

ADAPTIV REGULATOR BASERAD PÅ LÄRANDE MODELL

Examensarbete utfört av ANDREAS PAULI

Ansvarig handledare BJÖRN WITTENMARK

RE-74

TILLHÖR REFERENSBIBLIOTEKET
UTLÄNAS EJ

LUNDS TEKNISKA HÖGSKOLA
INST. FÖR REGLERINGSTEKNIK
FEB. 1970

ADAPTIV REGULATOR BASERAD PÅ LÄRANDE MODELL

Examensarbete utfört av ANDREAS PAULI

Ansvarig handledare BJÖRN WITTENMARK

LUNDS TEKNISKA HÖGSKOLA
INST. FÖR REGLERINGSTEKNIK
FEB. 1970

Innehållsförteckning

Summary1
Sammanfattning2
1. Inledning3
2. Reelltidsidentifiering baserad på "steepest descent" metoden5
3. Testexempel	
3.1 Öppet system10
3.2 Återkopplat system12
Diagram till kap. 316
4. Beskrivning av robot modell25
Diagram till kap. 428
5. Test av identifieringsmetoden på robot modell34
Skalning av systemparametrarna36
Diagram till kap. 539
6. Sammanfattning av resultat42
Referenser44
Appendix	
Programbeskrivning45

Summary.

In this work is made an attempt to find an adequate control law to an adaptive system by means of a computer.

The point is how to obtain sufficient identification of the parameters describing the process.

The method is based upon the following concepts: Find a mathematical model of the process. Adjust the parameters of the model by a gradient method.

$$x(t+1) = A(t) \cdot x(t) + B(t) \cdot u(t)$$

$$y(t) = C \cdot x(t)$$

The parameters of $A(t)$ and $B(t)$ identified in this way are used to design a feed-back loop so that the closed-loop system is given desirable properties.

The actual processes are of 2:nd order with one feed-signal $u(t)$ and all states measureable.

The method of identification was successful in processes where the parameters were slowly varying.

As to the robot that was later simulated, the parameters could not be identified sufficiently well to design a proper feed-back loop during all of the simulation. However during certain parts of the simulation a satisfactory feed-back was recorded.

Sammanfattning.

I detta examensarbete görs ett försök att med en dator finna en lämplig styr lag för ett adaptivt system. Problemet är att tillräckligt bra identifiera de parametrar, som beskriver den aktuella processen. Identifieringsmetoden bygger på att man konstruerar en matematisk modell, vars parametrar justeras med en gradient metod.

$$x(t+1) = A(t) \cdot x(t) + B(t) \cdot u(t)$$

$$y(t) = C \cdot x(t)$$

De sålunda identifierade parametrarna i $A(t)$ och $B(t)$ skall utnyttjas för att bestämma en återkoppling, så att det slutna systemet får önskade egenskaper.

De processer, som studeras är av 2:a ordningen med en insignal och samtliga tillstånd mätbara.

Identifieringsmetoden lyckades för processer där de ingående parametrarna varierade långsamt.

För den robot, som senare simulerades, kunde parametrarna inte identifieras tillräckligt väl för att under hela simuleringen utnyttja dessa till att återkoppla systemet.

Dock gick det att finna acceptabel återkoppling under delar av simuleringen.

1. Inledning.

I flygplan används s.k. autopiloter, vilka är styrsystem vilkas syfte är att hålla flygplanet på beordrad kurs. Flygplanets beteende är emellertid beroende av bl. a. hastighet och statiskt lufttryck, dvs för ett bestämt roderutslag får man olika reaktion vid olika flygfall. Denna inverkan önskar man i allmänhet eliminera.

I tidiga versioner av autopiloter bestämde man därför ett antal styrlagar för olika flygfall.

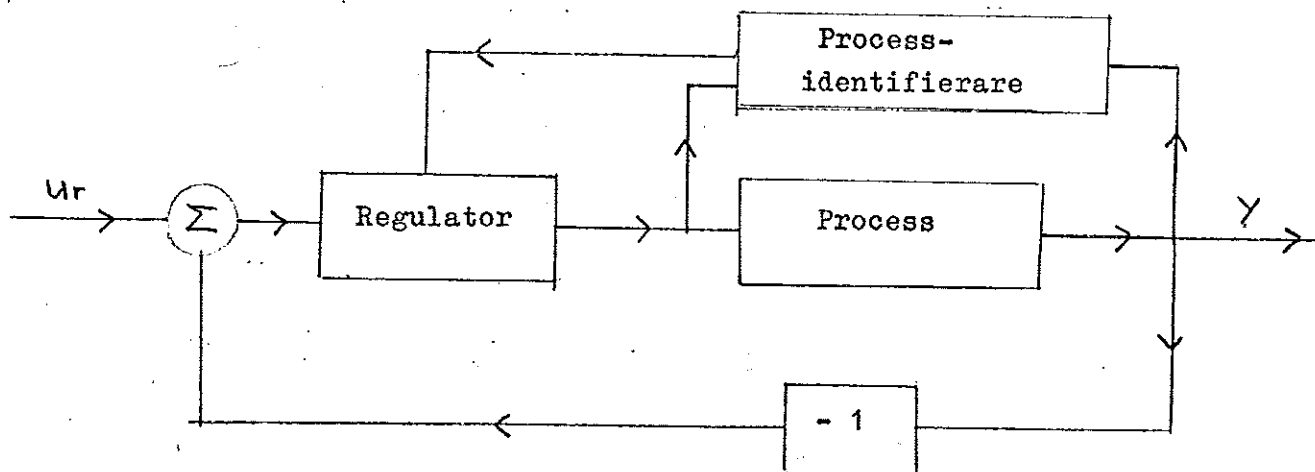
På senare år har emellertid utvecklats metoder, vilka direkt utnyttjar tillgängliga matsignaler för att bestämma lämpliga koefficienter i styrlagen.

Den metod som här skall undersökas, bygger på att man konstruerar en matematisk modell för den process man vill styra.

Parametrarna i denna modell identifieras.

Därför kallas den ibland "Lärande modell". Identifieringsmetoden beskrivs i kap. 2 .

Med kännedom om hur dessa parametrar varierar skall lämpliga koefficienter i regulatorn bestämmas.



}

Som process används dels olika processer utan fysikalisk anknytning och dels en simulering av en robot, vilken beskrives av ett 2:a ordn. linjärt dynamiskt system (kap 3).

$$x(t+1) = A(t) \cdot x(t) + B(t) \cdot u(t)$$

$$y(t) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x(t)$$

För att erhålla överföringsfunktionerna $H_1(q) = \frac{x_1(t)}{u(t)}$ resp

$$H_2(q) = \frac{x_2(t)}{u(t)} \quad \text{beräknas}$$

$$H(q) = C(qI - A)^{-1}$$

Uträknat fås

$$H_1(q) = \frac{b_1(q - a_{22}) + b_2 a_{12}}{q^2 - q(a_{11} + a_{22}) + a_{11} a_{22} - a_{12} a_{21}}$$

$$H_2(q) = \frac{b_1 a_{21} + b_2(q - a_{11})}{q^2 - q(a_{11} + a_{22}) + a_{11} a_{22} - a_{12} a_{21}}$$

Då alla tillståndsvariablerna är mätbara är det möjligt att med återkoppling placera polerna på önskat sätt.

Ett väsentligt krav är att det styrda systemets statiska förstärkning skall hållas någorlunda konstant. Detta kan i det betraktade fallet ordnas genom att införa en variabel förstärkning.

För dessa manipulationer krävs emellertid att god identifiering av parametrarna i modellen $x(t+1) = A(t) \cdot x(t) + B(t) \cdot u(t)$ erhålles.

Förutom detta krav måste systemet vara tillräckligt snabbt, ej ge för stor översläng samt ge acceptabel noggrannhet.

2. Reelltidsidentifiering baserad på "steepest descent" metoden.

Betrakta

$$x(t+1) = Ax(t) + Bu(t) \quad (2.1)$$

A och B okända. $x(t)$, $x(t+1)$ är direkt mätbara

Problemet är att finna uppskattningar av A och B så att följande gäller

$$x(t+1) = \hat{A} x(t) + \hat{B} u(t) + \tilde{x}(t)$$

där $\tilde{x}(t)$ är en felvektor

$$\tilde{x}(t) = x(t+1) - \hat{A}x(t) - \hat{B}u(t) \quad (2.2)$$

(2.1) och (2.2) ger tillsammans

$$\tilde{x}(t) = (A - \hat{A}) x(t) + (B - \hat{B}) u(t) \quad (2.3)$$

I fortsättningen kommer $x(t)$ resp. $u(t)$ att skrivas x resp. u . Likaså kommer $a_{ij}(t)$ resp. $b_i(t)$ att skrivas a_{ij} resp. b_i .

Genom att derivera någon norm av felet, låtsäga $\tilde{x}^T(t) \tilde{x}(t)$

partiellt m.a.p. \hat{a}_{ij} resp \hat{b}_i erhålles minimum för $\tilde{x}(t)$

m.a.p. \hat{a}_{ij} resp \hat{b}_i .

$$e^2 = \tilde{x}^T \tilde{x} = \sum_i (x_i(t+1) - \sum_j \hat{a}_{ij} x_j - \hat{b}_i u)^2$$

$$\frac{\partial e^2}{\partial \hat{a}_{ij}} = -2 \tilde{x}_i x_j \quad (2.4)$$

$$\frac{\partial e^2}{\partial \hat{b}_i} = -2 \tilde{x}_i u \quad (2.5)$$

Härav inses att om följande införes

$$\begin{aligned} \hat{a}_{ij}(t+1) &= \hat{a}_{ij}(t) + k \tilde{x}_i(t) x_j(t) \\ \hat{b}_i(t+1) &= \hat{b}_i(t) + k \tilde{x}_i(t) u(t) \end{aligned} \quad (2.7)$$

kommer, vid lämpligt val av k, felet att minska (steepest descent).

$x(t)$ utskrivnen i komponentform

$$\begin{bmatrix} \tilde{x}_1(t) \\ \dots \\ \tilde{x}_n(t) \end{bmatrix} = \begin{bmatrix} x_1(t+1) \\ \dots \\ x_n(t+1) \end{bmatrix} - \begin{bmatrix} \hat{a}_{11}x_1(t) + \dots & + \hat{a}_{1n}x_n(t) \\ \dots & \dots \\ \hat{a}_{n1}x_1(t) + \dots & + \hat{a}_{nn}x_n(t) \end{bmatrix} - \begin{bmatrix} \hat{b}_1 u(t) \\ \dots \\ \hat{b}_n u(t) \end{bmatrix} \quad (2.6)$$

Det bör påpekas att om tillräckligt många mätpunkter finns tillgängliga, fås ett ekvationssystem ur vilket \hat{a}_{ij} resp \hat{b}_i i princip kan lösas.

Dessa ekvationer utgör då normalekvationer enl. minsta kvadrat-metoden.

Ovan nämnda metod har använts vid analys av flygtest-data men leder till omfattande numeriska beräkningar varför den ej är så lämpad för flygsammanhang.

Därför behandlas i fortsättningen "steepest descent" metoden.

Sålunda gäller

$$\begin{aligned} \hat{a}_{ij}(t+1) &= \hat{a}_{ij}(t) + k \tilde{x}_i(t) x_j(t) \\ \hat{b}_i(t+1) &= \hat{b}_i(t) + k \tilde{x}_i(t) u(t) \end{aligned} \quad (2.7)$$

Hur bör k väljas?

Följande resonemang kan föras (Beckenbach [1])

Riktningen i vilken förändringen av uppskattningarna \hat{a}_{ij} resp \hat{b}_i bör ske, är känd nämligen $\tilde{x}_i(t) x_j(t)$ resp $\tilde{x}_i(t) u(t)$.

Men hur långt bör man gå?

Felen $\tilde{x}(t)$ beror på felaktiga uppskattningar $\hat{A}(t)$ och $\hat{B}(t)$.

Därför kunde, om uppskattningen valts bättre, felet varit noll.

Följande härledning inskränker sig ej till system med enbart en insignal utan behandlar det allmännare fallet med flera insignaler. Identifieringsmetoden kan tillämpas även på dylika system, förutsatt att insignalerna är oberoende.

Sålunda betrakta ekvationssystemet (2.6) radvis

$$\begin{aligned} \tilde{x}_i &= x_i(t+1) - \hat{A}_i(t+1) x(t) - \hat{B}_i(t+1) u(t) \quad (2.7) \\ &= x_i(t+1) - \left[\hat{A}_i(t) + k\tilde{x}_i x^T \right] x - \left[\hat{B}_i(t) + k\tilde{x}_i u^T \right] u \quad (2.2) \\ &= \tilde{x}_i(t) - k\tilde{x}_i(t) \left[x^T x + u^T u \right] = \tilde{x}_i(t) \left[1 - k(x^T x + u^T u) \right] \\ &\quad \neq 0 \end{aligned}$$

Härav inses att om k väljes enligt nedan

$$k = 1 / (x^T x + u^T u) \quad (2.8)$$

fås en uppskattning av A och B, som gör felet till noll.

Det återstår att visa att detta val garanterar att normen

$$\left\| \hat{A}_i(t) - A_i(t) \right\| + \left\| \hat{B}_i(t) - B_i(t) \right\| \rightarrow 0 \quad i=1,2,\dots,n$$

n betecknar systemordningen

$$\begin{aligned} &\left| (\hat{A}_i(t+1) - A_i)^2 + (\hat{B}_i(t+1) - B_i)^2 \right| \stackrel{2.7}{=} \\ &(\hat{A}_i(t) + k\tilde{x}_i x^T - A_i)^2 + (\hat{B}_i(t) + k\tilde{x}_i u^T - B_i)^2 = \\ &(\hat{A}_i(t) - A_i)^2 + (\hat{B}_i(t) - B_i)^2 + k^2 \tilde{x}_i^2 [x^T x + u^T u] + \\ &2k \tilde{x}_i \left[(\hat{A}_i(t) - A_i)x + (\hat{B}_i(t) - B_i)u \right] \\ &\quad - \tilde{x}_i(t) \end{aligned}$$

Sålunda fås med hjälp av (2.8) följande

$$\begin{aligned} (\hat{A}_i(t+1) - A_i)^2 + (\hat{B}_i(t+1) - B_i)^2 &= (\hat{A}_i(t) - A_i)^2 + (\hat{B}_i(t+1) - B_i)^2 - \\ &- \tilde{x}_i^2 / (x^T x + u^T u) \end{aligned}$$

Den sista termen är alltid positiv och därför gäller

$$\hat{A}_i \rightarrow A_i$$

$$\hat{B}_i \rightarrow B_i$$

Ett annat resonemang för att förklara metoden är följande

Låt systemet vara

$$x_i(t+1) = \sum_j a_{ij} x_j \quad i = 1, 2, \dots, n \quad n \text{ är systemordn.} \quad (2.9)$$

Ett kompaktare skrivsätt är

$$x_i(t+1) = A_i x$$

Man ser att (2.9), för varje värde på i , beskriver ett plan i den n -dimensionella rymden A_i .

Problemet är att finna ett \hat{A}_i , som satisfierar ovanstående.

Observera att enl. def. är x normalriktning till detta plan.

Metoden innebär, att man väljer ett startvärde på $\hat{A}_i(t)$.

En ny punkt $\hat{A}_i(t+1)$ erhålles genom att projicera den gamla vinkelrätt mot planet (2.9). $\hat{A}_i(t+1)$ kommer således att satisfiera (2.9)

I ekvationsform ser det ut enl. följande

$$\begin{cases} \hat{A}_i(t+1) = \hat{A}_i(t) + c x^T & (x \text{ är normalriktn.}) \\ x_i(t+1) = \hat{A}_i(t+1) x \end{cases} \quad (2.10)$$

$$\implies c = (x_i(t+1) - \hat{A}_i(t) x) / (x^T x)$$

Då det gäller att $x_i(t+1) - \hat{A}_i(t) x = \tilde{x}_i(t)$ erhålles

$$c = \tilde{x}_i(t) / x^T x$$

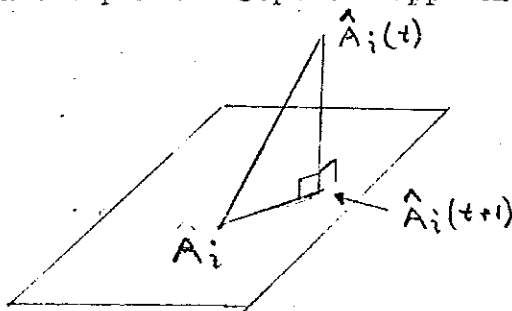
Således gäller

$$\hat{A}_i(t+1) = \hat{A}_i(t) + \tilde{x}_i(t) x / (x^T x)$$

vilket överensstämmer med det tidigare funna resultatet (2.7 - 2.8).

Den nya punkten $\hat{A}_i(t+1)$ projiceras i sin tur på det nya plan, som fås med nya mätsignaler x_j . Efter ett antal iterationer kommer man allt närmre det riktiga värdet A_i .

Att det nya värdet $\hat{A}_i(t+1)$ ligger närmre A_i än $\hat{A}_i(t)$ inses, ty dessa tre punkter "spänner upp" en rätvinklig triangel.



För speciella modeller kan något snabbare identifiering fås om ett minsta värde på k införes, dvs då $k_{opt} < k_{min}$ sättes $k = k_{min}$ förutsatt att stabilitetskriterierna uppfylles. Nedan följer en härledning av dessa.

Enl. ekv (2.2) gäller

$$\tilde{x}(t) = x(t+1) - \hat{A} x(t) - \hat{B} u(t) \quad (2.2)$$

Ett alternativt skrivsätt är

$$\tilde{x}_i(t) = x_i(t+1) - \sum_j \hat{a}_{ij}(t) x_j(t) - \hat{b}_i(t) u(t) \quad i = 1, 2, \dots, n$$

Tidigare infördes

$$\hat{a}_{ij}(t+1) = \hat{a}_{ij}(t) + k \tilde{x}_i(t) x_j(t) \quad \text{samt } k = 1/(x^T x + u^T u)$$

Detta tillsammans med (2.2) medför

$$\hat{a}_{ij}(t+1) = \hat{a}_{ij}(t) + k x_j(t) \left[x_i(t+1) - \sum_k \hat{a}_{ik}(t) x_k(t) - \hat{b}_i(t) u(t) \right]$$

Inför skiftoperatören q . Då fås

$$(1 + q^{-1}(k x_j^2(t) - 1)) \hat{a}_{ij}(t+1) = k x_j(t) \left[x_i(t+1) - \sum_{k \neq j} \hat{a}_{ik}(t) x_k(t) - \hat{b}_i(t) u(t) \right]$$

$$\hat{a}_{ij}(t+1) = \frac{\hat{a}_{ij}(t)}{1 + q^{-1}(k x_j^2(t) - 1)}$$

För att polen till detta uttryck skall hamna innanför enhetscirkeln kräves att $0 < k x_j^2 < 2$

$$k < 2 / (x_j^2) \quad j = 1, 2, \dots, n$$

Sammanfattning av ekvationer

$$\text{Givet system} \quad x(t+1) = Ax(t) + Bu(t) \quad (2.1)$$

Sök uppskattningar av A och B genom att

1. Sätt startvärden på \hat{A} och \hat{B} .
2. Förbättra dessa genom att (komponentvis) beräkna

$$\hat{a}_{ij}(t+1) = \hat{a}_{ij}(t) + k \tilde{x}_i(t) x_j(t) \quad i=1, \dots, n$$

$$\hat{b}_i(t+1) = \hat{b}_i(t) + k \tilde{x}_i(t) u(t) \quad j=1, \dots, n$$

$$\text{där } \tilde{x}_i(t) = x_i(t+1) - \sum_j \hat{a}_{ij}(t) x_j(t) - \hat{b}_i(t) u \quad (2.2)$$

$$\text{samt } k = 1 / \left(\sum_j x_j^2 + u^2 \right) \quad (2.8)$$

3. Testexempel

3.1 Icke återkopplat system.

För att testa den i föregående kapitel beskrivna identifieringsmetoden, simulerades ett 2:a ordn. linjärt dynamiskt system

$$x(t+1) = A(t) \cdot x(t) + B(t) \cdot u(t)$$

där $A(t)$ och $B(t)$ gavs värden enligt vad som senare följer.

Gemensamt för exempel 1, 2, 3, och 4 är följande:

$$A(t) = \begin{bmatrix} 1.5 & 1 \\ a_{21} & 0 \end{bmatrix} \quad B(t) = \begin{bmatrix} b_1 \\ 1 \end{bmatrix}$$

$$\hat{A}(0) = 0$$

$$\hat{B}(0) = 0$$

Insignalen u är en PRBS signal (Pseudo Random Binary Sequence)

Simuleringen utfördes under 300 samplingintervall.

Den tidigare omtalade (kap. 2) minsta estimatorförstärkningen sattes till 0.04 . Detta värde befanns för ett flertal simuleringar ge något snabbare konvergens.

Ex. 1

Samtliga parametrar konstanta.

$$a_{21} = -0.75 \quad ; \quad b_1 = 0.5$$

I detta exempel redovisas följande diagram: \hat{a}_{12} , \hat{a}_{21} , \hat{b}_1 , \hat{b}_2 ,

u , x_1 , x_2 .

Dessutom finns i diagramform förstärkningsfaktorn i ekv. 2.7

$$(k = 1 / (x^T x + u^T u)) \text{ med. } [\text{GAIN}]$$

Samtliga parametrar svänger snabbt in sig mot de riktiga värdena.

Ex. 2.

I denna simulering varieras 2 parametrar.

$$-0.5 > a_{21} > -1$$

$$b_1 = 0.5 (1 + \sin(0.04 \cdot I)) \text{ där } I = 1, 2, 3, \dots \dots 301$$

Detta motsvarar en variation av ca 4π för b_1 under simuleringen.

Observera att parametrarna identifieras radvis oberoende.

Följande diagram bifogas: \hat{a}_{12} , \hat{a}_{21} , \hat{b}_1 , \hat{b}_2

Även här svänger parametrarna snabbt in sig mot de riktiga värdena.

\hat{b}_1 följer variationen hos b_1 acceptabelt.

Ex. 3.

Som föregående exempel men med brus $N(0, 0.1)$ pålagt tillståndsvariablerna före identifieringen.

Följande diagram bifogas: \hat{a}_{12} , \hat{a}_{21} , \hat{b}_1 , \hat{b}_2

Jämför dessa diagram med motsvarande i exempel 2. Bruset gör att de identifierade parametrarna nu genomgående får ett "taggigare" utseende. Samtliga parametrar har identifierats något sämre.

Detta exempel verifierar att identifieringsmetoden är bruskänslig. Se även nästa exempel.

Ex. 4.

Samma som exempel 2 och 3 men med brus $N(0, 0.3)$ pålagt tillståndsvariablerna.

Följande diagram bifogas: \hat{a}_{12} , \hat{a}_{21} , \hat{b}_1 , \hat{b}_2

Parametrarna identifieras här väsentligen sämre.

För det i dessa fyra exempel använda systemet gäller att det är stabilt om

$$-0.5 > a_{21} > -1$$

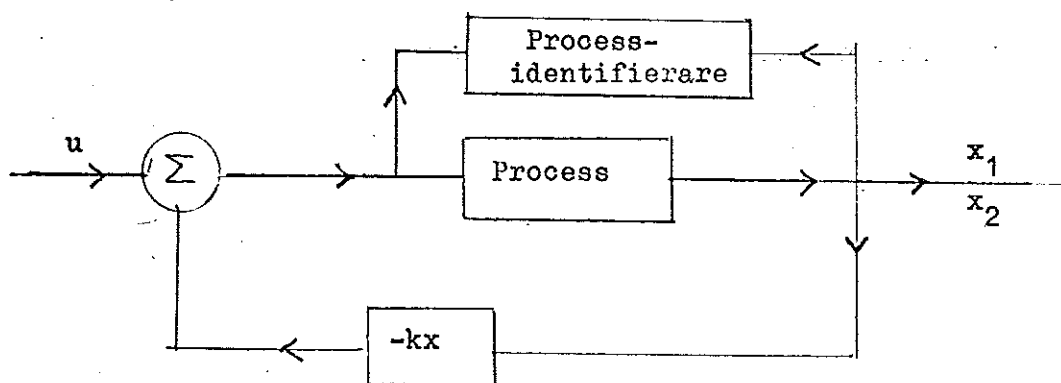
Då $B = \begin{bmatrix} 0.5 \\ 1 \end{bmatrix}$ inses att systemet är av icke -min-fas typ.

Detta spelar för själva identifieringsproblemet mindre roll.

I alla dessa simuleringar utom ex. 4 (brus $N(0,0.3)$) tycks de estimerade parametrarna svänga in sig mot de riktiga värdena. I de fall då dessa parametrar varierar följer de uppskattade acceptabelt efter. Dock konstaterades att en snabbare variation hos b_1 var svår att följa. Anledningen härtill omtalas senare.

3.2 Test av identifieringsmetod på återkopplat system.

För att utröna om de identifierade parametrarna kunde användas för att bestämma en återkoppling av det öppna systemet, gjordes ytterligare en serie simuleringar. För system där parametrarna är kända finns en metod att beräkna en återkoppling, så att det återkopplade systemet får önskade egenskaper (polerna får önskat läge). Denna metod förutsätter att alla tillståndsvariabler är mätbara,



Här görs ett försök att bestämma denna återkopplingsvektor genom att utnyttja de identifierade värdena på A .

Som modell för processen väljes

$$A(t) = \begin{bmatrix} 1.5 & 1 \\ a_{21} & 0 \end{bmatrix} \quad B(t) = \begin{bmatrix} b_1 \\ 1 \end{bmatrix}$$

$$a_{21} = -0.5 (1 + \sin(\omega t))$$

Således gäller att $0 > a_{21} > -1$ under simuleringen, vilket medför att det öppna systemet tidvis är instabilt.

Låt karakteristiska ekvationen för det önskade återkopplade systemet vara

$$q^2 - 0.4q + 0.25 = 0$$

Ex. 5

I denna simulering varieras en parameter, nämligen

$$a_{21} = -0.5 (1 + \sin(0.04 \cdot I)) \quad I = 1, 2, 3, \dots, 301$$

$$b_1 = 1.5$$

$$\hat{A}(0) = A(0) \quad ; \quad \hat{B}(0) = B(0)$$

Följande diagram bifogas: \hat{a}_{21} , \hat{b}_2 , u , x_1

Parametrarna i rad 1 i A och B berörs ej då de är konstanta och från början har de riktiga värdena.

Det konstaterades genom utskrift att rötterna till kar. ekv. för \hat{A} hela tiden låg tämligen nära de önskade lägena.

$$\text{Det återkopplade systemets kar. ekv. } q^2 + w_1 q + w_2 = 0$$

$$\text{Önskade värden } w_1 = -0.4 \quad ; \quad w_2 = 0.25$$

Med identifierade värden på A erhöles under simuleringen

$$-0.19 > w_1 > -0.56$$

$$0.17 > w_2 > 0.33$$

Det återkopplade systemet var hela tiden stabilt.

Sålunda kunde en acceptabel återkoppling beräknas då endast en parameter varierade.

