UNDERSÖKNING AV OPTIMALA SYSTEM

TORSTEN PÅLSSON

Rapport RE - 38 okt. 1968

UNDERSÖKNING AV OPTIMALA SYSTEM.

Examensarbete i Reglerteknik

av T. Pålsson

REFERENSER

(1) K. Mårtensson: Linear quadratic control package Part I - The continuous problem. RE 6802 Institutionen för Regleringsteknik LTH.

(2) K.J. Åström: Kompendium i Reglerteknik LTH.

(3) R. E. Kalman and T. S. Englar: A users manual for the automatic synthesis program. Chapter XI.

(4) R. E. Kalman: When is a linear control system optimal? NASA Report nr CR 475.

(5) A. R. M. Noton: Introduction to variational methods in control engineering.

(6) J. S. Tyler and F. B. Tuteur: The use of a quadratic performance index to design multivariable control systems. IEEE January 66.

(7) K. Mårtensson: To appear.

INLEDNING

Uppgiften består i att studera det så kallade linjär-kvadratiska optimeringsproblemet, och då speciellt att undersöka det stationära slutna systemet med avseende på dess egenvärden. För detta ändamål har utnyttjats en tidigare rapport från institutionen av K. Mårtensson (Ref. 1). Denna innehåller bl. a. en komplett programuppsättning för beräkning av den optimala styrlagen, dels med Runge-Kuttas metod och dels med hjälp av exponentialserier för den kanoniska ekvationen. Den senare metoden har här använts för att erhålla den stationära styrlagen, varefter det slutna systemets egenvärden har beräknats. Vid programmeringen har använts de tidigare skrivna subrutinerna.

Först undersöktes några mindre system vars styrlagar kan beräknas för hand, varefter några sjätte ordningens system med en insignal undersökts. Härvid har även något om observerbarhetens inverkan studerats. En artikel av Kalman (Ref3) ligger här till grund för beräkningarna. Av flervariabla system har beräknats ett modellföljningsystem för ett flygplan. Vidare anges en metod att beräkna egenvärdena till det optimala systemet utgående från karakteristiska ekvationen till Eulermatrisen.

2. PROBLEMETS LYDELSE

Betrakta ett linjärt tidsinvariant dynamiskt system givet av ekvationen

$$\frac{dx(t)}{dt} = A \cdot x(t) + B \cdot u(t) \qquad (2:1)$$

där $x(t)$ är en n-dimensionell tillståndsvektor, $u(t)$ en r-dimensionell vektor av insignaler, A en n x n matris och B en n x r matris.
Bilda den så kallade förlustfunktionen

$$V(u) = \tfrac{1}{2}\left\{x^T(t_1)\, Q_0\, x(t_1)\right\} + \tfrac{1}{2}\int_{t_0}^{t_1}\left\{x^T(s)\, Q_1\, x(s) + u^T(s)\, Q_2\, u(s)\right\} ds \qquad (2:2)$$

där t_0 och t_1 är givna tidpunkter. Vi antager att Q_0 och Q_1 är symmetriska positivt semidefinita matriser samt Q_2 är symmetrisk och positivt definit.

Uppgiften består nu i att bestämma en styrlag till systemet (2:1) sådan att förlustfunktionen (2:2) blir så liten som möjligt. Med hjälp av variationskalkyl erhålles två olika metoder att lösa problemet. Ref.(1) ger en redogörelse för dessa, Euler-Lagranges metod och Hamilton-Jacobis metod, varför här endast följer en resume´ därav.

Lösning av det linjär-kvadratiska optimeringsproblemet.

A. Euler-Lagranges metod

Denna lösning erhålles genom att undersöka variationen i förlustfunktionen i närheten av den optimala lösningskurvan. Genom att minimera Hamiltonfunktionen med avseende på u

$$2\mathcal{H}(x,p,u) = x^T\, Q_1\, x + u^T\, Q_2\, u + 2p^T(Ax + Bu)$$

där p är Lagranges multiplikator, erhålles

$$u = -\, Q_2^{-1} B^T p$$

De kanoniska ekvationerna blir

$$\frac{dx}{dt} = \dot{x}p = Ax - B\,Q_2^{-1}B^T p$$

$$\frac{dp}{dt} = -\dot{x}x = -\,Q_1 x - A^T p$$

med randvillkoren givna vid t_0 och t_1.

Inför 2n x 2n matrisen $\sum(t:t_1)$ som är fundamentalmatris till de kanoniska ekvationerna. Då gäller

$$\frac{d}{dt}\sum(t:t_1) = \begin{pmatrix} A & -BQ_2^{-1}B^T \\ -Q_1 & -A^T \end{pmatrix} \sum(t:t_1)$$

Dela upp $(t:t_1)$ i fyra undermatriser på följande sätt

$$\sum(t:t_1) = \begin{pmatrix} \sum_{11}(t:t_1) & \sum_{12}(t:t_1) \\ \sum_{21}(t:t_1) & \sum_{22}(t:t_1) \end{pmatrix}$$

Nu observeras att de kanoniska ekvationerna är linjära varför man kan erhålla

$$p(t) = S(t)\cdot x(t)$$

där

$$S(t) = (\sum_{21}(t:t_1) + \sum_{22}(t:t_1)Q_0)(\sum_{11}(t:t_1) + \sum_{12}(t:t_1)Q_0)^{-1}$$

och S symmetrisk. Styrlagen ges nu av

$$u(t) = -\,L(t)\cdot x(t)$$

med

$$L(t) = Q_2^{-1}B\,S(t)$$

B. Hamilton-Jacobis metod

Inför funktionalen

$$V(x,t) = \min_u \left\{ \tfrac{1}{2}\, x^T(t_1)Q_0x(t_1) + \tfrac{1}{2}\int_t^{t_1} \left(x^T(s)Q_1x(s) + u^T(s)Q_2u(s)\right)\right\} ds$$

På samma sätt som i A minimeras Hamiltonfunktionen. Funktionalen V(x,t) skall satisfiera Hamilton-Jacobi partiella differentialekvation. För att lösa denna fås av randvillkoret ansatsen

$$V(x,t) = \tfrac{1}{2}\, x^T(t)S(t)x(t)$$

där S är en n x n matris, som antages vara symmetrisk och positivt semidefinit. Genom att pröva ansatsen fås som villkor på matrisen S att den skall satisfiera differentialekvationen

$$\frac{dS}{dt} + A^TS + SA - S^TBQ_2^{-1}B^TS + Q_1 = 0$$

med randvillkoret givet vid sluttidpunkten t_1

$$S(t_1) = Q_0$$

Denna matrisdifferentialekvation innehåller n^2 icke-linjära differentialekvationer och är av s.k. Riccati-typ.Då S är symmetrisk reduceras antalet differentialekvationer från n^2 till $n(n+1)/2$.

Då $p=\text{grad}_x V^T$ blir den optimala styrlagen

$$u(t) = - Q_2^{-1}B^TS(t)$$

eller

$$u(t) = - L(t)\cdot x(t)$$

<u>Det slutna optimala systemet</u>.

Betrakta ett linjärt tidsinvariant dynamiskt system givet av ekvationen

$$\dot{x} = Ax + Bu$$

med förlustfunktionen

$$V(u) = \tfrac{1}{2}\, x^TQ_0x + \tfrac{1}{2}\int_0^T (x^TQ_1x + u^TQ_2u)\, dt$$

Låt nu $T \to \infty$ vilket innebär att vi erhåller ett stationärt värde på styrlagen $u = -L \cdot x$.
Det slutna systemet ges då av ekvationen

$$\dot{x} = (A - BL) \cdot x$$

Det förutsättes att samtliga tillståndsvariabler finns tillgängliga för återkopplingen. Uppgiften består nu i att undersöka det slutna systemets egenvärden då elementen i viktmatriserna i förlustfunktionen varieras. För ändamålet har skrivits ett program för beräkning av den optimala styrlagens stationära värde samt det slutna systemets egenvärden.

Program BUTTER

I ref.(1) finns två kompletta programuppsättningar för lösning av det linjär-kvadratiska optimeringsproblemet, dels med Euler-Lagranges metod, program LIOPCON och med Hamilton-Jacobis metod, program RKRICCE. Här har valts den förra lösningsmetoden och de i program LIOPCON ingående subrutinerna har direkt använts. För egenvärdesberäkningen har utnyttjats en subrutin, EIGUNS.

Ingångsstorheterna till program BUTTER är:

N - systemets ordning
NU - antalet insignaler
TIMEDIF - tiden mellan de ekvidistanta punkter i vilka S-matrisen beräknas.
ITER - ITER=0 betyder att fundamentalmatrisen beräknas för varje steg. ITER=1 betyder att fundamentalmatrisen beräknas endast i första steget. (Se subrutin RICCE)
EPS - För att erhålla det stationära värdet beräknas normen av skillnaden mellan två konsekutiva S-matriser. Då denna skillnad är mindre än EPS har det stationära värdet uppnåtts.
IMAX - Om inte S-matrisen konvergerat efter IMAX punkter avbryts beräkningen.

Programmet ger möjlighet att under samma körning beräkna det optimala systemet för olika värden på Q_0, Q_1, Q_2 och EPS. Dock kan endast ett system köras åt gången.
Utskrift sker av systemets parametrar, den beräknade S-matrisen, L-matrisen, matrisen (A-BL) samt egenvärdena till denna.
Se vidare listan.

3. SYSTEM MED EN INSIGNAL

Exempel

Betrakta systemet $\frac{dx}{dt} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$ (Dubbelintegratorn)

Med förlustfunktionen: $V(u) = \int_0^T (x^T Q_1 x + u^T Q_2 u)\, dt$

där $Q_1 = \begin{bmatrix} q_{11} & 0 \\ 0 & q_{22} \end{bmatrix}$ och $Q_2 = r$

Som lösning till Riccatiekvationen ansättes matrisen

$$S = \begin{bmatrix} s_{11} & s_{12} \\ s_{12} & s_{22} \end{bmatrix}$$

Stort värde på $T \Rightarrow \frac{dS}{dt} = 0$. Genom insättning i stationära Riccatiekvationen (SR) fås ett olinjärt ekvationssystem varur elementen i S-matrisen kan lösas.

$s_{11} = \sqrt{q_{11}(2\sqrt{q_{11} r} + q_{22})}$

$s_{12} = \sqrt{q_{11} r}$

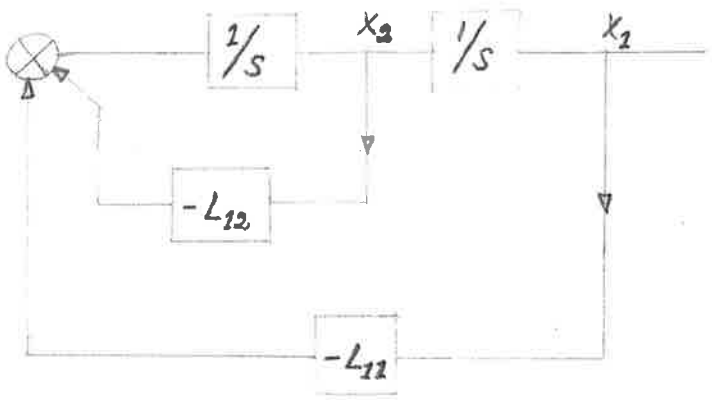
$s_{22} = \sqrt{r(2\sqrt{q_{11} r} + q_{22})}$

Det finns endast en positivt definit lösning till SR.

Återkopplingsmatrisen L blir

$$L = \left[\sqrt{q_{11} r^{-1}},\ \sqrt{r^{-1}(2 q_{11} r + q_{22})} \right]$$

Det optimala systemet får då följande utseende



Det optimala systemet blir

$$\mathbf{x} = \begin{bmatrix} 0 & 1 \\ -\sqrt{\frac{q_{11}}{r}} & -\sqrt{\frac{2\sqrt{q_{11}r} + q_{22}}{r}} \end{bmatrix} \mathbf{x}$$

Dess egenvärden ges av

$$\det(\mathbf{A}-\mathbf{BL}-s\mathbf{I}) = s^2 + \sqrt{\frac{2\sqrt{q_{11}r} + q_{22}}{r}}\, s + \sqrt{\frac{q_{11}}{r}} = 0$$

Undersök nu hur det optimala systemets egenvärden varierar då parametrarna q_{11}, q_{22} och r varieras.

a/ Sätt $q_{11} = q_{22} = 1$, variera r $(r>0)$

$$s_{1,2} = -\sqrt{\frac{2\sqrt{r} + 1}{4r}} \pm \sqrt{\frac{1 - 2\sqrt{r}}{4r}}$$ Se figur 1a.

b/ Sätt $q_{11} = 1$, $q_{22} = 0$, variera r

$$s_{1,2} = -\sqrt{\frac{\sqrt{r}}{2r}} \pm j\sqrt{\frac{\sqrt{r}}{2r}}$$ Se figur 1b.

c/ Sätt $q_{22} = r = 1$, variera q_{11} $(q_{11} \geq 0)$

$$s_{1,2} = -\frac{\sqrt{2\sqrt{q_{11}} + 1}}{2} \pm \frac{\sqrt{1 - 2\sqrt{q_{11}}}}{2}$$ Se figur 1c.

d/ Sätt $q_{11} = r = 1$, variera q_{22} $(q_{22} \geq 0)$

$$s_{1,2} = -\frac{\sqrt{2 + q_{22}}}{2} \pm \frac{\sqrt{q_{22} - 2}}{2}$$ Se figur 1d.

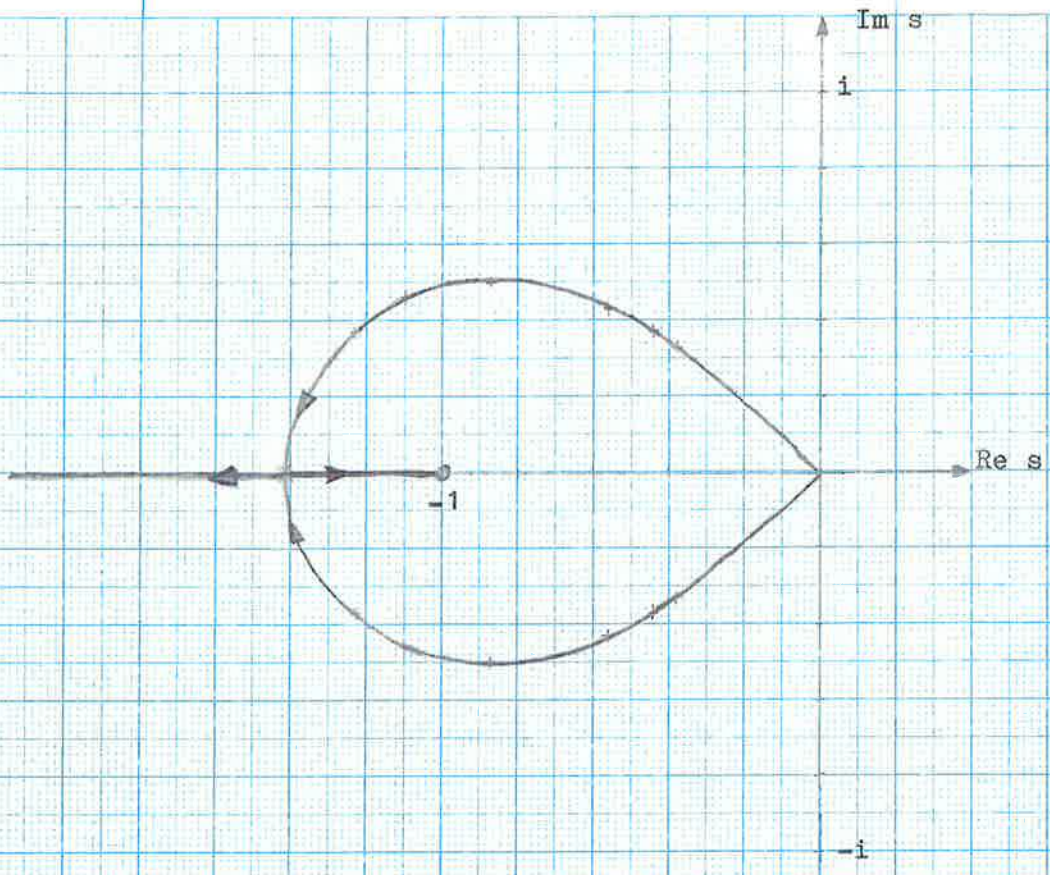
e/ Sätt $q_{11} = 0$, $q_{22} = q_{22}$, variera r

$$s_{1,2} = -\sqrt{\frac{q_{22}}{4r}} \pm \sqrt{\frac{q_{22}}{4r}}$$

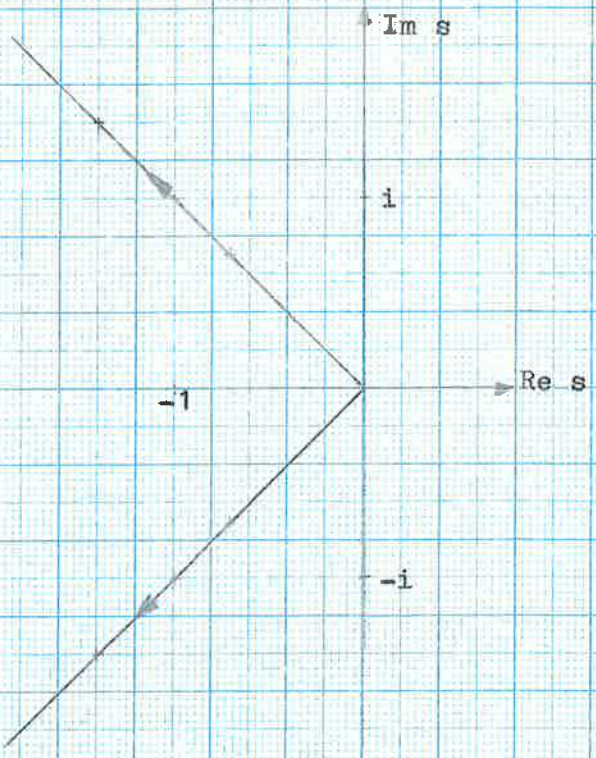
dvs ett egenvärde i noll och ett längs negativa reella axeln. Styrlagen är således inte asymptotiskt stabil. Lösningen till SR blir för detta fall

$$\mathbf{S} = \begin{bmatrix} 0 & 0 \\ 0 & \sqrt{rq_{22}} \end{bmatrix}$$ positivt semidefinit

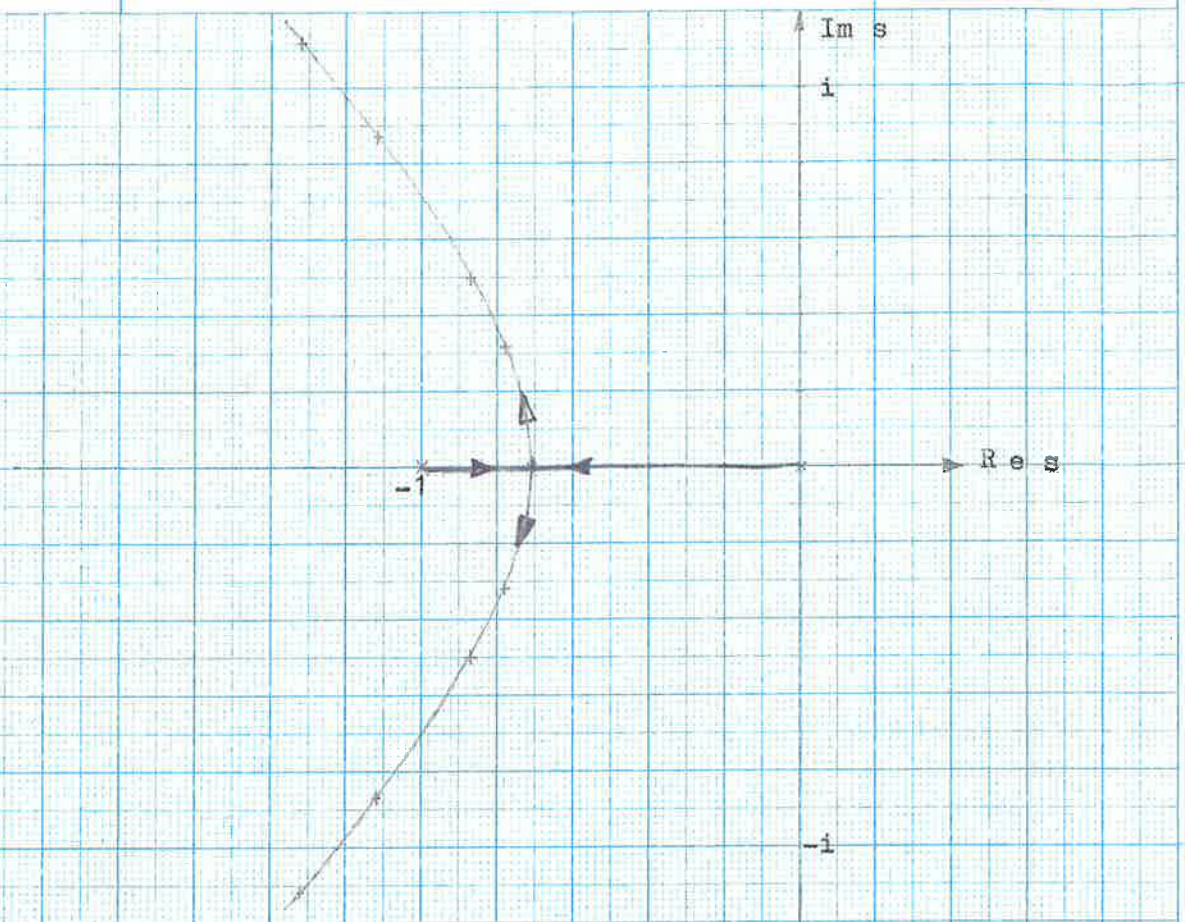
Rang $(\mathbf{A}\ \mathbf{Q}_1) = 1$ dvs systemet är ej observerbart.



