

IDENTIFIERING AV PARAMETRAR I TILLSTÄNDS-
MODELLER. EN ICKE-REKURSIV METOD SOM BYGGER
PÅ VARIATIONSKALKYL.

Examensarbete

av

Anders Edwardson Ulf Ekwall

Ansvarig handledare
Karl Eklund

LUNDS TEKNISKA HÖGSKOLA
INST. FÖR REGLERTEKNIK

April 1970

INNEHÅLLSFÖRTECKNING

	Sid.
1. Inledning	1
2. Problemformulering	3
3. Lösning med variationskalkyl	5
4. Algoritmer	10
5. Beskrivning av program	18
6. Numeriska exempel	24
7. Diskussion	61
8. Referenser	70
9. Appendix	71

ABSTRACT

This report presents a method to solve the following problem: Given is a mathematical model of a physical system (which is allowed to be a multivariable one):

$$\dot{x} = f (x, u, \alpha)$$

$$y = g (x, u, \alpha)$$

where x is the state vector, u the input vector, y the output vector and α is an unknown parameter vector. f and g are vector valued functions. The model is known except for some unknown parameters α_i . Furthermore a sequence of input-output pairs obtained from measurements on the physical system is given. Determine α by using all a priori information.

We solve the problem by minimizing the arbitrary chosen loss-functional

$$L = \frac{1}{2} \sum_{\text{all outputs}} \int_0^T (y_{\text{measured}} - y_{\text{model}})^2 dt$$

using calculus of variations. A program which uses this method has been written.

The method has been tested on simulated data generated by linear systems. In this case the algorithm gives a quite accurate estimate of the unknown parameters even if the output is contaminated with noise. The number of parameters which can be estimated is however limited. In the case of nonlinear models the computing time required for execution of the program soon becomes quite large which has limited the test runs.

1. INLEDNING

De allra flesta tillämpningar av den moderna reglerteknikens metoder fordrar en matematisk modell av den studerade fysikaliska processen. En sådan modell är i många fall känd i sina huvuddrag. Kanske dock med undantag av en eller flera parametrar, som visat sig vara alltför besvärliga att bestämma ur fysikaliska samband.

Därför är det ett önskemål att kunna identifiera dessa okända parametrar med hjälp av en svit in- och utsignalpar från det fysikaliska systemet. Flera sådana metoder finns idag tillgängliga. Många metoder tillåter dock inte a priorikunskap om processen utnyttjas. Detta examensarbete presenterar en undersökning av en metod {1} {5}, som medger att hänsyn tages till sådan kunskap. Problemet kan formuleras som:

Givet är en matematisk modell av ett fysikaliskt system.

$$\dot{x} = f(x, u, \alpha) \quad (1.1)$$

$$y = g(x, u, \alpha) \quad (1.2)$$

där f och g är funktioner av relativt godtyckligt utseende. Dessa är kända sånär som på ett antal parametrar, som är okända till sitt numeriska värde. α representerar dessa parametrar i (1.1) och (1.2). Flera in- respektive utsignaler får förekomma. Givet är vidare en uppmätt svit av in- och utsignalpar från det fysikaliska systemet.

Bestäm α , och utnyttja därvid all a prioriinformation!

En noggrannare problemformulering återfinnes i kap. 2. Problemet löses med hjälp av variationskalkyl, och i korthet går det till på följande sätt:

Vi startar med att göra en intelligent gissning på de okända parametrarna varigenom vi får en grovt bestämd modell. Denna

modell matas med samma insignal som användes för det fysikaliska systemet. (I vårt fall rör det sig om en PRBS-signal; Pseudo Random Binar Seqense). Om modellens och det fysikaliska systemets utsignaler skiljer sig, bör vi kunna modifiera vår gissning på de okända parametrarna.

I kap. 3 redogör vi närmare för denna modifierings allmänna teori, som i korthet kan beskrivas sålunda: Vi bildar, relativt godtyckligt, en förlustfunktional (L), där

$$L = \frac{1}{2} \sum_{\text{alla utsign. alla sampl. intervall}} \int (y_{\text{uppmätt}} - y_{\text{modell}})^2 dt.$$

L är alltså funktion av bl.a. de aktuella värdena hos de okända parametrarna α_i . (Dessa kommer vi i fortsättningen sammanfatta till en parametervektor α). Det gäller nu att genom successiva modifieringar av α minimera L. Vi har då enligt vår mening funnit det riktigaste möjliga värdet på α .

I och med kap. 4 lämnar vi de allmänna teorierna, och ger oss i kast med att formulera algoritmer för minimeringsproblemet lösning. Dessa blir formellt något enklare för specialfallet linjära system.

Med ledning av algoritmerna kan vi sedan tillverka datamaskinprogram, med vars hjälp identifieringen av den okända parametervektorn α lätt kan utföras. En allmän beskrivning av dessa program lämnas i kap 5.

Vi har också utfört identifieringar på kända system med hjälp av simulerade deterministiska samt brusbemängda data. En redogörelse för dessa simuleringar lämnas i kap. 6.

I kap. 7 diskuteras resultaten av dessa simuleringar. Metoden har i vissa fall överträffat förväntningarna, men den har ibland även slagit slint.

Slutligen i kap. 9 (Appendix) återfinnes programlistor.

2. PROBLEMFÖRMULERING

Vi betraktar modellen:

$$\dot{x} = f(x, u, \alpha) \quad (2.1)$$

$$y = g(x, u, \alpha) \quad (2.2)$$

där tillståndet x är en n -vektor, styrsignalen u en r -vektor och utsignalen y en p -vektor.

Modellen är bestämd i sin struktur, d v s systemordning, antalet in- och utsignaler och olinjariteternas form är givna. De parametrar i systemekvationerna som vi helt saknar eller endast har begränsad information om, betecknar vi med den tidsberoende vektorn α där $\alpha^T = (\alpha_1 \ \alpha_2 \ \dots \ \alpha_l)$.

Initialtillståndet $x(t_1)$ är givet. Dessutom har vi serier av uppmätta värden på insignalen $u_m(t)$ och utsignalen $y_m(t)$ till det fysikaliska systemet. Dessa in- och utsignaler är endast givna i diskreta tidpunkter i intervallet (t_1, t_N) .

Vårt problem är att med utgångspunkt från given information bestämma vektorn α så att en förlustfunktion, som anger systemets syfte minimeras. Denna förlustfunktion har i vårt fall tämligen godtyckligt fått följande utseende:

$$L(\alpha) = \frac{1}{2} \int_{t_1}^{t_N} (y_m - g(x, u, \alpha))^T (y_m - g(x, u, \alpha)) dF(t) \quad (2.3)$$

Differensen mellan $y_m(t)$ och modellens utsignal $g(t)$ viktas med inkrementet av en funktion $F(t)$ i förlustfunktionen. $F(t)$ väljes till en stegfunktion, vars ändring sker vid varje samplingstillfälle. Denna definition av $F(t)$ medför att vi endast behöver betrakta förlustfunktionen i samplingspunkterna t_i . Fysikaliskt är innebörden att vi endast tar hänsyn till avvikelser i samplingspunkterna, vilket är rimligt eftersom processens utsignal endast är känd vid dessa tillfällen.

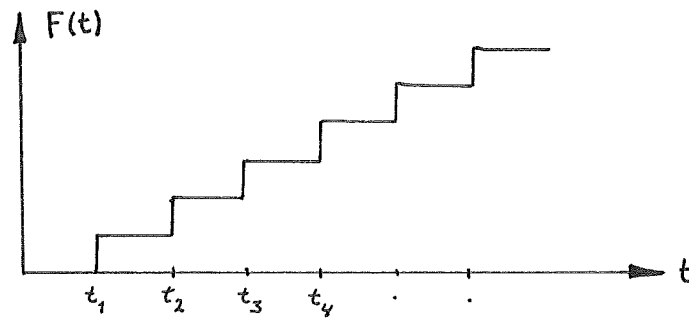


Fig. 2.1 - Stegfunktionen $F(t)$

Integralen i (2.3) kan nu uttrykas med en summa, som får ett tillskott i varje punkt t_i .

$$L(\alpha) = \frac{1}{2} \sum_{t_i=t_1}^{t_N} (y_m(t_i) - g(x(t_i), u(t_i), \alpha))^T (y_m(t_i) - g(x(t_i), u(t_i), \alpha)) \Delta F \quad (2.4)$$

Denna approximation av förlustfunktionen kommer att användas i fortsättningen.

3. LÖSNING MED VARIATIONSKALKYL

Några begrepp i variationskalkylen

Vi har valt att lösa vårt minimeringsproblem med variationskalkyl och vill därför införa några av de fundamentala begrepp, som vi använder.

Man kan i variationskalkylen ange ett kriterium för extremvärde hos en funktion, vars variabel är en funktion. Denna typ av funktion benämnes funktional. En liten förändring, variation, i variabeln ger en variation i funktionalen. Funktionalens variation kan t.ex. beräknas genom Taylorserieutveckling där termer upp till första ordningen medtages. Ett nödvändigt villkor för att funktionalen skall ha en extrempunkt för ett visst värde på variabeln är att funktionalens variation försvinner.

Ofta måste man ta hänsyn till att en funktional har bivillkor som inskränker rörelsefriheten för dess variabel. Även här kan villkor anges för extremvärde. Betrakta funktionalen

$$J(y) = \int_a^b F(x, y, y') dx$$

och låt bivillkoren vara

$$y(a) = A$$

$$y(b) = B$$

$$K(y) = \int_a^b G(x, y, y') dx = l$$

där även $K(y)$ är en funktional av y . Låt sedan $J(y)$ ha en extrempunkt för $y = y(x)$. Om då inte $y = y(x)$ är en extrempunkt till $K(y)$ existerar en konstant λ kallad Lagrangemultiplikator, sådan

att $y = y(x)$ är en extrempunkt till funktionalen $\int_a^b (F + \lambda G) dx$.

Härledning av formler för minimering av förlustfunktionen

Allmänt. Olinjära systemekvationer.

Vi skall minimera förlustfunktionalen

$$L(\alpha) = \frac{1}{2} \int_{t_1}^{t_N} (y_m - g(x, u, \alpha))^T (y_m - g(x, u, \alpha)) dF(t) \quad (3.1)$$

samtidigt som vi tar hänsyn till de begränsningar som ges av systemekvationerna (2.1). Genom att använda Lagrange-multiplikatorer inför vi $\dot{x} = f(x, u, \alpha)$ som bivillkor och får en ny förlustfunktional:

$$V(\alpha, x, \dot{x}) = \frac{1}{2} \int_{t_1}^{t_N} (y_m - g(x, u, \alpha))^T (y_m - g(x, u, \alpha)) dF(t) + \int_{t_1}^{t_N} \lambda^T(t) (\dot{x} - f(x, u, \alpha)) dt \quad (3.2)$$

Antag att denna funktional minimeras för $\alpha = \bar{\alpha}$, $x = \bar{x}$ och $\dot{x} = \bar{\dot{x}}$. Vi vill sedan bilda funktionalens variation och Taylorseriutvecklar V kring det nominella $\bar{\alpha}$, \bar{x} , $\bar{\dot{x}}$. Vi tar med termer upp till första ordningen. Detta medför att vi gör en variation δV i förlustfunktionalen. Seriutvecklingen ger:

$$\begin{aligned} \bar{V} + \delta V = & \frac{1}{2} \int_{t_1}^{t_N} (y_m - g(\bar{x}, u, \bar{\alpha}))^T (y_m - g(\bar{x}, u, \bar{\alpha})) dF(t) - \\ & \int_{t_1}^{t_N} (y_m - g(\bar{x}, u, \bar{\alpha}))^T g_{\alpha}(\bar{x}, u, \bar{\alpha}) \delta \alpha dF(t) - \\ & \int_{t_1}^{t_N} (y_m - g(\bar{x}, u, \bar{\alpha}))^T g_x(\bar{x}, u, \bar{\alpha}) \delta x dF(t) + \dots + \\ & \int_{t_1}^{t_N} \lambda^T(t) (\bar{\dot{x}} - f(\bar{x}, u, \bar{\alpha})) dt + \int_{t_1}^{t_N} \lambda^T(t) \delta \dot{x} dt - \int_{t_1}^{t_N} \lambda^T(t) f_x(\bar{x}, u, \bar{\alpha}) \delta x dt - \\ & \int_{t_1}^{t_N} \lambda^T(t) f_{\alpha}(\bar{x}, u, \bar{\alpha}) \delta \alpha dt + \dots \end{aligned} \quad (3.3)$$

$$g_{\alpha} = \frac{\partial g}{\partial \alpha}$$

$$g_x = \frac{\partial g}{\partial x}$$

där

$$f_{\alpha} = \frac{\partial f}{\partial \alpha}$$

$$f_x = \frac{\partial f}{\partial x}$$

Variationen δV erhålles om vi subtraherar V på båda sidor om likhetstecknet. Termen med variation i $\delta \dot{x}$ partialintegreras:

$$\int_{t_1}^{t_N} \lambda^T(t) \delta \dot{x} dt = \left[\lambda^T(t) \delta x \right]_{t_1}^{t_N} - \int_{t_1}^{t_N} \dot{\lambda}^T(t) \delta x dt$$

Termer med variation i \dot{x} försvinner alltså. Resultatet av integrationen insättes i δV samtidigt som vi samlar integralerna med variation i α resp. x var för sig.

$$\begin{aligned} \delta V = & \lambda^T(t_N) \delta x(t_N) - \lambda^T(t_1) \delta x(t_1) - \\ & \int_{t_1}^{t_N} (\dot{\lambda}^T(t) + (y_m(t) - g(\bar{x}, u, \bar{\alpha}))^T g_x(\bar{x}, u, \bar{\alpha})) \frac{dF}{dt} + \\ & \lambda^T(t) f_x(\bar{x}, u, \bar{\alpha}) \delta x dt - \\ & \int_{t_1}^{t_N} ((y_m(t) - g(\bar{x}, u, \bar{\alpha}))^T g_\alpha(\bar{x}, u, \bar{\alpha})) \frac{dF}{dt} + \\ & \lambda^T(t) f_\alpha(\bar{x}, u, \bar{\alpha}) \delta \alpha dt \end{aligned} \quad (3.4)$$

I variationskalkylen anges som vi ovan sett ett kriterium för att en funktional skall ha extremvärde som: Variationen av funktionalen skall försvinna i extrempunkten. Vi är alltså intresserade av att $\delta V = 0$.

1. Begynnelsevärdet i tillstånden är givet vilket medför att variationen i detsamma $\delta x(t_1) = 0$.

Ekvation (3.4) skall gälla för godtyckliga variationer $\delta \alpha$ och δx . För att då $\delta V = 0$ skall gälla erhåller vi:

2. Om $\delta \alpha = 0$ och $\delta x(t) = 0$ då $t \neq t_N$ skall $\lambda(t_N) = 0$ och vi har fått ett randvillkor för λ .

3. Om $\delta \alpha = 0$ och $\delta x(t) \neq 0$ skall $\dot{\lambda}^T(t) + f_x^T(\bar{x}, u, \bar{\alpha}) \cdot \lambda(t) + g_x^T(\bar{x}, u, \bar{\alpha})(y_m(t) - g(\bar{x}, u, \bar{\alpha})) \frac{dF}{dt} = 0$ (3.5)

Eftersom $F(t)$ är en stegfunktion med ändring i samplingsögonblicken t_i är dF/dt formellt en Dirac-funktion och karakteriseras alltså av:

$$\dot{\lambda}(t) = -f_x^T(\bar{x}, u, \bar{\alpha})\lambda(t) - g_x^T(\bar{x}, u, \bar{\alpha})(y_m - g(\bar{x}, u, \bar{\alpha}))\delta(t - t_i) \quad (3.6)$$

$$\lambda(t_N) = 0 \quad (3.7)$$

Vi observerar att randvillkoret för λ -ekvationen är givet i sluttidpunkten t_N . Differensen mellan mätvärdena y_m och modellens utsignal $g(x, u, \alpha)$ driver λ -ekvationen endast i samplingsögonblicken, vilket är naturligt eftersom mätvärdena bara är tillgängliga vid samplingstillfällena. Då λ beräknats enligt ovanstående gäller

$$\delta V = - \int_{t_1}^{t_N} ((y_m - g(\bar{x}, u, \bar{\alpha}))g_{\alpha}(\bar{x}, u, \bar{\alpha}) \frac{dF}{dt} + \lambda^T(t)f_{\alpha}(\bar{x}, u, \bar{\alpha}))\delta\alpha dt \quad (3.8)$$

Detta uttryck kan om $\delta\alpha \rightarrow 0$ omformas till gradienten av V med avseende på α .

$$\frac{dV}{d\alpha} = - \int_{t_1}^{t_N} ((y_m - g(\bar{x}, u, \bar{\alpha}))g_{\alpha}(\bar{x}, u, \bar{\alpha}) \frac{dF}{dt} + \lambda^T(t)f_{\alpha}(\bar{x}, u, \bar{\alpha}))dt \quad (3.9)$$

där gradienten av en funktion av flera variabler definieras som en radvektor.

Linjära systemekvationer

Då vi har linjära systemekvationer medför detta att vi betraktar ett specialfall av under A härledda formler. Den linjära modellen har följande principiella utseende.

$$\dot{x} = A(\alpha)x + B(\alpha)u \quad (3.10)$$

$$\bar{y} = C(\alpha)x + D(\alpha)u \quad (3.11)$$

Begynnelsevärdet av tillståndsvektorn $x(t_1) = C$.

Vår uppgift är nu att minimera

$$L(\alpha) = \frac{1}{2} \int_{t_1}^{t_N} (y_m(t) - C(\alpha)x - D(\alpha)u)^T (y_m(t) - C(\alpha)x - D(\alpha)u) dF(t) \quad (3.12)$$

Sätter vi sedan in de nya systemekvationerna i de formler som härletts i det allmänna fallet (3.6-9) får vi:

$$\dot{\lambda}(t) = -A^T \lambda(t) - C^T (y_m(t) - Cx - Du) \delta(t - t_i) \quad (3.13)$$

$$\lambda(t_N) = 0 \quad (3.14)$$

$$\delta V = - \int_{t_1}^{t_N} ((y_m - C\bar{x} - Du)^T \frac{\partial}{\partial \alpha} (C(\bar{\alpha})\bar{x} + D(\bar{\alpha})u) \frac{dF}{dt} + \lambda^T(t) \frac{\partial}{\partial \alpha} (A(\bar{\alpha})\bar{x} + B(\bar{\alpha})u)) d\alpha dt \quad (3.15)$$

och

$$\frac{dV}{d\alpha} = - \int_{t_1}^{t_N} (y_m - C\bar{x} - Du)^T \frac{\partial}{\partial \alpha} (C(\bar{\alpha})\bar{x} + D(\bar{\alpha})u) \frac{dF}{dt} + \lambda^T(t) \frac{\partial}{\partial \alpha} (A(\bar{\alpha})\bar{x} + B(\bar{\alpha})u) dt \quad (3.16)$$

4. ALGORITMER

Vi har följande givet:

- Initialgissningen $\alpha(0)$
- Initialvärdet $x(0)$
- $U(t)$ och $y_m(t)$, givna i tidpunkterna t_i , $i = 1, 2, \dots, N$

Problemet var att bestämma en ny α -vektor. Detta löses med hjälp av en minimeringsalgoritm av Fletcher-Powell typ, och denna kräver tillgång till värdet av förlustfunktionen och förlustfunktionens gradient med avseende på de obekanta parametrarna α_i .

Olinjärt system

Algoritmens principiella utseende blir:

1. Sätt $\alpha = \alpha(0)$
2. Integrera $\begin{cases} \dot{x} = f(x, u) \\ \dot{y} = g(x, u) \end{cases}$ där $x(t_1)$ är given och $x(0) = x(0)$
3. Spara $x(kp)$, $k = 1, \dots, Np$
4. Integrera $\dot{\lambda} = -f_x^T \lambda - g_x^T (y_m - g(x, u)) \delta(t - t_i)$
5. Evaluera V samt $dV/d\alpha$
6. Bestäm nytt α med FLEPOMIN
7. Gå till 2 om ej α konvergerat.

Kommentarer:

Systemekvationen i punkt 2 är olinjär och kräver en integrationsrutin av typ Runge-Kutta. Den adjungerade ekvationen i 4 är emellertid linjär men tidsvariabel.

Vid integration av systemekvationerna krävs ett kort integrationssteg h . Om $x(t)$ endast varierar långsamt, kan med god approximation ett väsentligt längre integrationssteg väljas i λ -ekvationen. Detta medför besparing av minnesutrymme, eftersom $x(t_k)$ måste sparas i varje t_k där $\lambda(t)$ beräknas.

