MMLC - A FORTRAN PACKAGE FOR LINEARLY CONSTRAINED MINIMAX OPTIMIZATION

J.W. Bandler and W.M. Zuberek

SOS-82-5-L2

August 1983

MMLC - A FORTRAN PACKAGE FOR LINEARLY CONSTRAINED MINIMAX OPTIMIZATION

J.W. Bandler and W.M. Zuberek

Abstract

MMLC is a package of Fortran subroutines for solving linearly constrained minimax optimization problems. It implements the combined LP and quasi-Newton methods introduced by Hald and Madsen. The package and documentation have been developed for use on the CDC 170/730 system with level 552 operating system and the Fortran Extended (FTN) version 4.8 compiler. This document contains a listing of the MMLC package.

This work was supported by the Natural Sciences and Engineering Research Council of Canada under Grant G0647.

The authors are with the Simulation Optimization Systems Research Laboratory and the Department of Electrical and Computer Engineering, McMaster University, Hamilton, Canada L8S 4L7.

W.M. Zuberek is on leave from the Institute of Computer Science, Technical University of Warsaw, Warsaw, Poland. He is now with the Department of Electrical Engineering, Texas A&M University, College Station, TX 77843, U.S.A.

I. INTRODUCTION

MMLC is a package of Fortran subroutines for solving linearly constrained minimax optimization problems. It is an extension and modification of the package MMLA1Q [1] due to Hald. The package implements combined LP and quasi-Newton methods [2].

The whole package is written in Fortran IV for the CDC 170/730 system with the NOS 1.4 level 552 operating system. It is available at McMaster University in the form of a library of binary relocatable subroutines in the group indirect file LIBRMML under the charge RJWBAND.

This document includes a listing of the package MMLC. The user's manual presented together with illustrative examples is found in [3]. The listing contains 1972 lines.

II. REFERENCES

[1] J. Hald (Adapted and Edited by J.W. Bandler and W.M. Zuberek), "MMLA1Q - A Fortran package for linearly constrained minimax optimization", Department of Electrical and Computer Engineering, McMaster University, Hamilton, Canada, Report SOS-81-14-UL, 1981.

[2] J. Hald and K. Madsen, "Combined LP and quasi-Newton methods for minimax optimization, Mathematical Programming, vol. 20, 1981, pp. 49-62.

[3] J.W. Bandler and W.M. Zuberek, MMLC - A Fortran package for linearly constrained minimax optimization", Department of Electrical and Computer Engineering, McMaster University, Hamilton, Canada, Report SOS-82-5-U2, 1983.