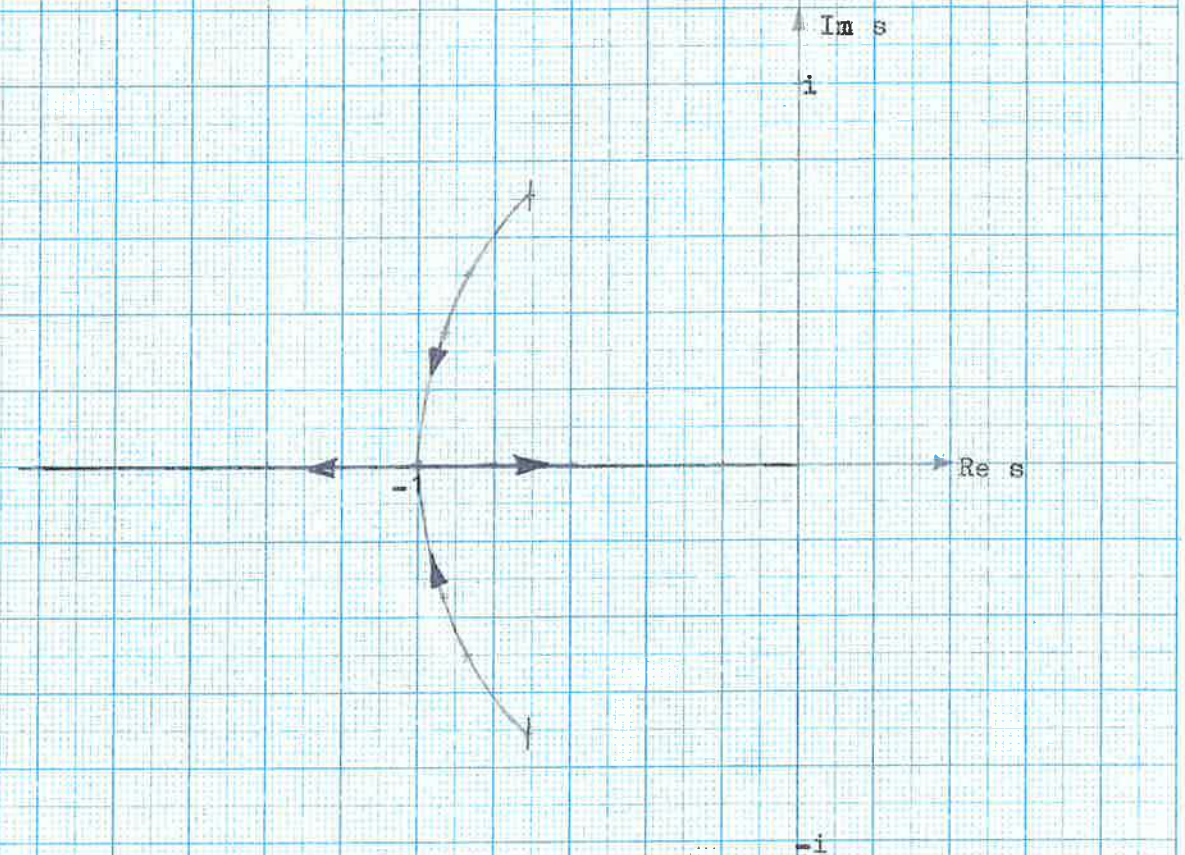
Figur 1a. Rotorten då $q_{11} = q_{22} = 1$ och r varieras (r minskas).



Figur 1b. Rotorten då $q_{11}=1$, $q_{22}=0$ och r varieras (r minskas).



Figur 1c. Rotorten då $q_{22} = r = 1$ och q_{11} varieras (q_{11} ökas).



Figur 1d. Rotorten då $q_{11} = r = 1$ och q_{22} varieras (q_{22} ökas)

Något om observerbarhetens inverkan på styrlagen.

Betrakta systemet: $\dot{x} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$

$y = [1,-1]\, x$

Överföringsfunktionen: $G(s) = \frac{1-s}{s^2-1}$

Välj som förlustfunktion: $V(u) = \int_0^\infty (y^T y + u^T u)\, dt$

Matrisen A har egenvärdena +1 och -1, dvs det öppna systemet är instabilt.

Rang $\begin{bmatrix} C \\ CA \end{bmatrix} = 1$, dvs systemet är ej observerbart, och det icke-observerbara egenvärdet har positiv realdel. Med r=1 erhålles följande lösningar till SR.

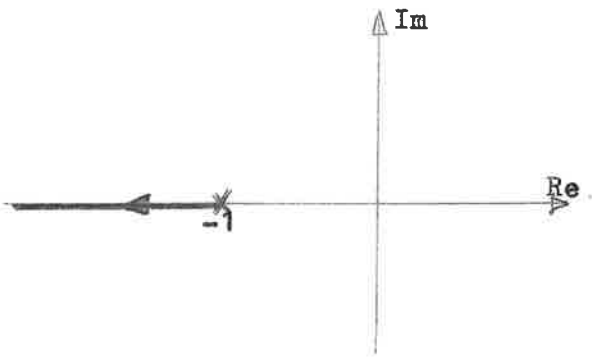
$S_1 = \begin{bmatrix} 3+\sqrt{2} & 1+\sqrt{2} \\ 1+\sqrt{2} & 1+\sqrt{2} \end{bmatrix}$ positivt definit

$S_2 = \begin{bmatrix} \sqrt{2}-1 & -\sqrt{2}+1 \\ -\sqrt{2}+1 & \sqrt{2}-1 \end{bmatrix}$ positivt semidefinit

Förlustfunktionen blir $V(u) = \int_0^\infty (x_1 - x_2)^2 + u^2\, dt$

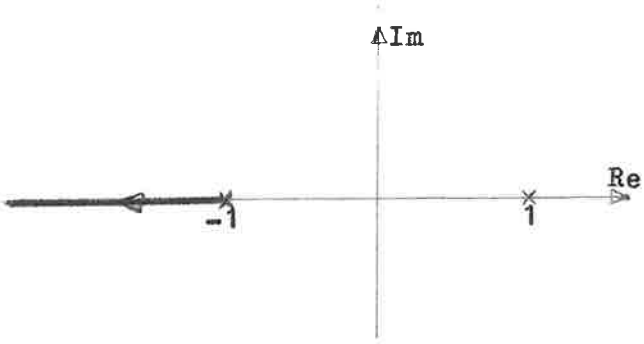
Med $x_1 = x_2$ blir $V(u)=0$ för $u = 0$. Detta motsvarar lösningen S_2 till SR, och detta är alltså den optimala lösningen till problemet. Det slutna systemets egenvärden beräknas då r varierar.

Med lösningen S_1 erhålles:



Ett egenvärde ligger kvar i -1.

Med lösningen S_2 erhålles:



Ett egenvärde ligger kvar i +1.

Härav framgår att det optimala systemet, svarande mot lösningen S_2 till SR, är instabilt.

Undersökning av det optimala systemets egenvärden då $Q_2 \to 0$.

Betrakta ett system med överföringsfunktionen

$$G(s) = \frac{b_m s^m + b_{m-1} s^{m-1} + \cdots + b_1 s + b_0}{s^n + a_{n-1} s^{n-1} + \cdots + a_1 s + a_0}$$

Systemet skrivet på kontrollerbar kanonisk form

$$\dot{x} = Ax + Bu$$

$$y = Cx$$

där

$$A = \begin{bmatrix} 0 & 1 & \cdot & \cdot & \cdot & \cdot & 0 \\ 0 & 0 & 1 & \cdot & \cdot & \cdot & 0 \\ \cdot & & & 1 & & & \cdot \\ \cdot & & & & \cdot & & \cdot \\ \cdot & & & & & \cdot & \cdot \\ -a_0 & \cdot & \cdot & \cdot & \cdot & \cdot & -a_{n-1} \end{bmatrix} \qquad B = \begin{bmatrix} 0 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 1 \end{bmatrix}$$

$$C = \begin{bmatrix} b_0, & b_1, & \cdot\ \cdot\ b_m, & 0, & \cdot\ \cdot\ 0 \end{bmatrix}$$

Välj som förlustfunktion $V(u) = \int_0^\infty (y^T y + u^T u)\, dt$ och undersök egenvärdena till det optimala systemet A - BL då $r \to 0$.

Ref.(4) ger: Då $r \to 0$ kommer m av det slutna systemets poler att gå till de m nollställena till G(s) i vänstra halvplanet (eller deras spegelbild med avseende på imaginära axeln, om några av nollställena är i högra halvplanet) och de andra n - m polerna kommer att konvergera till ett "Butterworth"-mönster med radien

$$\left(\frac{a_0^2 b_m^2}{b_0^2} + \frac{b_m^2}{r} \right)^{\frac{1}{2(n-m)}}$$

Enligt Kalman skall observerbarheten inte ha någon inverkan på resultatets giltighet. Med exempel skall här visas att så inte är fallet.

Som exempel har valts följande fyra överföringsfunktioner

$$G(s) = \frac{1}{s^6 + s^5 + 0.25s^4 - 0.5s^3 + 1.25s^2 - 0.5s - 2.5} = \frac{1}{N} \qquad (A)$$

$$G(s) = \frac{1 - s^2}{N} \qquad (B)$$

$$G(s) = \frac{1 - s + 0.5s^2}{N} \qquad (C)$$

$$G(s) = \frac{1 + s + 0.5s^2}{N} \qquad (D)$$

med polerna ± 1, -1 ± i, $\frac{1}{2}$ ± i, och nollställena (B) ± 1, (C) 1 ± i, och (D) -1 ± i.

Systemen skrivna på kontrollerbar kanonisk form.

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 2.5 & 0.5 & -1.25 & 0.5 & -0.25 & -1.0 \end{bmatrix} \qquad B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

$$C_A = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$C_B = \begin{bmatrix} 1 & 0 & -1 & 0 & 0 & 0 \end{bmatrix}$$

$$C_C = \begin{bmatrix} 1 & -1 & 0.5 & 0 & 0 & 0 \end{bmatrix}$$

$$C_D = \begin{bmatrix} 1 & 1 & 0.5 & 0 & 0 & 0 \end{bmatrix}$$

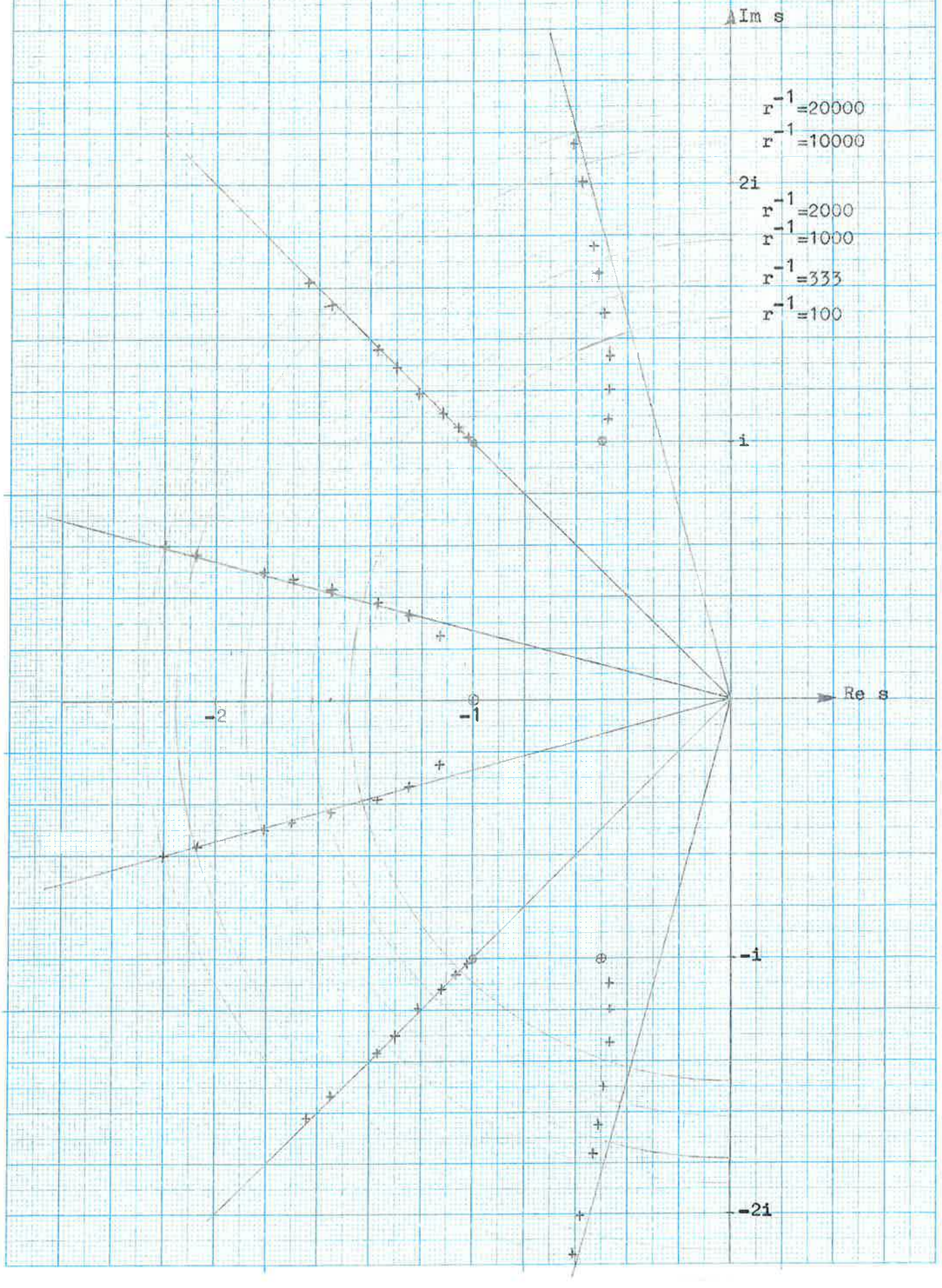
Dessa system har körts med program BUTTER. Därvid observeras att förlustfunktionen innehåller termen $y^T y$, men $y = Cx$ dvs $Q_1 = C^T C$.

Betrakta först systemen (A) och (C), som båda är observerbara. Resultatet framgår av figur 2 och 3. Då $r \to \infty$, vilket motsvaras av att förstärkningen i återkopplingsslingan går mot noll, är det slutna systemets egenvärden de samma som det öppna systemets egenvärden med negativ realdel. I detta fall är de -1, -1, -1 ± i, -1/2 ± i. De räta linjerna är asymptoter till rötterna.

Figur 2.

Rötternas lägen då $r \rightarrow 0$.

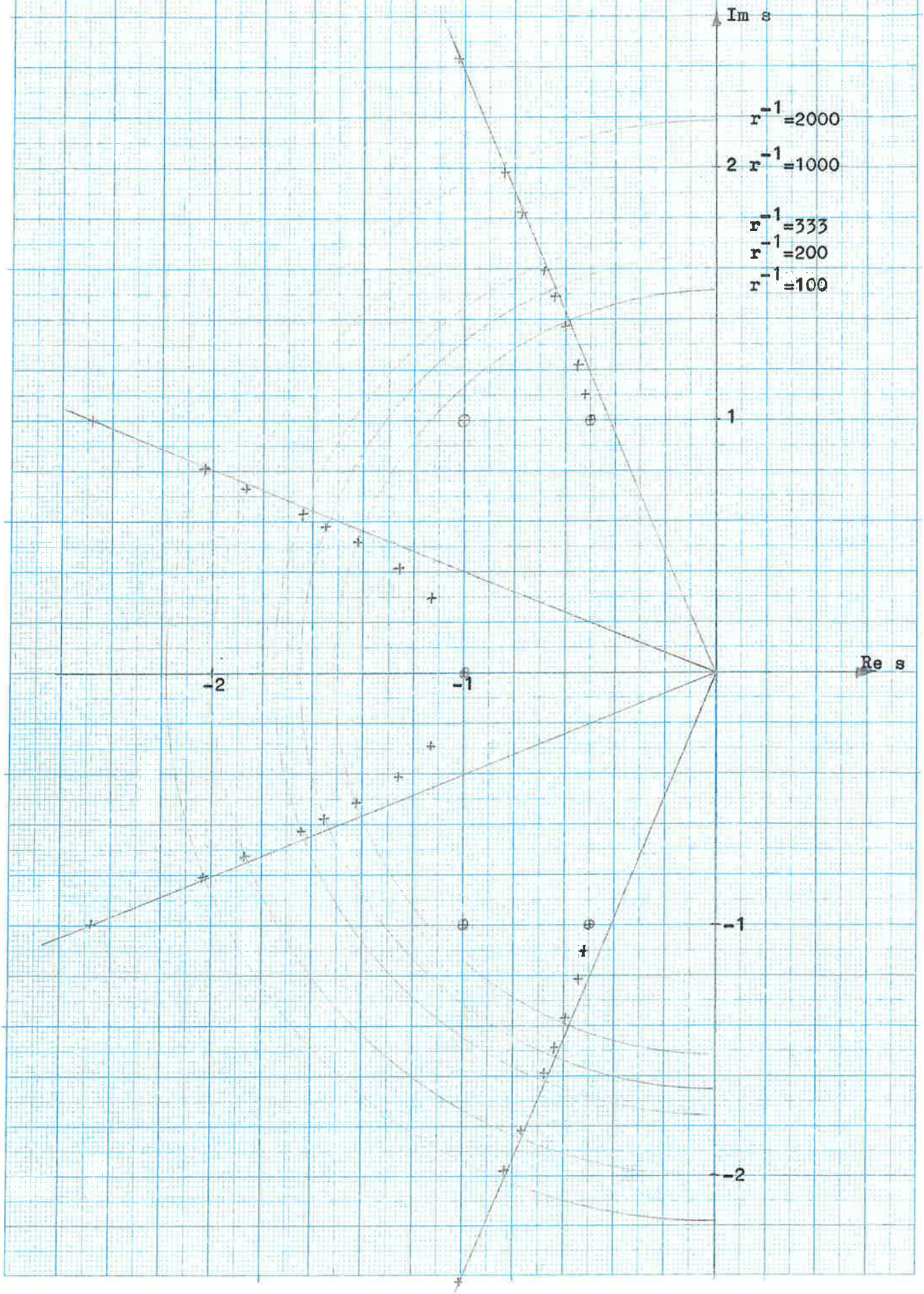
$G(s) = \frac{1}{N}$



Figur 3.

