CR-152237

SYSTEMS CONTROL, INC. (Vt)
1801 Page Mill Road
Palo Alto, California 94304

Telex: 348433

Telephone:
(415) 494-1165

DECEMBER 1978

FINAL REPORT

HUMAN OPERATOR IDENTIFICATION MODEL AND RELATED COMPUTER PROGRAMS

K.M. Kessler
J.N. Mohr

(NASA-CR-152237) HUMAN OPERATOR IDENTIFICATION MODEL AND RELATED COMPUTER PROGRAMS Final Report (Systems Control, Inc., Palo Alto, Calif.) 145 p HC A07/MF A01 CSCL 05H G3/54

N79-16551

Unclas
13150

Prepared for

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Ames Research Center
Moffett Field, California 94035

Under Contract No. NAS2-9754

TABLE OF CONTENTS

PRECEDING PAGE BLANK NOT FILMED

I. INTRODUCTION

This final report documents four computer programs developed and delivered under NASA ARC contract #NAS2-9754. All of these programs provide copmputational assistance in the analysis of man/machine systems. The programs are as follows:

(1) Modified Transfer Function Program (TF)

(2) Time Varying System Response Program (TVSR)

(3) Optimal Simulation Program (TVOPT)

(4) Linear Identification Program (SCIDNT)

All of these programs have been installed and are operational on the TSS-360 System at NASA-ARC. The Modified Transfer Function Program is a result of modifications to a previously delivered computer program described in "Human Operator Controller Model – (HOCOM); User's Guide," by J. Mohr and K. Kessler dated April 1978. The program converts the time domain state variable system representative to frequency domain transfer function system representation. This program is described in more detail in Section II.

The Time Varying System Response Program (TVSR) computes time histories of the input/output responses of the human operator model. It does this by means of a closed loop simulation derived from the system matrices calculated from the modified transfer function program. The link between these two programs is provided by writing the output of TF to output disc for storage. The simulation program reads this data and computes time histories. The simulation program was coded in such a way as to receive generic input data. That is, any linear system description can be written to disc (from card input, if desired) and read by the simulation program for subsequent processing. Specifically, Appendix A lists a stand-alone program which can be used to create a general output disc file to be read as input for the TVSR program. Due to the fact that the input data is obtained from a disc file, the TVSR program features a powerful mechanism to accurately simulate time varying manual systems. System matrices are read from disc for start and stop times. At appropriate intervals, the TVSR program interpolates between times to produce a smooth transition that approximates a time varying system. Several of these piece-wise linear constant coefficient simulations can be "pieced" together to create extremely versatile code. Section II describes the use of this linear simulation program (TVSR).

The third computer program is an optimal simulation program (TVOPT), described in Section IV. It is similar to the program TVSR in that it produces time histories of system states associated with an operator in the loop system. The first major difference of these programs is that TVOPT explicitly accounts for the reaction time of the operator whereas the TVSR program approximates the delay/prediction process by a lumped linear system. The second basic difference is that TVOPT was designed to operate as a stand-alone program. The system description is directly read from input cards whereas the TVSR program obtains its data from disc data created by the transfer function program or by some other means (see Appendix A). The third difference is that TVOPT cannot interpolate and, thus, represent systems constrained to a constant coefficient system. The last difference between these two codes is TVSR computes the expected value (and 1σ uncertainties) for the states as well as sample time histories, whereas TVOPT is limited to only sample time histories.

Both programs TVSR and TVOPT write the input/output time histories for the operator to disc for subsequent processing by the fourth computer program, the linear identification algorithm (SCIDNT). This last program is an open loop identification code which operates on the simulated data from TVOPT (or TVSR) or real operator data stored from motion simulators in an appropriate fashion. This code produces estimates of the various parameters associated with the operator model and is a modified version of the SCI Identification Program (also named SCIDNT) that has been delivered to various other government agencies. This program is described in Section V.

Each program description is self-contained within the appropriate section. The description includes a sample input, the correct procedure to execute the various programs on the TSS system, library routines and input/output disc files where appropriate. In the descriptions to follow, it is assumed the reader is familiar with the theoretical considerations involved with the algorithms; in particular, it is assumed the reader has knowledge of the optimal control human operator model, which these programs utilize.

In most cases, the notation used in the computer programs (as well as this user's manual) conforms to the widely used F, G, H etc. matrices defined in the optimal control literature. The operator notation also generally follows that used in the literature. The one exception is the SCIDNT program which uses different notation <u>internal</u> to the program. However, even in this case, every attempt was made to modify the output labels in this program to conform to those used in the transfer function and simulation programs.

II. MODIFIED TRANSFER FUNCTION PROGRAM (TF)

2.1 DESCRIPTION

The Modified Transfer Function Program (TF) converts a time domain state variable system representation to frequency domain transfer function system representation. The transfer equations may be computed for:

(1) plant/noise model only (no human model)

(2) human model only - open loop

(3) combined plant/noise/human model - open loop

(4) combined closed loop plant/noise/human

In addition, various frequency domain analysis aids are also computed. These include:

(1) plant/human operator state dynamics eigensystem

(2) printer plots/tables of Bode magnitude and phase diagrams of any specified transfer function element arising from systems 1,2,3,4, above

(3) poles/zeros of any specified transfer function element

(4) residue at all poles for any specified transfer function element.

The TF computer code also computes the covariance matrix associated with the closed loop system dynamics (plant/noise/human model).

The computer code has also been designed in conjunction with additional computer code (also delivered to Ames Research Center) called Time Varying System Response Program (TVSR) described in the next section. An option is available to write to disc, intermediate output from TF. The program TVSR then

PRECEDING PAGE BLANK NOT FILMED

reads this data and computes the time varying system response of user requested states and their associated 1σ uncertainties.

2.1.1 Program Functions

The basic options are determined by the system configuration specified by the user. Transfer functions for multi-input/multi-output plant/noise/human models may be computed for the following cases (see Figure 2.1):

(1) open loop plant dynamics only (open loop)

(2) human operator dynamics only (open loop)

(3) human operator/plant dynamics (open loop)

(4) closed loop system dynamics

(5) covariance matrix

The equations representing the above user options are given in Table 2.1.

The basic structure of the program allows for repeated application of generic transfer code which utilizes a constant coefficient differential system of the form:

$$\dot{x} = Fx + Gu$$

$$y = Hx + Du \qquad (2.1)$$

where x is an n component state vector, u is an m component input vector, and y is a p component output vector.

The frequency domain representation of the same system in terms of the complex frequency s is:

$$y(s) = T(s)\, u(s) \qquad (2.2)$$

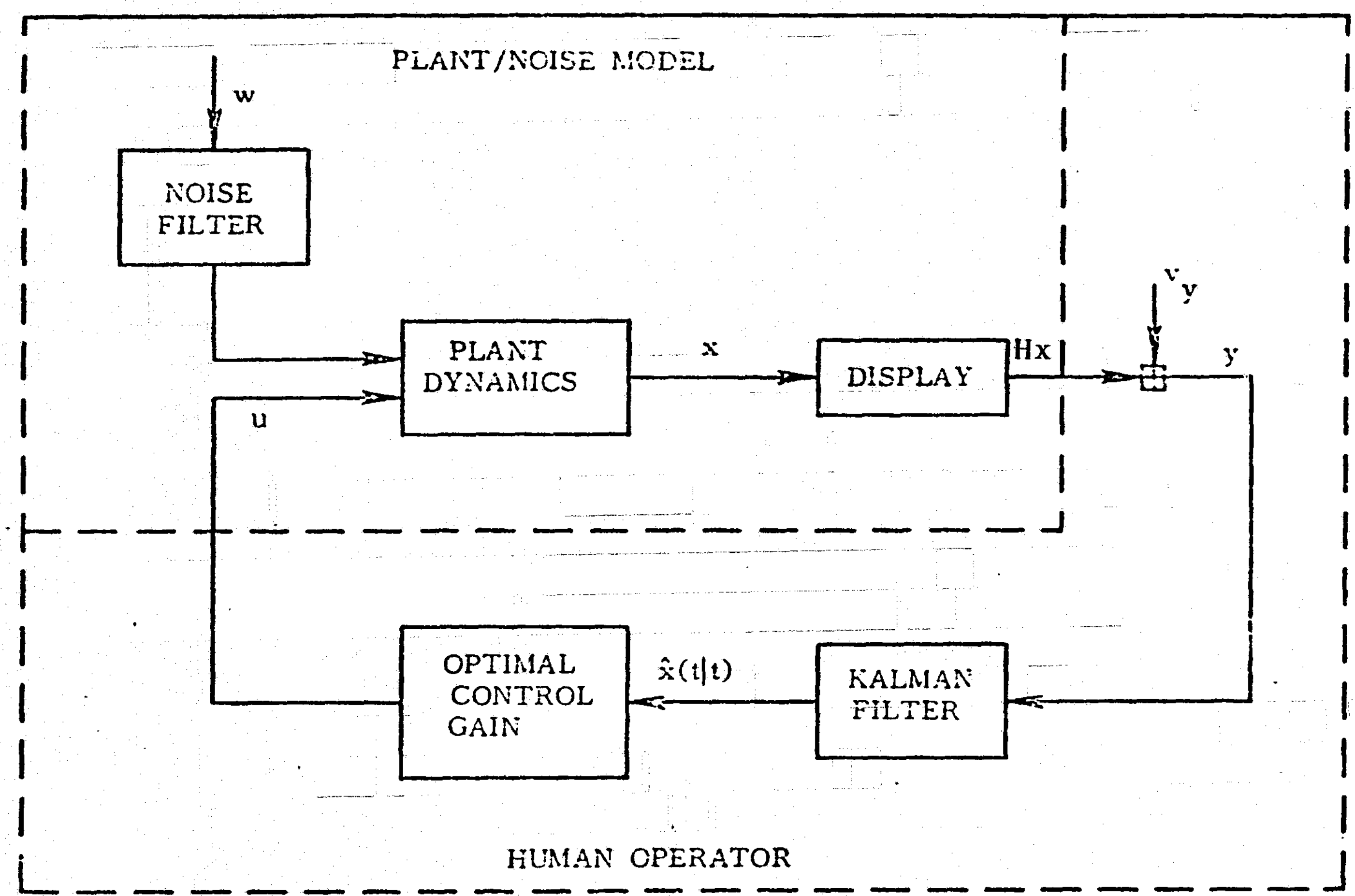


Figure 2.1 Structure of the Human Operator Controller Model used in TF

Table 2.1
Dynamic System Equations

Plant/Noise Model

$$\dot{x} = F_p x + G_p u + \Gamma_p \omega$$

$$y = H_p x + D_p u + v_y$$

$w \sim N(0,W)$; $v_y \sim N(0, Vy)$

Human Operator Model

$$\dot{\hat{x}} = (F_h - G_h\Lambda - KH_h - KD_h\Lambda)\hat{x} + Ky$$

$$u_d = -\Lambda\hat{x}$$

with augmented states

$$\dot{x}_a = F_a x_a + G_a u_d + CC v_a$$

$$u = H_a x_a + D_a u_d + BC v_a$$

where K satisfies

$$0 = F_h S + SF_h^T + \Gamma_h W \Gamma_h^T - SH_h^T Vy^{-1} H_h S$$

$$K = SH_h^T Vy^{-1}$$

and Λ satisfies

$$0 = PF_h + FP_h^T + Q_h - PG_h g^{-1} G_h^T P$$

$$\Lambda = g^{-1} G_h^T P$$

Table 2.1 (Concluded)

optimal cost functional

$$J = Jx + Ju$$

where

$$Jx = E \int_0^{\infty} x^T Q_h x \, dt$$

$$Ju = E \int_0^{\infty} u^T g u \, dt$$

<u>Closed Loop System</u>

$$\dot{z} = F_c Z + G_c \theta$$

$$y_c = H_c z$$

where

$$z = \begin{bmatrix} x \\ \hat{x} \\ x_a \end{bmatrix} ; \quad \theta = \begin{bmatrix} \underline{w} \\ \underline{v}_y \\ \underline{v}_a \end{bmatrix}$$

$$F_c = \left[\begin{array}{c|c|c} F_p & -G_p D_a \Lambda & G_p H_a \\ \hline KH_h & F_h - KH_h - G_h D_a \Lambda & G_h H_a \\ \hline 0 & -G_a \Lambda & F_a \end{array}\right]$$

$$G_c = \left[\begin{array}{c|c|c} \Gamma_p & 0 & G_p BC \\ \hline 0 & K & G_h BC \\ \hline 0 & 0 & CC BC \end{array}\right]$$

where $T(s)$ is a matrix of complex, rational expression, transfer functions between elements of the input $u(s)$ and the output $y(s)$. $T(s)$ can be expressed in terms of F, G, H, and D as:

$$T(s) = H(sI - F)^{-1}G + D$$

$$= \frac{1}{\Delta(s)}[H \operatorname{adj}(sI - F)G + \Delta(s)D] \qquad (2.3)$$

$\Delta(s)$ is the characteristic polynomial of the system and can be expressed as:

$$\Delta(s) = a_o + a_1 s + a_2 s^2 + \ldots + a_{n-1}s^{n-1} + s^n \qquad (2.4)$$

or as

$$\Delta(s) = (s-p_1)(s-p_2)\ldots(s-p_n) \qquad (2.5)$$

The p_k $k=1,\ldots,n$ are the poles of the system, Eq. (2.1). The adjoint of $(sI-F)$ is a matrix such that:

$$\frac{\operatorname{adj}(sI-F)}{\Delta(s)} = (sI-F)^{-1} \qquad (2.6)$$

In general, the elements of $\operatorname{adj}(sI-F)$ are polynomials in s and can be expressed in the form:

$$[\operatorname{adj}(sI-F)]_{ij} = b_{ijo} + b_{ij1}s + b_{ij2}s^2 + \ldots + b_{ij\ell}s^{\ell}$$

$$= (s-z_{ij1})(s-z_{ij2})\ldots(s-z_{ij\ell})$$

$$\triangleq N_{ij}(s) \qquad (2.7)$$

The z_{ijk} are the zeros of the transfer function between y_i and u_j. The number of zeros, ℓ, is less than or equal to n. Given the matrices F, G, H, and D, the program can compute the n coefficients a_k in the characteristic equation and the n+1 pxm arrays of coefficients b_{ijk}.

This generic transfer function code utilizes the Leverier algorithm described in the literature. Typically, systems on the order of 18-20 states can be determined with excellent accuracy utilizing both forward and backward computations. Numerical problems can and do occur with systems larger than 20 states.

In principal, then, transfer functions associated with (1) plant/noise model, (2) human operator model, and (3) closed loop system are restructured so as to appear as Eq. (2.1). A slight variation on this common procedure is used to compute the open loop plant/noise/human model transfer functions. For this case, the open loop plant/noise transfer functions are computed followed by the open loop human model transfer function. The associated adjoint matrices are multiplied together to produce the resultant system (the characteristic equation for each is similarly multiplied).

2.1.2 Computational Aids

Other computational aids for systems analysis are also available. These fall in the general categories of:

(1) State Dynamics Eigensystem

(2) Frequency Domain Analysis Aids

(3) Covariance Matrix Calculation of the Closed Loop System

(4) Optimal Cost Function of the Closed Loop System

These are briefly described in the following subsections.

2.1.2.1 State Dynamics Eigensystem

Given the state dynamics matrix of the time domain representation of a linear system, the program is capable of determining the complete eigensystem. The method of Householder orthogonal transformations is used. The following parameters are computed:

(1) Eigenvalue (pole) locations in the complex plane

(2) Natural frequency of each pole (magnitude of the eigenvalue)

(3) Damping factor of each pole $-RE(\lambda)/\|\lambda\|$

(4) Eigenvector corresponding to each eigenvalue. The eigenvector is normalized so that its largest component has unity magnitude

(5) Magnitude of each (possibly complex) component of each eigenvector

(6) Phase angle of each component of each eigenvector relative to the largest component of its own eigenvector

2.1.2.2 Frequency Domain Analysis Aids

The user may specify elements of the transfer function matrix which he wishes to be analyzed in detail. For each element specified, the following options are available.

<u>Bode Plots</u>

The program can produce printer plots of the Bode magnitude and phase diagrams for the transfer function element. The user must specify the maximum and minimum frequency for the plot. The user also has the option to plot the transfer function element multiplied by an arbitrary numerator and denominator polynomial in s.

Zeros

The program can compute the zeros of the transfer function element. These are the possibly complex roots of $N_{ij}(s) = 0$. The D.C. gains are also computed for convenience.

Residues

The program can compute the residues of the transfer function element evaluated at all of the system poles. The residue of the i,j element at the kth pole is defined as:

$$r_{ijk} = \lim_{s \to p_k} (s-p_k)T_{ij}(s)$$

2.1.2.3 Covariance Matrix Calculation of the Closed Loop System

The program computes the covariance matrix associated with the closed loop system. It utilizes the Q-R algorithm developed as part of the OPTSYS computer program. This algorithm fails with zero eigenvalues. A standard procedure is to introduce small perturbations on the order of 10^{-10} to 10^{-15} (depending on the size of the matrix) in selected elements of the system matrix so as to slightly move the eigenvalues away from zero.

2.1.2.4 Optimal Cost Function of the Closed Loop System

The program computes the individual costs JU, JX as well as the sum J (as defined in Table 2.1).

2.2 UTILIZATION

The Modified Transfer Function Program (TF) is very similar to a previous delivered transfer function computer program (see Section I). Basically, this new version replaces the augmented states, labeled x_a in Table 2.1 of the above referenced document, by new augmented states arising from a linear system approximation (Padé approximation) for the transfer function between the control output, u, and the desired operator output, u_d. To see how this is accomplished, refer to Figure 2.2. Here, the Standard Optimal Control Model (SOCM) is shown together with the linear system approximation required for the TF program. These blocks shown in Figure 2.2 must be lumped into an equivalent system; given by the augmented states listed in Table 2.1.

Combining the filter states with the augmented states yields the open loop transfer function of the operator in the form:

$$\frac{d}{dt}\begin{bmatrix} \hat{x} \\ x_a \end{bmatrix} = \begin{bmatrix} F^+ & (G_h - KD_h)H_A \\ -G_A\Lambda & F_A \end{bmatrix}\begin{bmatrix} \hat{x} \\ x_a \end{bmatrix} + \begin{pmatrix} K \\ 0 \end{pmatrix} y$$

$$u = (-D_A\Lambda \;\vdots\; H_A)\begin{pmatrix} \hat{x} \\ x_a \end{pmatrix} + BC\, v_a$$

where

$$F^+ = F_h - KH_h - (G_h - KD_h)D_a\Lambda$$

The open loop operator transfer function, u/y, is now computed rather than that discussed in the above referenced report. The human operator/plant dynamics (open loop) option now relies on u/y (given above) in the calculations.

Example

An example is given to show how the input matrices (provided by the user) are determined.

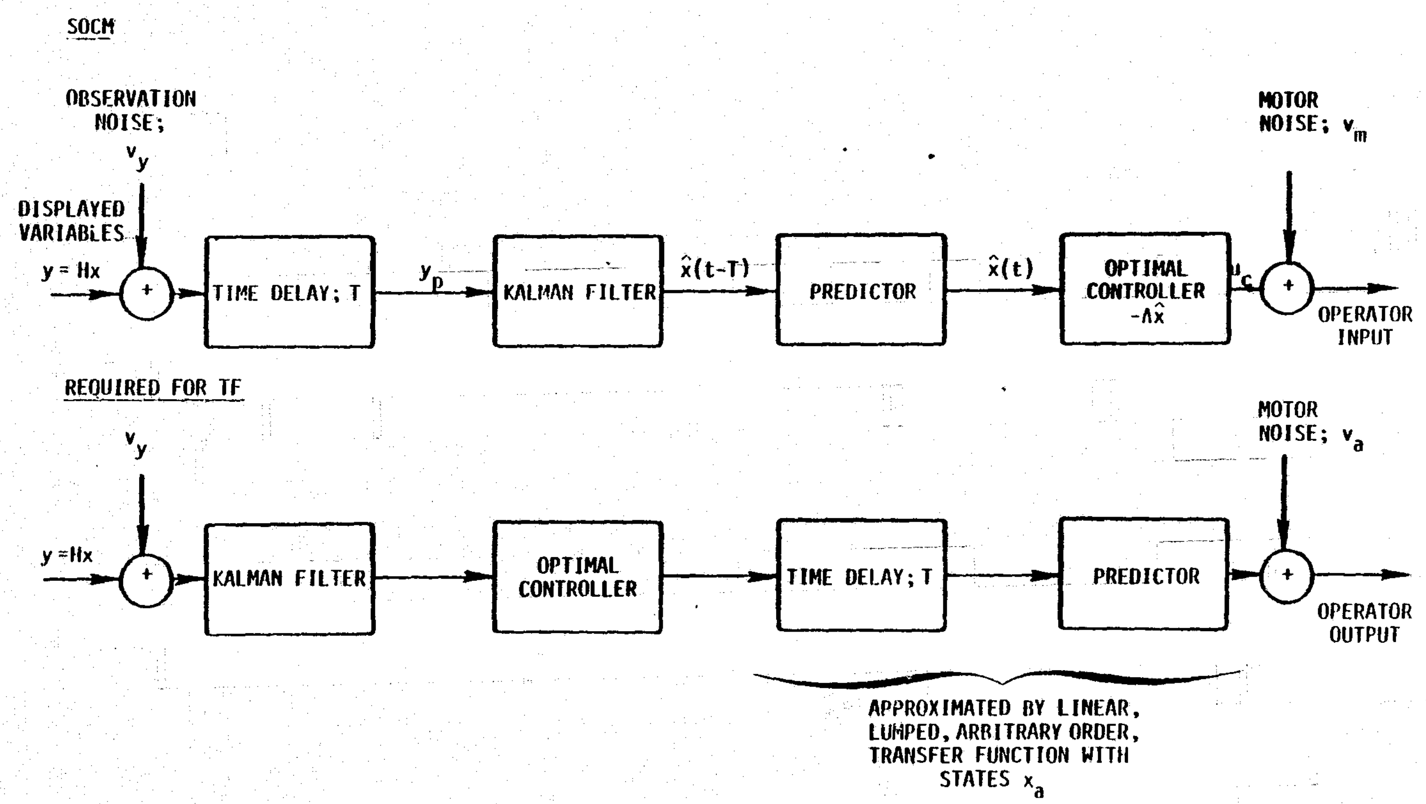


Figure 2.2 Human Operator Models

plant

$$\dot{X}_1 = -2X_1 + w \quad ; \quad \overline{w} = 0$$

$$\dot{X}_2 = X_1 + u$$

$$y = X_2 + v_y \quad ; \quad \overline{v}_y = 0$$

$$y_p(\text{perceived}) = y(t-T) \quad ; \quad T = \text{delay time}$$

The user must input values for

$$W = E(ww^T)$$

$$V_y = E(v_y \, v_y^T)$$

as well as Q, g arising from the optimal cost

$$J = \int \{\|X\|_Q^2 + \|u\|_g^2\} dt$$

To obtain a linear system approximation to the predictor (in the Kalman filter) and the delay in the control, we approximate the predictor as 1+Ts and the delay from a second order Padé approximation as

$$f(T) \cong \frac{6 - 2Ts}{6 + 4Ts + T^2s^2}$$

The predictor and Padé approximation are lumped into an equivalent transfer function of the form:

$$\frac{u}{u_d} = \left(\frac{6 - 2Ts}{6 + 4Ts + T^2s^2}\right)(1 + Ts)$$

$$= \frac{6 + 4Ts - 2T^2s^2}{6 + 4Ts + T^2s^2}$$

ORIGINAL PAGE IS
OF POOR QUALITY

This can be put into matrix form as

$$\frac{r}{u_d} = \frac{1}{6 + 4Ts + T^2s^2}$$

$$u = (6 + 4Ts - 2T^2s^2)r$$

and, after some manipulation,

$$\dot{x} = F_a X_a + G_a u_d + CC v_a$$

$$u = H_a X_a + D_a u_d + BC v_a$$

where

$$X_a^T = [r \ \dot{r}],$$

$$F_a = \begin{bmatrix} 0 & 1 \\ \frac{-6}{T^2} & \frac{-4}{T} \end{bmatrix} ; G_a = \begin{bmatrix} 0 \\ \frac{1}{T^2} \end{bmatrix} ; \Gamma_a = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$H_a = [\ 18 \ \ 12T\ ] \ ; \ D_a = -2$$

and BC is selected to produce "motor noise," v_a, of sufficient amount desired by the user.

Observe that the prediction and delay functions of the operator are lumped into one transfer function, currently up to order 5. This is only possible because of the linear assumptions for these processes.

The output data from the modified transfer function program is identical to that used in the original version. The input data is slightly modified to include data from the augmented states given by the matrices F_A, G_A, Γ_A, H_A, D_A, B_A, and V_A (= cov v_A).

Features

There are several features embodied in the code that require some discussion.

(1) It is possible to iterate on the values of V_y and V_A so as to converge to a solution (see Table 2.2, Group 1)

$$V_y = \rho_y E(\overline{y^2})$$

and

$$V_a = \rho_a E(\overline{y^2})$$

Either V_y (only) or V_a (only) or both, may be iterated. The iteration utilizes the subroutine MITER which in turn calls subroutines GRADE, ITERCK and AITERY. The algorithm for the iteration (in GRADE) for both V_y and V_a is

$$V(new) = 10^{**}[\log_{10} \hat{V} + \log_{10} V(old)]$$

where V is a generic representation of each diagonal element of the covariance matrices V_y or V_a. $\hat{V}$ is given by

$\underline{V_y}$

$$\hat{V}_i = \rho_y \text{ XCOV } (i,i) \; ; \; i = 1,2,\ldots,n$$

$$n = \# \text{ of elements in } V_y$$

$$\rho_y = \text{constant}$$

$$XCOV = E(y_c \, y_c^T); \text{ (see Table 2.1)}$$

$\underline{V_a}$

$\hat{V}_i = \rho_a$ XCOV (i,i) ; i = j+1,j+2,...,m

ρ_a = constant

j = # of measurements of operator (size of H_h)

m-j = # of elements in V_a

Modifications to this algorithm may be included in GRADE to improve convergence.

The "stopping" criterion (determined in ITERCK) is to check the diagonal elements for

$$\left| \frac{V(new) - V(old)}{V(old)} \right| \leq 0.03$$

for all elements. (That is, all elements must satisfy this criterion to stop the iteration process.) A dummy subroutine (AITERY) is coded to allow for further modifications of V_y to incorporate the utilization of perception thresholds in the display.

The user must be careful to structure the matrix H_c, since XCOV is the covariance matrix of $y_c = H_c Z$ and not Z. (If $H_c = I$, the two matrices are equivalent.)

(2) An arbitrary polynomial (up to order six) of the form N(s)/D(s) may be multiplied by any transfer function prior to obtaining a Bode plot. This provides additional plotting capability for the user.

(3) The input switch FCNTRL=T (in the input, Group 1 cards) causes the structure of F_c to be slightly modified. Using this option, F_c is given by:

$$F_c = \left[\begin{array}{c|c|c} F_p - G_p D_a \Lambda & 0 & G_p H_a \\ \hline KH_h & F_h - KH_h - G_h D_a \Lambda & G_h H_a \\ \hline -G_a \Lambda & 0 & F_a \end{array}\right]$$

It can be shown that this structure effectively bypasses the Kalman filter and the operator now only contains the optimal controller equations. This is obvious by inspection by noting $Z^T = [X^T \hat{X}^T X_a^T]$, and F_c above decouples the filter states $\hat{X}$ from the closed loop system. (The filter terms are still included, but are decoupled from the system.) This implementation was used to minimize changes to the structure of F_c.

2.3 INPUT SPECIFICATIONS

The program receives its input through cards and disk files.

2.3.1 Card Input

There are two cards required for each run. Card 1 is a title card for output identification. Card 2 is the option specification card. All succeeding input is dependent upon the options selected on card 2. All matrices utilizing card input are read in by rows. The default format is 8E10.4. All matrices utilizing disk file input are read in by rows. The default format is 4E20.13. Table 2.2 describes both the required and optional card input.

2.3.2 Disk Files

Depending on which options the user elects, the program can use up to seven disk files (logical units 9-14, 21).

2.3.2.1 Reading F and G (Plant Dynamics) from Disk (Unit 9)

The unit for reading F and G (plant dynamics) from disk is referred to as IUFG and has a default value of 9. This unit need only be assigned when DISK = T.

IUFG is a formatted file. F and G must be written by rows. They are read by a variable format stored in array IFMT2(13). The default value is (4E20.13) and may be altered via NAMELIST/OPTION/.

2.3.2.2 Scratch Unit (Unit 10)

Logical Unit 10 is a scratch file. It should always be assigned as a temporary file.

2.3.2.3 Output from Plant Dynamics (Unit 11)

Output from plant dynamics calculations is stored on a file referred to as IUP and has a default value of 11. This file contains the characteristic equation and the numerator matrices. It should be assigned whenever PLANT = T. It will usually be a temporary file. If it is assigned as a permanent file, on a subsequent run, the user may set JENTP = 1 on Card 2, Group 2 to produce further transfer functions without redoing previous calculations.

2.3.2.4 Output from Human Operator (Unit 12)

Output from the human operator is stored on a file referred to as IUH and has a default value of 12. It should be assigned whenever HUMAN = T. Its description and use is the same as for Unit 11, Section 2.3.2.3.

(Note: If assigned as a permanent file, the user can use file IUH as if it contained output from plant dynamics, and use JENTP = 1 on succeeding runs to get additional transfer functions.)

2.3.2.5 Output from Human Operator + Plant Dynamics (Unit 13)

Output from the human operator + plant dynamics is stored on a file referred to as IU3 and has a default value of 13. It should be assigned whenever HOPLT = T. Its description and use is the same as for Unit 11, Section 2.3.2.3 (see note, Section 2.3.2.4).

2.3.2.6 Output from Closed Loop (Unit 14)

Output from the closed loop is stored on a file referred to as IUC and has a default value of 14. It should be assigned whenever CLOSED = T. Its description and use is the same as for Unit 11, Section 2.3.2.3 (see note, Section 2.3.2.4).

2.3.2.7 Writing FC, GC, HC, WBIG, VY, VA to Disk (Unit 21)

The unit for writing FC, GC, HC, WBIG, VY and VA (if used) is referred to as IUSYS and has a default value of 21. It is a binary file and need only be assigned whenever SAVSYS = T.