III. LISTING OF THE MMLC PACKAGE

Subroutine	Number of lines (source text)	Number of words (compiled code)	Listing from page
MMLC1A	87	742	4
MMLA1Q	11	121	5
MMXOOZ	9	23	5
MMXOOQ	35	216	5
MMXOOV	26	235	6
MMXOOG	35	267	6
MMXOOH	67	435	7
MMXOOB	28	151	8
MMXPSZ	12	42	8
MMXPLM	11	37	8
MMXLLM	11	36	9
MMXHDR	16	47	9
MMXGLM	13	44	9
MMXGVL	11	41	9
MMLC8A	66	330	10
MMLC9A	245	1516	11
S2LA1Q	271	1441	14
FEASI	229	1360	19
MMLPA	280	1545	22
LINSYS	93	333	26
BFGS	43	215	28
ADDCL	92	357	29
DELCL	53	220	30
UTTRNS	36	130	31
UTRNS	33	141	31
RSOLV	21	76	32
TSOLV	19	72	32
HACUM	55	255	33
LIMIT	18	63	34

```
      SUBROUTINE MMLC1A (FDF,N,M,L,LEQ,C,DC,IC,X,DX,EPS,MAXF,KEQS,W,IW,L   000001
     1CH,IPR,IFALL)                                                        000002
      EXTERNAL FDF,MMX00Q,MMX00B                                           000003
C                                                                          000004
C          LEVEL 1 INTERFACE (STANDARD ENTRY)                              000005
C                                                                          000006
      DIMENSION C(1), DC(1,1), X(1), W(1)                                  000007
      COMMON /MMX000/ NCH,LV1,LV2,LG1,LG2,LMF,LMV,NRP,NRL,MXL,LMP,LML,LG   000008
     1H,DAT,TIM,LHT,H(8)                                                   000009
      NCH=LCH                                                              000010
      IF (LCH.LE.0) GO TO 40                                               000011
      I=IABS(IPR)                                                          000012
      J=I/10                                                               000013
      LG2=MOD(I,10)                                                        000014
      I=J/10                                                               000015
      LG1=MOD(J,10)                                                        000016
      J=I/10                                                               000017
      LV2=MOD(I,10)                                                        000018
      LV1=J                                                                000019
      LG1=LG1*LV1                                                          000020
      NRP=0                                                                000021
      CALL MMXPSZ (-1)                                                     000022
      CALL MMXPLM (-1)                                                     000023
      CALL MMXLLM (-1)                                                     000024
      CALL MMXHDR (-1,H)                                                   000025
      CALL MMXGLM (-1,-1)                                                  000026
      CALL MMXGVL (-1)                                                     000027
      IF (MXL.NE.0) LML=MXL*LMP+100                                        000028
      IF (MXL.EQ.0) MXL=LML+100                                            000029
      CALL DATE (DAT)                                                      000030
      CALL TIME (TIM)                                                      000031
      CALL MMX00B                                                          000032
      WRITE (LCH,10) N,M,L,LEQ,DX,EPS,MAXF,KEQS,IW,IPR                     000033
   10 FORMAT (11H0INPUT DATA/11H ----------//                              000034
     1 27H   NUMBER OF VARIABLES (N)  ,25(2H. ),I4//                       000035
     2 27H   NUMBER OF FUNCTIONS (M)  ,25(2H. ),I4//                       000036
     3 43H   TOTAL NUMBER OF LINEAR CONSTRAINTS (L)   ,17(2H. ),I4//       000037
     4 41H   NUMBER OF EQUALITY CONSTRAINTS (LEQ)   ,18(2H. ),I4//         000038
     5 21H   STEP LENGTH (DX)   ,25(2H. ),1PE10.3//                        000039
     6 19H   ACCURACY (EPS)   ,26(2H. ),1PE10.3//                          000040
     7 45H   MAX NUMBER OF FUNCTION EVALUATIONS (MAXF)  ,16(2H. ),I4//     000041
     8 43H   NUMBER OF SUCCESSIVE ITERATIONS (KEQS)   ,17(2H. ),I4//       000042
     9 22H   WORKING SPACE (IW)  ,26(2H. ),1H.,I6//                        000043
     * 26H   PRINTOUT CONTROL (IPR)  ,24(2H. ),1H.,I6/)                    000044
      NRL=NRL-24                                                           000045
      LML=LML-24                                                           000046
      IF (LV2.NE.0.OR.LV1.EQ.1) GO TO 30                                   000047
      WRITE (LCH,20)                                                       000048
   20 FORMAT (19H   STARTING POINT :)                                      000049
      NRL=NRL-1                                                            000050
      LML=LML-1                                                            000051
      CALL FDF (N,M,X,W(M+1),W(1))                                         000052
      CALL MMX00V (MMX00B,X,N,W,M)                                         000053
      IF (LG2.NE.0) CALL MMX00G (MMX00B,W(M+1),M,N)                        000054
   30 IF (IPR.GE.0) GO TO 40                                               000055
      I=M*N+M+1                                                            000056
      J=I+M                                                                000057
      K=J+M                                                                000058
      CALL MMX00H (MMX00B,FDF,N,M,X,W(M+1),W(1),W(J),W(K),W(I))            000059
   40 CALL SECOND (TBEG)                                                   000060
      CALL MMLC8A (MMX00Q,MMX00B,FDF,N,M,L,LEQ,C,DC,IC,X,DX,EPS,MAXF,KEQ   000061
     1S,W,IW,IFALL)                                                        000062
      CALL SECOND (TEND)                                                   000063
      IF (LCH.LE.0) RETURN                                                 000064
      IF (IFALL.EQ.-1) GO TO 90                                            000065
```

```
      IF (IFALL.EQ.-2) GO TO 70                                          000066
      IF (NRL.LT.9) CALL MMX00B                                          000067
      WRITE (LCH,50)                                                     000068
   50 FORMAT (//9H SOLUTION/9H --------)                                 000069
      NRL=NRL-4                                                          000070
      LML=LML-4                                                          000071
      CALL MMX00V (MMX00B,X,N,W,M)                                       000072
      CPU=TEND-TBEG                                                      000073
      IF (NRL.LT.9) CALL MMX00B                                          000074
      WRITE (LCH,60) IFALL,MAXF,KEQS,CPU                                 000075
   60 FORMAT (//29H   TYPE OF SOLUTION (IFALL)  ,24(2H. ),I4//           000076
     1 35H   NUMBER OF FUNCTION EVALUATIONS  ,21(2H. ),I4//              000077
     2 31H   NUMBER OF SHIFTS TO STAGE-2 ,23(2H. ),I4//                  000078
     3 31H   EXECUTION TIME (IN SECONDS) ,21(2H. ),1H.,F7.3/)            000079
      RETURN                                                             000080
   70 WRITE (LCH,80)                                                     000081
   80 FORMAT (//42H E M P T Y   F E A S I B L E   R E G I O N/)          000082
      RETURN                                                             000083
   90 WRITE (LCH,100)                                                    000084
  100 FORMAT (//40H I N C O R R E C T   P A R A M E T E R S/)            000085
      RETURN                                                             000086
      END                                                                000087
C                                                                        000088
C                                                                        000089
      SUBROUTINE MMLA1Q (FDF,N,M,L,LEQ,C,DC,IC,X,DX,EPS,MAXF,KEQS,W,IW,I 000090
     1FALL)                                                              000091
      EXTERNAL FDF,MMX00Z                                                000092
C                                                                        000093
C          LEVEL 2 INTERFACE (J.HALD ENTRY)                              000094
C                                                                        000095
      DIMENSION C(1), DC(1,1), X(1), W(1)                                000096
      CALL MMLC8A (MMX00Z,MMX00Z,FDF,N,M,L,LEQ,C,DC,IC,X,DX,EPS,MAXF,KEQ 000097
     1S,W,IW,IFALL)                                                      000098
      RETURN                                                             000099
      END                                                                000100
C                                                                        000101
C                                                                        000102
      SUBROUTINE MMX00Z (FUN,N,M,X,DF,F,K,NS)                            000103
C                                                                        000104
C     DUMMY SUBROUTINE WHICH FOR BASIC AND ORIGINAL ENTRIES SUBSTITUTES  000105
C     SUBROUTINE MMX00Q/11Q.                                             000106
C                                                                        000107
      EXTERNAL FUN                                                       000108
      DIMENSION X(N), DF(M,N), F(M)                                      000109
      RETURN                                                             000110
      END                                                                000111
C                                                                        000112
C                                                                        000113
      SUBROUTINE MMX00Q (FHH,N,M,X,DF,F,K,NS)                            000114
C                                                                        000115
C     PRINT RESULTS OF FUNCTION EVALUATION.                              000116
C                                                                        000117
      EXTERNAL FHH                                                       000118
      DIMENSION X(N), DF(M,N), F(M)                                      000119
      COMMON /MMX000/ LCH,LV1,LV2,LG1,LG2,LMF,LMV,NRP,NRL,MXL,LMP,LML,LG 000120
     1H,DAT,TIM,LHT,H(8)                                                 000121
      IF (LCH.LE.0) RETURN                                               000122
      IF (LV1+LV2.EQ.0) RETURN                                           000123
      IF (K.LE.LV2) GO TO 10                                             000124
      IF (LV1.EQ.0) RETURN                                               000125
      IF (MOD(K,LV1).NE.0) RETURN                                        000126
   10 IF (NRP.LE.LMP.AND.LML.GE.0) GO TO 30                              000127
      LV1=0                                                              000128
      LV2=0                                                              000129
      WRITE (LCH,20)                                                     000130
```

```
   20 FORMAT (//26H ( LISTING LIMIT REACHED )//)                          000131
      NRL=NRL-5                                                           000132
      LML=LML-5                                                           000133
      RETURN                                                              000134
   30 IF (NRL.LT.7) CALL FHH                                              000135
      WRITE (LCH,40) K,NS                                                 000136
   40 FORMAT (22H0FUNCTION EVALUATION :,I4,2H /,I2)                       000137
      NRL=NRL-2                                                           000138
      LML=LML-2                                                           000139
      CALL MMX00V (FHH,X,N,F,M)                                           000140
      IF (LG1+LG2.EQ.0) RETURN                                            000141
      IF (K.LE.LG2) GO TO 50                                              000142
      IF (K.LE.LV2) RETURN                                                000143
      IF (LG1.EQ.0) RETURN                                                000144
      IF (MOD(K,LG1).NE.0) RETURN                                         000145
   50 CALL MMX00G (FHH,DF,M,N)                                            000146
      RETURN                                                              000147
      END                                                                 000148
C                                                                         000149
C                                                                         000150
      SUBROUTINE MMX00V (FHH,X,N,F,M)                                     000151
C                                                                         000152
C     PRINT VALUES OF VARIABLES AND RESIDUAL FUNCTIONS.                   000153
C                                                                         000154
      DIMENSION X(N), F(M)                                                000155
      COMMON /MMX000/ LCH,LV1,LV2,LG1,LG2,LMF,LMV,NRP,NRL,MXL,LMP,LML,LG  000156
     1H,DAT,TIM,LHT,H(8)                                                  000157
      IF (LCH.LE.0) RETURN                                                000158
      K=MAX0(N,M)                                                         000159
      IF (NRL.LT.5) CALL FHH                                              000160
      WRITE (LCH,10)                                                      000161
   10 FORMAT (/30X,9HVARIABLES,18X,15HFUNCTION VALUES/)                   000162
      NRL=NRL-3                                                           000163
      LML=LML-3                                                           000164
      DO 40 I=1,K                                                         000165
      IF (NRL.LE.0) CALL FHH                                              000166
      IF (I.LE.N.AND.I.LE.M) WRITE (LCH,20) I,X(I),I,F(I)                 000167
      IF (I.LE.N.AND.I.GT.M) WRITE (LCH,20) I,X(I)                        000168
      IF (I.GT.N.AND.I.LE.M) WRITE (LCH,30) I,F(I)                        000169
   20 FORMAT (18X,I4,2X,1PE19.12,5X,I4,2X,1PE19.12)                       000170
   30 FORMAT (48X,I4,2X,1PE19.12)                                         000171
      NRL=NRL-1                                                           000172
      LML=LML-1                                                           000173
   40 CONTINUE                                                            000174
      RETURN                                                              000175
      END                                                                 000176
C                                                                         000177
C                                                                         000178
      SUBROUTINE MMX00G (FHH,G,M,N)                                       000179
C                                                                         000180
C     PRINT PARTIAL DERIVATIVES OF RESIDUAL FUNCTIONS.                    000181
C                                                                         000182
      DIMENSION G(M,N)                                                    000183
      COMMON /MMX000/ LCH,LV1,LV2,LG1,LG2,LMF,LMV,NRP,NRL,MXL,LMP,LML,LG  000184
     1H,DAT,TIM,LHT,H(8)                                                  000185
      IF (LCH.LE.0) RETURN                                                000186
      IF (NRL.LT.7) CALL FHH                                              000187
      MM=MIN0(M,LMF)                                                      000188
      NN=MIN0(N,LMV)                                                      000189
      WRITE (LCH,10)                                                      000190
   10 FORMAT (30H0  GRADIENTS ( DF.I / DX.J ) :)                          0001
      NRL=NRL-2                                                           000192
      LML=LML-2                                                           000193
      DO 60 K=1,NN,LCH                                                    000194
      IF (NRL.LT.5) CALL FHH                                              000195
```

```
      J1=K                                                                  000196
      J2=MIN0(NN,K+LGH-1)                                                   000197
      WRITE (LCH,20) (J,J=J1,J2)                                            000198
   20 FORMAT (1H0,9X,12HVARIABLES(J),10(I5,5X))                             000199
      WRITE (LCH,30)                                                        000200
   30 FORMAT (10X,12HFUNCTIONS(I))                                          000201
      NRL=NRL-3                                                             000202
      LML=LML-3                                                             000203
      DO 50 I=1,MM                                                          000204
      IF (NRL.LE.0) CALL FHH                                                000205
      WRITE (LCH,40) I,(G(I,J),J=J1,J2)                                     000206
   40 FORMAT (10X,I6,4X,10(1PE10.2))                                        000207
      NRL=NRL-1                                                             000208
      LML=LML-1                                                             000209
   50 CONTINUE                                                              000210
   60 CONTINUE                                                              000211
      RETURN                                                                000212
      END                                                                   000213
C                                                                           000214
C                                                                           000215
      SUBROUTINE MMX00H (FHH,FDF,N,M,X,DF,F,DG,DH,G)                        000216
C                                                                           000217
C     NUMERICAL VERIFICATION OF USER-DEFINED PARTIAL DERIVATIVES            000218
C     (VARIABLES ARE DISTURBED ONE BY ONE).                                 000219
C                                                                           000220
      DIMENSION X(N), DF(M,N), F(M), DG(M), DH(M,N), G(M)                   000221
      COMMON /MMX000/ LCH,LV1,LV2,LG1,LG2,LMF,LMV,NRP,NRL,MXL,LMP,LML,LG    000222
     1H,DAT,TIM,LHT,H(8)                                                    000223
      IF (LCH.LE.0) RETURN                                                  000224
      K=0                                                                   000225
      CALL FDF (N,M,X,DF,F)                                                 000226
      DO 60 I=1,N                                                           000227
      Z=X(I)                                                                000228
      DX=1.E-6*Z                                                            000229
      IF (ABS(DX).LT.1.E-10) DX=1.E-10                                      000230
      DX2=DX+DX                                                             000231
      X(I)=Z+DX                                                             000232
      CALL FDF (N,M,X,DH,F)                                                 000233
      DO 10 J=1,M                                                           000234
      DG(J)=DH(J,I)                                                         000235
   10 CONTINUE                                                              000236
      X(I)=Z-DX                                                             000237
      CALL FDF (N,M,X,DH,G)                                                 000238
      X(I)=Z                                                                000239
      DO 50 J=1,M                                                           000240
      Y=DF(J,I)                                                             000241
      Z=F(J)-G(J)                                                           000242
      IF (ABS(Z).LE.0.5E-13*(F(J)+G(J))) Z=0.0                              000243
      Z=Z/DX2                                                               000244
      IF (ABS(Y).LE.1.E-20.AND.ABS(Z).LE.1.E-20) GO TO 50                   000245
      IF (ABS(Z).LT.1.E-20) Z=SIGN(1.E-20,Z)                                000246
      R=100.0*ABS((Z-Y)/Z)                                                  000247
      IF (R.LE.1.0) GO TO 50                                                000248
      IF (SIGN(1.0,DG(J))+SIGN(1.0,DH(J,I)).EQ.0.0) GO TO 50                000249
      IF (K.NE.0) GO TO 30                                                  000250
      IF (NRL.LT.5) CALL FHH                                                000251
      WRITE (LCH,20)                                                        000252
   20 FORMAT (38H0VERIFICATION OF PARTIAL DERIVATIVES :/                    000253
     1 1H0,18X,52H DF.I / DX.J  :  USER DEFINED  NUMERICAL  DIFFERENCE)     000254
      NRL=NRL-4                                                             000255
      LML=LML-4                                                             000256
   30 K=K+1                                                                 000257
      IF (NRL.LE.0) CALL FHH                                                000258
      WRITE (LCH,40) J,I,Y,Z,R                                              000259
   40 FORMAT (19X,I5,3X,I4,6X,1PE10.3,2X,1PE10.3,4X,0PF6.1,2H %)            000260
```

```
      NRL=NRL-1                                                          000261
      LML=LML-1                                                          000262
   50 CONTINUE                                                           000263
   60 CONTINUE                                                           000264
      IF (K.NE.0) GO TO 80                                               000265
      IF (NRL.LT.2) CALL FHH                                             000266
      WRITE (LCH,70)                                                     000267
   70 FORMAT (47H0VERIFICATION OF PARTIAL DERIVATIVES PERFORMED.)         000268
      NRL=NRL-2                                                          000269
      LML=LML-2                                                          000270
