AD-A106 777 ROSE-HULMAN INST OF TECH TERRE HAUTE IN F/G 14/2
DOCUMENTATION FOR PROGRAM SIMICE.(U)
MAR 81 W W BOWDEN AFOSR-80-0180
UNCLASSIFIED AFOSR-TR-81-0616 NL

END
DATE
FILMED
11-81
DTIC

AFOSR-TR- 81-0616

12

LEVEL II

Documentation For Program SIMICE

written by

Professor Warren W. Bowden

Rose-Hulman Institute of Technology

Terre Haute, IN 47803

Air Force Grant [redacted] AFOSR-80-0130

81 11 06 034

March 23, 1981

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER AFOSR-TR-81-0616	2. GOVT ACCESSION NO. AD-A106 777	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) DOCUMENTATION FOR PROGRAM SIMICE		5. TYPE OF REPORT & PERIOD COVERED FINAL 1 May 80-30 April 81
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Warren W. BOWDEN		8. CONTRACT OR GRANT NUMBER(s) AFOSR-80-0180
9. PERFORMING ORGANIZATION NAME AND ADDRESS ROSE-HULMAN INSTITUTE OF TECHNOLOGY TERRE HAUTE, IN 47803		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 2308/D9 61102F
11. CONTROLLING OFFICE NAME AND ADDRESS AIR FORCE OFFICE OF SCIENTIFIC RESEARCH BUILDING 410 BOLLING AFB, DC 20332		12. REPORT DATE 23 March 1981
		13. NUMBER OF PAGES 42
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

WIND TUNNEL TESTING
ICING SIMILITUDE
SIMICE

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A computer code (SIMICE) has been developed to assist in establishing correct test conditions and in extrapolating icing test data to full scale. The code calculates the conditions under which a model should be tested to maintain similitude between it and a prototype. In particular the matched quantities are the collection efficiency, heat transfer flux, and the flux of liquid water approaching the body under test. The SIMICE code structure and operation are described and a listing of the code provided.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

81 11 06 034

DOCUMENTATION FOR PROGRAM SIMICE

Introduction

Program SIMICE calculates the conditions under which a model should be tested to maintain similitude between it and a prototype. It attempts to maintain the following constant between the model and prototype:

(1) collection efficiency, Em,
(2) heat transfer flux, q/A, and
(3) the flux of liquid water approaching the body under test, grams/meter^2/sec.

The program solves for the liquid water content (LWC), the free stream velocity (V_o), and average droplet diameter (D) of the model from various input data. The equations modelling these phenomena are nonlinear in form and are solved by the well-known Broyden technique.

The user has flexibility in the data input and output. The inputs from a file are checked for reasonableness. The user must input initial values of the independent variables which are at least reasonable approximations to the final solution or the Broyden procedure may fail.

The program writes out the following:

(1) intermediate results from the BROYDEN program to the file DELBRO.DAT;
(2) any error messages or flags which may be encountered to the file SIMICY.ERR; and
(3) the input data and calculated values to the terminal or lineprinter.

This method of handling the output permits the user to obtain the principal output in a highly-presentable form while, at the same time, having available intermediate and other output for more detailed inspection.

The program is highly structured, and copiously supplied with COMMENT statements and, therefore, easily understood. It consists of the main program SIMICE and 14 subprograms. Program SIMICE

a) reads in the necessary starting data (See DATA INPUT section);
b) sets up the system of equations to be solved;
c) calls on subroutine DELBRO (the Broyden routine) which solves the equations; and
d) writes out the final results.

The subprograms (described in greater detail below) perform various chores such as

a) calculation of physical properties;
b) calculation of values of such quantities as the heat transfer coefficient, h_c, collection efficiently, E_m, etc.

This structure makes it possible for the user to easily replace current correlations with improved versions if and when they become available.

Mathematical Background

The basic procedure used to find the conditions for the model tests scaled-down from the prototype test results according to the principals of similitude is as follows:

1) write three equations governing the relevant physical phenomena and their scaled-down and
2) solve these equations for the three unknowns.

The three unknowns referred to are:

LWC_f = Liquid Water Content for the model.

V_{of} = Free stream velocity for the model.

D_f = Average droplet diameter for the model.

Other variables are brought in as parameters which can be changed as necessary by the user of the program.

The physical phenomena whose equalities are to be maintained between the model and the prototype are:

(1) the collection efficiency, E_m;
(2) the heat transfer flux, q/A; and
(3) the flux of liquid water approaching the body under test.

The equations which model these phenomena are nonlinear in form. They are solved by the Broyden technique (2) using a program developed by Dr. J.D. Perkins.

To facilitate the numerical work the program referred to above uses the ratios

$$X(1) = (LWC)_m/(LWC)_f$$
$$X(2) = D_m/D_f$$
$$X(3) = V_{om}/V_{of}$$

where the subscript f refers to the model and m to the prototype.

The three equations are developed in greater detail below.

Collection Efficiency = E_m. The collection efficiency E_m is a function of K_o(9):

$$E_m = f_1(K_o)$$

where

$$K_o = \lambda/\lambda_s K$$
$$\lambda/\lambda_s = f_2(Re_D)$$

and

$$K = \frac{2}{9}\,\frac{\rho_w V_o D^2}{\mu_o L_c}$$

We want $(E_m)_p = (E_m)_m$.

$(E_m)_f$ will equal $(E_m)_m$ if $(K_o)_f = (K_o)_m$.

In the equations below the subscript f(referring to the model)became p,

So,
$$1 = \frac{(\lambda/\lambda_s)_p}{(\lambda/\lambda_s)_m}\left(\frac{K_p}{K_m}\right)$$

or
$$\left(\frac{K_m}{K_p}\right)\frac{(\lambda/\lambda_s)_m}{(\lambda/\lambda_s)_p} = 1$$

or
$$\frac{V_m\, D_m^2\, L_{cp}}{V_p\, D_p^2\, L_{cm}}\;\frac{f_2(N_{RoD})_m}{f_2(N_{RoD})_p} = 1$$

Let $L_{cm}/L_{cp} = R$ and the above equation becomes

$$R = \left(\frac{V_m}{V_p}\right)\left(\frac{D_m}{D_p}\right)^2\frac{f(N_{Re})_m}{f(N_{Re})_p}$$

or

$$\boxed{F_1 = R - \left(\frac{V_m}{V_p}\right)\left(\frac{D_m}{D_p}\right)^2\frac{f(N_{ReD})_m}{f(N_{ReD})_p} = 0}$$

<u>Heat Transfer Flux = q/A</u> . After Messinger(6), the heat flux to or from the body under study ($t_s \leq 32°F$) :

$$q/A = h_c[(t_{s\rho}-t_\infty - \frac{rV_\infty^2}{2g_cJC_{pa}}) + 2.90L_s\,(\frac{P_{si}-P_\infty}{P_o})]$$

$$+ R_w\,[C_w(32-t_\infty) - 144 - C_i(32-t_{se}) - \frac{V_\infty^2}{2g_cJ}]$$

We want

$$(q/A)_p = (q/A)_m$$

or $$(q/A)_m\,/\,(q/A)_p = 1$$

or $$\boxed{F_2 = 1.0 - (q/A)_m/\,(q/A)_p = 0}$$

<u>Water Flux = $(LWC)(V_o)$</u>

We want

$$(LWC)_p(V_o)_p = (LWC)_m(V_o)_m$$

or

$$\boxed{F_3 = 1. - \left(\frac{(LWC)_m}{(LWC)_p}\right) \times \left(\frac{V_{om}}{V_{op}}\right) = 0}$$

The above equations appear as follows in the program:

QAM = $(q/A)_m$

RAM = $(\lambda/\lambda_s)_m$

F(1) = 1.0 - QAM/$(q/A)_p$

F(2) = R - X(3)*X(2)**2*RAM/$(\lambda/\lambda_s)_p$

f(3) = 1. - X(1)*X(3)

Trial values of X(1), X(2) and X(3) are continued until these three functions are arbitrarily close to zero.

SUBPROGRAMS

Each subprogram is discussed briefly below.

SUBROUTINES DELBRO, BOUNDR, and GJINV

These three subroutines together solve systems of nonlinear equations iteratively. They are based on the Broyden technique (2) as modified by the More-Cosnard procedure(8). The program was written by Dr. J. D. Perkins, Imperial College, London.

The user must supply the set of functions F(I), set up the initial values of the independent variables X(I), and input boundary values of , BOUND(I), within which the program attempts to find a solution. Other input parameters are explained by rather detailed comments in DELBRO.

DELBRO may fail to find a solution for a variety of reasons:

(1) Initial Jacobian approximation may be singular because of:

(a) poor scaling of variables;
(b) equations are not independent;
(c) poor initial guesses.

The program works best if the independent variables are kept between -1 and +1 and the functions F(I) are kept near 1.0. Dr. Perkins (7) has suggested transformations of the type

$$X_i^* = (X_i - a)/b$$

$$f_i^* = f_i/c$$

(2) Bounds may impede progress towards a solution. It may be that a solution does not exist within the bounds, or, a poor starting point may have been chosen.

(3) The dimensions NW, NB are too small. See listing of program for requirements.

(4) The program may not have converged to the precision specified (ERROR) within the maximum number of iterations specified (MAXIT).

The subroutine BOUNDR checks the values of X_i calculated against the boundary values supplied by the program, sets INFO = (KOND) equal to 5, writes out a warning and returns control to DELBRO which, in turn, returns control to SIMICE.

Subroutine GJINV inverts the preliminary Jacobian matrix.

FUNCTIONS DENL, PSØ, CW

Function DENL calculates the density of liquid water (g/cc) from -50°C to +50°C. It is based on data and equations given by KELL (4). Below -20°C the results are extrapolated.

Function PSØ calculates the vapor pressure of ice (Kilopascals). It is based on the work of Arnold Wexler(13).

Function CW calculates the heat capacity of liquid water in the range -50°C to +40°C. It is based on data from Smithsonian Meteorological Tables (10).

FUNCTIONS MUMIX, KMIX, NPRMIX

Function MUMIX calculations the viscosity of air-water vapor mixtures. It is based on equations (9-3.9) and (9-5.4) of Reid, Prausnitz and Sherwood(9).

Function KMIX calculates the thermal conductivity of air-water vapor mixtures from equations (10-4.1), (10-6.1), (10-6.2) and (10-6.3) of Reid, Prausnitz and Sherwood(9).

Function NPRMIX calculates the Prandtl number of air-water vapor mixtures. It uses KMIX, MUMIX and heat capacity data on air from (5) and on water vapor from (3).

FUNCTIONS LAMDAS, EM, EMC and EMS

Function subprogram LAMDAS calculates a value of λ/λ_s for a given value of the drop Reynolds number N_{ReD}. The equations it uses are based on data tabulated in reference (11).

Function EM calculates the collection efficiency, E_m. It first calculates a value of $K = 2/9\rho_w V_o D^2/(\mu_a L_c)$, the Reynolds number and K_o = LAMDAS(N_{Re})*K. It then calls on EMC to calculate E_m for a cylinder, EMS to calculate E_m for a sphere, or EMØØ12 to calculate E_m for a NASA 0012 airfoil.

Functions EMC, EMS and EMØØ12 are based on equations obtained from curves given in references (1 , 11).

This set of Programs should be replaced by a procedure which integrates the basic differential equations (12) to obtain the collection efficiency, E_m .

FUNCTIONS HC, QA

Function HC calculates the heat transfer coefficient h_c from equations given in (14).

Function QA evaluates the heat flux as given by equations in Messinger (6).

References

1. Bowden, D.T., A.E. Gensemer and C.A. Skeen, Engineering Summary of Airframe Icing Technical Data, FAA ADS-4 (1964).

2. Broyden, C.G., Math. of Computation, 19, 577-593 (1965).

3. JANAF Thermochemical Tables (NBS), 2nd Edition (1970).

4. Kell, G.S., J. Chem. Eng. Data, 12, 66-69 (1967).

5. Kothandaraman, C.P. and S. Suframanyan, Heat and Mass Transfer Data Book, Wiley (1975).

6. Messinger, B.L., J. of Aeronautical Sciences, 20 (#1), 29-42 (1953).

7. Perkins, J.D., Private Communication, August 1980.

8. Pfeifer, G.D. and G.P. Maier, Engineering Summary of Powerplant Icing Technical Data, FAA-RD-77-76 (AD/A-045 087) (1977).