Integration av systemekvationerna

$u(t)$ är konstant över samplingsintervallet T , d v s $u(t)$ definieras enl. fig. 4.1. Samplingsintervallet T är konstant över hela mätserien.

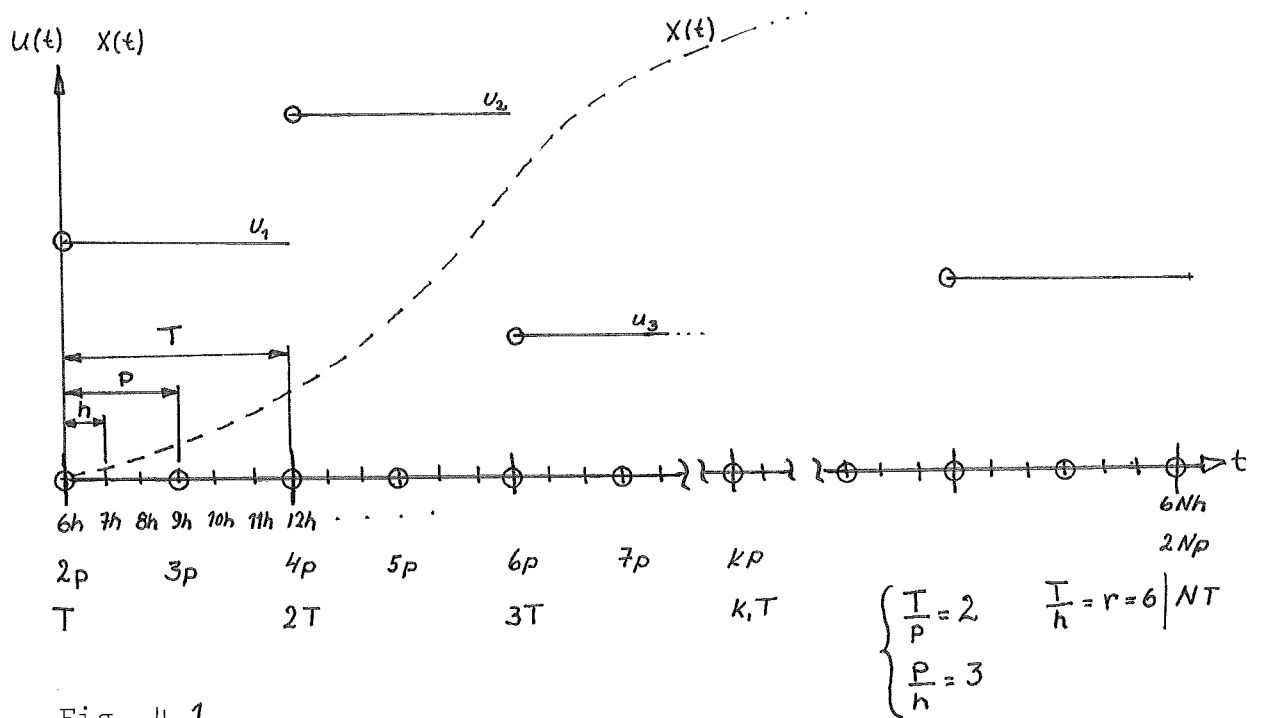


Fig. 4.1.

Definition av intervallindelningen. $x(kp)$ sparas för λ -beräkningen i de tidpunkter som är inringade på tidsaxeln.

De olinjära systemekvationerna $\dot{x} = f(x, u, \omega)$ integreras med $y = g(x, u, \omega)$

hjälp av en rutin typ Runge-Kutta. Integrationssteget h väljes så, att samplingsintervallet T är en heltalsmultipel av h , d v s $h = \frac{T}{r}$, där r är ett heltal. Integrationen utföres av subrutinen RK1ST, och $x(kp)$, $k = 1 \dots NP$ lagras för integreringen av λ -ekvationen.

Evaluering av förlustfunktionen V

Förlustfunktionen är enl. föregående:

$$L(\alpha) = \frac{1}{2} \sum_{i=1}^N (y_m(t_i) - g(x(t_i), u(t_i), \alpha))^T (y_m(t_i) - g(x(t_i), u(t_i), \alpha)) \quad (4.1)$$

Med $t_i = kT$ får vi

$$L(\alpha) = \frac{1}{2} \sum_{k=1}^N (y_m(kT) - g(x(kT), u(kT), \alpha))^T (y_m(kT) - g(x(kT), u(kT), \alpha)) \quad (4.2)$$

Beräkningen av förlustfunktionen kräver således att $x(t)$ är känd endast i samplingspunkterna.

Evaluering av förlustfunktionens gradient $dV/d\alpha$

Om vi, som förut, för in bivillkoret via lagrangemultiplikationer så gäller:

$$\frac{dV}{d\alpha} = - \int_{t_1}^{t_N} ((y_m - g(x, u, \alpha))^T g_{\alpha}(x, u, \alpha) \frac{dF}{dt} + \lambda^T f_{\alpha}(x, u, \alpha)) dt \quad (4.3)$$

eller, om vi som ovan sätter $t_i = kT$ och beaktar att $F(t)$ är en stegfunktion:

$$\begin{aligned} \frac{dV}{d\alpha} = & - \sum_{k=1}^N (y_m(kT) - g(x(kT), u(kT), \alpha))^T g_{\alpha}(x(kT), u(kT), \alpha) - \\ & - \int_{t_1}^{t_N} \lambda^T(t) f_{\alpha}(x(t), u(t), \alpha) dt \end{aligned} \quad (4.4)$$

Antag nu, att $\lambda(t)$ är känd i tidpunkterna k_p , där $k = n, n+1, \dots, nN$, och p har dimensionen tid och är en heltalsmultipel av h , och sådan att T även är en heltalsmultipel av p , ($T = np$). Approximeras andra termen i högerledet av ekvation (4.4) med en summa så fås:

$$\begin{aligned} \frac{dV}{d\alpha} = & - \sum_{k=1}^N (y_m(kT) - g(x(kT), u(kT), \alpha))^T g_{\alpha}(x(kT), u(kT), \alpha) - \\ & - \sum_{k=1}^N \lambda^T(k_p) f_{\alpha}(x(k_p), u(k_p), \alpha) p \end{aligned} \quad (4.5)$$

Integrationen av λ -ekvationen

För att vi skall kunna beräkna ovanstående gradient fordras tillgång till λ -funktionens värden i de diskreta tidpunkterna

$t = kp, k = n, \dots, nN.$

Vi har enl. kap. 3:

$$\begin{aligned} \dot{\lambda}(t) &= -f_x^T(x(t), u(t), \alpha)\lambda(t) - g_x^T(x(t), u(t), \alpha)(y_m(t) - \\ &- g(x(t), u(t), \alpha))\delta(t-t_i) \\ \lambda(t_N) &= 0 \end{aligned} \quad (4.6)$$

Som vi ser, så ges randvillkoret i slutet av mätserien. En rekursiv beräkning av $\lambda(t)$ där $t = kp$ måste alltså startas i sista tidpunkten t_N (där $\lambda=0$) och utföras med avtagande tid, d v s $k = nN, nN-1, \dots, n$. $\lambda(kp)$ beräknas alltså "baklänges".

Ekvation (4.6) är en linjär tidsvariant differentialekvation.

Inför beteckningarna

$$F(t) = f_x^T(x(t), u(t), \alpha)$$

$$G(t) = g_x^T(x(t), u(t), \alpha),$$

och ekvationen (4.6) kan skrivas:

$$\begin{cases} \dot{\lambda}(t) = -F(t)\lambda(t) - G(t)(y_m(t) - g(x(t), u(t), \alpha))\delta(t-t_i) \\ \lambda(t_N) = 0 \end{cases} \quad (4.7)$$

Antag, att $F(t)$ och $G(t)$ är konstanta över varje intervall $((k+1)p, kp)$. En rekursiv formel för λ är då:

$$\begin{aligned} \lambda(kp) &= e^{-F((k+1)p)(kp-(k+1)p)} \lambda((k+1)p) - \int_{(k+1)p}^{kp} e^{-F((k+1)p)(kp-s)} \\ &\cdot G(s)(y_m(s) - g(x(s), u(s), \alpha))\delta(s-t_k) ds \end{aligned} \quad (4.8)$$

Lägg märke till diracfunktionen i högerledet av ekvation (4.8)! Den innebär att ekvationen (4.8) endast drives i samplingspunkterna $k_1 T, k_1=1, \dots, N$. $\lambda(t)$ gör alltså ett språng vid tidpunkterna $t = k_1 T$ beroende på den nya differensen mellan modellens utsignal och de verkliga mätvärdena som tillkommer.

Detta är en följd av definitionen av y_m , där mätvärdet i tidpunkten t_i antas gälla hela samplingsintervallet. Se fig. 4.2!

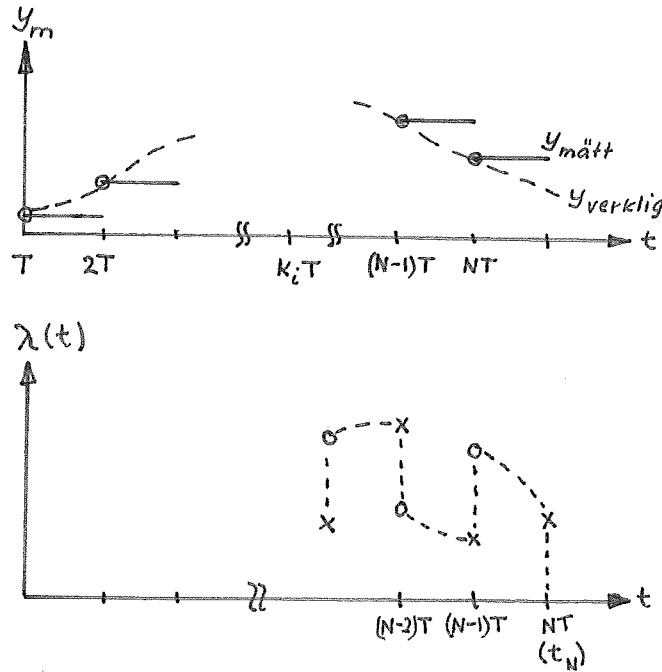


Fig. 4.2 - Definitionen på y_m . Nedre kurvan visar hur $\lambda(t)$ gör ett språng i samplingsögonblicken. De med o markerade värdena är värdet av $\lambda(t)$ vid tidpunkterna t_i , $i=N, \dots, 1$. Värdet efter den språngvisa ändringen är markerat med x.

Integralen i ekvationen (4.8) löses varvid vi erhåller:

$$\lambda(kp) = \begin{cases} e^{F((k+1)p)P} \cdot \lambda((k+1)p) & \text{om } (k+1)p \neq \forall t_i, i=1, \dots, N \\ e^{F((k+1)p)P} \cdot \lambda((k+1)p) + e^{F((k+1)p)P} \cdot G((k+1)p) \cdot (y_m((k+1)p) - g(x((k+1)p), u((k+1)p, \alpha))) & \text{om } (k+1)p = ngt t_i, i = 1, \dots, N. \end{cases} \quad (4.9)$$

Linjärt system

Algoritmens principiella utseende blir:

1. Sätt $\alpha = \alpha(0)$
2. Integrera $\begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases}$ där $x(t_i)$ är given
3. Spara $x(t_i)$, $i = 1, 2, \dots, N$
4. Integrera $\begin{cases} \dot{\lambda} = -A^T \lambda - C^T (y_m - Cx - Du) \delta(t - t_i) \\ \lambda(t_N) = 0 \end{cases}$
5. Evaluera V samt $\partial V / \partial \alpha$
6. Bestäm nytt α med FLEPOMIN
7. Gå till 2 om α ej konvergerat.

Integration av systemekvationerna

Antag, som ovan, att $u(t)$ är konstant över samplingsintervall
 vallet T , och att samplingsintervallet T är konstant. En
 rekursionsformel för $x(t)$ är då:

$$\begin{cases} x(t_{i+1}) = e^{AT} x(t_i) + \int_0^T e^{As} B ds \cdot u(t_i), & i=1, \dots, N \\ x(t_1) = C \end{cases} \quad (4.10)$$

För att kunna integrera systemekvationerna behöver vi beräkna

$$\Phi = e^{AT}$$

$$\Gamma = \int_0^T e^{As} B ds$$

Evaluering av förlustfunktionen V

Om $\dot{x} = Ax + Bu$, så är $V(\alpha, x, \dot{x}) = L(\alpha)$

$$L(\alpha) = \frac{1}{2} \int_{t_1}^{t_N} (y_m(t) - C(\alpha)x(t) - D(\alpha)u(t))^T (y_m(t) - C(\alpha)x(t) - D(\alpha)u(t)) dF(t) \quad (4.11)$$

Om $F(t)$ är stegfunktion med inkrementet \uparrow vid tidpunkterna t_i ,
 så gäller

$$L(\alpha) = \frac{1}{2} \sum_{i=1}^N (y_m(t_i) - Cx(t_i) - Du(t_i))^T (y_m(t_i) - Cx(t_i) - Du(t_i)) \quad (4.12)$$

Evaluering av gradienten av V

Gradienten är:

$$\frac{dV}{d\alpha} = - \int_{t_1}^{t_N} ((y_m - Cx - Du)^T \cdot \frac{\partial}{\partial \alpha} (C(\alpha)x + D(\alpha)u) \frac{dF}{dt} + \lambda^T(t) \cdot \frac{\partial}{\partial \alpha} (A(\alpha)x + B(\alpha)u) dt \quad (4.13)$$

Enl. ovan gäller då:

$$\begin{aligned} \frac{dV}{d\alpha} = & - \sum_{i=1}^N (y_m(t_i) - Cx(t_i) - Du(t_i))^T \cdot \frac{\partial}{\partial \alpha} (C(\alpha)x(t_i) + D(\alpha)u(t_i)) - \\ & - \int_{t_1}^{t_N} \lambda^T(t) \cdot \frac{\partial}{\partial \alpha} (A(\alpha)x(t) + B(\alpha)u(t)) dt \end{aligned} \quad (4.14)$$

Approximeras den 2:a termen i ekvationen (4.14) med en summa gäller:

$$\begin{aligned} \frac{dV}{d\alpha} = & - \sum_{i=1}^N (y_m(t_i) - Cx(t_i) - Du(t_i))^T \cdot \frac{\partial}{\partial \alpha} (C(\alpha)x(t_i) + D(\alpha)u(t_i)) - \\ & - \sum_{i=1}^N (\lambda^T(t_i) \cdot \frac{\partial}{\partial \alpha} (A(\alpha)x(t_i) + B(\alpha)u(t_i)))^T \end{aligned} \quad (4.15)$$

Med denna approximation av $\frac{dV}{d\alpha}$ behöves värdet av $\lambda(t)$ endast beräknas i samplingsögonblicken, vilket medför vissa beräkningstekniska fördelar.

Integration av λ -ekvationen

$\lambda(t)$ behöver enl. ovan endast beräknas i tidpunkterna t_i , $i = N, \dots, 1$. Vi har då:

$$\begin{cases} \dot{\lambda}(t) = -A^T \lambda(t) - C^T (y_m(t) - Cx(t) - Du(t)) \cdot \delta(t - t_i) \\ \lambda(t_N) = 0 \end{cases} \quad (4.16)$$

Observera att 2:a termen i λ -ekvationen endast ger ett bidrag vid tidpunkterna t_i , $i = N, \dots, 1$. En rekursionsformel för λ är:

$$\lambda(t_i) = e^{-A^T(t_i - t_{i+1})} \cdot \lambda(t_{i+1}) - \int_{t_{i+1}}^{t_i} e^{-A^T(t_i - s)} C^T (y_m(s) - Cx(s) - Du(s)) \cdot \delta(s - t_k) dt,$$

där $t_i - t_{i+1} = -T$.

Vi måste nu definiera tidpunkten t_k . Samma definition av $y_m(t)$ och för $u(t)$ som vid fallet olinjärt system ovan leder till $t_k = t_{i+1}$.

Vi får nu:

$$\lambda(t_i) = e^{A^T \cdot T} \cdot \lambda(t_{i+1}) + e^{A^T \cdot T} C^T (y_m(t_{i+1}) - Cx(t_{i+1}) - Du(t_{i+1}))$$

(4.18)

5. PROGRAMBESKRIVNING

Linjära system

Programmets principiella utseende framgår av fig. 5.1, där pilarna betecknar anrop.

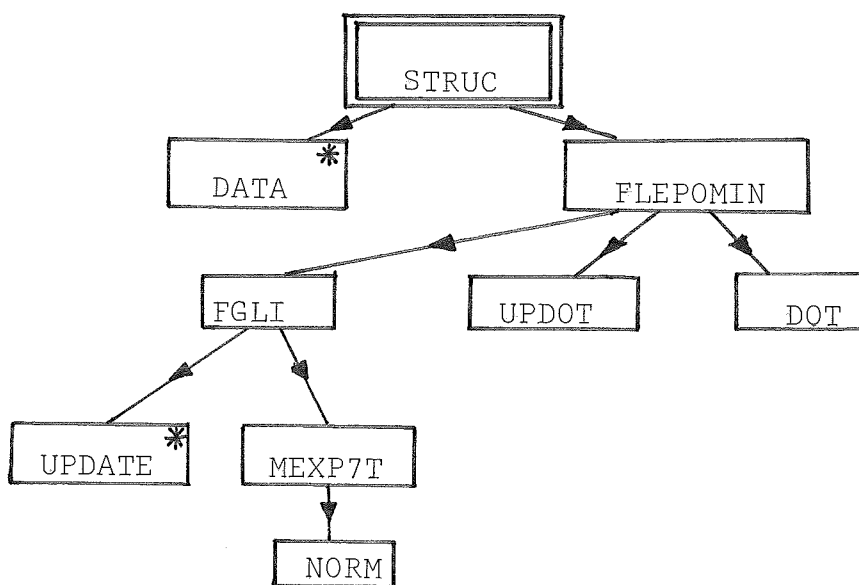


Fig. 5.2 - De med * märkta rutinerna måste ändras och kompileras vid varje körning.

Program STRUC är huvudprogrammet. Dess uppgift är dels att läsa in indata och utföra utskrifter, dels att anropa FLEPOMIN.

Indata är:

N	Systemordning	I 2
NU	Antal insignaler	I 2
NY	Antal utsignaler	I 2
MENU	Antal mätvärden	I 3
NUPA	Antal obek. parametrar	I 2
TSAMP	Samplingsintervallets längd	F 10.5
A,B,C,D	Systemmatriser	E 20.10
IAD	Heltalsmatriser som definierar partiella derivator av A,B,C,D med avseende på α_i	I 2
IBD		
ICD		
IDD		

XI	Initialvärdet på tillståndsvektorn	E 20.10
ALFA(I)	Initialvärdet på vektorn	E 20.10
EPS	Ett litet tal (ex. 10^{-8}) som används i FLEPOMIN	E 20.10
LIMIT	Maximalt antal iterationer i FLEPOMIN	I 3
LOADH	LOADH=0 medför att vektorn H läses in LOADH=1 " " " " ej " " I 1	
H	En vektor som innehåller en uppskattning på inversen av andraderivatormatrisen P för förlustfunktionen V med avseende på α . Denna matris är symmetrisk. H innehåller övre triangelmatri- sen av P lagrad efter rader.	