   80 RETURN                                                             000271
      END                                                                000272
C                                                                        000273
C                                                                        000274
      SUBROUTINE MMX00B                                                  000275
C                                                                        000276
C     CHANGE PAGE AND PRINT PAGE HEADER.                                 000277
C                                                                        000278
      COMMON /MMX000/ LCH,LV1,LV2,LG1,LG2,LMF,LMV,NRP,NRL,MXL,LMP,LML,LG 000279
     1H,DAT,TIM,LHT,H(8)                                                 000280
      IF (LCH.LE.0) RETURN                                               000281
      IF (NRP.LT.LMP) GO TO 20                                           000282
      LV1=0                                                              000283
      LV2=0                                                              000284
      WRITE (LCH,10)                                                     000285
   10 FORMAT (//27H ( LIMIT OF PAGES REACHED ))                          000286
   20 NRP=NRP+1                                                          000287
      NRL=MXL-5                                                          000288
      LML=LML-5                                                          000289
      WRITE (LCH,30) DAT,TIM,NRP                                         000290
   30 FORMAT (1H1/7H DATE :,A10,19X,6HTIME :,A10,20X,6HPAGE :,I3/        000291
     1 57H LINEARLY CONSTRAINED MINIMAX OPTIMIZATION (MMLC PACKAGE),15X, 000292
     2 9H(V:82.04))                                                      000293
      IF (LHT.LE.0) GO TO 50                                             000294
      WRITE (LCH,40) (H(J),J=1,LHT)                                      000295
   40 FORMAT (1H0,8A10)                                                  000296
      NRL=NRL-2                                                          000297
      LML=LML-2                                                          000298
   50 WRITE (LCH,60)                                                     000299
   60 FORMAT (1H0)                                                       000300
      RETURN                                                             000301
      END                                                                000302
C                                                                        000303
C                                                                        000304
      SUBROUTINE MMXPSZ (L)                                              000305
C                                                                        000306
C     DEFINE THE PAGE SIZE (I.E. THE NUMBER OF LINES PER PAGE).          000307
C                                                                        000308
      COMMON /MMX000/ LCH,LV1,LV2,LG1,LG2,LMF,LMV,NRP,NRL,MXL,LMP,LML,LG 000309
     1H,DAT,TIM,LHT,H(8)                                                 000310
      DATA LL/65/                                                        000311
      IF (L.GT.0) LL=MAX0(25,L)                                          000312
      IF (L.EQ.0) LL=0                                                   000313
      MXL=LL                                                             000314
      RETURN                                                             000315
      END                                                                000316
C                                                                        000317
C                                                                        000318
      SUBROUTINE MMXPLM (L)                                              000319
C                                                                        000320
C     DEFINE THE LIMIT OF PRINTED PAGES.                                 000321
C                                                                        000322
      COMMON /MMX000/ LCH,LV1,LV2,LG1,LG2,LMF,LMV,NRP,NRL,MXL,LMP,LML,LG 000323
     1H,DAT,TIM,LHT,H(8)                                                 000324
      DATA LL/10/                                                        000325
```

```
      IF (L.GT.0) LL=MIN0(50,L)                                          000326
      LMP=LL                                                             000327
      RETURN                                                             000328
      END                                                                000329
C                                                                        000330
C                                                                        000331
      SUBROUTINE MMXLLM (L)                                              000332
C                                                                        000333
C     DEFINE THE LIMIT OF PRINTED LINES.                                 000334
C                                                                        000335
      COMMON /MMX000/ LCH,LV1,LV2,LG1,LG2,LMF,LMV,NRP,NRL,MXL,LMP,LML,LG 000336
     1H,DAT,TIM,LHT,H(8)                                                 000337
      DATA LL/750/                                                       000338
      IF (L.GT.0) LL=L                                                   000339
      LML=LL                                                             000340
      RETURN                                                             000341
      END                                                                000342
C                                                                        000343
C                                                                        000344
      SUBROUTINE MMXHDR (L,T)                                            000345
C                                                                        000346
C     DEFINE THE HEADER LINE.                                            000347
C                                                                        000348
      DIMENSION T(1)                                                     000349
      COMMON /MMX000/ LCH,LV1,LV2,LG1,LG2,LMF,LMV,NRP,NRL,MXL,LMP,LML,LG 000350
     1H,DAT,TIM,LHT,H(8)                                                 000351
      DATA LL/0/                                                         000352
      IF (L.GE.0) LL=MIN0(8,L)                                           000353
      LHT=LL                                                             000354
      IF (L.LE.0) RETURN                                                 000355
      DO 10 I=1,LL                                                       000356
      H(I)=T(I)                                                          000357
   10 CONTINUE                                                           000358
      RETURN                                                             000359
      END                                                                000360
C                                                                        000361
C                                                                        000362
      SUBROUTINE MMXGLM (K,L)                                            000363
C                                                                        000364
C     DEFINE THE SIZE OF PRINTED JACOBIAN.                               000365
C                                                                        000366
      COMMON /MMX000/ LCH,LV1,LV2,LG1,LG2,LMF,LMV,NRP,NRL,MXL,LMP,LML,LG 000367
     1H,DAT,TIM,LHT,H(8)                                                 000368
      DATA KK/25/,LL/10/                                                 000369
      IF (K.GT.0) KK=K                                                   000370
      IF (L.GT.0) LL=L                                                   000371
      LMF=KK                                                             000372
      LMV=LL                                                             000373
      RETURN                                                             000374
      END                                                                000375
C                                                                        000376
C                                                                        000377
      SUBROUTINE MMXGVL (L)                                              000378
C                                                                        000379
C     DEFINE THE NUMBER OF JACOBIAN COLUMNS PRINTED IN ONE LINE.         000380
C                                                                        000381
      COMMON /MMX000/ LCH,LV1,LV2,LG1,LG2,LMF,LMV,NRP,NRL,MXL,LMP,LML,LG 000382
     1H,DAT,TIM,LHT,H(8)                                                 000383
      DATA LL/10/                                                        000384
      IF (L.GT.0) LL=MAX0(MIN0(10,L),5)                                  000385
      LGH=LL                                                             000386
      RETURN                                                             000387
      END                                                                000388
C                                                                        000389
C                                                                        000390
```

```
      SUBROUTINE MMLC8A (FQQ,FHH,FDF,N,M,L,LEQ,C,DC,IC,X,DX,EPS,MAXF,KEQ   000391
     1S,W,IW,IFALL)                                                        000392
C                                                                          000393
C     MMLC8A MINIMIZES THE MAXIMUM VALUE OF A SET OF NONLINEAR FUNCTIONS   000394
C     SUBJECT TO LINEAR EQUALITY AND INEQUALITY CONSTRAINTS. DERIVATIVES   000395
C     OF NONLINEAR FUNCTIONS ARE REQUIRED.                                 000396
C                                                                          000397
C     FOR A PROGRAM DESCRIPTION SEE:                                       000398
C     J. HALD: "MMLA1Q, A FORTRAN SUBROUTINE FOR LINEARLY CONSTRAINED      000399
C     MINIMAX OPTIMIZATION", REPORT NO. NI-81-1, INSTITUTE FOR NUMERICAL   000400
C     ANALYSIS, TECHNICAL UNIVERSITY OF DENMARK, DK-2800 LYNGBY, DENMARK   000401
C                                                                          000402
C     THE SUBROUTINES: MMLPA,FEASI,S2LA1Q,BFGS,ADDCL,DELCL,HACUM,          000403
C     UTRNS,UTTRNS,RSOLV,TSOLV,LIMIT,LINSYS  MUST BE AVAILABLE.            000404
C                                                                          000405
      DIMENSION C(1), DC(1,1), X(1), W(1)                                  000406
      EXTERNAL FQQ,FHH,FDF                                                 000407
      COMMON /MML000/ MARK                                                 000408
      DATA ZERO/0.0/                                                       000409
      MARK=1                                                               000410
C                                                                          000411
C       CHECK INPUT QUANTITIES                                             000412
C                                                                          000413
      IWR=2*M*N+5*N*N+4*M+8*N+4*IC+3                                       000414
      IFALL=-1                                                             000415
      IF (IW.LT.IWR.OR.N.LT.1.OR.M.LT.1.OR.L.LT.0.OR.LEQ.LT.0.OR.LEQ.GT.   000416
     1L.OR.LEQ.GT.N.OR.IC.LT.L.OR.DX.LE.ZERO.OR.EPS.LT.ZERO.OR.MAXF.LE.0   000417
     2) GO TO 10                                                           000418
C                                                                          000419
C       SPLIT UP THE WORK AREA                                             000420
C                                                                          000421
      N1=N+1                                                               000422
      NN=N+N                                                               000423
      NF=1                                                                 000424
      NF1=NF+M                                                             000425
      NDF=NF1+M                                                            000426
      NDF1=NDF+M*N                                                         000427
      NX1=NDF1+M*N                                                         000428
      NB=NX1+N                                                             000429
      NU=NB+N*N                                                            000430
      NR=NU+N*N                                                            000431
      NA=NU                                                                000432
      NCL=NA+NN*NN                                                         000433
      NWL=NCL+IC                                                           000434
      NWL1=NWL+IC                                                          000435
      NXX=NWL1+IC                                                          000436
      NW=NXX+N                                                             000437
      NW1=NW+N                                                             000438
      NW2=NW1+N                                                            000439
      NWM=NW2+N                                                            000440
      NAS=NWM+M                                                            000441
      NKS=NAS+N1                                                           000442
      NKS0=NKS+N1                                                          000443
      NKSTC=NKS0+N1                                                        000444
      NKSTF=NKSTC+IC                                                       000445
      IL=MAX0(1,IC)                                                        000446
      CALL MMLC9A (FQQ,FHH,FDF,N,M,L,LEQ,C,DC,IL,X,DX,EPS,MAXF,KEQS,N1,N   000447
     1N,W(NF),W(NF1),W(NDF),W(NDF1),W(NX1),W(NB),W(NU),W(NR),W(NA),W(NCL   000448
     2),W(NWL),W(NWL1),W(NXX),W(NW),W(NW1),W(NW2),W(NWM),W(NAS),W(NKS),W   000449
     3(NKS0),W(NKSTC),W(NKSTF),IFALL)                                      000450
      IF (IFALL.LT.0) GO TO 10                                             000451
      RETURN                                                               000452
   10 MAXF=0                                                               000453
      KEQS=0                                                               000454
      RETURN                                                               000455
```

```
      END                                                                   000456
C                                                                           000457
C                                                                           000458
      SUBROUTINE MMLC9A (FQQ,FHH,FDF,N,M,L,LEQ,C,DC,IC,X,DX,EPS,MAXF,KEQ    000459
     1S,N1,NN,F,F1,DF,DF1,X1,B,U,R,A,CLOC,WL,WL1,XX,W,W1,W2,WM,ASET,KSET   000460
     2,KSET0,KSTATC,KSTATF,IFALL)                                           000461
      DIMENSION C(IC), DC(IC,N), X(N), F(M), F1(M), WM(M), DF(M,N), DF1(    000462
     1M,N), XX(NN), X1(N), U(N,N), R(N,N), B(N,N), A(NN,NN), CLOC(IC), W    000463
     2L(IC), WL1(IC), W(N), W1(N), W2(N), ASET(N1)                          000464
      INTEGER KSET(N1),KSET0(N1),KSTATC(IC),KSTATF(M)                       000465
      LOGICAL DIV4,ACCUM,SHIFT                                              000466
      EXTERNAL FQQ,FHH,FDF                                                  000467
      COMMON /MML000/ MARK                                                  000468
      DATA XZERO,XONE,XP73,XM50/0.0,1.0,1.E73,1.E-50/                       000469
C                                                                           000470
C       SEPS IS AN EXPRESSION FOR THE MACHINE ACCURACY                      000471
C                                                                           000472
      SEPS=2.0*16.0**(-12)                                                  000473
C                                                                           000474
C       SET SOME CONSTANTS                                                  000475
C                                                                           000476
      DIV4=.FALSE.                                                          000477
      LI=L-LEQ                                                              000478
C                                                                           000479
C       INITIALIZE                                                          000480
C                                                                           000481
      KEQSET=0                                                              000482
      NCALL=0                                                               000483
      NSHIFT=0                                                              000484
      NSTEP=0                                                               000485
      FMMREF=XP73                                                           000486
      DX0=DX                                                                000487
      DO 20 I=1,N                                                           000488
      DO 10 J=1,N                                                           000489
      B(I,J)=XZERO                                                          000490
   10 CONTINUE                                                              000491
      B(I,I)=XONE                                                           000492
   20 CONTINUE                                                              000493
C                                                                           000494
C       FIND A FEASIBLE POINT                                               000495
C                                                                           000496
      IF (L.EQ.0) GO TO 80                                                  000497
      DO 40 I=1,L                                                           000498
      T=C(I)                                                                000499
      DO 30 J=1,N                                                           000500
      T=T+DC(I,J)*X(J)                                                      000501
   30 CONTINUE                                                              000502
      CLOC(I)=T                                                             000503
   40 CONTINUE                                                              000504
      NACT=0                                                                000505
      CALL FEASI (CLOC,DC,IC,LEQ,LI,N,XX,NACT,KSET,ASET,U,R,W1,W2,WL,WL1    000506
     1,W,KSTATC,IFALL,ACCUM,SEPS)                                           000507
      IF (IFALL.NE.0) RETURN                                                000508
      DO 50 I=1,N                                                           000509
      X(I)=X(I)+XX(I)                                                       000510
   50 CONTINUE                                                              000511
      DO 70 I=1,L                                                           000512
      T=C(I)                                                                000513
      DO 60 J=1,N                                                           000514
      T=T+DC(I,J)*X(J)                                                      000515
   60 CONTINUE                                                              000516
      CLOC(I)=T                                                             000517
   70 CONTINUE                                                              000518
C                                                                           000519
C       CALCULATE FUNCTION VALUES IN THE FIRST FEASIBLE POINT               000520
```

```
C                                                                       000521
   80 CALL FDF (N,M,X,DF,F)                                             000522
      CALL FQQ (FHH,N,M,X,DF,F,1,0)                                     000523
      FMM0=F(1)                                                         000524
      DO 90 I=1,M                                                       000525
      FMM0=AMAX1(FMM0,F(I))                                             000526
   90 CONTINUE                                                          000527
      XN=XZERO                                                          000528
      DO 100 I=1,N                                                      000529
      XN=XN+X(I)*X(I)                                                   000530
  100 CONTINUE                                                          000531
      XN=SQRT(XN)                                                       000532
      NACT0=0                                                           000533
      NCALL=1                                                           000534
C                                                                       000535
C       ITERATIVE LOOP STARTS HERE                                      000536
C                                                                       000537
  110 NACT=NACT0                                                        000538
      IF (NACT.EQ.0) GO TO 130                                          000539
      DO 120 I=1,NACT                                                   000540
      KSET(I)=KSET0(I)                                                  000541
  120 CONTINUE                                                          000542
C                                                                       000543
C       SOLVE THE LINEAR SUBPROBLEM                                     000544
C                                                                       000545
  130 CALL MMLPA (F,DF,CLOC,DC,M,N,N1,IC,LEQ,LI,DX,XXN,XX,NACT,KSET,ASET 000546
     1,U,R,W1,W2,F1,WM,WL,WL1,KSTATF,KSTATC,W,SEPS,ACCUM,FMM,IFALL)      000547
      IF (FMM.GE.FMM0) GO TO 400                                        000548
C                                                                       000549