Table 2.2
Card Input

GROUP NO.	CARD NO.	FORMAT	VARIABLE NAME	TYPE	DESCRIPTION
1	1	13A4	ITITLE	A	52 character title to be printed at the top of each output page
	2	NAMELIST /OPTION/	PLANT	L	= T, required for plant dynamics calculations (Default = F)
			HUMAN	L	= T, required for human operator calculations (Default = F)
			HOPLT	L	= T, required for human operator and plant dynamics calculations (Default = F)
			CLOSED	L	= T, required for closed loop calculations (Default = F)
			COVAR	L	= T, required to compute covariance (Default = F)
			MINOUT	L	= T, to obtain "minimal" output (Default = F)
			BACKWD	L	= T, to use backward Leverrier algorithm to compute adjoint matrix for no. of states < 5. (Automatically uses backward Leverrier algorithm if no. of states ≥5). (Default = F)
			DISK	L	= T, if plant dynamics F and G matrices are to be read from Unit 9.
			SAVSYS	L	= T, to write FC, GC, HC, W, Vy, VA to Unit 21 in binary format (Default = F)
			JUNIT	I	Unit no. all matrices (except F and G for plant dynamics if DISK = T) and matrix elements will be read from (Default = 5)
			IFMT(13)	A	Array containing format to be used when reading matrices from JUNIT (Default = (8E10.4)

Note: All Group and card numbers marked with * are conditional input.

Table 2.2 (Continued)

GROUP NO.	CARD NO.	FORMAT	VARIABLE NAME	TYPE	DESCRIPTION
			IFMT2(13)	A	Array containing format to be used when reading F and G from Unit 9 (Default = (4E20.13))
			ITERC	L	= T, for iterating on Vy and/or Va (Default = F)
			ITERVY	I	No. of iterations to do on Vy (Default = 0)
			ITERVA	I	No. of iterations to do on Va (Default = 0)
			RHO4	D.P.	Constant for Vy iteration (Default = .0314159
			RHOY	D.P.	Constant for Va iteration (Default = .092477)
			FCNTRL	L	= T, to switch FC matrix to control only. (Default = F)
2*					*Include Group 2 only if PLANT = T
	1	13A4	ICMMNT	A	Comment card for user identification of start of plant dynamics input.
	2	8I5	NSP	I	No. of states (≤30)
			NSP	I	No. of controls (≤4)
			NMP	I	No. of measurements (≤10)
			NOP	I	No. of noise (≤4)
			NTFP	I	No. of transfer functions desired (may be 0)
			JEIGFP	I	Eigenvalue flag for F = -1, compute and skip all other plant dynamics calculations = 0, do not compute = 1, compute and continue calculations
			JFSNGP	I	Singularity flag for F

Note: All Group and card numbers marked with * are conditional input.

Table 2.2 (Continued)

GROUP NO.	CARD NO.	FORMAT	VARIABLE NAME	TYPE	DESCRIPTION
					= 0, unknown = 1, F is singular
			JENTP	I	Entry flag = 0, regular run = 1, if adjoint matrices were previously computed and saved, and only specific transfer functions are desired
	3*	6I5			*Include NTFP sets of cards 3-3B
			IRWCLP(1)	I	Row no. of transfer function desired
			(2)	I	Column no. of transfer function desired. If IRWCLP (2) <0, the transfer function produced will be (IRWCLP(1), (-IRWCLP(2)+1)*S.
			(3)	I	= 0, zeros only = 1, zeros, residues = 2, bode plots only = 3, all
			(4)	I	Frequency (in radians/second) of lower bound of bode plot (log)
			(5)	I	Upper bound of bode plot (log)
			(6)	I	= 0, normal size bode plot = 1, double size bode plot (magnitude scale only)
	3A*	(I5,7E10.5)	NCOEFN	I	No. of coefficients in the numerator of the polynomial (N/D) which multiplies the transfer function before plotting. (≤ 6)
			COEFNP	D.P.	Coefficients in descending powers of S *Include only if IRWCLP(3)≥ 2

Note: All Group and card numbers marked with * are conditional input.

Table 2.2 (Continued)

GROUP NO.	CARD NO.	FORMAT	VARIABLE NAME	TYPE	DESCRIPTION
	3B*	(I5,7E10.5)	NCOEFD	I	No. of coefficients in the denominator of the polynomial (N/D) which multiplies the transfer function before plotting. (≤6)
			COEFDP	D.P.	Coefficients in descending powers of S *Include only if IRWCLP(3)≥2
	4-4n*	IFMT	FP	D.P.	F matrix (NSP x NSP) *Include only if JENTP = 0.
	5-5n*	IFMT	GP	D.P.	G matrix (NSP x NCP) *Include only if JENTP = 0, JEIGFP ≠ -1
	6-6n*	IFMT	HP	D.P.	H matrix (NMP x NSP) *Include only if JENTP = 0. JEIGFP ≠ -1
	7-7n*	IFMT	DP	D.P.	D matrix (NMP x NCP) *Include only if JENTP = 0. JEIGFP ≠ -1
	8-8n*	IFMT	GAMMP	D.P.	Γ matrix (NSP x NØP) *Include only if JENTP = 0, JEIGFP ≠ -1
3*					*Include only if HUMAN = T
	1	13A4	ICMMNT	A	Comment card for user identification of human operator input
	2	7I5	NSH	I	No. of states (≤15)
			NCH	I	No. of controls (≤4)
			NMH	I	No. of measurements (≤10)
			NOH	I	No. of noise (≤4)
			NTFH	I	No. of transfer functions desired (may be 0)
			JEIGFH	I	Eigenvalue flag (see Group 2 card 2)
			JFSNGH	I	Singularity flag (see Group 2, card 2)

Note: All Group and card numbers marked with * are conditional input.

Table 2.2 (Continued)

GROUP NO.	CARD NO.	FORMAT	VARIABLE NAME	TYPE	DESCRIPTION
	3*	6I5	IRWCLH(1-6)	I	See Group 2, card 3 *Include NTFH sets of cards 3-3B
	3A*	(I5,7E10.5)	NCOEFN	I	See Group 2, card 3A
			COEFNH	D.P.	See Group 2, card 3A
	3B*	(I5,7E10.5)	NCOEFD	I	See Group 2, card 3B
			COEFDH	D.P.	See Group 2, card 3B
	4-4n*	IFMT	FH	D.P.	F matrix (NSH x NSH)
	5-5n*	IFMT	GH	D.P.	G matrix (NSH x NCH)
	6-6n*	IFMT	HH	D.P.	H matrix (NMH x NSA)
	7-7n*	IFMT	DH	D.P.	D matrix (NMH x NCH)
	8-8n	IFMT	GAMMH	D.P.	Γ matrix (NSH x NØH)
	9-9n	IFMT	QH	D.P.	Q matrix (NSH x NSH)
4*					*Include only if HUMAN = T
	1	13A4	ICMMNT	A	Comment card for user identification of augmented states input
	2	I5	NSA	I	No. of augmented states
	3	IFMT	FA	D.P.	FA matrix (NSA x NSA)
	4	IFMT	GA	D.P.	GA matrix (NSA x NCH)
	5	IFMT	HA	D.P.	HA matrix (NCA x NSA)
	6	IFMT	DA	D.P.	DA matrix (NCH x NCH)
	7*	IFMT	VA	D.P.	VA matrix (NCH x NCH) *Include only if COVAR = T
5*					*Include only if HOPLT = T.
	1	13A4	ICMMNT	A	Comment card for user identification of human operator + plant dynamics input section
	2	I5	NTF3		No. of transfer functions desired (may be 0)

Note: All Group and card numbers marked with * are conditional input.

Table 2.2 (Continued)

GROUP NO.	CARD NO.	FORMAT	VARIABLE NAME	TYPE	DESCRIPTION
	3*	6I5	IRWCL3(1-6)	I	See Group 2, card 3 *Include NTF3 sets of cards 3-3B
	3A*	(I5,7E10.5)	NCOEFN	I	See Group 2, card 3A
			COEFN3	D.P.	See Group 2, card 3A
	3B*	(I5,7E10.5)	NCOEFD	I	See Group 2, card 3B
			COEFD3	D.P.	See Group 2, card 3B
6					*Include only if CLOSED = T
	1	13A4	ICMMNT	A	Comment card for user identification of closed loop input section.
	2	7I5	NMC	I	No. of measurements ($\leq$35)
			NTFC	I	No. of transfer functions desired (may be 0)
			JEIGFC	I	Eigenvalue flag See Group 2, card 2
			IDNTHC	I	= 0, HC matrix will be read in = 1, HC matrix set to identify
	3*	6I5	IRWCLC(1-6)	I	See Group 2, card 3 *Include NTFC sets of card 3-3B
	3A*	(I5,7E10.5)	NCOEFN	I	See Group 2, card 3A
			COEFNC	D.P.	See Group 2, card 3A
	3B*	(I5,7E10.5)	NCOEFD	I	See Group 2, card 3B
			COEFDC	D.P.	See Group 2, card 3B
	4	IFMT	CC	D.P.	Matrix for augmenting GC (NSA x NCH)
	5	IFMT	BC	D.P.	Matrix for augmenting GC (NCH x NCH)

Note: All Group and card numbers marked with * are conditional input.

Table 2.2 (Concluded)

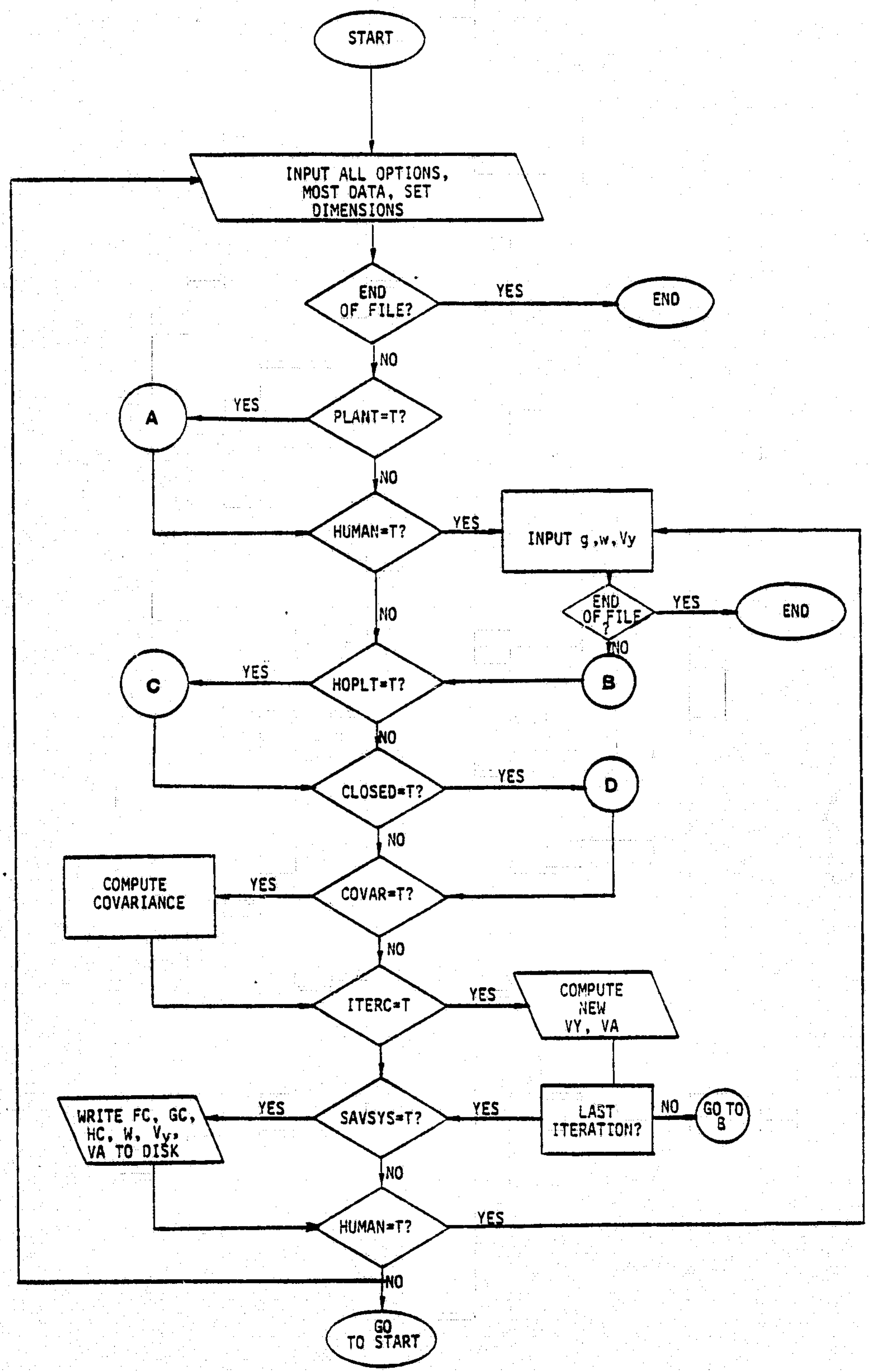
GROUP NO.	CARD NO.	FORMAT	VARIABLE NAME	TYPE	DESCRIPTION
	6*	IFMT	HCØRG	D.P.	H matrix for closed loop (NMC x NSC, NSC = NSP + NSH + NSA). *Include only if IDNTHC = 0
7*					*Include only if HUMAN = T
	1	IFMT	GSML	D.P.	g matrix (NCH x NCH)
	2	IFMT	WBIG	D.P.	W matrix (NOH x NOH)
	3	IFMT	VY	D.P.	Vy matrix (NMH x NMH)

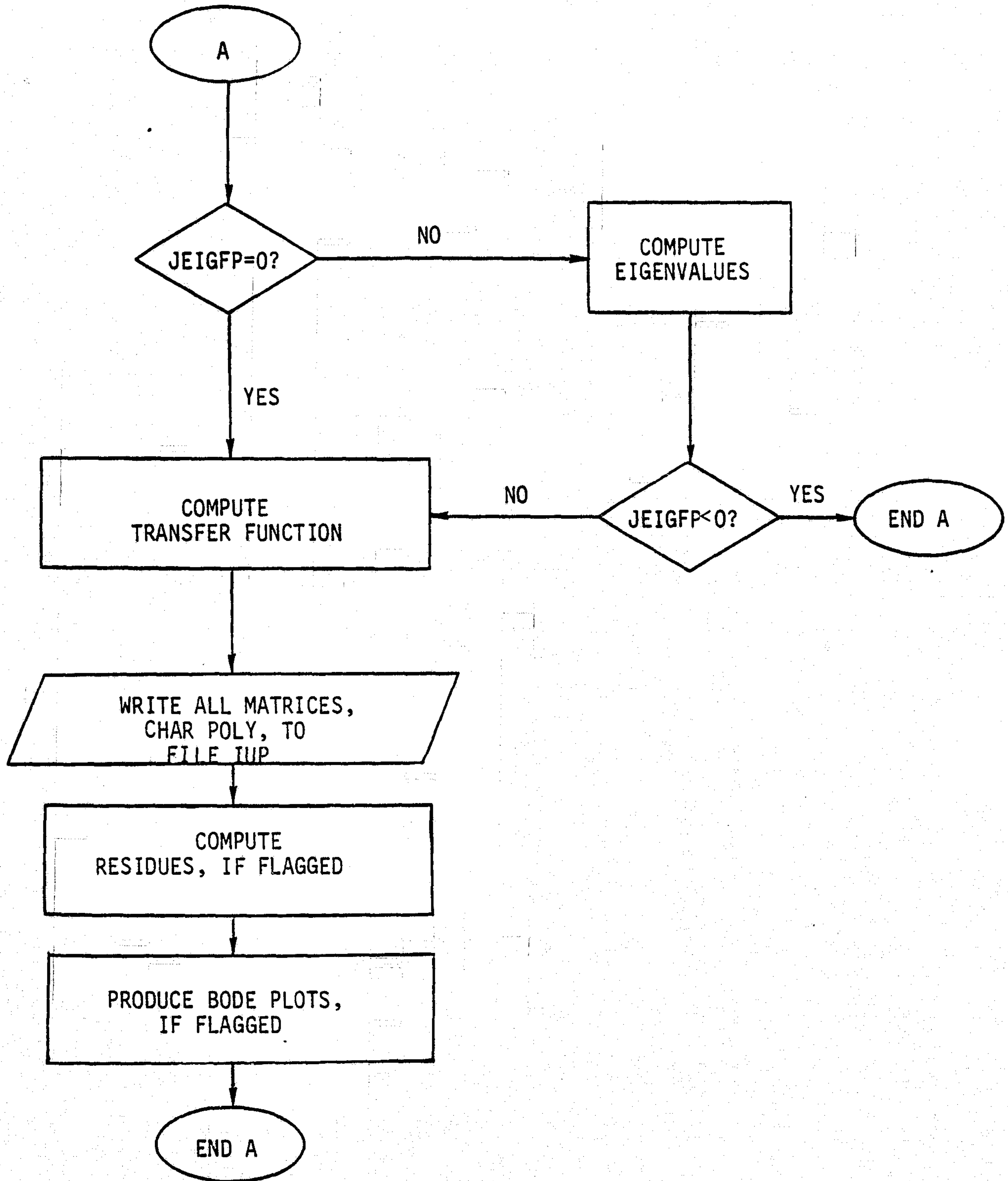
Note: All Group and card numbers marked with * are conditional input.

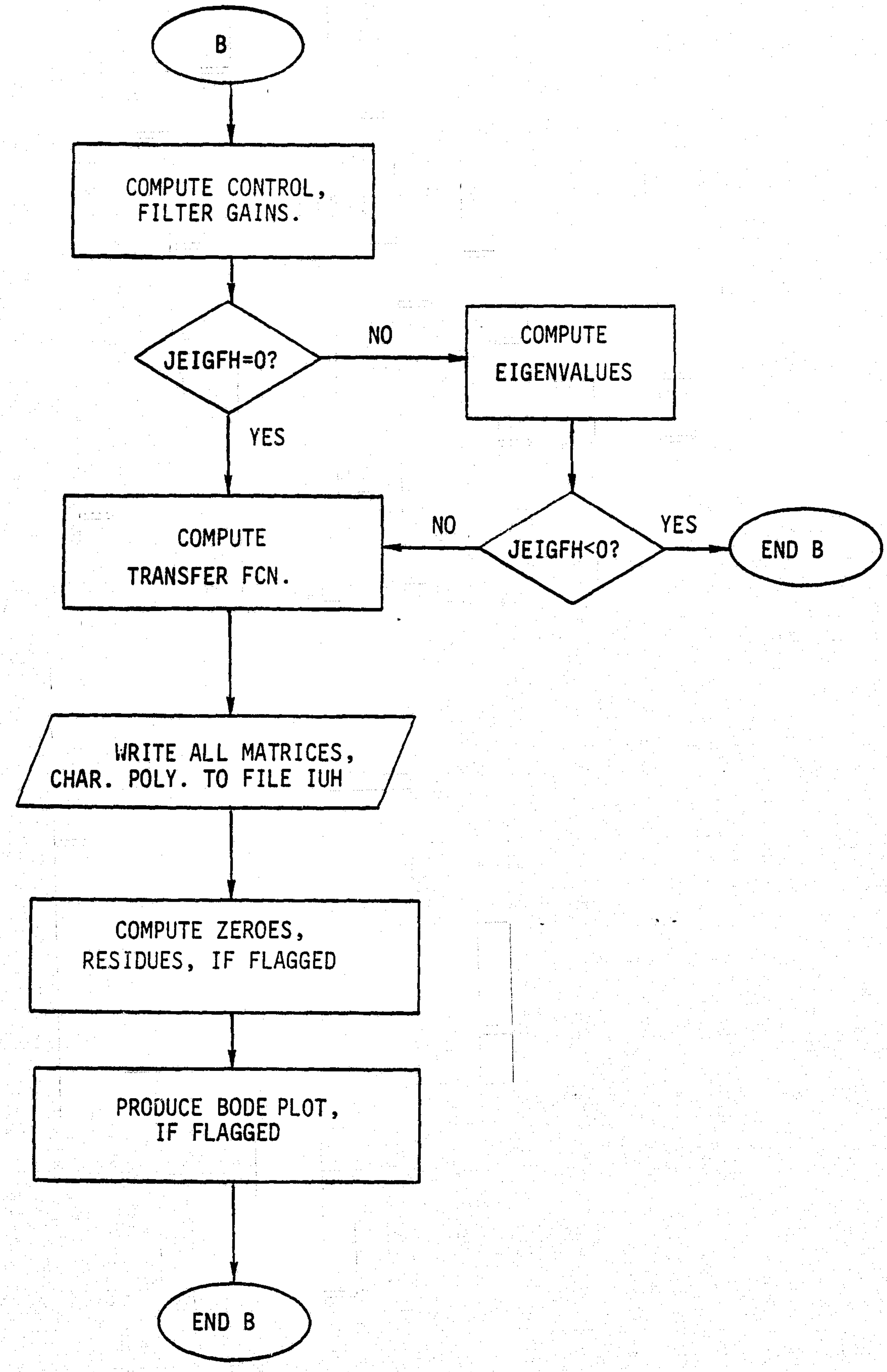
A = Alphanumeric
I = Integer
L = Logical
D.P. = Double Precision

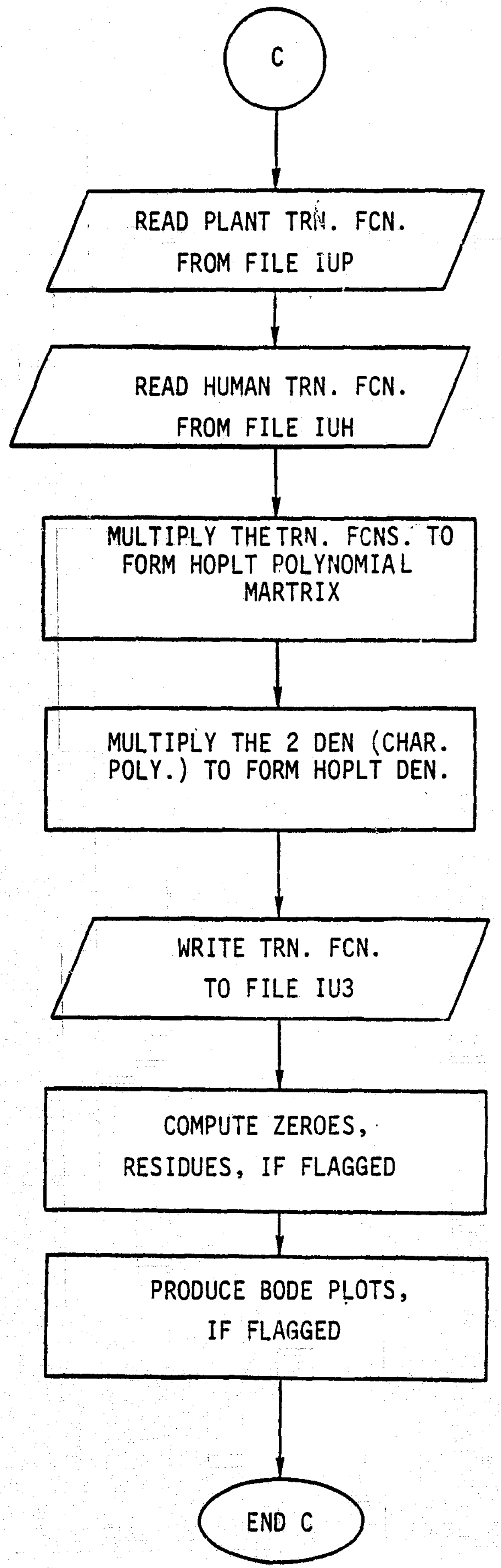
2.4 PROGRAM FLOWCHART

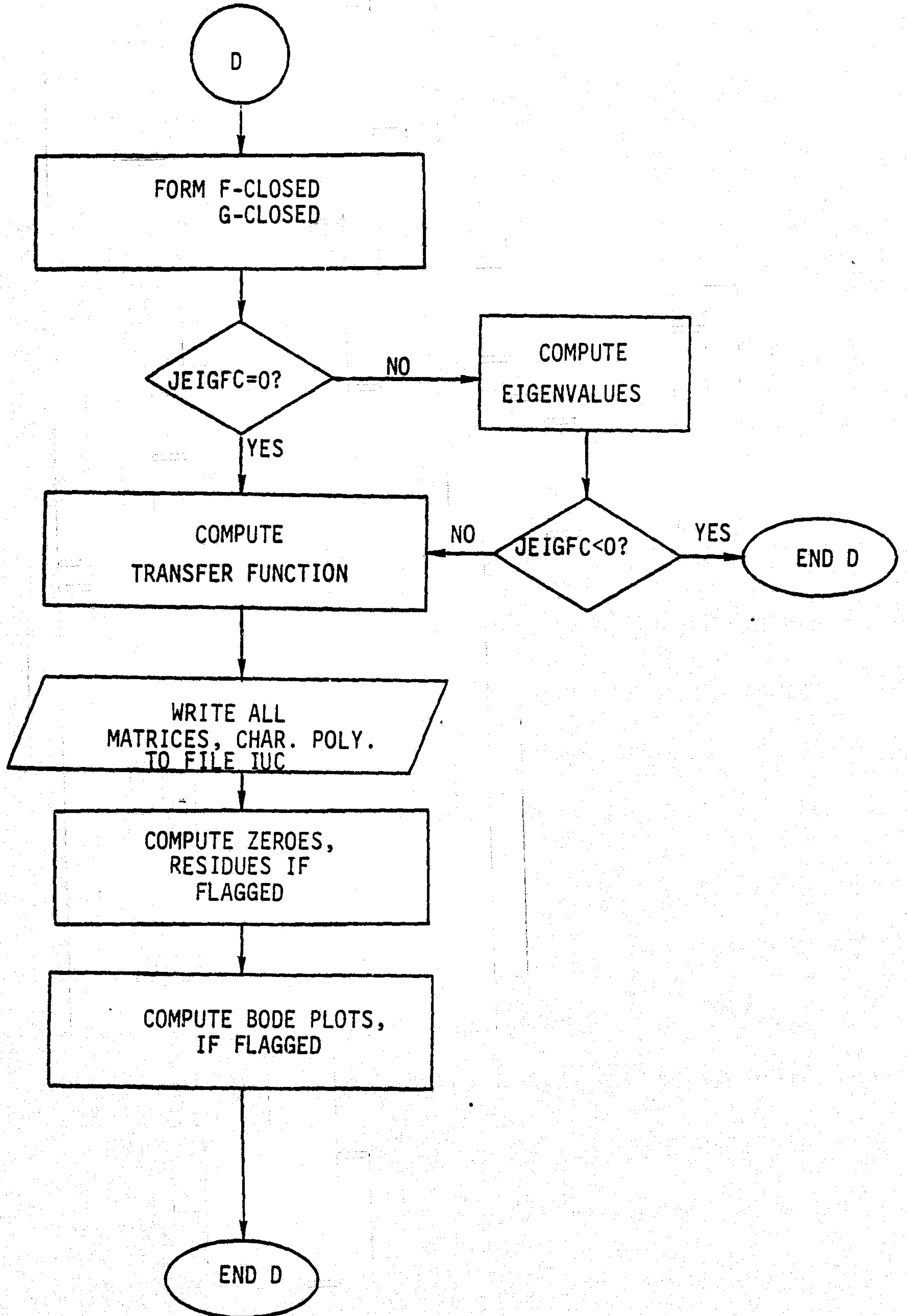
The following pages contain the flowchart for the transfer function program.


START
INPUT ALL OPTIONS, MOST DATA, SET DIMENSIONS
END OF FILE?
YES
END
NO
PLANT=T?
A
HUMAN=T?
INPUT g, w, Vy
END OF FILE ?
END
B
HOPLT=T?
C
CLOSED=T?
D
COVAR=T?
COMPUTE COVARIANCE
ITERC=T
COMPUTE NEW VY, VA
SAVSYS=T?
LAST ITERATION?
GO TO B
WRITE FC, GC, HC, W, Vy, VA TO DISK
HUMAN=T?
GO TO START


A
JEIGFP=0?
NO
COMPUTE EIGENVALUES
YES
COMPUTE TRANSFER FUNCTION
NO
JEIGFP<0?
YES
END A
WRITE ALL MATRICES, CHAR POLY, TO FILE IUP
COMPUTE RESIDUES, IF FLAGGED
PRODUCE BODE PLOTS, IF FLAGGED
END A


B
COMPUTE CONTROL, FILTER GAINS.
JEIGFH=0?
NO
COMPUTE EIGENVALUES
YES
COMPUTE TRANSFER FCN.
NO
JEIGFH<0?
YES
END B
WRITE ALL MATRICES, CHAR. POLY. TO FILE IUH
COMPUTE ZEROES, RESIDUES, IF FLAGGED
PRODUCE BODE PLOT, IF FLAGGED
END B


C
READ PLANT TRN. FCN.
FROM FILE IUP
READ HUMAN TRN. FCN.
FROM FILE IUH
MULTIPLY THE TRN. FCNS. TO FORM HOPLT POLYNOMIAL MARTRIX
MULTIPLY THE 2 DEN (CHAR. POLY.) TO FORM HOPLT DEN.
WRITE TRN. FCN.
TO FILE IU3
COMPUTE ZEROES,
RESIDUES, IF FLAGGED
PRODUCE BODE PLOTS,
IF FLAGGED
END C


D
FORM F-CLOSED
G-CLOSED
JEIGFC=0?
NO
COMPUTE
EIGENVALUES
YES
COMPUTE
TRANSFER FUNCTION
NO
JEIGFC<0?
YES
END D
WRITE ALL
MATRICES, CHAR. POLY.
TO FILE IUC
COMPUTE ZEROES,
RESIDUES IF
FLAGGED
COMPUTE BODE PLOTS,
IF FLAGGED
END D

2.5 TSS/360 OPERATION

The following control cards are required to execute the TF program on the NASA ARC TSS/360 system.

(1) LØGØN...

(2) DDEF FT10F001, VS, SCRATCH, DISP = NEW, RET = T

(3) DDEF FT11F001, VS, PLANT, DISP = NEW, RET = T

(4) DDEF FT12F001, VS, HUMAN, DISP = NEW, RET = T

(5) DDEF FT13F001, VS, HØPLT, DISP = NEW, RET = T

(6) DDEF FT14F001, VS, CLØSED, DISP = NEW, RET = T

(7) DDEF FT21F001, VS, TIME.F1, DISP = NEW

(8) JBLB NEWTFLIB

(9) LØAD BLØCK$$

(10) CALL MAIN$$

(11) {data

(12) LØGØFF

The previous transfer function described in the HOCOM user's manual (see Section I) is executed identically by the above sequence with the following exceptions:

(a) eliminate card #8

(b) replace card #9 by LOAD BLKDT1

(c) additional input data required in #11

The description of the above execution deck follows:

(1) Usual LØGØN card.

(2) Scratch file used by transfer function. It is always needed and should always be assigned temporary (RET = T parameter).

(3)-(6) Files needed depending on options chosen. I.e., if closed loop dynamics is not desired, card (6) can be eliminated. The RET = T parameter makes these files temporary.

(7) If the closed loop system is to be saved for use by the time varying program, this file must be assigned.

(8) Library of routines.

(9) Load in block data.

(10) Execute program.

(11) Input cards.

(12) Usual LØGØFF.