Rötternas lägen då $r \to 0$.

$$G(s) = \frac{1 - s + 0.5s^2}{N}$$

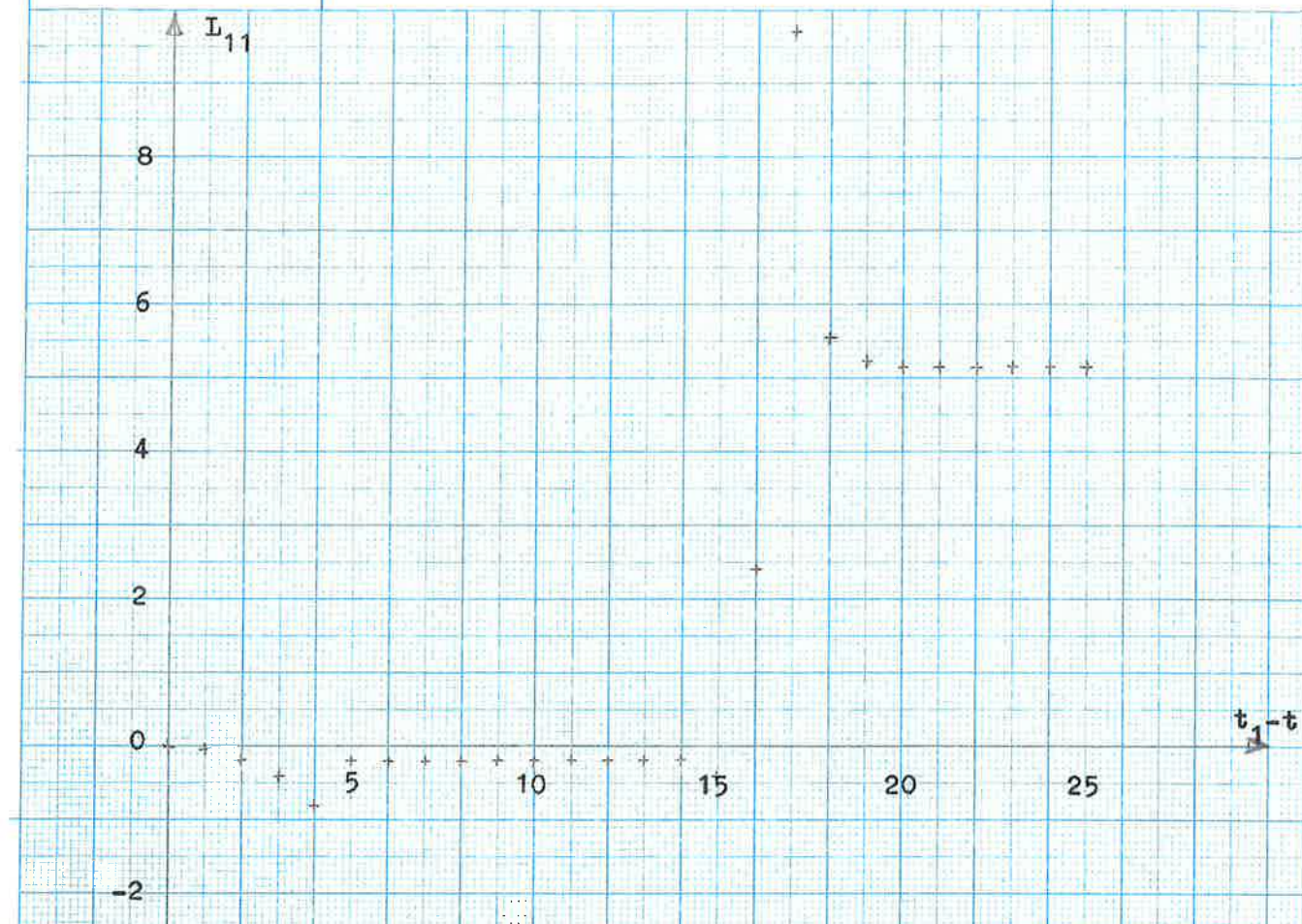


Cirklarna har radien $(|b_m|\sqrt{a_0^2 + r^{-1}})^{\frac{1}{6-m}}$

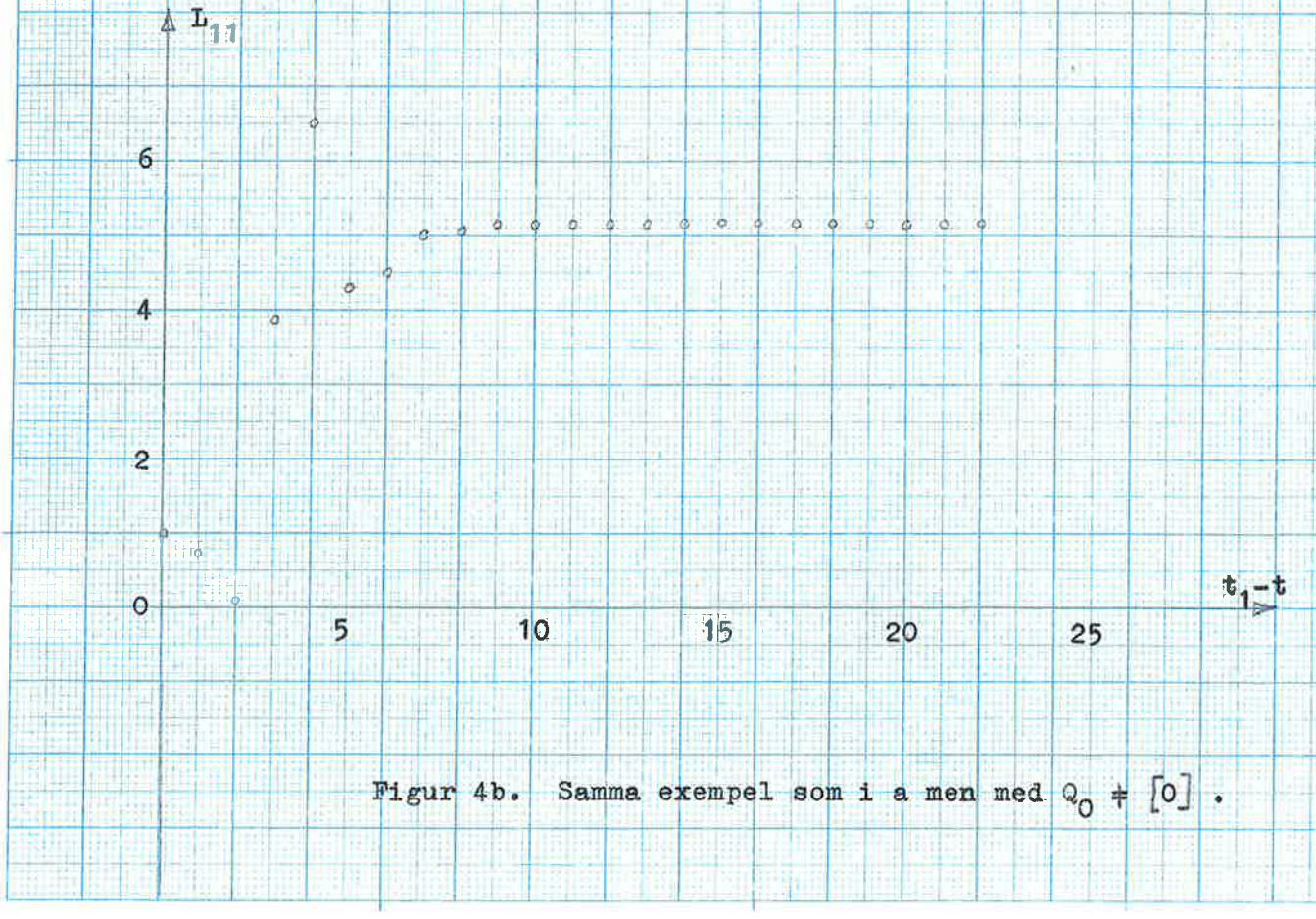
Överföringsfunktionerna för system (B) och (D) har faktorer gemensamma för täljare och nämnare. För båda systemen gäller rang $\begin{bmatrix} C \\ \vdots \\ CA^{n-1} \end{bmatrix} = 4$, dvs systemen är ej observerbara. Tillstånden motsvarande egenvärdena (+1, -1) och (-1 ± i) i system (B) resp. (D) är inte observerbara.

Dessa system har körts med program PLOTLOP som är en modifierad version av LIOPCON. Det ger möjlighet att plotta elementen i L-matrisen mot tidsdifferensen t_1-t. Se figur 4. Härav framgår att för system (B) erhålles med startvärdet $Q_0=[0]$ två stationära värden på styrlagen. Först konvergerar L-matrisen (S-matrisen) mot ett värde motsvarande en positivt semidefinit lösning till SR. Detta är den optimala lösningen till problemet, men den är instabil, och på grund av numeriska avrundningsfel vid beräkningen kommer lösningen att konvergera mot ett stabilt värde som svarar mot en positivt definit lösning till SR. Denna ger emellertid ett större värde på förlustfunktionen.

Om man i stället använder ett Q_0 som har full rang, dvs även de icke-observerbara tillstånden "straffas" vid sluttidpunkten, erhåller man inte den instabila lösningen. Tillståndsvariablerna antar snabbt höga värden och termen $x^T(t_1)Q_0x(t_1)$ kommer att dominera. Men här låter vi $t_1 \to \infty$ varför denna tidpunkt aldrig uppnås.
För system (D) erhålles den stabila optimala lösningen direkt eftersom de icke-observerbara tillstånden är stabila.
Som framgår av figur 4 sker för system (B) övergången från den instabila till den stabila lösningen snabbt. Om man försöker erhålla styrlagen genom att lösa Riccatiekvationen med Runge-Kuttas metod (program RKRICCE) erhålles vid övergången alltför stora värden på elementen i S-matrisen beroende på att ds/dt blir för stor.



Figur 4a. Exempel på hur styrlagen konvergerar för system (B) med $Q_0 = [0]$.



Figur 4b. Samma exempel som i a men med $Q_0 \neq [0]$.

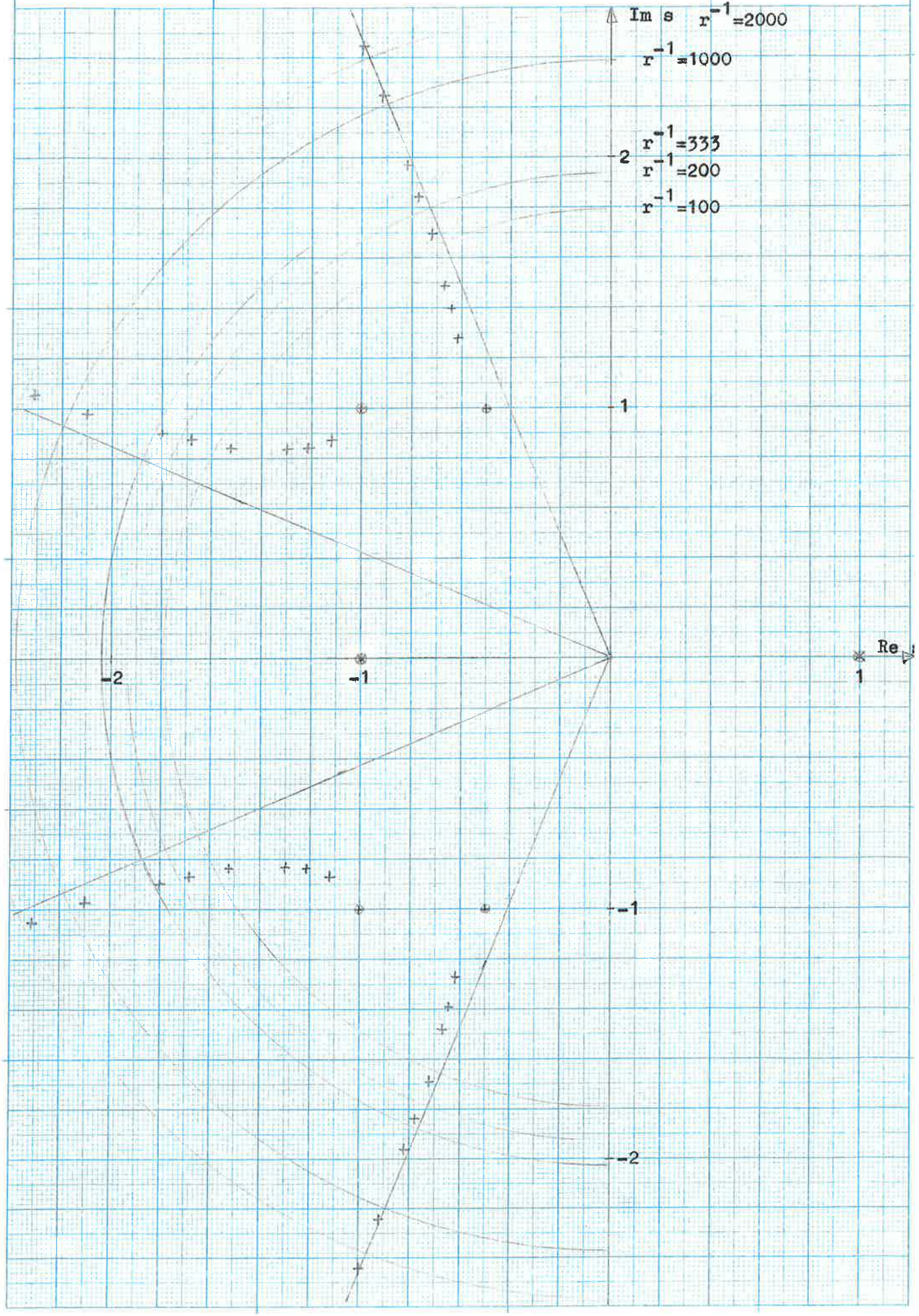
För system (B) erhålles egenvärdena svarande mot den instabila lösningen till SR om EPS ökas till 0.1. Se figur 5. Man erhåller som tidigare ett "Butterworth"-mönster, men ett egenvärde ligger kvar i +1. Det optimala systemet blir således instabilt. Om EPS minskas får man egenvärdena svarande mot den stabila lösningen. Resultatet härvid överensstämmer med Kalmans, dvs egenvärdet svarande mot nollstället till G(s) i +1 övergår i -1, men detta är inte det optimala systemet.

System (D) får samma utseende som (C). Se figur 3.

Figur 5.

Rötternas lägen då $r \to 0$.

$$G(s) = \frac{1 - s^2}{N}$$



4. FLERVARIABLA SYSTEM

Ref. (6) ger en metod att uttrycka det optimala systemets karakteristiska polynom som en explicit funktion av elementen i viktmatrisen i förlustfunktionen, varefter konventionell rotortmetod kan användas för att lokalisera polerna. Det bör observeras att metoden inte ger något uttryck på styrlagen.

Betrakta ett tidsinvariant linjärt dynamiskt system

$$\dot{x} = Ax + Bu$$
$$y = Cx \qquad (4:1)$$

Välj som förlustfunktion $V(u) = \int_0^T (x^T C^T Q\, C\, x + u^T u)\, dt$ (4:2)

där Q är en positivt semidefinit diagonalmatris.

Enligt tidigare blir de kanoniska ekvationerna

$$\begin{bmatrix} \dot{x} \\ \dot{p} \end{bmatrix} = \begin{bmatrix} A & -BB^T \\ -C^T QC & -A^T \end{bmatrix} \begin{bmatrix} x \\ p \end{bmatrix} = A_c \begin{bmatrix} x \\ p \end{bmatrix} \qquad (4:3)$$

där A_c är en 2n x 2n matris.

Karakteristiska ekvationen för det optimala systemet ges av

$$|sI - A + BL| = 0 = \prod_{i=1}^{n} (S - \alpha_i) \qquad (4:4)$$

Lösningen till kanoniska ekvationen, S-matrisen, är ingen enkel funktion av elementen i Q. Men de 2n egenvärdena till A_c innehåller egenvärdena till det optimala systemet A-BL och dess spegelbild kring imaginära axeln i s-planet. Antag nu att systemet beskrivet av A, B och C är fullständigt observerbart och kontrollerbart. Detta medför att det optimala systemet är stabilt, varför egenvärdena med negativ realdel är egenvärden till optimala systemet.

Karakteristiska ekvationen till matrisen A_c kan skrivas på formen

$$0 = |sI - A_c| = \Delta(S) \cdot \Delta(-S) = \bar{\Delta}(S) \cdot \det\left[(sI + A^T) - C^T Q C\, \frac{\Theta(S)}{\bar{\Delta}(S)}\, BB^T\right] \qquad (4:5$$

där $\bar{\Delta}(S) = (sI - A)$ och $\Theta(S) = \mathrm{adj}\,(sI - A)$

Det kan visas att för determinanten av en summa av matriser gäller en identitet som för två andra ordningens matriser blir

$$\det(N + M) = \det \begin{bmatrix} n_{11}+m_{11} & n_{12}+m_{12} \\ n_{21}+m_{21} & n_{22}+m_{22} \end{bmatrix} =$$

$$= |N| + \begin{vmatrix} n_{11} & n_{12} \\ m_{21} & m_{22} \end{vmatrix} + \begin{vmatrix} m_{11} & m_{12} \\ n_{21} & n_{22} \end{vmatrix} + |M| \qquad (4:6)$$

För två n:te ordningens matriser N och M får man 2^n determinanter som består av 1) alla kombinationer med insättning av rader från $|M|$ i $|N|$ 2) $|N|$ 3) $|M|$.
Genom att använda denna identitet på (4:5) erhålles

$$0 = \Delta(s)\,\Delta(-s) + (-1)^n \sum_{i=1}^{2(n-1)} k_i\, N_i(s)\, N_i(-s)$$

där $N_i(s)$ är polynom i s och k_i innehåller q_{ii} element eller en produkt av q_{ii} element.

Detta uttryck ger den karakteristiska ekvationen till det optimala systemet och dess spegelbild kring imaginära axeln, som explicit funktion av elementen i Q-matrisen. Det är skrivet på en form som ger möjlighet att använda rotortmetod då något q_{ii} element varieras.

