

LUND INSTITUTE OF TECHNOLOGY DEPARTMENT OF AUTOMATIC CONTROL Box 725 S 220 07 Lund 7 Sweden		Document name Report
		Date of issue June 1983
		Document number CODEN:LUTFD2/(TFRT-5300)/1-049/(1983)
Author(s) Klas Falkvall	Supervisor Joe Wall, Günter Stein, Björn Wittenmark.	
	Sponsoring organization	
Title and subtitle LQG-controller for an aircraft (LQG-regulator för flygplan)		
Abstract A method of design for multivariable systems is presented. It is used in an application design of a controller for an advanced aircraft. The main design criteria, singular values plot, is discussed in detail. A must for the use of the method is that computer assistance is available. Two different interactive program packages allowing LQG-design are presented and somewhat compared.		
Key words		
Classification system and/or index terms (if any)		
Supplementary bibliographical information		
ISSN and key title		ISBN
Language Swedish	Number of pages 49	Recipient's notes
Security classification		

DOCUMENT DATABL ID RT 3/81

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

Abstract

A method of design for multivariable systems is presented. It is used in an application design of a controller for an advanced aircraft. The main design criteria, singular values plot, is discussed in detail. A must for the use of the method is that computer assistance is available. Two different interactive program packages allowing LQG-design are presented and somewhat compared.

Innehållsförteckning

	sidan
Inledning	3
Något allmänt om flygplan	4
FSW-flygplanet	7
LQG-design	9
Modellbeskrivning	10
Singulara värde	12
Exempel på användning av singulara värde	14
LQG	16
LQG-konstruktion	18
Exempel på hur LQG-metoden kan användas	18
CAD-verktyg	21
Grundläggande algoritmer	21
Interaktiva programpaket	22
HoneyX	22
LQG-kommando i HoneyX	25
Synpac	27
Liten jämförelse HoneyX-Synpac	29
Tillämpningsexempel	30
Sammanfattning	32
Referenser	33
Bilagor 1-15	

Inledning

I det första avsnittet beskrivs några allmänna teorier som är speciella för flygplan. Dels redogörs för problem relaterade till flygplan, dels redogörs för de matematiska uppställningarna. I avsnittet FSW-flygplanet beskrivs det flygplan till vilket tillämpningsexemplet är relaterat.

LQG-designmetoden beskrivs i det efterföljande avsnittet. De problem som är unika för ett flervariabelt system, samt varför man använder återkoppling i LQG-konstruktion är omnämnda. Singulära värdet, som är ett storleksmått för flervariabla system, redogörs i detalj för, och en beskrivning av hur det kan användas ges i ett exempel.

De bakomliggande teorierna för LQG-konstruktion beskrivs i avsnittet LQG. Den matematiska grunden samt hur de olika stegen i konstruktionen beskrivs och belyses i ett exempel.

LQG-metoden förutsätter datorhjälp p.g.a. de omfattande räkneoperationerna. Två olika interaktiva programpaket, HoneyX och Synpac presenteras. Dessutom nämns de grundläggande algoritmer som särskilt behövs för LQG-konstruktion speciellt. I slutet av avsnittet görs en liten jämförelse mellan de två programpaketerna.

Det tidigare nämnda tillämpningsexemplet följer i det efterföljande avsnittet. Detta belyser hur en konstruktion av en regulator till ett flygplan kan gå till. Exemplet förtydligas med ett antal kurvor som bilägges.

Rapporten avslutas med en kort sammanfattning av densamma.

Något allmänt om flygplan

Inledning

Flygplan, speciellt de som är byggda för att uppträda i stridssituationer, är komplicerade att konstruera regulatorer för. Denna sorts flygplan skall kortfattat klara följande moment: Start, navigationsflygning till plats, strid, återflygning, inflygning samt landning. Det är ett antal parametrar, som man måste ta med i beräkningen vid formuleringen av de styrlagar, med vilkas hjälp piloten skall kunna framföra och använda flygplanet. Önskat eller nödvändigt uppträdande i luften, planets stabilitet eller hur det skall kunna kontrolleras är ett par exempel. Dessutom finns det några speciella problem relaterade till flygplan som man måste ta hänsyn till:

- a) Den komplexa påverkan luften har på vingarna.
- b) Det faktum att förflyttningar måste kunna ske mycket snabbt. Detta ställer stora krav på piloten.
- c) Att farkosten kan förflytta sig i alla tre dimensionerna samtidigt.

Ovanstående medför att flygplan numera styrs av datorer kombinerat med servomekanismer, där hänsyn bl.a. tas till olika flygtillstånd, till pilotens inverkan på flygplanet samt till pilotens fysiska och psykiska begränsningar. Konkreta faktorer som medfört att servomekanismer används är:

De nödvändiga krafterna för att kontrollera flygplanet har blivit stora.
Svarstiderna har blivit kortare.
Man önskar större svarsnoggrannhet.
På grund av de högre hastigheterna ändras flygplanets dynamiska karaktäristika fortare.

Några definitioner

En del uttryck används då man diskuterar ett flygplans olika prestanda och egenskaper.

Uppträdande

Kännetecknande för planets uppträdande i luften är maximal hastighet, högsta flyghöjd samt aktionsradie. Svarstiderna blir långa; minuter till timmar. Kraftekvationerna spelar störst roll.

Flygkvalitet

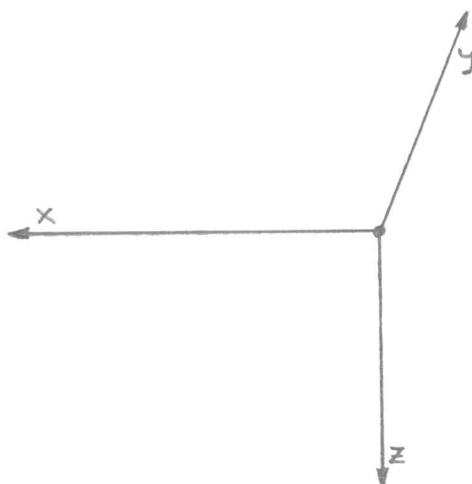
Detta omfattar stabilitet hos och kontroll av planet. Stabilitet betyder att flygplanet återvänder till sitt jämviktsläge efter en störning, flygning rakt fram med vingarna i samma nivå, om det är stabilt. Kontroll betyder att man kan styra flygplanet längs en godtycklig flygväg, att man kan ändra eller fixera jämviktsvillkor samt att flygplanet kan manövreras. På vilket sätt man kan förflytta planet till olika jämviktpunkter beror på om dess uppförande medger det.

Uppförande och kontroll

För dessa egenskaper är de karaktäristiska tiderna korta; motsvarande sekunder. Det är momentekvationerna som är de avgörande.

Rörelselagar och moment

Vid uppställandet av rörelselagar och momentekvationer måste ett koordinatsystem definieras. Detta görs i enlighet med figur 1.



Figur 1

Nollpunkten är placerad i flygplanets masscentrum. Flygplanet betraktas som en stel kropp.

Man säger att rörelseekvationerna relaterar rörelsen hos masscentrum till de externa krafterna, och att momentekvationerna relaterar rotationen kring masscentrum till de externa momenten. Vidare indelas ekvationerna i symmetriska och asymmetriska, eller longitudinella respektive latitudinella. Symmetriska ekvationer betyder att

de beskriver de rörelser som flygplanet gör när de båda vingarna förflyttar sig på ett likformigt sätt i luften.

Rörelselagarna fås genom att använda Newtons lagar för rörelse:

$$F_x = m(\ddot{u} + \omega q - v r)$$

$$F_y = m(\ddot{v} + u r - \omega p)$$

$$F_z = m(\ddot{w} + v p - u q)$$

Där

m = flygplanets massa (antas konstant under någon speciell dynamisk analys).

u, v, w är hastighetskomponenterna i respektive x, y, z -riktningarna. Nominell hastighet $U = \sqrt{u^2 + v^2 + w^2}$.

p, q, r är roll-, stignings- och gir- vinkelhastighetskomponenter.

Jorden är en tröghetsreferens och i det sammanhanget antas atmosfären vara fixerad till jorden.

(symmetrisk), och en för horisontell (asymmetrisk) rörelse. För den vertikala rörelsen används en modell enligt bilaga 1. Olika tillstånd, in- och utsignaler som är relaterade till den modellen framgår av bilaga 2. Vidare arbetar man med 10 olika flygtillstånd. Dessa beror på hastighet och på aktuell flyghöjd.

Modellen för vertikala rörelser används senare i ett tillämpningsexempel, som beskriver hur konstruktion av en regulator till ett sådant här flygplan kan gå till.

LQG-design

Flervariabla system

LQG - Linear Quadratic Gaussian - är en konstruktionsmetod som kan användas till flervariabla system. Dessa kännetecknas av att man har mer än en insignal och/eller mer än en utsignal i sitt system. Ett par företeelser uppträder hos flervariabla system som inte har någon motsvarighet hos Single-Input/Single-Output-system. I de flervariabla har man en slags användningskoppling: vissa givare brusar mer än andra, ställdon överstyrs olika, störningar uppträder bara i vissa återkopplingar osv. Vidare påverkas olika återkopplingar på olika sätt då någon åtgärd vidtas. Ibland påverkas bara en återkoppling, ibland påverkas flera. Detta medför att en analys av ett flervariabelt system återkoppling för återkoppling, vanligtvis inte ger t.ex. samma stabilitetsmarginaler som en analys av hela systemet på en gång skulle ge. I MIMO-system får man dessutom tillgripa ett annorlunda sätt att representera storlek på.

Återkoppling

Då man använder sig av LQG-metoden är det vanligt att man gör en återkoppling från utsignal till insignal. Det finns ett antal orsaker till att man gör detta:

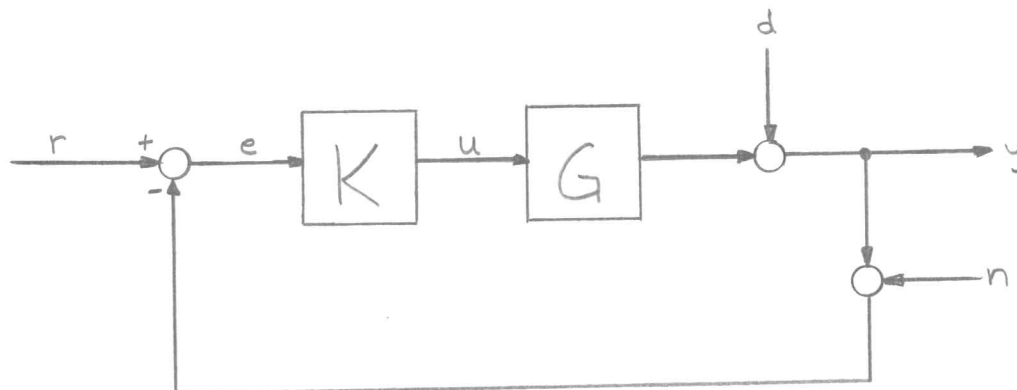
Man vill minska den inverkan skillnader mellan modell och det verkliga systemet har på resultatet. Modellen kan sällan bli exakt jämfört med verkligheten, och eftersom skillnaden (osäkerheten) är svår att bestämma mer än till sin storlek, mätt med något storleksmått, vill man gardera sig för de eventuella skillnaderna.

Man vill att inverkan från olika störningar skall vara så liten som möjligt. Om man önskar låg känslighet för störningar (stor störningsdämpning), kan detta dock inte erhållas vid de frekvenser då man önskar stora stabilitetsmarginaler.

Stabilitetsmarginaler vill man ha så att systemet inte blir instabilt. Man talar om att systemet skall vara robust. Robustheten är ett mått på hur stor en godtycklig störning kan tillåtas bli innan systemet blir instabilt.

Modellbeskrivning

Följande linjära, tidsinvarianta, ändligtdimensionella modell används.



Figur 3

I ett flervariabelt system ges förhållandet mellan det öppna och det slutna återkopplade systemet av $(I+GK)^{-1}$. Vi har vidare följande matematiska samband:

$$y = GK(I+GK)^{-1}(r-n) + (I+GK)^{-1}d$$

$$e = r - y = (I+GK)^{-1}(r-d) + GK(I+GK)^{-1}n$$

Detta innebär att när väl modellen G av det verkliga systemet är fastställd, och storleken på ostrukturerade felkällor är bestämd, så blir återkopplingsproblemet att konstruera en regulator så att:

i) $GK(I+GK)^{-1}$ är stabilt

ii) Ett stört system $G'K(I+G'K)^{-1}$ ($G'=G+\Delta G$ där ΔG är begränsad i storlek) är stabilt för alla möjliga G' tillåtna av de restriktioner som man har på felkällornas storlek.

iii) Systemet uppför sig på det sätt man önskar för alla G' .