Ex. 6

I detta exempel varieras 2 parametrar

$$a_{21} = -0.5 (1 + \sin(0.04 \cdot I))$$

$$b_1 = 1.5 + 0.5 \sin(0.04 \cdot I) \quad ; \quad I = 1, 2, 3, \dots, 301$$

Syftet var här samma som i föregående exempel att identifiera A och B samt därmed bestämma en återkoppling.

Följande diagram bifogas: \hat{a}_{21} , \hat{b}_1 , u, x_1

\hat{a}_{21} identifierades bra, vilket var att vänta då raderna identifieras oberoende av varandra. Däremot misslyckades försöket att identifiera \hat{b}_1 .

Förklaringen till detta är att insignalen u är väsentligen mindre än x_1 och x_2 . Detta medför att identifieringen av b_1 och b_2 blir långsammare än för parametrarna i A matrisen (jfr ekv. 2.7).

Det omvända skulle vara fallet om u vore större än x_1 och x_2 .

Skalning av tillst. variablerna skulle rätta till detta.

Den dåliga identifieringen gav som resultat en dålig återkoppling, dvs. polerna hamnade långt ifrån de lägen som önskades. Dock var det återkopplade systemet hela tiden stabilt.

Ex. 7.

För att belysa behovet av ordentlig insignalaktivitet, ökades frekvensen hos insignalen till det dubbla.

För systemet gäller

$$a_{21} = -0.5 (1 + \sin(0.04 \cdot I))$$

$$b_1 = 1.5 + 0.5 \sin(0.04 \cdot I) \quad I = 1, 2, \dots, 301$$

dvs. samma system som i föregående exempel.

Följande diagram bifogas: \hat{a}_{21} , \hat{b}_1 , u, x_1

\hat{a}_{21} identifieras som väntat bra.

\hat{b}_1 identifieras här bättre än i föregående exempel.

Systemet (återkopplade) var hela tiden stabilt.

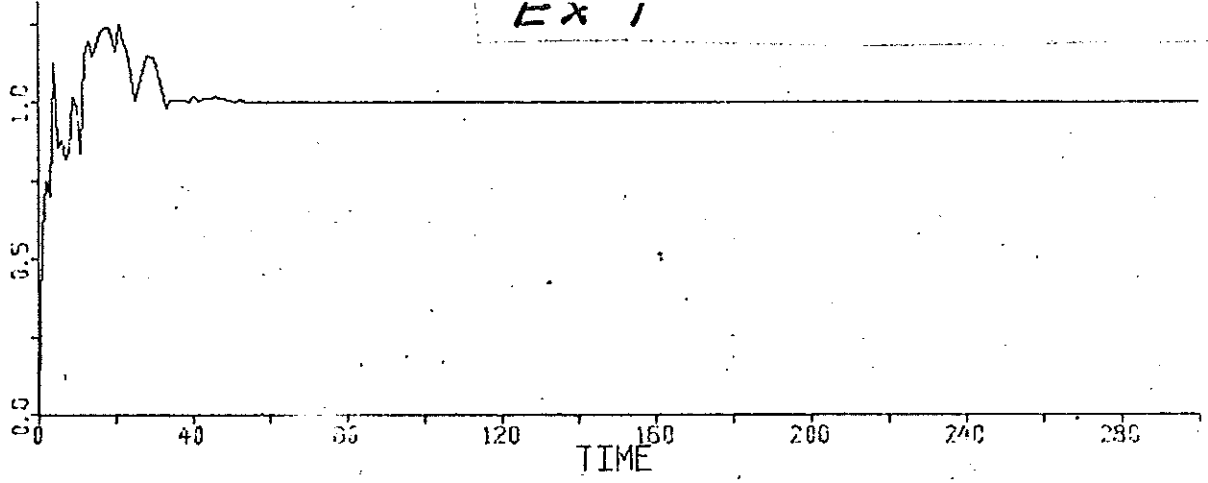
De i dessa exempel genomförda simuleringarna, tyder på att det är möjligt att identifiera parametrarna så väl, att en återkoppling går att bestämma med hjälp av dessa.

Identifieringen kan som visats i exempel 6 bli alltför långsam pga. dålig skalning. (jfr \hat{a}_{21} med \hat{b}_1)

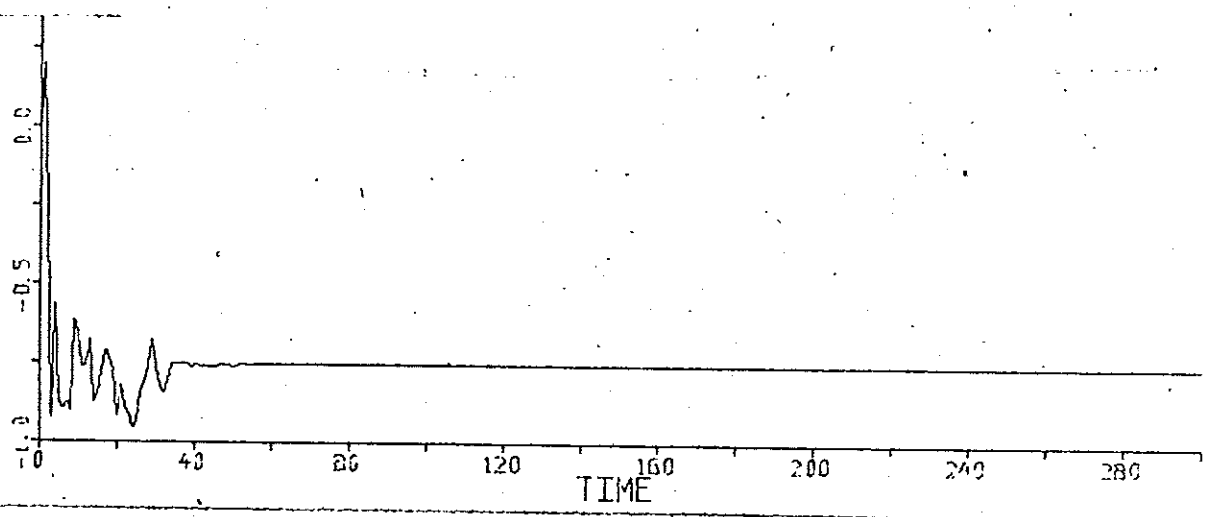
Exempel 7 visar att det är mycket väsentligt för identifieringsmetoden att ordentlig insignalaktivitet föreligger.

EX 1

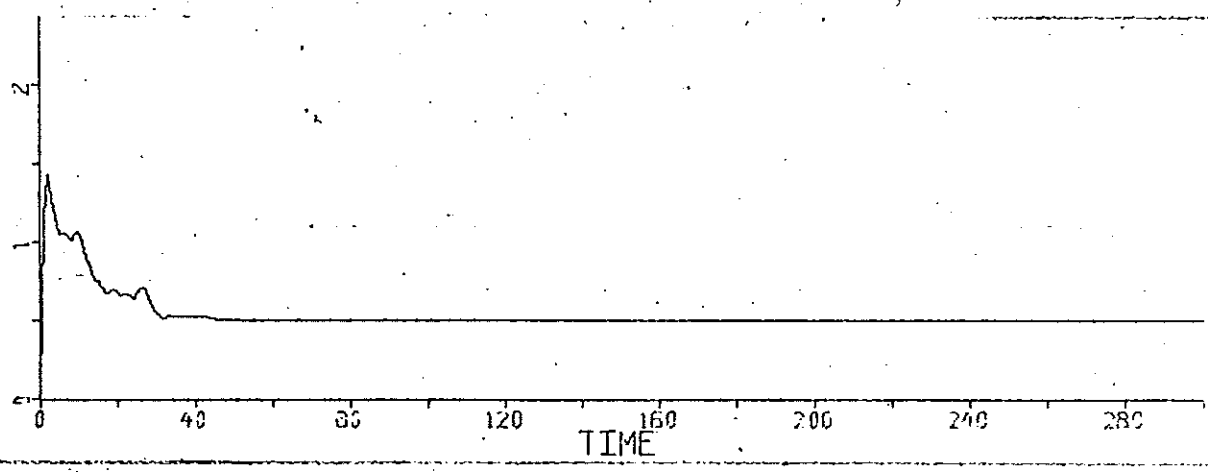
\hat{a}_{12}



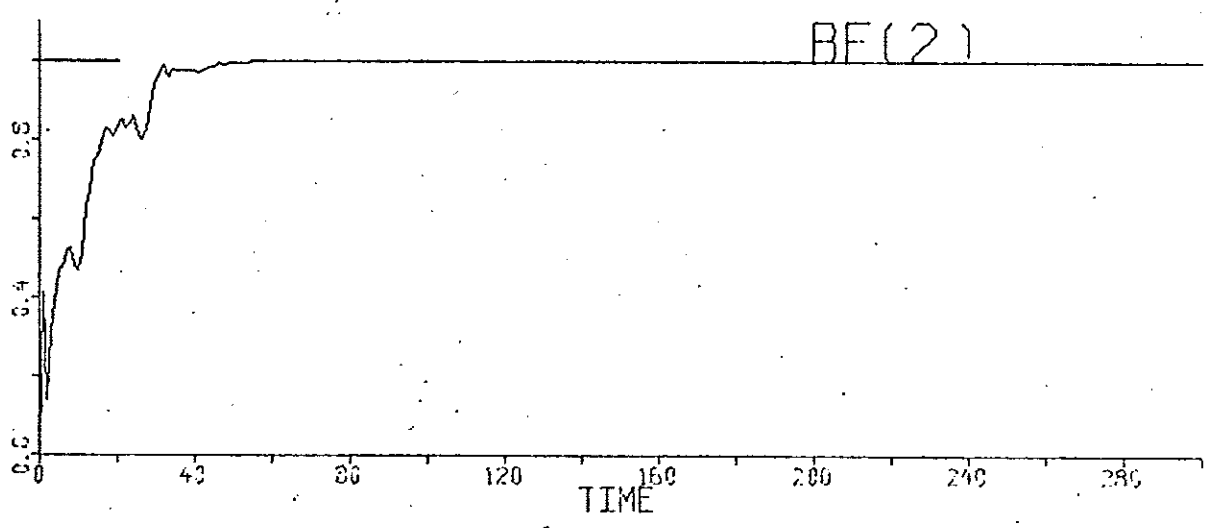
\hat{a}_{21}

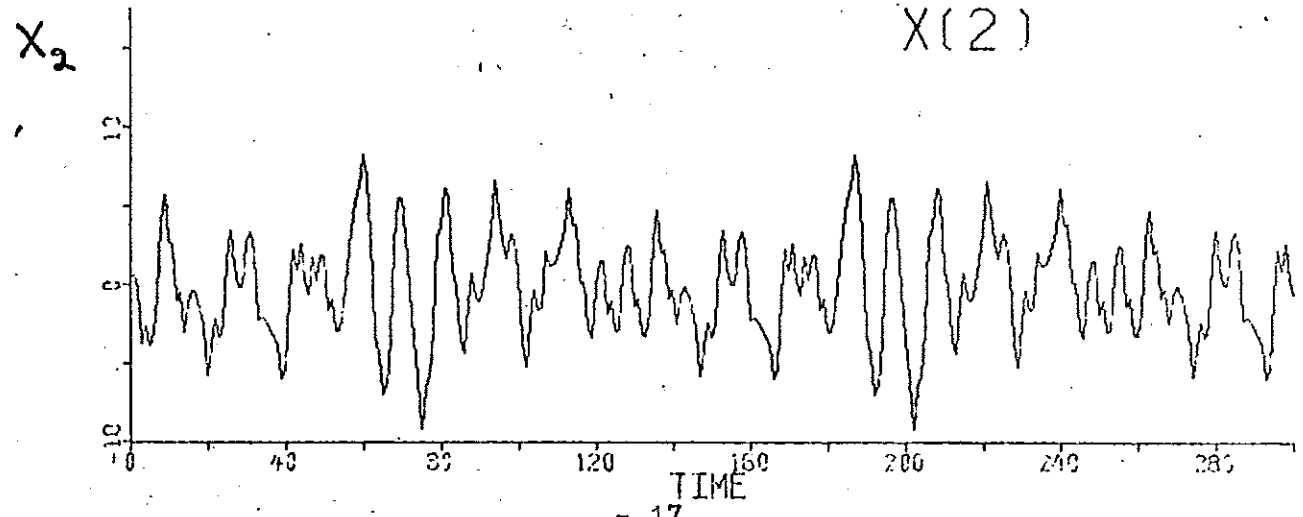
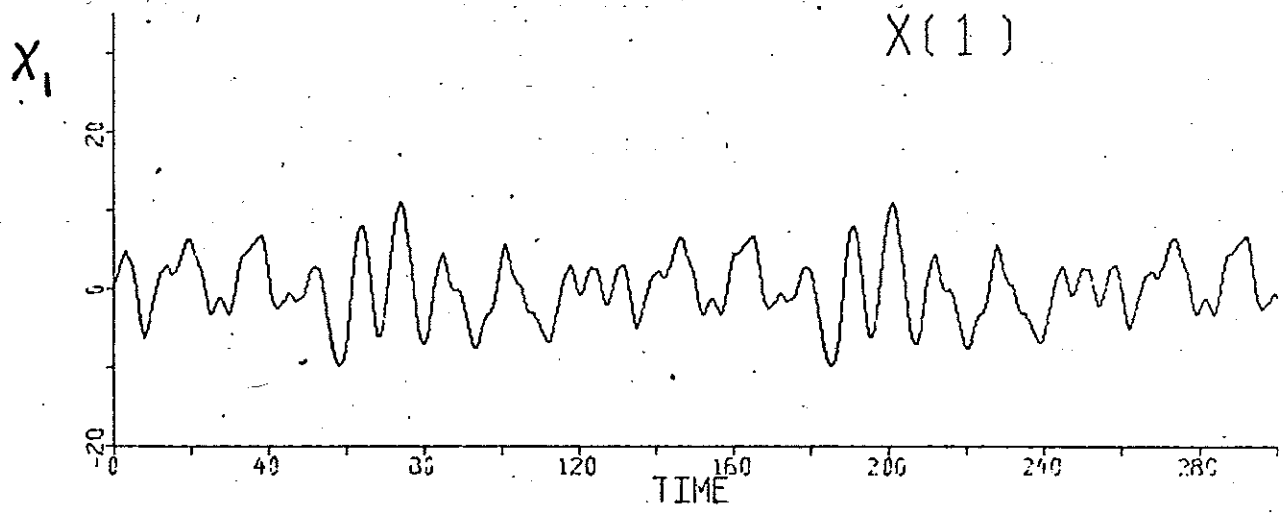
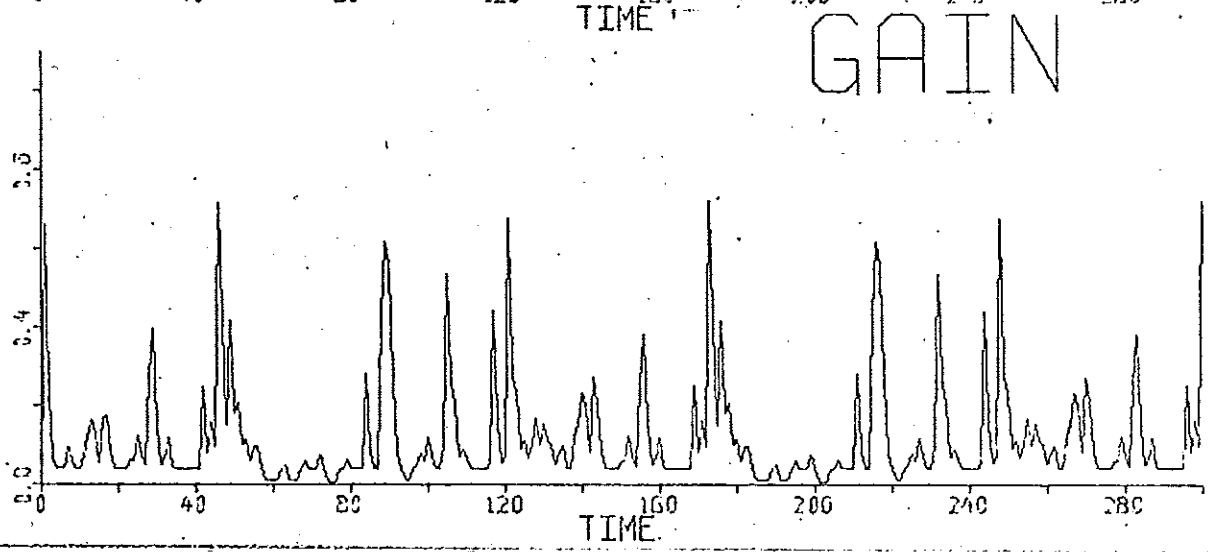
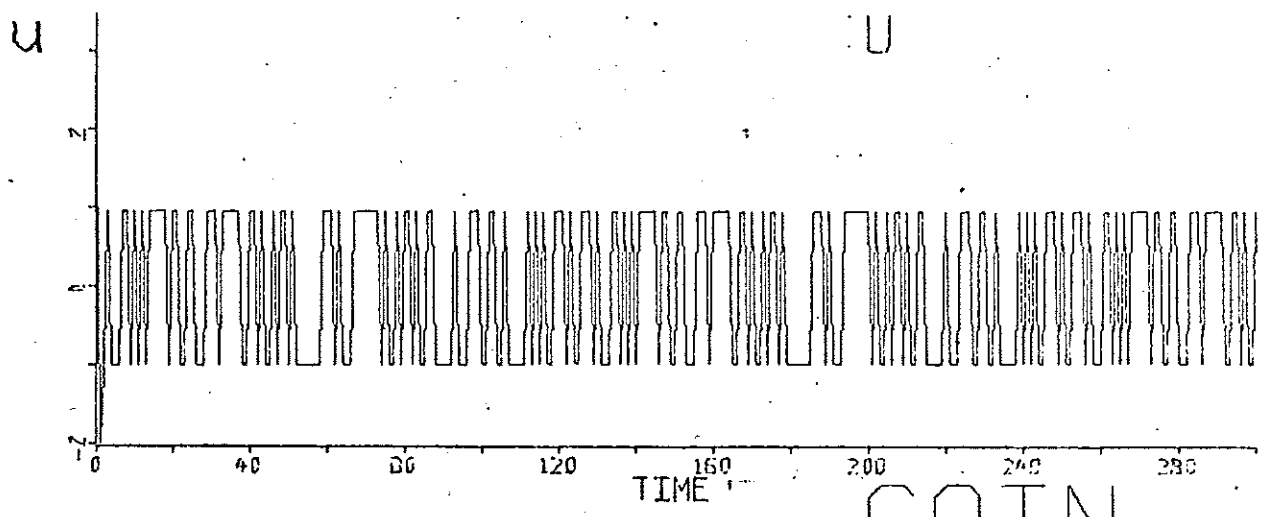


\hat{b}_1

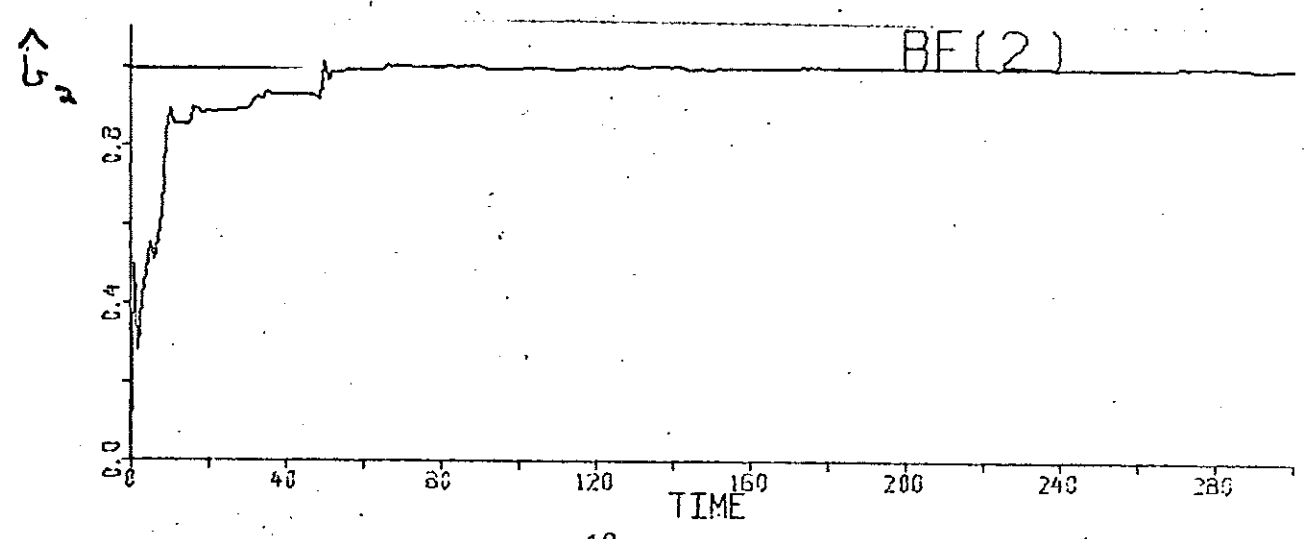
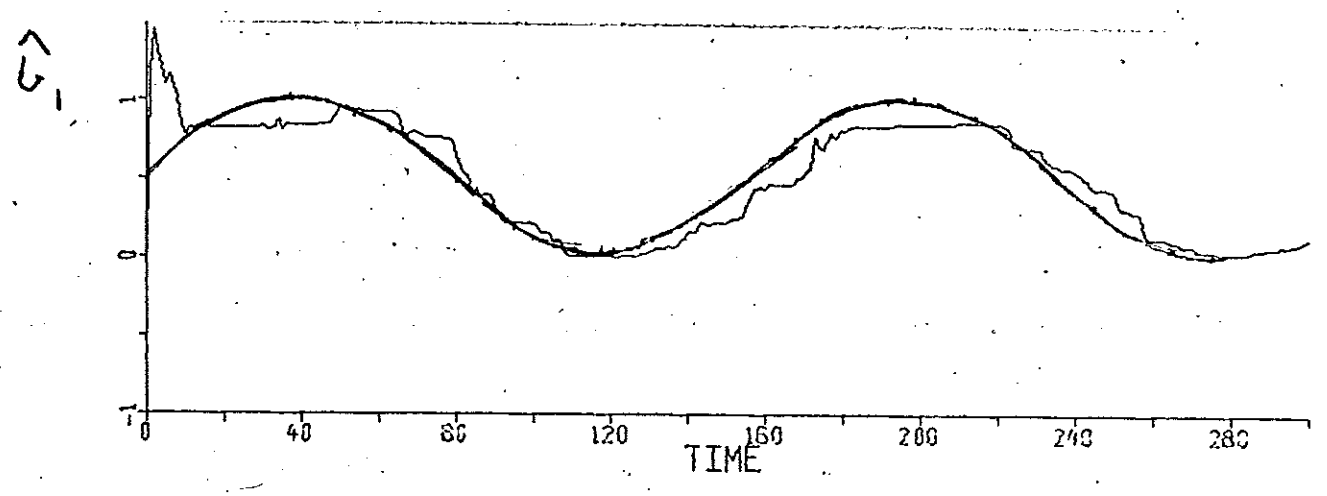
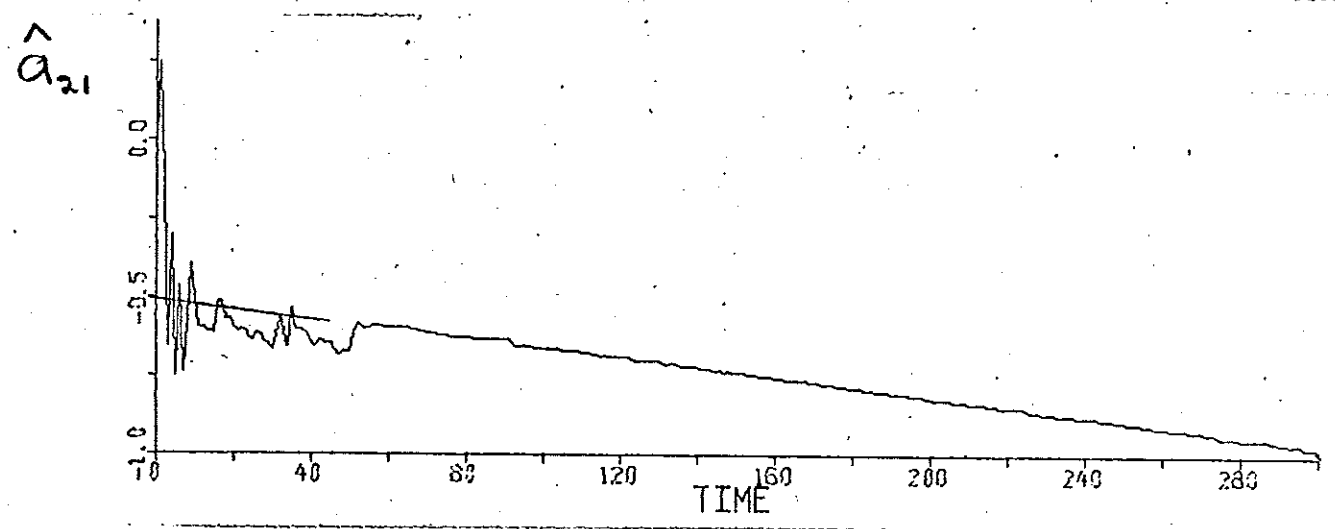
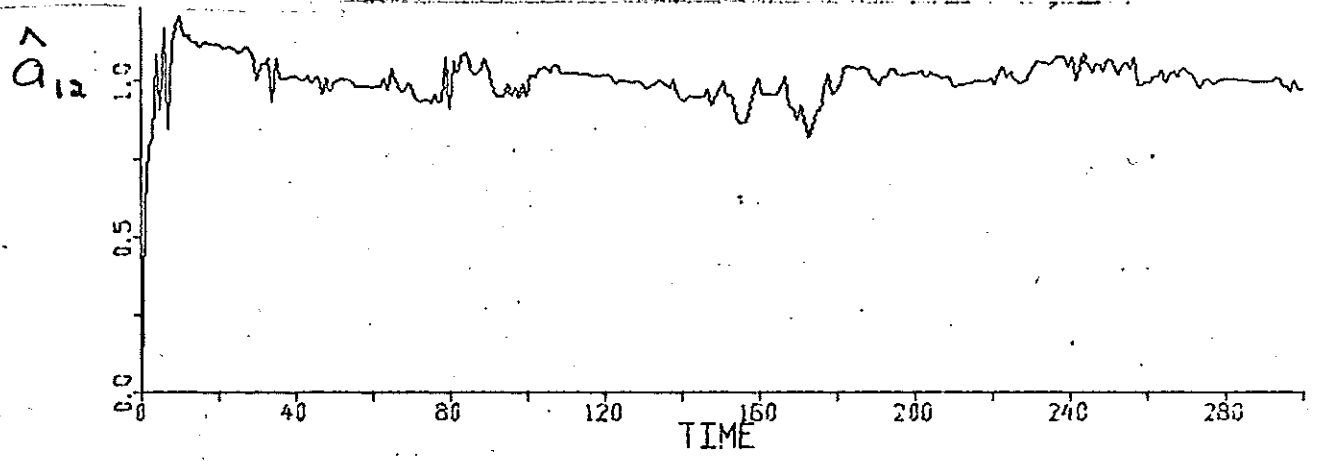


