ETT
INTERAKTIVT PROGRAMPAKET FÖR SIMULERING
AV
SJÄLVINSTÄLLANDE REGULATORER

BENGT CT BENGTSSON
BO EGARDT

RE-141 augusti 1974
Inst.för Reglerteknik
Lunds Tekniska Högskola

ETT INTERAKTIVT PROGRAMPAKET

FÖR SIMULERING AV SJÄLVINSTÄLLANDE REGULATORER

Examensarbete utfört våren 1974 vid
Institutionen för Reglerteknik, LTH

Bengt CT Bengtsson
Bo Egardt

Handledare: Staffan Selander
Torsten Söderström
Björn Wittenmark

SAMMANFATTNING

Denna rapport utgör dokumentation av resultat och erfarenheter som nåtts under ett examensarbete våren 1974.

Iförsta hand redovisas konstruktionen av ett interaktivt programpaket för simulering av vissa typer av adaptiva regulatorer, s.k. självinställande regulatorer. I enlighet med erfarenheterna vid institutionen är paketet kommandostyrt.

I rapporten ingår också en del, som presenterar några regulatoralgoritmer samt resultat av simuleringar med dessa. Bl.a. studeras en regulatortyp med en rekursiv maximum-likelihood-identifiering.

ABSTRACT

This MS thesis report provides documentation of results and experience obtained during the spring 1974.

The main purpose is to show the construction of an interactive program package, which is used for simulations of some special types of adaptive controllers, called self tuning regulators. According to the experience at the division, the program package is command driven.

A part of the report presents some regulator algorithms and the results obtained by simulating these. For example, one type of regulator with a recursive maximum-likelihood-identification is considered.

INNEHÅLLSFÖRTECKNING

I. INLEDNING

Examensarbetet har inriktats på konstruktionen av ett interaktivt programpaket för simulering av s.k. självinställande regulatorer. Programpaketet, som kallas STURE (av Self Tuning Regulators), har gjorts kommandostyrt med tanke på de goda erfarenheterna från bl.a. institutionens identifieringspaket IDPAC.

I programpaketet återfinns bl.a. en ny regulator, kallad STUREM, vilken använder en rekursiv maximum-likelihood-identifiering. För att undersöka egenskaperna för denna regulator redovisas även resultat av simuleringar i rapporten.

I kap II ges först en kortfattad presentation av de regulatorer, som finns implementerade i programpaketet. Programpaketets möjligheter och uppbyggnad visas i kap III och i kap IV jämförs de olika regulatorerna vid simulering på några olika system. Kap V och VI innehåller slutord resp referenser och slutligen ges två appendices, innehållande exempel på programpaketets resultatutskrifter resp listning av programhuvuden.

II. REGULATORERNA

I detta kapitel presenteras kortfattat de tre regulatoralgoritmer, som använts i de jämförande simuleringarna. Det bör kanske påpekas, att regulatorerna genomgående bär namnet STURE, som står för Self Tuning Regulator. Sålunda visas i avsnitt 1 och 2 principerna för de tidigare använda algoritmerna STURE1 resp STURE2, och i avsnitt 3 sammanfattas den nya regulatorn STUREM.
För närmare detaljer hänvisas till litteraturen enl referenslistan.

1. Regulatorn STURE1

Utgå från systemet

$$A^*(q^{-1})y(t) = q^{-(k+1)}B^*(q^{-1})u(t) + \lambda C^*(q^{-1})e(t) \tag{2.1.1}$$

där
$$A^*(q^{-1}) = 1 + a_1q^{-1} + \dots + a_nq^{-n}$$
$$B^*(q^{-1}) = b_1 + b_2q^{-1} + \dots + b_nq^{-(n-1)}$$
$$C^*(q^{-1}) = 1 + c_1q^{-1} + \dots + c_nq^{-n} \quad ,$$

k är systemets tidsfördröjning och e(t) är N(0,1) - fördelat vitt brus. Problemet att identifiera parametrarna i modellen (2.1.1) underlättas väsentligt, om vi antar att c_i=0 för varje i. Genom att utnyttja polynomidentiteten

$$1 = A^*(q^{-1})F^*(q^{-1}) + q^{-(k+1)}G^*(q^{-1}) \tag{2.1.2}$$

där
$$F^*(q^{-1}) = 1 + f_1q^{-1} + \dots + f_kq^{-k}$$
$$G^*(q^{-1}) = g_0 + g_1q^{-1} + \dots + g_{n-1}q^{-(n-1)}$$

kan då (2.1.1) skrivas på formen:

$$y(t+k+1) + \alpha_1 y(t) + \ldots + \alpha_n y(t-n+1) =$$

$$= \beta_0(u(t) + \beta_1 u(t-1) + \ldots + \beta_{n+k-1} u(t-n-k+1)) + \varepsilon(t+k+1) \qquad (2.1.3)$$

där $\varepsilon(t+k+1)$ är en linjärkombination av $e(t+k+1),\ldots, e(t+1)$.
Minimalvariansstrategin är

$$u(t) = \frac{1}{\beta_0}(\alpha_1 + \alpha_2 q^{-1} + \ldots + \alpha_n q^{-(n-1)})y(t) -$$

$$- (\beta_1 q^{-1} + \ldots + \beta_{n+k-1} q^{-(n+k-1)})u(t) \qquad (2.1.4)$$

Den självinställande regulatorns uppgifter är alltså: identifiera parametrarna i (2.1.3) med minsta-kvadratmetoden och styr sedan systemet enl (2.1.4). I allmänhet kan man inte entydigt identifiera samtliga parametrar i (2.1.3) när man applicerar styrlagen (2.1.4) - de parametervärden som erhålles utgör en linearmångfald. För att undvika drivande parametervärden väljes därför parametern β_0 i (2.1.3) fix i STURE1.

STURE1 konvergerar ofta mot en regulator, som ger minimal varians, även om C-polynomet inte är identiskt 1. Om systemet är icke-min-fas, så är det slutna systemet instabilt. Genom att använda olika trick, t.ex. att ge ett större värde på k än vad det verkliga systemet har, kan man utöka klassen av system, där regulatorn fungerar tillfredsställande.

Styrlagen (2.1.4) använder y(t) för beräkning av u(t). Om detta inte är möjligt, utan y(t-1) är det senaste kända värdet när u(t) skall beräknas, modifieras modellen i regulatorn så att den får ett k-värde, som är 1 enhet större än systemets verkliga värde. Regulatorn kan då sägas uppfatta situationen så, att den styr ett system med en enhets större fördröjning men utan fördröjning av styrsignalen.

2. Regulatorn STURE2

Istället för att estimera parametrarna i regulatorn, vilket görs i STURE1, kan man i varje steg identifiera parametrarna i systemet

$$A^*(q^{-1})y(t) = q^{-(k+1)}B^*(q^{-1})u(t) + e(t) \qquad (2.2.1)$$

med minsta-kvadratmetoden. Därefter löses identiteten (2.1.2) vilket ger en optimal styrlag

$$u(t) = -\frac{G^*(q^{-1})}{B^*(q^{-1})F^*(q^{-1})}\, y(t) \qquad (2.2.2)$$

Om denna regulator konvergerar, så konvergerar den mot den optimala regulatorn. Om däremot det faktiska C-polynomet är skilt från 1, konvergerar inte alltid regulatorn.

Man kan visa, att ovanstående styrlag ger ett instabilt slutet system, om systemet är icke-minimum-fas och de estimerade parametrarna skiljer sig det minsta från de verkliga. För att erhålla ett stabilt slutet system kan (2.1.2) ersättas med identiteten

$$B^-(q^{-1})C^*(q^{-1}) = A^*(q^{-1})F^*(q^{-1}) + q^{-(k+1)}G^*(q^{-1})B^{-*}(q^{-1}) \qquad (2.2.3)$$

där alltså $C^*(q^{-1}) = 1$. Polynomen B^- och B^{-*} fås ur uppdelningen

$$B(q) = q^{n-1}B^*(q^{-1}) = B^-(q)B^+(q)$$

där $B^-(q)$ har nollställen utanför enhetscirkeln och $B^+(q)$ innanför. Om B^- är av grad r och B^+ av grad n-r-1 så är F^* av grad k+r, G^* av grad n-1 och slutligen gäller:

$$B^{-*}(q^{-1}) = q^{-r}B^-(q).$$

Den optimala styrlagen ges nu av:

$$u(t) = -\frac{G^*(q^{-1})}{B^{+*}(q^{-1})F^*(q^{-1})}\,y(t) \qquad (2.2.4)$$

Styrlagarna (2.2.2) resp (2.2.4) kan emellertid också beräknas på ett annat sätt. Problemet är: minimera variansen under villkoret att det slutna systemet skall vara stabilt. Detta kan lösas genom att formulera ett s.k. linjärkvadratiskt styrproblem, som leder till en Riccati-ekvation. Denna ansats görs i STURE2-algoritmen. Styrlagen blir densamma som när man använder identiteterna ovan.

För beräkningarna används en tillståndsmodell med följande tillståndsvariabler:

$$\begin{aligned}
x_1(t) &= y(t)\\
x_2(t) &= -a_2y(t-1) - \ldots - a_ny(t-n+1) + b_1u(t-k) +\ldots+ b_nu(t-n-k+1)\\
&\ldots\\
x_k(t) &= -a_ky(t-1) -\ldots- a_ny(t-n+k-1) + b_1u(t-2) +\ldots+ b_nu(t-n-1)\\
x_{k+1}(t) &= -a_{k+1}y(t-1) -\ldots- a_ny(t-n+k) + b_1u(t-1) +\ldots+ b_nu(t-n)\\
&\ldots\\
x_n(t) &= -a_ny(t-1) + b_{n-k}u(t-1) +\ldots+ b_nu(t-k-1)\\
x_{n+1}(t) &= b_{n-k+1}u(t-1) +\ldots+ b_nu(t-k)\\
&\ldots\\
x_{n+k}(t) &= b_nu(t-1)
\end{aligned}$$

Detta ger systemekvationen på tillståndsform:

$$x(t+1) = \begin{array}{r} \\ \\ \\ \\ \\ \\ \\ \\ k\ \mathrm{st}\{ \\ \\ \end{array}\begin{pmatrix} -a_1 & 1 & & & & \\ -a_2 & & 1 & & 0 & \\ \cdot & & & \cdot & & \\ \cdot & & & & \cdot & \\ -a_n & & & & \cdot & \\ 0 & & 0 & & & 1 \\ \cdot & & & & & 1 \\ 0 & & & & & 0 \end{pmatrix} x(t) + \begin{pmatrix} 0 \\ \cdot \\ 0 \\ b_1 \\ b_2 \\ \cdot \\ \cdot \\ b_n \end{pmatrix}\begin{array}{l} \\ \}k\ \mathrm{st} \\ \\ \\ \\ \\ \\ \\ \end{array} u(t) + \begin{pmatrix} 1 \\ 0 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ 0 \end{pmatrix} e(t)$$

dvs $x(t+1) = \Phi x(t) + \Gamma u(t) + Ke(t)$ (2.2.5)

Lösningen av Riccati-ekvationen ger styrlagen $u(t) = -L\hat{x}(t)$ (2.2.6)

där
$$L = \Gamma^T S\Phi\{Q2 + \Gamma^T S\Gamma\}^{-1} \quad (2.2.7)$$

$$S = \Phi^T S\{\Phi - \Gamma L\} + Q1 \quad (2.2.8)$$

Denna styrlag ger $\min E\{\Sigma(x^T Q1x + u^T Q2u)\}$,
dvs vi får $\min E\{\Sigma y^2\}$ om $Q1 = \mathrm{diag}(1\ 0\ .\ .\ 0)$ och $Q2 = 0$.

Vid simuleringarna har för jämförelsens skull alla regulatorer använts med fördröjning, dvs när $u(t)$ skall beräknas har vi inte tillgång till $y(t)$. För STURE2 innebär detta, att styrlagen (2.2.6) ersätts med:

$u(t) = -L\hat{x}(t|t-1)$ (2.2.9)

där $\hat{x}(t|t-1)$ är tillståndsvektorn vid tidpunkten t, predikterad ett steg framåt.

3. Regulatorn STUREM

Med STURE2-algoritmen kan regulatorn konvergera mot en icke-optimal regulator om C-polynomet inte är identiskt 1. Detta beror på att minsta-kvadrat-metoden inte estimerar c-parametrarna. För att göra detta används i STUREM en annan identifieringsmetod - en rekursiv, approximativ maximum-likelihood-identifiering. Tillståndsmodellen, som används i STURE2, måste nu modifieras och innehålla även c-parametrarna.

Systemet vi betraktar är:

$$(1 + a_1q^{-1} + \ldots + a_nq^{-n})y(t) = q^{-(k+1)}(b_1 + b_2q^{-1} + \ldots + b_nq^{-(n-1)})u(t) +$$
$$+ (1 + c_1q^{-1} + \ldots + c_nq^{-n})e(t) \quad (2.3.1)$$

Inför nu följande tillståndsvariabler:

$$x_1(t) = y(t) - e(t)$$

$$x_2(t) = -a_2y(t-1) - \ldots - a_ny(t-n+1) + b_1u(t-k) + \ldots + b_nu(t-n-k+1) +$$
$$+ c_2e(t-1) + \ldots + c_ne(t-n+1)$$

. . .

$$x_k(t) = -a_ky(t-1) - \ldots - a_ny(t-n+k-1) + b_1u(t-2) + \ldots + b_nu(t-n-1) +$$
$$+ c_ne(t-1) + \ldots + c_ne(t-n+k-1)$$

$$x_{k+1}(t) = -a_{k+1}y(t-1) - \ldots - a_ny(t-n+k) + b_1u(t-1) + \ldots + b_nu(t-n) +$$
$$+ c_{k+1}e(t-1) + \ldots + c_ne(t-n+k)$$

. . .

$$x_n(t) = -a_ny(t-1) + b_{n-k}u(t-1) + \ldots + b_nu(t-k-1) + c_ne(t-1)$$

$$x_{n+1}(t) = b_{n-k+1}u(t-1) + \ldots + b_nu(t-k)$$

. . .

$$x_{n+k}(t) = b_nu(t-1)$$

Systemet kan då skrivas på standardformen:

$$x(t+1) = \begin{array}{r} \\ \\ \\ \\ \\ \\ \\ k \text{ st}\{ \\ \\ \end{array}\begin{pmatrix} -a_1 & 1 & & & & \\ -a_2 & & 1 & & 0 & \\ \cdot & & & \cdot & & \\ \cdot & & & & \cdot & \\ -a_n & & & & & \cdot \\ 0 & & 0 & & & 1 \\ \cdot & & & & & 1 \\ 0 & & & & & 0 \end{pmatrix} x(t) + \begin{pmatrix} 0 \\ \cdot \\ 0 \\ b_1 \\ b_2 \\ \cdot \\ \cdot \\ b_n \end{pmatrix}\begin{array}{l} \\ \}k \text{ st} \\ \\ \\ \\ \\ \\ \\ \end{array} u(t) + \begin{pmatrix} c_1-a_1 \\ c_2-a_2 \\ \cdot \\ \cdot \\ c_n-a_n \\ 0 \\ \cdot \\ 0 \end{pmatrix} e(t) \begin{array}{l} \\ \\ \\ \\ \\ \\ \}k \text{ st} \\ \\ \end{array}$$

$$y(t) = (1\ 0\ .\ .\ .\ 0)x(t) + e(t)$$

dvs:
$$\begin{cases} x(t+1) = \Phi x(t) + \Gamma u(t) + Ke(t) \\ y(t) = Cx(t) + e(t) \end{cases} \qquad (2.3.2)$$

STUREM har endast realiserats med en tidsfördröjning i regulatorn. Styrlagen får då, som nämndes i avsnitt 2, utseendet:

$$u(t) = -L\hat{x}(t|t-1)$$

Den predikterade tillståndsvektorn $\hat{x}(t|t-1)$ kan erhållas med vanlig Kalmanfiltrering:

$$\hat{x}(t+1|t) = \hat{\Phi}\hat{x}(t|t-1) + \hat{\Gamma}u(t) + \hat{K}(y(t) - \hat{C}\hat{x}(t|t-1)) \qquad (2.3.3)$$

Emellertid ger identifieringsalgoritmen resiudalen som delresultat och (2.3.3) kan ersättas med:

$$\hat{x}(t+1|t) = \hat{\Phi}\hat{x}(t|t-1) + \hat{\Gamma}u(t) + \hat{K}\hat{e}(t) \qquad (2.3.4)$$

Denna prediktering har nackdelen att $\hat{x}(t|t-1)$ beror på gamla parameterskattningar, och därför skrives (2.3.4) om till den slutliga versionen: (se nästa sida)

$$
\hat{x}(t+1|t) = \begin{bmatrix} -\hat{a}_1 & \cdot & \cdot & \cdot & -\hat{a}_n & 0 & \cdot & 0 \\ -\hat{a}_2 & & & \cdot & & & & \\ \cdot & & \cdot & & & & & \\ \cdot & \cdot & & & & & & \\ -\hat{a}_n & & & & & & & \\ 0 & & & & 0 & & & \\ \cdot & & & & & & & \\ 0 & & & & & & & \end{bmatrix} \begin{bmatrix} y(t) \\ y(t-1) \\ \cdot \\ \cdot \\ y(t-n+1) \\ 0 \\ \cdot \\ 0 \end{bmatrix} +
$$

$$
+ \begin{bmatrix} & & & \hat{b}_1 & \cdot & \cdot & \cdot & \hat{b}_n \\ 0 & & \cdot & & & & \cdot & \\ & \cdot & & & & \cdot & & \\ \hat{b}_1 & \cdot & \cdot & \cdot & \hat{b}_n & & & \\ \hat{b}_2 & \cdot & \cdot & \hat{b}_n & & & & \\ \cdot & & \cdot & & & 0 & & \\ \cdot & \cdot & & & & & & \\ \hat{b}_n & & & & & & & \end{bmatrix} \begin{bmatrix} u(t) \\ \cdot \\ \cdot \\ u(t-k) \\ \cdot \\ \cdot \\ \cdot \\ u(t-n-k+1) \end{bmatrix} +
$$

$$
+ \begin{bmatrix} \hat{c}_1 & \cdot & \cdot & \cdot & \hat{c}_n & 0 & \cdot & 0 \\ \hat{c}_2 & & & \cdot & & & & \\ \cdot & & \cdot & & & & & \\ \cdot & \cdot & & & & & & \\ \hat{c}_n & & & & & & & \\ 0 & & & & 0 & & & \\ \cdot & & & & & & & \\ 0 & & & & & & & \end{bmatrix} \begin{bmatrix} \hat{e}(t) \\ \hat{e}(t-1) \\ \cdot \\ \cdot \\ \hat{e}(t-n+1) \\ 0 \\ \cdot \\ 0 \end{bmatrix} \qquad (2.3.5)
$$

III. PROGRAMPAKETET

I detta kapitel presenteras kortfattat programpaketet STURE med avseende på programvaran. I avsnitt 1 ges en inledande översikt av paketet och dess kommandon. Avsnitt 2 behandlar paketet så som operatören ser det, medan avsnitt 3 behandlar programmens struktur.

1. Inledning

STURE är ett interaktivt programpaket för simulering av självinställande regulatorer, skrivet för institutionens processdator PDP 15. Paketet är huvudsakligen kommandostyrt. Bl.a. STURE:s förebild - institutionens identifieringspaket IDPAC (se {6},{7}) - har visat, att detta är mycket värdefullt, då det ger operatören mycket stor handlingsfrihet. Ett program styrt enligt principen frågor-och-svar har dock fördelen, att man inte kan glömma någonting. Vidare blir skrivarbetet minimalt, då inget kommando behöver anges. STURE har därför vissa inslag av frågor-och-svar samt ett kommando som lotsar operatören igenom alla kommandon, som är nödvändiga för att tilldela alla parametrar ett värde.

Paketet användes för att simulera verkan av STURE1, STURE2 och STUREM (se kap II) på linjära system störda av tidsinvariant brus. STURE1 och STURE2 kan köras med eller utan fördröjd styrsignal. Resultatet kan redovisas på radskrivare eller bildskärm. Vidare kan parametrar ändras under pågående simulering, kommentarer skrivas ut på valfritt medium och parametervärden listas på bildskärmen.

På följande sidor finns en lista över möjliga kommandon. Denna lista utgör en fil, INFO, och kan alltså lätt skrivas ut. Det framgår av listan att värdena på vissa parametrar kan ändras genom direkta kommandon. Dessa parametrars namn förklaras också i listan.

```
STURE - COMMAND STRUCTURE
*************************

GUIDE         AUTOMATICALLY MAKES THE COMMANDS SYST,REG,NOISE,YREF,INIT
-----
SYST          ASKS FOR THE SYSTEM PARAMETERS
----
REG           ASKS FOR THE REGULATOR PARAMETERS
---
NOISE         ASKS FOR THE NOISE PARAMETERS
-----
YREF          ASKS FOR THE REFERENCE PARAMETERS
----
INIT          ASKS FOR INITIAL VALUES OF THE PARAMETER ESTIMATES
----
SAVE VNAME1 [VNR11..] [VNAME2 [VNR21..]..]
------------------------------------------
              THE VARIABLES VNAME1(VNR11).. WILL BE SAVED ON THE
              DISK IN THE FOLLOWING SIMULATIONS
              VARIABLES:
                 A[LPHA] 1   -  A[LPHA] 6
                 B[ETA] 1    -  B[ETA] 6
                 C 1         -  C 6   ('GAMMA'='C')
                 LOSS
                 Y
                 U
                 YSUM
                 YREF
                 RYY 1       -  RYY 10
                 RYU 0       -  RYU 10
                 P 1 1       -  P 18 18  (P IS THE COVARIANCE MATRIX)

              NB.   MAXIMUM NUMBER OF SAVED VARIABLES:15

DISP          DISPLAYS THE PARAMETER VALUES
----
SIMU N        STARTS A NEW SIMULATION WITH N SAMPLE POINTS
------        THE SIMULATION IS STOPPED WHEN DATA SWITCH 0 IS SET

CONT N        CONTINUES THE SIMULATION FOR N SAMPLE POINTS
------        CONT IS NOT ALLOWED AFTER A NEW SAVE-COMMAND
              OR WHEN THE TYPE OF REGULATOR OR NA,NB,NC HAS BEEN CHANGED

RCALL         ELIMINATES THE EFFECTS OF SAVE-COMMANDS,CHANGES OF
-----         REGULATOR TYPE OR NA,NB,NC FOLLOWING THE LAST SIMULATION.
              CONT IS ALLOWED AFTER RCALL

[A]PLOT [VNAME1 [VNR11..]..] [SC YMIN1 YMAX1]
---------------------------------------------
              PLOTS THE VARIABLES VNAME1(VNR11)..
              IF NO VARIABLE NAMES ARE TYPED ALL SAVED VARIABLES
              WILL BE PLOTTED
              SCALES:  APLOT                      AUTOMATIC SCALING
                       PLOT ... SC YMIN1 YMAX1    YMIN1 MINIMUM VALUE
                                                  YMAX1 MAXIMUM VALUE
                       PLOT ...                   MINIMUM AND MAXIMUM VALUES
                                                  ARE GIVEN BY THE  VARIABLES
                                                  YMIN AND YMAX RESP (BELOW)
                                                  YMIN=YMAX GIVES
                                                  AUTOMATIC SCALING
```

PRINT N1 N2 — PRINTS INFORMATION ABOUT THE PRESENT SIMULATION FOLLOWED BY THE VALUES OF THE SAVED VARIABLES IN THE FIRST N1 POINTS AND THEN IN EVERY N2:TH POINT. IF CONT COMMANDS HAVE BEEN USED CHANGED PARAMETERS ARE PRINTED AND THE FIRST N1 POINTS OF THE CONTINUED SIMULATION ARE PRINTED FOLLOWED BY EVERY N2:TH POINT ETC

TEXT DEV ['FF']

THE FOLLOWING LINES TYPED ON THE TELETYPE ARE PRINTED ON MEDIUM DEV ('LP','TV' OR 'TT') [AFTER FORM FEED]. THE TRANSFER OF LINES IS STOPPED WHEN THE LINE TERMINATOR IS ALT MODE

STOP — EXECUTION OF THE PROGRAM PACKAGE IS TERMINATED

CHANGE OF A SINGLE PARAMETER VALUE:

KSYS N	TIME DELAY OF THE SYSTEM [0,10]
NA N	NUMBER OF A PARAMETERS IN THE REGULATOR MODEL [0,6]
NB N	NUMBER OF B PARAMETERS IN THE REGULATOR MODEL [0,6]
NC N	NUMBER OF C PARAMETERS IN THE REGULATOR MODEL [0,6]
KREG N	TIME DELAY OF THE REGULATOR MODEL [0,10]
KDEL N	DELAY OF THE REGULATOR SIGNAL [0,1]
RL R	EXPONENTIAL WEIGHTING FACTOR]0,1]
B0 R	PARAMETER BETA0 IN STURE1 .NE.0
P0 R	INITIAL VALUE OF THE COVARIANCE MATRIX IS P0*I >=0
ULIM R	LIMIT ON THE CONTROL SIGNAL. ULIM < 0: NO LIMIT
R0 R	RADIUS OF THE POLE-ENCLOSING CIRCLE]0,1]
Q2 R	PENALTY ON THE CONTROL SIGNAL >=0
ITER N	MAXIMUM NUMBER OF RICCATI ITERATIONS >=1
RLIM R	LIMIT ON THE ESTIMATED RESIDUALS (STUREM) >=0
RLEXP R	TIME DEPENDENCE OF RL (STUREM) [0,1]
YMIN R	MINIMUM VALUE OF THE Y SCALE
YMAX R	MAXIMUM VALUE OF THE Y SCALE

2. Paketet ur operatörens synvinkel

Operatören har flera olika kommandon till sitt förfogande. I föregående avsnitt gavs en lista med kortfattade beskrivningar av kommandona. Nedan beskrivs varje kommando något mer detaljerat.