Detta kan i sin tur översättas till frekvensvillkor på överföringsfunktionen GK , som konstruktionen måste tillfredsställa. Man vill ha en stor överföringsfunktion där kommandosignaler och/eller störningar är stora, och en liten överföringsfunktion där modellfelen dominerar. I övergångsområdet vill man ha bibehållen stabilitet och ett gott uppförande hos systemet, samt ett okänsligt system.

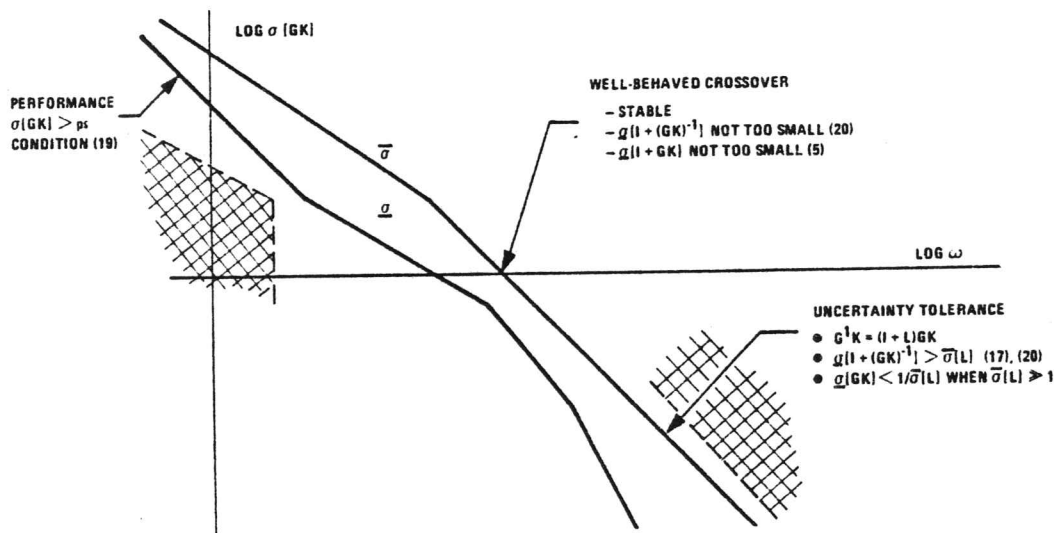


Figure 4. The Multivariable Feedback Design Problem

Figur 4

Kvaliteten på den modell man har bestäms av hur nära dess reaktioner på insignaler stämmer med den verkliga uppställningens. Skillnader måste man ta hänsyn till i sin modellkonstruktion. Vissa gränser för hur modellen tillåts reagera på olika signalers storlek och frekvens bör fastställas, så att man kan bestämma storleken hos den osäkerhet man har hos sin modell.

Hur denna skall representeras beror på de fel som uppstår i modellen. Speciellt vid låga frekvenser är det möjligt att representera osäkerheterna väl. Felen ökar dock med frekvensen. Känner konstruktören till noggrannheten hos sin modell vid olika frekvenser, så kan han ta med detta i designarbetet så att man får ett system som fungerar även i verkligheten.

Singulära värde

Som tidigare nämnts måste man representera storlek på ett annat sätt i flervariabla system än i SISO-system. Som en motsvarighet till Bode-diagram används σ -kurvor, som är en generalisering av en frekvenskurva. Med dessas hjälp kan man avgöra en del systemegenskaper t.ex storleken hos stabilitetsmarginaler, bandbredd och hur mycket känsligheten för störningar blir reducerad. Det singulära värdet för en matris A , σ , mäter hur nära matrisen är att bli singulär. Singulära värdetfunktioner är inte analytiska, medan däremot egenvärdetfunktioner är det. Därför har man förhållandet att $\underline{\sigma}(A) \leq \lambda(A) \leq \bar{\sigma}(A)$. Här betyder $\underline{\sigma}(A)$ minsta singulära, och $\bar{\sigma}(A)$ det största singulära värdet hos A . $\lambda(A)$ betyder egenvärdet av A .

Definition

$$\bar{\sigma}(A) = \max \|Ax\| \sqrt{\lambda_{\max}(A^*A)} \text{ där } \|x\|=1$$

$$\underline{\sigma}(A) = \min \|Ax\| \sqrt{\lambda_{\min}(A^*A)} \text{ där } \|x\|=1$$

(A^*A) är symmetrisk, vilket medför att man har positiva egenvärde.

Här betyder $\| \cdot \|$ normen av någon vektor. Det är ett storleksmått som definieras som roten ur skalärprodukten av en storhet multiplicerad med sig själv. $\lambda(\cdot)$ betyder egenvärde och $(\cdot)^*$ betyder konjugat-transponat.

Singulära värdet av GK, enligt modellen tidigare, d.v.s. $\sigma(GK)$ kan följdaktligen bestämmas och därvid måste vissa övervägande göras. Fel p.g.a. r och d blir små om $\underline{\sigma}(GK)$ är stort, då blir dock fel p.g.a. n stora. Man fastställer då en positiv, skalär funktion ℓ_a som anger den tillåtna

storleken hos modellosäkerheten och störningarna. Om då $G' = G + \Delta G$ så är G' begränsad i en omgivning till G om $\bar{\sigma}(\Delta G) < \ell_a(\omega)$.

De singulära värdena (maximala och minimala) för GK , $I+GK$ och $GK(I+GK)^{-1}$ kan uppritas som en funktion av frekvensen för analys av egenskaperna hos ett flervariabelt system. Härigenom kan systemet i sin helhet analyseras. Det är speciellt viktigt att $GK(I+GK)^{-1} < 1/\lambda$, där λ beskriver storleken hos de störningar som^m uppträder. Gäller ovanstående är systemet stabilt. För att få ett okänsligt system, så vill man ha $1/\sigma((I+GK)^{-1})$ eller $1/g(I+GK)$ inom önskat frekvensområde.

Exempel på användning av singulära värde

Vi har en lineariserad modell av en CH-47 helikopter.

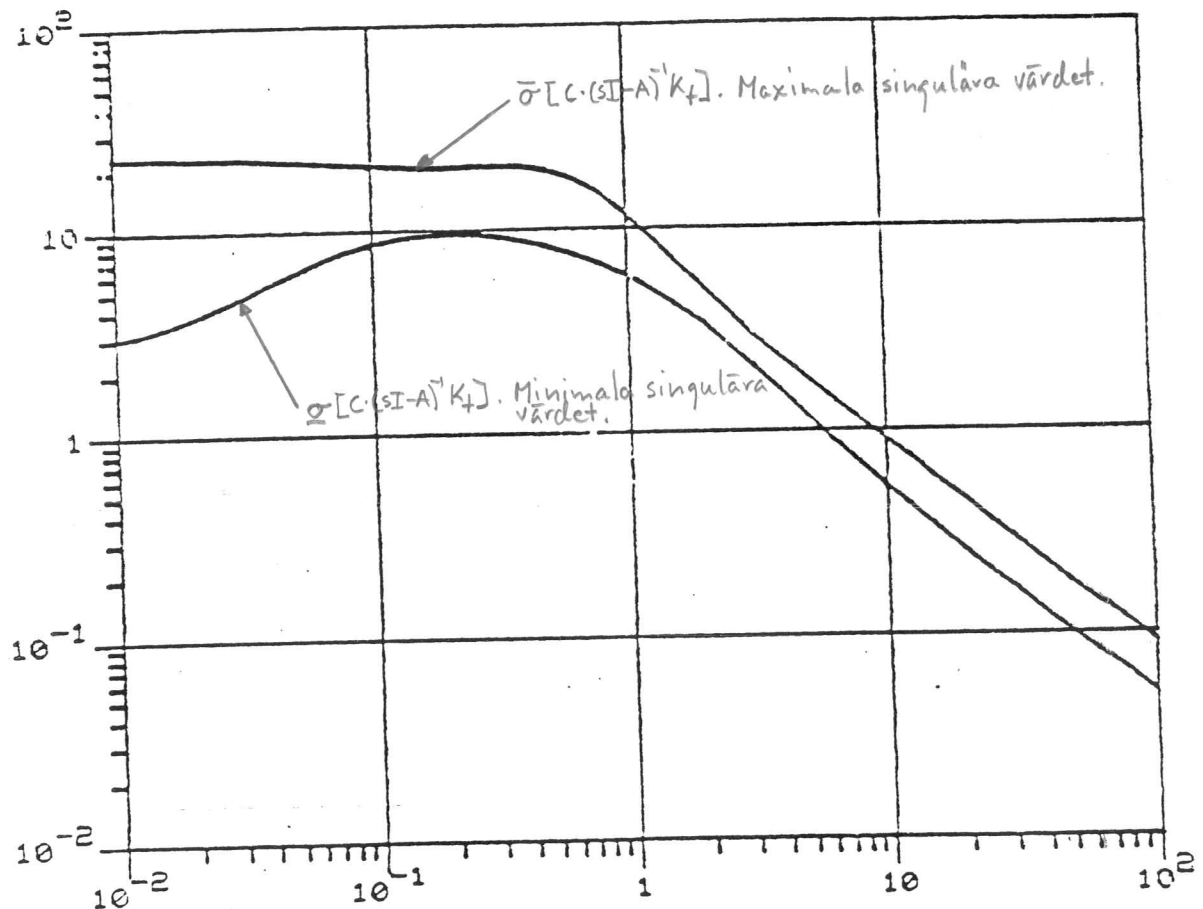
$$\dot{x} = Ax + Bu \quad y = Cx$$

där

$$A = \begin{bmatrix} -0.02 & 0.005 & 2.4 & -32 \\ -0.14 & 0.44 & -1.3 & -30 \\ 0 & 0.018 & -1.6 & 1.2 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0.14 & -0.12 \\ 0.36 & -8.6 \\ 0.35 & -0.009 \\ 0 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 57.3 \end{bmatrix}$$

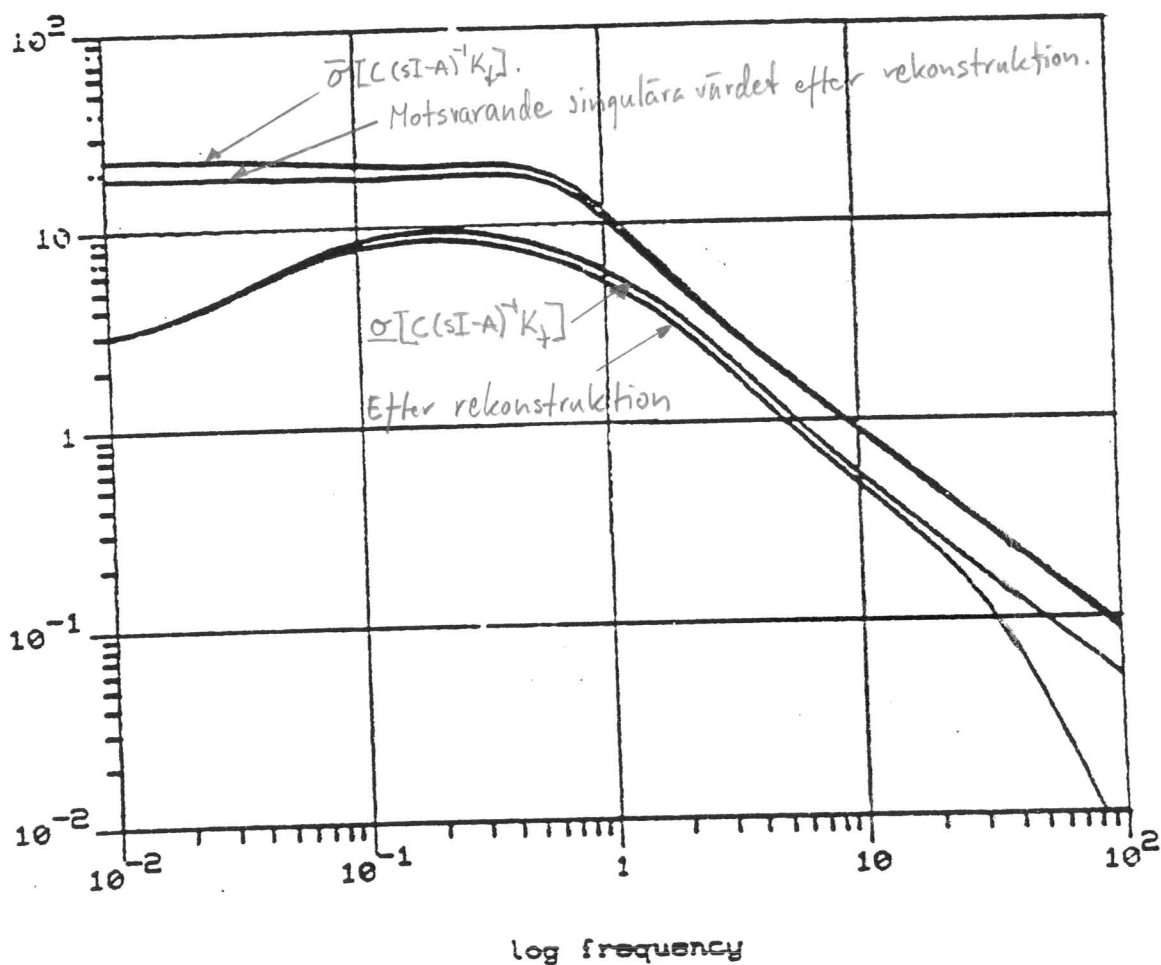
Dessa matriser formar ett system med namnet hel. Vi vill använda oss av ett Kalmanfilter och som ett mått på dess "riktighet" används de singulära värdena för överföringsfunktionen $C(sI-A)^{-1}K_f$, där K_f är förstärkningen i Kalmanfiltret.



