

FACILITY FORM 802

N71-33963
(ACCESSION NUMBER)

_____ (THRU)

_____ (PAGES)

NASA CR - 121912
(NASA CR OR TMX OR AD NUMBER)

_____ (CODE)

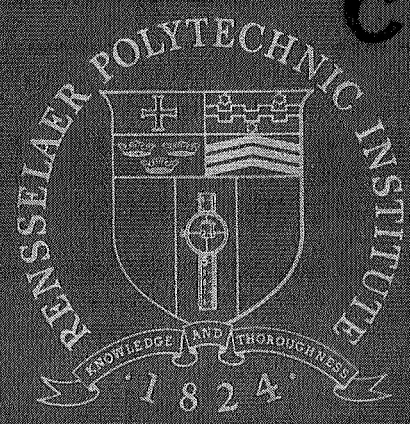
_____ (CATEGORY)

Final Report Vol. IV
 Contract No. NAS8-21131
 covering Period Nov. 4, 1969 - April 4, 1971
 NATIONAL AERONAUTICS AND
 SPACE ADMINISTRATION

User's Guide for Computer Program COMPDES

by
 Lutz Willner

CASE FILE
 COPY



Rensselaer Polytechnic Institute
 Troy, New York

Rensselaer Polytechnic Institute
Troy, New York

Final Report Vol. IV
Contract No. NAS8-21131
Covering Period Nov. 4, 1969 - April 4, 1971
NATIONAL AERONAUTICS AND
SPACE ADMINISTRATION

User's Guide for Computer Program COMPDES

by

Lutz Willner

Submitted on behalf of

Rob J. Roy
Professor
Systems Engineering Division

User's Guide for the Computer Program COMPDES

1. Introduction

In reference 1 the theories and algorithms for two digital computer programs, namely, a compensator design program and a pole placement program, are developed. The two programs are written in FORTRAN IV and are listed in appendices A and B of reference 1. The present user's guide describes the use of the compensator design program, COMPDES, only; the necessary instructions for the useage of the pole placement program are given in the form of comments at the beginning of that program.

The purpose of COMPDES is the design of a low-order compensator to stabilize a given controllable and observable linear time-invariant system in the presence of parameter uncertainties. The theoretical development for the algorithm can be found in reference 1.

2. Program Outline

COMPDES can be broken down into two major sections. The first section consists of a gradient procedure which tries to increase the stability of a system by computing appropriate feedback gains. The resulting closed-loop system is tested for stability with respect to the maximum possible parameter variations. The program terminates if stability can be guaranteed. If not, the program proceeds to program section two. Section one can be by-passed.

Program section two carries out the actual compensator design and sensitivity reduction. The compensator is composed of a state estimator and a set of 'all-state' feedback gains. The sensitivity function is minimized by means of the Davidon function minimization method. The

