.6) .

Det finns två sätt att behandla offset. Dels kan man skatta det, explicit eller med utökad modell, dels kan man använda differentierad modell (se Söderström-Stoica, kap 12.3). I matlab har vi åtgärdat detta genom 'detrend', som gör en medelvärdes-beräkning av signalen och sedan drar bort den. Därför har vi hittills bara tittat på avsnitt av signaler med konstant offset. Så småningom ska offseten skattas rekursivt och då har vi resultat att jämföra med när kvaliteten av den skattade offseten ska avgöras

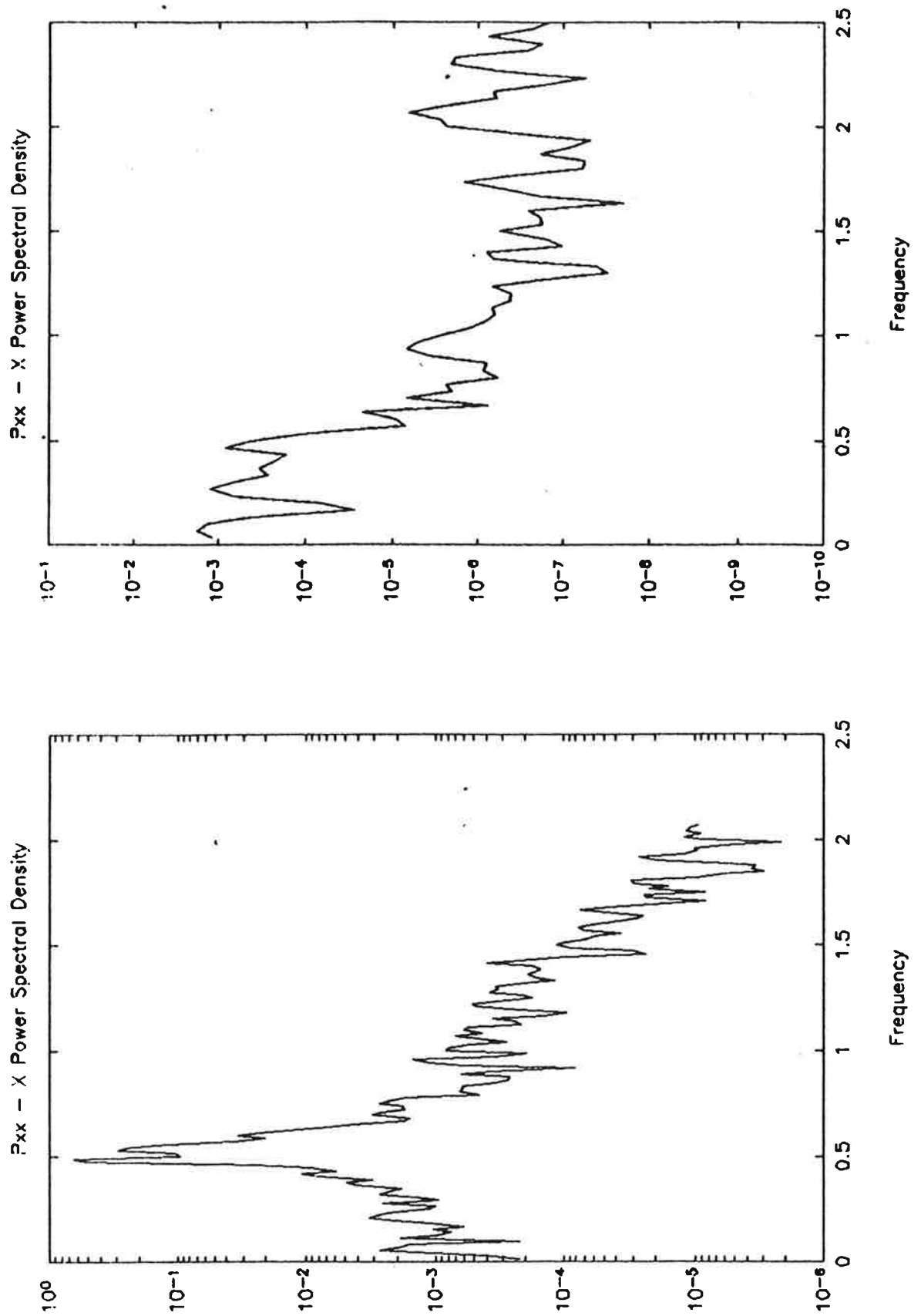


Figur 2.6 neutronflöde i simulerad nödstoppsituation

Ur signalernas spektrum ser man att resonansfrekvensen ligger runt 0.5 Hz (0.4-0.7) och är mer eller mindre dominerande. Man ser även att det finns andra resonansfrekvenser med samma styrka eller lägre, vid andra, intilliggande, frekvenser. Detta visade sig vara dynamik från de regelsystem som finns i reaktorn och innebär naturligtvis en risk vid identifieringen. Dessa "regulatorfrekvenser" hade en tendens att ändras med ändrad dämpkvot och dess frekvensvariationer var betydligt större än neutronflödesdynamikens, som var förvånansvärt stabil. Ett instabilare system innebär en större regulator-aktivitet, se figur 2.8-10

Signalerna samplas i verkligheten med 8 Hz vilket innebär att ett onödigt stort frekvensområde används vid modellanpassningen, informationen finns ju runt 0.5 Hz. Vi har därför valt att sampla ned signalen 2 ggr vilket ger en nyquistfrekvens på 2 Hz, vilket borde räcka. Vi får också mer tid för uppdateringar i den framtida rekursiva skattningen. Frågan är då hur mycket en nedsampling påverkar signalen och dess spektrum.

Nedsamlingen gjordes i MATLAB med 'Decimate'. För att undvika vinkningsfenomen måste signalen filtreras först och det görs i 'decimate' med ett 30:e ordningens FIR-filter. Ett FIR-filter innehåller visserligen bara nollställen, men det är ingen garanti för att man inte riskerar att införa ny dynamik i systemet. Det är dessutom svårt att kontrollera vid en AR-situation eftersom vi inte har någon insignal att prova modellen med. Det visar sig att signalens utseende inte förändras (synbart åtminstone) men dess spektrum påverkas dock en aning av detta filter.



Figur 2.7 Spektrum för signalen i fig. 2.3 (övre) , samt signal med tydlig påverkan av regulatorndynamik (undre).

3. IDENTIFIERING

3.1 ALLMÄNT

En anledning till de många skattningarna i matlab, är att försöka få så mycket förkunskaper som möjligt om processens uppförande, modell-ordning, dominerande frekvenser, robusthet men framförallt om dämpkvotens beteende för olika modellstrukturer och skattningsmetoder.

De modeller som vi försökt anpassa är dels AR-och dels ARMA. En av förutsättningarna var ett antagande om att systemet drevs av vitt brus, dvs att en AR-modell skulle räcka och frågan var om detta antagande var tillräckligt. Det visade sig vara delvis sant (se kap 3.2) .

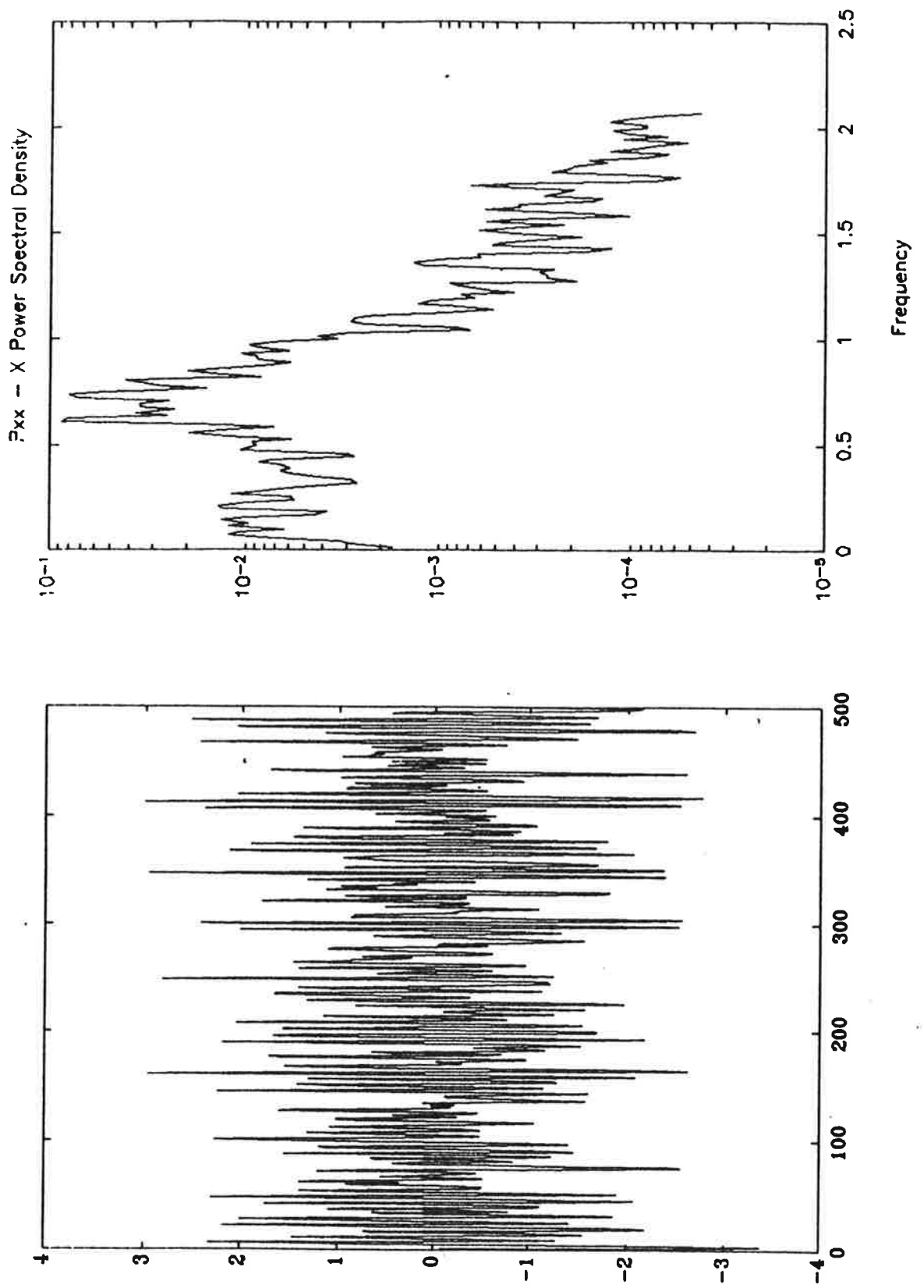
De verifieringsmetoder vi har kunnat utnyttja, har varit residual-test, Akaikes' FPE samt variansernas storlek (se kap. 3.4).

För att modellera ett system med fler dominerande resonansfrekvenser, krävs i allmänhet en modell av hög ordning. I vårt fall låg dessa frekvenser väldigt nära varandra, vilket gav lite olika resultat. Problemen uppstod när skillnaden i amplitud minskade.(ofta var den dominerande frekvensen mycket kraftigare i amplitud och då uppstod inget problem). Det krävdes ofta en ordning 4 eller 5 men sällan högre.

I fallen med stor amplitudskillnad räckte en modellordning på 2-3. I de fallen där skillnaden i amplitud och frekvens var liten, räckte även här en modell-ordning på 2-3

Genomgående visar det sig att dämpkvoten är ganska oberoende av om vi modellerar hela "sanningen" eller inte.

Exemplet som visas nedan är uppmätt neutronflöde vid lugn och normal drift, se figur 3.1. Man ser att bodeplotarna är likartade, se appendix A och att ordningarna blir ungefär de samma för AR och ARMA. Därför borde AR vara en fullgod modell, men skillnaden ligger i de alltför varierande dämpkvoterna i AR-modellen. Problemen med bestämning av AR-modell, ökar vid besvärligare driftslägen.



Figur 3.1 Exempel på mätdata med spektrum (överst), vid normaldrift

3.2 AR-anpassning

En AR-modell beskriver processen som:

$$A(q) y(t) = e(t) \quad ; e(t) \text{ vitt brus}$$

Resultatet av ett antal identifieringar med olika modellordning, är att det går att anpassa en AR-modell till processen men den är beroende av exakt "rätt" modellordning. Detta ställer stora krav på en bestämning av rätt modellordning och tittar man på de kriterier som står till buds tex. Akaikes' FPE, inser man att det kan bli svårt on-line, se tabell 1.

Genomgående är AR-modellerna mer varierande i kvalitet vad det gäller stabila dämpkvoter och frekvenser. Ofta var det svårt att med hjälp av gängse kriterier avgöra om en modell var bra, utan det blev ofta en kompromiss mellan stabil dämpkvot, parametrarnas varianser och dominerande frekvens. Då vi främst är intresserade av dämpkvoten, är dess känslighet för modellordning av stort intresse, och tabell 1 visar att även den varierar för olika modellordningar.

MODELLORDNING	AKAIKES' FPE	DOMINERANDE FREKVENNS	DÄMPKVOT
1	0,9597	-	-
2	0,3218	0,67	0,53
3	0,2951	0,72	0,54
4	0,2957	0,72	0,45
5	0,2960	0,71	0,59
6	0,2948	0,73	0,54
7	0,2903	0,77	0,32
8	0,2907	0,69	0,41
9	0,2924	0,69	0,43

TABELL 1. AR-identifiering som ger stora variationer i dämpkvoten.

Man ser även hur lite information FPE ger för val av modellordning

Kommandot AR i MATLAB skattar modellen med en minsta-kvadratmetod. En sådan är känslig för "outliers", eller abnorma mätvärden. Detta faktum tillsammans med antagandet om vitt brus ger en känslig skattning. I appendix A finns Bodeplot samt residualtest för ordning 3.

3.3 ARMA-anpassning

Nästa steg var att undersöka om en ARMA-modell gav ett bättre resultat. En ARMA modell modellerar bruset med C-polynom:

$$A(q) y(t) = C(t) e(t)$$

där $e(t)$ är vitt brus.

Pga det färgade bruset kan man inte använda MK-metoder utan att få systematiska fel i parametrarna (se Johansson R. kap 4.2). I MATLAB används en robustifierad PEM iterativ skattningsmetod, med möjlighet att påverka robustifiering o.dyl. Vi identifierade här samma mätdata som för AR och upptäckte en del skillnader.

Tabell 2 visar att modellerna blir betydligt bättre både vad det gäller parametrarnas varianser och dämpkvot. Den stora förbättringen är dämpkvotens okänslighet för modellordning. Intressant var att dämpkvoten oftast blev bra även för modeller med tveksamma varianser och residualer. Än en gång, vi är ju inte i första hand intresserade av att till varje pris beskriva en korrekt modell utan det är dämpkvotens uppförande som är det primära.

För $C=1$ fick vi ett AR-liknande uppförande dvs. svängigare dämpkvot och färre bra modeller. För $C=2-3$ fick vi fler bra modeller och stabilare dämpkvot. Då C ökades till 4 blev det sämre igen.

Vi har gjort ett antal ARMAX-identifieringar för lite olika reaktorer och driftfall. Dessa visar att modellordningarna, som nämnts ovan, gäller även för andra mätserier.

Då signalerna innehåller fler resonanstoppar än den intressanta, krävs en hög ordning för att modellera dem. En lägre ordningens modell innebär en approximation, där högsta toppen modelleras med ett polpar. Det gav förvånansvärt stabila frekvenser och polpar. Vi fick inga märkbart bättre resultat/dämpkvoter för modeller med högre ordning som försökte modellera fler frekvenser.

MODELLORDNING		AKAIKES' FPE	DOMINERANDE FREKVENNS	DÄMPKVOT
nA	nC			
2	2	0,2958	0,70	0,44
3	2	0,2946	0,72	0,45
4	2	0,2911	0,71	0,50
5	2	0,2902	0,72	0,51
6	2	0,2915	0,73	0,47
7	2	0,2918	0,71	0,41
8	2	0,2895	0,70	0,38
2	3	0,5070	0,72	0,51
3	3	0,4863	0,72	0,49
4	3	0,4902	0,73	0,49
5	3	0,4713	0,73	0,47
6	3	0,5201	0,72	0,52

TABELL 2. ARMA-identifiering för samma signal. Stabilare dämpkvot och frekvens än för AR-modellen. Samma uppförande för FPE.

3.4 Validering

De valideringsmöjligheter som finns är Akaikes' FPE, residualtest, parametrarnas storlek och varians samt simulering (se Johansson, kap 10). Vi har använt oss av alla utom simulering, som är lite svår att genomföra då en AR-modell inte har någon insignal.

Vi gjorde försök med steg och impulssvar för de skattade modellerna men de gav ingen ny information. Akaikes' FPE gav väldigt lite, då dess ändring var försumbar utom för extremt låga ordningar. Residualtest gav lite mer information om att ordningen kunde höjas men oftast hade de bästa modellerna ännu högre ordning.

Vi har därför genomgående förlitat oss på varianserna storlek i förhållande till parametrarna och delvis på residualtestens utseende.

4. Sammanfattning

Det visar sig att en ARMA-modell är lättare att anpassa, men att regulator-dynamiken dock spökar. I pol-nollställe-diagram ser man tydligt när modellen anpassat även denna dynamik. Dämpkvoten är dock i de flesta fall oberoende av om modellen beskriver all dynamik eller "bara" neutronflödesdynamiken.

Dessutom ser man att bodeplotarna är likartade och ordningarna blir ungefär de samma för AR och ARMA. Därför borde AR vara en fullgod modell, men skillnaden ligger i de alltför varierande dämpkvoterna i AR-modellen. Problemen ökar vid besvärligare driftslägen.

Då vi ska bestämma modellordningen online är det stabilare uppförandet i dämpkvot avgörande för val av modell. Detta resultat kan i och för sig bero på att skattningen görs med MK, vilken inte är konstant för färgat brus.

Det har varit intressant att prova identifiering i praktiken och eftersom signalen innehåller dels bruskomponent samt dels annan dynamik från återkopplade system är det ett givande identifieringsproblem.

Referenser

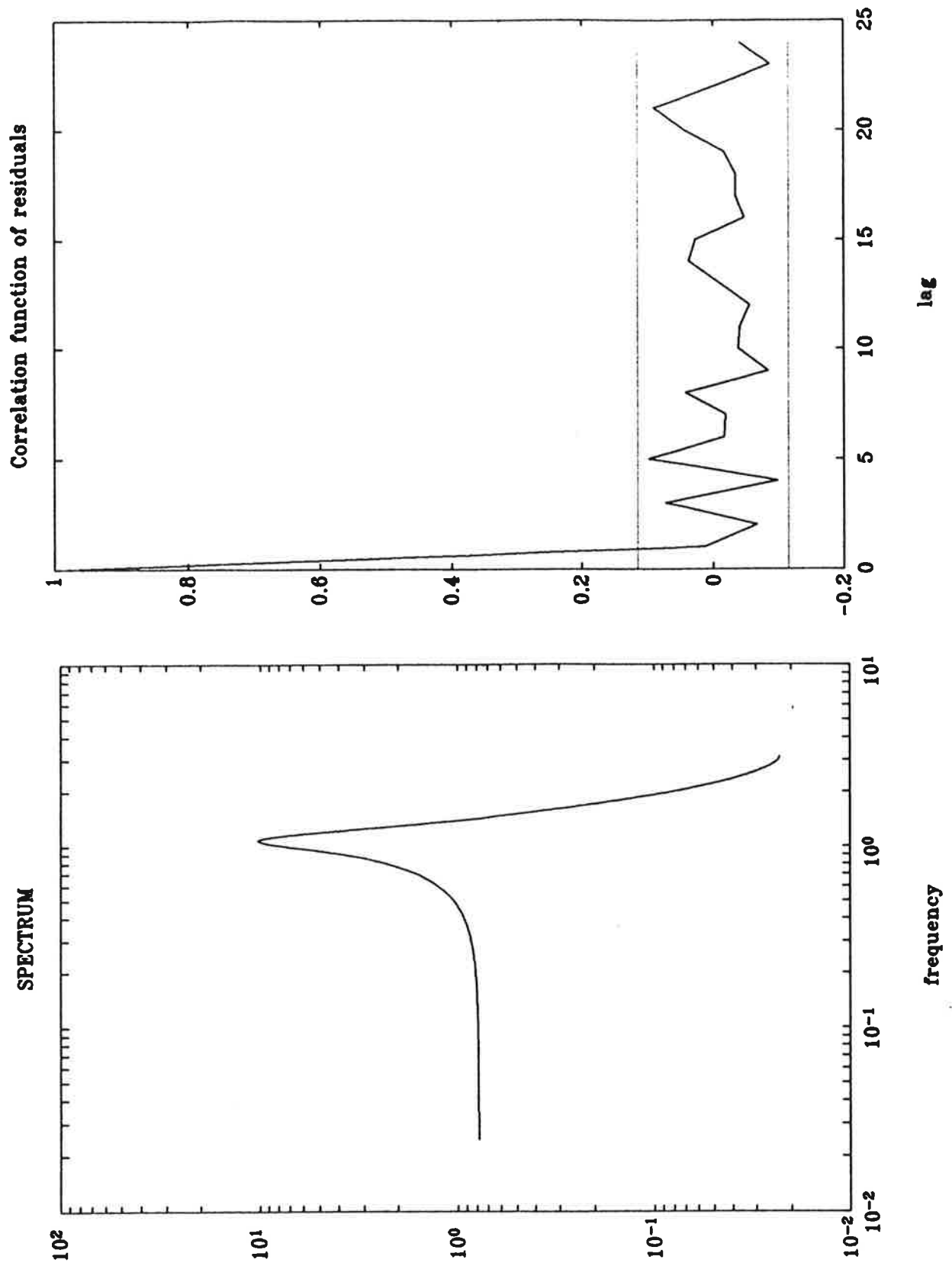
Johansson R. , "*lecture notes*", Lund,1990

Stoica P. och Söderström T. , *System Identification*, Prentice Hall,1989

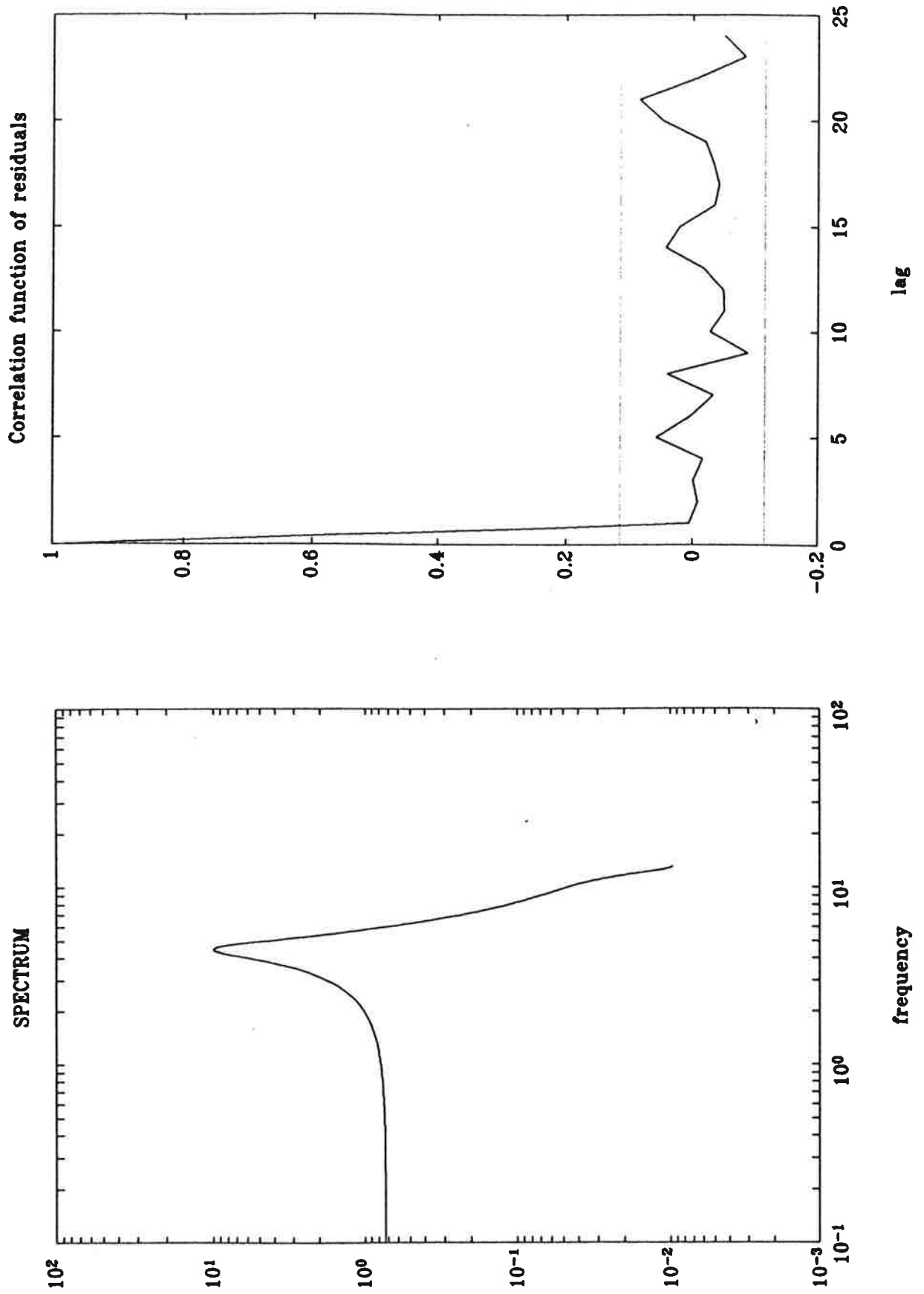
Åström K.J. och Wittenmark B. *Computer Controlled systems*, Prentice Hall, 1990

APPENDIX A

Plottar över residualtest och bodekurvor för en AR och ARMA-modell. Mätdata är de i Kapitel 3 använda. Modellordningen för AR är 3 c ARMA, $n_a=3, n_c=3$.



Figur A.1 Bode-diagram och residualtest för AR-modell av ordning 3



Figur A.2 Bodediagram och residualtest för en ARMA-modell med ordning $n_a=3, n_c=2$

Tabell 1. Akaikes FPE för ARMAX- och Box-Jenkinsmodeller

Ordning	FPE, ARMAX	FPE, Box-Jenkins
2	14.79	13.84
3	4.106	9.4
4	18.91	1.326
5	2.902	0.4074
6	90.27	0.248
7	6.727	0.194
8	0.5064	0.4169
9	0.3747	0.4641

Identifiering av återkopplad saltprocess

Frans Hermodsson
Håkan Svegerud

Institutionen for Reglerteknik
Lunds Tekniska Högskola
November 1990

Innehållsförteckning

1. Inledning	2
2. Problemuppsättning	3
3. Identifieringsförfarande	5
4. Slutsatser	10
A. Mätningar på saltprocessen	11
Referenser	14

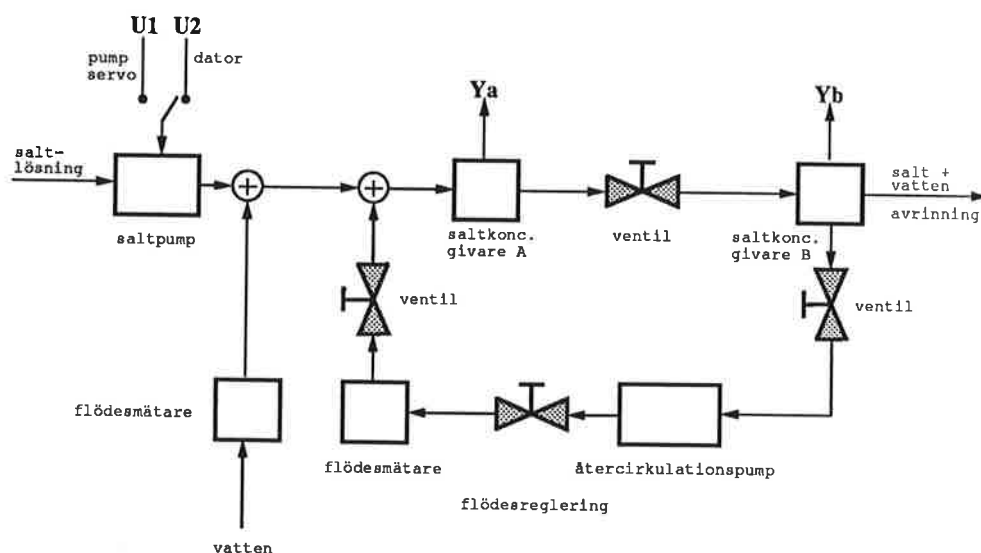
1. Inledning

Detta projekt i processidentifiering går ut på att identifiera ett system som består av en saltprocess som är uppbyggd på institutionen för Reglerteknik. Processen går att modifiera på ett antal olika sätt genom att öppna respektive stänga ett antal ventiler. Den process vi har valt att studera består av ett system som är återkopplat. Systemet är beskrivet av figur 1 där de fysikaliska delarna är identifierbara.

Vatten strömmar in i systemet via en flödesmätare. Till detta inflöde kan man tillsätta vatten med en saltkoncentration med hjälp av en pump som man styr med hjälp av en yttre spänning. Denna spänning valde vi att styra med hjälp av en IBM AT för att på ett smidigt sätt kunna sampla denna signal som en insignal. Som utsignaler från processen använde vi oss av de värdena som saltkoncentrationsgivarna A respektive B gav oss. I återkopplingslänken finns en pump för att hålla igång cirkulationen samt en flödesgivare. Vi ställde in de båda flödesgivarna så att de visade samma flöde. Genom att flödet är lika stort i inflödet till systemet som i återkopplingen så innebär det att flödet ut ur systemet också borde vara lika stort som i inflödet respektive återkopplingen.

Pumpen som tillsätter salt till vatten flödet är av så kallad slangpumpstyp. Det vill säga att saltlösningen pumpas fram genom en mjuk gummislang som kläms ihop av pumpen och då pumpen startar matas den saltbemängda vätskan fram genom ett hjul med tre utskjutande partier tvingar fram vätskan i gummislangen. Detta innebär att saltlösningen kommer att överföras till den övriga vätskan i form av små "saltvattenkvanta" och inte momentant som ett jämnt flöde. Vi ansåg dock att detta borde ha ringa betydelse och eventuellt endast märkas som brus.

I systemet ingår även två andra vätskevägar, dessa finns mellan saltkoncentrationsgivare A och B. Dessa består av dels en tio meter lång slang som fungerar som en stor fördröjning och dels av en vattenbehållare med en omrörare på botten. Vi valde att ej använda dessa två vägar utan att koncentrera oss på de mer direkta vägarna i systemet.



Figur 1.1 Saltprocessens uppbyggnad.

2. Problemuppställning

Vi har valt, att ur saltprocessen identifiera saltpumpen, den korta slangen och återkopplingen mellan saltkoncentrationsgivarna, se figur 2.1 .

Vi lyckades identifiera saltpumpen och den korta slangen med gott resultat, och återkopplingen med något sämre resultat, se figurerna 3.1-3.3 .

Saltpumpen

Hela processen styrs med hjälp av saltpumpen, som får sin styrsignal från antingen, en IBM AT eller ett pumpservo. Saltpumpen doserar saltmängden i processen, genom att pumpa ut en salt från en mättad lösning, bestående av vanligt vatten och koksalt.

Vi ansatte saltpumpen och ca. 10cm slang, som ett första ordningens system med en fördröjning. Insignalen var datorns PRBS-signal, utsignalen fås via saltkoncentrationsgivare A.

$$G_u(s) = \frac{K_u}{1 + sT_u} e^{-sT_{f_u}}$$

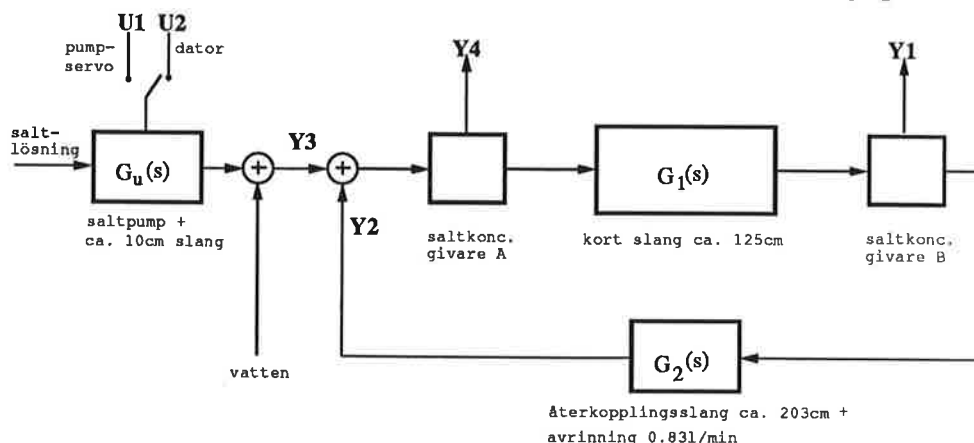
Kort slang

Den korta slangen ansattes också som ett första ordningens system med en fördröjning. Insignalen tas från saltkoncentrationsgivare A och utsignalen tas från saltkoncentrationsgivare B.

$$G_1(s) = \frac{K_1}{1 + sT_1} e^{-sT_{f_1}}$$

Återkoppling

När det gäller återkopplingen måste man göra vissa avväganden. Eftersom givarna sitter där dom sitter, måste avrinningen tas med i $G_2(s)$ för att saltkoncentrationsgivare B skall kunna användas som insignal. Utsignalen Y2 beräknades som Y4-Y3, där Y4 är mätbar och Y3 beräknas med hjälp av den



Figur 2.1 Utvalda delar av saltprocessen för identifiering

identifierade modellen för saltpumpen. Även här ansattes ett första ordningens system.

$$G_2(s) = \frac{K_2}{1 + sT_2} e^{-sT_{f2}}$$

3. Identifieringsförfarande

Allmänt

För att identifiera de tre olika delprocesserna saltpumpen, den korta slangen samt återkopplingen så gjorde vi först en del grundläggande antaganden.

Vi ansåg att då en saltkoncentrationspuls passerar fram genom slangen så kommer det att ske en långsam utslätning av denna pulsen och detta borde man kunna anse som ett system med en tidskonstant. Vidare antog vi att eftersom vätskan färdas med en måttlig hastighet så kommer det att uppstå tidsfördröjningar mellan de olika givarna, det brus som finns i systemet antog skulle kunna anses som normalfördelat eventuellt med någon typ av modulering. Vi bestämde oss för att använda någon typ av interaktiv identifieringsmetod [1] tex ARMAX eller Box Jenkins metod. Experimenten redovisas som försök 1-6 i appendix.

Den korta slangen

Den korta slangen ansåg vi att man kunde betrakta som ett första ordningens system med en liten tidskonstant och med en kort tidsfördröjning. Vid den första identifieringen använde vi oss av data från försök, se appendix. Som insignal använde vi oss av y_4 och som utsignal tog vi y_1 dvs signalerna från de båda saltkoncentrationsgivarna, se figur 2.1 . Genom att plotta signalerna i MATLAB så kunde vi bestämma tidsfördröjningen till ungefär tre sekunder.

Försök 1 misslyckades på grund av för dålig excitation av processen, för kort mätserie och att användandet av ARMAX som innebar att vi inte kunde modulera bruset som $e(t)$.

Genom att ta nya mätvärden, försök 2 och genom att använda Box-Jenkins metod så fick vi betydligt bättre överensstämmelse mellan beräknad och uppmätt utsignal från systemet. Efter en del experimenterande med inparametrarna till Box Jenkins funktionen så kom vi till resultatet som visas i figur 3.1 . Som synes så är överensstämmelsen mellan beräknad och uppmätt signal god.

Det system vi har fått fram ser på diskret form ut på detta vis

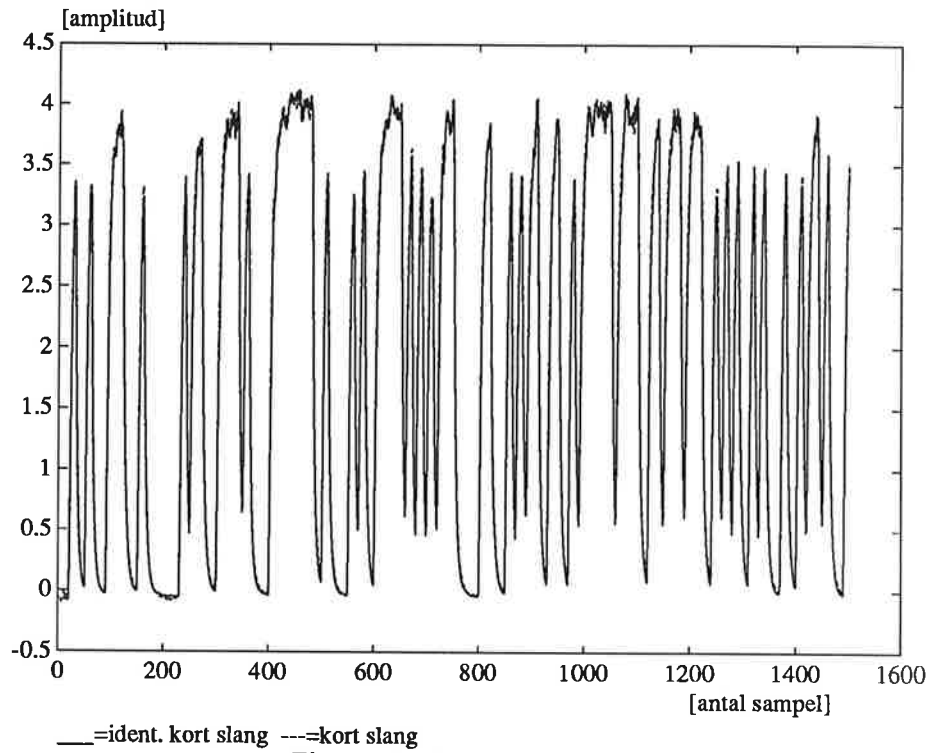
$$G_1(q) = \frac{b_1}{1 + f_1 q^{-1}} q^{-k_1} = \frac{0.3080}{1 - 0.6966 q^{-1}} q^{-12}$$

Detta system är alltså identifierat med samplingsfrekvensen 4 Hz.

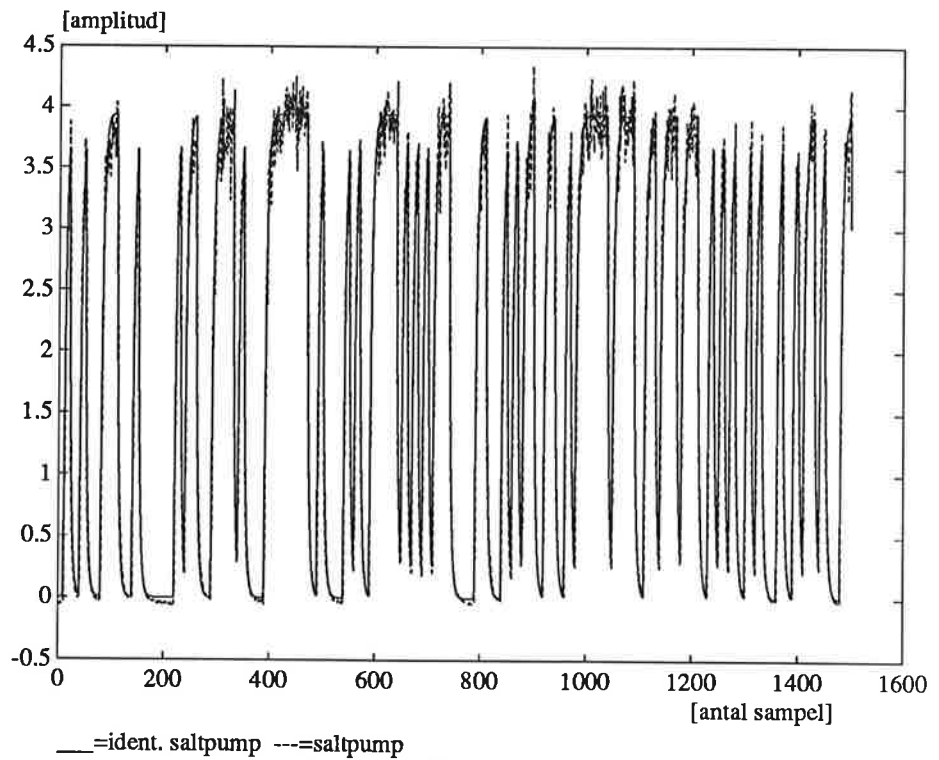
Saltpumpen

För saltpumpen gäller liknande förutsättningar som för den korta slangen men tidsfördröjningen uppskattade vi endast till ca 0.75 sekunder. Som mätdata använde vi försök 2 och som insignal använde vi spänningen som driver pumpen och som utsignal tog vi y_4 dvs signalen från givare a. Efter endast ett fåtal försök fick vi de resultat som finns att beskåda i figur 3.2.

Det system vi fått fram här är alltså identifierat med samplingsfrekvensen 4 Hz. Det visar sig senare att vi även behöver det för en samplingsfrekvens



Figur 3.1 Simulering av G_1



Figur 3.2 Simulering av G_u

som är 2 Hz. Istället för att göra nya mätningar så gjorde vi en omräkning till en samplingsfrekvens av 2 Hz. Systemet för 4 Hz ser ut på följande vis

$$G_u(q) = \frac{b_1}{1 + f_1 q^{-1}} q^{-k_1} = \frac{0.111}{1 - 0.7396 q^{-1}} q^{-2}$$

och för 2 Hz på följande vis

$$G_u(q) = \frac{b_2}{1 + f_2 q^{-1}} q^{-k_2} = \frac{0.1972}{1 - 0.6033 q^{-1}} q^{-1}$$

Beräkningarna från 4 till 2 Hz sampelfrekvens gjordes genom att först räkna om överföringsfunktionen till kontinuerlig form, därefter bytte vi sampelfrekvens och räknade om till diskret form igen.