Asymptotiska egenskaper hos det optimala systemet.

Om ett enda diagonalelement q_{ii} i Q-matrisen varieras och de andra hålles konstanta kan karakteristiska ekvationen uttryckas på formen

$$0 = |sI - A_c| = 1 + \frac{q_{ii}\, k\, N(s)N(-s)}{D(s)D(-s)} \qquad (4:8)$$

där k är en konstant och N(s),N(-s) är polynomen i (4:7) och D(s)D(-s) är den del av karakteristiska ekvationen som inte är funktion av q_{ii}. Om alla diagonalelement utom q_{ii} är noll blir $D(s) = \bar{\Delta}(s)$.

Antag att gradtalen av polynomen D(s) och N(s) är n resp. m. Då något q_{ii} element varieras kommer 2m av de 2n rötterna till D(s)D(-s) att sluta i nollställena till N(s)N(-s) för stora värden på q_{ii}. De återstående 2(n-m) rötterna kan approximeras med

$$\Delta(s)\,\Delta(-s) = k \prod_{i=1}^{n-m} \left(\frac{s}{z_i} + 1\right)\left(\frac{-s}{z_i} + 1\right)$$

där k är en konstant och z_i är rötternas värden.

Om $\left|\frac{s}{z_i}\right| \gg 1$ kommer rötterna att ligga på en cirkel kring origo.
Om (n-m) är jämnt ges asymptotvinklarna till rötterna av

$$\Theta = \frac{(2c + 1)\ 180}{2(n-m)} \qquad c = 0,1,2 \ldots\ldots$$

Om (n-m) är udda ges asymptotvinklarna av

$$\Theta = \frac{(2c + 1)\ 360}{2(n-m)} \qquad c = 0,1,2 \ldots\ldots$$

För stora värden på q_{ii} kommer rötterna till det optimala systemet att ligga på en halvcirkel kring origo i vänstra halvplanet. Funktioner av denna typ kallas Butterworth funktioner.

Exempel

Betrakta systemet

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} x + \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} u$$

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

med förlustfunktionen definierad enligt (4:2)

$$\bar{\Delta}(s) = |sI - A| = s^2 - 1$$

$$\Theta(s) = \begin{bmatrix} s & 1 \\ 1 & s \end{bmatrix}$$

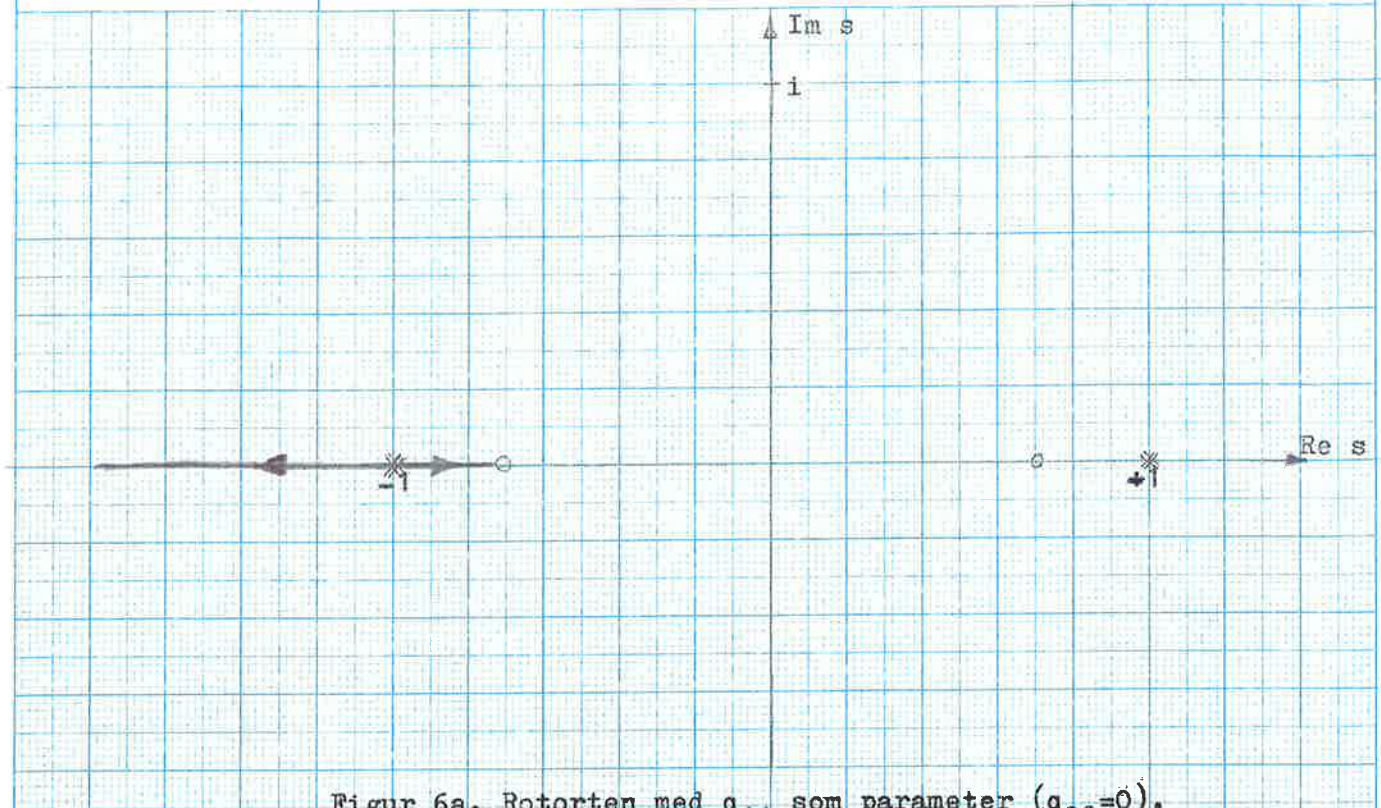
Insättning i ekvation (4:5) och tillämpning av (4:6) ger

$$0 = (s^2-1)^2 - q_{11}(2s^2-1) - q_{22}(s^2-2) + q_{11}q_{22}$$

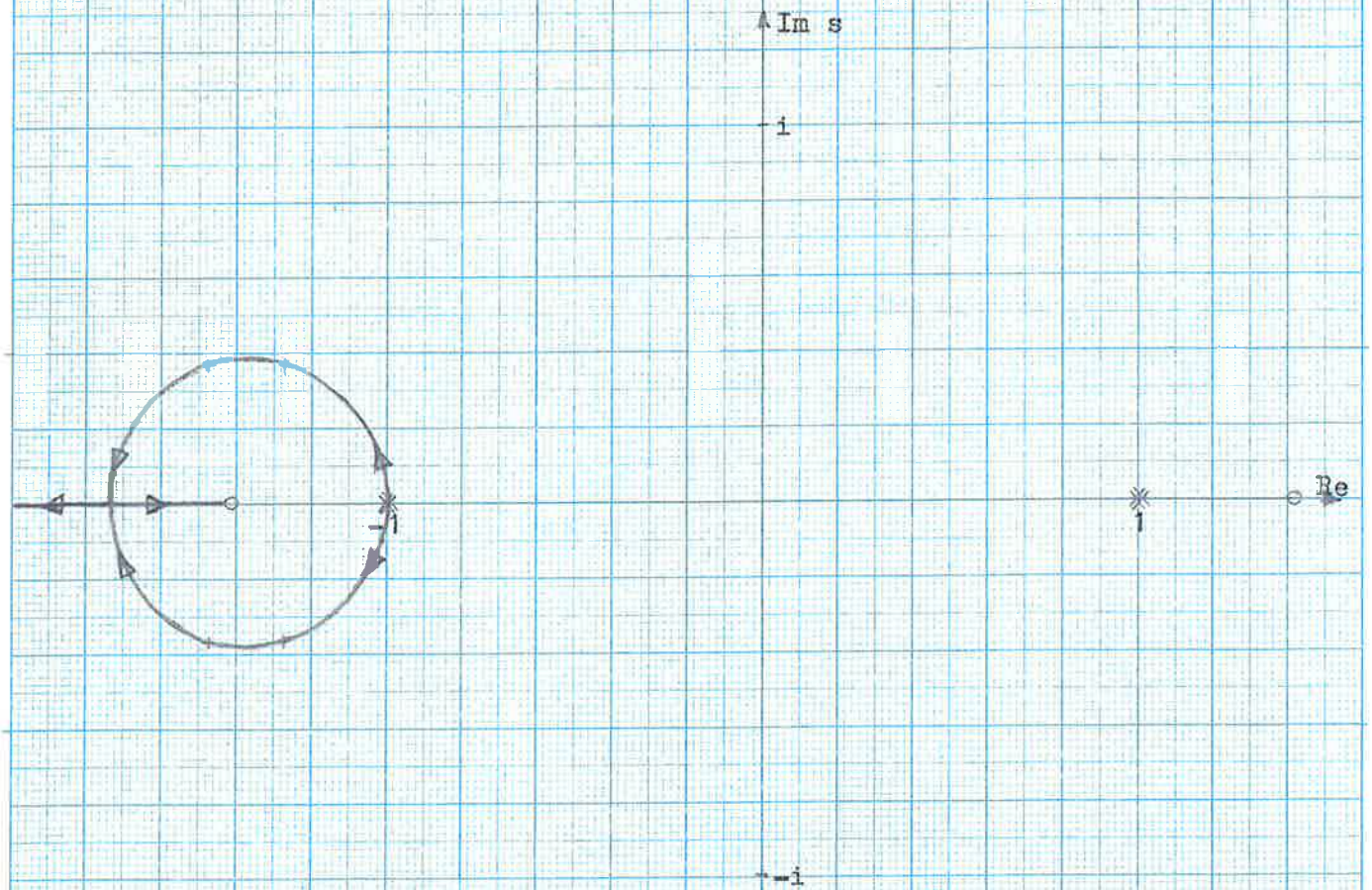
Rotorten för denna ekvation med q_{11} $(q_{22}=0)$ och q_{22} $(q_{11}=0)$ som parametrar är inritad i figur 6.

Beräkning av optimala flervariabla system.

Som nämnts erhåller man inget uttryck på styrlagen med den beskrivna metoden. Det är ej heller enkelt att konstruera något program, som genomför de algebraiska manipulationerna för något godtyckligt system, varför metoden endast omnämnes här.



Figur 6a. Rotorten med q_{11} som parameter (q_{22}=0).



Figur 6b. Rotorten med q_{22} som parameter (q_{11}=0).

Flervariabla system har dock undersökts varvid samma metod som tidigare använts, dvs den optimala styrlagen beräknas och det optimala systemet erhålles.

Betrakta ett system definierat av följande matriser

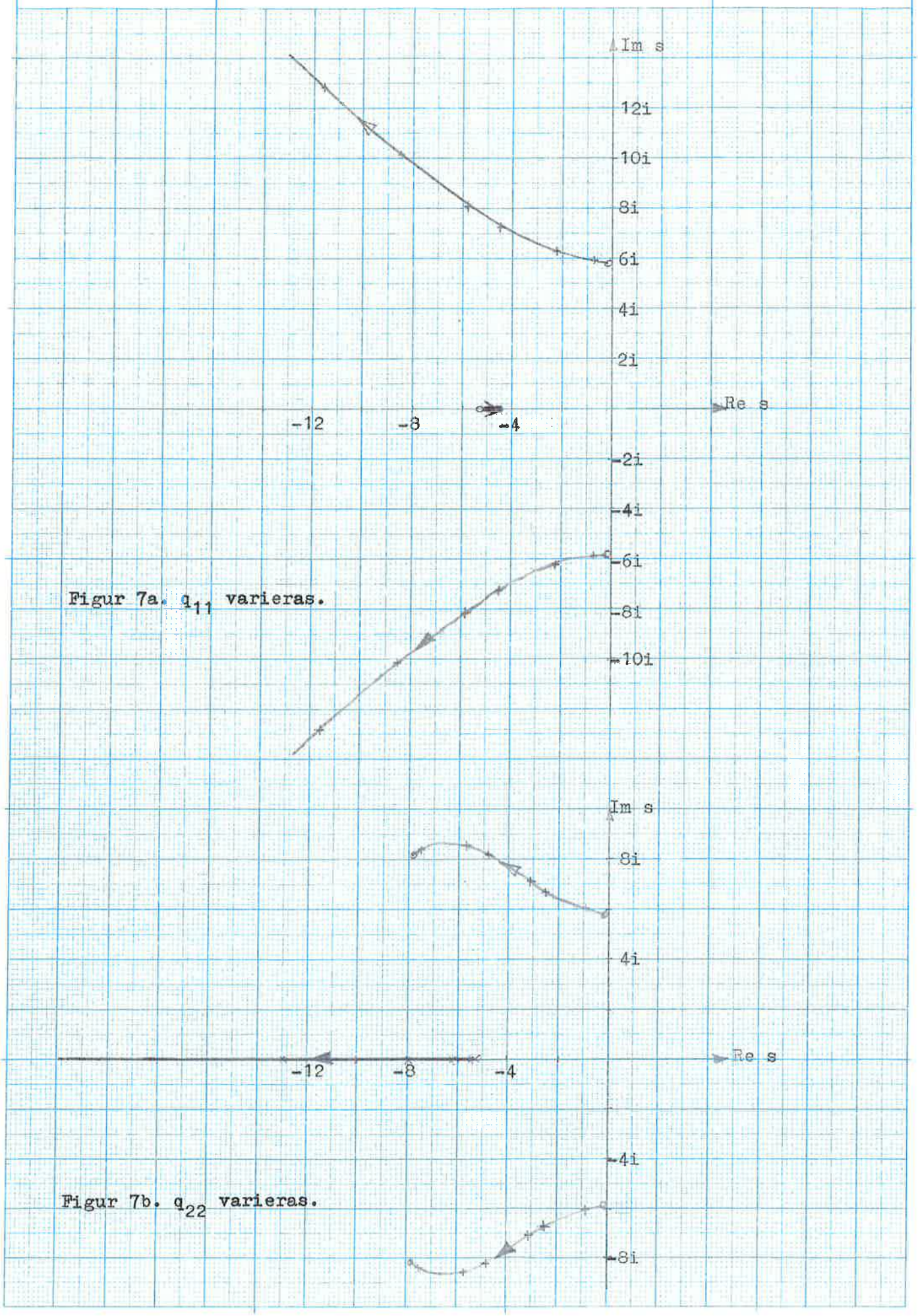
$$A = \begin{bmatrix} 1 & 2 & -3 \\ -4 & 5 & -0.6 \\ 7 & 8 & -0.9 \end{bmatrix} \qquad B = \begin{bmatrix} 0.1 & 0 & 2 \\ 0 & 10 & 0 \\ 0 & 0 & 100 \end{bmatrix} \qquad C = I$$

Med förlustfunktionen enligt (4:2)

Egen-värdena till A är 5.3, -0.124 ± i•5.89, dvs systemet är instabilt. Sök nu en återkopplingsmatris sådan att systemet är stabilt och stegsvaret har god dämpning. I figur 7 är rotorten inritad för variation av vart och ett av de tre q_{ii} elementen när de andra är noll. Av figurerna framgår att för små värden på parametrarna q_{ii} är egenvärdena till slutna systemet lika med egenvärdena till A med negativ realdel. Detta överensstämmer med den tidigare beskrivna metoden ty D(s) är här karakteristiska polynomet till matrisen A. Vidare framgår det att vissa grenar slutar för stora värden på q_{ii} i punkter som skulle motsvara nollställen till N(s). Med q_{33}=0.125 erhålles de komplexa rötterna -3.06 ± i1.92 samt roten -34.6. Ett tämligen bra dämpat system erhålles således med Q = diag.(0, 0, 0.125).

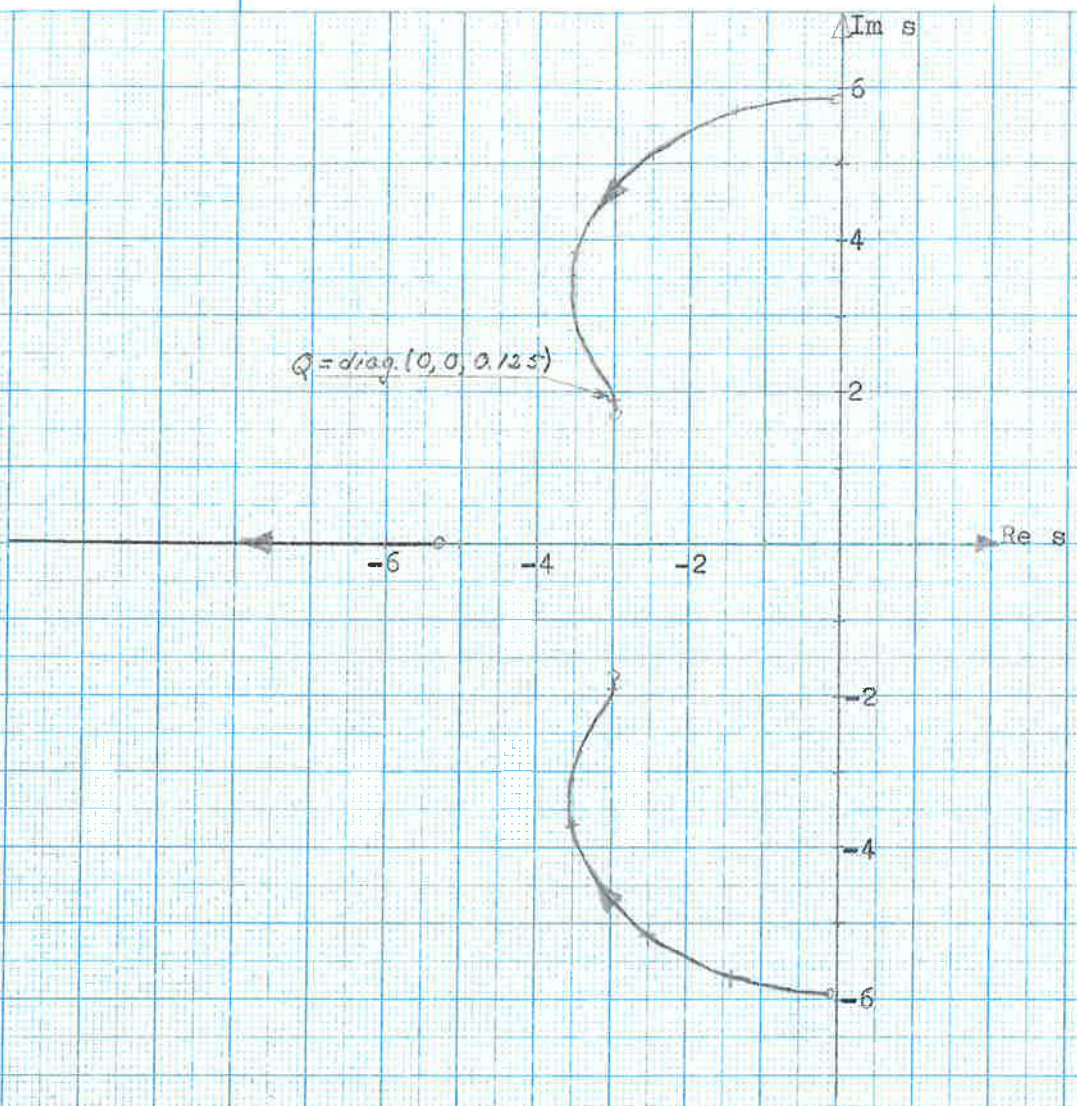
Vid körning av detta system med program BUTTER visade det sig att för stora värden på q_{ii} elementen går matrisen $\sum_{11}(t:t_1)$ + + $\sum_{12}(t:t_1)Q_0$ inte att invertera då tidsdifferensen Δt=1.0 som tidigare kunnat användas. Vid en minskning av Δt till 0.1 kan dock alla önskade värden erhållas.