9. Reid, R.C., J.M. Prausnitz and T.K. Sherwood, The Properties of Gases and Liquids, 3rd Edition, McGraw-Hill (1977).

10. Smithsonian Meteorological Tables (1951).

11. Sogin, H.H., A Design Manual For Thermal Anti-Icing Systems, WADC-TR 54-313 (AD 90156) (1954).

12. Tribus, Myron, G.B.W. Young, and L.M.K. Boelter, Trans. A.S.M.E., 70, 917-982 (1928).

13. Wexler, Arnold, J. of Research of the National Bureau of Standards, 81A, 5-20 (1977).

14. Whitaker, S., Elementary Heat Transfer Analysis, Pergamon (1976).

DATA INPUT

Data input is from a file named ICE·DAT. The input is unformatted: each number should be separated from others by commas or at least one space. Form and meaning of the input are shown in the following table:

Record	Variable Name	Variable Type	Significance
1	IB	Integer	Body type = 1 for cylinder = 2 for sphere = 3 for 0012 NASA Airfoil
1	IH	Integer	Heating condition = 1 No heat input = 2 Heat input
1	R	Real	Scale-up Ratio = $(LC)_m/(LC)_p$
1	LC	Real	Prototype characteristic dimension, cm
1	IPRINT	Integer	Variable controlling printout by DELBRO = -1 No printing = 0 Final results only = 1 X + F on every iteration = 2 X + F + W on every iteration
2	TMS	Real	Prototype surface temperature, °C
2	TMØ	Real	Prototype free stream temperature, °C
2	PMØ	Real	Prototype free stream total pressure, Kp
2	VMØ	Real	Prototype free stream velocity, meter/sec.
2	OMEGA	Real	Model(and prototype)specific humidity = (Kg water vapor)/(Kg moist air)
2	LWC	Real	Liquid Water Content, g/(cubic meter)
2	DM	Real	Model droplet diameter, microns
2	LIM	Integer	Program Control parameter: If LIM = -1 --- end of run ≠ -1 --- expects another set of data
3	TFS	Real	Model surface temperature, °C
	TFØ	Real	Model free stream temperatue, °C
	PPØ	Real	Model free stream pressure, KP
4	X (1) X (2) X (3)	Real	Initial values of ratio variables for BROYDEN

Record	Variable Name	Variable Type	Significance
5 & 6	BOUND(1)		
	BOUND(2)	Real	Boundary values for the X(I) variables:
	------		Lower bounds in BOUND(1), BOUND(2), BOUND(3);
	BOUND(6)		Upper bounds in BOUND(4), BOUND(5), BOUND(6)

A sample set of input data is given on page 10 .

```
[illegible]
  1 1 2.38 4.76 1
0.0 -10. 47. 116.79 0.004 2.05 32.03 1
0.0 -10. 47.
0.71 1.56 1.25
.5 .3 .5
2. 2. 2.
  1 1 2.38 4.76 1
0.0 -10. 47. 120. 0.004 2.05 32.03 1
0.0 -10. 47.
0.71 1.56 1.25
.5 .3 .5
2. 2. 2.
1 1 2.38 4.76 1
0.0 -10. 47. 115. 0.004 2.05 32.03 1
0.0 -10. 47.
0.71 1.56 1.25
0.5 0.3 0.5
2. 2. 2.
1 1 2.38 4.76 1
0.0 -10. 47. 110. 0.004 2.05 32.03 1
0.0 -10. 47.
0.71 1.56 1.25
0.5 0.3 0.5
2. 2. 2.
1 1 2.38 4.76 1
0.0 -10. 47. 100. 0.004 2.05 32.03 -1
0.0 -10. 47.
0.71 1.56 1.25
0.5 0.3 0.5
2. 2. 2.
```

DATA OUTPUT

Output from this program is illustrated on the following pages. The first set of output is the principal output which comes out to a terminal or line printer. The second set shows the intermediate results which can be written out to the file DELBRO.DAT. The third set shows various intermediate results which can be written out to the file SIMICY.ERR.

```
                         PROGRAM SIMICE
                                               28-JAN-81

  RUN NUMBER=  1 BROYDEN CONVERGED                    AFTER     15 CALLS
              MODEL AND PROTOTYPE BODY:CYLINDER
              PROTOTYPE(INPUT) CONDITIONS:
 LC = PROTOTYPE CHARACTERISTIC DIMENSION=      4.76CM
 TMS= PROTOTYPE SURFACE TEMPERATURE=           0.00C
 TMO= PROTOTYPE FREE STREAM TEMPERATURE=     -10.00C
 PMO= PROTOTYPE FREE STREAM TOTAL PRESSURE=  47.00KP
 VMO= PROTOTYPE FREE STREAM VELOCITY=       116.79METER/SEC
 LWC= PROTOTYPE LIQUID WATER CONTENT=          2.05GRAMS/CUBIC METER
 DM = PROTOTYPE DROPLET DIAMETER=             32.03MICRONS

               MODEL CONDITONS:
 LCF= MODEL CHARACTERISTIC DIMENSION=          2.00CM
 TFS= MODEL SURFACE TEMPERATURE=               0.00C
 TFO= MODEL STREAM TEMPERATURE=              -10.00C
 PFO= MODEL STREAM TOTAL PRESSURE=            47.00KP
 VPO= MODEL FREE STREAM VELOCITY=             96.01METER/SEC
 LWCF= MODEL LIQUID WATER CONTENT=             2.49GRAMS/CUBIC METER
 DF=MODEL WATER DROPLET DIAMETER=             20.00MICRONS


 OMEGA=      0.4000E-02(KG H2O VAPOR)/(KG MOIST AIR)


  RUN NUMBER=  2 BROYDEN CONVERGED                    AFTER     15 CALLS
              MODEL AND PROTOTYPE BODY:CYLINDER
              PROTOTYPE(INPUT) CONDITIONS:
 LC = PROTOTYPE CHARACTERISTIC DIMENSION=      4.76CM
 TMS= PROTOTYPE SURFACE TEMPERATURE=           0.00C
 TMO= PROTOTYPE FREE STREAM TEMPERATURE=     -10.00C
 PMO= PROTOTYPE FREE STREAM TOTAL PRESSURE=  47.00KP
 VMO= PROTOTYPE FREE STREAM VELOCITY-       120.00METER/SEC
 LWC= PROTOTYPE LIQUID WATER CONTENT=          2.05GRAMS/CUBIC METER
 DM = PROTOTYPE DROPLET DIAMETER=             32.03MICRONS

               MODEL CONDITONS:
 LCF= MODEL CHARACTERISTIC DIMENSION=          2.00CM
 TFS= MODEL SURFACE TEMPERATURE=               0.00C
 TFO= MODEL STREAM TEMPERATURE=              -10.00C
 PFO= MODEL STREAM TOTAL PRESSURE=            47.00KP
 VPO= MODEL FREE STREAM VELOCITY=             98.93METER/SEC
 LWCF= MODEL LIQUID WATER CONTENT=             2.49GRAMS/CUBIC METER
 DF=MODEL WATER DROPLET DIAMETER=             19.97MICRONS


 OMEGA=      0.4000E-02(KG H2O VAPOR)/(KG MOIST AIR)


  RUN NUMBER=  3 BROYDEN CONVERGED                    AFTER     14 CALLS
              MODEL AND PROTOTYPE BODY:CYLINDER
              PROTOTYPE(INPUT) CONDITIONS:
```

```
TMO= PROTOTYPE FREE STREAM TEMPERATURE=   -10.00C
PMO= PROTOTYPE FREE STREAM TOTAL PRESSURE= 47.00KP
VMO= PROTOTYPE FREE STREAM VELOCITY=      115.00METER/SEC
LWC= PROTOTYPE LIQUID WATER CONTENT=        2.05GRAMS/CUBIC METER
DM = PROTOTYPE DROPLET DIAMETER=           32.03MICRONS

                MODEL CONDITONS:


LCF= MODEL CHARACTERISTIC DIMENSION=        2.00CM
TFS= MODEL SURFACE TEMPERATURE=             0.00C
TFO= MODEL STREAM TEMPERATURE=            -10.00C
PFO= MODEL STREAM TOTAL PRESSURE=          47.00KP
VFO= MODEL FREE STREAM VELOCITY=           94.37METER/SEC
LWCF= MODEL LIQUID WATER CONTENT=           2.50GRAMS/CUBIC METER
DF=MODEL WATER DROPLET DIAMETER=           20.02MICRONS


OMEGA=     0.4000E-02(KG H2O VAPOR)/(KG MOIST AIR)


  RUN NUMBER=  4 BROYDEN CONVERGED                    AFTER    14 CALLS
               MODEL AND PROTOTYPE BODY:CYLINDER
               PROTOTYPE(INPUT) CONDITIONS:
LC = PROTOTYPE CHARACTERISTIC DIMENSION=    4.76CM
TMS= PROTOTYPE SURFACE TEMPERATURE=         0.00C
TMO= PROTOTYPE FREE STREAM TEMPERATURE=   -10.00C
PMO= PROTOTYPE FREE STREAM TOTAL PRESSURE= 47.00KP
VMO= PROTOTYPE FREE STREAM VELOCITY=      110.00METER/SEC
LWC= PROTOTYPE LIQUID WATER CONTENT=        2.05GRAMS/CUBIC METER
DM = PROTOTYPE DROPLET DIAMETER=           32.03MICRONS

                MODEL CONDITONS:
LCF= MODEL CHARACTERISTIC DIMENSION=        2.00CM
TFS= MODEL SURFACE TEMPERATURE=             0.00C
TFO= MODEL STREAM TEMPERATURE=            -10.00C
PFO= MODEL STREAM TOTAL PRESSURE=          47.00KP
VFO= MODEL FREE STREAM VELOCITY=           89.80METER/SEC
LWCF= MODEL LIQUID WATER CONTENT=           2.51GRAMS/CUBIC METER
DF=MODEL WATER DROPLET DIAMETER=           20.09MICRONS


OMEGA=     0.4000E-02(KG H2O VAPOR)/(KG MOIST AIR)


  RUN NUMBER=  5 BROYDEN CONVERGED                    AFTER    13 CALLS
               MODEL AND PROTOTYPE BODY:CYLINDER
               PROTOTYPE(INPUT) CONDITIONS:
LC = PROTOTYPE CHARACTERISTIC DIMENSION=    4.76CM
TMS= PROTOTYPE SURFACE TEMPERATURE=         0.00C
TMO= PROTOTYPE FREE STREAM TEMPERATURE=   -10.00C
PMO= PROTOTYPE FREE STREAM TOTAL PRESSURE= 47.00KP
VMO= PROTOTYPE FREE STREAM VELOCITY=      100.00METER/SEC
LWC= PROTOTYPE LIQUID WATER CONTENT=        2.05GRAMS/CUBIC METER
DM = PROTOTYPE DROPLET DIAMETER=           32.03MICRONS

                MODEL CONDITONS:
LCF= MODEL CHARACTERISTIC DIMENSION=        2.00CM
TFS= MODEL SURFACE TEMPERATURE=             0.00C
TFO= MODEL STREAM TEMPERATURE=            -10.00C
PFO= MODEL STREAM TOTAL PRESSURE=          47.00KP
```