Återkopplingen

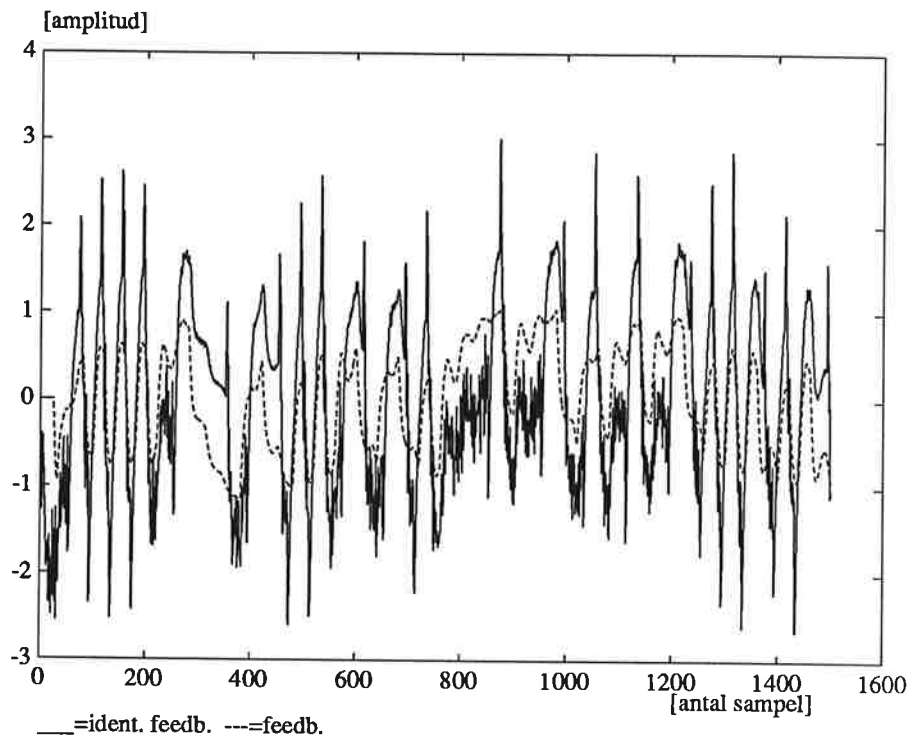
Överföringsfunktionen för återkopplingen ansåg vi även den vara ett system av första ordningen dock med betydligt större tidskonstant och tidsfördröjning än i de tidigare två fallen.

Då vi skulle identifiera överföringsfunktionen i återkopplingen stötte vi på problem eftersom saltkoncentrationsgivarna inte sitter så att man direkt kan använda den ena som insignal och den andra som utsignal. Vi beslöt oss därför att som det är beskrivet i kapitel 2 att inkludera utloppet från systemet i överföringsfunktionen. Detta gjorde vi för att kunna använda signalen från saltkoncentrationsgivare B som insignal till systemet dvs signal Y_a i figur 2.1. För att hitta en lämplig utsignal fick däremot anstränga oss lite mer. Vi ville ha signalen som i figur 2.1 kallas y_2 , men denna kunde vi inte mäta direkt. Vi kunde däremot mäta signalen y_4 med hjälp av saltkoncentrationsgivare A. Signalen y_3 kunde vi beräkna genom att vi tidigare identifierat överföringsfunktionen för saltpumpen dvs G_u och genom att vi kunde mäta insignalen till pumpen. Då vi hade både y_4 och y_3 så kunde vi alltså beräkna den önskade y_2 genom att $y_2 = y_4 - y_3$, se figur 2.1 .

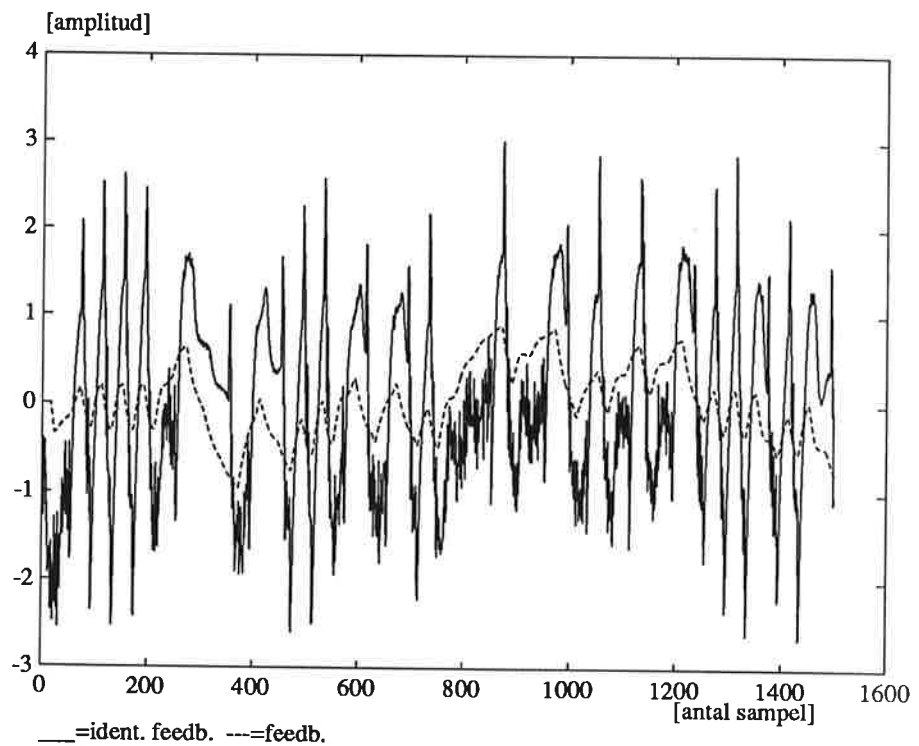
Genom att först ta upp en mätning där vi manuellt skickade en puls genom systemet kunde vi genom att plotta denna med Matlab uppskatta tidsfördröjningen till ca 13 sekunder, detta försök kallade vi 3 . Vid första identifieringen använde vi oss av värdena i försök 4 . Detta gav dock ett dåligt resultat och vi insåg att eftersom detta var ett så pass långsamt system så borde vi sänka samplingsfrekvensen. Vi gjorde ett nytt försök med halva samplingsfrekvensen, dvs 2 Hz detta blev försök 5 . Detta innebar att om vi skulle räkna ut signalen y_3 enligt beskrivningen ovan så behövde vi saltpumpens överföringsfunktion för den nya samplingsfrekvensen. För att slippa att göra nya försök och nya identifieringar med saltpumpen så räknade [2] vi ut den nya överföringsfunktionen för saltpumpen. Med värdena i försök 5 fick vi ett något bättre resultat, som dock ej speciellt bra. Vi gjorde ännu ett försök, nämligen 6 . Denna gången sänkte vi frekvensen i den PRBS signal som vi exiterade systemet med. Detta gav ännu något bättre resultat vi fick dock fortfarande dålig överensstämmelse i de höga frekvenserna. Vi provade till sist att filtrera in- och utsignalerna till systemet, innan vi utförde själva identifieringen, med ett Butterworthfilter. Genom att experimentera, fann vi att med ett 5:e ordningens Butterworthfilter av bandpassmodell fick vi en något bättre överensstämmelse i de höga frekvenserna. Bäst resultat fick vi med gränshfrekvenserna 0.3 Hz och 2.4 Hz.

Den identifieringsmetod vi använde oss av var även i detta fallet Box-Jenkins. Det system som vi tyckte var det som bäst överensstämde med verkligheten hade följande utseende

$$G_2(q) = \frac{b_1}{1 + f_1 q^{-1}} q^{-k_1} = \frac{0.0944}{1 - 0.8345 q^{-1}} q^{-24}$$

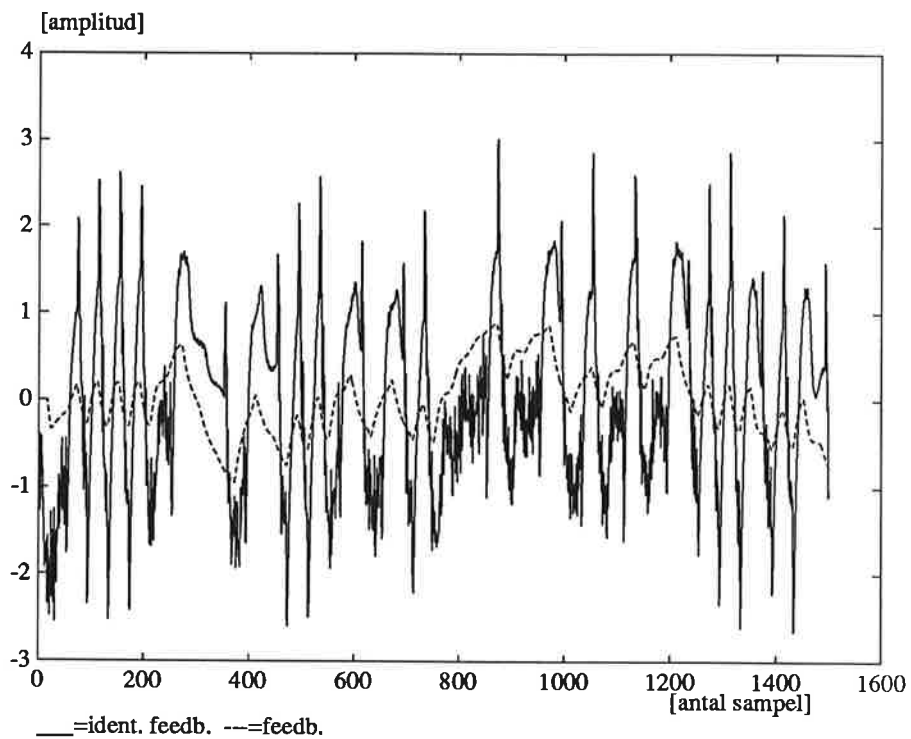


Figur 3.3 Simulering av G_2 med rätt tidsfördröjning dvs 12 sekunder



Figur 3.4 Simulering av G_2 med för liten tidsfördröjning, 8 sekunder

I figurerna 3.3-3.5 redovisar vi dels simuleringar genomförda med ovanstående modell och dels två simuleringar genomförda med för stor respektive för liten tidsfördröjning.



Figur 3.5 Simulering av G_2 med för stor tidsfördröjning, 16 sekunder

4. Slutsatser

Saltpumpen

Vår ansats av $G_u(s)$ visade sig vara riktig. Kurvorna (fig 3.1) överlappar varann perfekt. Systemet är okomplicerat och frånvaron av någon form av kraftigt mätbrus, gör identifieringen enkel.

Kort slang

Ansatzerna för den korta slangen stämde också bra. Identifieringen utfördes under närmast ideala förhållanden med ett enkelt system och låga brusförhållanden, vilket gjorde att vi fick näst intill identiska kurvor, se figur 3.1 .

Återkoppling

När $G_u(s)$ och $G_1(s)$ blev så pass lyckade som de blev, borde $G_2(s)$ med stöd av ovan nämnda överföringsfunktioner bli enkel att identifiera. Så blev nu inte fallet, tidskonstanten lyckades vi få fram, men amplituden stämmer ej.

En av anledningarna kan vara att vi förlorade högfrekvent information när vi beräknade $G_2(s)$:s utsignal. En annan anledning kan ha varit att det fanns luft i systemet under mätningarna. Dessutom var inte ventilen för inkommande vatten stabil. Den kunde ibland variera med ca. +/- 5 %, eftersom ventilen kärvade var det svårt att hålla en konstant flödesnivå.

A. Mätningar på saltprocessen

För mätningar på och insamling av mätdata från saltprocessen har en IBM AT med ett AD/DA-kort använts. Vi använde logger, ett program för insamling av mätdata och generering av PRBS-signalen

Insignalerna till datorn kommer från saltkoncentrationsgivarna. Utsignalen från datorn som styr saltpumpen är en PRBS-signal med valbara parametrar.

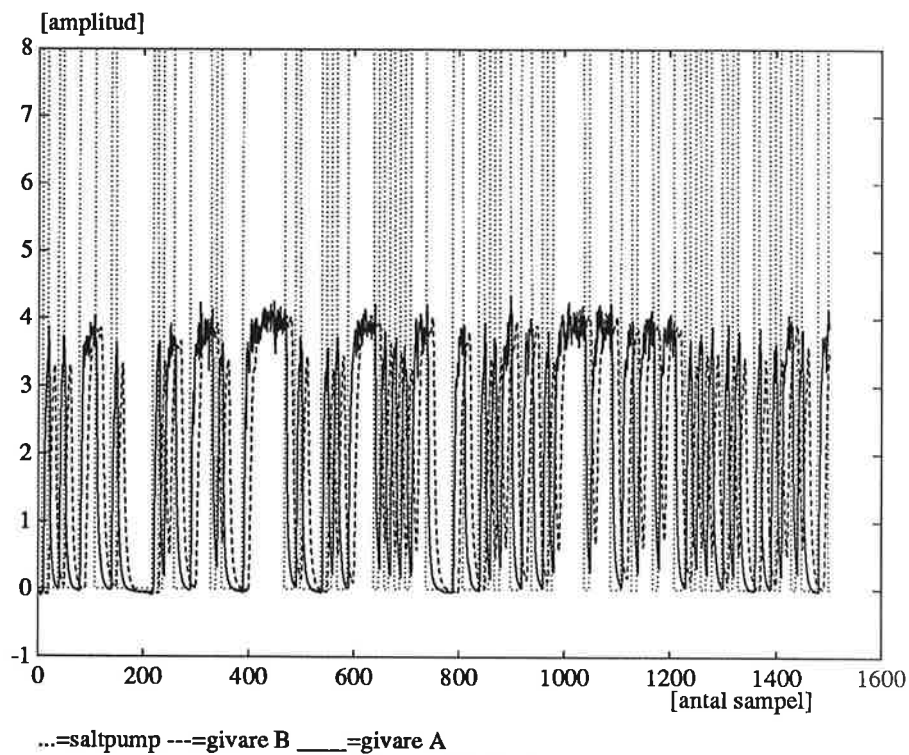
I försök 3 användes pumpservot som insignal, vilken kan generera en fyrkantspuls med valbar periodtid. I alla andra försök användes datorns PRBS-signal som processens insignal.

Försök 1 Mätdata finns i filen ex11.mat

- tsamp = 0.25s
- PRBS period = 20 sampel
- PRBS amplitud = 4.0V
- PRBS mean = 4.0V
- number of datapoints = 500
- vattenflöde in = 0.83l/min
- ingen återkoppling

Försök 2 Mätdata finns i filen ex12.mat

- tsamp = 0.25s
- PRBS period = 10 sampel
- PRBS amplitud = 4.0V
- PRBS mean = 4.0V
- number of datapoints = 1500
- vattenflöde in = 0.83l/min
- ingen återkoppling



Figur A.1 Mätförsök 2

Försök 3 Mätdata finns i filen ex21.mat

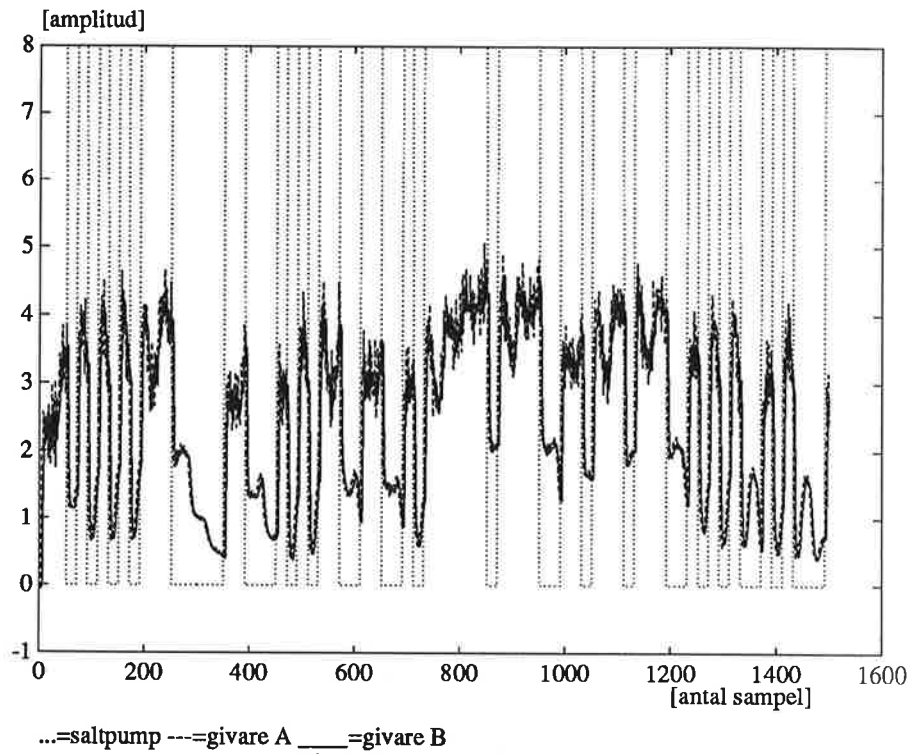
- tsamp = 0.25s
- en puls, manuell trigg
- pulslängd = 10s
- number of datapoints = 400
- vattenflöde in = 0.83l/min
- återkoppling

Försök 4 Mätdata finns i filen ex22.mat

- tsamp = 0.25s
- PRBS period = 10 sampel
- PRBS amplitude = 4.0V
- PRBS mean = 4.0V
- number of datapoints = 1500
- vattenflöde in = 0.83l/min
- återkoppling

Försök 5 Mätdata finns i filen ex23.mat

- tsamp = 0.5s
- PRBS period = 10 sampel
- PRBS amplitude = 4.0V
- PRBS mean = 4.0V
- number of datapoints = 1500
- vattenflöde in = 0.83l/min



Figur A.2 Mät försök 6

- återkoppling

Försök 6 Mätdata finns i filen ex24.mat

- tsamp = 0.5s
- PRBS period = 20 sampel
- PRBS amplitude = 4.0V
- PRBS mean = 4.0V
- number of datapoints = 1500
- vattenflöde in = 0.83l/min
- återkoppling

I samtliga mätningar har signalernas offset åtgärdats.

Referenser

- 1 R Johansson: Modeling and Identification, Department of Automatic Control Lund Institute of Technology Oct 1990
- 2 Karl J. Åström Björn Wittenmark: Computer Controlled System Theory and Design, 1990

PARAMETRISK ROBOTIDENTIFIERING
Projekt i Processidentifiering

Mårten Larsson
Katarina Fredriksson

Institutionen for Reglerteknik
Lunds Tekniska Högskola
november 1990

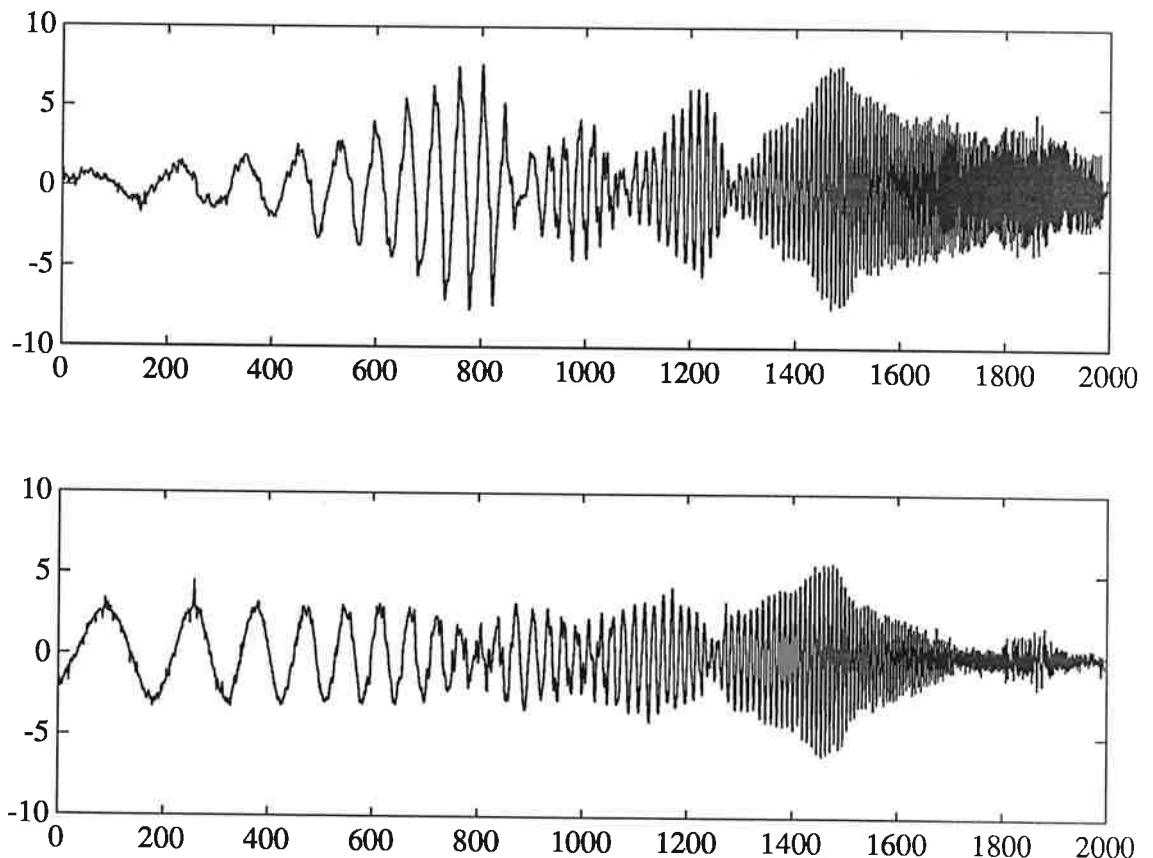
1. Inledning

Denna rapport beskriver ett projekt i Processidentifiering vid Institutionen för Reglerteknik hösten 1990. Arbetet gick ut på att ta fram en dynamisk modell för en robotarms rörelse med ett par olika identifieringsmetoder. Vi utgick från en inspelad dataserie från en robot hos ABB, IRB-3000. Vi har dessutom utfört egna experiment på institutionens robot.

2. Experimentvillkor

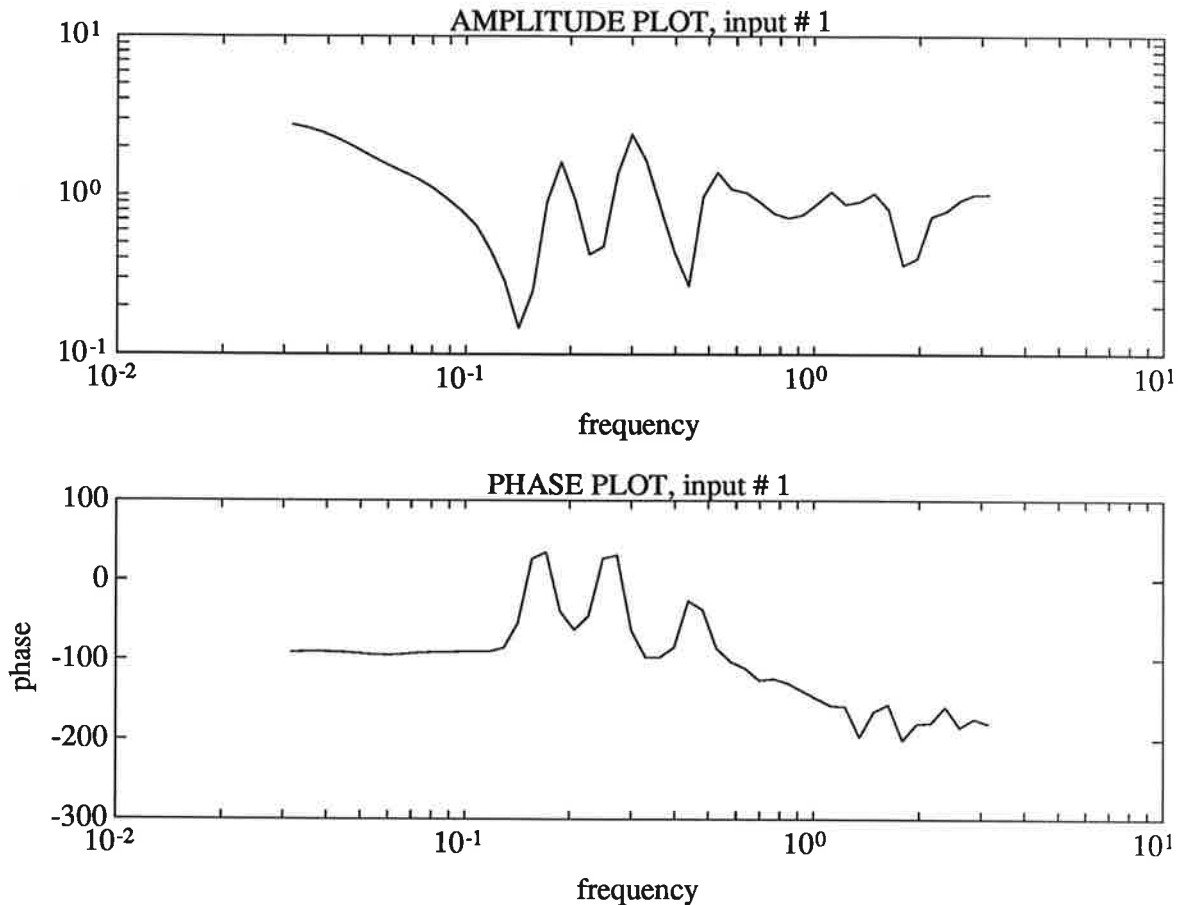
Insignal

Insignal till roboten var ström till motorn och utsignal var vinkelhastighet hos armen. Samplingsfrekvensen var 496 Hz. Experimentet var utfört på axel 1. I Fig.1. återfinns sambandet mellan insignal $u(t)$ och utsignal $y(t)$. Som insignal har valts en frekvenssvept sinussignal. Vid identifieringsförsök bör man dela sin mätserie i två delar och använda den ena delen vid identifiering och den andra till simulering. Valet av insignal i det här fallet omöjliggör detta pga att de båda mätserierna sträcker sig över olika frekvensområden.



Figur 1. Insignal (överst) och utsignal hos robotarmen vid experimentet.

Ett periodogram för dessa mätdata återfinns i Fig.2. Man kan urskilja tre resonanstoppar och ett första ordningens system. Härav kan man dra



Figur 2. Periodogram för den ofiltrerade mätdataserien.

slutsatsen att en sjunde ordningens modell borde vara tillräcklig för att beskriva systemet. Det intressanta frekvensområdet sträcker sig upp till ungefär 1 rad/sek i normerad frekvens.

Koherensfunktionen

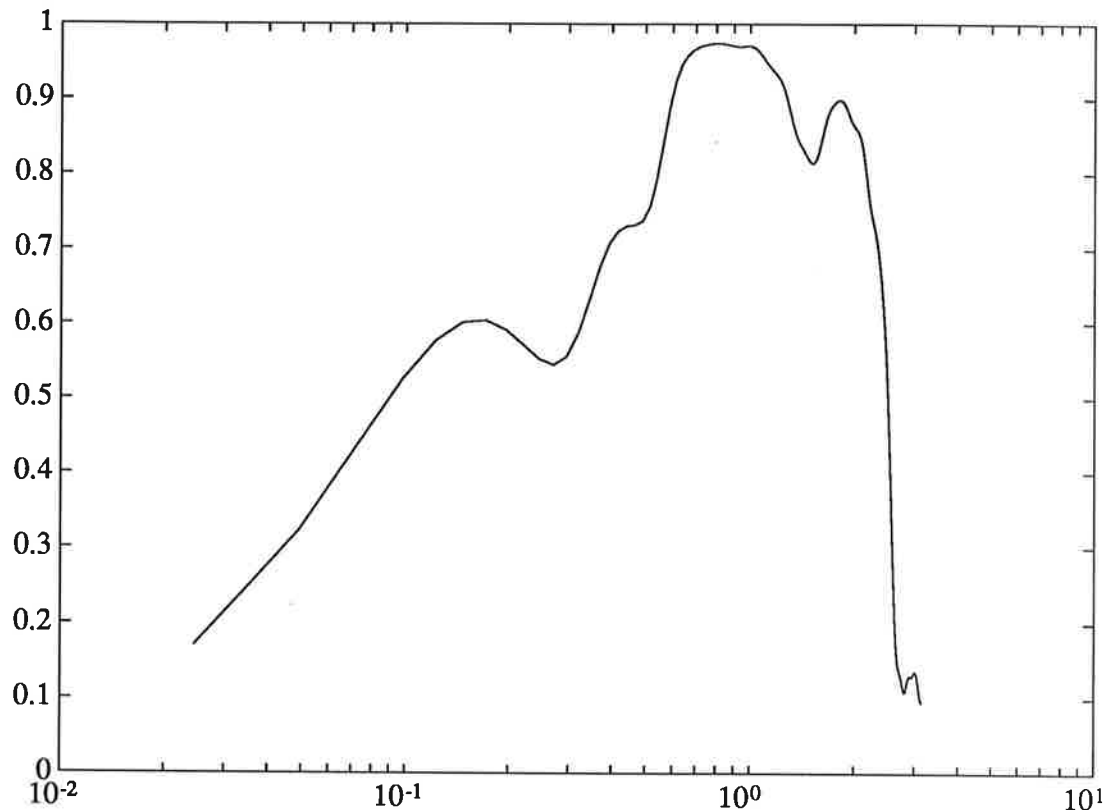
För att få reda på vilka frekvenser som är intressanta ur identifieringssynpunkt är det lämpligt att bilda koherensfunktionen ψ_{yu} mellan u och y . Denna beskrivs av sambandet

$$\psi_{yu}(w) = \frac{|\phi_{yu}(w)|}{\sqrt{\phi_u(w)\phi_y(w)}}$$

Koherensfunktionen återfinns i Fig.3. Det område där man kan förvänta sig att få god identifiering ligger mellan 0.09 och 2.0 rad/sek.

Filtrering

Vi konstaterade att mätsignalerna varken innehöll något medelvärde eller någon trend, vilka skulle ha givit ett felaktigt identifieringsresultat om man hade försummat att kompensera för dessa. Vi tog bort de allra mest högfrekventa signalerna genom att kapa mätserierna och lade sedan på ett fjärde ordningens Butterworth-bandpassfilter med brytfrekvenserna 0.03 resp. 2.4 rad/sek i enlighet med koherensfunktionen. Periodogrammet som erhålles efter filtreringen ges i Fig.4.



Figur 3. Koherensfunktionen mellan insignal och utsignal.

3. Modellstruktur

Tre olika modellstrukturer var aktuella i projektet:

- ARX
- ARMAX
- Box-Jenkins

ARX-modellen beskrivs av

$$A^*(q^{-1})y(t) = B^*(q^{-1})u(t - k) + e(t)$$

Den används när bruset kan approximeras som vitt brus. Den är lätt att räkna med och man kan snabbt skaffa sig en uppfattning om rätt modellordning. För att räkna fram parametrarna kan man använda sig av minsta kvadratmetoden.

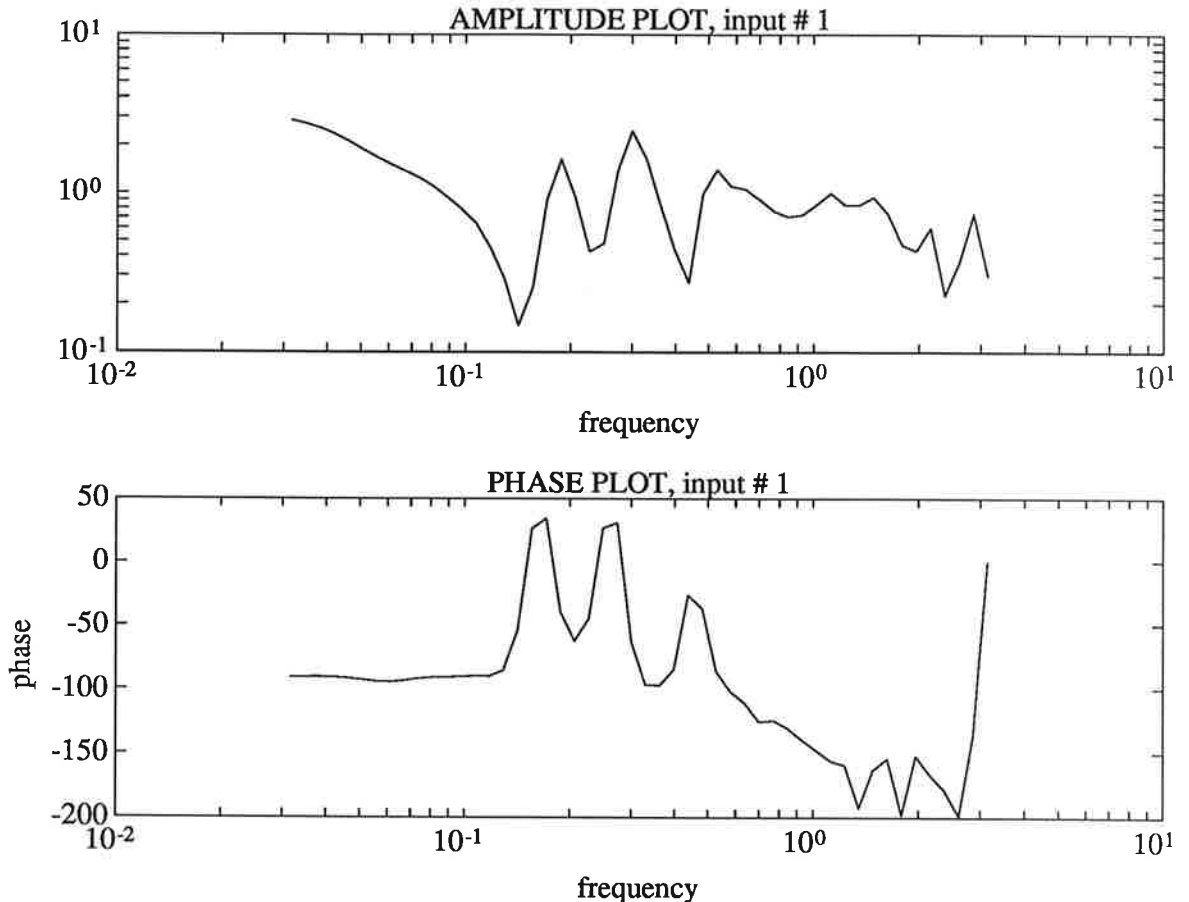
ARMAX-modellen beskrivs av

$$A^*(q^{-1})y(t) = B^*(q^{-1})u(t - k) + C^*(q^{-1})e(t)$$

Om man har färgat brus använder man denna modell. Vid beräkning av parametrarna använder man prediktionsfelsmetoden. Den kräver betydligt mer räknearbete än minsta kvadratmetoden.

Box-Jenkinsmodellen beskrivs av

$$y(t) = \frac{B^*(q^{-1})}{F^*(q^{-1})}u(t - k) + \frac{C^*(q^{-1})}{D^*(q^{-1})}e(t)$$



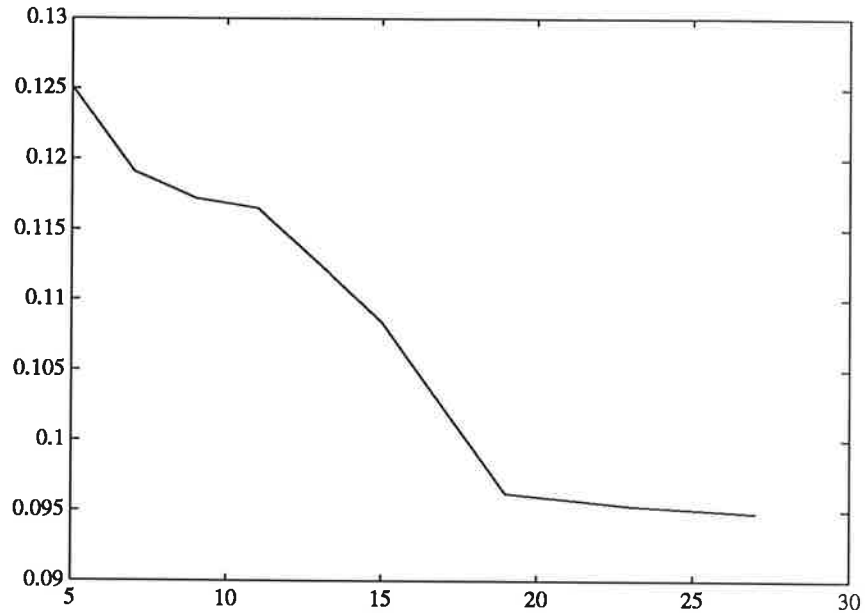
Figur 4. Periodogram över den förfilterade mätdataserien.

Även denna modell använder sig av prediktionsfelsmetoden vid beräkning av parametrarna. En fördel gentemot ARMAX-modellen är att bruset modelleras bättre.

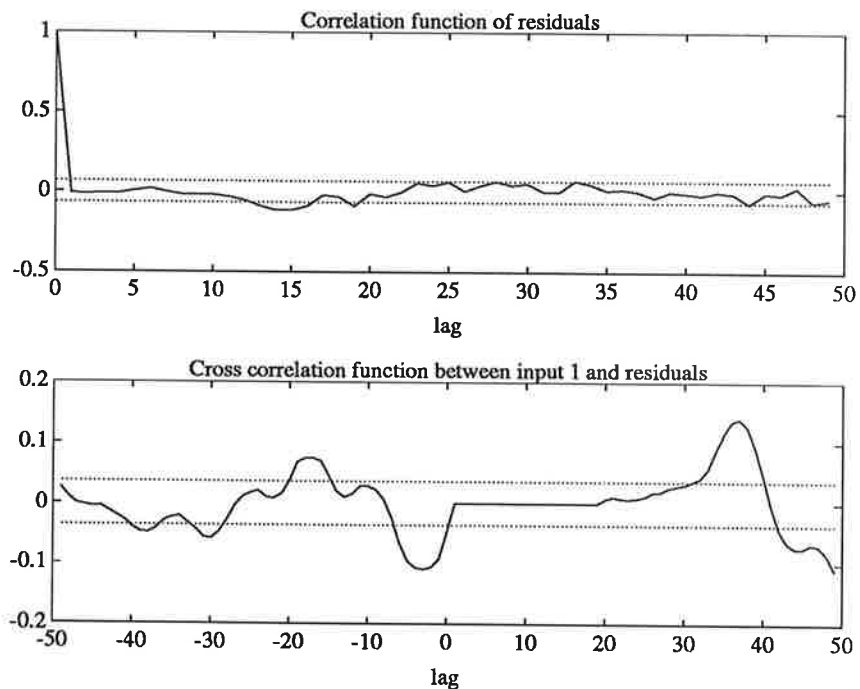
Vi började med att ansätta en ARX-modell och fick på så sätt en uppfattning om modellens ordning. Genom residualtester och studium av brusets spektralfördelning märkte vi också att bruset var färgat, varför vi ansåg oss behöva en modell som modellerar bruset bättre.

4. Identifiering med hjälp av en ARX-modell

Vi ansatte en ARX-modell där parametrarna uppskattades med minsta kvadrat-metoden. För att kunna avgöra systemets ordning använde vi matlabfunktionen `arxstruc`. Det visade sig dock att funktionen `selstruc` alltid valde den modell som hade den högsta ordningen. För att hitta den bästa modellordningen kan man i stället titta på Akaikes FPE som funktion av ordningstalet, eftersom FPE straffar på modellordningen. Detta åskådliggörs i Fig.5. Man ser i figuren att kurvan aldrig når något minimum för något rimligt ordningstal. Kurvan har dock ett tydligt knä vid 19:e ordningen. Residualernas autokorrelation och korskorrelationen mellan insignalen och residualerna återfinns i Fig.6. Man ser att experimentet har utförts i slutet loop på utbuktningen vid negativa tidsskillnader τ i korskorrelationskurvan. I samma kurva kan man observera utbuktningen vid $\tau = 30$. Den kan bero antingen på en för låg

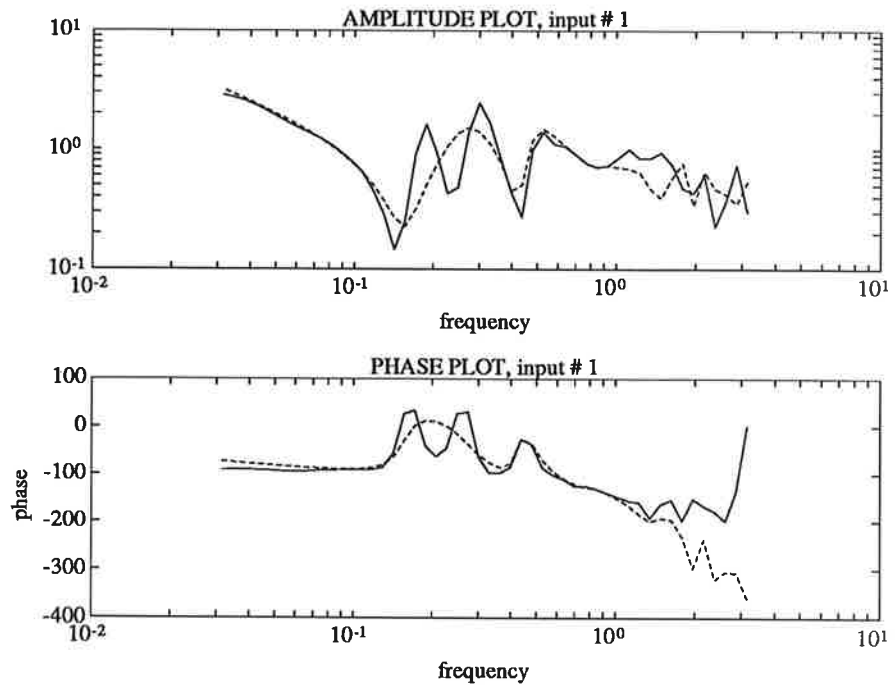


Figur 5. FPE som funktion av modellordning för ARX

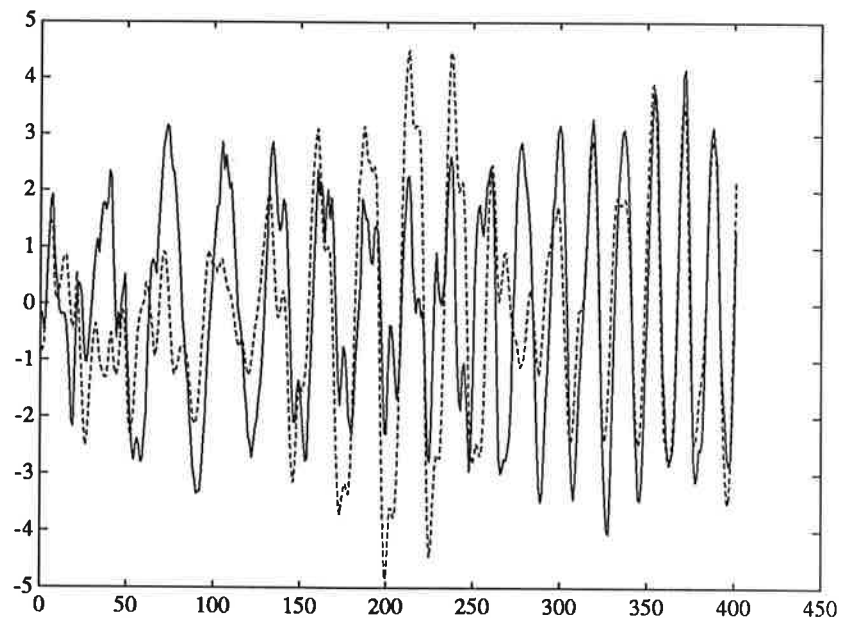


Figur 6. Autokorrelation och korskorrelation för ARX-modell av ordning 19

modellordning eller på att antagandet om att brusets vitt är felaktigt. Bodediagrammet i Fig.7. visar att modellen inte lyckades lösa upp de tre stora resonanstopparna. Simulering gav ganska bra resultat vid lägre frekvenser men betydligt sämre vid högre frekvenser. Fig.8. visar resultatet av en simulering. Detta stämmer med iakttagelserna i Bodediagrammet. Med modellordning 30 märktes ingen påtaglig förbättring. Vi konstaterade att vi behövde en modell som kunde modellera brusets bättre.