Figur 5

Genom ett begåvat val av viktningsmatriser kan lämpliga värde på förstärkningen vid olika frekvenser och även på övergångsfrekvensen erhållas. Dessutom vill man att σ - och $\bar{\sigma}$ -kurvorna skall ligga nära varandra. Därefter görs en rekonstruktion av tillstånden genom att använda en LQ-regulator.

hel.sigma



Figur 6

σ -kurvorna är här placerade i samma diagram för tydlighetens skull. Som man kan se när kurvorna efter rekonstruktion (den understa och näst översta) nästan upp till de för $C(sI-A)^{-1}K_f$

LQG

LQG-metoden är en systematisk metod att påverka de singulära värdetkurvor som är intressanta vid en konstruktion. Metoden kan användas direkt och är den enda för tillfället tillgänglig som direkt kan användas på flervariabla system.

Som grund för LQG-metoden har vi följande tillståndsbeskrivning.

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

$$y = Cx + \eta$$

där x : n -dimensionell tillståndsvektor
 u : m -dimensionell vektor för kontrollsignaler
 y : r -dimensionell vektor för utsignaler
 ξ, η : vitt brus

A, B och C -matriserna uppfyller

$$G(s) = C\phi(s)B$$

$$d(s) = C\phi(s)\xi \quad \text{där } \phi(s) = (sI - A)^{-1}$$

Aterkopplingsproblemet formuleras som ett optimeringsproblem för att hitta en styrlag $u(t) = f(y(\tau); \tau(t))$ så att uppförandekriteriet (förlustfunktionen):

$$J = E \left\{ \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T (z^T z + \rho u^T u) dt \right\} \quad \text{där } T \rightarrow \infty$$

minimeras. Vi har att $z = Hx$: m -dimensionell, ρ är en skalär och att E betyder väntevärde.

Lösningen till problemet ges av

$$u(t) = -K_c \hat{x}(t)$$

K_c är förstärkningen hos en Linjär-Kvadratisk Regulator (LQR) och fås ur följande Riccatiekvation (matrisekvation).

$$0 = PA + A^T P + H^T H - PB \frac{1}{\rho} B^T P$$

Och därvidlag är

$$K_c = \frac{1}{\rho} B^T P$$

Riccatiekvationer av högre ordningen är mycket svåra och tidskrävande att lösa för hand, och därför är LQG-metoden mycket beroende av att man har datorhjälp tillgänglig.

Tillståndsrekonstruktionen $\hat{x}(t)$ fås från Kalman-Bucy filtret.

$$\dot{\hat{x}} = A\hat{x} + Bu + K_f (y(t) - C\hat{x}(t))$$

Förstärkningen K_f löses ut ur en annan Riccatiekvation.

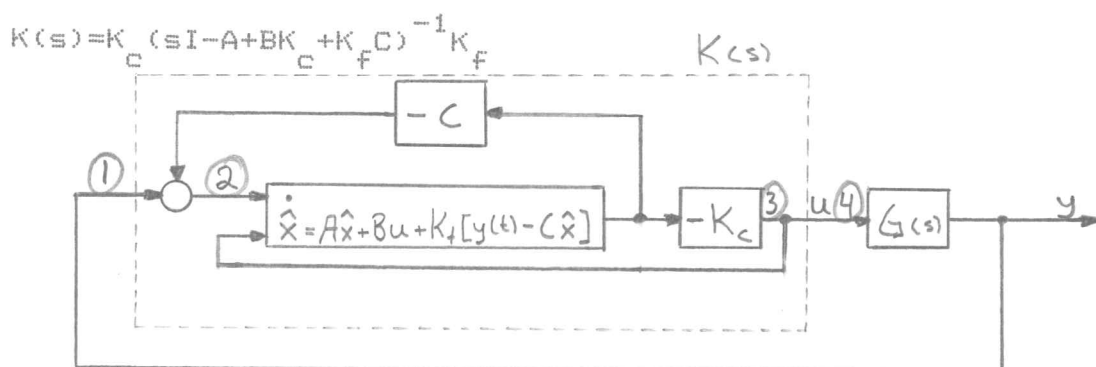
$$0 = A\Sigma + \Sigma A^T + \Gamma\Gamma^T - \Sigma C^T - C\Sigma$$

Och då är

$$K_f = \Sigma C^T -$$

Vi har att $\Gamma\Gamma^T$ är processbruset ξ 's intensitetsmatris, och μI , där μ är en skalär, är sensorbruset n 's motsvarande matris.

Den optimala LQG-regulatorn blir då kaskadkopplingen av LQ-regulatorn och KB-filtret för rekonstruktionen enligt:



Figur 7

Vi har olika överföringsfunktioner relaterade till de fyra numrerade platserna i figuren ovan.

- 1) Utsignalsida och vi har överföringsfunktionen $G(s)K(s)$. Denna kan fås att överrensstämma med $C\phi K_f$.
- 2) Här fås KB-filtrets överföringsfunktion $C\phi K_f$.
- 3) Här har vi LQ-regulatorns överföringsfunktion $K_c\phi B$.
- 4) Insignalsida och överföringsfunktionen $K(s)G(s)$. Denna kan närmas och fås lika med $K_c\phi B$.

Man kan konstruera m.a.p. utsignal- eller insignalsida, men det ger samma resultat i slutänden (matematiska dualiteter).

LQG-konstruktion

Denna utföres i två steg. Ett krav är att man har ett minimum-fas system. Det tillgår som följer:

- 1: Återkoppla alla tillstånd och tillse att singulära värdet kurvorna har önskat utseende beroende på specifikationer.
- 2: Rekonstruera alla tillstånden.

Konstruktionen kan sedan utföras på två olika sätt beroende på om man konstruerar m.a.p. insignal- eller utsignalsidan. Antingen konstrueras en LQ-regulator och sedan används ett KB-filter för rekonstruktionen (detta gäller insignalsidan) eller tvärtom. Resultatet blir detsamma ty, som tidigare nämnts, de är duala metoder.

Vi har LQR-överföringsfunktionen $T(s) = K_c \phi(s) B$ och med förhållandet $(I + T(j\omega))^* (I + T(j\omega)) = I + (H\phi(j\omega)B)^* (H\phi(j\omega)B) / \rho$, så kan man härleda ett förhållande mellan singulära värde:

$\sigma(T(j\omega)) \approx \sigma(H\phi(j\omega)B) / \rho$ under förutsättning att $\sigma(T) \gg 1$.

Övergångsfrekvensen fås ur $\omega_{cmax} = \bar{\sigma}(HB) / \sqrt{\rho}$.

Rekonstruktionen utföres med ett KB-filter där rekonstruktionsfelet endast beror på systemets brusällor, och inte på tillstånd eller styrkommando. Därför är KB-filtrets närvaro för tillståndsrekonstruktionen osynlig.

När rekonstruktionen är klar fås, under förutsättning att förhållandet $\hat{\phi} B = \phi B (I + K_c \phi B)^{-1}$ där $\hat{\phi} = (sI - A + BK_c)^{-1}$ gäller, att

$K(s)G(s) \rightarrow K_c \phi B$.

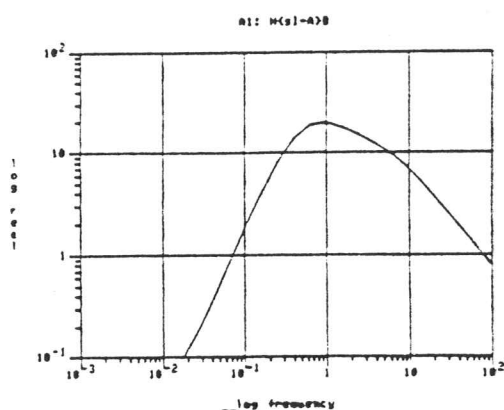
Exempel på hur LQG-metoden kan användas

Vi vill stabilisera ett flygplan. Informationen som finns tillgänglig är stigningshastighet (pitch rate), och vi ändrar vingklaffarnas läge för att reglera densamma. Vi använder LQG design och rekonstruerar med hjälp av ett KB-filter.

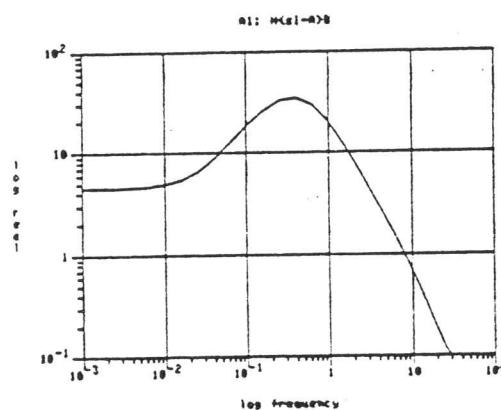
Först skall parametrar väljas till viktningsmatriserna $Q = H^T H$ och $R = \rho I$. Vidare vet vi att $\sigma(T(j\omega)) \approx \sigma(H(j\omega I - A)^{-1} B) / \rho$ vilket innebär att det räcker att ändra H (alternativt också ρ) för att få önskad singulära värdet kurva. Den aktuella kurvan

som studeras är den för $H(sI-A)^{-1}B$. Denna stämmer med en LQR-kurva så länge den har ett värde överstigande ett.

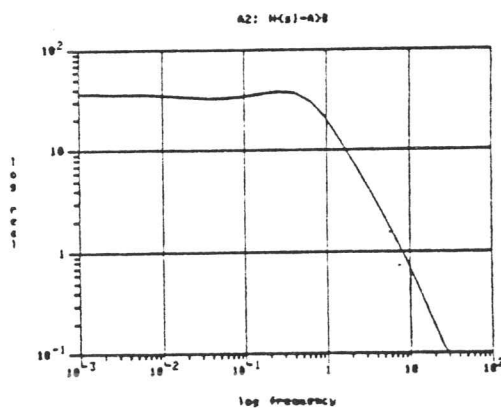
I det första valet av H mäter vi pitch rate. Kurvan för detta val har ett alldeles för lågt värde vid låga frekvenser, se figur 8. Därför viktas vi istället med avseende på pitch attitude (integralen av pitch rate). Figur 9. Läger vi sedan till mera förstärkning då frekvensen är < 0.1 rad/sek fås önskat utseende på kurvan $\sigma(H(sI-A)^{-1}B)$. Figur 10.



Figur 8



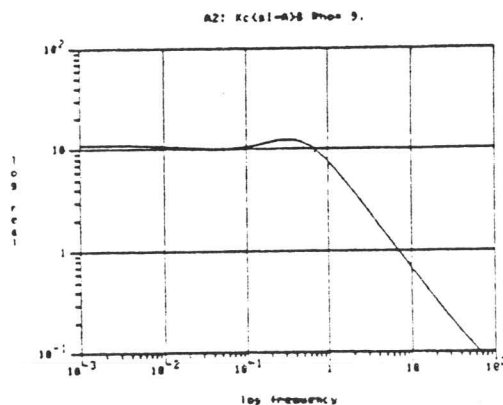
Figur 9



Figur 10

När nu väl viktmatriserna är bestämda skall LQ-regulatorn konstrueras. Detta görs genom att lösa Riccati-ekvationen. Därefter skapas funktionen $K(sI-A)^{-1}B$ och singularära värdekurvan uppritas och kan studeras.

Full State Loop Transfer

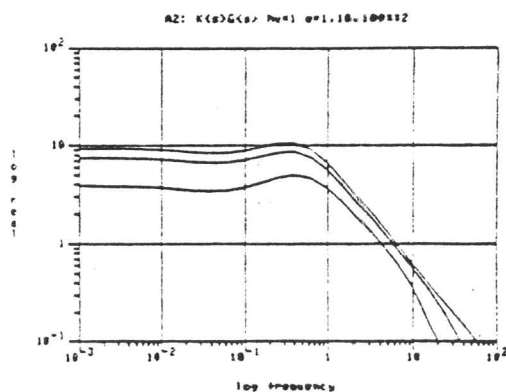


Figur 11

Man kan se att bandbredd och förstärkningen vid låga frekvenser är nästan samma som i figur 10.