T ex.

$$P = \begin{pmatrix} 1 & 0.1 & 0.3 \\ 0.1 & 3. & 0.4 \\ 0.3 & 0.4 & 5 \end{pmatrix}$$

$$H = \begin{pmatrix} 1. \\ 0.1 \\ 0.3 \\ 3. \\ 0.4 \\ 5. \end{pmatrix}$$

Om vektorn H ej läses in, sättes P till enhetsmatrisen i FLEPOMIN. E 20.10

Matriserna IAD, IBD, ICD, IDD beräknas sålunda:

Exempel

$$\dot{x} = \begin{pmatrix} \alpha_1 & 2 \\ 0 & -3 \end{pmatrix} x + \begin{pmatrix} 1 & 2 \\ \alpha_2 & 4 \end{pmatrix} u$$

$$y = \begin{pmatrix} 1 & \alpha_3 \end{pmatrix} x + \begin{pmatrix} \alpha_4 & 1 \end{pmatrix} u$$

$$\frac{\partial}{\partial \alpha} (A(\alpha)x) = \frac{\partial}{\partial \alpha} \begin{pmatrix} \alpha_1 x_1 + 2x_2 \\ -3x_2 \end{pmatrix} = \begin{pmatrix} x_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\Rightarrow IAD = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

IAD är alltså en N x NUPA-matris

$$\frac{\partial}{\partial \alpha} (B(\alpha)u) = \frac{\partial}{\partial \alpha} \begin{pmatrix} u_1 + 2u_2 \\ \alpha u_1 + 4u_2 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & u_1 & 0 & 0 \end{pmatrix}$$

$$\Rightarrow \text{IBD} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \text{ d v s en } N \times \text{NUPA-matris}$$

$$\frac{\partial}{\partial \alpha} (C(\alpha)x) = \frac{\partial}{\partial \alpha} (x_1 + \alpha_3 x_2) = (0 \quad 0 \quad x_2 \quad 0)$$

$$\Rightarrow \text{ICD} = (0 \quad 0 \quad 2 \quad 0) \text{ d v s en NY + NUPA-matris}$$

$$\frac{\partial}{\partial \alpha} (D(\alpha)u) = \frac{\partial}{\partial \alpha} (\alpha_4 u_1 + u_2) = (0 \quad 0 \quad 0 \quad u_1)$$

$$\Rightarrow \text{IDD} = (0 \quad 0 \quad 0 \quad 1) \text{ d v s en NY x NUPA-matris}$$

Subrutin DATA läser in våra in- och utsignalpar som är lagrade på lämpligt sätt, antingen på magnetband eller på kort. Denna subrutin måste alltså ändras om vi byter data för in- och utsignalpar.

Subrutin FLEPOMIN sköter själva minimeringen av förlustfunktionen över de obekanta parametrarna α_i . Det nya (modifierade)värdet på α beräknas enl. formeln

$$\text{ALFA}^{n+1} = \text{ALFA}^n + \beta \cdot H \cdot G$$

där β = skalär som bestämmer steglängden

G = gradienten

H = invers 2:a derivatormatris

FLEPOMIN fordrar tillgång till förlustfunktionen V och gradienten $\frac{dV}{d\alpha}$ för de aktuella parametervärdena α_i och anropar därför FGLI.

Subrutin FGLI. Denna subrutin utnyttjar de i kap.4 härledda formlerna. FGLI integrerar således först system- och adjungerande ekvationer med hjälp av subrutinen MEXP7T. Därefter beräknas förlustfunktionen V:s värde, och efter integreringen av λ -funktionen kan $dV/d\alpha$ beräknas enl. algoritmerna ovan.

Subrutin UPDATE uppdaterar systemmatriserna A B C D.

Samma exempel som på sid 19 ger

A(1,1) = ALFA (1)

B(2,1) = ALFA (2)

C(1,2) = ALFA (3)

D(1,1) = ALFA (4)

Denna subrutin måste alltså ändras om de obekanta parametrarna ändras.

Olinjära system

Programmets principiella utseende är enl fig 5.2

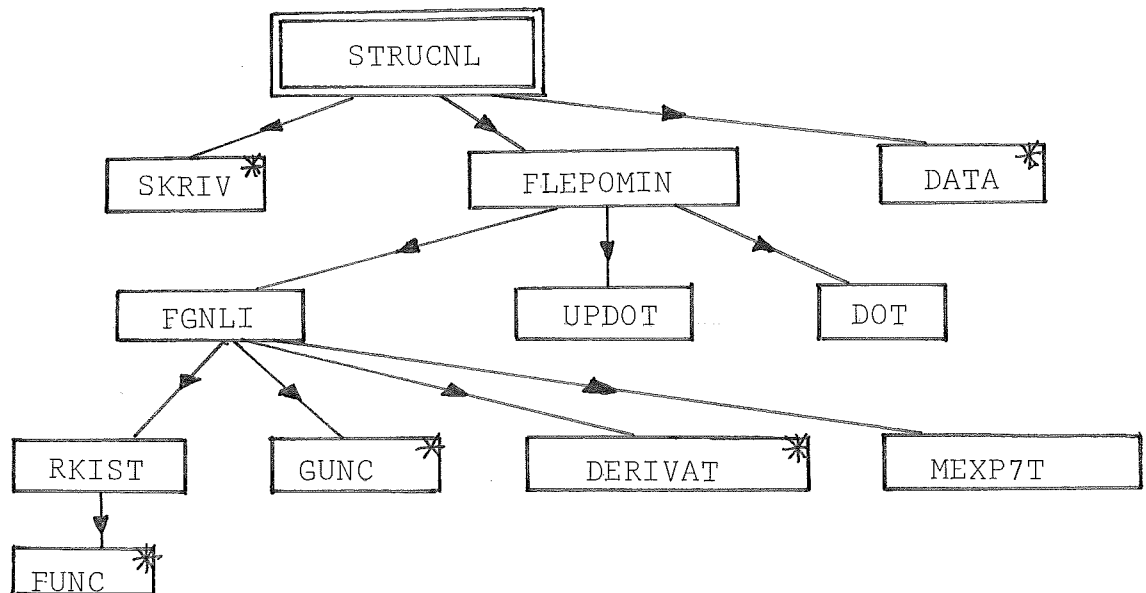


Fig 5.2

De med * märkta rutinerna måste ändras och kompileras vid varje körning.

Program STRUCNL är huvudprogrammet. Det sköter i princip om inläsning och utskrift av data samt anropar FLEPOMIN.

Indata är

NX	Systemordning	I 2
NU	Antal insignaler	I 2
NY	Antal utsignaler	I 2
NUPA	Antal okända parametrar	I 2

MENU	Antal mätvärden	I 3
TSAMP	Samplingsintervallets längd	F 10.5
XSAMP	Steglängden i Runge-Kutta	F 10.5
PSAMP	Intervallet mellan varje λ -beräkning	F 10.5
ALFA (I)	Initialvärdet på vektorn α	E 20.10
EPS	Se ovan!	
LIMIT	-"-	
LOADH	-"-	
H	-"-	

Subrutin DATA. Se ovan!

Subrutin FLEPOMIN. Se ovan!

Subrutin SKRIV sköter om utskrift av de aktuella f- samt g-funktionerna. Ex.:

$$\begin{aligned}F(1) &= ALFA(1) * X(2) ** 2 + U(2) \\F(2) &= X(1) + 3 * X(2) + 4 * U(1) ** 2 \\G(1) &= 3 * X(1) + EXP(X(2)) + U(1) * ALFA(3)\end{aligned}$$

Detta sker genom användning av FORMAT H-kod.

Subrutin RK1ST integrerar systemekvationerna med hjälp av Runge-Kutta.

Subrutin GUNC förser rutinen FGCLI med värdena på g-funktionerna för värdena på x, u och α . Exempelvis

$$\begin{aligned}GF(1) &= 3 * X(1) + EXP(X(2)) + U(1) * ALFA(3) \\GF(2) &= \dots \quad \text{o s v.}\end{aligned}$$

Subrutin DERIVAT förser rutinen FGCLI med värdena på df/dx , $df/d\alpha$, dg/dx , $dg/d\alpha$, för speciella värden på x, u och α .

Samma ex. som ovan:

$$\begin{aligned}\left(\frac{\partial f_1}{\partial x_1}\right)_{FX(1,1)} &= 2 * ALFA(1) * X(1) \\ \left(\frac{\partial f_1}{\partial x_2}\right)_{FX(1,2)} &= 0\end{aligned}$$

$$\left(\frac{\partial f_2}{\partial x_1} \right) = FX(2,1) = 1$$

$$\left(\frac{\partial f_2}{\partial x_2} \right) = FX(2,2) = 3$$

$$\left(\frac{\partial f_1}{\partial \alpha_1} \right) = FA(1,1) = X(2) ** 2$$

$$\left(\frac{\partial g_1}{\partial x_1} \right) = GX(1,1) = 3$$

$$\left(\frac{\partial g_1}{\partial \alpha_1} \right) = GA(1,1) = 0$$

Subrutinen FUNC förser rutinen RK1ST med värdena på f-funktionerna för speciella värden på x, u och α . Exempelvis

$$FU(1) = ALFA(1) * X(2) ** 2 + U(2)$$

$$FU(2) = \dots$$

⋮

Subrutinen FGCLI har samma funktion som rutinen FGLI ovan.

6. NUMERISKA EXEMPEL

Vår identifieringsmetod har vi i första hand testat för det linjära specialfallet. Under arbetets gång har bl.a. nedanstående frågeställningar funnits i åtanke. Många av dessa problem kan behandlas och besvaras med teoretisk analys. Vår uppgift har emellertid varit att via simulering undersöka vilka egenskaper de framtagna algoritmerna har i dessa avseenden.

- Hur många obekanta parametrar kan maximalt identifieras?
- Givet B och C samt insignal- och utsignalrelationen definierad av mätvärden. Hur många parametrar kan då bestämmas i A?
- Är konvergensen känslig för initialgissningar på vektorn α eller felet i initialvärdet av x?
- Hur inverkar brus på skattningarna av parametrarna α ?
- Är det någon skillnad att släppa loss parametrar perifert från diagonalen i A mot diagonalelement?
- Vad betyder samplingstiden för identifieringsnoggrannheten?
- Hur inverkar antalet mätdata på identifieringsnoggrannheten?

Linjära system

De linjära system vilka använts som fysikaliska modeller har utseende som följer:

System 1:

$$\begin{cases} \dot{x} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -3 & 0 \\ 0 & 0 & -1 \end{pmatrix} \cdot x + \begin{pmatrix} 1 & 0 \\ 0 & 2 \\ 0 & 1 \end{pmatrix} \cdot u \\ y = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} x + \lambda e \end{cases}$$

System 2:

$$\begin{cases} \dot{x} = \begin{pmatrix} -6 & 1 & 0 \\ -11 & 0 & 1 \\ -6 & 0 & 0 \end{pmatrix} \cdot x + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \cdot u \\ y = (1 \ 0 \ 0)x + \lambda e \end{cases}$$

System 1 är alltså ett tredje ordningens tidsinvariant system med två insignaler och två utsignaler. Egenvärdena är -1, -3 och -1. Notera att systemet kan skrivas på diagonalform trots att två egenvärden är lika.

Systemets blockschema:

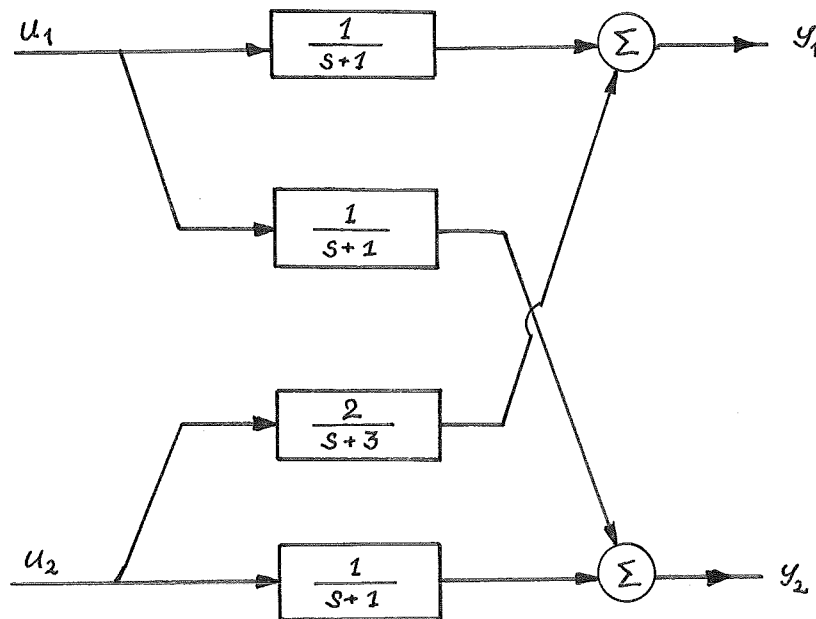


Fig.6.1

Vardera utsignalen påverkar således systemets två utsignaler och vi har en koppling i systemet.

Om detta till synes enkla systems egenheter hänvisas till {3}.

System 2 är även ett tidsinvariant system av tredje ordningen men har en insignal och en utsignal. Egenvärdena är -1, -2 och -3. Ovan är systemet skrivet på observerbar kanonisk form. Identifiering har även gjorts med samma mätdata på systemet transformerat till diagonalform, vars systemekvation är:

$$\begin{cases} \dot{\mathbf{x}} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & -3 \end{pmatrix} \mathbf{x} + \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \mathbf{u} \\ y = \begin{pmatrix} 1/2 & -1 & 1/2 \end{pmatrix} \mathbf{x} + \lambda e \end{cases}$$

För att erhålla mätdata till systemen lägger vi PRBS-signaler som insignal. Systemekvationerna integreras och ger oss tillståndsvektorn. Utsignalen påverkas förutom av tillstånden av brus λe , där e är en vektor. e genereras med en slumpvals-algoritm och har fördelningen $N(0,1)$. λ antar värdena 0, 0.1, 1.0 och anger spridningen på det brus som adderas till utsignalvektorn.

In- och utsignaler har lagrats på band i 500 samplingspunkter för system 1 och i 400 för system 2. Samplingsintervallets längd är i samtliga fall $T = 0.1$ sek. En uppsättning data finns för varje brusnivå λ , följaktligen 6 serier. Dessa finns plottade i diagram (6:1-6).

Olinjära system

Algoritmen och programmen för olinjära system har vi huvudsakligen testat med hjälp av data från de ovan beskrivna system 1 och 2. Vi har då kunnat göra direkta jämförelser mellan resultaten från testkörningarna med linjär metod resp. olinjär metod. Logiska programmeringsfel har på så vis kunnat spåras.

Avslutningsvis har vi emellertid gjort ett mindre antal identifieringar med hjälp av data från följande system:

$$\begin{cases} \dot{x} = -x + 5x^2 + u & (u = \text{PRBS-signal enl. ovan.}) \\ y = x \end{cases}$$

Samplingsintervallet $T = 0.01$ sek. Brusfritt.

EX. 1. $\lambda = 0.0$

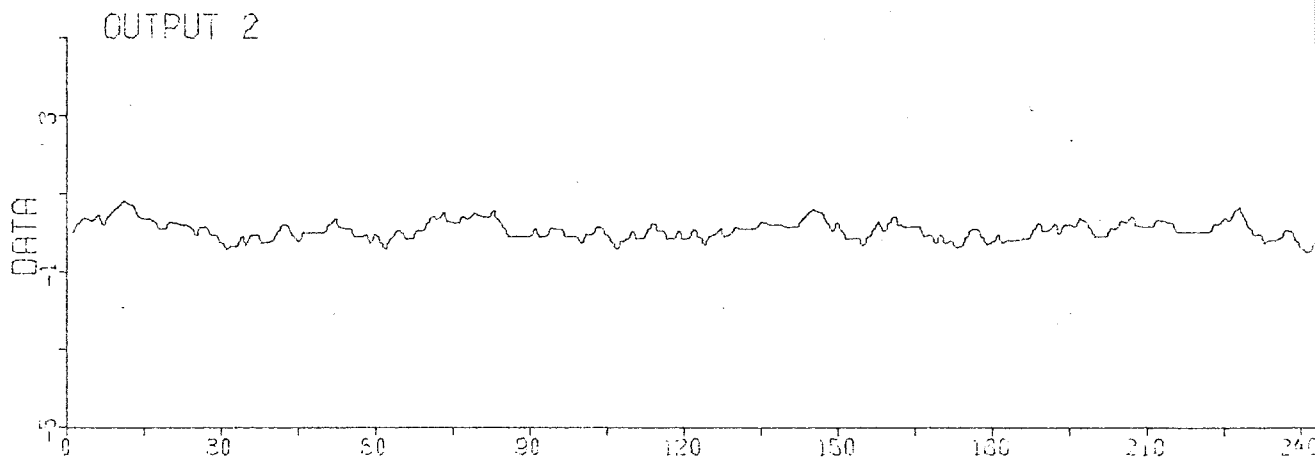
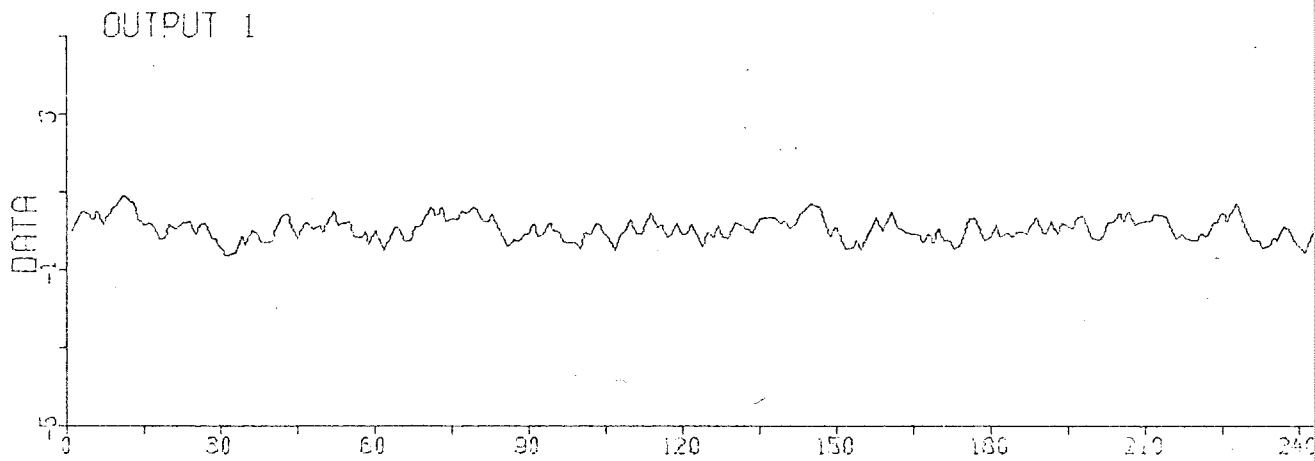
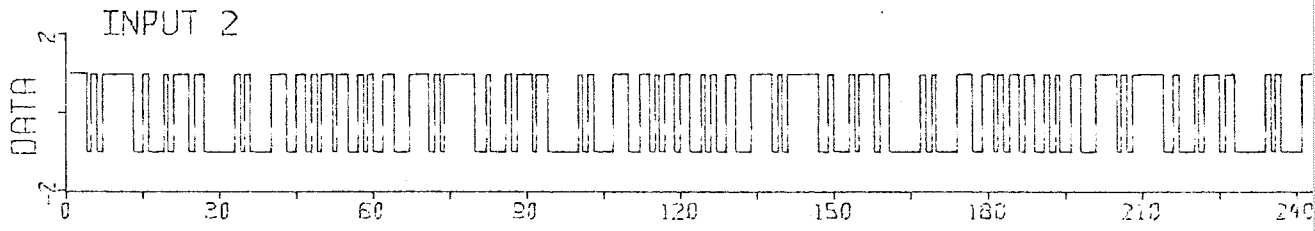
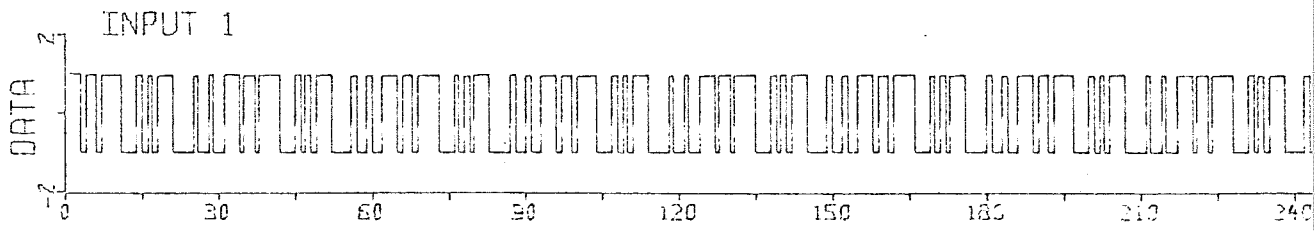