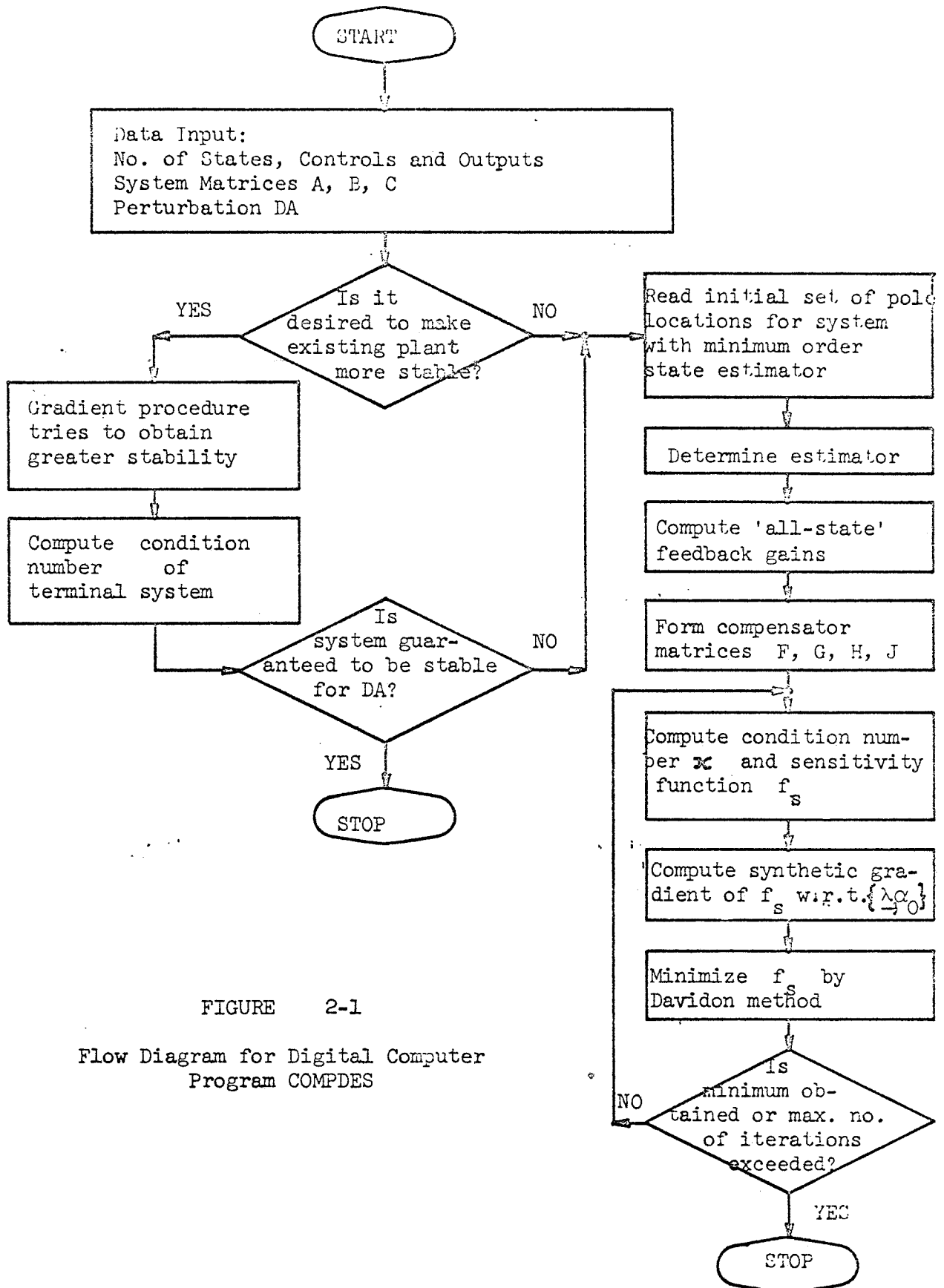


FIGURE 2-1

Flow Diagram for Digital Computer Program COMPDES

gradient of the sensitivity function with respect to the eigenvalues and ALO is required for the Davidon method and is synthetically generated. Thus the computed gradient may not be zero where the function actually has a minimum. To avoid numerical oscillation about such a minimum, the total number of iterations is limited. A flow chart of the COMPDES computer program is depicted in Figure 2-1.

3. Table of Input Parameters

Parameter	Input Format	Type	Explanation
NS	} 8I10	Int., simple	No. of System States
NC		" "	No. of Control Inputs
NF		" "	No. of Outputs
NFF		" "	Desired order of compensator, used in gradient method to increase stability.
IDELR		" "	$\begin{cases} \leq 0 & \text{DELR} = .2 \text{ by default} \\ > 0 & \text{read DELR} \end{cases}$
ISTOP		" "	$\begin{cases} \leq 0 & \text{STOPR is computer from other input data} \\ > 0 & \text{Read STOPR} \end{cases}$
IMATJ		" "	$\begin{cases} \leq 0 & \text{Matrix [FJ] = [0.] by default} \\ > 0 & \text{Read matrix [FJ]} \end{cases}$
NUL		" "	≥ 0 ; > 0 to generate a more conservative STOPR and condition no. α .
IPASS	8I10	Int., simple	$\begin{cases} \leq 0 & \text{Program tries to determine an appropriate compensator or order NFF via gradient method} \\ > 0 & \text{Gradient method is bypassed, compensator will be of order NFF=NS-NF} \end{cases}$