Till sist sker rekonstruktionen med hjälp av ett Kalman-Bucyfilter. Viktningsmatriserna i detta fallet är $M=qBB^T$, där man låter $q \rightarrow \infty$, och $N=\mu I$. Vid fixt värde på μ ökas sedan värdet successivt på q . $K(s)G(s) = K_c (sI - A + BK_c + K_f C)^{-1} K_f * C(sI - A)^{-1} B$ beräknas tillsammans med singularära värdet kurvan. Värdet på q ökas sedan tills sistnämnda kurva stämmer överrens med kurvan för $K_c (sI - A)^{-1} B$, se figur 12 nedan.

Loop Transfer Recovery



Figur 12

CAD-verktyg

LQG-metoden medför en hel del algebraiskt räknande, bl. a. lösandet av högre ordningens matrisekvationer. Detta gör, utom kanske för mindre, elementära designproblem, att det är nödvändigt med datorhjälp. Denna kan erhållas antingen genom att man gör vanliga batchkörningar eller genom att ett interaktivt programpaket finns tillhanda.

Grundläggande algoritmer

För att kunna lösa LQG-problem är det nödvändigt att det finns algoritmer implementerade som kan lösa följande:

- 1) Lösande av Riccatiekvationer för att få K_c och K_f , förstärkningsfaktorerna i respektive LQ-regulatorn och i KB-filtret. För detta finns det ett antal kända algoritmer att tillgå.
- 2) Beräkningen av olika svarskurvor i frekvensplanet för $G(j\omega)$, $K(j\omega)$, $GK(j\omega)$ osv. Utgående från matriserna i tillståndsmodellen skall t.ex. $G(j\omega) = C(sI - A)^{-1}B + D$ beräknas. Beroende på n =systemets ordningstal, m =antal insignaler och r =antal utsignaler kan olika algoritmer användas beroende på hur snabbt man önskar svar från datorn.
- 3) Singulära värde dekomposition (uppdelning) måste göras så att man kan rita önskade kurvor. För $G(j\omega)$ får man uppdelningen $G(j\omega) = U\Sigma V^H$, där U ($r \times r$) och V ($m \times m$) är ortogonala (dvs $U^H U = I$) och där $()^H$ betyder konjugat-transponat. Σ , som är en $r \times m$ matris har i diagonalen elementen $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_p \geq 0$, $p = \min(r, m)$. Vi är intresserade av σ_1 och σ_p enligt tidigare.

Förutom algoritmer för de tre ovanstående problemen behövs programvara för matrisberäkningar, för in/utmatning, för uppritande av kurvor etc.

Interaktiva programpaket

Mycket arbete har lagts ner på att konstruera CAD-verktyg att användas istället för den mera långsamma och svårarbetade batchkörningsmetoden. De två verktygen som beskrivs och jämförs i det följande är HoneyX, utvecklat vid MIT och Honeywell Inc. i USA, samt Synpac, utvecklat vid Institutionen för Reglerteknik LTH i Lund.

HoneyX

HoneyX är utvecklat och arbetar inom ett system hos Honeywell som heter Multics, Multiplexed Information and Computing Service.

Multics

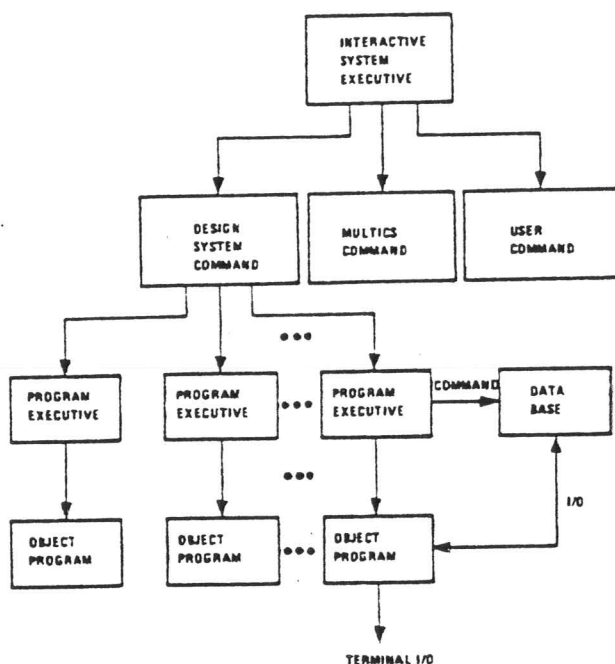
Multics hanterar bl.a. hårdvara såsom minne, terminaler, skrivare, kortläsare etc. Det innehåller vidare editor, möjligheter att skriva s.k. Macros, olika kommando, kompilatorer samt aktiva funktioner. Dessa sistnämnda används när någon form av aktuell data eller information behövs, eller så kan de användas som alldeles vanliga kommando. De aktiva funktionerna indelas i olika grupper, t.ex. de som rör datum och tid, aritmetik, in/utmatning osv.

Kommando representeras av ett program i systemets programbibliotek och exekveras genom att man skriver dess namn. De skall kunna utföra önskad uppgift, ta emot godtyckligt antal parametrar samt hantera eventuella fel. Kommandona är även de indelade i olika grupper, beroende på vilken funktion de har.

Minnet är hierarkiskt uppbyggt med directories och med filer inom ett directory. Det är möjligt att ha ett antal olika attribut i ett directory/en fil. I detsamma kan man lagra data, text, käll- eller objektкод eller låta det vara tomt.

Hur HoneyX är organiserat

Programpaketet HoneyX är indelat i tre hierarkiska nivåer enligt bild överst på nästa sida.



Figur 13

System executive

Denna ansvarar för kontakterna med omgivningen till designsystemet. Den gör systembiblioteket tillgängligt för användaren och skickar dennas kommando vidare till lämplig processor för exekvering. Kommando kan tillhöra tre olika klasser: HoneyX-kommando, Macroanrop eller Multicskommando.

Program executive

Ett interface mellan system executive och objektprogrammen. Den hanterar främst in- och utmatning inom filsystemet. Därvidlag måste den analysera kommando så att rätt filer allokeras. Eventuella argument transporteras vidare till objektprogrammet.

Objektprogram

Utför de önskade beräkningarna beroende på vilket kommando man givit. Den undersöker medföljande argument och kontrollerar så de räcker och har rätt form. Saknas något argument frågas efter detta om det inte har default värde.

Programmen i HoneyX:s bibliotek är indelade i två grupper: Generella matrisrutiner och Specialprogram med ett visst ändamål. För att kunna använda de sistnämnda måste man tillse att namnsättningen är enhetlig, så att ett "system" bildas. Programmen anropas genom ett kommando. Dettas allmänna

form är <kommandoidentitet> <lista med argument>, där kommandoidentitet är detsamma som programmets namn, och argumentet definierar de datafiler som behövs.

Macro_i_HoneyX

För skapandet av egna programrutiner och för tillägg till programpaketet kan man utnyttja två möjligheter. Antingen skriver man en "exec-com", som är en fil innehållande Multics och HoneyX kommando, eller så kan man skriva Fortran subrutiner. Dessa sistnämnda måste anropas från en exec-com som öppnar och stänger filer samt vidarebefordrar rätta parametrar till subrutinen.

Exec-com

Exec-com betyder uttytt "executive-commands" och är ett annat namn för en makro. Dylika kan skapas då man ofta utför en viss följd av kommando. Dessa skrivs istället samman i en makro, som anropas med dess namn, och sedan exekveras varje separat kommando i tur och ordning. I en makro kan do-loopar och villkorliga hopp användas.

Fortran subrutiner

En subroutine skrivs som ett vanligt program och den exekveras genom att dess namn skrivs ut på terminalen, eventuellt med tillhörande aktuella parametrar. Innan den kan användas måste man ha skrivit en exec-com för att kunna exekvera densamma.

LQG-kommando_i_HoneyX

HoneyX är ett programpaket som man lätt kan lära sig använda. Det är skräddarsytt för LQG-konstruktion, men innehåller även en del rutiner för SISO-system. Det krävs inte någon större kunskap om överordnade programpaket eller operativsystem för att kunna använda det.

För skapandet av A, B, C etc matriserna används ett kommando -Matini- där argumenten är namn och dimension. Här får man vara noga med namnsättningen. Ett gemensamt systemnamn måste användas, men detta behöver ej särskilt deklarerats. Matriserna skiljs åt genom att de heter systemnamn.matrisnamn.

När man skapat sina systemmatriser är nästa steg att välja sina viktningsmatriser. En av dessa (H eller Γ) används för att skatta en singulära värdet kurva. Här kan man välja att bruka sig av ett av två kommando, beroende på vilket av de två tillvägagångssätten för LQG-konstruktion som man valt.

T.ex. -Lqr_weight-, som plottar singulära värdena av $H(sI-A)^{-1}B$ där argumenten är systemnamn, frekvensgränser samt hur plottningen skall avgränsas.

Därefter kan man ändra i viktningsmatrisen, om man inte är nöjd med singulära värdet kurvorna. Eventuellt får man addera till en integrator om man måste höja förstärkningen vid låga frekvenser. Ändring av matriser görs med kommandot -Matchg-, argument namn, position och nytt värde.

Sedan används Lqr_weight och Matchg tills man är fullständigt tillfreds med kurvorna. Nästa steg blir då att beräkna K_c eller K_f samt att plotta förstärkningen hos

LQ-regulatorn eller hos KB-filtret. I det förstnämnda fallet används kommandot -Lqr-, som ger K_c samt en plot av sin-

gulära värdet av $K_c(sI-A)^{-1}B$. Argumenten skall vara samma som de för Lqr_weight.

Återstår att genomföra rekonstruktionen. Innan detta kan göras måste överföringsfunktionen $G(s)$ beräknas. Då används kommandot -Freq- med argumenten systemnamn samt frekvensgränser (samma som tidigare). Man erhåller då $systemnamn.g=C(sI-A)^{-1}B+D$. För rekonstruktionen finns också två kommando, precis som innan, beroende på vilken metod man valt. En viktig parameter är en viktningsskalär q, som genom successiv användning av kommandot ges ett allt större värde tills man är nöjd med rekonstruktionen.

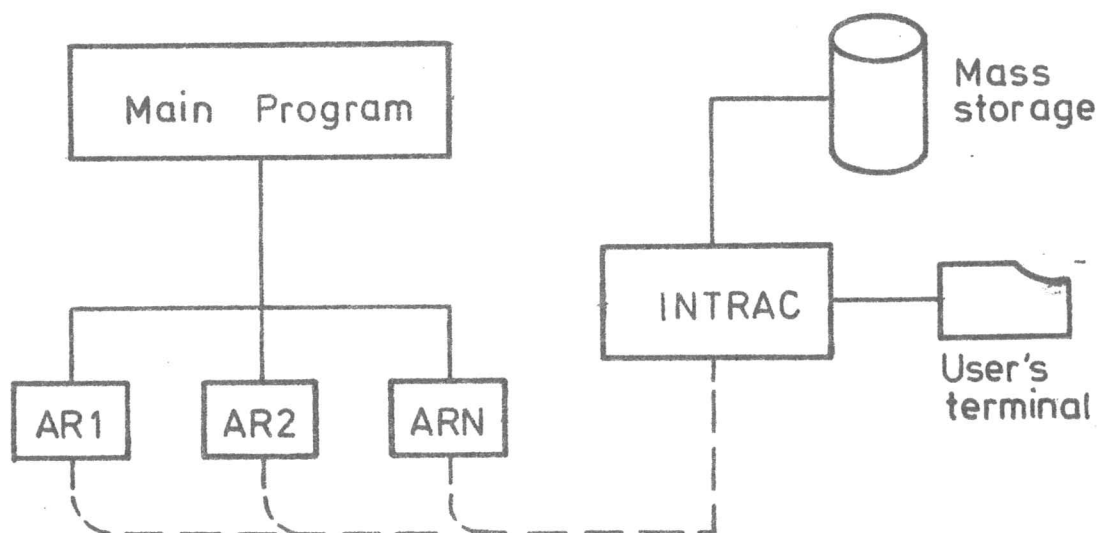
Används kommandot `-Lqr_recovery-` måste argumenten vara systemnamn, q , frekvensgränser och gränser för plotten. Sedan kommandot givits beräknas först KB-filtrets förstärkning K_f och $K(s) = K_f (sI - A + BK_f + K_f C)^{-1} K_f$, samt ritas en plott över de singulära värdena till $K(s)G(s)$. När $q \neq 0$ skall denna stämma med den för $K_f (sI - A)^{-1} B$. Är man sedan nöjd, är konstruktionen av regulatorn klar och det återstår att simulera systemet för att avgöra om det uppfyller de givna specifikationerna.