DIAGRAM 6.2

EX 1. $\lambda = 0.1$

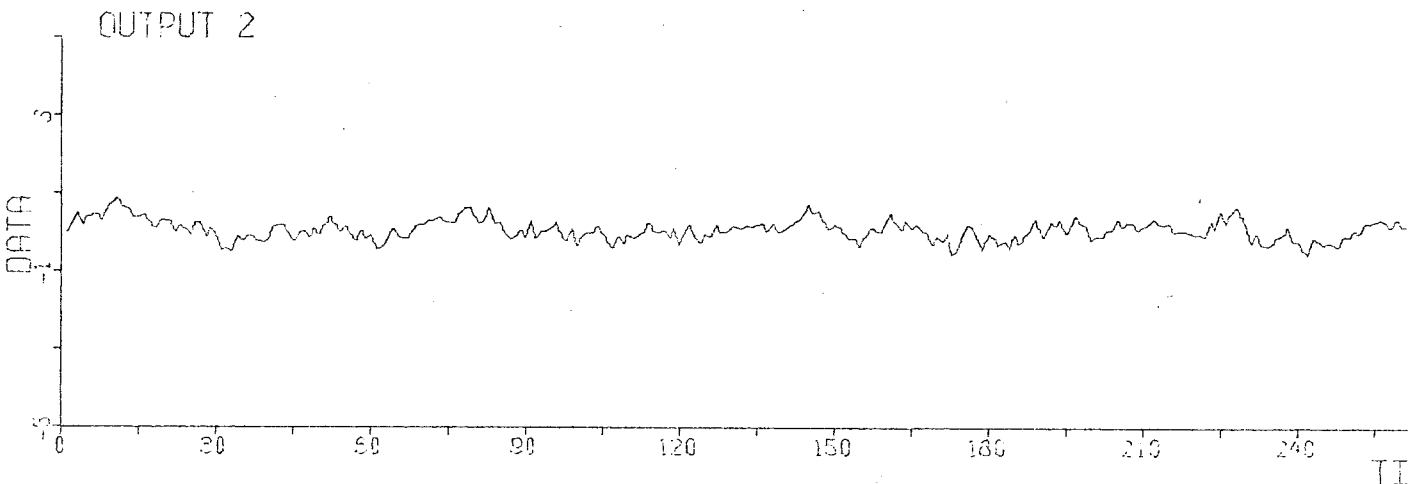
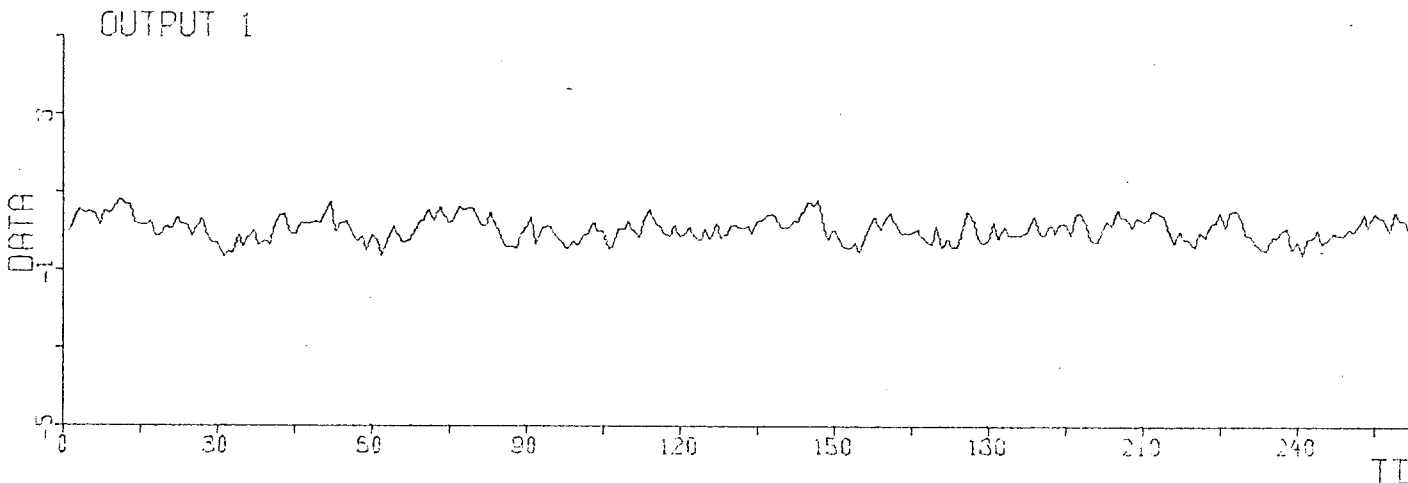
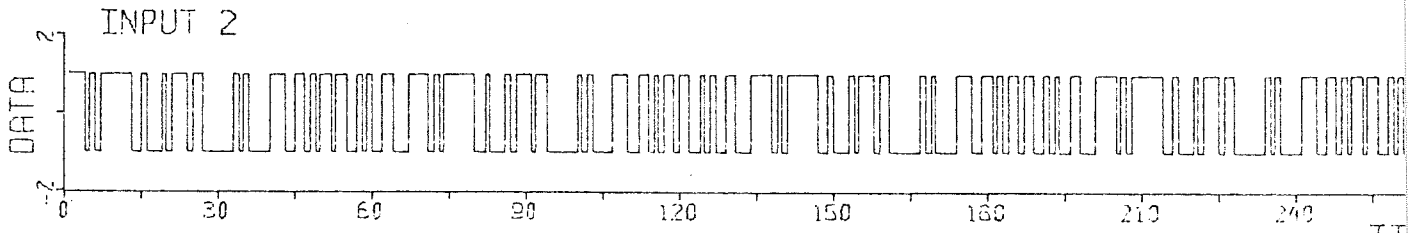
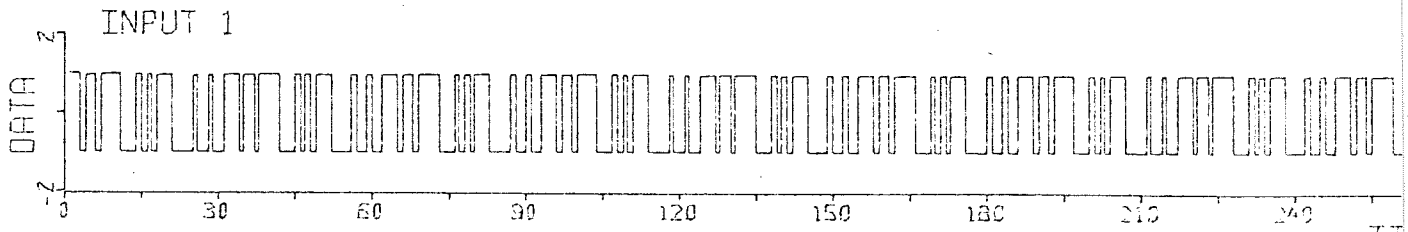
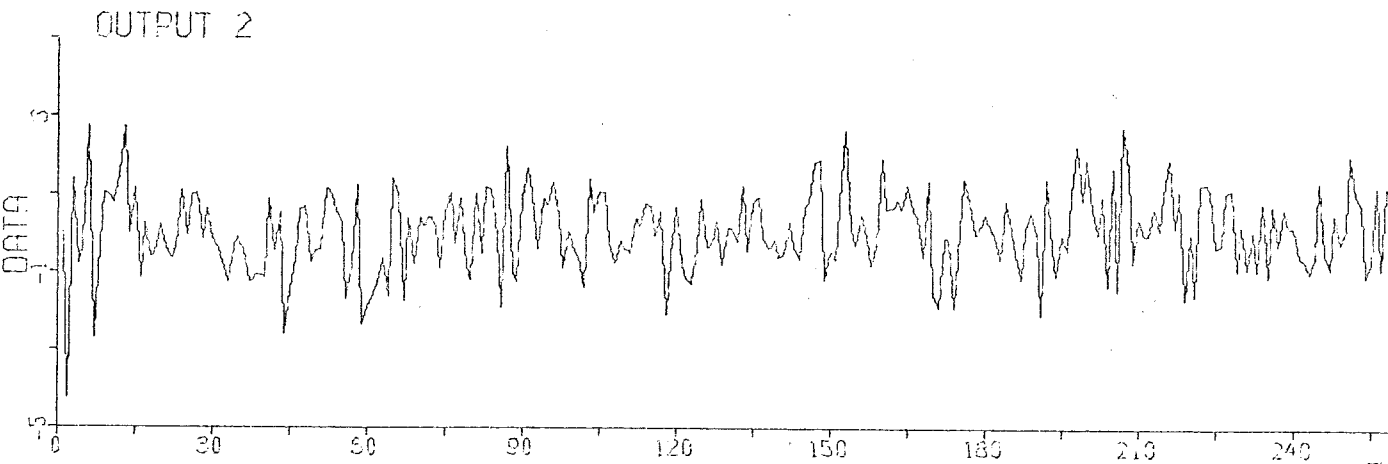
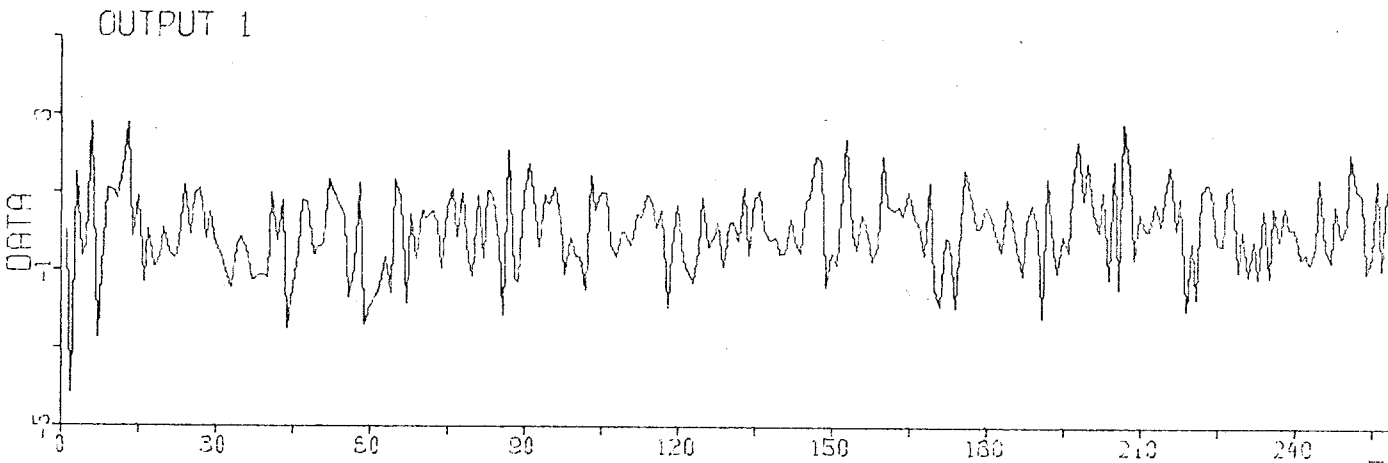
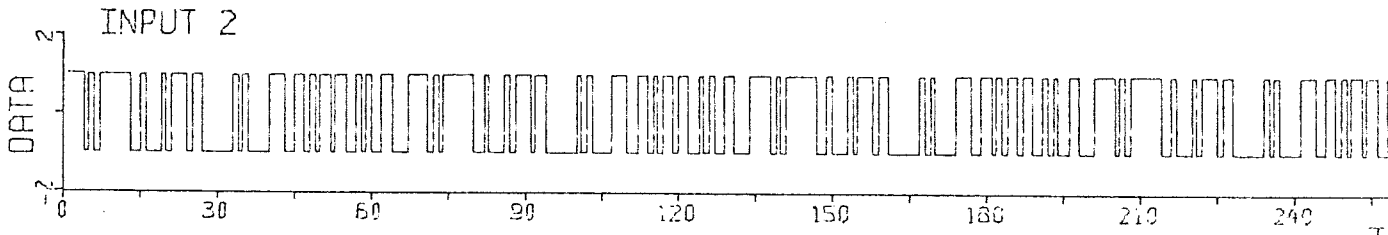
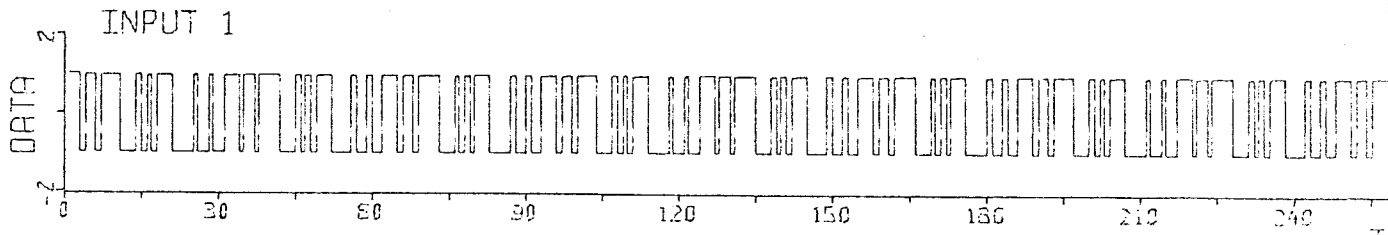
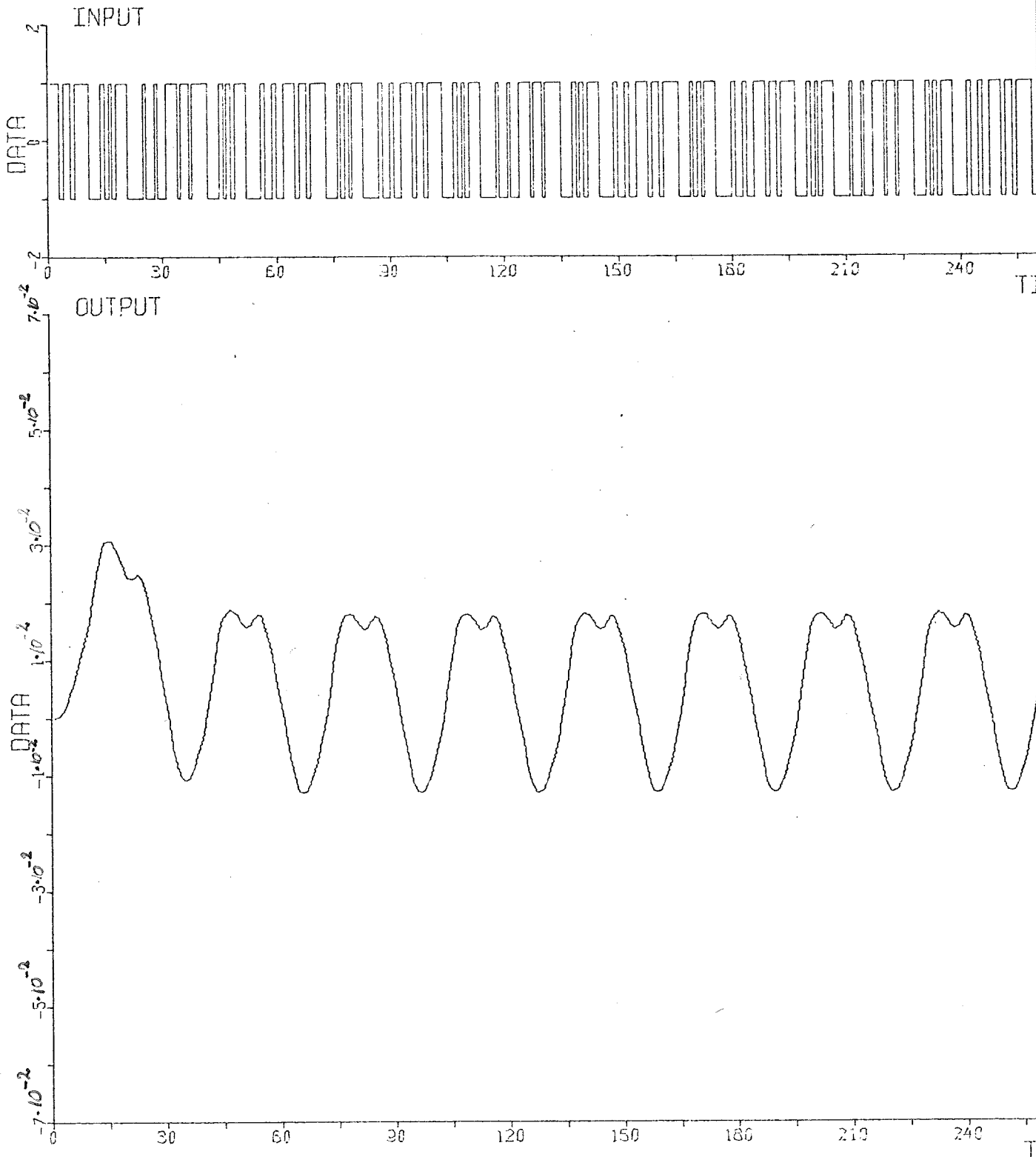


