UR1981-21

1981/21. FORTRAN programs for the vertical continuation of gravity data

R.G. Richardson

Abstract

The equivalent source technique provides a means of vertically continuing gravity observations that are irregularly distributed in space without the normal requirements of gridding, with accompanying filtering and distortion, and reduction of observations to a uniform plane.

A suite of programs is presented which represents an advance in the efficient implementation of the original algorithm.

INTRODUCTION

Gravity observations on an irregular grid at a variety of elevations may be synthesised by a source of discrete point masses (the equivalent source) on a plane at an arbitrary depth below the lowest observation height. Once the equivalent source is determined, the field can be re-calculated on a regular grid at any height above the plane of sources, allowing rapid determination of regional and residual anomalies.

The equivalent source technique was originally formulated by Dampney (1966; 1969) and applied by him to the Derby-Winnaleah area of north-east Tasmania, but it has since been enhanced and optimised for use with Department of Mines surveys (Leaman and Richardson, 1981). The following discussion of the technique is adapted from Dampney (1969).

The gravitational field g_z (x,y,z) observed at N observation points can be synthesised using N point masses at a suitable depth, and the equation

$$g_1 = a_{11}m_1 + a_{12}m_2 + \dots + a_{1k}m_k + \dots + a_{1N}m_N$$

$$g_i = a_{i1}m_1 + a_{i2}m_2 + \dots + a_{ik}m_k + \dots + a_{iN}m_N$$

$$g_N = a_{N1}m_1 + a_{N2}m_2 + \dots + a_{Nk}m_k + \dots + a_{NN}m_N$$

where

$$a_{ik} = \frac{K(z_i-h)}{[(x_i-\alpha_k)^2 + (y_i-\beta_k)^2 + (z_i-h)^2]^{3/2}}$$

Z = h is the horizontal plane containing the point masses m_k at (α_k, β_k, h)

g_i is the observed gravity field at (x_i, y_i, z_i).

This can be written in matrix form as

$$g = Am$$

which represents N simultaneous equations in N unknowns and is thus solvable.

The equation is solved by reducing R to a value of $N\sigma^2$ by varying the m_k in

$$R = (g-Am)^T(g-Am)$$

where
- T denotes the transpose operation
- N is the number of data points
- σ is the mean variance of the observed data.

The A values may be stored on disk or tape, stored in memory or re-computed each time as required. As implemented, the method avoids numerical round-off problems. The computation time is proportional to N^2 and thus for large N the data may need to be subdivided and the method applied to calculate the equivalent masses for each sub-area. These masses are then used as the first approximation to the point masses required to synthesise the complete data set.

PROGRAM DESCRIPTIONS

The normal program sequence is:

(i) INITIAL/MASS

Calculate a first approximation to the point masses and generate the file of a_{ik} values. If the data set is sufficiently small the a_{ik} values may best be stored in memory or if the data set is very large the a_{ik} values may have to be re-calculated at each usage.

(ii) GRAVTWO/A

Takes the observed gravity, initial approximations to the point masses, and the a_{ik} values and adjusts the point masses until $R < N\sigma^2$.

(iii) GRAVTWO/B

Calculates the gravity field at a series of points on a rectangular grid at any height above the plane of the equivalent source.

(a) INITIAL/MASS

This program (Appendix 1) sets up duplicate files (G2FA, G2FB) of the point masses, a control file (G2FC) used by program GRAVTWO/A to indicate the most up-to-date values for the point masses, and the a_{ik} values. The array dimensions should be adjusted to suit the number of data points.

Data input from File 5 is:

Line 3800 N number of data points

Line 4100 H the height of the source plane (negative below the datum).

Normally $2.5\Delta X < (Zi-H) < 6\Delta X$

where ΔX is the average observation separation.

The optimum value of H may be determined by plotting the rate of convergence of equal length runs of GRAVTWO/A for different values of H and choosing the H value corresponding to fastest convergence (*e.g.* Leaman and Richardson, 1981, fig. 14).