Synpac

Synpac är en del i en familj av olika interaktiva program-paket som är utvecklade vid Institutionen för Reglerteknik, LTH i Lund. Programmen kan användas till olika sorters analyser, till konstruktion eller till systembetraktelser. Kommunikationen mellan program och användare företas med hjälp av kommando. De paket som i skrivande stund finns tillhanda är Idpac, Modpac, Synpac och Polpac. De kommando som avges hanteras inom en uppsättning subrutiner med det gemensamma namnet Intrac. De olika tillämpningsprogrammen tillsammans med Intrac medger ett flertal skilda sätt att beskriva ett dynamiskt system på, t.ex tillståndsform eller in/utsignalpolynomform.

Intrac

Intrac är avsett att kombineras med ett tillämpningsprogram så att ett program för interaktiv kommunikation erhålles. Intrac innehåller kommando för: generering av Makros, tilldelning av värde till variabler, hopp, loopar och in/utmatning. Detta innebär att Intrac inte innehåller några tillämpningsanpassade avsnitt, och därför kan det användas tillsammans med vilket tillämpningsprogram som man önskar. Ett interaktivt program uppbyggt kring Intrac har följande principiella utseende.



Figur 14

Huvudprogrammet anropar subrutiner (AR1, AR2 osv) och Intrac. Intrac läser in ett avgivet kommando från t.ex. en terminal och analyserar detta. Beroende på om det är ett tillämpningskommando så skickas information till huvudprog-

rammet som i sin tur anropar rätt subroutin (action routine). Det innebär att oftast är subroutin=action routine. Ett annat sätt för Intrac att läsa ett kommando är från massminne i form av en del av en makro.

Kommando

Kommando har den generella formen <kommando> <argumentlista>, där argumenten antingen specificerar ett resultat av anropet eller indata till operationen som skall utföras. Kommandot kan vara av tre typer: Intrac-, Tillämpnings- eller Makrokommando. Det förekommer dessutom för vissa kommando att underkommando måste ges, för de fall att ytterligare data eller information behövs.

Man kan också indela kommando i generella eller tillämpningsorienterade. De förstnämnda hanteras av Intrac och de övriga hanteras av aktuellt tillämpningsprogrampaket. När ett kommando ges från en terminal kan man befinna sig i tre olika tillstånd. Direkta är det som programmet startar i, genererings är ett man hamnar i efter ett makroanrop och avbrutet när man då exekveringen av en makro tillfälligt avbrytes.

Aggregatfil

En aggregatfil består av en sammanlagring av ett antal individuella filer till en enda sekvensiellt uppbyggd fil. Varje enskild delfil i en aggregatfil är dock lika åtkomlig som om den vore ensam. Detta klaras med hjälp av :-notation. Aggregatfilerna ökar effektiviteten hos systemets databashanterare, och tillåter dessutom ett intellegent namngivande på de filer som tillsammans bildar ett system.

Systemfil

Detta är en textfil som innehåller en systembeskrivning av något önskat system. Namn, nyckelord etc gör det lättare att avgöra vilket och vilken sorts system som beskrivs. Aggregatfiler och datafiler lagras separat, endast deras namn finns medtagna i systembeskrivningen, som referenser till de aktuella filerna. En systemfil kan innehålla mer än en systembeskrivning.

Programpaketet Synpac

Synpac's främsta användningsområde är för konstruktion i flervariabla system med hjälp av s.k. linjär kvadratiska metoder. Specifikationerna för det system man konstruerar till kan uttryckas i antingen utseende på och tider hos ett stegsvar, i utseende på frekvenssvar eller i pol/nollställesplacering. För att kunna lösa konstruktionsuppgifterna innehåller Synpac rutiner för följande:

- 1) Matrishantering.
- 2) Systembeskrivning.
- 3) Egenvärdesberäkning.
- 4) Frekvenssvarsberäkning.
- 5) Simulering.
- 6) Grafisk visning av tids- och frekvenssvar.
- 7) Lösande av Riccatiekvationer.
- 8) Konstruktion av reducerade regulatorer.
- 9) Generering av deterministiska eller stokastiska testsignaler.

För LQG-konstruktion finns det kommando för konstruktion av återkopplad LQ-regulator, och för rekonstruktion av tillstånd med hjälp av ett Kalman-filter.

Liten jämförelse HoneyX-Synpac

Då erfarenheten av Synpac är liten hos rapportskrivaren kan inte något heltäckande jämförelse göras. En spontan reaktion är att Synpac är mycket mera generellt än HoneyX, och även betydligt svårare att sätta sig in i hur det skall användas. Framförallt p.g.a. att kommandona har en betydligt mera komplex struktur. HoneyX är, för LQG-konstruktion, mera tillrättalagt och rakt på. Man behöver inte lägga ner någon tid på att göra en systembeskrivning, utan det räcker att följa namngivningskonventionen. Detta kan visserligen leda till att det i efterhand kan vara svårare att studera en konstruktion utförd i HoneyX, just eftersom det saknas en systembeskrivning.

Sätten att konstruera på är väsentligen olika. I HoneyX använder man sig av singulära värde kurvor, medan man i Synpac får ta till de mere konventionella svars kurvorna eller pol/ nollställesplacering. De olika stegen i konstruktionen är dock lika eftersom samma teorier ligger i botten.

För den ovane användaren finns det ett underprogram till Synpac med namnet Anpac. Detta är en samling makros skrivna för lösandet av enkla kontrollproblem i kontinuerliga SISO-system.

Tillämpningsexempel

Denna tillämpning avser en regulator till det redan omnämnda FSW-flygplanet. Som tidigare nämndes finns det ett antal (14 stycken närmare bestämt) olika flygtillstånd som man använder sig av i analyserna. De varierar i flyghastighet och i flyghöjd. I exemplet används flygtillståndet: hastighet 1.2 mach, flyghöjd 30 000 fot. Modellen, tillståndsmatriserna, är kopplade till flygplanets färd och uppförande i vertikalled, och de har erhållits efter linearisering kring ett jämviktssläge. De olika tillstånden samt in- och utsignalerna till modellen är listade i bilaga 2.

Förberedande åtgärder

Den regulator som skall konstrueras skall med hjälp av ändringar i canardinställningen reglera flygplanets stigningshastighet (pitch rate). Det visar sig dock att en direkt användning av pitch rate ger en alldeles för låg, $\ll 1$, förstärkning vid låga frekvenser. I stället använder vi oss av pitch acceleration, som är ett mått på hur pitch rate förändrar sig. Vi introducerar dessutom extra förstärkning med en faktor 20 i systemet. Då det efter dessa ändringar fortfarande är för lite förstärkning kopplas en extra integrator med förstärkningen 1 000 gånger in på utgången. De ursprungliga tillståndsmatriserna finns i bilaga 3 (den högra uppsättningen). Efter ovanstående åtgärder erhåller vi tillståndsmatriser som i sitt ändrade skick har utseendet i enlighet med bilaga 4 (a och b-matriserna) och bilaga 5 (c-matrisen samt h, kc och kf-matriserna, se nedan).

Val av viktningsmatris

När nu dessa förberedande åtgärder är vidtagna kan ett val av viktningsmatris ske. Specifikationerna är att vi vill ha hög förstärkning vid låga frekvenser, låg förstärkning vid höga frekvenser samt en övergångsfrekvens på 10 rad/sek. Efter val av, och dessutom flera ändringar av, viktningsmatrisen H erhålles till sist en skattad singulara värdet kurva som tillfredsställande uppfyller specifikationerna. Se bilaga 6 där singulara värdet av $H(sI-A)^{-1}B$ är plottat. Viktningsmatrisen blir $\text{syst.}h = (20 \ 0.02 \ 0.32 \ 75 \ 100 \ 1)$, värdet på g är ett.

Som ett resultat av att vi erhållit denna H-matris, får vi efter lösande av Riccatiekvationen en förstärkning K_c hos

LQ-regulatorn som är $\text{syst.kc}=(20.21 \ 0.023 \ 0.347 \ 106.76 \ 173.76 \ 1)$. Jämför med sys.h ovan. Kurvan över singulära värden av $K_c(sI-A)^{-1}B$ är till största delen överrensstämmande med den över $H(sI-A)^{-1}B$. Jämför bilaga 6 med bilaga 7, kurvan över singulära värden av $K_c(sI-A)^{-1}B$.

Rekonstruktion

Till sist görs rekonstruktionen med hjälp av ett Kalman-Bucy-filter. Då visar det sig att ett val av $q=10\ 000$ ger god rekonstruktion. Kurvan över det singulära värdet av $K(s)G(s)$, bilaga 8, liknar de föregående med undantag för vid höga frekvenser.

Simulering

Systemet med den nykonstruerade regulatorn simuleras därefter. Det simuleras under 1.5 sekunder med simuleringssteg på 0.025 sekunder. Insignalen till systemet (kontrollsignalen) är en stegfunktion med start i $t=0$ och med storleken 0.5 radianer/sekund. Man kan efter avslutad simulering studera förändringar i de 5 första tillstånden i enlighet med bilaga 2 genom att studera bilagorna 9-13. Bilaga 9 visar hur pitch acceleration ändras för att få det önskade värdet på pitch rate. Bilaga 10 visar ändringarna i forward velocity osv. Enheterna på y-axeln framgår av bilaga 2. Utsignalen från systemet sys.y (bilaga nummer 14) visar att efter en viss tid, ≈ 1 sekund och med ca 40% översläng, erhåller man, dock med ett litet stationärt fel, önskad pitch rate.

När man sedan studerar canardutslagen, bilaga 15, ser det ut som om man har mycket stora dylika, men om man tar hänsyn till att vi själv har introducerat en extra förstärkning på 20 000 gånger, så blir utslagen mera troliga och riktiga. Man kan också se att canardvingarna tippar nedåt först vilket förväntas då man vill få ett flygplan att stiga fortare.

Sammanfattning

LQG-metoden för konstruktion av regulatorer har presenterats. Denna metoden tillsammans med interaktiv datorhjälp medger ett slagkraftigt och effektivt sätt att på modern väg konstruera för flervariabla återkopplade system.

Presentationen är inte att anse som heltäckande, utan mera ett sätt att visa på hur den kan användas i en tillämpning. Förutom för konstruktion av en regulator till ett flygplan kan metoden användas till flervariabla system generellt, oavsett vilken tillämpning som är aktuell.