DELBRO.DAT (37;

```
DELBRO*** NUMBER OF CALLS=    1
X        0.7100E+00      0.1560E+01       0.1250E+01
F       -0.4752E-01      0.5996E-01       0.1125E+00
DELBRO*** NUMBER OF CALLS=    2
X        0.7171E+00      0.1560E+01       0.1250E+01
F       -0.4755E-01      0.5996E-01       0.1036E+00
DELBRO*** NUMBER OF CALLS=    3
X        0.7100E+00      0.1576E+01       0.1250E+01
F       -0.4753E-01      0.2224E-01       0.1125E+00
DELBRO*** NUMBER OF CALLS=    4
X        0.7100E+00      0.1560E+01       0.1262E+01
F       -0.6543E-01      0.4558E-01       0.1036E+00
DELBRO*** NUMBER OF CALLS=    5
X        0.6164E+00      0.1603E+01       0.1297E+01
F       -0.1148E+00     -0.1020E+00       0.2004E+00
DELBRO*** NUMBER OF CALLS=    6
X        0.7149E+00      0.1409E+01       0.1225E+01
F       -0.1110E-01      0.4363E+00       0.1246E+00
DELBRO*** NUMBER OF CALLS=    7
X        0.5000E+00      0.3000E+00       0.1364E+01
F       -0.2096E+00      0.2203E+01       0.3178E+00
DELBRO*** NUMBER OF CALLS=    8
X        0.8279E+00      0.1892E+01       0.9930E+00
F        0.3105E+00     -0.3695E+00       0.1779E+00
DELBRO*** NUMBER OF CALLS=    9
X        0.8634E+00      0.2000E+01       0.5000E+00
F        0.8450E+00      0.3800E+00       0.5683E+00
DELBRO*** NUMBER OF CALLS=   10
X        0.8171E+00      0.1601E+01       0.1146E+01
F        0.9719E-01      0.8398E-01       0.6199E-01
DELBRO*** NUMBER OF CALLS=   11
X        0.8044E+00      0.1563E+01       0.1232E+01
F       -0.2213E-01      0.7344E-01       0.8951E-02
DELBRO*** NUMBER OF CALLS=   12
X        0.8221E+00      0.1595E+01       0.1248E+01
F       -0.4463E-01     -0.2329E-01      -0.2576E-01
DELBRO*** NUMBER OF CALLS=   13
X        0.8218E+00      0.1601E+01       0.1215E+01
F        0.2083E-02      0.1521E-02       0.1515E-02
DELBRO*** NUMBER OF CALLS=   14
X        0.8221E+00      0.1601E+01       0.1217E+01
F       -0.8571E-04     -0.2750E-03      -0.1145E-03
DELBRO*** NUMBER OF CALLS=   15
X        0.8220E+00      0.1601E+01       0.1216E+01
F       -0.2742E-05     -0.4408E-05      -0.3295E-05
***DELBRO***CONVERGENCE AFTER   15 CALLS
DELBRO*** NUMBER OF CALLS=    1
X        0.7100E+00      0.1560E+01       0.1250E+01
F       -0.5368E-01      0.6258E-01       0.1125E+00
DELBRO*** NUMBER OF CALLS=    2
X        0.7171E+00      0.1560E+01       0.1250E+01
F       -0.5370E-01      0.6258E-01       0.1036E+00
```

```
X        0.7100E+00      0.1576E+01      0.1250E+01
F       -0.5368E-01      0.2497E-01      0.1125E+00
DELBRO*** NUMBER OF CALLS=    4
X        0.7100E+00      0.1560E+01      0.1262E+01
F       -0.7197E-01      0.4828E-01      0.1036E+00
DELBRO*** NUMBER OF CALLS=    5
X        0.6164E+00      0.1604E+01      0.1303E+01
F       -0.1315E+00     -0.1075E+00      0.1940E+00
DELBRO*** NUMBER OF CALLS=    6
X        0.7090E+00      0.1403E+01      0.1220E+01
F       -0.1023E-01      0.4541E+00      0.1348E+00
DELBRO*** NUMBER OF CALLS=    7
X        0.5000E+00      0.3000E+00      0.1400E+01
F       -0.2720E+00      0.2200E+01      0.3000E+00
DELBRO*** NUMBER OF CALLS=    8
X        0.7914E+00      0.1946E+01      0.8868E+00
F        0.4500E+00     -0.3020E+00      0.2982E+00
DELBRO*** NUMBER OF CALLS=    9
X        0.6035E+00      0.2000E+01      0.5000E+00
F        0.8469E+00      0.3800E+00      0.6982E+00
DELBRO*** NUMBER OF CALLS=   10
X        0.8712E+00      0.1569E+01      0.1127E+01
F        0.1244E+00      0.1859E+00      0.1845E-01
DELBRO*** NUMBER OF CALLS=   11
X        0.7973E+00      0.1522E+01      0.1200E+01
F        0.1850E-01      0.2087E+00      0.4296E-01
DELBRO*** NUMBER OF CALLS=   12
X        0.8236E+00      0.1568E+01      0.1251E+01
F       -0.5540E-01      0.4222E-01     -0.3027E-01
DELBRO*** NUMBER OF CALLS=   13
X        0.8172E+00      0.1597E+01      0.1210E+01
F        0.4136E-02      0.2085E-01      0.1105E-01
DELBRO*** NUMBER OF CALLS=   14
X        0.8251E+00      0.1605E+01      0.1213E+01
F        0.1359E-03     -0.1965E-02     -0.6745E-03
DELBRO*** NUMBER OF CALLS=   15
X        0.8244E+00      0.1604E+01      0.1213E+01
F        0.2921E-05      0.3250E-05      0.7837E-05
***DELBRO***CONVERGENCE AFTER    15 CALLS
DELBRO*** NUMBER OF CALLS=    1
X        0.7100E+00      0.1560E+01      0.1250E+01
F       -0.4400E-01      0.5840E-01      0.1125E+00
DELBRO*** NUMBER OF CALLS=    2
X        0.7171E+00      0.1560E+01      0.1250E+01
F       -0.4403E-01      0.5840E-01      0.1036E+00
DELBRO*** NUMBER OF CALLS=    3
X        0.7100E+00      0.1576E+01      0.1250E+01
F       -0.4401E-01      0.2062E-01      0.1125E+00
DELBRO*** NUMBER OF CALLS=    4
X        0.7100E+00      0.1560E+01      0.1262E+01
F       -0.6169E-01      0.4398E-01      0.1036E+00
DELBRO*** NUMBER OF CALLS=    5
X        0.6152E+00      0.1603E+01      0.1294E+01
F       -0.1054E+00     -0.9876E-01      0.2041E+00
DELBRO*** NUMBER OF CALLS=    6
X        0.7181E+00      0.1412E+01      0.1227E+01
F       -0.1144E-01      0.4255E+00      0.1189E+00
DELBRO*** NUMBER OF CALLS=    7
X        0.5000E+00      0.3000E+00      0.1347E+01
F       -0.1799E+00      0.2204E+01      0.3263E+00
DELBRO*** NUMBER OF CALLS=    8
X        0.8380E+00      0.1860E+01      0.1044E+01
F        0.2425E+00     -0.3796E+00      0.1254E+00
DELBRO*** NUMBER OF CALLS=    9
X        0.8959E+00      0.2000E+01      0.5000E+00
F        0.8439E+00      0.3800E+00      0.5521E+00
DELBRO*** NUMBER OF CALLS=   10
```

```
X        0.8254E+00      0.1610E+01      0.1144E+01
F        0.1048E+00      0.6814E-01      0.5819E-01
DELBRO*** NUMBER OF CALLS=  11
X        0.8011E+00      0.1562E+01      0.1224E+01
F       -0.7897E-02      0.8256E-01      0.1929E-01
DELBRO*** NUMBER OF CALLS=  12
X        0.8184E+00      0.1591E+01      0.1256E+01
F       -0.5258E-01     -0.2435E-01     -0.2775E-01
DELBRO*** NUMBER OF CALLS=  13
X        0.8204E+00      0.1599E+01      0.1222E+01
F       -0.5505E-02     -0.2755E-02     -0.2904E-02
DELBRO*** NUMBER OF CALLS=  14
X        0.8206E+00      0.1600E+01      0.1219E+01
F       -0.3564E-04     -0.2451E-04     -0.2616E-04
***DELBRO***CONVERGENCE AFTER   14 CALLS
DELBRO*** NUMBER OF CALLS=   1
X        0.7100E+00      0.1560E+01      0.1250E+01
F       -0.3381E-01      0.5366E-01      0.1125E+00
DELBRO*** NUMBER OF CALLS=   2
X        0.7171E+00      0.1560E+01      0.1250E+01
F       -0.3384E-01      0.5366E-01      0.1036E+00
DELBRO*** NUMBER OF CALLS=   3
X        0.7100E+00      0.1576E+01      0.1250E+01
F       -0.3382E-01      0.1569E-01      0.1125E+00
DELBRO*** NUMBER OF CALLS=   4
X        0.7100E+00      0.1560E+01      0.1262E+01
F       -0.5086E-01      0.3910E-01      0.1036E+00
DELBRO*** NUMBER OF CALLS=   5
X        0.6116E+00      0.1602E+01      0.1283E+01
F       -0.7902E-01     -0.8943E-01      0.2150E+00
DELBRO*** NUMBER OF CALLS=   6
X        0.7272E+00      0.1423E+01      0.1234E+01
F       -0.1158E-01      0.3923E+00      0.1028E+00
DELBRO*** NUMBER OF CALLS=   7
X        0.5000E+00      0.5738E+00      0.1310E+01
F       -0.1132E+00      0.1893E+01      0.3452E+00
DELBRO*** NUMBER OF CALLS=   8
X        0.8318E+00      0.1715E+01      0.1156E+01
F        0.9399E-01     -0.2042E+00      0.3869E-01
DELBRO*** NUMBER OF CALLS=   9
X        0.8707E+00      0.1753E+01      0.9958E+00
F        0.3069E+00     -0.5956E-01      0.1330E+00
DELBRO*** NUMBER OF CALLS=  10
X        0.8311E+00      0.1617E+01      0.1203E+01
F        0.2964E-01     -0.2890E-01      0.7753E-04
DELBRO*** NUMBER OF CALLS=  11
X        0.8170E+00      0.1594E+01      0.1208E+01
F        0.2251E-01      0.2073E-01      0.1276E-01
DELBRO*** NUMBER OF CALLS=  12
X        0.8125E+00      0.1589E+01      0.1238E+01
F       -0.1818E-01     -0.2707E-02     -0.6028E-02
DELBRO*** NUMBER OF CALLS=  13
X        0.8163E+00      0.1594E+01      0.1226E+01
F       -0.9429E-03     -0.3298E-03     -0.4121E-03
DELBRO*** NUMBER OF CALLS=  14
X        0.8164E+00      0.1594E+01      0.1225E+01
F        0.5430E-04      0.6326E-04      0.3732E-04
***DELBRO***CONVERGENCE AFTER   14 CALLS
DELBRO*** NUMBER OF CALLS=   1
X        0.7100E+00      0.1560E+01      0.1250E+01
F       -0.1184E-01      0.4227E-01      0.1125E+00
DELBRO*** NUMBER OF CALLS=   2
X        0.7171E+00      0.1560E+01      0.1250E+01
F       -0.1187E-01      0.4227E-01      0.1036E+00
DELBRO*** NUMBER OF CALLS=   3
X        0.7100E+00      0.1576E+01      0.1250E+01
F       -0.1184E-01      0.3870E-02      0.1125E+00
DELBRO*** NUMBER OF CALLS=   4
```

```
X       0.7100E+00     0.1560E+01     0.1262E+01
F      -0.2754E-01     0.2740E-01     0.1036E+00
DELBRO*** NUMBER OF CALLS=    5
X       0.6029E+00     0.1599E+01     0.1262E+01
F      -0.2578E-01    -0.6822E-01     0.2395E+00
DELBRO*** NUMBER OF CALLS=    6
X       0.7455E+00     0.1448E+01     0.1246E+01
F      -0.6636E-02     0.3150E+00     0.7125E-01
DELBRO*** NUMBER OF CALLS=    7
X       0.5565E+00     0.1009E+01     0.1264E+01
F      -0.2890E-01     0.1215E+01     0.2963E+00
DELBRO*** NUMBER OF CALLS=    8
X       0.8042E+00     0.1598E+01     0.1240E+01
F       0.7776E-03    -0.3771E-01     0.3089E-02
DELBRO*** NUMBER OF CALLS=    9
X       0.8090E+00     0.1586E+01     0.1238E+01
F       0.3029E-02    -0.8006E-02    -0.1474E-02
DELBRO*** NUMBER OF CALLS=   10
X       0.8083E+00     0.1584E+01     0.1234E+01
F       0.7504E-02     0.2266E-02     0.2225E-02
DELBRO*** NUMBER OF CALLS=   11
X       0.8158E+00     0.1596E+01     0.1220E+01
F       0.2561E-01    -0.1036E-01     0.4785E-02
DELBRO*** NUMBER OF CALLS=   12
X       0.8064E+00     0.1582E+01     0.1240E+01
F       0.7311E-03     0.1399E-03     0.2300E-03
DELBRO*** NUMBER OF CALLS=   13
X       0.8063E+00     0.1582E+01     0.1240E+01
F       0.3982E-04    -0.2299E-05     0.1249E-04
***DELBRO***CONVERGENCE AFTER   13 CALLS
```

```
SIMICY.ERR( 37,
----------------------------------------------------------------------------------------------------

EM:NRE,K,KO=      0.1391E+03      0.1671E+02      0.5102E+01
EMC:KO,EM=      0.5102E+01      0.8429E+00
NRE OUT OF RANGE IN HC=      0.20667E+06
KALL=  1 X=  0.7100E+00  0.1560E+01  0.1250E+01
EM:NRE,K,KO=      0.7132E+02      0.1307E+02      0.5234E+01
EMC:KO,EM=      0.5234E+01      0.8460E+00
KALL=  2 X=  0.7171E+00  0.1560E+01  0.1250E+01
EM:NRE,K,KO=      0.7132E+02      0.1307E+02      0.5234E+01
EMC:KO,EM=      0.5234E+01      0.8460E+00
KALL=  3 X=  0.7100E+00  0.1576E+01  0.1250E+01
EM:NRE,K,KO=      0.7061E+02      0.1281E+02      0.5151E+01
EMC:KO,EM=      0.5151E+01      0.8441E+00
KALL=  4 X=  0.7100E+00  0.1560E+01  0.1262E+01
EM:NRE,K,KO=      0.7061E+02      0.1294E+02      0.5202E+01
EMC:KO,EM=      0.5202E+01      0.8453E+00
KALL=  5 X=  0.6164E+00  0.1603E+01  0.1297E+01
EM:NRE,K,KO=      0.6687E+02      0.1193E+02      0.4893E+01
EMC:KO,EM=      0.4893E+01      0.8380E+00
KALL=  6 X=  0.7149E+00  0.1409E+01  0.1225E+01
EM:NRE,K,KO=      0.8060E+02      0.1636E+02      0.6248E+01
EMC:KO,EM=      0.6248E+01      0.8658E+00
KALL=  7 X=  0.5000E+00  0.3000E+00  0.1364E+01
EM:NRE,K,KO=      0.3398E+03      0.3239E+03      0.6860E+02
EMC:KO=      0.6860E+02
EMC:KO,EM=      0.6860E+02      0.9776E+00
KALL=  8 X=  0.8279E+00  0.1892E+01  0.9930E+00
EM:NRE,K,KO=      0.7403E+02      0.1119E+02      0.4417E+01
EMC:KO,EM=      0.4417E+01      0.8256E+00
KALL=  9 X=  0.8634E+00  0.2000E+01  0.5000E+00
EM:NRE,K,KO=      0.1391E+03      0.1988E+02      0.6072E+01
EMC:KO,EM=      0.6072E+01      0.8652E+00
NRE OUT OF RANGE IN HC=      0.17367E+06
KALL= 10 X=  0.8171E+00  0.1601E+01  0.1148E+01
EM:NRE,K,KO=      0.7566E+02      0.1351E+02      0.5289E+01
EMC:KO,EM=      0.5289E+01      0.8474E+00
KALL= 11 X=  0.8044E+00  0.1563E+01  0.1232E+01
EM:NRE,K,KO=      0.7222E+02      0.1321E+02      0.5265E+01
EMC:KO,EM=      0.5265E+01      0.8468E+00
KALL= 12 X=  0.8221E+00  0.1595E+01  0.1248E+01
EM:NRE,K,KO=      0.6987E+02      0.1252E+02      0.5053E+01
EMC:KO,EM=      0.5053E+01      0.8418E+00
KALL= 13 X=  0.8218E+00  0.1601E+01  0.1215E+01
EM:NRE,K,KO=      0.7147E+02      0.1276E+02      0.5106E+01
EMC:KO,EM=      0.5106E+01      0.8430E+00
KALL= 14 X=  0.8221E+00  0.1601E+01  0.1217E+01
EM:NRE,K,KO=      0.7139E+02      0.1275E+02      0.5102E+01
EMC:KO,EM=      0.5102E+01      0.8429E+00
KALL= 15 X=  0.8220E+00  0.1601E+01  0.1216E+01
EM:NRE,K,KO=      0.7139E+02      0.1275E+02      0.5102E+01
EMC:KO,EM=      0.5102E+01      0.8429E+00
KALL,KOUNT= 15  2
EM:NRE,K,KO=      0.[illegible]E+03      0.1717E+02      0.5183E+01
```

```
NRE OUT OF RANGE IN HC=     0.21235E+06
KALL=  1 X=  0.7100E+00  0.1560E+01  0.1250E+01
EM:NRE,K,KO=       0.7328E+02       0.1343E+02       0.5323E+01
EMC:KO,EM=       0.5323E+01       0.8482E+00
KALL=  2 X=  0.7171E+00  0.1560E+01  0.1250E+01
EM:NRE,K,KO=       0.7328E+02       0.1343E+02       0.5323E+01
EMC:KO,EM=       0.5323E+01       0.8482E+00
KALL=  3 X=  0.7100E+00  0.1576E+01  0.1250E+01
EM:NRE,K,KO=       0.7256E+02       0.1317E+02       0.5238E+01
EMC:KO,EM=       0.5238E+01       0.8461E+00
KALL=  4 X=  0.7100E+00  0.1560E+01  0.1262E+01
EM:NRE,K,KO=       0.7256E+02       0.1330E+02       0.5290E+01
EMC:KO,EM=       0.5290E+01       0.8474E+00
KALL=  5 X=  0.6184E+00  0.1604E+01  0.1303E+01
EM:NRE,K,KO=       0.6836E+02       0.1219E+02       0.4959E+01
EMC:KO,EM=       0.4959E+01       0.8395E+00
KALL=  6 X=  0.7090E+00  0.1403E+01  0.1220E+01
EM:NRE,K,KO=       0.8344E+02       0.1700E+02       0.6405E+01
EMC:KO,EM=       0.6405E+01       0.8665E+00
KALL=  7 X=  0.5000E+00  0.3000E+00  0.1400E+01
EM:NRE,K,KO=       0.3403E+03       0.3243E+03       0.6866E+02
EMC:KO=     0.6866E+02
EMC:KO,EM=       0.6866E+02       0.9776E+00
KALL=  8 X=  0.7914E+00  0.1946E+01  0.8868E+00
EM:NRE,K,KO=       0.8282E+02       0.1217E+02       0.4599E+01
EMC:KO,EM=       0.4599E+01       0.8307E+00
NRE OUT OF RANGE IN HC=     0.10061E+06
KALL=  9 X=  0.6035E+00  0.2000E+01  0.5000E+00
EM:NRE,K,KO=       0.1429E+03       0.2043E+02       0.6168E+01
EMC:KO,EM=       0.6168E+01       0.8655E+00
NRE OUT OF RANGE IN HC=     0.17845E+06
KALL= 10 X=  0.8712E+00  0.1569E+01  0.1127E+01
EM:NRE,K,KO=       0.8085E+02       0.1474E+02       0.5622E+01
EMC:KO,EM=       0.5622E+01       0.8559E+00
KALL= 11 X=  0.7973E+00  0.1522E+01  0.1200E+01
EM:NRE,K,KO=       0.7824E+02       0.1470E+02       0.5681E+01
EMC:KO,EM=       0.5681E+01       0.8576E+00
KALL= 12 X=  0.8236E+00  0.1568E+01  0.1251E+01
EM:NRE,K,KO=       0.7285E+02       0.1329E+02       0.5277E+01
EMC:KO,EM=       0.5277E+01       0.8471E+00
KALL= 13 X=  0.8172E+00  0.1597E+01  0.1210E+01
EM:NRE,K,KO=       0.7394E+02       0.1324E+02       0.5229E+01
EMC:KO,EM=       0.5229E+01       0.8459E+00
KALL= 14 X=  0.8251E+00  0.1605E+01  0.1213E+01
EM:NRE,K,KO=       0.7340E+02       0.1308E+02       0.5179E+01
EMC:KO,EM=       0.5179E+01       0.8447E+00
KALL= 15 X=  0.8244E+00  0.1604E+01  0.1213E+01
EM:NRE,K,KO=       0.7344E+02       0.1309E+02       0.5183E+01
EMC:KO,EM=       0.5183E+01       0.8448E+00
KALL,KOND= 15   2
EM:NRE,K,KO=       0.1369E+03       0.1645E+02       0.5057E+01
EMC:KO,EM=       0.5057E+01       0.8419E+00
NRE OUT OF RANGE IN HC=     0.20350E+06
KALL=  1 X=  0.7100E+00  0.1560E+01  0.1250E+01
EM:NRE,K,KO=       0.7023E+02       0.1287E+02       0.5184E+01
EMC:KO,EM=       0.5184E+01       0.8449E+00
KALL=  2 X=  0.7171E+00  0.1560E+01  0.1250E+01
EM:NRE,K,KO=       0.7023E+02       0.1287E+02       0.5184E+01
EMC:KO,EM=       0.5184E+01       0.8449E+00
KALL=  3 X=  0.7100E+00  0.1576E+01  0.1250E+01
EM:NRE,K,KO=       0.6953E+02       0.1262E+02       0.5101E+01
EMC:KO,EM=       0.5101E+01       0.8429E+00
KALL=  4 X=  0.7100E+00  0.1560E+01  0.1262E+01
EM:NRE,K,KO=       0.6953E+02       0.1274E+02       0.5152E+01
EMC:KO,EM=       0.5152E+01       0.8441E+00
KALL=  5 X=  0.6152E+00  0.1603E+01  0.1294E+01
EM:NRE,K,KO=       0.6604E+02       0.1178E+02       0.4856E+01
```

```
EMC:KO,EM=      0.4830E+01      0.8371E+00
KALL=  6 X=  0.7161E+00  0.1412E+01  0.1227E+01
EM:NRE,K,KO=       0.7902E+02       0.1600E+02       0.6158E+01
EMC:KO,EM=      0.6158E+01      0.8655E+00
KALL=  7 X=  0.5000E+00  0.3000E+00  0.1347E+01
EM:NRE,K,KO=       0.3388E+03       0.3229E+03       0.6848E+02
EMC:KO=      0.6848E+02
EMC:KO,EM=      0.6848E+02      0.9776E+00
KALL=  8 X=  0.8380E+00  0.1860E+01  0.1044E+01
EM:NRE,K,KO=       0.7056E+02       0.1085E+02       0.4362E+01
EMC:KO,EM=      0.4362E+01      0.8240E+00
KALL=  9 X=  0.8959E+00  0.2000E+01  0.5000E+00
EM:NRE,K,KO=       0.1369E+03       0.1958E+02       0.6018E+01
EMC:KO,EM=      0.6018E+01      0.8650E+00
NRE OUT OF RANGE IN HC=      0.17101E+06
KALL= 10 X=  0.8234E+00  0.1610E+01  0.1144E+01
EM:NRE,K,KO=       0.7438E+02       0.1321E+02       0.5206E+01
EMC:KO,EM=      0.5206E+01      0.8454E+00
KALL= 11 X=  0.8011E+00  0.1562E+01  0.1224E+01
EM:NRE,K,KO=       0.7160E+02       0.1310E+02       0.5239E+01
EMC:KO,EM=      0.5239E+01      0.8462E+00
KALL= 12 X=  0.8184E+00  0.1591E+01  0.1256E+01
EM:NRE,K,KO=       0.6853E+02       0.1232E+02       0.5006E+01
EMC:KO,EM=      0.5006E+01      0.8406E+00
KALL= 13 X=  0.8204E+00  0.1599E+01  0.1222E+01
EM:NRE,K,KO=       0.7007E+02       0.1253E+02       0.5051E+01
EMC:KO,EM=      0.5051E+01      0.8417E+00
KALL= 14 X=  0.8206E+00  0.1600E+01  0.1219E+01
EM:NRE,K,KO=       0.7025E+02       0.1256E+02       0.5057E+01
EMC:KO,EM=      0.5057E+01      0.8418E+00
KALL,KOND= 14   2
EM:NRE,K,KO=       0.1310E+03       0.1574E+02       0.4929E+01
EMC:KO,EM=      0.4929E+01      0.8388E+00
NRE OUT OF RANGE IN HC=      0.19466E+06
KALL=  1 X=  0.7100E+00  0.1560E+01  0.1250E+01
EM:NRE,K,KO=       0.6717E+02       0.1231E+02       0.5042E+01
EMC:KO,EM=      0.5042E+01      0.8415E+00
KALL=  2 X=  0.7171E+00  0.1560E+01  0.1250E+01
EM:NRE,K,KO=       0.6717E+02       0.1231E+02       0.5042E+01
EMC:KO,EM=      0.5042E+01      0.8415E+00
KALL=  3 X=  0.7100E+00  0.1576E+01  0.1250E+01
EM:NRE,K,KO=       0.6651E+02       0.1207E+02       0.4961E+01
EMC:KO,EM=      0.4961E+01      0.8396E+00
KALL=  4 X=  0.7100E+00  0.1560E+01  0.1262E+01
EM:NRE,K,KO=       0.6651E+02       0.1219E+02       0.5011E+01
EMC:KO,EM=      0.5011E+01      0.8408E+00
KALL=  5 X=  0.6116E+00  0.1602E+01  0.1283E+01
EM:NRE,K,KO=       0.6372E+02       0.1137E+02       0.4750E+01
EMC:KO,EM=      0.4750E+01      0.8345E+00
KALL=  6 X=  0.7272E+00  0.1423E+01  0.1234E+01
EM:NRE,K,KO=       0.7462E+02       0.1500E+02       0.5901E+01
EMC:KO,EM=      0.5901E+01      0.8642E+00
KALL=  7 X=  0.5000E+00  0.5738E+00  0.1310E+01
EM:NRE,K,KO=       0.1743E+03       0.8687E+02       0.2411E+02
EMC:KO=      0.2411E+02
EMC:KO,EM=      0.2411E+02      0.9394E+00
KALL=  8 X=  0.8318E+00  0.1715E+01  0.1156E+01
EM:NRE,K,KO=       0.6608E+02       0.1102E+02       0.4539E+01
EMC:KO,EM=      0.4539E+01      0.8291E+00
KALL=  9 X=  0.8707E+00  0.1753E+01  0.9958E+00
EM:NRE,K,KO=       0.7506E+02       0.1225E+02       0.4808E+01
EMC:KO,EM=      0.4808E+01      0.8359E+00
KALL= 10 X=  0.8311E+00  0.1617E+01  0.1203E+01
EM:NRE,K,KO=       0.6731E+02       0.1190E+02       0.4870E+01
EMC:KO,EM=      0.4870E+01      0.8374E+00
KALL= 11 X=  0.8170E+00  0.1594E+01  0.1206E+01
EM:NRE,K,KO=       0.6800E+02       0.1220E+02       0.4972E+01
EMC:KO,EM=      0.4972E+01      [illegible]
```

```
EM:NRE,K,KO=      0.6658E+02      0.1198E+02      0.4923E+01
EMC:KO,EM=      0.4923E+01      0.8387E+00
KALL= 13 X=  0.8163E+00  0.1594E+01  0.1226E+01
EM:NRE,K,KO=      0.6704E+02      0.1202E+02      0.4926E+01
EMC:KO,EM=      0.4926E+01      0.8388E+00
KALL= 14 X=  0.8164E+00  0.1594E+01  0.1225E+01
EM:NRE,K,KO=      0.6707E+02      0.1203E+02      0.4929E+01
EMC:KO,EM=      0.4929E+01      0.8388E+00
KALL,KOND= 14  2
EM:NRE,K,KO=      0.1191E+03      0.1431E+02      0.4663E+01
EMC:KO,EM=      0.4663E+01      0.8323E+00
NRE OUT OF RANGE IN HC=     0.17696E+06
KALL=  1 X=  0.7100E+00  0.1560E+01  0.1250E+01
EM:NRE,K,KO=      0.6107E+02      0.1119E+02      0.4748E+01
EMC:KO,EM=      0.4748E+01      0.8344E+00
KALL=  2 X=  0.7171E+00  0.1560E+01  0.1250E+01
EM:NRE,K,KO=      0.6107E+02      0.1119E+02      0.4748E+01
EMC:KO,EM=      0.4748E+01      0.8344E+00
KALL=  3 X=  0.7100E+00  0.1576E+01  0.1250E+01
EM:NRE,K,KO=      0.6046E+02      0.1097E+02      0.4671E+01
EMC:KO,EM=      0.4671E+01      0.8325E+00
KALL=  4 X=  0.7100E+00  0.1560E+01  0.1262E+01
EM:NRE,K,KO=      0.6046E+02      0.1108E+02      0.4716E+01
EMC:KO,EM=      0.4718E+01      0.8337E+00
KALL=  5 X=  0.6029E+00  0.1599E+01  0.1262E+01
EM:NRE,K,KO=      0.5904E+02      0.1056E+02      0.4534E+01
EMC:KO,EM=      0.4534E+01      0.8289E+00
KALL=  6 X=  0.7455E+00  0.1448E+01  0.1246E+01
EM:NRE,K,KO=      0.6602E+02      0.1304E+02      0.5375E+01
EMC:KO,EM=      0.5375E+01      0.8495E+00
KALL=  7 X=  0.5565E+00  0.1009E+01  0.1264E+01
EM:NRE,K,KO=      0.9329E+02      0.2643E+02      0.9524E+01
EMC:KO,EM=      0.9524E+01      0.8957E+00
KALL=  8 X=  0.8042E+00  0.1598E+01  0.1240E+01
EM:NRE,K,KO=      0.6013E+02      0.1076E+02      0.4591E+01
EMC:KO,EM=      0.4591E+01      0.8304E+00
KALL=  9 X=  0.8090E+00  0.1586E+01  0.1238E+01
EM:NRE,K,KO=      0.6064E+02      0.1093E+02      0.4648E+01
EMC:KO,EM=      0.4648E+01      0.8319E+00
KALL= 10 X=  0.8083E+00  0.1584E+01  0.1234E+01
EM:NRE,K,KO=      0.6090E+02      0.1099E+02      0.4668E+01
EMC:KO,EM=      0.4668E+01      0.8324E+00
KALL= 11 X=  0.8158E+00  0.1596E+01  0.1220E+01
EM:NRE,K,KO=      0.6114E+02      0.1095E+02      0.4643E+01
EMC:KO,EM=      0.4643E+01      0.8318E+00
KALL= 12 X=  0.8064E+00  0.1582E+01  0.1240E+01
EM:NRE,K,KO=      0.6071E+02      0.1097E+02      0.4664E+01
EMC:KO,EM=      0.4664E+01      0.8323E+00
KALL= 13 X=  0.8063E+00  0.1582E+01  0.1240E+01
EM:NRE,K,KO=      0.6069E+02      0.1097E+02      0.4663E+01
EMC:KO,EM=      0.4663E+01      0.8323E+00
KALL,KOND= 13  2
```