Figur 7. Rotorten för det optimala systemet då q_{ii} elementen varieras.



Figur 7a. q_{11} varieras.

Figur 7b. q_{22} varieras.



Figur 7c. q_{33} varieras.

Modellföljningsystem för ett B-26 flygplan.

Tillståndsekvationen för denna metod av modellföljning är

$$\begin{bmatrix} \dot{x}_m(t) \\ \dot{x}_p(t) \end{bmatrix} = \begin{bmatrix} M & [0] \\ [0] & A_p \end{bmatrix} \begin{bmatrix} x_m(t) \\ x_p(t) \end{bmatrix} + \begin{bmatrix} [0] \\ B_p \end{bmatrix} u_p(t)$$

där index m och p står för modell resp. plan och L är en n:te ordningens matris som beskriver modellen. Ekvationen kan skrivas mera kompakt som

$$\dot{x}(t) = Ax(t) + Bu_p(t)$$

Genom att definiera utgångsmatrisen som

$$C = \left[I \;\vdots\; -I \right]$$

en n x 2n matris, kommer skillnaden mellan modellens tillståndsvektor $x_m(t)$ och flygplanets $x_p(t)$ att minimeras med förlustfunktionen definierad enligt (4:2).

Numeriska värden för flygplanet är

$$A_p = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -2.93 & -4.75 & 0.78 \\ 0.086 & 0 & -0.11 & -1.0 \\ 0 & -0.042 & 2.59 & -0.39 \end{bmatrix} \qquad B_p = \begin{bmatrix} 0 & 0 \\ 0 & -3.91 \\ 0.035 & 0 \\ -2.53 & 0.31 \end{bmatrix}$$

Modellen definieras som

$$M = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -1 & -73.14 & 3.18 \\ 0.086 & 0 & -0.11 & -1.0 \\ 0.0086 & 0.086 & 8.95 & -0.49 \end{bmatrix}$$

Q-matrisen definieras som tidigare.

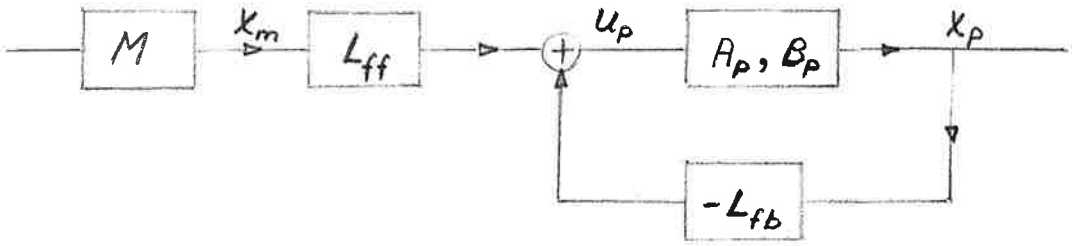
Det optimala systemet blir

$$\begin{bmatrix} \dot{x}_m(t) \\ \dot{x}_p(t) \end{bmatrix} = \begin{bmatrix} M & [0] \\ -B_p L_{ff} & A_p - B_p L_{fb} \end{bmatrix} \begin{bmatrix} x_m(t) \\ x_p(t) \end{bmatrix}$$

där återkopplingsmatrisen L har uppdelats enligt

$$L = \left[L_{ff} \;\vdots\; L_{fb} \right]$$

där index ff står för feedforward och fb feedback.



Blockschema för hela systemet.

Meningen med denna modellföljningsmetod är att förbättra svaret från styrobjektet genom att införa en återkoppling sådan att utgången från systemet följer signaler från modellen. Om amplituden på egenvärdena till det återkopplade systemet $A_p - B_p L_{fb}$ är avsevärt större än modellens kommer svaret från styrobjektet att följa modellsvaret. Men det återkopplade systemet är oberoende av modellen varför en optimal styrlag, L_{fb}, till detta, som ger de önskade egenvärdena, kan beräknas. Det härigenom erhållna värdet på Q-matrisen används sedan för att beräkna den totala styrlagen L.

Egenvärdena till modellmatrisen M är -1.065, 0.00275, -0.288 ± i2.94. Egenvärdena till det återkopplade systemet, då elementen i Q-matrisen varierar finns inritade i figur 8. Först varieras Q som $\varrho \cdot I$ varigenom ett lämpligt värde på ϱ kan erhållas. Därefter varieras vart och ett q_{ii} element med de övriga konstanta.

Man måste ta hänsyn till vissa praktiska begränsningar för förstärkningen i återkopplingsslingan. Således måste återkopplingsmatrisen, L_{fb}, vara mindre än

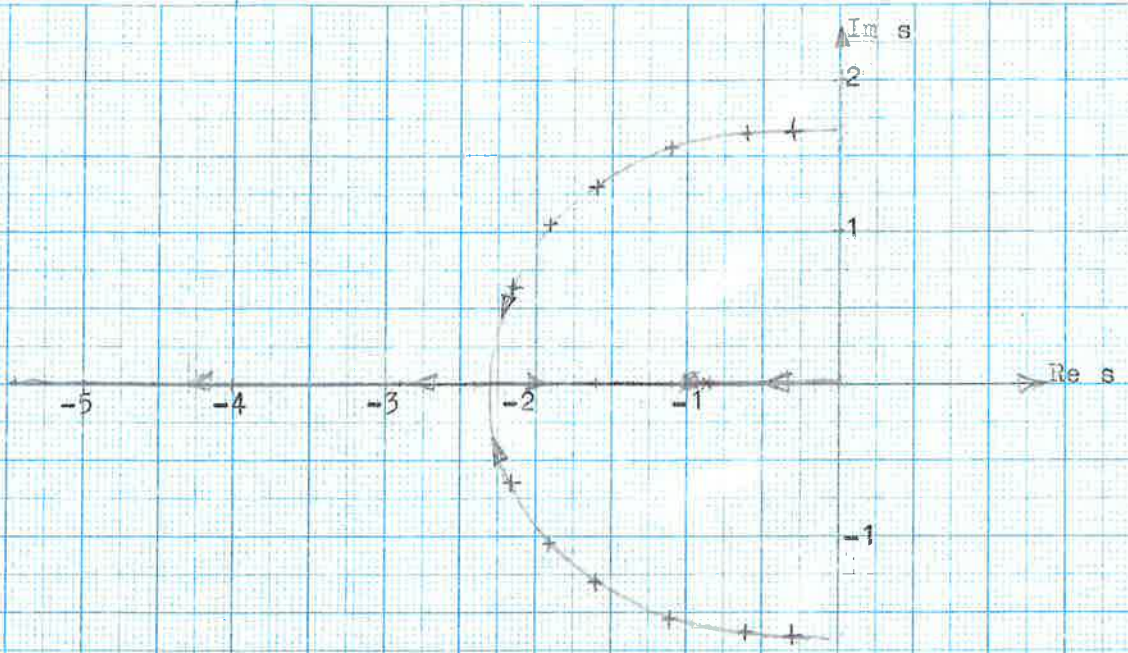
$$L_{fbmax} = \begin{bmatrix} 5 & 0.5 & 5 & 5 \\ 5 & 2 & 20 & 1 \end{bmatrix}$$

Med $Q = \varrho I$ erhålles för $\varrho \geq 3$ reella egenvärden och då ϱ ökas ytterligare kommer två egenvärden att förflyttas åt vänster längs negativa reella axeln, medan de två övriga går mot approx. -1.

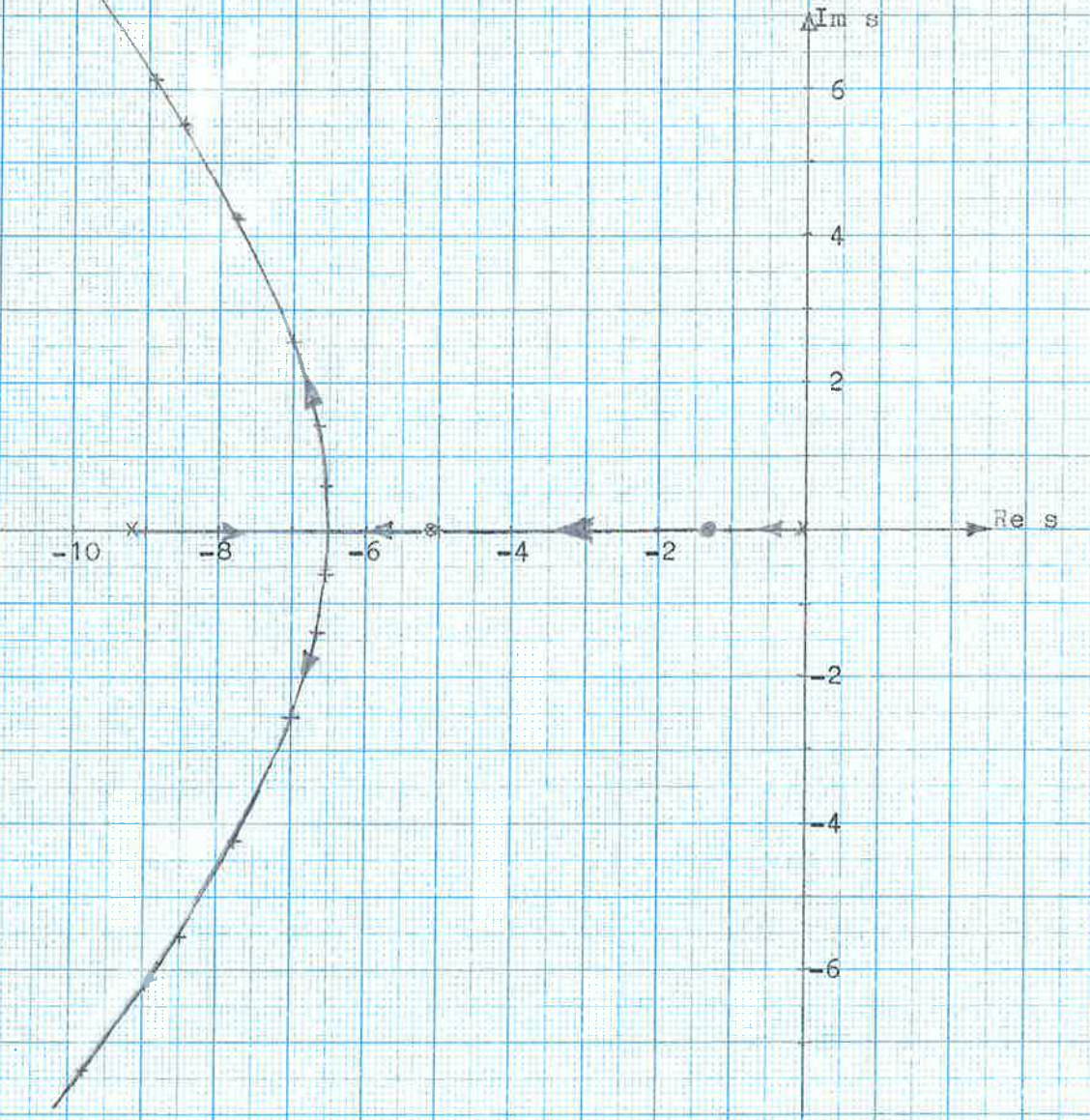
Det är med de två förstnämnda som systemets egenskaper kan ändras, ty de båda andras variation är obetydlig. I tabell 1 finns värden på L-matrisen samt egenvärdena till $A_p - B_p L_{fb}$ för $\rho = 5$. Härav framgår att L_{fbmax} ej överskrides samt att det till absolutbelopp minsta av de båda påverkbara egenvärdena, α_{min}, överstiger det största modellegenvärdet med en faktor 2. Transientsvaret (se figur 9) visar emellertid att systemets följsamhet inte är helt tillfredsställande med Q=5I.

En förbättring erhålles om man varierar vart och ett q_{ii} element medan de övriga sättes lika med 5. Vidare bör förhållandet mellan α_{min} och det största modellegenvärdet ökas till minst 3, vilket ger $\alpha_{min} = 10$. Värdet på varje q_{ii} för att erhålla detta finns i tabell 2. Härvid är det lämpligast att använda det minsta q_{ii} som ger önskat resultat, och i detta fall är systemet känsligast för variation i q_{22} eller q_{44}. Q=diag,(5, 6.4, 5, 5) skulle i så fall ge bäst resultat, men en ökning i q_{22} skulle direkt påverka elementet L_{fb22}, som redan ligger nära sin övre gräns. I stället har Q=diag.(5, 5, 5, 20) använts och för detta värde har den totala styrlagen beräknats. Se tabell 1. Av figur 9 fram-går att transientsvaret nu kan anses tillfredsställande.

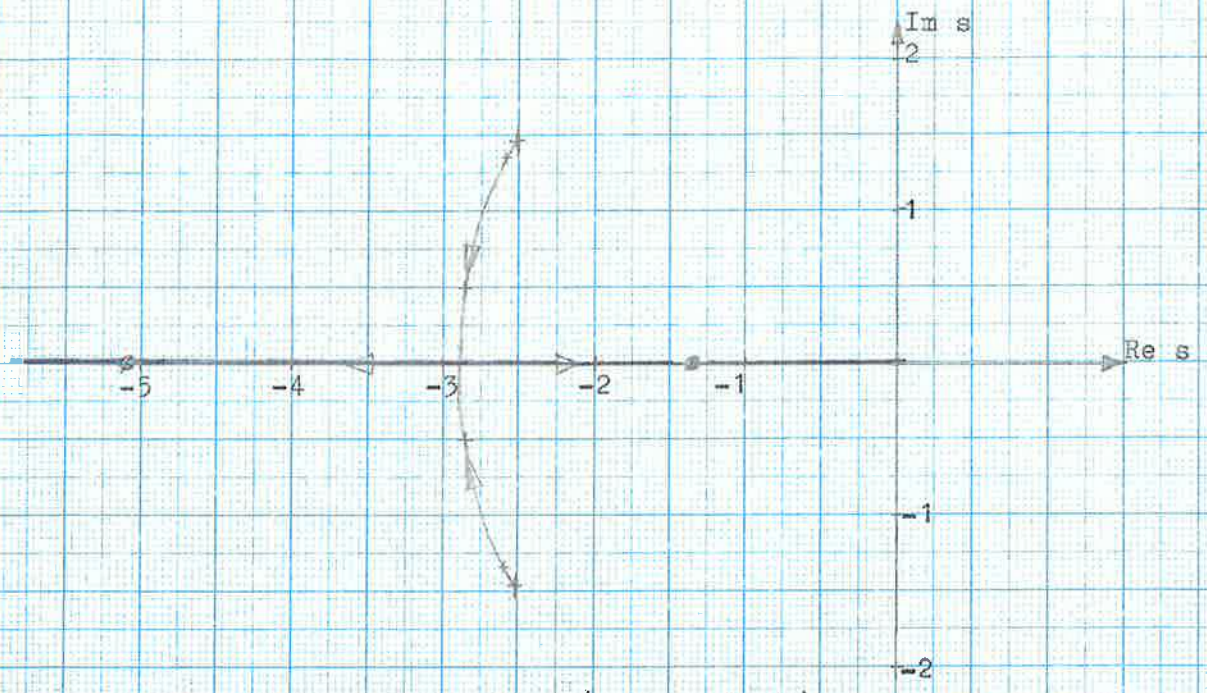
Figur 8. Rotorten för det optimala systemet då q_{ii} elementen varieras.



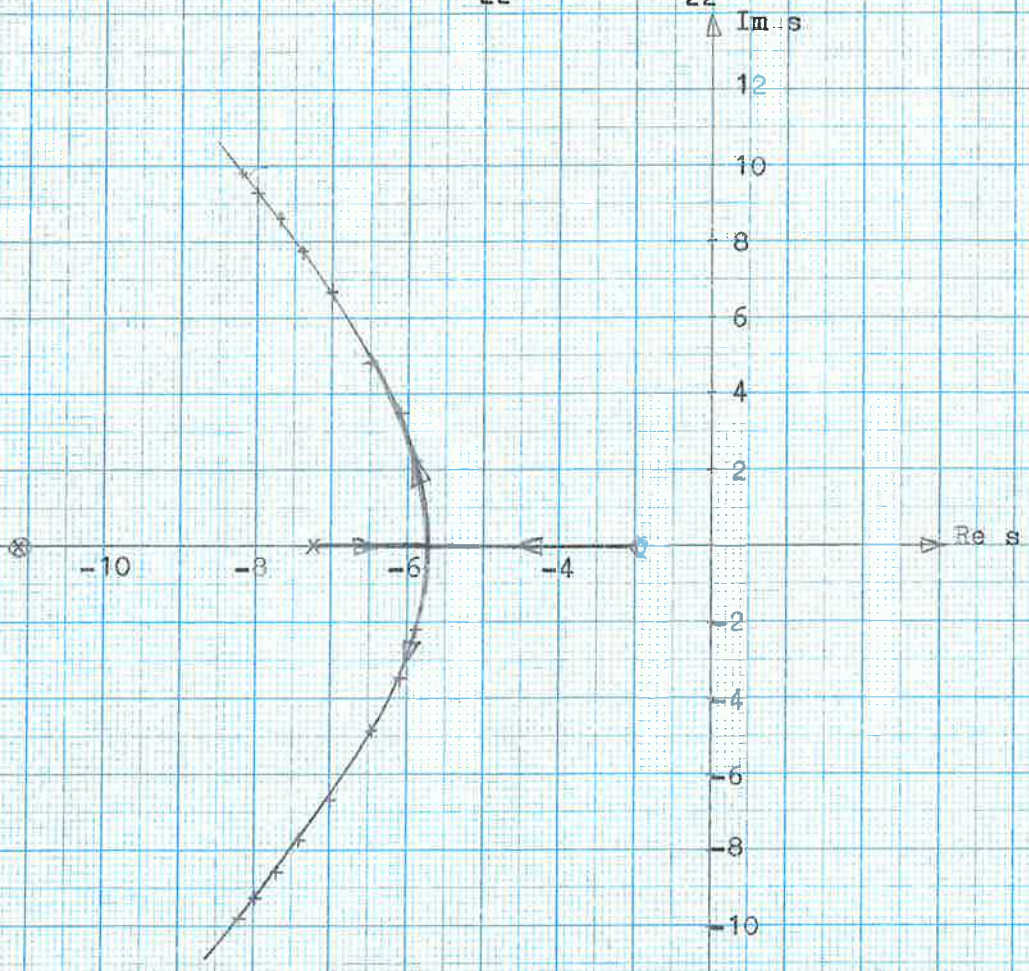
Figur 8a. Q= I, varieras.