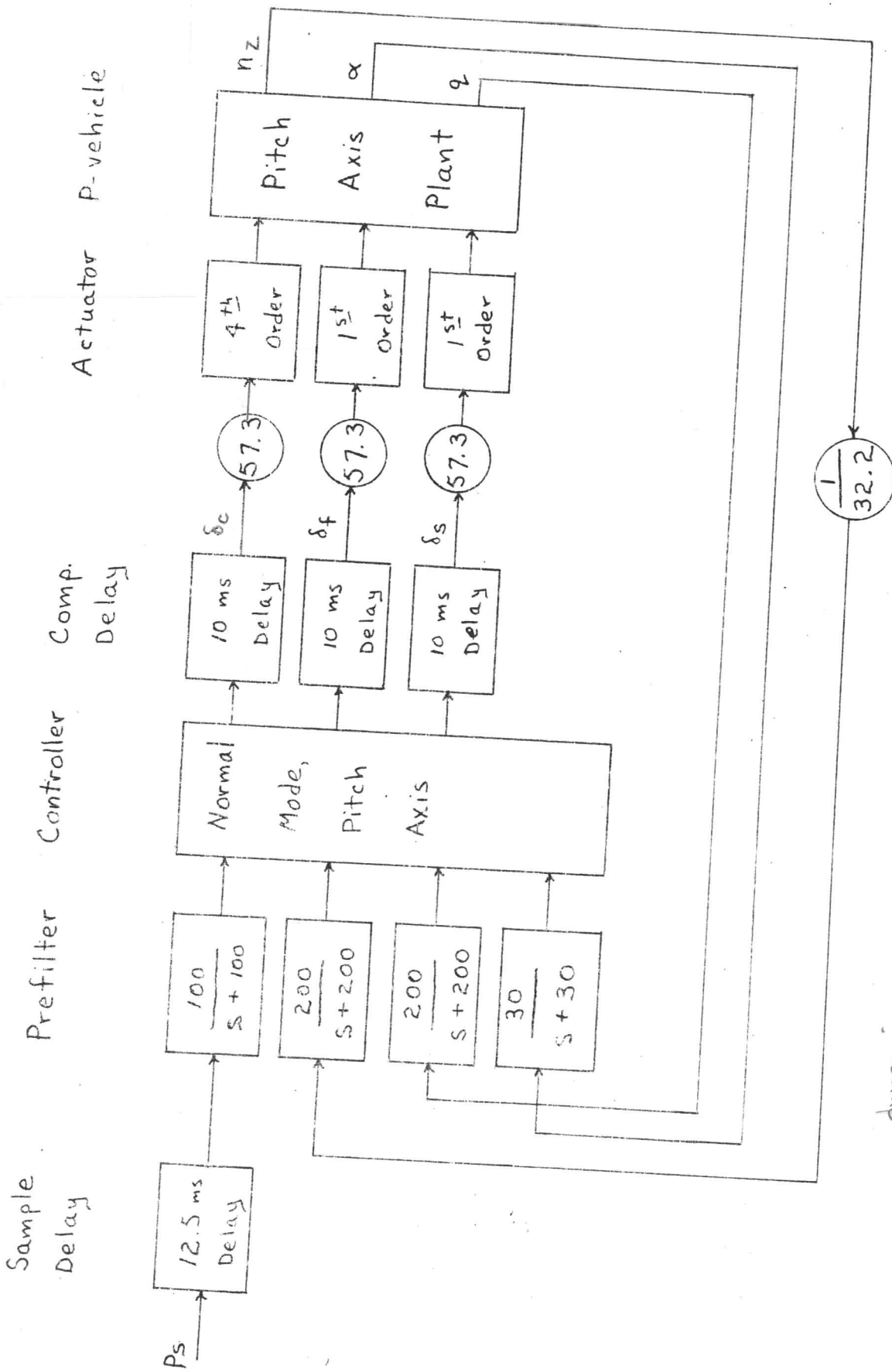
Som kan förstås av presentationerna av och jämförelsen mellan HoneyX och Synpac finns det olika angreppssätt för en konstruktion. I detta fallet har de varit utformningen av singulära värdetkurvor eller tids- och frekvenssvarsspecifikationer. Vilket sätt som används beror på den programvara som finns tillgänglig.

Det kan också konstateras att utan datorhjälp är det svårt att utföra LQG-konstruktion. Framförallt är det rena räkandet mycket omfattande, t.ex. lösande av Riccatiekvation. Ett interaktivt arbetssätt underlättar för användaren. Med hjälp av kommando och frågor/svar kan resultatet av t.ex. en ändring i en viktningsmatrix direkt utläsas. Vissa nödvändiga algoritmer måste finnas implementerade oavsett vilket konstruktionsmätt man använder.

En av styrkefaktorerna hos LQG-metoden är att den medger analys av och konstruktion för ett system i sin helhet. Avgörande egenskaper som robusthet och integritet kan kontrolleras. Integritet betyder systemets förmåga att tåla bortfall av givare eller ställdon och fortfarande kunna sköta sin uppgift. I övrigt är metoden lätt att lära sig använda, men svår att förstå hur den egentligen fungerar, främst matematiskt.

Referenser

- J. Blakelock -Automatic Control of Aircraft and Missiles.
- B. Etkin -Dynamics of flight. Stability and Control.
- John C Doyle -Multivariable design techniques based on singular value generalizations of classical control. AGARD grp lecture series No. 117 okt 1981.
 -Limitations on achievable performance of multivariable feedback systems. AGARD grp lecture series No. 117 sep 1981.
 -Robustness of multiloop linear feedback design 17th IEEE konferens, Decision and control San Diego CA jan 1979.
- G. Stein
 N.R. Sandell -Classical and modern methods for control system design. Dep. of Electrical Engineering and Computer Science. MIT USA.
- G. Stein -LQG based multivariable design: Frequency domain interpretation. AGARD grp lecture series No. 117 sep 1981.
- G. Stein
 S. Pratt -LQG multivariable design tools. AGARD grp lecture No. 117 sep 1981.
- J. Wieslander
 H. Elmqvist -INTRAC. A communication module for interactive programs. Inst. för Reglerteknik LTH 1980.
- J. Wieslander -Interactive programs. General guide. Inst. för Reglerteknik LTH 1978.
- J. Wieslander -Synpac commands. User's guide. Inst. för Reglerteknik LTH 1980.



Actuator $\frac{20}{s+20} \left[\underbrace{\left(\frac{75^2}{s^2+2 \cdot (0.7)75s+75^2} \right)}_{\text{dynamics}} \left(\frac{144}{s+144} \right) \right]$

*** NAME LIST DATA FOR - PVEHICLE ***

NUMBER	NAME	DESCRIPTION	UNIT
1	XU	1 Pitch acceleration 2 FORWARD VELOCITY	FT/SEC
2	XW	3 VERTICAL VELOCITY	FT/SEC
3	XQ	4 PITCH RATE	RADIAN/SEC
4	XTHETA	5 PITCH ATTITUDE 6 Integrator	RADIAN
1	NZ	NORMAL ACCELERATION	FT/SEC ²
2	ALPHA	ANGLE OF ATTACK	RADIAN
3	Q	PITCH RATE	RADIAN/SEC
4	THETA	PITCH ATTITUDE	RADIAN
5	QDOT	PITCH ACCELERATION	RADIAN/SEC ²
6	GAMMA	FLIGHT PATH ANGLE	RADIAN
7	GAMMADOT	FLIGHT PATH RATE	RADIAN/SEC
1	DCC	CANARD COMMAND	DEGREE
2	DFC	FLAPPERON COMMAND	DEGREE
3	DSC	STRAKE COMMAND	DEGREE

*** quadruple data for - pvehicle ***

matrix a size = 4 x 4

	1-column	2-column	3-column	4-column
1-row	-0.3001E-01	0.6249E-01	0.0000E+00	-0.3212E+02
2-row	0.2323E-01	-0.1254E+01	0.1188E+04	-0.1706E+01
3-row	0.4757E-03	0.9826E-02	-0.5257E+00	0.1148E-03
4-row	0.0000E+00	0.0000E+00	0.1000E+01	0.0000E+00

matrix b size = 4 x 3

	1-column	2-column	3-column
1-row	-0.4425E-01	0.1498E+00	0.8849E-01
2-row	-0.4217E+01	-0.2592E+01	-0.5672E+00
3-row	0.2050E+00	-0.1433E+00	-0.9742E-01
4-row	0.0000E+00	0.0000E+00	0.0000E+00

matrix c size = 7 x 4

	1-column	2-column	3-column	4-column
1-row	-0.2323E-01	0.1254E+01	0.4674E+01	0.1976E-04
2-row	-0.4447E-04	0.8381E-03	0.0000E+00	0.0000E+00
3-row	0.0000E+00	0.0000E+00	0.1000E+01	0.0000E+00
4-row	0.0000E+00	0.0000E+00	0.0000E+00	0.1000E+01
5-row	0.4757E-03	0.9826E-02	-0.5257E+00	0.1148E-03
6-row	0.4447E-04	-0.8381E-03	0.0000E+00	0.1000E+01
7-row	-0.2080E-04	0.1054E-02	0.3918E-02	0.1467E-05

matrix d size = 7 x 3

	1-column	2-column	3-column
1-row	0.4217E+01	0.2592E+01	0.5672E+00
2-row	0.0000E+00	0.0000E+00	0.0000E+00
3-row	0.0000E+00	0.0000E+00	0.0000E+00
4-row	0.0000E+00	0.0000E+00	0.0000E+00
5-row	0.2050E+00	-0.1433E+00	-0.9742E-01
6-row	0.0000E+00	0.0000E+00	0.0000E+00
7-row	0.3533E-02	0.2179E-02	0.4793E-03

fc10 mach #, alt = 1.2,30k_revised

*** quadruple data for - pvehicle ***

matrix a size = 4 x 4

	1-column	2-column	3-column	4-column
1-row	-0.1789E-01	0.4689E-01	0.0000E+00	-0.3210E+02
2-row	0.5287E-02	-0.9781E+00	0.8903E+03	-0.2136E+01
3-row	-0.1480E-02	0.1825E-01	-0.3983E+00	0.1211E-03
4-row	0.0000E+00	0.0000E+00	0.1000E+01	0.0000E+00

matrix b size = 4 x 3

	1-column	2-column	3-column
1-row	-0.3570E-01	0.1584E+00	0.5332E-01
2-row	-0.8771E+00	-0.3226E+01	-0.3947E+00
3-row	0.1676E+00	-0.8724E-01	-0.6235E-01
4-row	0.0000E+00	0.0000E+00	0.0000E+00

matrix c size = 7 x 4

	1-column	2-column	3-column	4-column
1-row	-0.5287E-02	0.9781E+00	0.4567E+01	-0.7907E-05
2-row	-0.7424E-04	0.1118E-02	0.0000E+00	0.0000E+00
3-row	0.0000E+00	0.0000E+00	0.1000E+01	0.0000E+00
4-row	0.0000E+00	0.0000E+00	0.0000E+00	0.1000E+01
5-row	-0.1480E-02	0.1825E-01	-0.3983E+00	0.1211E-03
6-row	0.7424E-04	-0.1118E-02	0.0000E+00	0.1000E+01
7-row	-0.7237E-05	0.1097E-02	0.5104E-02	0.3684E-05

matrix d size = 7 x 3

	1-column	2-column	3-column
1-row	0.8771E+00	0.3226E+01	0.3947E+00
2-row	0.0000E+00	0.0000E+00	0.0000E+00
3-row	0.0000E+00	0.0000E+00	0.0000E+00
4-row	0.0000E+00	0.0000E+00	0.0000E+00
5-row	0.1676E+00	-0.8724E-01	-0.6235E-01
6-row	0.0000E+00	0.0000E+00	0.0000E+00
7-row	0.9775E-03	0.3617E-02	0.4450E-03

fc9 mach #, alt = 0.5,50k_revised


```
matprt syst.(a b)
Number of rows = 6
Number of columns = 6
-2.0000E+01  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00
-8.8500E-01 -3.0010E-02  6.2490E-02 -3.2120E+01  0.0000E+00  0.0000E+00
-8.4340E+01  2.3230E-02 -1.2540E+00  1.1880E+03 -1.7060E+00  0.0000E+00
 4.1000E+00  4.7570E-04  9.8260E-03 -5.2570E-01  1.1480E-04  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  1.0000E+00  0.0000E+00  0.0000E+00
 0.0000E+00  0.0000E+00  0.0000E+00  0.0000E+00  1.0000E+03  0.0000E+00
```

```
Number of rows = 6
Number of columns = 1
```

```
1.0000E+00
0.0000E+00
0.0000E+00
0.0000E+00
0.0000E+00
0.0000E+00
```

>

matprt syst.(c h kc kf)