Parameter	Input Format	Type	Explanation
DELR	7F10.4	Real, simple	Only if IDELR > 0 ; DELR is the increment for the gradient method
STOPR	7F10.4	Real, simple	Only if ISTOP > 0 ; STOPR determines the minimum required system stability for termination of the gradient method.
A	5F15.6	Real, matrix	NS * NS System matrix; input by rows.
DA	5F15.6	" "	NS * NS System uncertainty matrix; input by rows.
B	5F15.6	" "	NS * NC System input matrix; input by rows.
C	5F15.6	Real, matrix	NF * NS System output matrix; input by rows.
ACC	5F15.6	Real, simple	Min. stability required for nominal system plus compensator.
AC		Real, simple	Min. Stability required for perturbed system plus compensator.
FJ	5F15.6	Real, matrix	Only if IMATJ > 0; NC * NF matrix of initial direct feedback gains to start gradient method.
IPOLE	}8I10	Int., simple	$\left\{ \begin{array}{l} \leq 0 \text{ Program generates initial set of stable system and compensator poles} \\ > 0 \text{ Read system and compensator poles.} \end{array} \right.$
IAREA		" "	$\left\{ \begin{array}{l} \leq 0 \text{ No region constraint for pole locations} \\ > 0 \text{ Read region constraint DR and weighting GA} \end{array} \right.$
LIMIT		" "	Maximum no. of iterations (for any given set of initial poles) to determine valid compensator.
KSHIFT		" "	≥ 1 , no. of different initial pole sets. If KSHIFT > 1, program will generate new initial sets.

Parameter	Input Format	Type	Explanation
DR	4F20.10	Real, simple	Only if IAREA > 0 { Region height 2*DR, width DR. Final design should have poles in or close to this region. Weighting of region constraints; choose GA ≤ .001
GA		" "	
EMS	4F20.10	Real, matrix	Only if IPOLE > 0, NS*2 matrix (i.e., real and im. part of poles). NS poles initially to be realized by system with 'all-state' feedback.
EMF	4F20.10	" "	Only if IPOLE > 0, NFF*2 matrix. NFF poles initially to be realized by the compensator.

4. Example

The following Computer Print-out represents a typical example.

The example chosen describes a 3rd order system with one input and two outputs (Note: for the program to work the system has to be in observer canonical form, i.e., the first part of the NF*NS output matrix C represents the identity matrix of rank NF, the remainder of the matrix C is zero). The card set-up for the input data is shown in Fig. 4-1.

Fig. 4-2 presents the actual computer print-out.

STATES INPUTS OUTPUTS COMP-ORD IDELR ISTCP IPATJ NUI
 NS = 3 NC = 1 NF = 2 NFF = C 0 0 1 0

SYSTEM MATRIX A.

```

-0.4000000D 01    -0.2000000D C1    C.1000000D C1    0.5000000D 00
-0.3000000D 01    0.2000000D C1    C.2500000D C1    0.8000000D 00
0.2000000D 01

```

MAXIMUM ABSOLUTE CHANGE DA OF THE ELEMENTS OF THE SYSTEM MATRIX A

```

0.4000000D 00    0.2000000D C0    C.1000000D C0    0.5000000D-01
0.3000000D 00    0.2000000D C0    C.2500000D C0    0.8000000D-01
0.2000000D 00

```

CONTROL INPUT MATRIX B

```

-0.2000000D 01    0.1000000D C1    C.3000000D C1

```

OUTPUT MATRIX C

```

0.1000000D 01    0.C    C.C    0.0
0.1000000D 01    0.C

```

ACCEPTABLE LARGEST REAL PART OF THE EIGENVALUES FOR THE WORST CASE OF
 PARAMETER UNCERTAINTY HAS TO BE LESS THAN AC = -C.20000
 EIGENVALUES OF NOMINAL SYSTEM ARE KEPT TO THE LEFT OF ACC = -2.00000

MINIMUM STABILITY REQUIRED FOR TERMINATION -4.09999943

STABILITY INCREASE INCREMENT = C.20000000

GRADIENT PROCEDURE TRIES TO DETERMINE A COMPENSATOR OF ORDER NFF = 0
 SUCH THAT THE CLOSED-LOOP SYSTEM DOES NOT BECOME UNSTABLE FOR THE GIVEN
 MAXIMUM PARAMETER UNCERTAINTIES (SEE MATRIX DA).