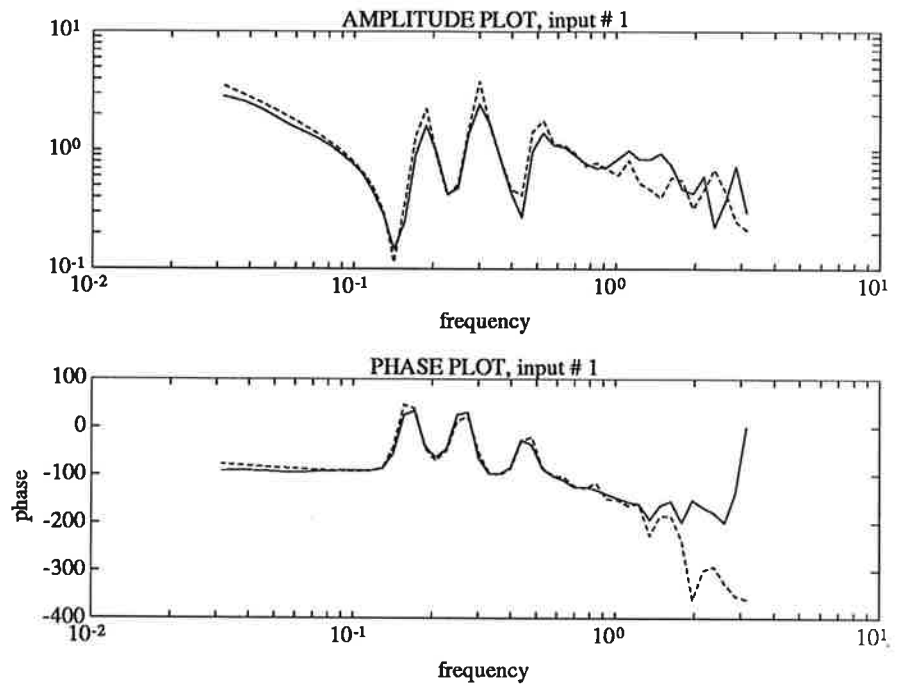
Figur 7. Bodediagram för systemet (heldragen) och ARX-modell av ordning 19 (streckad)



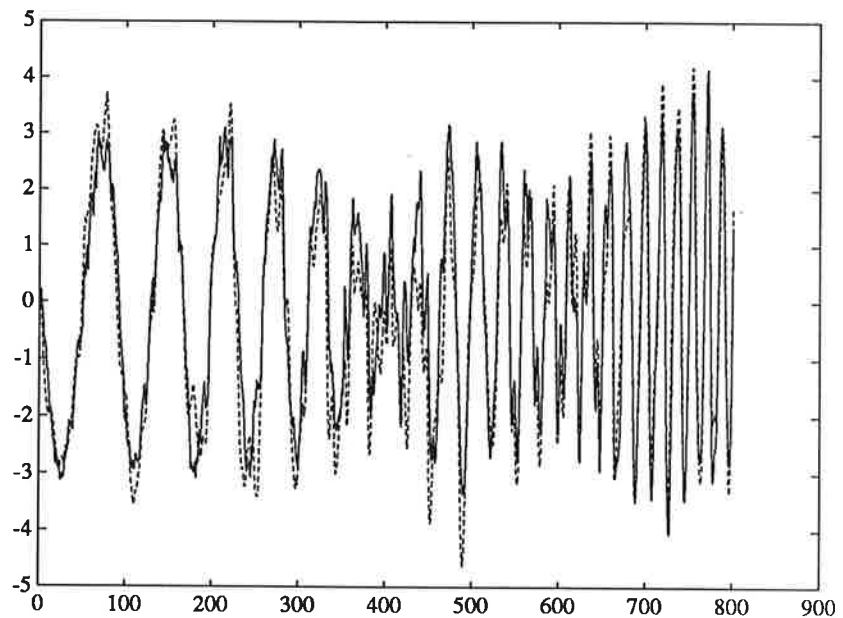
Figur 8. Simulering av modellen av ordning 19 (streckad)

5. Identifiering med hjälp av en Box-Jenkinsmodell

Med tanke på brusets spektralfördelning ansåg vi att en Box-Jenkinsmodell var lämplig. Vid försök att ta fram en sådan av sjunde ordningen upptäckte vi vid simulering att modellens utsignal anpassade sig efter systemets insignal i stället för dess utsignal. Detta berodde på numeriska problem för just denna ordning. Högre ordningar gav bra resultat. För att få bra anpassning i

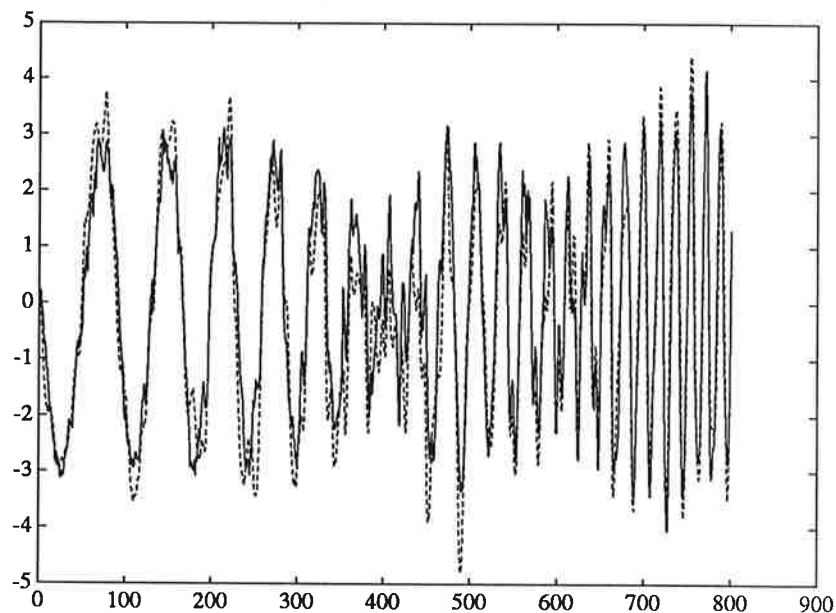


Figur 9. Bodediagram för systemet (heldragen) och ARMAX-modell av ordning 30 (streckad)



Figur 10. Simulering av modellen av ordning 30 (streckad)

frekvensplanet behövde vi gå högt upp i ordning för att sedan reducera gradtalet. Man vinner då mindre på att använda Box-Jenkins. Vi beslöt oss därför att gå över till identifiering med hjälp av ARMAX.



Figur 11. Simulering av den reducerade modellen av ordning 11, balanserad realisering (streckad)

6. Identifiering med hjälp av en ARMAX-modell

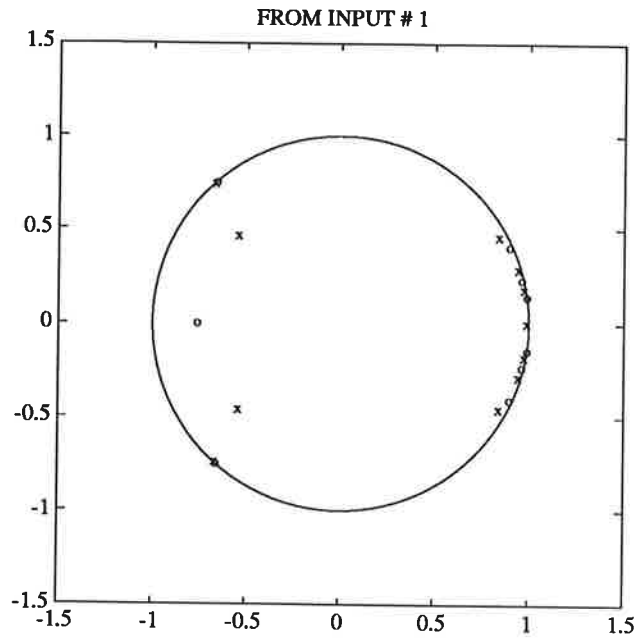
Vi började med att ansätta en sjunde ordningens ARMAX-modell. Detta var ej tillräckligt. För att kunna lösa upp de tre stora resonanstopparna i periodogrammet, se Fig.9, var vi tvungna att öka modellordningen till 30. Residualernas autokorrelation och korskorrelation med signalen höll sig inom felgränserna $\pm 3\sigma$. Simulering gav ett gott resultat. Man ser i Fig.10. att denna modell följde utsignalen betydligt bättre än motsvarande simulering av ARX-modellerna. Den framtagna modellen har en onödigt hög ordning. För att underlätta hantering av modellen är det lämpligt att göra modellreduktion. Vi utförde reduktion med hjälp av två olika metoder, dels via balanserad realisering dels genom strykning av vissa poler och nollställen.

Modellreduktion med hjälp av balanserad realisering

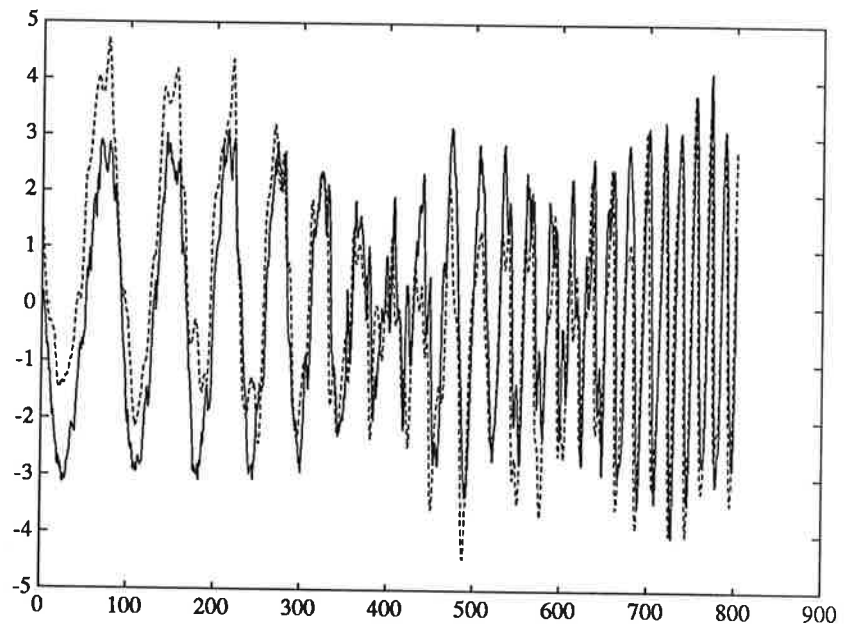
Vi utförde balanserad realisering av modellen av ordning 30 med hjälp av `dbalreal` i MATLAB. Gramianens egenvärden kunde inte direkt grupperas i två grupper med tydlig skillnad i storlek mellan det minsta egenvärdet i gruppen med de största egenvärdena och det största egenvärdet i gruppen med de minsta egenvärdena. Vi kunde inte direkt avgöra till vilken ordning modellen skulle reduceras för att erhålla en bra reducerad modell. Bra resultat erhöles vid val av ordning 11. I Fig.11 återfinns en simulering av en reducerad modell av ordning 11. Dess pol- och nollställediagram visas i Fig.12.

Modellreduktion genom strykning av poler och nollställen

I bodediagrammet valdes den frekvens upp till vilken vi ville ha god överensstämmelse med systemet, i vårt fall ungefär 1 rad/sek. Denna frekvens motsvarar cirka 60 grader i enhetscirkeln i z-planet. Alla poler och nollställen som låg utanför ± 60 grader ströks. Efter reduceringen återstod en elfte

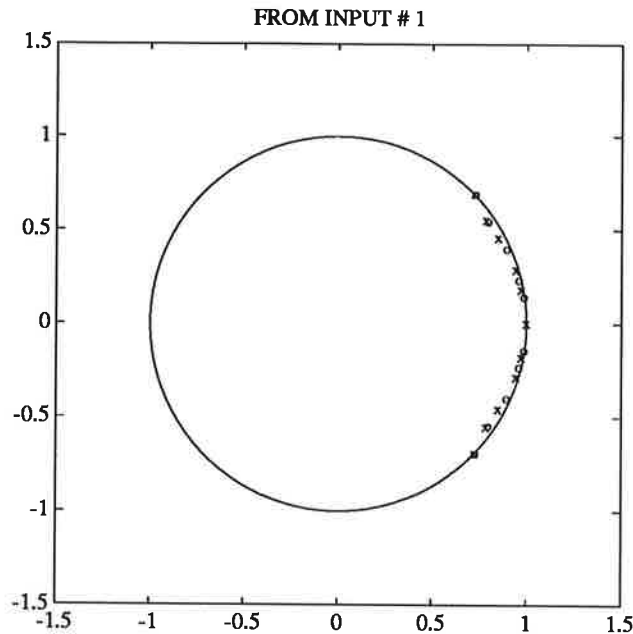


Figur 12. Pol- och nollställediagram för den reducerade modellen av ordning 11, balanserad realisering. Två nollställen ligger utanför bilden.



Figur 13. Simulering av den reducerade modellen av ordning 11, strykning av poler och nollställen (streckad)

ordningens modell. Denna korrigerades så att rätt statistisk förstärkning erhöles. Resultatet av simulering av denna modell återfinns i Fig.13 och pol- och nollställediagrammet i Fig.14.



Figur 14. Pol- och nollställediagram för den reducerade modellen av ordning 11, strykning av poler och nollställen

Som en jämförelse mellan de båda modellreduktionsmetoderna beräknades medelkvadratfelet:

$$MSE = \frac{1}{N} \sum_{n=1}^N (\hat{y}_i - y_i)^2$$

där \hat{y}_i är den reducerade modellens utsignal i punkt i och y_i är systemets utsignal i punkt i . Vi fick följande resultat:

$$\begin{aligned} MSE \text{ (balanserad realisering)} &= 0.5317 \\ MSE \text{ (strykning)} &= 3.0607 \end{aligned}$$

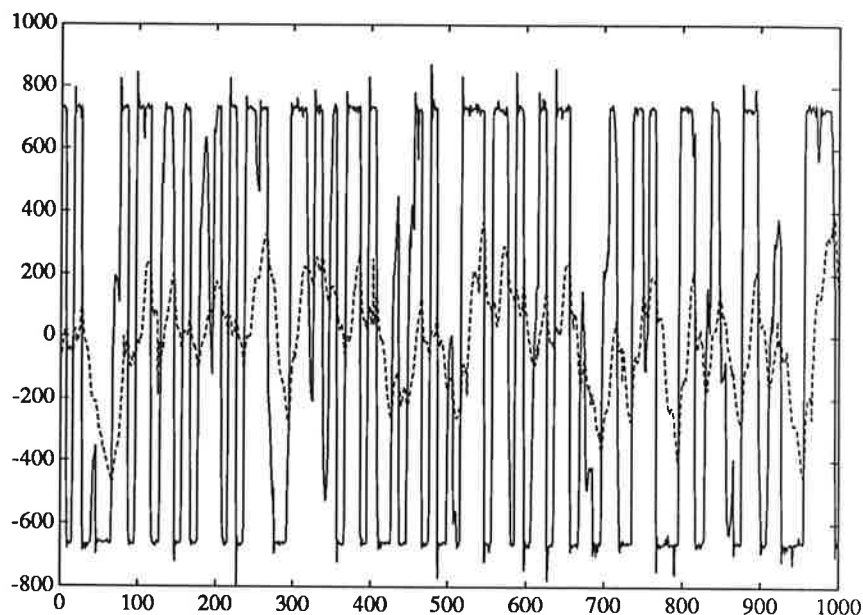
Balanserad realisering är alltså en noggrannare metod för förkortning av poler och nollställen. Enligt tidigare antagande om ett sjunde ordningens system reducerade vi modellen ytterligare med hjälp av balanserad realisering. En sjunde ordningens modell gav gott resultat upp till den normerade frekvensen 0.35 rad/sek. Högre frekvenser gav fel förstärkning. Modellen kunde inte anses vara acceptabel.

7. Egna experiment

Som avslutning gjordes identifiering på institutionens robot, en ABB IRB-6. Mätningarna gjordes på axel 1. Även i detta experiment var insignalen spänning till robotarmens motor och utsignalen var armens vinkelhastighet.

Experimentvillkor

Insignal vid experimentet var en PRBS. Samplingsintervallet var 4 ms. Experimentet utfördes i sluten loop med en PI-regulator. Mätserien återfinns i Fig.15. Om man studerar systemets periodogram finner man att en tredje ordningens modell borde kunna räkna för att beskriva systemet upp till 1 rad/sek.



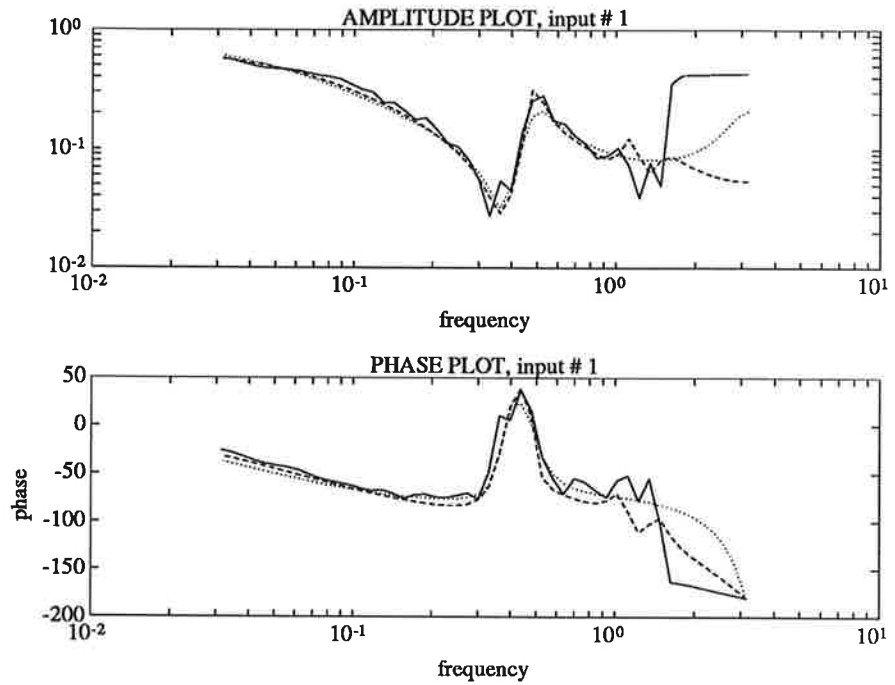
Figur 15. Insignal (heldragen) och utsignal (streckad) till systemet

Koherensfunktionen var stor upp till cirka 1 rad/sek, vilket var det intressanta området. Förfiltrering gjordes med ett åttonde ordningens Butterworth-lågpassfilter med brytfrekvensen 1.2 rad/sek. Medelvärden och trend togs bort. Mätserien delades sedan i två delar, den ena för identifiering och den andra för simulering.

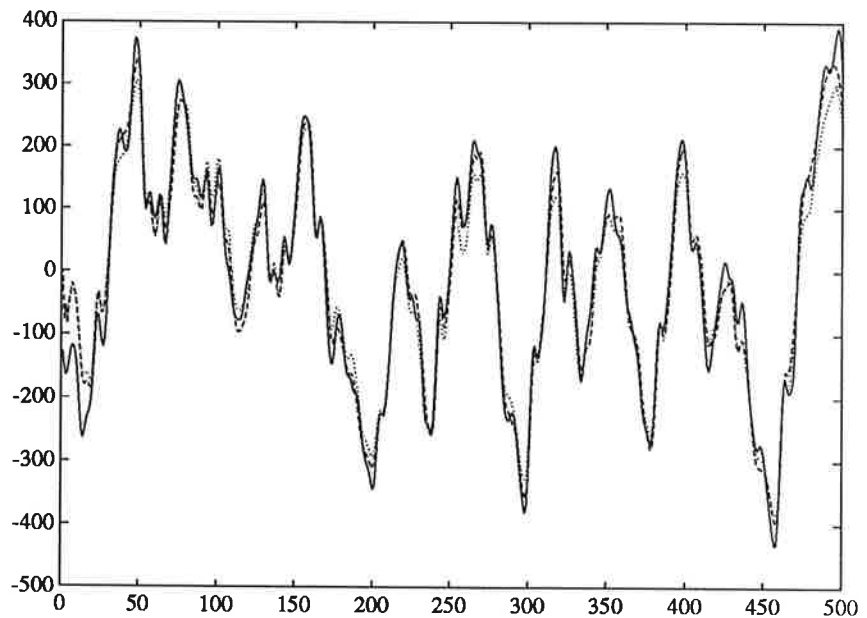
Identifiering

Vi började med att försöka hitta en ARMAX-modell som kunde beskriva systemet. Ett antal modeller av olika ordningar togs fram. Akaikes FPE studerades, men gav egendomliga värden, se tabell nedan. Ordning 4 gav en modell som var bra vid simulering, men inte lyckades lösa upp resonanstoppen i Bode diagrammet. De följande modellordningarna visade sig vara instabila map A-polynomet. Ordning 9 var den modell med lägst ordning som hade ett tillfredsställande Bodediagram och simulering, se Fig.16 och 17. Residualernas korrelation uppförde sig nöjaktigt.

En modell baserad på Box-Jenkinsstruktur beräknades. Modeller med ordning 2 till och med 9 togs fram. I Tab.1. återfinns Akaikes FPE för respektive modell. Man ser att FPE når sitt minimum för ordning 7 och att funktionen har ett skarpt knä vid ordning 4. Eftersom simulering och studium av Bodediagram gav likvärdiga resultat för de båda modellerna bestämde vi oss för den lägre modellordningen. För att kunna jämföra Box-Jenkinsmodellen med den tidigare beräknade ARMAX-modellen återfinns Bodediagram och simulering för de båda modellerna i Fig.16 och 17. Att ARMAX-modellen kräver ett högre gradtal beror på att den inte kan modellera brusets lika bra som Box-Jenkins.



Figur 16. Bodediagram för systemet (heldragen), ARMAX av ordn.9 (streckad) samt Box-Jenkins av ordn. 4 (prickad)



Figur 17. Utsignal från systemet (heldragen), simulering av ARMAX av ordn. 9 (streckad) samt simulering av Box-Jenkins av ordn. 4 (prickad)

8. Sammanfattning

Identifiering av en robotarms rörelse är en besvärlig uppgift i och med att identifieringen sker i sluten loop. På grund av resonanserna måste man gå högt upp i modellordning för att hitta en acceptabel modell. Denna kan dock reduceras med gott resultat.

Ekorrhjul

Projekt i kursen Processidentifiering

Institutionen för Reglerteknik

LTH

Kenth Nilsson E86

Istephan Tamras E86

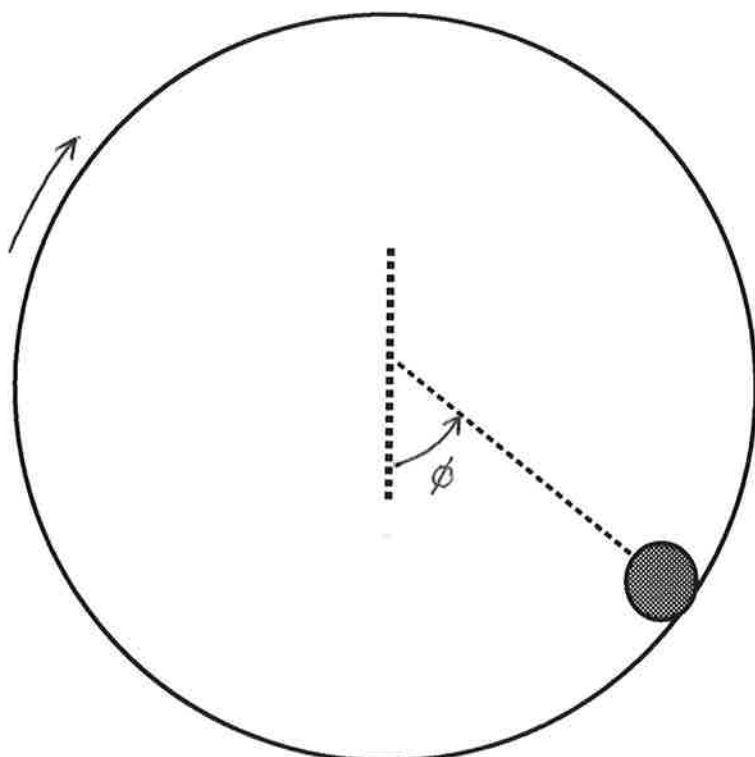
December 1990

1. Introduktion

Rapporten behandlar ett identifieringsprojekt i kursen Processidentifiering vid Institutionen för Reglerteknik, LTH. Målet med projektet var att ta fram en någorlunda bra modell av processdynamiken för att sedan kunna reglera processen. Som hjälpmedel vid identifieringen användes MATLAB samt System Identification Toolbox.

2. Processbeskrivning

Processen finns avbildad i fig. 2.1. Denna går under benämningen "ekorrhjul". En likströmsmotor driver runt ett aluminiumhjul och påverkar på så sätt en kula att inta ett visst läge beroende på inspänningen. Vinkeln ϕ som kulan avviker från vertikalanteret agerar som utsignal ifrån processen. Det visar sig emellertid att processen har ett nollställe nära noll, vilket innebär att kulan kommer att inta jämviktsläge ($\phi = 0$) efter en kort stund, oberoende av spänningen. Målsättningen blir därför att försöka reglera kulan kring jämviktsläget då denna utsätts för yttre påverkan.



Figur 2.1 Ekorrhjulet

3. Experimentellt

3.1 Visuell identifiering

- Resonans p.g.a kulan uppträder som en pendel. Resonansfrekvensen uppskattas till 1 Hz vilket medför $\omega \simeq 6\text{rad/s}$. Dämpningen (ζ) uppskattas till 0.1 – 0.4.
- Vinkelgivaren mättar vid 10V vilket motsvarar en vinkel ϕ på ca 30°. Lämpligt arbetsområde blir därför

$$u = [0.5V, -0.5V]$$

samt

$$\phi = [30^\circ, -30^\circ]$$

- Tidsfördröjningen från att spänning pålagts till kulan börjar röra på sig estimeras till mindre än en sekund. Tidsfördröjningen vid avstängning av motorn tills kulan intar jämviktsläge uppskattas till 4 sekunder.
- Uppskattning av modellordning: Spänning till vinkelhastighet 1:a ordningen. Vinkelhastighet till kulans läge 2:a ordningen (pendel). Ett 3:e ordningens system borde således passa.
- Val av samplingstid: Eftersom målsättningen var att kunna dämpa resonansfrekvensen så är ett lämpligt värde på $\omega \simeq 6\text{rad/s}$. Computer Controlled Systems rekommenderar $\omega h = 0.15 - 0.5$, så vi väljer $h = 0.1\text{s}$ och hoppas det går bra.

3.2 Experimentvillkor

Ett program kallat logger används för att sampla spänningen och vinkeln på processen under experimenten. Insignalen är en PRBS-signal där vi väljer $per=5$ för att kunna excitera resonansfrekvensen. Vi väljer att använda 400 datapunkter vilket blir en försökstid på 40 s. Resultatet av experimentet visas i fig. 3.1

4. Identifieringen

Efter inspelning av data så drar vi bort insignalens och utsignalens medelvärden för att få bort eventuell bias. En ARMAX-modell

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

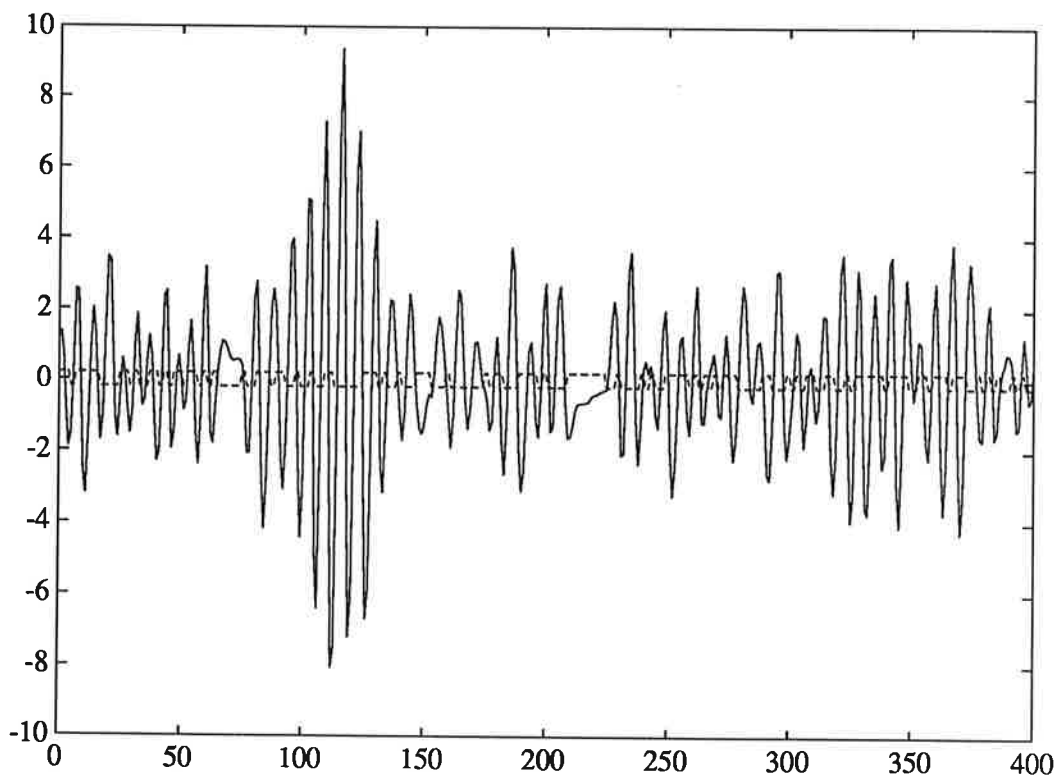
ansätts med hjälp av MATLAB-kommandot `ARMAX([y u], [nA,nB,nC,k])` och efter lite prövande kommer vi fram till att en modell med parametrarna

$$nA = 3, nB = 3, nC = 3, k = 1$$

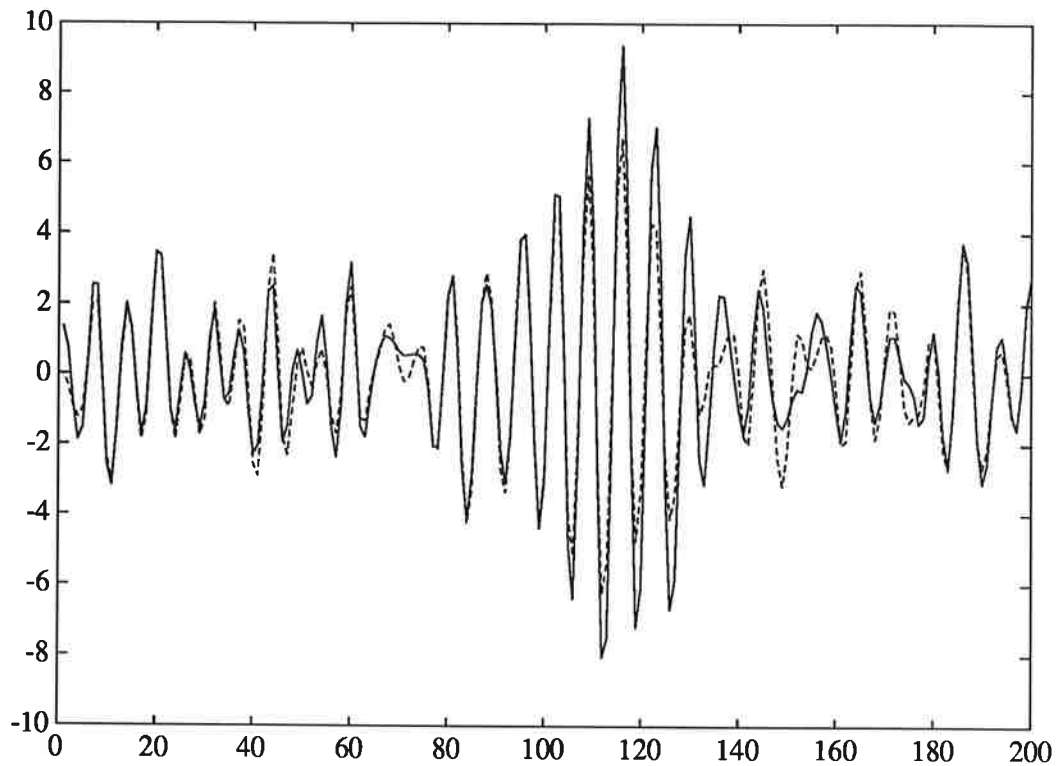
nog beskriver processen bäst.

5. Validering

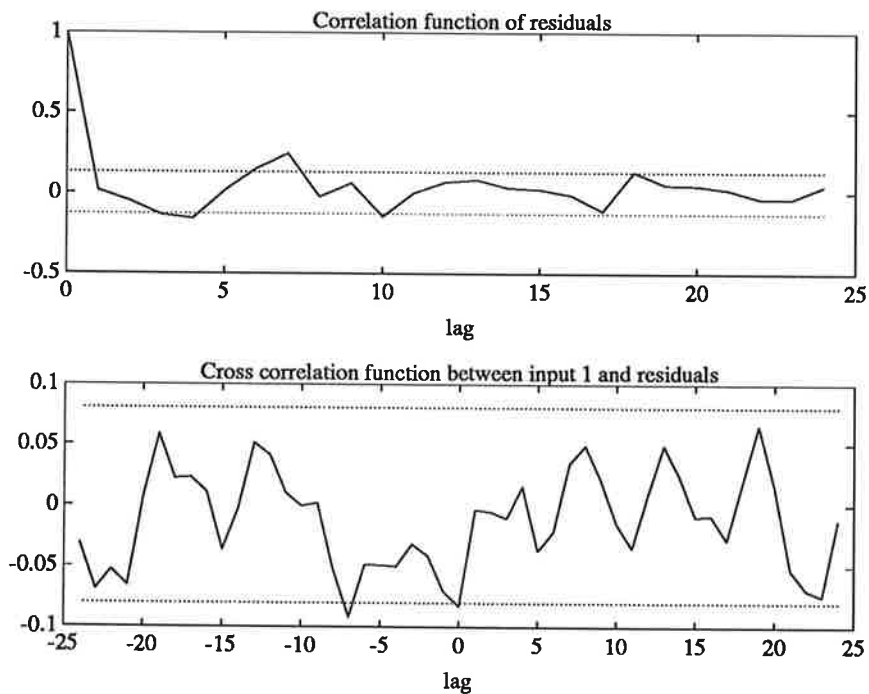
En jämförelse mellan verklig utsignal och modellens utsignal visas i figur 5.1. Man kan se att modellen stämmer väl överens med verkligheten i detta fallet. I fig. 5.2 visas korrelationsfunktionen hos residualerna samt korskorrelationen mellan residualerna och insignalen. Vi ser att både två liknar vitt brus och ligger innanför konfidensintervallen. Kommandot `Trf` i MATLAB används för att ta fram överföringsfunktionen samt brusspektrum för modellen. I fig. 5.3 visas bodediagrammet för B/A . Vi kan se att den förväntade resonansen vid 1 Hz framträder. Vi ser även det förväntade nollstället i nollan (Bodediagrammet har positiv lutning för små frekvenser). I fig. 5.4 visas bodediagrammet för brusspektrum C/A . Vi kan se att modellen klarar av bruset bra upp till resonansfrekvensen vid 1 Hz. Vårt system beskrivs antagligen bättre med skilda



Figur 3.1 Utsignal (Heldragen) och insignal (Plottad) vid experimentet

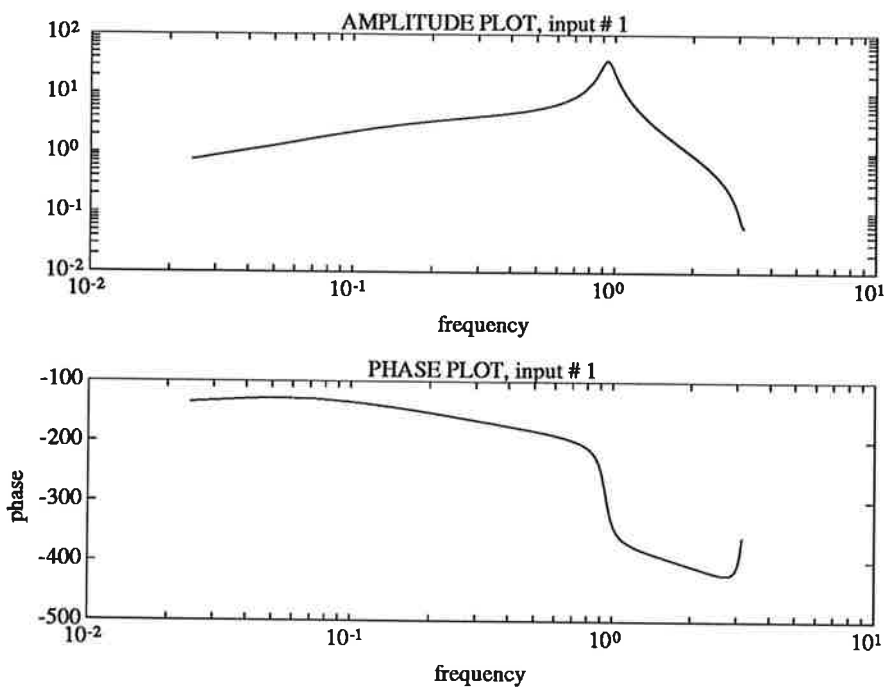


Figur 5.1 Verklig utsignal (Heldragen) samt modellens utsignal (Plottad)

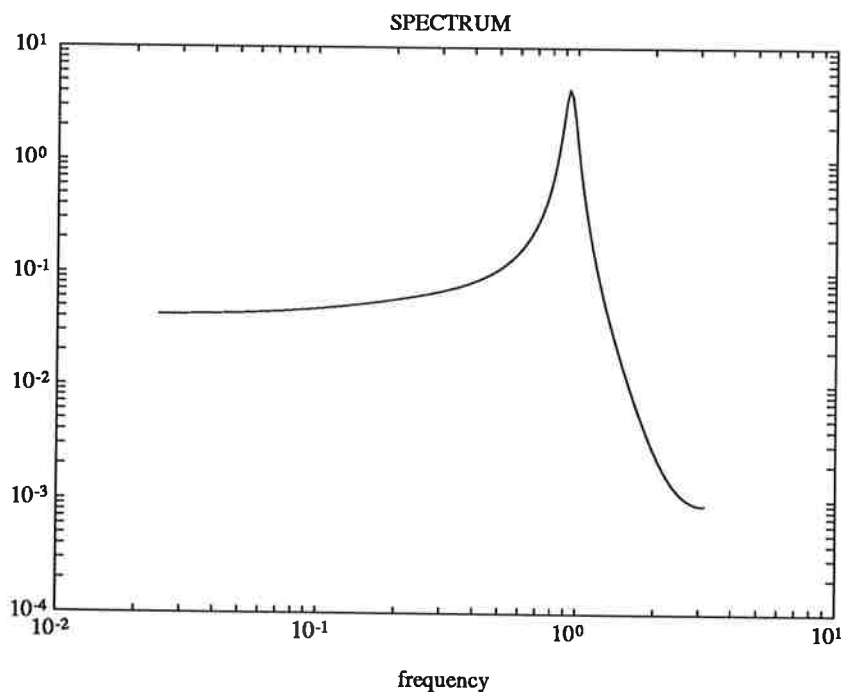


Figur 5.2 Korrelationsfunktionen hos residualerna samt korskorrelationen mellan residualerna och insignalen

system och bruspoler vilket leder till att en Box-Jenkins modell borde ge en bättre modell vid högre frekvenser. Vi har även testat en sådan modell men

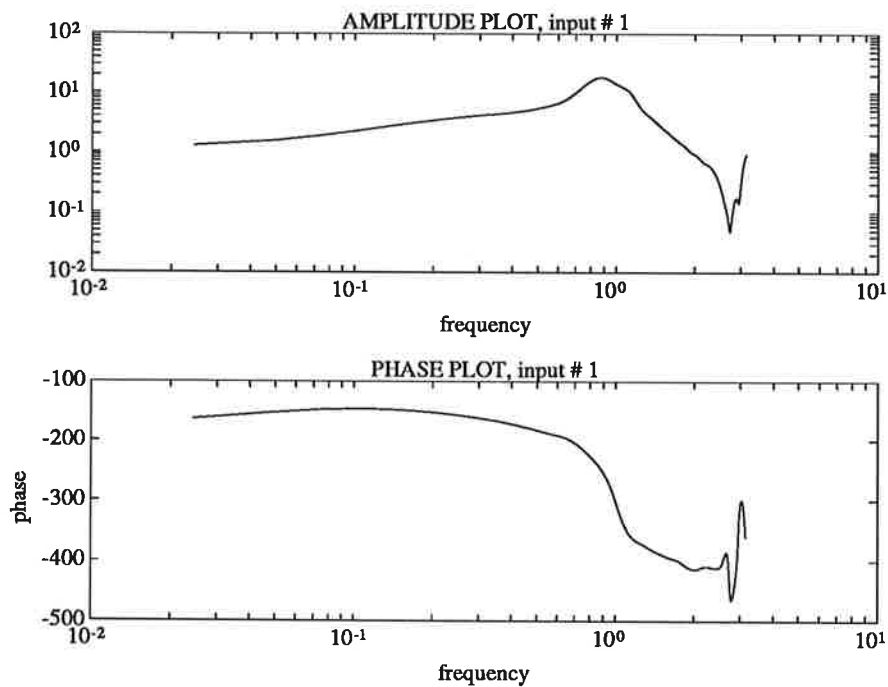


Figur 5.3 Bodediagram för överföringsfunktionen hos modellen

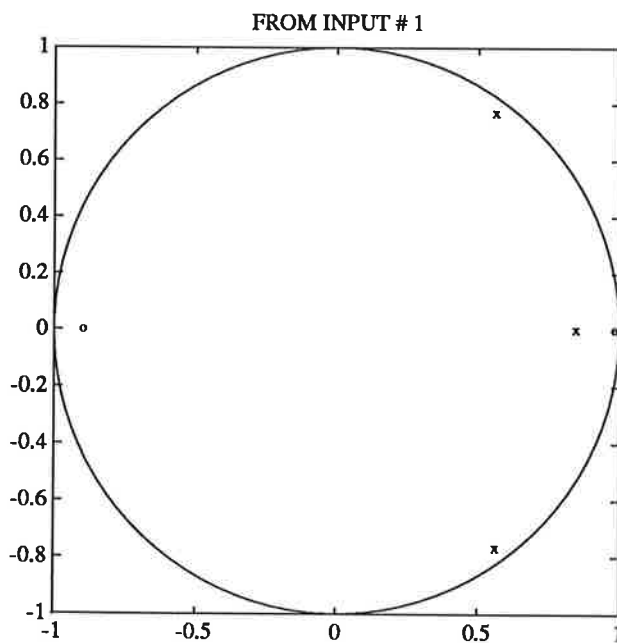


Figur 5.4 Bodediagram för brusspektrum hos modellen

eftersom skillnaderna blir så marginella i frågan om modellens beteende och vi bara är intresserade av frekvenser upp till resonansfrekvensen så nöjer vi oss med ARMAX-modellen. Nästa steg är att göra spektralanalys på använda data . Detta görs med hjälp av kommandot Spa i MATLAB . I fig. 5.5 visas Bodediagramet för överföringsfunktionen då vi använt detta kommandot. Även i detta diagram framträder det förväntade nollstället (jämför med figur 3.2). För att plotta pol-nollställediagrammet för modellen används kommandot Zpplot i MATLAB . Detta diagram visas i fig. 5.6 där man kan se att vi



Figur 5.5 Bodediagram för överföringsfunktionen vid användandet av verkliga data



Figur 5.6 Pol-nollställediagram för modellen

bl.a har två dåligt dämpade poler, samt ett nollställe i $q = 1$ vilket motsvarar $s = 0$ i kontinuerlig tid. Detta var precis vad vi hade förväntat oss med utgångspunkt ifrån den visuella identifieringen och vi anser därför att modellen är verifierad.