DIAGRAM 6.3

EX.1. $\text{LAMBDA} = 1.0$



EX. 2 LAMBDA=0.0



EX. 2. LAMBDA=0.1

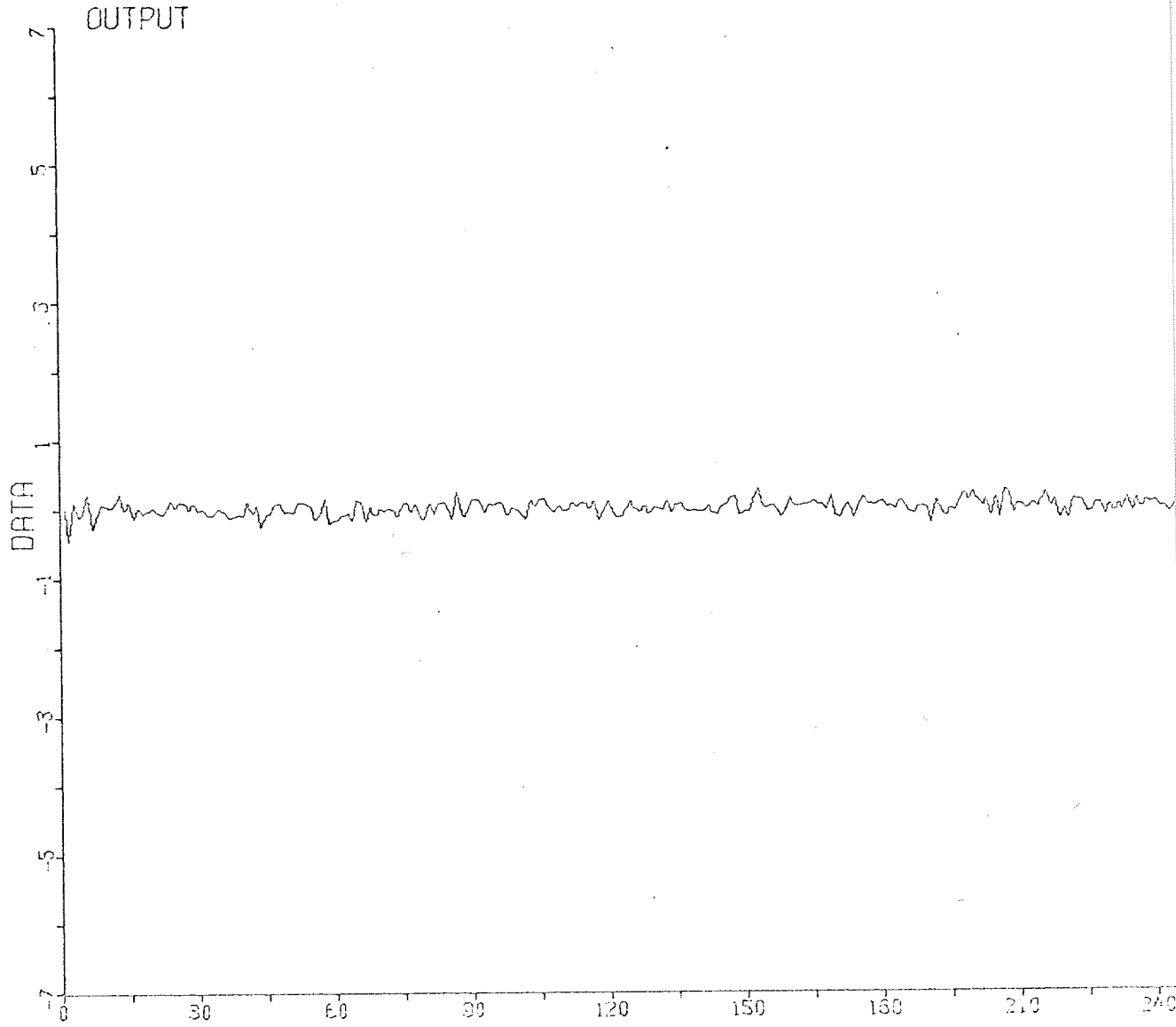
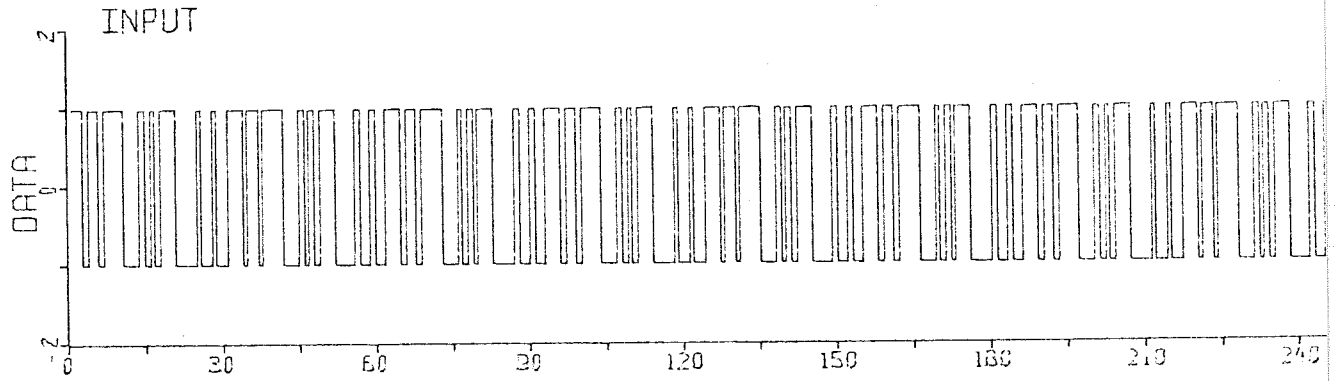
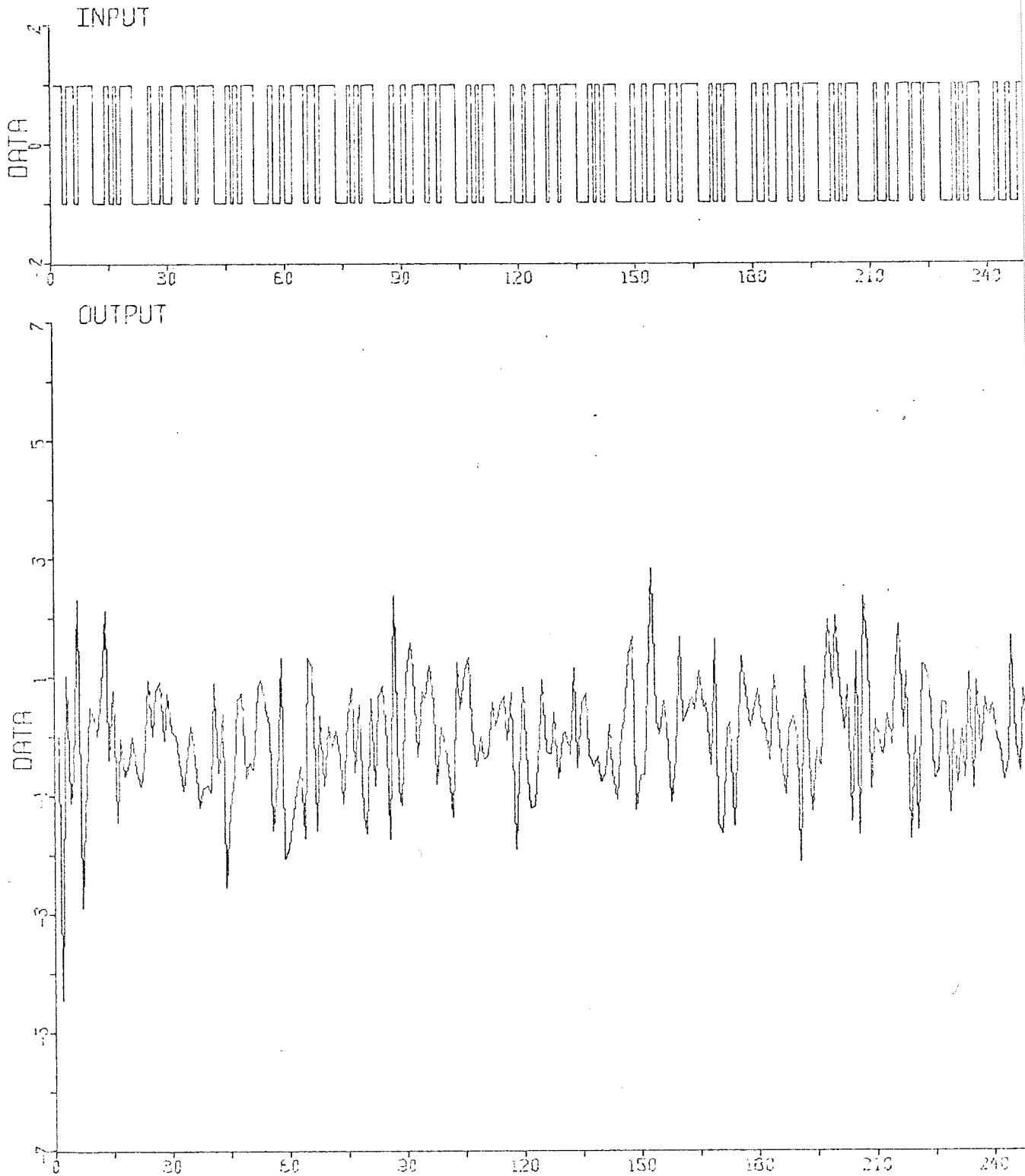


DIAGRAM 6.6

EX. 2

LAMBDA=1.0



Testkörningar

Nedan presenteras ett urval testkörningar. Varje typ av exempel är försett med en mindre kommentar. De beteckningar som användes i tabellerna är:

$x(0)$ - Tillståndsvektorns begynnelsevärde.

$\alpha(0)$ - Initialgissningen på de parametrar som skall identifieras.

$\left\{ \begin{array}{l} \alpha_{\text{resultat}} \\ \alpha_{\text{rätt}} \end{array} \right.$ - Det resultat identifieringen ger och motsvarande riktiga värde ur den fysikaliska modellen.

V_{min} - Den minimipunkt förlustfunktionen antar.

λ - Brusnivån på utsignalen.

Antal eval.- Antalet iterationer i FLEPOMIN. Om denna anges som a+b indikerar detta att FLEPOMIN omstartats p g a mycket dålig konvergens.

Tid i FLEPOMIN - Den tid som "antal eval." tar.

MENU - Antalet använda mätpunkter.

I tabellernas kolumner för V_{min} förekommer en beteckning av följande utseende:

x.x-yy

Detta skall uttydas som $x.x \cdot 10^{yy}$

Exempelvis 1.7-19 skall uttydas som $1.7 \cdot 10^{-19}$

Tabell 6.1

$$\begin{cases} \dot{x} = \begin{pmatrix} \alpha & 0 & 0 \\ 0 & -3 & 0 \\ 0 & 0 & -1 \end{pmatrix} x + \begin{pmatrix} 1 & 0 \\ 0 & 2 \\ 0 & 1 \end{pmatrix} u \\ y = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} x \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
$\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$	-2.	-1.00000000001	-1.	2.18-19	0	11	24 300	400

Tabell 6.2

$$\begin{cases} \dot{x} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -3 & 0 \\ 0 & 0 & \alpha \end{pmatrix} x + \begin{pmatrix} 1 & 0 \\ 0 & 2 \\ 0 & 1 \end{pmatrix} u \\ y = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} x \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
$\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$	-2.	-1.00000000002	-1.	1.7 -19	0	11	24 317	400

2 enkla testkörningar med endast 1 parameter att identifiera.

Tabell 6.3

$$\begin{cases} \dot{x} = \begin{pmatrix} \alpha_1 & 0 & 0 \\ 0 & \alpha_2 & 0 \\ 0 & 0 & -1 \end{pmatrix} x + \begin{pmatrix} 1 & 0 \\ 0 & 2 \\ 0 & 1 \end{pmatrix} u \\ y = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} x \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
$\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$	-2.	-1.00000000005	-1.	1.2 -18	0	24	60 317	400
$\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$	-2.	-2.99999999999	-3.					
$\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$	-2.	-1.00000000001	-1.	5.6 -19	0	26	18 776	100
$\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$	-2.	-2.99999999995	-3.					

Två parametrar att identifiera. Preliminär undersökning ang. betydelsen av antalet mätdata. Vi ser, att identifieringsnoggrannheten ej påverkas om antalet mätdata minskas från 400 till 100. En fullständigare undersökning av detta problem har gjorts med 7 identifierade parametrar. Se tab. 6.10 !

Tabell 6.4

$$\begin{cases} \dot{x} = \begin{pmatrix} \alpha_1 & 0 & 0 \\ 0 & \alpha_2 & 0 \\ 0 & 0 & \alpha_3 \end{pmatrix} x + \begin{pmatrix} 1 & 0 \\ 0 & 2 \\ 0 & 1 \end{pmatrix} u \\ y = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} x \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
$\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$	-2.	-1.00000000000	-1.	7.5 -20	0	32	47 316	200
$\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$	-2.	-3.00000000001	-3.					
$\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$	-2.	-1.00000000002	-1.					

Tabell 6.5

$$\begin{cases} \dot{x} = \begin{pmatrix} \alpha_1 & 0 & \alpha_2 \\ 0 & \alpha_3 & 0 \\ 0 & 0 & \alpha_4 \end{pmatrix} x + \begin{pmatrix} 1 & 0 \\ 0 & 2 \\ 0 & 1 \end{pmatrix} u \\ y = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 3 & 1 \end{pmatrix} x \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0	-2.	-1.0000000000	-1.	7.2 -19	0	36	59 148	200
0	0	-3.8	-11					
0	-2.	-3.0000000002	-3.					
	-2.	-1.0000000004	-1.					

Tabell 6.6

$$\begin{cases} \dot{x} = \begin{pmatrix} \alpha_1 & \alpha_2 & \alpha_3 \\ 0 & \alpha_4 & 0 \\ 0 & 0 & \alpha_5 \end{pmatrix} x + \begin{pmatrix} 1 & 0 \\ 0 & 2 \\ 0 & 1 \end{pmatrix} u \\ y = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} x \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0	-2.	-1.0000000007	-1.	5.9 -19	0	50	90 258	200
0	0	-9.3	-10					
0	0	6.3	-10					
	-2.	-2.9999999987	-3.					
	-2.	-1.0000000001	-1.					

Tabell 6.7

$$\begin{cases} \dot{x} = \begin{pmatrix} \alpha_1 & \alpha_2 & \alpha_3 \\ 0 & \alpha_4 & \alpha_5 \\ 0 & 0 & \alpha_6 \end{pmatrix} x + \begin{pmatrix} 1 & 0 \\ 0 & 2 \\ 0 & 1 \end{pmatrix} u \\ y = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} x \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0 0 0	-2. 0 0	-1.0050 0.0070 0.0783	-1. 0 0	9.6 -04	0	56	Avbruten Jobtiden överskr.	500
	-2. 0 -2.	-2.8503 -0.2791 -1.0869	-3. 0 -1.					
0 0 0	-2. 0 0	-1.0027 -0.0139 0.0132	-1. 0 0	1.5 -04	0	88	Avbruten Jobtiden överskr.	200
	-2. 0 -2.	-2.9863 -0.0063 -0.9930	-3. 0 -1.					
0 0 0	-2. 0 0	-1.0059 0.0076 0.0666	-1. 0 0	4.9 -05	0	80+8	Avbruten Printgr. överskr.	50
	-2. 0 -2.	-2.8773 -0.2503 -1.0704	-3. 0 -1.					
0 0 0	-2. 0 0	-1.5616 0.4440 0.3219	-1. 0 0	7.5 -04	0	80	Avbruten Jobtiden överskr.	10
	-2. 0 -2.	-2.9089 -0.4191 -1.2324	-3. 0 -1.					

I tabell 6.4-7 redovisas resultaten av körningar med successiv ökning av antalet okända parametrar i A-matrisen.

Vi ser, att antalet erforderliga iterationer ökar med ökat antal okända parametrar.

I och med körningarna i tabell 6.7 har även en undersökning av betydelsen av antalet mätdata gjorts. Eftersom det rör sig om 6 okända parametrar, fordras så många iterationer, att såväl jobtiden som printgränsen överskridits. En undersökning av konvergensen efter 56 iterationer visar emellertid, att denna skiljer sig mycket litet i fallen 500, 200 och 50 data, medan fallet 10 data visar en större avvikelse.

Tabell 6.8

$$\begin{cases} \dot{x} = \begin{pmatrix} \alpha_1 & 0 & 0 \\ 0 & \alpha_2 & 0 \\ 0 & 0 & \alpha_3 \end{pmatrix} x + \begin{pmatrix} \alpha_4 & 0 \\ 0 & \alpha_5 \\ 0 & \alpha_6 \end{pmatrix} u \\ y = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} x \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0	-2.	-1.0000000003	-1.	2.8 -19	0	46	212 282	500
0	-2.	-3.0000000003	-3.					
0	-2.	-1.0000000003	-1.					
	0	1.0000000001	1.					
	0	2.0000000002	2.					
	0	1.0000000001	1.					
0	-2.	-0.9461	-1.	4.5	0.1	34	157694	500
0	-2.	-2.9042	-3.					
0	-2.	-0.9740	-1.					
	0	1.0081	1.					
	0	1.9588	2.					
	0	0.9977	1.					

Försök att identifiera element från flera matriser samtidigt, vilket lyckades. 2:a körningen visar, att bruset har en störande inverkan på identifieringsnoggrannheten. Notera även att färre iterationer göres i fallet med "brusiga" data.

Tabell 6.9.

$$\begin{cases} \dot{x} = \begin{pmatrix} \alpha_1 & 0 & \alpha_4 \\ 0 & \alpha_2 & 0 \\ \alpha_5 & 0 & \alpha_3 \end{pmatrix} x + \begin{pmatrix} \alpha_6 & 0 \\ 0 & 2 \\ 0 & 1 \end{pmatrix} u \\ y = \begin{pmatrix} 1 & \alpha_7 & 0 \\ 1 & 0 & 1 \end{pmatrix} x + \lambda e \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0	-2.	-1.0002	-1.	2.6 -03	0	29	Avbruten Jobtiden överskr.	500
0	-2.	-2.8694	-3.					
0	-2.	-0.9832	-1.					
	0.5	-0.0082	0					
	0.5	0.0446	0					
	0	1.0018	1.					
	0	0.9755	1.					
0	-2.	-0.9559	-1.	4.5	0.1	35	Avbruten Jobtiden överskr.	500
0	-2.	-2.7708	-3.					
0	-2.	-0.8966	-1.					
	0.5	-0.0803	0					
	0.5	0.0377	0					
	0	1.0072	1.					
	0	0.9637	1.					
0	-2.	-0.8033	-1.	4.9 +02	1.0	35	Avbruten Jobtiden överskr.	500
0	-2.	-5.5399	-3.					
0	-2.	-0.5094	-1.					
	0.5	0.2767	0					
	0.5	-0.3764	0					
	0	1.1690	1.					
	0	1.1025	1.					

Undersökning av brusets inverkan på identifieringsnoggrannheten om många parametrar loss.

Tabell 6.10

Se tabell 6.9!

x(0)	$\alpha(0)$	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0	-2.	-1.0000000001	-1.	1.0 -19	0	52	110 571	200
0	-2.	-3.0000000002	-3.					
0	-2.	-1.0000000002	-1.					
	0.5	-6.1 -11	0					
	0.5	9.3 -11	0					
	0	1.0000000001	1.					
	0	1.0000000000	1.					
0	-2.	-1.0000000002	-1.	3.8 -19	0	61	70 956	100
0	-2.	-3.0000000000	-3.					
0	-2.	-1.0000000008	-1.					
	0.5	7.3 -12	0					
	0.5	6.5 -11	0					
	0	1.0000000001	1.					
	0	1.0000000000	1.					
0	-2.	-1.0000000003	-1.	2.4 -19	0	63	43 280	50
0	-2.	-2.9999999998	-3.					
0	-2.	-1.0000000004	-1.					
	0.5	4.7 -11	0					
	0.5	2.7 -10	0					
	0	1.0000000002	1.					
	0	1.0000000000	1.					
0	-2.	-1.6898	-1.	6.4 -04	0	80	Avbruten Printgr, överskr.	20
0	-2.	-3.0978	-3.					
0	-2.	-1.5347	-1.					
	0.5	0.4974	0					
	0.5	0.7596	0					
	0	1.0014	1.					
	0	1.0417	1.					
0	-2.	-1.9502	-1.	1.9 -04	0	78	Avbruten Printgr. överskr.	10
0	-2.	-2.7650	-3.					
0	-2.	-1.5376	-1.					
	0.5	0.6144	0					
	0.5	0.8839	0					
	0	0.9953	1.					
	0	0.9972	1.					

Undersökning av antalet mätdatas betydelse för identifieringsnoggrannheten. Vi ser, att noggrannheten ej blir större om antalet ökas över 50 st.

Tabell 6.11

Se tabell 6.9!

x(0)	$\alpha(0)$	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0	-2.	-0.8859	-1.	1.7	0.1	41	88 093	200
0	-2.	-3.0076	-3.					
0	-2.	-0.8891	-1.					
	0.5	-0.0540	0					
	0.5	0.0317	0					
	0	1.0096	1.					
	0	0.9879	1.					
0	-2.	-0.8295	-1.	8.7 -01	0.1	50	58 861	100
0	-2.	-3.0242	-3.					
0	-2.	-0.8647	-1.					
	0.5	-0.0785	0					
	0.5	0.0148	0					
	0	0.9779	1.					
	0	0.9989	1.					
0	-2.	-0.5982	-1.	3.6 -01	0.1	78	Avbruten Printgr. överskr.	50
0	-2.	-2.3776	-3.					
0	-2.	-0.6824	-1.					
	0.5	-0.2620	0					
	0.5	-0.3067	0					
	0	0.9827	1.					
	0	0.8091	1.					

Samma undersökning som i tabell 6.10, men med brusbe-
mängda data. Vi ser, att en minskning av antalet mätpunkter
medför ökat antal iterationer.

Tabell 6.12

$$\begin{cases} \dot{x} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \alpha_1 & 0 \\ 0 & 0 & -1 \end{pmatrix} \dot{x} + \begin{pmatrix} \alpha_2 & 0 \\ 0 & 2 \\ 0 & 1 \end{pmatrix} u \\ y = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} x + \lambda e \end{cases}$$

x(0)	$\alpha(0)$	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
10 5 10	-2. 0.67	-53.866 0.6781	-3. 1.	1.4 +03	0	30	93 308	500
1. 0.5 1.	-2. 0.67	-3.5946 0.8287	-3. 1.	1.5 +01	0	21	64 784	500
0.1 0.1 0.1	-2. 0.67	-3.0400 0.9818	-3. 1.	1.7 -01	0	31	138 447	500
0.1 0.1 0.1	-2. 0.67	-2.9849 0.9995	-3. 1.	4.7	0.1	17	52 491	500

Avsikten var att undersöka hur pass känslig metoden är för ett felaktigt värde på tillståndsvektorns begynnelsevärde. För det fysikaliska systemet gäller, att $x(0) = 0$, samt att tillstånden högst antar värdet c:a 0.4. I startvektorn $\begin{pmatrix} 0.1 \\ 0.1 \\ 0.1 \end{pmatrix}$ är alltså felet relativt stort, men α kan trots detta identifieras med fel först i andra decimalen. Se även tabell 6.13, där startvektorfelets inverkan som funktion av antalet mät-punkter belyses.

Tabell 6.13

Se tabell 6.12.

x(0)	$\alpha(0)$	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0.1	-2.	-1.0815	-1.	1.7 -01	0	36	Avbruten Jobtiden överskr.	500
0.1	-2.	-2.9766	-3.					
0.1	-2.	-0.9957	-1.					
	0.5	-0.0015	0					
	0.5	0.0439	0					
	0	0.9819	1.					
	0	0.9820	1.					
0.1	-2.	-1.0065	-1.	1.3 -01	0	56+15	82535	100
0.1	-2.	-2.9716	-3.					
0.1	-2.	-1.0225	-1.					
	0.5	-0.0728	0					
	0.5	-0.1173	0					
	0	0.9360	1.					
	0	0.9690	1.					
0.1	-2.	-0.9160	-1.	1.1 -01	0	60	40 996	50
0.1	-2.	-3.0513	-3.					
0.1	-2.	-1.0534	-1.					
	0.5	-0.1069	0					
	0.5	-0.1515	0					
	0	0.8644	1.					
	0	0.9582	1.					

Undersökning av hur ett fel i tillståndsvektorns begynnelsevärde påverkar identifieringen, om man endast har ett mindre antal mätvärden till förfogande.