DIRECT FEEDBACK MATRIX FJ

```

-0.1000000D 01    -0.2500000D C1

```

ROOTS	REAL PART	IMAG. PART
	-0.251839C 01	C.165C62D C1
	-0.251839C 01	-C.165C62D C1
	-0.463227E 00	C.C

MAXIMUM REAL PART OF ROOTS

```

-0.4632269C 00

```

CLOSED-LOOP SYSTEM MATRIX A(C.-L.).

```

-0.2000000D 01    0.2000000D C1    C.1000000D C1    -0.5000000D 00
-0.5500000D 01    0.2000000D C1    -C.5000000D C0    -0.6700000D 01
0.2000000D 01

```

FIGURE 4-2, i
 Computer Print-Out for 3rd Order Example

EIGVEC ERROR MESSAGES

SWI=0.3692C-11 ITER= 2 DIF=C.1866E-10

EIGENVECTORS CORRES TO MRP EIGENVALLE

RCW REAL PART	ROW IMAG PART	CCL REAL PART	CCL IMAG PART
-0.7049734D-01	0.C	C.1CCCCCDD C1	0.0
0.1000000C 01	0.C	C.2331752D C0	0.0
-0.7833232C 00	0.C	C.8372354D C0	0.0

GRADIENT MATRIX DUE TO FJ

0.2451565C 01 0.5716541D CC

DIRECT FEEDBACK MATRIX FJ

-0.1309494C 01 -0.2572168C C1

RCCTS	REAL PART	IMAG. PART
	-0.159931C 01	0.C
	-0.167694C 01	0.1E4524D C1
	-0.167694C 01	-0.1E4524D C1

MAXIMUM REAL PART OF ROOTS

-0.1599307C 01

CLOSED-LOOP SYSTEM MATRIX A(C.-L.).

-0.1381012D 01	0.2144235D C1	C.1CCCCCDD C1	-0.8094941C 00
-0.5572168C 01	0.2CCCCCDD C1	-C.142E482D C1	-0.6916503C 01
0.2000000D 01			

EIGVEC ERROR MESSAGES

SWI=0.2651D-14 ITER= 2 DIF=C.1485E-14

EIGENVECTORS CORRES TO MRP EIGENVALLE

RCW REAL PART	ROW IMAG PART	CCL REAL PART	CCL IMAG PART
0.2558633C-01	0.C	C.1CCCCCDD C1	0.0
0.1000000C 01	0.C	-C.1214326D C0	0.0
-0.5627712D 00	0.C	C.1635256D C0	0.0

GRADIENT MATRIX DUE TO FJ

0.3936033D 01 -0.477962ED CC

DIRECT FEEDBACK MATRIX FJ

-0.1324045D 01 -0.257C4C1D C1

RCCTS	REAL PART	IMAG. PART
	-0.165629C 01	0.C
	-0.163301C 01	C.1E6EE4D C1
	-0.163301C 01	-0.1E6EE4D C1

MAXIMUM REAL PART OF ROOTS

-0.1633008C 01

FIGURE 4-2, ii

Computer Print-Out for 3rd Order Example (continued)

```

CLOSED-LOOP SYSTEM MATRIX A(C,-L.).
-0.1351909D 01    0.314C8C1D C1    C.1CCCCCDD C1    -0.8240453C 00
-0.5570401C 01    0.2CCCCCDD C1    -C.1472136D C1    -0.6911202C 01
  0.2000000D 01

```

```

EIGVEC ERROR MESSAGES
SW1=0.77130D-11    ITER= 2    DIF=C.7487E-12

```

```

EIGENVECTORS CORRES TO MRP EIGENVALLE
RCW REAL PART    ROW IMAG PART    CLL REAL PART    CCL IMAG PART
0.2057685D 00    0.3487345C-C1    C.1CCCCCDD C1    0.0
0.1000000D 01    -0.13E7775D-1E    -C.6E538E3D-C1    0.3779265C 00
-0.4762001D 00    -0.254555CC CC    -C.721137CD-C1    0.6818475C 00

```

```

GRADIENT MATRIX DUE TO FJ
-0.2885096C 01    0.7E49145C CC

```

```

DIRECT FEEDBACK MATRIX FJ
-0.1321017C 01    -0.25712C3D C1

```

```

ROOTS    REAL PART    IMAG. PART
-0.164440C 01    C.C
-0.164238C 01    C.1E65CED C1
-0.164238C 01    -C.1E65CED C1

```

```

MAXIMUM REAL PART OF ROOTS
-0.1642382C 01

```

```

CLOSED-LOOP SYSTEM MATRIX A(C,-L.).
-0.1357965D 01    0.31424C7D C1    C.1CCCCCDD C1    -0.8210174C 00
-0.5571203D 01    0.2CCCCCDD C1    -C.1463052D C1    -0.6913610D 01
  0.2000000D C1