Number of rows = 1
Number of columns = 6

0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 1.0000E+00

Number of rows = 1
Number of columns = 6

2.0000E+01 2.0000E-02 3.2000E-01 7.5000E+01 1.0000E+02 1.0000E+00

Number of rows = 1
Number of columns = 6

2.0210E+01 2.2981E-02 3.4685E-01 1.0676E+02 1.7366E+02 1.0000E+00

Number of rows = 6
Number of columns = 1

5.2216E+03

-2.2649E+02

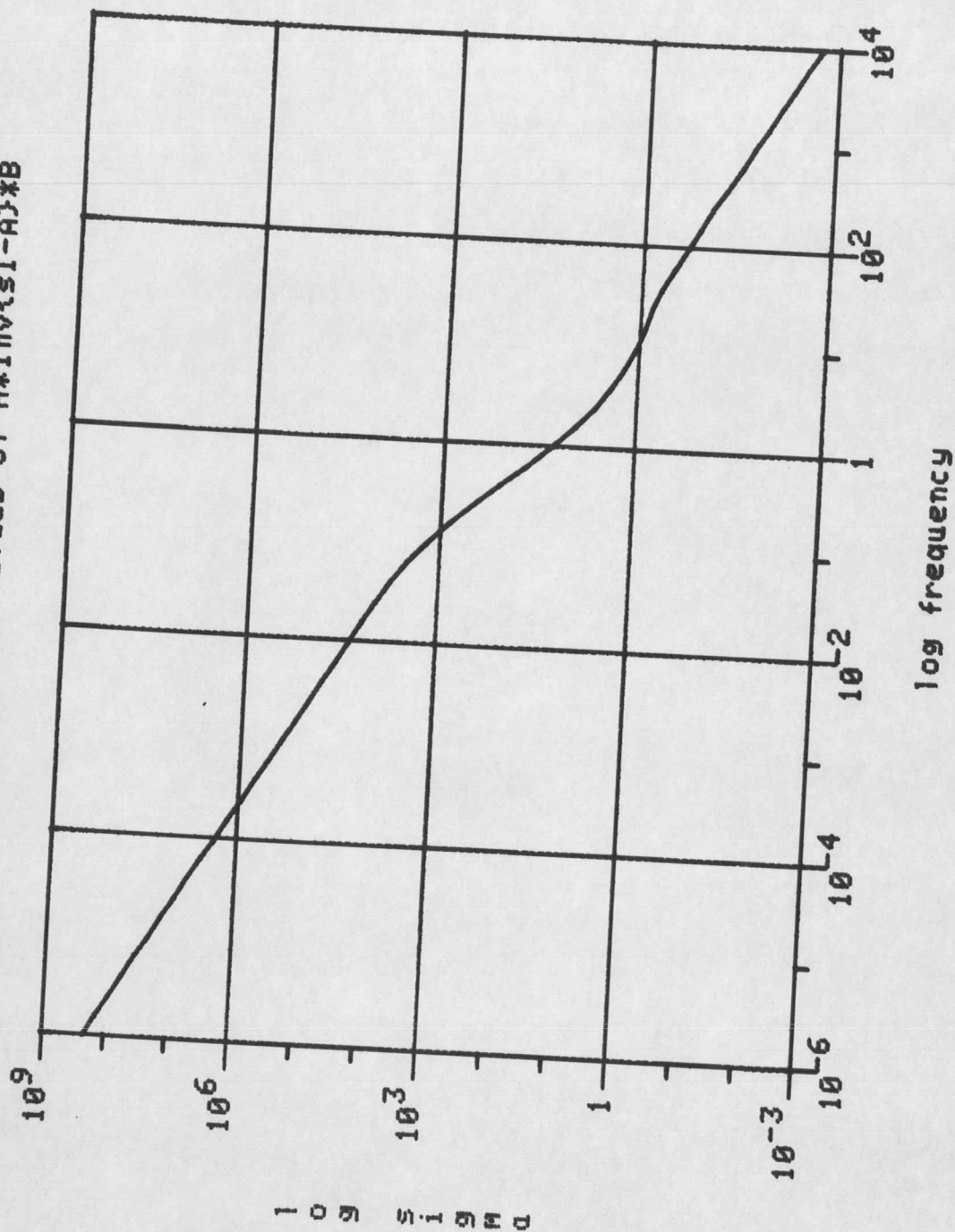
1.4952E+03

9.6864E+02

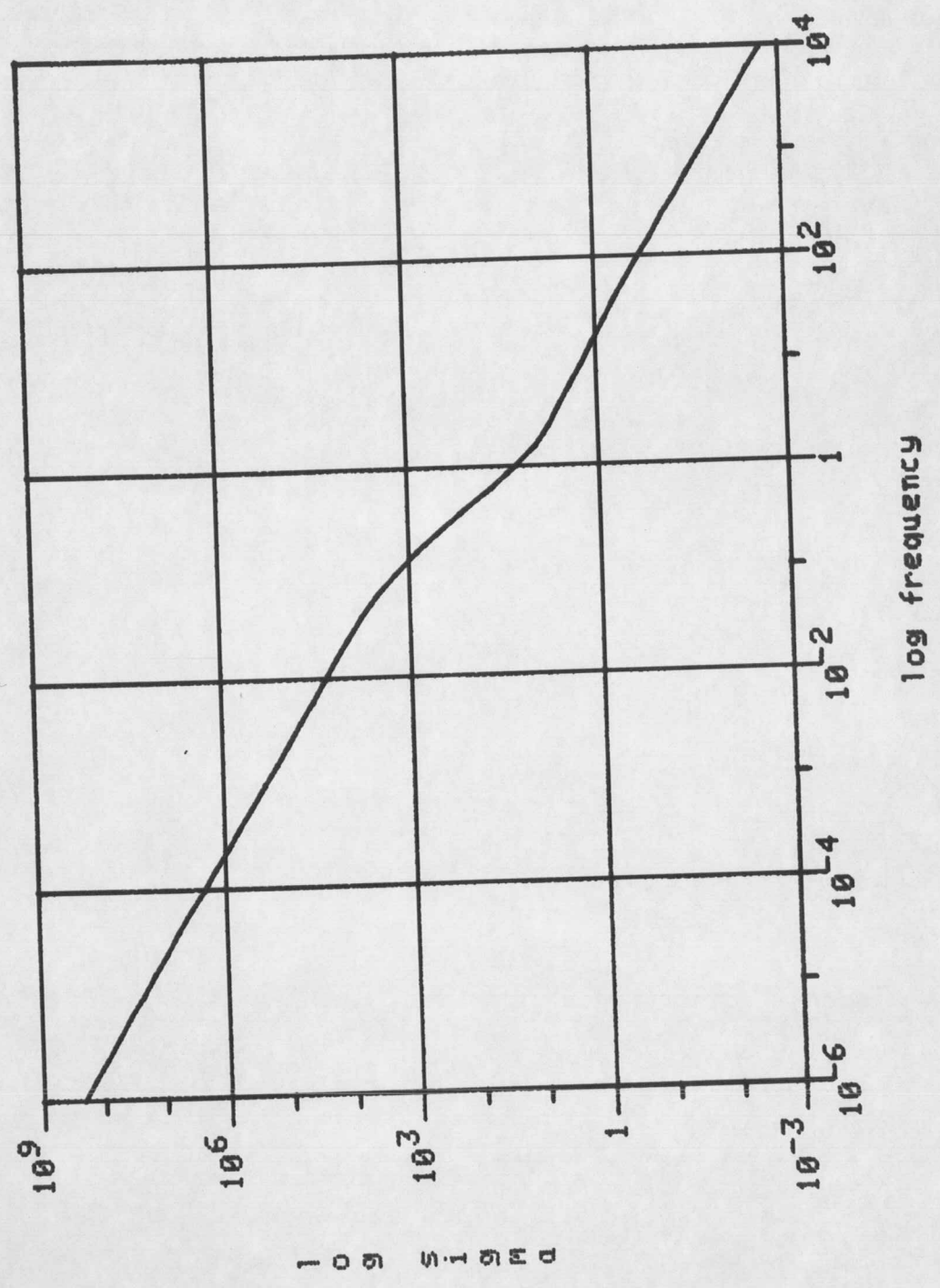
1.8068E+01

1.9009E+02

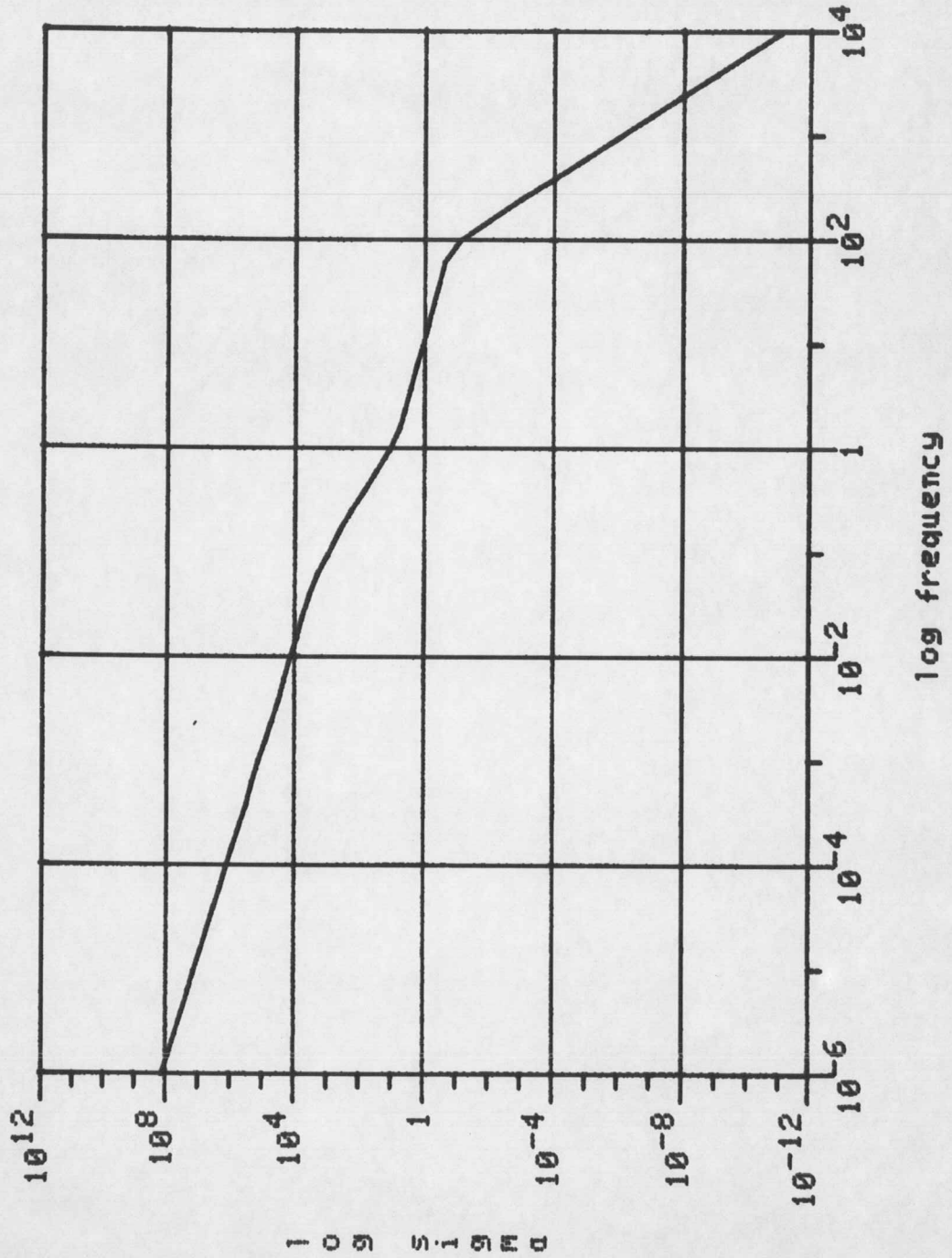
syst: Singular values of $H \cdot \text{inv}\{sI - A\} \cdot B$



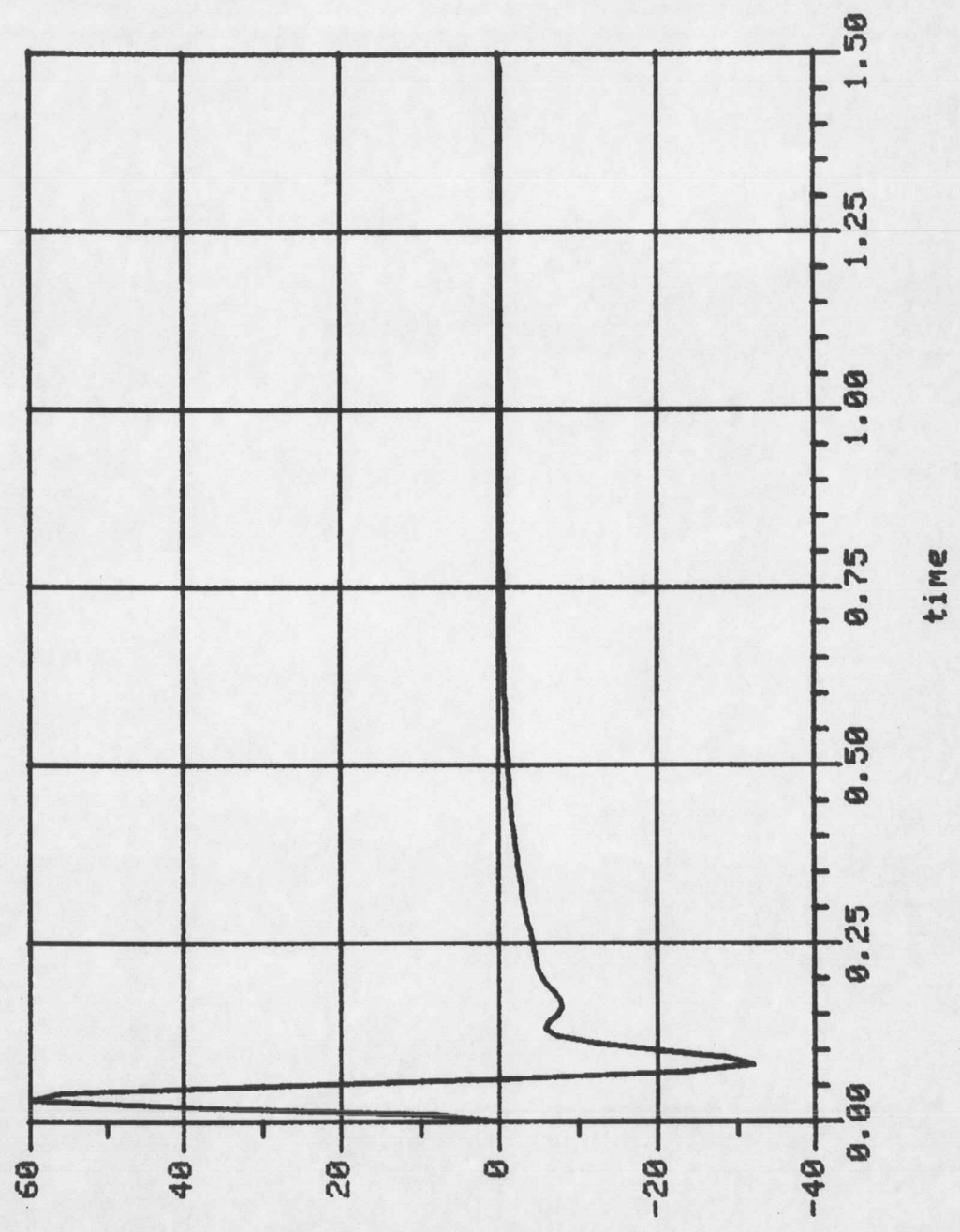
syst: Singular values of $Kc \cdot inv(sI - A)^{-1} \cdot B$