\hat{b}_2



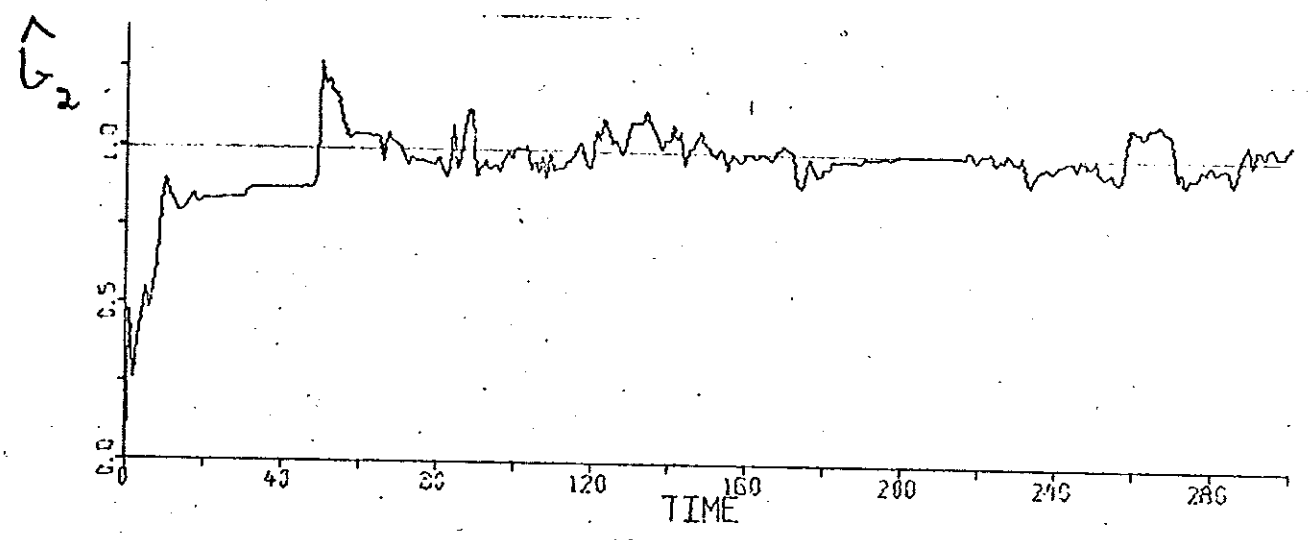
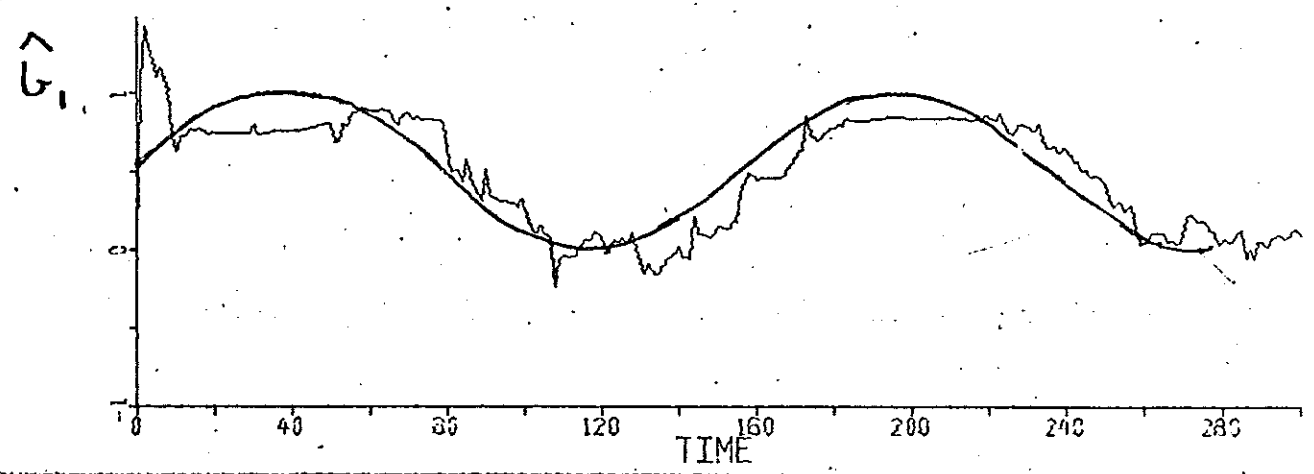
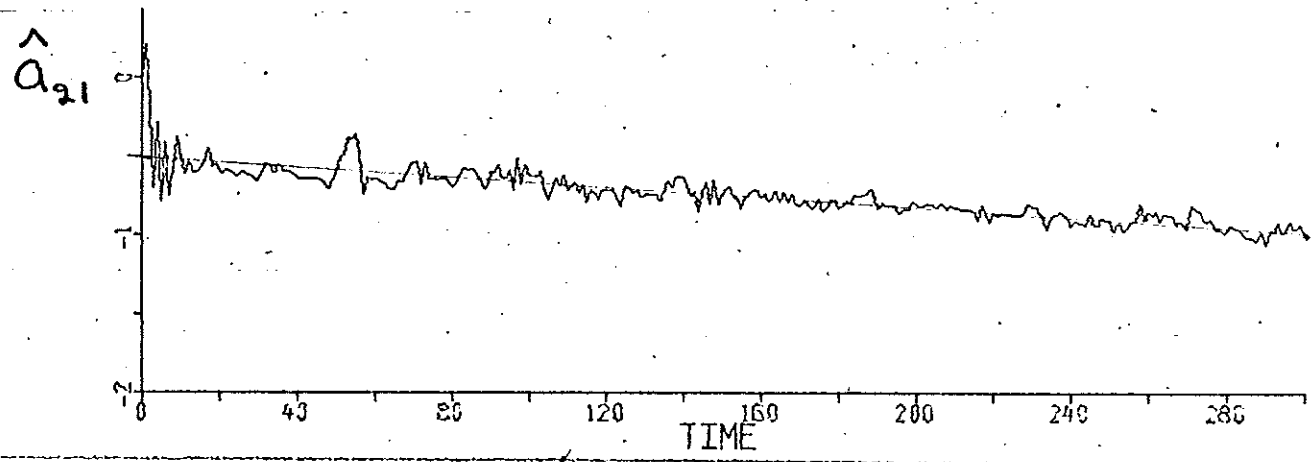
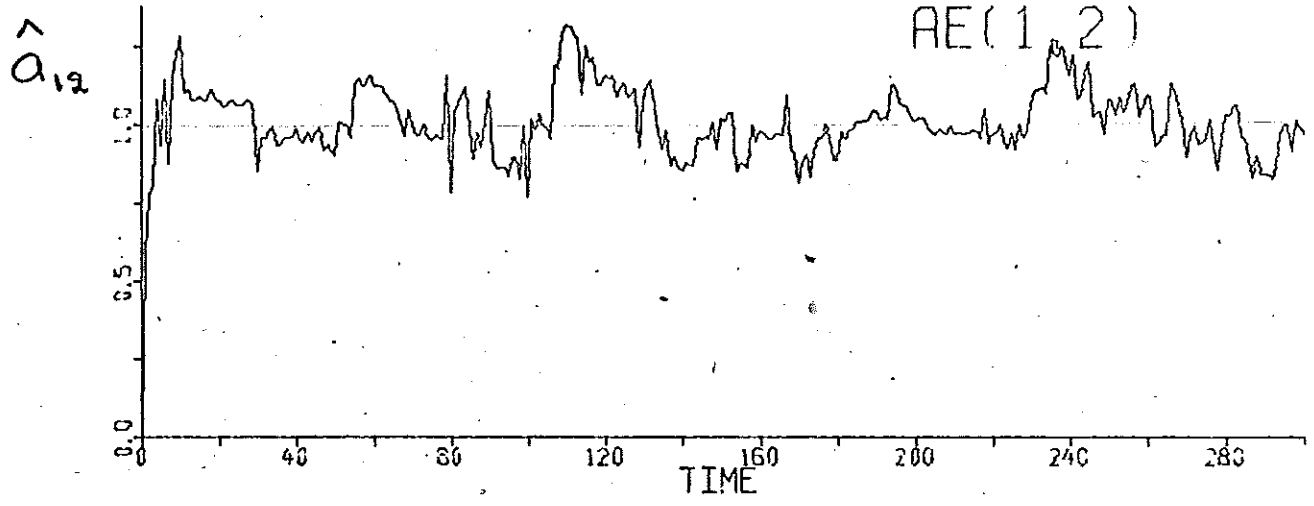


Ex. 2

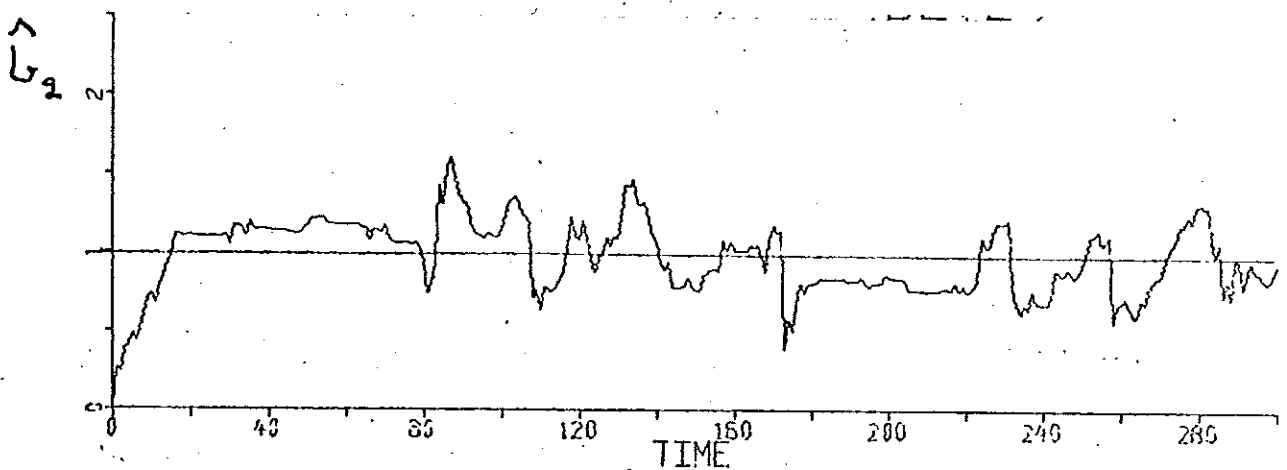
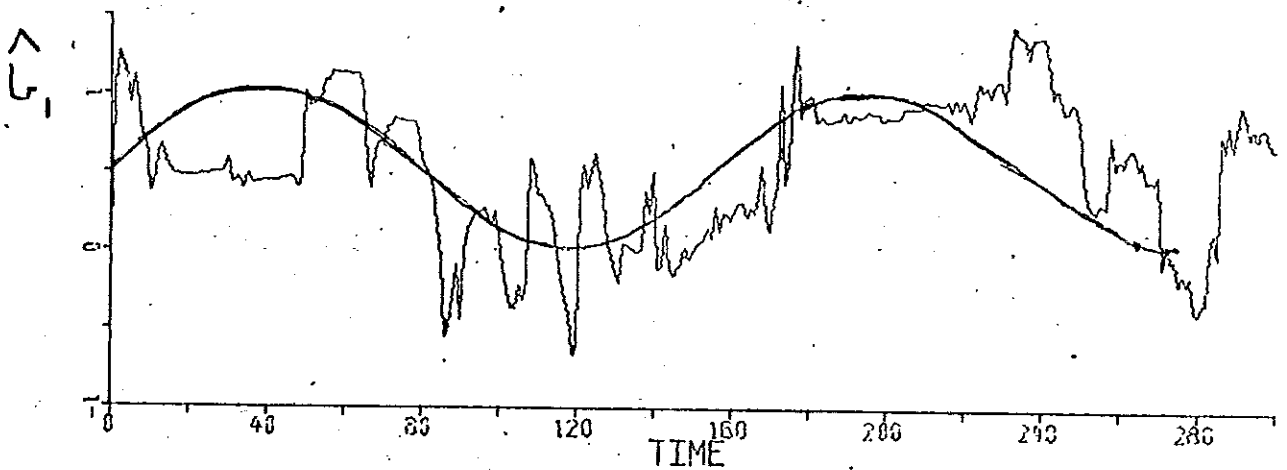
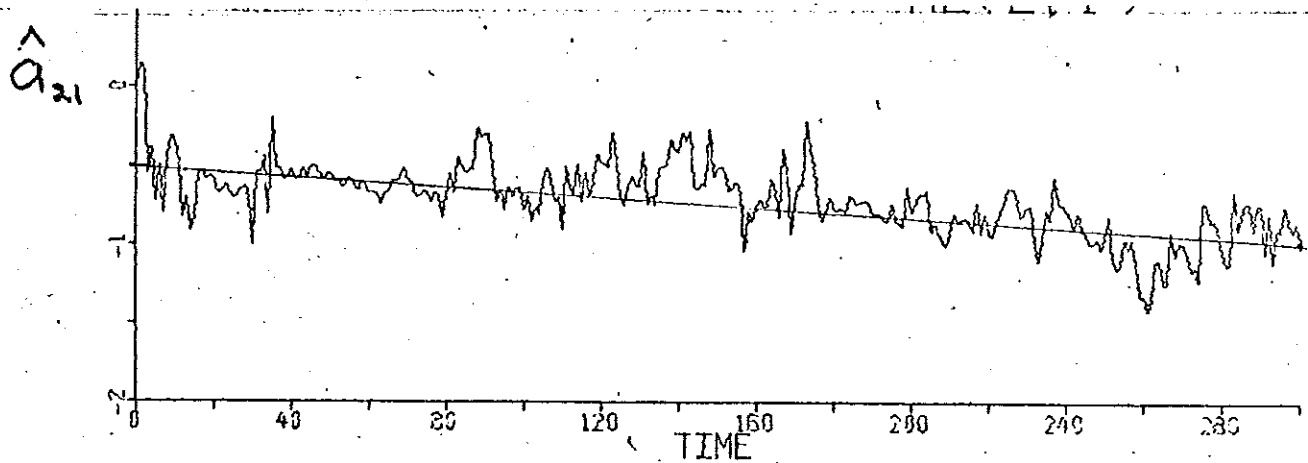
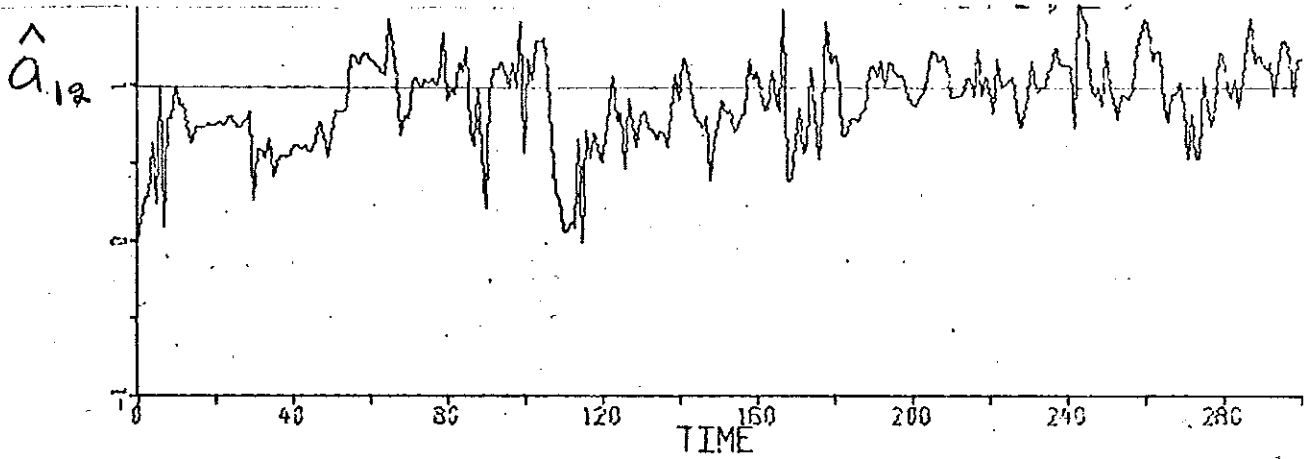


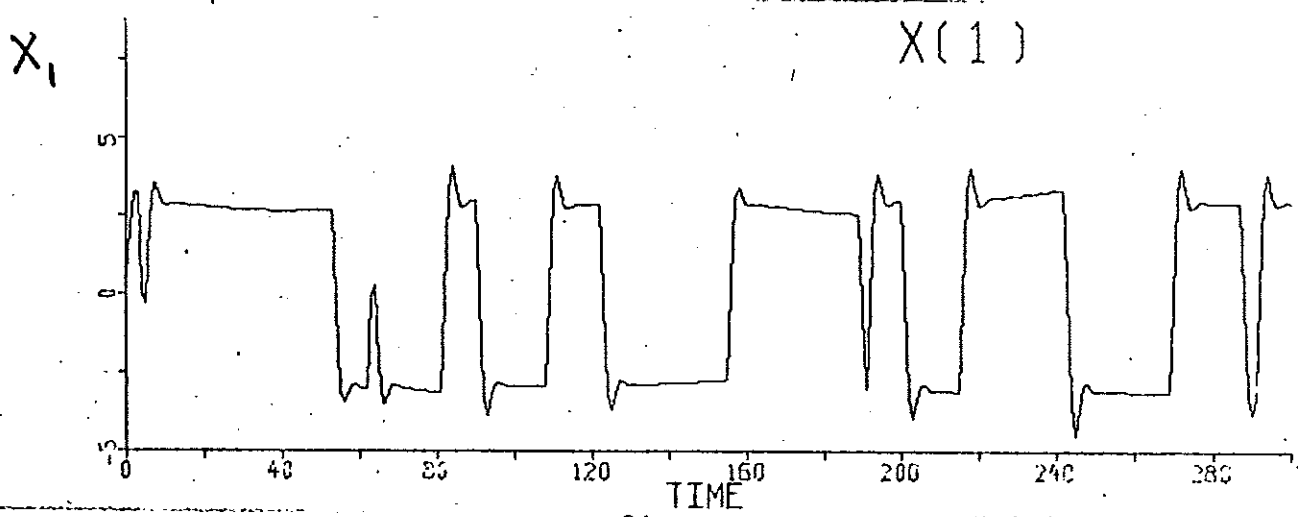
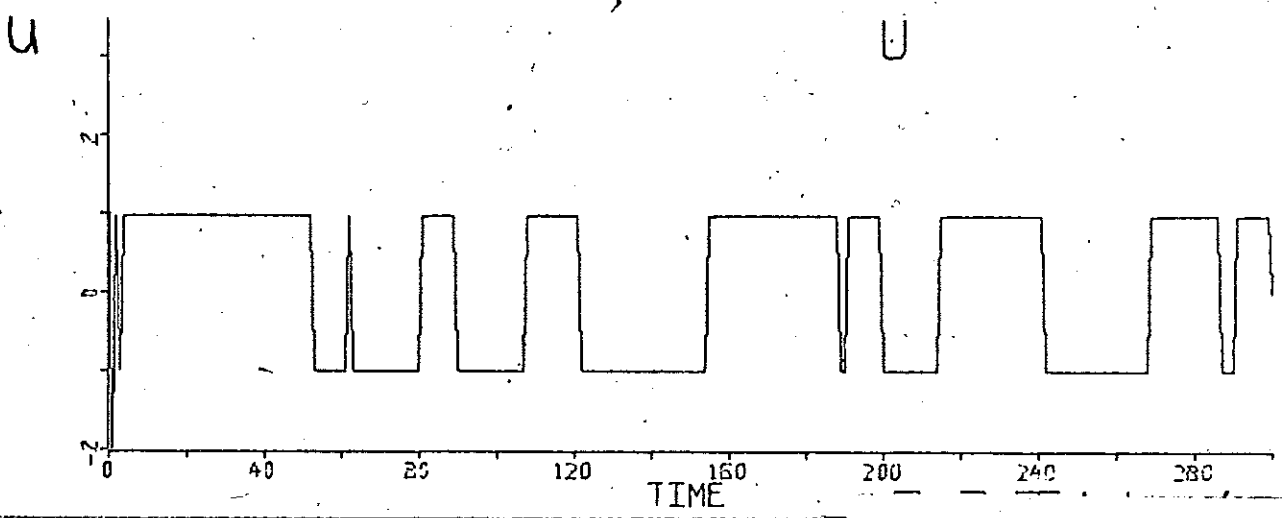
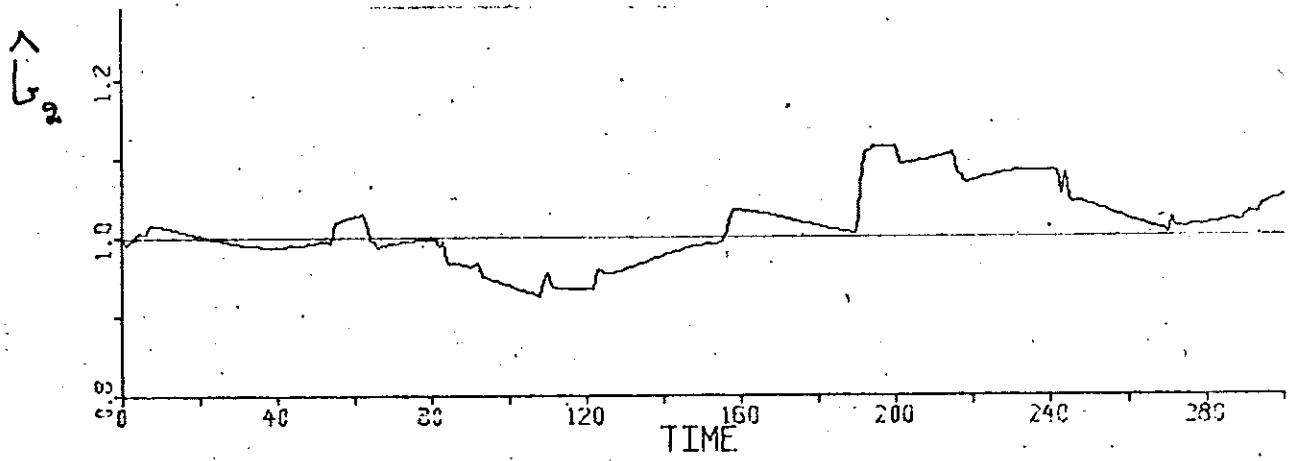
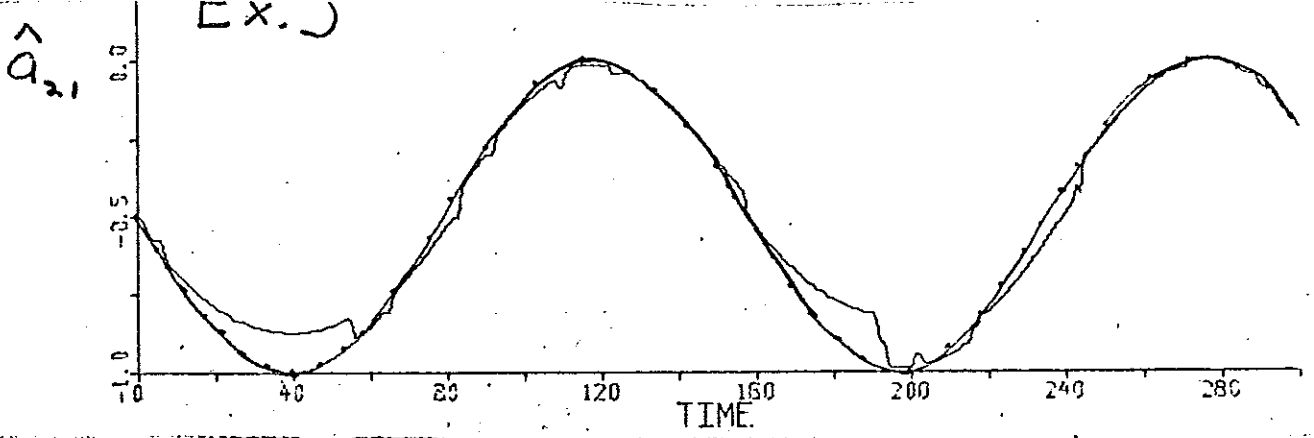
Ex 3

RE(1,2)