Tabell 6.14

$$\left\{ \begin{array}{l} \dot{x} = \begin{pmatrix} \alpha_1 & 0 & \alpha_2 \\ 0 & -3 & \alpha_3 \\ 0 & \alpha_4 & -1 \end{pmatrix} x + \begin{pmatrix} 1 & 0 \\ \alpha_5 & 2 \\ 0 & \alpha_6 \end{pmatrix} u \\ y = \begin{pmatrix} \alpha_7 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} x \end{array} \right.$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0	-2.	-0.9999999853	-1.				Avbruten	
0	-1.	-1.4	-08				Jobtiden	
0	-1.	2.2	-08	2.1 -15	0	48	överskr.	500
	1.	1.7	-08					
	0	6.4	-08					
	0	0.9999999978	1.					
	0	0.9999999554	1.					

Vi har här släppt loss 7 parametrar på något annorlunda sätt än i föregående ex. Trots att tiden överskrides ser vi att konvergensen är god.

Tabell 6.15

$$\left\{ \begin{array}{l} \dot{x} = \begin{pmatrix} \alpha_1 & 0 & \alpha_2 \\ 0 & -3 & \alpha_3 \\ 0 & \alpha_4 & -1 \end{pmatrix} x + \begin{pmatrix} 1 & 0 \\ \alpha_6 & 2 \\ 0 & \alpha_7 \end{pmatrix} u \\ y = \begin{pmatrix} \alpha_8 & 1 & 0 \\ 1 & \alpha_5 & 1 \end{pmatrix} x \end{array} \right.$$

V.g.V

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU	
		Resultat	Rätt						
0	-2.	-0.9999999999	-1.	8.1 -20	0	52	119 021	200	
0	-1.	-3.8	-10						0
0	-1.	8.7	-10						0
	1.	0.9882							0
	0.1	0.4941							0
	-0.1	1.1	-10						0
	0.5	0.0148							1.
	0.5	0.9999999999							1.
0	-2.	-1.0000000002	-1.						2.7 -19
0	-1.	3.0	-10	0					
0	-1.	-8.3	-10	0					
	0.1	0.5832		0					
	0.1	0.2916		0					
	-0.1	3.2	-10	0					
	0.9	0.4168		1.					
	0.5	0.9999999997		1.					

Vi skall här identifiera 8 parametrar. Det går till synes dåligt. Alla parametrar förutom α_4 , α_7 , α_5 , konvergerar mot rätta värden. Vid närmare undersökning visar sig detta vara kopplingseffekten för systemet som dyker upp på detta sätt. Överföringsfunktionerna överensstämmer nämligen med den för det fysikaliska systemet.

Korrekt överföringsfunktion:

$$G_{\text{fys}} = \begin{pmatrix} \frac{1}{s+1} & \frac{2}{s+3} \\ \frac{1}{s+1} & \frac{1}{s+1} \end{pmatrix}$$

Exemplens överföringsfunktion:

$$G_{\text{Ex1}} = \begin{pmatrix} \frac{1}{s+1} & \frac{2}{s+3} \\ \frac{1}{s+1} & \frac{1.0066s+3.0198}{(s+1)(s+3)} \end{pmatrix}$$

$$G_{\text{Ex2}} = \begin{pmatrix} \frac{1}{s+1} & \frac{2}{s+3} \\ \frac{1}{s+1} & \frac{s+3}{(s+1)(s+3)} \end{pmatrix}$$

Tabell 6.16

$$\begin{cases} \dot{x} = \begin{pmatrix} \alpha_1 & 0 & 0 \\ 0 & \alpha_2 & 0 \\ 0 & 0 & \alpha_3 \end{pmatrix} x + \begin{pmatrix} \alpha_4 & 0 \\ 0 & \alpha_5 \\ 0 & \alpha_6 \end{pmatrix} u \\ y = \begin{pmatrix} \alpha_7 & \alpha_8 & 0 \\ \alpha_9 & 0 & 1 \end{pmatrix} x \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0	-2.	-2.0000000000	-1.	7.9	0	28	44 897	127
0	-2.	-2.0000000000	-3.					
0	-2.	-1.1959	-1.					
	0	0.0000000000	1.					
	0	0.0000000000	2.					
	0	1.1025	1.					
	0	0.0000000000	1.					
	0	0.0000000000	1.					
	0	0.0000000000	1.					
0	-2.	-1.0000000005	-1.	4.6 -19	0	36	59 343	127
0	-2.	-3.0000000002	-3.					
0	-2.	-0.9999999995	-1.					
	0.5	1.1905	1.					
	0.5	1.3612	2.					
	0.5	0.9999999999	1.					
	0.5	0.8399	1.					
	0.5	1.4693	1.					
	0.5	0.8399	1.					

Undersökning av hur många parametrar som max. kan identifieras.

Observera att produkten

$$\begin{aligned} \hat{B}(2,2) \times \hat{C}(1,2) &= B(2,2) \times C(1,2) \\ \hat{B}(1,1) \times \hat{C}(1,1) &= B(1,1) \times C(1,1) \\ \hat{B}(1,1) \times \hat{C}(2,1) &= B(1,1) \times C(2,1) ; \end{aligned}$$

d v s produkterna är felfritt identifierade beroende av kopplingseffekten. V_{min}:s storleksordning implicerar minimum.

Observera även startvärdet α(0):s betydelse!

Tabell 6.17

$$\begin{cases} \dot{x} = \begin{pmatrix} \alpha_1 & 0 & \alpha_4 \\ 0 & \alpha_2 & 0 \\ \alpha_5 & 0 & \alpha_3 \end{pmatrix} x + \begin{pmatrix} \alpha_6 & 0 \\ 0 & \alpha_7 \\ 0 & 1 \end{pmatrix} u \\ y = \begin{pmatrix} 1 & \alpha_8 & 0 \\ 1 & 0 & \alpha_9 \end{pmatrix} x + \lambda e \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0	-2.	-0.9999999999	-1.					
0	-2.	-3.0000000014	-3.					
0	-2.	-1.0000000007	-1.					
	0.5	2.9	-10					
	0.5	1.6	-10	5.4 -19	0	49	80 212	127
	0	1.0000000001	1.					
	0	1.3581 ←	2.					
	0	1.4726 ←	1.					
	0	1.0000000003	1.					

Samma undersökning som tabell 6.16. Notera kopplingen mellan α_7 och α_8 . Produkten är rätt identifierad.

Tabell 6.18

$$\begin{cases} \dot{x} = \begin{pmatrix} \alpha_2 & 0 & 0 \\ 0 & \alpha_1 & 0 \\ 0 & 0 & -1 \end{pmatrix} x + \begin{pmatrix} b & 0 \\ 0 & 2 \\ 0 & 1 \end{pmatrix} u \\ y = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} x + \lambda e \end{cases}$$

Frusen parameter:

Körning 1 :b=0,5
 "- 2,3,4 :b=0,3

x(0)	$\alpha(0)$	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0 0 0	-2.5 -0.5	-2.9509 -0.5065	-3. -1.	7.4	0.1	20	62 013	500
0 0 0	-2.5 -0.5	-2.9732 -0.3739	-3. -1.	5.6	0	14	43 398	500
0 0 0	-2.5 -0.5	-2.9541 -0.3139	-3. -1.	1.0 +01	0.1	20	62 003	500
0 0 0	-2.5 -0.5	-4.6458 -0.4127	-3. -1.	5.0 +02	1.0	33+10	133 578	500

Här har vi frusit parameter B(1,1) = b. Dess riktiga värde är 1. Notera överensstämmelsen mellan den frusna parametern och α_2 . Om vi jämför överföringsmatriserna för körningarna med det fysikaliska systemets ser vi att den statiska förstärkningen överensstämmer.

$$G_{\text{fys.syst.}} = \begin{pmatrix} 1 \cdot \frac{1}{s+1} & \frac{2}{s+3} \\ 1 \cdot \frac{1}{s+1} & \frac{1}{s+1} \end{pmatrix}$$

$$G_{\text{körn.1}} = \begin{pmatrix} 0.8 \cdot \frac{1}{2.68s+1} & \frac{2}{s+2.9732} \\ 0.8 \cdot \frac{1}{2.68s+1} & \frac{1}{s+1} \end{pmatrix}$$

$$G_{\text{körn.3}} = \begin{pmatrix} 0.99 \cdot \frac{1}{1.97s+1} & \frac{2}{s+2.9509} \\ 0.99 \cdot \frac{1}{1.97s+1} & \frac{1}{s+1} \end{pmatrix}$$

Tabell 6.19

$$\begin{cases} \dot{x} = \begin{pmatrix} \alpha_1 & 0 & 0 \\ 0 & -2.5 & 0 \\ 0 & 0 & -1 \end{pmatrix} x + \begin{pmatrix} \alpha_2 & 0 \\ 0 & 2 \\ 0 & 1 \end{pmatrix} u \\ y = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} x + \lambda e \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0 0 0	-0.5 -0.5	-1.0376 1.0022	-1. 1.	2.4 -01	0	27	83 938	500
0 0 0	-0.5 -0.5	-0.9766 1.0088	-1. 1.	4.7	0.1	27	83 903	500
0 0 0	-0.5 -0.5	-1.2533 1.2297	-1. 1.	5.0 +02	1.0	26	80 811	500

Samma undersökning som i tabell 6.18 men med ett egenvärde fruset och med olika brusnivåer. Som synes erhåller vi relativt god konvergens, utom i fallet λ = 1.0.

Tabell 6.20

$$\begin{cases} \dot{x} = \begin{pmatrix} \alpha_1 & 0 & \alpha_2 \\ 0 & -2.5 & 0 \\ 0 & \alpha_3 & 1 \end{pmatrix} x + \begin{pmatrix} 1 & 0 \\ 0 & 2 \\ 0 & 1 \end{pmatrix} u \\ y = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} x + \lambda e \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0 0 0	-0.8 -0.01 0.01	-1.0401 -0.1238 -0.1082	-1.0 0 0	1.7 -01	0	38+3	Avbruten Jobtiden överskr.	500

Den frusna parametern är även nu A(2,2), men våra α_i ligger bara i A-matrisen. På våra α_i :s placering läses två egenvärden. Dålig konvergens.

Tabell 6.21

$$\begin{cases} \dot{x} = \begin{pmatrix} \alpha_1 & 0 & \alpha_2 \\ 0 & -2.5 & 0 \\ \alpha_3 & 0 & -1 \end{pmatrix} x + \begin{pmatrix} 1 & 0 \\ 0 & 2 \\ 0 & 1 \end{pmatrix} u \\ y = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} x + \lambda e \end{cases}$$

x(0)	$\alpha(0)$	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0	-2.	-1.0599	-1.	0.1941	0	25	87 446	500
0	0	-0.0825	0					
0	0	0.0785	0					
0	-2.	-0.9932	-1.	4.6909	0.1	21	73 451	500
0	0	-0.0792	0					
0	0	0.0995	0					
0	-2.	-0.8610	-1.	4.9 +2	1.0	39	Avbruten Jobtiden överskr.	500
0	0	0.2962	0					
0	0	-0.0004	0					
0	-1.5	-1.0527	-1.	0.1942	0	30+11	143 485	500
0	-0.1	-0.0824	0					
0	0.1	0.0768	0					

Våra α_i har omplacerats och endast ett egenvärde är låst. I de första körningarna anges ett V_{min}, men konvergensen osäker, eftersom med andra startvärden indikeras dålig konvergens.

Tabell 6.22

$$\begin{cases} \dot{x} = \begin{pmatrix} \alpha_1 & 0 & 0 \\ 0 & -2.5 & \alpha_2 \\ 0 & \alpha_3 & -1 \end{pmatrix} x + \begin{pmatrix} 1 & 0 \\ 0 & 2 \\ 0 & 1 \end{pmatrix} u \\ y = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} x + \lambda e \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0	-2.	-0.9982	-1.	0.058	0	13+15	97 994	500
0	-1.	-0.4567	0					
0	1.	0.0104	0					
0	-2.	-0.9996	-1.	0.059	0	32+ 8	139 851	500
0	0.2	-0.4415	0					
0	0.2	-0.0065	0					

Trots att vi ej har låst något egenvärde får vi dålig konvergens. Egenvärdet som skall bli - 3,0 rör sig mycket litet från det frusna värdet.

Tabell 6.23

$$\begin{cases} \dot{x} = \begin{pmatrix} \alpha_1 & 0 & 0 \\ 0 & -2.5 & \alpha_2 \\ 0 & \alpha_3 & -1 \end{pmatrix} x + \begin{pmatrix} 1 & 0 \\ \alpha_4 & 2 \\ 0 & 1 \end{pmatrix} u \\ y = \begin{pmatrix} 1 & 1 & 0 \\ 1 & \alpha_5 & 1 \end{pmatrix} x \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0	-2.	-0.9859	-1.	0.1245	0	36	Avbruten Jobtiden överskr.	500
0	-1.	-0.7610	0					
0	1.	-0.1264	0					
0	0	0.0652	0					
0	-2.	-0.9375	-1.	0.3121	0	32	Avbruten Jobtiden överskr.	500
0	0.2	-1.0979	0					
0	0.2	-0.3671	0					
0	0	0.1548	0					
0	0	-1.1209	0					

Först släppes en parameter i B-matrisen och därefter en i C-matrisen som är menade att få egenvärdet -2.5 att närma sig -3 som i den fysikaliska modellen. Egenvärdet närmar sig -3 men konvergensen är obefintlig.

Tabell 6.24

$$\begin{cases} \dot{x} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -3 & 0 \\ 0 & 0 & -1 \end{pmatrix} x + \begin{pmatrix} 1 & 0 \\ 0 & 2 \\ 0 & 1 \end{pmatrix} u \\ y = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} x + \begin{pmatrix} \alpha_1 & 0 \\ 0 & \alpha_2 \end{pmatrix} u + \lambda e \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0	0	0.5002	0.5	5.1	0.1	8	24 775	500
0	0	0.9926	1.					

Detta är det enda ex. som körts med D-matris. Det är lika lätt att identifiera parametrar i D-matrisen som i övriga matriser, när vi har $\lambda = 0.1$.

Tabell 6.25

$$\begin{cases} \dot{x} = \begin{pmatrix} \alpha & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & -3 \end{pmatrix} x + \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} u \\ y = \left(\frac{1}{2} \quad -1 \quad \frac{1}{2} \right) x + \lambda e \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0 0 0	-2.	-1.0000000003	-1.	5.2-20	0	23	43 056	400
0.1 0.1 0.1	-2.	-1.0209	-1.	3.1-04	0	27	50 552	400
0 0 0	-2.	-0.9454	-1.	1.9+00	0.1	27	51 988	400
0 0 0	-2.	-0.8013	-1.	1.9+02	1.0	17	32 713	400
0 0 0	-1.1	-1.0000000005	-1.	1.4-19	0	16	30 794	400
0 0 0	-3.	-1.0000000013	-1.	9.3-20	0	22	42 339	400
0 0 0	-4.	-1.	0	5	Avbruten Instabilt	400

System 3 har nu transformerats till diagonalform.

Vi identifierar först parameter A(1,1) med startvärde -2 och får god konvergens dock med ett ganska stort antal iterationer. När sedan samma körning görs med brusiga data, ser vi, att bruset liksom tidigare har stor inverkan på resultatet. Ett startvärde i x gör även att parametern blir svår att identifiera med stor precision. Vi har även tittat på olika startvärdens betydelse. När vi startar så långt ifrån det rätta värdet som

-4 tar FLEPOMIN så stora steg att vi går förbi det rätta värdet och kommer in i det instabila området.

Tabell 6.26

$$\left\{ \begin{array}{l} \dot{x} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & \alpha & 0 \\ 0 & 0 & -3 \end{pmatrix} x + \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} u \\ y = \left(\frac{1}{2} \quad -1 \quad \frac{1}{2} \right) x + \lambda e \end{array} \right.$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0 0 0	-1.	-1.9999999986	-2.	5.5 -20	0	16	30796	400
0 0 0	-3.	-1.9999999985	-2.	7.2 -20	0	20	38929	400

Två olika startvärden till parameter A(2,2). Båda fallen ger avvikelse i de två sista decimalerna.

Tabell 6.27

$$\left\{ \begin{array}{l} \dot{x} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & \alpha \end{pmatrix} x + \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} u \\ y = (0,5 \quad -1 \quad 0,5) x + \lambda e \end{array} \right.$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0 0 0	-4.	-3.0000000029	-3.	2.5-20	0	28	53 863	400
0 0 0	-2.	-3.0000000002	-3.	5.2-20	0	31	60 638	400

De två startvärdena på A (3,3) ger resultat med något olika avvikelse i sista decimalerna. Båda körningarna kräver ett mycket stort antal iterationer.

Tabell 6.28

$$\begin{cases} \dot{x} = \begin{pmatrix} \alpha_1 & 0 & 0 \\ 0 & \alpha_2 & 0 \\ 0 & 0 & -3 \end{pmatrix} x + \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} u \\ y = (0,5 \quad -1 \quad 0,5) x + \lambda e \end{cases}$$

x(0)	$\alpha(0)$	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0 0 0	-2. -1.	-1.0000000003 -1.9999999994	-1. -2.	4.6-20	0	50	109 756	400
0 0 0	-2. -1.	-1.0000000009 -1.9999999987	-1. -2.	1.5-19	0	72	45 010	100
0 0 0	-1.1 -2.1	-1.0000000005 -2.0000000004	-1. -2.	9.7-20	0	30	18 800	100
0 0 0	-1.1 -2.1	-1.0000000003 -1.9999999985	-1. -2.	2.3-19	0	27	59 848	400
0 0 0	-2. -3.	{-1.25...} {-1.49...}	-1. -2.	(2.4-02)	0	(64)	Avbruten Jobtiden överskr.	400

Vi släpper här loss två parametrar i diagonalen. Notera den kraftiga sänkningen av antalet iterationer när vi startar nära de riktiga värdena. Om vi minskar antalet mätdata blir inverkan på antalet iterationer uppenbar, vilket ej var fallet i system 1. I tre av de fyra avslutade körningarna identifieras α_2 med minst precision.

Tabell 6.29

$$\begin{cases} \dot{x} = \begin{pmatrix} \alpha_1 & 0 & 0 \\ 0 & \alpha_2 & 0 \\ 0 & 0 & \alpha_3 \end{pmatrix} x + \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} u \\ y = (0,5 \quad -1 \quad 0,5) x \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0	-2.	-1.716	-1.	(1.9+00)	0	57	Avbruten Jobtiden överskr.	400
0	-1.	-1.975	-2.					
0	-2.	-1.716	-3.					
0	-2.	-1.710	-1.	5.4-03	0	32+18	36 600	100
0	-1.	-1.949	-2.					
0	-2.	-1.710	-3.					
0	-2.	-1.19	-1.	(1.1-03)	0	54+ ..	Avbruten Jobtiden överskr.	400
0	-1.	-2.41	-2.					
0	-4.	-3.97	-3.					
0	-1.1	-1.025	-1.	(2.2-05)	0	44+...	Avbruten Jobtiden överskr.	400
0	-2.1	-2.056	-2.					
0	-3.1	-3.128	-3.					

När vi släpper loss hela diagonalen får vi stora svårigheter vid identifieringen. Om α_1 och α_3 har samma startvärde kan dessa parametrar ej skiljas åt. Även om vi har olika startvärden får vi ingen konvergens.

Tabell 6.30

$$\begin{cases} \dot{x} = \begin{pmatrix} -6 & 1 & 0 \\ -11 & 0 & 1 \\ -6 & 0 & 0 \end{pmatrix} x + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} u \\ y = (1 \ 0 \ 0)x \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0 0 0	A(1,1) = -3.	-6.0000000101	-6.	2.7-20	0	50	97 388	400
0 0 0	A(3,1) = -5.	-5.9999999984	-6.	9.9-20	0	16	31 137	400
0 0 0	A(2,1) = -10.	-11.0000000000	-11.	8.9-20	0	17	33 109	400
0 0 0	A(2,1) = -10.	-8.94956..	-11.	1.9+00	0.1	14	27 306	400
0 0 0	A(2,1) = -5.5	-11.0000000000	-11.	8.9-20	0	22	42 322	400

Körningarna innebär en försiktig sondering av system 2 på companionform. Det gäller att identifiera karakteristiska ekvationens koefficienter en i taget. Notera brusets förvånansvärt stora inverkan på identifieringsnoggrannheten. (rad 4)!

Tabell 6.31

$$\begin{cases} \dot{x} = \begin{pmatrix} -6 & 1 & 0 \\ \alpha_1 & 0 & 1 \\ \alpha_2 & 0 & 0 \end{pmatrix} x + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} u \\ y = (1 \ 0 \ 0) x \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0 0 0	-10. - 5.	-11.0000000030 - 5.9999999984	-11. - 6.	9.1-20	0	52	115 314	400
0 0 0	-10. - 5.	- 9.46... - 9.29...	-11. - 6.	1.9+00	0.1	62	Avbruten Jobtiden överskr.	400

Här har försök gjorts att identifiera 2 parametrar, vilket går utmärkt med deterministiska data. Bruset har även här katastrofal inverkan på identifieringsnoggrannheten.