```

```

EIGVEC ERROR MESSAGES
SW1=0.7376D-10    ITER= 2    DIF=C.4553E-11

```

```

EIGENVECTORS CORRES TO MRP EIGENVALLE
RCW REAL PART    ROW IMAG PART    CLL REAL PART    CCL IMAG PART
0.2042519D 00    0.361C422C-C1    C.1CCCCCDD C1    -0.4163336C-16
0.1000000C 01    -0.277555ED-1E    -C.671487CD-C1    0.3774747C 00
-0.4754359D 00    -0.25335E4D CC    -C.734C831D-C1    0.6788966C 00

```

```

GRADIENT MATRIX DUE TO FJ
-0.2902577D 01    0.7597035D CC

```

```

DIRECT FEEDBACK MATRIX FJ
-0.1320730D 01    -0.2571275D C1

```

```

ROOTS    REAL PART    IMAG. PART
-0.164327C 01    C.C
-0.164327C 01    C.1E6472D C1
-0.164327C 01    -C.1E6472D C1

```

FIGURE 4-2, iii

Computer Print-Out for 3rd Order Example (continued)

MAXIMUM REAL PART OF ROOTS

-0.1643272C 01

CLOSED-LOOP SYSTEM MATRIX A(C.-L.).

-0.1358539C 01	0.3142557D C1	C.1CCCCCDD C1	-0.8207304C 00
-0.5571279C 01	0.2CCCCCDD C1	-C.1462191D C1	-0.6913836C 01
0.2000000D 01			

EIGVEC ERROR MESSAGES

SW1=0.9037D-10 ITER= 2 DIF=C.5437E-11

EIGENVECTORS CORRES TO MRP EIGENVALLE

RCW REAL PART	ROW IMAG PART	CCL REAL PART	CCL IMAG PART
0.2041088C 00	0.36221CCD-C1	C.1CCCCCDD C1	0.4163336C-16
0.1000000D 01	0.13E7779D-1E	-C.672C672D-C1	0.3774326C 00
-0.4753631D C0	-0.252245CD CC	-C.7353225D-C1	0.6786178D 00

GRADIENT MATRIX DUE TO FJ

-0.2904210D 01 0.759198ED CC

DIRECT FEEDBACK MATRIX FJ

-0.1320730D 01 -0.2571279D C1

ROOTS REAL PART IMAG. PART

-0.164327C 01	C.C
-0.164327C 01	0.1E6472D C1
-0.164327C 01	-0.1E6472D C1

MAXIMUM REAL PART OF ROOTS

-0.1643272C 01

CLOSED-LOOP SYSTEM MATRIX A(C.-L.).

-0.1358539D 01	0.3142557D C1	C.1CCCCCDD C1	-0.8207304C 00
-0.5571279C 01	0.2CCCCCDD C1	-C.1462191D C1	-0.6913836C 01
0.2000000D 01			

EIGVEC ERROR MESSAGES

SW1=0.9037D-10 ITER= 2 DIF=C.5437E-11

EIGENVECTORS CORRES TO MRP EIGENVALLE

RCW REAL PART	ROW IMAG PART	CCL REAL PART	CCL IMAG PART
0.2041088C 00	0.36221CCD-C1	C.1CCCCCDD C1	0.4163336C-16
0.1000000D 01	0.13E7779D-1E	-C.672C672D-C1	0.3774326C 00
-0.4753631D 00	-0.252245CD CC	-C.7353225D-C1	0.6786178C 00

GRADIENT MATRIX DUE TO FJ

-0.2904210D 01 0.759198ED CC

DIRECT FEEDBACK MATRIX FJ

-0.1320730D 01 -0.2571279D C1

FIGURE 4-2, iv

Computer Print-Out for 3rd Order Example (continued)

```

RCCTS   REAL PART      IMAG. PART
        -0.164327E 01   C.C
        -0.164327E 01   C.1E6472D C1
        -0.164327E 01   -C.1E6472D C1

```

DETERMINE COMPENSATOR OF ORDER NFF = 1
AND ITERATE ON CONDITION-NUMBER.