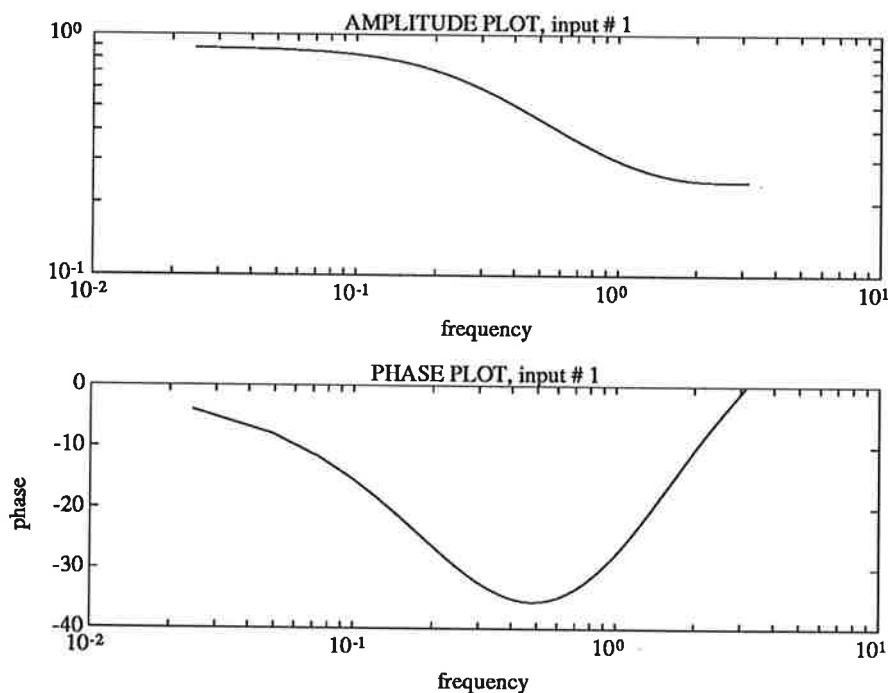
6. Reglering

6.1 Regulatordesign

Polplacering enligt Computer Controlled Systems används. Det önskade karakteristiska polynomet $A_m(q)$ samt observerarpolynomet $A_o(q)$ måste därför väljas. Vi väljer att relatera det karakteristiska polynoms poler till kontinuerlig tid och väljer dessa på en cirkelbåge med radien $\omega_m = 6 \text{ rad/s}$ och med halva öppningsvinkeln $= 45^\circ$. Observerarpolerna väljs på samma sätt men med $\omega_o = 2\omega_m$. De kontinuerliga polerna översätts sedan till diskret tid genom $z = e^{sh}$. Eftersom vi har en liten bias i utsignalen så väljs en icke integrerande regulator ty annars hade regulatorn fortsatt att integrera och vi hade till slut fått en "styrsignalexplotion". Med hjälp av MATLAB tas koefficienterna i R och S-polynomen fram. Följande R,S-regulator erhölls:

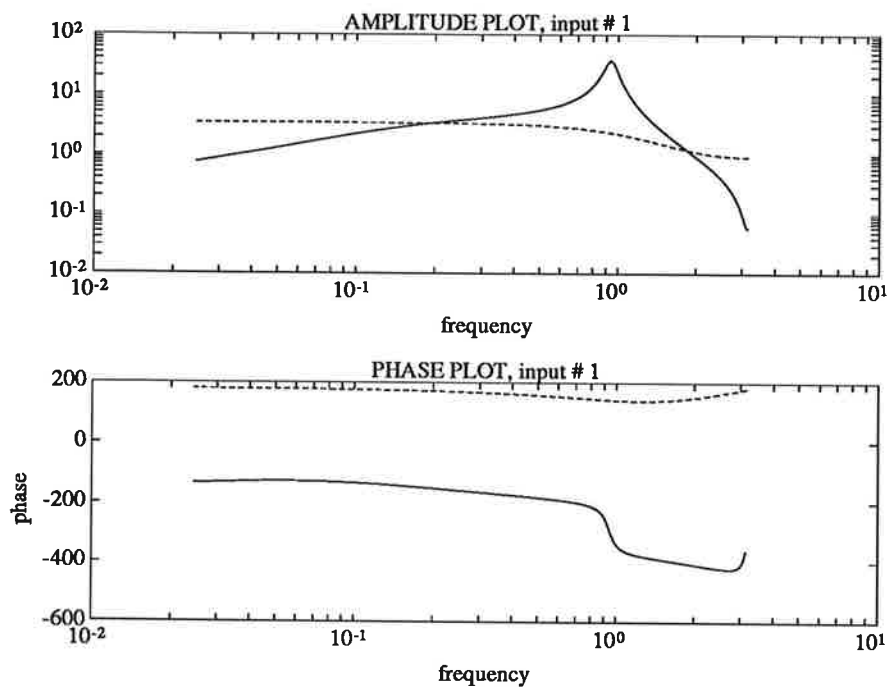
$$R(q) = q^2 - 0.3509q - 0.0694$$

$$S(q) = -0.0094q^2 + 0.1417q - 0.0681$$

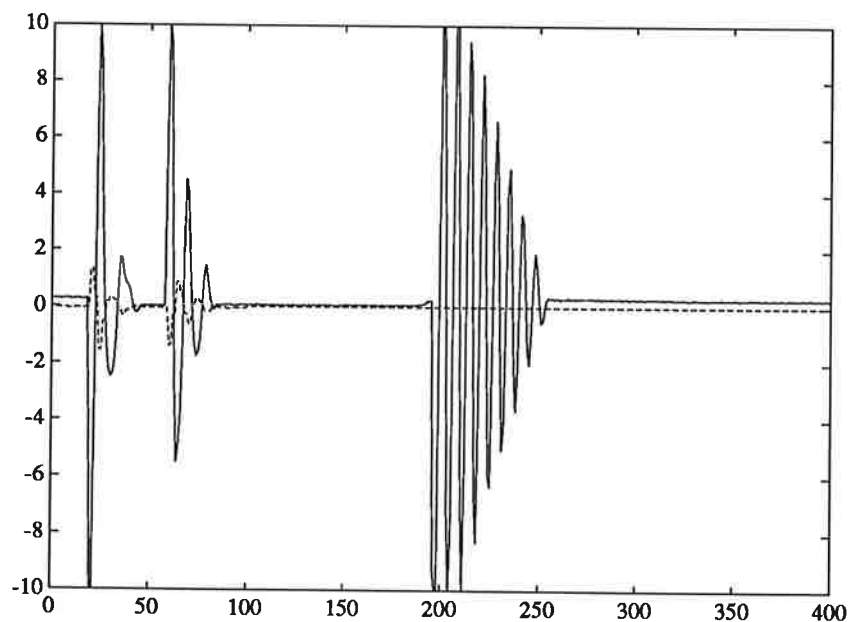


Figur 6.1 Bodediagram för regulatorn $-S/R$

Bodediagramet för regulatorn $-S/R$ visas i fig.6.1 och bodediagrammet för det öppna och slutna systemet visas i fig.6.2. Vi ser att resonansfrekvensen är kraftigt dämpad. Det är visserligen lägre frekvenser också men vi anser oss ändå relativt nöjda med regulatorn.



Figur 6.2 Bodediagram för det slutna och öppna systemet



Figur 6.3 Utsignal (Heldragen) och styrsignal (Plottad) vid yttre påverkan på processen

6.2 Resultat av regleringen

Vår målsättning var att försöka dämpa resonansen. Detta tycker vi att vi uppnått. I fig. 6.3 visas utsignalen i 3 fall. De två vänstra visar utsignal och styrsignal då vi genom yttre påverkan ändrat kulans läge 30° åt vänster respektive 30° åt höger. Det tredje visar processens utsignal vid samma yttre påverkan men utan regulatorn inkopplad.

Identifiering av robotaxel

ett projekt i kursen
Processidentifiering

på
Institutionen för Reglerteknik

av
Peter Nilsson

handledare
Ola Dahl
Klas Nilsson

december 1990

innehåll

1. Introduktion	1
2. SMIXP	1
3. Icke-parametriska metoder	2
4. Parametriska metoder i tidsplanet	5
5. Parametriska metoder i frekvensplanet	8
6. Jämförelse mellan metoderna i tidsplan och frekvensplan	10
7. Referenser och Hjälpmedel	12
8. Appendix	13

1. Introduktion

Detta projekt ingår i kursen Processidentifiering vid Institutionen för Reglerteknik, LTH. En elastodynamisk modell skulle identifieras fram. De data som använts kommer dels från en ASEA IRB-3000 och dels från institutionens egen IRB-6. Mätsignalerna kallas i fortsättningen Västeråsdata respektive LTHdata. Parametrisk identifiering i tidsplanet visade sig ge höga ordningstal, som dock kunde reduceras, medan en metod med parametrisk identifiering i frekvensplanet kombinerat med viss identifieringskunskap gav bra resultat vid mycket lägre ordning.

Som hjälpmedel vid identifieringen har Matlab och Identification Toolbox använts. Robotexperimenten och datainsamlingen vid institutionen har skett med realtidsprogrammering i Sun-VME-miljö.

2. SMIXP (System, Model, Identification, eXperiment, Purpose)

System

Systemet är en robotaxels dynamik. Främst har första axeln studerats eftersom där finns tydligast resonanser och därmed störst anledning att reglera väl. Insignal är strömmen dvs momentet och utsignalen är vinkelhastigheten.

Model

De modeller som prövats för tidsserieanalys är:

$$\text{ARX: } A(q^{-1})y(t) = B(q^{-1})u(t-k) + e(t)$$

$$\text{ARMAX: } A(q^{-1})y(t) = B(q^{-1})u(t-k) + C(q^{-1})e(t)$$

$$\text{OE: } y(t) = (A(q^{-1}) / B(q^{-1}))u(t-k) + e(t)$$

$$\text{Box-Jenkins: } y(t) = (B(q^{-1}) / F(q^{-1}))u(t-k) + (C(q^{-1}) / D(q^{-1}))e(t)$$

För metoderna i frekvensplanet antas:

$$H(s) = (B(s) / A(s)) * e^{-sT}$$

Identification

De modeller som beskrivits ovan finns i Identification Toolbox och de använder olika identifieringsalgoritmer: ARX används med minstakvadratmetoden (LS) och ARMAX, Output Error och Box Jenkins använder prediktionsfelsmetod.

eXperiment

Alla experiment utfördes i slutna loop med PI-regulator.

Västeråsdatan var inspelade med en sinusreferens med logaritmiskt svepande frekvens och en samplingsfrekvens på 496 Hz. LTHdatan spelade jag in själv med hjälp av lite realtidsmoduler och en koppling mellan VME och Matlab på Sun (modulen MatComm). Referenser genereras i Matlab och skickas ner till programmen i VME som ställer ut signaler och loggar de data man är intresserad av. Den loggade datan, i detta fall ström och vinkelhastighet, skickas sedan upp till Matlab. På detta vis gick experiment och mätdatainsamling mycket lätt och blev tidsmässigt en helt försumbar del i identifieringsarbetet.

Som referens användes dels bredbandig slumpdata, dels sinussvep. Sinus-svepningar är trevliga och pedagogiska för resonanser och antiresonanser ses direkt vid experimentet och i insignal-utsignal plottar. Frekvenserna till sinusfunktionen sveptes dels logaritmiskt, dels linjärt.

Koherensfunktionen beräknades för att se om mätserierna var att lita på, dvs om de beskrev en linjär överföringsfunktion. Om denna ej är nära ett kan det tyda på störning på y, störning på u (insignalen representeras inte av u), olinjäriteter (dvs det finns ingen överföringsfunktion mellan u och y) mm.

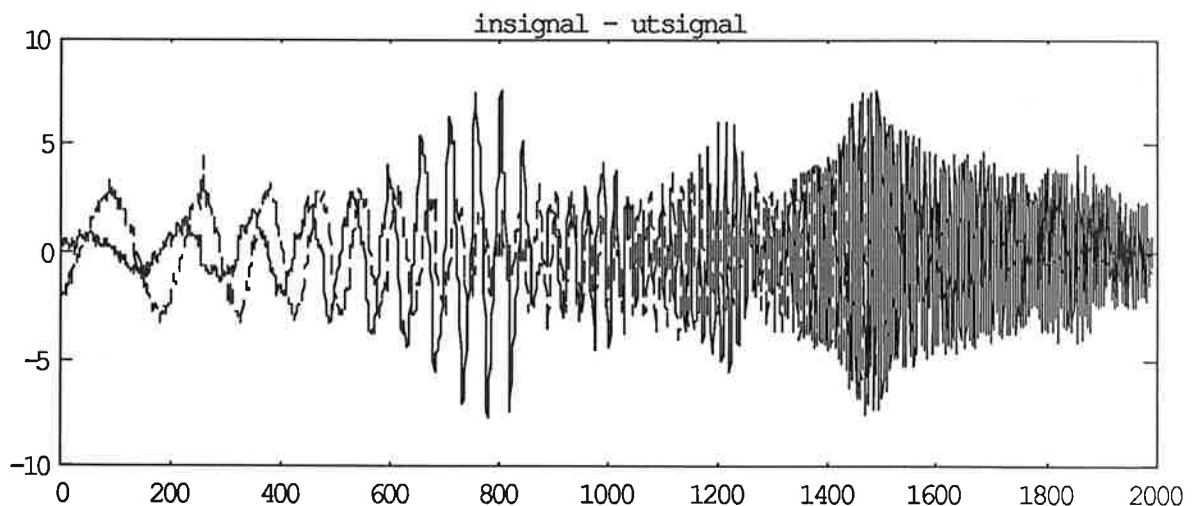
Purpose

Syftet var att efter identifiering kunna göra en bra regulator (och att förstå processidentifiering).

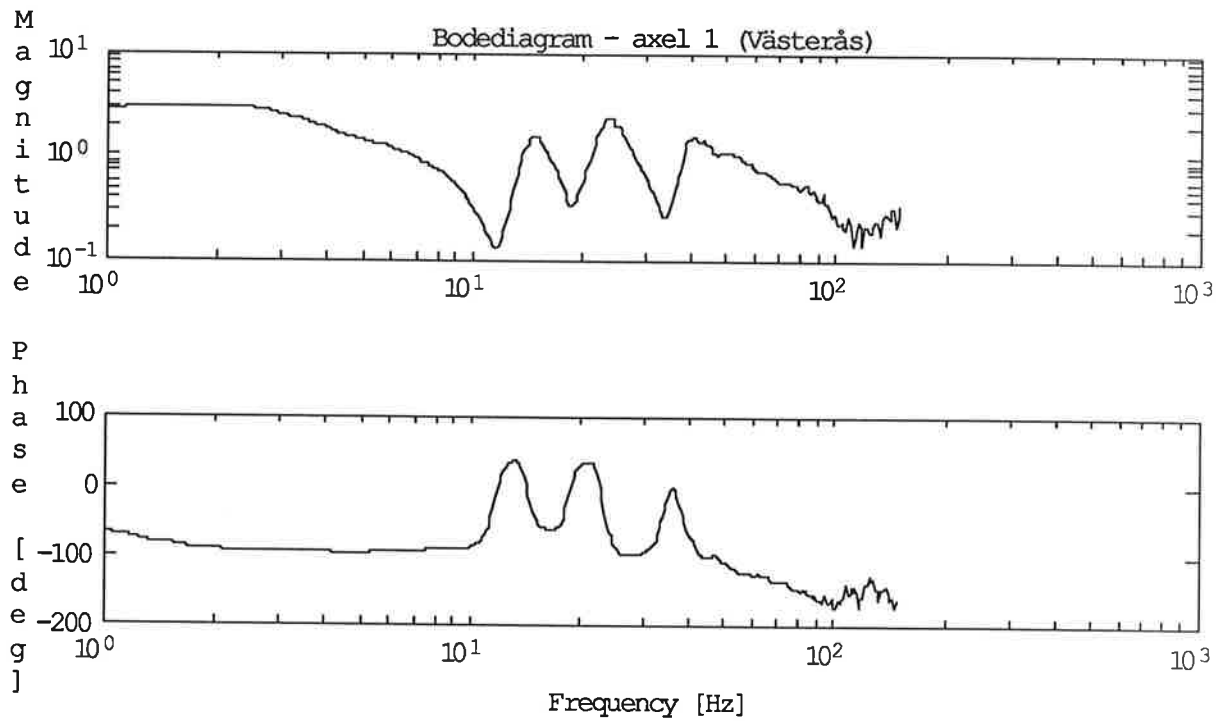
3. Icke-parametriska metoder

Överföringsfunktionen kan skattas med korspektrum genom insignalens autospektrum: $H = C_{uy}/C_{uu}$. Denna spektralanalys ger, om man tar ett större fönster än default i SPA i Identification Toolbox, tydliga resonanser.

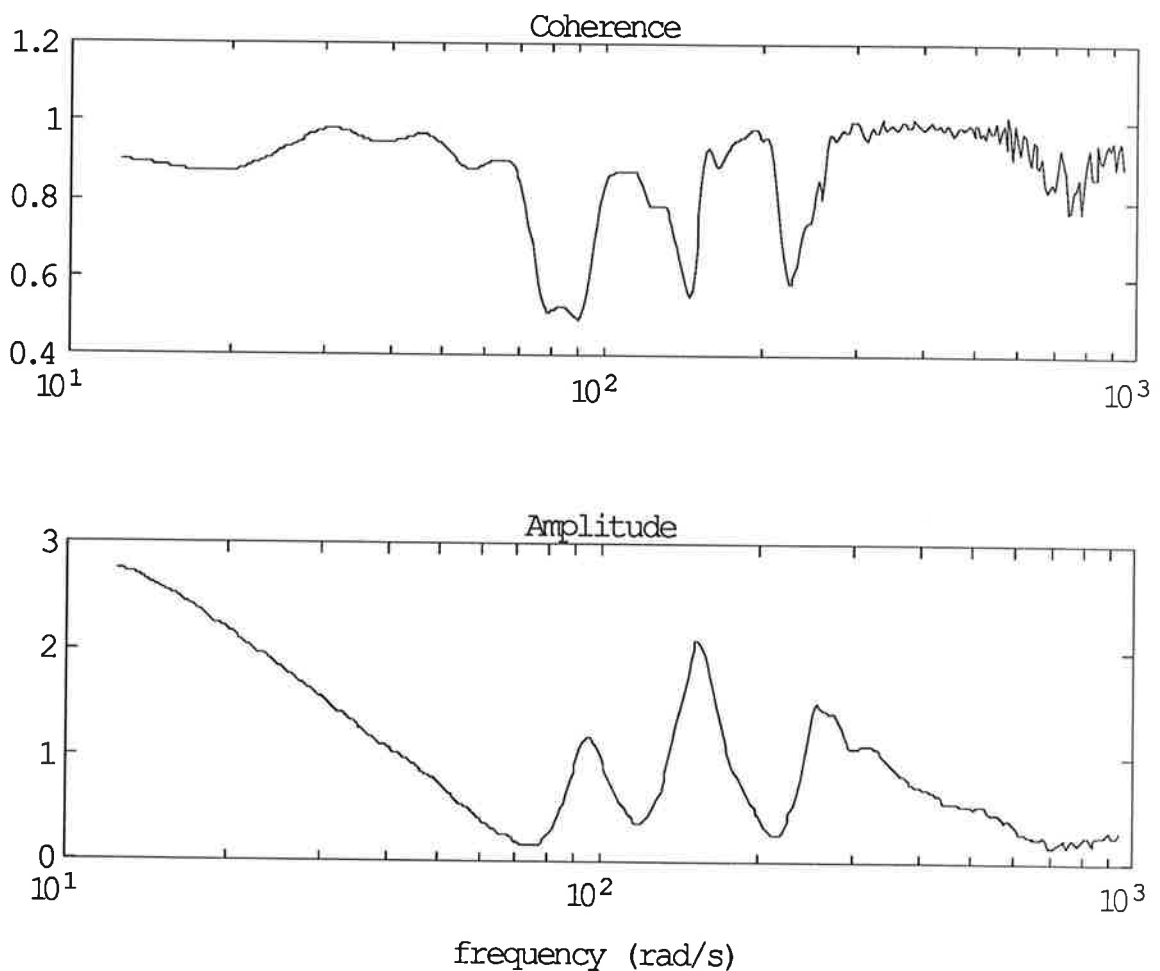
Bodediagrammet för denna analys tas i fortsättningen som ett, för teknologen viktigt, facit. Resultatet visas i figurerna 1-6.



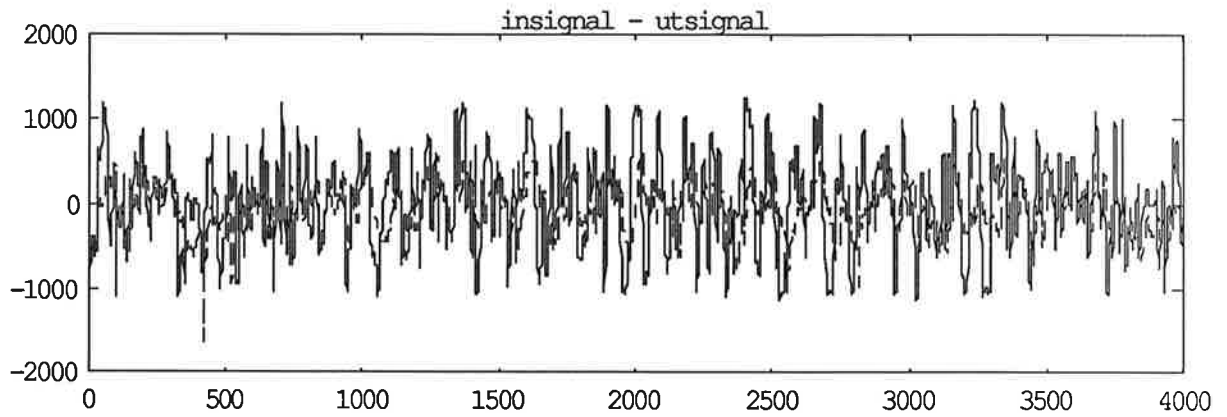
figur 1. Processens insignal och utsignal. Referensen till regulatorm är en logaritmiskt svept sinus och man ser tydliga resonanser. (Västeråsdata, sampelfrekvens = 496 Hz)



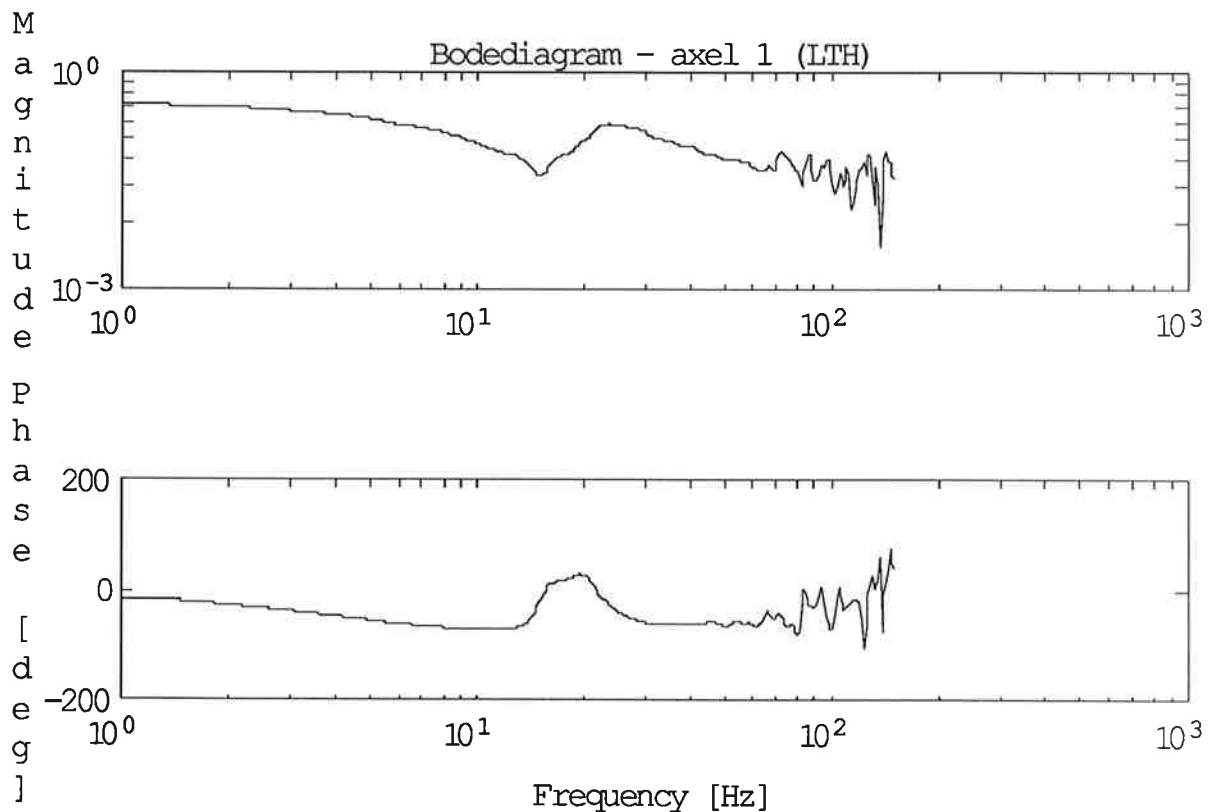
figur 2. Bodediagram för Västeråsdatan, axel1. Tre tydliga resonanser och en integrator ses.



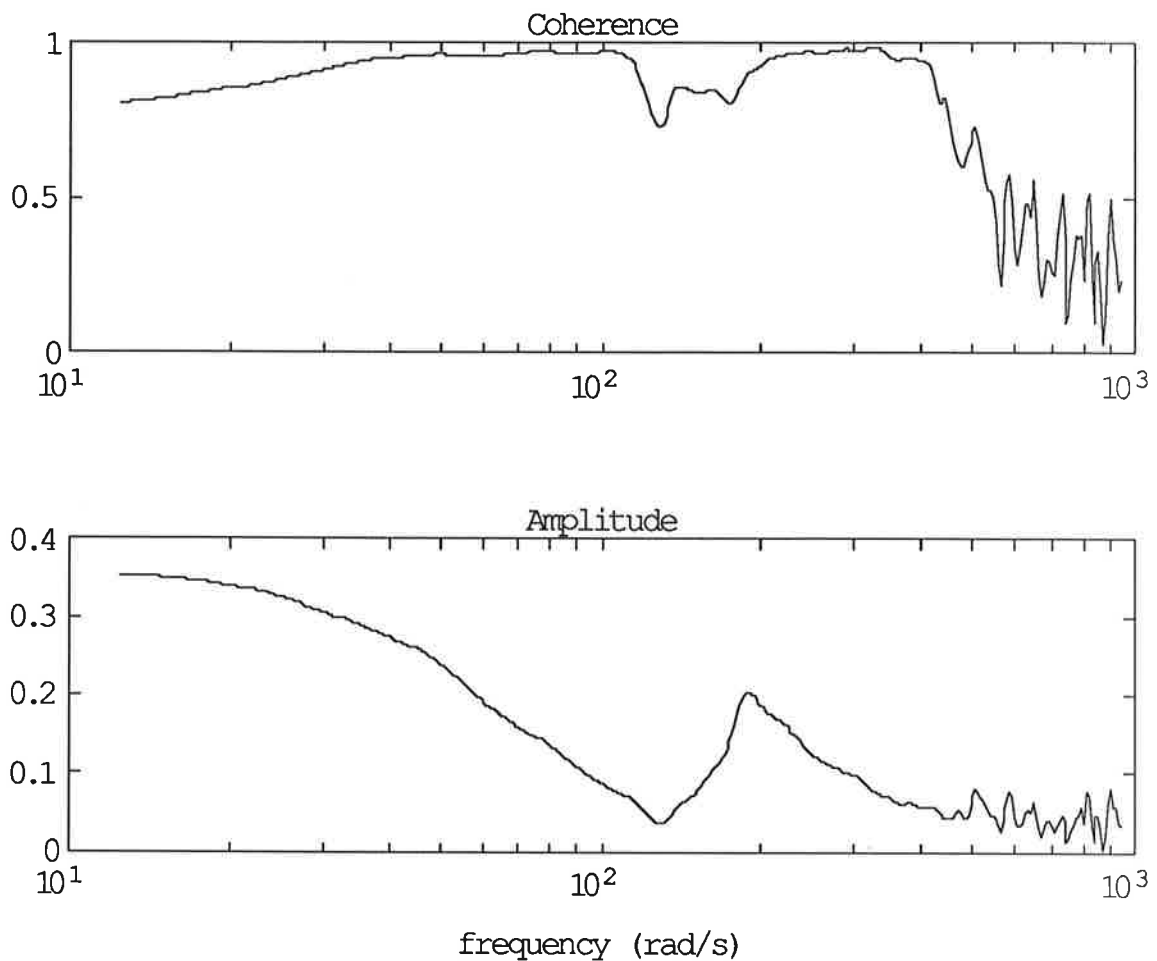
figur 3. Koherensen och Bodediagrammet. Koherensen går visserligen ner vid antiresonanser men det är väntat för där går mindre energi genom systemet.



figur 4. Processens insignal och utsignal. Referensen till regulatorm är en slumpsekvens, genererad med RAND i Matlab, som ska efterlikna vitt brus. (LTHdata, sampelfrekvens = 1000 Hz)



figur 5. Bodediagram för LTHdatan, axel 1. Över 70 Hz blir spektrumet tveksamt. Jämför med Koherensdiagrammet i figur 6.

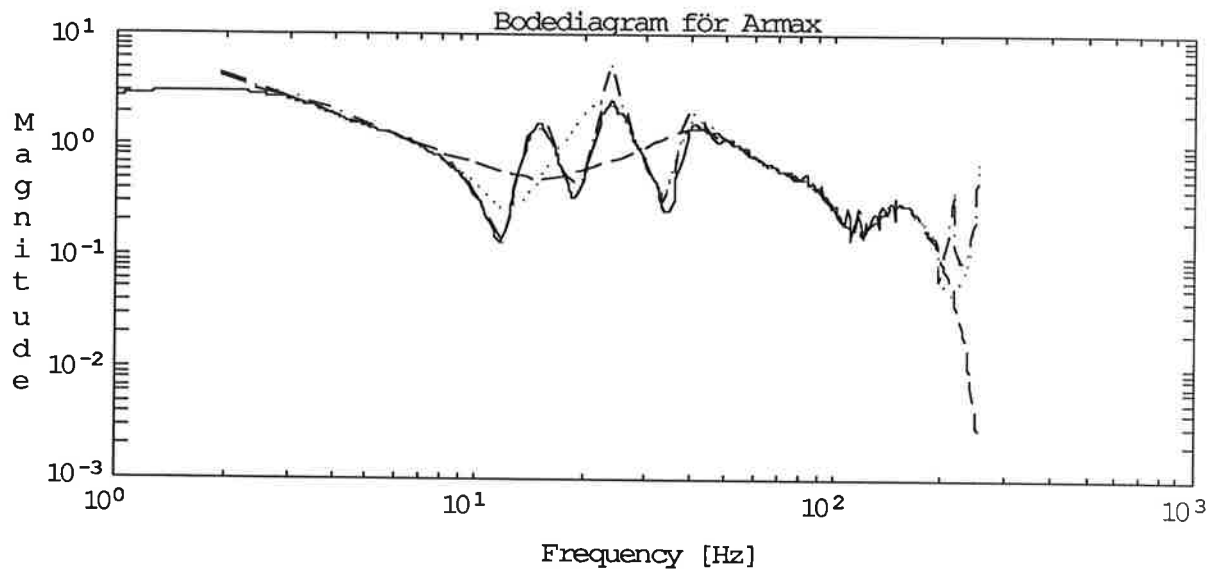


figur 6. Koherens för LTHdatan är god till mycket god. Över 400 rad/s (= 64 Hz) är datan inte att lita på för identifiering.

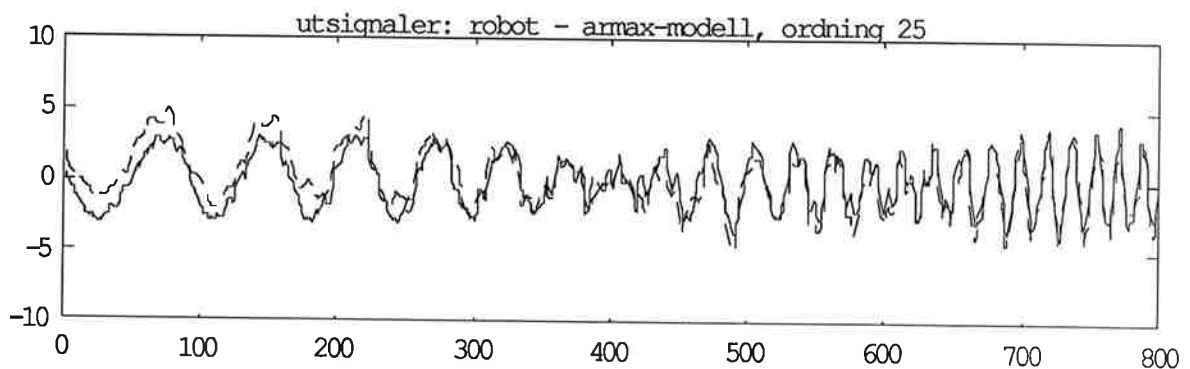
4. Parametriska metoder i tidsplanet

Modellanpassning

För att få en uppfattning om hur bra metoderna för tidsserieanalys var gjorde jag några anpassningar med Armax, Box-Jenkins och Output Error. Några exempel på resultaten från Armax visas i figur 7:



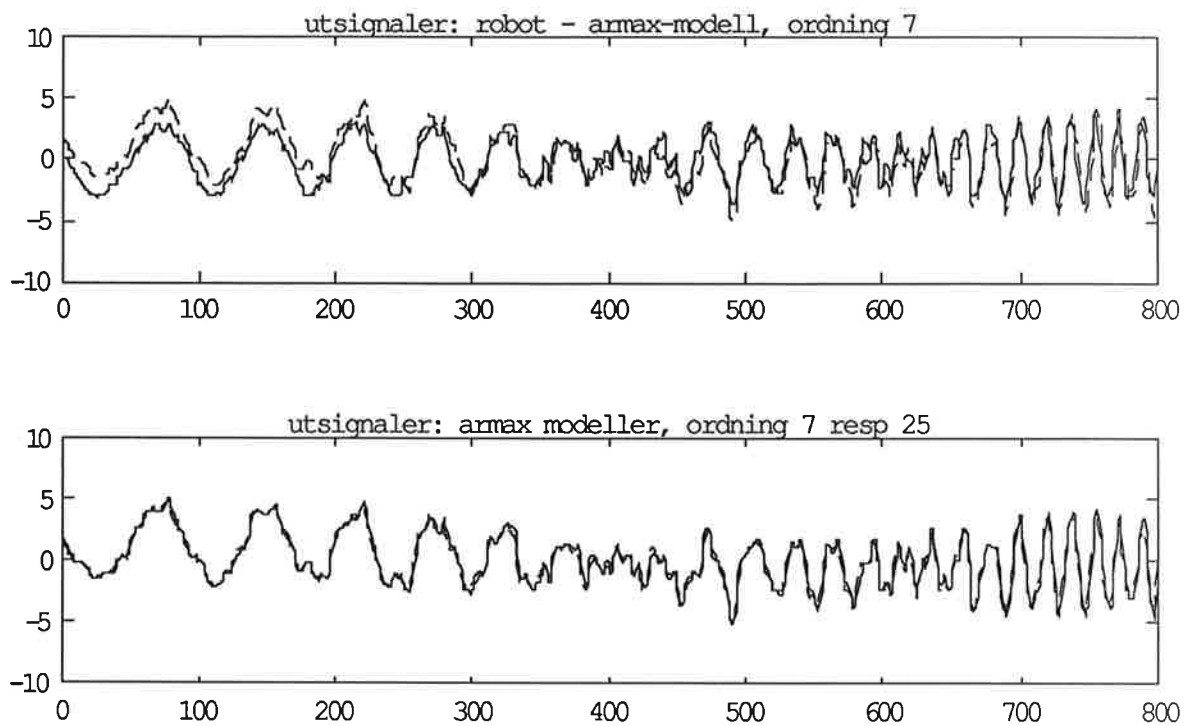
figur 7. Helderagen linje visar spektralanalyserad data och övriga linjer Armax-modeller av ordning 10,15 och 25. Modellererna klarar en, två respektive tre resonanser. (Västeråsdata)



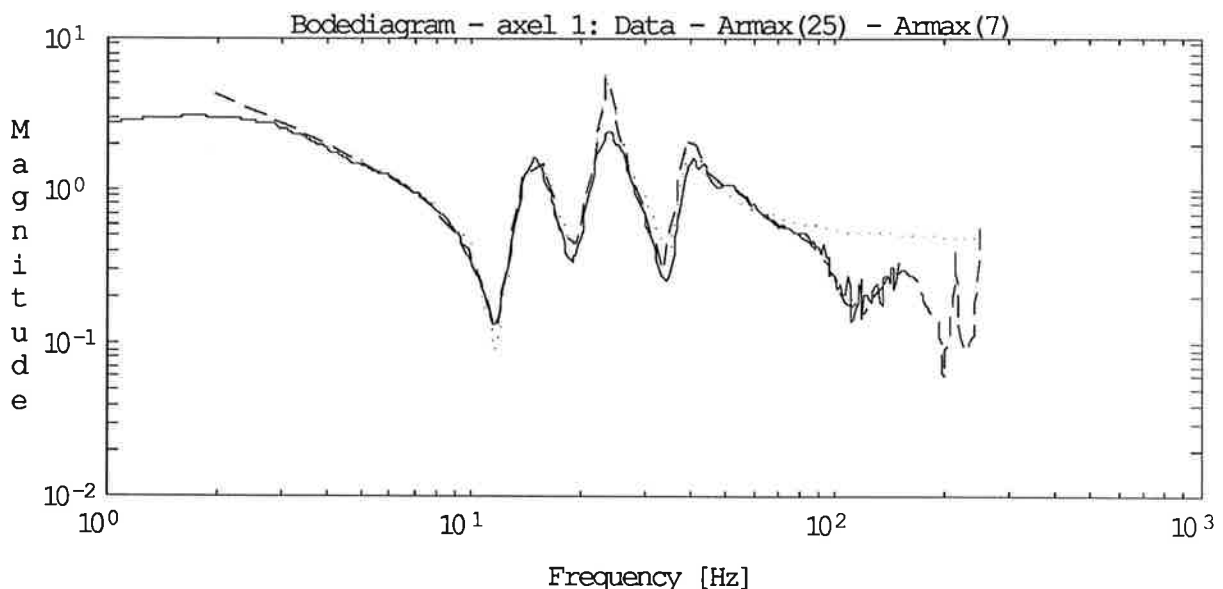
figur 8. Simulering av Armax-modellen, ordning 25.