Tabell 6.32

$$\begin{cases} \dot{x} = \begin{pmatrix} \alpha_1 & 1 & 0 \\ \alpha_2 & 0 & 1 \\ \alpha_3 & 0 & 0 \end{pmatrix} x + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} u \\ y = (1 \ 0 \ 0) x \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0	- 5.	- 5.9926..	- 6.	2.5-04	0	8	Avbruten Instabilt	400
0	-10.	-10.5422..	-11.					
0	- 5.	- 5.2887..	- 6.					
0	-5.992	- 5.7428..	- 6.	9.2-05	0	8	Avbruten Instabilt	400
0	-10.542	-10.6668..	-11.					
0	-5.288	- 5.5674	- 6.					

Här gäller det att identifiera karakteristiska ekvationens samtliga koefficienter. Systemet blir emellertid instabilt under exekveringens gång, varför denna avbrytes. Ett försök till omstart har gjorts, och vi ser, att förlustfunktionen V har minskat ytterligare innan systemet ånyo blir instabilt. Ytterligare omstarter borde ge konvergens.

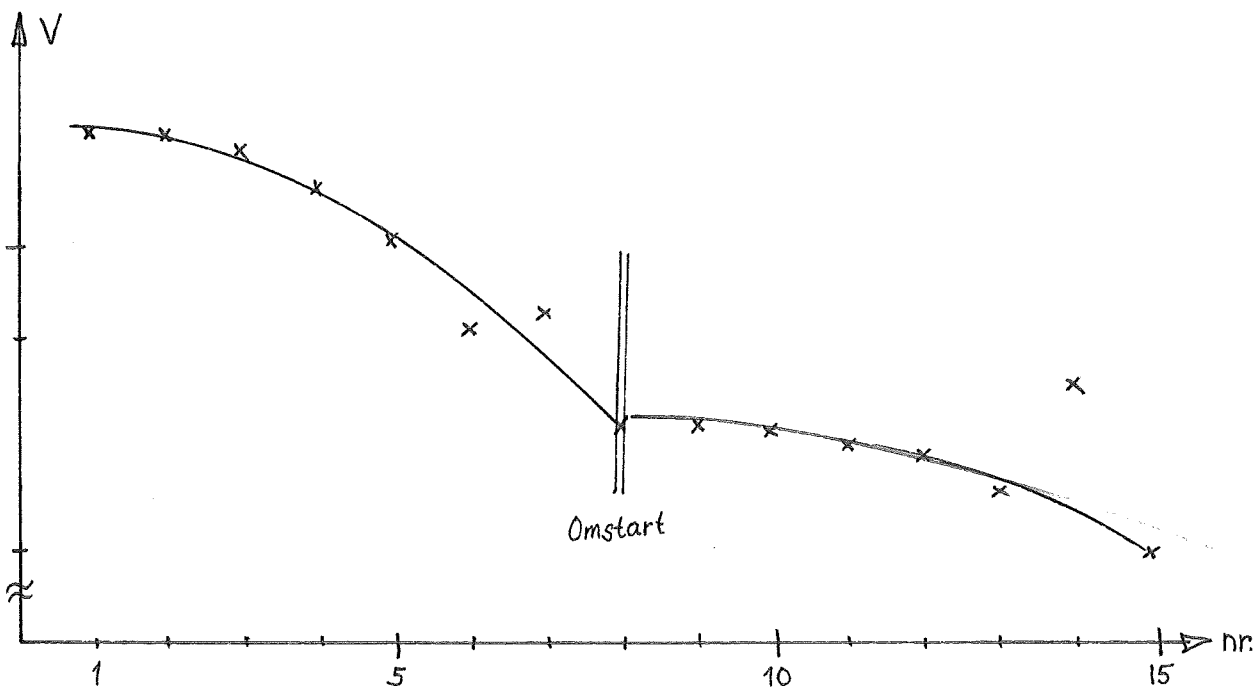


Fig 6.1 - Visar hur förlustfunktionens värde är avtagande.

Tabell 6.33

$$\dot{x} = \begin{pmatrix} -5.75 & 1 & 0 \\ -10.75 & 0 & 1 \\ -5.75 & 0 & 0 \end{pmatrix} x + \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{pmatrix} u$$

$$y = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} x$$

Förlustfunktionens värde
vid första evalueringen:

1.125-01

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0	0.25	-0.0001802	0.	1.4-06	0	54	Avbruten Jobtiden överskr.	400
0	0.25	0.0002107	0.					
0	1.25	0.9604662	1.					

Tabell 6.34

$$\dot{x} = \begin{pmatrix} \alpha_1 & 1 & 0 \\ \alpha_2 & 0 & 1 \\ \alpha_3 & 0 & 0 \end{pmatrix} x + \begin{pmatrix} -0.0001802 \\ 0.0002107 \\ 0.9604662 \end{pmatrix} u$$

$$y = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} x$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0	-5.75	-5.7671	-6.	1.04-06	0	30+25		400
0	-10.75	-10.7244	-11.					
0	-5.75	-5.7629	-6.					

Kommentar till tabell 6.33-34

Avsikten var att undersöka huruvida följande iterativa identifieringsmetod för överföringsfunktionen är användbar:

Antag, att vi har ofullständiga uppgifter om överföringsfunktionens samtliga parametrar. Vi vill finna de exakta värdena genom att:

- 1) Ställa systemet på companionform (idetta fall obs. bar kanonisk form).
- 2) Frysa matris A:s element med våra initialgissningar på a_i .
- 3) Identifiera B-matrisens element och använda initialgissningarna på b_i som begynnelsevärden.

- 4) Identifiera A-matrisens element med B-matrisen frusen enligt (3).
- 5) Ersätta ordet "gissningar" ovan med orden "senaste estimat".
- 6) GO TO 2.

Vi ser, att B-matrisens element lätt låter sig identifieras, trots att matris A är frusen. A-matrisen visar emellertid förvånansvärt dålig konvergens, speciellt med tanke på körningen i tabell 6.32, som visar, att det är möjligt att identifiera samtliga koefficienter i karakteristiska ekvationen.

Förlustfunktionen V har emellertid i en iteration minskats från $1.125 \cdot 10^{-1}$ till $1.045 \cdot 10^{-6}$, vilket ju visar, att metoden erbjuder en framkomlig väg.

Test av STRUCNL och FGCLI

För att testa STRUCNL och FGCLI, som ju dock var konstruerade för en olinjär modell, har vi som en försiktig början valt att göra identifieringen i det linjära systemen 1 och 2. Vi kan på så sätt göra direkta jämförelser med motsvarande körningar med STRUC och FGLI. Det är främst FGCLI vi är intresserade av.

$$\begin{cases} \dot{x} = \begin{pmatrix} -6 & 1 & 0 \\ \alpha & 0 & 1 \\ -6 & 0 & 0 \end{pmatrix} x + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} u \\ y = (1 \ 0 \ 0) x \end{cases} \quad \text{Tabell 6.35}$$

KÖRNING NR.	1	2	3
	TSAMP=0.100 PSAMP=0.050 XSAMP=0.025	TSAMP=0.100 PSAMP=0.050 XSAMP=0.010	TSAMP=0.100 PSAMP=0.020 XSAMP=0.010
Startvärde (0)	-10.	-10.	-10.
Förlustfknen 1:a ev. D:o med FGLI	2.2602522734-04 2.2602484068-04	2.2602484507-04 2.2602484068-04	2.2602484507-04 2.2602484068-04
Gradienten 1:a ev. D:o med FGLI	3.5059436446-04 4.7354031474-04	3.5059402713-04 4.7354031474-04	4.1174714318-04 4.7354031474-04
efter 18 iterat. D:o med FGLI	-11.0003264930 -11.0000000030	Endast 3 iterat.	Endast 3 iterat.
Rätt värde på	-11.		
Använd tid	10 min. 10 sek.	3 min. 10 sek.	3 min 08sek.

Här ser vi att FGCLI fordrar avsevärt mer maskintid än FGLI. I körning nr 1 ser vi, att tycks konvergera men långsammare än vid användning av FGLI. Detta beror med största sannolikhet på att gradienten hela tiden får ett något felaktigt värde.

Gradienten förbättras ej om vi gör tätare Runge-Kutta integrationer (se körning 2). Däremot förbättras förlustfunktionens värde.

Om vi gör tätare evalueringar av λ (i körning 3 göres 5 λ -evalueringar/samplingsintervall) förbättras även gradientens värde, men varje iteration tar mycket lång tid.

Tabell 6.36

$$\begin{cases} \dot{x} = \begin{pmatrix} -6 & 1 & 0 \\ \alpha_1 & 0 & 1 \\ \alpha_2 & 0 & 0 \end{pmatrix} x + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} u \\ y = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} x \end{cases}$$

Startvärde hos α	$\begin{pmatrix} -10. \\ -5. \end{pmatrix}$
Förlustfunkt. efter 1:a evalueringen	5.8488344743-04
D:o beräknad med hjälp av FGLI	5.8488344860-04
Gradienten efter 1:a evalueringen	$\begin{cases} 4.1018965220-04 \\ 7.4693891791-04 \end{cases}$
D:o beräknad med hjälp av FGLI	$\begin{cases} 4.6601036872-04 \\ 8.3179296150-04 \end{cases}$
TSAMP	0.100
PSAMP	0.020
XSAMP	0.010
Använd tid	3 min. 10 sek.
Antal medhugna evalueringar	1

Avsikten var att testa program FGCLI med NUPA \neq 1. Vi ser, att förlustfunktionen stämmer bra överens programmen emellan, medan gradienten beräknas med sämre resultat av FGCLI.

Tabell 6.37

$$\begin{cases} \dot{x} = \begin{pmatrix} \alpha_1 & 0 & 0 \\ 0 & \alpha_2 & 0 \\ 0 & 0 & -3 \end{pmatrix} x + \begin{pmatrix} 1 & 0 \\ 0 & 2 \\ 0 & 1 \end{pmatrix} u \\ y = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} x \end{cases}$$

Startvärde hos α	$\begin{pmatrix} -2. \\ -2. \end{pmatrix}$
Förlustfunkt. efter 1:a evalueringen	1.8705357588+00
D:o beräknad med hjälp av FGLI	1.8705357606+00
Gradienten efter 1:a evalueringen	$\begin{cases} -1.2654720135+00 \\ 2.5337184125+00 \end{cases}$
D:o beräknad med hjälp av FGLI	$\begin{cases} -1.1953883900+00 \\ 2.3027230559+00 \end{cases}$

forts.

TSAMP	0.100
PSAMP	0.020
XSAMP	0.010
Använd tid	3 min. 10 sek.
Antal medhugna evalueringar	1 st.

Här är det också frågan om ett test av FGnLI; denna gången med två in- och två utsignaler. Avsikten var att spåra eventuella logiska fel i programmet. Förlustfunktionen stämmer bra, och felen i gradienten skyller vi på en alltför gles evaluering av λ -ekvationen, varför vi tror oss kunna konstatera, att FGnLI ej är behäftad med något logiskt fel.

Efter detta lugnande besked kan vi ge oss i kast med en olinjär modell.

Tabell 6.38

För körningarna i tab 6.38-41 gäller:

$$\begin{cases} \dot{x} = -x + x^2 + u \\ y = \alpha x \end{cases}$$

TSAMP= 0.0100
PSAMP= 0.0050
XSAMP= 0.0025

x(0)	$\alpha(0)$	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0	0.9	1.0000000058	1.	2.4-17	0	6	23 497	200

Tabell 6.39

$$\begin{cases} \dot{x} = -x + \alpha x^2 + u \\ y = x \end{cases}$$

x(0)	$\alpha(0)$	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0	4.5	5.0000003370	5.	4,9-18	0	15	61 997	200

Dessa båda körningar är en första försiktig sondering av möjligheterna att identifiera koefficienterna i en olinjär modell. Som synes, är det inga problem att identifiera en lossläppt parameter.

Tabell 6.40

$$\begin{cases} \dot{x} = a_1 x + a_2 x^2 + u \\ y = x \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0	-0.9 4.5	-0.9679427619 4.4967896999	-1. 5.	3.3-07	0	51	Avbruten Jobtiden överskr.	200

Två parametrar loss. Det tycks erbjuda svårigheter att identifiera koefficienten framför x^2 -termen. Detta är inte speciellt förvånande, eftersom x^2 -termen är mycket liten.

Tabell 6.41

$$\begin{cases} \dot{x} = -x + x^2 + u \\ y = x + \sin \alpha x \end{cases}$$

x(0)	α(0)	α		V _{min}	λ	Ant. eval.	Tid i FLEPOMIN	MENU
		Resultat	Rätt					
0	0.5	0.0000000006	0.	2.5-17	0	8	31 971	200

Vi har här släppt loss en parameter belägen "innuti" en olinjär term, vilket inte tycks erbjuda några problem vid identifieringen.

7. DISKUSSION

Testkörningarna har gett entydiga svar på en del av frågeställningarna från föregående kapitel. Detta gäller framför allt brusets, initialvärdefelets $x(0)$ och mätpunktantalets inverkan på identifieringsnoggrannheten. Beträffande de övriga frågorna kan svaren ej bli lika enkla.

Inverkan av antalet mätpunkter.

Rent intuitivt inses, att ett ökat antal mätpunkter medför ökad information, vilket bör innebära ökad identifieringsnoggrannhet. Detta verifieras också av våra testkörningar. Dock är att märka, att vid användning av deterministiska data räcker ett visst begränsat antal mätpunkter; noggrannheten kan sedan ej bli större. För system 1 och med 7 identifierade parametrar ligger denna gräns vid omkring 50 mätpunkter. Se tab. 6.10!

Noteras bör också, att för samma startvärde fordras flera iterationer i FLEPOMIN om antalet mätpunkter minskas. Varje iteration tar emellertid då allt kortare maskintid i anspråk, varför minskning av antalet mätpunkter medför tidsvinst.

Brusets inverkan.

Vi kan utan vidare fastslå, att ökat brus ger mindre identifieringsnoggrannhet, vilket ju inte är speciellt förvånansvärt.

$\lambda = 0.1$ innebär, att brusets amplitud i 99% är ≤ 0.3 . Detta kan jämföras med utsignalernas värde, som för system 1 ligger inom ± 0.8 . Det rör sig alltså om en relativt hög brusnivå, och trots detta är relativa identifieringsfelet $< c:a 5\%$ i fallet med 6 identifierade parametrar.

$\lambda = 1.0$ har i samtliga fall visat sig ge en alltför hög brusnivå. Vi har dock kunnat identifiera parametrarna till storleksordning.

Brusets menliga inverkan minskar med ökat antal mätpunkter. Se tab. 6.11!

Initialvärdet $x(0)$.

Inverkan av ett felaktigt initialvärde hos tillståndsvariablerna kan i viss mån liknas vid brus. Felsets inverkan minskar med ökat antal mätpunkter. Se tab. 6.13!

Orsaken inses direkt i fallet linjära system om systemekvationerna löses. Vi får då:

$$y(t) = Du(t) + C e^{At} a + C \int_0^t e^{A(t-s)} Bu(s) ds, \text{ där } x(0) = a$$

Felkällan är att finna i termen $C e^{At} a$, där a är given med viss osäkerhet. Inverkan av begynnelsevärdet, dvs av denna term, på utsignalen minskar med ökad tid (t). Snabbheten på denna "utklingning" bestäms väsentligen av A -matrisen egenvärden.

Vektor α :s startvärde.

Inverkan av vektor α :s startvärde på konvergensen och på konvergensthastigheten kan synas helt slumpmässig. Man kan ej generellt säga, att ett startvärde nära det riktiga ger snabbare och säkrare konvergens än ett startvärde med större fel. Man bör emellertid undvika startvärdet 0, speciellt då för koefficienter framför in- och utsignaler. Koefficienten 0 innebär ju, att denna in- eller utsignals inverkan helt eller delvis försvinner. Se tab. 6.16! Vidare är det viktigt att kontrollera, att startvärdet ger ett stabilt system.

Startvärdets problematik belyses bäst om man betraktar förlustfunktionen och dess gradient samt iterationsformeln

$$\alpha_{k+1} = \alpha_k - \beta HG, \text{ där } \beta \text{ är en pos. skalär.}$$

För fallet med två obekanta parametrar α_1 och α_2 , kan $V=f(\alpha)$ illustreras med nivåkurvor och $dV/d\alpha$ med pilar enligt fig 7.1. Vi har gjort två testkörningar med olika startvärden, men med lika "avstånd" från det riktiga α -värdet. Det gäller här system 2; körningar enligt tab. 6.28.

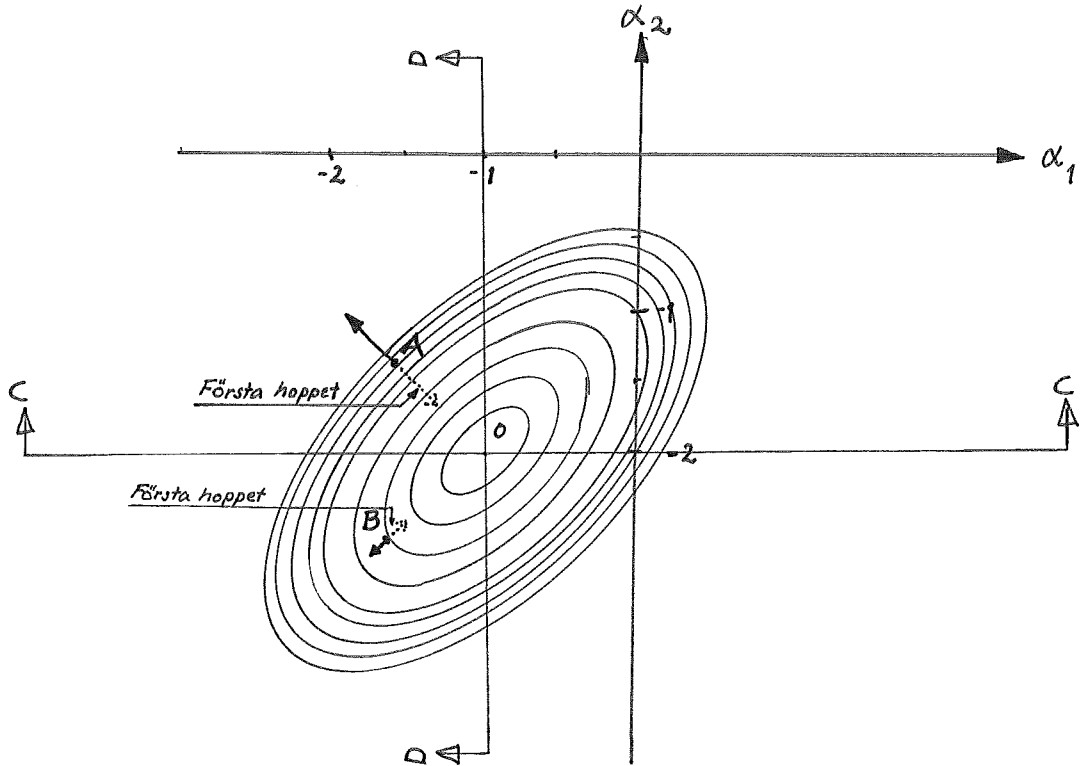


Fig 7.1 - Illustrerar förlustfunktionen åskådliggjord meddelst nivåkurvor.

Vi ser, att redan efter första iterationssteget eller "hoppet" har A kommit mycket närmare "målet" 0 än B. A når målet efter 50 iterationer, medan B efter 64 iterationer fortfarande har nästan halva vägen kvar.

Förlustfunktionens "nivåyta" illustreras även i fig. 7.2, som visar utseendet av snittet c-c och D-D i ovanst. fig.