C       CALCULATE FUNCTION VALUES IN THE NEW POINT                      000550
C                                                                       000551
      DO 140 I=1,N                                                      000552
      X1(I)=X(I)+XX(I)                                                  000553
  140 CONTINUE                                                          000554
      CALL FDF (N,M,X1,DF1,F1)                                          000555
      NCALL=NCALL+1                                                     000556
      CALL FQQ (FHH,N,M,X1,DF1,F1,NCALL,1)                              000557
      IF (MARK.EQ.0) GO TO 410                                          000558
      FMM1=F1(1)                                                        000559
      DO 150 I=1,M                                                      000560
      FMM1=AMAX1(FMM1,F1(I))                                            000561
  150 CONTINUE                                                          000562
C                                                                       000563
C       REVISE THE STEP LENGTH                                          000564
C                                                                       000565
      IF ((FMM0-FMM1).GT.0.25*(FMM0-FMM)) GO TO 160                     000566
      DX=0.25*XXN                                                       000567
      DIV4=.TRUE.                                                       000568
      GO TO 180                                                         000569
  160 IF (DIV4) GO TO 170                                               000570
      IF ((FMM0-FMM1).GT.0.75*(FMM0-FMM)) DX=XXN+XXN                    000571
  170 DIV4=.FALSE.                                                      000572
C                                                                       000573
C       UPDATE THE HESSIAN APPROXIMATION                                000574
C                                                                       000575
  180 DO 190 J=1,N                                                      000576
      W(J)=XZERO                                                        000577
      W1(J)=XZERO                                                       000578
  190 CONTINUE                                                          000579
      DO 210 I=1,NACT                                                   000580
      K=KSET(I)                                                         000581
      IF (K.LE.L) GO TO 210                                             000582
      KK=K-L                                                            000583
      T=-ASET(I)                                                        000584
      DO 200 J=1,N                                                      000585
```

```
      W1(J)=W1(J)+T*DF1(KK,J)                                           000586
      W(J)=W(J)+T*DF(KK,J)                                              000587
  200 CONTINUE                                                          000588
  210 CONTINUE                                                          000589
      DO 220 I=1,N                                                      000590
      W2(I)=W1(I)-W(I)                                                  000591
  220 CONTINUE                                                          000592
      CALL BFGS (B,N,W2,XX,W,SEPS)                                      000593
C                                                                       000594
C       TEST IF THE NEW POINT IS ACCEPTABLE                             000595
C                                                                       000596
      IF ((FMM0-FMM1).LE.0.01*(FMM0-FMM)) GO TO 320                     000597
C                                                                       000598
C       COMPARE THE NEW ACTIVE SET WITH THE PRECEDING                   000599
C                                                                       000600
      IF (NACT0.NE.NACT) GO TO 250                                      000601
      DO 240 I=1,NACT                                                   000602
      K=KSET(I)                                                         000603
      DO 230 J=1,NACT                                                   000604
      IF (K.EQ.KSET0(J)) GO TO 240                                      000605
  230 CONTINUE                                                          000606
      GO TO 250                                                         000607
  240 CONTINUE                                                          000608
      KEQSET=KEQSET+1                                                   000609
      GO TO 260                                                         000610
  250 KEQSET=1                                                          000611
C                                                                       000612
C       INTRODUCE THE NEW POINT                                         000613
C                                                                       000614
  260 NSTEP=NSTEP+1                                                     000615
      XN=XZERO                                                          000616
      FMM0=FMM1                                                         000617
      NACT0=NACT                                                        000618
      DO 270 I=1,N                                                      000619
      X(I)=X1(I)                                                        000620
      XN=XN+X(I)**2                                                     000621
      DO 270 J=1,M                                                      000622
  270 DF(J,I)=DF1(J,I)                                                  000623
      XN=SQRT(XN)                                                       000624
      DO 280 I=1,M                                                      000625
      F(I)=F1(I)                                                        000626
  280 CONTINUE                                                          000627
      DO 290 I=1,NACT0                                                  000628
      KSET0(I)=KSET(I)                                                  000629
  290 CONTINUE                                                          000630
      IF (L.EQ.0) GO TO 320                                             000631
      DO 310 I=1,L                                                      000632
      T=C(I)                                                            000633
      DO 300 J=1,N                                                      000634
      T=T+DC(I,J)*X(J)                                                  000635
  300 CONTINUE                                                          000636
      CLOC(I)=T                                                         000637
  310 CONTINUE                                                          000638
C                                                                       000639
C       TEST OF CONVERGENCE CRITERION                                   000640
C                                                                       000641
  320 IF (XXN.LE.EPS*XN) GO TO 410                                      000642
      IF (XXN.LE.SEPS*XN) GO TO 400                                     000643
      IF (XXN.LE.XM50) GO TO 410                                        000644
      IF (NCALL.GE.MAXF) GO TO 420                                      000645
C                                                                       000646
C       TEST FOR SWITCH TO STAGE-2                                      000647
C                                                                       000648
      SHIFT=FMM0.LE.FMMREF.AND.KEQSET.GE.KEQS.AND.NSTEP.GE.N            000649
      IF (.NOT.SHIFT) GO TO 110                                         000650
```

```
      IF (NACT.EQ.N1) GO TO 380                                          000651
C                                                                        000652
C       TEST FOR POSITIVE DEFINITENESS OF THE HESSIAN APPROX.            000653
C       IN A RELEVANT DIRECTION                                          000654
C                                                                        000655
      DO 340 I=1,NACT                                                    000656
      K=KSET(I)                                                          000657
      IF (K.GT.L) GO TO 340                                              000658
      T=ASET(I)                                                          000659
      DO 330 J=1,N                                                       000660
      W1(J)=W1(J)+T*DC(K,J)                                              000661
  330 CONTINUE                                                           000662
  340 CONTINUE                                                           000663
      DO 360 I=1,N                                                       000664
      T=XZERO                                                            000665
      DO 350 J=1,N                                                       000666
      T=T+B(I,J)*W1(J)                                                   000667
  350 CONTINUE                                                           000668
      W(I)=T                                                             000669
  360 CONTINUE                                                           000670
      T=XZERO                                                            000671
      DO 370 I=1,N                                                       000672
      T=T+W(I)*W1(I)                                                     000673
  370 CONTINUE                                                           000674
      IF (T.LE.XZERO) GO TO 110                                          000675
C                                                                        000676
C       SHIFT TO STAGE-2                                                 000677
C                                                                        000678
  380 NSHIFT=NSHIFT+1                                                    000679
      FMMREF=FMM0-10.0*SEPS*ABS(FMM0)                                    000680
      XXNMAX=AMAX1(DX0,DX+DX)                                            000681
      CALL S2LA1Q (FQQ,FHH,FDF,N,M,L,LEQ,C,CLOC,DC,IC,X,XXNMAX,B,NACT,KS 000682
     1ET,ASET,N1,KSTATF,KSTATC,A,XX,NN,F,DF,X1,F1,DF1,W1,W2,EPS,MAXF,NCA 000683
     2LL,XXN,NSTEP,SEPS,IFALL)                                           000684
      IF (IFALL.LE.4) GO TO 410                                          000685
      FMM0=-XP73                                                         000686
      DO 390 I=1,M                                                       000687
      FMM0=AMAX1(FMM0,F(I))                                              000688
  390 CONTINUE                                                           000689
      DX=AMAX1(DX,0.5*XXN)                                               000690
      KEQSET=1                                                           000691
      GO TO 110                                                          000692
C                                                                        000693
C       RETURN                                                           000694
C                                                                        000695
  400 IFALL=2                                                            000696
  410 MAXF=NCALL                                                         000697
      KEQS=NSHIFT                                                        000698
      EPS=XXN                                                            000699
      RETURN                                                             000700
  420 IFALL=3                                                            000701
      GO TO 410                                                          000702
      END                                                                000703
C                                                                        000704
C                                                                        000705
      SUBROUTINE S2LA1Q (FQQ,FHH,FDF,N,M,L,LEQ,C,CLOC,DC,IC,X,XXNMAX,B,N 000706
     1ACT,KSET,ASET,N1,KSTATF,KSTATC,DZ,ZZ,NN,F,DF,X1,F1,DF1,W,W1,EPS,MA 000707
     2XF,NCALL,XXN,NSTEP,SEPS,IFALL)                                     000708
C                                                                        000709
C     STAGE-2 (QUASI-NEWTON) ALGORITHM FOR LINEARLY CONSTRAINED          000710
C     MINIMAX OPTIMIZATION.                                              000711
C                                                                        000712
      DIMENSION C(IC), CLOC(IC), DC(IC,N), X(N), B(N,N), ASET(N1), DZ(NN 000713
     1,NN), ZZ(NN), F(M), DF(M,N), X1(N), F1(M), DF1(M,N), W(N), W1(N)   000714
      INTEGER KSET(N1),KSTATF(M),KSTATC(IC)                              000715
```

```
      EXTERNAL FQQ,FHH,FDF                                              000716
      COMMON /MML000/ MARK                                              000717
      DATA XZERO,XONE,XP73,XM50/0.0,1.0,1.E73,1.E-50/                   000718
C                                                                       000719
C       INITIALIZE                                                      000720
C                                                                       000721
      LI=L-LEQ                                                          000722
      LE1=LEQ+1                                                         000723
      IFALL=0                                                           000724
      SSEPS=SQRT(SEPS)                                                  000725
      KK0=KSET(NACT)-L                                                  000726
      NACT1=NACT-1                                                      000727
      NZ=N+NACT1                                                        000728
      NSTEP2=0                                                          000729
      XXN=XZERO                                                         000730
      DO 10 I=1,M                                                       000731
      KSTATF(I)=0                                                       000732
   10 CONTINUE                                                          000733
      IF (L.EQ.0) GO TO 30                                              000734
      DO 20 I=1,L                                                       000735
      KSTATC(I)=0                                                       000736
   20 CONTINUE                                                          000737
   30 DO 40 I=1,NACT                                                    000738
      K=KSET(I)                                                         000739
      IF (K.LE.L) KSTATC(K)=1                                           000740
      IF (K.GT.L) KSTATF(K-L)=1                                         000741
   40 CONTINUE                                                          000742
C                                                                       000743
C       ITERATIVE LOOP STARTS HERE                                      000744
C                                                                       000745
C       SET UP THE ITERATION MATRIX AND THE RIGHTHAND SIDE              000746
C                                                                       000747
   50 DO 70 I=1,N                                                       000748
      DO 60 J=1,N                                                       000749
      DZ(I,J)=B(I,J)                                                    000750
   60 CONTINUE                                                          000751
      ZZ(I)=-DF(KK0,I)                                                  000752
   70 CONTINUE                                                          000753
      IF (NACT.EQ.1) GO TO 150                                          000754
      DO 140 J=1,NACT1                                                  000755
      K=KSET(J)                                                         000756
      JN=J+N                                                            000757
      IF (K.GT.L) GO TO 90                                              000758
      ZZ(JN)=-CLOC(K)                                                   000759
      DO 80 I=1,N                                                       000760
      DZ(I,JN)=DC(K,I)                                                  000761
      DZ(JN,I)=DZ(I,JN)                                                 000762
   80 CONTINUE                                                          000763
      GO TO 110                                                         000764
   90 KK=K-L                                                            000765
      ZZ(JN)=F(KK)-F(KK0)                                               000766
      DO 100 I=1,N                                                      000767
      DZ(I,JN)=DF(KK0,I)-DF(KK,I)                                       000768
      DZ(JN,I)=DZ(I,JN)                                                 000769
  100 CONTINUE                                                          000770
  110 DO 120 I=N1,NZ                                                    000771
      DZ(I,JN)=XZERO                                                    000772
  120 CONTINUE                                                          000773
      T=ASET(J)                                                         000774
      DO 130 I=1,N                                                      000775
      ZZ(I)=ZZ(I)-T*DZ(JN,I)                                            000776
  130 CONTINUE                                                          000777
  140 CONTINUE                                                          000778
  150 RES0=XZERO                                                        000779
      DO 160 I=1,NZ                                                     000780
```

```
      RES0=RES0+ZZ(I)**2                                                  000781
  160 CONTINUE                                                            000782
      RES0=SQRT(RES0)                                                     000783
C                                                                         000784
C       CALCULATE THE QUASI-NEWTON STEP                                   000785
C                                                                         000786
      CALL LINSYS (DZ,ZZ,NN,NZ,K,SEPS)                                    000787
      IF (K.EQ.NZ) GO TO 170                                              000788
      IFALL=8                                                             000789
      RETURN                                                              000790
C                                                                         000791
C       CONTROL STEP LENGTH                                               000792
C                                                                         000793
  170 XXN1=XZERO                                                          000794
      ALFA=XONE                                                           000795
      DO 180 I=1,N                                                        000796
      XXN1=XXN1+ZZ(I)**2                                                  000797
  180 CONTINUE                                                            000798
      XXN1=SQRT(XXN1)                                                     000799
      IF (XXN1.GT.XXNMAX) ALFA=XXNMAX/XXN1                                000800
C                                                                         000801
C       WILL OTHER CONSTRAINTS OR FUNCTIONS BECOME ACTIVE ?               000802
C                                                                         000803
      STEP=XP73                                                           000804
      IF (LI.EQ.0) GO TO 210                                              000805
      DO 200 I=LE1,L                                                      000806
      IF (KSTATC(I).NE.0) GO TO 200                                       000807
      T=XZERO                                                             000808
      DO 190 J=1,N                                                        000809
      T=T+ZZ(J)*DC(I,J)                                                   000810
  190 CONTINUE                                                            000811
      IF (T.GE.XZERO) GO TO 200                                           000812
      T=-CLOC(I)/T                                                        000813
      IF (T.GT.STEP) GO TO 200                                            000814
      STEP=T                                                              000815
  200 CONTINUE                                                            000816
  210 T0=XZERO                                                            000817
      DO 220 I=1,N                                                        000818
      T0=T0+ZZ(I)*DF(KK0,I)                                               000819
  220 CONTINUE                                                            000820
      F0=F(KK0)                                                           000821
      DO 240 I=1,M                                                        000822
      IF (KSTATF(I).NE.0) GO TO 240                                       000823
      T=XZERO                                                             000824
      DO 230 J=1,N                                                        000825
      T=T+ZZ(J)*DF(I,J)                                                   000826
  230 CONTINUE                                                            000827
      T=T0-T                                                              000828
      IF (T.GE.XZERO) GO TO 240                                           000829
      T=(F(I)-F0)/T                                                       000830
      IF (T.GT.STEP) GO TO 240                                            000831
      STEP=T                                                              000832
  240 CONTINUE                                                            000833