Reduktion

Om man omformar modellen till en balanserad realisering ser man att vissa tillstånd, relativt sett, har liten inverkan på insignal-utsignalsambandet. Detta tillsammans med att bodediagrammet antyder ordning sju gör att man gärna försöker en reducering. Efter att ha strukit 18 tillstånd i den balanserade realiseringen ser man i följande plottar att detta till synes grova ingrepp knappt märks i en simulering.



figur 9. Skillnaderna mellan Armax av ordning 25 resp 7 är försumbara i en simulering.



figur 10. Heldragen linje visar spektralanalyserad data, streckad och prickad visar Armax-modeller av ordning 25 respektive 7 (reducerad).

Giltighetstest

För att få en uppfattning om hur bra modell som erhållits gjordes följande tester:

- Det från spektralanalysen erhållna "facit" plottades mot tidsseriens frekvensfunktion. Därefter får identifieraren med tillhörande öga och hjärna bedömma om modellen är bra. Se figurer 7 och 10.
- Residualen ska bli vitt brus. När residualens kovariansfunktion plottades blev den nära noll (krävs inte för tidsförskjutning noll).

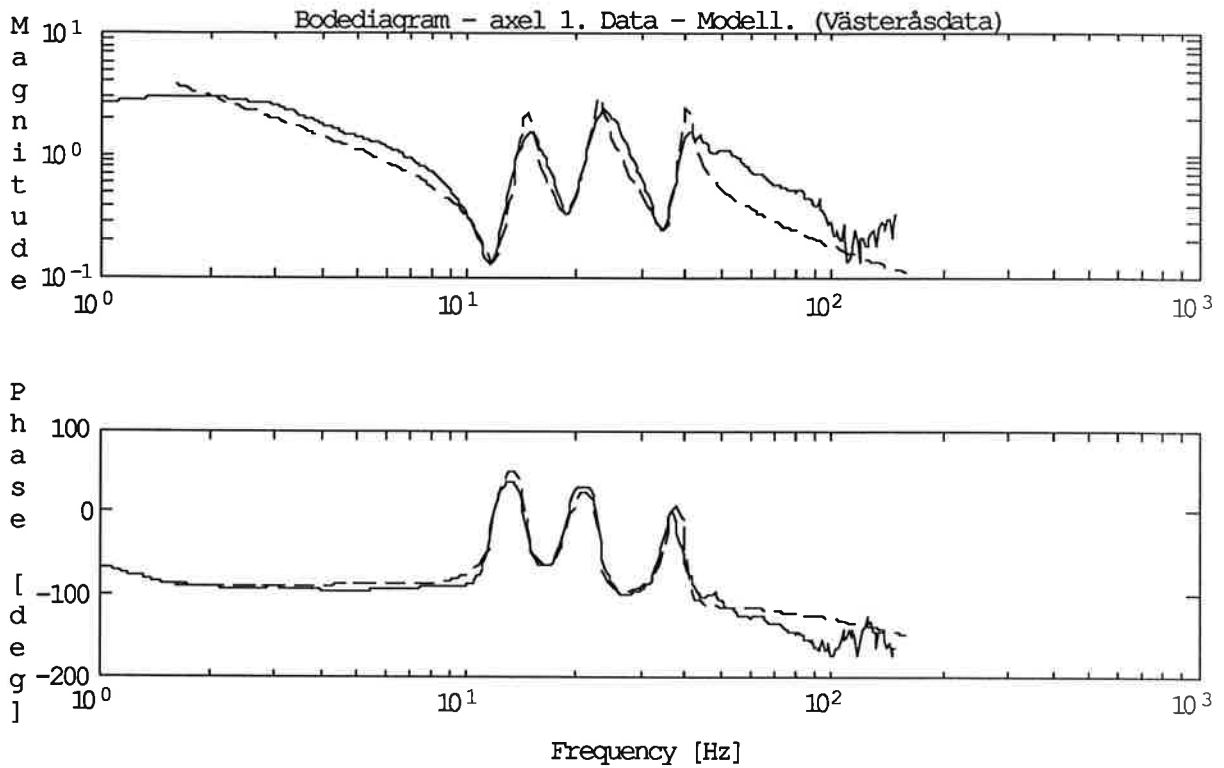
- Residualen ska ej bero på tidigare insignaler: $E(u(t)e^{(t+T)}) = 0$, $T > 0$. Utan återkoppling ska $E(u(t)e^{(t+T)}) = 0$, alla T . I en plot såg man på bulan för $T < 0$ att återkoppling använts.
- Plotning av residualen. När insignalen var en svept sinus syntes "sinusen" i denna plot om modellen var dålig (av för låg ordning).
- I en simulering ska utsignalen från modellen självklart följa den verkliga utsignalen. Resultat kan ses figurer 8 och 9. Denna verkliga utsignal bör ej vara densamma som använts för identifieringen. I mitt fall, där den praktiska insamlingen av data går snabbare än en inloggning, innebar detta inget problem.

5. Parametriska metoder i frekvensplanet

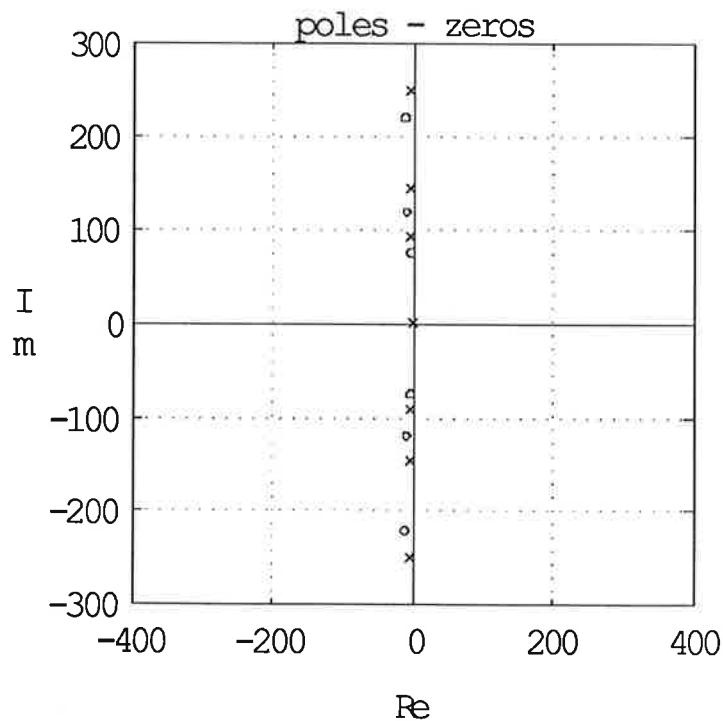
Armax krävde för hög ordning pga att den viktade alla frekvenser lika mycket. Det vore därför önskvärt om man kunde föra över lite kunskap från identifieraren till identifieringsalgoritmen så att denna förstod vad som var "viktigt". Detta kan göras med en direkt minstakvadratanpassning i frekvensplanet. I Matlab kan detta göras genom att man går in i bodediagrammet och väljer ut intressanta punkter såsom resonanser, antiresonanser och lågfrekvensnivån (detta görs genom klickning i Matlabs plotfönster). Därefter väljer man gradtal på A- och B-polynomen samt tidsfördröjning. En identifiering av axel 1, Västeråsdata går då till enligt:

1. Spektralanalys (SPA, Identification Toolbox) med något större fönster än default.
2. För in kunskap i identifieringen. Funktionen FR2TF (frequency function to transfer function) gör en bodeplot, i vilken det intressanta markeras medelst mustryckning, och frågar sedan efter gradtalen och fördröjning. I detta exempel finns tre antiresonanser vilket borde ge tre komplexkonjugerade nollställen, dvs sex nollställen. Vidare kan antalet poler antas vara sju, ty vi har tre resonanser och en integrator. Fördröjningen väljes till en halv sampelperiod.
3. Efter interpolering vid de valda frekvenserna anropas en funktion för minstakvadratanpassning i frekvensplanet.
4. Den erhållna överföringsfunktionen plottas mot "facit". Om resultatet därvid ej är tillfredsställande upprepas algoritmen från punkt 2. I annat fall har vi nu två polynom alternativt två vektorer med nollställen respektive poler, som nu kan användas för att bygga en RST-regulator.

Det goda resultatet och pol-nollställesdiagrammet som fås kan ses i följande figurer:

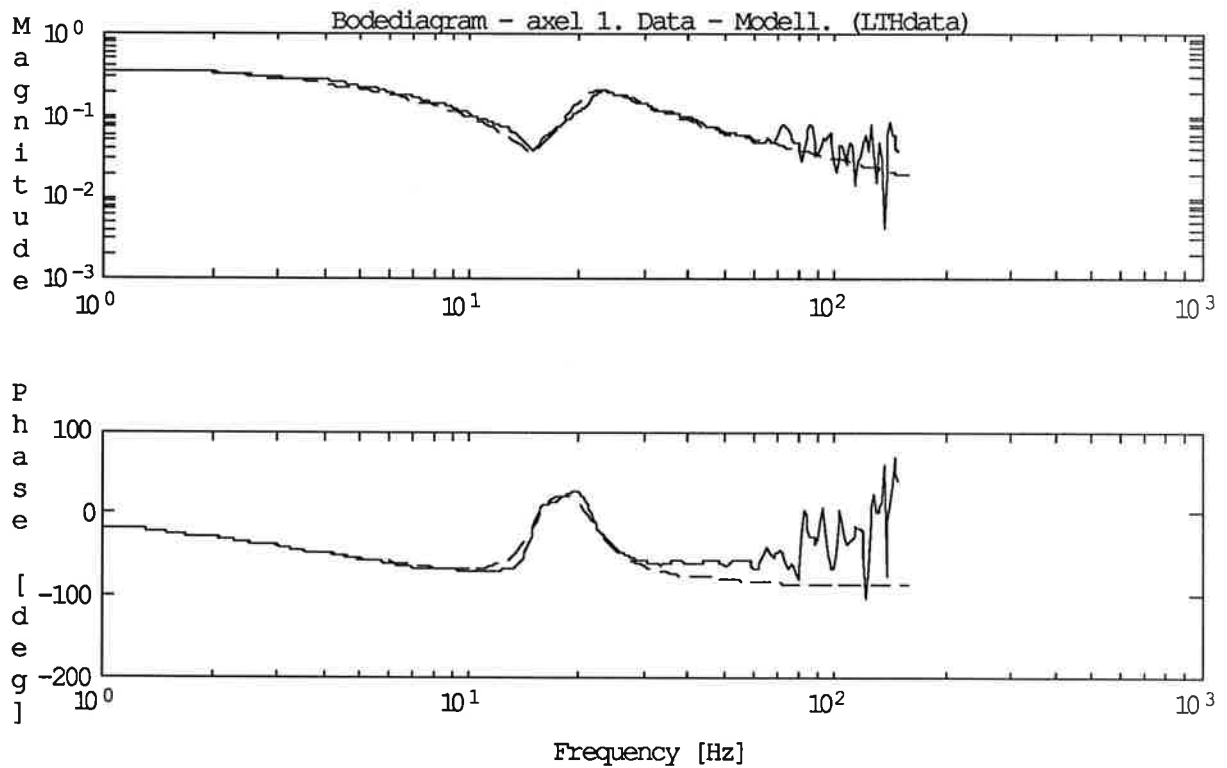


figur 11. Heldragen linje visar spektralanalyserad data och streckad visar modellens frekvensfunktion. Modellen är av ordning 7.

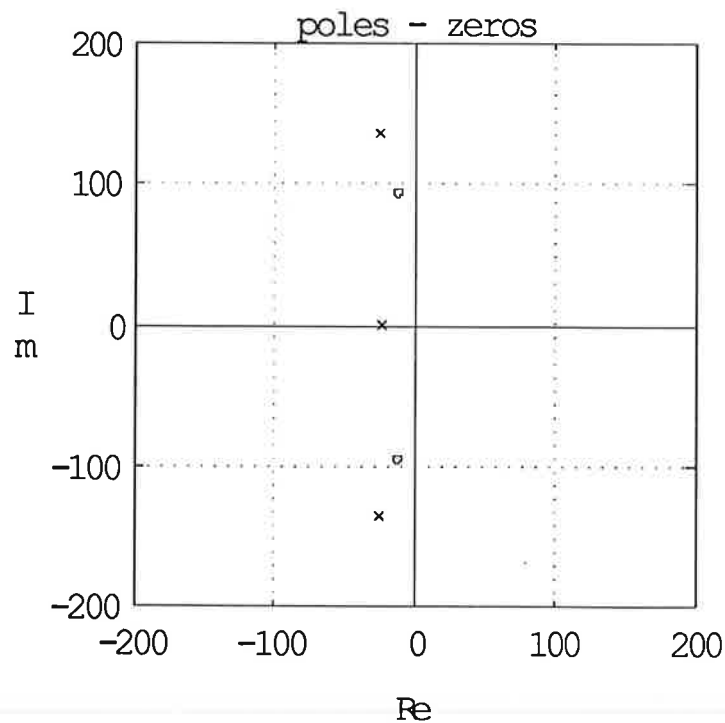


figur 12. Modellens poler och nollställen, markerade x respektive o, ligger mycket nära imaginära axeln. Integratorn i origo förklarar 90⁰-fasförskjutningen.

För LTHdatan väljes 3 poler och 2 nollställen och resultatet blir som i följande bild:



figur 13. Heldragen linje visar spektralanalyserad data och streckad visar modellens frekvensfunktion. Modellen är av ordning 3.



figur 14. Modellens poler och nollställen, markerade x respektive o.

6. Jämförelse mellan metoderna i tidsplan och frekvensplan

En jämförelse mellan modellanpassning i tidsplanet med Armax och minsta-kvadratanpassning i frekvensplanet kan fås dels i ett bodediagram, dels i en simulering. Man ser i följande plottar att resultaten är nästan desamma. Jag har alltså fått samma resultat genom att göra på följande två sätt:

1.

a) Anpassa en modell, t ex Armax, i tidsplanet. Välj ordning mycket över vad som kan antas.

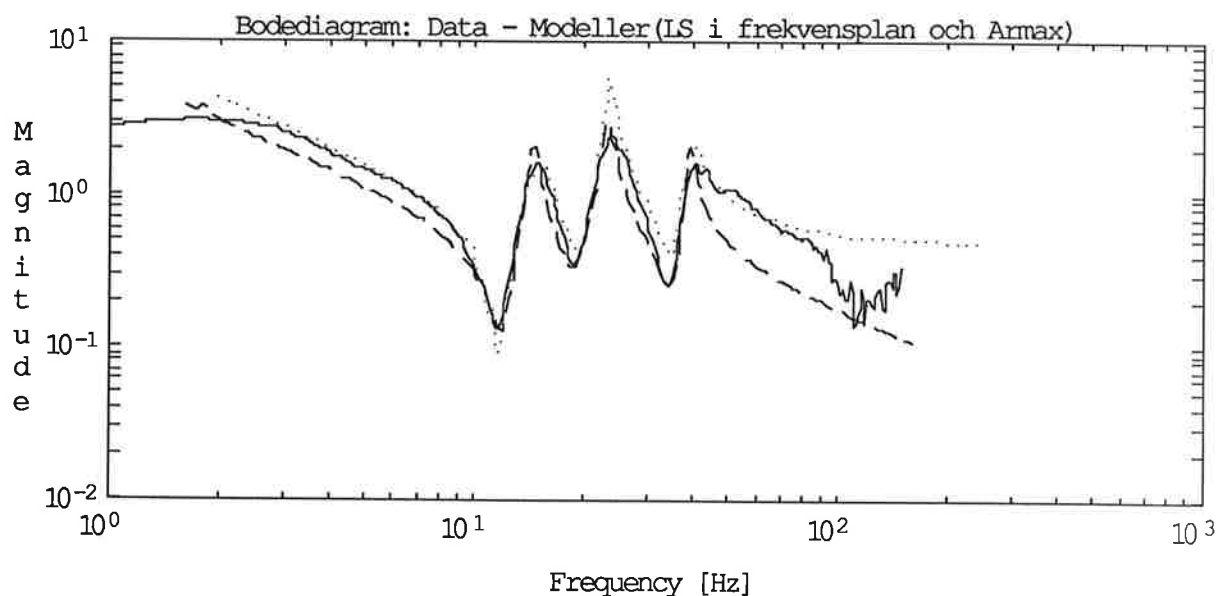
b) Gör en balanserad realisering. Reducera därefter modellen genom att stryka tillstånd med liten inverkan på insignal-utsignalsambandet.

2.

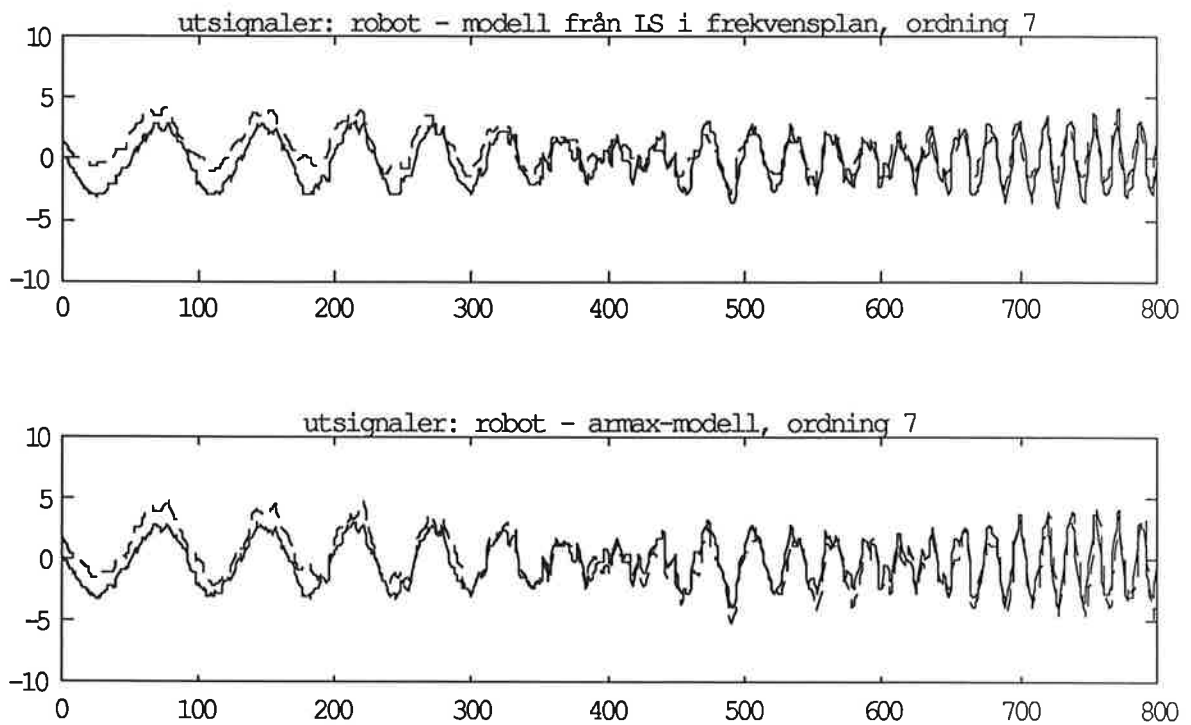
a) Välj intressanta punkter i ett bodediagram.

b) Anpassa i frekvensplanet med minstakvadratmetoden.

Båda metoderna behöver hjälp av identifieraren: I 1b) genom att han förstår hur många och vilka tillstånd som kan strukas och i 2a) genom att han väljer punkter av intresse.



figur 15. Heldragen linje är spektralanalyserad data, streckad är LS-anpassad i frekvensplan av ordning 7 och prickad Armax av ordning sju.



figur 16. Simulering av modellerna som erhållits dels från LS ifrekvensplan, dels från reducerad Armax.

7. Referenser och Hjälpmedel

Söderström T. (1984): Lecture notes in identification

Johansson R. (1990): Modelling and Identification, Kompendium

Glad T. och Ljung L. (1981): Reglerteknik, Grundläggande teori

Eriksson P. (1989): Kompendium i Digital Signalbehandling

Pro-Matlab (1990)

Ljung L: System Identification Toolbox

Gustavsson K., Lilja M. och Lundh M. (1990): A Collection of Matlab Routines for Control System Analysis and Synthesis

Nilsson P. (1990): A Bunch of Matlab Routines for Identification and Regulator Design

Nilsson K. (1990): MatComm, program för kommunikation mellan Matlab på Sun och VME

8. Appendix

Här beskrivs några Matlab-funktioner som jag gjort:

function **visval**(z, th, G);

visual validation plots: - spa-spectrum and theta-spectrum

- noise spectrum (in the same plot)
- $e(t)e(t+T)$ and $e(t)u(t+T)$ functions
- $e(t)$
- real and simulated output

$z = [y \ u]$

th = theta format computed by arx, armax etc

G = spectrum.

function Cuy = **coher2**(uy, M, G, T)

Computes the coherence function.

The coherence and the amplitude are shown.

uy = input output matrix

M = window size (see SPA)

G = transfer function generated by SPA

T = sampling interval

Cuy = [frequencies coherence]

function [a, b] = **fr2tf**(Giw)

frequencies to transfer function

This function makes a bodeplot and allows you to choose interesting frequencies. From these frequencies the

A and B polynoms are calculated. A new bodeplot, from these polynoms, is done in the old bodeplot.

Finally a pole-zero plot is done and the coefficients, poles and zeros written.

function [bm, am, a0] = **placepzo**(b, a)

placepzo : place poles, zeros and observer

function [r, s, t] = **rstpol**(b, a, bm, am, a0, h)

RST regulator computed with polynomial synthesis

The resulting polynoms are in discrete time if h is given otherwise in continuous time.

function [r, s, t] = **rstss**(b, a, bm, am, a0, h)

RST regulator computed from state space representation

The resulting polynoms are in discrete time if h is given otherwise in continuous time.

IDENTIFIERING AV FLEXIBELT SERVO
Projektarbete i Processidentifiering

Anders Ringdahl

Institutionen for Reglerteknik
Lunds Tekniska Högskola
december 1990.

Innehållsförteckning

1. Inledning	2
2. Mekanisk modellbygge	3
2.1 Linjär modell	3
2.2 Förenklad modell	4
2.3 Slutsatser	5
3. Experiment	7
3.1 Visuell identifiering	7
3.2 Inledande experiment	8
3.3 Experimentvillkor	9
4. Identifieringen	10
4.1 Icke parametrisk modell	10
4.2 Parametrisk modell	11
4.3 Valdiering	12
4.4 Tidskontinuerlig design	16
5. Reglering av processen	19
5.1 Regulatordesign	19
5.2 Simulering	20
5.3 Reglering	20
6. Referenser	23

1. Inledning

Som ett led i kursen processidentifiering vid Institutionen För Reglerteknik, Lunds Tekniska Högskola, har ett projektarbete genomförts under hösten 1990. Rapporten behandlar identidieringen av ett flexibelt servo.

Målet har varit att skapa en modell av överföringsfunktionen, så att hastighetsreglering kan göras. En härledning av det mekaniska systemet finns i kapitel 2, där problemen med olinjära friktioner också behandlas.

I kapitel 3 beskrivs experimentförfarandet och svårigheten att identifiera en huvudresonans (låg frekvens), då systemet påverkas av en högfrekvent periodisk störning.

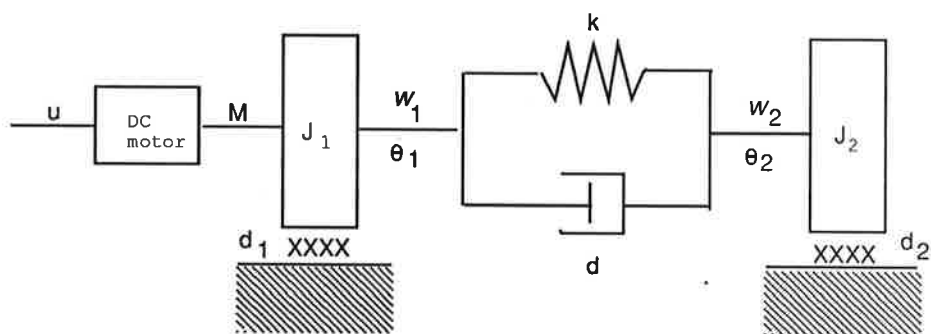
Identifiering och valdiering finns i kapitel 4, där tyngdpunkten ligger vid att skatta en parametrisk modell utifrån modellstrukturerna ARX, ARMAX och Box-Jenkins. Denna modell används sedan i kapitel 5 för att göra en enkel regulatordesign med hjälp av polplacering.

2. Mekanisk modellbygge

Genom att utnyttja de fysikaliska samband som råder för processen, kan man bygga en matematisk modell som är en stor tillgång till vid identifieringsarbete. De samband som redovisas i 2.1 och 2.2, bygger på stor del på resultat hämtade från A. Wallenborg [2]. Sambanden som tas upp i de första avsnitten, bygger på att processen kan antas vara linjär. I avsnitt 2.3 diskuteras lite om de olinjära effekterna och sätt att undvika dessa.

2.1 Linjär modell

Ett flexibelt servosystem kan modelleras med hjälp av två massor vilka är sammankopplade med en veka axel, se 2.1. J_1 och J_2 är massornas respektive tröghetsmoment. Den veka axeln beskrivs av en fjäderkonstant (k) och en dämpkoefficient (d). Förluster såsom friktioner i lager etc. , beskrivs av dämpkoefficienterna d_1 respektive d_2 .



Figur 2.1 Flexibelt servo

Modellen förutsätter att motorns dynamik kan försummas och att dess moment är proportionellt mot inspänning till motorn, enligt följande.

$$M_m = k_m I = k_m \frac{U}{R} = k_u U \quad (2.1)$$

Momentbalans ekvationen ger

$$J_1 \dot{\omega}_1 = k(\theta_2 - \theta_1) - d_1 \omega_1 - d(\omega_1 - \omega_2) + k_u u \quad (2.2)$$

$$J_2 \dot{\omega}_2 = k(\theta_2 - \theta_1) - d_2 \omega_2 + d(\omega_1 - \omega_2) \quad (2.3)$$

För att kunna skriva ekvationerna 2.2 och 2.3 på tillståndsform, definieras följande:

$$\begin{cases} \dot{\theta}_1 = \omega_1 \\ \dot{\theta}_2 = \omega_2 \end{cases} \quad (2.4)$$

Tillståndvariablerna och utsignalen sätts till

$$\begin{cases} x_1 = \omega_1 \\ x_2 = \omega_2 \\ x_3 = \theta_2 - \theta_1 \\ y = k_{\omega_1} \omega_1 \end{cases} \quad (2.5)$$

Processen kan alltså beskrivas av ett tredje ordningens system, med följande tillståndsekvationer

$$\dot{x}(t) = \begin{pmatrix} \frac{-(d_1+d_2)}{J_1} & \frac{d}{J_1} & \frac{k}{J_1} \\ \frac{d}{J_2} & \frac{-(d_1+d_2)}{J_2} & \frac{-k}{J_2} \\ -1 & 1 & 0 \end{pmatrix} x(t) + \begin{pmatrix} \frac{k_u}{J_1} \\ 0 \\ 0 \end{pmatrix} u(t) \quad (2.6)$$

$$y(t) = \begin{pmatrix} k_{\omega_1} & 0 & 0 \end{pmatrix} x(t) \quad (2.7)$$

Överföringsfunktionen

Med en tillståndsmodell som ges av ekvationerna 2.6 och 2.7 fås följande överföringsfunktion

$$G(s) = \frac{k_u k_{\omega_1} (J_2 s^2 + (d_2 + d)s + k)}{A(s)} \quad (2.8)$$

$$A(s) = J_1 J_2 s^3 + (J_1(d_2 + d))s^2 + (k(J_1 + J_2) + d_1 d_2 + d d_2)s + (d_1 + d_2)k \quad (2.9)$$

2.2 Förenklad modell

Poler

Om dämpkoefficienterna är små, kommer systemet att få två dåligt dämpade poler vilka är komplexkonjugerad. Deras naturliga frekvens ges av

$$\omega_0^2 \approx \frac{k(J_1 + J_2)}{J_1 J_2} \quad (2.10)$$

Den tredje polen dyker upp på den reelaxeln och ges av

$$s \approx -\frac{d_1 + d_2}{J_1 + J_2} \quad (2.11)$$

De tre polerna kommer alltså att ligga på, eller mycket nära den imaginära-axeln.

Nollställen

Systemet har också två nollställen, som i fallet med dålig dämpning, approximativt kan ses som komplexkonjugerade med den naturliga vinkelfrekvensen

$$\omega_n^2 \approx \frac{k}{J_2} \quad (2.12)$$

Förenklad överföringsfunktion

$$G(s) = \frac{K_u K_{\omega_1} (J_2 s^2 + k)}{s(J_1 J_2 s^2 + k(J_1 + J_2))}$$

Motorn

I den ovanstående modellen har det förutsatts att motorns dynamik kan försummas. Motorns dynamik kan beskrivas som

$$G_{mot}(s) \approx \frac{K_m}{1 + sT} \quad (2.13)$$

där T är motorns stigtid. Beroende på stigtiden kommer dynamik i motorn mer eller mindre påverka det totala systemet.

2.3 Slutsatser

Den framtagna modellen bygger på att systemet är linjärt, men i ett mekaniskt system som detta finns där friktion som inte är linjär. Friktionen kan delas in i tre delar.

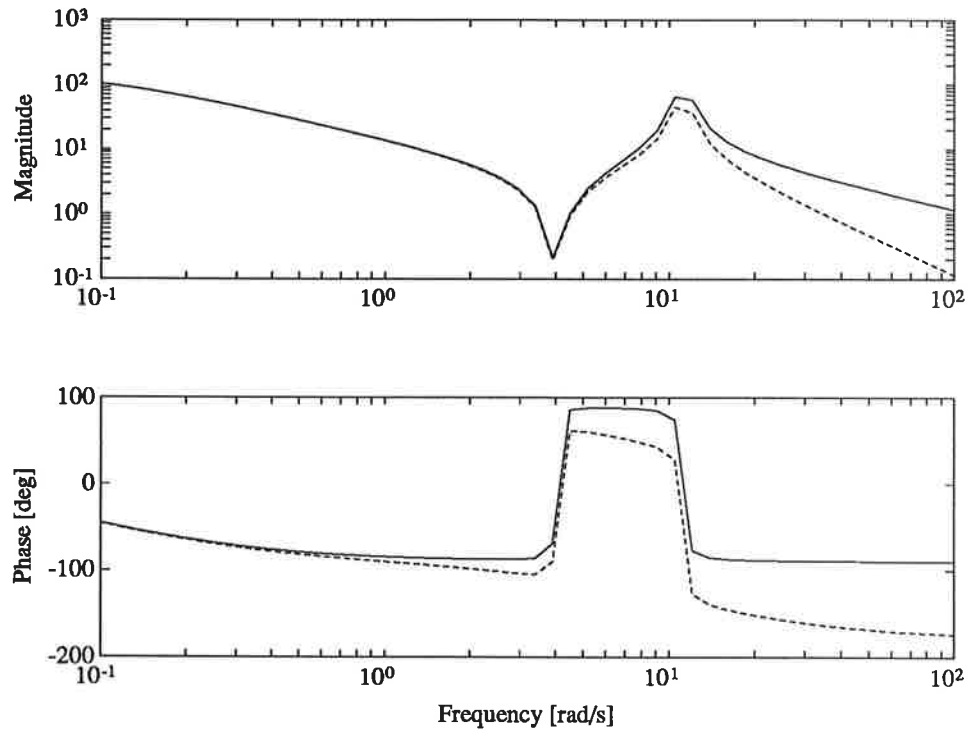
- Viskös friktion
- Statisk friktion
- Coulumb friktionen

Den viskosa friktionen är propotionel mot hastigheten och är medtagen i modellen (d_1 och d_2). Den Statiska friktionen påverkar systemet endast då det ej roterar, d. v. s. vid start och när man byter rotationsriktning. Columb friktionen är en konstan kraft som endast beror av rotationsriktningen.

För att slippa de två olinjära friktionerna vid identifieringsexperimentet, kan man sätta upp två enkla experimentvillkor:

- Byt ej rotationsriktning under data insamlingen.
- Hastigheten för ej vara så låg att statiska friktionen påverkar systemet.

Om dessa villkor uppfylls, bör systemet ha ordningen tre eller fyra beroende på motorn. I fig 2.2 visas bodediagrammet för ett flexibelt servo med data tagna från [2] . Både fallet med försumbar dynamik för motorn, och inverkan av denna presenteras.



Figur 2.2 Bodediagram för ett flexibelt servo. (—):Utan motordynamik. (····) : Med motordynamik, motorns stigtid $T = 0.1s$

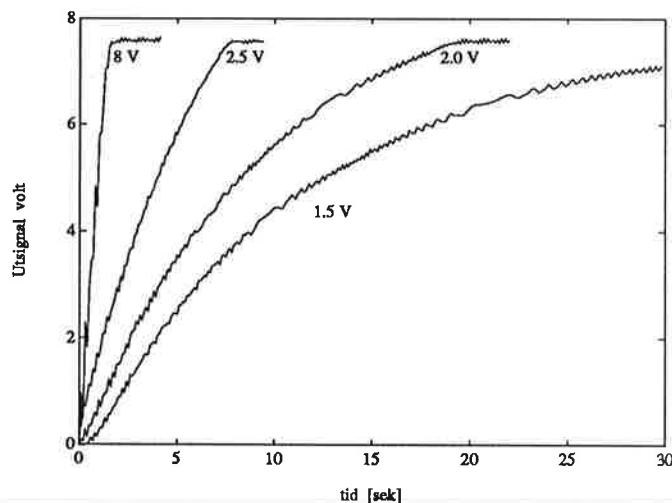
3. Experiment

Experimenten och datainsamlingen utfördes med hjälp av en PC och programmet LOGGER. Detta program skapar en PRBS-signal där möjligheten finns att sätta amplitud, likströmsnivå, signalens period och samplingsfrekvens. I kapitel 3.1 redovisas de resultat som framkom vid den visuella identifieringen. I 3.2 tas problem upp med störningar som verkar på systemet och i 3.3 sätts de slutgiltiga experimentvillkoren upp.

3.1 Visuell identifiering

I kap2 visades det att om systemet var dåligt dämpat skulle man få två poler som gav upphov till en egensvängning. Att systemet var dåligt dämpat var ingen svårighet att upptäcka, och med enkla experiment uppskattades egenfrekvensen att ligga i intervallet 2 till 3 Hz.

Styrsignalen som datorn ger ligger i intervallet $[-10\text{ V}, +10\text{ V}]$. Vid de inledande försöken upptäcktes det att motorn bottenade redan vid $\pm 0.02\text{ V}$. För att inte få problem med A/D-omvandlingen, fick en spänningsdelare som minskade styrspänningen med $1/11$ kopplas in. Detta medförde att styrsignalen kunde ligga i intervallet $[-1.4\text{ V}, +1.4\text{ V}]$, utan att utsignalen bottenade. Då processens stigtid är väldigt stor, se fig 3.1, kan man utnyttja insignaler som ligger utanför detta intervall. Det enda kravet är att signalen inte ligger på så länge, så att utsignalen bottenar. Vinsten med det är att medelvärdet på PRBS-signalen kan ligga högre, vilket medför att man kommer längre ifrån det område där den statistiska friktionen verkar. Detta samtidigt som amplituden på insignalen kan göras relativt stor, vilket medför att man får bättre excitation på systemet.



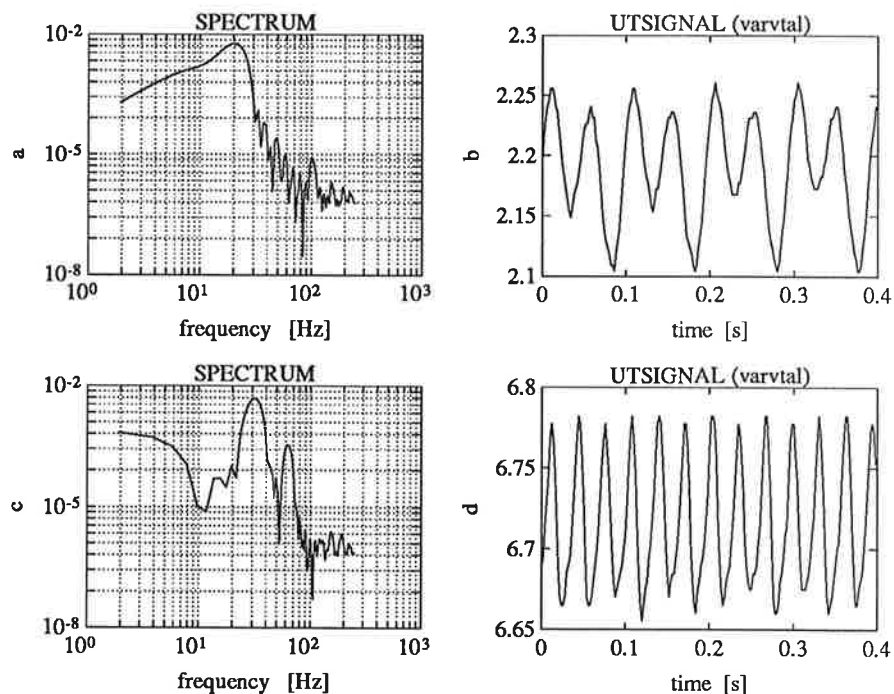
Figur 3.1 Stegsvår för olika styrsignaler. Max utsignal $\approx 7.7\text{V}$

3.2 Inledande experiment

Med resultat från den visuella identifieringen, kunde sedan experimentvillkor ställas upp. Systemet exiterades med en PRBS-signal med frekvensen 3 Hz och samplingsfrekvensen valdes till 50 Hz.

Den matematiskt framtagna modellen visar att systemet skulle vara av ordningen tre eller fyra, men vid försök att anpassa en ARMAX-modell med denna ordningen erhöles inga godtagbara resultat. Visserligen fanns det modeller av 4: ordningen som hade god överensstämmelse med det verkliga systemet, men med residuaer som visade att modellordningen var för låg. Vidare visade korskorrelationen att det fanns mer information att hämta från insignalen. Med ökad modellordning rättades residualerna och korskorrelationen mellan ut- och insignal till, men modellens och det verkliga systemets utsignal visade inte upp någon likhet.

Vid undersökning av systemets utsignal, visade det sig att denna hade en överlagrad störnsignal med en viss frekvens förutom vanligt brus. För att undersöka detta genomfördes experiment där insignalen hölls konstant, och spektralanalys gjordes på utsignalen se fig 3.2. Här ser man att hastighetsgivaren ger ifrån sig en brussignal där en frekvens dominerar, och att denna frekvens ökar med ökad styrsignal. Vid försöken ovan hade en styrsignal med medelvärdet 1.1 V och en amplitud på 0.5 V exiterat systemet. Detta medförde att den störning som givaren gav, låg approximativt i intervallet [10 Hz, 65 Hz]. Då f_s var 50 Hz medförde detta att vikning inträffade för frekvenser större än 25 Hz. För att rätta till problemet med den högfrekventa störningen måste samplingsfrekvensen ökas.



Figur 3.2 Spectrum för utsignal vid konstant inspänning. a,b) $u_{in} = 1.75V$ c,d) $u_{in} = 2.5V$

3.3 Experimentvillkor

För att tillgodose alla krav som framkommit vid tidigare försök, fick följande experimentvillkor ställas upp.

- Systemet får ej byta rotationsriktning.
- Systemet får aldrig vara i vila.
- Sampla så fort att problem ej uppkommer med vikning.

Detta löstes genom följande metod

$U_{in_{medel}}$	=	1.1	V
$U_{in_{amplitud}}$	=	0.6	V
f_{samp}	=	100	Hz
$PRBS_{period}$	=	17	
antal mätpunkter	=	6000	

Med den höga samplingsfrekvensen slipper man nu vikningseffekten, men en ny svårighet uppstår istället. Problemet blir nu att för lite kommer att hända mellan varje datapunkt för att identifieringen skall bli effektiv, signalen blir översamplad. Ända sättet att lösa detta är att sampla långsammare. För att kunna göra detta utan att på nytt få problem med vikningseffekten, måste man LP-filtrera signalen. Rent praktiskt löstes detta genom följande metodik.