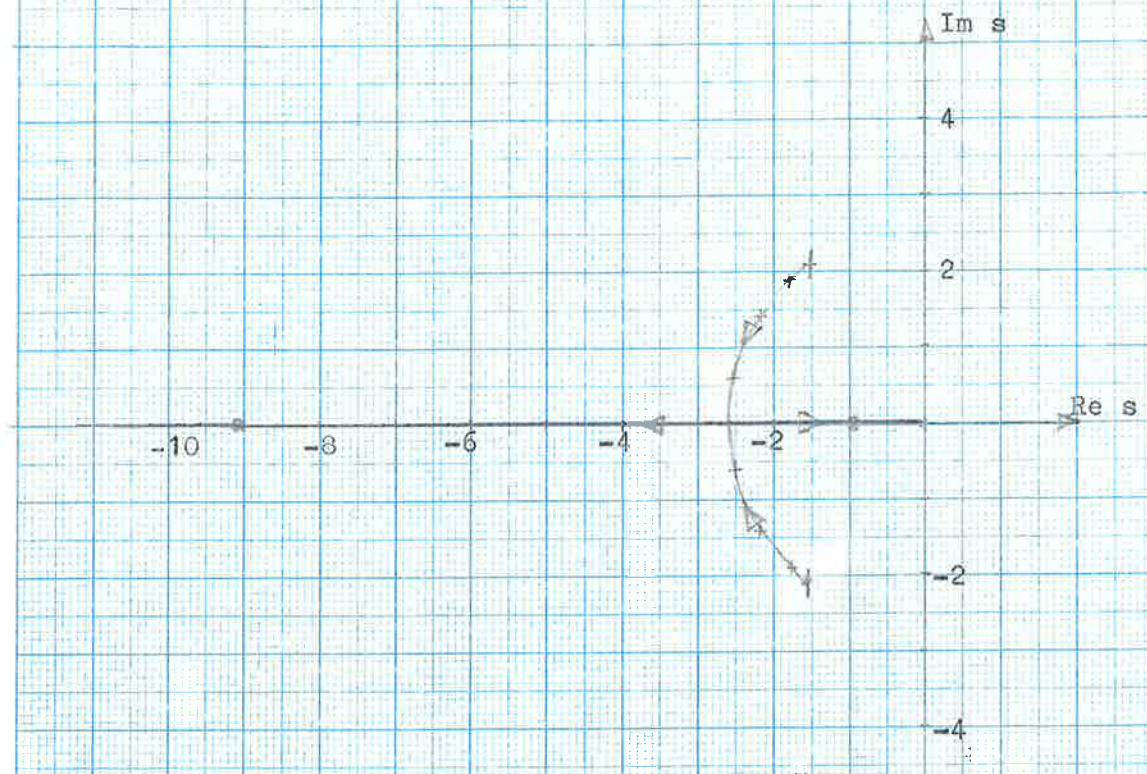
Figur 8b. Q = diag.(q_{11}, 5, 5, 5), q_{11} varieras



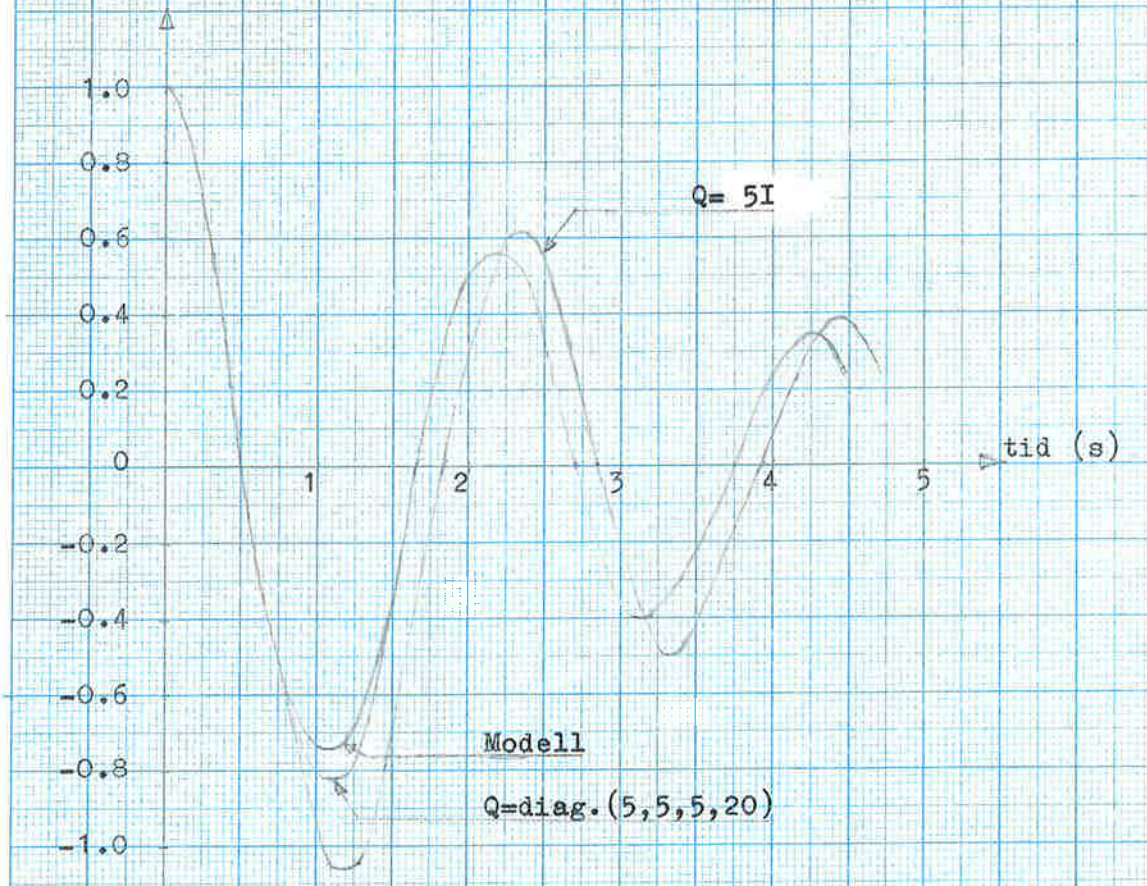
Figur 8c. Q = diag.(5, q_{22}, 5, 5), q_{22} varieras.



Figur 8d. Q = diag.(5, 5, q_{33}, 5), q_{33} varieras.



Figur 8e. Q = diag.(5, 5, 5, q_{44}), q_{44} varieras.



Figur 9. Stegsvaret.

TABELL 1

Q	L_{ff}	L_{fb}	Egenvärden till $A_p - B_p L_{fb}$
5,5,5,5	0.104 0.377 -3.63 4.16 2.035 2.211 -15.3 2.59	-2.53 -0.185 1.58 -2.34 -2.21 -1.83 0.70 -0.01	-0.99, -1.3, -5.13, -9.14
5,5,5,20	0.101 0.344 -2.15 5.61 2.045 2.172 -1.54 2.42	-0.201-0.185 1.42 -4.42 -2.23 -1.83 0.16 -0.26	-0.908, -0.66, -9.09, -11.2