      IF (STEP.GT.ALFA) GO TO 250                                         000834
      IFALL=9                                                             000835
      ALFA=STEP                                                           000836
C                                                                         000837
C       SCALE THE STEP                                                    000838
C                                                                         000839
  250 DO 260 I=1,NZ                                                       000840
      ZZ(I)=ALFA*ZZ(I)                                                    000841
  260 CONTINUE                                                            000842
      XXN1=ABS(ALFA)*XXN1                                                 000843
C                                                                         000844
C       CALCULATE FUNCTION VALUES AND RESIDUALS IN THE NEW POINT          000845
```

```
C                                                                       000846
      XN1=XZERO                                                         000847
      DO 270 I=1,N                                                      000848
      X1(I)=X(I)+ZZ(I)                                                  000849
      XN1=XN1+X1(I)**2                                                  000850
  270 CONTINUE                                                          000851
      XN1=SQRT(XN1)                                                     000852
      NCALL=NCALL+1                                                     000853
      CALL FDF (N,M,X1,DF1,F1)                                          000854
      CALL FQQ (FHH,N,M,X1,DF1,F1,NCALL,2)                              000855
      IF (MARK.EQ.0) GO TO 520                                          000856
      DASET0=XZERO                                                      000857
      IF (NACT.EQ.1) GO TO 290                                          000858
      DO 280 I=N1,NZ                                                    000859
      IF (KSET(I-N).GT.L) DASET0=DASET0-ZZ(I)                           000860
  280 CONTINUE                                                          000861
  290 RES=XZERO                                                         000862
      T=ASET(NACT)+DASET0                                               000863
      DO 300 I=1,N                                                      000864
      W(I)=-T*DF(KK0,I)                                                 000865
      W1(I)=-T*DF1(KK0,I)                                               000866
  300 CONTINUE                                                          000867
      IF (NACT.EQ.1) GO TO 350                                          000868
      DO 340 J=1,NACT1                                                  000869
      K=KSET(J)                                                         000870
      JN=J+N                                                            000871
      T=ASET(J)+ZZ(JN)                                                  000872
      IF (K.GT.L) GO TO 320                                             000873
      S=C(K)                                                            000874
      DO 310 I=1,N                                                      000875
      SS=DC(K,I)                                                        000876
      W(I)=W(I)+T*SS                                                    000877
      W1(I)=W1(I)+T*SS                                                  000878
      S=S+SS*X1(I)                                                      000879
  310 CONTINUE                                                          000880
      RES=RES+S**2                                                      000881
      GO TO 340                                                         000882
  320 KK=K-L                                                            000883
      DO 330 I=1,N                                                      000884
      W(I)=W(I)-T*DF(KK,I)                                              000885
      W1(I)=W1(I)-T*DF1(KK,I)                                           000886
  330 CONTINUE                                                          000887
      RES=RES+(F1(KK0)-F1(KK))**2                                       000888
  340 CONTINUE                                                          000889
  350 DO 360 I=1,N                                                      000890
      RES=RES+W1(I)**2                                                  000891
  360 CONTINUE                                                          000892
      RES=SQRT(RES)                                                     000893
C                                                                       000894
C       UPDATE THE HESSIAN APPROXIMATION                                000895
C                                                                       000896
      DO 370 I=1,N                                                      000897
      W1(I)=W1(I)-W(I)                                                  000898
  370 CONTINUE                                                          000899
      CALL BFGS (B,N,W1,ZZ,W,SEPS)                                      000900
C                                                                       000901
C       TEST IF THE RESIDUAL HAS DECREASED                              000902
C                                                                       000903
      IF (NSTEP2.EQ.0) GO TO 390                                        000904
      IF (RES.LE.0.999*RES0) GO TO 390                                  000905
C                                                                       000906
C       IF NO - TEST FOR MACHINE ACCURACY                               000907
C                                                                       000908
      IF (XXN1.GT.SSEPS*(XXNMAX+XN1).OR.NSTEP2.LT.2) GO TO 380          000909
      IFALL=2                                                           000910
```

```
      RETURN                                                            000911
  380 IFALL=5                                                           000912
      RETURN                                                            000913
C                                                                       000914
C       IF YES - INTRODUCE THE NEW POINT                                000915
C                                                                       000916
  390 NSTEP2=NSTEP2+1                                                   000917
      NSTEP=NSTEP+1                                                     000918
      XN=XZERO                                                          000919
      DO 400 I=1,N                                                      000920
      X(I)=X1(I)                                                        000921
      DO 400 J=1,M                                                      000922
  400 DF(J,I)=DF1(J,I)                                                  000923
      XN=XN1                                                            000924
      XXN=XXN1                                                          000925
      FMAX=-XP73                                                        000926
      DO 410 I=1,M                                                      000927
      T=F1(I)                                                           000928
      FMAX=AMAX1(T,FMAX)                                                000929
      F(I)=T                                                            000930
  410 CONTINUE                                                          000931
      ASET(NACT)=ASET(NACT)+DASET0                                      000932
      IF (ASET(NACT).GT.XZERO) IFALL=6                                  000933
      IF (NACT.EQ.1) GO TO 430                                          000934
      DO 420 I=1,NACT1                                                  000935
      IN=I+N                                                            000936
      ASET(I)=ASET(I)+ZZ(IN)                                            000937
      IF (KSET(I).GT.LEQ.AND.ASET(I).GT.XZERO) IFALL=6                  000938
  420 CONTINUE                                                          000939
  430 IF (L.EQ.0) GO TO 470                                             000940
      DO 450 J=1,L                                                      000941
      T=C(J)                                                            000942
      DO 440 I=1,N                                                      000943
      T=T+DC(J,I)*X(I)                                                  000944
  440 CONTINUE                                                          000945
      CLOC(J)=T                                                         000946
  450 CONTINUE                                                          000947
C                                                                       000948
C       TEST IF THE ACTIVE SET IS COMPLETE                              000949
C                                                                       000950
      T=FMAX+RES                                                        000951
      DO 460 I=1,M                                                      000952
      IF (F(I).LE.T) GO TO 460                                          000953
      IFALL=7                                                           000954
      RETURN                                                            000955
  460 CONTINUE                                                          000956
C                                                                       000957
C       TEST CONVERGENCE CRITERION                                      000958
C                                                                       000959
  470 IF (XXN.GT.EPS*XN) GO TO 480                                      000960
      IF (NACT.LT.N1) IFALL=1                                           000961
      RETURN                                                            000962
  480 IF (XXN.GT.SEPS*XN) GO TO 490                                     000963
      IFALL=2                                                           000964
      RETURN                                                            000965
  490 IF (XXN.GT.XM50) GO TO 500                                        000966
      IFALL=0                                                           000967
      RETURN                                                            000968
  500 IF (NCALL.LT.MAXF) GO TO 510                                      000969
      IFALL=3                                                           000970
      RETURN                                                            000971
  510 IF (IFALL.GT.4) RETURN                                            000972
      GO TO 50                                                          000973
  520 IFALL=4                                                           000974
      RETURN                                                            000975
```

```
      END                                                                   000976
C                                                                           000977
C                                                                           000978
      SUBROUTINE FEASI (C,DC,IC,LE,LI,N,X,NACT,KSET,ASET,U,R,DL,RIGHT,CU    000979
     1P,DLDC,W,KSTAT,IFALL,ACCUM,SEPS)                                      000980
C                                                                           000981
C     THE SUBROUTINE FINDS A FEASIBLE POINT FOR A SET OF LINEAR             000982
C     EQUALITY AND INEQUALITY CONSTRAINTS.                                  000983
C                                                                           000984
      DIMENSION C(IC), DC(IC,N), X(N), ASET(N), U(N,N), R(N,N), DL(N), R    000985
     1IGHT(N), CUP(IC), DLDC(IC), W(N)                                      000986
      INTEGER KSET(N),KSTAT(IC)                                             000987
      LOGICAL ACCUM,OBJECT                                                  000988
      DATA XZERO,XP73/0.0,1.E73/                                            000989
C                                                                           000990
C       INITIALIZE                                                          000991
C                                                                           000992
      EPS=(N+10)*SEPS                                                       000993
      ACCUM=.FALSE.                                                         000994
      NACTIN=NACT                                                           000995
      NACT=0                                                                000996
      LE1=LE+1                                                              000997
      LELI=LE+LI                                                            000998
      DO 10 I=1,N                                                           000999
      X(I)=XZERO                                                            001000
   10 CONTINUE                                                              001001
      IFALL=0                                                               001002
      IF (LELI.EQ.0) RETURN                                                 001003
      DO 20 I=1,LELI                                                        001004
      KSTAT(I)=0                                                            001005
   20 CONTINUE                                                              001006
C                                                                           001007
C       MAKE ACTIVE THE EQUALITY CONSTRAINTS PLUS OTHER CONSTRAINTS         001008
C       AS DEFINED IN KSET                                                  001009
C                                                                           001010
      IF (LE.EQ.0) GO TO 50                                                 001011
      IF (LE.GT.N) GO TO 410                                                001012
      DO 40 I=1,LE                                                          001013
      RIGHT(I)=-C(I)                                                        001014
      DO 30 J=1,N                                                           001015
      R(J,I)=DC(I,J)                                                        001016
   30 CONTINUE                                                              001017
      KSET(I)=I                                                             001018
      KSTAT(I)=1                                                            001019
   40 CONTINUE                                                              001020
      CALL ADDCL (U,R,N,NACT,LE,RIGHT,W,ACCUM,.FALSE.,EPS)                  001021
      IF (NACT.LT.LE) GO TO 410                                             001022
   50 IF (NACTIN.LT.1) GO TO 80                                             001023
      DO 70 K=1,NACTIN                                                      001024
      KK=KSET(K)                                                            001025
      IF (KK.LT.LE1.OR.KK.GT.LELI) GO TO 70                                 001026
      NACT1=NACT+1                                                          001027
      IF (NACT1.GT.N) GO TO 80                                              001028
      DO 60 I=1,N                                                           001029
      R(I,NACT1)=DC(KK,I)                                                   001030
   60 CONTINUE                                                              001031
      CALL UTTRNS (U,N,NACT,ACCUM,R(1,NACT1),W)                             001032
      CALL ADDCL (U,R,N,NACT,1,RIGHT,W,ACCUM,.FALSE.,EPS)                   001033
      IF (NACT.LT.NACT1) GO TO 70                                           001034
      RIGHT(NACT1)=-C(KK)                                                   001035
      KSET(NACT1)=KK                                                        001036
      KSTAT(KK)=1                                                           001037
   70 CONTINUE                                                              001038
   80 CALL TSOLV (R,N,NACT,RIGHT,X)                                         001039
      IF (NACT.EQ.N) GO TO 100                                              001040
```

```
      NACT1=NACT+1                                                      001041
      DO 90 I=NACT1,N                                                   001042
      X(I)=XZERO                                                        001043
   90 CONTINUE                                                          001044
  100 CALL UTRNS (U,N,NACT,ACCUM,X,W)                                   001045
C                                                                       001046
C       UPDATE THE CONSTRAINTS                                          001047
C                                                                       001048
      IF (LI.EQ.0) RETURN                                               001049
      DO 120 I=LE1,LELI                                                 001050
      T=C(I)                                                            001051
      DO 110 J=1,N                                                      001052
      T=T+DC(I,J)*X(J)                                                  001053
  110 CONTINUE                                                          001054
      CUP(I)=T                                                          001055
  120 CONTINUE                                                          001056
C                                                                       001057
C       INITIALIZE INEQUALITY CONSTRAINT LOOP                           001058
C                                                                       001059
      DO 130 I=LE1,LELI                                                 001060
      IF (CUP(I).LT.XZERO.AND.KSTAT(I).EQ.0) KSTAT(I)=-1                001061
  130 CONTINUE                                                          001062
C                                                                       001063
C       ACTIVATE VIOLATED INEQUALITY CONSTRAINTS ONE BY ONE             001064
C       USE THE STRONGEST VIOLATED AS OBJECTIVE CONSTRAINT              001065
C                                                                       001066
  140 FMIN=XP73                                                         001067
      DO 150 I=LE1,LELI                                                 001068
      IF (KSTAT(I).NE.-1) GO TO 150                                     001069
      IF (CUP(I).GE.FMIN) GO TO 150                                     001070
      FMIN=CUP(I)                                                       001071
      NEW=I                                                             001072
  150 CONTINUE                                                          001073
      IF (FMIN.GE.XZERO) RETURN                                         001074
      DO 160 I=1,N                                                      001075
      RIGHT(I)=DC(NEW,I)                                                001076
  160 CONTINUE                                                          001077
      CALL UTTRNS (U,N,NACT,ACCUM,RIGHT,W)                              001078
      KSTAT(NEW)=1                                                      001079
C                                                                       001080
C       CALCULATE MULTIPLIERS FOR THE NEW ACTIVE CONSTRAINT AND DROP    001081
C       CONSTRAINTS WITH POSITIVE MULTIPLIERS IN ORDER TO ACHIEVE       001082
C       THE DIRECTION OF STEEPEST INCREMENT                             001083
C                                                                       001084
  170 IF (NACT.EQ.LE) GO TO 220                                         001085
      CALL RSOLV (R,N,NACT,RIGHT,ASET)                                  001086
      AMAX=-XP73                                                        001087
      DO 180 I=LE1,NACT                                                 001088
      IF (ASET(I).LT.AMAX) GO TO 180                                    001089
      K=I                                                               001090
      AMAX=ASET(I)                                                      001091
  180 CONTINUE                                                          001092
      IF (AMAX.LT.XZERO) GO TO 220                                      001093
      KSTAT(KSET(K))=0                                                  001094
      DO 190 I=LE1,LELI                                                 001095
      IF (KSTAT(I).EQ.-2) KSTAT(I)=0                                    001096
  190 CONTINUE                                                          001097
      IF (ACCUM) GO TO 200                                              001098
      ACCUM=.TRUE.                                                      001099
      CALL HACUM (U,N,NACT,W)                                           001100
  200 CALL DELCL (K,U,R,N,NACT,RIGHT,.TRUE.)                            001101
      IF (K.GT.NACT) GO TO 170                                          001102
      DO 210 I=K,NACT                                                   001103
      KSET(I)=KSET(I+1)                                                 001104
  210 CONTINUE                                                          001105
```

```
      GO TO 170                                                          001106
C                                                                        001107
C       CALCULATE THE PROJECTED GRADIENT                                 001108
C                                                                        001109
  220 T=XZERO                                                            001110
      DLN2=XZERO                                                         001111
      IF (NACT.EQ.0) GO TO 240                                           001112
      DO 230 I=1,NACT                                                    001113
      T=T+RIGHT(I)**2                                                    001114
      DL(I)=XZERO                                                        001115