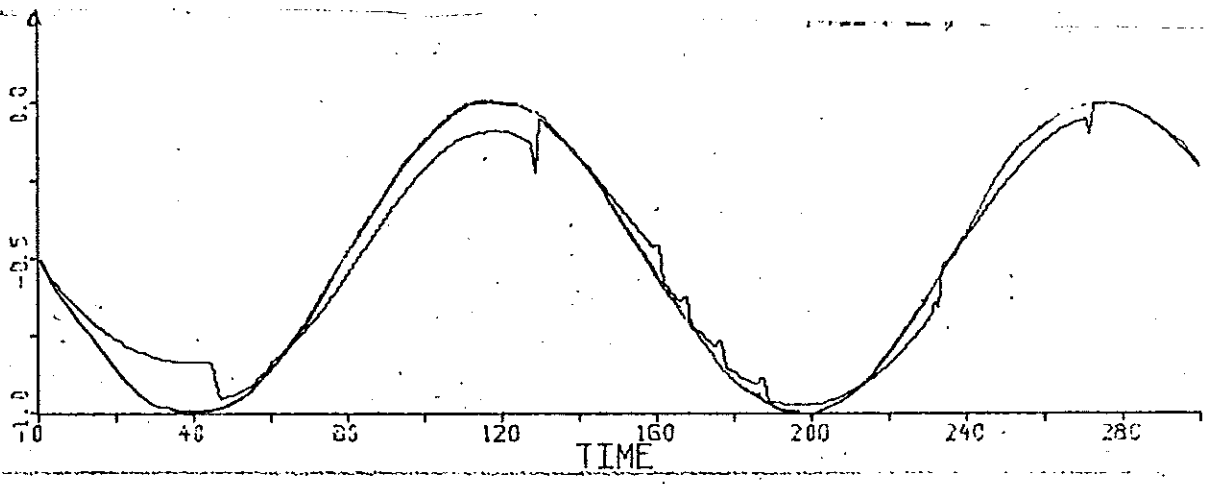
Ex. 4



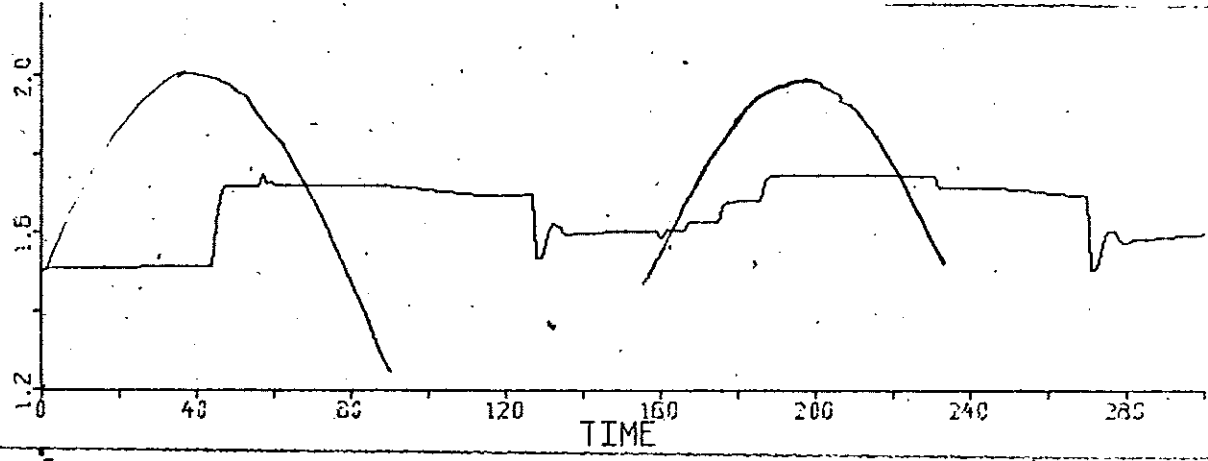


LX: 0

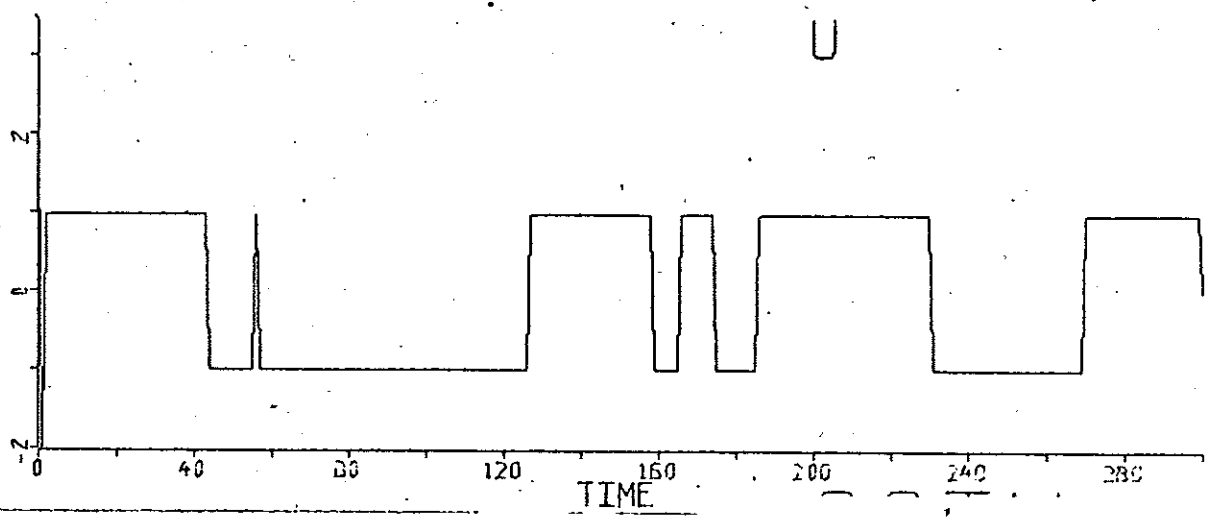
\hat{a}_{21}



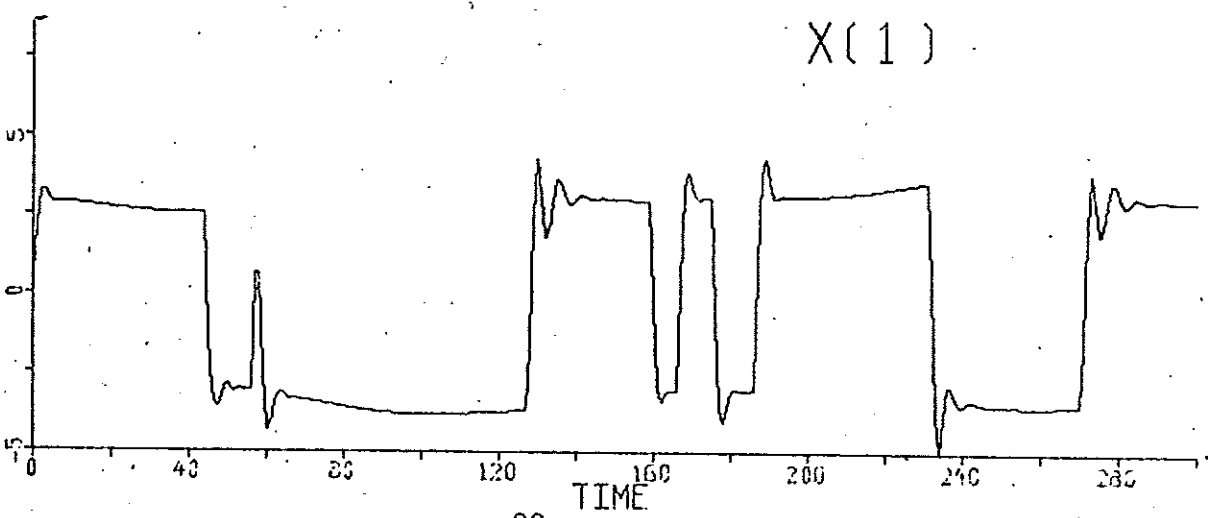
\hat{c}_1



u

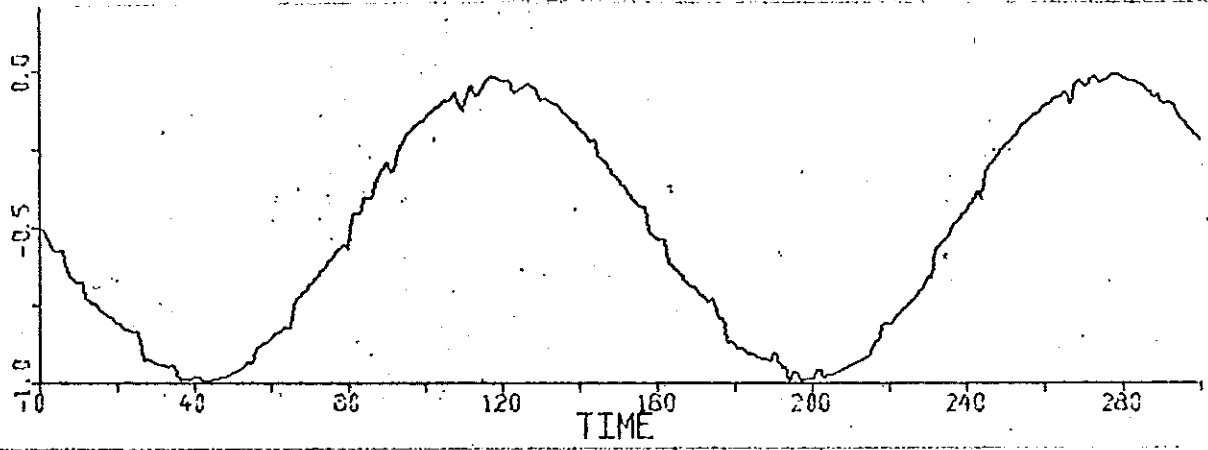


X_1

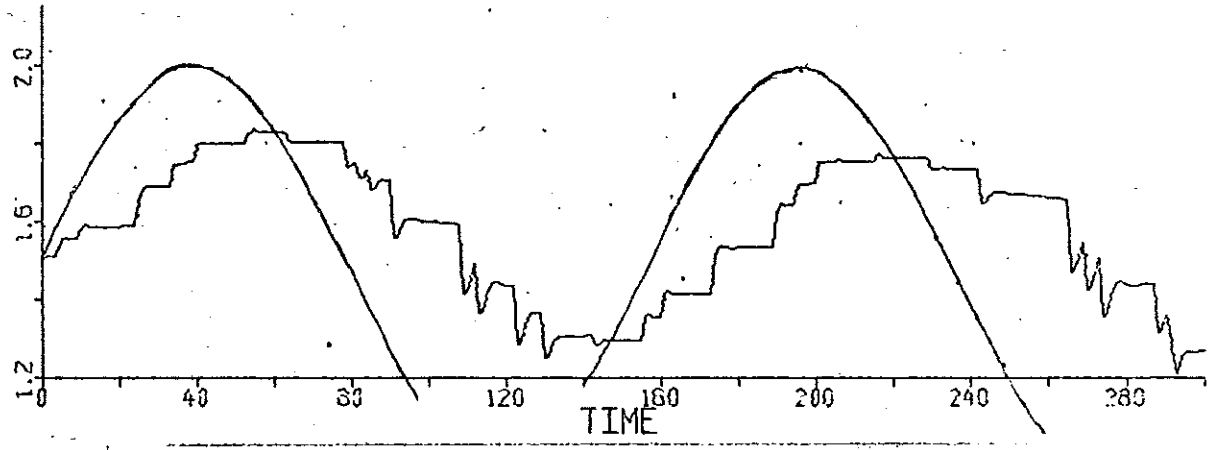


Ex. 7

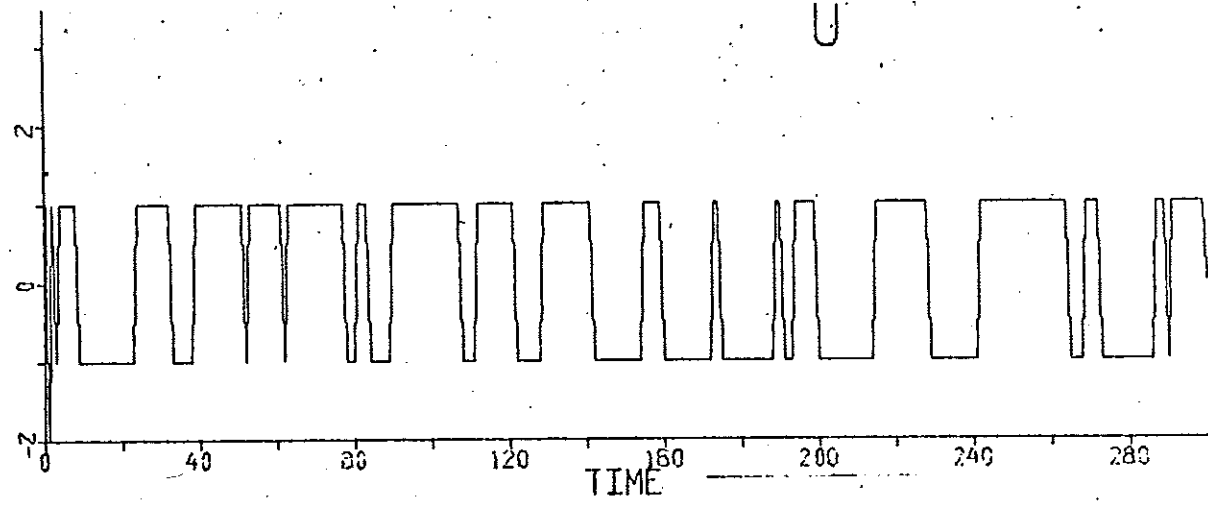
\hat{a}_{21}



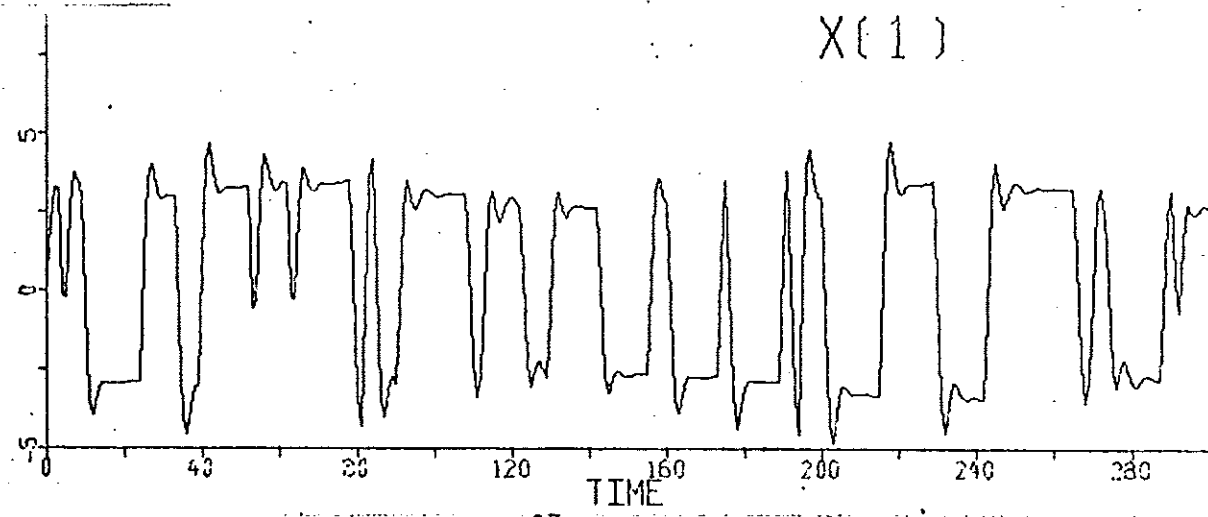
\hat{u}_1



u



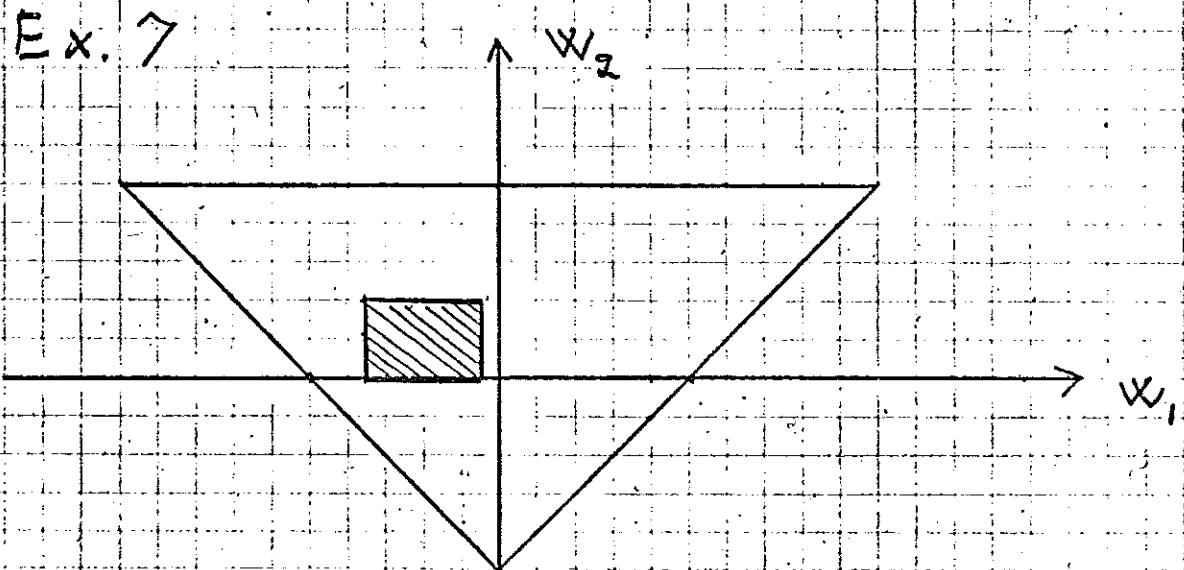
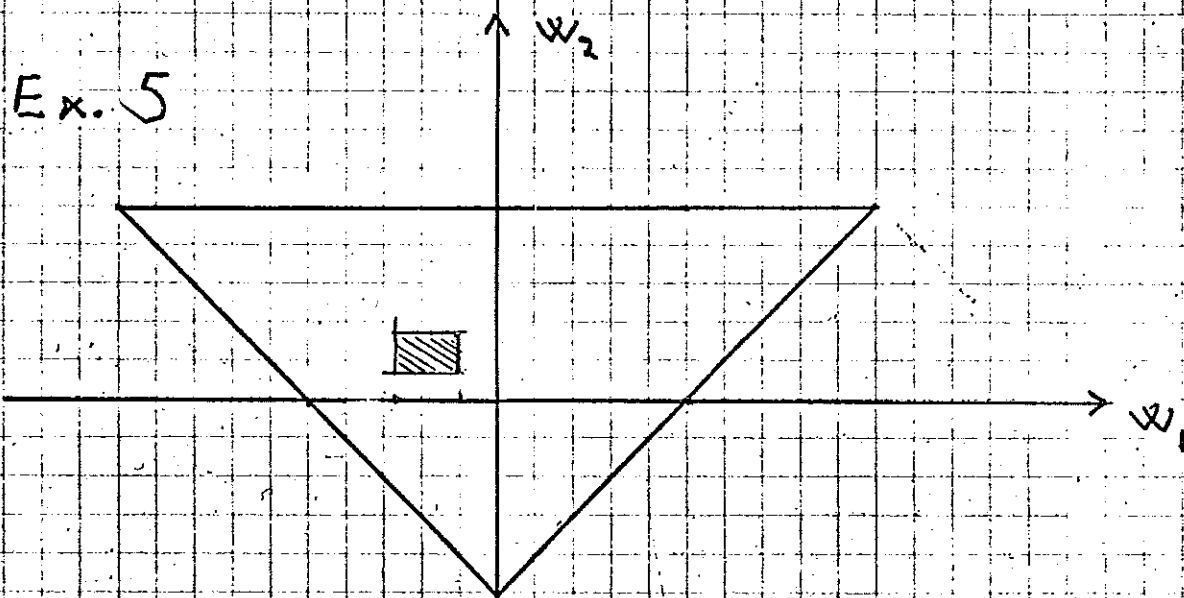
x_1



Grafisk representation av läget för karakteristiska ekvationens koefficienter. (Ex. 5 och Ex. 7)

Systemet är stabilt om $(w_1; w_2)$ innanför triangeln.

Här har max. och min. värden ritats in.
gränser

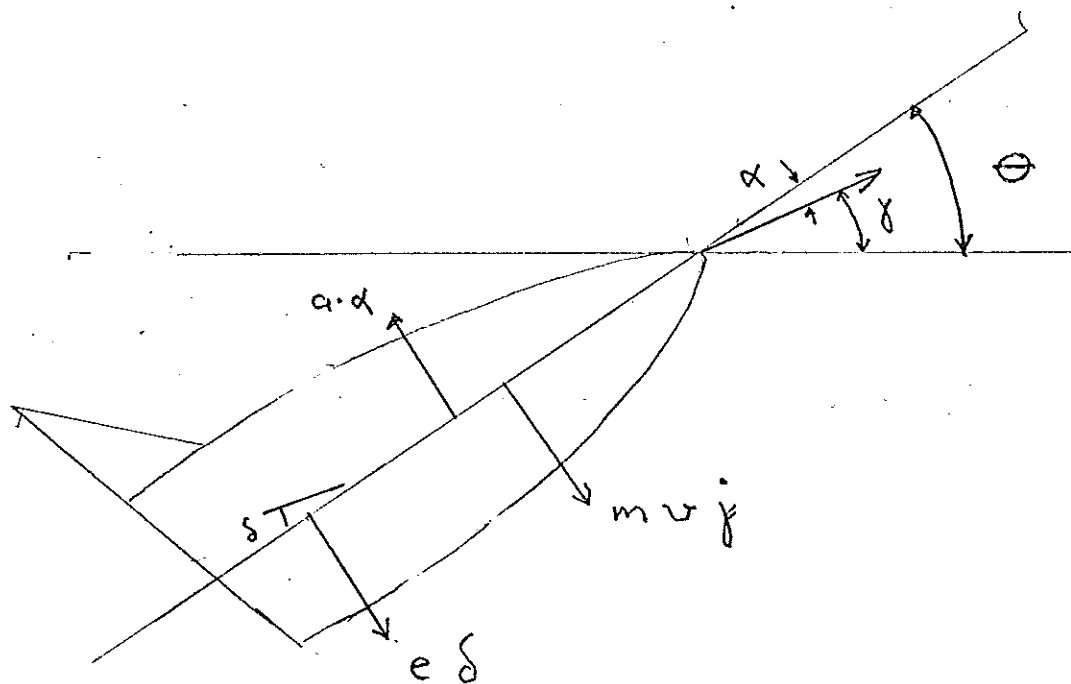


4. Beskrivning av robot-modell.

Här följer en presentation av den robot modell som användes vid simuleringarna.

Definition av vinklar

- α - anfallsvinkel
- Θ - attitydvinkel
- v - hastighetsvektor
- δ - roderutslag



För att beskriva roboten, ställes följande ekvationer upp.

$$m v \dot{\gamma} = a \left(\alpha + \frac{w}{v} \right) - e \delta \quad (\text{tvärkraft})$$

$$J \ddot{\theta} + b \dot{\theta} + c \left(\alpha + \frac{w}{v} \right) = e \delta \quad (\text{moment})$$

$$\alpha + \gamma = \theta \quad (\text{geometri})$$

De i ekvationerna ingående "konstanterna" a , b , c , och e varierar under flygningens gång.

Följande samband gäller:

$$a = p_s M^{3/2}$$

$$b = 0.005 p_s M^{1/2}$$

$$c = 0.04 (1 - 10|\alpha|) p_s M^2$$

$$e = 0.2 p_s M$$

$$m = 150 \text{ kg}$$

$$J = 60 \text{ kpm}^2$$

$$M = 1 \implies v = 300 \text{ m/s}$$

$$p_s = 10^5 e^{-H/7200} \text{ N/m}^2 \quad (\text{halvering var } 5:e \text{ km})$$

För att förenkla beräkningarna införes nedanstående approximationer:

- försumma termen $-e\delta$ i tvärkraftsekvationen
- sätt $w=0$ (vertikal turbulens)
- försumma olineäriteter i c , dvs sätt $c = 0.04 p_s M^2$.

Mättsignaler

Tippvinkelhastighet $\dot{\theta}$

Normalacceleration $N_z = v \dot{\gamma}$

Insignal

Roderutslag δ

Följande tillståndsvariabler införes

$$x_1 = \dot{\phi} \quad (\text{tippvinkelhast.})$$

$$x_2 = N_z = v \dot{\chi} \quad (\text{Normalacc.})$$

Tillståndsekvationerna blir då

$$x(t+1) = \begin{bmatrix} -\frac{b}{J} & -\frac{cm}{Ja} \\ \frac{a}{m} & -\frac{a}{mv} \end{bmatrix} x(t) + \begin{bmatrix} \frac{e}{J} \\ 0 \end{bmatrix} u(t)$$

$$y(t) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x(t)$$

De olika flygfall som har simulerats beskrives här

1. Start från markhöjd. Sluthöjd 3 km. Acceleration från hastigheten 0 m/s till 600 m/s . Därefter konstant hastighet till $t \approx 13$ sek (brinnslut) och sedan retardation r med $rr = 3 \cdot 10^{-4} (M = 0.5) p_g \text{ m/s}^2$.
2. Flygning till markmål. Starthöjd 3 km. Utgångshastighet 300 m/s Acceleration under 2 sek (200 m/s^2) till 700 m/s .
3. Flygning till höjdmål. Starthöjd 12 km. Sluthöjd 16 km . Utgångshastighet 300 m/s. Acceleration under 2 sek (200 m/s^2) till hastigheten 700 m/s .

Vid låg höjd och 6 - 700 km/s hastighet är modellens egenfrekvens 30 rad/s och rel. dämpningen 0.1 - 0.2 .

Egenfrekvensen minskar med ökad höjd eller minskad hastighet.

Då det gäller att $\omega_c \propto p_g$ fås en halvering av ω_c då p_g minskat till en fjärdedel, vilket motsvarar ca 10 km högre flyghöjd. Sålunda blir $\omega_c \approx 15$ rad/s vid flygning på ca 12 km höjd och med en hastighet av 6 - 700 m/s .

- Det kontinuerliga systemet har samplats med $T = 0.05$ sek.

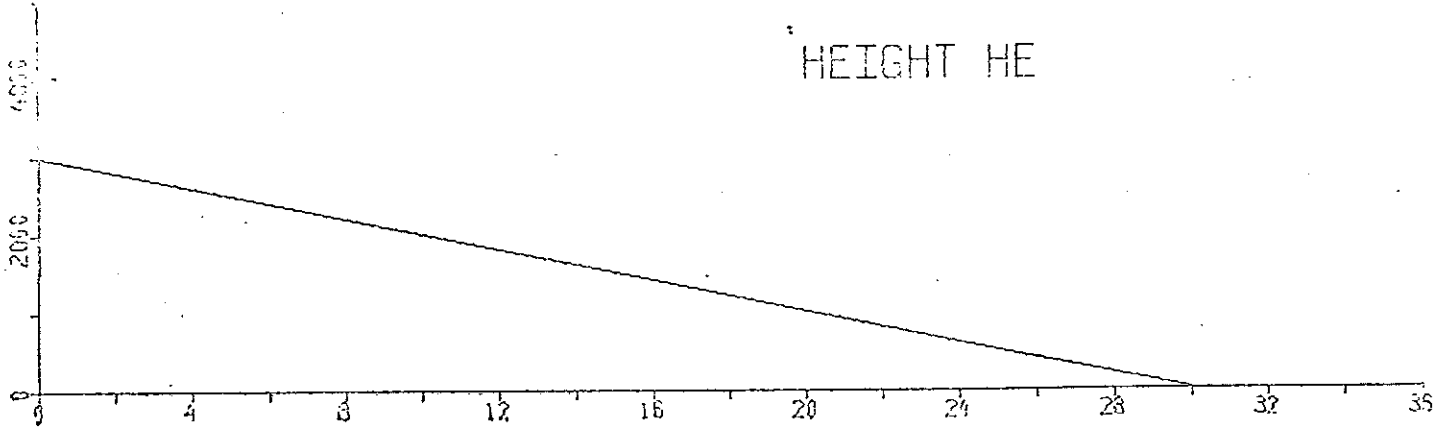
Då flygtiden är 30 sek fås 600 samplingpunkter.

Diagram bifogas för flygfall 2.

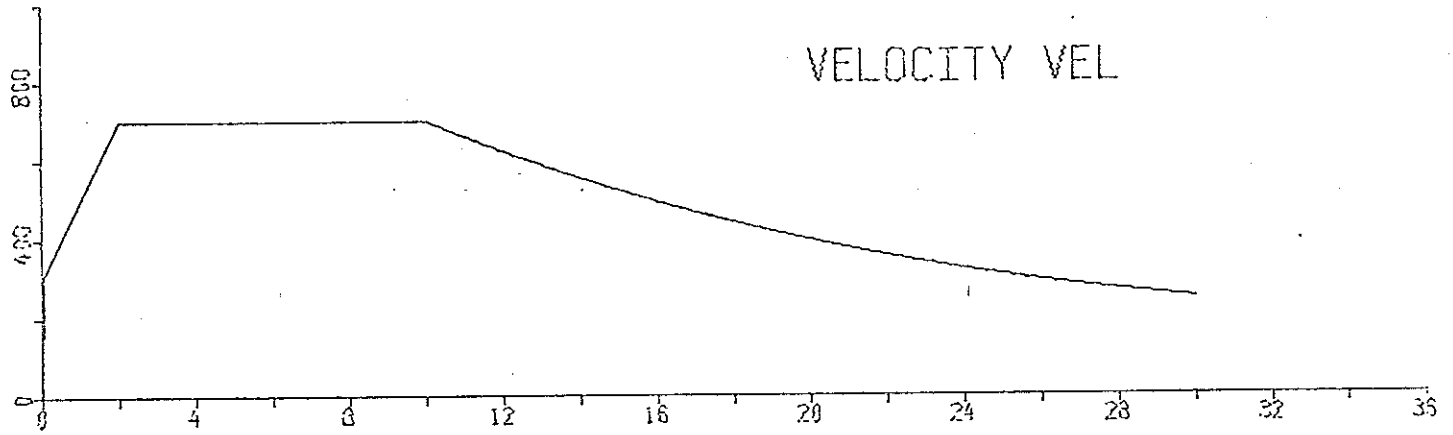
Vid uppläggnen av dessa simuleringar har Jan Andersson SAAB varit inkopplad.