PROGRAM LISTING

```
    11        999999      5555555555   WW      WW   WW      WW   BBBBBB
    11        999999      5555555555   WW      WW   WW      WW   BBBBBB
  1111      99      99    55           WW      WW   WW      WW   BB
  1111      99      99    55           WW      WW   WW      WW   BB
    11      99      99    55555555     WW      WW   WW      WW   BB
    11      99      99    55555555     WW      WW   WW      WW   BB
    11        99999999            55   WW  WW  WW   WW  WW  WW   BBBBBB
    11        99999999            55   WW  WW  WW   WW  WW  WW   BBBBBB
    11              99            55   WW  WW  WW   WW  WW  WW   BB
    11              99            55   WW  WW  WW   WW  WW  WW   BB
    11            99      55      55   WWWW  WWWW   WWWW  WWWW   BB
    11            99      55      55   WWWW  WWWW   WWWW  WWWW   BB
  111111    999999          555555     WW      WW   WW      WW   BBBBBB
  111111    999999          555555     WW      WW   WW      WW   BBBBBB
```

RSTS V7.0-07 M7 ROSE-HULMAN
JOB PRIORITY = 168   SEQUENCE #10
QUEUED ON 28-JAN-81 AT 09:18 AM

SIMICE.FOR   HEADER = 2   1 COPY
PROAUX.FOR   HEADER = 2   1 COPY

```
3333333333   7777777777             000000
3333333333   7777777777             000000
        33           77           00      00
        33           77           00      00
      33           77             00    0000
      33           77             00    0000
    3333         77               00  00  00
    3333         77               00  00  00
        33     77                 0000    00
        33     77                 0000    00
33      33     77         ;;      00      00
33      33     77         ;;      00      00
  333333       77         ;;        000000
  333333       77         ;;        000000
                        ;;
                        ;;
```