TABELL 2

q_{ii}	Värde på q_{ii} för $\lvert\alpha_{min}\rvert=10$	α_{min}
q_{11}	640	$-8.5 \pm i5.3$
q_{22}	6.4	-10
q_{33}	422	$-7.6 \pm i6.4$
q_{44}	16	-10

```
      PROGRAM BUTTER
C
C     COMPUTES THE STADY-STATE VALUE OF THE OPTIMAL CONTROL LAW
C     OF CONTINUOUS LINEAR DYNAMIC SYSTEMS WITH QUADRATIC LOSS.
C     BY CALLING EIGUNS THE EIGENVALUES OF THE CLOSED-LOOP
C     SYSTEM MATRIX(A-BL) ARE COMPUTED.
C
C     N-NUMBER OF STATES(MAX 10).
C     NU-NUMBER OF INSIGNALS(MAX 10).
C     TIMEDIF-TIME DIFFERENCE BETWEEN THE POINTS.
C     ITER-ITER=0 MEANS THAT THE FUNDAMENTAL MATRIX WILL BE COMPUTED
C     FOR EACH STEP, ITER=1 MEANS THAT THE FUNDAMENTAL MATRIX IS COMPU-
C     TED ONLY FOR THE FIRST STEP AND THEN USED IN THE OTHER STEPS.
C     IMAX-MAX. NUMBER OF POINTS IN WHICH THE S-MATRIX
C     IS COMPUTED IF IT HAS NOT CONVERGED.
C     EPS-A TOLERANCE FOR THE NORM OF THE DIFFERENCE BETWEEN
C     TWO CONSEQUENT S-MATRICES FOR ACCEPTANCE OF THE STATIONARY
C     VALUE.
C     NR-IF NR .GT.0, A NEW EXAMPLE IS EXECUTED. IF NR .LE.0
C     THERE ARE NO MORE DATA TO BE EXECUTED
C     NEWEPS-NEWEPS=0 MEANS THAT THE SAME VALUE OF EPS AS IN THE PRECEDING
C     EXAMPLE IS USED. EPS=1 MEANS THAT IN THE PRESENT EXAMPLE IS A NEW
C     VALUE OF EPS USED.
C     NEWQ0,NEWQ1,NEWQ2-DITO FOR THE MATRICES Q0,Q1,Q2.
C
C     SUBROUTINE REQUIRED
C             RICCE
C             MEXP7T
C             GJRV
C             NORM
C             EIGUNS
C
      DIMENSION A(10,10),B(10,10),Q0(10,10),Q1(10,10),Q2(10,10)
      DIMENSION S(10,10),UL(10,10),C(10,10),SG(10,10),ST(10,10)
      DIMENSION BL(10,10),ABL(10,10),EIGR(10),EIGI(10)
C
      READ 1000,N,NU,IMAX,ITER,TIMEDIF
 1000 FORMAT(4I3,F10.5)
      READ 1001,((A(I,J),J=1,N),I=1,N)
      READ 1001,((B(I,J),J=1,NU),I=1,N)
 1001 FORMAT(4E20.10)
      PRINT 1023
 1023 FORMAT(30H1PRINTOUTS FROM PROGRAM BUTTER,/)
      PRINT 1002
 1002 FORMAT(14H THE SYSTEM IS,/)
      PRINT 1003
 1003 FORMAT(9H MATRIX A,/)
      DO 2 K=1,N
    2 PRINT 1004,(A(K,J),J=1,N)
 1004 FORMAT(6E20.10)
      PRINT 1005
 1005 FORMAT(/,9H MATRIX B,/)
      DO 4 K=1,N
    4 PRINT 1004,(B(K,J),J=1,NU)
C
      PRINT 1011,TIMEDIF
 1011 FORMAT(/,36H TIME DIFFERENCE BETWEEN THE POINTS=,F10.5,/)
      IF(ITER) 14,12,14
   12 PRINT 1012
 1012 FORMAT(48H THE FUNDAMENTAL MATRIX IS COMPUTED AT EACH STEP,/)
      GO TO 16
```

```
   14 PRINT 1013
 1013 FORMAT(54H THE FUNDAMENTAL MATRIX IS COMPUTED ONLY AT FIRST STEP)
   16 CONTINUE
C
      READ 2002, NR,NEWEPS,NEWQ0,NEWQ1,NEWQ2
 2002 FORMAT(5I1)
      IF(NR) 200,200,210
  210 CONTINUE
      PRINT 900
  900 FORMAT(1H1)
C
      IF(NEWEPS) 212,212,211
  211 READ 2004,EPS
 2004 FORMAT(E20.10)
      PRINT 2003,EPS
 2003 FORMAT(/,33H THE TOLERANCE FOR CONVERGENCE IS,E20.10,/)
  212 CONTINUE
C
      IF(NEWQ0) 216,216,215
  215 READ 1001,((Q0(I,J),J=1,N),I=1,N)
      PRINT 1006
 1006 FORMAT(/,10H MATRIX Q0,/)
      DO 6 K=1,N
    6 PRINT 1004,(Q0(K,J),J=1,N)
  216 CONTINUE
C
      IF(NEWQ1) 219,219,218
  218 READ1001,((Q1(I,J),J=1,N),I=1,N)
      PRINT 1007
 1007 FORMAT(/,10H MATRIX Q1,/)
      DO 8 K=1,N
    8 PRINT 1004,(Q1(K,J),J=1,N)
  219 CONTINUE
C
      IF(NEWQ2) 221,221,220
  220 READ 1001,((Q2(I,J),J=1,NU),I=1,NU)
      PRINT 1040
 1040 FORMAT(/,10H MATRIX Q2,/)
      DO 225 K=1,NU
  225 PRINT 1004,(Q2(K,J),J=1,NU)
  221 CONTINUE
C
      DO 20 I=1,NU
      DO 20 J=1,NU
   20 UL(I,J)=Q2(I,J)
      CALL GJRV(UL,NU,1.0E-008,IERR,10)
      IF(IERR+1) 23,22,23
   22 PRINT 1017
 1017 FORMAT(//,39H THE MATRIX Q2 IS NOT POSITIVE DEFINITE)
      GO TO 118
   23 DO 25 I=1,NU
      DO 25 J=1,N
      R=0.
      DO 24 K=1,NU
   24 R=R+UL(I,K)*B(J,K)
   25 C(I,J)=R
      DO 27 I=1,NU
      DO 27 J=1,N
      R=0.
      DO 26 K=1,N
   26 R=R+C(I,K)*Q0(K,J)
   27 UL(I,J)=R
```

```
      DO 28 I=1,N
      DO 28 J=1,N
   28 SG(I,J)=QQ(I,J)
   29 ICOUNT=0
C
C     START THE LOOP
   30 CONTINUE
      ICOUNT=ICOUNT+1
      TD=TIMEDIF*FLOAT(ICOUNT)
      IF(ITER-1) 36,40,36
   36 CONTINUE
C
      CALL RICCE(A,B,QQ,Q1,Q2,S,N,NU,10,10,TD,IERR)
C
      IF(IERR+1) 51,38,51
   38 PRINT 1019
 1019 FORMAT(/,33H AN INVERSION HAS FAILED IN RICCE)
      GO TO 118
C
C     ITERATION
C
   40 IF(ICOUNT-1) 46,42,46
   42 CONTINUE
C
      CALL RICCE(A,B,SG,Q1,Q2,S,N,NU,10,10,TD,IERR)
C
      IF(IERR+1) 51,44,51
   44 PRINT 1019
      PRINT 1020
 1020 FORMAT(/,50H THE PROBLEM IS IMPOSSIBLE TO SOLVE WITH ITERATION)
      GO TO 118
   46 CONTINUE
C
      CALL ITERATE(A,B,SG,Q1,Q2,S,N,NU,10,10,TD,IERR)
C
      IF(IERR+1) 51,50,51
   50 PRINT 1019
      GO TO 118
   51 CONTINUE
      DO 52 I=1,N
      DO 52 J=1,N
   52 ST(I,J)=SG(I,J)-S(I,J)
   53 CALL NORM(ST,N,10,P)
      IF(P-EPS) 60,60,54
   54 CONTINUE
      DO 55 I=1,N
      DO 55 J=1,N
   55 SG(I,J)=S(I,J)
   56 CONTINUE
      IF(ICOUNT-IMAX) 30,57,57
   57 PRINT 1041
 1041 FORMAT(/,31H THE MATRIX S HAS NOT CONVERGED)
      GO TO 118
   60 PRINT 1021
 1021 FORMAT(/,18H COMPUTED S-MATRIX,/)
      DO 62 K=1,N
   62 PRINT 1004,(S(K,J),J=1,N)
      PRINT 1022
 1022 FORMAT(/,26H COMPUTED L-MATRIX(U=-L*X),/)
      DO 66 I=1,NU
      DO 66 J=1,N
      R=0.
```

```
      DO 64 K=1,N
   64 R=R+C(I,K)*S(K,J)
   66 UL(I,J)=R
      DO 68 K=1,NU
   68 PRINT 1004,(UL(K,J),J=1,N)
  101 CONTINUE
      DO 104 I=1,N
      DO 104 J=1,N
      T=0.
      DO 102 K=1,NU
  102 T=T+B(I,K)*UL(K,J)
  104 BL(I,J)=T
      DO 106 I=1,N
      DO 106 J=1,N
  106 ABL(I,J)=A(I,J)-BL(I,J)
      PRINT 1050
 1050 FORMAT(/,13H MATRIX(A-BL),/)
      DO 108 K=1,N
  108 PRINT 1004,(ABL(K,J),J=1,N)
C
C
      CALL EIGUNS(ABL,EIGR,EIGI,N,10,0,IERR)
      IF(IERR-1) 114,110,112
  110 PRINT 1052
 1052 FORMAT(/,34H THE TRIDIAGONALISATION HAS FAILED)
      GO TO 118
  112 PRINT 1054
 1054 FORMAT(/,36H THE MULLER METHOD FAILS TO CONVERGE)
      GO TO 118
  114 PRINT 1056
 1056 FORMAT(/,31H COMPUTED EIGENVALUES OF (A-BL),/)
  115 PRINT 1057
 1057 FORMAT(39H      REAL PART           IMARGINARY PART,/)
  116 PRINT 1058,(EIGR(I),EIGI(I),I=1,N)
 1058 FORMAT(2E20.10)
C
  118 CONTINUE
      GO TO 16
C
  200 CONTINUE
      CALL EXIT
      END
```

```
      SUBROUTINE RICCE(A,B,Q0,Q1,Q2,S,N,NU,IA,IB,TD,IERR)
C     THE SUBROUTINE COMPUTES THE SOLUTION TO THE RICATTIEQUATION
C     DS/DT=(AT)*S+S*A-S*B*(Q2-1)*(BT)*S+Q1 WITH S(T1)=Q0,
C     BY USING THE EXPONENTIAL SERIES FOR THE CANONICAL EQUATION.
C     AUTHOR,K.MORTENSSON 05/10-67.
C
C     A,Q0,Q1,S=NXN-MATRICES,S(T) IS THE SOLUTION.
C     B=NXNU-MATRIX.
C     Q2=NUXNU-MATRIX.
C     IA AND IB ARE THE DIMENSION PARAMETERS.
C     TD IS THE DIFFERENCE T1-T.
C     IERR IS RETURNED=-1 IF ANY INVERSION HAS FAILED.
C     MAXIMUM ORDER OF THE SYSTEM=10.
C     THE ROUTINE HAS AN ENTRY POINT CALLED ITERATE.
C     WHEN THE ROUTINE IS CALLED WITH ITERATE,WHICH REQUIRES THAT
C     A PREVIOUS CALL TO RICCE HAS BEEN MADE,USE IS MADE OF THE IN THE
C     FIRST CALL COMPUTED FUNDAMENTALMATRIX.Q0 IS THEN SET EQUAL TO
C     THE PREVIOUSLY COMPUTED S OUTSIDE THE ROUTINE BEFORE CALLING.
C
C     SUBROUTINE REQUIRED
C             MEXP7T
C             NORM
C             GJRV
C
      DIMENSION A(IA,IA),B(IA,IB),Q0(IA,IA),Q1(IA,IA),Q2(IB,IB),S(IA,IA)
      DIMENSION C(10,10),EA(20,20),EB(20,20)
C
C     COMPUTATION OF EULERMATRIX
C
      DO 10 I=1,N
      DO 10 J=1,N
      EA(I,J)=-A(I,J)*TD
      C(I,J)=Q2(I,J)
      NPI=N+I
      NPJ=N+J
      EA(NPI,J)=Q1(I,J)*TD
   10 EA(NPI,NPJ)=A(J,I)*TD
C
      CALL GJRV(C,NU,1.0E-008,IERR,10)
      IF(IERR+1) 15,60,15
   15 DO 20 I=1,N
      DO 20 J=1,N
      R=0.0
      DO 21 L=1,NU
      DO 21 M=1,NU
   21 R=R+B(I,L)*C(L,M)*B(J,M)
      NPJ=N+J
   20 EA(I,NPJ)=R*TD
C
C     COMPUTATION OF EB=EXP(EA)
C
      M=N+N
      III=0
      CALL MEXP7T(EA,EB,M,20,III)
C
      GO TO 29
C
      ENTRY ITERATE
C
   29 DO 30 I=1,N
      DO 30 J=1,N
```

```
      NPI=N+I
      R=0.0
      DO 31 K=1,N
      NPK=N+K
   31 R=R+EB(NPI,NPK)*QQ(K,J)
   30 C(I,J)=EB(NPI,J)+R
C
      DO 40 I=1,N
      DO 40 J=1,N
      R=0.0
      DO 41 K=1,N
      NPK=N+K
   41 R=R+EB(I,NPK)*QQ(K,J)
   40 EA(I,J)=EB(I,J)+R
C
      CALL GJRV(EA,N,1.0E-008,IERR,20)
      IF(IERR+1) 45,60,45
   45 DO 50 I=1,N
      DO 50 J=1,N
      R=0.0
      DO 51 K=1,N
   51 R=R+C(I,K)*EA(K,J)
   50 S(I,J)=R
C
   60 RETURN
      END
```

```
      SUBROUTINE GJRV(A,N,EPS,IERR,IA)
C
C     INVERTS ASYMME-RIC MATRICES, HAS EMERGENCY EXIT,
C     REQUIRES N**2+4*N WORDS OF ARRAY STORAGE
C
C     A IS THE NAME OF THE MATRIX TO BE INVERTED
C     N IS THE ORDER OF A
C     EPS IS A VALUE TO BE USED AS A TOLERANCE FOR
C     ACCEPTANCE OF THE SINGULARITY OF A GIVEN MATRIX
C     IERR IS AN INTEGER VARIABLE WHICH WILL CONTAIN ZERO
C     UPON RETURN IF INVERSION IS COMPLETED OR -1 IF SOME
C     PIVOT ELEMENT HAS AN ABSOLUTE VALUE LESS THAN EPS
C     IA IS THE DIMENSION PARAMETER
C     MAXIMUM ORDER OF A=40
C     THE ORIGINAL MATRIX IS DESTROYED
C     IF IERR IS RETURNED =-1 THEN THE INVERSION HAS FAILED
C     OTHERWISE THE RESULTING INVERSE IS PLACED IN A
C
C     SUBROUTINE REQUIRED
C             NONE
C
      DIMENSION A(IA,IA),B(40),C(40),IP(40),IQ(40)
      IERR=0
      DO 140 K=1,N
      PIVOT=0.0
      DO 120 I=K,N
      DO 2 J=K,N
      IF(ABSF(A(I,J))-ABSF(PIVOT)) 2,2,1
    1 PIVOT=A(I,J)
      IP(K)=I
      IQ(K)=J
    2 CONTINUE
  120 CONTINUE
      IF(ABSF(PIVOT)-EPS) 100,100,3
    3 IF(IP(K)-K) 4,6,4
    4 DO 5 J=1,N
      IPX=IP(K)
      Z=A(IPX,J)
      A(IPX,J)=A(K,J)
    5 A(K,J)=Z
    6 IF(IQ(K)-K) 7,9,7
    7 DO 8 I=1,N
      IPX=IQ(K)
      Z=A(I,IPX)
      A(I,IPX)=A(I,K)
    8 A(I,K)=Z
    9 DO 13 J=1,N
      IF(J-K) 11,10,11
   10 B(J)=1.0/PIVOT
      C(J)=1.0
      GO TO 12
   11 B(J)=-A(K,J)/PIVOT
      C(J)=A(J,K)
   12 A(K,J)=0.0
      A(J,K)=0.0
   13 CONTINUE
      DO 14 I=1,N
      DO 14 J=1,N
   14 A(I,J)=A(I,J)+C(I)*B(J)
  140 CONTINUE
      DO 20 KP=1,N
```

```
      K=N+1-KP
      IF(IP(K)-K) 15,17,15
   15 DO 16 I=1,N
      IPX=IP(K)
      Z=A(I,IPX)
      A(I,IPX)=A(I,K)
   16 A(I,K)=Z
   17 IF(IQ(K)-K) 18,20,18
   18 DO 19 J=1,N
      IPX=IQ(K)
      Z=A(IPX,J)
      A(IPX,J)=A(K,J)
   19 A(K,J)=Z
   20 CONTINUE
      GO TO 21
  100 IERR=-1
   21 RETURN
      END
```

```
      SUBROUTINE MEXP7T(A,B,N,IA,NOTRACE)
C
C     COMPUTES B=EXP(A) BY ORIGIN SHIFT AND SERIES EXPANSION USING 7
C     TERMS.
C     AUTHOR,K.MORTENSSON 15/11-67.
C
C     A-NXN-MATRIX.
C     B-NXN-MATRIX.
C     IA-DIMENSION PARAMETER.
C     NOTRACE=0 MEANS THAT NO TRACE
C     COMPUTATION WILL BE PERFORMED.
C     MAXIMUM ORDER OF A AND B=20.
C     THE MATRIX A IS DESTROYED.
C
C     SUBROUTINE REQUIRED
C             NORM
C
      DIMENSION  A(IA,IA),B(IA,IA),C(7,20,20)
      IF(NOTRACE) 1,5,1
    1 TRAA=0.
      DO 2 I=1,N
    2 TRAA=TRAA+A(I,I)
      IF(TRAA) 3,5,3
    3 TRAA=TRAA/N
      DO 4 I=1,N
    4 A(I,I)=A(I,I)-TRAA
    5 KDIV=0
      DO 6 I=1,N
      DO 6 J=1,N
    6 C(1,I,J)=A(I,J)
      DO 10 LOP=2,7
      DO 10 I=1,N
      DO 10 J=1,N
      R=0.
      DO 8 K=1,N
    8 R=R+C(LOP-1,I,K)*A(K,J)
   10 C(LOP,I,J)=R/LOP
   12 DO 14 I=1,N
      DO 14 J=1,N
   14 B(I,J)=C(7,I,J)
      CALL NORM(B,N,IA,P)
      IF(P-1.0E-010) 20,20,16
   16 REST=P*1.0E+010
   15 KDIV=KDIV+1
      RQ=2.0**(KDIV*7)
      IF(RQ-REST) 15,15,17
   17 DO 18 LOP=1,7
      PKVAD=2.0**(KDIV*LOP)
      DO 18 I=1,N
      DO 18 J=1,N
   18 C(LOP,I,J)=C(LOP,I,J)/PKVAD
   20 DO 22 I=1,N
      DO 22 J=1,N
   22 B(I,J)=0.0
      DO 26 I=1,N
   26 B(I,I)=1.0
      DO 28 LOP=1,7
      DO 28 I=1,N
      DO 28 J=1,N
   28 B(I,J)=B(I,J)+C(LOP,I,J)
      IF(KDIV) 46,46,36
```

```
36 DO 44 IPK=1,KOIV
   DO 40 I=1,N
   DO 40 J=1,N
   R=0.
   DO 38 K=1,N
38 R=R+B(I,K)*B(K,J)
40 C(1,I,J)=R
   DO 42 I=1,N
   DO 42 J=1,N
42 B(I,J)=C(1,I,J)
44 CONTINUE
46 IF(NOTRACE) 47,50,47
47 IF(TRAA) 49,50,49
49 CC=EXPF(TRAA)
   DO 48 I=1,N
   DO 48 J=1,N
48 B(I,J)=CC*B(I,J)
50 RETURN
   END
```

```
      SUBROUTINE NORM(A,N,IA,S)
C
C     THIS SUBROUTINE COMPUTES THE MINIMAXNORM OF A WHERE
C     A=NXN-MATRIX
C     S IS THE RESULTING NORM
C     IA IS THE DIMENSION PARAMETER
C
C     SUBROUTINE REQUIRED
C             NONE
C
      DIMENSION A(IA,IA)
      S=S1=0.0
      DO 20 J=1,N
      R=0.0
      DO 10 I=1,N
      R=R+ABSF(A(I,J))
   10 CONTINUE
      IF(R.GT.S1) 15,20
   15 S1=R
   20 CONTINUE
C     S1=MAX OVER THE COLUMNS
      DO 40 I=1,N
      R=0.0
      DO 30 J=1,N
      R=R+ABSF(A(I,J))
   30 CONTINUE
      IF(R.GT.S) 35,40
   35 S=R
   40 CONTINUE
C     S=MAX OVER THE ROWS
C
      IF(S.GT.S1) 50,60
   50 S=S1
   60 RETURN
      END
```

```
      SUBROUTINE EIGUNS(AM,EIGR,EIGI,N,IA,IPR,IERR)
C
C     SUBROUTINE FOR COMPUTING THE EIGENVALUES OF AN ARBITRARY REAL
C     MATRIX BY TRIDIAGONALISATION,DETERMINANT EVALUATION AND MULLER
C     ITERATIVE PROCESS.
C     REFERENCE,F4 UTEX ELIMEVPR
C     AUTHOR,K.MORTENSSON 20/10-67
C
C     AM-NXN-MATRIX WHOSE EIGENVALUES ARE TO BE DETERMINED.
C     EIGR-VECTOR OF DIMENSION N CONTAINING THE REAL PART OF THE
C     EIGENVALUES.
C     EIGI-VECTOR CONTAINIG THE CORRESPONDING IMAGINARY PART.
C     IA-DIMENSION PARAMETER.
C     IPR=1 MEANS THAT PRINTOUTS WILL BE MADE IN THE SUBROUTINE.
C     IPR=0 MEANS THAT NO PRINTOUTS WILL BE MADE.
C     IERR IS RETURNED 0 IF THE SUBROUTINE HAS SUCCEEDED IN FINDING THE
C     EIGENVALUES.IERR=1 MEANS THAT THE TRIDIAGONALISATION HAS FAILED,
C     IERR=2 MEANS THAT THE MULLER METHOD FAILS TO CONVERGE.
C     THE MATRIX AM IS DESTROYED
C
C     SUBROUTINE REQUIRED
C             TRIDH
C             TRIRTMU
C             FUNCT
C             MODFNT
C             CSQRN
C             SCAPRODB(CODAP CODED)
C
      DIMENSION AM(IA,IA),EIGR(IA),EIGI(IA)
      DIMENSION A(20,20),B(20,20),GR(200),GI(200),RTR(20),RTI(20),S(20),
     1IVEC(15)
      COMMON/HHW/B,EP,IPRINT
      COMMON/HH/GR,GI,EF,EI,EPC,ED
      COMMON/HHH/ A,RTR,RTI,S,K,IERROR
C
      IPRINT=IPR
      IERROR=0
      JVT=0
      GR(1)=GI(1)=0.
      GR(2)=GI(2)=GR(3)=0.1
      GI(3)=-0.1
      DO 1 I=1,N
      DO 1 J=1,N
    1 A(I,J)=AM(I,J)
      EF=1.0E-7
      EI=1.0E-14
      EPC=1.0E-6
      ED=1.0E-7
      EP=1.0E+5
      TRAB=0.0
      DO 30 K=1,N
   30 TRAB=TRAB+A(K,K)
      IF(IPR-1) 3,2,3
    2 DO 4 I=1,15
    4 IVEC(I)=40404040404040408
      PRINT 100,(IVEC(I),I=1,15)
  100 FORMAT(1H ,15(R8),/)
      PRINT 101
  101 FORMAT(20X,33H PRINTOUTS FROM SUBROUTINE EIGUNS,/)
      PRINT 220,N
  220 FORMAT(16H MATRIX OF ORDER,I4)
```

```
C       START TO TRANSFORM TO HESSENBERG FORM
      3 I=1
     24 IF(1.GE.N-I)21,22
     22 IPO=I+1
        IPT=I+2
C       PARTIAL PIVOTING
        IMAX=IPO
        DO 11 J=IPT,N
        IF(ABSF(A(IMAX,I)).GE.ABSF(A(J,I)))11,12
     12 IMAX=J
     11 CONTINUE
        IF(IMAX.EQ.IPO)13,14
     14 DO 15 J=I,N
        T=A(IPO,J)
        A(IPO,J)=A(IMAX,J)
     15 A(IMAX,J)=T
        DO 16 J=1,N
        T=A(J,IPO)
        A(J,IPO)=A(J,IMAX)
     16 A(J,IMAX)=T
     13 IF(A(IPO,I).EQ.0.)32,33
     33 DO 17 J=IPT,N
        AM(J,IPO)=A(J,I)/A(IPO,I)
     17 A(J,I)=0.
        AM(IPO,IPO)=1.
        DO 18 J=IPT,N
        DO 18 K=IPO,N
     18 A(J,K)=A(J,K)-AM(J,IPO)*A(IPO,K)
        NMI=N-I
        DO 19 K=1,N
        DO 20 J=IPO,N
     20 S(J)=A(K,J)
        CALL SCAPRODB(S(IPO),AM(IPO,IPO),NMI,A(K,IPO))
     19 CONTINUE
        GO TO 613
     32 DO 612 J=IPT,N
    612 AM(J,IPO)=0.
    613 I=I+1
        GO TO 24
     21 TRAA=0.
        DO 31 K=1,N
     31 TRAA=TRAA+A(K,K)
        IF(IPRINT-1) 6,5,6
      5 PRINT 201,TRAB,TRAA
    201 FORMAT(16H ORIGINAL TRACE=E20.10,29H, TRACE OF HESSENBERG MATRIX=E
       120.10)
      6 CALL TRIDH(N)
        IERR=IERROR
        KKK=IERR+1
        GO TO(7,9,9),KKK
      7 DO 8 I=1,N
        EIGR(I)=RTR(I)
      8 EIGI(I)=RTI(I)
      9 IF(IPRINT-1) 52,50,52
     50 PRINT 102
    102 FORMAT(/,20X,40H END OF PRINTOUTS FROM SUBROUTINE EIGUNS,/)
        PRINT 100,(IVEC(I),I=1,15)
     52 RETURN
        END
```

```
      SUBROUTINE TRIDH(NA)
      DIMENSION A(20,20),B(20,20),BM(20),S(20),RTR(20),RTI(20),GR(200),
     1 GI(200)
C
      COMMON/HHW/B,EP,IPRINT
      COMMON/HHH/ A,RTR,RTI,S,K,IERROR
      N=NA
      DO 1 J=1,N
      DO 1 L=1,N
    1 B(J,L)=A(J,L)
      K=1
      NMO=N-1
      NMT=N-2
      TRA=0.
      MB=0
    7 I=K
   76 IF(1.GE.N-I)30,75
   75 IPO=I+1
      IPT=I+2
      IF(A(IPO,I).EQ.0.)4,5
    4 IF(IPRINT-1) 101,100,101
  100 PRINT 50,I
   50 FORMAT(32H THE MATRIX IS REDUCIBLE AT STEP,I3)
  101 RTR(I)=A(I,I)
      RTI(I)=0.
      TRA=TRA+A(I,I)
      K=K+1
      IF(K.EQ.N-1)30,7
    5 DO 10 J=IPO,N
      IF(A(I,J).EQ.0.)10,12
   10 CONTINUE
      GO TO 4
   12 NMI=N-I
      SUM=ABSF(A(I,IPT))
      DO 19 J=IPT,NMO
      IF(SUM.LT.ABSF(A(I,J+1)))70,19
   70 SUM=ABSF(A(I,J+1))
   19 CONTINUE
      IF(SUM.GE.EP*ABSF(A(I,IPO)))2,3
    2 KPO=K+1
      MB=MB+1
      KN=N-K+1
      IF(MB-2)20,21,22
   20 IF(IPRINT-1) 103,102,103
  102 PRINT 51,I,MB
   51 FORMAT(8H AT STEP,I3,32H PRECONDITION TRANSFORMATION NO.,I2)
  103 DO 23 J=K,N
      DO 23 L=KPO,N
   23 A(J,L)=A(J,L)+A(J,K)
      DO 24 J=K,N
      DO 72 L=KPO,N
   72 S(L)=-1.
      S(K)=1.
   24 CALL SCAPRODB(A(K,J),S(K),KN,A(K,J))
      GO TO 7
   21 IF(IPRINT-1) 111,110,111
  110 PRINT 51,I,MB
  111 DO 25 J=K,N
      DO 25 L=KPO,N
   25 A(J,L)=A(J,L)-A(J,K)
      DO 26 J=K,N
```

```
      DO 73 L=K,N
   73 S(L)=1.
   26 CALL SCAPRODB(A(K,J),S(K),KN,A(K,J))
      GO TO 7
   22 MO=MB-1
      IF(IPRINT-1) 105,104,105
  104 PRINT 52,MO
   52 FORMAT(/I3,56H PRECONDITIONS HAVE BEEN MADE. TRIDIAGONALIZATION FA
     1ILS.)
  105 IERROR=1
      RETURN
    3 DO 27 L=IPT,N
   27 BM(L)=A(I,L)/A(I,IPO)
      BM(IPO)=1.
      DO 28 J=IPO,IPT
      DO 28 L=IPT,N
   28 A(J,L)=A(J,L)-A(J,IPO)*BM(L)
      NM=1
      DO 29 J=IPO,NMO
      NM=NM+1
   29 CALL SCAPRODB(A(IPO,J),BM(IPO),NM,A(IPO,J))
      CALL SCAPRODB(A(IPO,N),BM(IPO),NMI,A(IPO,N))
      IF(I.EQ.NMT)34,31
   31 IF(A(IPT,IPO).EQ.0.)36,37
   36 IF(IPRINT-1) 113,112,113
  112 PRINT 50,IPO
  113 DO 56 J=K,IPO
   56 TRA=TRA+A(J,J)
      GO TO 90
   37 SUM=ABSF(A(IPO,IPT+1))
      DO 32 J=IPT,NMT
      IF(SUM.LT.ABSF(A(IPO,J+2)))71,32
   71 SUM=ABSF(A(IPO,J+2))
   32 CONTINUE
      IF(SUM.GE.EP*ABSF(A(IPO,IPT)))33,34
   33 DO 35 J=IPT,N
      IF(A(IPO,J).EQ.0.)35,39
   35 CONTINUE
      GO TO 36
   39 DO 38 J=K,N
      DO 38 L=K,N
   38 A(J,L)=B(J,L)
      I=I+1
      GO TO 2
   90 DO 91 J=K,I
   91 S(J+1)=A(J,J+1)*A(J+1,J)
      CALL TRIRTMU(IPO,IPRINT)
      IF(IERROR-2) 96,97,96
   97 RETURN
   96 K=IPT
      GO TO 7
   34 I=I+1
      IPO=I+1
      IPT=I+2
      NMI=N-I
      GO TO 76
   30 DO 60 J=K,N
   60 TRA=TRA+A(J,J)
      IF(IPRINT-1) 107,106,107
  106 PRINT 95,TRA
   95 FORMAT(29H TRACE OF TRIDIAGONAL MATRIX=E20.10)
  107 DO 61 J=K,NMO
```

```
 61 S(J+1)=A(J,J+1)*A(J+1,J)
    IF(IPRINT-1) 89,55,89
 55 PRINT 85,(A(J,J),J=1,N)
 85 FORMAT(/23H --TRIDIAGONAL MATRIX--/14H MAIN DIAGONAL/(6E20.10))