PLOTS FROM ROBOT2

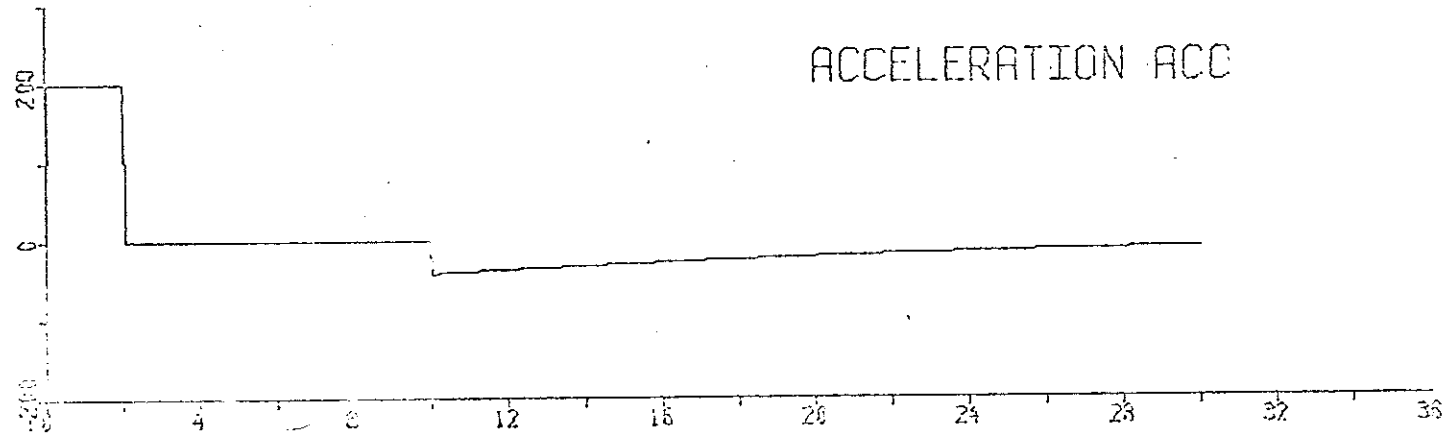
HEIGHT HE



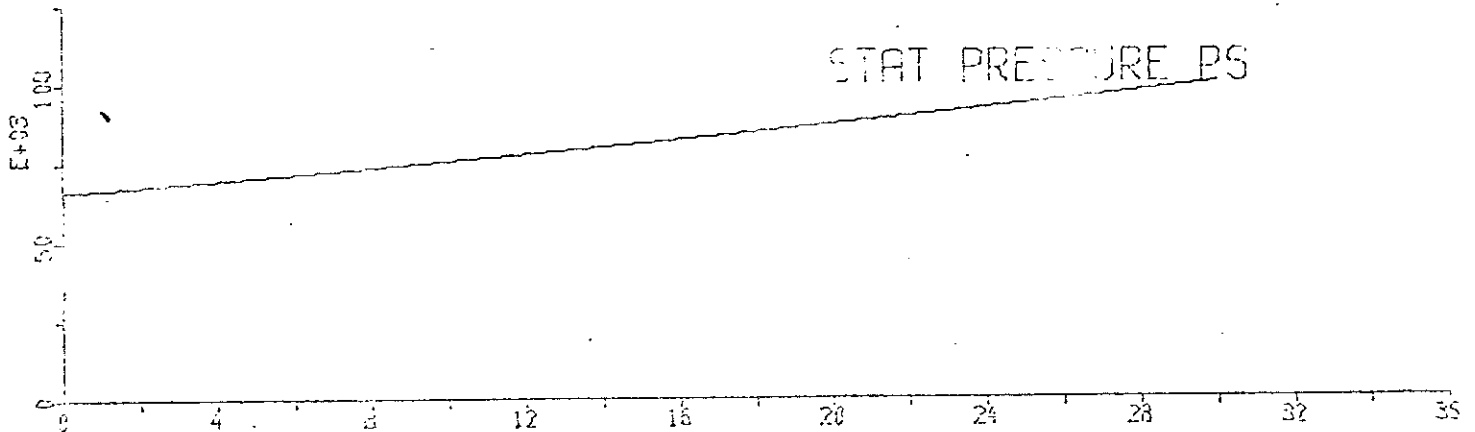
VELOCITY VEL

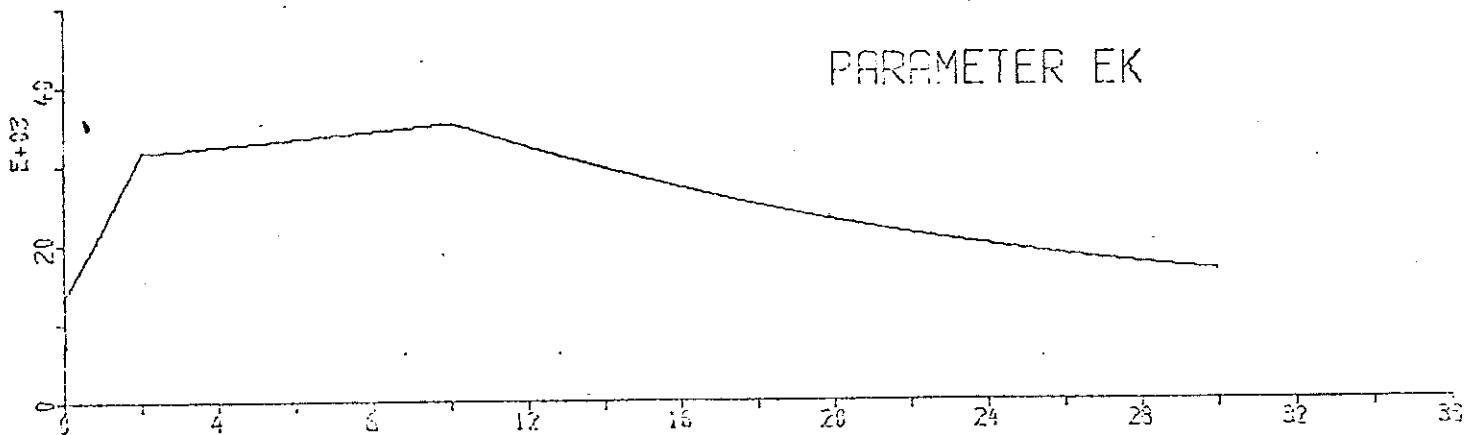
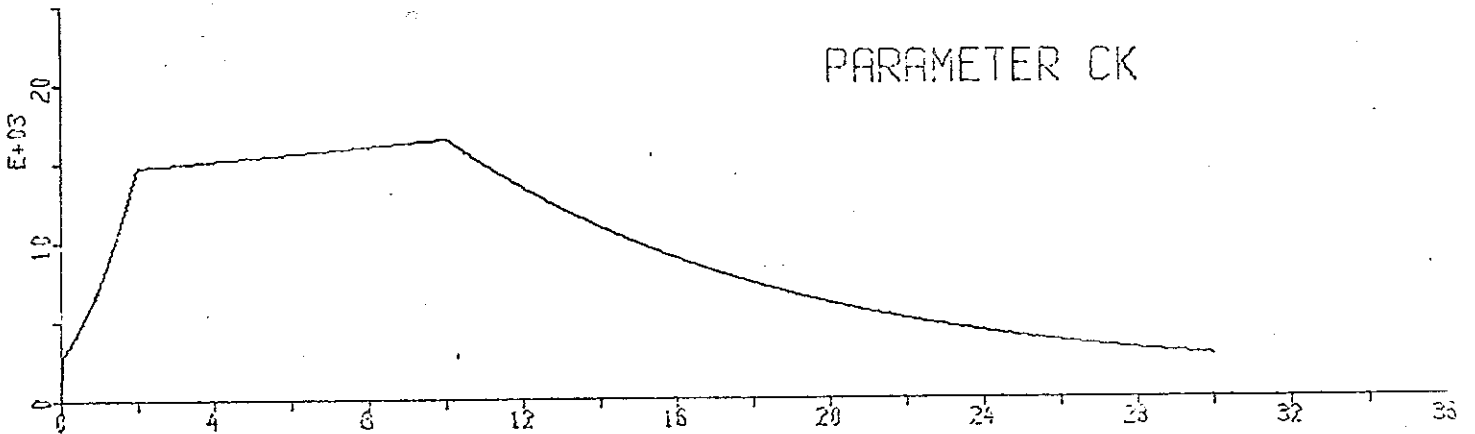
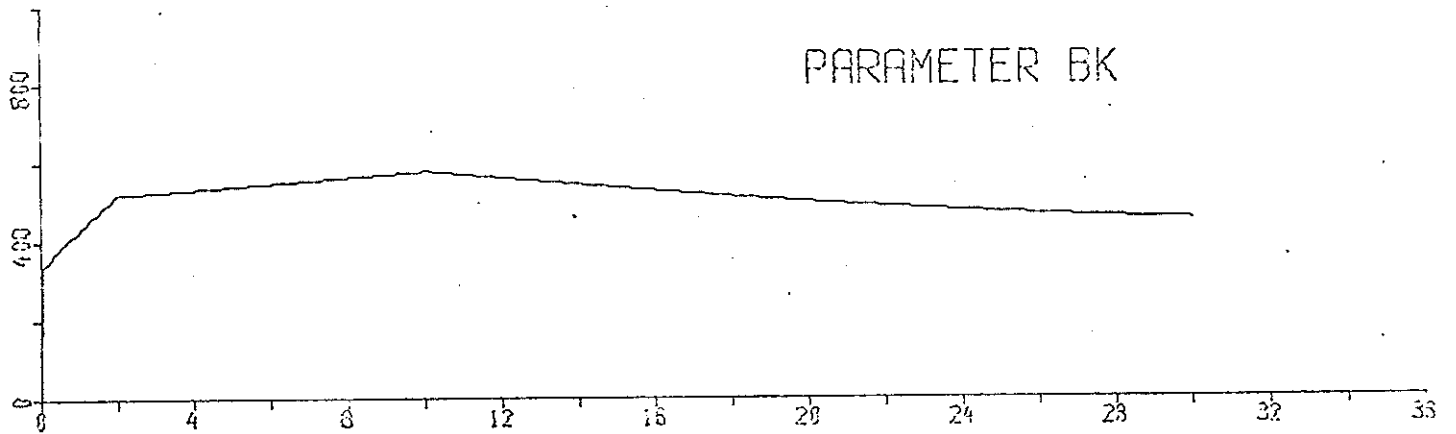
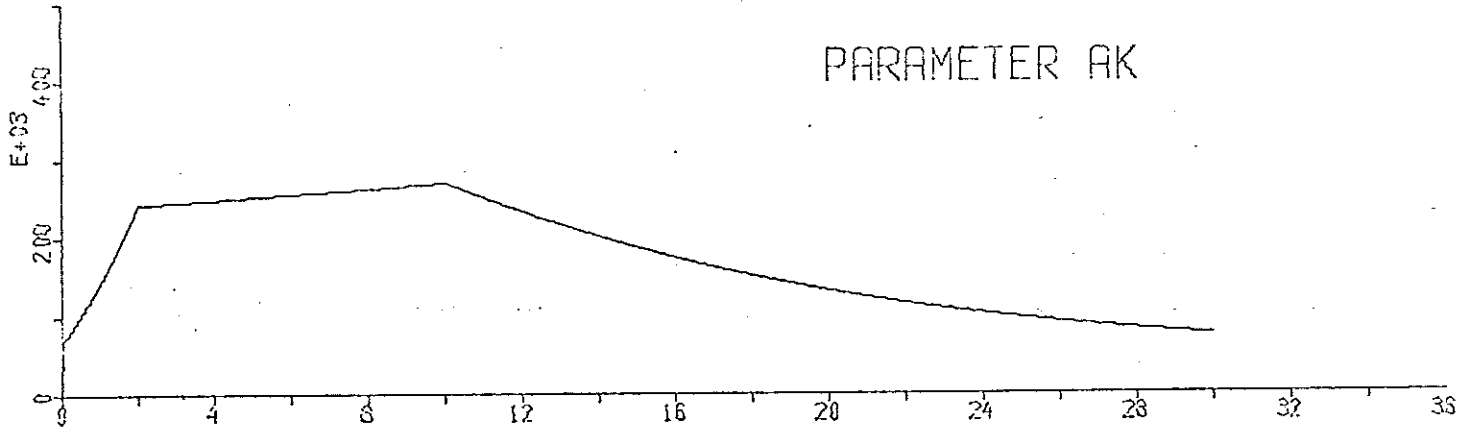


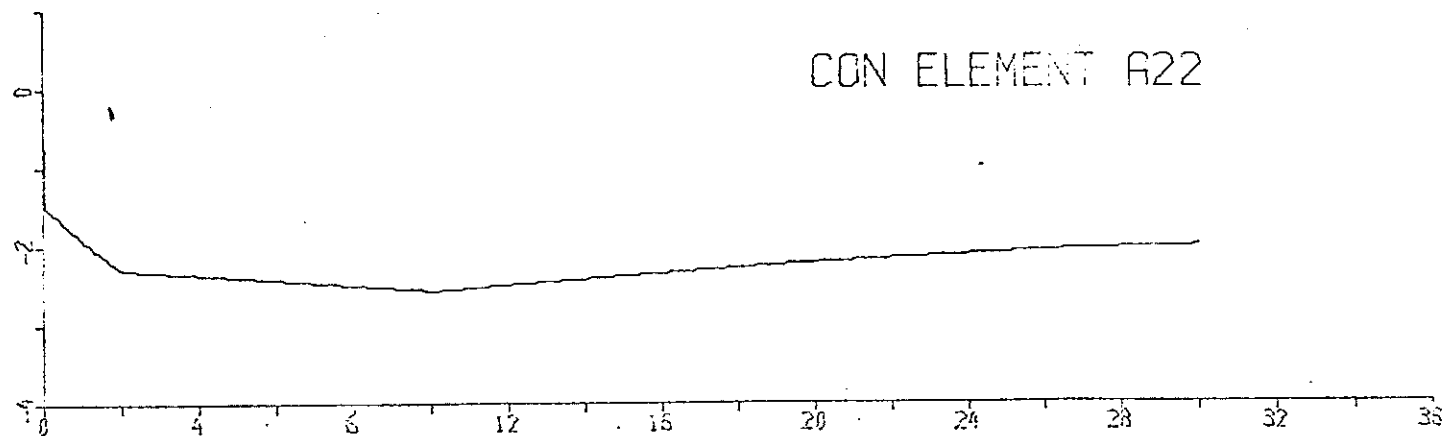
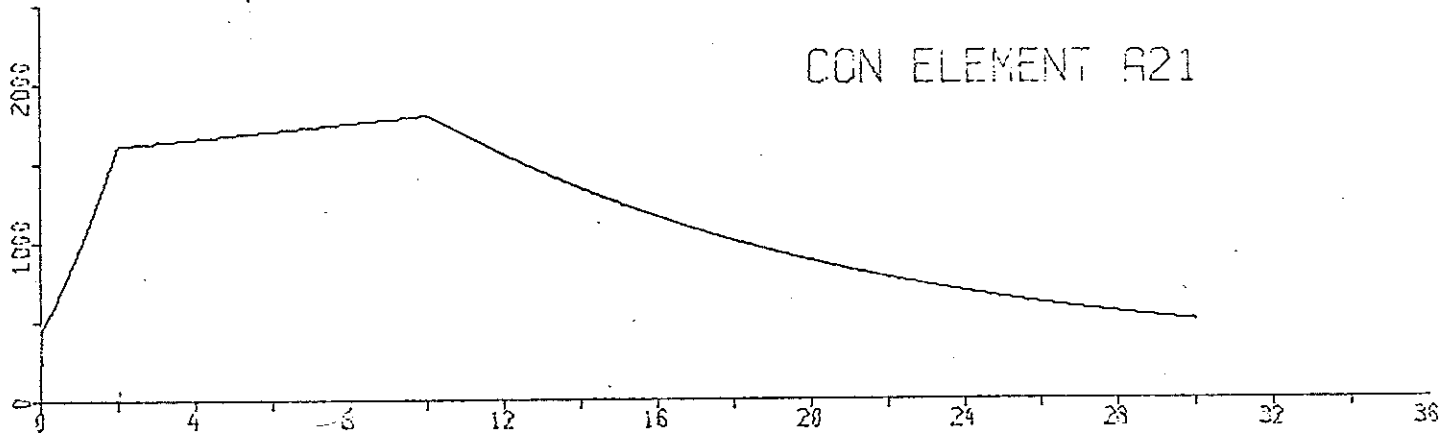
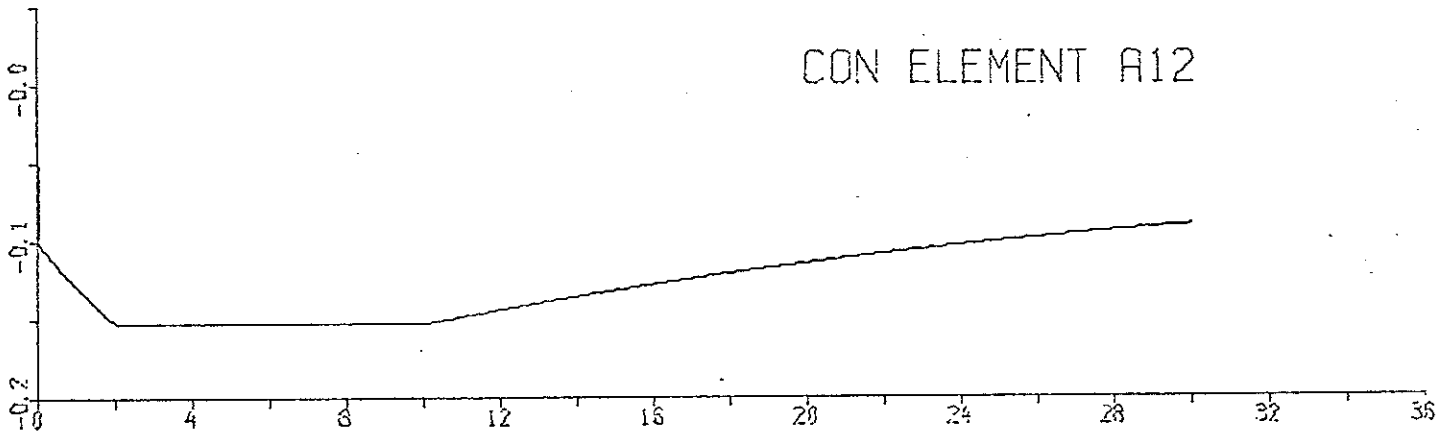
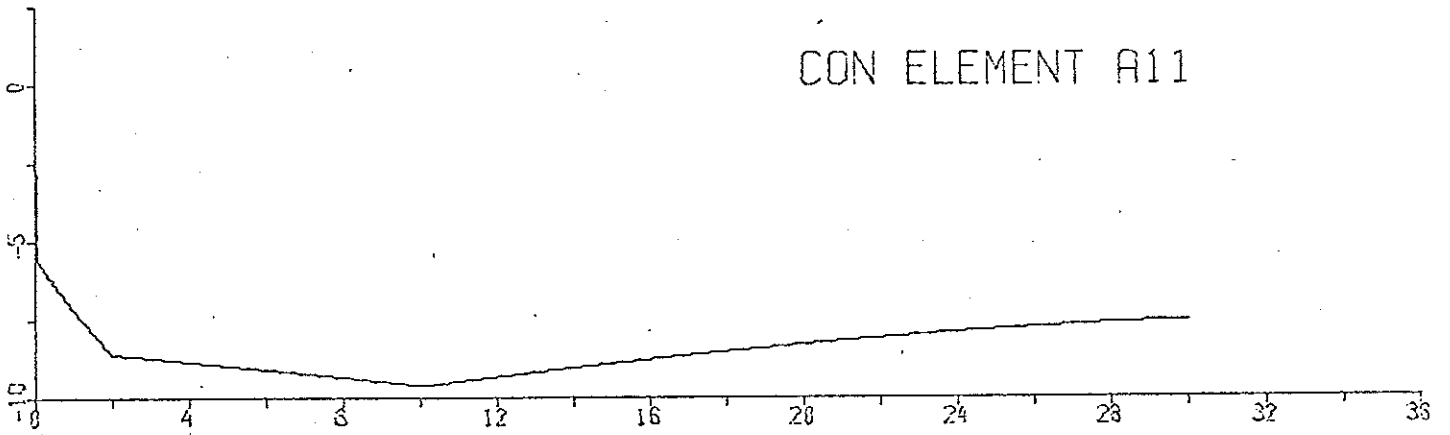
ACCELERATION ACC

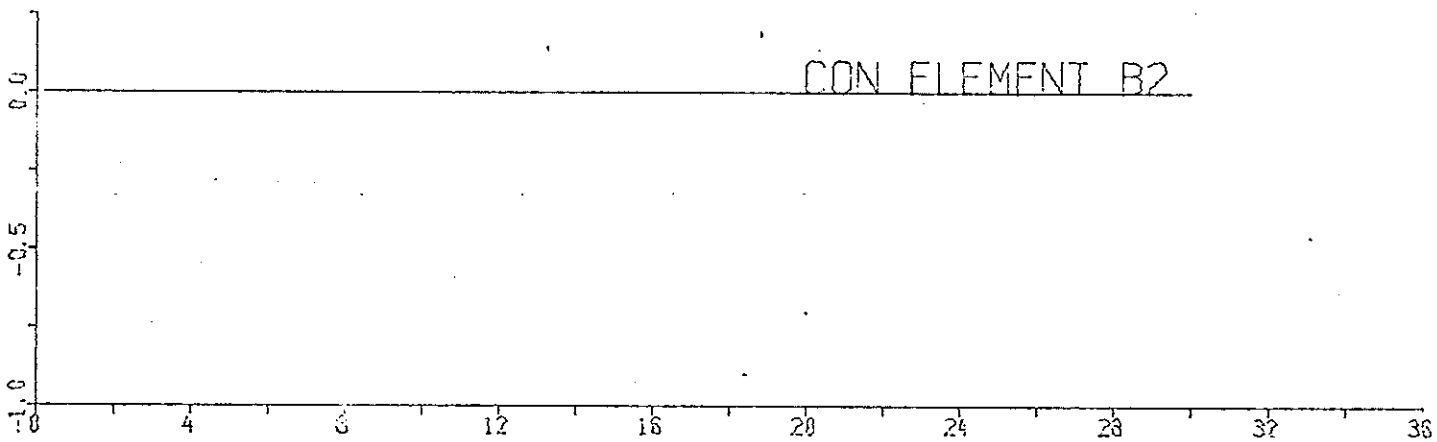
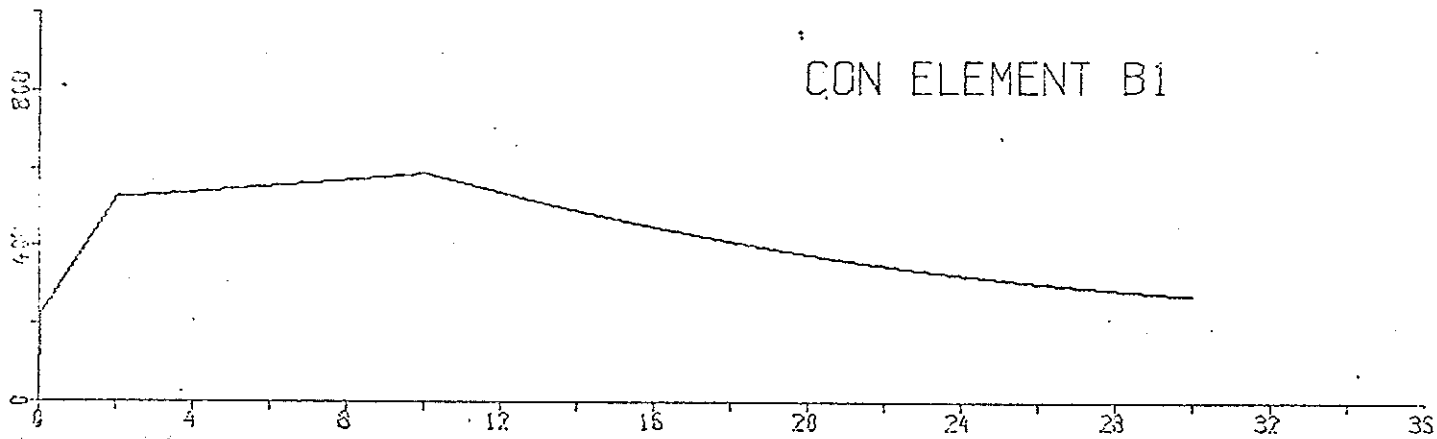