2.6 SAMPLE EXECUTION

The following represents a sample execution (with description and output listing).

```
+F FOR ID AND TV
 &OPTION
  PLANT=T,
  HUMAN=T,
  CLOSED=T,
  SAVSYS=T,
  COVAR=T,
 &END
INPUT FOR PLANT DYNAMICS
     2     1     1     1  0     0     0     0
-2.0        0.0
1.0             1.0E-05
0.0
1.0
0.0         1.0
0.0
1.0
0.0
INPUT FOR HUMAN DYNAMICS
     2     1     1     1     1     0     0
     1     1     3    -1     1     0
     1   1.0
     1   1.0
-2.0        0.0
1.0             1.0E-05
0.0
1.0
0.0         1.0
0.0
1.0
0.0
0.0         0.0
0.0         1.0
AUGMENTED STATES INPUT
     2
0.0         1.0
-266.67     -26.667
0.0
44.444
18.         1.8
-2.0
0.1048
CLOSED LOOP INPUT
     6     1     0     1
     1     1     3    -1     1     0
     1   1.0
     1   1.0
0.0
0.0
1.0
ADDITIONAL HUMAN INPUT
.05
17.
.00123
```

```
*** INPUT ***
ITITLE = SAMPLE RUN --- TRANSFER FUNCTION
THE OPTIONS HAVE BEEN SET AS FOLLOWS
  PLANT DYNAMICS =T
  HUMAN DYNAMICS = T
  CLOSED LOOP RESPONSE = T
  HUMAN + PLANT DYNAMICS = F
  MINIMAL OUTPUT FLAG = F
  BACKWARD LEVERRIER FOR NO. STATES LESS THAN 5 = F
ALL MATRICES WILL BE READ WITH FORMAT (8(E10.4) )
FROM UNIT  5
COVARIANCE FLAG = T
DISK READ FOR F AND G (PLANT) = F
WRITE FC,GC,HC,WBIG,VY,VA TO DISK FLAG = T
F AND G (PLANT) ON DISK USE FORMAT (4(E20.13) )
UNIT FOR FC,GC,HC,WBIG,VY,VA = 21
SWITCH TO SET FC MATRIX TO CONTROL ONLY = F
FLAG TO ITERATE ON VY AND/OR VA = F
***PLANT DYNAMICS INPUT
NO. OF STATES =   2
NO. OF CONTROLS =    1
NO. OF MEASUREMENTS =    1
NO. OF NOISE SOURCES =    1
EIGENVALUE FLAG FOR F =  0
SINGULARITY FLAG FOR F =  0
ENTRY FLAG =  0
  0 TRANSFER FUNCTIONS WILL BE DONE.
SAMPLE RUN --- TRANSFER FUNCTION


F, PLANT DYNAMICS
                1            2
     1    -2.0000D 00      0.0000
     2     1.0000D 00  1.0000D-05


H, PLANT DYNAMICS
                1            2
     1         0.0000  1.0000D 00


D, PLANT DYNAMICS
                1            2
     1         0.0000      0.0000


GAMMA, PLT. DYN. INPUT
                1
     1     1.0000D 00
     2         0.0000


G, AS INPUT
                1
     1         0.0000
     2     1.0000D 00
SAMPLE RUN --- TRANSFER FUNCTION


 AUGMENTED G AND GAMMA MATRIX
```

```
      1          0.0000   1.0000D 00
      2       1.0000D 00      0.0000

IN MATRIX OF TRANSFER FUNCTIONS , THE FIRST    1 COLUMNS = Y OVER U , THE SECOND    1 COLUMNS = Y OVER W
***HUMAN DYNAMICS INPUT
NO. OF STATES =    2
NO. OF CONTROLS =    1
NO. OF MEASUREMENTS =    1
NO. OF NOISE SOURCES =    1
EIGENVALUE FLAG FOR F =   0
SINGULARITY FLAG FOR F =   0

  1 TRANSFER FUNCTIONS WILL BE DONE.
ROW COL.  OPTION  LOG MIN  LOG MAX  PLOT FLAG
 1   1      3       -1        1         0       NUM. COEF..10000E 01
                                                DEN. COEF..10000E 01
SAMPLE RUN --- TRANSFER FUNCTION



F, HUM. DYN. INPUT
                 1            2
      1    -2.0000D 00      0.0000
      2     1.0000D 00   1.0000D-05


G, HUM. DYN. INPUT
                 1
      1          0.0000
      2       1.0000D 00


H, HUM. DYN. INPUT
                 1            2
      1          0.0000   1.0000D 00


D, HUM. DYN. INPUT
                 1
      1          0.0000


GAMMA, HUM. DYN. INPUT
                 1
      1       1.0000D 00
      2          0.0000


Q, HUM. DYN. INPUT
                 1            2
      1          0.0000      0.0000
      2          0.0000   1.0000D 00
SAMPLE RUN --- TRANSFER FUNCTION

***AUGMENTED STATES INPUT.
NO. OF STATES =    2
NO. OF CONTROLS =    1
NO. OF MEASUREMENTS =    1



FA
                 1            2
      1          0.0000   1.0000D 00
```

41

ORIGINAL PAGE IS
OF POOR QUALITY

```
GA
                1
     1      0.0000
     2   4.4444D 01


HA
                1            2
     1   1.8000D 01   1.8000D 00


DA
                1
     1  -2.0000D 00


VA
                1
     1   1.0480D-01
SAMPLE RUN --- TRANSFER FUNCTION

*** CLOSED LOOP RESPONSE INPUT
NO. OF STATES =    6
NO. OF CONTROLS =    3
NO. OF MEASUREMENTS =    7
SET HC = IDENTITY MATRIX FLAG =    1
EIGENVALUE FLAG =   0
  1 TRANSFER FUNCTIONS WILL BE DONE.
ROW COL.   OPTION   LOG MIN   LOG MAX   PLOT FLAG
  1   1      3        -1         1          0        NUM. COEF..10000E 01
                                                     DEN. COEF..10000E 01


CC MATRIX
                1
     1      0.0000
     2      0.0000


BC
                1
     1   1.0000D 00


UNAUGMENTED CLOSED LOOP HC
                1           2           3           4           5           6
     1   1.0000D 00      0.0000      0.0000      0.0000      0.0000      0.0000
     2      0.0000   1.0000D 00      0.0000      0.0000      0.0000      0.0000
     3      0.0000      0.0000   1.0000D 00      0.0000      0.0000      0.0000
     4      0.0000      0.0000      0.0000   1.0000D 00      0.0000      0.0000
     5      0.0000      0.0000      0.0000      0.0000   1.0000D 00      0.0000
     6      0.0000      0.0000      0.0000      0.0000      0.0000   1.0000D 00
*** END OF INPUT.
*** PLANT DYNAMICS OUTPUT
0A TEST OF THE NUMERIC ACCURACY OF THE [illegible] LEVERRIER METHOD IS THAT THE N-TH B MATRIX
 SHOULD BE IDENTICALLY ZERO ACCORDING TO THE THEORY.
 IT WAS CALCULATED FOR THIS PROBLEM AS---


N-TH B MATRI
                1            2
     [illegible]16D       [illegible]000[illegible]
     2   -6.5512D-17 -6.5513D-17
LEVERRIER ALGORITHM WAS COMPUTED IN THE FORWARD DIRECTION ONLY.
```

```
 NUMERATOR MATRIX POLYNOMIAL OF THE TRANSFER FUNCTION
+--------------------------------------------------------+
NUMERATOR MATRIX COEFFICIENT FOR S** 2

                  1           2
     1       0.0000      0.0000
NUMERATOR MATRIX COEFFICIENT FOR S** 1

                  1           2
     1   1.0000D 00      0.0000
NUMERATOR MATRIX COEFFICIENT FOR S** 0

                  1           2
     1   2.0000D 00  1.0000D 00
 MATRIX OF DC GAINS

                  1           2
     1  -1.0000D 05 -5.0000D 04
SAMPLE RUN --- TRANSFER FUNCTION

+------------------------------------+
   POLES OF THE TRANSFER FUNCTION
+------------------------------------+
COEFFICIENTS OF THE CHARACTERISTIC EQUATION IN DESCENDING POWERS OF S =

     1.00000D 00      1.99999D 00     -2.00000D-05
POLES OF THE TRANSFER FUNCTION AS COMPUTED FROM THE ABOVE POLYNOMIAL =

      0   (-2.00000D 00) + J(      0.00000)
  MAG, ANGLE, ZETA =      0.2000D 01    0.1800D 03    0.1000D 01
      1   ( 1.00000D-05) + J(      0.00000)
  MAG, ANGLE, ZETA =      0.1000D-04        0.0000   -0.1000D 01
SAMPLE RUN --- TRANSFER FUNCTION

**** CASE NO.    1

GSNL
                  1
     1   5.0000D-02

WBIG
                  1
     1   1.7000D 01

VY
                  1
     1   1.2300D-03
SAMPLE RUN --- TRANSFER FUNCTION

*** CONTROL AND FILTER GAINS.
```

CONTROL GAIN

	1	2
1	6.9098D-01	4.4721D 00

FILTER GAIN

	1
1	9.0636D 01
2	1.3464D 01

SAMPLE RUN --- TRANSFER FUNCTION

*** HUMAN OPERATOR DYNAMICS OUTPUT.
MATRICES FOR HUMAN DYNAMICS.

FHH

	1	2	3	4
1	-2.0000D 00	-9.0636D 01	0.0000	0.0000
2	2.3820D 00	-4.5194D 00	1.8000D 01	1.8000D 00
3	0.0000	0.0000	0.0000	1.0000D 00
4	-3.0710D 01	-1.9876D 02	-2.6667D 02	-2.6667D 01

GHH

	1
1	9.0636D 01
2	1.3464D 01
3	0.0000
4	0.0000

HHH

	1	2	3	4
1	1.3820D 00	8.9443D 00	1.8000D 01	1.8000D 00

A TEST OF THE NUMERIC ACCURACY OF THE ONE-WAY LEVERRIER METHOD IS THAT THE N-TH B MATRIX
SHOULD BE IDENTICALLY ZERO ACCORDING TO THE THEORY.
IT WAS CALCULATED FOR THIS PROBLEM AS---

N-TH B MATRI

	1	2	3	4
1	1.3642D-11	-2.9104D-11	1.7521D-10	-9.6634D-13
2	-5.4570D-13	-3.6380D-11	3.4561D-11	2.0918D-12
3	2.7285D-12	-3.3651D-11	3.6380D-11	9.0949D-13
4	-9.0949D-16	5.4813D-10	1.4552D-11	-2.4556D-11

LEVERRIER ALGORITHM WAS COMPUTED IN THE FORWARD DIRECTION ONLY.
SAMPLE RUN --- TRANSFER FUNCTION

+--+
NUMERATOR MATRIX POLYNOMIAL OF THE TRANSFER FUNCTION
+--+

NUMERATOR MATRIX COEFFICIENT FOR S** 4

	1
1	0.0000

NUMERATOR MATRIX COEFFICIENT FOR S** 3

	1
1	2.4568D 02

NUMERATOR MATRIX COEFFICIENT FOR S** 2

```
                  1
    1    -2.[illegible]D 0[illegible]
NUMERATOR MATRIX COEFFICIENT FOR S** 1


                  1
    1    -4.6775D 04
NUMERATOR MATRIX COEFFICIENT FOR S** 0


                  1
    1    -1.4019D 05
 MATRIX OF DC GAINS


                  1
    1    -8.2295D 00
SAMPLE RUN --- TRANSFER FUNCTION

+-----------------------------------+
'  POLES OF THE TRANSFER FUNCTION   '
+-----------------------------------+
COEFFICIENTS OF THE CHARACTERISTIC EQUATION IN DESCENDING POWERS OF S =

     1.00000D 00      3.31864D 01      1.02322D 03      7.01979D 03      1.70357D 04
POLES OF THE TRANSFER FUNCTION AS COMPUTED FROM THE ABOVE POLYNOMIAL =

      0   (-1.25314D 01) + J( 2.53228D 01)
  MAG, ANGLE, ZETA =      0.2825D 02     0.1163D 03     0.4435D 00
      1   (-1.25314D 01) + J(-2.53228D 01)
  MAG, ANGLE, ZETA =      0.2825D 02    -0.1163D 03     0.4435D 00
      2   (-4.06181D 00) + J( 2.20050D 00)
  MAG, ANGLE, ZETA =      0.4620D 01     0.1516D 03     0.8793D 00
      3   (-4.06181D 00) + J(-2.20050D 00)
  MAG, ANGLE, ZETA =      0.4620D 01    -0.1516D 03     0.8793D 00
SAMPLE RUN --- TRANSFER FUNCTION

+-------------------------------------+
'  ZEROES OF THE TRANSFER FUNCTION    '
+-------------------------------------+
ZEROES OF THE TRANSFER FUNCTION BETWEEN Y( 1) AND U( 1) =

      1   ( 1.99992D 01) + J(      0.00000)
  MAG, ANGLE, ZETA =      0.2000D 02         0.0000    -0.1000D 01
      2   (-6.66657D 00) + J(      0.00000)
  MAG, ANGLE, ZETA =      0.6667D 01     0.1800D 03     0.1000D 01
      3   (-4.28005D 00) + J(      0.00000)
  MAG, ANGLE, ZETA =      0.4280D 01     0.1800D 03     0.1000D 01
COEFFICIENTS OF THE NUMERATOR POLYNOMIAL IN ASCENDING POWERS OF S =

   -1.40195D 05     -4.67750D 04     -2.22403D 03      2.45680D 02
RESIDUES OF THE TRANSFER FUNCTION=

           POLE LOCATION                       RESIDUE AT THE POLE          COMBINED RESIDUES=MAG*COS(WD*T+PHI)
                                                                                MAGNITUDE            PHI(DEG)

    (-1.25314D 01) + J( 2.53228D 01)     ( 1.35698D 02) + J( 1.40361D 02)        3.90463D 02         -45.96773
    (-4.06181D 00) + J( 2.20050D 00)     (-1.28564D 01) + J(-6.35190D 00)        2.86835D 01         153.71120
FUNCTION TO BE PLOTTED IS Y( 1) TO U( 1)
TRANSFER FUNCTION TO BE PLOTTED

NUMERATOR COEFFICIENTS IN DESCENDING ORDER
```

[illegible]

DENOMINATOR COEFFICIENTS IN DESCENDING ORDER

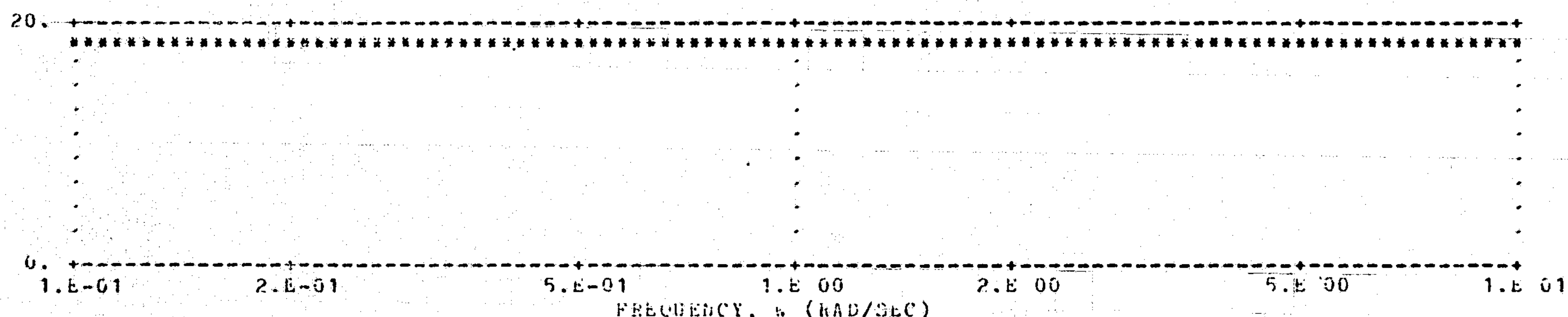
0.1000000E 01 0.3318643E 02 0.1023222E 04 0.7019792E 04 0.170[illegible]75E 05

BODE PLOT VALUES

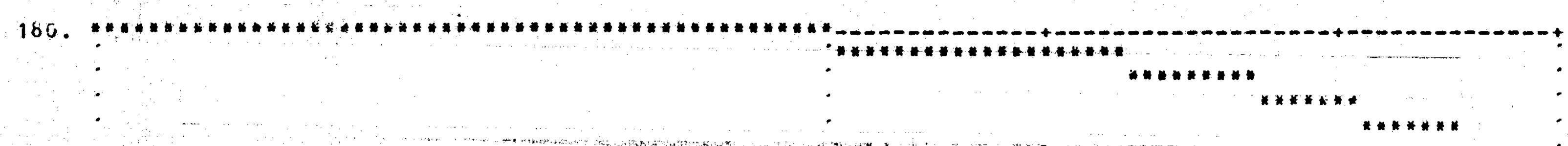
FREQUENCY	MAGNITUDE	PHASE
0.10000E 00	0.18309E 02	0.17[illegible]55E 03
0.10471E 00	0.18309E 02	0.17953E 03
0.10965E 00	0.18309E 02	0.17951E 03
0.11482E 00	0.18309E 02	0.17948E 03
0.12023E 00	0.18309E 02	0.17946E 03
0.12589E 00	0.18309E 02	0.17943E 03
0.13183E 00	0.18310E 02	0.17941E 03
0.13804E 00	0.18310E 02	0.17938E 03
0.14454E 00	0.18310E 02	0.17935E 03
0.15136E 00	0.18310E 02	0.17932E 03
0.15849E 00	0.18311E 02	0.17929E 03
0.16596E 00	0.18311E 02	0.17925E 03
0.17378E 00	0.18311E 02	0.17922E 03
0.18197E 00	0.18312E 02	0.17918E 03
0.19055E 00	0.18312E 02	0.17914E 03
0.19953E 00	0.18313E 02	0.17910E 03
0.20893E 00	0.18313E 02	0.17906E 03
0.21878E 00	0.18314E 02	0.17901E 03
0.22909E 00	0.18314E 02	0.17897E 03
0.23988E 00	0.18315E 02	0.17892E 03
0.25119E 00	0.18316E 02	0.17887E 03
0.26303E 00	0.18316E 02	0.17881E 03
0.27542E 00	0.18317E 02	0.17876E 03
0.28840E 00	0.18318E 02	0.17870E 03
0.30200E 00	0.18319E 02	0.17864E 03
0.31623E 00	0.18320E 02	0.17857E 03
0.33113E 00	0.18322E 02	0.17850E 03
0.34674E 00	0.18323E 02	0.17843E 03
0.36308E 00	0.18324E 02	0.17836E 03
0.38019E 00	0.18326E 02	0.17828E 03
0.39811E 00	0.18328E 02	0.17819E 03
0.41687E 00	0.18330E 02	0.17811E 03
0.43652E 00	0.18332E 02	0.17802E 03
0.45709E 00	0.18334E 02	0.17792E 03
0.47863E 00	0.18337E 02	0.17782E 03
0.50119E 00	0.18339E 02	0.17771E 03
0.52481E 00	0.18342E 02	0.17760E 03
0.54954E 00	0.18346E 02	0.17749E 03
0.57544E 00	0.18349E 02	0.17736E 03
0.60256E 00	0.18353E 02	0.17723E 03
0.63096E 00	0.18357E 02	0.17710E 03
0.66069E 00	0.18362E 02	0.17695E 03
0.69183E 00	0.18367E 02	0.17680E 03
0.72444E 00	0.18373E 02	0.17664E 03
0.75858E 00	0.18378E 02	0.17648E 03
0.79433E 00	0.18385E 02	0.17630E 03
0.83176E 00	0.18392E 02	0.17611E 03
0.87096E 00	0.18400E 02	0.17591E 03
0.91201E 00	0.18408E 02	0.17570E 03
0.95499E 00	0.18417E 02	0.17548E 03
0.10000E 01	0.18426E 02	0.17525E 03
0.10471E 01	0.18437E 02	0.17500E 03
0.10965E 01	0.18448E 02	0.17474E 03
0.11482E 01	0.18460E 02	0.17446E 03
0.12023E 01	0.18473E 02	0.17416E 03
0.12589E 01	0.18487E 02	0.17385E 03
0.13[illegible] 01	[illegible]850[illegible] 02	0.[illegible]51E [illegible]
0.13804E 01	0.18517E 02	0.17315E 03
0.14454E 01	0.18534E 02	0.17277E 03

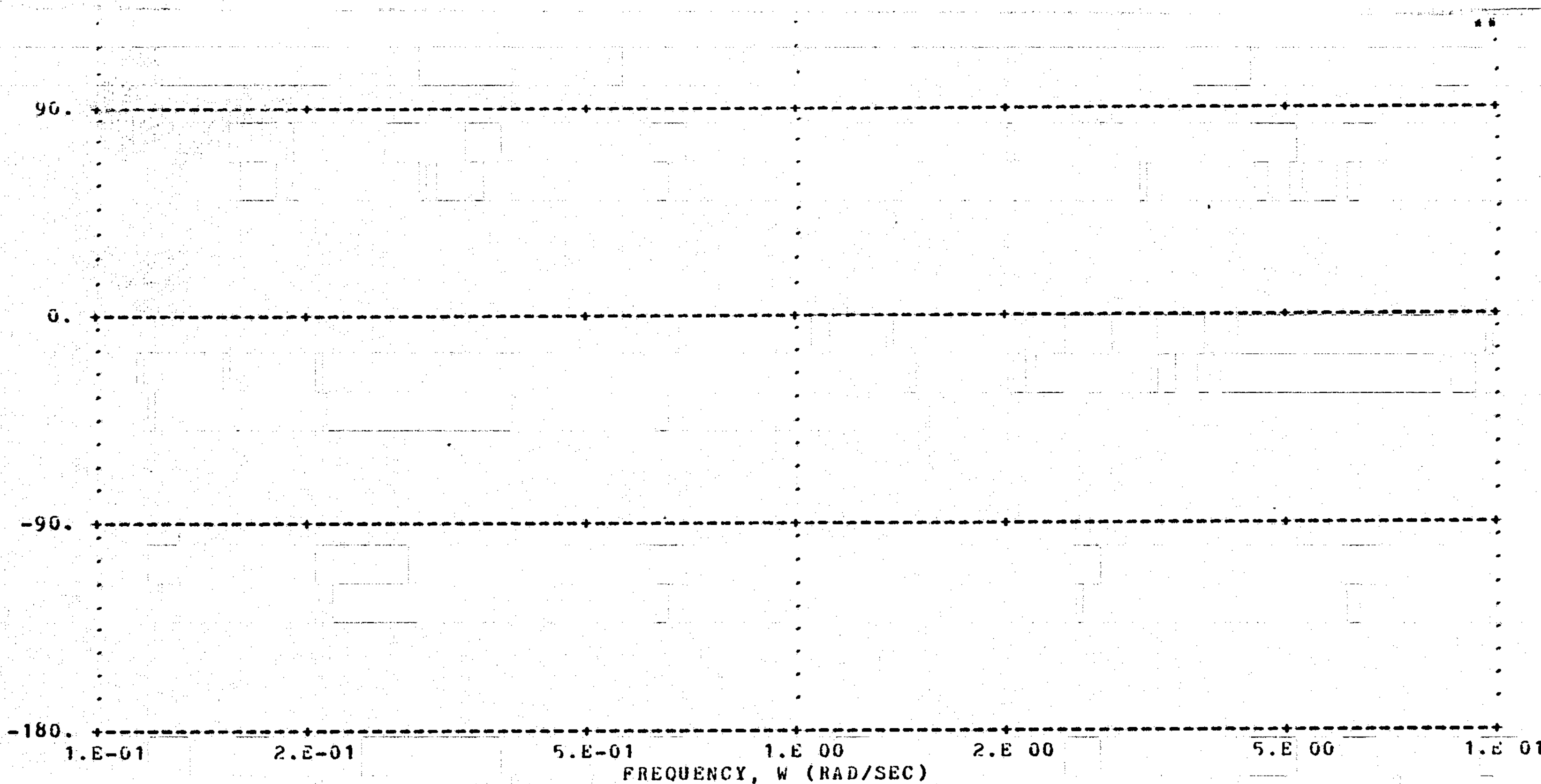
0.15649E 01	0.18570E 02	0.17194E 03
0.165[illegible] 01	0.18589E 02	0.17146E 03
0.175[illegible] 01	0.[illegible]609E 02	0.[illegible]99E
0.18197E 01	0.18630E 02	0.17047E 03
0.19055E 01	0.18652E 02	0.16991E 03
0.19953E 01	0.18674E 02	0.16932E 03
0.20893E 01	0.18697E 02	0.16869E 03
0.21878E 01	0.18719E 02	0.16801E 03
0.22909E 01	0.18742E 02	0.16730E 03
0.23988E 01	0.18764E 02	0.16653E 03
0.25119E 01	0.18785E 02	0.16573E 03
0.26303E 01	0.18806E 02	0.16487E 03
0.27542E 01	0.18825E 02	0.16396E 03
0.28840E 01	0.18842E 02	0.16300E 03
0.30199E 01	0.18857E 02	0.16199E 03
0.31623E 01	0.18869E 02	0.16093E 03
0.33113E 01	0.18879E 02	0.15982E 03
0.34674E 01	0.18884E 02	0.15865E 03
0.36308E 01	0.18886E 02	0.15744E 03
0.38019E 01	0.18884E 02	0.15618E 03
0.39811E 01	0.18878E 02	0.15488E 03
0.41687E 01	0.18867E 02	0.15354E 03
0.43652E 01	0.18852E 02	0.15216E 03
0.45709E 01	0.18833E 02	0.15075E 03
0.47863E 01	0.18810E 02	0.14931E 03
0.50119E 01	0.18784E 02	0.14784E 03
0.52481E 01	0.18755E 02	0.14635E 03
0.54954E 01	0.18724E 02	0.14483E 03
0.57544E 01	0.18693E 02	0.14330E 03
0.60256E 01	0.18661E 02	0.14174E 03
0.63096E 01	0.18631E 02	0.14017E 03
0.66069E 01	0.18603E 02	0.13858E 03
0.69183E 01	0.18578E 02	0.13696E 03
0.72443E 01	0.18558E 02	0.13532E 03
0.75858E 01	0.18544E 02	0.13366E 03
0.79433E 01	0.18537E 02	0.13196E 03
0.83176E 01	0.18539E 02	0.13022E 03
0.87096E 01	0.18550E 02	0.12844E 03
0.91201E 01	0.18571E 02	0.12660E 03
0.95499E 01	0.18604E 02	0.12471E 03
0.10000E 02	0.18650E 02	0.12275E 03

BODE PLOT...MAGNITUDE OF RESPONSE



BODE PLOT...PHASE OF THE RESPONSE





*** OUTPUT FROM CLOSED LOOP RESPONSE.

FC, CLOSED LOOP RESPONSE

	1	2	3	4	5	6
1	-2.0000D 00	0.0000	0.0000	0.0000	0.0000	0.0000
2	1.0000D 00	1.0000D-05	1.3820D 00	8.9443D 00	1.8000D 01	1.8000D 00
3	0.0000	9.0636D 01	-2.0000D 00	-9.0636D 01	0.0000	0.0000
4	0.0000	1.3464D 01	2.3820D 00	-4.5194D 00	1.8000D 01	1.8000D 00
5	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000D 00
6	0.0000	0.0000	-3.0710D 01	-1.9876D 02	-2.6667D 02	-2.6667D 01

HC CLOSED LOOP RESPONSEG

	1	2	3	4	5	6
1	0.0000	0.0000	1.3820D 00	8.9443D 00	1.8000D 01	1.8000D 00
2	1.0000D 00	0.0000	0.0000	0.0000	0.0000	0.0000
3	0.0000	1.0000D 00	0.0000	0.0000	0.0000	0.0000
4	0.0000	0.0000	1.0000D 00	0.0000	0.0000	0.0000
5	0.0000	0.0000	0.0000	1.0000D 00	0.0000	0.0000
6	0.0000	0.0000	0.0000	0.0000	1.0000D 00	0.0000
7	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000D 00

GC, CLOSED LOOP RESP

	1	2	3
1	1.0000D 00	0.0000	0.0000
2	0.0000	0.0000	1.0000D 00
3	0.0000	9.0636D 01	0.0000
[illegible]	0.00[illegible]	1.3[illegible] 01	[illegible]
5	0.[illegible]00	0.0000	0.0000
6	0.0000	0.0000	0.0000

LEVERRIER ALGORITHM WAS COMPUTED IN THE FORWARD AND BACKWARD DIRECTIONS.
SAMPLE RUN --- TRANSFER FUNCTION

NUMERATOR MATRIX POLYNOMIAL OF THE TRANSFER FUNCTION

NUMERATOR MATRIX COEFFICIENT FOR S** 6

	1	2	3
1	0.0000	0.0000	0.0000
2	0.0000	0.0000	0.0000
3	0.0000	0.0000	0.0000
4	0.0000	0.0000	0.0000
5	0.0000	0.0000	0.0000
6	0.0000	0.0000	0.0000
7	0.0000	0.0000	0.0000

NUMERATOR MATRIX COEFFICIENT FOR S** 5

	1	2	3
1	0.0000	2.4568D 02	8.9443D 00
2	1.0000D 00	0.0000	0.0000
3	0.0000	0.0000	1.0000D 00
4	0.0000	9.0636D 01	0.0000
5	0.0000	1.3464D 01	1.0000D 00
6	0.0000	0.0000	0.0000
7	0.0000	0.0000	0.0000

NUMERATOR MATRIX COEFFICIENT FOR S** 4

	1	2	3
1	0.0000	-1.7327D 03	3.6950D 01
2	3.3186D 01	0.0000	0.0000
3	1.0000D 00	2.4568D 02	4.4131D 01
4	0.0000	1.7676D 03	0.0000
5	0.0000	6.2878D 02	4.4131D 01
6	0.0000	0.0000	0.0000
7	0.0000	-5.4595D 03	-1.9876D 02

NUMERATOR MATRIX COEFFICIENT FOR S** 3
SAMPLE RUN --- TRANSFER FUNCTION

	1	2	3
1	2.4568D 02	-5.1223D 04	-1.9469D 03
2	7.7754D 02	0.0000	0.0000
3	3.3186D 01	-1.7327D 03	8.8087D 02
4	9.0636D 01	3.8191D 04	1.1369D-13
5	1.3464D 01	6.2592D 03	8.8087D 02
6	0.0000	-5.4595D 03	-1.9876D 02
7	0.0000	-3.4286D 04	-3.4711D 03

NUMERATOR MATRIX COEFFICIENT FOR S** 2

	1	2	3
1	-2.2240D 03	-2.3374D 05	-3.6430D 04
2	9.2438D 03	0.0000	0.0000
3	1.0232D 03	-5.1223D 04	8.8520D 03
4	1.6063D 03	1.7804D 05	-7.2760D-12