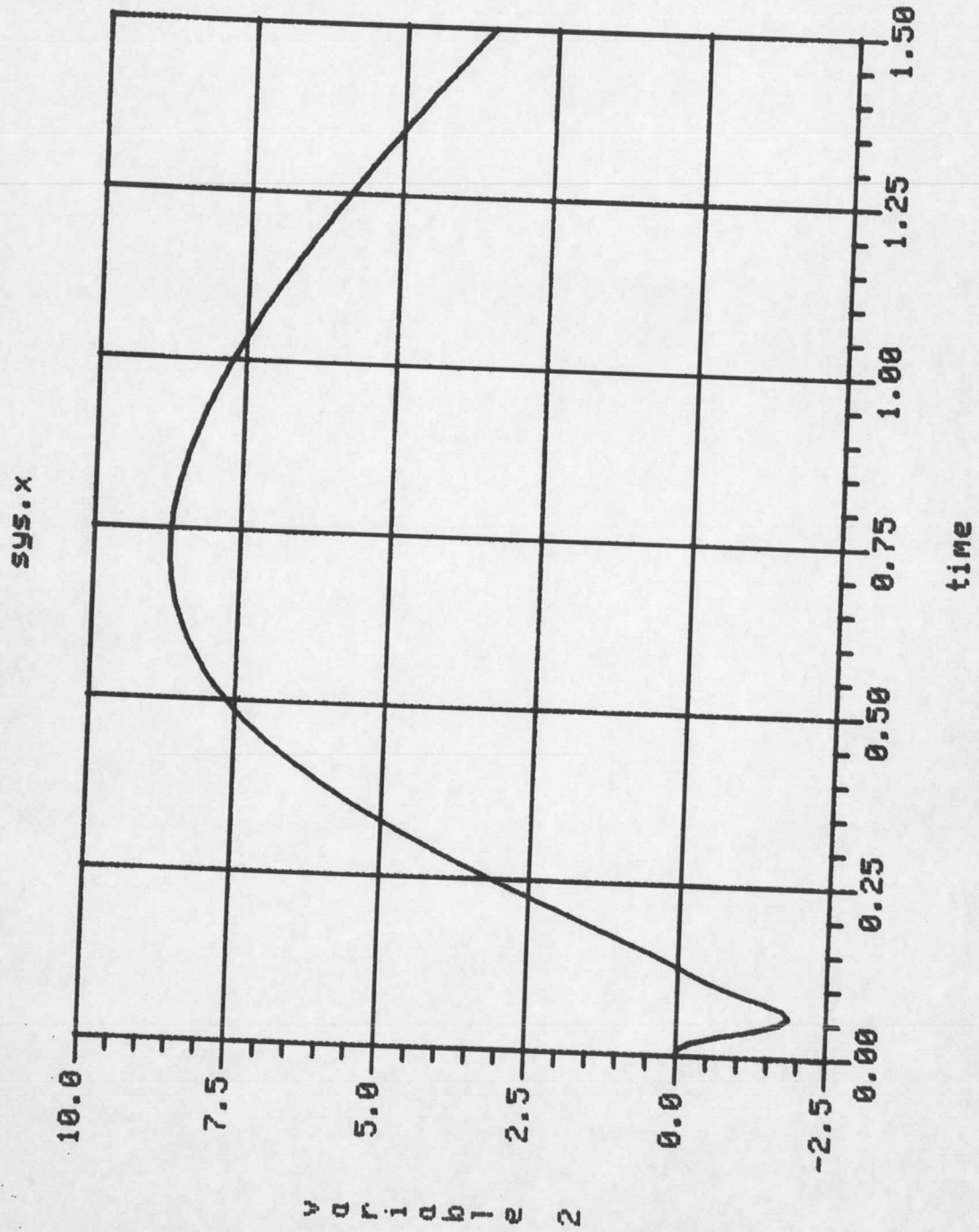
syst: Singular values of $K(s)*G(s)$ $q=10000$

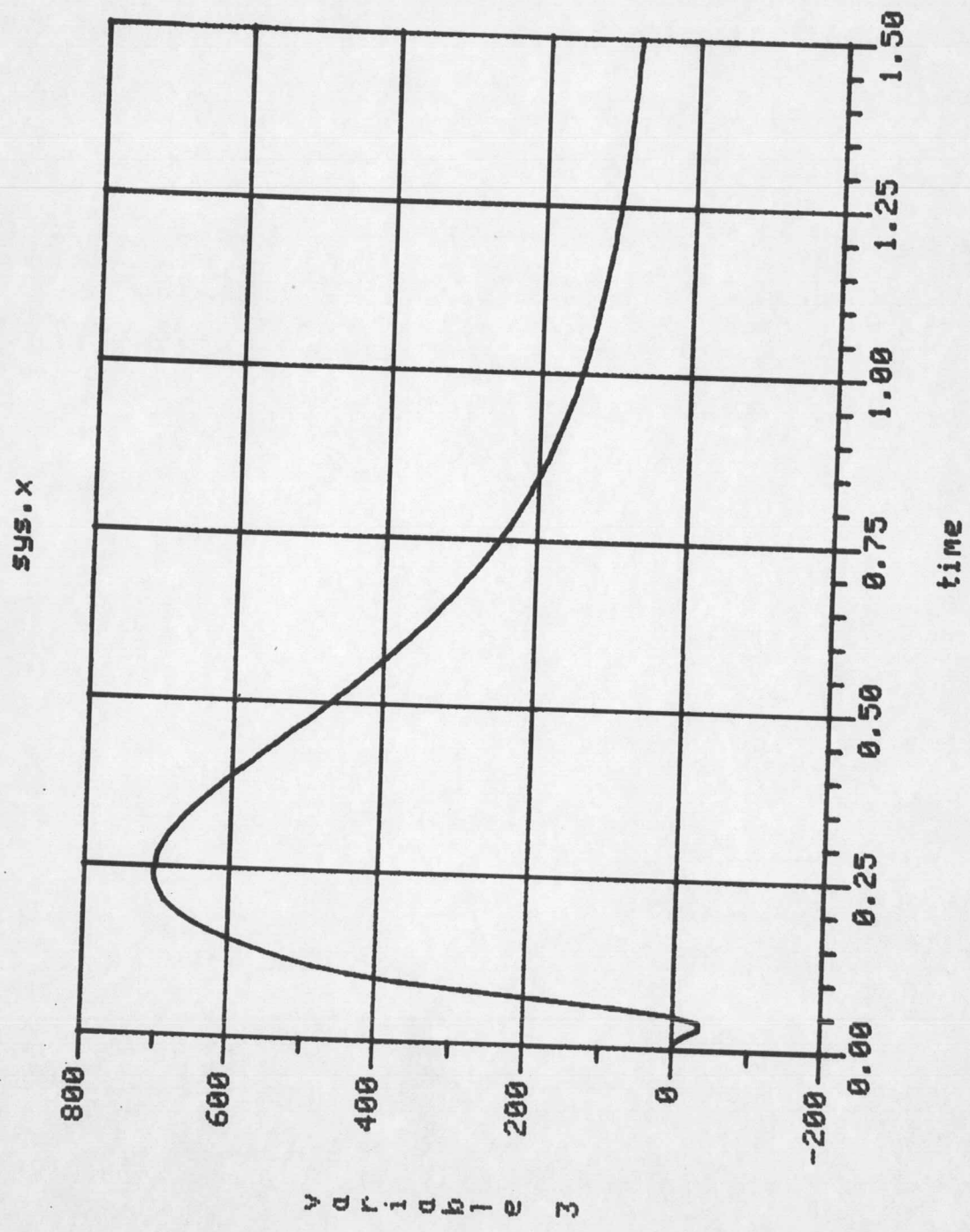


SYS.X

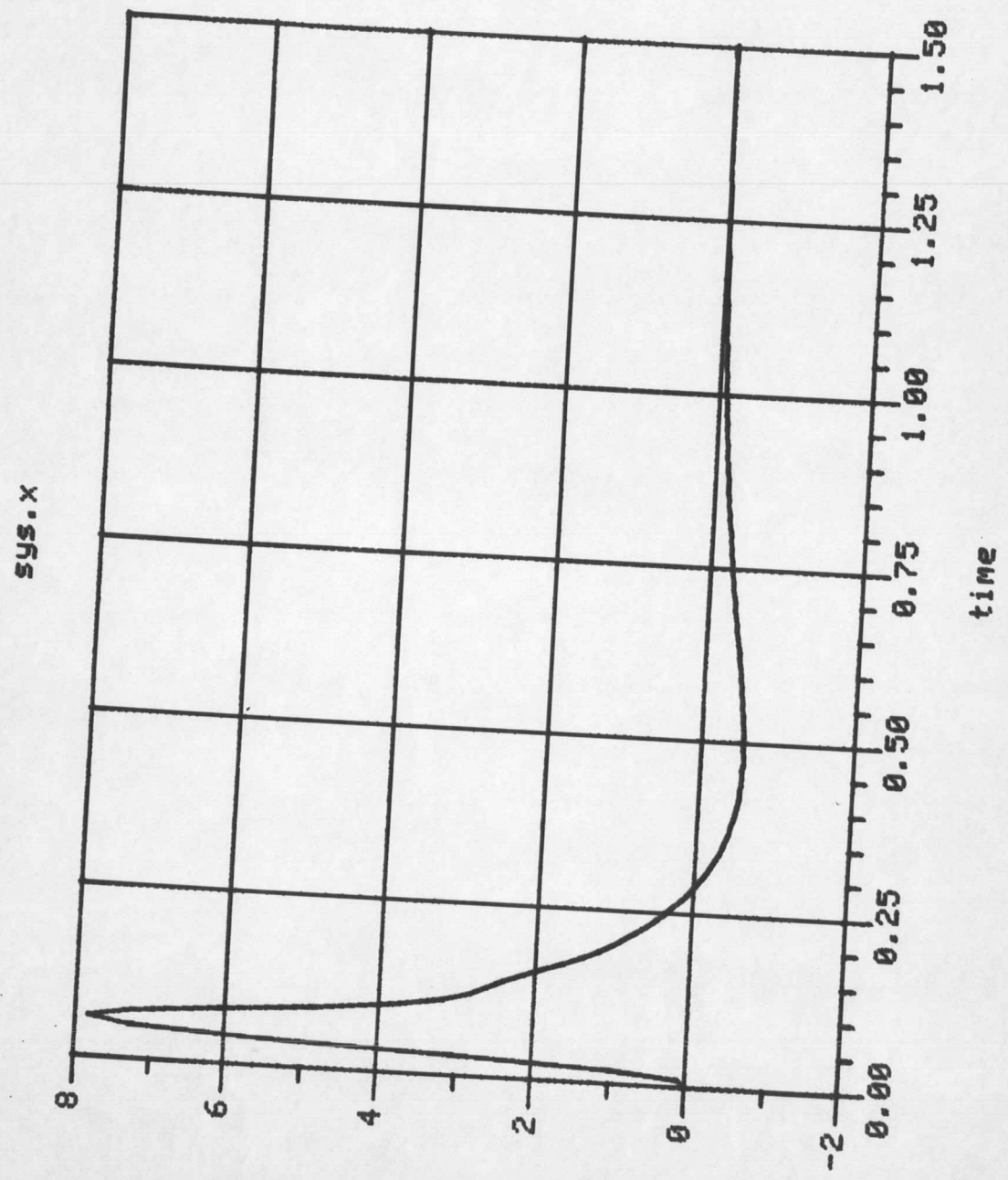


variable 1





variable 3



variable 4