Inhämtningskommandon

(i anslutning till varje kommando ges en kopia på de frågor, som efterhand kommer upp på bildskärmen)

GUIDE går automatiskt igenom kommandona SYST, REG, NOISE, YREF och INIT.

SYST frågar efter systemets parametrar:

```
S Y S T
=======

NORD (ORDER OF SYSTEM)

KSYS (TIME DELAY OF SYSTEM)

APOLYNOMIAL ( 3 NUMBERS!)

BPOLYNOMIAL ( 3 NUMBERS!)

CPOLYNOMIAL ( 3 NUMBERS!)
```

NORD (systemets ordning) får vara högst 5.

De tre polynomen innehåller koefficienterna a_i, b_i och c_i i modellen (2.1.1).

REG frågar efter regulatortyp och för denna regulator relevanta parametrar. För STURE1 är frågorna följande:

```
R E G
=====

TYPE OF REGULATOR    0:NONE (U=0)
                     1:STURE1
                     2:STURE2
                     3:STUREM

KREG (TIME DELAY OF MODEL)

KDEL (DELAY OF REGULATOR SIGNAL)

NA NB (NUMBER OF ALPHA AND BETA PARAMETERS)

B0 (PARAMETER BETA0)

ULIM (LIMIT ON THE CONTROL SIGNAL)
     <0: NO LIMIT
     >0: LIMIT

RL (EXPONENTIAL WEIGHTING FACTOR)

P0 (INITIAL VALUE OF THE COVARIANCE MATRIX IS P0*I)
```

Frågorna för STURE2 och STUREM är nästan desamma, så endast de för STUREM visas. För STURE2 blir det följande skillnader: NC, RLEXP och RLIM utgår samt KDEL tillkommer.

```
R E G
=====

TYPE OF REGULATOR     0:NONE (U=0)
                      1:STURE1
                      2:STURE2
                      3:STUREM

KREG (TIME DELAY OF MODEL)

NA NB NC (NUMBER OF ALPHA, BETA AND GAMMA PARAMETERS)

R0 (RADIUS OF CIRCLE ENCLOSING ALL POLES)

Q2 (PENALTY ON CONTROL SIGNAL)

RLEXP (TIME DEPENDENCE OF RL)

RLIM (LIMIT ON THE RESIDUALS)

ULIM (LIMIT ON THE CONTROL SIGNAL)
    <0: NO LIMIT
    >0: LIMIT

RL (EXPONENTIAL WEIGHTING FACTOR)

P0 (INITIAL VALUE OF THE COVARIANCE MATRIX IS P0*I)
```

NOISE frågar efter startvärdet till brusgeneratorn samt önskad standardavvikelse:

```
N O I S E
=========

MAC0 (ODD INTEGER FOR MCNODI)

DEVIA (STANDARD DEVIATION OF NOISE)
```

YREF frågar efter typ av referenssignal samt amplitud och period för denna:

```
Y R E F
=======

REFERENCE:  1: CONSTANT
            2: SQUARE WAVE
            3: TRIANG WAVE
            4: SINUSOIDAL

AMPLITUDE

PERIOD (INTEGER)
```

INIT frågar efter initialvärden på parameterestimaten:

```
I N I T
=======

ALPHA  (INITIAL VALUES OF ALPHA PARAMETERS)   2 NUMBERS!

BETAO  (INITIAL VALUES OF BETA PARAMETERS)    3 NUMBERS!

GAMO   (INITIAL VALUES OF GAMMA PARAMETERS)   2 NUMBERS!
```

För samtliga kommandon ovan gäller att om man besvarar en fråga med enbart Carriage Return antages tidigare värde gälla.

Sparande

SAVE Ex. SAVE A 1 2 LOSS Y U

Angivna variabler sparas på skivan i följande simuleringar. Högst 15 variabler får sparas. I kommandotabellen anges vilka variabler som kan sparas.

A, B och C är parameterestimaten.

LOSS är förlustfunktionen Σy^2.

Y, U, YSUM och YREF är utsignal, styrsignal, kumulerad utsignal samt referenssignal.

RYY N och RYU N är $\Sigma y(t)y(t-N)$ resp $\Sigma y(t)u(t-N)$.

P A B är element (A,B) i kovariansmatrisen P.

Om mer än en variabel i ett fält skall sparas, räcker det att ange fältnamnet en gång. Ex. SAVE P 1 1 12 15 åstadkommer att P(1,1) och P(12,15) sparas.

Fortsättning av en påbörjad simulering (se nedan) är inte tillåtet om ett nytt SAVE-kommando gjorts (eller om regulatortyp,NA, NB eller NC ändrats). För att upphäva verkan av dessa ändringar finns kommandot

RCALL efter vilket fortsatt simulering med de tidigare värdena på ovan nämnda parametrar är tillåtet.

Simulering

SIMU Ex. SIMU 500. En simulering i angivet antal punkter utföres. Tidigare gjorda simuleringar går förlorade.

Om dataswitch 0 på operatörspanelen sättes eller om utsignalens absolutbelopp någon gång överstiger 10^{70} avbryts simuleringen. Härvid räddas automatiskt erhållna resultat, så att förloppet kan studeras eller, om avbrottet berodde på switchen, simuleringen kan fortsättas.

CONT Ex. CONT 500. En tidigare påbörjad simulering fortsättes med slutvärdena från föregående simulering som startvärden. Bortsett från utskriftens typografi (se PRINT nedan) blir alltså resultatet detsamma av SIMU 1000 som av t.ex. SIMU 500 följt av CONT 500, såvida inga parametrar ändrats mellan dessa kommandon. Bortsett från vad som nämndes i anslutning till RCALL ovan får alla parametrar ändras mellan SIMU och CONT. Efter SIMU får följa ett godtyckligt antal CONT. Beträffande avbrytning av en simulering, se SIMU.

Plottning och utskrift

[A]PLOT Ex. APLOT A 1 B 1 2

PLOT U Y SC -10 11.5

PLOT LOSS

Angivna variabler presenteras grafiskt på bildskärmen. Alla sparade variabler kan plottas. Om inga variabler anges plottas samtliga sparade variabler. Reglerna för variabelnamnen är identiska med de som gäller för SAVE.

Skalor: APLOT ger automatisk skalning.

PLOT ... SC R1 R2 ger R1 och R2 som min- resp maxvärde på abscissan.

PLOT ... använder som min- och maxvärden de speciella parametrarna YMIN och YMAX (se kommandotabellen). Om dessa är lika fås automatisk skalning.

Variablerna plottas i den ordning som de ges i kommandot, utom U som plottas sist i histogramform. Förtydligande text kan skrivas in genom att i kommandot efter " skriva önskad kommentar. I nästa kapitel finns smakprov på plottningar.

PRINT Ex. PRINT 10 25 ger utskrift av de 10 första punkterna och därefter var 25:e punkt. Samtliga sparade variabler skrivs ut. Om N1 och N2 (se kommandotabellen) inte båda är 0 skrivs den sista punkten alltid ut. N2=0 har samma effekt som ett oändligt stort N2. I appendix 1 ges exempel på utskrift från PRINT. Observera att ändrade parametrar skrivs ut efterhand samt att förloppet med N1 och N2 omstartas efter varje CONT.

Direkt åtkomliga parametrar

Genom kommandon av typ NA 3 kan 17 olika parametrar åsättas värden.(se kommandotabellen). Här intar ITER, YMIN och YMAX en särställning därigenom att de till skillnad från övriga parametrar inte finns med i SYST eller REG, utan kan ändras enbart med "direktkommando". Observera att ITER får värdet 10 om operatören inte anger något annat.

Övriga kommandon

DISP ger utskrift på bildskärmen av för tillfället intressanta parametrar. Exempel på utskrift framgår av nästa sida.

TEXT Kommando: TEXT DEV 'FF'
Ger utskrift på angivet medium av följande rader skrivna på teletypen. DEV kan vara TT, TV eller LP beroende på om utskriften önskas på teletypen,bildskärmen eller radskrivaren. FF anger att utskriften skall börja på ny sida. Utskrift på teletypen sker endast under exekvering av en MACRO. Överföringen avslutas när ALT MODE påträffas.

STOP avslutar körningen.

Programpaketet är dessutom utrustat med faciliteter för generering och exekvering av "superkommandon", s.k. MACRO:s. Beskrivning av dessa återfinns i {6} och {7}.

```
SYSTEM AND REGULATOR PARAMETERS (STUREM)

NORD         3
KSYS         0
APOL        -1.7000    0.7000    0.0050
BPOL        -1.0000    2.3000   -1.3000
CPOL         0.0000    0.0000    0.0000
KREG         0
NA           2
NB           3
NC           2
R0           1.0000
Q2           0.0100
ALPH0       -1.7000    0.7000
BETA0       -1.0000    2.3000   -1.3000
GAM0         0.0000    0.0000
ULIM        -1.0000   (NO LIMIT)
P0           0.0000
RL           0.9900
RLEXP        0.0100
RLIM        10.0000
ITER        10

MISCELLANEOUS

YREF          TRIANG WAVE          MAC0       19                YMIN     -3.0000
YAMPL       10.0000                DEVIA       1.0000           YMAX      3.0000
IYPER      200

SAVED VARIABLES:
   A 1       A 2       B 1       B 2       B 3       C 1       C 2    LOSS     U(T)
  Y(T)      P 1.1     P 2.2
```

Exempel på utskrift av DISP

3. Programstruktur

Paketet består huvudsakligen av följande program:

Huvudprogrammet:	MAIN
Program, som frågar efter parametervärden:	BEGIN, GUIDE, SYST, REG, NOISE, YREF och INIT
Program för plottning och utskrift:	PLOCOM, PLOT, ADPRI, PRINT, DISP och TEXT
Program för simulering:	SIMU, SYSTEM, CORI och INPOL
Regulatorerna:	STURE1, STURE2, STURE3 och STUREM.(STURE2 och STURE3 är samma regulator med resp utan fördröjning av styrsignalen)
Identifieringsalgoritmerna:	RTLSID och RTMLE
Program för kommandoavkodning o dyl:	INTRAC, MACHDL, RESEX, WBUFF, ARGIN, COMLIN, SETPAR och GETIN
Program för stränghantering o dyl:	CODE, DECODE, NUMER, GAC, PAC, GETCHA och PUTCHA.
Programmet för utskrift av felmeddelanden:	ERROR

Programmen har skrivits med avsikten att programhuvud och övriga kommentarer skall räcka för att förstå funktionen. Programhuvudena har listats i appendix 2.
Informationsflödet sker huvudsakligen över COMMON-block. Dessa är listade med vidare referenser i MAIN:s programhuvud.
Fig 3.3.1 ger en grov bild av vilka program som anropas vid de olika kommandona. Observera att flera andra program är aktiva vid många tillfällen. Det skulle inta vara möjligt att åstadkomma ett blockschema som var både fullständigt och läsbart.

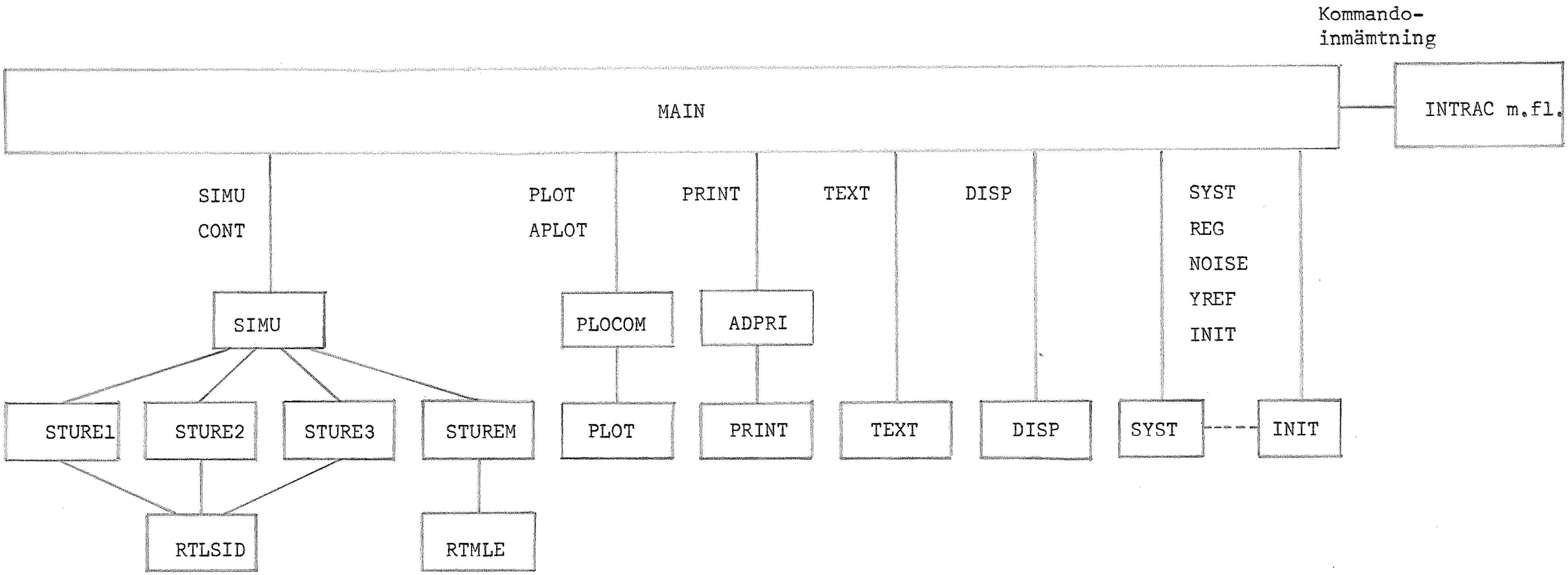


Fig 3.3.1 Programstruktur

I några situationer måste "tjuvknep" tillgripas. Fig 3.3.2 visar gången vid simulering.

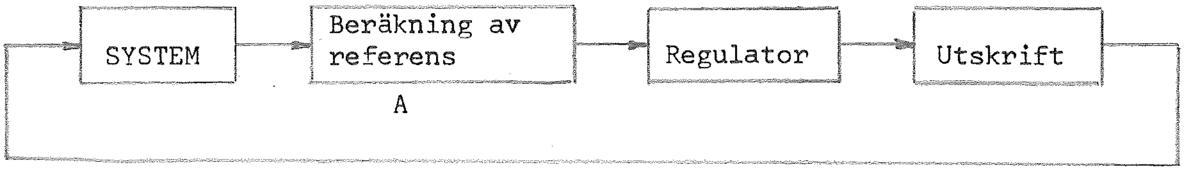


Fig 3.3.2

Programmet SYSTEM, som simulerar det verkliga systemet, är skrivet så, att y(t) beräknas för givet u(t). Regulatorn STURE1 är vidare skriven för att ge en tidsfördröjning i regulatorn. För att beräkningen av referenssignalen skall stämma måste vid A i fig 3.3.2 vektorn Y ha y(t) som första element. Följande situationer uppstår nu:

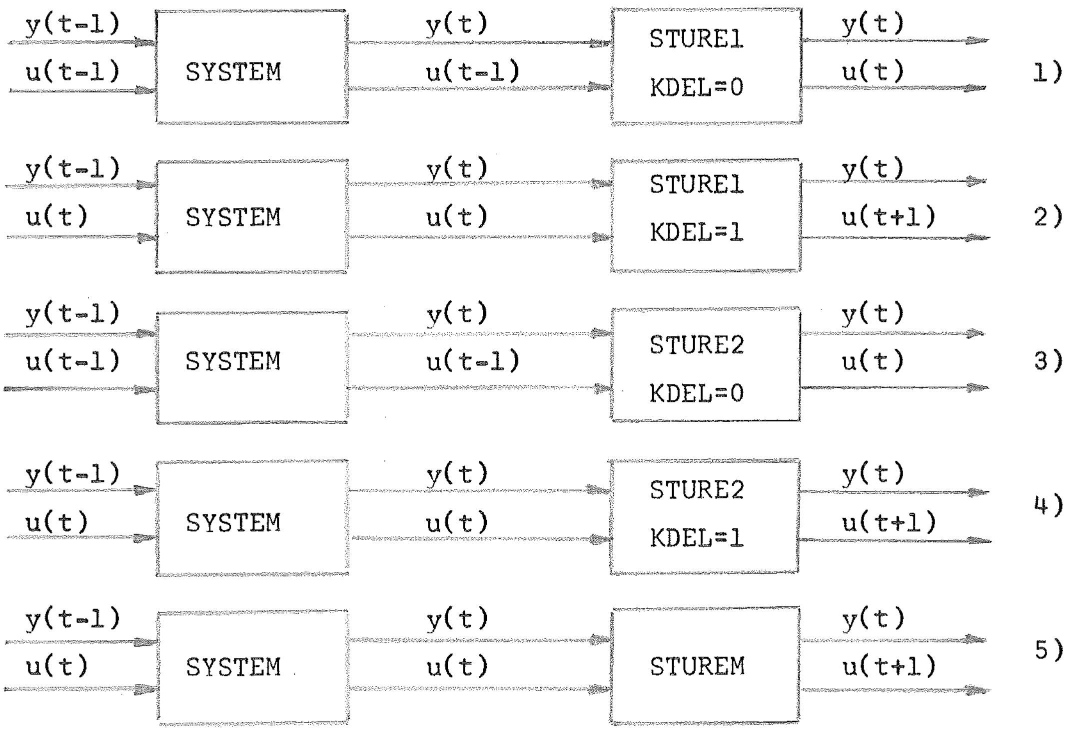


Fig 3.3.3

I fallen 2), 4) och 5) måste vid utskrift samt vid beräkning av RYU andra elementet i U-vektorn användas i stället för första.
I fallen 1) och 3) däremot består problemet i att indata till SYSTEM inte är i enlighet med vad som nämndes ovan. Genom att ge SYSTEM ett värde på systemets fördröjning som är ett mindre än den verkliga, får man önskat uppträdande. Eftersom vidare samma program används för STURE1 i fallen 1) och 2) måste i fall 1) även regulatorn få uppgift om en tidsfördröjning som är ett mindre än den verkliga. Samtliga dessa åtgärder görs automatiskt på programnivå.

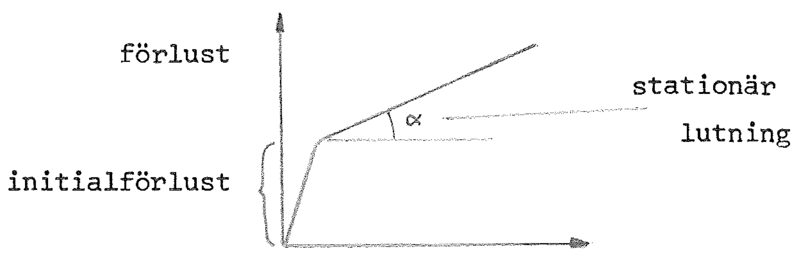
IV. SIMULERINGAR

I detta kapitel redovisas de resultat som erhållits vid simulering av de tre regulatorerna applicerade på några olika system. Särskilt uppmärksammas naturligtvis egenskaperna för den nya regulatorn STUREM. I avsnitt 1 ges en inledande presentation av förutsättningar, beteckningar m.m.. För fyra olika system ges i avsnitt 2-5 teoretiska beräkningar, kommentarer om förväntat resultat och resultat från simulering. I avsnitt 6 ges slutligen en sammanfattning av gjorda erfarenheter.

1. Inledning

Trots att i programpaketet har implementerats både STURE1 och STURE2 också utan fördröjning i regulatorn, har samtliga simuleringar gjorts med fördröjning i regulatorn. Detta för att ge en "rättvis" jämförelse med STUREM, som endast finns med fördröjning.

Vid simuleringarna ägnas intresse framför allt åt två detaljer - parameterestimat och förlustfunktion. Den förlustfunktion, som presenteras av programmet är definierad som $\Sigma y^2(s)$, där y är utsignalen från systemet. I allmänhet har förlusten följande förlopp:



De intressanta värdena är här dels lutningen α, som anger hur stor förlusten är per samplingsintervall i stationärt tillstånd, dels initialförlusten. Om inget annat anges, redovisas därför endast dessa två värden, och bara i undantagsfall visas förlustfunktionens tidsförlopp.

Vid simuleringarna har framkommit misstanken, att det i bruset, som genereras av en biblioteksrutin MCNODI, finns kovarianser, som är skilda från 0. Detta skulle kunna förklara, att simuleringarna ofta uppvisar en stationär medelförlust, som t.o.m. är något lägre än den teoretiskt optimala.

Några kommentarer skall också ges angående de parametervärden, som använts. Förklaring av samtliga använda parameternamn ges i kap III. De parametrar, som framför allt kommer att varieras, är:

P0 - initialvärdet av kovariansmatrisen i identifieringen är P0•I, där I är enhetsmatrisen

ULIM - begränsning av styrsignalen

Dessa ändringar kommer att anges i varje fall, medan däremot följande parametrar är oförändrade genom nästan alla simuleringarna:

DEVIA = 1	
MACO = 19	
B0 = 1	(STURE1)
KDEL = 1	(fördröjning i regulatorn - se ovan)
R0 = 1	(STURE2,STUREM)
Q2 = 0	(se avsnitt II.2)
RL = 0.99	
RLEXP = 0.01	(STUREM)
RLIM = 10	(STUREM)
ITER = 10	(STURE2,STUREM)

Avviker värdena i någon simulering från ovanstående, påpekas detta särskilt.

För de flesta exemplen har simuleringarnas längd varit 1000 punkter, vilket i allmänhet ger en god bild av uppförandet för regulatorn. I speciella fall har dock längre simuleringar gjorts.

I de följande avsnitten presenteras de tre regulatorernas inverkan på fyraolika system. I varje avsnitt visas först hur min-variansregulatorn ser ut för systemet, därefter resultat av reglering med resp STURE-algoritm och slutligen ges en sammanställning av resultaten.

2. Exempel 1

Vi inleder med att betrakta systemet

$$(1 + aq^{-1})y(t) = q^{-1}bu(t) + (1 + cq^{-1})e(t) \tag{4.2.1}$$

med $a=-0.95$, $b=1$, $c=-0.5$.

Systemet är ett stabilt min-fas-system av 1:a ordningen, och den optimala regulatorn för detta system kan enl avsnitt II.2 erhållas om följande identitet löses:

$$C^*(q^{-1}) = A^*(q^{-1})F^*(q^{-1}) + q^{-(k+1)}G^*(q^{-1}) \tag{4.2.2}$$

För systemet (4.2.1) blir identiteten (observera att fördröjningen i regulatorn ökar k-värdet med 1):

$$1 + cq^{-1} = (1 + aq^{-1})(1 + fq^{-1}) + q^{-2}g \quad \text{som ger}$$

$$\begin{cases} f = c-a = 0.45 \\ g = -af = -a(c-a) = 0.4275 \end{cases}$$

Styrlagen är:

$$u(t) = -\frac{G^*(q^{-1})}{B^*(q^{-1})F^*(q^{-1})}y(t-1) = \frac{a(c-a)}{b(1 + (c-a)q^{-1})}y(t-1) =$$

$$= \frac{\frac{a}{b}(c-a)}{1 + (c-a)q^{-1}}y(t-1) = -\frac{0.4275}{1 + 0.45q^{-1}}y(t-1) \tag{4.2.3}$$

Utsignalens varians erhålles med hjälp av polynomet F^*:

$Ey^2 = 1 + f^2 = 1.2025$

Vi skall nu studera hur de olika regulatorerna klarar att reglera systemet (4.2.1).

STURE1

Regulatorn estimerar här parametrarna i styrlagen (4.2.3), men då bruset inte är okorrelerat (dvs C-polynomet inte identiskt 1) kan vi inte vänta oss, att estimaten skall konvergera mot de optimala värdena i (4.2.3), dvs $\alpha_1 = -0.4275$, $\beta_1 = 0.45$.
Som framgår av diagram 4.2.1 är också parameterestimaten "oroliga", men den stationära förlusten är ändå så bra som 1.22. Simuleringen är här gjord med P0=10, ULIM=-1 (dvs ingen begränsning av styrsignalen) samt initialvärdena 0 på parameterestimaten. STURE1 fungerar alltså här utmärkt, särskilt med tanke på den korta räknetiden, ungefär 15 sek per 1000 samplingspunkter.

STURE2

STURE2 identifierar istället parametrarna i systemet, dock inte c-parametrarna. Styrlagen beräknas sedan som i (4.2.3) men utan c-parametrar, och därför blir villkoren för optimal reglering:

$$\begin{cases} \frac{a^2}{b} = 0.4275 \\ a = -0.45 \end{cases} \Rightarrow \begin{cases} a=-0.45 \\ b=0.47 \end{cases} \qquad (4.2.4)$$

Parameterestimaten framgår av diagram 4.2.2, och förlusten är 1.21 per steg. Ingen signifikant förbättring av regleringen märks alltså jämfört med STURE1. Räknetiden är något längre - 25 sek per 1000 punkter. Simuleringen är gjord med P0=10, ULIM=10 och initialvärdena 0 på parameterestimaten.