ORIGINAL PAGE IS OF POOR QUALITY

```
     6          0.0000 -3.4286D 04 -3.4711D 03
     7     -5.4595D 03 -4.6733D 04 -2.9514D 04
 NUMERATOR MATRIX COEFFICIENT FOR S** 0


                 1           2           3
     1     -1.4019D 05  2.6039D 00 -2.8039D 05
     2      1.4019D 05      0.0000      0.0000
     3      1.7036D 04 -2.8039D 05  6.2701D 04
     4      1.0808D 05 -2.1617D 00 -7.2760D-11
     5      1.4651D 04 -2.9301D-01  6.2701D 04
     6     -2.3367D 04  4.6734D-01 -4.6734D 04
     7          0.0000      0.0000      0.0000
  MATRIX OF DC GAINS


                 1           2           3
     1     -5.0000D-01  1.0000D-05 -1.0000D 00
     2      5.0000D-01      0.0000      0.0000
     3      6.0757D-02 -1.0000D 00  2.2362D-01
     4      3.8548D-01 -7.7095D-06 -2.5949D-16
     5      5.2251D-02 -1.0450D-06  2.2362D-01
     6     -8.3337D-02  1.6667D-06 -1.6667D-01
     7          0.0000      0.0000      0.0000
 SAMPLE RUN --- TRANSFER FUNCTION

 NUMERATOR MATRIX COEFFICIENT FOR S** 1


                 1           2           3
     1     -4.6775D 04 -2.8039D 05 -2.0512D 05
     2      6.3811D 04      0.0000      0.0000
     3      7.0198D 03 -2.3374D 05  4.5868D 04
     4      3.4978D 04  2.1617D 05 -4.3656D-11
     5      5.0554D 03  2.9301D 04  4.5868D 04
     6     -5.4595D 03 -4.6733D 04 -2.9514D 04
     7     -2.3367D 04  4.6734D-01 -4.6734D 04
 SAMPLE RUN --- TRANSFER FUNCTION

 +----------------------------------+
    POLES OF THE TRANSFER FUNCTION
 +----------------------------------+
 COEFFICIENTS OF THE CHARACTERISTIC EQUATION IN DESCENDING POWERS OF S =

     1.00000D 00     3.51864D 01     8.43915D 02     1.07989D 04     8.22983D 04     2.67816D 05     2.80389D 05
 POLES OF THE TRANSFER FUNCTION AS COMPUTED FROM THE ABOVE POLYNOMIAL =

      0   (-7.08451D 00) + J( 1.68931D 01)
   MAG, ANGLE, ZETA =      0.1832D 02     0.1128D 03     0.3867D 00
      1   (-7.08451D 00) + J(-1.68931D 01)
   MAG, ANGLE, ZETA =      0.1832D 02    -0.1128D 03     0.3867D 00
      2   (-7.73186D 00) + J( 7.60143D 00)
   MAG, ANGLE, ZETA =      0.1084D 02     0.1355D 03     0.7131D 00
      3   (-7.73186D 00) + J(-7.60143D 00)
   MAG, ANGLE, ZETA =      0.1084D 02    -0.1355D 03     0.7131D 00
      4   (-3.55369D 00) + J(     0.00000)
   MAG, ANGLE, ZETA =      0.3554D 01     0.1800D 03     0.1000D 01
      5   (-2.00000D 00) + J(     0.00000)
   MAG, ANGLE, ZETA =      0.2000D 01     0.1800D 03     0.1000D 01
 SAMPLE RUN --- TRANSFER FUNCTION

 +----------------------------------+
```

+------------------------------------+

ROOTS OF THE TRANSFER FUNCTION BETWEEN Y(1) AND U(1)

```
     1   ( 1.99992D 01) + J(      0.00000)
  MAG, ANGLE, ZETA =        0.2000D 02          0.0000    -0.1000D 01
     2   (-6.66657D 00) + J(      0.00000)
  MAG, ANGLE, ZETA =        0.6667D 01     0.1800D 03     0.1000D 01
     3   (-4.28005D 00) + J(      0.00000)
  MAG, ANGLE, ZETA =        0.4280D 01     0.1800D 03     0.1000D 01
COEFFICIENTS OF THE NUMERATOR POLYNOMIAL IN ASCENDING POWERS OF S =

   -1.40195D 05    -4.67750D 04    -2.22403D 03     2.45680D 02
RESIDUES OF THE TRANSFER FUNCTION=
```

POLE LOCATION	RESIDUE AT THE POLE	COMBINED RESIDUES=MAG*COS(WD*T+PHI) MAGNITUDE	PHI(DEG)
(-7.08451D 00) + J(1.68931D 01)	(-6.73486D-01) + J(-6.93353D-01)	1.93321D 00	134.16728
(-7.73166D 00) + J(7.60143D 00)	(1.14165D 00) + J(1.09210D 00)	3.15978D 00	-43.72915
-3.55369D 00	3.75779D-01		
-2.00000D 00	-1.31212D 00		

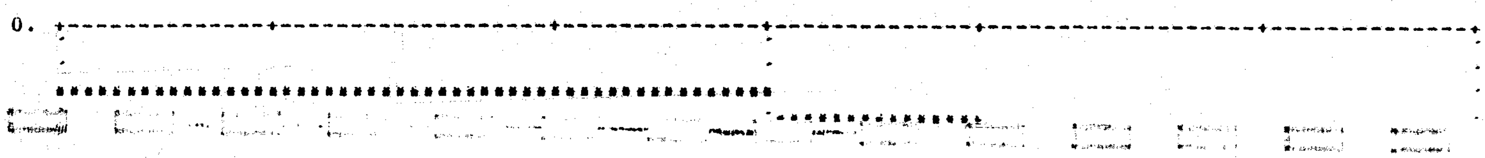
FUNCTION TO BE PLOTTED IS Y(1) TO U(1)
TRANSFER FUNCTION TO BE PLOTTED

NUMERATOR COEFFICIENTS IN DESCENDING ORDER
0.2456795E 03-0.2224027E 04-0.4677496E 05-0.1401949E 06
DENOMINATOR COEFFICIENTS IN DESCENDING ORDER
0.1000000E 01 0.3518642E 02 0.8439154E 03 0.1079889E 05 0.8229826E 05 0.2678160E 06 0.2803895E 06
BODE PLOT VALUES

FREQUENCY	MAGNITUDE	PHASE
0.10000E 00	-0.60312E 01	0.17644E 03
0.10471E 00	-0.60323E 01	0.17627E 03
0.10965E 00	-0.60334E 01	0.17610E 03
0.11482E 00	-0.60346E 01	0.17592E 03
0.12023E 00	-0.60360E 01	0.17572E 03
0.12589E 00	-0.60375E 01	0.17552E 03
0.13183E 00	-0.60391E 01	0.17531E 03
0.13804E 00	-0.60409E 01	0.17509E 03
0.14454E 00	-0.60428E 01	0.17486E 03
0.15136E 00	-0.60450E 01	0.17462E 03
0.15849E 00	-0.60473E 01	0.17437E 03
0.16596E 00	-0.60499E 01	0.17410E 03
0.17378E 00	-0.60527E 01	0.17382E 03
0.18197E 00	-0.60558E 01	0.17353E 03
0.19055E 00	-0.60591E 01	0.17323E 03
0.19953E 00	-0.60628E 01	0.17291E 03
0.20893E 00	-0.60669E 01	0.17258E 03
0.21878E 00	-0.60713E 01	0.17224E 03
0.22909E 00	-0.60762E 01	0.17187E 03
0.23988E 00	-0.60815E 01	0.17149E 03
0.25119E 00	-0.60874E 01	0.17109E 03
0.26303E 00	-0.60937E 01	0.17068E 03
0.27542E 00	-0.61007E 01	0.17024E 03
0.28840E 00	-0.61084E 01	0.16979E 03
0.30200E 00	-0.61168E 01	0.16931E 03
0.31623E 00	-0.61259E 01	0.16882E 03
0.33113E 00	-0.61359E 01	0.16830E 03
0.34674E 00	-0.61469E 01	0.16775E 03
0.36308E 00	-0.61589E 01	0.16719E 03
0.38019E 00	-0.61719E 01	0.16659E 03
0.39811E 00	-0.61862E 01	0.16598E 03
0.41687E 00	-0.62019E 01	0.16533E 03
0.43652E 00	-0.621[illegible]E 01	0.1[illegible]E 03

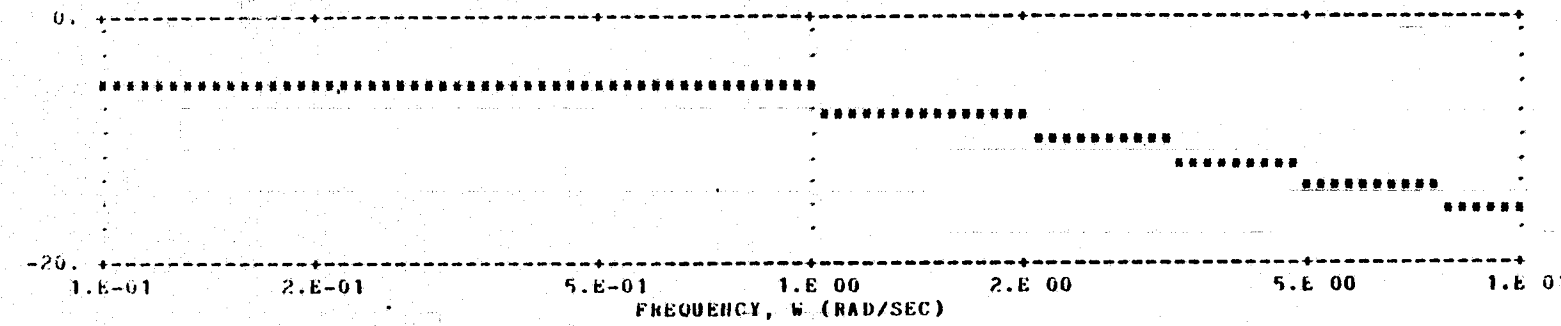
0.72444E 00	-0.65447E 01	0.15506E 03
0.75858E 00	-0.65918E 01	0.15399E 03
0.79433E 00	-0.66428E 01	0.15285E 03
0.83176E 00	-0.66979E 01	0.15168E 03
0.87096E 00	-0.67575E 01	0.15046E 03
0.91201E 00	-0.68218E 01	0.14920E 03
0.95499E 00	-0.68912E 01	0.14790E 03
0.10000E 01	-0.69658E 01	0.14656E 03
0.10471E 01	-0.70459E 01	0.14517E 03
0.10965E 01	-0.71320E 01	0.14374E 03
0.11482E 01	-0.72242E 01	0.14227E 03
0.12023E 01	-0.73227E 01	0.14075E 03
0.12589E 01	-0.74279E 01	0.13920E 03
0.13183E 01	-0.75400E 01	0.13761E 03
0.13804E 01	-0.76591E 01	0.13597E 03
0.14454E 01	-0.77855E 01	0.13430E 03
0.15136E 01	-0.79192E 01	0.13259E 03
0.15849E 01	-0.80604E 01	0.13085E 03
0.16596E 01	-0.82092E 01	0.12907E 03
0.17378E 01	-0.83655E 01	0.12726E 03
0.18197E 01	-0.85293E 01	0.12542E 03
0.19055E 01	-0.87004E 01	0.12354E 03
0.19953E 01	-0.88788E 01	0.12164E 03
0.20893E 01	-0.90641E 01	0.11972E 03
0.21878E 01	-0.92561E 01	0.11776E 03
0.22909E 01	-0.94544E 01	0.11578E 03
0.23988E 01	-0.96587E 01	0.11378E 03
0.25119E 01	-0.98683E 01	0.11175E 03
0.26303E 01	-0.10083E 02	0.10970E 03
0.27542E 01	-0.10301E 02	0.10762E 03
0.28840E 01	-0.10524E 02	0.10552E 03
0.30199E 01	-0.10749E 02	0.10338E 03
0.31623E 01	-0.10976E 02	0.10121E 03
0.33113E 01	-0.11205E 02	0.99012E 02
0.34674E 01	-0.11435E 02	0.96770E 02
0.36308E 01	-0.11664E 02	0.94484E 02
0.38019E 01	-0.11892E 02	0.92146E 02
0.39811E 01	-0.12119E 02	0.89751E 02
0.41687E 01	-0.12343E 02	0.87291E 02
0.43652E 01	-0.12565E 02	0.84756E 02
0.45709E 01	-0.12783E 02	0.82137E 02
0.47863E 01	-0.12997E 02	0.79423E 02
0.50119E 01	-0.13207E 02	0.76604E 02
0.52481E 01	-0.13412E 02	0.73666E 02
0.54954E 01	-0.13613E 02	0.70598E 02
0.57544E 01	-0.13810E 02	0.67385E 02
0.60256E 01	-0.14003E 02	0.64015E 02
0.63096E 01	-0.14193E 02	0.60472E 02
0.66069E 01	-0.14380E 02	0.56745E 02
0.69183E 01	-0.14566E 02	0.52818E 02
0.72443E 01	-0.14752E 02	0.48680E 02
0.75858E 01	-0.14939E 02	0.44318E 02
0.79433E 01	-0.15129E 02	0.39721E 02
0.83176E 01	-0.15324E 02	0.34679E 02
0.87096E 01	-0.15526E 02	0.29784E 02
0.91201E 01	-0.15736E 02	0.24428E 02
0.95499E 01	-0.15956E 02	0.18802E 02
0.10000E 02	-0.16188E 02	0.12966E 02

BODE PLOT...MAGNITUDE OF RESPONSE

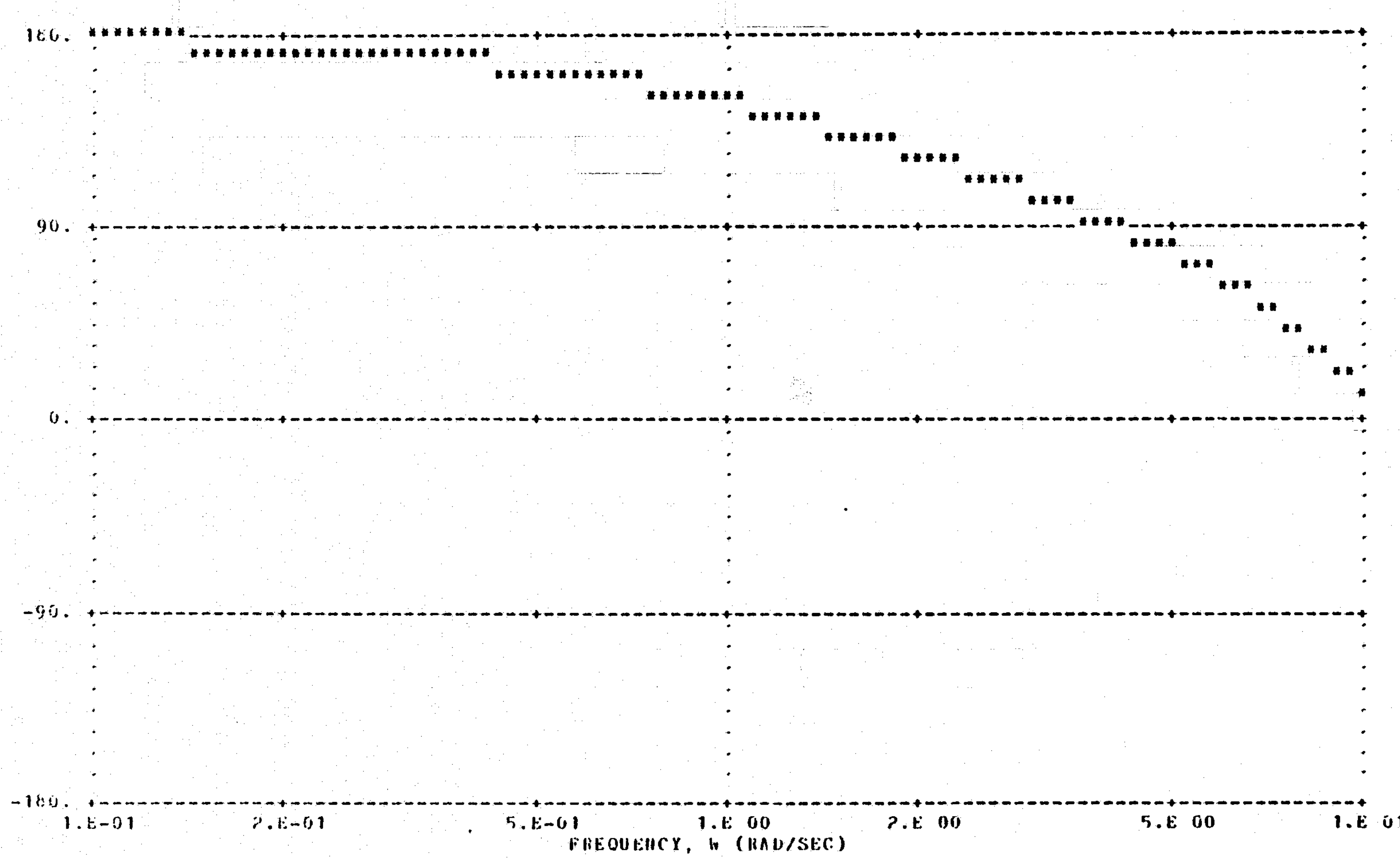


0.10000E 02-0.16188E 02 0.12900E 02

BODE PLOT...MAGNITUDE OF RESPONSE



BODE PLOT...PHASE OF THE RESPONSE



THE STEADY STATE COVARIANCE MATRIX IS

1 2 3 4 5 6 7

1.E-01 2.E-01 5.E-01 1.E 00 2.E 00 5.E 00 1.E 01

FREQUENCY, W (RAD/SEC)

THE STEADY STATE COVARIANCE MATRIX IS

	1	2	3	4	5	6	7
1	9.8678D 00	-3.2829D 00	-4.8355D-01	-3.2829D 00	-4.8355D-01	8.1886D-01	2.2167D 00
2	-3.2829D 00	4.2560D 00	4.8355D-01	2.5261D 00	3.7207D-01	-4.6767D-01	-9.3535D-01
3	-4.8355D-01	4.8355D-01	8.6547D-02	3.7207D-01	6.9986D-02	-5.5087D-02	-3.5119D-01
4	-3.2829D 00	2.5261D 00	3.7207D-01	2.5261D 00	3.7207D-01	-4.6767D-01	-9.3535D-01
5	-4.8355D-01	3.7207D-01	6.9986D-02	3.7207D-01	6.9986D-02	-5.5087D-02	-3.5119D-01
6	8.1886D-01	-4.6767D-01	-5.5087D-02	-4.6767D-01	-5.5087D-02	1.0877D-01	-1.6376D-15
7	2.2167D 00	-9.3535D-01	-3.5119D-01	-9.3535D-01	-3.5119D-01	-1.5543D-15	3.6947D 00

COST INDEX

RJX = -0.328290D 01
RJU = -0.327400D 01
RJ = -0.655690D 01

*** FC, GC, HC, WEIG, VY, AND VA WRITTEN TO DISK.

ORIGINAL PAGE IS OF POOR QUALITY

Table 2.3
Description of Sample Execution

LINE NUMBER(S)	GROUP TYPE	CARD TYPE	DESCRIPTION
1	1	1	Title for run
2-8		2	Option card. Plant, human and closed loop dynamics will be done. Covariance will be calculated and closed loop system will be saved on file for use by time varying program.
9	2	1	Comment card indicating start of plant dynamics input
10		2	Indicates 2 states, 1 control, 1 measurement, 1 noise, no transfer functions, no eigenvalues, singularity of FP unknown, regular run
11-12		4	FP matrix
13-14		5	GP matrix
15		6	HP matrix
16		7	DP matrix
17-18		8	ΓP matrix
19	3	1	Comment card indicating start of human dynamics input
20		2	Indicates 2 states, 1 control, 1 measurement, 1 noise, 1 transfer function desired, no eigenvalues, singularity of FH unknown
21		3	Transfer function desired is for row 1, column 1; zeros, residues, and bode plots are desired; plot bounds are -1 to 1 and normal size plots should be done
22		3A	1 coefficient in the numerator of polynomial multiplying transfer function before plotting; value is 1.0

Table 2.3 (Continued)

LINE NUMBER(S)	GROUP TYPE	CARD TYPE	DESCRIPTION
23		3B	1 coefficient in the denominator of polynomial multiplying transfer function before plotting; value is 1.0
24-25		4	FH matrix
26-27		5	GH matrix
28		6	HH matrix
29		7	DH matrix
30-31		8	ΓH matrix
32-33		9	QH matrix
34	4	1	Comment card indicating start of augmented states input
35		2	No. of augmented states is 2
36-37		3	FA matrix
38-39		4	GA matrix
40		5	HA matrix
41		6	DA matrix
42		7	VA matrix
43	6	1	Comment card indicating start of closed loop input
44		2	No. of rows in unaugmented HC is 6, 1 transfer function desired, no eigenvalues, HC to be set to identity matrix
45		3	Transfer function desired is for row 1, column 1; zeros, residues and bode plots with bounds of -1 to 1 are desired; normal size plots

Table 2.3 (Concluded)

LINE NUMBER(S)	GROUP TYPE	CARD TYPE	DESCRIPTION
46		3A	1 coefficient in numerator of polynomial multiplying transfer function before plotting; value is 1.0
47		3B	1 coefficient in denominator of polynomial multiplying transfer function before plotting; value is 1.0
48-49		4	CC matrix for augmenting GC
50		5	BC matrix for augmenting GC
51	7	1	Comment card for additional human dynamics input
52		2	g matrix
53		3	W matrix
54		4	Vy matrix

2.7 MAIN PROGRAM (ONLY) LISTING

The following is a computer listing for the program.

```
C
C   TRANSFER FUNCTION PROGRAM WITH SEVERAL OPTIONS---
C   (1) PLANT DYNAMICS - COMPUTE Y/U OR Y/W
C        XDOT = FX + GU + GAMMAW
C       Y     = HX + DU
C
C   (2) HUMAN OPERATOR DYNAMICS - COMPUTE U/Y
C       SEE ROUTINE MHUMAN FOR STRUCTURE.
C
C   (3) HUMAN OPERATOR + PLANT DYNAMICS
C       COMPUTE   E/F, WHERE
C       E =(A*ADJB*C)  *(A2*ADJB2*C2)  , FIRST ( ) FROM (1) ABOVE,
C                                        SECOND( ) FROM (2) ABOVE
C       F =(DETD)*(DETD1)
C
C   (4) CLOSED LOOP RESPONSE
C        ZDOT = FZ +GAMMAW    , SEE ROUTINE MCLOSE FOR DESCRIPTION OF F AN
C       Y     = HZ
C
C   (5)
C     C  INPUT---
C     ALL INPUT IS PERFORMED IN INIT AND INIT2.
C     THERE ARE 7 GROUPS OF INPUT SETS.
C     GROUP 1 = TITLE AND OPTIONS
C     GROUP 2 = INPUT FOR PLANT DYNAMICS
C     GROUP 3 = INPUT FOR HUMAN DYNAMICS
C      GROUP 4 = INPUT FOR AUGMENTED STATES
C     GROUP 5 = INPUT FOR HUMAN OPERATOR + PLANT DYNAMICS
C     GROUP 6 = INPUT FOR CLOSED LOOP RESPONSE
C     GROUP 7 = ADDITIONAL INPUT FOR HUMAN DYNAMICS
C
C     THE PROGRAM LOGIC IS SUCH THAT IF THE OPTION IS SET SO THAT
C     HUMAN DYNAMICS ARE DONE, THE PROGRAM ITERATES ON GROUP 7 INPUT.
C     OTHERWISE, IT WILL ITERATE STARTING AT GROUP 1.
C
C     ALL MATRICES ARE READ BY ROWS ACCORDING TO FORMAT IN ARRAY IFMT.
C     (SEE GROUP 1 INPUT DESCRIPTION)
C
C     ALL CARDS MARKED WITH * ARE CONDITIONAL INPUT.
C
C     GROUP 1  ALWAYS INCLUDED
C      CARD  FORMAT        DESCRIPTION
C       1    (13A6)        TITLE(13) = TITLE TO BE PRINTED AT THE TOP
C                                      OF EACH PAGE
C       2    NAMELIST / OPTION /
C                          PLANT = T, TO DO PLANT DYNAMICS (DEFAULT=F)
C                          HUMAN = T, TO DO HUMAN DYNAMICS (DEFAULT=F)
C                          HOPLT = T, TO DO HUMAN OPERATOR + PLANT DYNAMICS
C                                  (DEFAULT=F)
C                          CLOSED = T,TO DO CLOSED LOOP RESPONSE
C                                   (DEFAULT=F)
C                          COVAR = T, TO DO COVARIANCES
C                                  (DEFAULT=F)
C                          IFMT =  13 WORD ARRAY FOR VARIABLE INPUT FORMAT.
C                                  ALL CARD MATRICES ARE READ WITH THIS FOR
C                                  (DEFAULT = 8E10.4)
C                          MINOUT = T, FOR MINIMAL OUTPUT
C                          BACKWD = T, TO USE BACKWARD LEVERRIER IN
C                                   COMPUTING ADJOINT EVEN IF NO. OF STATES
C                                   IS LESS THAN 5. (AUTOMATICALLY TRIES
C                                   BACKWARD IF NO. STATES .GE. 5 )
```