- Datainsamling med $f_s = 100\text{ Hz}$
- LP-filtrera signalen i matlab (brytfrekvens 20 Hz)
- Tag varannan datapunkt av den filtrerade signalen $\Rightarrow f_s = 50\text{ Hz}$

4. Identifieringen

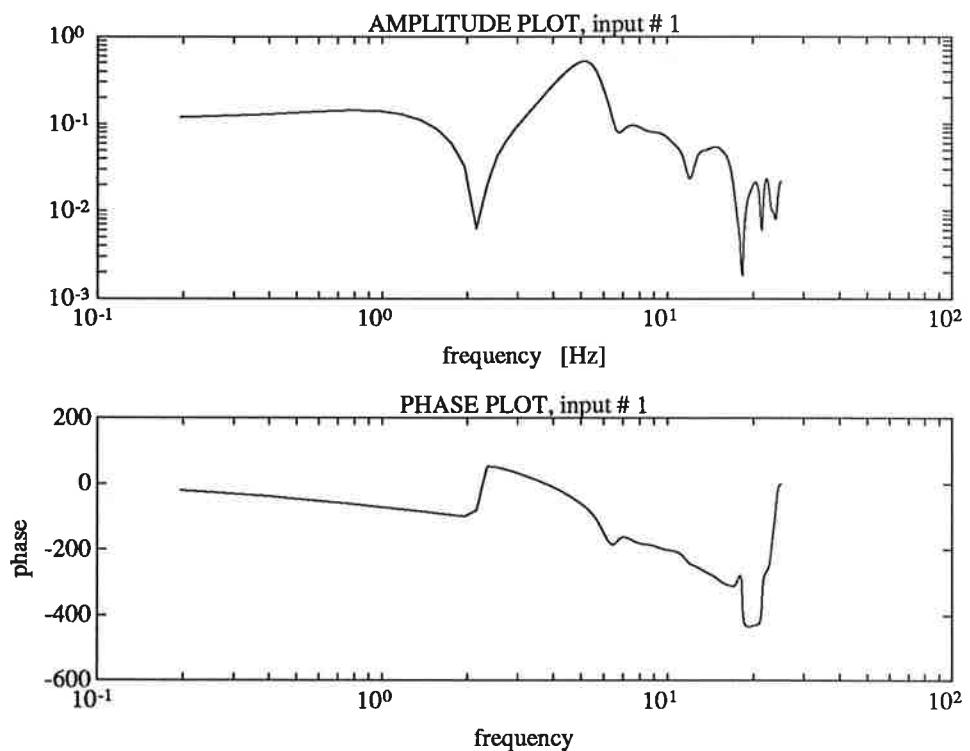
Allt identifieringsarbet har gjorts i MATLAB och hela identifieringen bygger på en signal som LP-filtrerats och fått minskad samplingsfrekvens, se kapitel 3. I avsnitt 4.1 tas en icke parametrisk skattning fram. I avsnitt 4.2 och 4.3 tas en parametrisk modell fram, vars syfte är att användas vid regulatordesignen.

4.1 Icke parametrisk modell

Via sambandet 4.1 kan man skatta den diskreta överföringsfunktionen.

$$\hat{H}(e^{i\omega h}) = \frac{\hat{S}_{yu}(i\omega)}{\hat{S}_{uu}(i\omega)} \quad (4.1)$$

Fig 4.1 visar skattningen av bodediagrammet. En direkt jämförelse kan ej göras med bodediagrammet som visas i fig 2.2, då det är i kontinuerlig tid. Men man kan se de karakteristiska dragen för ett system av tredje ordningen med ett komplexkonjugerat nollställe. Vi får en notch och en resonansfrekvens. Den enda avikelsen från det förväntade bodediagrammet är att amplituden har närmare nollutning istället för minus ett för låga frekvenser. En förklaring till detta kan vara att PRBS-signalen ger dålig excitation i detta frekvensområde. Resultatet här indikerar, trots avikelsen, att härledningen i kapitel 2 stämmer för systemet.



Figur 4.1 Det skattade bodediagrammet

Rent praktiskt medförde detta följande värden på PRBS-signalen.

4.2 Parametrisk modell

Allmänt kan en modell skrivas som

$$y(t) = G(q^{-1})u(t) + H(q^{-1})e(t) \quad (4.2)$$

där $e(t)$ är brus. För att skatta en modell användes följande modellstrukturer.

ARX

ARX modellen ges av följande samband

$$A(q)y(k) = B(q)u(k - nk) + e(k) \quad (4.3)$$

Denna modell är enkel att skatta, men ger ingen möjlighet att modulera bruset. På grund av den bruskaraktistik som processen har, borde inte denna modell ge något bättre resultat.

ARMAX

En ARMAX-modell ges av följande samband

$$A(q)y(k) = B(q)u(k - nk) + C(q)e(k) \quad (4.4)$$

I denna modell har man möjlighet att modulera bruset. Dock är man bunden av att A polynomet modulerar både insignal och bruset.

Box-Jenkins

Denna modell beskrivs av

$$y(k) = \frac{B(q)}{F(q)}u(k - nk) + \frac{C(q)}{D(q)}e(k) \quad (4.5)$$

Denna modell ger möjligheten att modulera insignalen och bruset med skilda polynom.

Sammanfattning

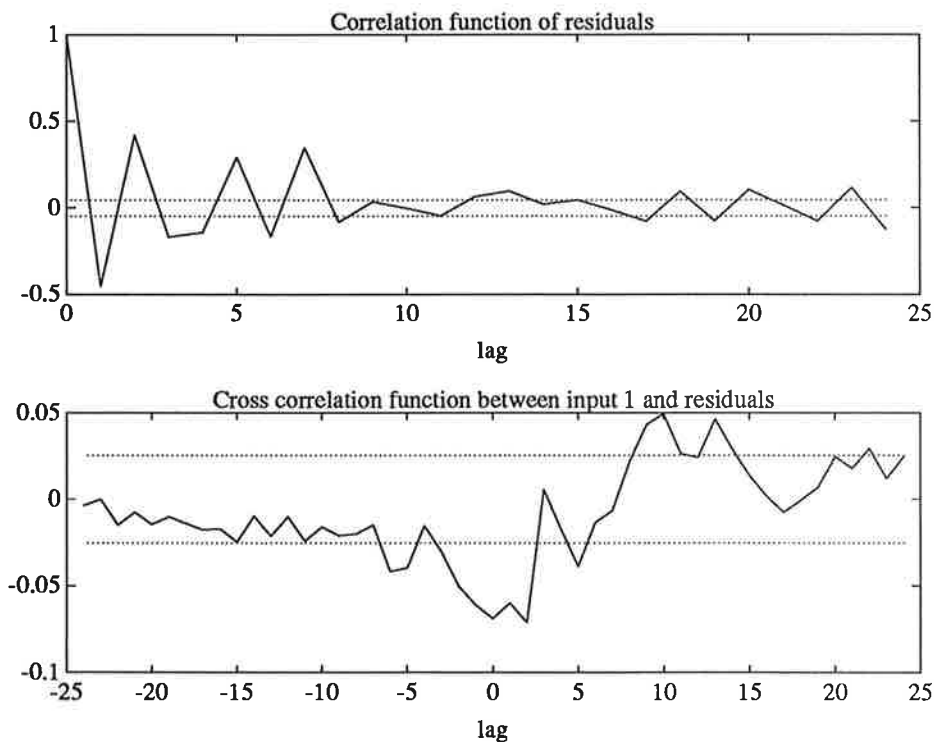
De modeller som borde vara av störst intress är ARMAX och BJ. A polynomet bör rimligtvis ha gradtal tre eller fyra och B polynomet två. Då bruset har en olinjär del, den hastighets beroende frekvensen från mätgivaren, så måste det vara av intresse att testa olika C och D polynom så att man modulerar bruset så bra som möjligt. Ett antal identifieringar gjordes för olika gradtal på polynomen, se tabell 4.1 .

Modell	deg A, deg F	deg B	deg C	deg D	nk
ARX	2 - 5	0 - 3			1 - 5
ARMAX	3 - 4	1 - 3	0 - 4		1 - 3
Box-Jenkins	2 - 4	0 - 2	0 - 4	2 - 4	2 - 3

Tabell 4.1 Tabel för olika modellordningar

4.3 Valdiering

Vid valdiering av modellordning finns det en rad metoder att tillgå. De flesta bygger dock på att bruset som påverkar systemet kan moduleras utifrån vitt brus. Bruset på detta system har tyvärr frekvens komponenter som är beroende av insignalen. Detta medför att antagandet om att vitt brus påverkar systemet, ej kan göras. Om man gör residualanalys kommer detta att få ett principutseende enligt fig 4.2 .



Figur 4.2 Korrelationen och korskorrelationen för residualen och insignal, på en ARMAX modell med dag A = 3, deg B = 2, deg C = 3 och nk = 2

Korrelationen för residualen visar att modellordningen är för låg, vilket beror på att modellen ej plockar ut alla de frekvenser som hastighetsgivaren ger upphov till. Korskorrelationen mellan residual och insignal visar att det finns mer information att hämta från insignalen. Detta stämmer också, då en komponent i bruset är beroende av insignalen. Detta är en olinjäritet som ej går att modulera med givna modellstrukturer.

De valdieringstest som kan vara rimliga att göra på detta systemet är.

- Överensstämmelse mellan modellens- och sytemets utsignal
- Rimliga pol och nollställen
- Jämförelse med mekaniskmodell
- Jämförelse mellan förlustfunktioner
- Korsvaldiering med mätdata som ej används vid identifiering
- Variansen på parameterskattningarna
- *F*-test (med resevation)

Inledande valdiering

För att sortera bland alla framtagna modeller användes en kombination av; överensstämmelse mellan modellens och systemets utsignal, rimliga pol och nollställen och jämförelse med mekanisk modell. Problemet med att jämföra modellens utsignal med systemets, är att metoden är väldigt subjektiv och risk finns att man överanpassar modellen till data. Förhoppningsvis elimineras lite av detta genom jämförelsen med den matematiska modellen. De modeller som valdes återfinns i tabell 4.2.

Modell	Struktur	deg A, deg F	deg B	deg C	deg D	nk
\mathcal{M}_1	ARMAX	3	1	3		2
\mathcal{M}_2	ARMAX	3	2	3		2
\mathcal{M}_3	Box-Jenkins	3	1	3	3	2

Tabell 4.2 Valda modeller efter inledande valdiering.

De ARX modeller som stämde överens med det verkliga systemet, hade så höga gradtal så de inte var intressanta. För att kunna välja mellan de övriga modellerna, måste metoder med mer objektivitet användas. Ett sätt är att använda metoder som utnyttjar förlustfunktionerna för de olika modellerna, se tabell 4.3.

Modell	Förlustfunktion	Antal parametrar
\mathcal{M}_1	0.001985	$n_1=8$
\mathcal{M}_2	0.001128	$n_2=9$
\mathcal{M}_3	0.000749	$n_3=11$

Tabell 4.3 Förlustfunktioner och antal parametrar för de olika modellerna

F-test

Med hjälp av detta test kan man avgöra vilken av de två ARMAX modellerna som skall väljas, dock kan testet ej användas för att testa \mathcal{M}_3 jämfört med de två ARMAX modellerna. Anta att man har ett system med n_0 parametrar

$$S: Ay = Bu + Ce \quad e \in (0, \sigma) \quad (4.6)$$

För de två modellerna \mathcal{M}_1 och \mathcal{M}_2 kan följande hypotes sättas upp

$$H_0: n_0 \leq n_1 \leq n_2 \quad (4.7)$$

I vårt fall kommer teststorheten τ ej att tillhöra $F(\infty, 1)$, vilket innebär att H_0 ej kan antas. \mathcal{M}_2 skall väljas framför \mathcal{M}_1 .

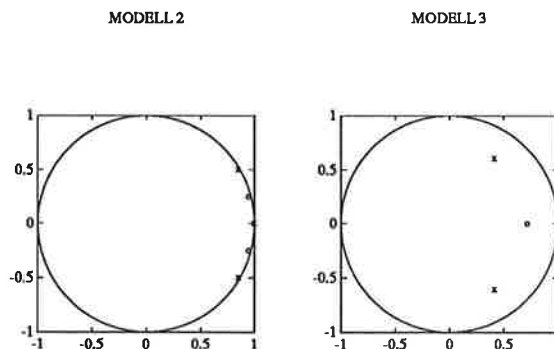
Modellerna \mathcal{M}_2 och \mathcal{M}_3

		\mathcal{M}_2			
$B(q^{-1})$	=		$0.065q^{-2}$	$- 0.122q^{-3}$	$+ 0.061q^{-4}$
$\text{var}(\hat{\theta}_{nB})$	=		0.0010	0.0018	0.0010
$A(q^{-1})$	=	1	$- 2.695q^{-1}$	$+ 2.664q^{-2}$	$- 0.969q^{-3}$
$\text{var}(\hat{\theta}_{nA})$	=	0	0.0034	0.0066	0.0034
$C(q^{-1})$	=	1	$- 1.935q^{-1}$	$+ 1.771q^{-2}$	$- 0.756q^{-3}$
$\text{var}(\hat{\theta}_{nC})$	=	0	0.0119	0.0183	0.0118
		\mathcal{M}_3			
$B(q^{-1})$	=		$0.040q^{-2}$	$- 0.029q^{-3}$	
$\text{var}(\hat{\theta}_{nB})$	=		0.0009	0.0011	
$F(q^{-1})$	=	1	$- 1.823q^{-1}$	$+ 1.361q^{-2}$	$- 0.536q^{-3}$
$\text{var}(\hat{\theta}_{nF})$	=	0	0.0078	0.0104	0.0100
$C(q^{-1})$	=	1	$- 0.591q^{-1}$	$+ 0.740q^{-2}$	$+ 0.093q^{-3}$
$\text{var}(\hat{\theta}_{nC})$	=	0	0.0292	0.0250	0.0294
$D(q^{-1})$	=	1	$- 1.032q^{-1}$	$- 0.200q^{-2}$	$+ 0.679q^{-3}$
$\text{var}(\hat{\theta}_{nD})$	=	0	0.0219	0.0371	0.0213

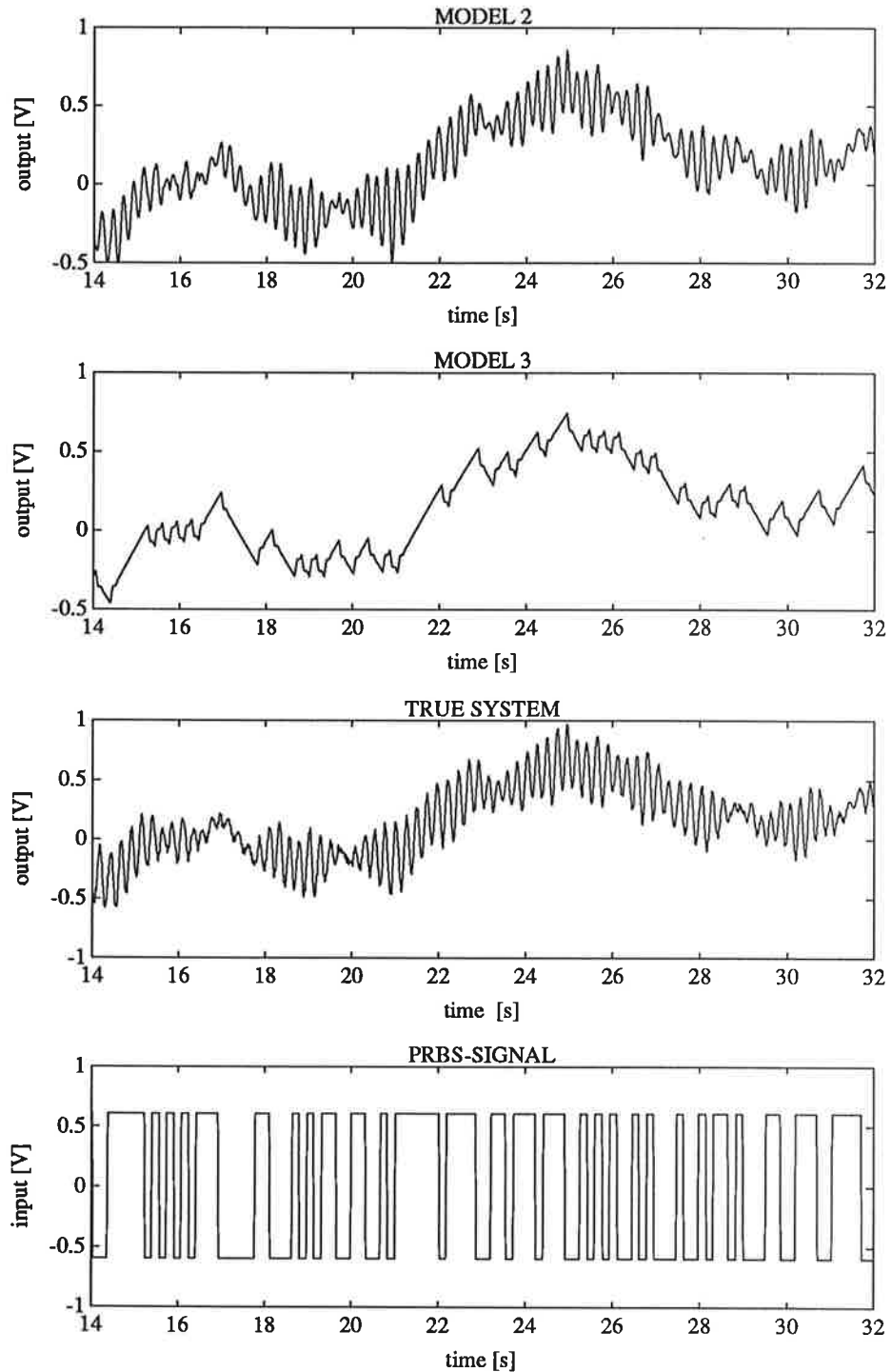
Tabell 4.4 Skattade parametervärden och dess skattade varianser för modell 1 och 2

Utsignalerna för modellerna och systemet finns presenterade i fig 4.4. Där ser man att \mathcal{M}_2 plockar upp de snabba svängningarna bättre än \mathcal{M}_3 , däremot följer \mathcal{M}_3 den långsamma svängningen lika bra. Förklaringen till detta ser man i fig 4.3, som visar att \mathcal{M}_2 's poler är mycket snabbare.

Vilken av de två modellerna skall man välja ? Med tanke på resultatet i avsnitt 4.1, borde det naturliga valet var \mathcal{M}_2 med sina två nollställen. Då man dessutom kan anta att dämpfaktorerna är små, skall pol och nollställen ligga mycket nära enhetscirkeln. Tittar man på varianserna för parameterskattningarna, upptäcker man att denna är nästan en faktor 2 större för \mathcal{M}_3 , ännu ett argument som talar för \mathcal{M}_2 . Trots detta kan \mathcal{M}_3 vara av intresse, då denna modell modulera bruset på ett annat sätt.



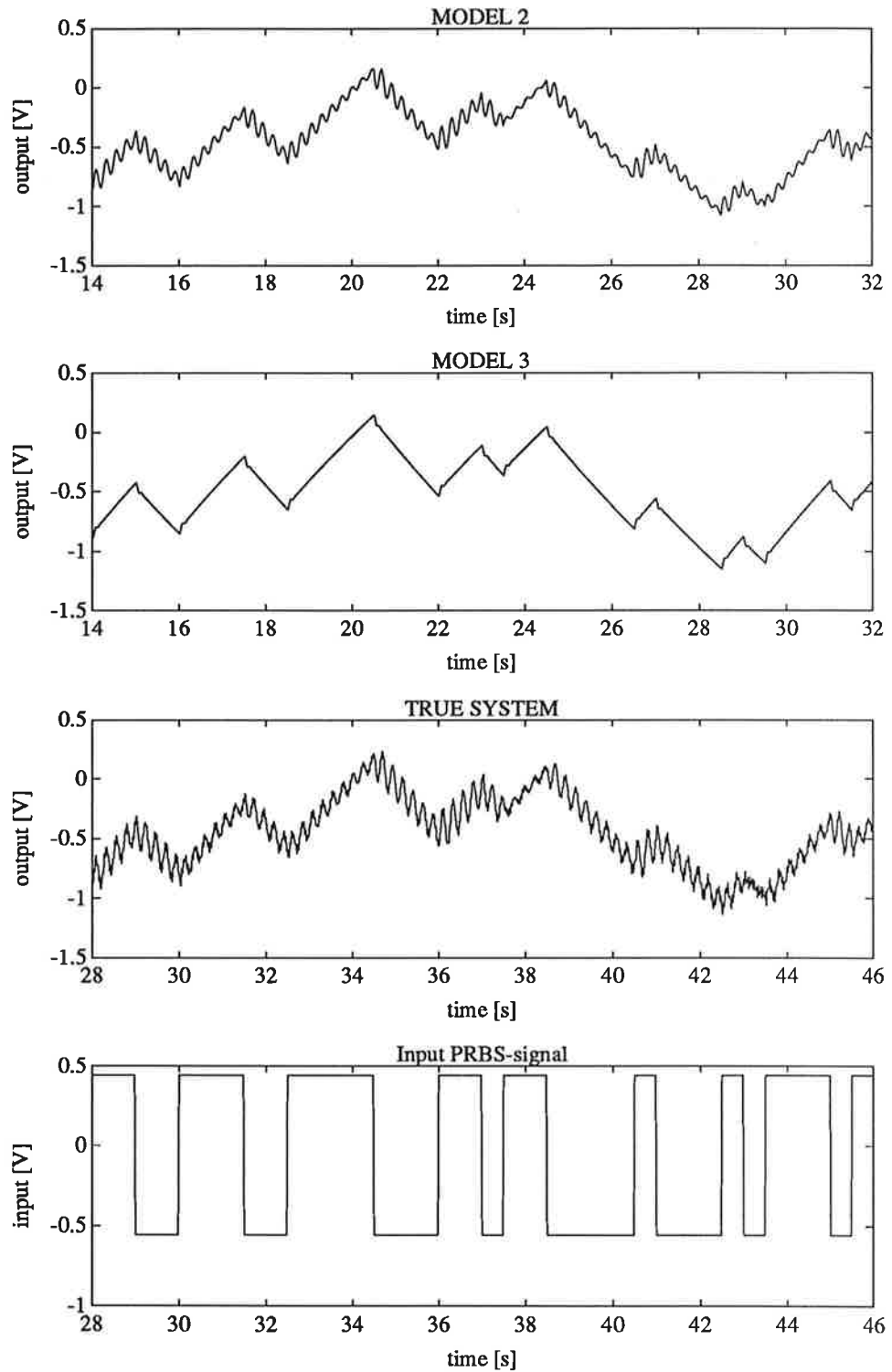
Figur 4.3 Pol och nollställe för modellerna



Figur 4.4 Utsignal för modellerna och systemet

Korsvaldiering med mätdata som ej används vid identifiering

Ett sista test för modellerna, som kan visa om någon av dem är för hårt anpassad till data. Processen har exiterats med exakt samma PRBS-signal, förutom det att perioden på signalen är satt till 50. I fig 4.5 kan man se att \mathcal{M}_2 fortfarande följer systemets utsignal. Antagandet man måste göra utifrån detta resultat är att Box-Jenkins modellen inte tillför något, utan man kan välja den enklare ARMAX modellen.



Figur 4.5 Utsignaler för modellerna och system vid korsvaldiering

4.4 Tidskontinuerlig design

Målet med identifieringen är att skapa en modell, så att reglerdesign på systemet kan göras. Då det är lättare att specificera egenskaperna i kontinuerlig tid, överförs den diskreta modellen till kontinuerlig tid i följande steg.

- Omvandla modellen till tillståndsrepresentation.
- Överför till kontinuerlig tid.
- Utför modellreduktion.
- Identifiera dämpkoefficienter och resonansfrekvenser.

Tillståndsrepresentation och överföring till kontinuerlig tid.

Reglerdesignen skulle kunna utföras i diskret tid men som tidigare påpekats är det lättare att utföra desing i kontinuerlig tid. Dessutom blir man ej låst vid den samplingstid som användes vid identifieringen. \mathcal{M}_2 får en tillståndsform enligt 4.8 .

$$x(n+1) = \begin{pmatrix} 2.6948 & -2.6645 & 0.9693 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} x(n) + \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (4.8)$$

$$y(n) = \begin{pmatrix} 0 & 0.0648 & -0.1217 & 0.0612 \end{pmatrix} x(n) \quad (4.8')$$

Fyra tillstånd behövs för att representera \mathcal{M}_2 , detta p. g. a. den extra tidsfördröjningen ($n_k = 2$). Med hjälp av *Matlab* utförs "invers sampling" och beräkning av nollställen och poler för det kontinuerliga systemet.

$$p_{1,2} = -0.7 \pm 26.7i$$

$$p_3 = -0.1$$

$$p_4 = -1853$$

$$n_{1,2} = -1.62 \pm 13.45i$$

$$n_3 = 48.8$$

Polerna får anses vara rimliga, där den snabba polen förmodligen beskriver motorns dynamik, svårare är det att förklara det icke stabila nollstället. Resultatet visar dock att modellreduktion kan göras.

Modellreduktion

När balanserad modellreduktion gjordes på modellen, blev resultatet det förväntade för polerna, p_4 ströks. Däremot ströks inga nollställen och in direktterm infördes. Anledningen till detta kan vara att processen innehåller en tidsfördröjning i storleksordningen 1 till 2 sampel, d.v.s $\tau \approx [20ms, 40ms]$. Slutsaten man kan dra är att man ändå kan stryka det icke stabila nollstället, och ej ta med någon direktterm i modellen.

Tidskontinuerlig överföringsfunktion

Med den reducerade modellen fås följande överföringsfunktion.

$$\hat{G}(s) = \frac{s^2 + 3.2568s + 184.7}{s^3 + 1.5575s^2 + 713.5s + 75.72} \quad (4.9)$$

Obervera att tidsfördröjningen ej tagits med.

Processen kan skrivas som

$$G(s) = \frac{s^2 + \omega_n \zeta_n s + \omega_n^2}{(s + p_1)(s^2 + \omega_o \zeta_p s + \omega_o^2)} \quad (4.10)$$

Med följande parametervärden

$$\omega_o = 26.7081 \text{ rad/s}$$

$$\zeta_p = 0.0272$$

$$p = 0.1061$$

$$\omega_n = 13.5893 \text{ rad/s}$$

$$\zeta_n = 0.1198$$

Dessa värden får antas vara rimliga, och de konfirmerar antagandet om små dämpstermer, som gjorts i tidigare avsnitt.

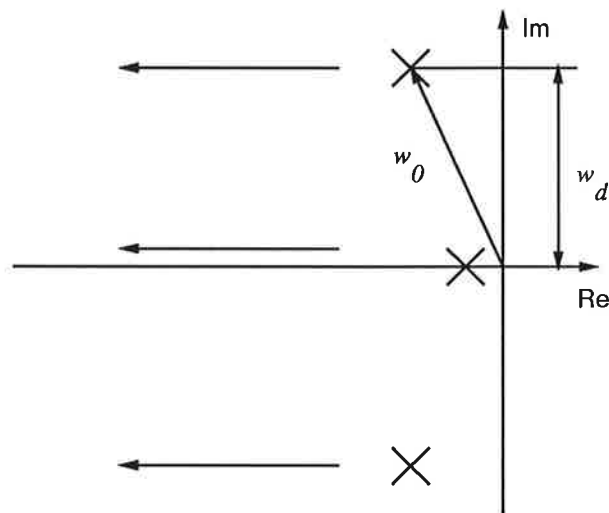
5. Reglering av processen

5.1 Regulatordesign

Regulatorndesignen baseras på polplacering och följer den procedur som redovisas i [3], kap 10. För att genomföra designen måste man bestämma det slutna systemets poler och observeraren.

Slutna systemet och observerarens karakteristiska polynom

För att bestämma dessa bör man studera processens poler.



Figur 5.1 Singularitetsdiagram för processens karakteristiska polynom

De två komplexkonjugerade polerna förklarar fjäderns beteende. Polen på reella axeln visar hur processens stigtid beror på tröghetsmoment och den linjära delen av friktionen. Att försöka ändra på den frekvens fjäder svänger med är ingen ide', det man kan göra är att öka dämpfaktorn. Fjäderns svängningsfrekvens bestäms av sambandet 5.1, dvs. imaginärdelen av polen.

$$\omega_d = \omega_0 \sqrt{1 - \zeta_p^2} \quad (5.1)$$

Genom att placera de två komplexkonjugerade polerna i A_m längs pil 1, behåller man svängningsfrekvensen men ökar dämpfaktorn. På detta sätt hoppas jag att ta bort egensvängningen som fjädern ger upphov till. Genom att placera polen på den reella axeln längre åt vänster, snabbar man upp systemet.

För observeraren, A_o gäller samma regler, det är polen på den reella axeln som det skall manipuleras med. Skillnaden mellan observerarpolynomet och slutna-lopens karakteristiska polynom A_m , är att det förstnämnda bör vara snabbare.

RST - Regulatorn

Fyra olika regulatorer designades. Skillnaden var att läget på den reela polen varierades i både A_m och A_o , medan ω_0 och ζ hölls konstanta.

$$\omega_0 = 37.38 \text{ rad/s}$$

$$\zeta = 0.7$$

Regulator	A_m		A_o	
Regulator 1	2.122	20Proc _{pol}	4.244	40Proc _{pol}
Regulator 2	4.244	40Proc _{pol}	4.244	40Proc _{pol}
Regulator 3	2.244	40Proc _{pol}	6.366	60Proc _{pol}
Regulator 4	6.366	60Proc _{pol}	6.366	60Proc _{pol}

Då designen av regulatorn gjordes i diskret tid, fick de framtagna A_m och A_o samplas. Inga nollställen förkortades, dvs. att det slutna systemets nollställe var samma som för processen. Vidare designades regulatorn så att den innehöll en integrator. Följande gällde för alla regulatorer

Deg A_m	=	3	Deg R	=	3
Deg A_o	=	3	deg S	=	3
Deg B^-	=	2	deg T	=	3
Deg B^+	=	0	sampleintervall	=	30 ms

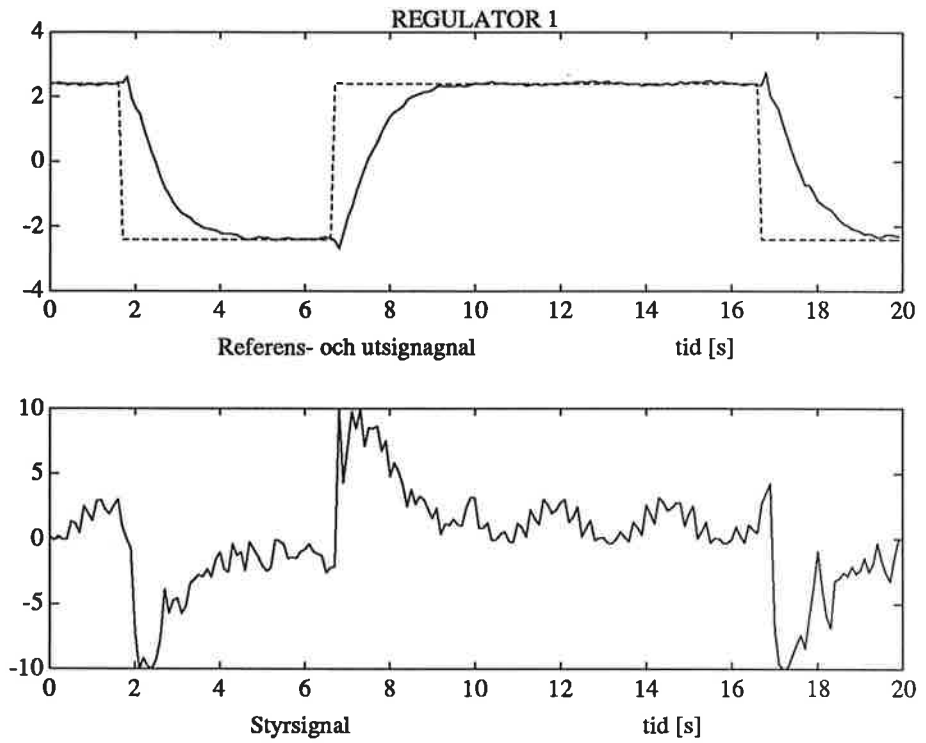
5.2 Simulering

De framtagna modellerna simulerades och förväntat resultat erhöles. Ju snabbare polen för A_m och A_o gjordes, desto snabbare blev stegsvaret. Dock fanns där en övre gräns hur snabba poler man kunde ha, innan systemet blev instabilt. Det skall dock påpekas att inga störningar påverkade systemet under simuleringen.

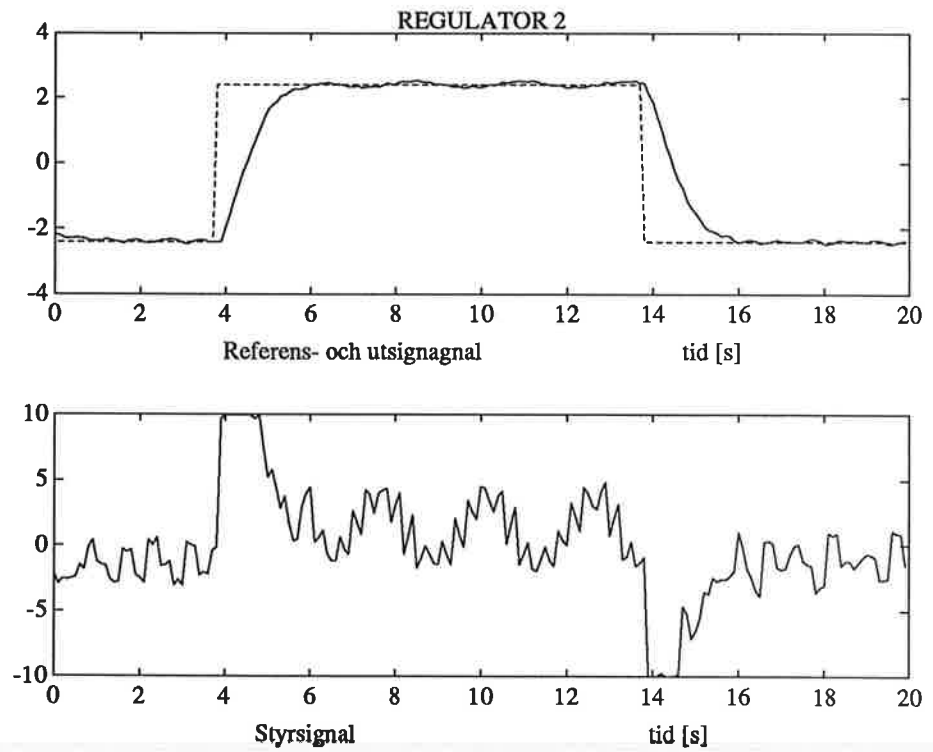
5.3 Reglering

För att reglera processen användes programmet TOOLBOX och för mätvärdesinsamling LOGGER. Anledning till att samplefrekvensen valts så låg, var problem att få TOOLBOX att fungera med snabbare. Detta gör att resultaten påverkas av de vinkningseffekterna, som de högfrekventa störningar ger upphov till. Resultatet för de olika regulatorerna visas i fig5.2, fig5.3, fig5.4 och fig5.5 .

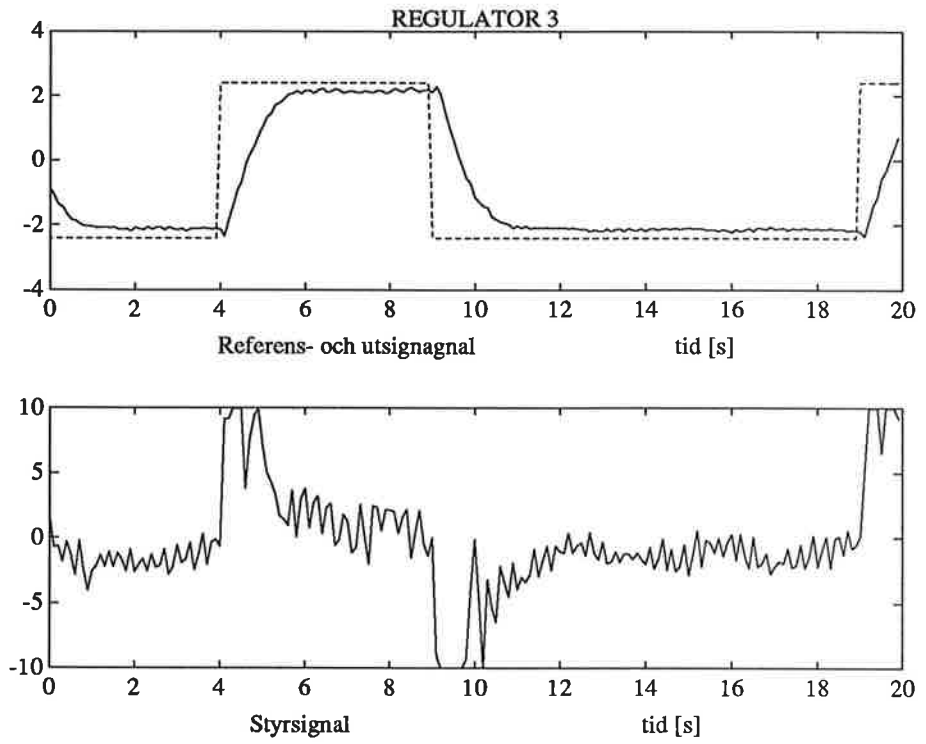
Ingen större skillnad i stigtid mellan de olika regulatorerna kan upptäckas. Däremot så skiljer sig de olika regulatorerna åt, med avseende på hur väl utsignalen följer referensvärdet och styrsignalens uppförande. Den regulator som ger bäst resultat är nummer ett. Och slutsaten man får dra, är att jag nog i designen försökt ta ut mer av processen än vad denna klarar av. En långsammare design skulle förmodligen ge ett bättre utseende på styrsignalen. Dock skall det tilläggas att fjäderns egensvängning har till stor del dämpats ner.



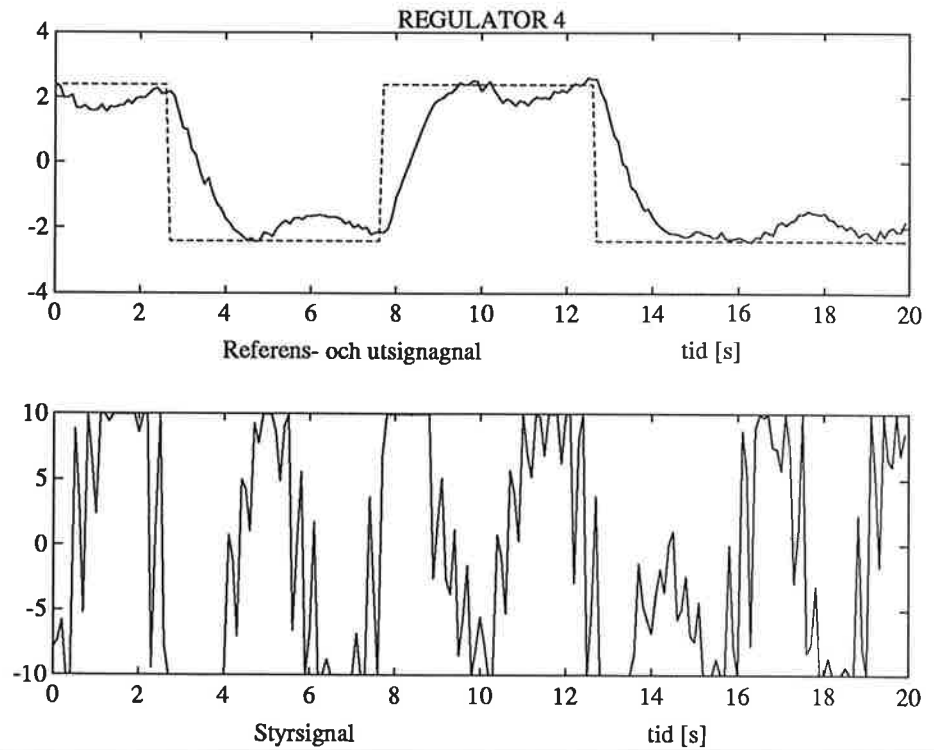
Figur 5.2 Reglering av processen med regulator 1



Figur 5.3 Reglering av processen med regulator 2



Figur 5.4 Reglering av processen med regulator 3



Figur 5.5 Reglering av processen med regulator 4

6. Referenser

JOHANSSON, R (1990): "Kompendium: Modeling and Identification," Institutionen för Reglerteknik, Lunds Tekniska Högskola, Lund, Sweden.

WALLENBERG, A (1987): "Control of Flexible Servo Systems," CODEN: LUTFD/TFRT-3188, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

ÅSTRÖM, K. J. AND B. WITTENMARK (1990): *Computer Controlled Systems*, Prentice Hall.

SÖDERSTRÖM, T. AND P. STOICA (1989): *Systemidentification*, Prentice Hall.

Modeling and control of the ball and beam process

Ulf Jönsson
Henrik Olsson

Department of Automatic Control
Lund Institute of Technology
December 1990

Table of Contents

1. Introduction	2
2. Process description	2
3. Identification	5
3.1 Identification of the beam process	5
3.2 Identification of the ball process.	11
4. Controller design	18
4.1 Design of controller for the beam process	19
4.2 Design of controller for the total system	21
5. Conclusions	28
References	29

1. Introduction

This project was done as part of a course in system identification given at Lund Institute of Technology. The aim of the project was to apply theoretical knowledge, gained at lectures, to a lab process. The process in this case was the so called ball and beam process. To justify the identified models of the process, these were used for design of a controller for the position of the ball on the beam. This controller was then implemented and tried out on the real process. The outline of the report is as follows. Chapter 2 describes the ball and beam process and derives differential equations for it. Chapter 3 deals with the identification of the process. Design of controllers for the process is treated in Chapter 4, and finally conclusions are drawn and suggestions for future improvements are made in Chapter 5.

2. Process description

The ball and beam process, shown in figure 2.1, consists of a beam, whose angle is controlled by a DC-motor, and a ball that can roll along a track on the beam. The input to the process is the voltage to the DC-motor and the outputs are the beam angle, ϕ , and the ball position, x .

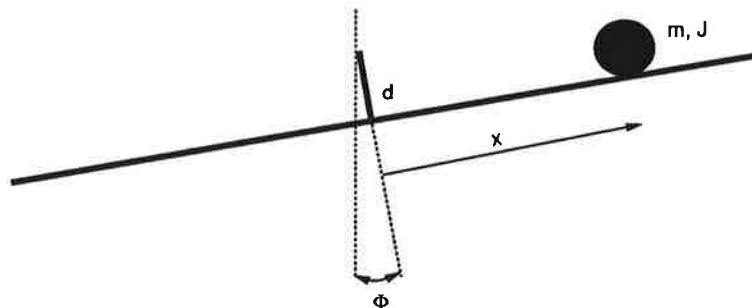


Figure 2.1 The ball and beam process.