  230 CONTINUE                                                           001116
  240 NACT1=NACT+1                                                       001117
      IF (NACT.EQ.N) GO TO 260                                           001118
      DO 250 I=NACT1,N                                                   001119
      DLN2=DLN2+RIGHT(I)**2                                              001120
      DL(I)=RIGHT(I)                                                     001121
  250 CONTINUE                                                           001122
  260 T=T+DLN2                                                           001123
      IF (T.GT.XZERO.AND.DLN2.GT.EPS*EPS*T) GO TO 280                    001124
      S=(N+1)*ABS(C(NEW))                                                001125
      DO 270 I=1,N                                                       001126
      S=S+ABS(DC(NEW,I)*X(I))*(N+3-I)                                    001127
  270 CONTINUE                                                           001128
      IF (CUP(NEW).LT.-EPS*S) GO TO 410                                  001129
      KSTAT(NEW)=0                                                       001130
      GO TO 140                                                          001131
  280 CALL UTRNS (U,N,NACT,ACCUM,DL,W)                                   001132
C                                                                        001133
C       PROJECT GRADIENTS ON THE PROJECTED GRADIENT                      001134
C                                                                        001135
      DO 300 I=LE1,LELI                                                  001136
      T=XZERO                                                            001137
      DO 290 J=1,N                                                       001138
      T=T+DL(J)*DC(I,J)                                                  001139
  290 CONTINUE                                                           001140
      DLDC(I)=T                                                          001141
  300 CONTINUE                                                           001142
C                                                                        001143
C       CALCULATE STEP LENGTH "ANES" TO MAKE THE OBJECTIVE CONSTRAINT    001144
C       EQUAL ZERO, AND CALCULATE THE STEP LENGTH "AMIN" TO THE          001145
C       NEAREST INACTIVE CONSTRAINT UNDER CONSIDERATION                  001146
C                                                                        001147
      ANES=-CUP(NEW)/DLN2                                                001148
  310 AMIN=XP73                                                          001149
      DO 320 I=LE1,LELI                                                  001150
      IF (KSTAT(I).NE.0) GO TO 320                                       001151
      T=DLDC(I)                                                          001152
      IF (T.GE.XZERO) GO TO 320                                          001153
      T=-CUP(I)/T                                                        001154
      IF (T.GT.AMIN) GO TO 320                                           001155
      AMIN=T                                                             001156
      K=I                                                                001157
  320 CONTINUE                                                           001158
C                                                                        001159
C       WILL THE OBJECTIVE CONSTRAINT GET ACTIVE ?                       001160
C       IF NOT, MAKE ACTIVE THE CLOSEST                                  001161
C                                                                        001162
      OBJECT=ANES.LE.AMIN                                                001163
      ALFA=AMIN1(AMIN,ANES)                                              001164
      NACT1=NACT+1                                                       001165
      IF (OBJECT) GO TO 350                                              001166
      DO 330 I=1,N                                                       001167
      R(I,NACT1)=DC(K,I)                                                 001168
  330 CONTINUE                                                           001169
      CALL UTTRNS (U,N,NACT,ACCUM,R(1,NACT1),W)                          001170
```

```
      CALL ADDCL (U,R,N,NACT,1,RIGHT,W,ACCUM,.TRUE.,EPS)                001171
      IF (NACT1.EQ.NACT) GO TO 340                                      001172
      KSTAT(K)=-2                                                       001173
      GO TO 310                                                         001174
  340 KSTAT(K)=1                                                        001175
      KSET(NACT)=K                                                      001176
C                                                                       001177
C       TAKE THE STEP                                                   001178
C                                                                       001179
  350 IF (ALFA.EQ.XZERO) GO TO 380                                      001180
      DO 360 I=1,N                                                      001181
      X(I)=X(I)+ALFA*DL(I)                                              001182
  360 CONTINUE                                                          001183
      DO 370 I=LE1,LELI                                                 001184
      T=CUP(I)+ALFA*DLDC(I)                                             001185
      IF (KSTAT(I).EQ.-1.AND.T.GE.XZERO) KSTAT(I)=0                     001186
      CUP(I)=T                                                          001187
  370 CONTINUE                                                          001188
  380 IF (.NOT.OBJECT) GO TO 170                                        001189
C                                                                       001190
C       ACTIVATE THE OBJECTIVE CONSTRAINT                               001191
C                                                                       001192
      DO 390 I=1,N                                                      001193
      R(I,NACT1)=RIGHT(I)                                               001194
  390 CONTINUE                                                          001195
      CALL ADDCL (U,R,N,NACT,1,RIGHT,W,ACCUM,.FALSE.,EPS)               001196
      IF (NACT.EQ.NACT1) GO TO 400                                      001197
      KSTAT(NEW)=0                                                      001198
      GO TO 140                                                         001199
  400 KSET(NACT)=NEW                                                    001200
      GO TO 140                                                         001201
C                                                                       001202
C       NO FEASIBLE POINTS                                              001203
C                                                                       001204
  410 IFALL=3                                                           001205
      RETURN                                                            001206
      END                                                               001207
C                                                                       001208
C                                                                       001209
      SUBROUTINE MMLPA (F,DF,C,DC,M,N,N1,IC,LE,LI,XNMAX,XN,X,NACT,KSET,A 001210
     1SET,U,R,DL,RIGHT,FUP,DLDF,CUP,DLDC,KSTATF,KSTATC,W,SEPS,ACCUM,FMAX 001211
     2,IFALL)                                                           001212
C                                                                       001213
C     THE SUBROUTINE SOLVES A LINEARLY CONSTRAINED LINEAR MINIMAX       001214
C     PROBLEM. THE STARTING POINT MUST BE FEASIBLE.                     001215
C                                                                       001216
      DIMENSION F(M), DF(M,N), C(IC), DC(IC,N), X(N), ASET(N1), U(N,N), 001217
     1R(N,N), DL(N), RIGHT(N), FUP(M), DLDF(M), CUP(IC), DLDC(IC), W(N) 001218
      INTEGER KSET(N1),KSTATF(M),KSTATC(IC)                             001219
      LOGICAL ACCUM                                                     001220
      DATA XZERO,XONE,XP73/0.0,1.0,1.E73/                               001221
C                                                                       001222
C       INITIALIZE                                                      001223
C                                                                       001224
      LE1=LE+1                                                          001225
      LELI=LE+LI                                                        001226
      XNMAX2=XNMAX**2                                                   001227
      XN2=XZERO                                                         001228
      EPS=N*SEPS                                                        001229
      ACCUM=.FALSE.                                                     001230
      IFALL=0                                                           001231
      DO 10 I=1,N                                                       001232
      X(I)=XZERO                                                        001233
   10 CONTINUE                                                          001234
      FMAX=-XP73                                                        001235
```

```
      DO 20 I=1,M                                                       001236
      KSTATF(I)=0                                                       001237
      T=F(I)                                                            001238
      IF (T.LE.FMAX) GO TO 20                                           001239
      FMAX=T                                                            001240
      KSET0=I                                                           001241
   20 FUP(I)=T                                                          001242
      IF (LELI.EQ.0) GO TO 40                                           001243
      DO 30 I=1,LELI                                                    001244
      KSTATC(I)=0                                                       001245
      CUP(I)=C(I)                                                       001246
   30 CONTINUE                                                          001247
C                                                                       001248
C       ACTIVATE INITIAL ACTIVE SET                                     001249
C                                                                       001250
   40 NACTIN=NACT                                                       001251
      NACT=0                                                            001252
      IF (LE.EQ.0) GO TO 70                                             001253
      DO 60 I=1,LE                                                      001254
      DO 50 J=1,N                                                       001255
      R(J,I)=DC(I,J)                                                    001256
   50 CONTINUE                                                          001257
      KSET(I)=I                                                         001258
      KSTATC(I)=1                                                       001259
   60 CONTINUE                                                          001260
      CALL ADDCL (U,R,N,NACT,LE,RIGHT,W,ACCUM,.FALSE.,EPS)              001261
      IF (NACT.EQ.LE) GO TO 70                                          001262
      XN=XZERO                                                          001263
      IFALL=3                                                           001264
      RETURN                                                            001265
   70 IF (NACTIN.LT.LE1) GO TO 100                                      001266
      DO 90 K=1,NACTIN                                                  001267
      KK=KSET(K)                                                        001268
      IF (KK.LT.LE1.OR.KK.GT.LELI) GO TO 90                             001269
      NACT1=NACT+1                                                      001270
      IF (NACT1.GT.N) GO TO 100                                         001271
      DO 80 I=1,N                                                       001272
      R(I,NACT1)=DC(KK,I)                                               001273
   80 CONTINUE                                                          001274
      CALL UTTRNS (U,N,NACT,ACCUM,R(1,NACT1),W)                         001275
      CALL ADDCL (U,R,N,NACT,1,RIGHT,W,ACCUM,.FALSE.,EPS)               001276
      IF (NACT.LT.NACT1) GO TO 90                                       001277
      EPS=EPS+SEPS                                                      001278
      KSET(NACT1)=KK                                                    001279
      KSTATC(KK)=1                                                      001280
   90 CONTINUE                                                          001281
C                                                                       001282
C       TRANSFORM OBJECTIVE FUNCTION GRADIENT                           001283
C                                                                       001284
  100 KSTATF(KSET0)=1                                                   001285
      DO 110 J=1,N                                                      001286
      RIGHT(J)=-DF(KSET0,J)                                             001287
  110 CONTINUE                                                          001288
      KSET0=KSET0+LELI                                                  001289
      CALL UTTRNS (U,N,NACT,ACCUM,RIGHT,W)                              001290
C                                                                       001291
C       ITERATIVE LOOP                                                  001292
C                                                                       001293
C       CALCULATE MULTIPLIERS AND FIND THE LARGEST                      001294
C                                                                       001295
  120 ASET0=-XONE                                                       001296
      IF (NACT.EQ.0) GO TO 240                                          001297
      CALL RSOLV (R,N,NACT,RIGHT,ASET)                                  001298
      IF (NACT.EQ.LE) GO TO 240                                         001299
      AMAX=-XP73                                                        001300
```

```
      DO 130 I=LE1,NACT                                                 001301
      IF (KSET(I).GT.LELI) ASET0=ASET0-ASET(I)                          001302
      IF (ASET(I).LE.AMAX) GO TO 130                                    001303
      K=I                                                               001304
      AMAX=ASET(I)                                                      001305
  130 CONTINUE                                                          001306
      IF (AMAX.LT.XZERO.AND.ASET0.LT.XZERO) GO TO 240                   001307
      IF (AMAX.GT.ASET0) GO TO 180                                      001308
C                                                                       001309
C       CHANGE OBJECTIVE FUNCTION                                       001310
C                                                                       001311
      DO 140 I=LE1,NACT                                                 001312
      IF (KSET(I).LE.LELI) GO TO 140                                    001313
      K=I                                                               001314
      GO TO 150                                                         001315
  140 CONTINUE                                                          001316
  150 DO 170 I=1,K                                                      001317
      T=R(I,K)                                                          001318
      IF (K.EQ.NACT) GO TO 170                                          001319
      K1=K+1                                                            001320
      DO 160 J=K1,NACT                                                  001321
      IF (KSET(J).GT.LELI) R(I,J)=R(I,J)-T                              001322
  160 CONTINUE                                                          001323
  170 RIGHT(I)=RIGHT(I)+T                                               001324
      KK=KSET0                                                          001325
      KSET0=KSET(K)                                                     001326
      KSET(K)=KK                                                        001327
C                                                                       001328
C       DELETE ACTIVE CONSTRAINT NUMBER K                               001329
C                                                                       001330
  180 KK=KSET(K)                                                        001331
      IF (KK.GT.LELI) KSTATF(KK-LELI)=0                                 001332
      IF (KK.LE.LELI) KSTATC(KK)=0                                      001333
      IF (ACCUM) GO TO 190                                              001334
      ACCUM=.TRUE.                                                      001335
      CALL HACUM (U,N,NACT,W)                                           001336
  190 CALL DELCL (K,U,R,N,NACT,RIGHT,.TRUE.)                            001337
      EPS=EPS+SEPS                                                      001338
      IF (K.GT.NACT) GO TO 210                                          001339
      DO 200 I=K,NACT                                                   001340
      KSET(I)=KSET(I+1)                                                 001341
  200 CONTINUE                                                          001342
C                                                                       001343
C       DELETE LINEAR DEPENDENCE LABELS                                 001344
C                                                                       001345
  210 DO 220 I=1,M                                                      001346
      IF (KSTATF(I).EQ.-2) KSTATF(I)=0                                  001347
  220 CONTINUE                                                          001348
      IF (LI.EQ.0) GO TO 120                                            001349
      DO 230 I=LE1,LELI                                                 001350
      IF (KSTATC(I).EQ.-2) KSTATC(I)=0                                  001351
  230 CONTINUE                                                          001352
      GO TO 120                                                         001353
C                                                                       001354
C       IS THERE AN UNBOUNDED SOLUTION ?                                001355
C                                                                       001356
  240 IF (NACT.EQ.N) GO TO 490                                          001357
C                                                                       001358
C       CALCULATE THE PROJECTED GRADIENT                                001359
C                                                                       001360
      K=NACT+1                                                          001361
      T=XZERO                                                           001362
      DLN2=XZERO                                                        001363
      DO 250 I=K,N                                                      001364
      DLN2=DLN2+RIGHT(I)**2                                             001365
```

```
      DL(I)=RIGHT(I)                                                    001366
  250 CONTINUE                                                          001367
      IF (K.EQ.1) GO TO 270                                             001368
      DO 260 I=1,NACT                                                   001369
      T=T+RIGHT(I)**2                                                   001370
      DL(I)=XZERO                                                       001371
  260 CONTINUE                                                          001372
  270 T=T+DLN2                                                          001373
      IF (T.GT.XZERO.AND.DLN2.GT.EPS*EPS*T) GO TO 280                   001374
      IFALL=2                                                           001375
      GO TO 490                                                         001376
  280 CALL UTRNS (U,N,NACT,ACCUM,DL,W)                                  001377
C                                                                       001378
C       PROJECT GRADIENTS ON THE PROJECTED GRADIENT                     001379
C                                                                       001380
      DO 300 I=1,M                                                      001381
      T=XZERO                                                           001382
      DO 290 J=1,N                                                      001383
      T=T+DL(J)*DF(I,J)                                                 001384
  290 CONTINUE                                                          001385
      DLDF(I)=T                                                         001386
  300 CONTINUE                                                          001387
      IF (LELI.EQ.0) GO TO 330                                          001388
      DO 320 I=1,LELI                                                   001389
      T=XZERO                                                           001390
      DO 310 J=1,N                                                      001391
      T=T+DL(J)*DC(I,J)                                                 001392
  310 CONTINUE                                                          001393
      DLDC(I)=T                                                         001394
  320 CONTINUE                                                          001395
C                                                                       001396
C       CALCULATE STEP LENGTH                                           001397
C                                                                       001398
  330 SMINC=XP73                                                        001399
      IF (LI.EQ.0) GO TO 350                                            001400