EPS ($=N\sigma^2$) the error value at which the approximation is sufficiently good.

CON sets the minimum acceptable rate of convergence at each iteration. If convergence is too slow and the number of iterations is greater than LOOPS then A = C*A.

Try CON = 0.1.

LOOPS is used to decide when to increase A.

Try LOOPS = 2.

C the factor by which A will be increased if convergence is not sufficiently fast.

Try C = 1.5.

LIMA the upper value allowed for A.

Try LIMA = 2.8.

The above variables are passed to GRAVTWO/A and GRAVTWO/B via files G2FA and G2FB.

Data input from File 2 is:

Line 6800 X_i, Y_i, Z_i, G_i the N sets of observation coordinates and gravity values are read and the height is converted to the same units as X and Y at line 7100.

The program calculates the average height of the observations and the maximum and minimum X and Y coordinates. Using this information, the average number of point masses, within a distance of three times the average height of observations above the source plane from each observation point, is calculated. The initial approximations to the point masses are then calculated using:

$$m_i = \frac{G_i * (H-Z_i)^2}{N_{avge}}$$

N a_{ik} values are calculated for each value of I using

$$A_{ik}(K) = \frac{Z_i - H}{((x_i-x_k)^2 + (y_i-y_k)^2 + (Z_i-H)^2)^{3/2}}$$

and then written to File 1.

(b) GRAVTWO/A

This program (Appendix 2) uses the files G2FA and G2FB which were created by INITIAL/MASS. The terminology used within the program is that of Dampney (1969) and solves the equation

$$g = Am$$

using an over-relaxed method of steepest descent.

Data input from File 8 is:

Line 4000 G_i the N observed gravity values.

The a_{ik} values are copied from tape to disk for higher access speeds (Line 4700 to Line 5500) and the most recently written file of mass values is read from disk (Line 2700 to Line 3800). Computation terminates when the sum of the squares of the difference between the observed gravity value and calculated gravity value is less than EPS or when the processor time exceeds some specified value. Every 200 processor seconds (approximately) the current point mass values are written to disk and the control file (G2FC) updated accordingly.

If the rate of convergence is too slow as defined by CON and LOOPS, the value of A is increased by a factor C (Line 9700) until LIMA is reached. If the solution becomes non-convergent, the A value is halved and the point masses recalculated to ensure convergence.

The arrays (Line 2100) should be dimensioned to suit the data. If the data set is sufficiently small, the a_{ik} values may be stored in memory, or if the data set is very large the a_{ik} values may need to be calculated each time they are referenced. The latter would require storage of the coordinates of the observation points.

(c) GRAVTWO/B

This program (Appendix 3) uses the mass values calculated by GRAVTWO/A and the X and Y coordinates of the observation points to calculate the anomaly on a rectangular grid at an arbitrary height above the equivalent source plane.

Data input from File 7 is:

Line 1800	Z1	the height of the desired continuation plane (negative below the datum)
	GSPACE	the X and Y spacing of the grid in the same units as the observation point coordinates (separate X and Y values could be specified with slight modifications to the program).

Data input from File 5 is:

Line 4300 X_i, Y_i the N sets of observation coordinates.

The grid limits are set in Lines 2800 to 3200.

Data output to File 8 is:

Line 4600 GSPACE) as for input
Z1)

Line 6900 I a row counter for the gravity values.

Line 7100 G_{grid} a row of calculated data from Xmin to Xmax.

The program calculates the gravity anomaly using the formula

$$G_{grid} = \sum_{k=1}^{N} \frac{m_k(Z1-H)}{((X_k-X_{grid})^2 + (Y_k-Y_{grid})^2 + (Z1-H)^2)^{3/2}}$$

CONCLUSIONS

The equivalent source technique provides a reliable method for removing many of the problems associated with gravity surveys in areas of irregular topography. Computation time is long, with the number of iterations depending on the distribution of observation points and on the anomaly wavelengths present. For a set of 1013 gravity observations from eastern Tasmania, the processor times on a Burroughs B6700 were

Program	*Time* (seconds)
INITIAL/MASS	330
GRAVTWO/A	42 000
GRAVTWO/B	250

Although these times are long the costs, when compared to the cost of drilling, are small.