23

SIMICE.FOR (37,

```
-------------------------------------------------------------------------------------------
      PROGRAM SIMICE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C        PROGRAM SIMICE CALCULATES THE SCALED-DOWN CONDITIONS UNDER C
C        WHICH A MODEL SHOULD BE TESTED TO SIMULATE THE EXPERI-     C
C        MENTAL CONDITIONS OF A PROTOTYPE UNDERGOING ICING.         C
C                                                                   C
C          VERSION I  SEPTEMBER 1980                                C
C          PROGRAM DEVELOPED UNDER AIR FORCE GRANT NAM 80-009,      C
C          ROSE-HULMAN INSTITUTE OF TECHNOLOGY,TERRE HAUTE,IN 47803 C
C          SEVERAL REVISIONS BETWEEN SEPT.,1980 AND DEC.,1980       C
C          ADDITIONAL REVISIONS JAN.,1981                           C
C          W.W.BOWDEN                                               C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
C          VARIABLES,UNITS:
C
C          PROTOTYPE(INPUT) VARIABLES:
C                         TMS=PROTOTYPE SURFACE TEMPERATURE,CELSIUS
C                         TMO=AMBIENT(FREE STREAM) TEMPERATURE,CELSIUS
C                         VMO=FREE STREAM VELOCITY,METER/SEC
C                         PMO=FREE STREAM TOTAL PRESSURE,KILOPASCALS
C                                             (101.325KP=1 ATM)
C                         OMEGA=SPECIFIC HUMIDITY,(KG WATER VAPOR)/
C                                             (KG MOIST AIR)
C                         LWC=LIQUID WATER CONTENT,GRAMS/(CUBIC METER)
C                         DM= WATER DROPLET DIAMETER,MICRONS
C                                            (1 MICRON=1.E-6METER)
C                         LC= CHARACTERISTIC DIMENSION OF MODEL,CM
C                         R = SCALEUP RATIO=LC/LCP
C                         IB=MODEL TYPE PARAMETER:=1 ---CYLINDER
C                                             =2 ---SPHERE
C                                             =3 ---0012 NASA
C                                                   AIRFOIL
C                                             =4 ---FUTURE CHOICE
C                                             =5 ---FUTURE CHOICE
C                         IH=HEATING CONDITION PARAMETER:
C                                             =1 ---NO HEAT INPUT
C                                             =2 ---HEAT INPUT
C
C                           IPRINT=PRINTING OPTION IN DELBRO
C                                 =-1---NO PRINTING
C                                 = 0---FINAL RESULTS ONLY
C                                 = 1---X+F ON EVERY ITERATION
C                                 = 2---X+F+W ON EVERY ITERATION
C                           LIM= -1---LAST CASE
C                              =  1(OR ANY INTEGER NOT EQUAL TO -1)
C                                    ---ANOTHER CASE EXPECTED
C
C          MODEL(OUTPUT) VARIABLES:
C                    TFS =MODEL SURFACE TEMPERATURE,C
C                    PFO =FREE STREAM TOTAL PRESSURE,KP
C                    TFO =MODEL FREE STREAM TEMPERATURE,C
C                    VFO =MODEL STREAM VELOCITY,METER/SEC(CALCULATED)
```

24

```
C                 LWCF=LIQUID WATER CONTENT(CALCULATED)
C                 DF  =MODEL DROPLET DIAMETER(CALCULATED)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
          REAL LC,LCL,LCF,LCLIM,LCOUT,LWC,LWCL,LWCOUT
          REAL LWCF,LAMDAS,NRE
          REAL*8 MODEL(3),TYPE(3),MESS(9,5)
          INTEGER*2 DATES(5)
          REAL*8 X(3),F(3),BOUND(6),W(100)
         COMMON /INOUT/ IOUT,IOERR
         COMMON /EMNRE/ NRE
     DATA MESS(2,1),MESS(2,2),MESS(2,3),MESS(2,4),MESS(2,5)
   $ /'BROYDEN','CONVERGE','D      ',' ','AFTER'/
     DATA MESS(3,1),MESS(3,2),MESS(3,3),MESS(3,4),MESS(3,5)
   $ /'BROYDEN','FAILED:','JACOBIAN','SINGULAR','     '/
     DATA MESS(5,1),MESS(5,2),MESS(5,3),MESS(5,4),MESS(5,5)
   $ /'BROYDEN','FAILED:','BOUNDS L','IMITING','AFTER'/
     DATA MESS(7,1),MESS(7,2),MESS(7,3),MESS(7,4),MESS(7,5)
   $ /'BROYDEN','FAILED:','ITERATIO','NS MORE','THAN 20='/
     DATA MESS(9,1),MESS(9,2),MESS(9,3),MESS(9,4),MESS(9,5)
   $ /'BROYDEN','FAILED:','DIMENSIO','NS NW,NB',' WRONG'/
     DATA MODEL/'CYLINDER','  SPHERE','0012NASA'/
     DATA TYPE/  ' ',  ' ',' AIRFOIL'/
C
C  INITIAL DATA
C
          OPEN(UNIT=13,NAME='DELBRU.DAT',TYPE='NEW')
          OPEN(UNIT=17,NAME='ICE.DAT',TYPE='OLD')
          OPEN(UNIT=19,NAME='SIMICY.ERR',TYPE='NEW')
          IOERR=19
          IRUN=0
          INA=17
          IOUT=7
          IBLIM=3
          IHLIM=2
          RLIM=10.0
          LCLIM=50.0
          N=3
          NB=6
          NW=18
          MAXIT=20
          ERROR=1.E-4
        CALL DATE(DATES)
        WRITE(IOUT,13) DATES
C
C
10        CONTINUE
          ICHECK=1
          KALL=0
          IRUN=IRUN+1
C
C      READ IN PROTOTYPE CONDITIONS AND FOLLOWING MODEL
C          VARIABLES:  TFO=FREE STREAM TEMPERATURE,C
C                      TFS=SURFACE TEMPERATURE,C
C                      PPO=AMBIENT (FREE STREAM) PRESSURE,KP
C          AND INITIAL VALUES OF X-VARIABLES;AND BOUNDS.
C
C
          READ(INA,*) IB,IH,R,LC,IPRINT
          READ(INA,*) TMS,TMO,PMO,VMO,OMEGA,LWC,DM,LIM
          READ(INA,*) TFS,TFO,PPO,X,BOUND
C
C     CHECK OF INPUT VARIABLES FOR REASONABLESS:
C
       IF(IB.LT.0.OR.IB.GT.IBLIM)                 ICHECK=2
       IF(IH.LT.0.OR.IH.GT.IHLIM)                 ICHECK=3
       IF(R.LT.0.0.OR.R.GT.RLIM)                  ICHECK=4
       IF(LC.LT.0.0.OR.LC.GT.LCLIM)               ICHECK=5
```

```
      IF(TMO.LT.(-40.0).OR.TMO.GT.4.0)              ICHECK=7
      IF(VMO.LT.10.0.OR.VMO.GT.250.)                ICHECK=8
      IF(PMO.LT.46.5.OR.PMO.GT.101.325)             ICHECK=9
      IF(OMEGA.LT.0.0.OR.OMEGA.GT. 0.0082643)       ICHECK=10
      IF(LWC.LT.0.0.OR.LWC.GT.3.5)                  ICHECK=11
      IF(DM.LT.5.0.OR.DM.GT.70.0)                   ICHECK=12
C
C     BRANCH TO ERROR/WARNING MESSAGE ABOUT INPUT DATA:
C
        GO TO (21,22,23,24,25,26,27,28,29,30,31,32),ICHECK
21      GO TO 50
22      WRITE(IOUT,122) IB
        CALL EXIT
23      WRITE(IOUT,123) IH
        CALL EXIT
24      WRITE(IOUT,124) R
        GO TO 40
25      WRITE(IOUT,125) LC
        GO TO 40
26      WRITE(IOUT,126) TMS
        GO TO 40
27      WRITE(IOUT,127) TMO
        GO TO 40
28      WRITE(IOUT,128) VMO
        GO TO 40
29      WRITE(IOUT,129) PMO
        GO TO 40
30      WRITE(IOUT,130) OMEGA
        GO TO 40
31      WRITE(IOUT,131) LWC
        GO TO 40
32      WRITE(IOUT,132) DM
40      CONTINUE
        WRITE(IOUT,133)
        GO TO 50
C
C
C
C     INTERNAL VARIABLE UNITS ARE AS FOLLOWS:
C              (TEMPERATURES)          = KELVIN
C              (PRSSURES)              = KILOPASCALS
C              (DROPLET DIAMETER)      = MICRONS
C              (BODY DIMENSION)        = CM
C              (VELOCITIES)            = METER/SEC
C              (LIQUID WATER CONTENT)= GM/(CUBIC METER)
C
50        CONTINUE
          TMSL=TMS+273.15
          TMOL=TMO+273.15
          TFSL=TFS+273.15
          TFL =TF0+273.15
C
C     SET UP EQUATIONS TO BE SOLVED BY BROYDEN PROCEDURE
C
C
C     INPUT/OUTPUT VARIABLE NAMES:
C
C              Y1=MOLE FRACTION AIR
C     IN BROYDEN: X(I)=INDEPENDENT VARIABLES
C                 F(I)=FUNCTIONS WHOSE ZEROS ARE TO BE FOUND
C                 X(1)=LWC;X(2)=D;X(3)=V0
C
C       SET UP INITIAL VALUES OF VARIABLES AND THEIR BOUNDS:
C
          LCF=LC/R
          Y1N=(1.0-OMEGA)/28.9752
          Y1D=Y1N+OMEGA/18.016
          Y1=Y1N/Y1D
```

```
              QAP=QA(TMOL,TMSL,PMO,VMO,Y1,DM,LC,LWC,IB)
              RAP=LAMDAS(NRE)
              NB=6
              NW=18
200           CONTINUE
              KALL=KALL+1
             WRITE(IOERR,2015) KALL,X
2015         FORMAT(6X,'KALL=',I3,' X=',3E12.4)
              IF(KALL.GT.MAXIT) GO TO 270
C
C             SET EXPRESSIONS FOR F(1),F(2),ETC. FOR BROYDEN
C
              VF=VMO/X(3)
              DF=DM/X(2)
              LWCF=LWC/X(1)
              F(1)=1.0-QAP/QA(TFL,TFSL,PPO,VF,Y1,DF,LCF,LWCF,IB)
              F(2)=R-X(3)*X(2)**2*RAP/LAMDAS(NRE)
              F(3)=1.-X(1)*X(3)
C
C   CALLING BROYDEN SUBROUTINE
C
          CALL DELBRO(N,X,F,ERROR,KOND,KALL,BOUND,NB,W,NW,IPRINT)
                IF(KOND.NE.0) GO TO 280
                GO TO 200
270       CONTINUE
          KOND=7
280         CONTINUE
C
C          OUTPUT SECTION
C
                WRITE(IOERR,7) KALL,KOND
              LWCF=LWC/X(1)
              DF=DM/X(2)
              VF=VMO/X(3)
              WRITE(IOUT,15) IRUN,(MESS(KOND,I),I=1,5),KALL
              WRITE(IOUT,17) MODEL(IB),TYPE(IB)
              WRITE(IOUT,19) LC
              WRITE(IOUT,41) TMS,TMO,PMO,VMO,LWC,DM,
     $                       LCF,TFS,TFO,PPO,VF,LWCF,DF
              WRITE(IOUT,45) OMEGA
              IF(LIM.NE.-1) GO TO 10
              CALL EXIT
C
C  FORMATS
C
7             FORMAT(6X,'KALL,KOND=',2I3)
13       FORMAT(1H1,30X,'PROGRAM SIMICE'/T50,5A2)
15       FORMAT(T10,'RUN NUMBER=',I3,1X,5A8,I3,' CALLS')
17       FORMAT(T20,'MODEL AND PROTOTYPE BODY:',2A8/T20,
     $ 'PROTOTYPE(INPUT) CONDITIONS:')
19       FORMAT(T8,'LC = PROTOTYPE CHARACTERISTIC DIMENSION=',3X,F5.2,
     $ 'CM')
41       FORMAT(T8,'TMS= PROTOTYPE SURFACE TEMPERATURE=',7X,F6.2,'C'/
     $ T8,'TMO= PROTOTYPE FREE STREAM TEMPERATURE=',3X,F6.2,'C'/
     $ T8,'PMO= PROTOTYPE FREE STREAM TOTAL PRESSURE=',F6.2,'KP'/
     $ T8,'VMO= PROTOTYPE FREE STREAM VELOCITY=',4X,F8.2,'METER/SEC'/
     $ T8,'LWC= PROTOTYPE LIQUID WATER CONTENT=',4X,F8.2,
     $'GRAMS/CUBIC METER'/
     $ T8,'DM = PROTOTYPE DROPLET DIAMETER=',8X,F8.2,'MICRONS'//
     $ T20,' MODEL CONDITONS:'/T8,'LCF=',
     $ ' MODEL CHARACTERISTIC DIMENSION=',7X,F5.2,'CM'/
     $ T8,'TFS= MODEL SURFACE TEMPERATURE=',11X,F6.2,'C'/
     $ T8,'TFO= MODEL STREAM TEMPERATURE=',12X,F6.2,'C'/
     $ T8,'PPO= MODEL STREAM TOTAL PRESSURE=',8X,F7.2,'KP'/
     $ T8,'VPO= MODEL FREE STREAM VELOCITY=',6X
     $ ,F8.2,'METER/SEC'/T8,'LWCF= MODEL LIQUID WATER ',
     $ 'CONTENT=',7X,F8.2,'GRAMS/CUBIC METER'/T8,'DF=MODEL '
```

```
45    FORMAT(//T8,'OMEGA=',E15.4,'(KG H2O VAPOR)/(KG MOIST AIR)'
     $//)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     FORMAT STATEMENTS ASSOCIATED WITH CHECKS ON INPUT DATA          C
C     NOTE:PROTOTYPE VARIABLES,INITIAL VALUES OF X(I),AND VALUES
C     OF BOUND(I) ARE NOT CHECKED.
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
122     FORMAT(8X,'PROTOTYPE TYPE PARAMETER OUT OF RANGE=',I2,
     $ ' CHECK INPUT DATA.')
123     FORMAT(8X,'HEATING CONDITION PARAMETER OUT OF RANGE='
     $ ,I2,' CHECK INPUT DATA.')
124     FORMAT(8X,'SCALE-UP RATIO OUT OF EXPECTED RANGE=',F10.2,
     $' ---CHECK INPUT DATA.')
125     FORMAT(8X,'PROTOTYPE CHARACTERISTIC DIMENSION OUT OF RANGE=',
     $ F10.3,'CM,---CHECK INPUT DATA;CALCULATIONS ALLOWED TO'
     $ ' CONTINUE.')
126     FORMAT(8X,'PROTOTYPE SURFACE TEMPERATURE OUT OF RANGE=',F10.2
     $,'C,CHECK INPUT DATA;CALCULATIONS ALLOWED TO CONTINUE.')
127     FORMAT(8X,'PROTOTYPE FREE STREAM TEMPERATURE OUT OF RANGE=',
     $ F10.2,'C,CHECK INPUT DATA;CALCULATIONS ALLOWED TO ',
     $ 'CONTINUE.')
128     FORMAT(8X,'PROTOTYPE FREE STREAM VELOCITY OUT OF RANGE=',
     $ F10.3,'METER/SEC,CHECK INPUT DATA;CALCULATIONS ALLOWED  ',
     $'TO CONTINUE.')
129     FORMAT(8X,'PROTOTYPE FREE STREAM PRESSURE OUT OF RANGE=',
     $ F12.3,'KILOPASCALS,CHECK INPUT DATA;CALCULATIONS  ',
     $ 'ALLOWED TO CONTINUE.')
130   FORMAT(8X,'PROTOTYPE SPECIFIC HUMIDITY OUT OF RANGE=',E15.4,
     $ 'KG WATER VAPOR/KG MOIST AIR;CHECK INPUT DATA; '
     $ 'CALCULATIONS ALLOWED TO CONTINUE.')
131   FORMAT(8X,'PROTOTYPE LIQUID WATER CONTENT OUT OF RANGE=',
     $ F10.3,'GM/CUBIC METER,CHECK INPUT DATA;CALCULATIONS  ',
     $ 'ALLOWED TO CONTINUE.')
132   FORMAT(8X,'PROTOTYPE DROPLET SIZE OUT OF RANGE=',F10.2,
     $ 'MICRONS,CHECK INPUT DATA;CALCULATIONS ALLOWED TO  '
     $ 'CONTINUE.')
133         FORMAT(8X,'COMPUTATIONS CONTINUING WITH WHAT MAY BE',
     $ ' SPURIOUS DATA.')
C
C     END OF FORMATS FOR WARNING MESSAGES ON INPUT DATA.
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
            END
```