When analyzing the ball and beam process it is convenient to regard the beam process and the ball process separately. This separation can only be done if it is assumed that the ball has no effect on the beam movement, i.e. if the mass of the ball is small. The beam process, with transfer function G_ϕ , describes the relation between the input voltage u to the DC-motor and the beam angle ϕ , and the ball process, with transfer function G_x , describes the relation between the beam angle ϕ and the ball position x . A block diagram of the process is shown in figure 2.2.

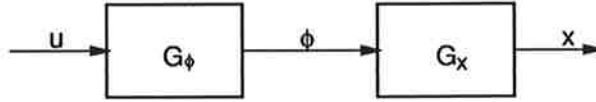


Figure 2.2 Block diagram of the ball and beam process

The beam process

The beam is connected to the DC-motor's shaft. Therefore, from electromechanics, the transfer function from input voltage to beam angle, G_ϕ , can be expected to be

$$G_\phi(s) = \frac{b}{s(s + \tau)} \quad (2.1)$$

There may also be resonances present. These can either originate from the connection between the motor and the beam or from the beam itself. The transfer function G_ϕ with n resonances present is

$$G_\phi(s) = \frac{b}{s(s + \tau) \prod_{i=1}^n (s^2 + 2\zeta_i\omega_i + \omega_i^2)} \quad (2.2)$$

The ball process

The derivation of a differential equation for the ball process is a straightforward application of balance relations from mechanics (see also [ola]). The coordinates defined in figure 2.3 will be used.

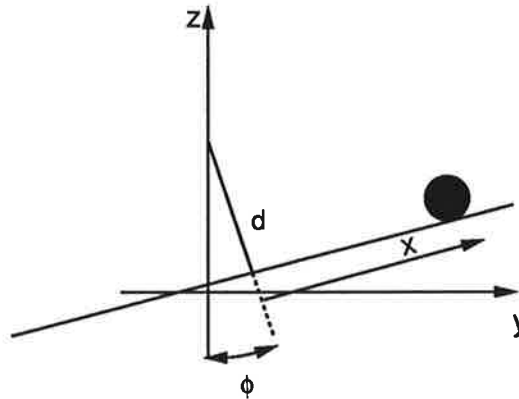


Figure 2.3 Definition of coordinates

Figure 2.4 shows the forces acting on the ball. The definition of the coordinates gives

$$\begin{aligned} y &= d \sin(\phi) + x \cos(\phi) \\ z &= d - d \cos(\phi) + x \sin(\phi) \end{aligned} \quad (2.3)$$

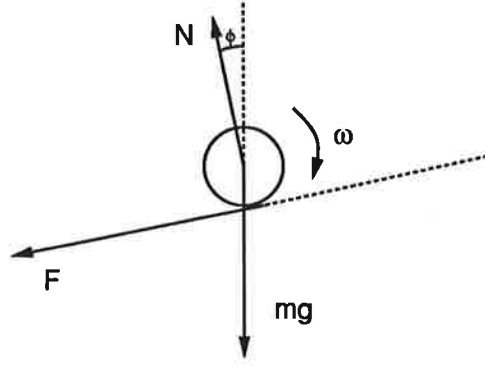


Figure 2.4 Forces acting on the ball

Differentiating the equations above twice gives

$$\begin{aligned}\ddot{y} &= d\ddot{\phi} \cos(\phi) - d\dot{\phi}^2 \cos(\phi) + \ddot{x} \cos(\phi) - 2\dot{x}\dot{\phi} \sin(\phi) - x\ddot{\phi} \sin(\phi) - x\dot{\phi}^2 \cos(\phi) \\ \ddot{z} &= d\ddot{\phi} \sin(\phi) + d\dot{\phi}^2 \cos(\phi) + \ddot{x} \sin(\phi) + 2\dot{x}\dot{\phi} \cos(\phi) + x\ddot{\phi} \cos(\phi) - x\dot{\phi}^2 \sin(\phi)\end{aligned}\quad (2.4)$$

The force balance for the ball is given by (see Figure 2.4)

$$\begin{aligned}m\ddot{y} &= -N \sin(\phi) - F \cos(\phi) \\ m\ddot{z} &= N \cos(\phi) - F \sin(\phi) - mg\end{aligned}\quad (2.5)$$

The force N is eliminated by multiplying the upper equations in (2.4) and (2.5) with $\cos(\phi)$, and by multiplying the lower equations with $\sin(\phi)$, this results in

$$m(\ddot{y} \cos(\phi) + \ddot{z} \sin(\phi)) = -F - mg \sin(\phi) = m(d\ddot{\phi} + \ddot{x} - x\dot{\phi}^2) \quad (2.6)$$

If it is assumed that the friction between the ball and the beam is such that the ball is rolling along the beam, then the force F is given by

$$Fr = J\dot{\omega} = J\frac{\ddot{x}}{r} = \alpha mr^2 \frac{\ddot{x}}{r} \Rightarrow F = \alpha m\ddot{x} \quad (2.7)$$

From (2.6) and (2.7), the following differential equation is obtained, which describes the dynamics from the angle ϕ of the beam to the the position x of the ball on the beam

$$\begin{cases} (1 + \alpha)\ddot{x} - x\dot{\phi}^2 + d\ddot{\phi} = -g \sin(\phi) \\ \alpha = \frac{J}{mr^2} \end{cases} \quad (2.8)$$

This equation is not linear, and therefore it is not possible to describe its dynamics with models such as ARMAX or Box-Jenkins. Since linear models are convenient both for identification and control, it was decided only to work with such in this project. A linear approximation of equation (2.8) is therefore desired. If (2.8) is linearized around $\phi = 0$ the right hand side becomes $-g\phi$. The nonlinear term on the left hand side is harder to linearize. However, consider that the process must be controlled by a controller which makes the

beam movements slow. The term $x\dot{\phi}^2$, which is proportional to the centrifugal force, will therefore be negligible, and an approximate and linear model is thus given by

$$(1 + \alpha)\ddot{x} + d\ddot{\phi} = g\phi \quad (2.9)$$

transfer function for the ball process G_x is obtained by taking the Laplace transform of (2.9).

$$G_x(s) = -\frac{g + ds^2}{(1 + \alpha)s^2} \quad (2.10)$$

Since d is small (0.02m) and since the process dynamics are slow, the term ds^2 will be small compared to $g = 9.81\text{m}^2/\text{s}$. The following transfer function is therefore a reasonable linear approximation of the dynamics in (2.9)

$$G_x(s) = -\frac{g}{(1 + \alpha)s^2} \quad (2.11)$$

The approximations done to obtain this linear model should have been analyzed theoretically in order to justify them. This was however not done in this project.

3. Identification

From frequency analysis done earlier in a laboratory exercise, it was known that the transfer function for the beam process G_ϕ , i.e. from u to ϕ , consists of approximately an integrator and a resonance. The resonance causes the beam to vibrate and the ball to bump occasionally and thus gives an unpleasant performance of the total system. To design a good controller for the total system it is therefore necessary to identify this resonance properly. This is very hard if the identification is done by measuring corresponding values of u and x only, since we have approximately a double integrator from ϕ to x (see equation 2.11). This means that the ball position will hardly be affected by the high frequency resonance in G_ϕ . The low-pass character of the ball process makes an accurate identification of the resonance impossible. We therefore decided to identify the two processes G_ϕ and G_x separately, G_ϕ using a high excitation in the frequency range where the resonance was expected, and G_x using excitation at lower frequencies. The two identifications were also done at different sampling rates because of the above mentioned difference in interesting frequencies. The identified G_ϕ was then used for design of a controller for the beam process and the identified G_x for design of a controller for the ball process in series with the controlled beam process.

3.1 Identification of the beam process

This identification was done without having the ball on the beam. We assumed that this would not affect the process significantly.

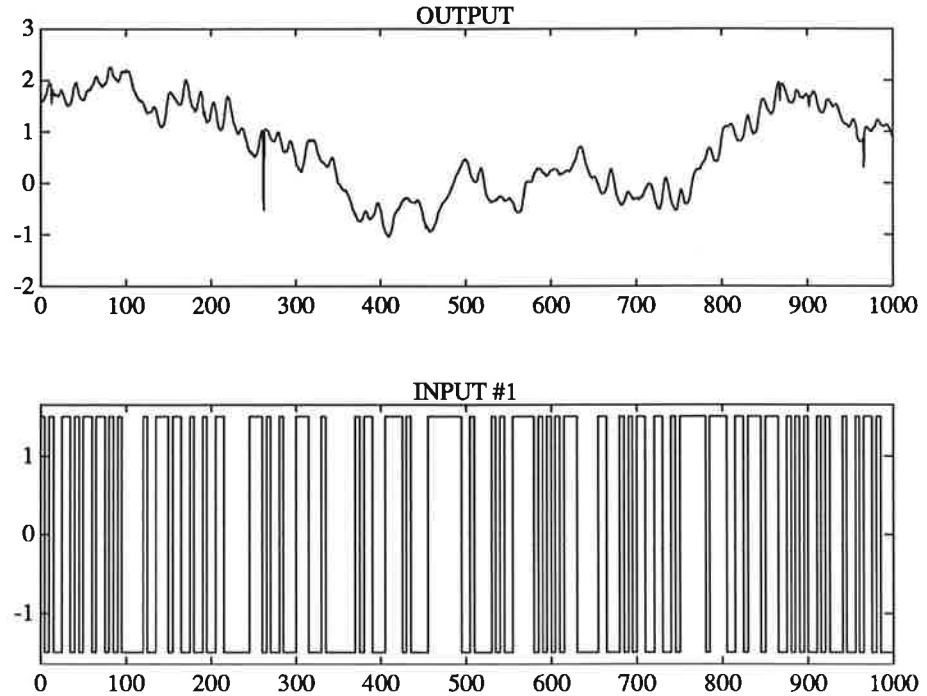


Figure 3.1 Recorded process data. Output ϕ (upper curve) and input u (lower curve)

Data generation

The program `logger` was used to collect data for the identification and to generate an input signal, u , to the process. As input signal a PRBS signal was chosen with *period* = 5, *amplitude* = 1.5V and *tsamp* = 5ms. The period was chosen to excite the expected resonance at 12Hz. 10 000 data points were collected. 2000 were used for the identification and 2000 for cross validation. The first 1000 recorded input and output data can be seen in figure 3.1. caused by the A/D-conversion. These were removed by setting the value of ϕ at these points to the mean value of the two surrounding points. The mean value of the data was then removed using the matlab command `detrend`. The first 1000 detrended data points with outliers removed can be seen in figure 3.2.

Choice of model

Different model structures for the identification of the beam processes can be considered. The two types mainly used in this identification were ARMAX and Box-Jenkins. These were tried with different orders of the polynomials in the model structures. The two types gave models that when validating turned out to be almost equally good. The best ARMAX model of the form

$$A(q)y(t) = q^{-k}B(q)u(t) + C(q)e(t)$$

had $\deg(A) = 3$, $\deg(B) = 0$, $\deg(C) = 2$ and $k = 4$. However a Box-Jenkins model with the structure

$$y(t) = \frac{q^{-k}B(q)}{F(q)}u(t) + \frac{C(q)}{D(q)}e(t)$$

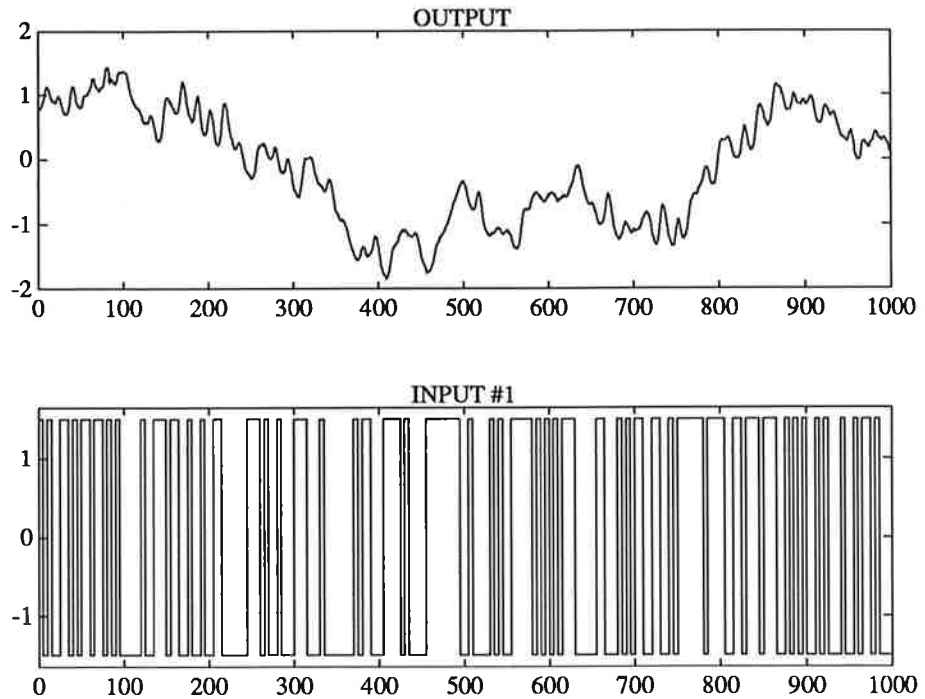


Figure 3.2 Data used for identification. Output ϕ (upper curve) and input u (lower curve)

and with $\deg(B) = 0$, $\deg(C) = 2$, $\deg(D) = 3$, $\deg(F) = 3$ and $k = 4$ gave slightly better result when cross validating with unused data. This model was obtained by the command

```
>> thbj12334=bj(iddata,[1 2 3 3 4]);
```

The coefficients of the model was presented by

```
>> present(thbj12334)
```

with the following result.

```
This matrix was created by the command BJ on 11/10 1990 at 15:53
Loss fcn: 0.0005373 Akaike's FPE: 0.0005422 Sampling interval 1
The polynomial coefficients and their standard deviations are
B
```

```
=
0 0 0 0 0.0023
0 0 0 0 0.0001
F =
1.0000 -2.7679
2.6558 -0.8877
0 0.0043 0.0084 0.0042
C =
1.0000 1.1163
0.1559
0 0.0518 0.0508
```

D =
 1.0000 -0.2847 -0.9927
 0.2835
 0 0.0508 0.0027 0.0505

The transfer function of the model can be seen in figure 3.3. As expected the model consists of an integrator plus a resonance.

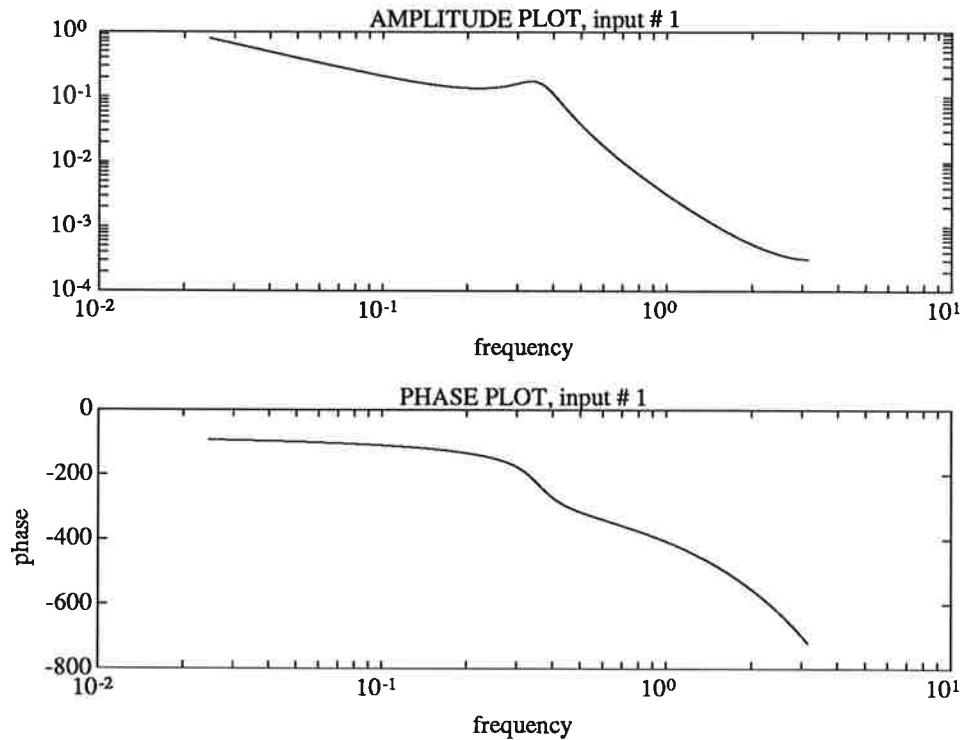


Figure 3.3 Computed transfer function of the estimated model.

the empirical transfer function estimate, ETFE, obtained by dividing the FFT of the output with that of the input of the data used for identification. This estimate is shown in figure 3.4. As can be seen the empirical transfer function does only correspond for low frequencies with the transfer function of the identified model. This is probably due to noise in the higher frequency range. A slope of -1 and a -90° phase can be seen for low frequencies. The transfer function obtained by spectral analysis shows the same poor result for higher frequencies. This was however not considered serious since the accuracy of the model in higher frequencies could be examined by cross validation. The poles of the estimated Box-Jenkins model are shown in figure 3.5. an integrator as expected. A time constant as in (2.1) could not be seen even if the model order was increased.

Validation

The identified model can be validated in different ways. Cross validation with unused data was considered the most important for our purposes and therefore the result of this determined our choice of model for subsequent controller design. 2000 unused data points were used to cross validate the model. This gave a satisfying result as can be seen in figure 3.6. Most of the dynamics in the recorded data can be reproduced and the deviation between the recorded and

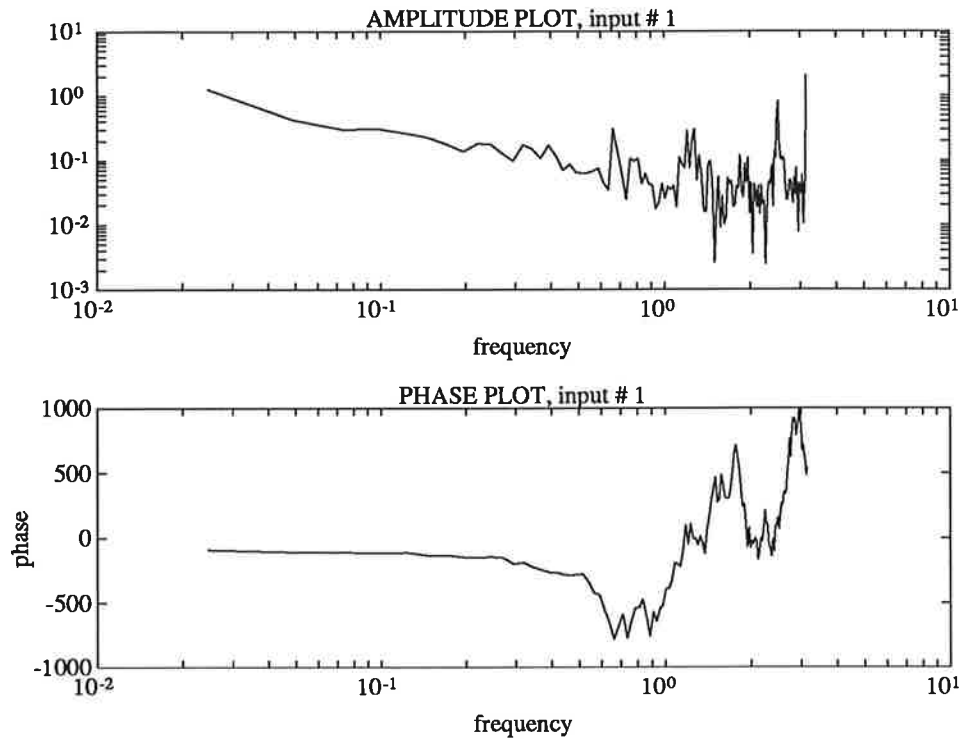


Figure 3.4 The empirical transfer function estimate

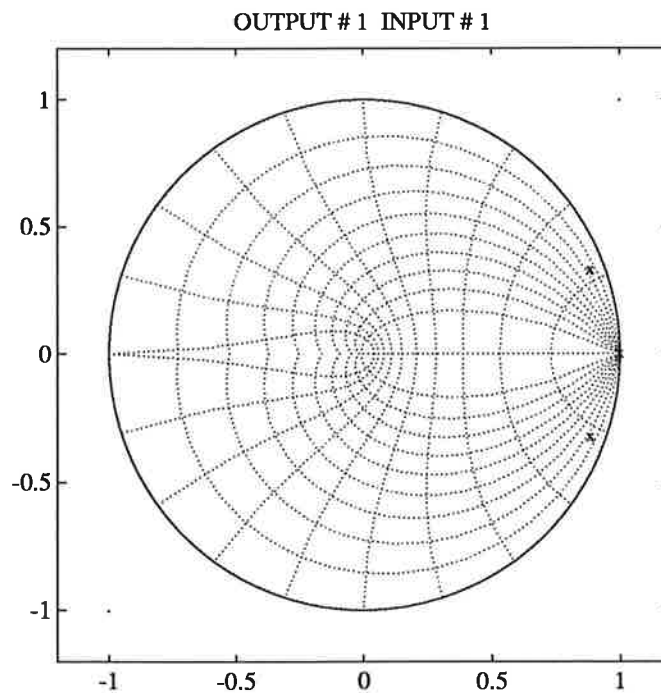


Figure 3.5 Pole location for the Box-Jenkins model

simulated data is mainly caused by inexact integration but this will not lead to any trouble in the use of the model for controller design. Some differences may also be attributed to backlash caused by poor attachment of the beam on the DC-motor's shaft. The backlash can however not be included in a linear model and is therefore left without consideration. The autocorrelation

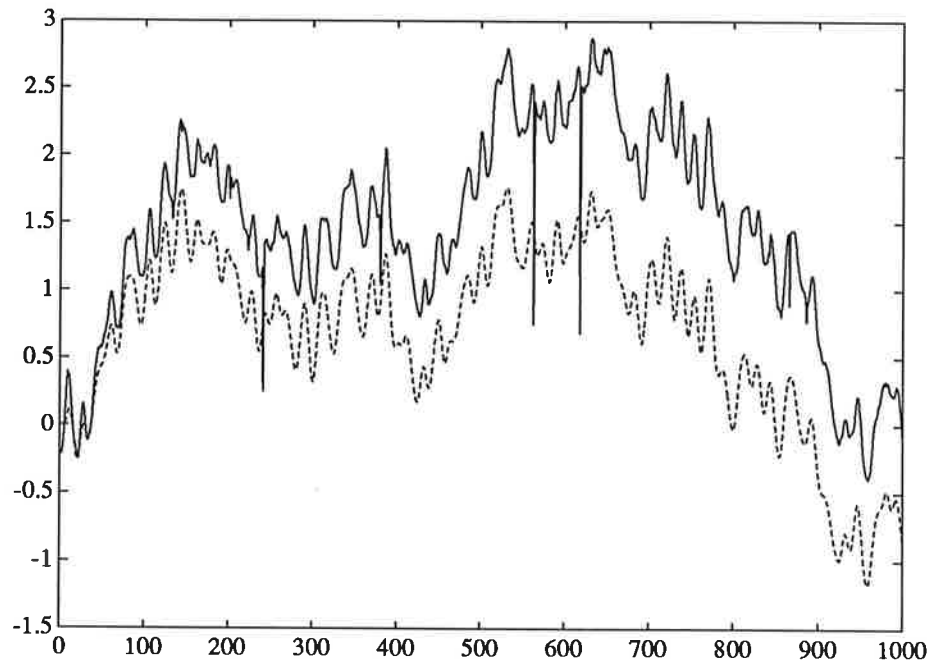


Figure 3.6 Cross validation of Box-Jenkins model. Recorded output (solid line) and output when simulating with estimated model (dashed line)

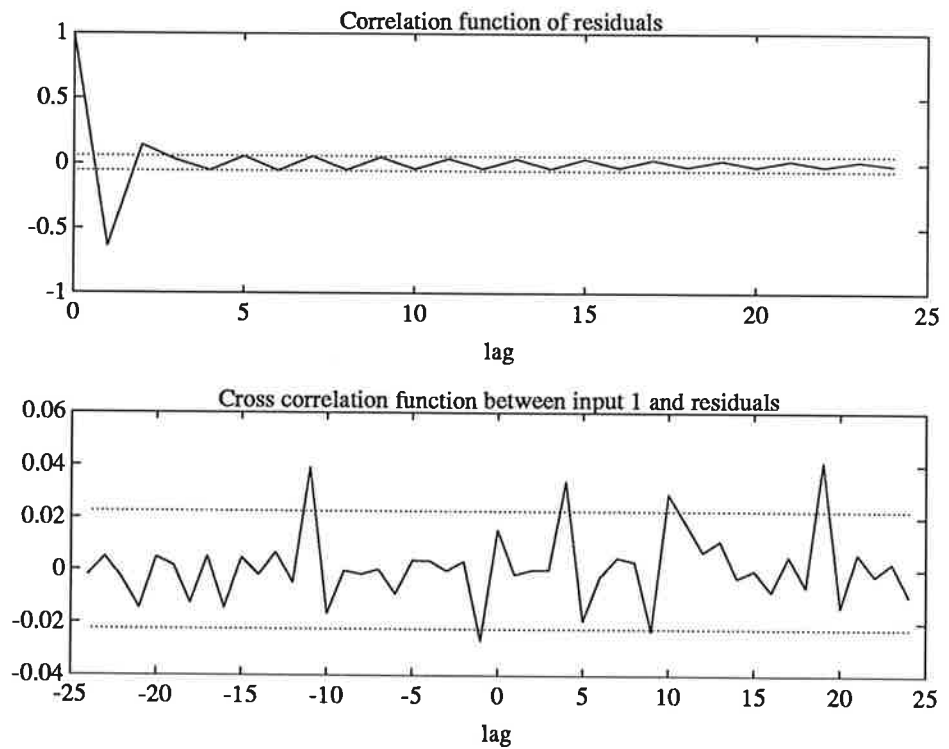


Figure 3.7 Residual test after cross validation

for the residuals and the cross correlation between residuals and input after simulating with unused data are shown in figure 3.7. The residuals show white noise character except for the auto correlation for $lag = 1$. This correlation could not be removed by increasing the model order and was therefore not

considered to be due to a model of too low order.

Results

The Box-Jenkins model was found to be the best and therefore the following transfer function model will be used for controller design.

$$\hat{G}_\phi = \frac{0.0023q}{q^4 - 2.77q^3 + 2.66q^2 - 0.89q} \quad (3.1)$$

with

$$t_{\text{samp}} = 0.005\text{s}$$

3.2 Identification of the ball process.

There are two major problems with the identification of the ball process. One is that the beam angle only can be controlled by the input voltage to the DC-motor. The beam angle will therefore not be constant between sampling intervals, which means that a systematic error is introduced in the identification. The reason for the systematic error is that discrete time identification methods assume constant input signal between sampling intervals. No attention to this was taken during the identification.

The other problem is that the ball and beam process is unstable. The solution is to perform the identification experiments in closed loop. A cascade control structure consisting of an inner loop with a P-regulator for the beam process and an outer loop with a PD-regulator for the total system was sufficient to stabilize the system, see figure 3.8. The regulator was implemented in Simnon as a somewhat modified version of the regulator presented in [simnon]. Identifiability can sometimes be lost when the identification experiments are performed in closed loop. However, only direct parametric identification methods (identification of G_x from measurements of the ball position x and the beam angle ϕ) were considered and the input x_{ref} was chosen to be a PRBS. Therefore no problems with identifiability were expected, see [ss] for further reference.

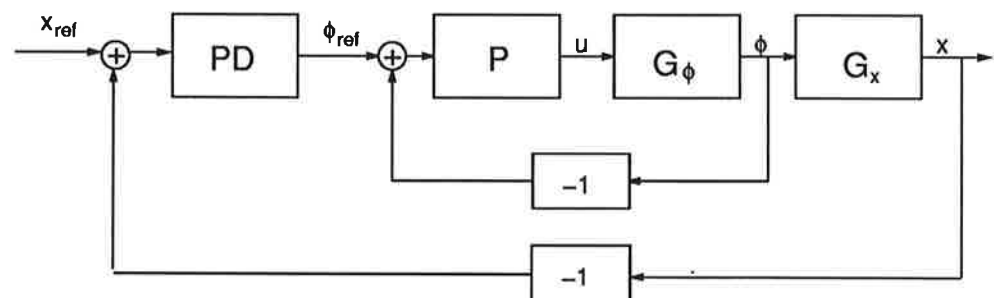


Figure 3.8 Cascade control used for the identification experiments

Data generation

The program logger was used to collect data for the identification and also to generate an input signal, x_{ref} , to the process. A PRBS with *period* = 10, *amplitude* = 0.35V and *tsamp* = 50ms was chosen as input signal. The long

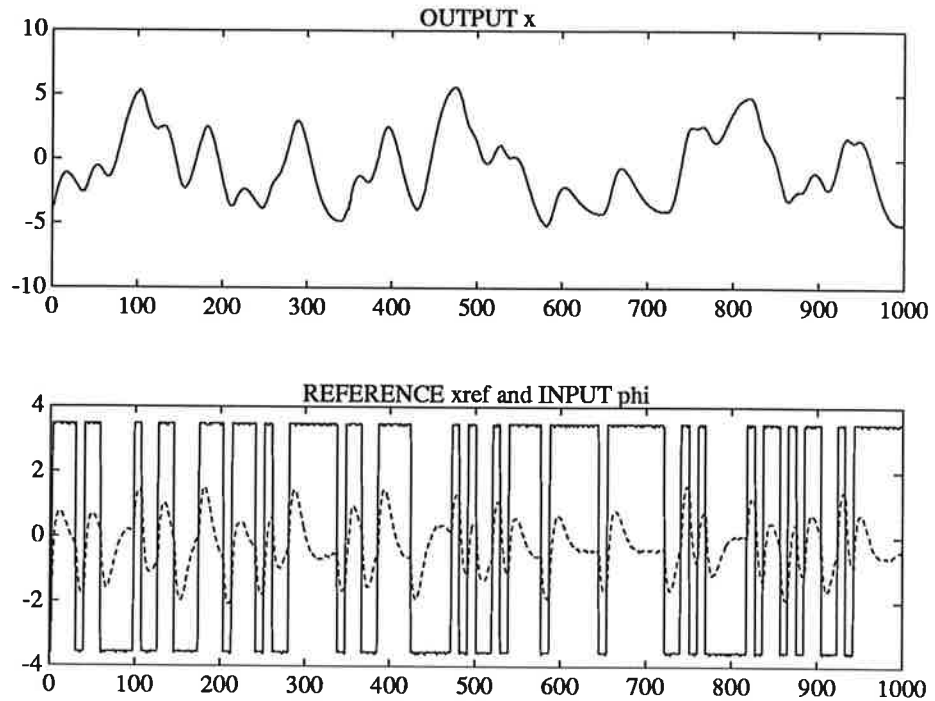


Figure 3.9 Recorded process data. Output x (upper curve), the reference signal x_{ref} and the input signal ϕ (lower curve)

period of the PRBS was chosen because the ball movements along the beam arc very slow and therefore excitation at low frequencies was necessary. 10 000 data points were collected. The first 2000 were used for identification and the next 2000 for cross validation. The first 1000 recorded data can be seen in figure 3.9. The mean value of the data was then removed using the matlab command `detrend`. The first 1000 detrended data points can be seen in figure 3.10.

Choice of model

Different types of models and identification methods were considered. Prediction error methods, with ARMAX, ARX and Box-Jenkins models, and instrumental variable methods were used. The methods were tried with different model orders and then compared. A Box-Jenkins model

$$y(t) = \frac{q^{-k}B(q)}{F(q)}u(t) + \frac{C(q)}{D(q)}e(t)$$

with $\deg(B) = 0$, $\deg(C) = 2$, $\deg(D) = 3$, $\deg(F) = 2$ and $k = 1$ was chosen. This model was obtained with the command

```
>> thbj12321=bj(iddata,[1 2 3 2 1])
```

The coefficients of the model was presented by

```
>> present(thbj12321)
```

with the following result

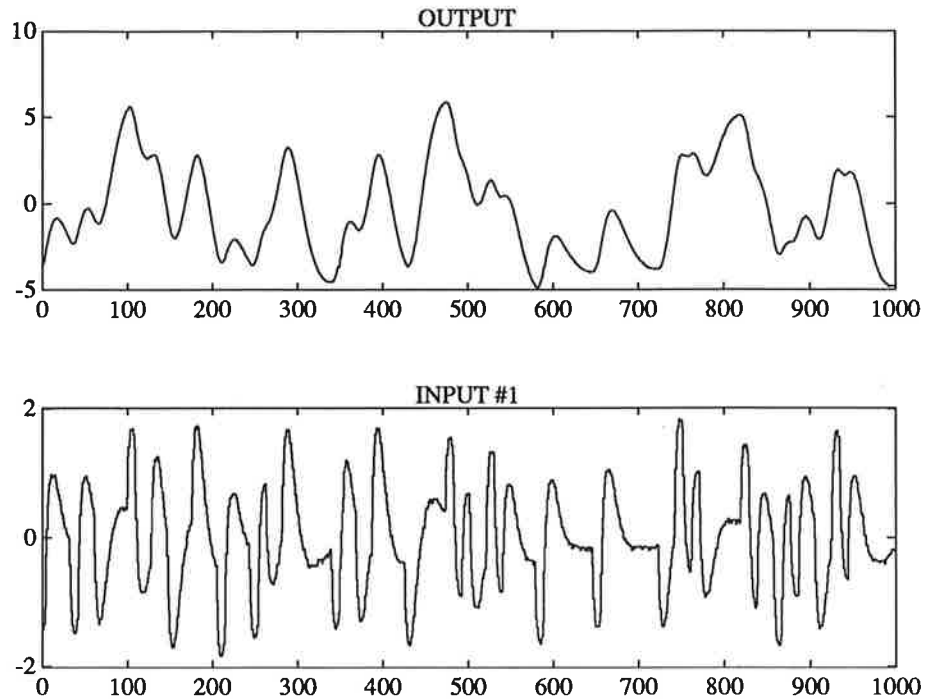


Figure 3.10 Data used for identification. Output z (upper curve) and input ϕ in (lower curve)

This matrix was created by the command BJ on 11/13 1990 at 14:32
 Loss fcn: 0.001444 Akaike's FPE: 0.001455 Sampling interval 1
 The polynomial coefficients and their standard deviations are

B =

0	-0.0246
0	0.0001

F =

1.0000	-1.9962	0.9962
0	0.0006	0.0006

C =

1.0000	-1.2169	0.3445
0	0.0234	0.0241

D=

1.0000	-1.7355	0.4755	0.2600
0	0.0187	0.0369	0.0182

The pole locations for the estimated model are shown in figure 3.11. The exact locations of the poles are,

$$z_1 = 0.9981 + 0.0014i$$

$$z_2 = 0.9981 - 0.0014i$$

thus the pole locations are equivalent to a resonance at a low frequency. The transfer function of the model can be seen in figure 3.12. In accordance with

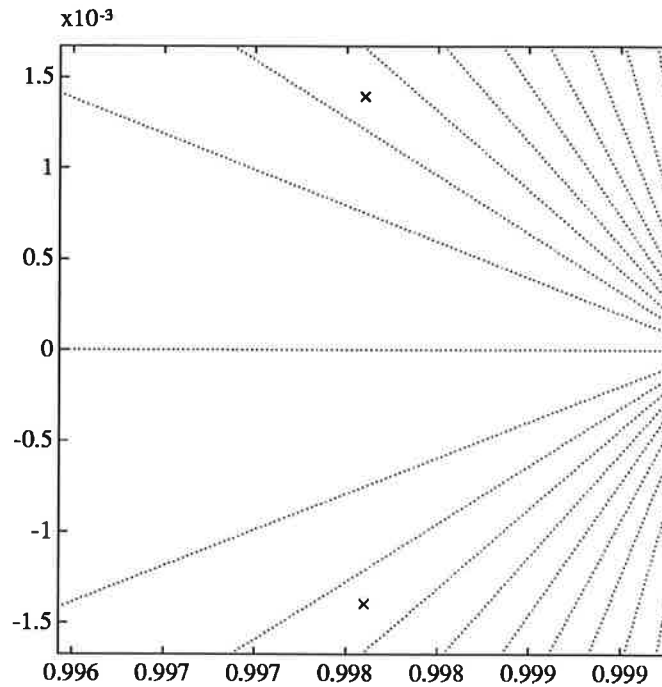


Figure 3.11 Pole locations for the model G_z

the pole locations, a well damped resonance can be seen at the frequency 0.0025Hz. The estimated model therefore behaves as a double integrator for frequencies above approximately 0.007Hz. The reason that the estimated model is not a pure double integrator as expected, is probably that the PRBS does not excite well at the extremely low frequencies. Therefore the identification is only accurate at frequencies excluding the lower extremes. The empirical transfer function estimate, ETFE, is for comparison shown in figure 3.13. The ETFE also suggests a double integrator for low frequencies, while noise dominates at high frequencies.

Validation

As before, cross validation with unused data was considered the most important validation method. 2000 unused data points were used to cross validate the model. The result of the cross validation in figure 3.14 shows that the dynamics in the ball process have been captured, but a drift can be seen. This drift is probably caused by different initial values for the simulated data and the recorded data. An interesting experiment would be to compare the estimated Box-Jenkins model with the double integrator model that was obtained in the theoretical analysis in Chapter 2. This will be investigated in the next section.

The auto correlation for the residuals and the cross correlation between the residuals and the input ϕ are shown in figure 3.15. The residuals show white noise character and the cross correlation between the angle ϕ and the auto-correlation for the residuals are also acceptable.