MINIMUM STABILITY REQUIRED FOR TERMINATION -4.79999924

COMPENSATOR DESIGN - INITIAL VALUES. ISHIFT = 1

DIRECT FEEDBACK MATRIX FJ
-0.1231167D 01 -0.1554874D C2

COMPENSATOR OUTPUT MATRIX FH
-0.2834564D 01

COMPENSATOR INPUT MATRIX FG
0.2827917C 01 0.6671861D C1

COMPENSATOR MATRIX FF
-0.1913589D 01

```

RCCTS   REAL PART      IMAG. PART
        -0.900000E 01   C.C
        -0.400000E 01   C.1CCCCCD C1
        -0.400000E 01   -C.1CCCCCD C1
        -0.300000E 01   C.C

```

ALC = 1.000000COND.-NUMBER = 222.5C9CC3

ALC = 1.00000 ROOT1 = -C.3CCCCCD C1 COND.-NUMBER = 0.222909D 03 FUNCTION VALUE = 0.5937

ITERATION NUMBER KOUNT= 1

ALC = 1.00200 ROOT1 = -C.2555431D C1 COND.-NUMBER = 0.2212218D 03 FUNCTION VALUE = 0.5936

ITERATION NUMBER KOUNT= 1

ALC = 1.00400 ROOT1 = -C.255E62D C1 COND.-NUMBER = 0.2195591D 03 FUNCTION VALUE = 0.5936

ITERATION NUMBER KOUNT= 1

ALC = 1.00800 ROOT1 = -C.2557724D C1 COND.-NUMBER = 0.2163055D 03 FUNCTION VALUE = 0.5935

ITERATION NUMBER KOUNT= 1

FIGURE 4-2, v

Computer Print-Out for 3rd Order Example (continued)

```

ALO =      1.01601  ROOT1 = -C.2595448D C1  COND.-NUMBER =  0.2100730D 03  FUNCTION VALUE =  0.5933
ITERATION NUMBER  KOUNT=  1

ALO =      1.03202  ROOT1 = -C.2590897D C1  COND.-NUMBER =  0.1986163D 03  FUNCTION VALUE =  0.5930
ITERATION NUMBER  KOUNT=  1

ALO =      1.06403  ROOT1 = -C.2581793D C1  COND.-NUMBER =  0.1791212D 03  FUNCTION VALUE =  0.5923
ITERATION NUMBER  KOUNT=  1

ALO =      1.12807  ROOT1 = -C.2563586D C1  COND.-NUMBER =  0.1501772D 03  FUNCTION VALUE =  0.5910
ITERATION NUMBER  KOUNT=  1

ALO =      1.25614  ROOT1 = -C.2527172D C1  COND.-NUMBER =  0.1154399D 03  FUNCTION VALUE =  0.9888
ITERATION NUMBER  KOUNT=  1

ALO =      1.51228  ROOT1 = -C.2E54344D C1  COND.-NUMBER =  0.8297166D 02  FUNCTION VALUE =  0.9858
ITERATION NUMBER  KOUNT=  1

ALO =      1.46258  ROOT1 = -C.2E68476D C1  COND.-NUMBER =  0.8750232D 02  FUNCTION VALUE =  0.9863
ITERATION NUMBER  KOUNT=  1

ALO =      1.42769  ROOT1 = -C.2E78396D C1  COND.-NUMBER =  0.9094286D 02  FUNCTION VALUE =  0.9866
ITERATION NUMBER  KOUNT=  1

ALO =      1.40311  ROOT1 = -C.2E65383D C1  COND.-NUMBER =  0.9353388D 02  FUNCTION VALUE =  0.9869
ITERATION NUMBER  KOUNT=  1

ALO =      1.38567  ROOT1 = -C.2E9C343D C1  COND.-NUMBER =  0.9547476D 02  FUNCTION VALUE =  0.9871
ITERATION NUMBER  KOUNT=  1

ALO =      1.37315  ROOT1 = -C.2E939C3D C1  COND.-NUMBER =  0.9692715D 02  FUNCTION VALUE =  0.9872
ITERATION NUMBER  KOUNT=  1