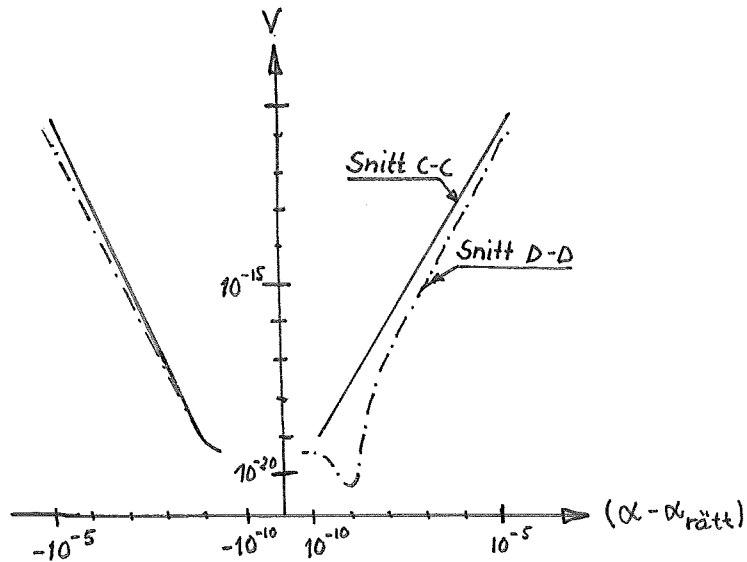


Fig. 7.2

Vi ser i diagrammet (som är logaritmiskt) att kurvornas lutningar är ≈ 2 . Därav följer, att förlustfunktionens värde är ung. proportionellt mot kvadraten på α_i :s avvikelse från sitt rätta värde. Diagrammet uppvisar även ett oväntat läge för minimipunkten hos funktionen $V = f(\alpha_2)$. Orsaken till denna avvikelse härrör sig säkerligen från avrundningsfel i de numeriska beräkningarna.

Om samma element (A(1,1) och A(2,2)) släpps loss i system 1, visar det sig, att nivåkurvorna får en approximativt cirkulär form. Se fig 7.3!

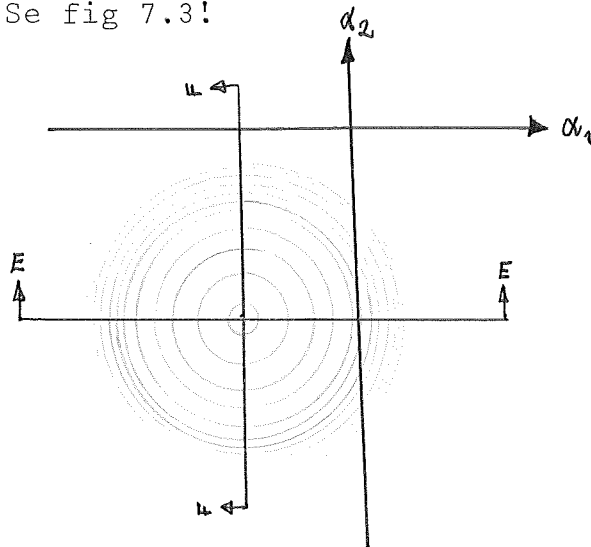


Fig 7.3. - Visar utseendet av nivåkurvorna när A(1,1) och A(2,2) "släpps loss" i system 1. Snitten E-E och F-F finns uppritade i fig. 7.4.

Av symmetriskäl följer då att konvergensen är mindre känslig för valet av startpunkt och att denna överhuvudtaget underlättas. Vi tror, att det är skillnaden i förlustfunktionernas topografi som är orsaken till att system 2 visat sig vara besvärligare ur identifieringssynpunkt än system 1. Se även anmärkningen sid 69!

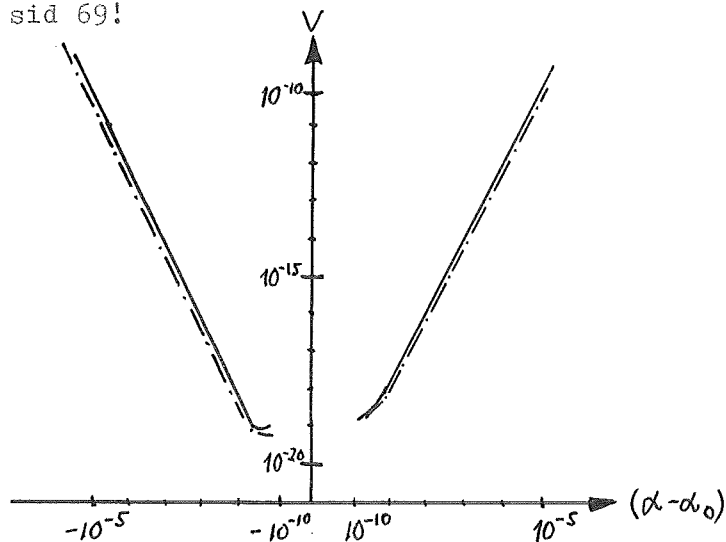


Fig 7.4.

Eftersom matris H inverkar på stegningen, går denna ej alltid i gradientens motsatta riktning. (Detta inträffar i själva verket endast i första steget). FLEPOMIN varierar även värdet på β . Därför sker ibland olyckligtvis ett hopp in i det instabila området. Om då värdet hos V blir alltför stort, avbryter maskinen exekveringen. Någon form av stabilitetstest vore därför på sin plats i FLEPOMIN.

Efter ett hopp ut i det instabila området kan man mycket väl göra omstart från det näst sista α -värdet vid en senare körning. Se tab. 6.32! Instabilitetsproblemet blir särskilt påtagligt om stabilitetsgränsen ligger i närheten av "operationsområdet".

Hur många parametrar kan max. identifieras?

Detta är den intressantaste frågan, och svaret torde vara en mätare på metodens duglighet. Ett generellt svar kan emellertid ej ges.

Teoretiskt är det möjligt att i system 1 identifiera max 12 parametrar. Det har lyckats oss att i detta system identifiera max 7 parametrar med ett fel först i 10:e decimalen. Om vi ökar antalet parametrar ytterligare, kan vi ej garantera konvergens för samtliga parametrar. Däremot kan vi beräkna en korrekt överföringsfunktion med hjälp av de identifierade parametrarna. Detta har vi gjort i fallet med 9 identifierade parametrar.

Förlustfunktionens värde var denna gång av en sådan storleksordning, att vi har anledning att misstänka ett minimum, trots att vissa parametrar fått felaktigt värde i identifieringen. V kan alltså minimeras för ett speciellt värde på två koplade parametrar.

Om B och C är givna, har vi med framgång kunnat identifiera 5 obekanta parametrar i A (system 1). Försöket med 7 obekanta (tab. 6.14) tyder på att det är möjligt att identifiera 7 obekanta parametrar, även om detta fordrar många iterationer.

Frusna parametrar.

De försök som gjorts där någon parameter fixerats till ett från den fysikaliska modellen avvikande värde (den har "frusits") har visat svårigheter vid identifieringen. I något fall lyckas vi identifiera parametrar med fel först i andra decimalen. Om sedan samma försök göres med annat startvärde får vi mycket dålig konvergens.

Det är emellertid anmärkningsvärt, att vi, om vi fryser karakteristiska ekvationens koefficienter (se tab. 6.33), kan göra en god identifiering av matris B i system 2.

Försöken har inte givit några entydiga resultat eller implikationer; möjligtvis ger de en svag antydning, om att en frusen parameter ställer till med de största svårigheterna inom den egna matrisen.

Testkörningarna med STRUCNL och FGCLI.

Som vi sett ovan, fungerar metoden bra om vi har en linjär modell och tillämpar algoritmerna för det linjära specialfallet. Då har vi bl.a. kunnat utföra direkt integration av såväl systemekvationerna som λ -funktionerna.

Dessa båda integrationer måste utföras på annat sätt när det gäller olinjära modeller. Systemekvationen integreras utan alltför avgörande fel med en rutin typ Runge-Kutta. En jämförelse av förlustfunktionens värde mellan körningarna 1 och 2 i tab. 6.35 visar detta. En jämförelse mellan 1 och 2 visar också att påverkan på gradienten är obetydlig om integrationssteget i Runge-Kutta göres mindre.

Större problem erbjuder tydligen integrationen av λ -ekvationerna. En jämförelse mellan körningarna 2 och 3 i tab. 6.35 visar, att en relativt stor förbättring inträffar om tätare evalueringar av λ göres. Dessa evalueringar tar emellertid mycket lång tid att utföra. Jämför tiderna och antalet medhanna iterationer mellan 2 och 3 i tab. 6.35! I körning 3 har λ evaluerats 2,5 ggr tätare än i körning 2, och i körning 2 hann datamaskinen med 3 fullständiga iterationssteg mot 1 i körning 3.

Dilemmats förklaring är enkel. Vi har i algoritmen ersatt integralen med en summa. Se fig. 7.5!

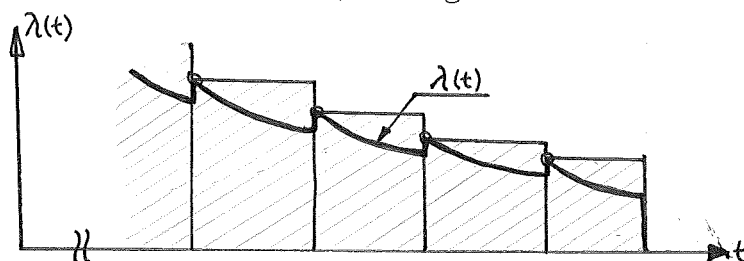


Fig. 7.5. Den streckade ytan visar approximationen av $\int \lambda(t) dt$

Vid tätare evaluering blir överensstämmelsen mellan summa och integral större. En tänkbar utväg att slippa en alltför tät och tidskrävande λ -evaluering är följande approximation av integralen. Se fig. 7.6!

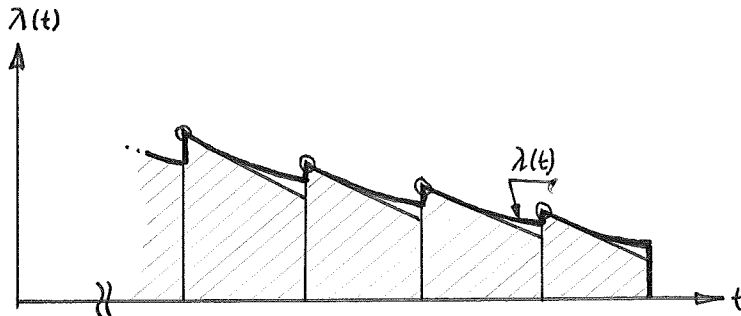


Fig 7.6.

Eftersom λ -ekvationens derivata är tillgänglig i de inringade punkterna kan λ -ekvationen lineariseras från dessa punkter och sedan lätt integreras.

Det är en optimeringsfråga, hur långt man skall driva exaktheten hos denna integration av λ -ekvationen. En exakt integral ger en exakt gradient i varje iterationssteg, och därmed fordras färre antal iterationer, men detta är till priset av mycket lång maskintid/iterationssteg. Försöket i tabell 6.35 körning 1 visar, att α konvergerar trots ett något felaktigt värde på gradienten i varje steg.

Generell slutsats.

Våra testkörningar med linjära system visar, att algoritmerna fungerar och är användbara. De visar även att metoden tillvaratager all a prioriinformation, och att konvergensen blir säkrare ju riktigare och pålitligare denna a prioriinformation är.

Anmärkning

En annan tänkbar förklaring till våra identifieringssvårigheter i system 2 kan vara följande:

Tumregel_för_val_av_PRBS_data: PRBS-signalen skall väljas sådan, att

$N \times (\text{samplingsintervallets längd}) = (c:a 5) \times (\text{största tidskonstant})$

Detta medför, att N i vårt fall bör vara minst c:a 50, men av diagram 6.4 framgår att $N = c:a 31$. De data som vi haft till förfogande har alltså ej uppfyllt de villkor som identifieringsmetoden fordrar. I system 1 är däremot $N = 127$.

8. REFERENSER

- 1 Eklund, Karl: Identifiering av parametrar i system på tillståndsform. Intern rapport. Inst. reglerteknik LTH 1969.
- 2 Gelfand-Fomin: Calculus of variations. Prentice-Hall Inc. 1963.
- 3 Rosenbrock, H.H.: On the design of linear multi-variable control systems. IFAC-papper. Session 1. Paper 1A.
- 4 Åström, K.J.: Lectures on the identification problem - The least squares method. Kompendium. Inst. reglerteknik LTH 1968.
- 5 Åström K.J.: Computational Aspects of a Class of Identification Problems. Technical Report No 69-14. Dept. och Elect. Eng. USC. Oct. 1969

9. APPENDIX

Programlistor

PROGRAM STRUC

PROGRAM STRUCNL

FTN5.5B

PROGRAM STRUC

C

```

DIMENSION A(10,10),B(10,5),C(5,10),D(5,5),IAD(10,10),IBD(10,10),IC
10(5,10),IDD(5,10),YM(5,500),UM(5,500),ALFA(10),XI(10),H(55)
COMMON A,B,C,D,IAD,IBD,ICD,IDD,YM,UM,XI,N,NU,NY,MENU,NUPA,TSAMP
INTEGER CONV
EXTERNAL FGLI
READ 1,N,NU,NY,MENU,NUPA,TSAMP
1 FORMAT(3I2,I3,I2,F10.5)
READ 2,((A(I,J),J=1,N),I=1,N)
2 FORMAT(4E20,10)
READ 2,((B(I,J),J=1,NU),I=1,N)
READ 2,((C(I,J),J=1,N),I=1,NY)
READ 2,((D(I,J),J=1,NU),I=1,NY)
READ 3,((IAD(I,J),J=1,NUPA),I=1,N)
3 FORMAT(40I2)
READ 3,((IBD(I,J),J=1,NUPA),I=1,N)
READ 3,((ICD(I,J),J=1,NUPA),I=1,NY)
READ 3,((IDD(I,J),J=1,NUPA),I=1,NY)
READ 2,(XI(I),I=1,N)
READ 2,(ALFA(I),I=1,NUPA)
CALL DATA
READ 30,EPS,LIMIT
30 FORMAT(E20,10,I3)
READ 50,LOADH
50 FORMAT(I1)
IF(LOADH) 51,51,52
51 K=NUPA*(NUPA+1)/2
READ 2,(H(I),I=1,K)
52 CONTINUE
PRINT 4
4 FORMAT(/,9H MATRIX A,/)
DO 5 I=1,N
5 PRINT 6,(A(I,J),J=1,N)
6 FORMAT(6E20,10)
PRINT 7
7 FORMAT(//,9H MATRIX B,/)
DO 8 I=1,N
8 PRINT 6,(B(I,J),J=1,NU)
PRINT 9
9 FORMAT(//,9H MATRIX C,/)
DO 10 I=1,NY
10 PRINT 6,(C(I,J),J=1,N)
PRINT 62
62 FORMAT(/,9H MATRIX D,/)
DO 63 I=1,NY
63 PRINT 6,(D(I,J),J=1,NU)
PRINT 11
11 FORMAT(//,11H MATRIX IAD,/)
DO 12 I=1,N
12 PRINT 13,(IAD(I,J),J=1,NUPA)
13 FORMAT(40I3)
PRINT 14
14 FORMAT(//,11H MATRIX IBD,/)
DO 15 I=1,N
15 PRINT 13,(IBD(I,J),J=1,NUPA)

```

```
PRINT 16
16 FORMAT(//,11H MATRIX ICD,/)
DO 17 I=1,NY
17 PRINT 13,(ICD(I,J),J=1,NUPA)
PRINT 60
60 FORMAT(//,11H MATRIX IDD,/)
DO 61 I=1,NY
61 PRINT 13,(IDD(I,J),J=1,NUPA)
PRINT 18
18 FORMAT(//,10H VECTOR X1,/)
DO 19 I=1,N
19 PRINT 6,XI(I)
PRINT 20
20 FORMAT(//,12H VECTOR ALFA,/)
DO 21 I=1,NUPA
21 PRINT 6,ALFA(I)
PRINT 31,EPS
31 FORMAT(//,5H EPS=,E20.10)
PRINT 32,LIMIT
32 FORMAT(//,7H LIMIT=,I3)
PRINT 33,TSAMP
33 FORMAT(//,7H TSAMP=,E20.10)
IF(LOADH) 53,53,54
53 PRINT 500
500 FORMAT(//,47H INITIAL GUESS OF INVERSE OF SECOND DERIVATIVES,/)
PRINT 6,(H(I),I=1,K)
54 K1=MSLEFT(XMS)
C
CALL FLEPOMIN(NUPA,ALFA,F,EPS,CONV,LIMIT,H,LOADH,FGLI)
C
K2=MSLEFT(XMS)
K=K1-K2
PRINT 55,K
55 FORMAT(//,17H TIME IN FLEPOMIN,I6.3H MS)
IF(CONV) 35,35,34
35 LOADH=1
C
CALL FLEPOMIN(NUPA,ALFA,F,EPS,CONV,LIMIT,H,LOADH,FGLI)
C
PRINT 36
36 FORMAT(//,33H FLEPOMIN RESTARTED AFTER FAILURE,/)
34 PRINT 37
37 FORMAT(//,31H ESTIMATED PARAMETERVECTOR ALFA,/)
DO 38 I=1,NUPA
38 PRINT 6,ALFA(I)
PRINT 39,F
39 FORMAT(//,31H MINIMUM VALUE OF LOSS FUNCTION,5X,E20.10)
PRINT 40
40 FORMAT(//,47H ESTIMATED INVERSE MATRIX OF SECOND DERIVATIVES,/)
K=NUPA*(NUPA+1)/2
PRINT 6,(H(I),I=1,K)
CALL EXIT
END
```

```
PROGRAM STRUCNL
DIMENSION YM(5,500), UM(5,500), ALFA(10), XI(10), H(55),
*W(10),Z(10)
COMMON UM,YM,XI,MENII,NX,NUPA,NP,NXS,TSAMP,PSAMP,XSAMP,NU,NY,JS
COMMON /FUNCTION/ TE, W, Z
INTEGER CONV
EXTERNAL FGCLI
READ 1, NX, NU, NY, NUPA, MENII, TSAMP, XSAMP, PSAMP
1 FORMAT (4I2, I3, 3F10.5)
READ 2, (ALFA(I), I=1,NUPA)
2 FORMAT (4E20.10)
READ 2, (XI(I), I=1,NX)
CALL GENO
READ 3, EPS, LIMIT, LOADH
3 FORMAT (E20.10, I3, I1)
IF (LOADH) 4,4,5
4 K=NUPA*(NUPA+1)/2
READ 2, (H(I), I=1,K)
5 CONTINUE
CALL SKRIV
PRINT 8
8 FORMAT (///, 10H VECTOR XI,/, 1X, 9(1H*),//)
DO 9 I=1,NX
9 PRINT 10, XI(I)
10 FORMAT (6E20.10)
PRINT 11
11 FORMAT(///, 12H VECTOR ALFA,/, 1X, 11(1H*),//)
DO 12 I=1,NUPA
12 PRINT 10, ALFA(I)
PRINT 13, EPS
13 FORMAT(///, 7H EPS =, E20.10)
PRINT 14, LIMIT
14 FORMAT (///, 11H LIMIT= , I3)
PRINT 15, TSAMP
15 FORMAT (///, 7H TSAMP=, E20.10)
PRINT 16, XSAMP
16 FORMAT (///, 7H XSAMP=, E20.10)
PRINT 17, PSAMP
17 FORMAT (///, 7H PSAMP=, E20.10)
IF (LOADH) 18, 18, 20
18 PRINT 19
19 FORMAT (///, 47H INITIAL GUESS OF INVERSE OF SECOND DERIVATIVES,/)
PRINT 10, (H(I), I=1,K)
20 K1=MSLEFT(XMS)
NP=INT(TSAMP/PSAMP+0.5)
NXS=INT(PSAMP/XSAMP+0.5)
CALL FLEPOMIN (NUPA,ALFA,F,EPS,CONV,LIMIT,H,LOADH,FGCLI)
K2=MSLEFT(XMS)
K=K1-K2
PRINT 21,K
21 FORMAT (///, 17H TIME IN FLEPOMIN, I6, 3H MS)
IF (CONV) 22, 22, 231
22 LOADH=1
CALL FLEPOMIN (NUPA,ALFA,F,EPS,CONV,LIMIT,H,LOADH,FGCLI)
PRINT 23
23 FORMAT (///, 33H FLEPOMIN RESTARTED AFTER FAILURE,/)

```

FTN5.5B

20/03-70

```
231 PRINT 24
24  FORMAT (//, 31H ESTIMATED PARAMETERVECTOR ALFA,/)
    DO 25 I=1,NUPA
25  PRINT 10, ALFA(I)
    PRINT 26, F
26  FORMAT (//, 31H MINIMUM VALUE OF LOSS FUNCTION,5X, E20.10)
    PRINT 27
27  FORMAT (//,47H ESTIMATED INVERSE MATRIX OF SECOND DERIVATIVES,/)
    K=NUPA*(NUPA+1)/2
    PRINT 10, (H(I), I=1,K)
    CALL EXIT
    END
```