PRDRUX.FOR (37,

```
      FUNCTION DENL(T)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C       THIS FUNCTION CALCULATES THE DENSITY OF LIQUID WATER IN THE C
C       RANGE -50 C TO +50C .IT IS BASED ON AN EQUATION GIVEN BY    C
C       KELL IN J.CHEM.ENG.DATA,12,66-69(1967). BELOW -20 C THE     C
C       RESULTS ARE EXTRAPOLATED.                                   C
C          UNITS: (T)=KELVIN                                        C
C                 (DENL)=GRAMS/CC                                   C
C        W.W.BOWDEN                       21 AUGUST 1980            C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
         REAL*8 A(7),TLOC,NUM,DENOM
         DATA A /0.9998396D0,18.224944D-3,-7.92221D-6,
     $ -55.44846D-9,149.7562D-12,-393.2952D-15,
     $ 18.159725D-3/
         TLOC=T-273.15D0
      NUM=A(1)+TLOC*(A(2)+TLOC*(A(3)+TLOC*(A(4)+TLOC*(A(5)+
     $ TLOC*A(6)))))
         DENOM=1.0+A(7)*TLOC
         DENL=NUM/DENOM
         RETURN
         END
C
C
C
          REAL FUNCTION MUMIX(T,Y1)
C        MIXTURES AT TEMPERATURE T. IT IS BASED ON EQUATIONS        C
C        (9-3.9),P.396 AND (9-5.4),P.411 OF REID,PRAUSNITZ AND      C
C        SHERWOOD(1977).                                            C
C                                                                   C
C           UNITS:(T)=KELVIN                                        C
C                 (MU)=MICROPOISE                                   C
C             Y1=MOLE FRACTION OF AIR                               C
C       W.W.BOWDEN                         AUGUST 21,1980           C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
         REAL M,MUMIX,M1,M2,MU1,MU2,MUF
         OMEGA(TS)=1.16145/(TS)**(0.14874)+0.52487/EXP(
     $ 0.7732*TS)+2.16178/EXP(2.43787*TS)
         MUF(SIG,M,OMEG)=26.69*SQRT(M*T)/(SIG**2*OMEG)
         TS1=T/78.6
         TS2=T/809.1
         M1=28.9752
         M2=18.016
         SIG1=3.711
         SIG2=2.641
         Y2=1.-Y1
         OMEGA1=OMEGA(TS1)
         OMEGA2=OMEGA(TS2)
         MU1=MUF(SIG1,M1,OMEGA1)
         MU2=MUF(SIG2,M2,OMEGA2)
       PHI12=(1.0+SQRT(MU1/MU2)*SQRT(SQRT(M2/M1)))**2/SQRT(
     $ 8.0*(1.0+M1/M2))
       PHI21=PHI12*M1/M2*MU2/MU1
       MUMIX=Y1*MU1/(Y1+PHI12*Y2)+Y2*MU2/(Y2+Y1*PHI21)
         RETURN
```

```
      RETURN
      END
C
C
      REAL FUNCTION KMIX(T,Y1)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C        THIS FUNCTION CALCULATES THE THERMAL CONDUCTIVITY OF         C
C        AIR-WATER MIXTURES.IT IS BASED ON EQUATIONS (10-4.1),        C
C        P.499 AND (10-6.1),(10-6.2) ,(10-6.3),P.508 OF REID,         C
C        PRAUSNITZ AND SHERWOOD.                                      C
C           UNITS: (T)= KELVIN                                        C
C                  (K)= KCAL/(METER-HR-C)                             C
C             Y1=MOLE FRACTION OF AIR                                 C
C     W.W.BOWDEN                           AUGUST 29,1980             C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
         REAL K1,K2,KMIX,M1,M2
      K1(T)=20.280098E-3*(T/263.15)**0.891329
      K2(T)=12.05E-3*(T/250.)**(1.38692)
         M1=28.9752
         M2=18.016
         Y2=1.-Y1
      A12=(1.0+SQRT(K1(T)/K2(T))*SQRT(SQRT(M2/M1)))**2/
     $ SQRT(8.0*(1.0+M1/M2))
      A21=A12*K2(T)/K1(T)*M1/M2
      KMIX=Y1*K1(T)/(Y1+A12*Y2)+Y2*K2(T)/(Y2+Y1*A21)
      RETURN
      END
C
C
C
C
         REAL FUNCTION NPRMIX(T0,Y1)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     THIS FUNCTION CALCULATES THE PRANDTL NUMBER OF AIR-   C
C     WATER VAPOR MIXTURES.                                 C
C    W.W.BOWDEN NOV.,1980                                   C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
         REAL KMIX,MUMIX,NPRMIX
         CPAIR=0.240
         CPH2O=7.967/18.016
         CPMIX=Y1*CPAIR+(1.-Y1)*CPH2O
         NPRMIX=CPMIX*MUMIX(T0,Y1)*1.E-7/KMIX(T0,Y1)*3600.
         RETURN
C
C
         END
       FUNCTION CW(T)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     FUNCTION CW CALCULATES THE HEAT CAPACITY OF LIQUID    C
C     WATER IN THE RANGE -50C TO +50C.                      C
C    W.W.BOWDEN                           AUGUST,1980       C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
          COMMON /INOUT/ IOUT,IOERR
     IFLAG=0
        IF(T.LT.223.15.OR.T.GT.323.15) IFLAG=1
        IF(IFLAG.EQ.1) WRITE(IOERR,11) T
        IF(IFLAG.EQ.1) CALL EXIT
      IF(T.GT.223.15.AND.T.LT.273.15) CW=0.9648266-0.53229597
     $ E-02*(T-273.15)
      IF(T.GE.273.15.AND.T.LE.323.15) CW=1.0057583-0.33745813E-3*
     $ (T-273.15)**2 + 0.20916964E-08*(T-273.15)**4
       RETURN
11        FORMAT(6X,'T IN CW OUT OF RANGE',E15.3)
       END
C
C
       REAL FUNCTION PSO(TS)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

```
C           BASED ON THE WORK OF ARNOLD WEXLER,EQUATION (54),P.11 IN        C
C           JOURNAL OF RESEARCH OF THE NATIONAL BUREAU OF STANDARDS,         C
C           81A,#1,5-20(1977).                                               C
C                 THE VAPOR PRESSURE OF ICE AT THE TRIPLE POINT,273.16K      C
C           ,IS TAKEN TO BE 611.657 PASCALS.                                 C
C           UNITS:TEMPERATURE,KELVIN                                         C
C                 PRESSURE,KILOPASCALS                                              C
C            W.W.BOWDEN                          8/18/80                     C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
                REAL*8 K(6),SUM,TSD
                COMMON /INOUT/IOUT,IOERR
             DATA K/-0.58653696D4,0.22241033D2,0.137490D-1,
     $ -0.34031775D-4,0.26967687D-7,0.6918651D0/
             IF(TS.GT.273.16.OR.TS.LT.213.15) GO TO 100
           SUM=0.0
           TSD=TS
                  DO 10 J=1,5
                  SUM=SUM+K(J)*TSD**(J-2)
10                CONTINUE
           SUM=SUM+K(6)*DLOG(TSD)
           PSO=DEXP(SUM)/1000.
           RETURN
100               WRITE(IOERR,13) TS
13           FORMAT(6X,'IN PSO:TS=',F15.3,' GREATER THAN 273.15 OR'
     $ ,' OUT OF RANGE')
             CALL EXIT
             END
C
C
                FUNCTION PLO(T)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C              THIS FUNCTION CALCULATES THE VAPOR PRESSURE OF LIQUID        C
C              WATER IN THE RANGE -50C TO +50C. THE EQUATIONS ARE FROM      C
C              THE FOLLOWING SOURCES:                                       C
C                    0 C TO 50 C:WEXLER(1976)                               C
C                    0 C TO -15 C:WEXLER(1977) AND BOTTOMLEY(1978)          C
C                    -15 C TO -50 C:CORRELATED FROM DATA IN SMT(1951)       C
C              UNITS: T:KELVIN                                              C
C                     P:KILOPASCALS                                         C
C                W.W.BOWDEN                          8/19/80                C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
               REAL*8 TX,TD,B,C,D,DP,G(8),K(6),SUM
               DATA B,C,D/42.164D0,2.8482D0,0.0580200/
               DATA G/-0.29912729D04,-0.60170128D04,0.188764385D02,
     $               -0.28354721D-1,0.17838301D-4,-0.84150417D-9,
     $                0.44412543D-12,0.2858487D01/
               DATA K/-0.53728198D04,0.21120260D02,0.23145673D-01,
     $               -0.64600326D-05,-0.97969796D-07,0.19452535D0/
               IOUT=13
               IRANGE=1
          TD=T
          IF(T.LT.258.15.AND.T.GE.223.15) IRANGE=2
          IF(T.GE.258.15.AND.T.LT.273.15) IRANGE=3
          IF(T.GE.273.15.AND.T.LE.323.15) IRANGE=4
               GO TO (10,20,30,40) , IRANGE
10        CONTINUE
          WRITE(IOUT,13) T
          CALL EXIT
20        CONTINUE
C
C     T LESS THAN 258K
C
           SUM=0.0
           DO 25 J=1,5
           SUM=SUM+K(J)*TD**(J-2)
25         CONTINUE
           SUM=SUM+K(6)*DLOG(TD)
```

```
      GO TO 100
30          CONTINUE
C
C   T BETWEEN 258.15 AND 273.15
C
        TX=TD-273.14D0
        DP=-(TX)*(B+TX*(C+TX*D))*1.D-3
C
C  UNITS HERE MMHG
C
        DP=DP*1.D3/7.50062D0
C
C  UNITS HERE PASCALS
C
        PLO=(PSO(T)+DP)/1000.
        GO TO 100
40      CONTINUE
C
C  T GREATER THAN 273.15
C
      SUM=0.0
      DO 45 I=1,7
      SUM=SUM+G(I)*TD**(I-3)
45    CONTINUE
      SUM=SUM+G(8)*DLOG(TD)
      PLO=DEXP(SUM)/1000.
100   CONTINUE
      RETURN
13    FORMAT(6X,'TEMPERATURE OUT OF RANGE IN PLO=',E15.4)
      END
C
C
          FUNCTION HC(TO,TS,PO,VO,Y1,LC,IB)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C       THIS FUNCTION CALCULATES THE HEAT TRANSFER COEFFICIENT    C
C       ,HC,FOR ONE OF SEVERAL BODIES AS A FUNCTION OF THE         C
C       REYNOLDS NUMBER,NRE, AND OTHER VARIABLES.                  C
C            IF IB= 1 HC REFERS TO A CYLINDER                      C
C               IB= 2 HC REFERS TO A SPHERE                        C
C               IB= 3 HC REFERS TO A NASA 001 AIRFOIL              C
C            TWO DUMMY SLOTS HAVE BEEN INSERTED FOR EXPANSION.     C
C       SOURCES FOR THE CORRELATIONS ARE AS FOLLOWS:               C
C            CYLINDER AND SPHERE:STEPHEN WHITAKER,ELEMENTARY       C
C            HEAT TRANSFER ANALYSIS(PERGAMON,1976):                C
C                 CYLINDER:PP.212-216;SPHERE:PP.218-218            C
C            NASA 001 AIRFOIL:1)SOGIN,H.H.,WADC TR54-313(1954)     C
C                             2)SCHMIDT,E.AND K.WENNER,NACA        C
C                             TECH. MEMORANDUM 1050(1943)          C
C                             3)SEIBERT,O.,NACA TECH. MEM.1040     C
C                             (1944).                              C
C        W.W.BOWDEN                     SEPTEMBER 3,1980           C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
        REAL KMIX,NRE,NNU,NPR,NPRD,MUMIX,LC,NRED,MW,LCLOC
        REAL NPRMIX
        COMMON /INOUT/ IOUT,IOERR
          NNU(NRED)=(0.4*SQRT(NRED)+0.06*(NRED)**0.66667)*
     S NPRD**(0.4)*SQRT(SQRT(MUMIX(TO,Y1)/MUMIX(TS,Y1)))
        NIB=5
        MW=28.9742*Y1+(1.-Y1)*18.016
        R=8.31431
        RHO=PO*MW/(R*TO)
        LCLOC=LC/100.
        NRE=LCLOC*RHO*VO/(MUMIX(TO,Y1)*1.E-7)
                DO 5 I=1,NIB
                IF(IB.EQ.I) GO TO 10
                [illegible]
```