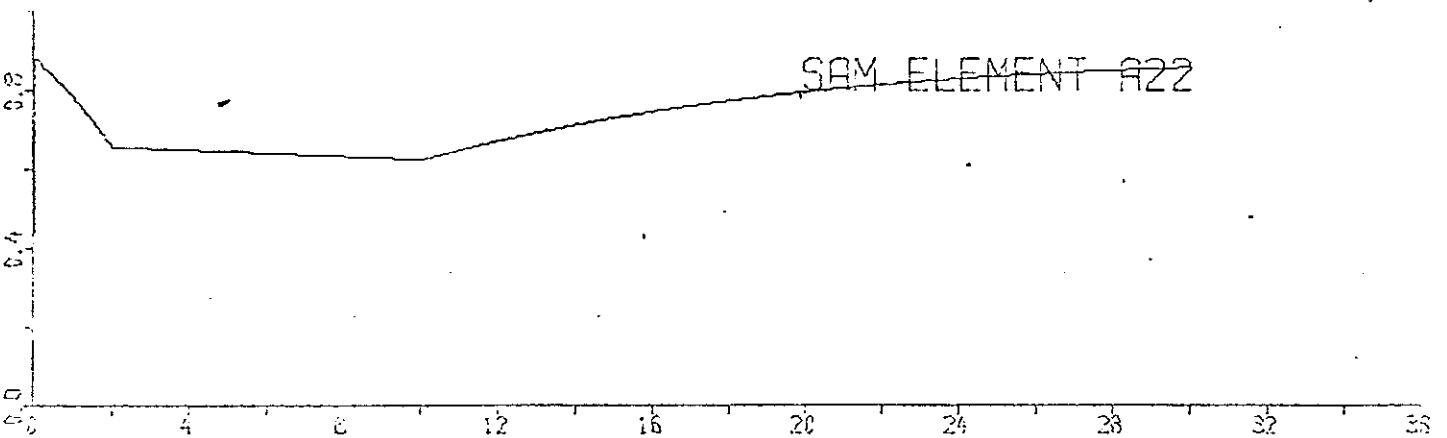
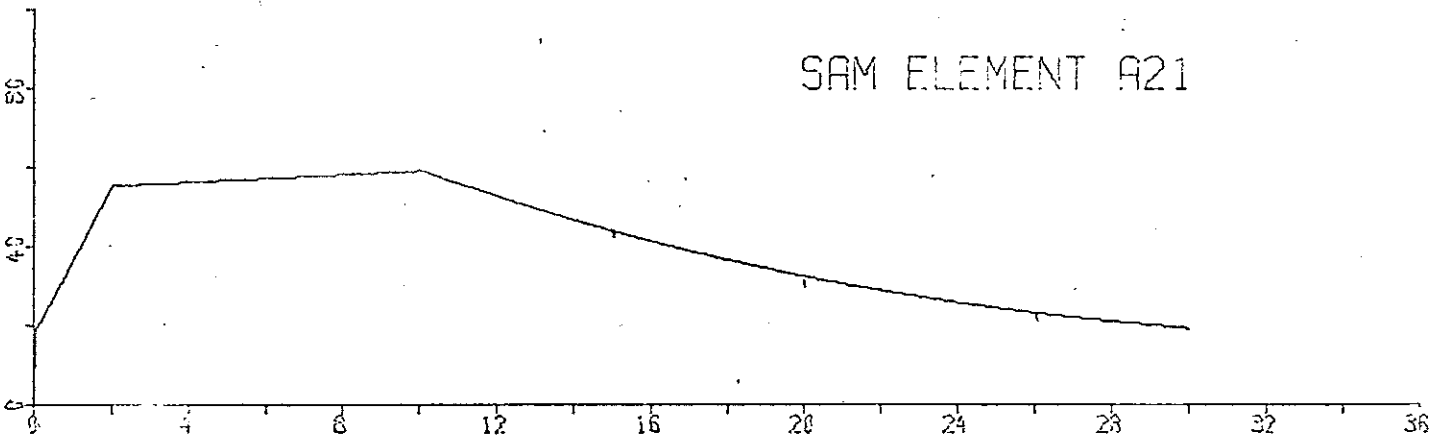
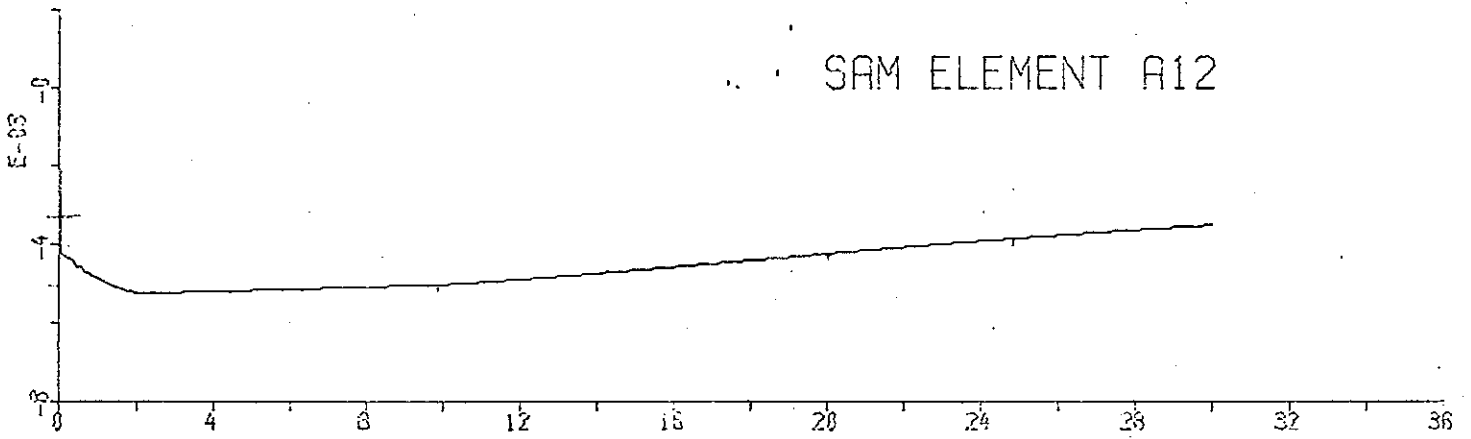
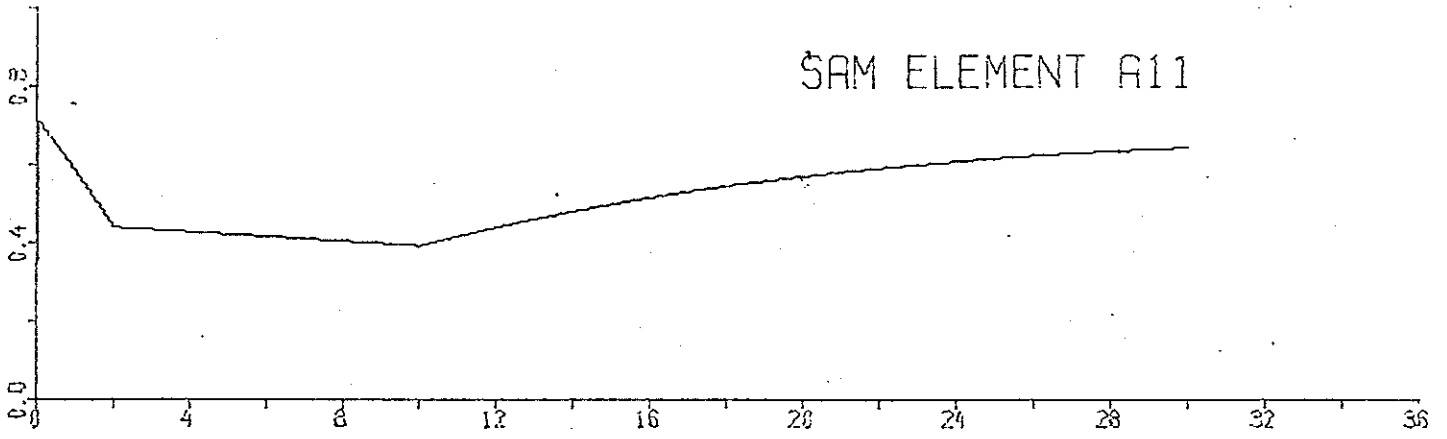
STAT PRESSURE PS

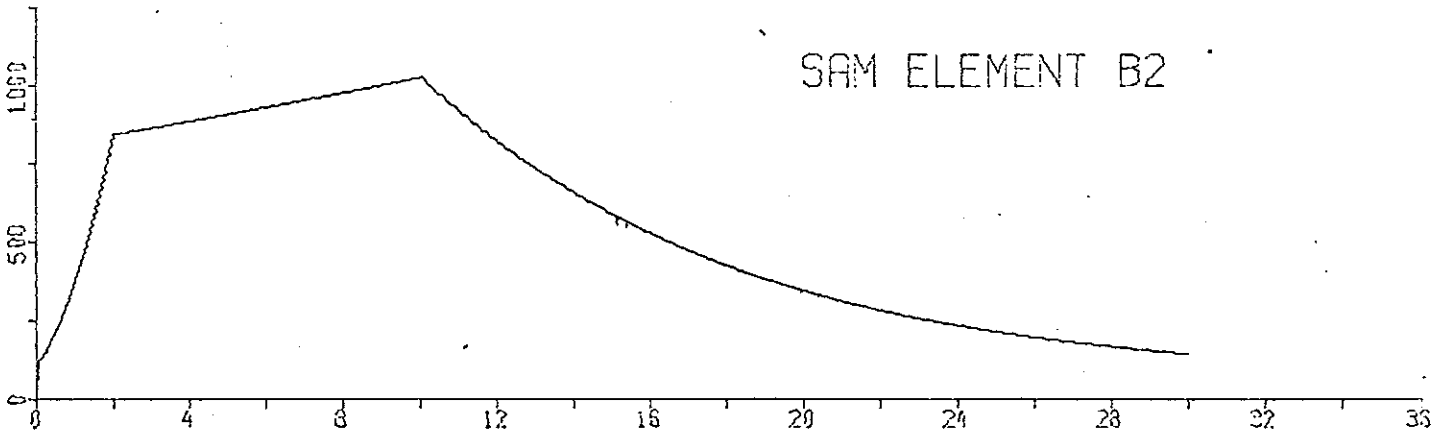
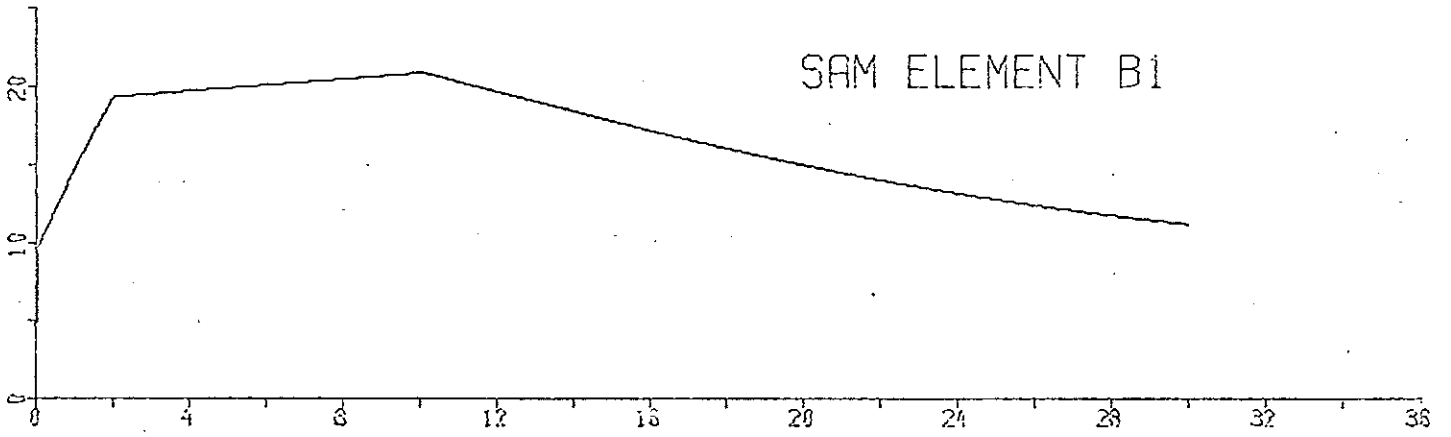












5. Test av identifieringsmetod på robot-modell.

Den aerodynamiska process, som betraktas, kan beskrivas av ett 2:a ordn. linjärt dynamiskt system.

$$x(t+1) = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} x(t) + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} u(t)$$

där x_1 och x_2 i verkligheten motsvaras av de samplade värdena på δ och N_z . u motsvaras av δ .

Uppgiften är att studera hur överföringsfunktionen $H(q) = \frac{x_1(t)}{u(t)}$

varierar under de olika flygfallen.

Målet är, att med kännedom om denna variation bestämma en styrlag så att robotens beteende blir tämligen oberoende av variationen hos flygparametrarna i A och B.

Uträknat fås överföringsfunktionen

$$H(q) = \frac{b_1(q - a_{22}) + b_2 a_{12}}{q^2 - (a_{11} + a_{22})q + a_{11}a_{22} - a_{12}a_{21}}$$

Nollstället till ovanstående befanns variera tämligen lite.

$0.86 < z < 0.92$	för flygfall	1
$0.87 < z < 0.91$	"	2
$0.97 < z < 0.98$	"	3

En konsekvens av nollställets läge blev emellertid en alltför stor översläng. Därför infördes en kaskadkopplad länk $G(q) = \frac{(1-a)q}{q-a}$

För flygfall 1, 2 resp 3 befanns $a=0.85$, $a=0.90$ resp $a=0.95$ ge en godtagbar översläng.

För att eliminera polernas rörelser, återkopplades tillståndsvariablerna. Liksom tidigare användes här en subrutin från programbiblioteket för att bestämma återkopplingsvektorn k (RECONK).

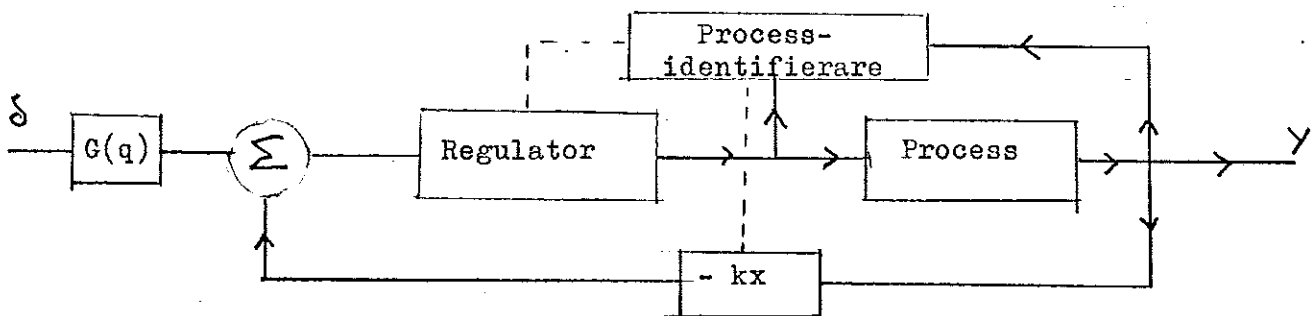
Till en början användes härför de riktiga värdena på A och B .

Detta för att konstatera om det överhuvudtaget fanns förutsättningar att identifiera A och B under dessa flygfall.

Denna förhoppning verkade till en början inte alls gå att uppfylla.

Parentetiskt kan nämnas att för flygfall 1 blev det styrda systemets uppförande acceptabelt även utan denna återkoppling, dvs. systemets transienta beteende var godtagbart. För flygfall 3 var emellertid så ej fallet (alltför dåligt dämpat).

Ett annat krav är att systemets statiska förstärkning skall vara konstant. Därför måste en förstärkningsregulator ingå.



Målet är således att identifiera A och B så bra att med kännedom om dessa kunna bestämma

1. en återkoppling, så att systemet får önskade egenskaper
2. en variabel förstärkning, så att det styrda systemets statiska förstärkning blir konstant.

Beträffande punkt 1. uppstod som tidigare nämnts svårigheter.

Vad punkt 2. anbelangar konstaterades det, efter en rad försök, att för förstärkningsreglering i alla tre flygfallen krävdes kännedom om b_1 , b_2 , a_{22} och a_{12} . Det verkade inte alls förhoppningsfullt.

Skalning av systemparametrarna.

Det verkade som om metoden att identifiera A och B inte alls kunde fås att fungera för de givna flygfallen.

Men varför lyckades identifieringen för enstaka parametrar?

Det stod nu klart att A och B var dåligt konditionerade matriser.

Försök gjordes att skala tillståndsvariablerna x_1 och x_2 , vilket gav något bättre resultat, dock ej tillräckligt bra.

Då uppstod tanken att skala samtliga parametrar i A och B.

Observera att x_1 , x_2 och u då kommer att skalas olika för de båda raderna i A och B.

Hur skulle denna skalning väljas?

De skalade systemekvationerna ser ut enligt nedan.

$$\left[\frac{\alpha_{11} x_1(t+1)}{\alpha_{11}} \right] = \left[\frac{a_{11} \alpha_{11} x_1(t)}{\alpha_{11}} \right] + \left[\frac{a_{12} \alpha_{12} x_2(t)}{\alpha_{12}} \right] + \left[\frac{b_1 \beta_1 u(t)}{\beta_1} \right]$$

$$\left[\frac{\alpha_{22} x_2(t+1)}{\alpha_{22}} \right] = \left[\frac{a_{21} \alpha_{21} x_1(t)}{\alpha_{21}} \right] + \left[\frac{a_{22} \alpha_{22} x_2(t)}{\alpha_{22}} \right] + \left[\frac{b_2 \beta_2 u(t)}{\beta_2} \right]$$

Det synes rimligt att på något vis välja skalningen efter hur parametrarna i processen varierar. Följande villkor tänktes ge en bra skalning :

$$\frac{\Delta a_{11}}{\alpha_{11}} = \frac{\Delta a_{12}}{\alpha_{12}} = \frac{\Delta b_1}{\beta_1} = \frac{\Delta a_{21}}{\alpha_{21}} = \frac{\Delta a_{22}}{\alpha_{22}} = \frac{\Delta b_2}{\beta_2}$$

där Δa_{11} Δa_{12} osv

betecknar variationen hos resp. parameter.

Ur diagram ^{från} robotsimuleringarna bestämdes med ögonmått vägningsvektorer för de olika flygfallen.

$$V = (\alpha_{11} : \alpha_{12} : \beta_1 : \alpha_{21} : \alpha_{22} : \beta_2)$$

$$V_1 = (0.5 : 0.005 : 20 : 50 : 0.35 : 950) \quad \text{flygfall 1}$$

$$V_2 = (0.4 : 0.0018 : 12 : 40 : 0.30 : 900) \quad \text{" 2}$$

$$V_3 = (0.05 : 0.002 : 2.5 : 10 : 0.05 : 50) \quad \text{" 3}$$

Observera att dessa vägningsvektor är valda för de speciella flygfall som simuleras.

Därigenom får styrlagen vissa drag av "programmerad styrlag".

Med införandet av dessa vägningsvektorer blev identifieringen väsentligen bättre.

För flygfall 1 blev det dock inte så bra, kanske beroende på olyckligt vald vägningsvektor.

För flygfall 2 och 3 blev emellertid efter ca 150 samplingintervall identifieringen tillräckligt bra för att kunna utnyttja de identifierade parametrarna till att bestämma en återkopplingsvektor k ($u = u_r - k x$).

Vidare konstaterades det att efter dessa 150 samplingintervall gick det att bestämma en tillräckligt bra förstärkningsregulator.

Denna regulator infördes dock ej i de följande gjorda simuleringarna.

Likaså tycktes nollstället ^{1/}överföringsfunktionen $H(q)$ gå att bestämma tämligen bra. Ej heller denna information utnyttjades utan polen i den kaskadkopplade länken fick konstant värde enligt föregående.

Önskad kar ekv. för det återkopplade systemet valdes till

$$q^2 - 0.4 q + 0.25 = 0$$

Ex. 8 visar de identifierade parametrarna för flygfall 2 efter 175 samplingintervall.

Följande diagram bifogas: \hat{a}_{11} , \hat{a}_{12} , \hat{a}_{21} , \hat{a}_{22} , \hat{b}_1 , \hat{b}_1 ,

w_1 , w_2 , \hat{b}_2 , u , x_1 , x_2

där w_1 och w_2 är koeff. till kar. ekv.

Som synes identifieras parametrarna väl.

w_1 och w_2 antager efter dessa 175 samplingintervall inga otill-låtna värden, dvs. det återkopplade systemet är stabilt.

Under de inledande 175 samplingintervallen skulle systemet varit instabilt 59 gånger om de estimerade parametrarna utnyttjats för återkoppling.

Detta exempel visar att det är möjligt att identifiera parametrarna i den givna processen, åtminstone efter viss tid.

Valet av vägningsvektor är antagligen tämligen känsligt.

Ett bättre val skulle ev. kunna ge ännu bättre identifiering.

Området där tillräckligt bra identifiering erhöles, skulle troligtvis varit större om mer aktivitet på insignalen varit för handen.

Ex. 9

För att belysa beroendet av stor insignalaktivitet gjordes ytterligare en simulering på flygfall 2 . Denna gång med 5 gånger så stor aktivitet på insignalen som i föregående exempel.

Följande diagram bifogas: \hat{a}_{11} , \hat{a}_{12} , \hat{a}_{21} , \hat{a}_{22} , \hat{b}_1 , \hat{b}_1 ,
 w_1 , w_2 , \hat{b}_2 , u , x_1 , x_2 .

De estimerade parametrarna utnyttjas här efter 100 samplingtidpunkter för att bestämma en återkoppling.

Såsom framgår av diagrammen identifieras parametrarna mycket bra.

w_1 resp w_2 avviker mycket lite från de önskade -0.40 resp 0.25 .

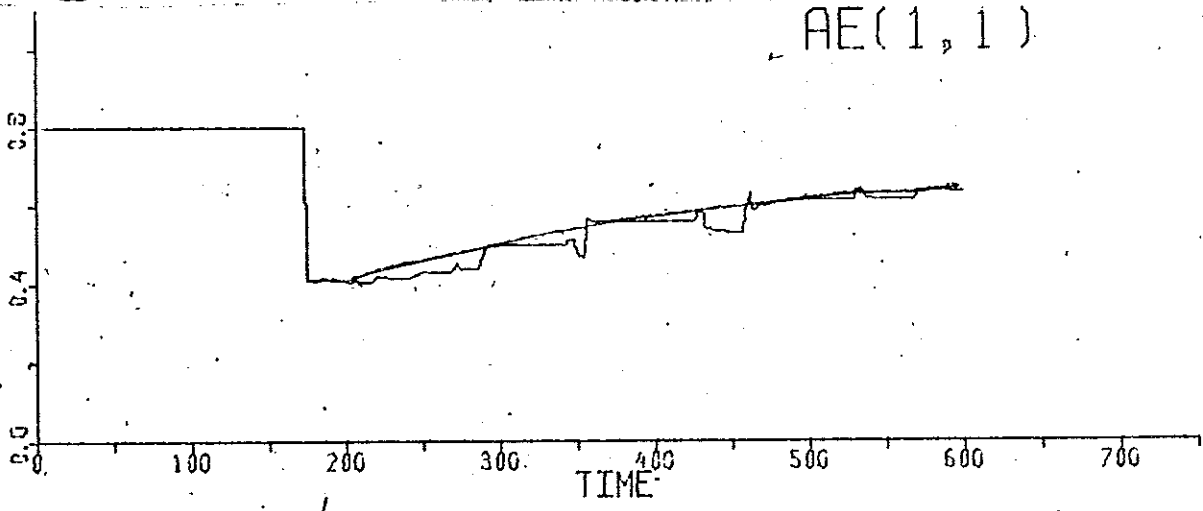
Systemet skulle varit instabilt 22 gånger om de estimerade parametrarna utnyttjats för att bestämma en återkoppling under de inledande 100 samplingintervallen.

Då så stor styraktivitet vanligtvis ej förekommer, måste en speciell signal superponeras den egentliga styrsignalen.

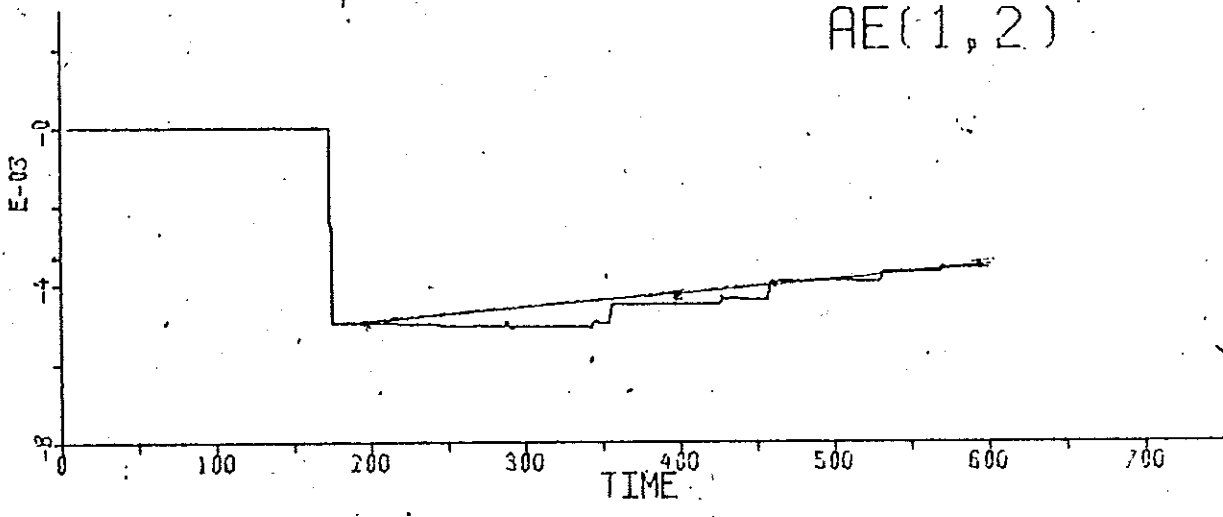
Den superponerade signalen skall givetvis vara av betydligt mindre amplitud än den egentliga styrsignalen.