REFERENCES

DAMPNEY, C.N.G. 1966. *Geophysical studies in Tasmania.* M.Sc. thesis, University of Tasmania:Hobart.

DAMPNEY, C.N.G. 1969. The equivalent source technique. *Geophysics* 34:39-53.

LEAMAN, D.E.; RICHARDSON, R.G. 1981. Gravity survey of the East Coast Coalfields. *Bull.geol.Surv.Tasm.* 60.

[15 April 1981]

APPENDIX 1

Programme, INITIAL/MASS

NITIAL/MASS (04/09/79)

```
$SET $
$RESET FREE
$OPT=+1
$SET VECTORMODE
C PROGRAM INITIAL/MASS
C BASED ON C.N.G.DAMPNEY 1966 AND 1969
C CALCULATES THE INITIAL MASSES FOR GRAVTWO/A AND SETS UP THE
C NECESSARY FILES FOR THE RUNS OF GRAVTWO/A
C ALSO SETS UP THE FILES OF AIK COEFFICIENTS
C FILE1=AIK COEFFICIENTS ON PETAPE
C FILE2=STATION DATA
C FILE3=MASS OUTPUT
C FILE7=MASS OUTPUT
C FILE4=GRAVTWO RECOVERY CONTROL FILE (G2FC)
C FILE5=CONTROL FILE
C FILE6=PRINTER
FILE   6(KIND=PRINTER)
FILE   5(TITLE="CONTROL/MASSES",KIND=PACK,FILETYPE=7)
FILE   4(TITLE="G2FC ", KIND=PACK,FILETYPE=0,UNITS=WORDS,
     . MAXRECSIZE=14,MINRECSIZE=14,BLOCKSIZE=14,
     . FILEKIND=DATA,AREAS=2,INTMODE=EBCDIC,EXTMODE=EBCDIC,
     . AREASIZE=1,PROTECTION=PROTECTED,SECURITYTYPE=CLASSA)
FILE   3(TITLE="G2FA ",KIND=PACK,FILETYPE=0,UNITS=WORDS,
     . MAXRECSIZE=14,MINRECSIZE=14,BLOCKSIZE=420,
     . FILEKIND=DATA,AREAS=200,INTMODE=EBCDIC,EXTMODE=EBCDIC,
     . AREASIZE=450,SECURITYTYPE=CLASSA)
FILE   7(TITLE="G2FB",KIND=PACK,FILETYPE=0,UNITS=WORDS,
     . MAXRECSIZE=14,MINRECSIZE=14,BLOCKSIZE=420,
     . FILEKIND=DATA,AREAS=200,INTMODE=EBCDIC,EXTMODE=EBCDIC,
     . AREASIZE=450,SECURITYTYPE=CLASSA)
FILE   2(TITLE="(GMGL0026)CCP/SORT",KIND=PACK,FILETYPE=7)
FILE   1(TITLE="GMGL0030DATA1",KIND=PETAPE,FILETYPE=7,SAVEFACTOR=999,
     1 SECURITYTYPE=CLASSA)
      DIMENSION M(1100),G(1100),X(1100),Y(1100),Z(1100),AIK(1100)
C DIMENSION M ETC. WITH DIMENSION M(N),ETC.
      REAL M,LIMA
      EQUIVALENCE (AIK(1),M(1)),(G(1),M(1))
      READ(5,100) N
  100 FORMAT(I4)
C THE  NUMBER OF DATA POINTS
      READ(5,200) H,EPS,CON,LOOPS,C,LIMA
  200 FORMAT(3F10.3,I3,2F10.3)
C H IS THE DEPTH OF THE PLANE OF POINT MASSES USED FOR
C     CONTINUATION (NEGATIVE IF BELOW THE ELEVATION DATUM)