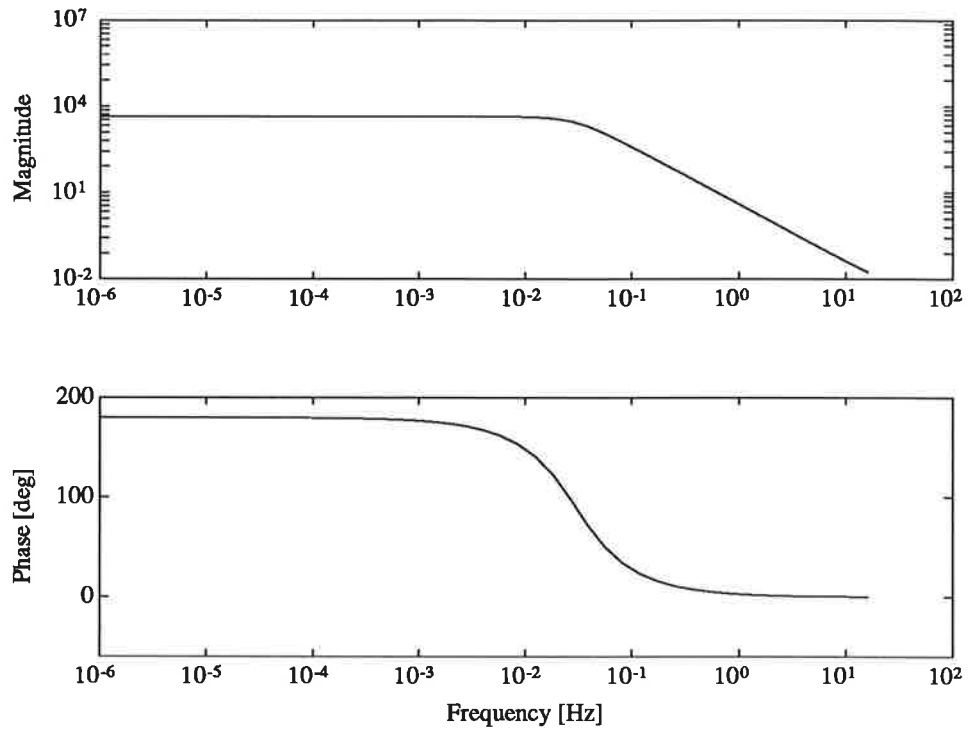


Figure 3.12 Transfer function of the estimated model, G_z

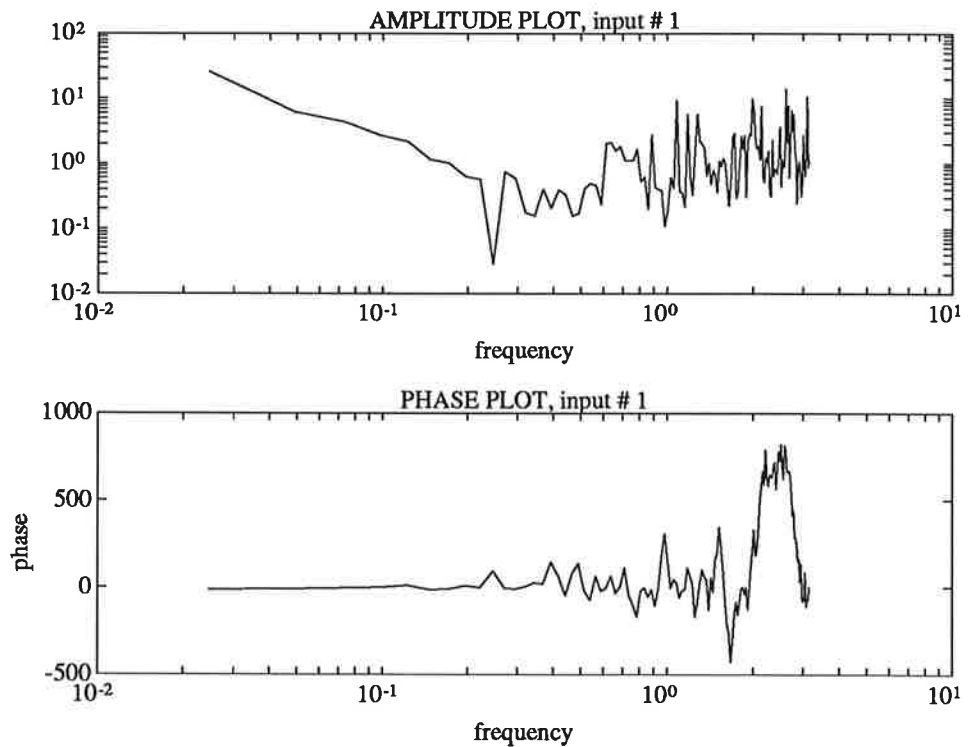


Figure 3.13 The empirical transfer function estimate for the ball process G_z

Identification using continuous time techniques

It was derived in Chapter 2 that the ball process can be approximated with a double integrator. The estimated Box-Jenkins model also indicates that the ball process is a double integrator. It is therefore interesting to investigate the

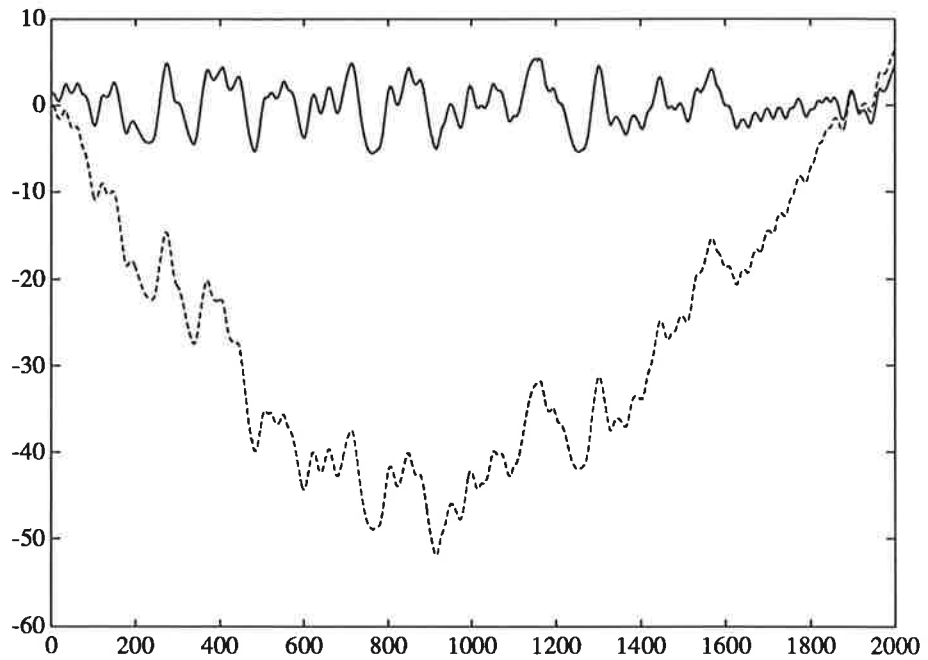


Figure 3.14 Cross validation of Box-Jenkins model. Recorded output (solid line), output when simulating with estimated model (dashed line)

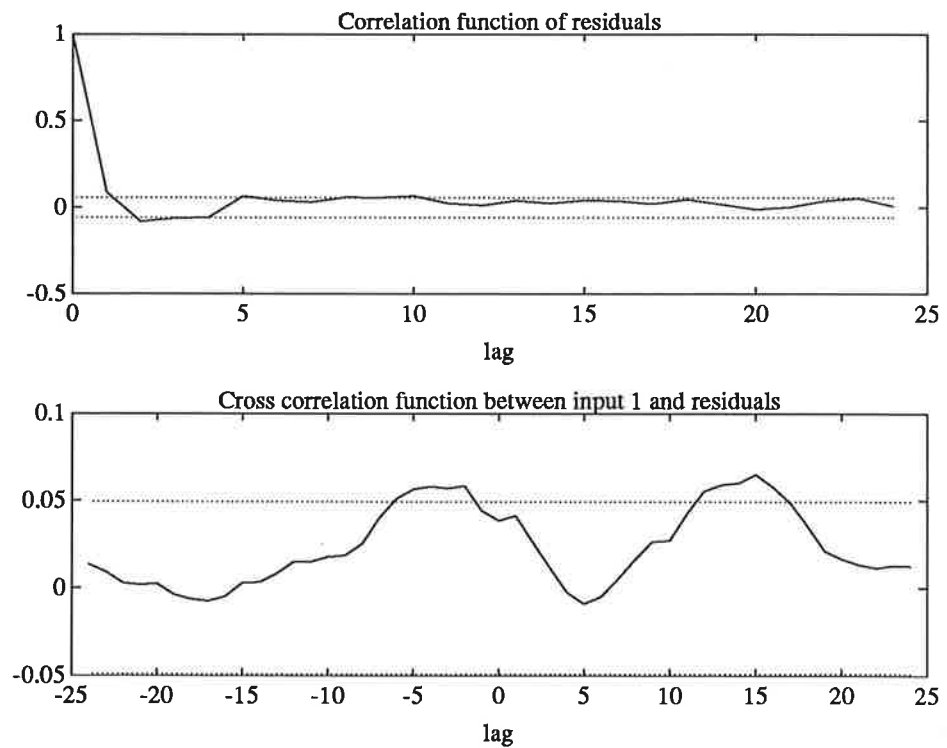


Figure 3.15 Residual test of the Box-Jenkins model

behavior of a pure double integrator model. This will be done in this section where a continuous time identification method will be used to estimate a model with the following structure (see [rolf] for further reference about continuous

time identification)

$$G(s) = \frac{c}{s^2} \quad (3.2)$$

that is a double integrator with the constant c to be estimated. In Chapter 2 c was found to be

$$\begin{cases} c = \frac{-g}{(1+\alpha)} \\ \alpha = \frac{J}{mr^2} = \frac{2}{5} \end{cases} \Rightarrow c = -\frac{5g}{7} \approx 7.0$$

The continuous time identification was performed as follows. An operator λ defined as

$$\lambda = \frac{a}{s+a} \Leftrightarrow s = \frac{a(1-\lambda)}{\lambda}$$

was used to transform (3.2). This results in

$$G^*(\lambda) = \frac{c\lambda^2}{a^2(1-2\lambda+\lambda^2)}$$

which also can be written as

$$\mathbf{x}_f(t) = \mathbf{x}(t) - 2[\lambda\mathbf{x}(t)] + [\lambda^2\mathbf{x}(t)] = \frac{c}{a^2}[\lambda^2\phi(t)]$$

It is now possible to estimate c by using the least squares method

$$\hat{c} = a^2(\Phi'\Phi)^{-1}(\Phi'X_f)$$

where

$$X_f = \begin{bmatrix} \mathbf{x}_f(N) \\ \vdots \\ \mathbf{x}_f(1) \end{bmatrix}, \Phi = \begin{bmatrix} [\lambda^2\phi(N)] \\ \vdots \\ [\lambda^2\phi(1)] \end{bmatrix}$$

The filtering with λ was done in discrete time and the filter constant, a , was chosen to be 1. The motivation for the choice of filter constant was that the ETFE in figure 3.13 shows much noise above 1Hz. The resulting estimate of c was

$$\hat{c} = -9.3$$

which may be compared with the theoretical value $c = -7.0$. The difference in these values are of course due to the scale factors when measuring \mathbf{x} and ϕ , which are measured in Volts and not in meters and radians respectively. Therefore the theoretical value of c is of no practical importance unless the scale factors are determined.

Figure 3.16 shows a cross validation of the model estimated with the continuous time technique. Simulations with the theoretical model and with the estimated Box-Jenkins model are for comparison also shown. The figure shows that both the Box-Jenkins model and the model obtained with continuous time identification techniques captures the dynamics of the true data but they differ somewhat in their drift. The models used for simulation were the zero-order hold equivalents of the continuous time models.

Result

The Box-Jenkins model was found acceptable despite the drift, and the following model will therefore be used in the subsequent regulator design.

$$\hat{G}_x = -\frac{0.0246q}{q^2 - 1.9962q + 0.9962} \quad (3.3)$$

with

$$t_{\text{samp}} = 0.05\text{s}$$

4. Controller design

To control the ball and beam process two controllers were designed, one inner, G_{R_ϕ} , for the beam process and one outer, G_{R_x} , for the total process including the inner controller as shown in figure 4.1. G_{R_ϕ} was designed entirely in the frequency domain by introducing filters in series with G_ϕ and then using a feedback with -1 . G_{R_x} was designed by state feedback and by using a reduced order observer to reconstruct the states. To avoid a stationary error an integrator was introduced as an extra state.

The controllers were implemented in Simnon and therefore the models had to be resampled to 80Hz sampling frequency in order to do the controller

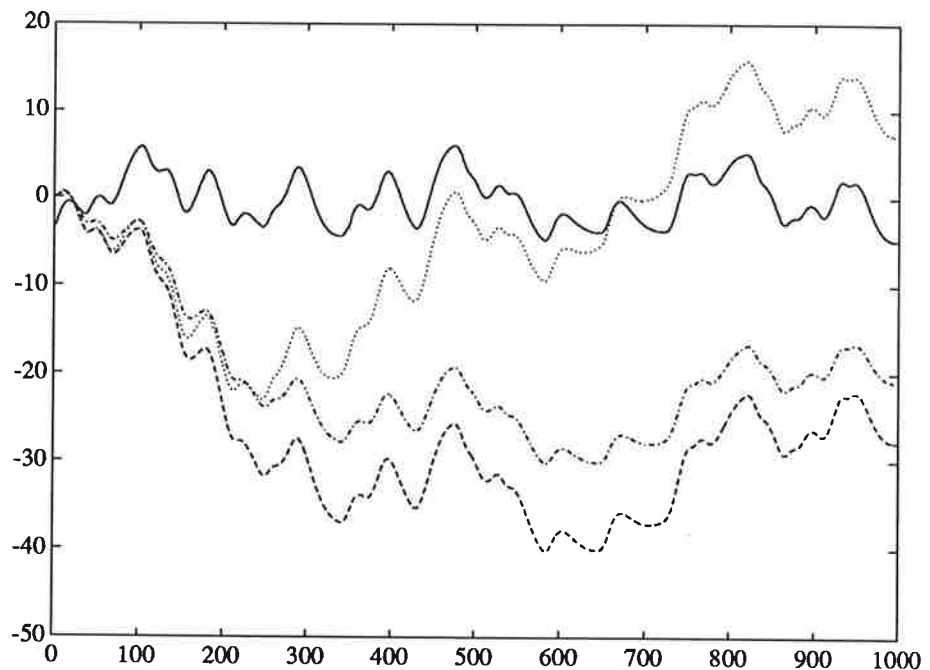


Figure 3.16 Simulation with different models, true output data (solid line), output from model estimated by continuous time technique (dashed line), theoretical model (dash-dotted line) and the Box Jenkins model (dotted line) respectively

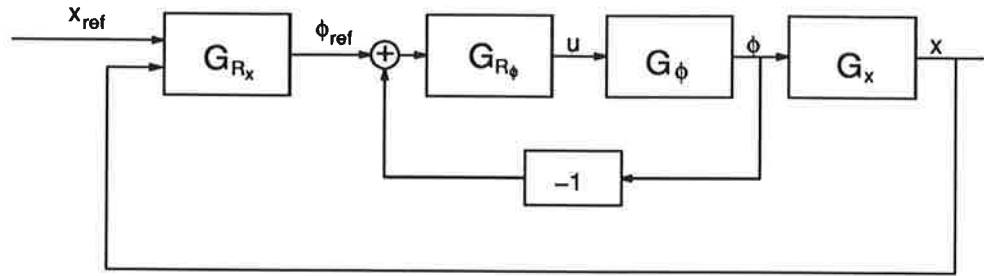


Figure 4.1 Controllers for the ball and beam process

design since this was the fastest frequency that could be used. This is a lower frequency than desired since we are interested in process frequencies up to approximately 15Hz to cover the resonance and therefore would have liked a sampling frequency of at least 150Hz.

4.1 Design of controller for the beam process

The beam dynamics are approximately ten times faster than the ball dynamics, therefore the design of $G_{R\phi}$ is not so critical. The purpose of $G_{R\phi}$ is just to get a smooth movement of the beam to the desired angle. To achieve this, the resonance has to be well damped. This can be done by introducing a notch filter, G_{notch} , in series with G_ϕ . The notch filter consists mainly of two zeros placed close to the resonance poles in order to cancel these out. To further smooth the behavior a second order LP-filter of Butterworth type, G_{LP} , was placed in series with G_ϕ . Both G_{notch} and G_{LP} contain gains to get as high cross over frequency as possible for the open loop system while retaining an acceptable stability margin. An extra gain was then introduced after the design to facilitate adjustments when trying out the controller on the process. The structure of the controller for the beam process can be seen in figure 4.2.

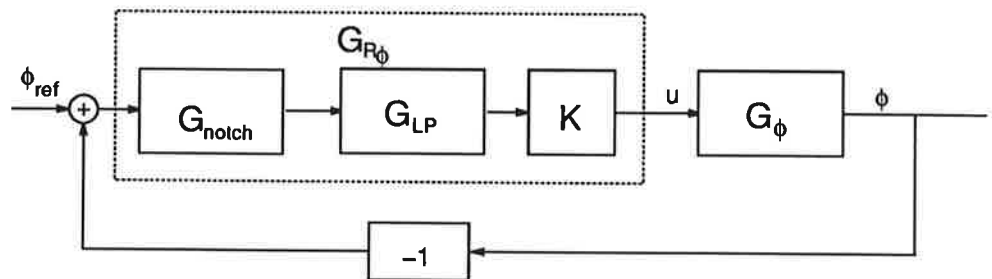


Figure 4.2 Controller structure for the beam process

The controller was designed by first resampling the identified model and then interactively by plotting the Bodediagram for the open loop system $G_\phi G_{notch} G_{LP}$ when varying G_{notch} and G_{LP} . The frequency responses for the Butterworth filter, notch filter and total open loop system can be found in figures 4.3 and 4.4. To do the design the command sequence in listing 4.1 was used.

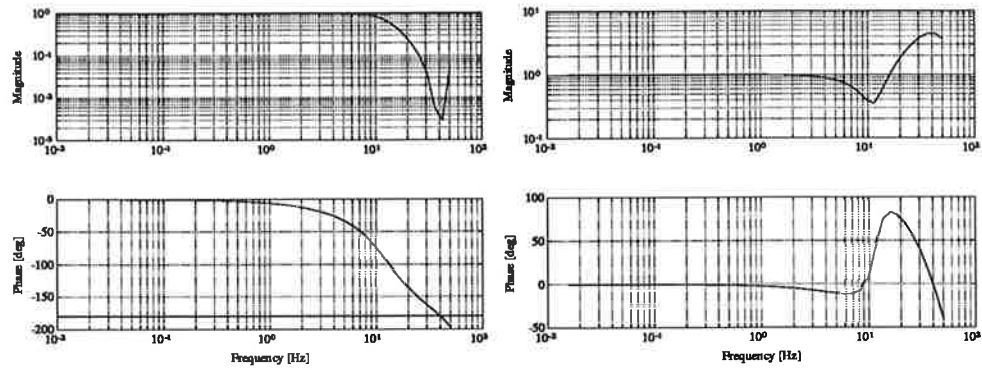


Figure 4.3 Frequency response for the Butterworth LP-filter (left) and the notch filter (right)

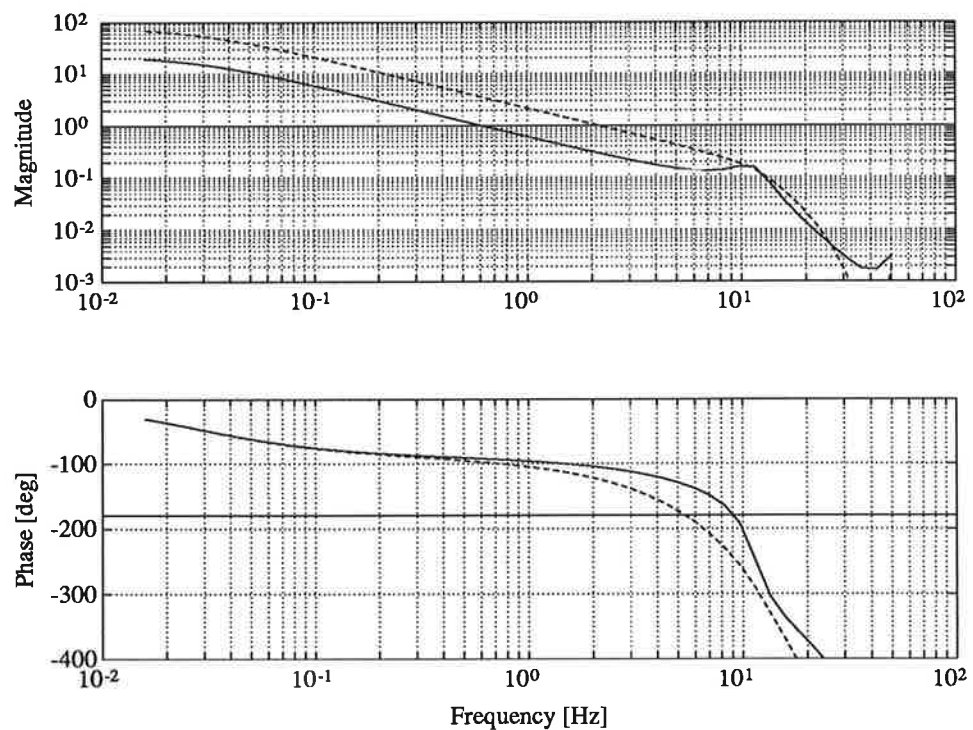


Figure 4.4 Open loop frequency response for the system in figure 4.2, uncompensated system (solid line) and compensated system (dashed line).

Results

The controller was found to work well when trying it out for beam angle control. The gain K could be increased to 1.8, with the beam movement still being smooth. Vibrations in a direction orthogonal to ϕ were however noticed. What caused these vibrations could not be entirely investigated but they probably arose because of poor alignment in the attachment of the beam. Interestingly they were much more dominant when the beam moved in counter clockwise direction. Since these vibrations were unmeasurable, the identification and the controller design could not be done with consideration taken to them. They were removed by limiting the maximum input voltage to the DC-motor. This voltage is normally restricted to $|u|_{max} \leq 1$ but this interval now had to be reduced to $|u|_{max} \leq 0.25$. Another way of eliminating the vibrations could have been to generate smooth trajectory from the step in reference value and

```

>>[a1,b1,c1,d1,f1]=polyform(thbjof12334);
>>[phi gamma cc dd]=tf2ss(b1,[f1,0]);
>>[A B]=d2c(phi,gamma,0.005);
>>[phi1 gamma1]=c2d(A,B,0.0125);
>>[bb1 aa1]=ss2tf(phi1,gamma1,cc,dd,1)
bb1 =
      0   -0.0002   0.0092   0.0203   0.0019
aa1 =
      1.0000  -2.0930   1.8369  -0.7425   0.0000
>>[bf af]=butter(2,.30)
bf =
      0.1311   0.2622   0.1311
af =
      1.0000  -0.7478   0.2722
>>numnot=1.6*[1,-2*0.85*cos(7/8),0.85^2]
numnot =
      1.6000  -1.7435   1.1560
>>dennot=[1,0,0]
dennot =
      1   0   0
>>s2=conv(bf,numnot)
s2 =
      0.2098   0.1910  -0.0958   0.0745   0.1516
>>r2=conv(af,dennot)
r2 =
      1.0000  -0.7478   0.2722   0   0
>>K=1.8
K =
      1.8

```

Listing 4.1 Matlab sequence used for design of G_{R_ϕ}

then use this trajectory as reference to the system. Step responses for the closed loop system with the two different restrictions on u are shown in figures 4.5 and 4.6 .

The final controller was

$$G_{R_\phi} = 1.8 \frac{0.2098q^4 + 0.1910q^3 - 0.0958q^2 + 0.0745q + 0.1516}{q^4 - 0.7478q^3 + 0.2722q^2}$$

and

$$t_{\text{samp}} = 0.0125\text{s}$$

with $|u|_{\text{max}}$ restricted to 0.25.

4.2 Design of controller for the total system

A state-feedback design was chosen for G_{R_x} . One difficulty with this is that the inner loop, consisting of the beam process and its controller, is of high order which implies design of a complex controller. To avoid this, the system

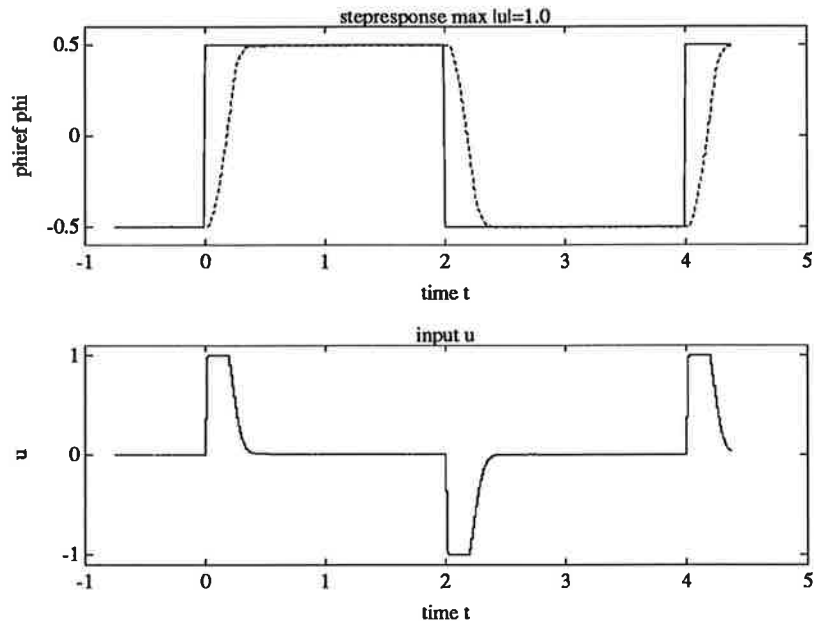


Figure 4.5 Step response for system when input limited to ± 1

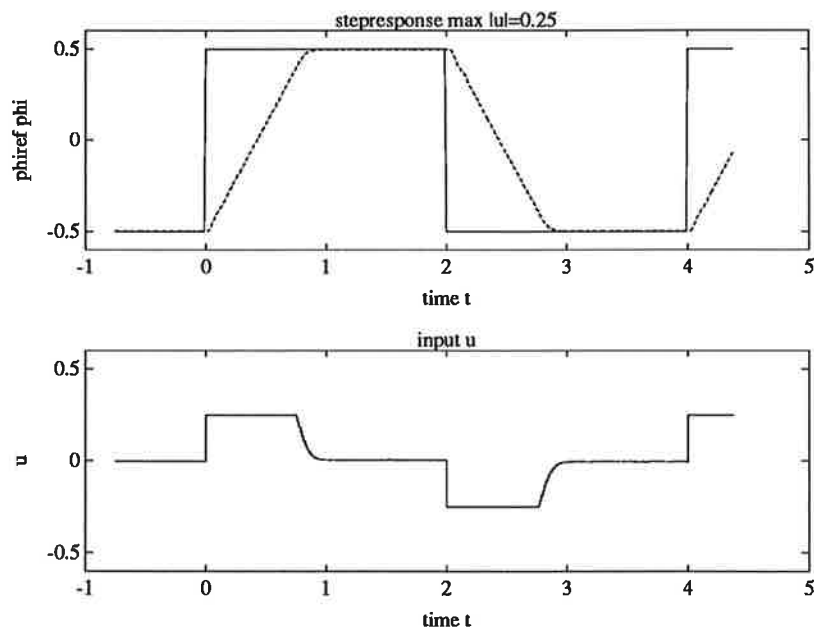


Figure 4.6 Step response for system when input limited to ± 0.25

order was reduced by approximating the inner loop with a first order system. A state corresponding to an integrator was included in order to avoid error in stationarity and a state-feedback and an observer were designed. A block diagram of the total system is shown in figure 4.7. Numerical values are shown in the last section, where the results are given.

First order approximation of the inner loop

The seventh order inner loop, consisting of the beam process and its regulator,

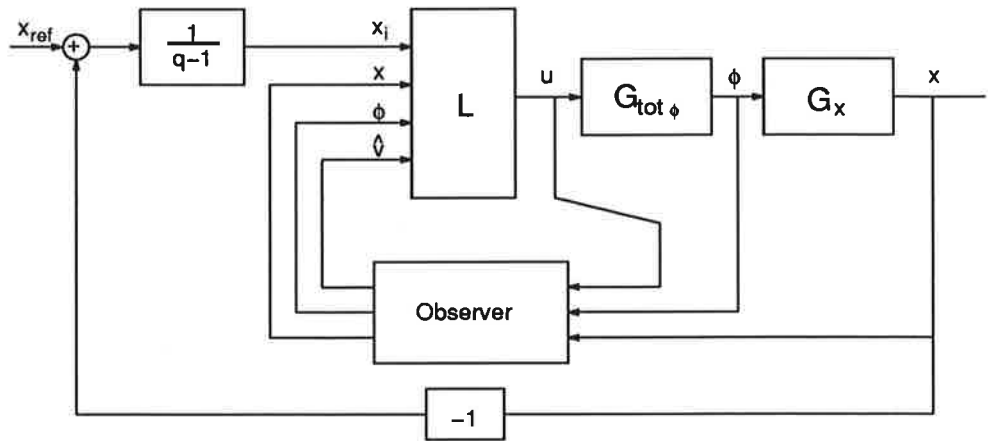


Figure 4.7 Block diagram of the total system

was approximated with the following first order continuous time system

$$G(s) = \frac{\omega_0}{(s + \omega_0)}$$

where

$$\omega_0 = 2\pi \cdot 8$$

As can be seen in figure 4.8 this approximation is good only for low frequencies. The bandwidth of the total system will however be approximately 0.3Hz, and therefore good agreement is needed only at low frequencies.

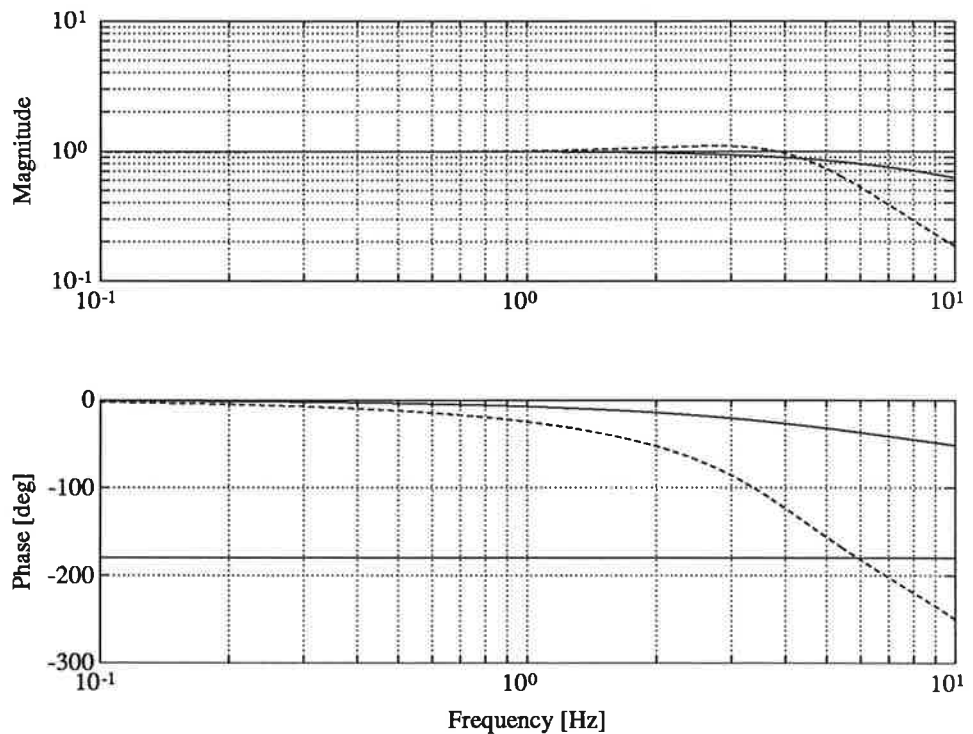


Figure 4.8 Transfer functions for $G_{tot\phi}$ and its first order approximation

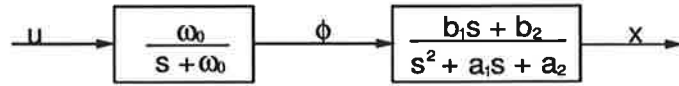


Figure 4.9 Block diagram of the reduced system used for control design

The reduced total system

The total system after reduction consists of the first order approximation of the inner loop and the ball process. The estimated transfer function for the ball process \hat{G}_x in 3.3 was translated to continuous time with a second order transfer function as result. Thus, the block diagram of the total system after reduction is as shown in figure 4.9.

This reduced system has a state-space representation as in (4.1).

$$\frac{dx}{dt} = \begin{pmatrix} -\omega_0 & 0 & 0 \\ b_1 & -a_1 & 1 \\ b_2 & -a_2 & 0 \end{pmatrix} x + \begin{pmatrix} \omega_0 \\ 0 \\ 0 \end{pmatrix} u$$

$$y = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} x \quad (4.1)$$

where

$$x(t) = \begin{pmatrix} \phi(t) \\ x(t) \\ v(t) \end{pmatrix}$$

The third state in the state vector, v , has an approximate physical interpretation as the speed of the ball. This is due to the fact that the estimated transfer function \hat{G}_x is almost a double integrator. The continuous time state model above was discretized with sampling frequency 80Hz (the sampling frequency used in the resulting controller). The resulting discrete time model has the following structure

$$x(t+1) = \begin{pmatrix} \phi_{11} & \phi_{12} & \phi_{13} \\ \phi_{21} & \phi_{22} & \phi_{23} \\ \phi_{31} & \phi_{32} & \phi_{33} \end{pmatrix} x(t) + \begin{pmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \end{pmatrix} u(t)$$

$$y = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} x \quad (4.2)$$

where

$$\phi_{12} = \phi_{13} = 0$$

This state model will be used for design of the discrete time observer and the state-feedback controller.

Design of the observer

The total reduced system in (4.2) has three states, two of which can be measured. It is therefore necessary with either a third order observer or a first order reduced observer. The second alternative was chosen. The design of the reduced order observer is straightforward (see [rhodes] for further reference). The observer uses the restricted value of the input voltage u to the DC-motor,

see section 4.1. The block diagram of the reduced order observer is shown in figure 4.10 with the matrices defined as follows

$$\Phi = \left(\begin{array}{cc|c} \Phi_{11} & \Phi_{12} & \phi_{13} \\ \Phi_{21} & \Phi_{22} & \phi_{23} \\ \hline \Phi_{31} & \Phi_{32} & \phi_{33} \end{array} \right) = \begin{pmatrix} \phi_{11} & \phi_{12} & \phi_{13} \\ \phi_{21} & \phi_{22} & \phi_{23} \\ \phi_{31} & \phi_{32} & \phi_{33} \end{pmatrix}$$

$$\Gamma = \begin{pmatrix} \Gamma_1 \\ \Gamma_2 \end{pmatrix} = \begin{pmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \end{pmatrix}$$

$$K = \begin{pmatrix} k_1 & k_2 \end{pmatrix}$$

$$\Psi = \Phi_{21} - K\Phi_{11} + \Phi_{22}K - K\Phi_{12}K$$

$$\Delta = \Gamma_2 - K\Gamma_1$$

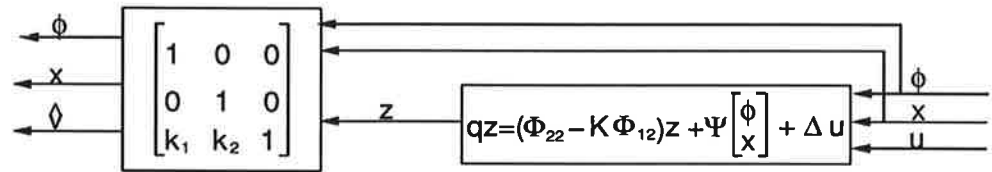


Figure 4.10 Block diagram of the reduced order observer

The coefficients k_1 and k_2 were chosen so that the observer pole was placed in $z = 0.95$. This makes the observer dynamics faster than the dynamics for the closed loop system (see next section for pole locations of the closed loop system). Notice that the choice of k_1 and k_2 is not unique, only the scalar product

$$K\Phi_{12} = \begin{pmatrix} k_1 & k_2 \end{pmatrix} \begin{pmatrix} \phi_{13} \\ \phi_{23} \end{pmatrix}$$

need to be specified. This gives a degree of freedom in the choice of k_1 and k_2 . Since $\phi_{13} = 0$, the value of k_1 does not affect the pole location for the reduced order observer. However, the Ψ and Δ matrices are affected by the value of k_1 . In the project k_1 was chosen to 0. This turned out to give a good observer and no further investigation of how the observer performance depended on k_1 was undertaken. The resulting matrices with $k_1 = 0$ are

$$\Psi = \begin{pmatrix} \psi_1 & \psi_2 \end{pmatrix} = \begin{pmatrix} \phi_{31} - k_2\phi_{21} & \phi_{32} - k_2(\phi_{22} - \phi_{33} + k_2\phi_{23}) \end{pmatrix}$$

$$\Delta = \gamma_3 - k_2\gamma_2$$

Design of the state feedback

The third order reduced system in (4.2) was extended with a fourth state corresponding to an integrator (see figure 4.7), and a state-feedback law was

designed for the extended system with the following state space representation.

$$\mathbf{x}_e(t+1) = \begin{pmatrix} \phi_{11} & \phi_{12} & \phi_{13} & 0 \\ \phi_{21} & \phi_{22} & \phi_{23} & 0 \\ \phi_{31} & \phi_{32} & \phi_{33} & 0 \\ 0 & -1 & 0 & 1 \end{pmatrix} \mathbf{x}_e(t) + \begin{pmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \\ 0 \end{pmatrix} u(t) + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \mathbf{x}_{\text{ref}}(t)$$

$$\mathbf{y} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \mathbf{x}_e \quad (4.3)$$

where

$$\mathbf{x}_e(t) = \begin{pmatrix} \phi(t) \\ \mathbf{x}(t) \\ v(t) \\ \mathbf{x}_i(t) \end{pmatrix}$$

and the control law is defined as

$$u = -L\mathbf{x}_e = - \begin{pmatrix} l_1 & l_2 & l_3 & l_4 \end{pmatrix} \begin{pmatrix} \phi \\ \mathbf{x} \\ v \\ \mathbf{x}_i \end{pmatrix}$$

The poles of the closed loop system were placed in the following locations, see figure 4.11.

$$p_1 = 0.5335$$

$$p_2 = 0.9782$$

$$p_3 = 0.9790 + j0.0056$$

$$p_4 = 0.9790 - j0.0056$$

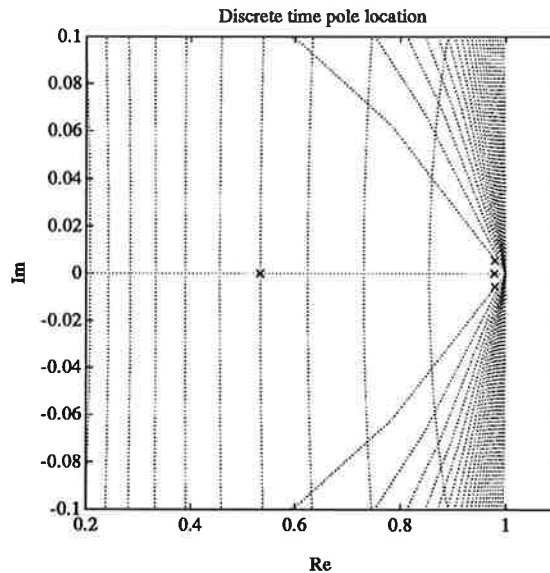


Figure 4.11 Pole locations for state feedback design (discrete time)

Since it is easier to interpret the physical implications of continuous time poles the corresponding continuous time pole locations are shown in figure 4.12

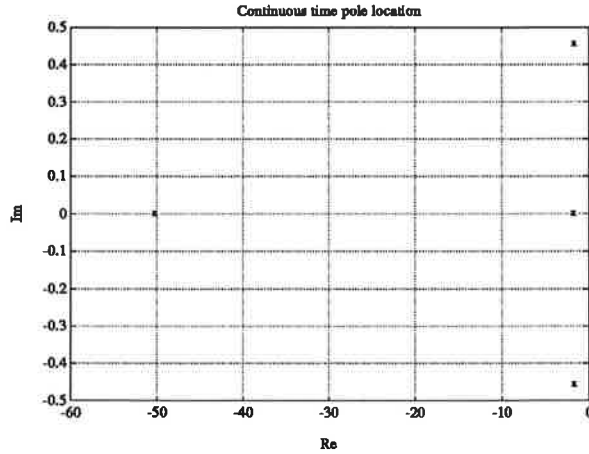


Figure 4.12 Pole locations for state feedback design (continuous time)

Results

The following system was obtained, see equation (4.3)

$$\mathbf{x}_e(t+1) = \begin{pmatrix} 0.5335 & 0 & 0 & 0 \\ -0.0029 & 0.9991 & 0.0125 & 0 \\ -0.0915 & 0.0000 & 1.0000 & 0 \\ 0 & -1 & 0 & 1 \end{pmatrix} \mathbf{x}_e(t) + \begin{pmatrix} 0.4665 \\ 0.0009 \\ -0.0317 \\ 0 \end{pmatrix} u(t) + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \mathbf{x}_{\text{ref}}(t)$$

$$y(t) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \mathbf{x}_e(t)$$

This gave the following reduced order observer

$$z(t+1) = (\Phi_{22} - K\Phi_{12})z(t) + \Psi \begin{pmatrix} \phi \\ \mathbf{x} \end{pmatrix} + \Delta u$$

$$\hat{v} = \begin{pmatrix} k_1 & k_2 & 1 \end{pmatrix} \begin{pmatrix} \phi \\ \mathbf{x} \\ z \end{pmatrix}$$

where

$$K = \begin{pmatrix} k_1 & k_2 \end{pmatrix} = \begin{pmatrix} 0.0000 & 4.000 \end{pmatrix}$$

$$\Phi_{22} - K\Phi_{12} = 0.95$$

$$\Psi = \begin{pmatrix} \psi_1 & \psi_2 \end{pmatrix} = \begin{pmatrix} -0.0859 & -0.199532 \end{pmatrix}$$

$$\Delta = 0.0222$$

and the control law was

$$u = -L\mathbf{x}_e = - \begin{pmatrix} l_1 & l_2 & l_3 & l_4 \end{pmatrix} \begin{pmatrix} \phi \\ \mathbf{x} \\ \hat{v} \\ \mathbf{w}_1 \end{pmatrix}$$

with

$$L = \begin{pmatrix} l_1 & l_2 & l_3 & l_4 \end{pmatrix} = \begin{pmatrix} 0.0990 & -0.8573 & -0.5004 & 0.0067 \end{pmatrix}$$

From the step response in figure 4.13 it is seen that the resulting controller gives a closed loop system without overshoot. The ball movement along the beam was however a bit jerky. This was to some extent caused by dirt and dents on the ball track, which disturbed the measurement of the ball position. This problem might be overcome by using a full order observer, which is less sensitive to disturbances.

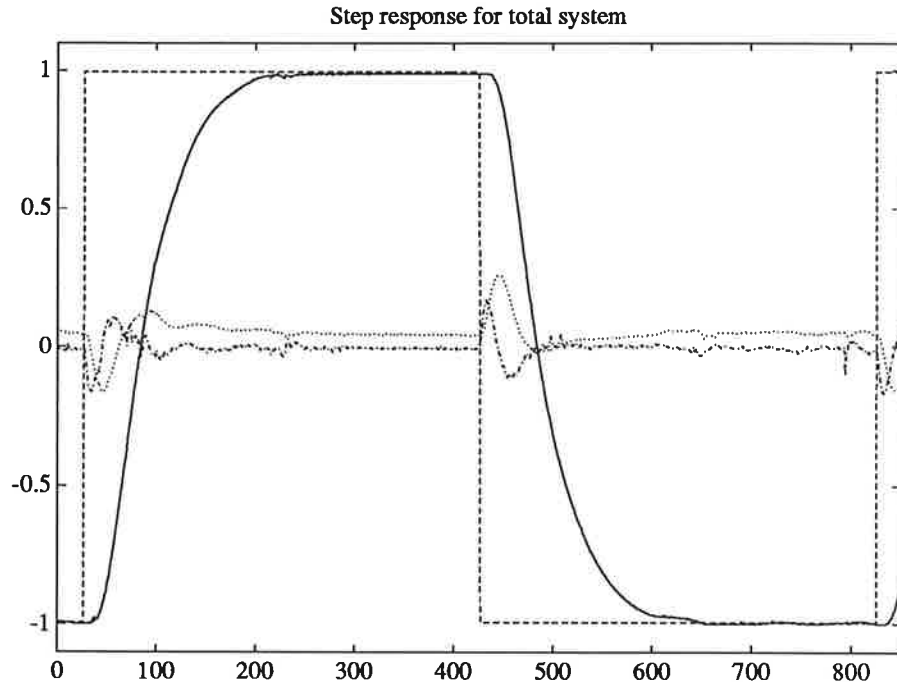


Figure 4.13 Step response for the total system with 80Hz sampling frequency, position reference (dashed line), ball position (solid line), beam angle (dotted line), input voltage to the DC-motor (dash-dotted line).

5. Conclusions

The project has shown that the theory developed during lectures could be applied to a real world process with good results. A few modifications of the identification procedure could however improve the results. Using an anti-aliasing filter when recording data for the identification could give better accuracy for the estimated models. However, no problems were noticed. Taking into consideration the fact that the identification of the ball process was done without a piecewise constant input signal might also be an improvement.

The controller performance could also be enhanced. For example the sensitivity to disturbances in the ball position signal could be reduced by filtering

of this signal. One way of doing this is to build a full order observer instead of the reduced one that was used. The controller should also be complemented with an anti-windup mechanism. Another improvement would be to use an antialiasing filter in the controller.

References

- [1] Ola Dahl. Solutions to exercises in system identification. Lund Institute of Technology.
- [2] Rolf Johansson. Lecture notes in system identification. Lund Institute of Technology, 1990.
- [3] Ian B. Rhodes. Lecture notes in linear systems. University of California, Santa Barbara, 1990.
- [4] Torsten Söderström and Petre Stoica. *System Identification*. Prentice Hall International (UK) Ltd, 1989.
- [5] SSPA Systems. *SIMNON User's Guide for MS-DOS Computers*. Göteborg, 1990.