STUREM

STUREM identifierar också c-parametrarna, och en betraktelse av styrlagen (4.2.3) ger villkoren på estimaten för att den optimala regulatorn skall fås:

$$\begin{cases} \frac{a}{b}(c-a) = -0.4275 \\ c-a = 0.45 \end{cases} \Rightarrow \begin{cases} \frac{a}{b} = -0.95 \\ c-a = 0.45 \end{cases} \qquad (4.2.5)$$

De estimat, som alltså ger en optimal regulator, utgör en lineär mångfald. Estimaten framgår av diagram 4.2.3 och kvantiteterna c-a och a/b är plottade i diagram 4.2.4. Insvängningen mot värdena enl (4.2.5) framgår här tydligt. Den stationära förlusten är samma som för STURE2, dvs 1.21 per steg. Simuleringen använder P0=10, ULIM=10 och initialvärden på parametrarna är 0. Räknetiden är längre än för de båda andra regulatorerna, c:a 40 sek per 1000 steg.

Sammanställning ex 1

	STURE1	STURE2	STUREM
Initialförlust,total	0	50	50
Stationär förlust per steg	1.22	1.21	1.21
Optimal d:o	1.20	1.20	1.20
P0	10	10	10
ULIM	-1	10	10
Initiala parametervärden	0	0	0
Räknetid per 1000 steg (sek)	15	25	40

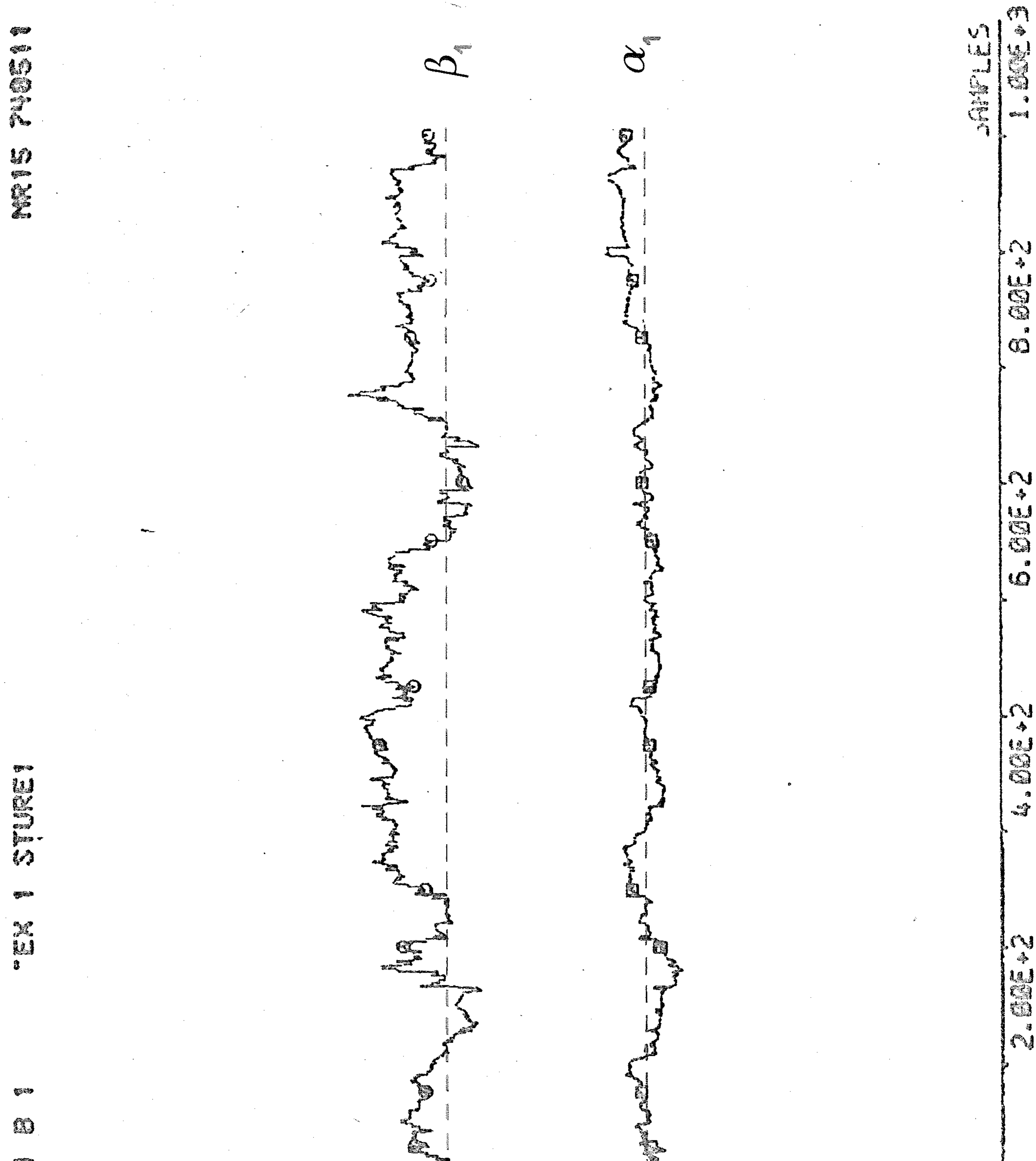
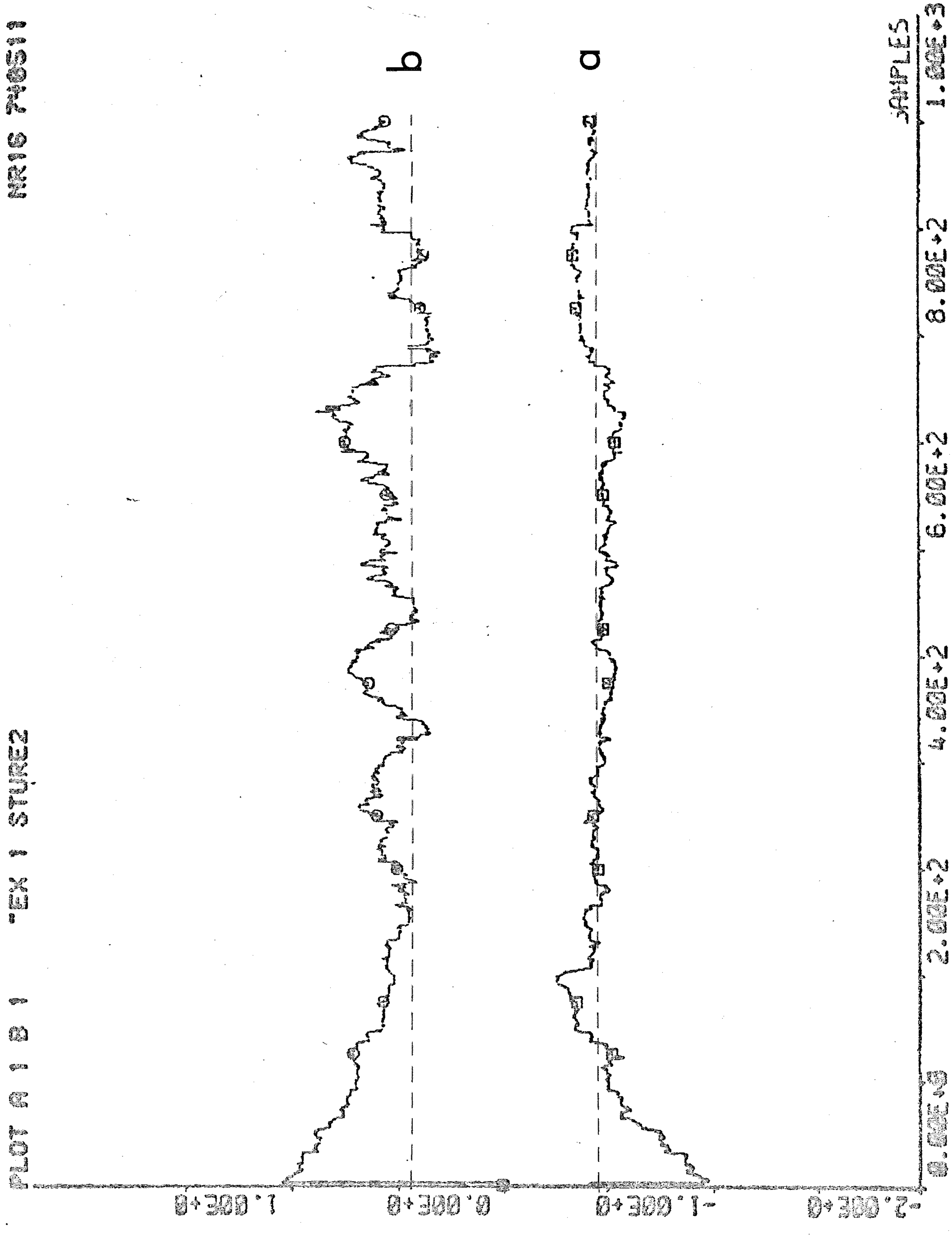

PLOT A 1 B 1
EX 1 STURE
MR15 740511
β_1
α_1
SAMPLES
0.00E+0
2.00E+2
4.00E+2
6.00E+2
8.00E+2
1.00E+3
1.00E+0
0.00E+0
-1.00E+0
-2.00E+0

Diagram 4.2.1



1.00E+0
0.00E+0
-1.00E+0
-2.00E+0
0.00E+0
2.00E+2
4.00E+2
6.00E+2
8.00E+2
1.00E+3
SAMPLES
a
b

Diagram 4.2 2

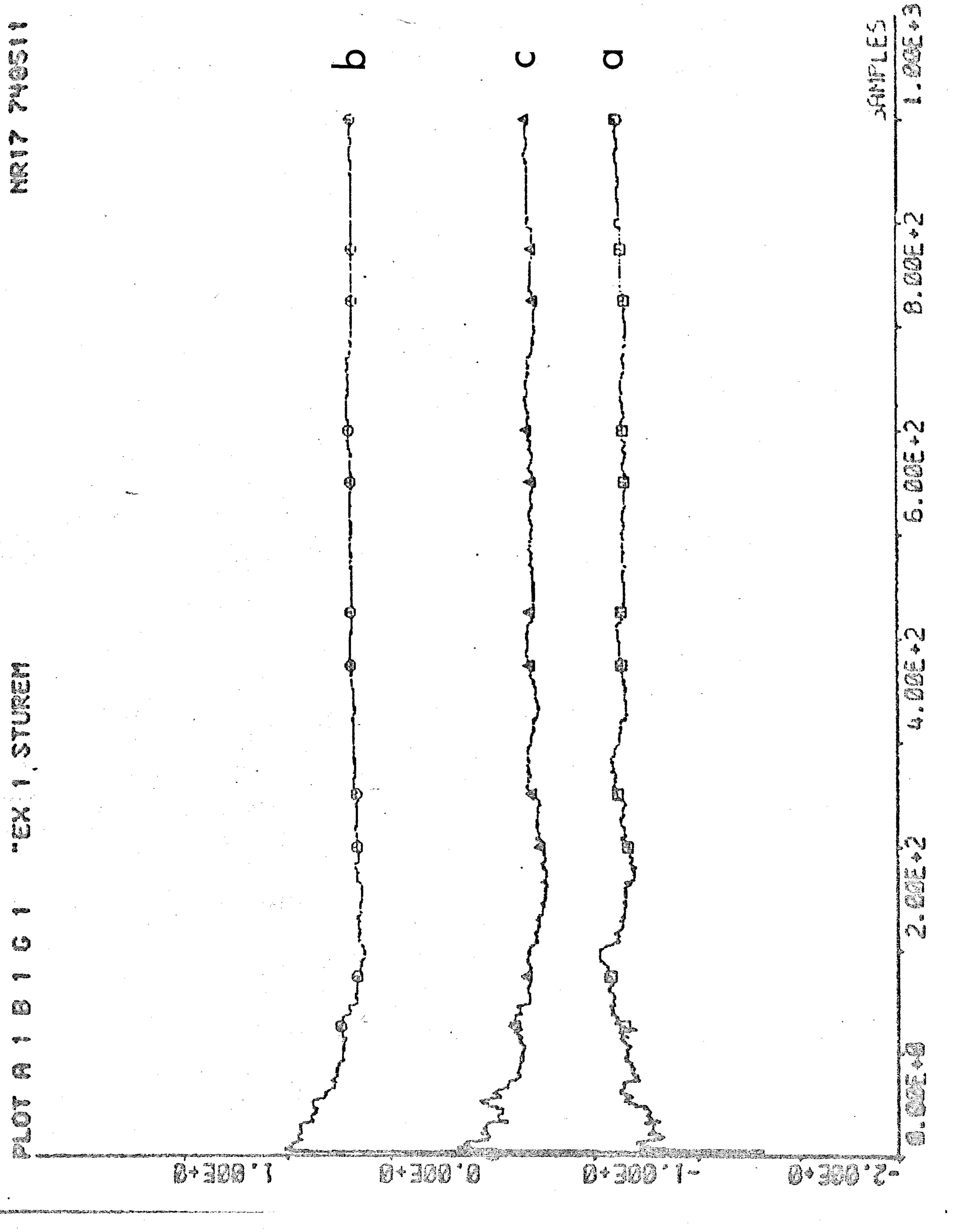

PLOT A 1 B 1 G 1 "EX 1 STUREM
NR17 740511
b
c
a
SAMPLES
0.00E+0
2.00E+2
4.00E+2
6.00E+2
8.00E+2
1.00E+3
1.00E+0
0.00E+0
-1.00E+0
-2.00E+0

Diagram 4.2.3

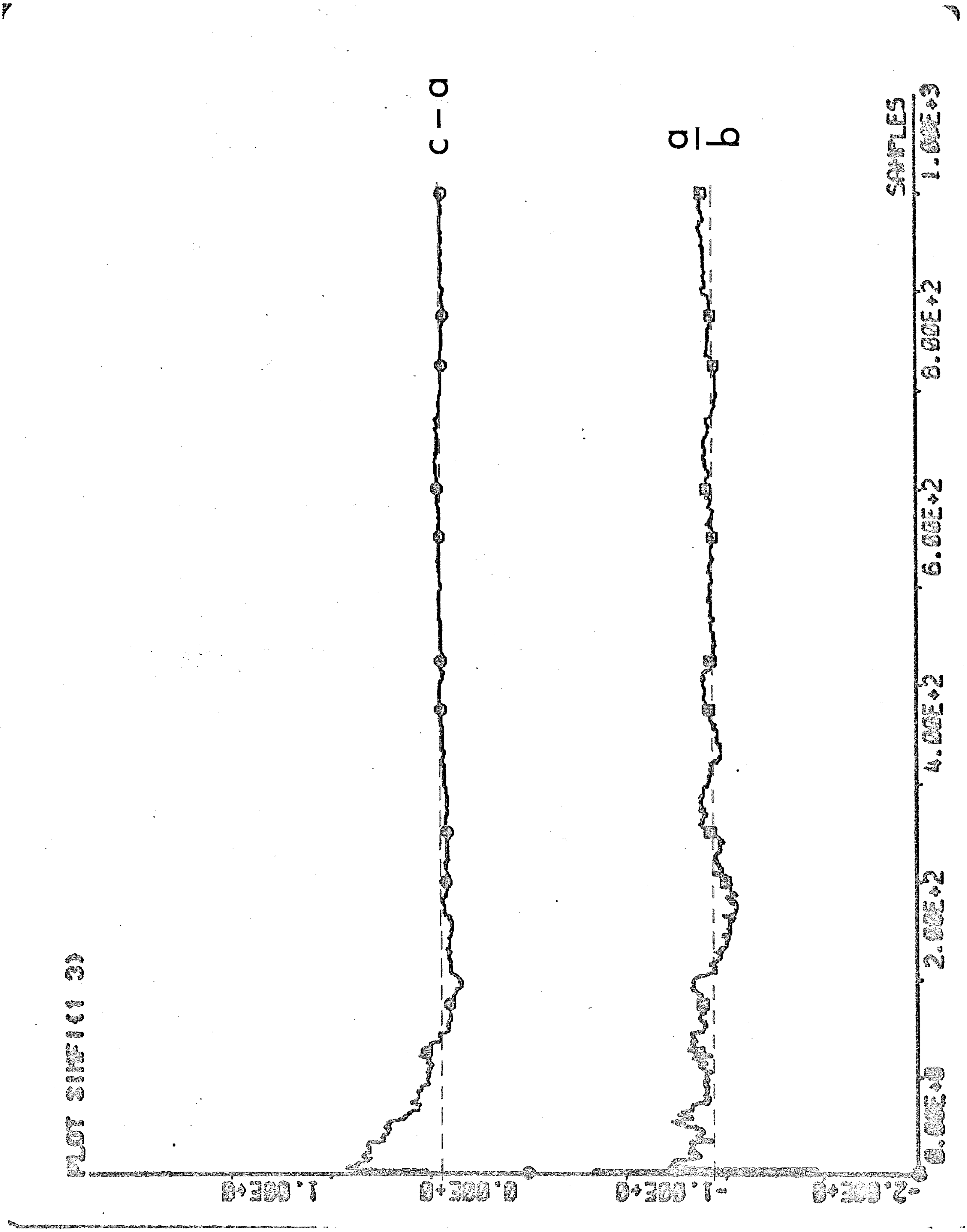

PLOT SIMFI(1 3)
c − a
a/b
SAMPLES
1.00E+0
0.00E+0
-1.00E+0
-2.00E+0
0.00E+0
2.00E+2
4.00E+2
6.00E+2
8.00E+2
1.00E+3

Diagram 4.2.4

3. Exempel 2

Vi skall nu studera systemet (behandlat i {1}, sid 61 ff)

$$(1 + aq^{-1})y(t) = q^{-2}bu(t) + (1 + cq^{-1})e(t) \qquad (4.3.1)$$

och särskiljer två fall:

I. a=-1.5	II. a=-1.5
b=1	b=1
c=0	c=-0.9

Systemet är alltså ett instabilt min-fas-system av 1:a ordningen med i fall I C-polynomet identiskt lika med 1, i fall II inte. Identiteten (4.2.2) blir i detta fall:

$$1 + cq^{-1} = (1 + aq^{-1})(1 + f_1q^{-1} + f_2q^{-2}) + q^{-3}g \quad \text{som ger}$$

		I	II	
$f_1=c-a$		1.5	0.6	
$f_2=-a(c-a)$	dvs:	2.25	0.9	(4.3.2)
$g=a^2(c-a)$		3.375	1.35	

Styrlagen blir:

$$u(t) = \frac{-\frac{a^2}{b}(c-a)}{1 + (c-a)q^{-1} - a(c-a)q^{-2}}\, y(t-1) \qquad (4.3.3)$$

och teoretiskt optimal förlust $1 + f_1^2 + f_2^2 = \begin{cases} 8.31 & \text{(fall I)} \\ 2.17 & \text{(fall II)} \end{cases}$
per samplingsintervall.

STUREI

Regulatorn skall estimera parametrarna i (4.3.3) och för optimal reglering skall gälla:

Fall I: $\alpha_1 = -3.375$
$\beta_1 = 1.5$
$\beta_2 = 2.25$

Fall II: $\alpha_1 = -1.35$
$\beta_1 = 0.6$
$\beta_2 = 0.9$

För fall I är C-polynomet=1 och STURE1 bör alltså inte ha några problem (att öppna systemet är instabilt skall inte betyda något). Diagram 4.3.1 visar också att parameterestimaten svänger kring de rätta värdena. Här har simulerats 2000 punkter beroende på att α_1 är så "orolig". Förlusten är stationärt 8.40 jfrt med 8.31 för optimala fallet. P0=10, ULIM=-1 samt initialvärdena är 0 i simuleringen. I fall II är C-polynomet skilt från 1, men diagram 4.3.2 visar att estimaten ändå svänger in mot de rätta värdena. Förlusten är 2.3 per steg. Parametervärdena är samma som i fall I och ibåda fallen är räknetiden c:a 15 sek per 1000 steg.

STURE2

Utesluts c-parametrarna i (4.3.3) får vi de villkor, som måste gälla för att STURE2 skall ge optimal regulator:

Fall I:
$$\begin{cases} \frac{a^3}{b} = -3.375 \\ a = -1.5 \\ a^2 = 2.25 \end{cases} \Rightarrow \begin{cases} a = -1.5 \\ b = 1 \end{cases}$$

Fall II:
$$\begin{cases} \frac{a^3}{b} = -1.35 \\ a = -0.6 \\ a^2 = 0.9 \end{cases} \Rightarrow \text{lösning saknas}$$

Vi träffar alltså här på första fallet, då STURE2 inte kan konvergera mot den optimala regulatorn. Detta inträffar dock endast i fall II, och vi inleder med att studera fall I. Diagram 4.3.3 visar att estimaten snabbt konvergerar mot de rätta, och förlusten är också bra: 8.25 per steg. I fall II däremot fås alltså inte den optimala regulatorn, utan STURE2 ger en suboptimal regulator. Diagram 4.3.4 visar parameterestimaten och förlusten är här 4.1 per steg, att jämföra med det optimala 2.17. I båda simuleringarna är P0=10, ULIM=50 och initialvärdena på parametrarna 0.1, det senare för att undvika de kraftiga styrsignaler i uppstarten som blir följden av initialvärdena 0. Räknetiden är i båda simuleringarna c:a 45 sek per 1000 steg.

STUREM

I STUREM tas hänsyn även till c-parametrarna och styrlagen (4.3.3) ger villkor på estimaten för att STUREM skall ge den optimala regulatorn:

Fall I:

$$\begin{cases} \frac{a^2}{b}(c-a) = 3.375 \\ c-a = 1.5 \\ a(c-a) = -2.25 \end{cases} \Rightarrow \begin{cases} a = -1.5 \\ b = 1 \\ c = 0 \end{cases}$$

Fall II:

$$\begin{cases} \frac{a^2}{b}(c-a) = 1.35 \\ c-a = 0.6 \\ a(c-a) = -0.9 \end{cases} \Rightarrow \begin{cases} a = -1.5 \\ b = 1 \\ c = -0.9 \end{cases}$$

Vi noterar här, att estimaten måste vara lika med systemets verkliga parametrar för att optimal reglering skall uppnås. För fall I visar diagram 4.3.5 att parameterestimaten är mycket goda och förlusten blir också liten: 8.25. Till skillnad från STURE2 fungerar STUREM bra också i fall II. Estimaten svänger snabbt in sig mot de riktiga värdena, vilket framgår av diagram 4.3.6. Förlusten är stationärt 2.15, vilket visar att STUREM i motsats till STURE2 ger optimal regulator. I både fall I och fall II har STUREM simulerats med initialvärdena 0.1, P0=10 och ULIM=10. Räknetiden rör sig om c:a 55 sek på 1000 punkter.