C EPS IS THE ERROR VALUE AT WHICH THE APPROXIMATION IS
C    SUFFICIENTLY GOOD
C CON IS THE REQUIRED CONVERGENCE BEFORE INCREASING A
C LOOPS IS THE NUMBER OF ITERATIONS BEFORE CONVERGENCE
C CAN BE SPED UP,IF NECESSARY
C C IS THE CONVERGENCE FACTOR
C LIMA (REAL) IS THE MAXIMUM VALUE ALLOWED FOR A
      LOCK(5)
      I=0
      A=1.2
      WRITE(4,201) I
  201 FORMAT(I2)
      CLOSE(4,DISP=CRUNCH)
C USED TO INDICATE THE FILE CONTAINING THE MASSES
      WRITE(3,202) N,H,EPS,CON,LOOPS,C,LIMA,A
      WRITE(7,202) N,H,EPS,CON,LOOPS,C,LIMA,A
  202 FORMAT(I5,3E18.12/I5,3E18.12)
      XMAX=-1.E16
      XMIN=1.E16
      YMAX=XMAX
      YMIN=XMIN
      SUM=0.0
      DO 10 I=1,N
      READ(2,102) XI,YI,ZZ,G(I)
  102 FORMAT(10X,2F9.4,F8.2,28X,F8.2)
C CHOOSE FORMAT TO SUIT DATA
      ZZ=ZZ*0.001
C CONVERT TO KMS.
      Z(I)=ZZ
      SUM=SUM+ZZ
      XMAX=AMAX1(XMAX,XI)
      XMIN=AMIN1(XMIN,XI)
      YMAX=AMAX1(YMAX,YI)
```

```
      YMIN=AMIN1(YMIN,YI)
      X(I)=XI
      Y(I)=YI
   10 CONTINUE
      ZAVGE=SUM/FLOAT(N)
      AREA=(XMAX-XMIN)*(YMAX-YMIN)
C ASSUME RECTANGULAR DATA DISTRIBUTION FOR CALCULATION OF EQUIVALENT
C MASSES.
      FN=(6.0*3.141592*(ZAVGE-H)/AREA)*FLOAT(N)
C USE 3 RADII AS GOOD APPROXIMATION TO MASSES
      DO 15 I=1,N
      M(I)=G(I)*((H-Z(I))**2)/FN
   15 CONTINUE
      WRITE(3,203) (M(I),I=1,N)
      WRITE(7,203) (M(I),I=1,N)
  203 FORMAT(4E18.12)
      LOCK(2)
      CLOSE(3,DISP=CRUNCH)
      CLOSE(7,DISP=CRUNCH)
      WRITE(6,400)
  400 FORMAT(' INITIAL MASSES COMPLETED')
C NOW TO COMPUTE AIK
      OPEN(1,MAXRECSIZE=N,MINRECSIZE=N,BLOCKSIZE=N,MYUSE=VALUE(OUT),
     . UNITS=VALUE(WORDS),EXTMODE=VALUE(EBCDIC),INTMODE=VALUE(EBCDIC)
      DO 20 I=1,N
      XI=X(I)
      YI=Y(I)
      ZI=Z(I)-H
      ZI2=ZI**2
      DO 30 K=1,N
      XK=XI-X(K)
      YK=YI-Y(K)
      W=XK**2+YK**2+ZI2
      AIK(K)=ZI/(W*SQRT(W))
   30 CONTINUE
      CALL PUTOUT(AIK,N)
   20 CONTINUE
      CLOSE (1)
      WRITE(6,401)
  401 FORMAT(' AIK S COMPLETE')
      STOP
      END
      SUBROUTINE PUTOUT(A,N)
C N WORD UNFORMATTED WRITE
      DIMENSION A(N)
      WRITE(1) A
      RETURN
      END
```