```
          WRITE(IOERR,13) IB
          CALL EXIT
10        CONTINUE
          NPROD=NPRMIX(TO,Y1)
               GO TO (20,30,40,50),IB
20          CONTINUE
          HC=NNU(NRE)*KMIX(TO,Y1)/LCLOC
          IF(NRE.LT.1.0.OR.NRE.GT.1.E5) GO TO 100
          RETURN
30                CONTINUE
                  HC=(2.0+NNU(NRE))*KMIX(TO,Y1)/LCLOC
                  RETURN
40        CONTINUE
          WRITE(IOERR,15) IB
          CALL EXIT
50        CONTINUE
          WRITE(IOERR,15) IB
          CALL EXIT
100       WRITE(IOERR,17) NRE
             RETURN
13              FORMAT(6X,'BODY PARAMETER OUT OF RANGE=',I2)
15              FORMAT(6X,'IB WRONG=',I3)
17              FORMAT(6X,'NRE OUT OF RANGE IN HC=',E15.5)
          END
C
C
C
        FUNCTION QA(TO,TS,PO,VO,Y1,D,LC,LWC,IB)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C       FUNCTION QA CALCULATES THE HEAT FLUX FROM MESSINGER'S   C
C       EQUATIONS(J.A.SCI.,20,29-42(1953))                      C
C       UNITS ARE THOSE USED BY MESSINGER                       C
C     W.W.BOWDEN                      DECEMBER,1980             C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
          REAL LS,LC,LWC,LWCLOC
            TOLOC=TO*1.8-459.68
            TSLOC=TS*1.8-459.68
            VOLOC=VO/0.3048
            LWCLOC=LWC*(0.3048)**3/454.
            CW=1.0
            CI=0.47
            RW=VOLOC*LWCLOC*EM(TO,PO,VO,Y1,D,LC,IB)
            LS=1219.82
            YH2O=1.-Y1
            PSLOC=PSO(TS)
            R=0.875
            CPAIR=0.2394
            PROD=64.4*778.0
        P1=(2.90*LS*(YH2O-PSLOC/PO)-TOLOC+TSLOC-R/CPAIR*
     $ VOLOC**2/PROD)*HC(TO,TS,PO,VO,Y1,LC,IB)
        P2=RW*(CW*(32.-TOLOC)-144.-CI*(32.-TSLOC)-VOLOC**2/PROD)
        QA=P1+P2
        RETURN
        END
C
C
               FUNCTION EM(T,P,VO,Y1,D,LC,IB)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C           THIS FUNCTION CALCULATES THE COLLECTION EFFICIENCY,EM,  C
C           OF THREE BODIES:A CYLINDER OF DIAMETER LC,              C
C                           A SPHERE OF DIAMETER LC, AND            C
C                           OR NASA 001 AIRFOIL.                    C
C           IT IS BASED ON EXPRESSIONS AND DATA FROM THE FOLLOWING  C
C           SOURCES:                                                C
C                   1. G.D.PFIEFFER,ET AL.,ENGINEERING SUMMARY OF   C
C                      POWERPLANT ICING TECHNICAL DATA,AD/A-045 087 C
C                   2. W.W.SOGIN,A DESIGN MANUAL FOR THERMAL ANTI-  C
C                      [illegible]                                  C
```

```
C                    3. OTHER SOURCES OF DATA GIVEN IN THE FUNCTIONS C
C                       USED BY EM.                                  C
C           UNITS OF INPUT DATA:                                     C
C                                     (P)= KILOPASCALS               C
C                                     (T)= KELVIN                    C
C                                     (D)= MICRONS                   C
C                                     (V0)=METER/SEC                 C
C                                     (LC)=CM                        C
C                                                                    C
C           W.W.BOWDEN                          25 AUGUST 1980       C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      REAL K,K0,MU,MULOC,MW,LCLOC,LAMDAS,B(4),C(4),E(4),NRE
     S ,MUMIX
         REAL LC
            COMMON /INOUT/ IOUT,IOERR
            COMMON /EMNRE/ NRE
            EM0012(X)=X
         DLOC=D*3.281E-6
         LCLOC=LC/100./0.3048/2.0
         VOLOC=V0/0.3048
         DENLOC=DENL(T)*62.4
         TLOC=T*1.8
         MULOC=MUMIX(T,Y1)*1.E-7*2.20462/3.2808399
         PLOC=P/101.325*14.696
         R=10.7319
         MW=28.9752*Y1+18.016*(1.-Y1)
         ILLAG=0
            K=(DLOC*0.5)**2*DENLOC*VOLOC*2.0/(LCLOC*MULOC*9.)
                    RHOAIR=PLOC*MW/(R*TLOC)
                    NRE=RHOAIR*VOLOC*DLOC/MULOC
                    K0=LAMDAS(NRE)*K
             WRITE(IOERR,33) NRE,K,K0
33           FORMAT(6X,'EM:NRE,K,K0=',3E15.4)
         IF(IB.EQ.1) EM=EMC(K0)
         IF(IB.EQ.2) EM=EMS(K0)
         IF(IB.EQ.3) EM=EM0012(K0)
         IF(IB.GT.3) WRITE(IOERR,13) IB
         IF(IB.GT.3) CALL EXIT
         IF(ILLAG.EQ.1) CALL EXIT
         RETURN
13                FORMAT(6X,'BODY PARAMETER,IB, OUT OF RANGE=',I3)
         END
C
C
               REAL FUNCTION LAMDAS(NRE)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C         FUNCTION LAMDAS CALCULATES THE 'RANGE' AS A FUNCTION   C
C         OF THE DROP REYNOLDS NUMBER. IT IS BASED ON DATA       C
C         TABULATED IN H.H.SOGIN,WADC TR54-313(1954)            C
C      W.W.BOWDEN                    AUGUST,1980               C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
              REAL NRE,LAMDAS,LAM,NRED
              REAL*8 B(4),C(4),X1,X2,X3,X4
              COMMON /INOUT/ IOUT,IOERR
        DATA B /0.17797562D0,0.44015126D-3,0.27469307D0,
     S          0.37743733D-2/
        DATA C /0.34487129D0,0.71310500D-07,0.2236679D-01,
     S          0.13819799D-04/
         LAM(X1,X2,X3,X4)=(1.0+X1*NRED+X2*NRED**2)/
     S                    (1.+X3*NRED+X4*NRED**2)
         NRED=NRE
         IF(NRE.LT.0.0.OR.NRE.GT.10000.) GO TO 40
         IF(NRE.GT.500.)                 GO TO 30
                LAMDAS=LAM(B(1),B(2),B(3),B(4))
                RETURN
30       CONTINUE
         LAMDAS=LAM(C(1),C(2),C(3),C(4))
```

```
40          CONTINUE
            WRITE(IOERR,13) NRE
            CALL EXIT
13     FORMAT(6X,'NRE OUT OF RANGE IN LAMDAS=',E15.4)
       END
C
C
          FUNCTION EMC(K0)
       REAL K0
       INTEGER RANGE
         COMMON /INOUT/ INOUT,IOERR
       FEM(A0,A1,A2,A3)=A0+A1*K0+A2*K0**2+A3*K0**3
       IF(K0.LT.0.0) GO TO 120
       IF(K0.LE.0.1) RANGE =1
       IF(K0.GT.0.1.AND.K0.LT.0.3) RANGE=2
       IF(K0.GE.0.3.AND.K0.LT.1.0) RANGE=3
       IF(K0.GE.1.0.AND.K0.LT.6.0) RANGE=4
       IF(K0.GE.6.0.AND.K0.LE.10.) RANGE=5
       IF(K0.GT.10.)               RANGE=6
      GO TO (10,20,30,40,50,60) , RANGE
10        EMT=0.0
          GO TO 80
20        A0=-0.035
          A1= 0.3
          A2= 0.5
          A3= 0.0
      GO TO 70
30        A0=-0.31381
          A1= 1.9039
          A2=-2.03463
          A3=0.909091
      GO TO 70
40        A0=0.183123
          A1=0.343423
          A2=-0.0632076
          A3=0.00416364
      GO TO 70
50        A0=1.05529
          A1=-0.08100119
          A2=0.0107143
          A3=-0.000416667
      GO TO 70
60        A0=0.83324
          A1=0.0063575
          A2=-0.0000914342
          A3=0.429012E-6
120   WRITE(IOERR,13) K0
13    FORMAT(6X,'EMC:K0=',E15.4)
70        EMT=FEM(A0,A1,A2,A3)
          WRITE(IOERR,23) K0,EMT
        IF(EMT.GT.1.0) EMT=1.0
80        EMC=EMT
          RETURN
23      FORMAT(6X,'EMC:K0,EM=',2E15.4)
          END
C
          FUNCTION EMS(K0)
       REAL K0
       INTEGER RANGE
         COMMON /INOUT/ IOUT,IOERR
       FEM(A0,A1,A2,A3)=A0+A1*K0+A2*K0**2+A3*K0**3
       IF(K0.LT.0.0) GO TO 120
       IF(K0.LE.0.07) RANGE =1
       IF(K0.GT.0.07.AND.K0.LT.0.4) RANGE=2
       IF(K0.GE.0.4.AND.K0.LT.1.0) RANGE=3
       IF(K0.GE.1.0.AND.K0.LT.6.0) RANGE=4
       IF(K0.GE.6.0.AND.K0.LE.10.) RANGE=5
       IF(K0.GT.10.)               RANGE=6
```

```
        GO TO (10,20,30,40,50,60) , RANGE
10            EMT=0.0
              GO TO 80
20            A0=-0.130495
              A1=2.12278
              A2=4.77231
              A3=5.06444
        GO TO 70
30            A0=-0.0371429
              A1=0.935714
              A2=-0.357143
              A3=0.499436E-12
        GO TO 70
40            A0=0.22
              A1=0.39504
              A2=-0.0794048
              A3=0.0055556
        GO TO 70
50            A0=0.87
              A1=0.01
              A2=-0.137435E-12
              A3=0.570531E-14
        GO TO 70
60            A0=0.0
              A1=0.0
              A2=0.0
              A3=0.0
120     WRITE(IOERR,13) K0
13      FORMAT(6X,'EMS:K0=',E15.4,' K0 OUT OF RANGE')
70            EMT=FEM(A0,A1,A2,A3)
              WRITE(IOERR,23) K0,EMT
23         FORMAT(6X,'EMS:K0,EM=',2E15.4)
80            EMS=EMT
              RETURN
              END
C
C
                SUBROUTINE DELBRO(N,X,F,EPS,INFO,KALL,BOUND,
     $ NB,W,NW,IPRINT)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C            ROUTINE DUE TO DR.J.D.PERKINS                          C
C                   IMPERIAL COLLEGE,LONDON  SW7 2BY                C
C            ALGORITHM DUE TO BROYDEN PLUS MORE-COSNARD MODIFICATION C
C                                                                   C
C            SPEEDUP LIBRARY ---REQUIRES F(K),K=1,N                 C
C            VARIABLE USEAGE                                        C
C            --------------                                         C
C              N=NUMBER OF VARIABLES                                C
C              F=FUNCTION VECTOR,DIMENSION (N)                      C
C              X=SOLUTION VECTOR,DIMENSION (N)                      C
C              EPS=CONVERGENCE CRITERIA:TEST IS:IF(DABS(F).LE.EPS)  C
C                                                 CONVERGED         C
C              INFO:                                                C
C                    ON RETURN FROM DELBRO:-                        C
C                    =0 --- ITERATIONS ARE CONTINUING. REQUIRES
C                           EVALUATION OF F CORRESPONDING TO X      C
C                    =2 --- CONVERGED. X CONTAINS A GOOD APPROXIMATION C
C                           TO THE SOLUTION.                        C
C                    =3 --- FAILED DUE TO INITIAL JACOBIAN APPROXIMA- C
C                           TION BEING SINGULAR.POSSIBLE REASONS:   C
C                           POOR SCALING OF PROBLEM;EQUATIONS ARE NOT C
C                           INDEPENDENT;POOR INITIAL GUESS.         C
C                    =5 --- FAILED BECAUSE BOUNDS ARE IMPEEDING     C
C                           PROGRESS.POSSIBLE REASONS:NO SOLUTION   C
C                           WITHIN THE BOUNDS;POOR INITIAL GUESS.   C
C                    =9 --- FAILED BECAUSE DIMENSIONS NW,NB ARE TOO C
C                           SMALL.                                  C
```

```
C             KALL=SCALAR SET TO 1 ON INITIAL CALL AND INCREMENTED BY C
C                  ONE ON SUBSEQUENT CALLS.                           C
C             BOUND=BOUNDS ARRAY.SET LOWER BOUNDS IN LOWER N          C
C                  POSITIONS AND UPPER BOUNDS IN UPPER N POSITIONS,   C
C                  DIMENSION (2*N).                                   C
C             NB --- MUST BE SET EQUAL TO 2*N.                        C
C             W=JACOBIAN MATRIX,DIMENSION (N**2+3*N)                  C
C             NW --- MUST BE SET EQUAL TO N**2+3*N                    C
C             IPRINT=PRINTING CONTROL PARAMETER:                      C
C                         -1 NO PRINTING                              C
C                          0 FINAL RESULTS                            C
C                          1 X+F EVERY ITERATION                      C
C                          2 X+F+W EVERY ITERATION                    C
CCCCCCCCCCCCCC
C          BROYDEN'S METHOD---JULY 1978                               C
C          WITH BROYDEN INITIALIZATION USING GAUSS-JORDAN INVERSION   C
C          OF BO.                                                     C
C                                                                     C
C          MODIFIED ON 28 JUNE 1979 FOR TESTING WITH EUCLIDIAN NORM   C
C          FOR CONTROLLING FUNCTION INCREASING.                       C
C          MODIFICATION NOTED WITH .................................  C
C                                                                     C
C          MODIFIED 18 SEPTEMBER 1979 FOR ADDING CONTROL ON STEP      C
C          GENERATED ACCORDING TO MORE-COSNARD PAPER.                 C
C          MODIFICATION NOTED WITH !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!  C
C                                                                     C
C          INSTALLED ON RHIT  11/70 AUGUST 1980                       C
C                W.W.BOWDEN                          20 AUGUST 1980   C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
            DIMENSION X(N),F(N),W(NW),BOUND(NB)
              IMPLICIT REAL*8(A-H,O-Z)
             COMMON /INOUT/ IOUT,IOERR
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C         ADDED 14 SEPTEMBER 1979
            IO=13
          IF(KALL.NE.1) GO TO 2
          XNORM=0.0
                    DO 4010 I=1,N
4010                XNORM=XNORM+X(I)*X(I)
          XNORM=DSQRT(XNORM)
          DELTA=10.0
          IF(XNORM.NE.0.0) DELTA=10.*XNORM
2         IF(KALL.LE.(N+1)) GO TO 3
          XNORM =0.0
                    DO 4020 I=1,N
4020                XNORM=XNORM+X(I)**2
          XNORM=DSQRT(XNORM)
          IF(DELTA.LT.10.*XNORM) DELTA=10.*XNORM
3         CONTINUE
C !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
                N2=N*N
                N3=N2+N
                N4=N3+N
                N5=N4+N
                INFO=0
                IF(NW.GE.N5) GO TO 5
C**          INSUFFICIENT WORKING SPACE
                INFO=9
                IF(IPRINT.GE.0) WRITE(IO,1000)
1000            FORMAT(6X,'DELBRO*** DIMENSION OF W TOO SMALL')
                RETURN
C**          ITERATION PRINT-OUT
5               IF(IPRINT.LE.0) GO TO 10
                WRITE(IO,1050) KALL
1050            FORMAT(6X,'DELBRO*** NUMBER OF CALLS= ',I3)
                WRITE(IO,1020) (X(I),I=1,N)
                WRITE(IO,1030) (F(I),I=1,N)
```