      DO 340 I=LE1,LELI                                                 001401
      IF (KSTATC(I).NE.0) GO TO 340                                     001402
      T=DLDC(I)                                                         001403
      IF (T.GE.XZERO) GO TO 340                                         001404
      T=-CUP(I)/T                                                       001405
      IF (T.GT.SMINC) GO TO 340                                         001406
      NEWC=I                                                            001407
      SMINC=T                                                           001408
  340 CONTINUE                                                          001409
  350 SMINF=XP73                                                        001410
      K=KSET0-LELI                                                      001411
      T0=DLDF(K)                                                        001412
      F0=FUP(K)                                                         001413
      DO 360 I=1,M                                                      001414
      IF (KSTATF(I).NE.0) GO TO 360                                     001415
      T=T0-DLDF(I)                                                      001416
      IF (T.GE.XZERO) GO TO 360                                         001417
      T=(FUP(I)-F0)/T                                                   001418
      IF (T.GT.SMINF) GO TO 360                                         001419
      SMINF=T                                                           001420
      NEWF=I                                                            001421
  360 CONTINUE                                                          001422
      STEP=AMIN1(SMINF,SMINC)                                           001423
C                                                                       001424
C       IN CASE THE STEP IS TOO LONG REDUCE AND RETURN                  001425
C                                                                       001426
      S=STEP                                                            001427
      CALL LIMIT (XNMAX2,X,XN2,DL,DLN2,S,N)                             001428
      IF (S.EQ.STEP) GO TO 370                                          001429
      IFALL=1                                                           001430
```

```
      STEP=S                                                            001431
      GO TO 440                                                         001432
C                                                                       001433
C       INCLUDE THE NEW FUNCTION/CONSTRAINT                             001434
C                                                                       001435
  370 NACT1=NACT+1                                                      001436
      KK0=KSET0-LELI                                                    001437
      IF (SMINF.LT.SMINC) GO TO 400                                     001438
      DO 380 I=1,N                                                      001439
      R(I,NACT1)=DC(NEWC,I)                                             001440
  380 CONTINUE                                                          001441
      CALL UTTRNS (U,N,NACT,ACCUM,R(1,NACT1),W)                         001442
      CALL ADDCL (U,R,N,NACT,1,RIGHT,W,ACCUM,.TRUE.,EPS)                001443
      IF (NACT.EQ.NACT1) GO TO 390                                      001444
      KSTATC(NEWC)=-2                                                   001445
      GO TO 330                                                         001446
  390 KSTATC(NEWC)=1                                                    001447
      KSET(NACT)=NEWC                                                   001448
      GO TO 430                                                         001449
  400 DO 410 I=1,N                                                      001450
      R(I,NACT1)=DF(KK0,I)-DF(NEWF,I)                                   001451
  410 CONTINUE                                                          001452
      CALL UTTRNS (U,N,NACT,ACCUM,R(1,NACT1),W)                         001453
      CALL ADDCL (U,R,N,NACT,1,RIGHT,W,ACCUM,.TRUE.,EPS)                001454
      IF (NACT.EQ.NACT1) GO TO 420                                      001455
      KSTATF(NEWF)=-2                                                   001456
      GO TO 330                                                         001457
  420 KSTATF(NEWF)=1                                                    001458
      KSET(NACT)=NEWF+LELI                                              001459
  430 EPS=EPS+SEPS                                                      001460
      IF (STEP.EQ.XZERO) GO TO 120                                      001461
C                                                                       001462
C       TAKE THE STEP AND UPDATE LINEAR FUNCTIONS                       001463
C                                                                       001464
  440 FMAX=-XP73                                                        001465
      XN2=XZERO                                                         001466
      DO 450 I=1,N                                                      001467
      X(I)=X(I)+STEP*DL(I)                                              001468
      XN2=XN2+X(I)**2                                                   001469
  450 CONTINUE                                                          001470
      DO 460 I=1,M                                                      001471
      T=FUP(I)+STEP*DLDF(I)                                             001472
      IF (T.GT.FMAX) FMAX=T                                             001473
      FUP(I)=T                                                          001474
  460 CONTINUE                                                          001475
      IF (LELI.EQ.0) GO TO 480                                          001476
      DO 470 I=1,LELI                                                   001477
      CUP(I)=CUP(I)+STEP*DLDC(I)                                        001478
  470 CONTINUE                                                          001479
  480 IF (IFALL.EQ.0) GO TO 120                                         001480
C                                                                       001481
C       RETURN                                                          001482
C                                                                       001483
  490 XN=SQRT(XN2)                                                      001484
      NACT=NACT+1                                                       001485
      KSET(NACT)=KSET0                                                  001486
      ASET(NACT)=ASET0                                                  001487
      RETURN                                                            001488
      END                                                               001489
C                                                                       001490
C                                                                       001491
      SUBROUTINE LINSYS (A,B,IDIM,N,NR,EPS)                             001492
C                                                                       001493
C     THE SUBROUTINE SOLVES A SYSTEM OF LINEAR EQUATIONS                001494
C     USING GAUSSIAN ELIMINATION.                                       001495
```

```
C                                                                       001496
      DIMENSION A(IDIM,IDIM), B(N)                                      001497
      DATA XONE,XM50/1.0,1.E-50/                                        001498
      NR=0                                                              001499
C                                                                       001500
C       A IS CONSIDERED TO BE OF RANK K-1 IF THE ABSOLUTE VALUE         001501
C       OF THE K-TH PIVOT IS LESS THAN K*EPS.                           001502
C                                                                       001503
      IF (N-1) 120,10,20                                                001504
   10 IF (ABS(A(1,1)).LT.XM50) RETURN                                   001505
      NR=1                                                              001506
      B(1)=B(1)/A(1,1)                                                  001507
      RETURN                                                            001508
C                                                                       001509
C       EQUILIBRATION IN THE INFINITY NORM                              001510
C                                                                       001511
   20 DO 40 I=1,N                                                       001512
      AM=ABS(A(I,1))                                                    001513
      DO 30 J=2,N                                                       001514
      S=ABS(A(I,J))                                                     001515
      IF (AM.LT.S) AM=S                                                 001516
   30 CONTINUE                                                          001517
      IF (AM.LT.XM50) AM=XONE                                           001518
      B(I)=B(I)/AM                                                      001519
      DO 40 J=1,N                                                       001520
   40 A(I,J)=A(I,J)/AM                                                  001521
C                                                                       001522
C       ELIMINATION                                                     001523
C                                                                       001524
      N1=N-1                                                            001525
      DO 90 K=1,N1                                                      001526
      NR=K-1                                                            001527
C                                                                       001528
C       FIND PIVOTAL ROW                                                001529
C                                                                       001530
      AM=ABS(A(K,K))                                                    001531
      I0=K                                                              001532
      K1=K+1                                                            001533
      DO 50 I=K1,N                                                      001534
      S=ABS(A(I,K))                                                     001535
      IF (S.LE.AM) GO TO 50                                             001536
      AM=S                                                              001537
      I0=I                                                              001538
   50 CONTINUE                                                          001539
      IF (AM.LT.2*K*EPS) RETURN                                         001540
      IF (I0.EQ.K) GO TO 70                                             001541
C                                                                       001542
C       INTERCHANGE EQUATIONS K AND I0                                  001543
C                                                                       001544
      DO 60 J=K,N                                                       001545
      S=A(K,J)                                                          001546
      A(K,J)=A(I0,J)                                                    001547
      A(I0,J)=S                                                         001548
   60 CONTINUE                                                          001549
      S=B(K)                                                            001550
      B(K)=B(I0)                                                        001551
      B(I0)=S                                                           001552
C                                                                       001553
C       STORE PIVOT IN AM AND ELIMINATE IN ROWS K+1 TO N                001554
C                                                                       001555
   70 AM=A(K,K)                                                         001556
      DO 90 I=K1,N                                                      001557
      S=A(I,K)/AM                                                       001558
      DO 80 J=K1,N                                                      001559
      A(I,J)=A(I,J)-S*A(K,J)                                            001560
```

```
   80 CONTINUE                                                          001561
   90 B(I)=B(I)-S*B(K)                                                  001562
      NR=N1                                                             001563
      IF (ABS(A(N,N)).LT.2*N*EPS) RETURN                                001564
C                                                                       001565
C       A HAS FULL RANK                                                 001566
C                                                                       001567
      NR=N                                                              001568
C                                                                       001569
C       BACK SUBSTITUTION                                               001570
C                                                                       001571
      B(N)=B(N)/A(N,N)                                                  001572
      K=N                                                               001573
      DO 110 I=2,N                                                      001574
      K1=K                                                              001575
      K=K-1                                                             001576
      S=B(K)                                                            001577
      DO 100 J=K1,N                                                     001578
      S=S-A(K,J)*B(J)                                                   001579
  100 CONTINUE                                                          001580
      B(K)=S/A(K,K)                                                     001581
  110 CONTINUE                                                          001582
  120 RETURN                                                            001583
      END                                                               001584
C                                                                       001585
C                                                                       001586
      SUBROUTINE BFGS (B,N,Y,XX,W,SEPS)                                 001587
C                                                                       001588
C     UPDATES A HESSIAN APPROXIMATION USING BFGS-FORMULA.               001589
C                                                                       001590
      DIMENSION B(N,N), Y(N), XX(N), W(N)                               001591
      DATA XZERO/0.0/                                                   001592
      EPS=(N+10)*SEPS                                                   001593
      DO 20 I=1,N                                                       001594
      T=XZERO                                                           001595
      DO 10 J=1,N                                                       001596
      T=T+B(I,J)*XX(J)                                                  001597
   10 CONTINUE                                                          001598
      W(I)=T                                                            001599
   20 CONTINUE                                                          001600
      YXX=XZERO                                                         001601
      WXX=XZERO                                                         001602
      YN=XZERO                                                          001603
      XXN=XZERO                                                         001604
      WN=XZERO                                                          001605
      DO 30 I=1,N                                                       001606
      YN=YN+Y(I)**2                                                     001607
      XXN=XXN+XX(I)**2                                                  001608
      WN=WN+W(I)**2                                                     001609
      YXX=YXX+Y(I)*XX(I)                                                001610
      WXX=WXX+W(I)*XX(I)                                                001611
   30 CONTINUE                                                          001612
      YN=SQRT(YN)                                                       001613
      XXN=SQRT(XXN)                                                     001614
      WN=SQRT(WN)                                                       001615
      IF (YN.EQ.XZERO.OR.WN.EQ.XZERO.OR.XXN.EQ.XZERO) RETURN            001616
      IF (ABS(YXX).LT.EPS*YN*XXN) RETURN                                001617
      IF (ABS(WXX).LT.EPS*WN*XXN) RETURN                                001618
      DO 40 I=1,N                                                       001619
      B(I,I)=B(I,I)+Y(I)**2/YXX-W(I)**2/WXX                             001620
   40 CONTINUE                                                          001621
      IF (N.EQ.1) RETURN                                                001622
      DO 50 I=2,N                                                       001623
      I1=I-1                                                            001624
      DO 50 J=1,I1                                                      001625
```

```
      B(I,J)=B(I,J)+Y(I)*Y(J)/YXX-W(I)*W(J)/WXX                          001626
   50 B(J,I)=B(I,J)                                                     001627
      RETURN                                                            001628
      END                                                               001629
C                                                                       001630
C                                                                       001631
      SUBROUTINE ADDCL (U,R,N,KCOL,KNEW,RIGHT,W,ACCUM,LRIGHT,EPS)       001632
C                                                                       001633
C     UPDATES HOUSEHOLDER FACTORIZATION.                                001634
C     THE NEW COLUMNS MUST HAVE BEEN TRANSFORMED AS RIGHTHAND SIDES.    001635
C                                                                       001636
      DIMENSION U(N,N), R(N,N), RIGHT(N), W(N)                          001637
      LOGICAL ACCUM,LRIGHT                                              001638
      DATA XZERO/0.0/                                                   001639
      K1=KCOL+1                                                         001640
      K2=KCOL+KNEW                                                      001641
C                                                                       001642
C       COLUMN LOOP STARTS HERE                                         001643
C                                                                       001644
      DO 170 K=K1,K2                                                    001645
      S=XZERO                                                           001646
      T=XZERO                                                           001647
      IF (K.EQ.1) GO TO 20                                              001648
      KK=K-1                                                            001649
      DO 10 I=1,KK                                                      001650
      T=T+R(I,K)**2                                                     001651
   10 CONTINUE                                                          001652
   20 DO 30 I=K,N                                                       001653
      S=S+R(I,K)**2                                                     001654
   30 CONTINUE                                                          001655
      T=T+S                                                             001656
      T=SQRT(T)                                                         001657
      S=SQRT(S)                                                         001658
C                                                                       001659
C       RETURN IF THE NEW COLUMN DEPENDS LINEARLY ON THE                001660
C       PRECEDING COLUMNS                                               001661
C                                                                       001662
      IF (T.EQ.XZERO) RETURN                                            001663
      IF (S.LT.T*EPS) RETURN                                            001664
C                                                                       001665
C       PERFORM HOUSEHOLDER TRANSFORMATION                              001666
C                                                                       001667
      TT=R(K,K)                                                         001668
      T=ABS(TT)                                                         001669
      ALFA=SQRT(S*(S+T))                                                001670
      BETA=-SIGN(S,TT)                                                  001671
      R(K,K)=BETA                                                       001672
      W(K)=(TT-BETA)/ALFA                                               001673
      IF (K.EQ.N) GO TO 80                                              001674
      KK=K+1                                                            001675
      DO 40 I=KK,N                                                      001676
      W(I)=R(I,K)/ALFA                                                  001677
   40 CONTINUE                                                          001678
C                                                                       001679
C       TRANSFORM THE REMAINING COLUMNS                                 001680
C                                                                       001681
      IF (K.EQ.K2) GO TO 80                                             001682
      DO 70 J=KK,K2                                                     001683
      T=XZERO                                                           001684
      DO 50 I=K,N                                                       001685
      T=T+W(I)*R(I,J)                                                   001686