APPENDIX 2

Programme, GRAVTWO/A

GRAVTWC/A (10/15/79)

```
$SET$
$RESET FREE
$OPT=+1
$SET VECTORMODE
C PPROGRAM GRAVTWO/A
C BASED ON THE PROGRAM GRAVTWO PARTA OF C.N.G. DAMPNEY
C CALCULATES THE EQUIVALENT SOURCES NEEDED TO APPROXIMATE THE
C GRAVITIONAL FIELD OBSERVED AT A NUMBER OF IRREGULARLY DISTRIBUTED
C DATA POINTS.
C THIS PROGRAM MUST BE RUN AFTER INITIAL/MASS WHICH SETS UP CONTROL
C AND SAVE FILES
C UPON SATISFACTORY APPROXIMATION THE PROGRAM TERMINATES
FILE   5(TITLE="G2FC",KIND=PACK,FILETYPE=7)
C USED TO INDICATE THE TITLE OF FILE4
FILE   4(KIND=PACK)
C FILE4 IS DUMP FILE - TITLE SET DYNAMICALLY
FILE   6(KIND=PRINTER)
FILE   3(TITLE="AIK",KIND=PACK,FILETYPE=0,AREASIZE=2,AREAS=999)
FILE   2(TITLE="GMGL00300ATA1",KIND=PETAPE,FILETYPE=7)
FILE   8(TITLE="(GMGL0026)CCP/SORT",KIND=PACK,FILETYPE=7)
       DIMENSION F(1100),DU(1100),G(1100),AIK(1100),
     . G2FA(2),G2FB(2)
       REAL EPS,CON,C,LIMA,A,U2,U1,SUM1,SUM2,SUM3,FI,LAMBDA,H,
     . M(1100)
       INTEGER IB
       DATA G2FA/"G2FA.    "/,G2FB/"G2FB.    "/
       READ(5,100) IC
   100 FORMAT(I2)
       LOCK(5)
       IF (IC .EQ. 0) OPEN(4,TITLE=G2FA,FILETYPE=7)
C DATA IN G2FA
       IF (IC .EQ. 1) OPEN(4,TITLE=G2FB,FILETYPE=7)
C DATA IN G2FB
       READ(4,101) N,H,EPS,CON,LOOPS,C,LIMA,A
   101 FORMAT(I5,3E18.12/I5,3E18.12)
       READ(4,102) (M(I),I=1,N)
   102 FORMAT(4E18.12)
       LOCK(4)
       DO 5 I=1,N
       READ(8,103) G(I)
   103 FORMAT(64X,F8.2)
C CHOOSE FORMAT FOR PROJECT
     5 CONTINUE
       LOCK(8)
       ITIME=12000
C RUN IN INCREMENTS OF 200 SECS
       OPEN(3,MAXRECSIZE=N,MINRECSIZE=N,BLOCKSIZE=N,UNITS=VALUE(WORDS
     . EXTMODE=VALUE(EBCDIC),MYUSE=VALUE(OUT),INTMODE=VALUE(EBCDIC))
C SET ATTRIBUTES OF FILE TO HOLD COEFFICIENTS
       DO 13 I=1,N
       CALL INBIN(AIK,N,2)
       CALL OUTBIN(AIK,N,3)
    13 CONTINUE
       CLOSE(2)
C COPY AIK TO TEMPORARY DISK FILE
       IB=0
       U2=1.E+66
     1 CONTINUE
C OLD S1 LABEL
       IB=IB+1
       U1=0.0
C
C CALCULATE F(I)
       FIND(3=1)
C FIND FIRST RECORD OF AIK FILE
       DO 11 I=1,N
       CALL INBIN(AIK,N,3)
       SUM1=0.0
       DO 10 K=1,N
       SUM1=SUM1+M(K)*AIK(K)
    10 CONTINUE
       FI=G(I)-SUM1