```
      [illegible]
      WRITE(IO,1060) (W(I),I=1,N5)
1060  FORMAT(6X,'W: ',8E15.4)
C**   CONVERGENCE TEST
              DO 20 I=1,N
              IF(DABS(F(I)).GT.EPS) GO TO 30
20            CONTINUE
      INFO=2
      IF(IPRINT.LT.0) RETURN
      WRITE(IO,1010) KALL
1010  FORMAT('***DELBRO***CONVERGENCE AFTER ',I4,' CALLS')
      IF(IPRINT.GT.0) RETURN
      WRITE(IO,1020) (X(I),I=1,N)
      WRITE(IO,1030) (F(I),I=1,N)
1020  FORMAT(6X,'X ',8E15.4)
1030  FORMAT(6X,'F ',8E15.4)
      RETURN
C**   NO CONVERGENCE --- ITERATIONS CONTINUE.
30    IF(KALL.EQ.1) GO TO 50
      IF(KALL.GT.(N+1)) GO TO 110
C**   INITIALIZATION PHASE--- STORE COLUMN OF B0.
      L=KALL-1-N
      STEPX=W(N5)
              DO 40 I=1,N
              L=L+N
40            W(L)=(F(I)-W(N2+I))/STEPX
C**   REMOVING PERTURBATION FROM FROM X(KALL-1)
      X(KALL-1)=X(KALL-1)-STEPX
      IF(KALL.EQ.(N+1)) GO TO 90
      GO TO 70
C**   FIRST CALL --- STORING INITIAL F.
50    DO 60 I=1,N
60    W(N2+I)=F(I)
C**   ADDING PERTURBATION TO X(KALL) AND CHECKING BOUNDS.
C.......................................................................
C
C     MODIFIED 28 JUNE 1979
      FNORM0=0.0
              DO 65 I=1,N
65            FNORM0=FNORM0+F(I)**2
      FNORM0=DSQRT(FNORM0)
C.........................................................................
70    STEPX=0.01*X(KALL)
      IF(DABS(X(KALL)).LT.1.E-6) STEPX=0.01
      X(KALL)=X(KALL)+STEPX
      W(N5)=STEPX
      IF(X(KALL).LT.BOUND(KALL)) GO TO 80
      IF(X(KALL).GT.BOUND(N+KALL)) GO TO 80
      RETURN
80    W(N5)=-W(N5)
      X(KALL)=X(KALL)-2.0*STEPX
      RETURN
C**   LAST INITIALISATION CALL - INVERTING B0.
90    DO 100 I=1,N
100   F(I)=W(N2+I)
      CALL GJINV(N,N2,N3,N4,N5,W,INFO)
      DO 105 L=1,N2
105   W(L)=-W(L)
      IF(INFO.EQ.0) GO TO 220
C**   B0 SINGULAR.
      IF(IPRINT.LT.0)RETURN
      WRITE(IO,1040)
1040  FORMAT(6X,'DELBRO *** INITIAL JACOBIAN APPROXIMATION IS '
     $ 'SINGULAR.')
      RETURN
C**   UPDATING PHASE - CHECKING FUNCTION INCREASES.
C..................MODIFIED 28 [illegible] 79
110   FNORM=0.0
      [illegible]
```

```
                   FNORM=FNORM+F(I)*F(I)
120                CONTINUE
                   FNORM=DSQRT(FNORM)
                   IF (FNORM.LT.100.*FNORM0) GO TO 140
C.........................................................
C**                FUNCTION INCREASE TOO LARGE - REDUCING STEP.
                   IF(IPRINT.LE.0) GO TO 125
                   WRITE(10,1070)
1070     FORMAT(6X,'DELBRO *** STEP CAUSED TOO LARGE A FUNCTION ',
     $ 'INCREASE.')
125                DO 130 I=1,N
                   PI=W(N3+I)
                   X(I)=X(I)-PI
                   PI=PI*0.05
                   X(I)=X(I)+PI
130                W(N3+I)=PI
                   RETURN
C**                UPDATE
140                DO 150 I=1,N
150                W(N2+I)=F(I)-W(N2+I)
C..................ADDED 28 JUNE 79
                   FNORM0=FNORM
C..............................
                   L=0
                   BETA=0.0
                   DO 170 I=1,N
                   WYI=0.0
                   DO 160 J=1,N
                   L=L+1
160                WYI=WYI+W(L)*W(N2+J)
                   W(N4+I)=WYI
170                BETA=BETA+WYI*W(N3+I)
                   IF(DABS(BETA).GT.1.0D-15) GO TO 180
                   IF(IPRINT.LE.0) GO TO 220
                   WRITE(10,1080)KALL
1080    FORMAT(6X,'DELBRO *** NO. OF CALLS= ',I4,' DENOMINATOR'
     $ ,' FAILURE.')
                   GO TO 220
180                DO 200 I=1,N
                   W(N4+I)=W(N4+I)+W(N3+I)
                   WPI=0.0
                   L=I-N
                   DO 190 J=1,N
                   L=L+N
190                WPI=WPI+W(L)*W(N3+J)
200                W(N2+I)=WPI
                   L=0
                   DO 210 I=1,N
                   DO 210 J=1,N
                   L=L+1
210                W(L)=W(L)-W(N4+I)*W(N2+J)/BETA
C**                NEW BROYDEN STEP
220                L=0
                   DO 240 I=1,N
                   PI=0.0
                   DO 230 J=1,N
                   L=L+1
230                PI=PI+W(L)*F(J)
                   W(N2+I)=F(I)
                   W(N3+I)=PI
240                X(I)=X(I)+PI
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C                  ADDED ON 14 SEPT 79
C
                   DNORM=0.0
                   DO 4100 I=1,N
4100               DNORM=DNORM+W(N3+I)*W(N3+I)
                   [illegible]
```

```
      DNORM=DNRM2(...)
      IF (DNORM.LE.DELTA) GO TO 4200
      DO 4150 I=1,N
      STORE=W(N3+I)/DNORM*DELTA
      X(I)=X(I)-W(N3+I)+STORE
      W(N3+I)=STORE
4150  CONTINUE
4200  CONTINUE
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C**   CHECKING ON BOUNDS
      CALL BOUNDR(N,N3,IPRINT,X,W,BOUND,NB,NW,INFO)
      IF(INFO.EQ.0.OR.IPRINT.LT.0)RETURN
      WRITE(10,1090)
1090  FORMAT(6X,'DELBRO *** FAILED TO FIND A SOLUTION')
      RETURN
      END
C
C
      SUBROUTINE BOUNDR(N,N2,IPRINT,X,W,BOUND,NB,NW,INFO)
      DIMENSION X(N),W(NW),BOUND(NB)
      IMPLICIT REAL*8(A-H,O-Z)
C**   CHECKING LOWER BOUNDS.
      IO=13
      KACT=0
      DO 10 I=1,N
      IF(X(I).GE.BOUND(I)) GO TO 10
      W(N2+I)=W(N2+I)-X(I)+BOUND(I)
      X(I)=BOUND(I)
      KACT=KACT+1
10    CONTINUE
C**   CHECKING UPPER BOUNDS.
      DO 20 I=1,N
      IF(X(I).LE.BOUND(N+I)) GO TO 20
      W(N2+I)=W(N2+I)-X(I)+BOUND(N+I)
      X(I)=BOUND(N+I)
      KACT=KACT+1
20    CONTINUE
C**   CHECKING FOR FINITE STEP.
      IF(KACT.EQ.0)RETURN
      DO 30 I=1,N
      IF(DABS(W(N2+I)).GT.1.0D-6)RETURN
30    CONTINUE
C**   BOUNDS IMPEDING PROGRESS.
      INFO=5
      IF(IPRINT.LT.0)RETURN
      WRITE(IO,1000)  (X(I),I=1,N)
1000  FORMAT(6X,'DELBRO *** BOUNDS ARE IMPEDING PROGRESS',
     $ ' X ',8E15.4)
      RETURN
      END

      SUBROUTINE GJINV(N,N2,N3,N4,N5,W,INFO)
C
C     GAUSS-JORDAN INVERSION OF A MATRIX.
C     THE MATRIX IS STORED IN THE FIRST N2
C     LOCATIONS OF W, 3*N EXTRA LOCATIONS
C     ARE USED AS WORKING SPACE (IROW,JCOL,Y)
C
      DIMENSION W(N5)
      IMPLICIT REAL*8(A-H,O-Z)
C**   INITIALISATION.
      DO 5 I=1,N
      W(N2+I)=-1.0
5     W(N3+I)=-1.0
C**   BEGIN ELIMINATION.
      DO 45 K=1,N
C**   SEARCH FOR LARGEST PIVOT.
      PIVOT=0.0
```

```
      DO 15 I=1,N
      IF(W(N2+I).GT.0.0) GO TO 15
      DO 10 J=1,N
      IF(W(N3+J).GT.0.0) GO TO 10
      L=(I-1)*N+J
      IF(DABS(W(L)).LE.DABS(PIVOT)) GO TO 10
      PIVOT=W(L)
      IROWK=I
      JCOLK=J
10    CONTINUE
15    CONTINUE
C**   TEST FOR SINGULAR MATRIX.
      IF(DABS(PIVOT).GT.1.0D-12) GO TO 20
      INFO=3
      RETURN
C**   NORMALISE PIVOT ROW ELEMENTS.
20    AK=FLOAT(K)
      W(N2+IROWK)=AK
      W(N3+JCOLK)=AK
      L=N*(IROWK-1)
      LIROWK=L
      DO 30 J=1,N
      L=L+1
30    W(L)=W(L)/PIVOT
      W(IROWK+JCOLK)=1.0/PIVOT
C**   ELIMINATION.
      L=JCOLK-N
      DO 40 I=1,N
      L=L+N
      AIJCK=W(L)
      IF(I.EQ.IROWK) GO TO 40
      W(L)=-W(L)/PIVOT
      DO 35 J=1,N
      IF(J.EQ.JCOLK) GO TO 35
      M=(I-1)*N+J
      W(M)=W(M)-AIJCK*W(LIROWK+J)
35    CONTINUE
40    CONTINUE
45    CONTINUE
C**   UNSCRAMBLE THE INVERSE,
C**   FIND ALL ROWS/COLS WHICH DO NOT NEED UNSCRAMBLING.
      DO 145 J=1,N
      IF(DABS(W(N2+J)-W(N3+J)).GT.0.1) GO TO 145
      W(N2+J)=0.0
      W(N3+J)=0.0
145   CONTINUE
C**   UNSCRAMBLING BY ROWS - FIND A ROW FOR INTERCHANGE.
245   DO 345 J=1,N
      JP=J
      IF(W(N2+J).GT.0.1) GO TO 545
345   CONTINUE
C**   NO ROWS LEFT.
      DO 445 J=1,N
445   W(N2+J)=DABS(W(N2+J))
      GO TO 61
C**   PUTTING ROW JP IN WORKING SPACE.
545   LO=N*(JP-1)
      DO 46 J=1,N
46    W(N4+J)=W(LO+J)
C**   LOOP FOR INTERCHANGING ROWS BEGINS
47    K=IDINT(W(N2+JP)+0.5)
      W(N2+JP)=-W(N2+JP)
C**   FINDING K IN W(N3+-)
      DO 50 J=1,N
      JP=J
      IF(IDINT(W(N3+J)+0.5).EQ.K) GO TO 55
50    CONTINUE
C**   [illegible]
```

```
C**      NOW ... CHANGE.
55       LO=N*(JP-1)
         DO 60 J=1,N
         WW=W(N4+J)
         W(N4+J)=W(LO+J)
60       W(LO+J)=WW
         IF(W(N2+JP).GT.0.1) GO TO 47
         GO TO 245
C**      UNSCRAMBLING BY COLS. - FIND A COL FOR INTERCHANGE.
61       DO 62 J=1,N
         JP=J
         IF(W(N3+J).GT.0.1) GO TO 63
62       CONTINUE
C**      NO COLS LEFT.
         RETURN
C**      PUTTING COL JP IN WORKING SPACE.
63       L=JP-N
         DO 65 I=1,N
         L=L+N
65       W(N4+I)=W(L)
C**      LOOP FOR COL INTERCHANGES BEGINS.
70        K=IDINT(W(N3+JP)+0.5)
          W(N3+JP)=-1.0
C**       FINDING K IN W(N2+-)
          DO 75 J=1,N
          JP=J
          IF(IDINT(W(N2+J)+0.5).EQ.K) GO TO 80
75        CONTINUE
C**       COL INTERCHANGE.
80        L=JP-N
          DO 85 I=1,N
          L=L+N
          WW=W(N4+I)
          W(N4+I)=W(L)
85        W(L)=WW
          IF(W(N3+JP).GT.0.1) GO TO 70
          GO TO 61
          END
```

LMED

8