    PRINT 86,(A(J-1,J),J=2,N)
 86 FORMAT(14H SUPERDIAGONAL/(6E20.10))
    PRINT 62,(A(J,J-1),J=2,N)
 62 FORMAT(12H SUBDIAGONAL/(6E20.10))
 89 CALL TRIRTMU(N,IPRINT)
    IF(IERROR-2) 93,94,93
 94 RETURN
 93 SUMR=0.
    SUMI=0.
    DO 87 J=1,N
    SUMR=SUMR+RTR(J)
 87 SUMI=SUMI+RTI(J)
    IF(IPRINT-1) 109,108,109
108 PRINT 88,SUMR,SUMI
 88 FORMAT(19H SUM OF EIGENVALUES,2E20.10)
109 RETURN
    END
```

```
      SUBROUTINE TRIRTMU(NINPUT,IPRINT)
C     SUBROUTINE FOR COMPUTING THE EIGENVALUES OF THE TRIDIAGONAL
C     MATRIX BY DETERMINANT EVALUATION AND THE MULLER ITERATION PROCESS.
      DIMENSION FR(200),FI(200),A(20,20),GR(200),GI(200),RTR(20),RTI(20)
     1,S(20)
      COMMON/HH/GR,GI,EF,EI,EPC,ED
      COMMON/HHH/ A,RTR,RTI,S,K,IERROR
      N=NINPUT
      SUM=A(K,K)**2
      KPO=K+1
      DO 600 I=KPO,N
  600 SUM=SUM+A(I,I)**2+A(I-1,I)**2+A(I,I-1)**2
      EMAX=SUM*EF
      NM=K-1
      NPLO=NM+1
  760 KK=1
      DO 750 I=1,3
      CALL FUNCT(N,GR(I),GI(I),TEMR,TEMI)
      IF(ABSF(TEMR)+ABSF(TEMI))751,751,799
  799 CALL MODFNT(NM,TEMR,TEMI,GR(I),GI(I),FR(I),FI(I))
  750 CONTINUE
      NG=1
      NGT=NG+2
  758 IF(NG-200)780,781,781
  781 IF(IPRINT-1) 101,100,101
  100 PRINT 805
  805 FORMAT(21H MULLER METHOD FAILS./)
  101 IERROR=2
      RETURN
  780 NGO=NG+1
      ABR=GR(NG)-GR(NGT)
      ABI=GI(NG)-GI(NGT)
      BBR=GR(NGO)-GR(NGT)
      BBI=GI(NGO)-GI(NGT)
      DAR=FR(NG)-FR(NGT)
      DAI=FI(NG)-FI(NGT)
      DBR=FR(NGO)-FR(NGT)
      DBI=FI(NGO)-FI(NGT)
      BDAR=BBR*DAR-BBI*DAI
      BDAI=BBR*DAI+BBI*DAR
      ADBR=ABR*DBR-ABI*DBI
      ADBI=ABR*DBI+ABI*DBR
      UNR=BDAR-ADBR
      UNI=BDAI-ADBI
      ABBR=ABR*BBR-ABI*BBI
      ABBI=ABR*BBI+ABI*BBR
      AMBR=ABR-BBR
      AMBI=ABI-BBI
      DENR=ABBR*AMBR-ABBI*AMBI
      DENI=ABBR*AMBI+ABBI*AMBR
      DEN=DENR**2+DENI**2
      ASDBR=ABR*ADBR-ABI*ADBI
      ASDBI=ABR*ADBI+ABI*ADBR
      BSDAR=BBR*BDAR-BBI*BDAI
      BSDAI=BBR*BDAI+BBI*BDAR
      BNUR=ASDBR-BSDAR
      BNUI=ASDBI-BSDAI
      CAR=(UNR*DENR+UNI*DENI)/DEN
      CAI=(UNI*DENR-UNR*DENI)/DEN
      CBR=(BNUR*DENR+BNUI*DENI)/DEN
      CBI=(BNUI*DENR-BNUR*DENI)/DEN
```

```
      CCR=FR(NGT)
      CCI=FI(NGT)
      CBSR=CBR*CBR-CBI*CBI
      CBSI=2.*CBR*CBI
      FACR=4.*(CAR*CCR-CAI*CCI)
      FACI=4.*(CAR*CCI+CAI*CCR)
      QDR=CBSR-FACR
      QDI=CBSI-FACI
      CALL CSQRN(QDR,QDI,CDR,CDI)
      IF((-CBR)*CDR+(-CBI)*CDI)752,753,753
  752 DDR=-CBR-CDR
      DDI=-CBI-CDI
      GO TO 754
  753 DDR=CDR-CBR
      DDI=CDI-CBI
  754 DD=DDR**2+DDI**2
      DBR=2.*(CCR*DDR+CCI*DDI)/DD
      DBI=2.*(CCI*DDR-CCR*DDI)/DD
      DB=DBR**2+DBI**2
      CM=GR(NGT)**2+GI(NGT)**2
      IF(EMAX-CM)610,610,611
  611 CM=EMAX
  610 GR(NGT+1)=GR(NGT)+DBR
      GI(NGT+1)=GI(NGT)+DBI
      NGT=NGT+1
      ABS=ABSF(GR(NGT))+ABSF(GI(NGT))
      IF(ABSF(GR(NGT)-ABS*ED)785,786,786
  785 GR(NGT)=0.
      GO TO 787
  786 IF(ABSF(GI(NGT))-ABS*ED)788,787,787
  788 GI(NGT)=0.
  787 IF(DB-EI*CM)765,756,756
  756 CALL FUNCT(N,GR(NGT),GI(NGT),TEMPR,TEMPI)
      IF(ABSF(TEMPR)+ABSF(TEMPI))765,765,766
  766 CALL MODFNT(NM,TEMPR,TEMPI,GR(NGT),GI(NGT),FR(NGT),FI(NGT))
      NG=NG+1
      GO TO 758
  751 NGT=I
      NIT=0
      GO TO 500
  765 NIT=NGT
  500 RTR(NPLO)=GR(NGT)
      RTI(NPLO)=GI(NGT)
      IF(IPRINT-1) 759,102,759
  102 PRINT 801,NPLO,RTR(NPLO),RTI(NPLO),NIT
  801 FORMAT(15H EIGENVALUE NO.,I3,1H=2E20.10,3H, (I4,12H ITERATIONS))
  759 NM=NM+1
      NPLO=NM+1
      IF(NM-N)770,761,761
  770 ABSO=ABSF(GR(NGT))+ABSF(GI(NGT))
      IF(ABSO-EI)797,797,794
  794 GO TO (795,797),KK
  795 KK=2
      IF(ABSF(GI(NGT))/ABSO-EPC)797,797,796
  796 GI(NGT)=-GI(NGT)
      NIT=0
      GO TO 500
  797 TER = GR(NGT) * 1.001 + .001
      TEI = GI(NGT) * 1.001 + .001
      GR(1)=TER*.75
      GR(2)=TER
      GR(3)=TER*1.25
```

```
      GI(1)=TEI*.75
      GI(2)=TEI
      GI(3)=TEI*1.25
      GO TO 760
  761 CONTINUE
      END
```

```
      SUBROUTINE FUNCT(NA,AGR,AGI,FNR,FNI)
C     SUBROUTINE FOR COMPUTING THE CHARACTERISTIC DETERMINANT OF A
C        TRIDIAGONAL MATRIX
      DIMENSION A(20,20),RTR(20),RTI(20),S(20)
      COMMON/HHH/ A,RTR,RTI,S,K,IERROR
      POR=1.
      POI=0.
      PTR=A(K,K)-AGR
      PTI=-AGI
      KPO=K+1
      DO 3 I=KPO,NA
      TI=A(I,I)-AGR
      TTR=TI*PTR+AGI*PTI
      TTI=TI*PTI-AGI*PTR
      FNR=TTR-S(I)*POR
      FNI=TTI-S(I)*POI
      POR=PTR
      POI=PTI
      PTR=FNR
      PTI=FNI
    3 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MODFNT(NR,FNR,FNI,AGR,AGI,FUNR,FUNI)
C     SUBROUTINE FOR EVALUATING THE FUNCTION VALUES FOR USE IN THE MULLER
C        PROCESS
      DIMENSION A(20,20),RTR(20),RTI(20),S(20)
      COMMON/HHH/ A,RTR,RTI,S,K,IERROR
      TR=1.
      TI=0.
      DO 1 I=K,NR
      WR=TR
      WI=TI
      TTR=RTR(I)-AGR
      TTI=RTI(I)-AGI
      TR=WR*TTR-WI*TTI
    1 TI=WR*TTI+WI*TTR
      WR=TR*TR+TI*TI
      FUNR=(FNR*TR+FNI*TI)/WR
      FUNI=(FNI*TR-FNR*TI)/WR
      END
```

```
      SUBROUTINE CSQRN(XR,XI,YR,YI)
C     SUBROUTINE FOR EXTRACTING THE SQUARE ROOT OF A COMPLEX NUMBER
      IF(XR*XI)1,2,1
    2 IF(XR)3,4,3
    3 IF(XR)5,5,6
    5 YR=0.
      YI=SQRTF(ABSF(XR))
      GO TO 10
    6 YR=SQRTF(XR)
      YI=0.
      GO TO 10
    4 IF(XI)7,8,7
    8 YR=0.
      YI=0.
      GO TO 10
    7 YR=SQRTF(ABSF(XI)/2.)
      IF(XI)9,11,11
    9 YI=-YR
      GO TO 10
   11 YI=YR
      GO TO 10
    1 UR=ABSF(XR)+SQRTF(XR*XR+XI*XI)
      U=SQRTF(2.*UR)
      IF(XR)12,13,13
   12 YR=XI/U
      YI=U/2.
      GO TO 10
   13 YR=U/2.
      YI=XI/U
   10 CONTINUE
      END
```

```
          IDENT     SCAPRODB
          CODAP
          REM       AN ACCURATE REAL SCALER PRODUCT SUBROUTINE.
          ENTRY     SCAPRODB
SCAPRODB  SLJ       **
+         LDA       SCAPRODB
          INA       1
          STA       =STEMP1
          LDA   B7  TEMP1
          SAU       POWERB
          SAL       RECORD
          ALS       24
          SAU       CONTINUE
          SAU       POWERA
          RAO       TEMP1
          INA       1
          SAL       STORE+1
          LDA   B7  TEMP1
          SAU       STORE
          ALS       24
          SAL       LOOP
          SIL   B1  S ORE
          SIU   B2  STORE+1
          ENA       0                    INITIALIZE
          STA       =SSUMU                 LOCATIONS.
          STA       =SSUML                 *
          STA       =SS
          ENI   B1  0                     *
LOOP      ENA   B1  0                    TEST FOR
          SUB       **                     COMPLETION OF SUMMATION
          AJP   M   CONTINUE                 A(1)*B(1)+...+A(N)*B(N).
          SLJ       FINISH
CONTINUE  LDA   B1  **                    EXTRACT
          AJP   P   F                       MANTISSA
          SST       =03777000000000000        OF A(I)
+         SLJ       G                           IN FIXED=POINT,
F         SCL       =03777000000000000            FRACTIONAL
G         ALS       11                              FORMAT.
          AJP   N   RECORD                  TEST A(I) FOR ZERO.
          INI   B1  1                       A(I)=0. THEN
          SLJ       LOOP                      A(I)*B(I)=0.
RECORD    STA       TEMP1                   MANTISSA OF A(I).
          LDA   B1  **                      EXTRACT
          AJP   P   J                         MANTISSA
          SST       =03777000000000000          OF B(I)
          SLJ       K                             IN FIXED-POINT
J         SCL       =03777000000000000              FRACTIONAL
K         ALS       11                                FORMAT.
          AJP   N   MULTIPLY                TEST B(I) FOR ZERO.
          INI   B1  1                       B(I)=0. THEN
          SLJ       LOOP                      A(I)*B(I)=0.
MULTIPLY  MUF       TEMP1                   MULTIPLY THE MANTISSAS TOGETHER.
          SCO   B2  1                       SCALE THE DOUBLE-LENGTH PRODUCT.
          STA       =SCFU                   NORMALIZED DOUBLE-LENGTH
          STQ       =SCFL                     MANTISSA OF A(I)*B(I).
POWERA    LDA   B1  **                      EXTRACT
          AJP   P   E                         EXPONENT
          SCM       =07777777777777777          OF A(I)
E         ALS       1                             IN INTEGER FORMAT.
          SCM       =04000000000000000            *
          ARS       37                            *
```

```
         STA        TEMP1                    *                                 6
POWERB   LDA     B1 **                     EXTRACT                             6
         AJP     P  V                        EXPONENT                          6
         SCM        =07777777777777777         OF B(I)                         6
V        ALS        1                            IN INTEGER FORMAT.            6
         SCM        =04000000000000000       *                                 6
         ARS        37                       *                                 6
         ADD        TEMP1                  COMPUTE SUM                         6
         STA        TEMP1                  OF THE EXPONENS.                    6
         ENA     B2 0                      ADJUST                              6
         INA        -1                       EXPONENT                          6
         ADD        TEMP1                      DUE TO THE NORMALIZATION.       6
         STA        =SR                    CORRECT EXPONENT OF A(I)*B(I).      6
         LDA        SUMU                   TEST THE PARTIAL                    6
         AJP     N  NOTZERO                  SUM FOR ZERO.                     6
         LDA        CFU                    THE PARTIAL SUM                     6
         LDQ        CFL                      A(1)*B(1)+...A(I-1)*B(I-1)=0.     6
         STA        SUMU                       THEN THE PARTIAL SUM A(I)*B(I)  6
         STQ        SUML                         +...+A(I)*B(I)=A(I)*B(I).     6
         LDA        R                        *                                 6
         STA        S                        *                                 6
         INI     B1 1                        *                                 6
         SLJ        LOOP                     *                                 6
NOTZERO  LDA        S                      COMPARE S AND R                     6
         SUB        R                        BY FORMING S-R.                   6
         STA        TEMP1                      S-R IN TEMP1.                   6
         AJP     N  NOTEQUAL               IF S-R=0, THEN IT IS                6
         LDA        CFU                      NOT NECESSARY                     6
         LDQ        CFL                        TO ALIGN                        6
         SLJ        ALIGNED                      BINARY POINT.                 6
NOTEQUAL AJP     P  NOCHANGE               IF S IS LESS THAN R,                6
         LAC        TEMP1                    INTERCHANGE A(I)*B(I)             6
         STA        TEMP1                      AND THE PARTIAL SUM.            6
         LDA        SUMU                   INTERCHANGE CFU(UPPER               6
         LDQ        CFU                      HALF MANTISSA OF A(I)*B(I))       6
         STA        CFU                        AND SUMU(UPPER HALF             6
         STQ        SUMU                         MANTISSA OF THE PARTIAL SUM)  6
         LDA        SUML                   INTERCHANGE CFL(LOWER               6
         LDQ        CFL                      HALF MANTISSA OF A(I)*B(I))       6
         STA        CFL                        AND SUML(LOWER HALF MANTISSA    6
         STQ        SUML                         OF THE PARTIAL SUM).          6
         LDA        R                      INTERCHANGE                         6
         LDQ        S                        R(EXPONENT OF A(I)*B(I))          6
         STA        S                          AND S(EXPONENT                  6
         STQ        R                            OF THE PARTIAL SUM).          6
NOCHANGE ENA        91                     CHECK SHIFT COUNT TO BE             6
         SUB        TEMP1                    NECESSARY TO ALIGN BINARY         6
         AJP     P  ALIGN                      POINT BEFORE SHIFTING. IF IT    6
         INI     B1 1                            IS LARGER THAN 91, A(I)*B(I)  6
         SLJ        LOOP                           IS TOO SMALL.               6
ALIGN    LDA        CFU                    ALIGN BINARY                        6
         LDQ        CFL                      POINT BY SHIFTING                 6
         LRS      7 TEMP1                      A(I)*B(I) RIGHT BY S-R PLACES.  6
ALIGNED  LRS        2                      GIVE TWO MORE SIGN BITS.            6
         SCL        =04000000000000000     PLACE 0 AT THE                      6
         STA        CFU                      BEGINNING OF UPPER HALF.          6
         ENA        0                      PLACE 0 AT THE                      6
         LRS        1                        BEGINNING OF                      6
         STQ        CFL                        LOWER HALF.                     6
         LDA        SUMU                                                       6
         LDQ        SUML                                                       6
         LRS        2                      GIVE TWO MORE SIGN BITS.            6
```

```
          SCL       =0400000000000000   PLACE 0 AT THE                          6
          STA       SUMU                  BEGINNING OF UPPER HALF.              6
          ENA       0                   PLACE 0                                 6
          LRS       1                     AT THE BEGINNING                      6
          STQ       SUML                    OF LOWER HALF.                      6
STARTADD  LDA       SUML                ADD                                     6
          ADD       CFL                   LOWER HALVES.                         6
          STA       TEMP1                                                       6
          LDQ       TEMP1                                                       6
          ENA       0                   CLEAR A-REG AND                         6
          LLS       1                     BRING POSSIBLE CARRY INTO A-REG.      6
          ADD       SUMU                ADD CARRY                               6
          ADD       CFU                   AND UPPER HALVES.                     6
          AJP    P  NOCARRY             TEST FOR END-AROUND CARRY.              6
          STA       =STEMP2             END-AROUND                              6
          ENA       2                     CARRY EXISTS.                         6
          LRS       1                       MAKE AN                             6
          STQ       TEMP1                     END-AROUND                        6
          ADD       TEMP1                       CARRY.                          6
          STA       TEMP1                       °                               6
          LDQ       TEMP1                       °                               6
          ENA       0                           °                               6
          LLS       1                           °                               7
          ADD       TEMP2                       °                               7
NOCARRY   LLS       1                   SIGN BIT AT THE BEGINNING OF A≠REG      7
          LRS       2                   THESE TWO OPERATIONS SUPPLY             7
          LLS       2                     CORRECT FILLER BITS(SIGN BITS).       7
          SCQ    B2 95                  NORMALIZE THE                           7
          STA       SUMU                  DOUBLE-LENGTH                         7
          STQ       SUML                    SUM.                                7
          ENA    B2 0                   ADJUST                                  7
          INA       -94                   EXPONENT                              7
          ADD       S                       DUE TO                              7
          STA       S                         THE NORMALIZATION.                7
          INI    B1 1                   A(1)∘B(1)+...+A(I)∘B(I) FORMED.         7
          SLJ       LOOP                  REPLACE I BY I+1. BACK TO LOOP.       7
FINISH    LDA       SUMU                THE SCALER PRODUCT                      7
          AJP    Z  ZERO                                                        7
          AJP    P  PLUS                 A(1)∘B(1)+...A(N)∘B(N) FORMED.        7
          LAC       SUMU                  ROUND OFF THE DOUBLE≠                 7
PLUS      ARS       1                       LENGTH ANSWER                       7
          INA       1000B                     TO SINGLE≠LENGTH(36               7
          SCA    B2 1                           BITS MANTISSA).                  7
+         SSK       SUMU                        °                               7
          SLJ       NOCOMP                      °                               7
+         SCM       =077777777777777777         °                               7
NOCOMP    ARS       11                          °                               7
          SCL       =037770000000000000         °                               7
          STA       SUMU                        °                               7
          ENA    B2 0                   ADJUST EXPONENT                         7
          ADD       S                     DUE TO THE ROUND-OFF.                 7
          ALS       36                  REPACK THE                              7
+         SSK       SUMU                  EXPONENT                              7
          SLJ       NOTCOMP                 AND THE MANTISSA                    7
+         SCM       =077777777777777777       INTO 48-BITS STANDARD             7
NOTCOMP   SCM       =020000000000000000         FLOATING-POINT FORMAT.          7
          SCL       =040007777777777777         °                               7
          ADD       SUMU                        °                               7
STORE     STA       °°                  STORE THE ANSWER.                       7
          ENI    B1 °°                                                          7
          ENI    B2 °°                                                          7
          SLJ       °°                  NORMAL RETURN.                          7
```

```
ZERO    ENA     0                                                    74
        SLJ     STORE                                                74
        END                                                          74
```