Sammanställning ex 2

Fall I

	STURE1	STURE2	STUREM
Initialförlust,total	20000	20000	0
Stationär förlust per steg	8.40	8.25	8.25
Optimal d:o	8.31	8.31	8.31
PO	10	10	10
ULIM	-1	50	10
Initiala parametervärden	0	0.1	0.1
Räknetid per 1000 steg (sek)	15	45	55

Fall II

	STURE1	STURE2	STUREM
Initialförlust,total	2500	1000	300
Stationär förlust per steg	2.3	4.1	2.15
Optimal d:o	2.17	2.17	2.17
PO	10	10	10
ULIM	-1	50	10
Initiala parametervärden	0	0.1	0.1
Räknetid per 1000 steg (sek)	15	45	55

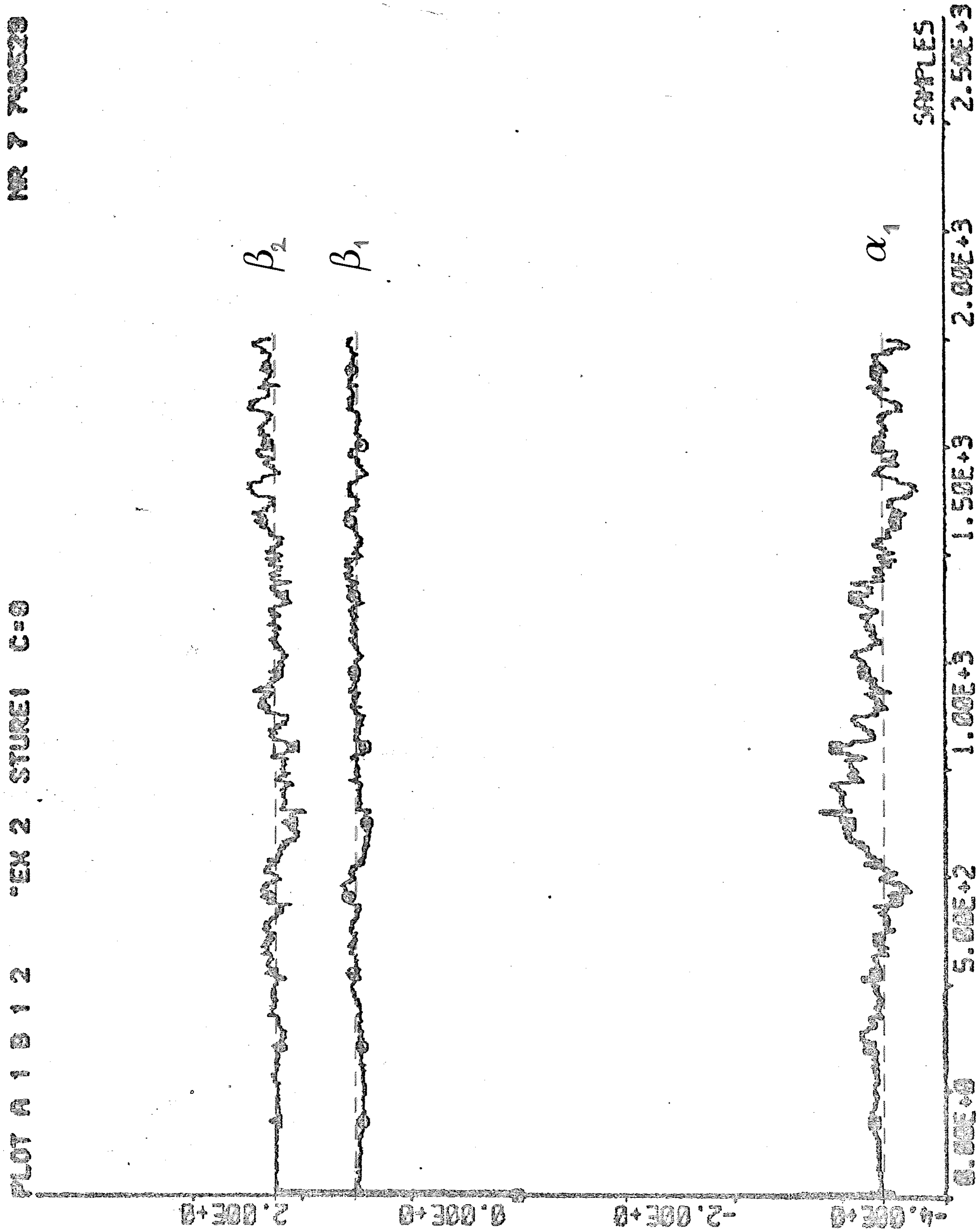
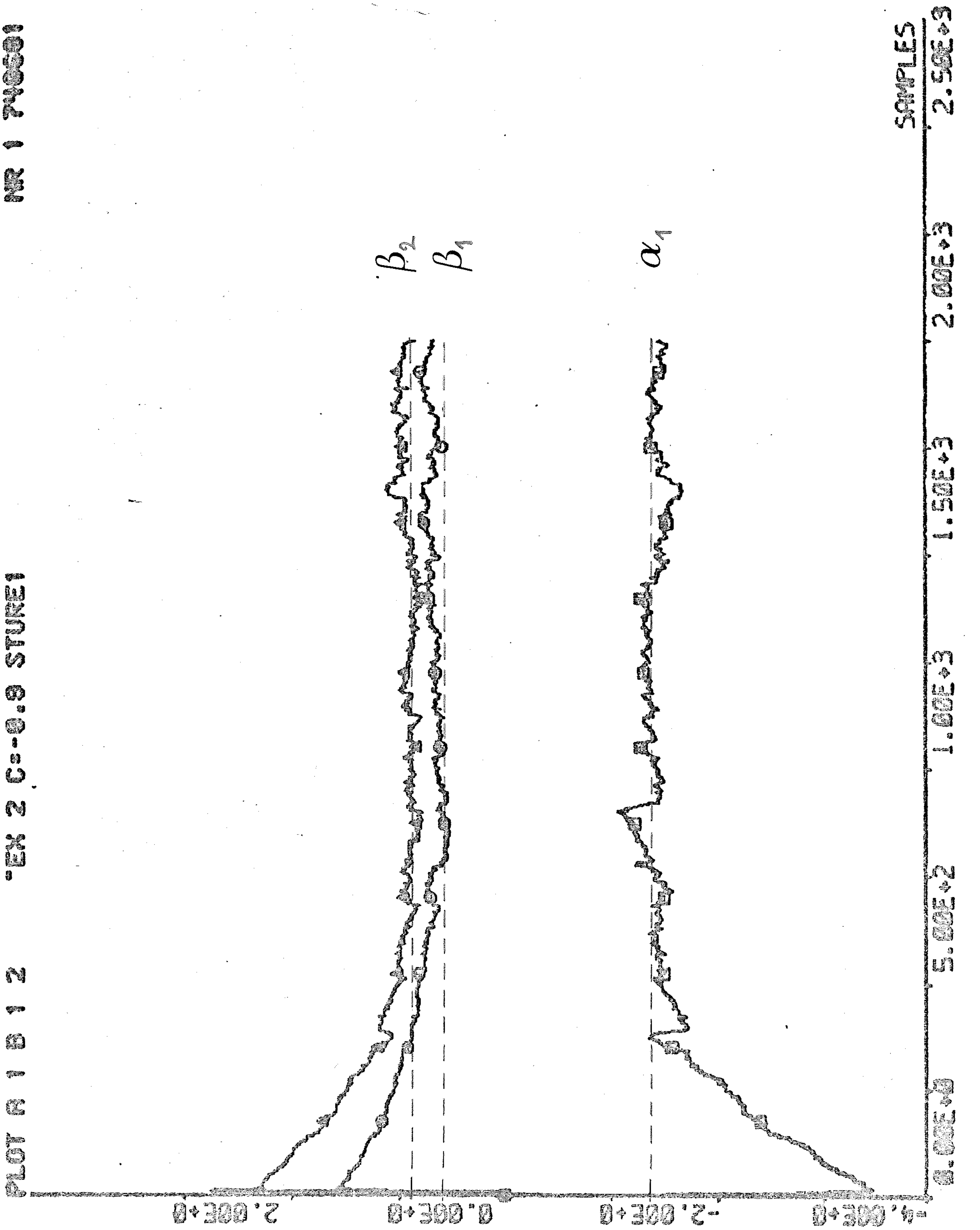

PLOT A 1 B 1 2 'EX 2 STURE1 C:0
β_2
β_1
α_1
2.00E+0
0.00E+0
-2.00E+0
-4.00E+0
0.00E+0
5.00E+2
1.00E+3
1.50E+3
2.00E+3
2.50E+3
SAMPLES

Diagram 4.3.1



SAMPLES
2.00E+0
0.00E+0
-2.00E+0
-4.00E+0
5.00E+2
1.00E+3
1.50E+3
2.00E+3
2.50E+3
β_2
β_1
α_1

Diagram 4.3.2

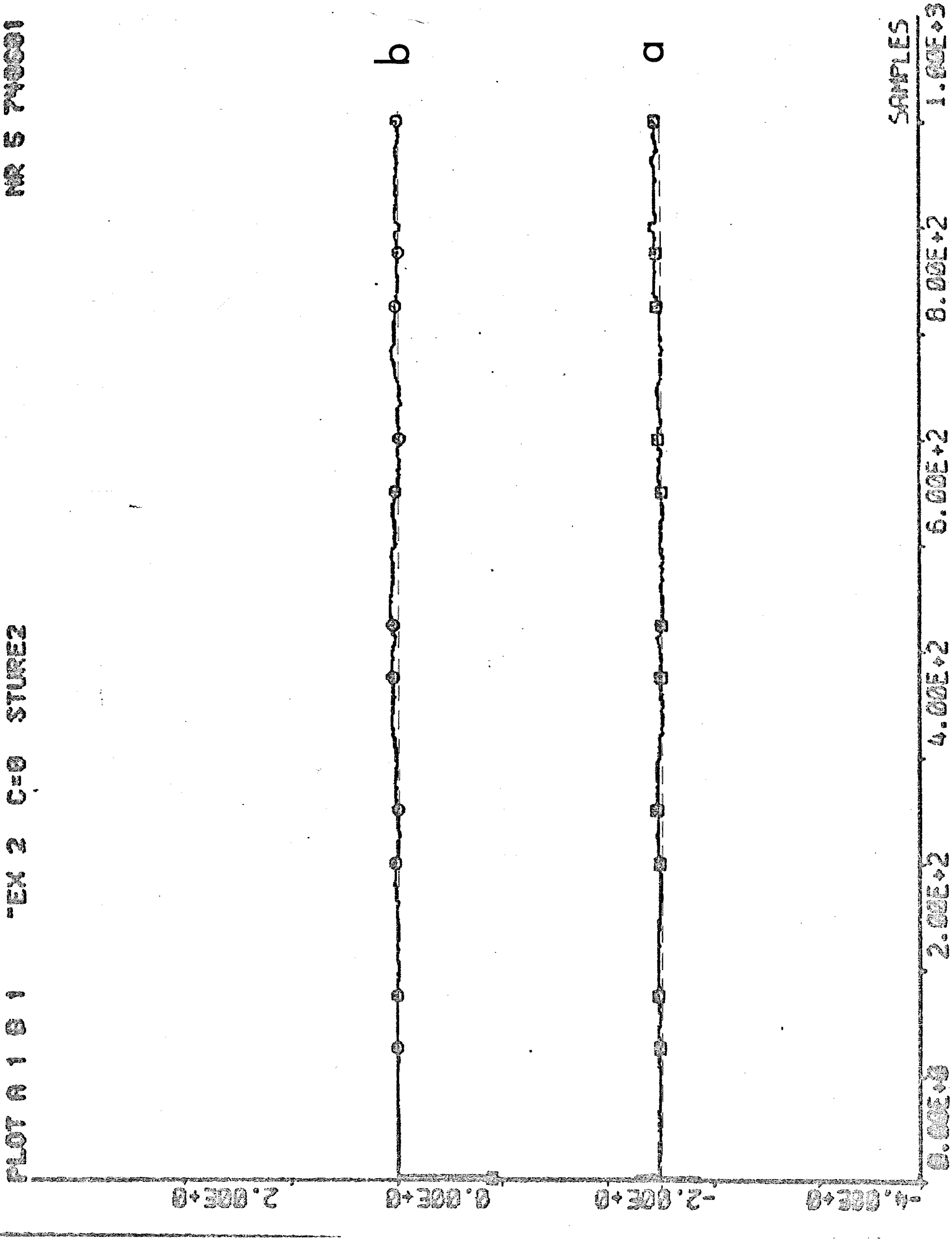

PLOT A 1 B 1 "EX 2 C=0 STURE2
SAMPLES
b
a
0.00E+0
2.00E+2
4.00E+2
6.00E+2
8.00E+2
1.00E+3
2.00E+0
0.00E+0
-2.00E+0
-4.00E+0

Diagram 4.3.3

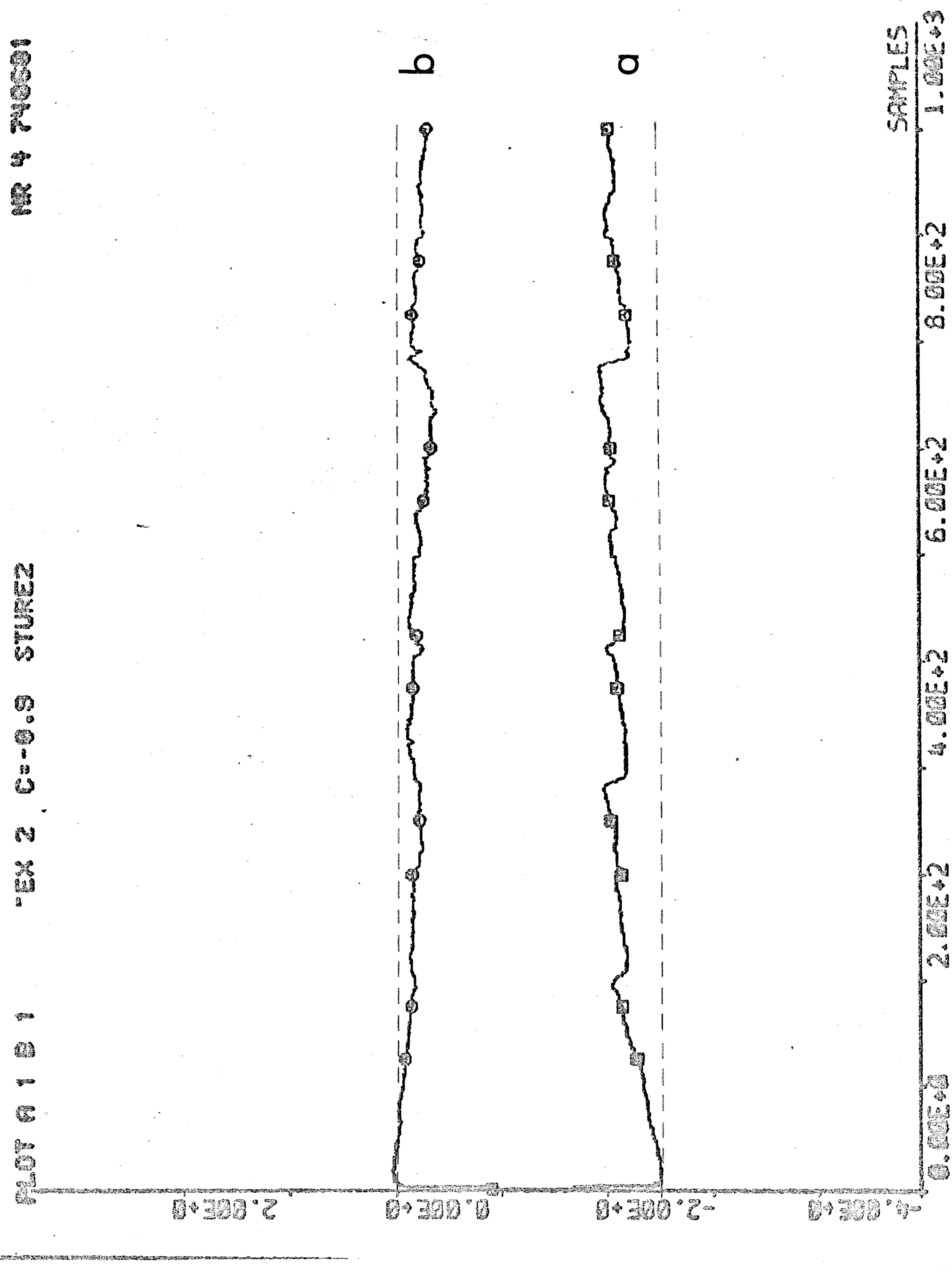

PLOT A 1 B 1 'EX 2 C=-0.9 STUREZ NR 4 740601
b
a
SAMPLES
0.00E+0
2.00E+2
4.00E+2
6.00E+2
8.00E+2
1.00E+3
2.00E+0
0.00E+0
-2.00E+0
-4.00E+0

Diagram 4.3.4

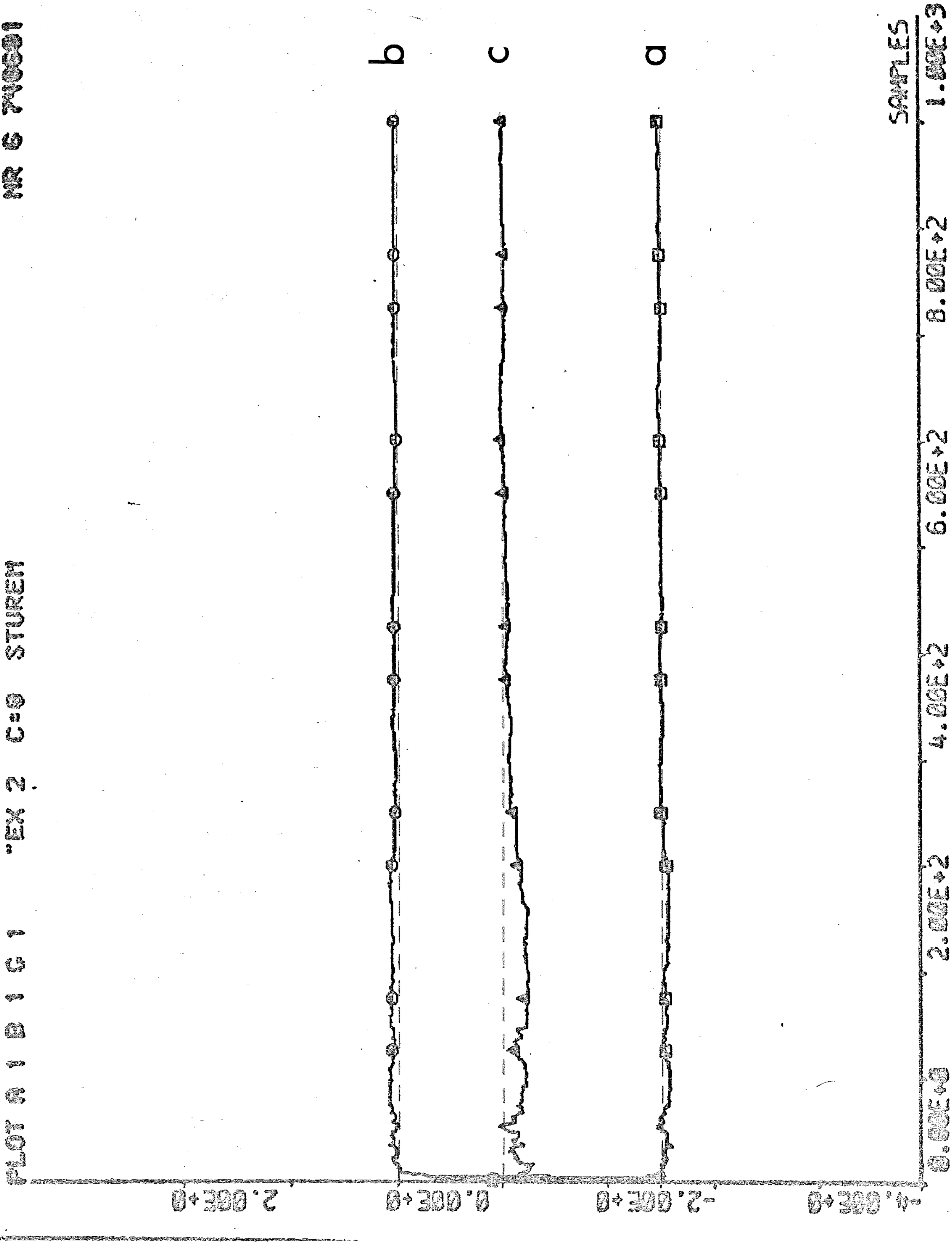

b
c
a
SAMPLES
1.00E+3
8.00E+2
6.00E+2
4.00E+2
2.00E+2
2.00E+0
0.00E+0
-2.00E+0
-4.00E+0

Diagram 4.3.5

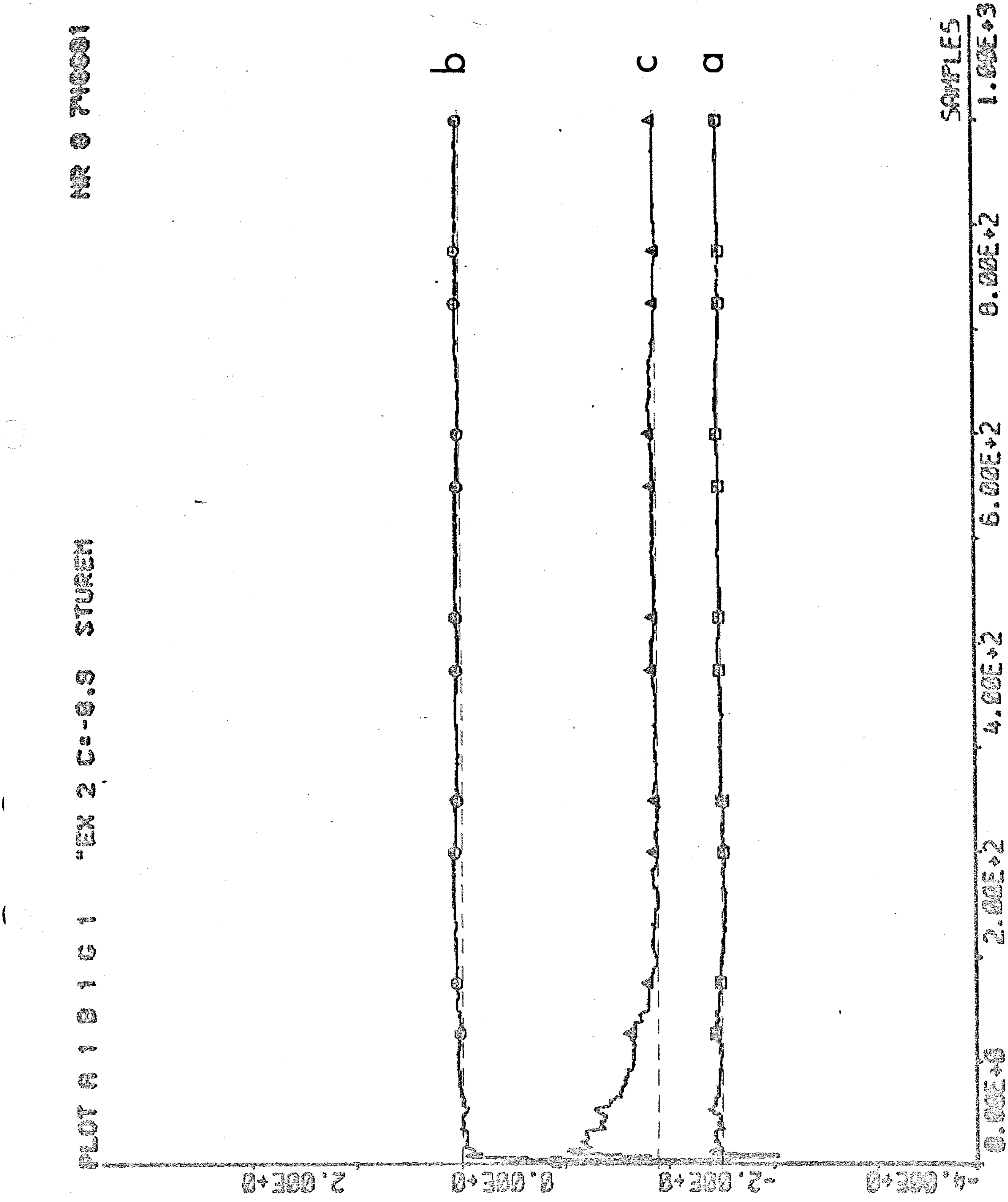

b
c
a
SAMPLES
0.00E+0
2.00E+2
4.00E+2
6.00E+2
8.00E+2
1.00E+3
2.00E+0
0.00E+0
-2.00E+0
-4.00E+0

Diagram 4.3.6

4. Exempel 3

Systemet vi skall uppehålla oss vid i detta avsnitt är:

$$(1 + aq^{-1})y(t) = q^{-1}(b_1 + b_2q^{-1})u(t) + (1 + cq^{-1})e(t) \qquad (4.4.1)$$

med $a=-0.95$, $b_1=1$, $b_2=2$ samt 3 olika fall för c-parametrarna:

I $c=0$

II $c=-0.3$

III $c=-0.7$

Detta system behandlas i {3}, ex 6.3. Systemet är alltså ett stabilt 2:a ordningens icke-min-fas-system. Vid härledningen av min-varians-regulatorn ersätts nu identiteten (4.2.2) av identiteten (2.2.3), som för systemet (4.4.1) får utseendet:

$$\begin{aligned}(b_1q^{-1} + b_2)(1 + cq^{-1}) &= (1 + aq^{-1})(f_0 + f_1q^{-1} + f_2q^{-2}) + \\ &\quad + q^{-2}(b_1 + b_2q^{-1})(g_0 + g_1q^{-1})\end{aligned} \qquad (4.4.2)$$

Detta ger följande ekvationssystem för den optimala regulatorn:

$$\begin{cases} b_2 = f_0 \\ b_1 + b_2c = f_1 + af_0 \\ b_1c = f_2 + af_1 + b_1g_0 \\ 0 = af_2 + b_1g_1 + b_2g_0 \\ 0 = b_2g_1 \end{cases} \qquad (4.4.3)$$

(4.4.3) har lösningen

$$\begin{cases} f_0 = b_2 \\ f_1 = b_1 + b_2(c-a) \\ f_2 = \dfrac{b_2(c-a)(b_1-ab_2)}{b_2-ab_1} \\ g_1 = 0 \end{cases} \qquad (4.4.4)$$

Detta ger den optimala styrlagen (se II.2):

$$u(t) = -\frac{G^*(q^{-1})}{B^{+*}(q^{-1})F^*(q^{-1})}\, y(t-1) = -\frac{g_0 + g_1 q^{-1}}{f_0 + f_1 q^{-1} + f_2 q^{-2}}\, y(t-1) =$$

$$= \frac{\dfrac{a(c-a)(b_1-ab_2)}{b_2(b_2-ab_1)}}{1 + \dfrac{b_1 + b_2(c-a)}{b_2}\, q^{-1} + \dfrac{(c-a)(b_1 - ab_2)}{b_2 - ab_1}\, q^{-2}}\, y(t-1) \qquad (4.4.5)$$

Med insatta värden fås styrlagen i resp fall:

I $\quad u(t) = \dfrac{-0.444}{1 + 1.45q^{-1} + 0.934q^{-2}}\, y(t-1)$

II $\quad u(t) = \dfrac{-0.304}{1 + 1.15q^{-1} + 0.639q^{-2}}\, y(t-1)$

III $\quad u(t) = \dfrac{-0.117}{1 + 0.75q^{-1} + 0.246q^{-2}}\, y(t-1)$

Det slutna systemet ser nu lite annorlunda ut jfrt med min-fas-fallet:

$$y(t) = \frac{F^*(q^{-1})}{B^-(q^{-1})}\, e(t) \qquad (4.4.6)$$

vilket i vårt exempel blir:

$$y(t) = \frac{b_2 + (b_1 + b_2(c-a))q^{-1} + \dfrac{b_2(c-a)(b_1 - ab_2)}{b_2 - ab_1}\, q^{-2}}{b_2 + b_1 q^{-1}}\, e(t) \qquad (4.4.7)$$

För att erhålla utsignalens varians får vi här lösa en integral, vilket gjorts enl {5} och gett resultatet:

I $\quad Ey^2 = 2.18$

II $\quad Ey^2 = 1.55$

III $\quad Ey^2 = 1.08$

Det är nu dags att studera hur regulatorerna klarar sin uppgift på systemet (4.4.1).