59

ORIGINAL PAGE IS
OF POOR QUALITY

```
C                               ELEMENTS WILL BE READ FROM.
C                               (DEFAULT=5)
C                         DISK = T IF THE PLANT F AND G ARE TO
C                               BE READ OFF OF UNIT 9
C                         ITERC = T, TO ITERATE ON VY AND/OR VA
C                         FCNTRL = T, TO SWITCH FC MATRIX TO CONTROL ONLY
C                               (DEFAULT = F)
C                         ITERVY = NO. OF ITERATIONS FOR VY ( DEFAULT=0)
C                         ITERVA = NO. OF ITERATIONS FOR VA ( DEFAULT = 0)
C                         RHOY = CONSTANT FOR VY (DEFAULT=.0314159)
C                         RHOA = CONSTANT FOR VA (DEFAULT=.092477)
C                         SAVSYS = T, IF FC, GC, HC, WEIG, VY, VA ARE TO
C                               BE WRITTEN TO DISK UNIT 21
C                         IFMT2 = 13 WORD ARRAY FOR VARIABLE FORMAT USED
C                               FOR FILES USED WHEN DISK=T OR SAVFCA=T.
C                               ( DEFAULT=4E20.15 )
C
C  GROUP 2* INCLUDE ONLY IF PLANT=T
C   CARD  FORMAT          DESCRIPTION
C     1   13A6            ICMMNT(13) = COMMENT CARD DENOTING START OF
C                                      PLANT DYNAMICS INPUT. STRICTLY
C                                      FOR USER IDENTIFICATION.
C     2   16I5            NSP = NO. OF STATES
C                         NCP = NO. OF CONTROLS
C                         NMP = NO. OF MEASUREMENTS
C                         NOP = NO. OF NOISE
C                         NTFP = NO. OF SPECIFIC TRANSFER FUNCTIONS TO BE
C                               DONE
C                         JEIGFP = EIGENVALUE FLAG FOR F
C                               -1, COMPUTE AND SKIP ALL OTHER PLANT
C                                   DYN. CALCULATIONS.
C                                0, DO NOT COMPUTE
C                                1, COMPUTE AND CONTINUE
                          JFSNGP = SINGULARITY FLAG FOR F
                                0, NOT KNOWN
                                1, F KNOWN TO BE SINGULAR
60                        JENTP = ENTRY FLAG
C                               0, REGULAR CALCULATIONS
C                               1, IF TRN. FCN. WAS COMPUTED AND SAVED
C                                  IN A PREVIOUS RUN, AND ONLY SPECIFIC
C                                  TRANSFER FUNCTIONS ARE NOW DESIRED.
C     3*  16I5            READ NTFP SETS OF CARD TYPE 3-3B (NTFP MAY = 0)
C                         IRWCLP(1) = ROW NO. OF TRN. FCN. DESIRED
C                               (2) = COL. NO. OF TRN. FCN. DESIRED
C                               (3) = 0, FOR ZEROES
C                                     1, FOR ZEROES + RESIDUES
C                                     2, FOR BODE PLOTS
C                                     3, FOR EVERYTHING
C                               (4) = LOWER BOUND OF BODE PLOT (LOG)
C                               (5) = UPPER BOUND OF BODE PLOT (LOG)
C                               (6) = 0, NORMAL SIZE BODE PLOT
C                                     1, DOUBLE SIZE BODE PLOT
C     3A* I5,7E10.5       INCLUDE ONLY IF IRWCLP(3) .GE. 2
C                         NCOEFN = NO. OF COEFFICIENTS IN NUMERATOR
C                               OF THE N/D POLYNOMIAL WHICH MULTIPLIES
C                               THE TRN. FCN. BEFORE PLOTTING.
C                         COEFNP(I,J) = COEFFICIENTS OF THE NUMERATOR
C                                     IN DESCENDING ORDER.
C     3B* I5,7E10.5       INCLUDE ONLY IF IRWCLP(3) .GE. 2
C                         NCOEFD  = NO. OF COEFFICIENTS IN DENOMINATOR
C                         COEFDP(I,J) = DENOMINATOR COEFFICIENTS IN
C                                     DESCENDING ORDER.
C   4-4X* IFMT            F MATRIX  INCLUDE ONLY IF JENTP = 0
C   5-5X* IFMT            G MATRIX  INCLUDE ONLY IF JEIGFP .NE. -1,JENTP=0
C
```

```
C      7-7X* IFMT          D MATRIX  INCLUDE ONLY IF JEIGFP .NE. -1,JENTP=0
C
C      8-6X* IFMT          GAMMA MTRX INCLUDE ONLY IF JEIGFP .NE. -1,JENTP=0
C
C     GROUP 3*  INCLUDE ONLY IF HUMAN=T
C      CARD  FORMAT        DESCRIPTION
C        1    13A6         ICMMNT(13) = SEE CARD 1, GROUP 2
C
C        2   16I5          NSH = NO. OF STATES
C                          NCH = NO. OF CONTROLS
C                          NMH = NO. OF MEASUREMENTS
C                          NOH = NO. OF NOISE
C                          NTFH = NO. OF TRN. FCNS. DESIRED
C                          JEIGFH = EIGENVALUE FLAG. SEE CARD 2, GROUP 2
C                          JFSNGH = SINGULARITY FLAG FOR F. SEE CARD 2,
C                                   GROUP 2
C        3*  16I5          READ NTFH SETS OF CARD TYPE 3-3B (NTFH MAY = 0)
C                          IRWCLH(6) = SEE CARD 3 , GROUP 2
C       3A*  I5,7E10.5     NCOEFN      = SEE CARD 3A, GROUP 2
C                          COEFNH(I,J)=SEE CARD 3A, GROUP 2
C       3B*  I5,7E10.5     NCOEFD      =SEE CARD 3B, GROUP 2
C                          COEFDH(I,J)=SEE CARD 3B, GROUP 2
C
C      4-4X  IFMT          F MATRIX
C
C      5-5X  IFMT          G MATRIX
C
C      6-6X  IFMT          H MATRIX
C
C      7-7X  IFMT          D MATRIX
C
C      8-8X  IFMT          GAMMA MATRIX
C
C      9-9X  IFMT          Q MATRIX
C     GROUP 4*  INCLUDE ONLY IF HUMAN=T
C      CARD  FORMAT        DESCRIPTION
C        1   13A4          ICMMNT(13), SEE CARD 1, GROUP 2
C        2     I5          NSA = NO. OF AUGMENTED STATES
C        3   IFMT          FA MATRIX
C        4   IFMT          GA MATRIX
C        5   IFMT          HA MATRIX
C        6   IFMT          DA MATRIX
C        7*  IFMT          VA MATRIX, INCLUDE ONLY IF COVAR=T
C
C     GROUP 5*  INCLUDE ONLY IF HOPLT=T
C      CARD  FORMAT        DESCRIPTION
C        1   13A6          ICHMNT(13) = SEE CARD 1, GROUP 2
C
C        2   I5            NTF3 = NO. OF TRANSFER FUNCTIONS TO BE DONE
C
C        3*  16I5          READ NTF3 SETS OF CARD TYPE 3-3B (NTF3 MAY=0)
C                          IRWCL3(6)   = SEE CARD 3 , GROUP 2
C       3A*  I5,7E10.5     NCOEFN      = SEE CARD 3A, GROUP 2
C                          COEFN3(I,J)= SEE CARD 3A, GROUP 2
C       3B*  I5,7E10.5     NCOEFD      = SEE CARD 3B, GROUP 2
C                          COEFD3(I,J)= SEE CARD 3B, GROUP 2
C
C     GROUP 6*  INCLUDE ONLY IF CLOSED=T
C      CARD  FORMAT        DESCRIPTION
C        1   13A6          ICMMNT(13) = SEE CARD 1, GROUP 2
C        2   16I5          NRC = NO. OF ROWS IN HC
C                          NTFC=NO. OF TRN FCNS. DESIRED
C                          JEIGFC = EIGENVALUE FLAG, SEE CARD 2, GROUP 2
C                          IDNTHC = H = IDENTITY MATRIX FLAG
C                                   0, H WILL BE READ IN
```

```
C                                           I, H WILL BE SET TO IDENTITY
C       3*  1615          REAL NTFC SETS OF CARD TYPE 3-3E (NTFC MAY = 0)
C                         IRWCLC(6)    = SEE CARD 3 , GROUP 2
C      3A*  I5,7E10.5     NCOEFN       = SEE CARD 3A, GROUP 2
C                         COEFNC(I,J)= SEE CARD 3A, GROUP 2
C      3B*  I5,7E10.5     NCOEFD       = SEE CARD 3B, GROUP 2
C                         COEFDC(I,J)= SEE CARD 3B, GROUP 2
C
C
C      4   IFMT           CC MATRIX ( FOR GC )
C      4A  IFMT           BC MATRIX ( FOR GC )
C      5*  IFMT          INCLUDE ONLY IF IDNTHC = 0
C                         H MATRIX
C
C
C   GROUP 7*  INCLUDE ONLY IF HUMAN=T, ALL MATRICES FOR GAINS CALC.
C    CARD  FORMAT         DESCRIPTION
C      1   13A6           ICMMNT(13) = SEE CARD 1, GROUP 2
C
C      2   IFMT           G SMALL MATRIX
C
C      3  IFMT           W BIG MATRIX
C
C      4  IFMT           VY MATRIX
C
C
C  FILES USED--
C    9  INPUT UNIT FOR F AND G ( PLANT ) WHEN DISK=TRUE
C   10  SCRATCH UNIT (ALWAYS NEEDED)
C   11  OUTPUT UNIT FOR PLANT DYNAMICS (TEMPORARY OR PERMANENT)
C   12  OUTPUT UNIT FOR HUMAN DYNAMICS (     "       "       "     )
C   13  OUTPUT UNIT FOR HOPLT DYNAMICS (     "       "       "     )
C   14  OUTPUT UNIT FOR CLOSED LOOP    (     "       "       "     )
C
      IMPLICIT  REAL*8 ( A-H, O-Z )
C
      LOGICAL  PLANT, HUMAN, CLOSED, HOPLT
     .        , MINOUT, BACKWD, DMATXP, DMATXH, DMATXC
      LOGICAL IEOF
      LOGICAL COVAR
      LOGICAL SAVSYS, FCNTRL, ITERC, LITER, AITER
C
      COMMON  / LOGIC  / PLANT, HUMAN, HOPLT, CLOSED, MINOUT
     .                  ,BACKWD, DMATXP, DMATXH, DMATXC
     .                  ,ITERC, FCNTRL, AITER
      COMMON  / SAVE   /  SAVSYS
      COMMON  / UNIT52 / IUFG,  IUSYS
      COMMON  / MAXDIM / NSPMX, NCPMX, NMPMX, NOPMX, NSHMX, NCHMX
     .                 , NMHMX, NOHMX, NS3MX, NC3MX, NM3MX
     .                 , NSCMX, NCCMX, NMCMX
      COMMON / UNITS / IUP, IUH, IU3, IUC, JUNIT
      COMMON /VAR/ COVAR
C
C  PLANT DYNAMICS
C
      DIMENSION FP(30,30),GP(30,4),HP(10,30),DP(10,4),GAMMP(30,4)
      DIMENSION PP(31),IRWCLP(6,40),G(30,8),NDEGNP(40)
      DIMENSION COEFNP(6,40),NDEGDP(40),COEFDP(6,40)
C
C  HUMAN DYNAMICS
C
      DIMENSION FH(15,15),GH(15,4),HH(10,15),DH(10,4),GAMMH(15,4)
      DIMENSION CH(15,15),FGAIN(15,10),CGAIN(4,15),IRWCLH(6,16)
      DIMENSION PH([illegible]),NDEGNH(16),COEFNH(6,16),NDEGDH(16)
      DIMENSION COEFDH(6,16),GSML(4,4),WBIG(4,4),VY(10,10)
```

```
C
C  AUGMENTED STATES
C
      DIMENSION FA(5,5), GA(5,4), HA(4,5), DA(4,4), VA(4,4)
C
C  HUMAN OPERATOR + PLANT DYNAMICS
C
      DIMENSION P3(51),NDEGN3(16),COEFN3(6,16)
      DIMENSION NDEGD3(16),COEFD3(6,16),IRWCL3(6,16)
C
C  CLOSED LOOP DYNAMICS
C
      DIMENSION FC(35,35),GC(35,19),HC(35,35),PC(36),XCOV(35,35)
      DIMENSION NDEGNC(1225),COEFNC(6,1225),NDEGDC(1225),COEFDC(6,1225)
     DIMENSION  IRWCLC(6,1225), DC(35,35)
      DIMENSION  CC ( 4,4 ), HCORG(35,35), BC (4,4 )
C
C  OTHER ARRAYS
C
      DIMENSION BPLT(165),EIGR(35),EIGI(35),WORK(5000)
      DIMENSION FWORK( 35,35 )
      DIMENSION YU(36,1225),NDEGYU(1225)
C
C  SET CONSTANTS ACCORDING TO PARAMETERS
C
      MXCO=6
      MXYU=36
      NSPMX=30
      NCPMX=8
      NMPMX=10
      NOPMX=4
      NSHMX=15
      NCHMX=4
      NMHMX=10
      NOHMX=4
      NS3MX=30
      NC3MX=8
      NM3MX=10
      NSCMX=35
      NCCMX=19
      NMCMX=35
C
C  READ ALL INPUT EXCEPT GSML, VY, WBIG AND
C  DO SOME ITITIALIZATION
C
  100 CONTINUE
      CALL INIT ( FP, GP, HP, DP, GAMMP, G, IRWCLP                  -
     .           , NDEGNP, NDEGDP, COEFNP, COEFDP                  -
     .           , FH, GH, HH, GAMMH, OH, IRWCLH, NDEGNH           -
     .           , NDEGDH, COEFNH, COEFDH, DH                      -
     .           , IRWCL3, NDEGN3, NDEGD3, COEFN3, COEFD3          -
     .           , HCORG, DC, FA, GA, HA, DA, VA                     -
     .           , IRWCLC, NDEGNC, NDEGDC, COEFNC, COEFDC         -
     .           , CC, BC                                          -
     .           , MXCO, IEOF )
      IF (IEOF ) GO TO 999
C
C  PLANT DYNAMICS
C
      IF ( .NOT. PLANT) GO TO 200
      CALL MPLANT ( FP, GP, HP, DP, PP, YU , MXYU , IRWCLP, NDEGYU     -
     .             , NDEGNP, NDEGDP, COEFNP, COEFDP, MXCO             -
     .             , BPLT, EIGR, EIGI, WORK, FWORK )
C
C  HUMAN DYNAMICS
C
```

63

ORIGINAL PAGE IS
OF POOR QUALITY

```
  230 CONTINUE
      IF ( .NOT. HUMAN) GO TO 700
C
C  READ IN GSML, WBIG, VY
C
  250 CONTINUE
      CALL INIT2 ( GSML, WBIG, VY,  JUNIT, IEOF )
      IF ( IEOF ) GO TO 999
  260 CONTINUE
C
C  COMPUTE THE CONTROL AND FILTER GAINS ( LAMBDA AND K )
C
      CALL MGAINS ( FH, GH, HH, QH, GSML, WBIG, VY, GAMMH
     .             , WORK, CGAIN, FGAIN )
C
C  COMPUTE NEW F MATRICES FOR HUMAN AND CLOSED LOOP
C  AND NEW H MATRIX FOR CLOSED LOOP
C
      CALL MFHFC ( FP, G, FH, GH, HH, DH, FA, GA, HA, DA, CGAIN, FGAIN
     .            , WORK, FHH, FC ,HCORG, HC )
      CALL MHUMAN ( FHH, GHH, HHH, DHH, HA, DA, FGAIN, CGAIN, PH
     .             , YU , MXYU , IRWCLH, NDEGYU, NDEGNH, NDEGDH
     .             , COEFNH, COEFDH, MXCO, BPLT, EIGR, EIGI, WORK )
C
C  HUMAN OPERATOR + PLANT DYNAMICS
C
      IF ( .NOT. HOPLT) GO TO 400
      CALL MHOPLT ( P3, YU , MXYU , IRWCL3, NDEGYU
     .             , NDEGN3, NDEGD3, COEFN3, COEFD3, MXCO
     .             , BPLT, EIGR, EIGI, WORK , PP, PH )
C
C  CLOSED LOOP RESPONSE
C
  400 CONTINUE
      IF ( .NOT. CLOSED ) GO TO 600
      CALL MCLOSE ( G, GH, GAMMP, FGAIN, FC, GC, HC, DC, CC, BC
     .             , FC, YU , MXYU , IRWCLC, NDEGYU
     .             , NDEGNC, NDEGDC, COEFNC, COEFDC, MXCO
     .             , BPLT, EIGR, EIGI, WORK, FWORK  )
C
      IF(.NOT.COVAR) GO TO 450
C
C  COMPUTE COVARIANCE
C
      CALL MCOV(XCOV,WBIG,VY,VA,FC,GC,HC,WORK, FWORK)
      CALL MCOST(XCOV,QH,GSML,CGAIN,RJX,RJU,RJ,WORK)
C
C
 450  CONTINUE
      IF ( .NOT. ITERC ) GO TO 500
C
C  COMPUTE NEW VY AND/OR VA
C
      CALL MITER ( XCOV, VY, VA, WORK, LITER, AITER )
      IF ( LITER ) GO TO 500
      GO TO 260
  500 CONTINUE
C
C  SAVE THE SYSTEM
C
      IF ( .NOT. SAVSYS ) GO TO 600
      CALL FCSAVE ( FC, GC, HC, WBIG, VY, VA )
C
C  GET A NEW GSML, WBIG AND VY
C
```

```
      GO TO 250
C
C     GET A NEW CASE
C
  700 CONTINUE
      GO TO 100
C
C
  999 CONTINUE
      STOP
      END
```

III. TIME VARYING SYSTEM RESPONSE PROGRAM (TVSR)

3.1 DESCRIPTION

The Time Varying System Response Program (TVSR) solves the differential equation

$$\dot{z}(t) = F_c(t)z(t) + G_c(t)\theta(t)$$

$$y(t) = H_c(t)x(t) \qquad (3.1)$$

The system matrices F_c, G_c and H_c are not computed in the program, but rather are read in from a disc file. The modified transfer function program described in the previous section, writes out these system matrices in a suitable format. Appendix A describes a stand-alone program which also can be used as generic input.

The time varying response output can be of three types:

(1) mean (expected) value of states/outputs

(2) sample time histories

(3) state/output covariance

The system matrices are read in at the start and end point for some interval of time, which has been specified. These matrices need not have the same coefficients at the beginning and end. An algorithm interpolates between these matrices at intervals specified by the user. By this method, a linear piecewise solution may be obtained for the time varying system.

An added feature of the code is that several time segments can be "pieced" together to make a longer time record. At the end of any one record, all relevant information is stored on

PRECEDING PAGE BL...

disc. The next time segment uses this information as new start conditions.

This sequence is shown in the following figure.

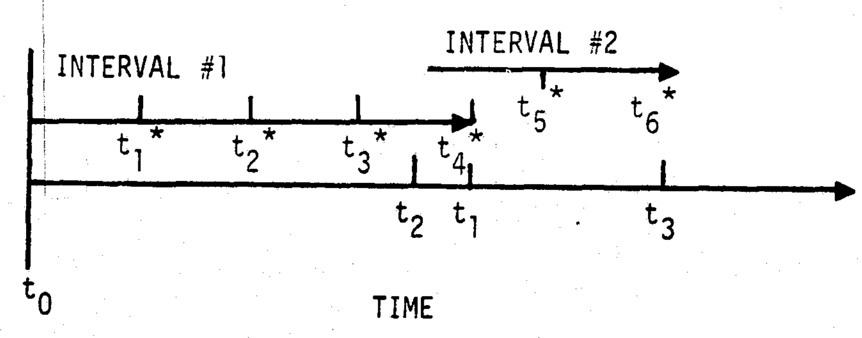


Figure 3.1

(1) Interval #1 (from t_0 to t_1) requires one execution of the program. The system matrices (F_c, G_c, H_c) must be previously stored on disc corresponding to $t = t_0$ and $t = t_1$. Four interpolation intervals have been specified (corresponding to time points t_1^*, t_2^*, t_3^* and t_4^*). The state at t_2 has been specified for storage for initializing interval #2. (Note: t_2 need not be equal to t_1, i.e. $t_2 \leq t_1$.)

(2) Interval #2 (from t_2 to t_3) is now executed with two interpolations (t_5^* and t_6^*) using initial conditions stored from the previous execution.

The interpolation algorithm cues the value stored in the output vector, y, in the location INDEXV (card group 2). The variable, such as altitude, may be used. It is important to specify H_c properly so as to ensure the variable appearing in the correct location in the output vector, y.

The structure of the code is such that the form of $G_c\,\theta$ can be augmented from that written to disc from the transfer function program. The transfer function program assumes θ is the form:

$$\theta = \begin{bmatrix} w(t) \\ v_y(t) \\ v_a(t) \end{bmatrix} \; ; \; \begin{matrix} w \sim N(0,W) \\ v_y \sim N(0,V_y) \\ v_a \sim N(0,V_a) \end{matrix}$$

where W, V_y and V_a are stored on the disc. For increased versatility, a deterministic input of the form $p(t)$; where $p(t)$ is a polynomial function (with coefficients read into the TVSR program) of time. This requires augmentation of $G_c\,\theta$ of the form:

$$\left[\begin{array}{c|c} & G_{c_1} \\ G_c & G_{c_2} \\ & G_{c_a} \end{array}\right] \begin{bmatrix} w \\ v_y \\ v_a \\ p \end{bmatrix}$$

where now G_{c_1}, G_{c_2} and G_{c_a} must also be read in.

The covariance of the augmented θ vector is assumed to be of the form:

$$\begin{bmatrix} W & 0 & 0 & 0 \\ 0 & V_y & 0 & 0 \\ 0 & 0 & V_a & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

since p(t) is deterministic. This matrix is used in the state covariance calculation which employs OPTSYS in the solution of the steady state values. The state covariance is evaluated only at the times designated as linear interpolation times (t_i^* in the figure), since the system matrices only change at these times. The covariance matrix can be evaluated even when sample function option is selected, since this uses a separate algorithm for its calculation.

The covariance that is printed is actually the covariance of y, the output. This is equivalent to the state covariance only when $H_c = I$.

One note of caution is required in the utilization of the output vector, $y = H_c Z$. Specifically, the first NC variables (where NC = number of control variables - output of the operator) of y is constrained to be u where (see Section II)

$$u = -D_a \Lambda \hat{x} + H_a x_a$$

Thus, y is assumed to be of the form

$$y = \left[\begin{array}{c|c|c} 0 & -D_a \Lambda & H_a \\ \hline & H_c & \end{array}\right] \begin{bmatrix} x \\ \hat{x} \\ x_a \end{bmatrix}$$

where H_c is read in from cards (or set to I internal in the program).

3.2 INPUT SPECIFICATIONS

The program receives its input through cards and disk files.

3.2.1 Card Input

There are two cards required for each run. Card 1 is a title card for output identification. Card 2 is the option specification card. All succeeding input is dependent upon the options selected on Card 2. All matrices utilizing card input are read in by rows. The default format is 8E10.4 which may be changed by Card 2. Table 3.1 describes the required and optional card input.

3.2.2 Disk Files

The program uses four disk files:

(1) System at start time (input only)

(2) System at end time (input only)

(3) Z and H*Z vectors (input/output)

(4) Steady state and steady state output covariance (input/output)

The user need only be concerned with the structure of (1) and (2).

3.2.2.1 File for System at Start Time

The unit for the system at start time is referred to as IUFC1 and has a default value of 10. It is a sequential binary file. The file must contain seven logical records. All records must appear on the file, even if the user overrides them by providing card input or sets H_c to the identity matrix.

The format of file IUFC1 is given in Table 3.2.

Table 3.1

Card Input

CARD	FORMAT	VARIABLE NAME	TYPE	DESCRIPTION
1	13A4	ITITLE	A	52 character title to be printed at the top of each output page
2	Namelist /Option/	MEAN	L	= T, to compute mean calculations (Default = F)
		SAMPLE	L	= T, to compute the sample function (Default = F)
		PLOT	L	= T, to produce printer plots (Default = F)
		COVAR	L	= T, to compute covariance and save on disk (Default = F)
		HIDENT	L	= T, to set H1, H2 to the identity matrix (Default = F, use H1, H2 on disk files IUFC1, IUFC2)
		WCARD	L	= T, to read W1, W2 from cards (Default = F, use W1, W2 on disk files IUFC1, IUFC2)
		VYCARD	L	= T, to read VY1, VY2 from cards (Default = F, use VY1, VY2 on disk files IUFC1, IUFC2)
		ZCARD	L	= T, to read initial Z vector from cards (Default = F, use Z vector on disk file IUZ)
		VACARD	L	= T, to read VA1, VA2 from cards (Default = F, use VA1, VA2 on disk files IUFC1, IUFC2)
		IUFC1	I	Unit No. containing FC1, GC1, HC1, W1, VY1, VA1 (Default = 10)

A = Alphanumeric
I = Integer
L = Logical
D.P. = Double Precision

Table 3.1 (Continued)

CARD	FORMAT	VARIABLE NAME	TYPE	DESCRIPTION
2		IUFC2	I	Unit No. containing FC2, GC2, HC2, W2, VY2, VA2 (Default = 11)
		IUZ	I	Unit No. containing Z, H*Z vectors (Default = 12)
		IUCOV	I	Unit No. containing steady state and steady state output covariance (Default = 13)
		LASTZ	I	= n, record no. of Z vector on file IUZ to be used as initial Z. (This only needs to be set if you do not want the last Z written.) = 0, will use last Z written or will start a new file if ZCARD = T. (Default =0)
		NPRINT	I	Increment for printing (Default = 0, last Z only will be printed)
		NSTORE	I	Increment for saving Z and H*Z vectors (Default = 0, nothing is saved)
		NINT	I	No. of intervals between TSTART and TSTOP (Default = 1)
		NTERMS	I	No. of terms to use in computing PHI and PSI (transition matrices) (Default = 10)
		TSTART	D.P.	Start time (Default = 0.0)
		TSTOP	D.P.	Stop time (Default = 0.0)
		TDELT	D.P.	Delta time between TSTART and TSTOP (Default = 1.0)
		NCOEF	I	No. of coefficients in p matrix ($\leq$5, Default = 3)
		ISEED	I	Seed for random in generator (Default = 328765)

A = Alphanumeric
I = Integer
L = Logical
D.P. = Double Precision

Table 3.1 (Continued)

CARD	FORMAT	VARIABLE NAME	TYPE	DESCRIPTION
2		NPSIZE	I	No. of rows in p matrix (Default=0)
		DELTAV	D.P.	Desired total change in velocity
		INDEXV	I	Location of velocity in H*Z vector (Default=1)
		IFMT(13)	I	Array containing format for reading matrices from cards (Default = 8E10.4)
3A*	13A4	ICMMT	A	Comment card for user identification of p matrix
3-3n*	IFMT	PSML	D.P.	Matrix of coefficients (NPSIZE x NCOEF) used in computing θ. Coefficients should be in decreasing order. *Include only if NPSIZE>0
4A*	13A4	ICMMT	A	Comment card for user identification of W1 matrix
4-4n*	IFMT	W1	D.P.	Matrix (NNP x NNP) to be used in white noise sequence of sample function and/or covariance. *Include only if WCARD = T and (Sample = T or Covar = T).
5A*	13A4	ICMMT	A	Same as 4A except for W2.
5-5n*	IFMT	W2	D.P.	Same as 4-4n except for W2

A = Alphanumeric
I = Integer
L = Logical
D.P. = Double Precision

Table 3.1 (Continued)

CARD	FORMAT	VARIABLE NAME	TYPE	DESCRIPTION
6A*	13A4	ICMMT	A	Comment card for user identification of VY1 matrix
6-6n*	IFMT	VY1	D.P.	Matrix (NMH x NMH) to be used in white noise sequence of sample function and/or covariance. *Include only if VYCARD = T and (Sample = T or Covar = T)
7A*	13A4	ICMMT	A	Same as 6A except for VY2
7-7n*	IFMT	VY2	D.P.	Same as 6-6n except for VY2
8A*	13A4	ICMMT	A	Comment card for user identification of VA1 matrix
8-8n*	IFMT	VA	D.P.	Matrix (NCA x NCA) for use in computing covariance. *Include only if VACARD = T and COVAR = T.
9A*	13A4	ICMMT	A	Same as 8A except for VA2
9-9n*	IFMT	VA2	D.P.	Same as 8-8n except for VA2
10A*	13A4	ICMMT	A	Comment card for user identification of GCONE1 matrix. *Include only if NPSIZE>0.
10-10n*	IFMT	GCONE1	D.P.	Matrix (NSP x NPSIZE) for augmenting GC1.
11A*	13A4	ICMMT	A	Same as 10A* except for GCONE2
11-11n*	IFMT	GCONE2	D.P.	Same as 10-10n* except for GC2
12A*	13A4	ICMMT	A	Comment card for user identification of GCTWO1 matrix. *Include only if NPSIZE>0.
12-12n*	IFMT	GCTWO1	D.P.	Matrix (NSH x NPSIZE) for augmenting GC1.
13A*	13A4	ICMMT	A	Same as 12A* except for GCTWO2
13-13n*	IFMT	GCTWO2	D.P.	Same as 12-12n* except for GC2

A = Alphanumeric
I = Integer
L = Logical
D.P. = Double Precision

Table 3.1 (Continued)

CARD	FORMAT	VARIABLE NAME	TYPE	DESCRIPTION
14A*	13A4	ICMMT	A	Comment card for user identification of GCA1 matrix. *Include only if NPSIZE>0.
14-14n*	IFMT	GCA1	D.P.	Matrix (NSA x NPSIZE) for augmenting GC1
15A*	13A4	ICMMT	A	Same as 14A* except for GCA2
15-15n*	IFMT	GCA2	D.P.	Same as 14-14n* except for GC2
16A*	13A4	ICMMT	A	Comment card for user identification of initial Z vector
16-16n*	IFMT	Z	D.P.	Initial Z vector (NSC long) *Include only if ZCARD = T.
17A*	13A4	ICMMT	A	Comment card for user identification of plot input section. *Include only if PLOT = T.
17B*	16I5	NVARZ	I	No. of Z states to plot
		NVARY	I	No. of H*Z variables to plot
		ISD	I	= 0, do not plot standard deviation = 1, plot standard deviation associated with each variable
		ISCALE	I	= 0, use same scale for all curves = 1, plot each curve on own scale
		KREC	I	= 0, plot all output = 1, plot only output from current run
		INC	I	= 1, plot every point = n, plot every nth point Plots are done from the output files (IUZ and IUCOV).

A = Alphanumeric
I = Integer
L = Logical
D.P. = Double Precision

Table 3.1 (Concluded)

CARD	FORMAT	VARIABLE NAME	TYPE	DESCRIPTION
17B*				Therefore, setting INC =2 will give a plot of every other point on the output file. Maximum number of points that may be plotted is 2000.)
17C*	(5I5,5A4)	MVAR(1-5)	I	List of Z and H*Z variables to plot
		ID(1-5)	A	4 character identification of each variable plotted As many sets of cards 17B*-17C* as desired may be included. Plots can only be done if NSTORE ≠ 0.

A= Alphanumeric
I = Integer
L = Logical
D.P. = Double Precision

Table 3.2

Structure for Files IUFC1 and IUFC2,
Systems at Start and End Times

LOGICAL RECORD NO.	DESCRIPTION
1	NSP = No. of states (plant) NNP = No. of noise (plant) NSH = No. of states (human) NMH = No. of measurements (human) NSC = No. of states (closed loop) NIC = No. of inputs (closed loop) NMC = No. of measurements (closed loop) NSA = No. of states (augmented) NCA = No. of controls (augmented)
2	F Matrix (closed loop) (NSC x NSC)
3	G Matrix (closed loop) (NSC x NIC)
4	H Matrix (closed loop) (NMC x NSC)
5	W Matrix (NNP x NNP)
6	Vy Matrix (NMH x NMH)
7	VA Matrix (NCA x NCA)

3.2.2.2 File for System at End Time

The unit for the system at end time is referred to as IUFC2 and has default value of 11. Its description is identical to IUFC1 (System at Start Time, Section 3.2.2.1).

3.2.2.3 File for Z and H*Z Vectors

The unit containing the Z and H*Z vectors is referred to as IUZ and has default value of 12. It is a random access binary file. Its structure is defined and its records are written by the program. IUZ is used as an input file in two cases: (1) when a Z vector computed in a previous run is used as the initial Z vector in a later run and (2) when printer plots are desired.

The structure of file IUZ is given in Table 3.3.

Table 3.3
Structure for File IUZ, Z and H*Z Vectors

RECORD NO.	DESCRIPTION
1	IREC = record no. of last record written on this file (<2000) NSC = no. of states (closed loop) NMC = no. of measurements (closed loop)
2 - IREC	Time Z vector (NSC long) H*Z vector (NMC long)

3.2.4 Covariance File

The unit for the steady state and steady state output covariance is referred to as IUCOV and has default value of 13. It is a random access binary file. Its structure is defined and records are written by the program. It is used as an input file when printer plots with standard deviation are desired.

The structure of file IUCOV is given in Table 3.4.

Table 3.4
Structure for File IUCOV, Steady State and Steady State Output Covariance

RECORD NO.	DESCRIPTION
1	JREC = record no. of last record written on this file ($\leq$2000) NSC = no. of states (closed loop) NMC = no. of measurements (closed loop)
2 - JREC	IREC = record no. of file IUZ this covariance is associated with YCOV = vector of the square roots of the diagonal elements of the steady state output covariance matrix (NSC long) ZCOV = vector of the square roots of the diagonal elements of the steady state covariance matrix (NMC long)

3.3 PROGRAM FLOWCHART

The program flowchart for the TVSR program is shown in Figure 3.3.

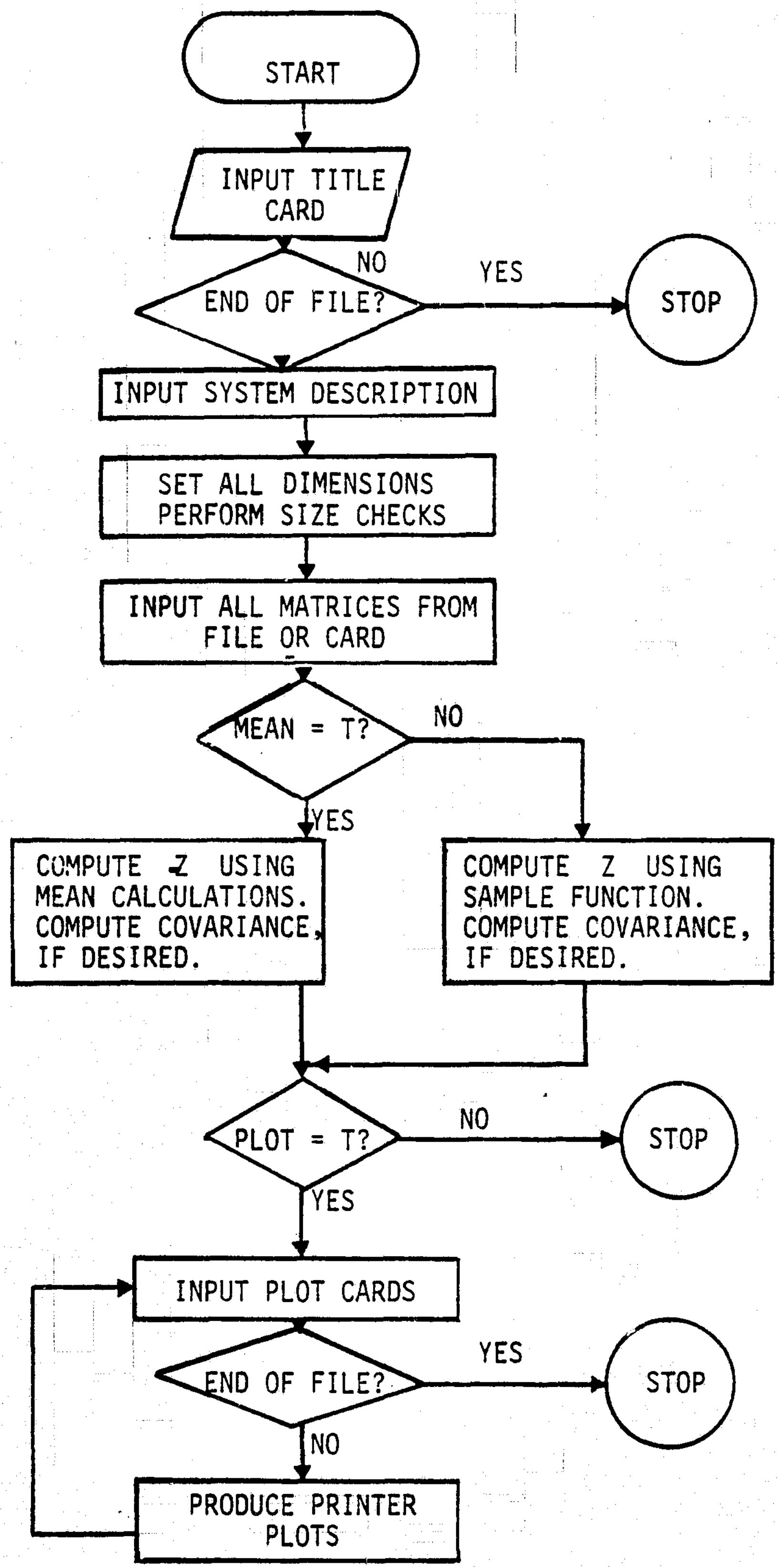


Figure 3.3 TVSR Program Flowchart

3.4 TSS/360 OPERATION

The following control cards are required to execute the TVSR program on the NASA ARC TSS/360 system.

```
 (1)  LØGØN...
 (2)  DDEF FT10F001, VS, F1, DISP=OLD
 (3)  DDEF FT11F001, VS, F2, DISP=OLD
 (4)  DDEF FT12F001, VI, ZØUT, DISP=NEW
 (5)  DDEF FT13F001, VI, CØVØUT, DISP=NEW, RET=T
 (6)  AMES IMSL
 (7)  JBLB TVLIB
 (8)  LØAD TVBLØCK
 (9)  CALL TVMAIN
(10)  {data
(11)  LØGØFF
```

The description of the above execution deck, follows:

(1) Usual LØGØN card.