Ex. 8

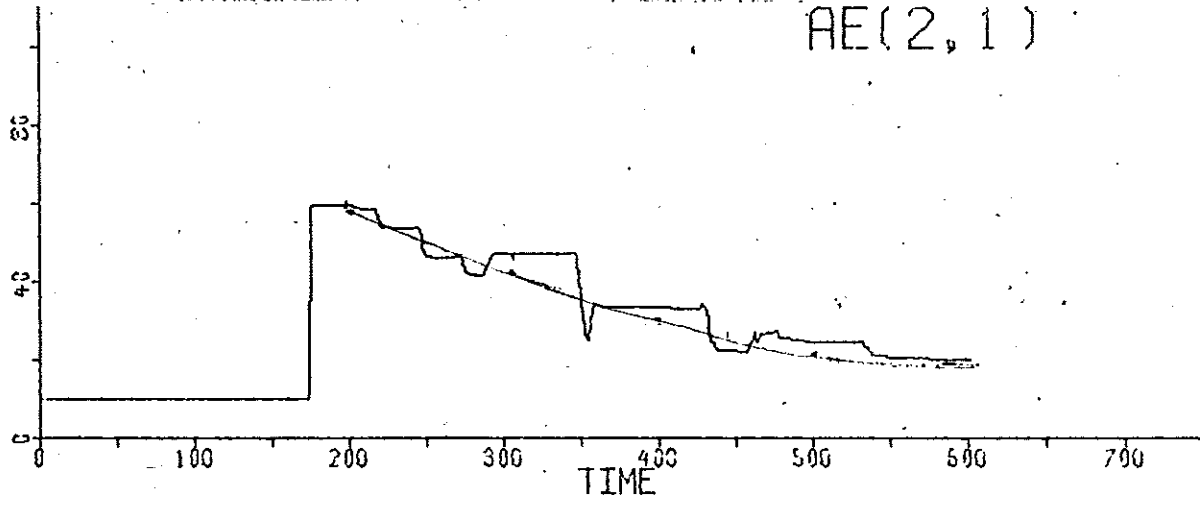
\hat{a}_{11}



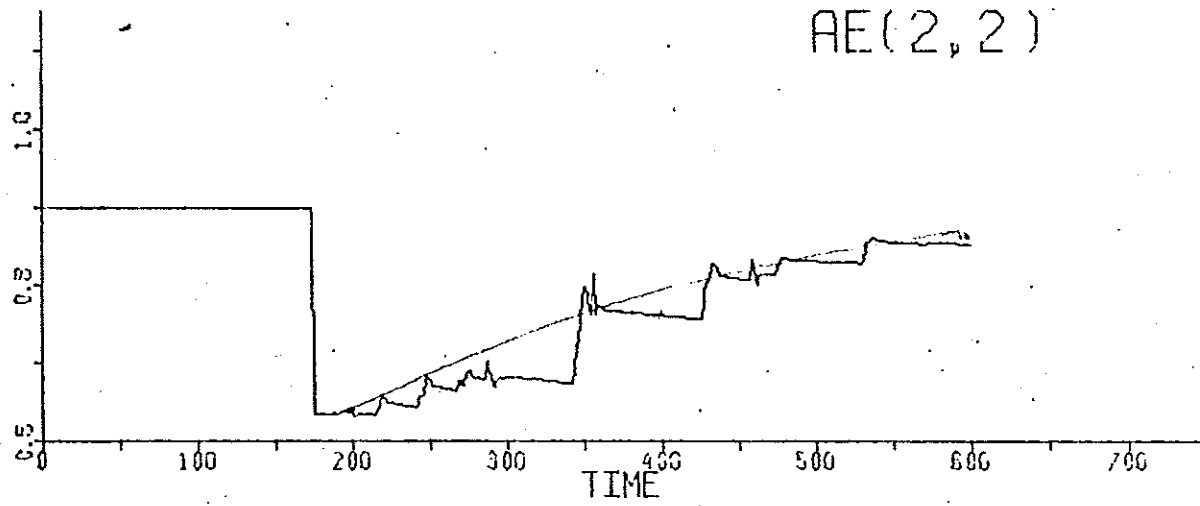
\hat{a}_{12}



\hat{a}_{21}



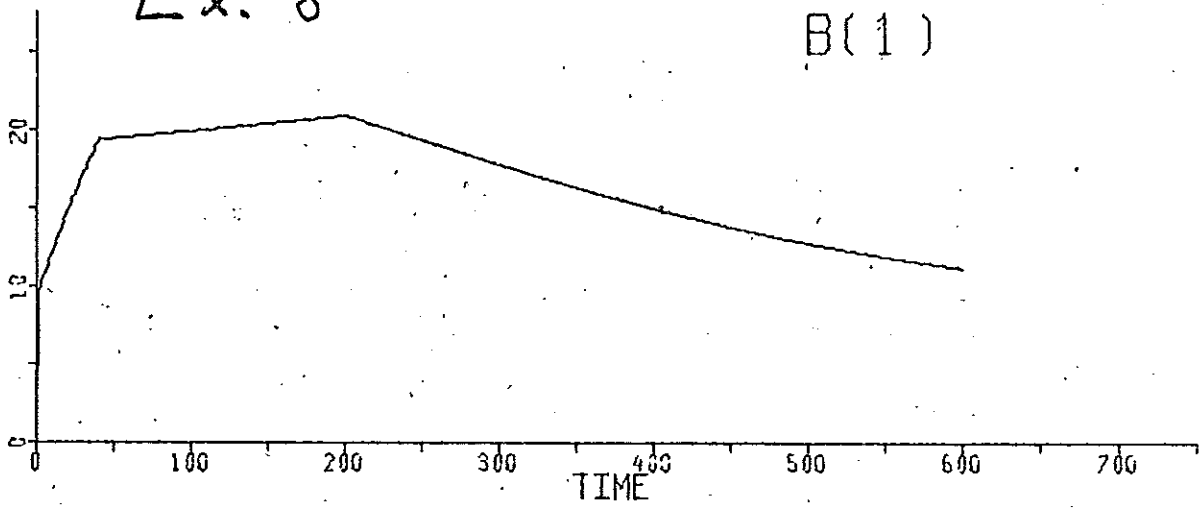
\hat{a}_{22}



Ex. 8

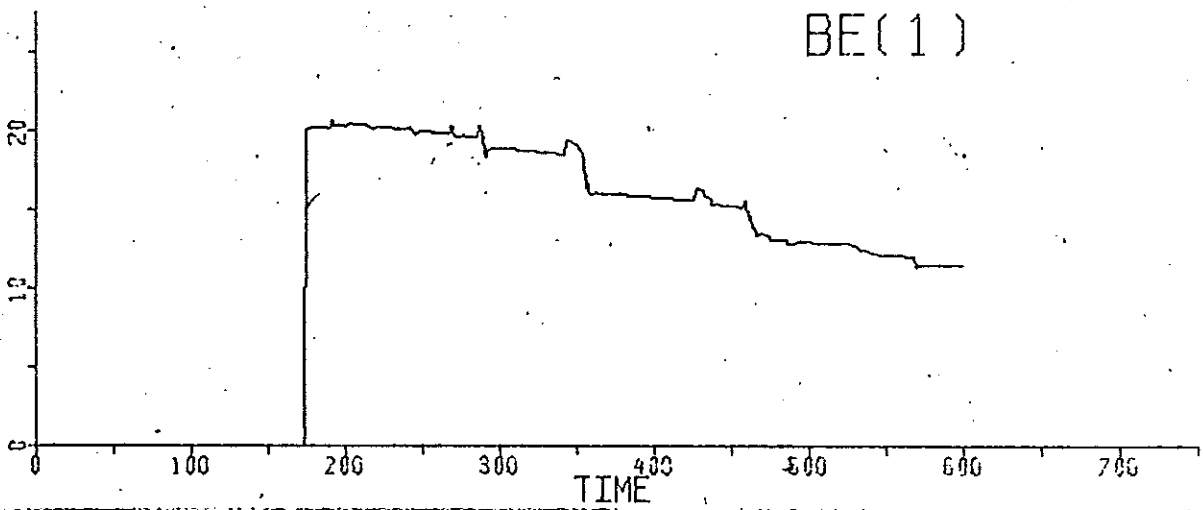
ψ_1

B(1)



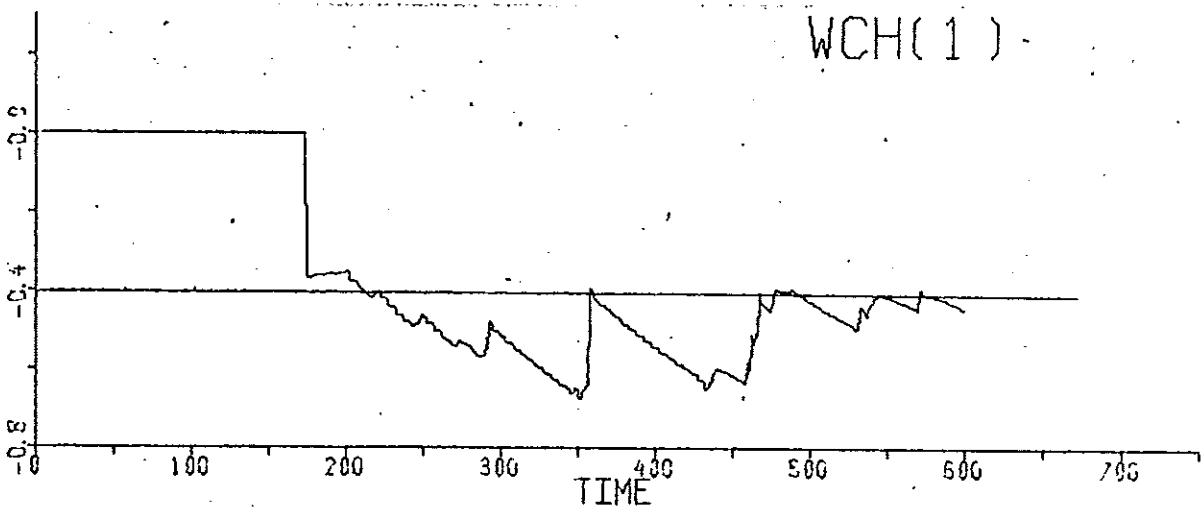
ψ_2

BE(1)



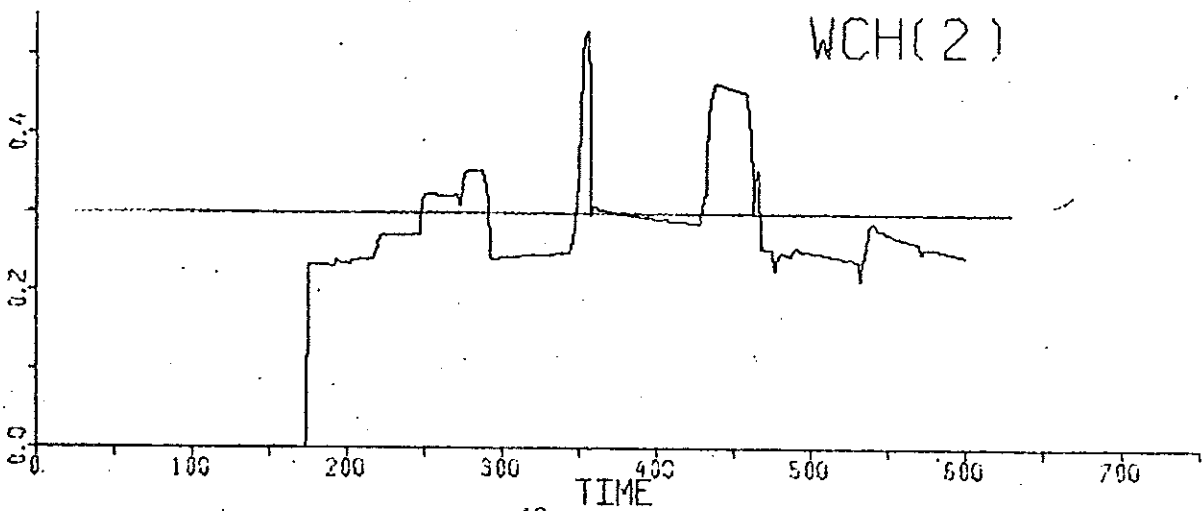
w_1

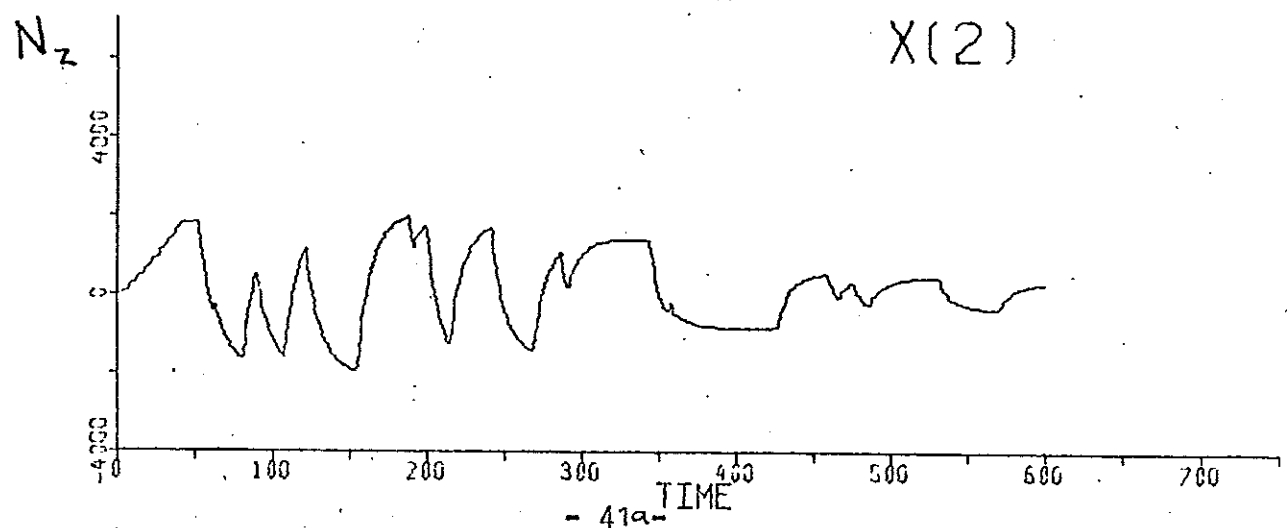
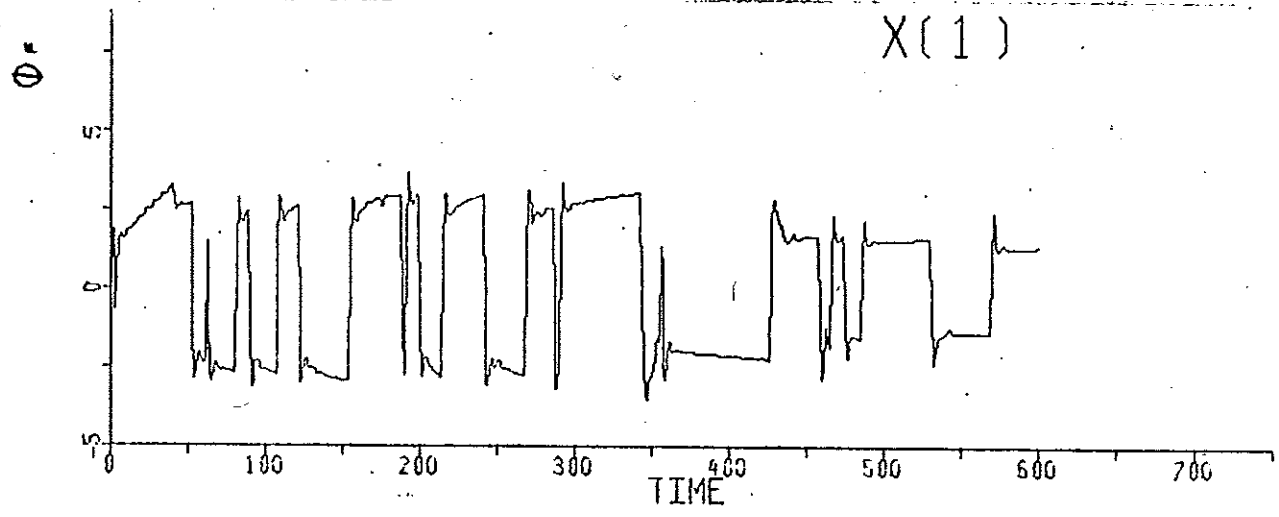
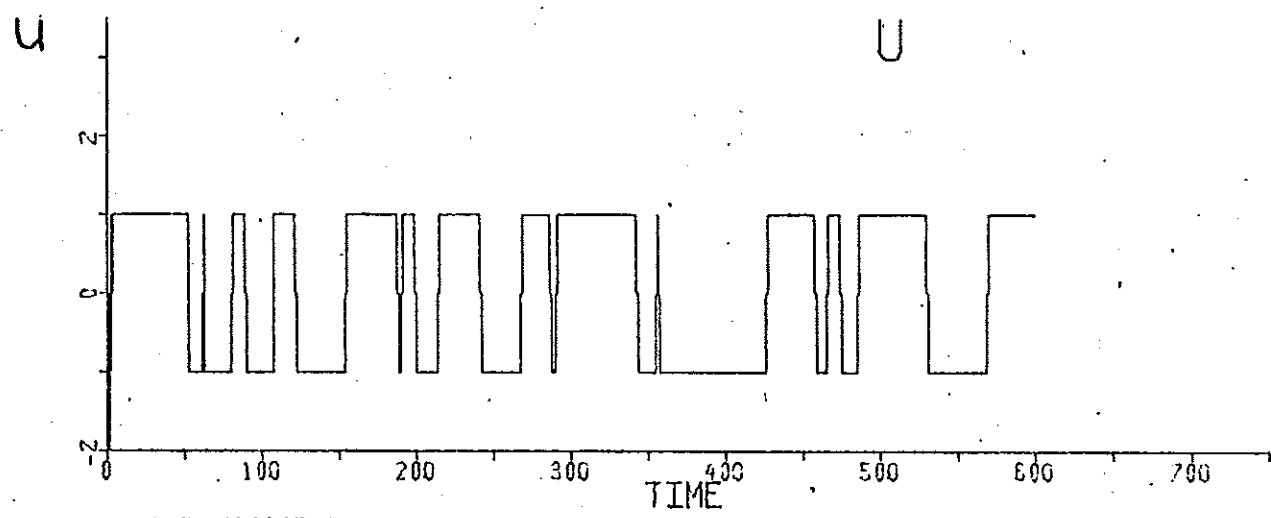
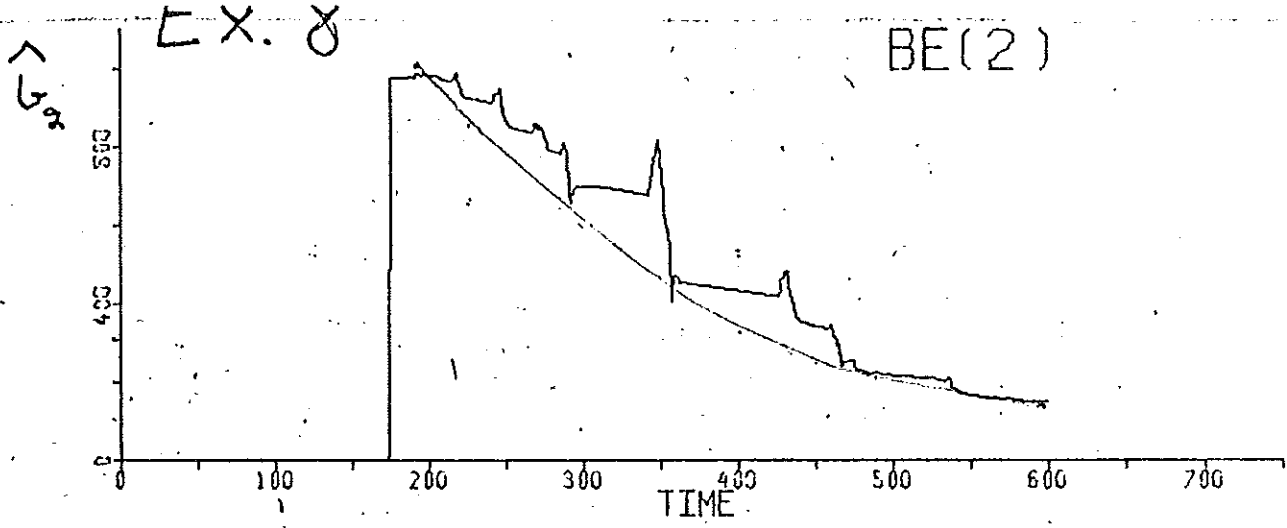
WCH(1)



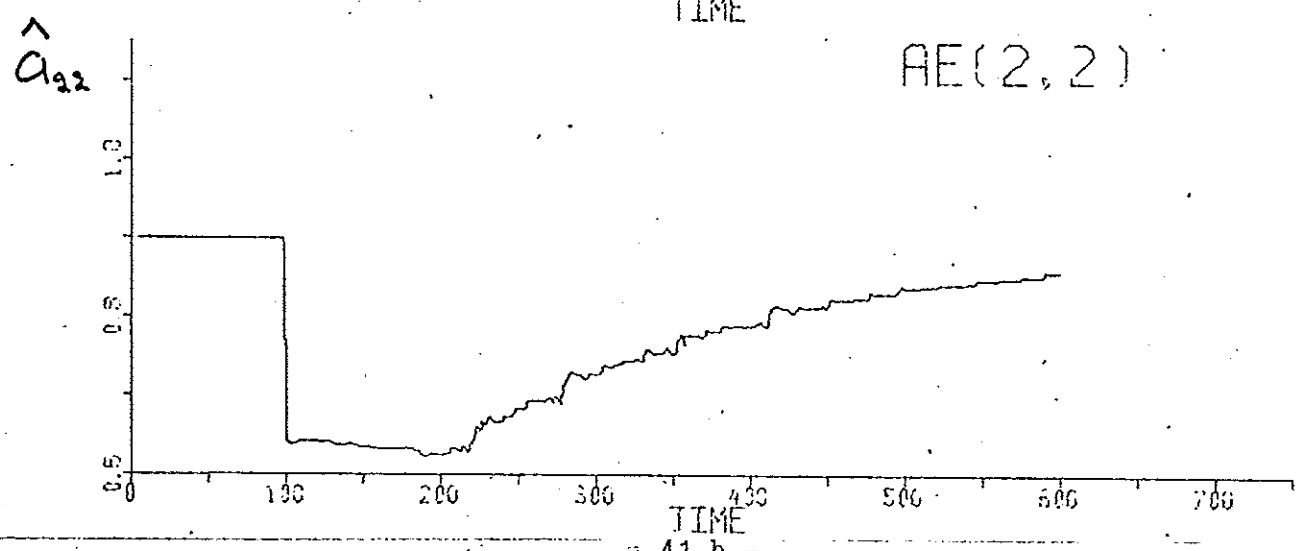
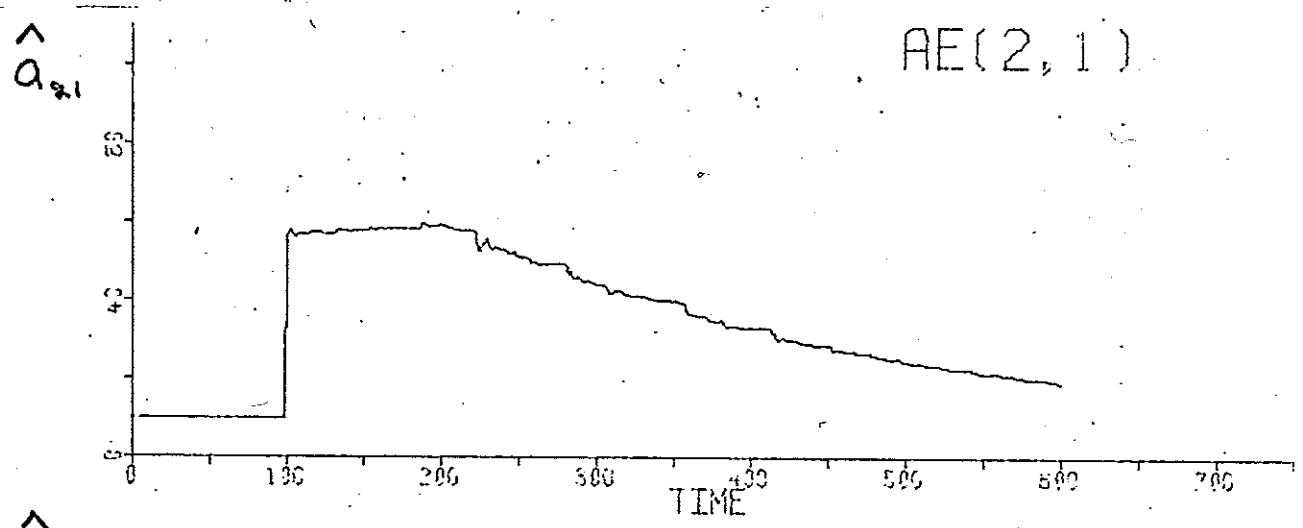
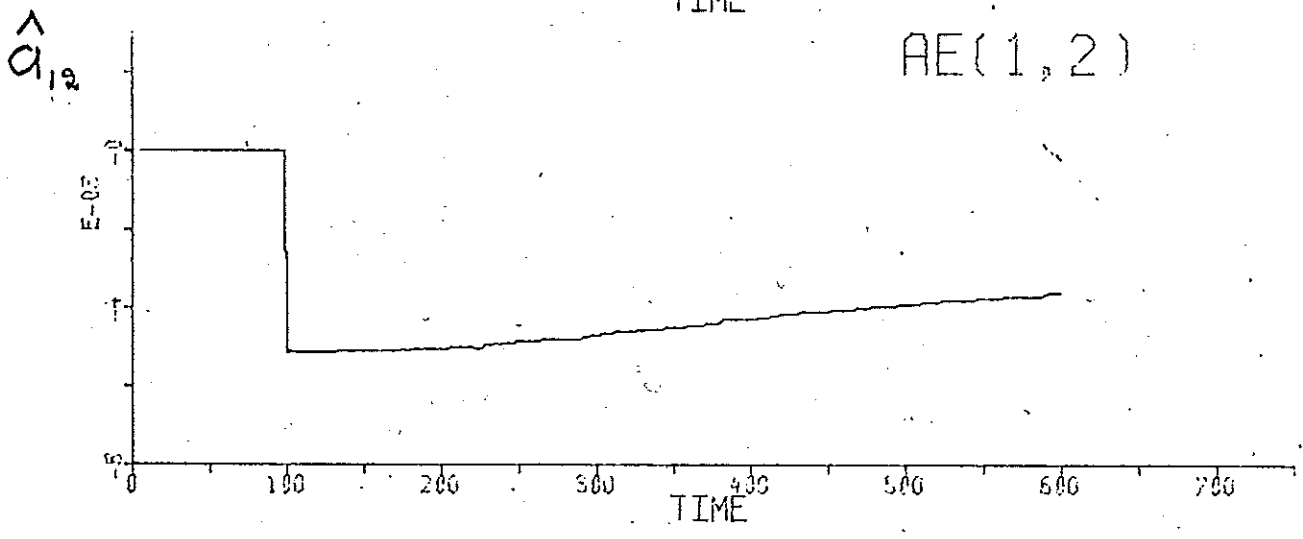
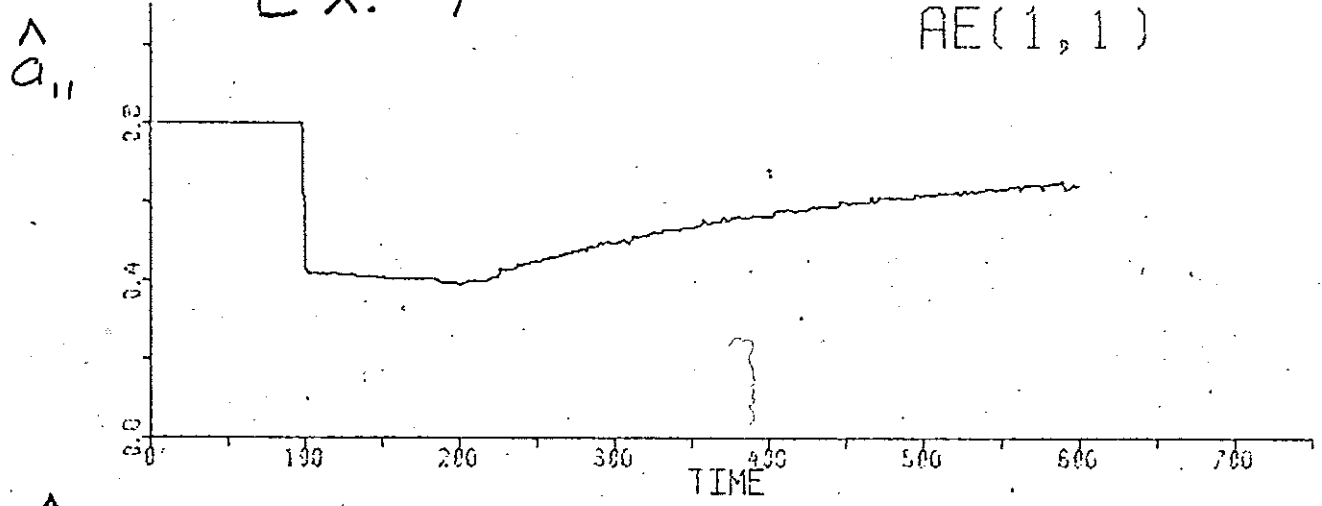
w_2

WCH(2)