STURE1

Systemet (4.4.1) är icke-min-fas och enl avsnitt II.1 är STURE1-algoritmen då instabil. Vid simulering har också konstaterats, att algoritmen divergerar i samtliga tre fall om rätt k-värde ges till regulatorn. Parameterestimaten är på "rätt väg" i ung 10 steg, men antar därefter helt orimliga värden. I avsnitt II.1 nämns att egenskaperna hos STURE1 kan förbättras kraftigt med hjälp av olika tricks. Några sådana försök har inte gjorts på detta exempel, utan vi nöjer oss med att konstatera divergensen i "normalfallet". Inga diagram ges då dessa saknar intresse.

STURE2

För fall I har vi anledning att vänta oss en bra reglering med STURE2, då C-polynomet är identiskt 1. Diagram 4.4.1 visar också, att parametrarna svänger kring de rätta värdena. Förlusten ligger också nära den optimala: 2.2 per steg.
I fall II och III kan man inte lösa ekv-systemet (4.4.4) för optimal regulator m.a.p. a, b_1 och b_2 när c-parametrarna utesluts. Detta innebär, att STURE2 inte kan konvergera mot min-varians-regulatorn. Man får dock i alla fall en bra reglering - STURE2 ger en suboptimal regulator, som trots fladdrande parameterestimat (se diagram 4.4.2 resp 4.4.3) ger en låg förlust: i fall II 1.6 (optimalt 1.55) och i fall III 1.7 (optimalt 1.08) per samplingsintervall. För samtliga fall verkar det som om sam-variationer hos estimaten inte ger sämre regulator (jfr ex 1 STUREM). Samtliga simuleringar har initialvärden 0, P0=10 och ULIM=10. Räknetiden är ungefär 90 sek för 1000 punkter.

STUREM

Vi väntar oss i fall I, att STUREM skall uppföra sig ungefär som STURE2, eftersom ingen c-parameter finns. STURE2 gav en bra regulator, och det gör STUREM också - förlusten blir liksom för STURE2 2.2 per steg. Parameterestimaten svänger också in mot systemets verkliga, även om det går något långsamt, vilket framgår av diagram 4.4.4. För fall II och III kan vi vänta oss ett bättre resultat för STUREM än för STURE2, som ju inte gav optimala regulatorn. Parameterestimaten i fall II enl diagram 4.4.5 verkar svänga in mot de verkliga värdena, om än långsamt. Estimaten i fall III däremot tenderar knappast gå in mot de rätta värdena, vilket skulle kunna förklaras med att kanske också andra parameterkombinationer än de rätta ger en optimal regulator. Eventuellt är detta också möjligt i fall I och fall II. Trots att estimaten alltså är lite "underliga" visar sig förlusten vara mycket bra: 1.56 i fall II, 1.1 i fall III. STUREM är alltså här en betydligt bättre regulator än STURE2. Parametervärden vid simuleringarna var samma som för STURE2: P0=10, ULIM=10 samt initialvärden 0. Räknetiden är ungefär densamma - 100 sek för 1000 punkter.

Sammanställning ex 3

	Fall I		Fall II		Fall III	
	S2	SM	S2	SM	S2	SM
Initialförlust,total	4500	700	5000	700	400	5500
Stationär förlust per steg	2.2	2.2	1.6	1.56	1.7	1.1
Optimal d:o	2.18		1.55		1.08	
P0	10	10	10	10	10	10
ULIM	10	10	10	10	10	10
Initiala parametervärden	0	0	0	0	0	0
Räknetid per 1000 steg (sek)	90	100	90	100	90	100

OBS! STURE1 divergerar i samtliga fall.

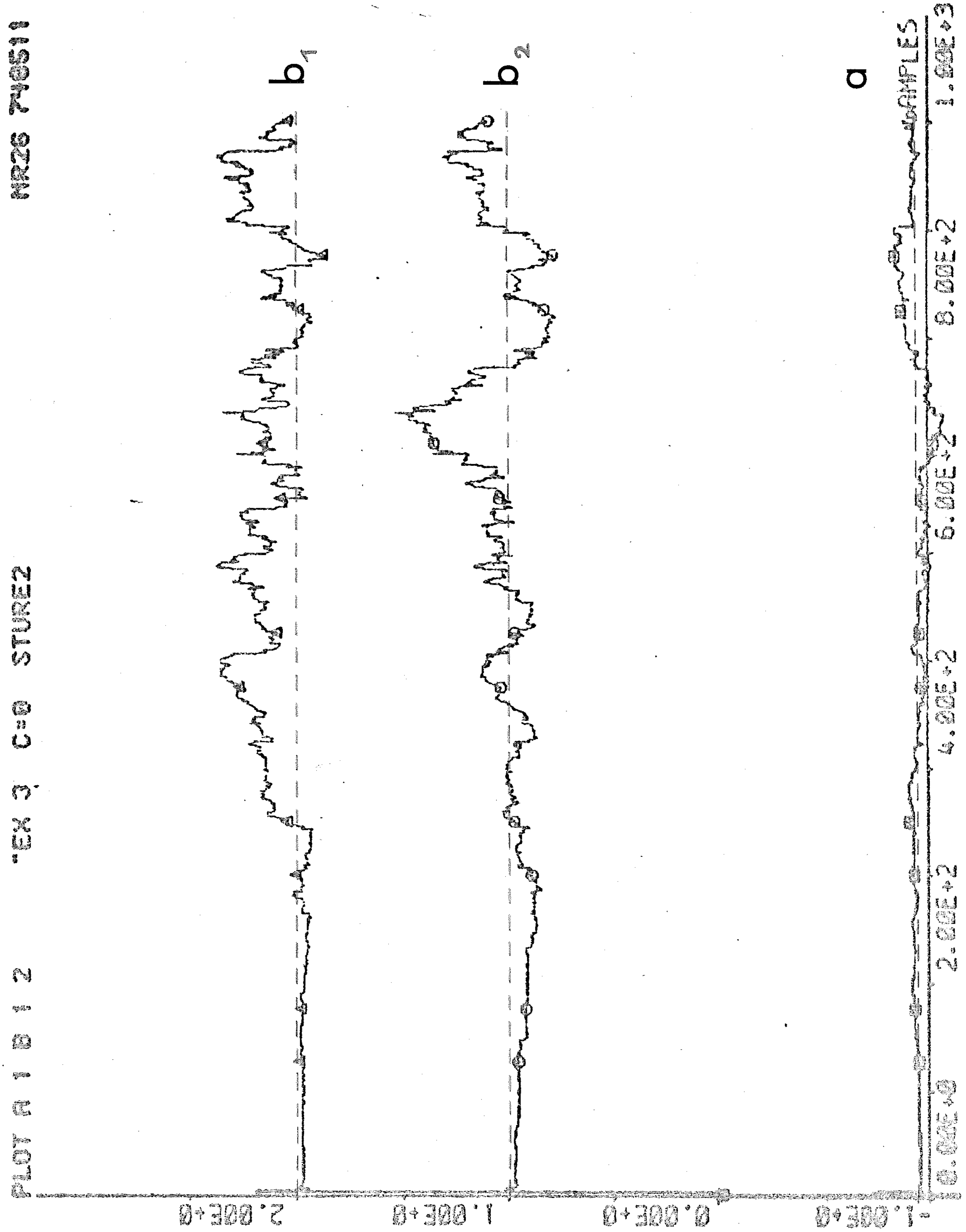

PLOT A ! B ! 2 "EX 3 C=0 STURE2
MR26 740511
b1
b2
a
SAMPLES
2.00E+0
1.00E+0
0.00E+0
-1.00E+0
0.00E+0
2.00E+2
4.00E+2
6.00E+2
8.00E+2
1.00E+3

Diagram 4.4.1

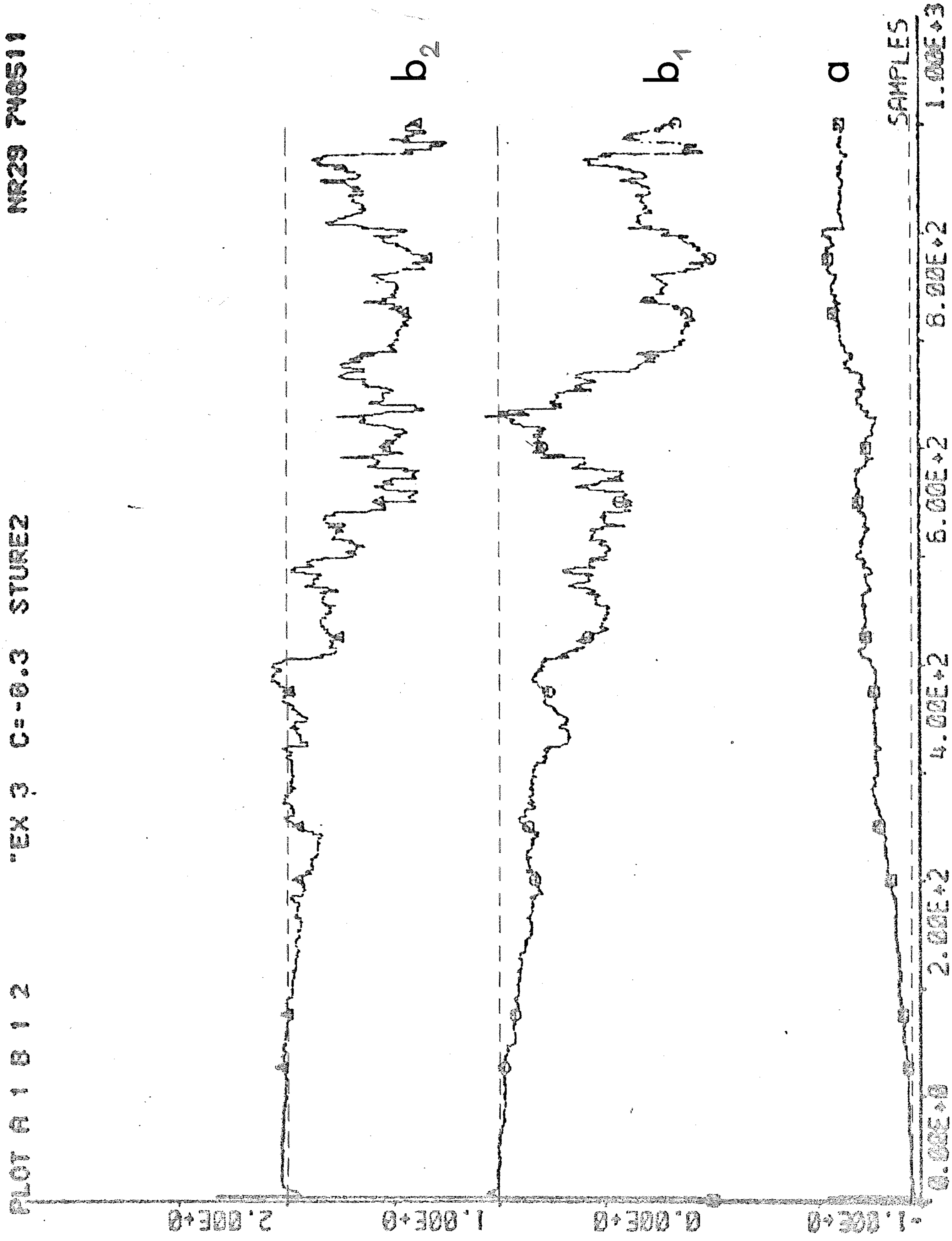

PLOT A 1 B 1 2
'EX 3 C=-0.3 STURE2
MR29 740511
b2
b1
a
SAMPLES
0.00E+0
2.00E+2
4.00E+2
6.00E+2
8.00E+2
1.00E+3
-1.00E+0
0.00E+0
1.00E+0
2.00E+0

Diagram 4.4.2

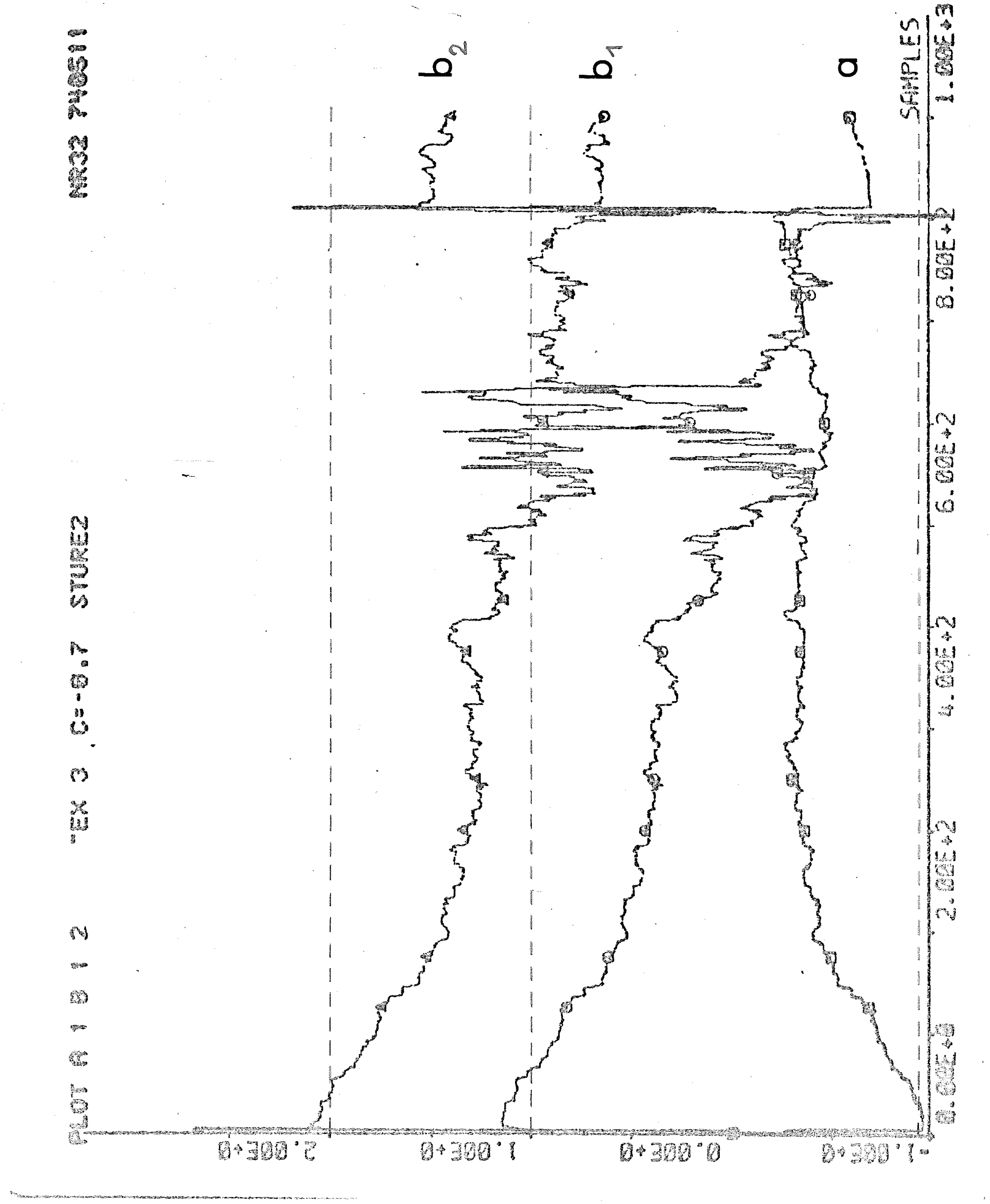

PLOT A 1 B 1 2 "EX 3 C=-0.7 STURE2 MR32 740511
b2
b1
a
SAMPLES
0.00E+0
2.00E+2
4.00E+2
6.00E+2
8.00E+2
1.00E+3
-1.00E+0
0.00E+0
1.00E+0
2.00E+0

Diagram 4.4.3

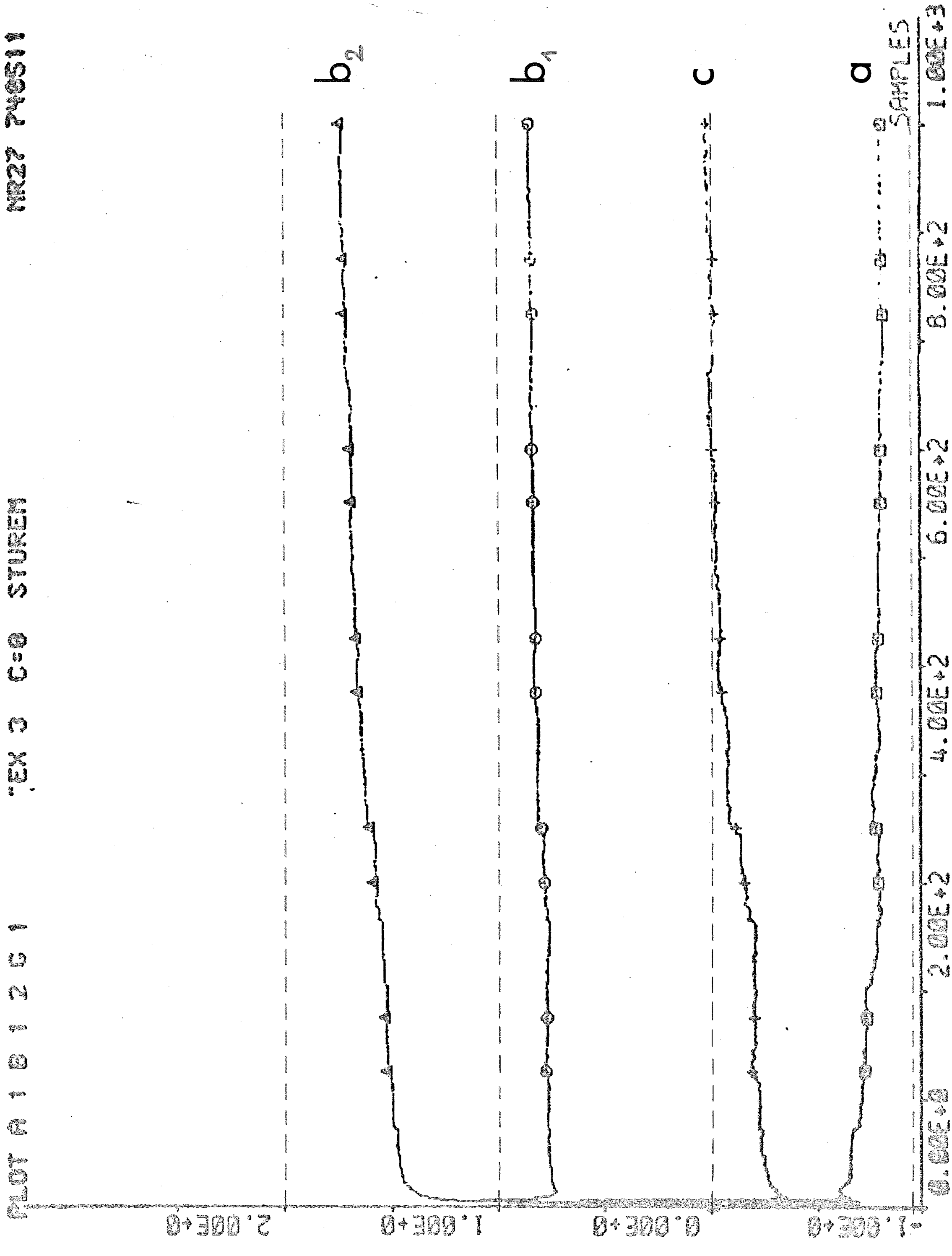
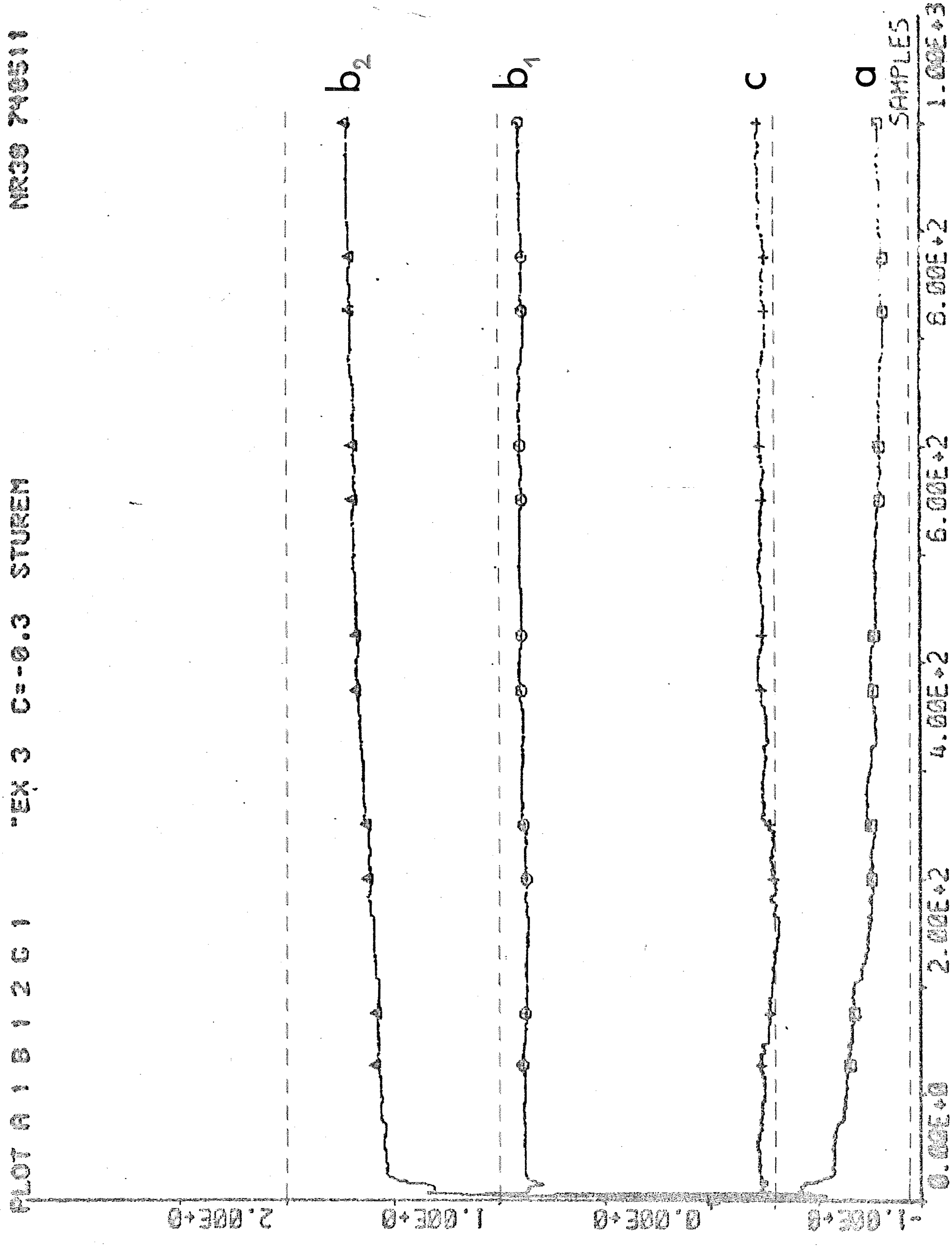
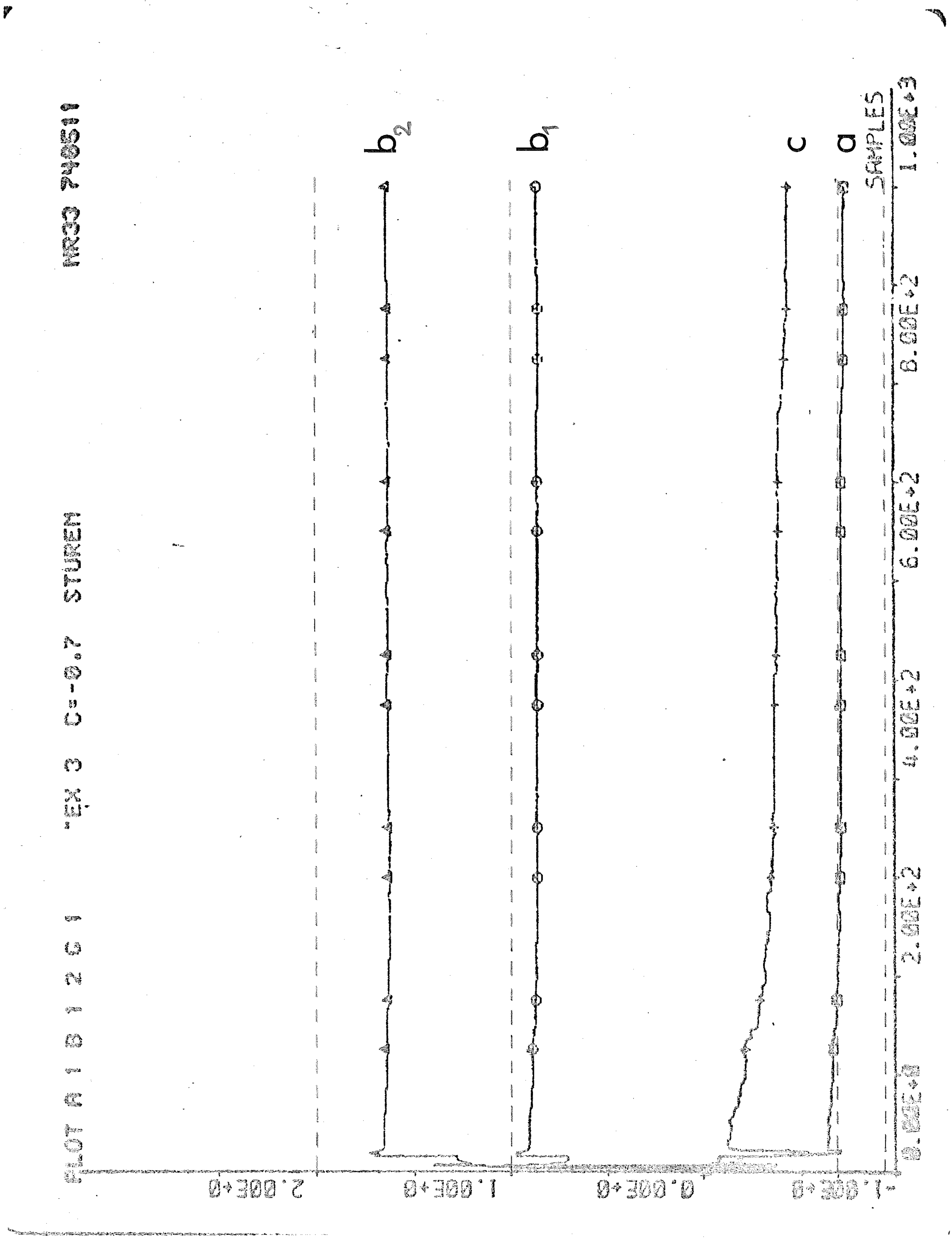