(2) File containing the system at start time.

(3) File containing the system at end time.

(4) File containing Z-vector output.

(5) File for covariance output.

(6) Invoke IMSL library (for random no. generator GGNØF).

(7) Assign library TVLIB.

(8) Load in block data.

(9) Execute program.

(10) Input cards.

(11) Usual LØGØFF.

3.5 SAMPLE EXECUTION

The following represents a sample execution (with description and output listing).

```
SAMPLE RUN OF TIME VARYING PROGRAM
 $OPTION
   TSTART=0.0,
   TSTOP=.5,
   TDELT=.05,
   SAMPLE=T,
   ZCARD=T,
   NPSIZE=1,
   NCOEF=1,
   NPRINT=1,
   NSTORE=1,
   PLOT=T,
   NINT=1,
   DELTAV=0.0,
 &END
PSML MATRIX
1.0
GCONE1 MATRIX
1.0
0.0
GCONE2 MATRIX
1.0
0.0
GCTWO1 MATRIX
0.0
0.0
GCTWO2 MATRIX
0.0
0.0
GCA1 MATRIX
0.0
0.0
GCA2 MATRIX
0.0
0.0
INITIAL Z VECTOR
0.0
0.0
0.0
0.0
0.0
0.0
PLOT INPUT
     1     1     0     1     0     1
     1     3                   IDZ IDHZ
```

Table 3.5

Description of Sample Execution

LINE NUMBER(S)	CORRESPONDING CARD TYPE	DESCRIPTION
1	1	Title card
2-15	2	Option card specifying start time of 0.0, end time of 0.5, delta time of 0.5, sample function, initial Z to be read in from cards, 1 row in the p matrix, 1 coefficient (column) in the p matrix, every time point to be printed, every time point to be stored, printer plots to be done, one interval, expected velocity change is 0.0. (Velocity change (DELTV) is meaningless when no. of intervals (NINT) is 1.)
16	3A	Comment card identifying p matrix
17	3	p matrix
18	10A	Comment card identifying GCONE1 matrix
19-20	10	GCONE1 matrix
21	11A	Comment card identifying GCONE2 matrix
22-23	11	GCONE2 matrix
24	12A	Comment card identifying GCTWO1 matrix
25-26	12	GCTWO1 matrix
27	13A	Comment card identifying GCTWO2 matrix
28-29	13	GCTWO2 matrix
30	14A	Comment card identifying GCA1 matrix
31-32	14	GCA1 matrix
33	15A	Comment card identifying GCA2 matrix
34-35	15	GCA2 matrix
36	16A	Comment card identifying initial Z matrix

Table 3.5 (Concluded)

LINE NUMBER(S)	CORRESPONDING CARD TYPE	DESCRIPTION
37-42	16	Initial Z vector
43	17A	Comment card identifying plot input
44	17B	Plot input asking for 1 Z state to be plotted, 1 H*Z output to be plotted, no standard deviation to be plotted, each curve to be plotted on its own scale, all output to be plotted, every point to be plotted
45	17C	Z state 1 is to be plotted, H*Z output 3 is to be plotted, description of Z(1) is IDZ, description of H*Z(3) is IDHZ

```
TITLE = SAMPLE RUN --- TIME VARYING PROGRAM
MEAN CALCULATION FLAG = F
SAMPLE CALCULATION FLAG = T
COMPUTE COVARIANCE FLAG = F
INPUT W1,W2 FROM CARDS FLAG = F
INPUT VY1,VY2 FROM CARDS FLAG =F
INPUT INITIAL Z FROM CARDS FLAG = T
INPUT VA1,VA2 FROM CARDS FLAG = F
PLOT FLAG = T
SET H TO IDENTITY FLAG = F
UNIT FOR FC1, GC1 = 10
UNIT FOR FC2, GC2 = 11
UNIT FOR INITIAL Z (IF DISK) AND/OR OUTPUT Z    = 12
UNIT NO. FOR COVARIANCE OUTPUT = 13
MAX. NO. RECORDS ALLOWED ON Z I/O FILE =  2000
MAX. NO. RECORDS ALLOWED ON COVARIANCE FILE =  2000
START TIME =         0.00
STOP  TIME =         0.50
DELTA TIME =         0.05
PRINT INCREMENT =  1
INCREMENT FOR SAVING Z =  1
NO. OF INTERVALS FROM FC1 TO FC2 =  1
NO. OF COEFFICIENTS IN THETA =  1
NO. OF TERMS TO COMPUTE PHI,PSI = 10
SEED FOR RANDOM NO. GENERATOR =       328765
FORMAT FOR READING MATRICES = (8(E10.4) )
RECORD NO. OF INITIAL Z MATRIX, IF KNOWN (FOR ZCARD=F)=    0
TOTAL VELOCITY CHANGE (DELTAV) =      0.0000
INDEX OF H*Z (Y) TO USE IN INTERPOLATION =    1
NO. OF STATES (PLANT) =  2
NO. OF NOISE (PLANT) =  1
NO. OF STATES (HUMAN) =  2
NO. OF MEASUREMENTS (HUMAN) =  1
NO. OF STATES (CLOSED) =  6
NO. OF INPUTS (CLOSED) =  3
NO. OF MEASUREMENTS (CLOSED) =  7
NO. OF AUGMENTED STATES =  2
NO. OF AUGMENTED CONTROLS =  1
NO. OF ROWS IN PSML =  1
```

SAMPLE RUN --- TIME VARYING PROGRAM

PSML

	1
1	1.0000D 00

FC1

	1	2	3	4	5	6
1	-2.0000D 00	0.0000	0.0000	0.0000	0.0000	0.0000
2	1.0000D 00	1.0000D-05	1.3620D 00	6.9443D 00	1.8000D 01	1.8000D 00
3	0.0000	9.0636D 01	-2.0000D 00	-9.0636D 01	0.0000	0.0000
4	0.0000	1.3464D 01	2.3620D 00	-4.5194D 00	1.8000D 01	1.8000D 00
5	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000D 00
6	0.0000	0.0000	-3.0710D 01	-1.9876D 02	-2.6667D 02	-2.6667D 01

FC2

	1	2	3	4	5	6
1	-2.0000D 00	0.0000	0.0000	0.0000	0.0000	0.0000
2	1.0000D 00	1.0000D-05	1.3820D 00	6.9443D 00	1.8000D 01	1.8000D 00

ORIGINAL PAGE IS
OF POOR QUALITY

4	0.0000	1.3464D 01	2.3820D 00	-4.5194D 00	1.8000D 01	1.8000D 00
5	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000D 00
6	0.0000	0.0000	-3.0710D 01	-1.9876D 02	-2.6667D 02	-2.6667D 01

GC1

	1	2	3
1	1.0000D 00	0.0000	0.0000
2	0.0000	0.0000	1.0000D 00
3	0.0000	9.0636D 01	0.0000
4	0.0000	1.3464D 01	1.0000D 00
5	0.0000	0.0000	0.0000
6	0.0000	0.0000	0.0000

GC2

	1	2	3
1	1.0000D 00	0.0000	0.0000
2	0.0000	0.0000	1.0000D 00
3	0.0000	9.0636D 01	0.0000
4	0.0000	1.3464D 01	1.0000D 00
5	0.0000	0.0000	0.0000
6	0.0000	0.0000	0.0000

SAMPLE RUN --- TIME VARYING PROGRAM

HC1

	1	2	3	4	5	6
1	0.0000	0.0000	1.3820D 00	8.9443D 00	1.8000D 01	1.8000D 00
2	1.0000D 00	0.0000	0.0000	0.0000	0.0000	0.0000
3	0.0000	1.0000D 00	0.0000	0.0000	0.0000	0.0000
4	0.0000	0.0000	1.0000D 00	0.0000	0.0000	0.0000
5	0.0000	0.0000	0.0000	1.0000D 00	0.0000	0.0000
6	0.0000	0.0000	0.0000	0.0000	1.0000D 00	0.0000
7	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000D 00

HC2

	1	2	3	4	5	6
1	0.0000	0.0000	1.3820D 00	8.9443D 00	1.8000D 01	1.8000D 00
2	1.0000D 00	0.0000	0.0000	0.0000	0.0000	0.0000
3	0.0000	1.0000D 00	0.0000	0.0000	0.0000	0.0000
4	0.0000	0.0000	1.0000D 00	0.0000	0.0000	0.0000
5	0.0000	0.0000	0.0000	1.0000D 00	0.0000	0.0000
6	0.0000	0.0000	0.0000	0.0000	1.0000D 00	0.0000
7	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000D 00

W1

	1
1	1.7000D 01

W2

	1
1	1.7000D 01

VY1

	1
1	1.2300D-03

VY2

```
VA1
                  1
      1     1.0480D-01
SAMPLE RUN --- TIME VARYING PROGRAM


VA2
                  1
      1     1.0480D-01


GCONE1
                  1
      1     1.0000D 00
      2         0.0000


GCONE2
                  1
      1     1.0000D 00
      2         0.0000


GCTWO1
                  1
      1         0.0000
      2         0.0000


GCTWO2
                  1
      1         0.0000
      2         0.0000


GCA1
                  1
      1         0.0000
      2         0.0000


GCA2
                  1
      1         0.0000
      2         0.0000
SAMPLE RUN --- TIME VARYING PROGRAM


INITIAL Z
                  1
      1         0.0000
      2         0.0000
      3         0.0000
      4         0.0000
      5         0.0000
      6         0.0000


AUGMENTED GC1
                  1              2             3          4
      1     1.0000D 00      0.0000        0.0000   1.0000D 00
```

3	0.0000	9.0636D 01	0.0000	0.0000
4	0.0000	1.3464D 01	1.0000D 00	0.0000
5	0.0000	0.0000	0.0000	0.0000
6	0.0000	0.0000	0.0000	0.0000

AUGMENTED GC2

	1	2	3	4
1	1.0000D 00	0.0000	0.0000	1.0000D 00
2	0.0000	0.0000	1.0000D 00	0.0000
3	0.0000	9.0636D 01	0.0000	0.0000
4	0.0000	1.3464D 01	1.0000D 00	0.0000
5	0.0000	0.0000	0.0000	0.0000
6	0.0000	0.0000	0.0000	0.0000

```
  2  0.000 THETA        0.00000    0.00000    0.00000    0.00000
           H*Z          0.00000    0.00000    0.00000    0.00000    0.00000    0.00000    0.00000
```

SAMPLE RUN --- TIME VARYING PROGRAM

INTERVAL NO. 1
NO. OF STEPS TO BE TAKEN = 10
STARTING TIME OF THIS INTERVAL = 0.000
END TIME OF THIS INTERVAL = 0.500
INTERPOLATION CONSTANT = 0.000
INITIAL VELOCITY = 0.000
CURRENT VELOCITY = 0.000

FC

	1	2	3	4	5	6
1	-2.0000D 00	0.0000	0.0000	0.0000	0.0000	0.0000
2	1.0000D 00	1.0000D-05	1.3820D 00	8.9443D 00	1.8000D 01	1.8000D 00
3	0.0000	9.0636D 01	-2.0000D 00	-9.0636D 01	0.0000	0.0000
4	0.0000	1.3464D 01	2.3820D 00	-4.5194D 00	1.8000D 01	1.8000D 00
5	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000D 00
6	0.0000	0.0000	-3.0710D 01	-1.9876D 02	-2.6667D 02	-2.6667D 01

GC

	1	2	3	4
1	1.0000D 00	0.0000	0.0000	1.0000D 00
2	0.0000	0.0000	1.0000D 00	0.0000
3	0.0000	9.0636D 01	0.0000	0.0000
4	0.0000	1.3464D 01	1.0000D 00	0.0000
5	0.0000	0.0000	0.0000	0.0000
6	0.0000	0.0000	0.0000	0.0000

HC

	1	2	3	4	5	6
1	0.0000	0.0000	1.3820D 00	8.9443D 00	1.8000D 01	1.8000D 00
2	1.0000D 00	0.0000	0.0000	0.0000	0.0000	0.0000
3	0.0000	1.0000D 00	0.0000	0.0000	0.0000	0.0000
4	0.0000	0.0000	1.0000D 00	0.0000	0.0000	0.0000
5	0.0000	0.0000	0.0000	1.0000D 00	0.0000	0.0000
6	0.0000	0.0000	0.0000	0.0000	1.0000D 00	0.0000
7	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000D 00

WC

	1
1	1.7000D 01

SAMPLE RUN --- TIME VARYING PROGRAM

```
                 1
      1      1.2300D-03

VAC
                 1
      1      1.0460D-01

PHI
                 1             2             3             4             5             6
      1      9.0484D-01      0.0000        0.0000        0.0000        0.0000        0.0000
      2      5.0233D-02    1.1227D 00    2.0560D-02    3.1899D-03    5.0290D-01    6.6100D-02
      3      8.3903D-02    3.0052D 00    8.2093D-01   -3.0052D 00   -2.2204D-16   -1.3878D-17
      4      1.7076D-02    6.8184D-01    5.3717D-02    4.4403D-01    5.0290D-01    6.6100D-02
      5     -1.0159D-03   -7.5932D-02   -2.8121D-02   -9.8645D-02    7.4649D-01    2.0083D-02
      6     -7.3901D-02   -3.8769D 00   -9.0041D-01   -1.6762D 00   -8.4979D 00   -1.0329D-01

PSI
                 1             2             3             4             5             6
      1      4.7581D-02      0.0000        0.0000        0.0000        0.0000        0.0000
      2      1.2480D-03    5.2729D-02    9.1616D-04    2.9444D-03    1.5810D-02    1.8720D-03
      3      1.5121D-03    8.6927D-02    4.6069D-02   -8.6927D-02      0.0000        0.0000
      4      2.8896D-04    1.7654D-02    1.8752D-03    3.8019D-02    1.5810D-02    1.8720D-03
      5     -1.0792D-05   -1.0375D-03   -5.1450D-04   -2.1762D-03    4.5433D-02    7.3372D-04
      6     -1.0159D-03   -7.5932D-02   -2.8121D-02   -9.8645D-02   -2.5351D-01    2.0083D-02
SAMPLE RUN --- TIME VARYING PROGRAM

   3  0.050 THETA    0.75910E 00-.71375E-010.68469E-010.10000E 01
              H*Z    -.12279E 000.83700E-01-.27490E-02-.21183E 00-.44346E-010.51806E-020.26297E 00
   4  0.100 THETA    -.50435E 01-.26986E-01-.38545E 000.10000E 01
              H*Z    0.43625E 00-.11666E 00-.13207E-01-.12908E 00-.52555E-010.22935E-010.37438E 00
   5  0.150 THETA    0.12948E 010.60853E-010.38505E 000.10000E 01
              H*Z    0.80679E 000.36300E-020.44538E-010.18583E 000.58606E-010.28692E-01-.27490E 00
   6  0.200 THETA    -.20452E 01-.54380E-010.46275E 000.10000E 01
              H*Z    -.48613E 00-.46448E-010.68299E-01-.51951E-010.51237E-010.40742E-02-.52553E 00
   7  0.250 THETA    -.19052E 01-.44962E-01-.35645E 000.10000E 01
              H*Z    -.60354E 00-.85097E-010.25396E-01-.13176E 00-.65815E-02-.12318E-01-.78255E-01
   8  0.300 THETA    0.49293E 01-.64308E-020.40436E 000.10000E 01
              H*Z    0.10047E 000.20513E 000.39263E-01-.29569E-010.14337E-01-.91299E-020.96577E-01
   9  0.350 THETA    0.15489E 01-.99591E-020.24605E 000.10000E 01
              H*Z    0.11299E 000.30669E 000.71404E-010.41770E-010.44621E-01-.66702E-02-.10432E 00
  10  0.400 THETA    0.39468E 010.46170E-010.54610E 000.10000E 01
              H*Z    -.41888E-020.51306E 000.12757E 000.28676E 000.12804E 00-.25191E-01-.60682E 00
  11  0.450 THETA    -.64893E 010.22516E-010.31837E 000.10000E 01
              H*Z    -.94363E 000.20305E 000.13615E 000.33639E 000.14671E 00-.64567E-01-.66584E 00
  12  0.500 THETA    0.17870E 01-.66722E-010.46781E 000.10000E 01
              H*Z    -.15138E 010.31634E 000.10207E 000.65161E-010.71166E-01-.96531E-01-.27936E 00
PLOT NO.   1
NO. OF VARIABLES TO BE PLOTTED (Z) =   1
NO. OF VARIABLES TO BE PLOTTED (H*Z) = 1
STANDARD DEVIATION PLOT FLAG =   0
SCALE FLAG =   1
ENTIRE FILE PLOT FLAG =   0
PLOT INCREMENT =   1
Z STATES TO BE PLOTTED =       1
H*Z STATES TO BE PLOTTED =       3
      0.000      0.000000      0.000000
      0.0500.837004E-01-.274900E-02
      0.100-.116659E 00-.132070E-01
      0.1500.363004E-020.445379E-01
      0.200-.464481E-010.682991E-01
      0.250-.850973E-010.253957E-01
      0.3000.205126E 000.392630E-01
```

```
  0.4000.513060E 000.127566E 00
  0.4500.203051E 000.136150E 00
  0.5000.316336E 000.162071E 00

 A STATE      -3.00E-01 -2.00E-01 -1.00E-01      0.00  1.00E-01  2.00E-01  3.00E-01  4.00E-01  5.00E-01  6.00E-01  7.00E-01
   IDZ         +---------+---------+---------+---------+---------+---------+---------+---------+---------+---------+
 B STATE        -2.00E-02       0.00  2.00E-02  4.00E-02  6.00E-02  8.00E-02  1.00E-01  1.20E-01  1.40E-01  1.60E-01
   IDHZ        ---------+---------+---------+---------+---------+---------+---------+---------+---------+---------+-
   TIME

      0.00000  .                  B          A                                                                        .
      0.05000  .                  B          .      A                                                                 .
      0.10000  .            B     A                                                                                   .
      0.15000  .                  .          A         B                                                              .
      0.20000  .                  .     A    .                   B                                                    .
      0.25000  .                  .  A       . B                                                                      .
      0.30000  .                  .          .       B          A                                                     .
      0.35000  .                  .          .                      B     A                                           .
      0.40000  .                  .          .                                              A B                       .
      0.45000  .                  .          .                 A                                B                     .
      0.50000  .                  .          .                            A       B                                   .
```

3.6 MAIN PROGRAM LISTING

The following is a computer listing for the program.

```
C
C  PROGRAM TO SOLVE
C        ZDOT = FC*Z + GC*THETA
C
C  THE SOLUTION IS GIVEN BY
C        Z(K+1) = PHI*Z(K) + PSI*GC*THETA
C    WHERE  PHI = STATE TRANSITION MATRIX
C           PSI = INPUT TRANSITION MATRIX
C
C                 0
C         THETA = 0      , FOR MEAN CALCULATIONS
C                PSML
C
C                 W      , FOR SAMPLE FUNCTION WHITE NOISE SEQUENCE
C         THETA = VY       GENERATED FROM N(0,W),N(0,VY) & N(0,VA)
C                VA
C                PSML
C
C
C  INPUT--     ( * CARDS ARE CONDITIONAL )
C    CARD NO. FORMAT    DESCRIPTION
C
C       1      13A6      TITLE = TITLE TO BE PRINTED AT TOP OF PAGE
C
C       2                NAMELIST /OPTION/
C                        MEAN = T, FOR MEAN CALCULATIONS
C                             = F, DO NOT COMPUTE MEAN
C                               ( DEFAULT=F )
C                        SAMPLE = T, TO COMPUTE THE SAMPLE FUNCTION
C                               = F, DO NOT COMPUTE
C                                 ( DEFAULT = F )
C                        PLOT = T, TO PLOT ON PRINTER
C                             = F, DO NOT PLOT
C                               ( DEFAULT = F )
C                        WCARD = T , TO READ IN W1,W2 FROM CARDS
C                              = F, USE W1,W2 MATRICES STORED ON DISK
C                               ( DEFAULT = F)
C                        VYCARD = T, TO READ VY1,VY2 FROM CARDS
C                               = F, TO USE VY1, VY2 STORED ON DISK
C                                 ( DEFAULT=F )
C                        ZCARD  = T, IF INITIAL Z VECTOR IS CARD INPUT
C                               = F, IF INITIAL Z VECTOR IS ON DISK
C                                 ( DEFAULT=F )
C                        VACARD = T, TO READ VA1,VA2 FROM CARDS
C                               = F, USE VA1,VA2 STORED ON DISK
C                                 ( DEFAULT = F )
C                        NPSIZE = NO. OF ROWS IN PSML ( DEFAULT=0 )
C                        COVAR  = T, TO COMPUTE COVARIANCE AND SAVE
C                               = F, DO NOT COMPUTE COVARIANCE
C                                 (DEFAULT=F )
C                        IUFC1  = UNIT NO. OF FC1, GC1, W1, VY1, P1, HC1
C                                 ( DEFAULT=10 )
C                        IUFC2  = UNIT NO. FOR FC2, GC2, W2, VY2, P2, HC2
C                                 ( DEFAULT=11 )
C                        IUZ    = UNIT NO. FOR INPUT/OUTPUT Z VECTORS
C                               ( DEFAULT=12 )
C                        IUCOV = UNIT NO. FOR COVARIANCE OUTPUT, IF ANY
                              ( [illegible] )
C                        LASTZ = RECORD NO. OF INITIAL Z TO BE USED ON
C                                FILE. THIS ONLY NEEDS TO BE SET IF
```

```
C                               = 0, WILL USE LAST Z WRITTEN OR WILL START
C                                 A FILE IF ZCARD=TRUE
C                               ( DEFAULT=0 )
C                     NPRINT = INCREMENT FOR PRINTING
C                               ( DEFAULT=0, ONLY LAST Z PRINTED)
C                     NSTORE = INCREMENT FOR STORING Z VECTOR,
C                               EVERY NSTORE VECTOR WILL BE SAVED
C                               ( DEFAULT=0, NOTHING IS SAVED)
C                     NINT = NO. OF INTERVALS BETWEEN TSTART, TSTOP
C                             (FOR INTERPOLATION BETWEEN FC'S ET. AL.)
C                             ( DEFAULT=1 )
C                     NTERMS = NO. OF TERMS TO USE IN COMPUTING
C                               PHI, PSI
C                               ( DEFAULT = 10 )
C                     TSTART = START TIME
C                               ( DEFAULT = 0.0)
C                     TSTOP  = END TIME
C                               ( DEFAULT=0.0)
C                     TDELT  = DELTA TIME BETWEEN TSTART,TSTOP
C                               ( DEFAULT=1.0)
C                     NCOEF  = NO. OF COEFFICIENTS IN PSML MATRIX
C                               ( COL. DIMENSION OF PSML )
C                               ( DEFAULT=3 )
C                     ISEED  = SEED FOR RANDOM NO. GENERATOR
C                               ( DEFAULT=328765 )
C                     IFMT  = FORMAT FOR CARD MATRIX INPUT
C                              ( DEFAULT = 8E10.5 )
C                     HIDENT = T, SET H TO IDENTITY MATRIX
C                            = F, USE ORIGIINAL H
C                               ( DEFAULT=F )
C                     DELTAV = DESIRED TOTAL CHANGE IN VELOCITY
C                     INDEXV = LOCATION OF VELOCITY IN OUTPUT VECTOR
C                                H*Z  (DEFAULT=1)
C
C       3*     13A6       ICMMT = COMMENT CARD TO IDENTIFY PSML MATRIX
C                                 FOR USER'S PURPOSE ONLY
C                                 INCLUDE ONLY IF NPSIZE .GT. 0
C
C       4-4N* IFMT        PSML MATRIX, READ IN BY ROWS
C                         NO. OF ROWS OF PSML = NPSIZE
C                         INCLUDE ONLY IF NPSIZE .GT. 0
C
C       5A*    13A6       ICMMT = COMMENT CARD FOR USER IDENTIFICATION
C                                 OF W1 MATRIX INPUT.
C                                 INCLUDE ONLY IF (SAMPLE=T OR COVAR=T)
C                         INCLUDE ONLY IF MEAN = FALSE OR COVAR=TRUE
C                                       AND WCARD=T
C       5-5N* IFMT        W1 MATRIX
C
C       6A*    13A6       ICMMT = SAME AS 5A* EXCEPT FOR W2
C       6-6N* IFMT        SAME AS 5* EXCEPT FOR W2
C
C       7A*    13A6       ICMMT = SAME AS 5A* EXCEPT FOR VY1
C       7-7N* IFMT        SAME AS 5* EXCEPT FOR VY1
C
C       8A*    13A6       ICMMT = SAME AS 5A* EXCEPT FOR VY2
C       8-8N* IFMT        SAME AS 5* EXCEPT FOR VY2
C       9A*    13A6      SAME A 5A* EXCEPT FOR VA1
C       9-9N* IFMT       SAME AS 5* EXCEPT FOR VA1
C
C       10A*   13A6      SAME AS 5A* EXCEPT FOR VA2
C       10-10N*          SAME AS 5* EXCEPT FOR VA2
C
C       11A*   13A6      COMMENT CARD FOR START OF GCONE1
C     11-11N*  IFMT      GCONE1 MATRIX
```

```
C       12A*   13A6       COMMENT CARD FOR START OF GCONE2
C    12-12N*   IFMT       GCONE2 MATRIX
C
C       13A*   13A6       COMMENT CARD FOR START OF GCTWO1
C    13-13N*   IFMT       GCTWO1 MATRIX
C
C       14A*   13A6       COMMENT CARD FOR START OF GCTWO2
C    14-14N*   IFMT       GCTWO2 MATRIX
C
C       15A*   13A6       COMMENT CARD FOR START OF GCA1
C    15-15N*   IFMT       GCA1 MATRIX
C
C       16A*   13A6       COMMENT CARD FOR START OF GCA2
C    16-16N*   IFMT       GCA2 MATRIX
C
C       17A*   13A6        ICMMT = COMMENT CARD FOR USER IDENTIFICATION
C                          OF INITIAL Z VECTOR. INCLUDE ONLY IF ZCARD=TRUE.
C       17-17N*            AS MANY CARDS AS NEEDED FOR Z VECTOR INITIAL
C                          VALUE.  INCLUDE ONLY IF ZCARD=TRUE.
C
C         18A* 13A6        COMMENT CARD FOR IDENTIFICATION OF PLOT INPUT
C                          TOTAL NO. OF VARIABLES THAT MAY BE SPECIFIED
C                          MUST BE .LE. 5
C         18B*  16I5       NVARZ = NO. OF Z STATES TO PLOT
C                          NVARY = NO. OF H*Z VARIABLES TO PLOT
C                                 (DO NOT INCLUDE STD. DEV. CURVES IN NO.)
C                          ISD = 0, DO NOT PLOT STD. DEV.
C                              = 1, PLOT STD. DEV. ASSOCIATED WITH EACH
C                                VARIABLE.
C                          ISCALE = 0, USE SAME SCALE FOR ALL CURVES
C                                 = 1, USE DIFFERENT SCALE FOR EACH CURVE
C                          KREC = 0, PLOT ALL OUTPUT
C                               = 1, PLOT ONLY THE CURRENT RUN
C                          INC = 1, PLOT EVERY POINT
C                              = N, PLOT EVERY NTH POINT
C         18C (5I5,5A4)   NVARZ(5) = LIST OF Z OR H*Z STATES TO PLOT
C                                     ( COLS. 1-25 )
C                          YID(5) = 4 CHARACTER DESCRIPTION OF EACH
C                                   VARIABLE PLOTTED (COLS. 26-45)
C
C GENERAL NOTES--
C   (1) ALL MATRICES ARE READ IN BY ROWS FOR CARD INPUT
C
C   (2) CARDS 3-17N ARE INCLUDED ONLY IF MEAN=T OR SAMPLE=T
C              3*  THRU  4N* ARE USED ONLY IF NPSIZE .GT. 0
C              5A* THRU  6N* ARE USED ONLY IF SAMPLE=TRUE OR COVAR=TRUE,
C                                             AND WCARD=TRUE
C              7A* THRU  8N* ARE USED ONLY IF SAMPLE=TRUE OR COVAR=TRUE,
C                                             AND VYCARD=TRUE
C              9A* THRU 10N* ARE USED ONLY IF COVAR=TRUE, VACARD=TRUE,
C                                               AND NPSIZE .GT. 0
C             11A* THRU 16N* ARE USED ONLY IF NPSIZE .GT. 0
C             17A* THRU 17N* ARE USED ONLY IF ZCARD = TRUE
C             18A* THRU 18C* ARE USED ONLY IF PLOT = TRUE
C
C   (3) UNIT USED--
C        5 - CARD READ
C        6 - PRINT
C       10 - INPUT FILE CONTAINING FC1, GC1, HC1, W1, VY1, VA1
C       11 -      ''        ''        FC2, GC2, HC2, W2, VY2, VA2
C       12 - IN/OUT ''        ''      Z VECTORS
C       13 - OUTPUT FILE FOR COVARIANCE, RANDOM ACCESS
C
C     PARAMETER NS...,MI=4,...4,N...,NM...,NCO...,GCA...
C     PARAMETER NPLT=100
```

```
C     PARAMETER  NZRLC=100, NCOREC=100
C
      IMPLICIT REAL (A-H,O-Z)
      REAL PLT
      LOGICAL MEAN, WCARD, VYCARD, ZCARD, COVAR, VACARD                 -
     .        , SAMPLE, PLOT, HIDENT
C
      COMMON  / MAXDIM / MXNS, MXNI, MXNM, MXNNP, MXNMH, MXNCO, MXNGCA  -
     .                 , MXPLT, MXZ, MXCOV
      COMMON  / LOGICS / MEAN, COVAR, WCARD, VYCARD, ZCARD, VACARD      -
     .                 , SAMPLE, PLOT, HIDENT
      COMMON  / UNITS  / IUFC1, IUFC2, IUZ, IUCOV
C
C     DIMENSION  FC1( NS,NS ), FC2( NS,NS ), FCC( NS,NS )               -
C    .          , GC1( NS,NI ), GC2( NS,NI ), GCC( NS,NI )              -
C    .          , W1 (NNP,NNP), W2 (NNP,NNP), WC (NNP,NNP)              -
C    .          , VY1(NMH,NMH), VY2(NMH,NMH), VYC(NMH,NMH)              -
C    .          , HC1( NM,NS ), HC2( NM,NS ), HCC( NM,NS )              -
C    .          , P1( NGCA, NGCA ), P2( NGCA, NGCA ), PC( NGCA, NGCA )  -
C    .          , PHI( NS,NS ), PS1( NS,NS ), THETA( NI,2)              -
C    .          , Z( NS,2)    , PSML( NGCA, NCO )                       -
C    .          , ZCOV( NS,NS ), YCOV( NM,NM ), PSIGC( NS,NI )          -
C    .          , PLT(NPLT,11)                                          -
C    .          , WORK( WRK )
C
      DIMENSION  FC1( 35,35 ), FC2( 35,35 ), FCC( 35,35 )   -
     .          , GC1( 35,19), GC2( 35,19 ), GCC( 35,19 )  -
     .          , W1 ( 4, 4 ), W2 ( 4, 4 ), WC ( 4, 4 )  -
     .          , VY1 (10,10 ), VY2(10,10 ), VYC( 10,10 )  -
     .          , HC1(35,35 ), HC2 (35,35 ), HCC( 35,35 )  -
     .          , PHI( 35,35 ), PS1 (35,35), THETA( 19,2 )  -
     .          , Z( 35,2 ), PSML ( 5,5 )  -
     .          , ZCOV ( 35,35 ), YCOV ( 35,35 ), PSIGC ( 35,19 )  -
     .          , PLT ( 2000, 11 )  -
     .          , WORK ( 10600 )
      DIMENSION FCCOV( 35,35 ), GCORG1(35,19), GCORG2(35,19)          -
     .         , GCONE1(30,5), GCONE2(30,5)                            -
     .         , GCTWO1(30,5),  GCTWO2(30,5), GCA1(5,5), GCA2(5,5)       -
     .         , VA1(4,4), VA2(4,4), VAC(4,4)
C
      EXTERNAL THETAM, THETAS
C
C  CONSTANTS BASED ON PARAMETERS
C
C     MXNS = NS
C     MXNI = NI
C     MXNM = NM
C     MXNNP = NNP
C     MXNMH = NMH
C     MXNCO=NCO
C     MXNGCA = NGCA
C     MXZ  =NZREC
C     MXCOV = NCOREC
C     MXPLT = NPLT
C
C
      MXNS = 35
      MXNI = 19
      MXNM = 35
      MXNNP = 4
      MXNMH = 10
      MXNCO = 5
      MXNGCA = 5
      MXZ = 2000
      MXCOV = 2000
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C
C  READ THE OPTIONS, SET ALL DIMENSIONS, CHECK SOME VALUES
C
  100 CONTINUE
      CALL INITL
C
C  INPUT ALL MATRICES
C
      IF ( .NOT. MEAN .AND. .NOT. SAMPLE ) GO TO 300
      CALL INPUT ( PSML, FC1, FC2, GCORG1, GCORG2, HC1, HC2              -
     .           , W1, W2, VY1, VY2, Z, VA1, VA2                         -
     .           , GCONE1, GCONE2, GCTWO1, GCTWO2, GCA1, GCA2 )
C
C  FORM THE AUGMENTED GC AND HC MATRICES
C
      CALL FORMGC ( GCORG1, GCONE1, GCTWO1, GCA1, GC1 )
      CALL FORMGC ( GCORG2, GCONE2, GCTWO2, GCA2, GC2 )
C
      IF ( .NOT. MEAN ) GO TO 200
C
C  COMPUTE Z FOR MEAN CALCULATIONS
C
      CALL MINTEG ( FC1, FC2, FCC, GC1, GC2, GCC, HC1, HC2, HCC          -
     .            , W1, W2, WC, VY1, VY2, VYC, VA1, VA2, VAC             -
     .            , Z, PSML, PHI, PSI, PSIGC, THETA, ZCOV, YCOV, WORK   -
     .            , FCCOV, THETAM  )
      GO TO 300
C
C  COMPUTE Z FOR SAMPLE FUNCTION
C
  200 CONTINUE
      CALL MINTEG ( FC1, FC2, FCC, GC1, GC2, GCC, HC1, HC2, HCC          -
     .            , W1, W2, WC, VY1, VY2, VYC, VA1, VA2, VAC             -
     .            , Z, PSML, PHI, PSI, PSIGC, THETA, ZCOV, YCOV, WORK   -
     .            , FCCOV, THETAS )
C
C  PLOTTING
C
  300 CONTINUE
      IF ( .NOT. PLOT ) GO TO 999
      CALL  MPLOT ( PLT, Z, WORK, ZCOV, YCOV )
C
C
  999 CONTINUE
      STOP
C
      END
```