```

FIGURE 4-2, vi

Computer Print-Out for 3rd Order Example (continued)


```

ALO = 1.36405 ROOT1 = -C.2E9E45CD C1 COND.-NUMBER = 0.9806241C 02 FUNCTION VALUE = 0.9873
ITERATION NUMBER KOUNT= 1

ALO = 1.37261 ROOT1 = -C.2E94C57D C1 COND.-NUMBER = 0.9699118C 02 FUNCTION VALUE = 0.9872
ITERATION NUMBER KOUNT= 1

ALO = 1.37211 ROOT1 = -C.2E942CCD C1 COND.-NUMBER = 0.9705061C 02 FUNCTION VALUE = 0.9872
ITERATION NUMBER KOUNT= 1

ALO = 1.37164 ROOT1 = -C.2E94332D C1 COND.-NUMBER = 0.9710586C 02 FUNCTION VALUE = 0.9872
ITERATION NUMBER KOUNT= 1

ALO = 1.37121 ROOT1 = -C.2E94455D C1 COND.-NUMBER = 0.9715729C 02 FUNCTION VALUE = 0.9872
ITERATION NUMBER KOUNT= 1

ALO = 1.37080 ROOT1 = -C.2E9457CD C1 COND.-NUMBER = 0.9720521C 02 FUNCTION VALUE = 0.9872
1

```

THE BELOW COMPENSATOR GUARANTEES STABILITY ONLY FOR A TOTAL
UNCERTAINTY OF 0.C277644E

COMPENSATOR DESIGN - FINAL VALUES.

DIRECT FEEDBACK MATRIX FJ
-0.1691622C 01 -0.152C581D C2

COMPENSATOR OUTPUT MATRIX FH
-0.2269640D 01

COMPENSATOR INPUT MATRIX FG
0.4328633D 01 0.E397799D C1

COMPENSATOR MATRIX FF
-0.1968502D 01

RCCTS	REAL PART	IMAG. PART
	-0.758312C 01	C.C
	-0.416168C 01	C.241976D C1
	-0.416168C 01	-0.241976D C1
	-0.288457C 01	C.C

FIGURE 4-2, vii

Computer Print-Out for 3rd Order Example (continued)

5. Explanation of Computer Print-Out, Fig. 4-2

The top half of example page i gives a partial listing of the input parameter.

The bottom half shows the beginning of the iterative gradient procedure to obtain a more stable system. The initial direct feedback gain matrix was inputted as -1. -2.5 resulting in a closed loop system whose least stable root has a maximum real part (MRP) of -.463. The following pages list the iterations produced by the gradient method in order to improve the system stability. The stability is improved such that the least stable closed-loop system root has a MRP = -1.643.

Since the closed-loop system without compensator (recall NFF was chosen 0), could not guarantee the required minimum stability $-AC = -.2$ in the presence of the chosen parameter uncertainty DA the program tries to determine a low-order compensator to obtain the required minimum stability. The compensator order is $NF = NF - NF = 1$. The compensator matrices FJ, FH, FG and FF, to realize the inputted set of initial system-plus compensator poles (EMS, EMF), are computed and listed in the top half of example page v.

The closed-loop system formed by the original system and the compensator yields a condition number of $\kappa = 222.909$ for $ALO = 1$ (ALO multiplies matrix FG and divides matrix FH).

In order to achieve a high relative stability the function

$$f_s(\underline{\lambda}, ALO) = 1 + \frac{\text{Re}(\lambda_{\max}) - ACC}{\kappa \|DA\| - ACC},$$

where $\underline{\lambda}$ are the eigenvalues of the closed-loop system, is minimized.

The iterations on ALO, the COND. - NUMBER and the FUNCTION VALUE are listed on pages v, vi and vii. It should be noted that the program terminates because of too many iterations. Although the function f_s achieves at some iteration step a value that is lower than the terminal FUNCTION VALUE the minimization procedure will in general, not be able to return to the lowest function value because the function gradient is synthetically computed.

The terminal design values and closed-loop system eigenvalues are listed at the bottom of example page vii.

6. Changes in Program

The COMPDES program taken to compute the 3rd order example is the same as listed in Appendix A of reference 1, with 2 exceptions. In the MAIN program,

- a) insert: READ (1, 10) IPASS between ISN0077 and ISN0078,
- b) take out ISN0140 (IF(IMATJ.EQ.0)GO TO 1099) and substitute IF(IPASS.GT.0)GO TO 1099.

Reference:

1. Lutz Willner, "Compensator Design for Low-Sensitivity Linear Time-Invariant Systems," Final Report - Vol. II, Contract No. NAS8-21131, Covering the period Nov. 4, 1969 - April 4, 1971, NASA.