PLOT A 1 B 1 2 C 1
EX 3 C=0 STUREM
NR27 740511
SAMPLES
a
b1
b2
c
-1.00E+0
0.00E+0
1.00E+0
2.00E+0
0.00E+0
2.00E+2
4.00E+2
6.00E+2
8.00E+2
1.00E+3

Diagram 4.4.4



PLOT A 1 B 1 2 C 1 "EX 3 C=-0.3 STUREM NR38 740511
b2
b1
c
a
SAMPLES
0.00E+0
2.00E+2
4.00E+2
6.00E+2
8.00E+2
1.00E+3
2.00E+0
1.00E+0
0.00E+0
-1.00E+0

Diagram 4.4.5



PLOT A 1 B 1 2 C 1
"EX 3 C=-0.7 STUREN
MR30 740511
b2
b1
c
a
SAMPLES
2.00E+0
1.00E+0
0.00E+0
-1.00E+0
0.00E+0
2.00E+2
4.00E+2
6.00E+2
8.00E+2
1.00E+3

Diagram 4.4.6

5. Exempel 4

Vi skall slutligen studera systemet

$$y(t) = \frac{-q^{-1} + 1.3q^{-2}}{1 - 0.7q^{-1}} u(t),$$

som störs på utgången av drivande brus v(t), givet av:

$$v(t) = \frac{1}{(1-0.7q^{-1})(1-q^{-1})} e(t), \quad \text{där } e(t) \text{ är vitt brus.}$$

Detta ger oss slutligen systemet

$$(1 + \alpha_1 q^{-1})(1 + \alpha_2 q^{-1})y(t) = -q^{-1}(1 + \beta_1 q^{-1})(1 + \beta_2 q^{-1})u(t) + e(t)$$

med

$$\begin{array}{ll} \alpha_1 = -1 & \beta_1 = -1 \\ \alpha_2 = -0.7 & \beta_2 = -1.3 \end{array} \qquad (4.5.1)$$

Det är alltså fråga om ett 3:e ordningens system, som är icke-min-fas och med A-polynomets nollställen i 0, 0.7 och 1, B-polynomets i 1.3 och 1.

Identiteten (2.2.3) blir för systemet (4.5.1):

$$q^{-1} + \beta_2 = (1 + \alpha_1 q^{-1})(1 + \alpha_2 q^{-1})(f_0 + f_1 q^{-1} + f_2 q^{-2}) + \\ + q^{-2}(1 + \beta_2 q^{-1})(g_0 + g_1 q^{-1}) \qquad (4.5.2)$$

För att få den optimala regulatorn, skall vi alltså lösa ekv-systemet

$$\begin{cases} f_0 = \beta_2 \\ 1 = f_1 + f_0(\alpha_1 + \alpha_2) \\ 0 = f_2 + f_1(\alpha_1 + \alpha_2) + f_0\alpha_1\alpha_2 + g_0 \\ 0 = f_2(\alpha_1 + \alpha_2) + f_1\alpha_1\alpha_2 + g_0\beta_2 + g_1 \\ 0 = f_2\alpha_1\alpha_2 + \beta_2 g_1 \end{cases} \qquad (4.5.3)$$

Detta ger oss:

$$\begin{cases} f_0 = -1.3 \\ f_1 = -1.21 \\ f_2 = -4.6518 \\ g_0 = 3.5048 \\ g_1 = -2.5048 \end{cases}$$

dvs styrlagen för min-varians-regulatorn är:

$$u(t) = -\frac{G^*(q^{-1})}{B^{+*}(q^{-1})F^*(q^{-1})}\,y(t) = -\frac{g_0 + g_1q^{-1}}{-(1+\beta_1q^{-1})(f_0 + f_1q^{-1} + f_2q^{-2})}\,y(t-1) =$$

$$= \frac{3.5048 - 2.5048q^{-1}}{(1-q^{-1})(-1.3 - 1.21q^{-1} - 4.6518q^{-2})}\,y(t-1) \qquad (4.5.4)$$

Det slutna systemet är:

$$y(t) = \frac{F^*(q^{-1})}{B^-(q^{-1})}\,e(t) = \frac{f_0 + f_1q^{-1} + f_2q^{-2}}{b_2 + q^{-1}}\,e(t) =$$

$$= \frac{1.3q^2 + 1.21q + 4.6518}{1.3q^2 - q}\,e(t) \qquad (4.5.5)$$

På samma sätt som i avsnitt IV.4. har utsignalens varians beräknats: $Ey^2 = 62.$

Vi skall nu se hur de olika regulatorerna löser uppgiften att reglera systemet (4.5.1).

STURE1

Eftersom det betraktade systemet är icke-min-fas, så ger STURE1 ett instabilt slutet system. Några försök att förbättra regleringen med hjälp av olika tricks har inte heller här gjorts. För redogörelse av sådana försök hänvisas till {1}, sid 83 ff. Av ovanstående anledning presenteras inga diagram för STURE1.

STURE2, STUREM

Eftersom systemet (4.5.1) har ett C-polynom, som är identiskt lika med 1, väntas att STURE2 och STUREM skall ha samma egenskaper vid regleringen. Detta har också verifierats vid simuleringarna, och därför är resultaten i fortsättningen giltiga för båda regulatorerna. Systemet, som skall regleras, är stört av brus, som driver. Av denna anledning kan man misstänka, att styrsignalen ibland måste vara stor, för att regleringen skall bli bra. Därför har ingen begränsning av styrsignalen införts.

Vi skall börja "försiktigt" med att studera en simulering med P0=0 och Q2=0 samt med rätta initialvärden. Parameterestimaten ligger alltså helt rätt hela tiden, och som väntat blir regleringen bra. Av diagram 4.5.1 framgår att ovanstående förmodan om styrsignalen var riktig. Förlusten är nära den optimala - 63 per steg.

Vi inför nu lite osäkerhet på estimaten - P0 får värdet 0.01. Fortfarande är initialvärdena de riktiga. Estimaten svänger snabbt in sig, vilket framgår av diagram 4.5.2. Däremot är, trots de goda estimaten, regleringen dålig upp till ung punkt 1000 (se diagram 4.5.3). Förlusten är under dessa 1000 samplingsintervall i medeltal hela 5000 per steg. Samtidigt kan noteras, att Riccati-ekvationen måste itereras maximalt antal gånger - 10 - i nästan varje punkt. Efter dessa c:a 1000 punkter blir regleringen drastiskt bättre och förlusten blir bara c:a 60 per steg. Denna goda reglering har dock visat sig vara en tillfällighet. En simulering med P0=10 och i övrigt samma förutsättningar har gett samma uppförande, men till synes "omotiverat" fås en kraftig spik i styrsignalen och därmed utsignalen efter drygt 1000 samplingspunkter (se diagram 4.5.4). Också här har Riccati-ekvationen mycket svårt att konvergera.

Riccati-ekvationenes svårighet att konvergera skulle kunna förklaras med problemets art - ett nollställe till B-polynomet och därmed en pol till det slutna systemet ligger på enhetscirkeln. Den dåliga konvergensen skulle i sin tur kunna resultera i en endast positivt semidefinit matris S och därmed i en singulär term i ekv (2.2.7). Detta kan man tänka sig att försöka avhjälpa genom att sätta Q2 större än 0. Resultatet av denna åtgärd med Q2=0.1 ses i diagram (4.5.5) Som tidigare gäller P0=0.01 och riktiga initialvärden. Parameterestimaten blir snabbt bra och regleringen upp till ung punkt 1500 är också bra -

förlusten är c:a 70 per steg. Därefter uppträder emellertid samma fenomen som tidigare, och också i detta fall konvergerar iterationen av Riccati-ekvationen dåligt.

Ovanstående får anses vara resultat av en inledande undersökning av problemet. Ytterligare simuleringar redovisas inte. Som avslutning skall dock nämnas att vi försökt förbättra resultaten genom att skala problemet (dvs använt ett $RO < 1$) för att få det slutna systemets poler strängt innanför enhetscirkeln. Metoden har gett dåligt resultat-det principiella uppförandet är samma som tidigare - men är, tror vi, ändå värd att undersöka närmare.

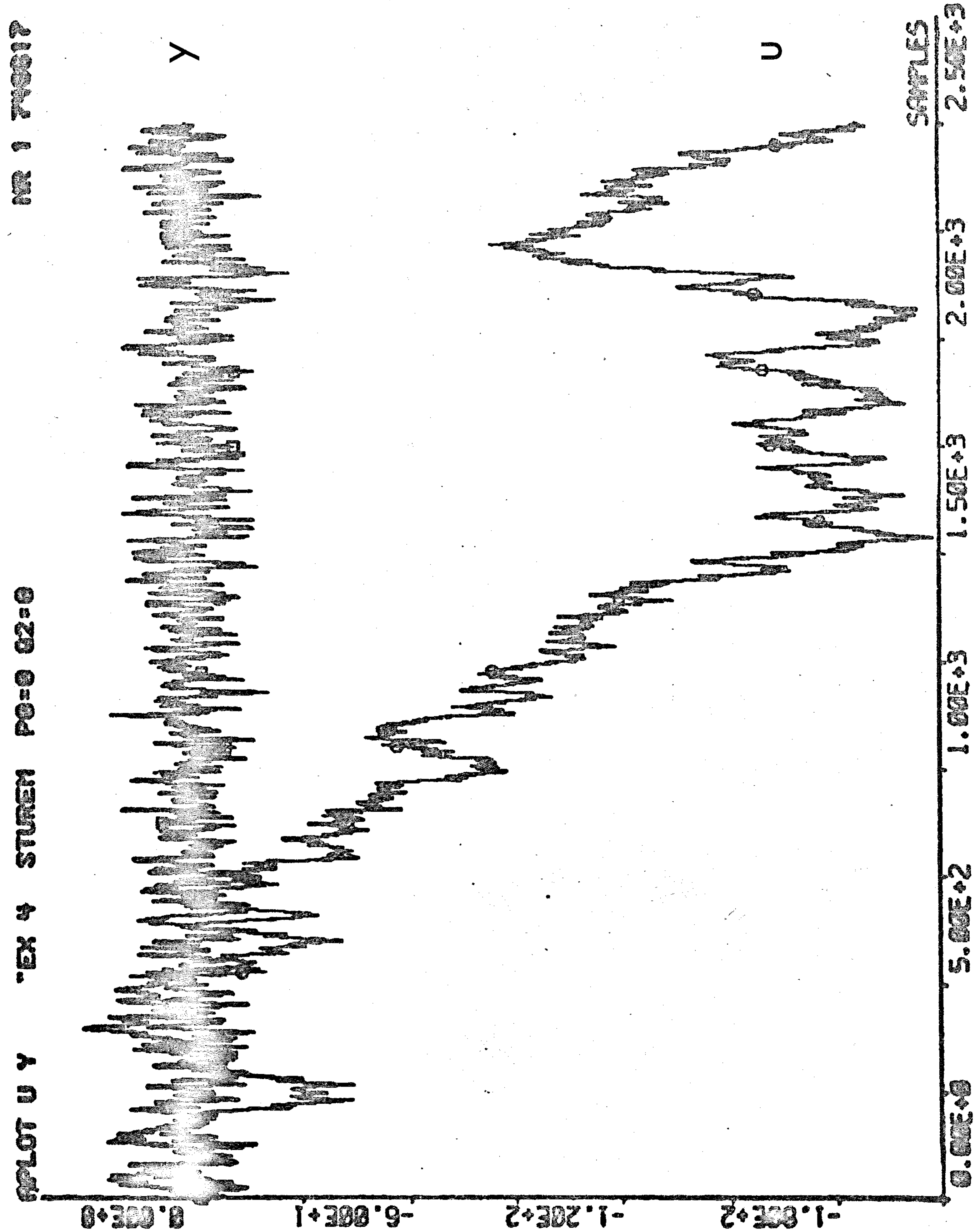

APLOT U Y
Y
U
SAMPLES
0.00E+0
5.00E+2
1.00E+3
1.50E+3
2.00E+3
2.50E+3
0.00E+0
-6.00E+1
-1.20E+2
-1.80E+2

Diagram 4.5.1

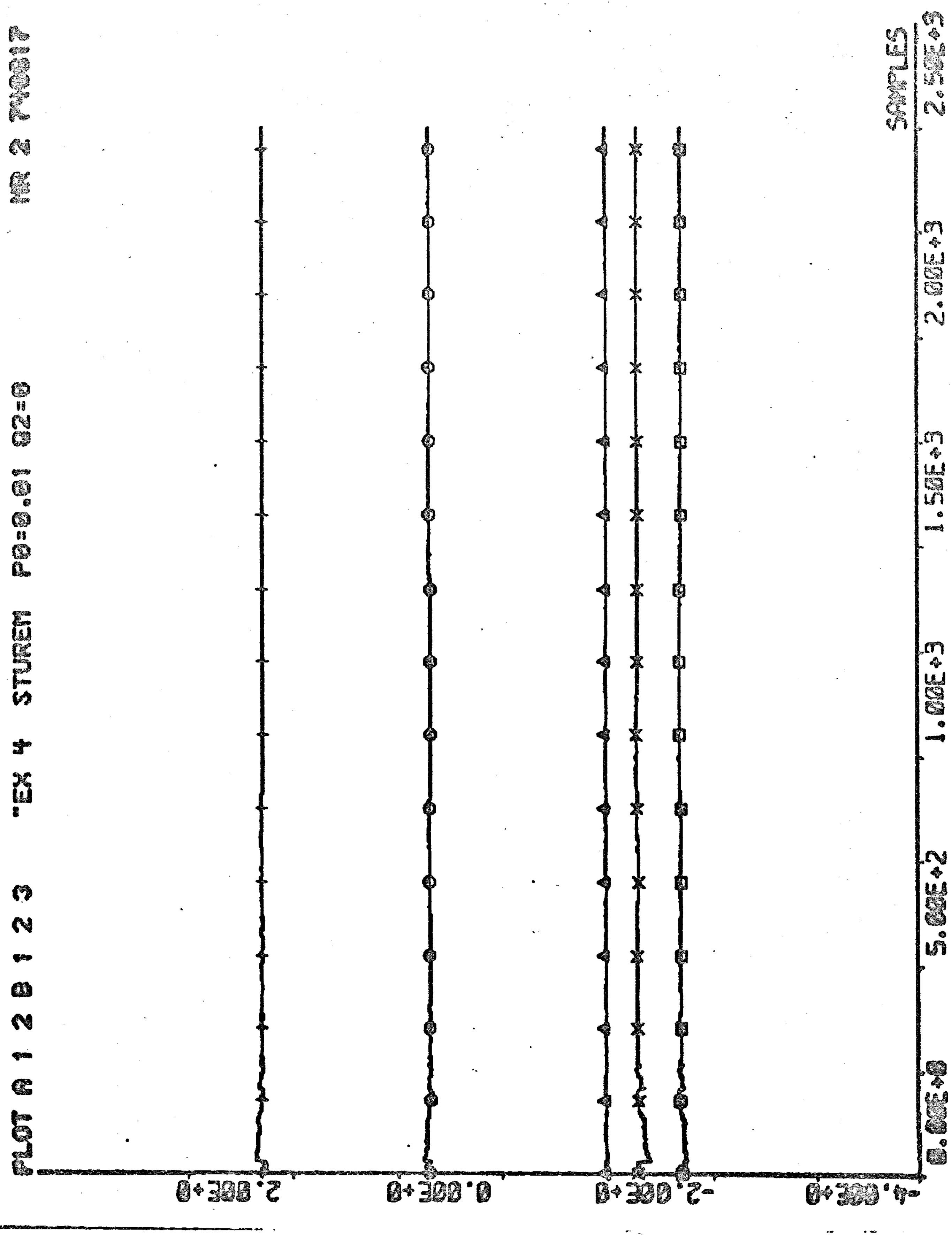

PLOT A 1 2 B 1 2 3 "EX 4 STUREM P0=0.01 B2=0 NR 2 790817
SAMPLES
0.00E+0
5.00E+2
1.00E+3
1.50E+3
2.00E+3
2.50E+3
2.00E+0
0.00E+0
-2.00E+0
-4.00E+0