   50 CONTINUE                                                          001687
      DO 60 I=K,N                                                       001688
      R(I,J)=R(I,J)-T*W(I)                                              001689
   60 CONTINUE                                                          001690
```

```
   70 CONTINUE                                                          001691
C                                                                       001692
C       TRANSFORM THE RIGHTHAND SIDE                                    001693
C                                                                       001694
   80 IF (.NOT.LRIGHT) GO TO 110                                        001695
      T=XZERO                                                           001696
      DO 90 I=K,N                                                       001697
      T=T+W(I)*RIGHT(I)                                                 001698
   90 CONTINUE                                                          001699
      DO 100 I=K,N                                                      001700
      RIGHT(I)=RIGHT(I)-T*W(I)                                          001701
  100 CONTINUE                                                          001702
C                                                                       001703
C       ACCUMULATE THE TRANSFORMATIONS IN U                             001704
C       U MUST HAVE BEEN INITIALIZED                                    001705
C                                                                       001706
  110 IF (ACCUM) GO TO 130                                              001707
      DO 120 I=K,N                                                      001708
      U(I,K)=W(I)                                                       001709
  120 CONTINUE                                                          001710
      GO TO 170                                                         001711
  130 DO 160 I=1,N                                                      001712
      T=XZERO                                                           001713
      DO 140 J=K,N                                                      001714
      T=T+U(I,J)*W(J)                                                   001715
  140 CONTINUE                                                          001716
      DO 150 J=K,N                                                      001717
      U(I,J)=U(I,J)-T*W(J)                                              001718
  150 CONTINUE                                                          001719
  160 CONTINUE                                                          001720
  170 KCOL=KCOL+1                                                       001721
      RETURN                                                            001722
      END                                                               001723
C                                                                       001724
C                                                                       001725
      SUBROUTINE DELCL (K,U,R,N,KCOL,RIGHT,LRIGHT)                      001726
C                                                                       001727
C     DELETES COLUMN NUMBER K IN THE FACTORIZED MATRIX.                 001728
C     K MUST SATISFY      1.LE.K.LE.KCOL                                001729
C     U MUST HAVE BEEN ACCUMULATED.                                     001730
C                                                                       001731
      DIMENSION U(N,N), R(N,N), RIGHT(N)                                001732
      LOGICAL LRIGHT                                                    001733
C                                                                       001734
C       DELETE COLUMN NUMBER K                                          001735
C                                                                       001736
      KCOL=KCOL-1                                                       001737
      IF (K.GT.KCOL) RETURN                                             001738
      DO 10 J=K,KCOL                                                    001739
      J1=J+1                                                            001740
      DO 10 I=1,J1                                                      001741
   10 R(I,J)=R(I,J1)                                                    001742
C                                                                       001743
C       TRANSFORM TO UPPER TRIANGULAR FORM                              001744
C       USING STANDARD GIVENS TRANSFORMATIONS                           001745
C                                                                       001746
      DO 60 KK=K,KCOL                                                   001747
      K1=KK+1                                                           001748
      X=R(KK,KK)                                                        001749
      Y=R(K1,KK)                                                        001750
      A=SQRT(X*X+Y*Y)                                                   001751
      C=X/A                                                             001752
      S=Y/A                                                             001753
      R(KK,KK)=C*X+S*Y                                                  001754
      IF (KK.EQ.KCOL) GO TO 30                                          001755
```

```
      DO 20 J=K1,KCOL                                                   001756
      X=R(KK,J)                                                         001757
      Y=R(K1,J)                                                         001758
      R(KK,J)=C*X+S*Y                                                   001759
      R(K1,J)=C*Y-S*X                                                   001760
   20 CONTINUE                                                          001761
   30 IF (.NOT.LRIGHT) GO TO 40                                         001762
      X=RIGHT(KK)                                                       001763
      Y=RIGHT(K1)                                                       001764
      RIGHT(KK)=C*X+S*Y                                                 001765
      RIGHT(K1)=C*Y-S*X                                                 001766
C                                                                       001767
C       ACCUMULATE THE TRANSFORMATIONS                                  001768
C                                                                       001769
   40 DO 50 I=1,N                                                       001770
      X=U(I,KK)                                                         001771
      Y=U(I,K1)                                                         001772
      U(I,KK)=C*X+S*Y                                                   001773
      U(I,K1)=C*Y-S*X                                                   001774
   50 CONTINUE                                                          001775
   60 CONTINUE                                                          001776
      RETURN                                                            001777
      END                                                               001778
C                                                                       001779
C                                                                       001780
      SUBROUTINE UTTRNS (U,N,KCOL,ACCUM,R,W)                            001781
C                                                                       001782
C     TRANSFORM THE VECTOR R AS A RIGHTHAND SIDE.                       001783
C                                                                       001784
      DIMENSION U(N,N), R(N), W(N)                                      001785
      LOGICAL ACCUM                                                     001786
      DATA XZERO/0.0/                                                   001787
C                                                                       001788
C       IF THE TRANSFORMATIONS HAVE BEEN ACCUMULATED                    001789
C       DO SIMPLE MATRIX-MULTIPLICATION                                 001790
C       ELSE TRANSFORM RIGHTHAND SIDES                                  001791
C                                                                       001792
      IF (ACCUM) GO TO 40                                               001793
      IF (KCOL.EQ.0) RETURN                                             001794
      DO 30 K=1,KCOL                                                    001795
      T=XZERO                                                           001796
      DO 10 I=K,N                                                       001797
      T=T+R(I)*U(I,K)                                                   001798
   10 CONTINUE                                                          001799
      DO 20 I=K,N                                                       001800
      R(I)=R(I)-T*U(I,K)                                                001801
   20 CONTINUE                                                          001802
   30 CONTINUE                                                          001803
      RETURN                                                            001804
   40 DO 50 I=1,N                                                       001805
      W(I)=R(I)                                                         001806
   50 CONTINUE                                                          001807
      DO 70 K=1,N                                                       001808
      T=XZERO                                                           001809
      DO 60 I=1,N                                                       001810
      T=T+U(I,K)*W(I)                                                   001811
   60 CONTINUE                                                          001812
      R(K)=T                                                            001813
   70 CONTINUE                                                          001814
      RETURN                                                            001815
      END                                                               001816
C                                                                       001817
C                                                                       001818
      SUBROUTINE UTRNS (U,N,KCOL,ACCUM,R,W)                             001819
C                                                                       001820
```

```
C     TRANSFORM THE VECTOR R OPPOSITE A RIGHTHAND SIDE.                 001821
C                                                                       001822
      DIMENSION U(N,N), R(N), W(N)                                      001823
      LOGICAL ACCUM                                                     001824
      DATA XZERO/0.0/                                                   001825
      K1=KCOL+1                                                         001826
      IF (ACCUMD GO TO 40                                               001827
      IF (KCOL.EQ.0) RETURN                                             001828
      DO 30 KK=1,KCOL                                                   001829
      K=K1-KK                                                           001830
      T=XZERO                                                           001831
      DO 10 J=K,N                                                       001832
      T=T+U(J,K)*R(J)                                                   001833
   10 CONTINUE                                                          001834
      DO 20 J=K,N                                                       001835
      R(J)=R(J)-T*U(J,K)                                                001836
   20 CONTINUE                                                          001837
   30 CONTINUE                                                          001838
      RETURN                                                            001839
   40 DO 50 I=1,N                                                       001840
      W(I)=R(I)                                                         001841
   50 CONTINUE                                                          001842
      DO 70 I=1,N                                                       001843
      T=XZERO                                                           001844
      DO 60 J=1,N                                                       001845
      T=T+U(I,J)*W(J)                                                   001846
   60 CONTINUE                                                          001847
      R(I)=T                                                            001848
   70 CONTINUE                                                          001849
      RETURN                                                            001850
      END                                                               001851
C                                                                       001852
C                                                                       001853
      SUBROUTINE RSOLV (R,N,KCOL,RIGHT,X)                               001854
C                                                                       001855
C     PERFORM BACK SUBSTITUTION ON RIGHT.                               001856
C                                                                       001857
      DIMENSION R(N,N), RIGHT(N), X(N)                                  001858
C                                                                       001859
C       CALCULATE ALFA USING BACK SUBSTITUTION ON R                     001860
C                                                                       001861
      K=KCOL                                                            001862
      K1=K+1                                                            001863
   10 IF (K.EQ.0) RETURN                                                001864
      T=RIGHT(K)                                                        001865
      IF (K1.GT.KCOL) GO TO 30                                          001866
      DO 20 J=K1,KCOL                                                   001867
      T=T-X(J)*R(K,J)                                                   001868
   20 CONTINUE                                                          001869
   30 X(K)=T/R(K,K)                                                     001870
      K1=K                                                              001871
      K=K-1                                                             001872
      GO TO 10                                                          001873
      END                                                               001874
C                                                                       001875
C                                                                       001876
      SUBROUTINE TSOLV (R,N,KCOL,RIGHT,X)                               001877
C                                                                       001878
C     PERFORM BACK SUBSTITUTION ON RIGHT USING THE                      001879
C     TRANSPOSED TRIANGULAR MATRIX.                                     001880
C                                                                       001881
      DIMENSION R(N,N), RIGHT(N), X(N)                                  001882
      IF (KCOL.EQ.0) RETURN                                             001883
      X(1)=RIGHT(1)/R(1,1)                                              001884
      IF (KCOL.EQ.1) RETURN                                             001885
```

```
      DO 20 I=2,KCOL                                                    001886
      I1=I-1                                                            001887
      T=RIGHT(I)                                                        001888
      DO 10 J=1,I1                                                      001889
      T=T-X(J)*R(J,I)                                                   001890
   10 CONTINUE                                                          001891
      X(I)=T/R(I,I)                                                     001892
   20 CONTINUE                                                          001893
      RETURN                                                            001894
      END                                                               001895
C                                                                       001896
C                                                                       001897
      SUBROUTINE HACUM (U,N,KCOL,W)                                     001898
C                                                                       001899
C     ACCUMULATES HOUSEHOLDER VECTORS STORED IN LOWER TRIANGLE          001900
C     OF THE FIRST KCOL COLUMNS OF U IN AN ORTHONORMAL MATRIX U.        001901
C     THE HOUSEHOLDER VECTORS MUST HAVE TWO NORM EQUAL TO TWO.          001902
C     KCOL.GE.1 .                                                       001903
C                                                                       001904
      DIMENSION U(N,N), W(N)                                            001905
      DATA XZERO,XONE/0.0,1.0/                                          001906
C                                                                       001907
C       INITIALIZE USING LAST TRANSFORMATION                            001908
C                                                                       001909
      K1=KCOL+1                                                         001910
      DO 10 I=KCOL,N                                                    001911
      W(I)=U(I,KCOL)                                                    001912
   10 CONTINUE                                                          001913
      DO 20 I=KCOL,N                                                    001914
      U(I,I)=XONE-W(I)**2                                               001915
   20 CONTINUE                                                          001916
      IF (KCOL.EQ.N) GO TO 40                                           001917
      DO 30 I=K1,N                                                      001918
      I1=I-1                                                            001919
      T=W(I)                                                            001920
      DO 30 J=KCOL,I1                                                   001921
      S=-T*W(J)                                                         001922
      U(I,J)=S                                                          001923
   30 U(J,I)=S                                                          001924
   40 IF (KCOL.EQ.1) RETURN                                             001925
C                                                                       001926
C       ACCUMULATE REMAINING TRANSFORMATIONS                            001927
C                                                                       001928
      DO 100 KK=2,KCOL                                                  001929
      K=K1-KK                                                           001930
      DO 50 I=K,N                                                       001931
      W(I)=U(I,K)                                                       001932
   50 CONTINUE                                                          001933
      T=W(K)                                                            001934
      KP1=K+1                                                           001935
      U(K,K)=XONE-T*T                                                   001936
      DO 60 I=KP1,N                                                     001937
      U(I,K)=-T*W(I)                                                    001938
   60 CONTINUE                                                          001939
      DO 90 L=KP1,N                                                     001940
      S=XZERO                                                           001941
      DO 70 I=KP1,N                                                     001942
      S=S+W(I)*U(I,L)                                                   001943
   70 CONTINUE                                                          001944
      U(K,L)=-T*S                                                       001945
      DO 80 I=KP1,N                                                     001946
      U(I,L)=U(I,L)-S*W(I)                                              001947
   80 CONTINUE                                                          001948
   90 CONTINUE                                                          001949
  100 CONTINUE                                                          001950
```

```
      RETURN                                                            001951
      END                                                               001952
C                                                                       001953
C                                                                       001954
      SUBROUTINE LIMIT (XNMAX2,X,XN2,P,PN2,ALFA,N)                      001955
C                                                                       001956
C     LIMIT THE STEP LENGTH ALFA.                                       001957
C                                                                       001958
      DIMENSION X(N), P(N)                                              001959
      DATA XZERO/0.0/                                                   001960
      XTP=XZERO                                                         001961
      DO 10 I=1,N                                                       001962
      XTP=XTP+X(I)*P(I)                                                 001963
   10 CONTINUE                                                          001964
      B=XTP/PN2                                                         001965
      T=SQRT(B*B+(XNMAX2-XN2)/PN2)                                      001966
      AP=T-B                                                            001967
      AM=-T-B                                                           001968
      IF (ALFA.GT.AP) ALFA=AP                                           001969
      IF (ALFA.LT.AM) ALFA=AM                                           001970
      RETURN                                                            001971
      END                                                               001972
```