       U1=U1+FI**2
       F(I)=FI
    11 CONTINUE
       WRITE(6,200) U1
   200 FORMAT(E18.12)
```

```
      IF (U1 .LE. U2) GOTO 22
C CONVERGENT
      A=A*0.5
      DO 20 I=1,N
      M(I)=M(I)+A*LAMBDA*DU(I)
   20 CONTINUE
      IB=0
      WRITE(6,201)
  201 FORMAT(' NOT CONVERGENT')
      GOTO 1
C FINISHED WITH NON-CONVERGENT CASE
   22 CONTINUE
      IF (TIME(2) .GE. ITIME .OR. U1.LT. EPS) CALL SAVE(N,H,EPS,CON,
     .                        LOOPS,C,LIMA,A,M,ITIME)
C SAVE FILE TO ENABLE RESTART
      IF (U1 .LT. EPS .OR. ITIME .GT. 120000) GOTO 3
C ERROR REDUCED TO CRITERION OR TIME IS LONGER THAN 2000 CPU SECONDS
      WRITE(6,202)
  202 FORMAT(' CONVERGENT')
      IF (IB .GT. LOOPS .AND. ((U2-U1)/U2) .LT. CON) A=C*A
      U2=U1
      IF (A .LT. 1.0) A=1.0
      IF (A .GT. LIMA) A=LIMA
C FORCE A TO REMAIN IN RANGE
C
C CALCULATE DU/DJ(K)
      FIND(3=1)
      DO 15 I=1,N
      CALL INBIN(AIK,N,3)
      SUM1=0.0
      DO 14 K=1,N
      SUM1=SUM1-2.0*F(K)*AIK(K)
   14 CONTINUE
      DU(I)=SUM1
   15 CONTINUE
C
C NOW CALCULATE LAMBDA
      SUM2=0.0
      SUM3=0.0
      FIND(3=1)
      DO 31 I=1,N
      SUM1=0.0
      CALL INBIN(AIK,N,3)
      DO 30 K=1,N
      SUM1=SUM1-DU(K)*AIK(K)
   30 CONTINUE
      SUM2=SUM2+SUM1**2
      SUM3=SUM3+SUM1*F(I)
   31 CONTINUE
      LAMBDA=SUM3/SUM2
      DO 32 I=1,N
      M(I)=M(I)-A*LAMBDA*DU(I)
   32 CONTINUE
      GOTO 1
    3 CONTINUE
      IF (U1 .LT. EPS) WRITE(6,203)
  203 FORMAT(' CONVERGED ADEQUATELY')
      WRITE(6,204)
  204 FORMAT(' CIVILISED END')
      STOP
      END
      SUBROUTINE SAVE(N,H,EPS,CON,LOOPS,C,LIMA,A,M,ITIME)
C USED TO DUMP OUT MOST RECENT APPROXIMATIONS TO MASSES
      REAL LIMA,M
      DIMENSION M(2),G2FA(2),G2FB(2)
      DATA G2FA/"G2FA.     "/,G2FB/"G2FB.     "/
      OPEN(5,FILETYPE=7)
      READ(5,100) IC
  100 FORMAT(I2)
      LOCK(5)
      II=0
      IF (IC .EQ. 0) II=1
      IF (II .EQ. 0) OPEN(4,TITLE=G2FA,FILETYPE=7)
C OPEN G2FA FOR WRITE
      IF (II .EQ. 1) OPEN(4,TITLE=G2FB,FILETYPE=7)
C OPEN G2FB FOR WRITE
      WRITE(4,200) N,H,EPS,CON,LOOPS,C,LIMA,A
  200 FORMAT(I5,3E18.12/I5,3E18.12)
      WRITE(4,201) (M(I),I=1,N)
  201 FORMAT(4E18.12)
```