Diagram 4.5.2

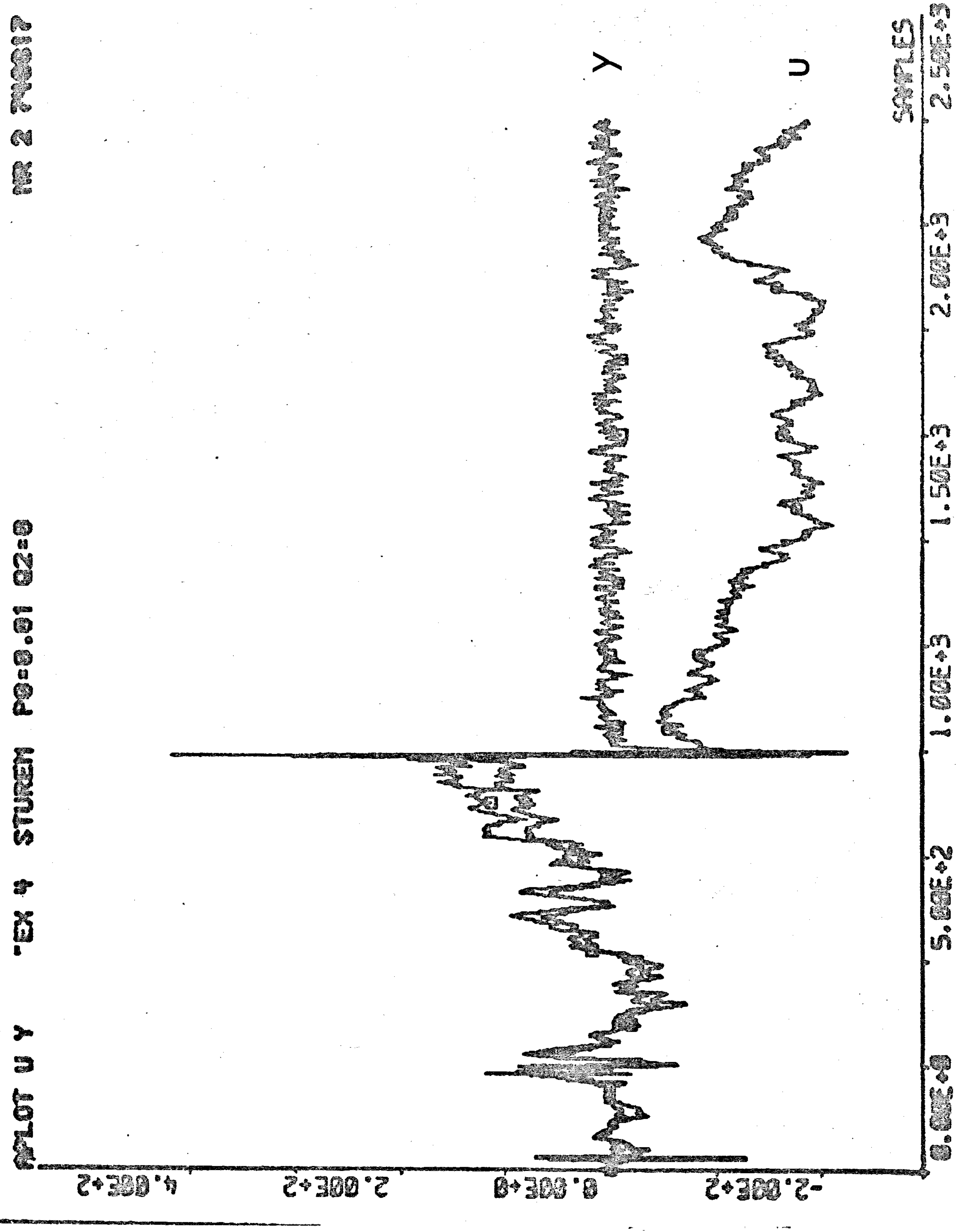

APLOT U Y
Y
U
SAMPLES
0.00E+0
5.00E+2
1.00E+3
1.50E+3
2.00E+3
2.50E+3
4.00E+2
2.00E+2
0.00E+0
-2.00E+2

Diagram 4.5.3

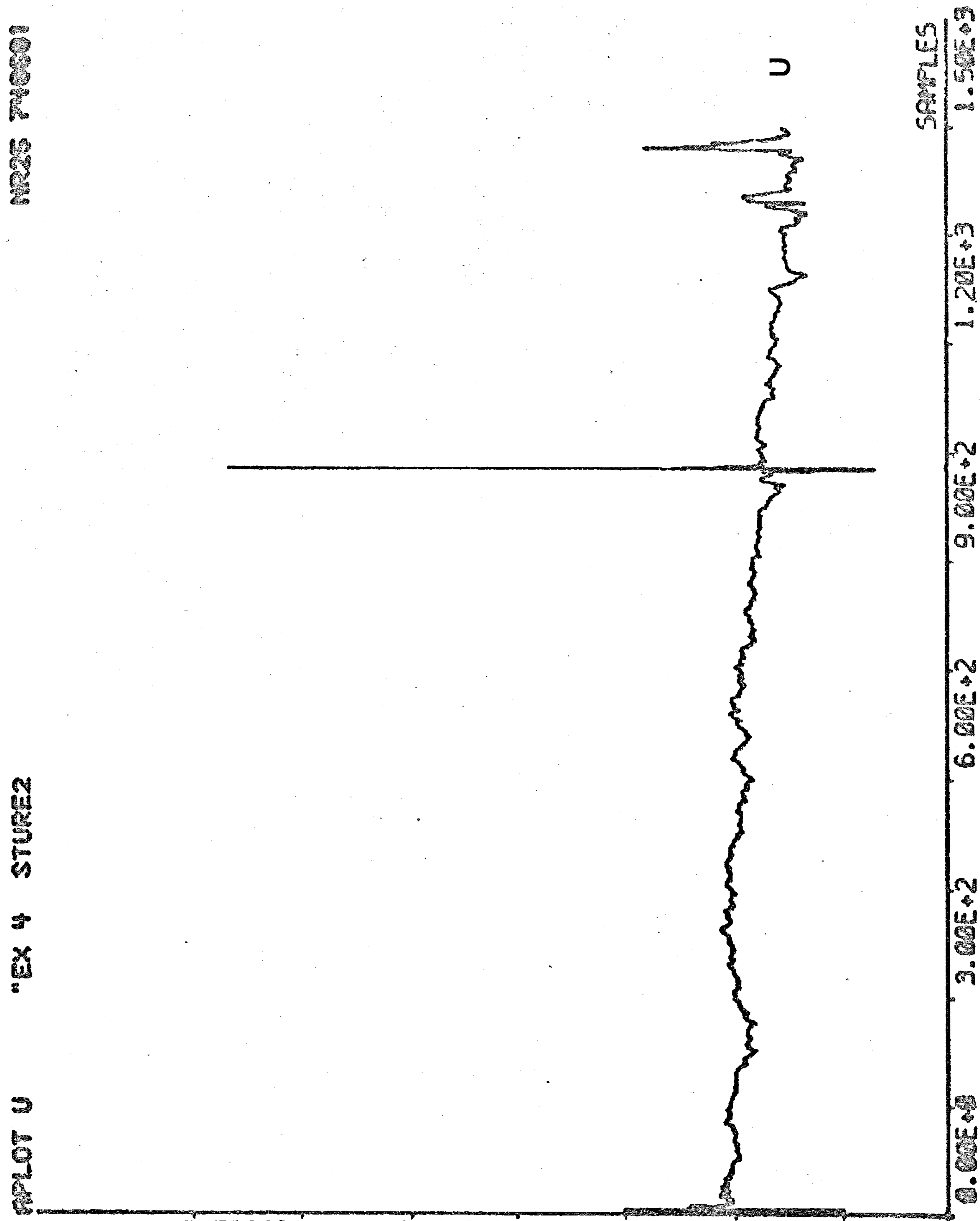

PLOT U
"EX 4 STURE2
U
SAMPLES
0.00E+0
3.00E+2
6.00E+2
9.00E+2
1.20E+3
1.50E+3
-5.00E+2
0.00E+0
5.00E+2
1.00E+3

Diagram 4.5.4

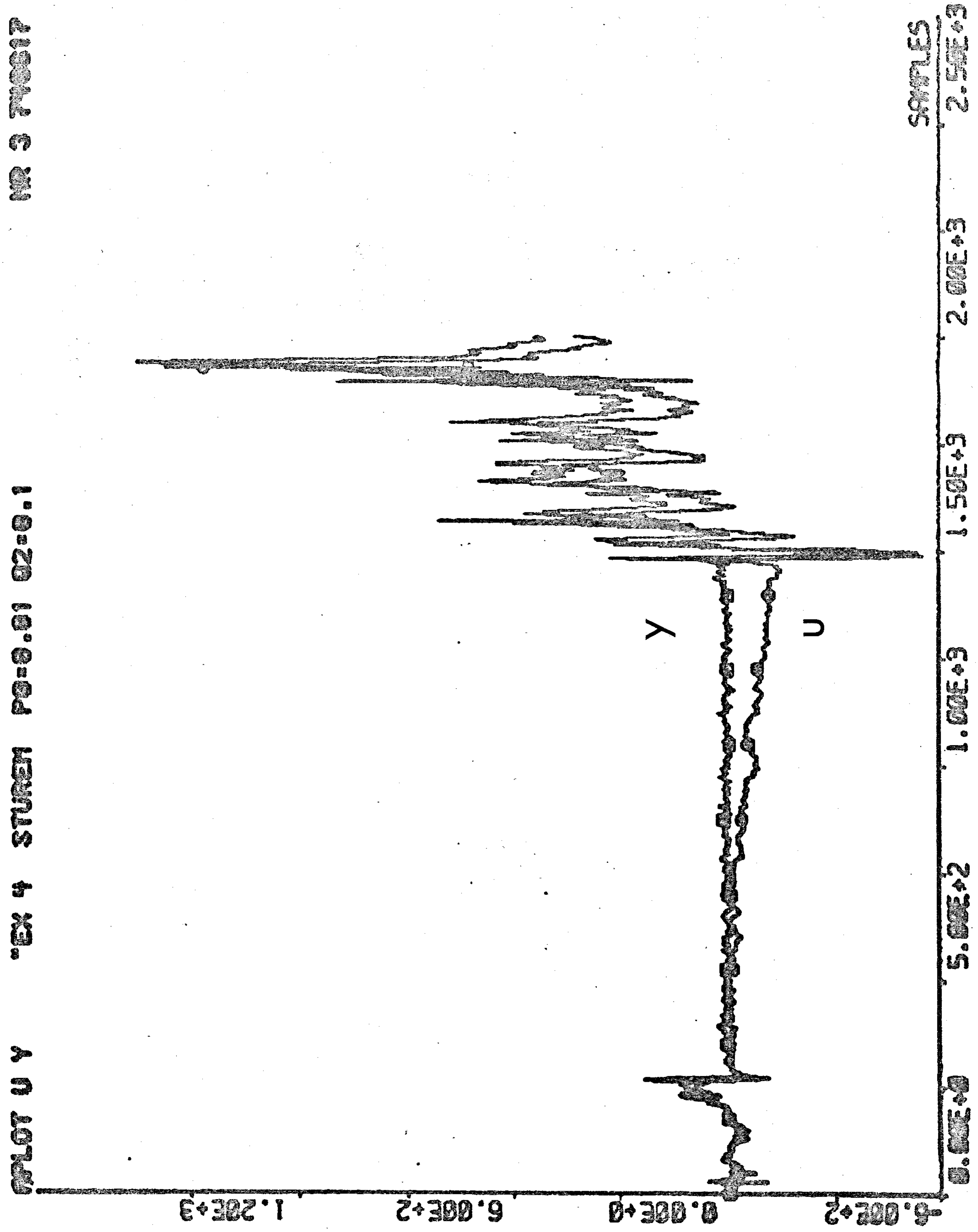

PPLOT U Y
STUREM
SAMPLES
0.00E+0
5.00E+2
1.00E+3
1.50E+3
2.00E+3
2.50E+3
1.20E+3
6.00E+2
0.00E+0
-6.00E+2
Y
U

Diagram 4.5.5

6. Sammanfattning

STURE1 är den regulator, som visat sig avgjort enklast och snabbast av de tre studerade. Det förefaller också som den är "robust" på så sätt att inställning av parametrar är okritisk. Den reglerar minimum-fas-system bra men på icke-min-fas-system är STURE2 och STUREM klart överlägsna, även om STURE1 kan fås att fungera genom olika tricks.

STURE2 och STUREM uppvisar i stort sett samma egenskaper - de är båda betydligt mer komplicerade än STURE1 och har därför längre räknetid men reglerar i gengäld även icke-min-fas-system (se dock ex. 4).
Den något längre räknetiden för STUREM jämfört med STURE2 förefaller vara marginell, varför STUREM på oss verkar vara att föredra. Detta särskilt som STURE2 inte estimerar c-parametrarna och därför ibland bara konvergerar mot en suboptimal regulator.

V. SLUTORD

Vi har nu redogjort för det programpaket, som utgör huvudresultatet av examensarbetet, och även visat hur det kan användas för att studera olika regulatorers egenskaper.
Naturligtvis finns det stort utrymme att förbättra paketet och därmed öka användarens möjligheter. Under arbetets gång har vi försökt göra paketet någorlunda flexibelt - sålunda är flera program förberedda för ytterligare kommandons införande. Detta gäller också s.k. direktkommandon (som tilldelar en parameter ett värde) med inhämtning och testning. Tänkbara önskemål på en ny version av STURE är

- handhavande av flera filer,
- flera sparade variabler,
- möjlighet att plotta uttryck i parameterestimat (t.ex. c_1-a_1)
- möjlighet att plotta referenslinjer,
- möjlighet att plotta valfritt antal punkter per sida,
- samtidig plottning.

Slutligen vill vi rikta ett varmt tack till våra handledare

- Staffan Selander, som tåligt understött oss med mycken programmeringskunskap,
- Torsten Söderström, som diskuterat mycket "regulatoruppförande" med oss och hjälpt oss få arbetet "i hamn", samt
- Björn Wittenmark, som introducerade oss i arbetet, och visade tålamod trots vår "tröga" start.

VI. REFERENSER

En utförligare presentation och diskussion av STURE1-algoritmen ges i

{1} B Wittenmark: A Self Tuning Regulator (Report 7311)

{2} KJ Åström/B Wittenmark: On Self Tuning Regulators (Report 7317)

STURE2 presenteras närmare i

{3} KJ Åström/B Wittenmark: Analysis of a Self Tuning Regulator for Non Minimum Phase Systems (IFAC Budapest sept 1974)

Identifieringsmetoden, använd i STUREM, återfinns i

{4} T Söderström: An On-line Algorithm for Approximate Maximum Likelihood Identification of Linear Dynamic Systems (Report 7308)

För beräkning av varianser har använts metoder enl

{5} Formelsamling i reglerteknik (Sigma-Tryck, TLTH)

Vidare information om detaljer i programpaketet ges i

{6} I Gustavsson/S Selander/J Wieslander: IDPAC, User's Guide (Report 7331)

{7} Programmerarhandbok PDP 15

APPENDICES

A1. Exempel på utskrifter

A2. Listning av programhuvuden

A1. Exempel på utskrifter

Som exempel på programpaketets resultatutskrifter har tagits en utskrift från en simulering av ex 1 med regulatorn STUREM. Simuleringen är exakt den som redovisas i avsnitt IV.2.
För att visa presentationen av ändrad parameter och CONT har följande kommandon getts efter det att alla parametrar fått sina värden:

```
SIMU 500
ITER 11
CONT 500
PRINT 10 50
```

S T U R E M
****

DATE: 74-06-18

SIMULATION NUMBER 1

ORDER OF SIMULATED SYSTEM	NORD:	1
TIME DELAY OF SIMULATED SYSTEM	KSYS:	0
A-POLYNOMIAL	APOL:	-0.95000
B-POLYNOMIAL	BPOL:	1.00000
C-POLYNOMIAL	CPOL:	-0.50000
STANDARD DEVIATION OF THE NOISE	DEVIA:	1.00000
ODD INTEGER FOR MCNODI	MACO:	19
REGULATOR: STUREM		
RADIUS OF POLE-ENCLOSING CIRCLE	RO:	1.00000
PENALTY ON CONTROL SIGNAL	Q2:	0.00000
TIME DELAY OF MODEL	KREG:	0
INITIAL VALUE OF COVARIANCE MATRIX	PO:	10.00000*I
EXPONENTIAL WEIGHTING FACTOR	RL:	0.99000
TIME DEPENDENCE OF RL	RLEXP:	0.01000
LIMIT ON RESIDUALS	RLIM:	10.00000
MAX NUMBER OF RICCATI ITERATIONS	ITER:	10
NUMBER OF A-PARAMETERS	NA:	1
NUMBER OF B-PARAMETERS	NB:	1
NUMBER OF C-PARAMETERS	NC:	1
INITIAL VALUES OF THE REGULATOR		
A-PARAMETERS	ALPHO:	0.00000
B-PARAMETERS	BETAO:	0.00000
C-PARAMETERS	GAMO:	0.00000
LIMIT ON THE CONTROL SIGNAL	ULIM:	10.00000
CONSTANT YREF	AMPL:	0.00000

	A 1	B 1	C 1	LOSS	U(T)	Y(T)
1	0.00000	0.00000	0.00000	2.7616	0.00000	-1.6618
2	-0.43679	0.00000	0.43679	4.9447	0.00000	-1.4775
3	-0.26391	0.00000	0.26391	5.0070	-10.000	0.24959
4	-0.32626	0.99046	0.19842	112.95	10.000	-10.390
5	-1.3208	0.89759	-0.52427	112.99	-1.4924	0.20342
6	-0.89145	0.96495	-0.12143E-01	114.51	-1.6306	-1.2321
7	-0.84777	1.0126	0.81051E-01	131.61	2.4421	-4.1352
8	-0.72305	1.0087	0.17273	131.70	0.96595	-0.30363
9	-0.74247	0.99399	0.13771	131.86	-0.64633	-0.39242
10	-0.76131	0.98244	0.10717	131.95	0.83062	-0.29876
50	-0.69922	0.87756	0.14910E-01	184.53	-0.26499	-0.88723
100	-0.66436	0.73202	-0.15141	228.07	-0.77180E-01	0.20949

150	-0.59339	0.66290	-0.18885	287.84	-0.59552	-0.27190
200	-0.52961	0.62621	-0.18763	317.43	0.12886	0.63468
250	-0.65338	0.63954	-0.26341	367.45	0.15618	0.16577
300	-0.66591	0.66147	-0.24737	435.95	-0.48550	0.31014
350	-0.61616	0.66539	-0.20743	503.54	-0.36830	1.4853
400	-0.62406	0.68056	-0.19520	572.35	-0.48566	0.78096
450	-0.65364	0.67879	-0.22013	623.33	0.14200	-0.42349E-01
500	-0.62147	0.69407	-0.17663	672.06	-0.61671	0.57764

LAST SAMPLE POINT WHERE THE RICCATI EQUATION DID NOT CONVERGE IN ITER STEPS: 0
TOTAL NUMBER OF SAMPLE POINTS WHERE THE RICCATI EQUATION DID NOT CONVERGE IN ITER STEPS: 0

MAX NUMBER OF RICCATI ITERATIONS ITER: 11

501	-0.61955	0.69434	-0.17474	672.24	0.44147E-01	-0.42011
502	-0.61957	0.69418	-0.17498	672.25	0.14707	0.98802E-01
503	-0.62013	0.69389	-0.17580	672.40	-0.10459	-0.39393
504	-0.62037	0.69305	-0.17707	672.92	0.20288	0.71566
505	-0.62018	0.69342	-0.17645	673.32	-0.37386	0.63705
506	-0.61652	0.69343	-0.17350	674.15	-0.86876E-01	-0.91158
507	-0.61754	0.68853	-0.18062	679.73	0.39764	2.3625
508	-0.61799	0.68551	-0.18488	680.08	-1.0989	0.58352
509	-0.61118	0.68696	-0.17759	681.07	0.24706	-0.99631
510	-0.61110	0.68690	-0.17760	681.13	0.27664	-0.25153
550	-0.65000	0.68537	-0.21003	719.24	0.88580E-01	-0.53313
600	-0.63655	0.69035	-0.19411	782.60	0.66873	-0.65747
650	-0.65038	0.69342	-0.20100	837.81	-0.11351	-0.48956
700	-0.64111	0.70328	-0.18226	907.42	0.21054	-0.67443
750	-0.63553	0.70525	-0.17583	971.99	0.36680	-1.8334
800	-0.65897	0.68721	-0.21681	1044.5	-0.25462E-01	-1.5273
850	-0.63175	0.68790	-0.19602	1089.9	0.25022	-0.99644E-01
900	-0.61362	0.68547	-0.18544	1158.0	0.12409E-01	-0.38853
950	-0.61866	0.69202	-0.18148	1221.1	0.42073	0.62050
1000	-0.61449	0.69245	-0.17802	1278.5	0.28699	1.0130

LAST SAMPLE POINT WHERE THE RICCATI EQUATION DID NOT CONVERGE IN ITER STEPS: 0
TOTAL NUMBER OF SAMPLE POINTS WHERE THE RICCATI EQUATION DID NOT CONVERGE IN ITER STEPS: 0

A2. Listning av programhuvuden

Följande sidor innehåller programhuvuden, hörande till program, som används speciellt i STURE-paketet. För information om övriga program, bl.a. kommandoavkodningsrutinerna, hänvisas till {6} och {7}.
Följande program ingår i listningen:

MAIN
BEGIN
GUIDE
SYST
REG
NOISE
YREF
INIT
PLOCOM
PLOT
ADPRI
PRINT
DISP
TEXT
SIMU
SYSTEM
STURE1
STURE2
STURE3
STUREM
RTMLE
ERROR
SETPAR
GETIN
CODE
DECODE
NUMER
EQU

```
C       MAIN PROGRAM FOR STURE - PROGRAM PACKAGE,
C       SIMULATING SELFTUNING REGULATORS
C
C       CONTENTS OF COMMON/PAR/:
C
C               NSAMP - NUMBER OF SAMPLE POINTS
C               NORD  - ORDER OF THE SYSTEM
C               KSYS  - TIME DELAY OF SIMULATED SYSTEM
C               APOL  - A-POLYNOMIAL OF THE SYSTEM
C               BPOL  - B-POLYNOMIAL OF THE SYSTEM
C               CPOL  - C-POLYNOMIAL OF THE SYSTEM
C               IRTYP - TYPE OF THE REGULATOR
C               B0    - PARAMETER BETA0 IN STURE1
C               KREG  - TIME DELAY OF THE MODEL
C               KDEL  - DELAY OF REGULATOR SIGNAL
C               NA    - NUMBER OF A-PARAMETERS
C               NB    - NUMBER OF B-PARAMETERS
C               NC    - NUMBER OF C-PARAMETERS
C               ALPH0 - INITIAL VALUES OF A-PARAMETERS
C               BETA0 - INITIAL VALUES OF B-PARAMETERS
C               GAM0  - INITIAL VALUES OF C-PARAMETERS
C               R0    - RADIUS OF POLE-ENCLOSING CIRCLE
C               Q2    - PENALTY ON CONTROL SIGNAL
C               P0    - INITIAL VALUE OF THE COVARIANCE MATRIX
C               RL    - EXPONENTIAL WEIGHTING FACTOR
C               ULIM  - LIMIT ON THE CONTROL SIGNAL
C               IYTYP - TYPE OF REFERENCE SIGNAL
C               YAMPL - AMPLITUDE OF REFERENCE SIGNAL
C               IYPER - PERIOD OF REFERENCE SIGNAL
C               DEVIA - STANDARD DEVIATION OF THE NOISE
C               MAC0  - ODD INTEGER FOR MCNODI
C               CHPAR - LOGICAL VECTOR INDICATING CHANGED PARAMETERS
C               ISAVE - VECTOR CONTAINING CODE NUMBERS OF SAVED VARIABLES
C               IDATE - VECTOR CONTAINING THE DATE
C               NSIM  - NUMBER OF THE CURRENT SIMULATION
C               RLEXP - TIME DEPENDENCE OF RL
C               RLIM  - LIMIT ON THE RESIDUALS
C               ITER  - MAX NUMBER OF RICCATI ITERATIONS
C               NRICC - NUMBER OF RICCATI NON-CONVERGENCES
C               LRICC - LAST POINT WITH RICCATI NON-CONVERGENCE
C
C       CONTENTS OF COMMON/SIM/: SEE SUBROUTINE SIMU
C
C       CONTENTS OF COMMON/COMINF/: SEE SUBROUTINE INTRAC
C
C       CONTENTS OF COMMON/MACINF/: SEE SUBROUTINE INTRAC
C
C       CONTENTS OF COMMON/RETAB/: SEE SUBROUTINE PLOT
C
C       CONTENTS OF COMMON/SYCO/: SEE SUBROUTINE PLOT
C
C       AUTHORS: B BENGTSSON/B EGARDT MAY 1974
C
C       SUBROUTINES REQUIRED:
C           INTRAC
C           GETIN
C           CODE
C           SYST
C           REG
C           NOISE
C           BEGIN
C           ERROR
C           SETPAR
```

```
C          YREF
C          INIT
C          SIMU
C          PLOCOM
C          ADPRI
C          DISP
C          GUIDE
C          TEXT
C
```

```
      SUBROUTINE BEGIN
C
C     USED AT START-UP. ASKS FOR DATE AND NUMBER OF FIRST
C     SIMULATION.
C
C     SUBROUTINES REQUIRED:
C          GETIN
C          SETPAR
C
C     AUTHORS: B BENGTSSON/B EGARDT   MAY 1974
C
```

```
      SUBROUTINE GUIDE
C
C     MAKES INITIAL COMMANDS AUTOMATICALLY
C
C     AUTHORS: B BENGTSSON/B EGARDT MAY 1974
C
C     SUBROUTINES REQUIRED:
C          SYST
C          REG
C          NOISE
C          YREF
C          INIT
C
```

```
      SUBROUTINE SYST
C
C     GETS VALUES OF THE FOLLOWING SYSTEM PARAMETERS:
C
C     NORD (ORDER OF THE SYSTEM)
C     KSYS (TIME DELAY OF THE SYSTEM)
C     APOL (A-POLYNOMIAL)
C     BPOL (B-POLYNOMIAL)
C     CPOL (C-POLYNOMIAL)
C
C     AUTHORS: B BENGTSSON/B EGARDT MAY 1974
C
C     SUBROUTINES REQUIRED:
C          GETIN
C
```

```
      SUBROUTINE REG
C
C     GETS THE FOLLOWING REGULATOR PARAMETERS:
C
C     IRTYP   (TYPE OF REGULATOR)
C     KREG    (TIME DELAY OF MODEL)
C     KDEL    (DELAY OF REGULATOR SIGNAL)
C     NA      (NUMBER OF ALPHA PARAMETERS)
C     NB      (NUMBER OF BETA PARAMETERS)
C     NC      (NUMBER OF GAMMA PARAMETERS)
C     R0      (RADIUS OF CIRCLE ENCLOSING ALL POLES)
C     Q2      (PENALTY ON INPUT SIGNAL)
C     B0      (PARAMETER BETA0 IN STURE1)
C     RLEXP   (TIME DEPENDENCE OF RL
C     RESLIM  (LIMIT ON RESIDUALS)
C     ULIM    (LIMIT ON THE CONTROL SIGNAL)
C     RL      (EXPONENTIAL WEIGHTING FACTOR)
C     P0      (INITIAL VALUE OF THE VARIANCE MATRIX)
C
C     AUTHORS: B BENGTSSON/B EGARDT MAY 1974
C
C     SUBROUTINES REQUIRED:
C          GETIN
C
```

```
      SUBROUTINE NOISE
C
C     GETS THE FOLLOWING PARAMETERS OF THE NOISE:
C
C     MACO  (ODD INTEGER FOR MCNODI)
C     DEVIA (STANDARD DEVIATION OF THE NOISE)
C
C     AUTHORS: B BENGTSSON/B EGARDT MAY 1974
C
C     SUBROTINES REQUIRED:
C          GETIN
C
```

```
      SUBROUTINE YREF
C
C     GETS THE FOLLOWING PARAMETERS OF THE REFERENCE SIGNAL:
C
C     IYTYP (TYPE OF REFERENCE)
C     YAMPL (AMPLITUDE)
C     IYPER (PERIOD)
C
C     AUTHORS: B BENGTSSON/ B EGARDT MAY 1974
C
C     SUBROUTINES REQUIRED:
C          GETIN
C
```

```
      SUBROUTINE INIT
C
C     GETS THE FOLLOWING INITIAL VALUES OF THE PARAMETER ESTIMATES:
C
C     ALPH0
C     BETA0
C     GAM0
C
C     AUTHORS: B BENGTSSON/B EGARDT MAY 1974
C
C     SUBROUTINES REQUIRED:
C          GETIN
C
```

```
      SUBROUTINE DISP
C
C     DISPLAYS THE PARAMETER VALUES
C
C     AUTHORS: B BENGTSSON/B EGARDT MAY 1974
C
C     SUBROUTINES REQUIRED:
C          DECODE
C
```

```
      SUBROUTINE TEXT
C
C     READS A TEXT STRING FROM INPUT (TELETYPE OR MACRO)
C     AND PRINTS IT ON TELETYPE, DISPLAY OR LINE PRINTER.
C     AN ALT MODE LINE TERMINATOR RETURNS CONTROL TO THE
C     MAIN PROGRAM.
C
C     COMMAND
C     TEXT DEV [FF]
C     DEV      - OUTPUT DEVICE (TT, TV OR LP)
C     FF       - INDICATES THAT A FORM FEED SHALL BE MADE
C
C     AUTHOR STAFFAN SELANDER 1974-04-08
C     REV BB/BE MAY 1974
C
C     SUBROUTINES REQUIRED
C             ARGIN
C             INTRAC
C             GAC
C             PAC
C
```

```
      SUBROUTINE ADPRI(N1,N2)
C
C     ADMINISTRATES THE WORK REQUESTED BY THE COMMAND: PRINT N1 N2
C
C     IN THE FILE SIMFI THE DATA OF THE SIMULATION(S) ARE STORED.
C     IN THE FILE CTLG INFORMATION ABOUT SIMFI IS STORED.
C     FOR FURTHER INFORMATION: SEE SUBROUTINE SIMU.
C     ADPRI READS THE PARAMETER VALUES FROM CTLG AND MAKES AN
C     APPROPRIATE CALL FOR PRINT.
C     USES DUM1,DUM2 OF COMMON/SLASK/ TO TRANSFER THE PARAMETER
C     VALUES TO PRINT.
C
C     N1,N2  -  SEE SUBROUTINE PRINT
C
C     AUTHORS: B BENGTSSON/B EGARDT MAY 1974
C
C     SUBROUTINES REQUIRED:
C          FILES
C          PRINT
C
```

```
      SUBROUTINE PRINT(N1,N2,NPRINT,GT9LOG,PRILOG)
C
C     FOR EACH SIMULATION COMMAND THIS SUBROUTINE
C     PRINTS THE VALUES OF THE SAVED VARIABLES IN
C     THE FIRST N1 POINTS AND THEN IN EVERY N2:TH POINT.
C
C     FIRST THE INITIAL PARAMETER VALUES ARE ALL LISTED
C     THEN ONLY THOSE PARAMETERS ARE LISTED THAT HAVE BEEN
C     CHANGED SINCE THE LAST CONT-COMMAND.
C
C     FLAGS HAVE BEEN SET IN CHPAR CORRESPONDING TO CHANGED PARA-
C     METERS. THE CODE IN CHPAR:
C
C     CHPAR(1)=.TRUE.       -    NORD HAS BEEN CHANGED
C           2               -    KSYS
C           3               -    APOL
C           4               -    BPOL
C           5               -    CPOL
C           6               -    DEVIA
C           7               -    B0
C           8               -    R0
C           9               -    Q2
C          10               -    KREG
C          11               -    KDEL
C          12               -    RL
C          13               -    RLEXP
C          14               -    RLIM
C          15               -    ITER
C          16               -    ULIM
C          17               -    IYTYP,YAMPL,IYPER
C          18               -    IRTYP
C          19               -    NA
C          20               -    NB
C          21               -    NC
C          22               -    ISAVE
C
C     NPRINT                -    NUMBER OF SAMPLE POINTS ALREADY
C                                WRITTEN.
C
C     GT9LOG                -  RETURNED .TRUE. IF MORE THAN 9
C                              VARIABLES HAVE BEEN SAVED
C
C     PRILOG                -  .TRUE. IF THE FIRST 9 SAVED
C                              VARIABLES HAVE ALREADY BEEN PRINTED
C
C     DUM1 AND DUM2 OF COMMON/SLASK/ ARE USED TO TRANSFER DATA
C     FROM THE ADMINISTRATING PROGRAM ADPRI TO PRINT. DUM1 AND
C     DUM2 COULD BE USED FOR OTHER PURPOSES OUTSIDE ADPRI/PRINT.
C
C     AUTHORS: B BENGTSSON/B EGARDT MAY 1974
C
C     SUBROUTINES REQUIRED:
C         FILES
C         DECODE
C
```

```
      SUBROUTINE PLOCOM(APLOT)
C
C     MAKES A COMMAND WHICH SUITS SUBROTINE PLOT AND CALLS PLOT
C
C     COMMAND SHOULD BE OF TYPE:
C
C     [A]PLOT [VNAME1 [VNR11..] [VNAME2 [VNR21..]..] [SC YMIN YMAX]]
C
C     IF NO VARIABLES ARE TYPED ALL SAVED VARIABLES WILL BE PLOTTED
C     THE COMMAND APLOT GIVES AUTOMATIC SCALING
C     THE COMMAND PLOT GIVES USERS SCALING
C        (THE SCALES ARE GIVEN IN THE COMMAND OR - IF SC YMIN YMAX
C         IS OMITTED - BY THE RESERVED VARIABLES YMIN,YMAX.
C         WHEN THE SCALES ARE GIVEN IN THE COMMAND ONLY THE CURRENT
C         PLOT IS AFFECTED)
C
C     APLOT   -    .TRUE. : AUTOMATIC SCALING
C                  .FALSE.: USERS SCALING (LIMITS GIVEN IN COMMAND
C                           OR BY YMIN,YMAX IN COMMON/RETAB/)
C     THE PLOT OF U WILL BE OF HISTOGRAM TYPE
C
C     AUTHORS: B BENGTSSON/B EGARDT MAY 1974
C
C     SUBROUTINES REQUIRED:
C          FILES
C          NUMER
C          GAC
C          PAC
C          CODE
C          PLOT
C
```

```
      SUBROUTINE PLOT
C
C     PLOTS CURVES ON DISPLAY FROM DISK RESIDENT FILES
C
C     THE RESERVED VARIABLES NPLX,YMAX AND YMIN DETERMINES
C     THE NUMBER OF DATA POINTS TO BE PLOTTED PER PAGE AND
C     THE SCALES TO BE USED. IF BOTH YMIN AND YMAX ARE SET
C     TO ZERO THE PROGRAM WILL CHOOSE THE SCALES. IT IS
C     POSSIBLE TO ADD VALUES FOR YMIN AND YMAX TO THE
C     COMMAND. THESE VALUES WILL ONLY AFFECT THE SCALES
C     ON THE CURRENT PLOT AND WILL NOT CHANGE THE VALUES
C     OF THE RESERVED VARIABLES.
C
C     NORMALLY THE CURVES ARE PLOTTED VERSUS SAMPLE NUMBER
C     BUT IF THE COMMAND TIME S/M/H HAS BEEN GIVEN THE
C     CURVES ARE PLOTTED VERSUS TIME (SECONDS, MINUTES
C     OR HOURS).
C
C     WHEN MORE THAN ONE CURVE IS PLOTTED THE CURVES ARE
C     MARKED BY
C     SQUARE, OCTAGON, TRIANGLE, PLUS, CROSS, ASTERISK,
C     HORIZ. BAR AND VERT. BAR RESP.
C
C     COMMAND:
C     PLOT ['HP'] FNAM1[(C11 ..)] [['HP'] FNAM2 ..] [YMIN YMAX]
C     HP       - INDICATES THAT THE FOLLOWING FILE SHALL
C                BE PLOTTED AS A HISTOGRAM
C     YMIN     - MINIMUM VALUE FOR THIS PLOT
C     YMAX     - MAXIMUM VALUE FOR THIS PLOT
C     WITH YMIN AND YMAX OMITTED THE SCALING IS DETERMINED
C     BY THE RESERVED VARIABLES YMIN AND YMAX.
C
C     AUTHOR STAFFAN SELANDER 1972-09-22
C     REVISED STAFFAN SELANDER 1973-06-08
C     REVISED STAFFAN SELANDER 1973-11-01
C     REVISED STAFFAN SELANDER 1974-02-19
C
C     SUBROUTINES REQUIRED:
C             ARGIN
C             NEWPIC
C             (FILES)
C             (ERASE)
C             (AXIS)
C             (SCALE)
C             (LINE)
C
```

```
C
      SUBROUTINE SIMU(ITYPE,ITIME)
C
C     SIMULATES THE CLOSED LOOP SYSTEM AND STORES THE RESULT IN THE
C     FILE SIMFI. PARAMETER VALUES ARE STORED IN THE FILE CTLG.
C     DEPENDING ON THE COMMAND TWO DIFFERENT TYPES OF SIMULATION
C     CAN BE MADE:
C     SIMU  - A NEW SIMULATION IS DEFINED (THE OLD ONE IS NOT STORED)
C     CONT  - THE SIMULATION IS CONTINUED WITH OLD VALUES AS START
C             VALUES (INFORMATION ABOUT CHANGED PARAMETER VALUES IN CTLG).
C             SEVERAL CONT COMMANDS ARE PERMITTED. SIMU MUST BE CALLED
C             BEFORE CONT. NO NEW SAVE COMMAND IS PERMITTED BETWEEN
C             SIMU AND CONT COMMANDS.
C     THE FILES ARE ORGANIZED AS FOLLOWS:
C     SIMFI - EVERY ROW CONTAINS VALUES OF THE VARIABLES TO BE SAVED
C             (CODE NUMBERS OF THESE VARIABLES ARE CONTAINED IN THE
C             VECTOR ISAVE, WHERE ISAVE(1)=NUMBER OF SAVED VARIABLES
C             (MAX 15); THE CODE NUMBERS ARE LISTED IN SUBROUTINE CODE)
C     CTLG  - THE FIRST ROW CONTAINS THE VALUES OF THE PARAMETERS
C             IN COMMON/PAR/ AT THE TIME FOR SIMU COMMAND.
C             THE SUBSEQUENT ROWS CONTAIN VALUES OF THOSE PARAMETERS,
C             WHICH HAVE BEEN CHANGED SINCE THE LAST SIMU/CONT COMMAND.
C
C     ITYPE - 1: SIMU COMMAND
C             2: CONT COMMAND
C     ITIME - TOTAL NUMBER OF SAMPLE POINTS SINCE THE INITIAL
C             SIMU COMMAND UNTIL THIS SIMULATION (NSAMP IN THIS
C             SIMULATION IS NOT INCLUDED)
C
C     CONTENTS OF COMMON/SIM/:
C
C     U     - VECTOR OF PROCESS INPUTS OF DIMENSION NB+K+1
C             U(1)=U(T)
C             . . .
C             U(NB+K+1)=U(T-NB-K)
C     Y     - VECTOR OF PROCESS OUTPUTS OF DIMENSION NA+K+1
C             Y(1)=Y(T)
C             . . .
C             Y(NA+K+1)=Y(T-NA-K)
C     YSC   - VECTOR OF SCALED PROCESS OUTPUTS OF DIMENSION NA+K+1
C             YSC(1)=Y(T)-YREF   (STURE1: (Y(T)-YREF)/B0)
C             . . .
C             YSC(NA+K+1)=Y(T-NA-K)-YREF  (STURE1: (Y(T-NA-K)-YREF)/B0)
C     E     - VECTOR OF RESIDUALS OF DIMENSION NC
C             E(1)=E(T)
C             . . .
C             E(NC)=E(T-NC)
C     TETA  - VECTOR OF ESTIMATED PARAMETERS OF DIMENSION NA+NB+NC
C             TETA(1)=ALPHA(1)
C             . . .
C             TETA(NA)=ALPHA(NA)
C             TETA(NA+1)=BETA(1)
C             . . .
C             TETA(NA+NB)=BETA(NB)
C             TETA(NA+NB+1)=GAMMA(1)
C             . . .
C             TETA(NA+NB+NC)=GAMMA(NC)
C     RYY   - VECTOR OF COVARIANCES OF DIMENSION 11
C             RYY(1)=SUM(Y(T)Y(T)) = LOSS FUNCTION
C             . . .
C             RYY(11)=SUM(Y(T)Y(T-10))
C     RYU   - VECTOR OF COVARIANCES OF DIMENSION 11
C             RYU(1)=SUM(Y(T)U(T))
C             . . .
```

```
C               RYU(11)=SUM(Y(T)U(T-10))
C        P      - VARIANCE MATRIX OF THE PARAMETER ESTIMATES
C                 OF ORDER (NA+NB+NC)
C        S      - SOLUTION OF THE RICCATI EQUATION
C                 OF ORDER MAX(NA,(NB+K),NC)
C        MAC    - ODD INTEGER FOR MCNODI
C        YSUM   - SUM(Y(T))
C        UOLD   - OLD INPUT VALUE (STUREM)
C        YOLD   - OLD OUTPUT VALUE (STUREM)
C        RESOLD- OLD RESIDUAL (STUREM)
C        RLNEW - TIME DEPENDENT EXPONENTIAL WEIGHTING
C                FACTOR (STUREM)
C        RESID - VECTOR OF ESTIMATED RESIDUALS (STUREM)
C        X      - STATE VECTOR (STUREM)
C        Z      - STATE VECTOR (STUREM)
C
C        MAXIMUM VALUES:
C        SYSTEM: NORD=5,KREG=10
C        STURE1: NA=6,NB=6,KREG=10
C        STURE2,3,M: NA=6,NB=6,NC=6,NB+KREG=8
C
C        AUTHORS: B BENGTSSON/B EGARDT MAY 1974
C
C        SUBROUTINES REQUIRED:
C             SYSTEM
C             STURE1
C             STURE2
C             STURE3
C             STUREM
C             FILES
C             MOVE
C
```

```
         SUBROUTINE SYSTEM
C
C        COMPUTES THE OUTPUT FROM THE SYSTEM
C
C        MODEL:Y(T)+A(1)*Y(T-1)+ .....+A(NORD)*Y(T-NORD)=
C        B(1)*U(T-KSYS-1)+ ... +B(NORD)*U(T-KSYS-NORD)+
C        +E(T)+C(1)*E(T-1)+ ... +C(NORD)*E(T-NORD)
C        WHERE (E(J)) IS WHITE NOISE WITH
C        STANDARD DEVIATION DEVIA.
C
C        AUTHORS: B BENGTSSON/B EGARDT MAY 1974
C
C        SUBROUTINES REQUIRED:
C             MOVE
C             SCAPRO
C
```

```
      SUBROUTINE STURE1(RES,DENOM)
C
C     SELFTUNING REGULATOR BASED ON LEAST SQUARES IDENTIFICATION
C     AND MINIMUM VARIANCE CONTROL.
C     THE ALGORITHM IS BASED ON THE MODEL:
C
C     Y(T+K+1)+A(1)*Y(T-1)+...+A(NA)*Y(T-NA)=
C     B0*(U(T)+B(1)*U(T-1)+...+B(NB)*U(T-NB))+E(T+K+1)     (*)
C
C     AT EACH STEP THE LEAST SQUARES ESTIMATES OF THE MODEL
C     PARAMETERS ARE COMPUTED. THE PROCESS INPUT U(T+1) TO BE
C     APPLIED TO THE PROCESS AT TIME T+1 IS THEN COMPUTED FROM
C
C     U(T+1)=(AE(1)*Y(T)+...+AE(NA)*Y(T-NA+1))/B0-
C           -BE(1)*U(T)-...-BE(NB)*U(T-NB+1)
C     WHERE AE AND BE ARE THE PARAMETER ESTIMATES.
C
C     WHEN APPLYING THE ALGORITHM THE PROCESS OUTPUT IS THUS READ
C     AT TIME T. THE PROCESS INPUT U(T+1) TO BE APPLIED AT TIME
C     T+1 IS THEN COMPUTED AT THE TIME INTERVALL (T,T+1)
C
C     RES-RESIDUAL
C     DENOM-THE FACTOR 1+FI*P*PIT IN RTLSID
C
C     USES DUM7 AND DUM8 (RTLSID) OF COMMON SLASK
C
C     AUTHORS: K.J. ASTROM/B. WITTENMARK AUG 71,REV 72.08.02 BW
C     REVISED B BENGTSSON/B EGARDT  MAY 1974
C
C     SUBROUTINES REQUIRED:
C           RTLSID
C           SCAPRO
C           MOVE
C
```

```
      SUBROUTINE STURE2(IND)
C
C     SELFTUNING REGULATOR BASED ON LEAST SQUARES IDENTIFICATION
C     AND MINIMUM VARIANCE CONTROL
C     THE ALGORITHM IS BASED ON THE MODEL
C
C     Y(T)+A(1)*Y(T-1)+...+A(NA)*Y(T-NA)=
C     B(1)*U(T-K-1)+...+B(NB)*U(T-K-NB)      (*)
C
C     AT EACH STEP THE LEAST SQUARES ESTIMATES OF THE MODEL
C     PARAMETERS ARE COMPUTED. THE PROCESS INPUT U(T+1) TO BE
C     APPLIED TO THE PROCESS AT TIME T+1 IS THEN COMPUTED
C     FROM THE SOLUTION OF THE RICCATI EQUATION WHICH MINIMIZES
C     SUM Y(T)**2 UNDER THE CONSTRAINT THAT ALL POLES OF
C     THE CLOSED LOOP SYSTEM ARE WITHIN THE CIRCLE WITH RADIUS RO
C
C     WHEN APPLYING THE ALGORITHM THE PROCESS OUTPUT IS THUS READ
C     AT TIME T. THE PROCESS INPUT U(T+1) TO BE APPLIED AT TIME T+1
C     IS THEN COMPUTED AT THE TIME INTERVAL (T,T+1)
C
C     IND-  1: THE RICCATTI EQUATION HAS NOT CONVERGED AFTER
C              ITER STEPS
C          -N: CONVERGENCE AFTER N STEPS
C
C
C     USES DUM5-DUM8 OF COMMON /SLASK/
C
C     AUTHOR KJ ASTROM 72-01-11  REV 72-07-13 BW
C          REV 05-74 BB/BE
C
C     SUBROUTINES REQUIRED:
C          RTLSID
C          SCAPRO
C          MOVE
C          CORI
C          NORM
C
```

```
      SUBROUTINE STURE3(IND)
C
C     SELFTUNING REGULATOR BASED ON LEAST SQUARES IDENTIFICATION
C     AND MINIMUM VARIANCE CONTROL
C     THE ALGORITHM IS BASED ON THE MODEL
C
C     Y(T)+A(1)*Y(T-1)+...+A(NA)*Y(T-NA)=
C     B(1)*U(T-K-1)+...+B(NB)*U(T-K-NB)        (*)
C
C     AT EACH STEP THE LEAST SQUARES ESTIMATES OF THE MODEL
C     PARAMETERS ARE COMPUTED. THE PROCESS INPUT U(T) TO BE
C     APPLIED TO THE PROCESS AT TIME T IS THEN COMPUTED
C     FROM THE SOLUTION OF THE RICCATI EQUATION WHICH MINIMIZES
C     SUM Y(T)**2 UNDER THE CONSTRAINT THAT ALL POLES OF
C     THE CLOSED LOOP SYSTEM ARE WITHIN THE CIRCLE WITH RADIUS RO
C
C     WHEN APPLYING THE ALGORITHM THE PROCESS OUTPUT IS THUS READ
C     AT TIME T. THE PROCESS INPUT U(T) TO BE APPLIED AT TIME T
C     IS THEN IMMEDIATELY COMPUTED. I.E. THERE IS NO TIME
C     DELAY IN THE REGULATOR.
C
C     IND-  1: THE RICCATTI EQUATION HAS NOT CONVERGED AFTER
C              ITER STEPS
C          -N: CONVERGENCE AFTER N STEPS
C
C
C     USES DUM5-DUM8 OF COMMON /SLASK/
C
C     AUTHOR B WITTENMARK 72-08-02  REV 05-74 BB/BE
C
C     SUBROUTINES REQUIRED:
C          RTLSID
C          SCAPRO
C          MOVE
C          CORI
C          NORM
C
```

```
      SUBROUTINE STUREM(IND)
C
C     SELF-TUNING REGULATOR BASED ON MAXIMUM LIKELIHOOD IDENTIFICATION
C     AND MINIMUM VARIANCE CONTROL
C
C     THE ALGORITHM IS BASED ON THE MODEL:
C
C     Y(T)+A(1)*Y(T-1)+...+A(NA)*Y(T-NA)=
C     B(1)*U(T-KREG-1)+...+B(NB)*U(T-KREG-NB)+
C     E(T)+C(1)*E(T-1)+...+C(NC)*E(T-NC)
C
C     AT EACH STEP THE MAXIMUM LIKELIHOOD ESTIMATES OF THE
C     MODEL PARAMETERS ARE COMPUTED. THE PROCESS INPUT U(T+1)
C     TO BE APPLIED TO THE PROCESS AT TIME T+1 IS THEN COMPUTED
C     FROM THE SOLUTION OF THE RICCATI EQUATION WHICH MINIMIZES
C     SUM Y(T)**2 UNDER THE CONSTRANT THAT ALL POLES OF THE
C     CLOSED LOPP SYSTEM ARE WITHIN THE CIRCLE WITH RADIUS RO.
C
C     THE PROCESS OUTPUT IS READ AT TIME T.
C     DURING THE TIME INTERVAL (T,T+1) THE PROCESS
C     INPUT U(T+1) IS COMPUTED.
C
C     IND   1: THE RICCATI EQUATION HAS NOT CONVERGED AFTER ITER STEPS
C          -N: COVERGENCE AFTER N STEPS
C
C     AUTHORS:     B BENGTSSON / B EGARDT    MAY 1974
C
C     USES DUM3 - DUM6 OF COMMON/SLASK/
C
C     SUBROUTINES REQUIRED:
C          RTMLE
C          SCAPRO
C          MOVE
C          CORI
C          NORM
C
```

```
      SUBROUTINE RTMLE
C
C     PERFORMS REAL TIME MAXIMUM LIKELIHOOD ESTIMATION
C     THE FOLLOWING MODEL IS USED:
C
C     A(Q)*Y(T) = B(Q)*U(T)+C(Q)*E(T)
C
C     USES DUM3 - DUM5 OF COMMON/SLASK/
C
C     AUTHORS:  B BENGTSSON / B EGARDT MAY 1974
C
C     SUBROUTINES REQUIRED:
C          INPOL
C          MOVE
C          SCAPRO
C
```

```
      SUBROUTINE ERROR
C
C     PRINTS ERROR MESSAGES ON THE TELEPRINTER
C
C     IERR  -  ERROR INDICATOR
C              <0: INTERNAL ERROR
C               0: NO ERROR
C              >0: ERROR FROM INTRAC
C
C     AUTHORS:  B BENGTSSON/B EGARDT  MAY 1974
C
C     SUBROUTINES REQUIRED:
C           NONE
C



      SUBROUTINE SETPAR(ICNT,PNAME,NR,ITYP)
C
C     SUBROUTINE TO SET PARAMETER VALUES
C     TESTS IF THE GIVEN VALUE IS ALLOWED
C     AND GIVES ERROR MESSAGES
C
C     ICNT  - 1: NO COMMAND BEFORE NUMBER
C             2: COMMAND BEFORE NUMBER
C     PNAME - NAME OF THE PARAMETER
C     NR    - NUMBER OF THE PARAMETER IN
C             THE FOLLOWING LIST:
C             (NB. THE PARAMETERS 1 - 25
C             CORRESPONDS TO COMMANDS NBR 25 - 50
C             (SEE MAIN PROGRAM))
C             KSYS    -  1
C             KREG    -  2
C             KDEL    -  3
C             NA      -  4
C             NB      -  5
C             NC      -  6
C             R0      -  7
C             Q2      -  8
C             B0      -  9
C             RL      - 10
C             RLEXP   - 11
C             RLIM    - 12
C             ULIM    - 13
C             P0      - 14
C             ITER    - 15
C             YMIN    - 16
C             YMAX    - 17
C             (FREE)  - 18-25
C             NORD    - 26
C             IRTYP   - 27
C             MAC0    - 28
C             DEVIA   - 29
C             IYTYP   - 30
C             YAMPL   - 31
C             IYPER   - 32
C             NSIM    - 33
C     ITYP  - 1: INTEGER
C             2: REAL
C
C     AUTHORS: B BENGTSSON/B EGARDT MAY 1974
C
C     SUBROTINES REQUIRED:
C          GETIN
C          EQU
C
```

```
      SUBROUTINE GETIN(ICNT,KIND,NBR,GET,IGET,GOON)
C
C     SUBROUTINE TO GET PARAMETER VALUES
C     TESTS THE TYPE OF GIVEN NUMBER AND GIVES ERROR MESSAGES
C
C     ICNT   -   1: NO COMMAND BEFORE NUMBERS
C                2: COMMAND BEFORE NUMBERS
C     KIND   -   1: INTEGER
C                2: REAL
C     NBR    -   NUMBER OF VALUES TO COME
C     GET    -   VECTOR CONTAINING REAL NUMBERS
C     IGET   -   VECTOR CONTAINING INTEGERS
C     GOON   -   RETURNED TRUE IF THE LINE IS EMPTY OR
C                COMMAND IS INCORRECT
C
C     AUTHORS: B BENGTSSON/B EGARDT MAY 1974
C
C     SUBROUTINES REQUIRED:
C         INTRAC
C         ARGIN
C

      SUBROUTINE CODE(IVNBR,COMSC,YMIN1,YMAX1)
C
C     CONVERTS A HOLLERITH STRING OF NAMES OF VARIABLES
C     INTO A VECTOR,CONTAINING A LIST OF INTEGERS,
C     REPRESENTING THE VARIABLES. THE STRING SHOULD BE
C     OF TYPE:
C     VNAME1 [VNR11..] [VNAME2 [VNR21..]..] [SC YMIN YMAX]
C     SC YMIN YMAX - USED ONLY FOR PLOT COMMANDS
C     THE FOLLOWING CODE IS USED:
C     A[LPHA] 1      -    1
C     ...
C     A[LPHA] 6      -    6
C     B[ETA] 1       -   11
C     ...
C     B[ETA] 6       -   16
C     C 1            -   21
C     ...
C     C 6            -   26      ('GAMMA' IS EQUIVALENT TO 'C')
C     LOSS           -   34
C     Y              -   35
C     U              -   36
C     YSUM           -   37
C     YREF           -   38
C     RYY 1          -   41
C     ...
C     RYY 10         -   50
C     RYU 0          -   60
C     ...
C     RYU 10         -   70
C     P 1 1          -  101
C     ...
C     P 18 18        - 1818
C
C     IVNBR          -    VECTOR OF VARIABLE NUMBERS
C                         (IVNBR(1) - NUMBER OF VARIABLES)
C
C     COMSC,YMIN1,YMAX1 ARE USED FOR PLOT COMMAND:
C     COMSC          -    RETURNED .TRUE. IF THE COMMAND CONTAINS
C                         SC YMIN1 YMAX1
C     YMIN1,YMAX1    -    VALUES FOR USERS SCALING IN THIS PLOT
C
C
C     AUTHORS: B BENGTSSON/B EGARDT MAY 1974
C
C     SUBROUTINES REQUIRED:
C         ARGIN
C
```

```
      SUBROUTINE DECODE(ISAVE,VNAME,VNR)
C
C     DECODES VARIABLE NAMES FROM CODE NUMBERS IN ISAVE
C     AND LOADS RESULT INTO VNAME (NAME) AND VNR (ELEMENT IN ARRAY).
C     BOTH VNAME AND VNR SHOULD BE WRITTEN IN A-CODE.
C
C     AUTHORS: B BENGTSSON/B EGARDT MAY 1974
C
C     SUBROUTINE REQUIRED:
C          NUMER
C
```

```
      SUBROUTINE NUMER(K,D10,D1)
C
C     CONVERTS AN INTEGER < 100 TO TWO FIGURES IN A-CODE.
C
C     K:INTEGER TO BE CONVERTED.
C
C     EXAMPLE: K=53
C     D10=1H5
C     D1 =1H3
C
C     AUTHORS: B BENGTSSON/B EGARDT MAY 1974
C
C     SUBROUTINES REQUIRED:
C          NONE
C
```

```
      SUBROUTINE EQU(VAR1,VAR2)
C
C     USED TO MAKE AN EQUIVALENCE BETWEEN ONE REAL
C     AND TWO INTEGER VARIABLES.
C
C     SUBROUTINES REQUIRED:
C          NONE
C
C     AUTHORS:   B BENGTSSON/B EGARDT    MAY 1974
C
```