IV. OPTIMAL SIMULATION PROGRAM (TVOPT)

The time varying optimal simulation program (TVOPT), as previously discussed, utilizes an exact closed loop simulation. That is, the time delay T, in the perceived display and Kalman filter predictor equations, is explicitly accounted for. (The program TVSR relies on a lumped linear approximation to this delay time.) The program TVOPT is more restrictive than TVOPT because it does not compute covariance terms but is limited to sample time histories.

4.1 DESCRIPTION

The system to be simulated is given by:

plant

$$\dot{x}(t) = Fx(t) + Gu(t) + \Gamma w(t)$$

$$y(t) = Hx(t) + v_y(t)$$

$$y_p(t) = y(t-T)$$

estimator

$$\dot{\hat{x}}(t-T|t-T) = F\hat{x}(t-T|t-T) + Gu(t-T) + K[y_p(t) - H\hat{x}(t-T|t-T)]$$

predictor

$$\dot{\hat{x}}(t|t-T) = F\hat{x}(t|t-T) + Gu(t) + e^{FT}K[y_pH) - Hx(t-T|t-T)]$$

control

$$u(t) = -\lambda\hat{x}(t|t-T)$$

Defining $\gamma(t) = \hat{x}(t|t-T)$ and $p(t+T) = q(t) = \hat{x}(t|t)$, then

$$\dot{z}(t) = F_c\, z(t) + G_c\, \theta(t)$$

where

$$z^T(t) = [x^T(t),\ \gamma^T(t),\ q^T(t)]$$

$$F_c = \begin{bmatrix} F & -G\lambda & 0 \\ 0 & F-G\lambda & 0 \\ KH & -G\lambda & F-KH \end{bmatrix}$$

$$G_c = \begin{bmatrix} G & \Gamma & 0 & 0 \\ G & 0 & 0 & e^{FT}K \\ G & 0 & K & 0 \end{bmatrix}$$

$$\theta^T(t) = [v_m{}^T(t),\ w^T(t),\ v_y{}^T(t),\ b^T(t-T)]$$

$$b(t) = v_y(t) + H[xH) - q(t)]$$

The displayed variable (input to the operator) is given by $y_p(t-T)$ where

$$y_p(t) = [H \quad 0 \quad 0]z(t) + v_y(t)$$

and the operator's output, $u_m(t)$, is given by

$$u_m(t) = [0 \quad -\lambda \quad 0]z(t) + v_m(t)$$

It is these last two equations which are computed as written to an output disc file ($y_p(t-T)$ and $u_m(t)$).

The solution for $z(t)$ is obtained from

$$z(k+1) = \phi_1 z(k) + \phi_2 \theta(k)$$

where

$$\phi_1 = \exp F_c \Delta \; ; \; \Delta = \text{integration step size}$$

$$\phi_2 = \int_0^{\Delta} \phi_1 G_c \, dt$$

The time delay, T, is restricted to be integer values of the integration step size, i.e. $T = N\Delta$; N is input. The only user option is mean values of the states. This is implemented in the same way as the sample function case but with the σ-values on all noise sources set to zero. The user must specify the matrices F, G, H, Γ, Q (in the optimal cost), W (= cov w), V_y (= cov v_y), V_m (= cov v_m), g (in the optimal cost), and $x(t = 0)$.

4.2 INPUT SPECIFICATIONS

All input is accomplished through cards. All matrices are read in by rows in format (qE10.4). Table 4.1 describes required and optional card input.

Table 4.1
Card Input Specifications

CARD NO.	FORMAT	DESCRIPTION
1	(13A4)	ITITLE(13) = title to be printed at the top of each output page
2	NAMELIST /TVOPT/	MEAN=T, to compute mean calculations (Default=F) SAMPLE=T, to compute sample function (Default=F) STAT=T, to compute mean and standard deviation for V_m, W, V_y, u, y and Z(1-NS). (Default=F) PLØT=T, to get printer plots of z, u, and/or y (Default=F) DELTA = delta time (Default=1.0) N = time delay (Default=1) NPTS = total no. of points to compute (Default=10) ISAVE = increment for saving points on units 10 and 11 (Default=1) NS = no. of states (no default) NC = no. of controls (no default) NØ = no. of outputs (no default) NM = no. of measurements (no default) NTERMS = no. of terms in expansion series in computing ϕ and ψ (routine DISC) (Default=10) ISEED = seed for random no. generator (Default= 1487621)
3-3N	(8E10.4)	F matrix (NS x NS)
4-4N	(8E10.4)	G matrix (NS x NC)
5-5N	(8E10.4)	H matrix (NM x NS)
6-6N	(8E10.4)	Γ matrix (NS x NØ)

ORIGINAL PAGE IS
OF POOR QUALITY

Table 4.1 (Continued)

CARD NO.	FORMAT	DESCRIPTION
7-7N	(8E10.4)	Q matrix (NS x NS)
8-8N	(8E10.4)	W matrix (NØ x NØ)
9-9N	(8E10.4)	V_y matrix (NM x NM)
10-10N	(8E10.4)	V_m matrix (NC x NC)
11-11N	(8E10.4)	g matrix (NC x NC)
12-12N	(8E10.4)	XO matrix (NS x 1)
13*	NAMELIST /PLOTS/	NZPLØT = no. of z variables to plot (max. of 5) (Default=0) NUPLØT = no. of u variables to plot (max. of 5) (Default=0) NYPLØT = no. of y variables to plot (max. of 5) (Default=0) NZ(1-5) = list of z variables to plot NU(1-5) = list of u variables to plot NY(1-5) = list of y variables to plot INC = plot increment (this will plot every n^{th} point stored on unit 10 and/or unit 11. I.e., if ISAVE=2 and INC=2, every 4^{th} point computed will be plotted.) (Default=1) INDSCZ=T, to independently scale each z variable (Default=F) INDSCU=T, to independently scale each u variable (Default=F) INDSCY=T, to independently scale each y variable (Default=F) PRINT=T, to echo the data to be plotted (Default=F)

Table 4.1 (Concluded)

CARD NO.	FORMAT	DESCRIPTION
14-14N	4A4	YLAB(4, 1-NZPLOT+NVPLOT+NYPLOT) = label of up to 16 characters. One card per variable in order of z variables, u variables and y variables.

Cards 13—14* are only required when PLOT=T. As many sets of cards 13—14 as desired may be stacked.

4.3 OUTPUT FILES

The program uses 2 output files. Unit 10 contains z-vector output and unit 11 contains u-vector and y-vector output. Tables 4.2 and 4.3 describe these files.

Table 4.2
Z-Vector Output (Unit 10)

RECORD NO.	DESCRIPTION**
1	TIME, z(1), z(2), ..., z(NSC)***
2	⋮
⋮	
NSAVED*	

*NSAVED = NPTS/ISAVE
**All output variables are double precision
***NSC = 3** no. of states

Table 4.3
U-Vector and Y-Vector Output (Unit 11)

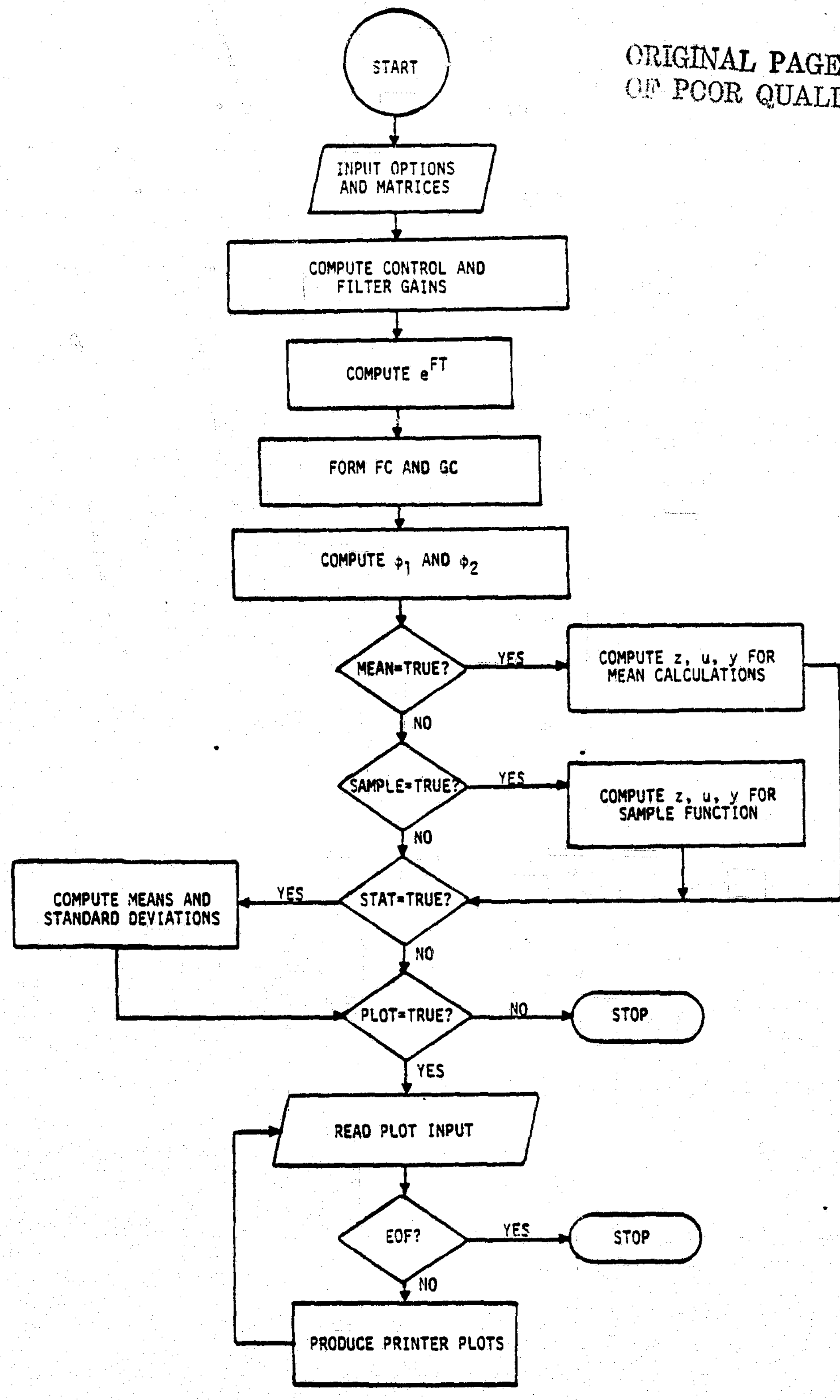
RECORD NO.	DESCRIPTION**
1	TIME, u(1), u(2), ..., u(NC),*** y(1), y(2), ..., y(NM)****
2	⋮
⋮	
NSAVED*	

*NSAVED = NPTS/ISAVE
**All output variables are double precision
***NC = no. of controls
****NM = no. of measurements

4.4 PROGRAM FLOWCHART OF TVOPT

The program flowchart for the TVOPT program is shown in Figure 4.1.

ORIGINAL PAGE IS
OF POOR QUALITY



4.5 TSS/360 OPERATION

The following control cards are required to execute the TVOPT program on the NASA ARC TSS/360 system.

(1) LØGØN...

(2) DDEF FT10F001, VS, ZØUT, DISP=NEW, RET=T

(3) DDEF FT11F001, VS, UYØUT, DISP=NEW

(4) AMES IMSL

(5) JBLB TVØPT.LIB

(6) CALL MAIN$$

(7) {input data

(8) LØGØFF

The description of the above execution deck follows:

(1) Usual LØGØN card.

(2) Data definition card for unit 10, z-vector output. The DISP=NEW parameter indicates the file named ZØUT does not exist. The RET=T parameter (RETAIN=TEMPORARY) means the file will automatically be erased at the end of the job.

(3) Data definition card for unit 11, u-vector and y-vector output. Since the RET=T parameter does not appear on this card, the file UYØUT will be permanent. The user will have to use the ERASE command in a later run to delete it.

(4) Invokes the IMSL library. Routine GGNØF (random no. generator) is used from IMSL for the sample function.

(5) Specifies that the library TVØPT.LIB should be searched for all routines first.

(6) Execute the program.

(7) Input data.

(8) Usual LØGØFF card.

4.6 SAMPLE EXECUTION

The following represents a sample execution (with description and output listing).

```
TP FOR TD AND TV
 &OPTION
  PLANT=T,
  HUMAN=T,
  CLOSED=T,
  SAVSYS=T,
  COVAR=T,
 &END
INPUT FOR PLANT DYNAMICS
     2     1     1     1  0     0     0     0
-2.0        0.0
1.0             1.0E-05
0.0
1.0
0.0         1.0
0.0
1.0
0.0
INPUT FOR HUMAN DYNAMICS
     2     1     1     1     1     0     0
     1     1     3    -1     1     0
     1  1.0
     1  1.0
-2.0        0.0
1.0             1.0E-05
0.0
1.0
0.0         1.0
0.0
1.0
0.0
0.0         0.0
0.0         1.0
AUGMENTED STATES INPUT
     2
0.0         1.0
-266.67     -26.667
0.0
44.444
18.         1.0
-2.0
0.1048
CLOSED LOOP INPUT
     5     1     0     1
     1     1     3    -1     1     0
     1  1.0
     1  1.0
0.0
0.0
1.0
ADDITIONAL HUMAN INPUT
.05
17.
.00123
```

V. IDENTIFICATION PROGRAM (SCIDNT)

The identification algorithm delivered to NASA ARC is a modified version of an aircraft identification code commonly referred to as SCIDNT. The basic modification utilizes the output error method to estimate the parameters associated with the human operator model. Specifically, the parameters can include any combinations of the parameters associated with the optimal cost function as well as the parameters associated with the Kalman filter/predictor (e.g. the process noise/measurement noise covariances as well as the prediction time).

5.1 DESCRIPTION

The identification algorithm assumes a structure for the human operator model of the form:

$$\dot{\hat{x}}(t|t-T) = F^* \hat{x}(t|t-T) + G^* y_a(t-T)$$

$$u(t) = -\lambda \hat{x}(t \; t-T) + v_m(t) \qquad (5.1)$$

where

$$F^* = (I + e^{FT} KHT)(F - G\lambda - e^{FT} KH)$$

$$G^* = (I + e^{FT} KHT)e^{FT} K$$

The input to the system is $y_a(t-T)$, the displayed variable to the operator, and the output is $u(t)$, the controlled output of the operator.

This model is based on the assumption that the operator has an "internal" aircraft model that is suitably modified by a

PRECEDING PAGE

Kalman filter derived from the internal model and state noise. An optimal controller is also implemented which ultimately produces the stick motion (output) of the operator.

In the above equations, the following definitions are used:

(1) K = Kalman gains = $PH^T V_y^{-1}$

where P is the solution of the filter Riccati equation

$$FP + PF^T + \Gamma Q \Gamma^T - PH^T V_y^{-1} HP = 0$$

(2) λ = optimal control gains = $g^{-1} G^T S$

where S is the solution of the control Riccati equation

$$SF + F^T S + Q - SG g^{-1} G^T S = 0$$

(3) T = operator's "reaction" time (in seconds)

The input to the algorithm are the matrices, covariances, etc. associated with the plant (e.g. F, G, H, W, etc.). At each iteration of the parameters to be estimated, F^*, G^* and λ are computed as well as the sensitivity matrices $\partial F^*/\partial\theta$, $\partial G^*/\partial\theta$, $\partial\lambda/\partial\theta$ where θ is a generic representation of the parameter(s) to be estimated.

The sensitivity equation for F^* is given by

$$F^*_\theta = (I + K'HT)(F_\theta - K'H_\theta - K'_\theta H - G\lambda_\theta - G_\theta \lambda)$$

$$+ [TK'H_\theta + TK'_\theta H + T_\theta K'H][F - K'H - G\lambda]$$

where $K' = e^{FT} K \approx (I + FT)K = K + TFK$

The sensitivity equation for G^* is given by

$$G^*_\theta = (I + K'HT)(K'_\theta) + (TK'H_\theta + TK'_\theta H + T_\theta K'H)K'$$

Also

$$K'_\theta = K_\theta + TFK_\theta + TF_\theta K + T_\theta FK$$

$$= (I + TF)K_\theta + (TF_\theta + T_\theta F)K$$

$$\lambda_\theta = \frac{\partial}{\partial\theta}(g^{-1} G^T S) = g^{-1} G^T S_\theta + g^{-1} G^T_\theta S - g^{-1} g_\theta g^{-1} G^T S$$

$$= g^{-1}[G^T S_\theta + G^T_\theta S - g_\theta \lambda]$$

where S_θ is obtained implicitly from

$$(F - G\lambda)^T S_\theta + S_\theta (F - G\lambda) = - Q_\theta - \lambda^T g_\theta \lambda - (F_\theta - G_\theta \lambda)^T S - S(F_\theta - G_\theta \lambda)$$

Depending on the parameter set, θ, to be estimated, the previous sensitivity equation(s) are constructed and computed in a sequence of various subroutines.

In the SCIDNT implementation, F^*, G^* and λ are computed transparently to the user. That is, at any iteration step, the matrices K', H, T, λ are computed (as well as the sensitivity equation) and are used to compute F^*, G^* and λ. Currently, the state equation matrices (and partials) are computed in subroutine STATG and measurement equation matrices (and partials) in subroutine MEASG. (The "general equation" mode of previous versions of SCIDNT are used for the generality required for this model.)

STATG assumes that the user is using the general linear model (specified of the TITLE card). The four integers in columns 69-80 of the parameter cards indicate the position of each parameter in the system matrices. The parameters enter into the STATG routine in the P-array and the indices in the IPDX array act as pointers from the P-array to a set of local plant and human operator matrices. These matrices local to STATG are:

FAC (NS x NS)	- plant system matrix
GAC (NS x NQAC)	- plant control matrix
GAMAC (NS x NDAC)	- plant noise matrix
QAC (NDAC x NDAC)	- plant process noise covariance (=W)
HAC (NPAC x NS)	- plant measurement matrix
DAC (NPAC x NQAC)	- plant measurement/control matrix (=0)
RAC (NPAC x NPAC)	- plant measurement noise covariance (=V_y)
A (NS x NS)	- state weighting matrix (=Q)
B (NQAC x NQAC)	- control weighting matrix (=g)
TAU (scalar)	- operator's prediction/delay time (=T)

IPDX elements are calculated by the user assuming the matrices are stacked columnwise in the order shown above. NQAC, NPAC and NDAC are the number of controls, measurements and disturbances, respectively, for the plant model. F*, G*, λ and DF, DG and Dλ (partials of F, G and λ) with respect to the parameters) for the operator model are calculated by STATG from FAC, GAC, GAMAC, QAC, HAC, DAC, RAC, A, B and TAU.

The entry points of STATG and their functions are described below:

STATG (main entry) - Calculates operator model F^* and G^* given FAC, GAC, GAMAC, QAC, HAC, DAC (=0), RAC, A, B and TAU

MEASG - Calculates λ and D (=0)

GQGG - Calculates $\Gamma Q \Gamma^T$ for operator model (since process noise is not implemented, this is always zero)

DERVG - Calculates F^*, G^* (operator)

DGQGG - Calculates $\partial/\partial\theta$ $(\Gamma Q \Gamma^T)$ for operator (always zero, since process noise is not implemented)

DERH - Calculates $\lambda\theta$ (operator)

DERR - Calculates $\partial/\partial\theta$ (V_m) for operator. Since IRCMP=0 is assumed, elements of V_m are not identified. It is, however, computed "after the fact" by computing the standard deviation of the residual between the input control (stick output of the operator) read from data and the estimated control of the operator.

Although restrictions on Γ, V_m and $\Gamma Q \Gamma^T$ (Q is used internal to STATG, however this corresponds to W external to STATG) have been pointed out above, the user can still identify elements in GAMAC, RAC, QAC existing in the previous versions of SCIDNT.

All STATG entries interface with the existing SCIDNT's UPDATE subroutine. Other SCIDNT's subroutines changed from the previous version are:

- OUTERR - arrays depending upon number of controls, enlarged to accommodate 5 controls

- INREAD - reads controls/measurements

In order to avoid overwriting intermediate results which might be in arrays DUM, DUM2, DDM in common block /DDM/, all scratch storage used by STATG is in local array D. Dimensioning information for D appears in the comments of the code for STATG. Other arrays which must be re-dimensioned (if number of measurements, states, controls, or process noise is increased above current maximum) are:

FAC, DFAC (NS**2)
GAC, DGAC (NS*NQAC)
GAMAC, DGAMAC (NS*NN)
QAC, DQAC (NN*NN)
HAC, DHAC (NPAC*NS)
DAC, DDAC (NPAC*NQAC)
RAC, DRAC (NPAC*NPAC)
AAC, DAAC (NS*NS)
BAC, DBAC (NQAC*NQAC)
P (NS*NS), K (NS*NPAC), KPR (NS*NPAC),
IKPRHT (NS*NS), FKHGC(NS*NS),
DP, DK, DKPR (same as P, K, KPR),
EVLRKF, EVLIKF, EVLROC, EVLIOC (NS)
EVCKF, EVCOC, EVCIKF, EVCIOC (NS*NS)
S, DS (NS*NS) C, DC (NQAC*NS)

Also, matrices such as DUM, DUM2, DUM1, FAA, HAA, etc. must be re-dimensioned in OUTER2RR.

5.2 PROGRAM STRUCTURE

In order to facilitate the future (potential) upgrading of the identification algorithm, the following section describes the program structure generic to the original version of SCIDNT. Currently, the identification algorithm utilizes the output error method embedded in the subroutine OUTERR. Utilization of the subroutine DRIVER can be used to incorporate process noise in the model.

The basic program structure is shown in Figure 5.1.

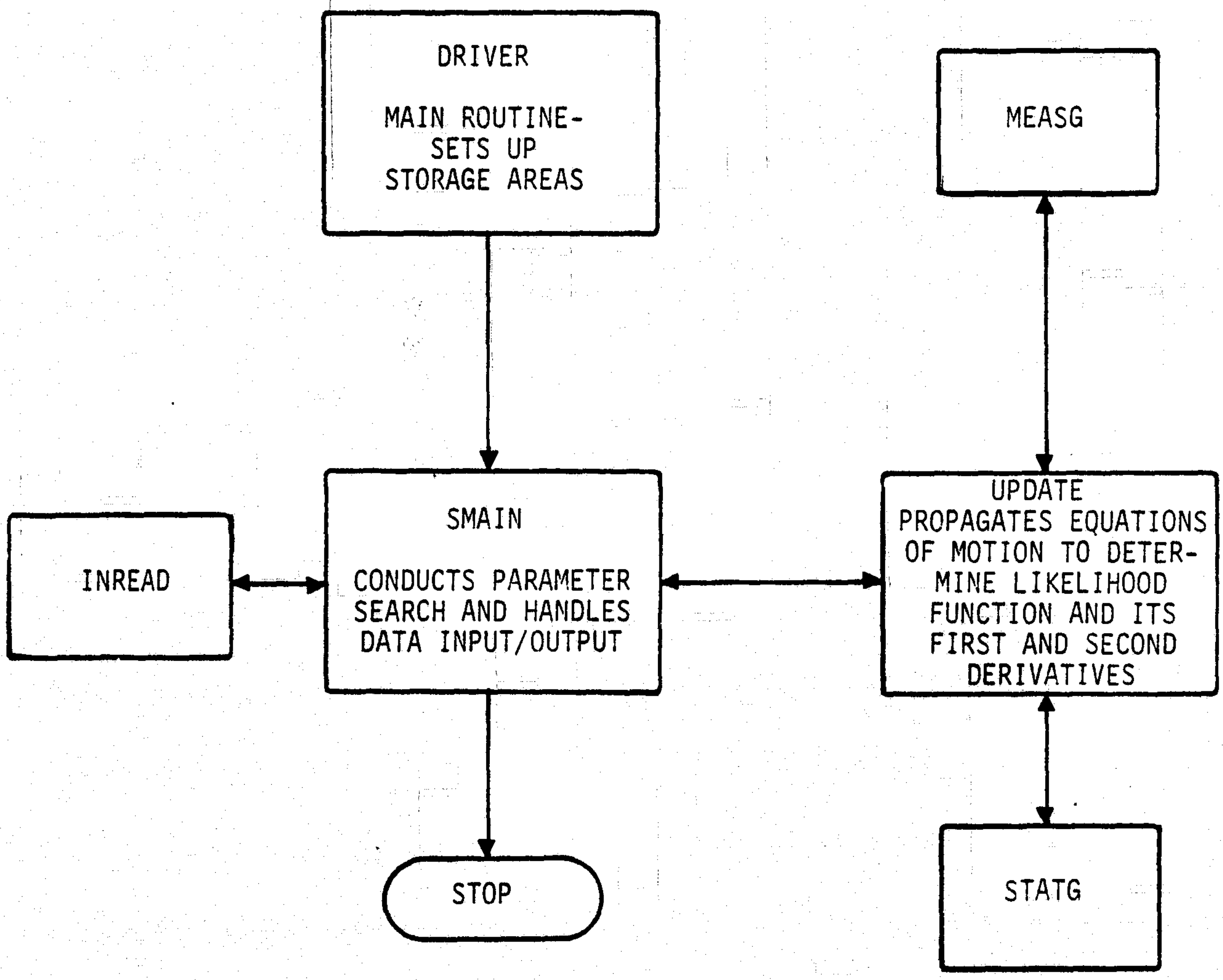


Figure 5.1 Basic Program Structure

DRIVER and OUTERR are two versions of the main routine. DRIVER should be used if process noise is present. DRIVER may also be used if process noise is not present, but in this case OUTERR may be used. OUTERR requires less storage than DRIVER to work with the same number of data points.