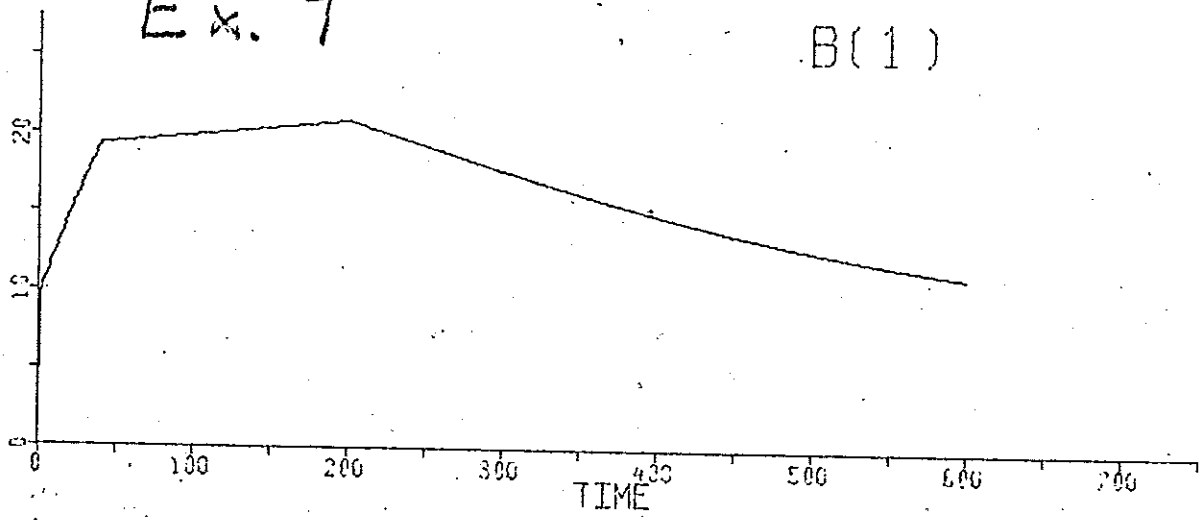
Ex. 9



Ex. 9

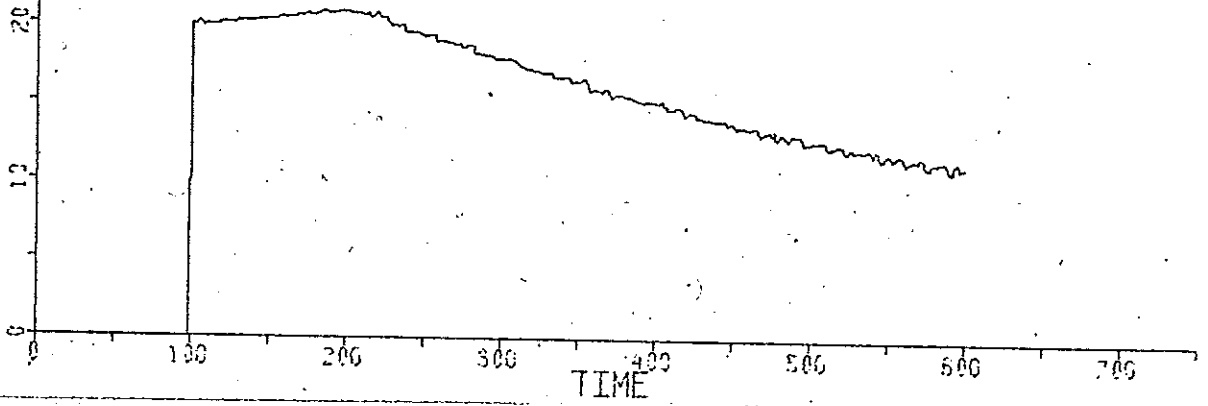
B(1)

ψ_1



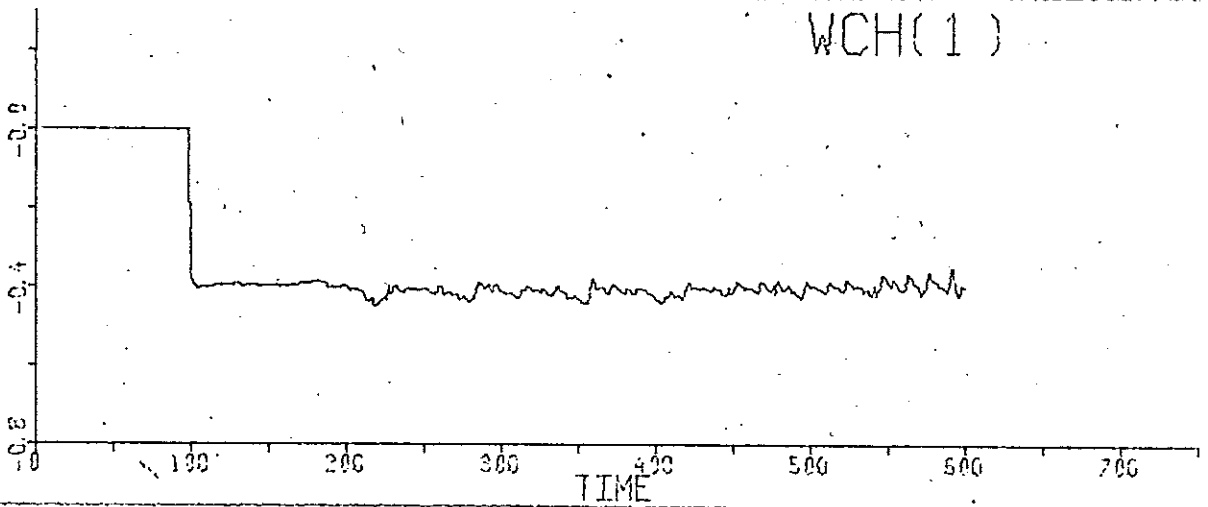
BE(1)

ψ_2



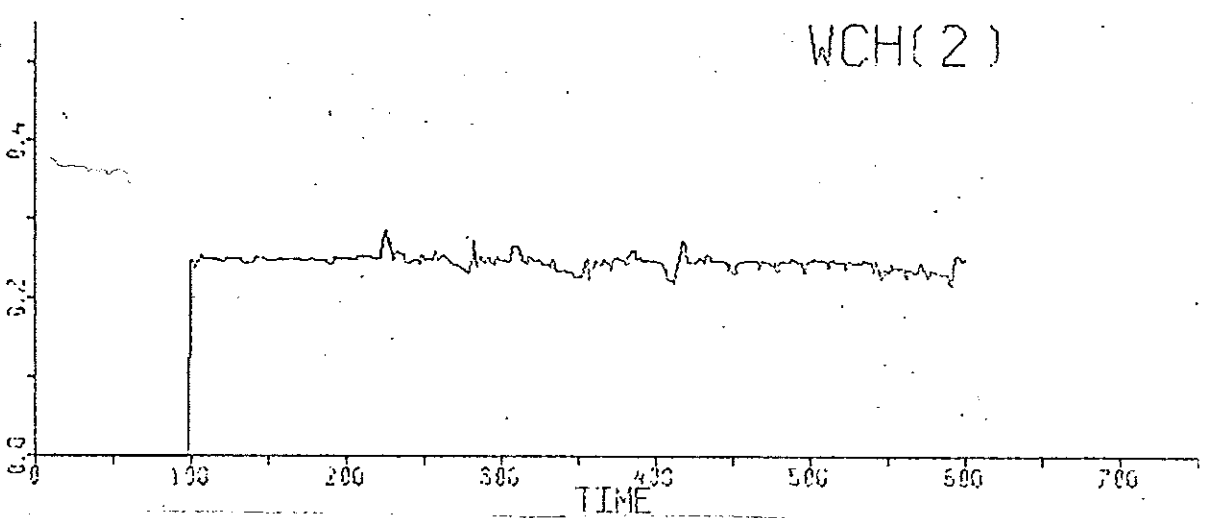
WCH(1)

ψ_1



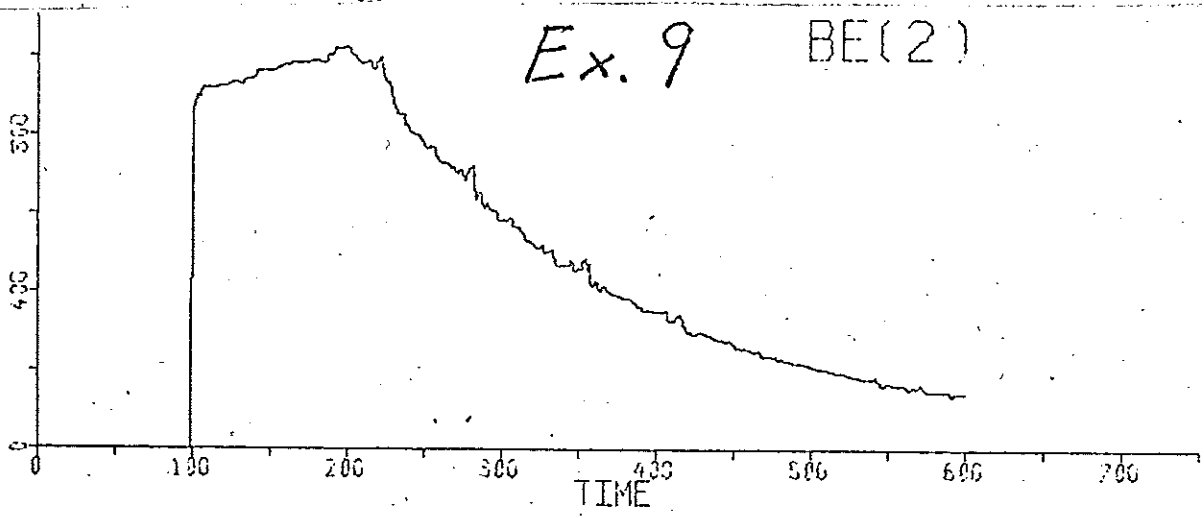
WCH(2)

ψ_2



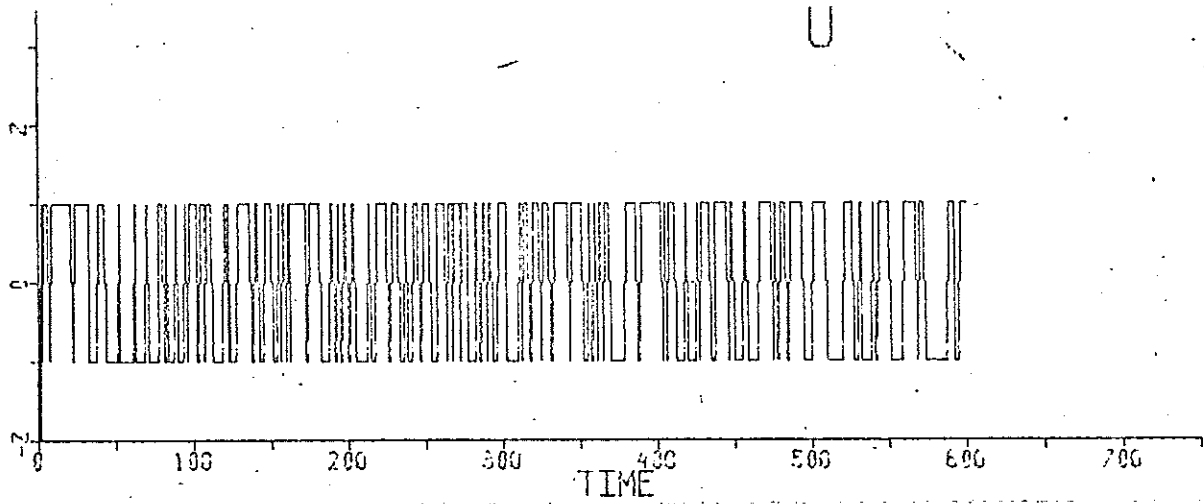
Ex. 9 BE(2)

\hat{u}_g



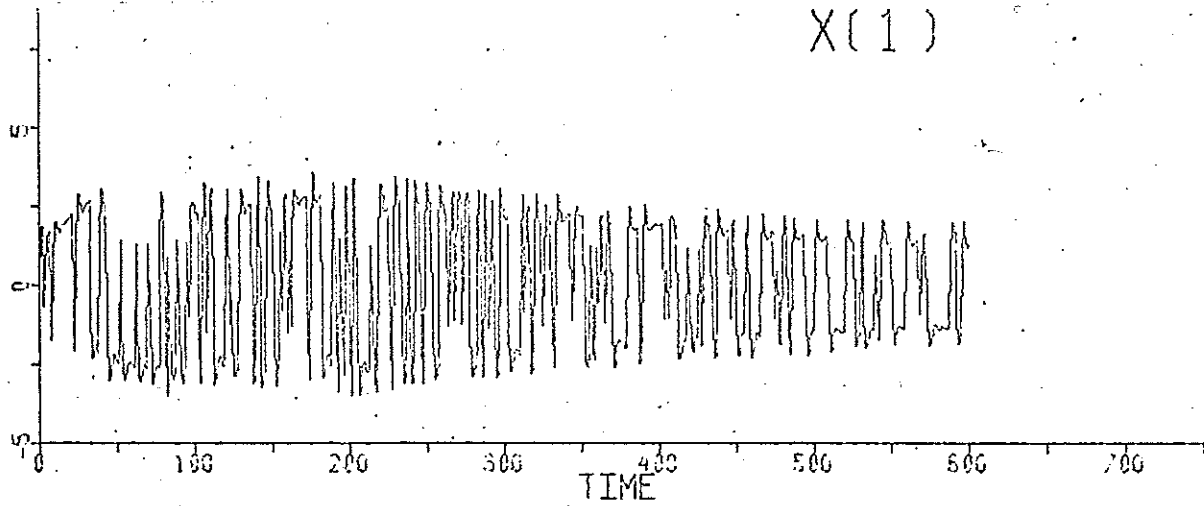
u

U



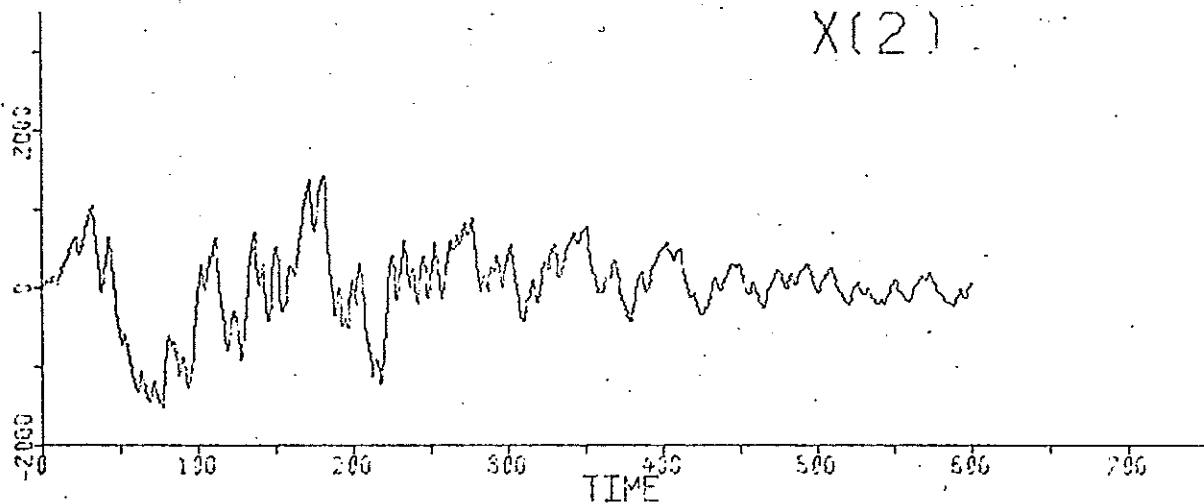
X_1

X(1)



X_2

X(2)



6. Sammanfattning av resultat

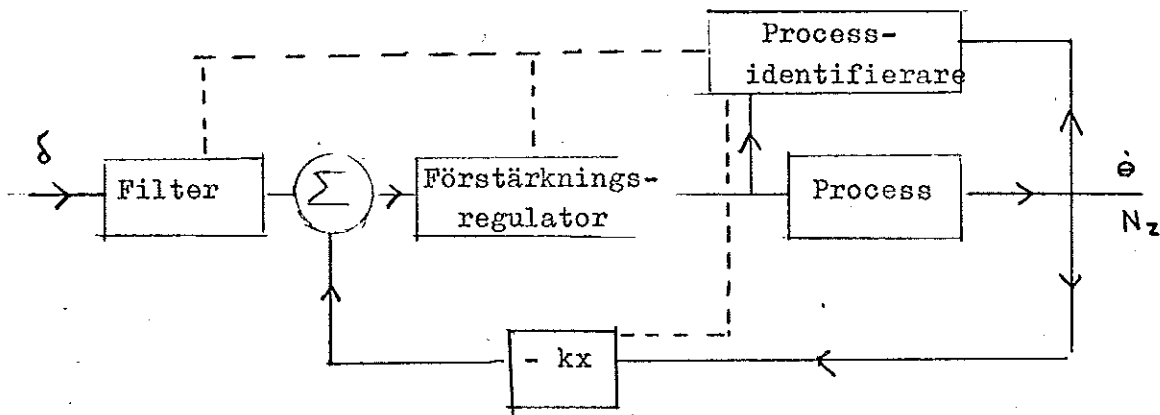
Den här beskrivna metoden att identifiera parametrarna i modellen $x(t+1) = A(t) x(t) + B(t) u(t)$ tycks efter viss "förfining" kunna fås att fungera även på en mer realistisk process.

Gjorda simuleringar på testexempel visar att metoden är

1. långsam, dvs. den kan endast följa tämligen långsamt varierande parametrar
2. känslig för brus på mätsignalerna
3. beroende av att parametrarna och deras variation är av samma storleksordning

Punkt 3 var den direkta anledningen till att skalning infördes. Denna kan ses som ett införande av information till systemet om hur parametrarna i verkligheten varierar. Det förefaller ju rimligt att identifieringen då skulle förbättras, vilket också verifierades. Då det åtminstone efter ett antal samplingtidpunkter tycks kunna gå att identifiera parametrarna tillräckligt bra, bör det gå att

1. återkoppla så att polerna till det fysikaliska systemet får önskat läge
2. bestämma en variabel förstärkning så att det styrda systemets statiska förstärkning blir konstant
3. bestämma nollstället till det fysikaliska systemet, för att därigenom bestämma lämpligt läge för polen i den kaskadkopplade länken. Detta för att det styrda systemet ej skall få för stor översläng.



För att förbättra identifieringen ytterligare måste en noggrannare analys av tänkbara flygfall genomföras. Detta för att få tillgång till "riktigare" vägningsvektorer för processen. En tänkbar möjlighet vore ett antal på förhand bestämda vägningsvektorer, där valet bestäms av någon mätbar storhet hos processen t.ex. acceleration, hastighet, lufttryck etc.

Desstuom bör begränsningar på \hat{A} och \hat{B} införas.

Slutligen skall sägas att det vore intressant att prova metoden på en flygplansmodell. Detta för att vissa parametrar har en helt annan variation hos flygplan och vissa förenklingar hos identifieringsmodellen därför ev. vore möjliga.

Dessutom förutsätter metoden tämligen avancerad digital utrustning, vilket man i allmänhet inte kostar på en robot eftersom instrumenteringen där bör vara billig pga. robotens arbetssätt.

REFERENSER

- [1.] Beckenbach, E : Modern mathematics for engineers,
Mc. GRAW-HILL 1961.
- [2.] Margolis M, Leondes C T, :On the Theory of Adaptive Control
System. The Learning Model approach:
IFAC Moscow 1960, Vol 2 p 556.
- [3.] Skoog H : Examensarbete vid LTH, Re 21 .
- [4.] Schaeperkoetter RL, Transier KG : Parameter identification,
Sperry Rand Engineering Review part 2 1968 p 25.

Appendix 1

Programuppbyggnad.

Under arbetets gång har i huvudsak använts 2 program.
För att någorlunda lätt kunna ändra i programmen har ett antal subrutiner använts.

SUBROUTINE MODEL

Denna subrutin används för att generera nya tillstånds variabler i modellen $x(t+1) = A(t) \cdot x(t) + B(t) \cdot u(t)$.

SUBROUTINE ESTIM

Denna subrutin identifierar parametrarna enligt den tidigare beskrivna metoden (steepest descent)

Förutom dessa har ur programbiblioteket lånats ett antal subrutiner för olika ändamål.

SUBROUTINE RECONK

Denna subrutin bestämmer återkopplingsvektorn k ($u = u_r - k x(t)$), så att polerna får önskat läge.

För generering av insignal har använts PRBS (Pseudo Random Binary Sequence) och RANF (genererar rektangelfördelad stokastisk variabel)

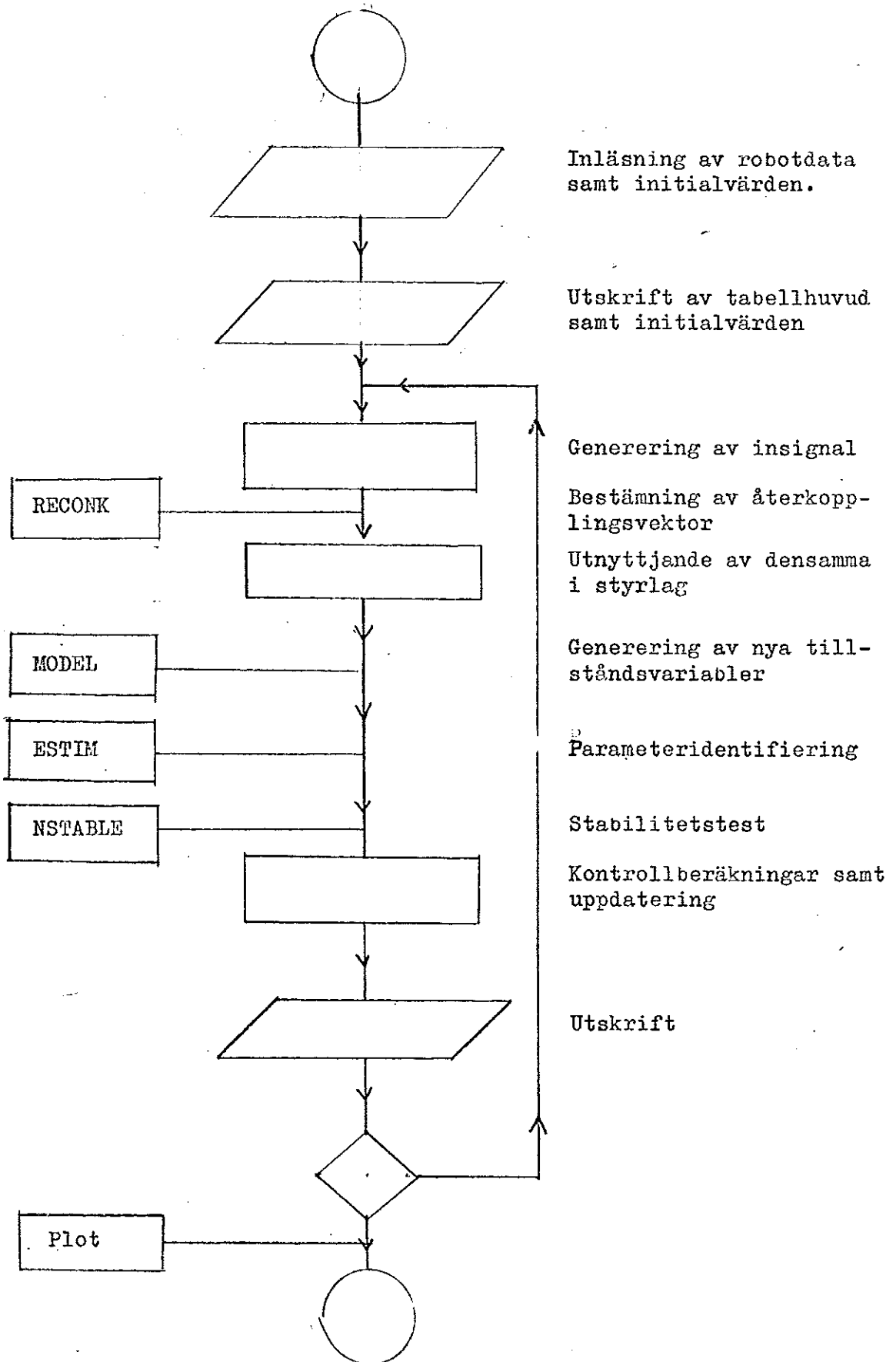
Normalfördelat brus har genererats med hjälp av RANSS

För stabilitets-test har FUNCTION NSTABLE använts.

Vissa beräkningsresultat har dessutom presenterats i form av diagram. Därför har plottningsrutiner använts.

Program ROBTTEST har följande blockstruktur.

Detta program användes vid simuleringar enl . kap. 5.



```

SUBROUTINE MODEL(X,XB,U,A,B,N,M)
DIMENSION A(M,M), B(M), X(M), XB(M), XC(9)
DO 3 I = 1,N
XC(I) = 0
DO 2 J = 1,N
2 XC(I) = XC(I) + A(I,J) * X(J)
3 XC(I)=XC(I)+ B(I)*U
DO 4 K=1,N
XB(K)=X(K)
4 X(K)=XC(K)
RETURN
END

```

X Aktuell tillståndsvariabel
 XB Föregående tillståndsvariabel
 U Insignal
 N Systemordning
 M Dimension hos deklarerade fält
 A, B Systemparametrar

```

SUBROUTINE ESTIM(XN,XB,AE,BE,AK,BK,U,N,XFEL,M)
DIMENSION XFEL(M), XN(M), XB(M),AE(M,M),BE(M)
DO 70 I= 1,N
D=0
DO 60 J=1,N
60 D = D + AE(I,J) * XB(J)
70 XFEL(I) =XN(I) - D- BE(I) * U
DO 90 I = 1,N
DO 80 J = 1,N
80 AE(I,J) = AE(I,J) + AK * XFEL(I) *XB(J)
90 BE(I) = BE(I) + AK *XFEL(I) * U
RETURN
END

```

XN aktuell tillståndsvariabel
 AE resp BE är identifierade parametrarna
 AK Förstärkning hos identifieraren (se ekv. 2.7)
 Denna kunde beräknats inom subrutinen men har här bestämts utanför
 XFEL Felvektor (skrevs ibland ut för kontrolländamål)
 BK "Slaskparameter"

```

SUBROUTINE ESTIM(XN,XR,AE,RE,AK,BK,U,N,XFEL,M)
DIMENSION XFEL(M), XN(M), XB(M),AE(M,M),BE(M)
* AK(M,M), BK(M), OX(9)
DO 70 I= 1,N
D=0.
DO 60 J=1,N
60 D = D + AE(I,J) * XB(J)
70 XFEL(I) =XN(I) - D- BE(I) * U
DO 99 I= 1,N
DO 91 J= 1,N
OX(J) = XB(J) * AK(I,J)
AE(I,J) = AE(I,J) / AK(I,J)
91 CONTINUE
BE(I) = BE(I) / BK(I)
UN = U * BK(I)
S = 0.
DO 92 J = 1,N
92 S = OX(J) **2          + S
S = S + UN**2
GF = 1. / S
DO 80 J = 1,N
80 AE(I,J) = AE(I,J) + GF * XFEL(I) *OX(J)
BE(I) = BE(I) + GF *XFEL(I) * UN
DO 93 J = 1,N
93 AE(I,J) = AE(I,J) *AK(I,J)
BE(I) = BE(I) * BK(I)
99 CONTINUE
RETURN
END

```

Detta är den modifierade versionen av SUBROUTINE ESTIM, i vilken systemparametrarna skalas innanför subrutinen. Skalningsvärdena för A resp. B införes via de formella parametrarna AK resp BK. Förstärkningsfaktorn (2.8) beräknas här innanför subrutinen.

Övriga formella parametrar har samma betydelse som i tidigare beskrivna subrutinen