APPENDIX 3

Programme, GRAVTWO/B

RAVTWO/B (03/14/80)

```
      $SET$
      $RESET FREE
      $OPT=+1
      $SET VECTORMODE
      C PROGRAM GRAVTWO/B
      C BASED ON C.N.G. DAMPNEY 1966, FIGURE 15
      FILE  3(KIND=PACK,FILETYPE=7)
      FILE  4(TITLE="G2FC",KIND=PACK,FILETYPE=7)
      FILE  5(TITLE="(GMGL0026)CCP/SORT",KIND=PACK,FILETYPE=7)
      FILE  6(KIND=PRINTER)
      FILE  7(TITLE="G2B/CONTROL",KIND=PACK,FILETYPE=7)
      FILE  8(KIND=PACK,FILETYPE=0,UNITS=WORDS,MAXRECSIZE=14,MINRECSIZE=14,
           . BLOCKSIZE=420,FILEKIND=DATA,SECURITYTYPE=CLASSA,AREAS=300,
           . AREASIZE=450)
            DIMENSION X(1100),Y(1100),M(1100),G(500),G2FA(2),G2FB(2)
            REAL M
            DATA G2FA/"G2FA.      "/,G2FB/"G2FB.      "/
            READ(7,100) Z1,GSPACE
      C HEIGHT OF CALCULATION,GRID SPACING
        100 FORMAT(2F10.1)
            LOCK(7)
            READ(4,101) I
        101 FORMAT(I2)
            LOCK(4)
            IF (I .EQ. 0) OPEN(3,TITLE=G2FA)
            IF (I .EQ. 1) OPEN(3,TITLE=G2FB)
      C SET TITLE ACCORDING TO LAST FILE WRITTEN
            XMIN=583.
            XMAX=608.
      C X RANGE
            YMIN=366.
            YMAX=391.
      C Y RANGE
            NX=((XMAX-XMIN)/GSPACE)+1.
            NY=((YMAX-YMIN)/GSPACE)+1.0
            READ(3,102) N,H
        102 FORMAT(I5,E18.12)
            READ(3)
      C SKIP ONE RECORD ON INPUT FILE
            READ(3,103) (M(I),I=1,N)
        103 FORMAT(4E18.12)
            LOCK(3)
            READ(5,104) (X(I),Y(I),I=1,N)
        104 FORMAT(10X,2F9.4)
            LOCK(5)
            WRITE(8,200) GSPACE,Z1
        200 FORMAT('UP. CONT. GRID',F6.2,' HEIGHT',F6.2,'KM&')
            BMAX=-1.E+25
            BMIN=1.E+25
            A1=Z1-H
            A2=A1**2
            A4=YMIN
            DO 10 I=1,NY
            A3=XMIN
            DO 20 J=1,NX
            GG=0.0
            DO 30 K=1,N
            XX=X(K)-A3
            YY=Y(K)-A4
            ZZ=XX**2+YY**2+A2
            GG=GG+(M(K)*A1)/(ZZ*SQRT(ZZ))
         30 CONTINUE
            BMIN=AMIN1(BMIN,GG)
            BMAX=AMAX1(BMAX,GG)
            G(J)=GG
            A3=A3+GSPACE
         20 CONTINUE
      C OUTPUT IN CHOSEN FORMAT
            WRITE(8,201) I
        201 FORMAT(I3)
            WRITE(8,202) (G(J),J=1,NX)
        202 FORMAT(9F8.1)
            A4=A4+GSPACE
         10 CONTINUE
            CLOSE(8,DISP=CRUNCH)
            WRITE(6,203) BMIN,BMAX
        203 FORMAT(' MIN IS',F8.1/' MAX IS',F8.1)
```

```
7800      STOP
7900      END
```

```
      LOCK(4)
      OPEN(5,FILETYPE=7)
      WRITE(5,100) II
      LOCK(5)
      ITIME=ITIME+12000
      RETURN
      END
      SUBROUTINE INBIN(A,N,IFILE)
C  N WORD UNFORMATTED READ
      DIMENSION A(N)
      READ(IFILE) A
      RETURN
      END
      SUBROUTINE OUTBIN(A,N,IFILE)
C  N WORD UNFORMATTED WRITE
      DIMENSION A(N)
      WRITE(IFILE) A
      RETURN
      END
```