Both routines read cards which specify page heading, and number of states, controls, measurements, and process noise sources. This information is passed to the subroutines so that arrays may be dynamically dimensioned. Currently, the maximum allowable values are

number of states $\leq$ 5
number of controls $\leq$ 3
number of measurements $\leq$ 5
number of process noise sources $\leq$ 1
number of data points $\leq$ 301 for DRIVER
$\leq$ 501 for OUTERR
number of identifiable parameters $\leq$ 18 for OUTERR
$\leq$ 10 for DRIVER

SMAIN is called by DRIVER or OUTERR. SMAIN sets up the remaining program for identifying parameters. It reads cards which specify the parameter values, which parameters are to be fixed, bounds on the parameters, and other input data. SMAIN conducts the iterative maximum likelihood parameter search with successive calls to UPDATE. SMAIN also handles printout, printer plots, and writes time histories on tape or disc files.

UPDATE is called by SMAIN. UPDATE solves linear ordinary differential equations for state time histories and for the sensitivity of the likelihood function with respect to parameters that are to be identified. It also computes the likelihood function itself. It calls STATG to determine state dynamics and control distribution matrices and their gradients in terms of the current estimates of the parameters. Similarly, it calls MEASG for the measurement equations.

STATG and MEASG are subroutines which define the matrices in the state and measurement equation, respectively.

5.3 INPUT SPECIFICATIONS

SCIDNT requires two classes of inputs. The first type, which will be referred to as "card-input," defines the number of states, controls, etc., and denotes the parameters of the model which are to be identified. SCIDNT reads card-input from unit 5.

The second type of input consists of tabular values of the measurement and control time history. SCIDNT calls a subroutine INREAD once before beginning the identification algorithm. Subroutine INREAD reads the values of measurements y and controls u for the entire time period of the experiment. The user must supply his own version of INREAD for "real" data. This is because in general, INREAD must read data in many different formats for various types of simulation or flight test data. For simulated data, INREAD is compatible with data written to disc from the TVOPT program.

Card-input to the SCIDNT computer program consists of a sequence of cards constructed from various "card types." That is, several cards of a single type may be required in the input sequence. The card-input sequence is described in the following paragraphs. The card types are described in Table 5.1.

(1) One card of type 1. It contains run identification information which is printed at the top of each page of output.

(2) One card of type 2. The number of states, measurements, controls, and process noise sources specified on this card will be used in array dimension limits throughout the program. If they exceed the maximum dimensions allowed in the program, then a message will be printed and execution will stop. (The limits may be increased by re-coding DRIVER, the main routine.)

Table 5.1
Input Card Formats

CARD TYPE	COLUMNS	FORMAT	VARIABLE NAME	DESCRIPTION
1	1-76	19A4	TITLE	Program identification information. Specify GENE as the first four characters.
2	1-5	I5	K1MAX	Maximum number of iterations (default value is 6)
	6-10	I5	K2MAX	Maximum number of step cuts (default value is 4)
	11-20	D10.0	DL1	Sample time interval of time histories
	21-25	I5	NNS	Number of states (default value is 5)
	26-30	I5	NNP	Number of measurements (default value is 5)
	31-35	I5	NNQ	Number of controls (default value is 3 for a lateral case and 2 for a longitudinal case)
	36-40	I5	NNG	Number of process noise sources, must be ≥1 (default value is 1)
	41-45	I5	NMAXP	Total number of non-zero parameters
3	1-5	I5	IPLØT	=3, if both printer plots and a tape of the variables T, Y, and YPLØT (time, measurements, and measurement estimates) are to be made =2, if only the tape is to be made =1, if only the printer plots are to be made =0, if neither is to be done

Table 5.1 (Continued)

CARD TYPE	COLUMNS	FORMAT	VARIABLE NAME	DESCRIPTION
	6-10	I5	NØISE	=1, if process noise is to be included in the model
				=0, if no process noise is to be included in the model
	11-15	I5	IRCMP	=0, if estimate of measurement noise covariance (R matrix) is computed internally in program (usual procedure)
				=1, if estimate of measurement noise covariance is fixed to input values
	16-20	I5	IPRNT	=0, if extra printout for diagnostic purposes is not required
				=1, if extra printout for diagnostic purposes is desired
	21-25	I5	IINFØ	=0, if the a priori information matrix is not to be input, but zero filled
				=1, if the a priori information matrix is input (see card type 7)
	26-30	I5	IEIGF	=0, if the eigenvalues of F are not desired
				=1, if the eigenvalues of the initial and final F are to be computed and printed
	31-35	I5	IGUST	=1, if estimate of wind gust is to be plotted (ignored if NØISE = 0)
				=0, if not
	36-40		INC	Printer plot interval

Table 5.1 (Concluded)

CARD TYPE	COLUMNS	FORMAT	VARIABLE NAME	DESCRIPTION
4	1-5	I5	ICØLK(i) i=1, NP	=0, if the i^{th} column of K, the Kalman gain matrix, is to be zero filled
	20-25	I5		=1, if the i^{th} column of K is to be retained (usual procedure)
5	2-5	I4	J1	Parameter number as used in the model
	7-10	A4	J2	=blank, if the parameter is not to be identified =non-blank (such as an asterisk) if it is to be identified
	11-30	D20.4	P(J1)	Initial parameter value
	31-50	D20.4	PL(J1)	Lower bound on parameter
	51-68	D18.4	PU(J1)	Upper bound on parameter. If both the lower and upper bounds are zero on input, then they are defaulted to -1000 and +1000, respectively
	69-80	4I3	IPX	Parameter location
6	1	A1	ECHK	Any character (not a blank) to denote the end of cards of type 5
7	1-80	8D10.0	INFO(i,j), j=1,...no. of parameters being identified	i row of the information matrix (continue the row on additional cards as needed). Repeat for each row of INFO, beginning each row on a new card.
8	blank			
9	blank			
10	blank			

If an input variable on this card has a standard value, then the variable will default to the standard value if the input value is zero (see Table 5.1). The sample time interval is data dependent and has no default value. The program computes its own integration interval to strike a balance between computational speed and accuracy (see section 5.4).

(3) One card of type 3. This card flags various program options.

(4) One card of type 4. This card specifies which columns of the Kalman gain matrix, K, are to be retained or filled with zeroes. It is infrequently but sometimes advantageous to delete columns of K to enhance convergence of the parameter search. The usual practice is to retain all the columsn of K.

(5) Up to MMAXP cards of type 5, one for each non-zero or identified parameter in the model.

(6) One card of type 6 to designate the end of cards of type 5 in the input sequence.

(7) If IINFØ = 1 on card type 3, then the a priori information matrix is input here by including as many cards of type 7 as are necessary. If IINFØ = 0, then no cards of type 7 should be used.

(8) Three blank cards.

Card type 5 (see Table 5.1) is required for input of the system matrices. The matrices have the dimensions as shown in Table 5.2.

Table 5.2
Input System Matrices

MATRIX	DIMENSIONS
F	NS x NS
G	NS x NQ
Γ	NW x NG
W	NG x NG
H	HD x NS
D	NP x NQ
V_y	NP x NP
Q	NS x NS
g	NQ x NQ
T	scalar

where NS = # of states
NP = # of measurements
NG = # of process noise sources
NQ = # of control inputs

The user must allow for at least one column of Γ and one element of V_y.

The user defines the characteristics of both constant parameters and parameters to be identified using "card-input" of type 5. The last four integer entries define the location of the particular parameter in the system matrices. The location is an integer number specifying the vector location in the matrix list; i.e., each matrix is stored by column and the matrices are stored in the order shown in Table 5.2. Thus, location 1 implies F(1,1) and location NS*NS+1 implies G(1,1), etc. By using more than one of the four integer location inputs possible with each parameter, the user can direct SCIDNT to place a given parameter in up to four matrix elements. If a particular matrix location does not have a parameter value assigned, SCIDNT assumes a value of 0.0.

In summary, the input required for each parameter according to the format defined above is: blank, parameter sequence number (1 to 38), a non-blank character if this parameter is to be identified, the parameter value, allowed lower limit on parameter value, allowed upper limit on parameter value, and one to four location specification indices.

An example parameter card format is given below:

If

NS = 5
NP = 5
NQ = 2
NG = 1

then the parameter card

```
1 * -2.62   -1000.0   1000.0   1  21  42  62
```

will give these results for F and H:

$$F = \begin{bmatrix} [1] & & & & [21] \\ -2.62 & 0 & 0 & 0 & -2.62 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$H = \begin{bmatrix} [42] & & & & [62] \\ -2.62 & 0 & 0 & 0 & -2.62 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

F(1,1) = location 1 = 1

F(1,5) = location 5(5-1) + 1 = 21

H(1,1) = location $NS^2 + NS \cdot NQ + NS \cdot NGN + NGN^2 + 5(1-1) + 2 = 42$

H(1,5) = location of H(1,1) + 5(5-1) = 62

5.3.1 Subroutine INREAD

The purpose of subroutine INREAD is to retrieve measurement and control time history data from cards, magnetic tape, or other mass storage devices.

Calling Sequence: CALL INREAD (U, NQ, NDPMAX, Y, NP, T, TI, TF, NDPA)

Inputs:

NQ = number of controls

NP = number of measurements

NDPMAX = maximum number of data storage locations (i.e. data points) for each measurement or control

Outputs:

T = array for storing the time of each data point

U = array for storing the control time histories

Y = array for storing measurement time histories

NDPA = actual number of data points ready by the program

Array Dimensions:

T(NDPMAX)

U(NU, NDPMAX)

Y(NY, NDPMAX)

Notes:

(1) The program uses the array T only for the plot outputs. The identification algorithm does not require T.

(2) TI and TF are the initial and final times of the measurement time histories. The user can ignore these two arguments while writing INREAD, except that two dummy arguments must be included in the argument list.

(3) INREAD should either read the value of NDPA before reading T, Y, and U or should count the number of columns of these arrays as it reads them. The main

program does not initialize NDPA before calling INREAD.

(4) INREAD should not reset the values of NQ, NP, or NDPMAX.

(5) INREAD may perform any desired preprocessing of Y and U.

5.4 OUTPUT GUIDE AND EFFECTIVE USE OF THE PROGRAM

Three forms of output can be obtained from Linear SCIDNT. They are:

(1) Tabular printout showing the progress of the parameter estimation process.

(2) Printer plots of time histories for measurements and measurement estimates using the final values of identified parameter and inputs.

(3) Measurement, measurement estimates and input time history data stored on magnetic tape for use in generating off-line plots.

The first type of output is always produced. Types 2 and 3 are optional and may be selected by the user (see input data cards description).

The option is provided to put measurement and control time history data on magnetic tape, primarily for the purpose of producing off-line plots analogous to the printer plots. Data is written to tape (currently designated as being on logical unit 2) in unformatted records as follows:

RECORD	DATA
1	NS, NQ, NP, NEW
	NS = Number of states
	NQ = Number of controls
	NP = Number of measurements
	NEW = Number of data points per variable
2 thru NEW+1	T(K), U(K), Y(K), YPLØT(K)
	T = Time for k^{th} data step (1 variable per record)
	U = Controls for k^{th} data step (NQ variables per record)
	Y = Actual measurements for k^{th} data step (NP variables per record)
	YPLØT = Estimated measurements for k^{th} data step (NP variables per record)
NEW+2	End-of-file mark

The computer program's applicability is quite general. The following guidelines may make the program more useful for the evaluation of stability and control derivatives from test data.

5.4.1 Data Integration Step Size

The program uses a second-order Runge-Kutta method to solve ordinary differential equations for the states and for the parameter sensitivites. This method is accurate only if the step size is much smaller than the smallest time constant of the system. However, a step size that is too small will require excessive execution time and may lead to problems with roundoff errors. A sampling rate of 20-50 times the highest system natural frequency (Hz) should be an adequate compromise.

Linear SCIDNT automatically chooses the integration step size. Subroutine INTSTP sets the integration step size DTINT to

$$DTINT = DTSMPL/K$$

where DTSMPL is the sample period for the data and K is the smallest integer greater than ten times the magnitude of the largest eigenvalue of the system dynamics matrix F. This value is adjusted before each parameter identification iteration to reflect changes in the F matrix.

The user can make estimates of relative program running time using estimates of the largest norm of the system eigenvalues.

5.4.2 Primary and Secondary Parameters

The computer program can identify any or all unknown parameters if enough information is available about these parameters from the data. To identify a large number of parameters, a very careful procedure is often required to ensure convergence. In the procedure which has been found to be most successful, the parameters are divided into two or more groups in the decreasing order of the effect they have on the response. Initially, only the most important parameters are identified, leaving the remaining ones fixed at *a priori* values. Once a reasonable convergence is achieved on these parameters, the second group may be added to the list of identifiable parameters. The identification is carried out using this new set of parameters, which are identified until a reasonable convergence is reached. This procedure is repeated until all the parameters are included in the identified set.

5.4.3 Initial Parameter Estimates

Good *a priori* values (e.g., from other wind tunnel data or flight tests) should be used for start-up. If they are not available, it may be necessary to use a least-squares type of procedure to obtain starting values.

5.4.4 Data Record Length

Sufficient data length should be used to identify parameters. If the data length is too short, the identified model may yield a good time history match even though the parameter estimates are inaccurate. A data length equal to 2 to 3 times the longest period of the system should prove adequate. Keep in mind the required maximum number of data points.

5.4.5 Data Sampling Interval

In order to assure adquate information content in the data, a sampling interval of at least 25 times the highest system natural frequency is required. A faster sampling rate provides somewhat more accurate parameter estimates because of some additional information available in the sampled data. However, the algorithm realizes diminishing returns in terms of increased parameter accuracy as the parameter estimates approach the continuous time case.

5.4.6 Diagnostics

ABNORMAL TERMINATION CONDITION	CORRECTIVE ACTION
Step Cut Limit Exceeded	(1) Increase step cut limit and restart using parameter estimates from final iteration of terminated run.
	(2) Delete parameters with low confidence (F-value small) from the search.
Iteration Limit Exceeded	(1) Restart program using parameter values from final iteration of terminated run.
	(2) Increase iteration limit.

NORMAL TERMATION, BUT PROBLEMS IN RESULT	CORRECTIVE ACTION
High Value of a Particular Measurement's Noise Covariance	(1) Faulty data: Inspect for correct sign, slipping, lost values, "shot" noise.
	(2) Incorrect values for "fixed" parameter.
High Standard Deviation on a Particular Parameter Estimate	(1) Insufficient excitation of the mode which that parameter influences. Rerun identifying only the more important parameters.
	(2) Too many parameters identified for that particular data record (i.e., overparameterization). Rerun, identifying only the more important parameters.
	(3) Parameter influence masked by a parallel influence from another, much larger parameter (i.e., identifiability).

5.5 PROGRAM FLOWCHART (STATG)

The program flowchart for STATG is shown in Figure 5.2. Note that STATG uses the VASP routines

TRANP
UNITY
INV

These must be present in a user library when the program is loaded and executed. Also, STATG uses the extended VASP routines

ALA
APALB
APALBT
ATXB
AXB
AXBT
AXNXBT
AZERO

These routines were used because they have more general calling sequences, and, in many cases, combine the operation of transposition with multiplication or addition, resulting in more efficient use of temporary storage and fewer subroutine calls for complicated matrix algebra expressions. Descriptions and calling sequences for these routines are as follows (N followed by a matrix name, e.g. NA, denotes a 2-element integer array containing the numbers of rows and columns of the matrix).

CALL ALA (A, NA, B, ALPHA) (B = ALPHA*A)

Scales A by ALPHA, storing result in B. A and B can be the same matrix.

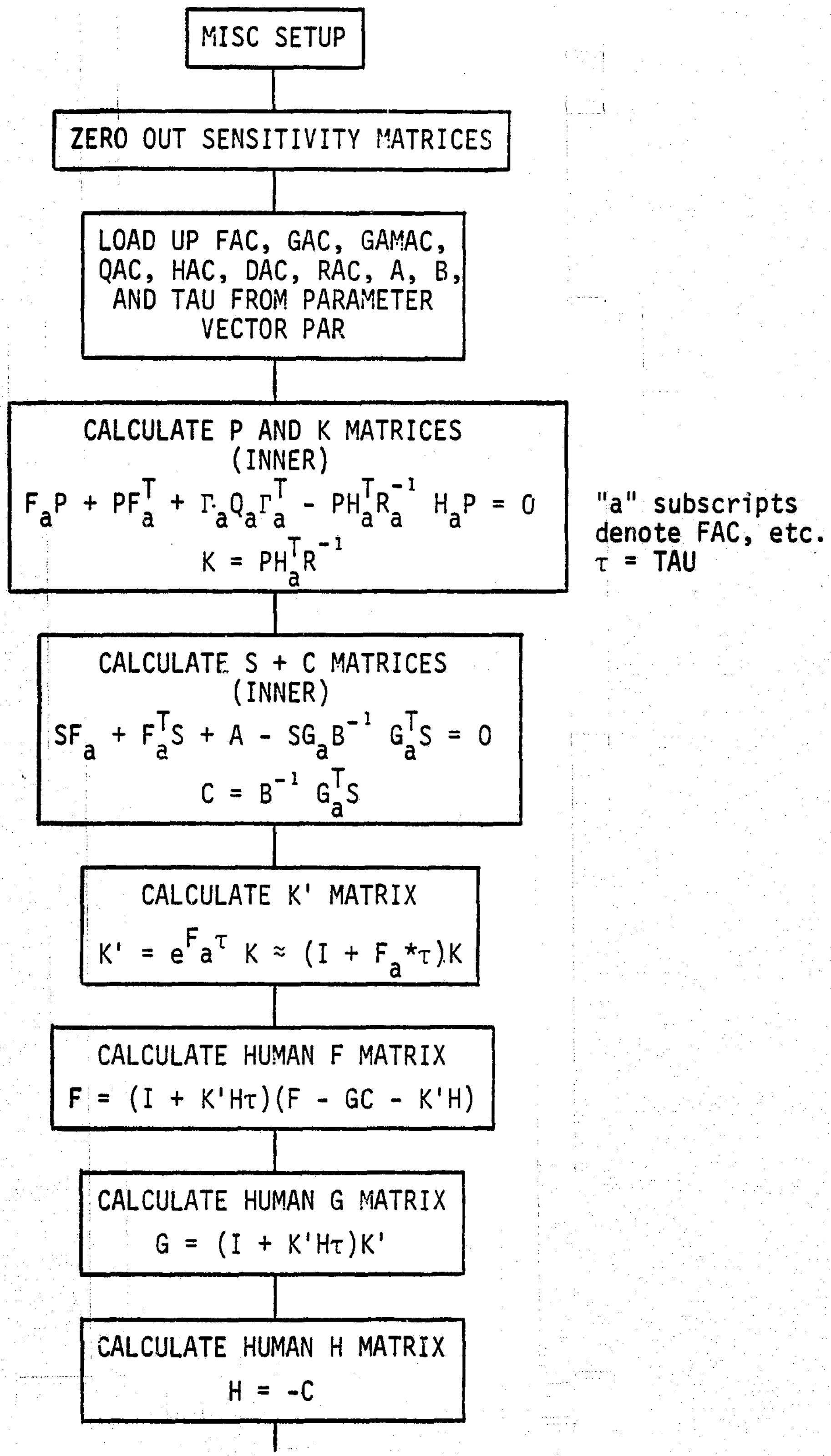


Figure 5.2 Program Flowchart (STATG)

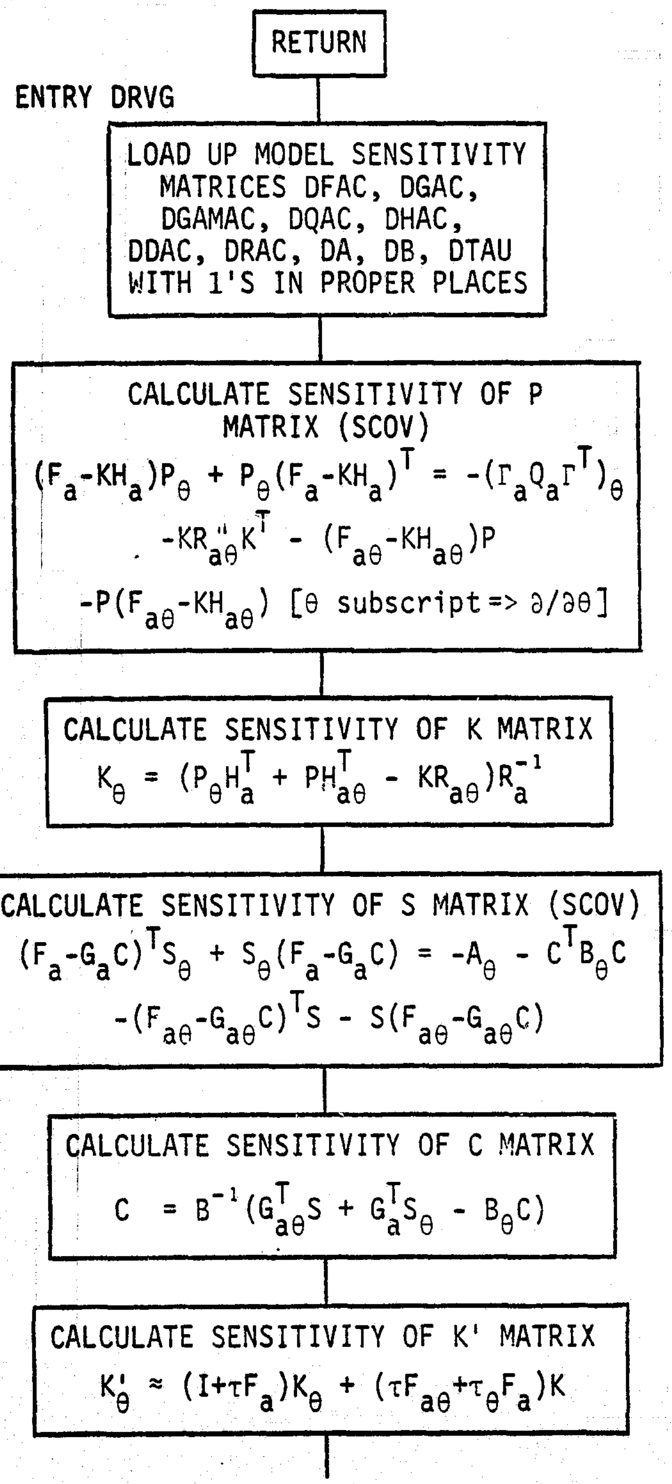


Figure 5.2. Program Flowchart (STATG) (Continued)

CALCULATE SENSITIVITY OF <u>HUMAN</u> F MATRIX (DF)

$$F_\theta = (I+K'H_a\tau)(F_{a\theta}-K'H_{a\theta}-K'_\theta H_a-G_aC_\theta-G_{a\theta}C)$$
$$+(\tau K'H_{a\theta}+\tau K'_\theta H_a+\tau_\theta K'H_a)(F_a-K'H_a-G_aC)$$

CALCULATE SENSITIVITY OF <u>HUMAN</u> G MATRIX (DG)

$$G_\theta = (K+K'H_a\tau)K'_\theta+(\tau K'H_{a\theta}+\tau K'_\theta H_a+\tau_\theta K'H_a)K'$$

CALCULATE SENSITIVITY OF <u>HUMAN</u> H MATRIX (DH)

$$H_\theta = -C_\theta$$

RETURN

Figure 5.2 Program Flowchart (STATG) (Concluded)

CALL APALB (A, NA, B, C, ALPHA) ($C = A + ALPHA*B$)

Scales B by ALPHA and adds it to A, storing result in C. A, B, and/or C may be same matrix. Used for matrix addition and subtraction when ALPHA = 1 or -1. (No multiplies are performed in these cases.)

CALL APALBT (A, NA, B, C, ALPHA) ($C = A + ALPHA*B^T$)

Scales transpose of B and adds to A, storing result in C. A, B, and/or C may be same matrix. Note that A + B must be square.

CALL ATXB (A, NA, B, NB, C, NC) ($C = A^T*B$)

Multiplies transpose of A by B, storing result in C. A + C or B + C may not be same matrix. NC is output parameter.

CALL AXB (A, NA, B, NB, C, NC) ($C = A*B$)

Multiplies A by B, storing result in C. A + C or B + C may not be same matrix. NC is output parameter.

CALL AXBT (A, NA, B, NB, C, NC) ($C = A*B^T$)

Multiplies A by transpose of B, storing result in C. NC is output parameter. A + C or B + C may not be the same matrix.

CALL AXWXBT (A, NA, W, B, NB, C, D) ($C = A*W*B^T$)

Multiplies A by W by transpose of B, storing result in C. D is an NA(2) x 1 scratch vector. W is assumed to be dimensioned (NA(2) x NB(2). A + C or W + C or B + C must not be same matrix. D must also be a separate storage area from A, W, B, or C.

CALL AZERO (A, NA) ($A = 0$)

Zeros out matrix A.

5.6 TSS/360 OPERATION

The following control cards are required to execute SCIDNT on the NASA ARC TSS/360 system.

```
(1)  LØGØN...
(2)  DDEF FT10F001, VS, TIME.HISTORY, DISP=ØLD
(3)  JBLB SCIDNT
(4)  JBLB HUMANØP
(5)  LØAD BLØCKD$$
(6)  CALL ØUTER2$$
(7)  {input data
(8)  LØGØFF
```

The description of the above execution deck follows:

(1) Usual LØGØN card.

(2) Data definition card specifying the file on which time history is stored.

(3) Assign library SCIDNT.

(4) Assign library HUMANØP.

(5) Load in block data.

(6) Execute program.

(7) Data.

(8) Usual LØGØFF.

APPENDIX A

The following program is a "stand-alone" computer program which can be used to create a general output disc file to be read as input for the TVSR program. Specifically, the program reads the input data from cards and writes to disc file in the identical format as does the TF program. Since the TVSR program inputs data in identically this format, this stand-alone code can be used to generate input data for TVSR without the need to exercise the TF program.

The input format for TVSR is given in Table 3.2. The following program can serve to create this data from an input card deck; an example is also shown. Note: In the example, card 1 indicates 3 x's. These x's correspond to (1) the number of plant states, (3) the number of human states and (3) the number of augmented states. In the case when the input vector is not augmented (with the deterministic input, p(t)), these x values are arbitrary. For the case when the input vector is augmented, care should be taken in inserting the correct values.

```
      IMPLICIT  REAL*8 (A-H,O-Z)
      DIMENSION  IFMT(3)
      DIMENSION F(15,15), G(15,15), H(15,15), W(4,4), VY(5,5)
      DIMENSION  VA(5,5)
C
      DATA  IFMT  / '(8E', '10.5', ')'  /
C
C  READ DIMENSIONS
C
      READ(5,100) NS, NNP, NSH, NMH, NSC, NI, NM, NSA, NCA
  100 FORMAT(9I5)
C
C  READ MATRICES AT START TIME
C
      CALL RDMAT(F,NS,NS,IFMT,5)
      CALL RDMAT(G,NS,NI,IFMT,5)
      CALL RDMAT(H,NM,NS,IFMT,5)
      CALL RDMAT(W,NNP,NNP,IFMT,5)
      CALL RDMAT(VY,NMH,NMH,IFMT,5)
      CALL RDMAT( VA, NCA, NCA, IFMT, 5 )
C
C  WRITE DIMENSIONS AND MATRICES AT START TIME
C
      WRITE(10) NS, NNP, NSH, NMH, NSC, NI, NM, NSA, NCA
      CALL BWTMAT(F,NS,NS,10)
      CALL BWTMAT(G,NS,NI,10)
      CALL BWTMAT(H,NM,NS,10)
      CALL BWTMAT(W,NNP,NNP,10)
      CALL BWTMAT(VY,NMH,NMH,10)
      CALL BWTMAT(VA, NCA, NCA, 10)
      ENDFILE  10
C
C  READ MATRICES AT END TIME
C
      CALL RDMAT(F,NS,NS,IFMT,5)
      CALL RDMAT(G,NS,NI,IFMT,5)
      CALL RDMAT(H,NM,NS,IFMT,5)
      CALL RDMAT(W,NNP,NNP,IFMT,5)
      CALL RDMAT(VY,NMH,NMH,IFMT,5)
      CALL RDMAT( VA, NCA, NCA, IFMT, 5)
C
C  WRITE DIMENSIONS AND MATRICES AT END TIME
C
      WRITE(11) NS, NNP, NSH, NMH, NSC, NI, NM, NSA, NCA
      CALL BWTMAT(F,NS,NS,11)
      CALL BWTMAT(G,NS,NI,11)
      CALL BWTMAT(H,NM,NS,11)
      CALL BWTMAT(W,NNP,NNP,11)
      CALL BWTMAT(VY,NMH,NMH,11)
      CALL BWTMAT(VA, NCA, NCA, 11)
      ENDFILE  11
      STOP
      END
```

```
      ROU     RC    ( A     , NC    MT,    IT)
      IMPLICIT  REAL*8 (A-H,O-Z)
      DIMENSION  A(NR,NC)
      DO 100 I=1,NR
      READ(IUNIT,IFMT) (A(I,J),J=1,NC)
  100 CONTINUE
      RETURN
      END
```

```
SUBROUTINE  BWTMAT(A,NR,NC,IUNIT)
IMPLICIT  REAL*8(A-H,O-Z)
DIMENSION A(NR,NC)
WRITE(IUNIT ) ((A(I,J),I=1,NR),J=1,NC)
RETURN
END
```

```
     X     1     X     1     7     5     5     X     3
0.0         0.0         0.0         0.0         0.0         0.990       0.0
0.0         0.0         -0.263      0.0         0.0         0.0         1.0
0.0         0.263       0.0         0.0         0.0         0.139       0.0
0.0         0.0365      0.0         0.0         0.0         1.0         0.0
0.0         0.0         -0.263      0.0         0.0         0.0         1.0
      7.E-70.0          0.0         0.0         0.139
                  7.E-7
0.0         0.0         0.0         0.0         -0.990
                            2.88E-4
                                       -2.88E-4
            1.0                                 -1.0
-1.0                    -0.1405     1.0098
                        1.0098      -0.1405
                                    -.707         -.707
                                    .707         -.707
1.0
1.0
```