PROGRAM ESTPM

ESTIMATION OF SECONDARY TERRESTRIAL POSITIONS FOR MAPPING

DOCUMENTATION

J. A. R. BLAIS

July 1979

Physical Geodesy Section,
Geodetic Survey Division,
Surveys and Mapping Branch,
Dept. of Energy, Mines + Resources,
615 Booth St., Ottawa K1A 0E9.

TABLE OF CONTENTS

Page No.

INTRODUCTION	1
OPERATING INSTRUCTIONS	2
REFERENCES	5
PROGRAM ESTPM	A.1
SUBROUTINE NADATS	B.1
SUBROUTINE ATSNAD	C.1
SUBROUTINE DATAST	D.1
SUBROUTINE LSQSLN	E.1
SUBROUTINE CXSLN	F.1
SUBROUTINE GRID	G.1
FUNCTION KFN	H.1
COMPASS ROUTINE JCMEM	I.1
COMPASS ROUTINE KOREA	J.1
COMPASS ROUTINES COMSAVE & COMREST	K.1

omgre

INTRODUCTION

The program ESTPM has been developped to estimate corrections to geodetic positions by interpolating between the positional corrections which are known for a number of control stations. The estimation is done first, at the datum level with Vening-Meinesz' formula, then at the regional level with a least-squares complex polynomial for regional trends and finally at the local level using a covariance function for residual interpolation. The theoretical aspects of this approach and its approach in geodetic horizontal control densification are discussed in [Blais 1979; Boal and Junkins, 1978].

The analysis of ESTPM results are greatly simplified with the optional graphical outputs. The given correction vectors at the control stations and the estimated correction vectors at grid and other points provide a useful graphical representation of the results for assessment and analysis purposes. These outputs can be displayed using an interactive graphical computer terminal or a CALCOMP drum plotter.

The estimated coefficients and related information can also be saved or catalogued for future uses. These include the transformation of positions at a later date and/or further graphical analysis especially in overlapping areas.

OPERATING INSTRUCTIONS

The techniques of dynamic dimensioning have been used in developping the program ESTPM to facilitate its use in different contexts. Explicitly, all the matrix and vector dimensions are computed within the program at execution time depending on the amount of data read in and the chosen degree for the complex polynomial. When insufficient central memory is available, a diagnostic message is printed in the output listing giving the amount of additional core required and the execution is aborted.

The control information required by ESTPM is read in from two cards:

<u>Card One</u> : <u>Identification of Project</u>.
The 80 characters on this card are reproduced without any modification in the output listings for identification purposes.

<u>Card Two</u> : <u>Control Parameters</u>.

col. 1-2 : degree of complex polynomial; the degree must be set to 99 when the coefficient information is to be read in from TAPE11.

col. 3 : 0 - no transformation of datum.
1 - transformation: NAD27 → ATS77.
2 - transformation: ATS77 → NAD27.

col. 4 : 0 - no plot.
1 - plot with point numbers.
2 - plot without point numbers.

col. 5 : 0 - no grid.
1 - usual grid.
2 - grid symbols without vectors.

col. 6-10: Western limit in longitude (F5.1)
col. 11-15: Eastern limit in longitude (F5.1)
col. 16-20: Southern limit in latitude (F5.1)
col. 21-25: Northern limit in latitude (F5.1)
col. 26-30: scale in degrees of latitude per inch (F5.3)
default = 1.0 degree per inch
col. 31-35: scale of vectors in seconds per inch (F5.3)
default = 1.0 second per inch
col. 36-40: spacing of grid in degrees (F5.3)
default = 1.0 degree spacing
col. 41-45: maximum vector length in inches (F5.3)
default = 3.0 inches
col. 46-50: size of title letters in inches (F5.3)
default = 0.21 inches
col. 51-55: size of symbols in inches (F5.3)
default = 0.14 inches.
col. 56-60: size of point numbers in inches (F5.3)
default = 0.07 inches
col. 61-65: normalized unit distance for interpolation purposes (F5.3)
default = 0.1 (of radial dispersion)
col. 66-70: normalized unit distance for extension beyond peripheral control stations (F5.3)
default = 0.1 (of radial dispersion)
col. 80 : blank - GALS format for data cards.
otherwise - format read in from following card (8A10)
"Z" - no checking for position codes 4, 5, 6, 04, 05, 06 and 96.

Data Files

TAPE2 : first set of coordinates of control stations.
- not required when degree = 99.

TAPE3 : second set of coordinates of control stations.
- not required when degree = 99.

TAPE4 : first set of coordinates of other stations to be transformed.

TAPE11 : binary file of previously estimated coefficients and related information.
- required only when degree = 99.
- always prepared by ESTPM when degree ≠ 99.

Note

The system SORT/MERGE is used to find the matching stations on TAPE2 and TAPE3. Only the control stations within the grid limits are kept for the modelling. The resulting transformations are then only applied to those control stations and the other stations on TAPE4 within the area defined by the peripheral control stations and the parameter RUE.

REFERENCES

BLAIS, J.A.R. (1979): Least-Squares Estimatio of Geodetic Horizontal Control Densification. Presented at the Canadian Geophysical Union Meeting at U.N.B. (June 1979).

BOAL, J.D. and JUNKINS, D.R. (1978): Adjustment of Inertial Survey System Data with Satellite Doppler and Conventional Control. Collected Papers (1978), Geodetic Survey of Canada.

```
      PROGRAM  ESTPM  (INPUT,OUTPUT,TAPE1=INPUT,TAPE2,TAPE3,TAPE4,
     .                TAPE10=OUTPUT,TAPE11,TAPE12)
      COMMON A(1)
      COMPLEX CZ,ZZ,EE,FF,GG,CF,CZZ
      COMMON /REF/ X0,Y0,U0,V0,SX,SY,SU,SV,WP
      COMMON /GDL/ GW1,GW2,GN1,GN2,GRD,SGD,SCV,RCV,SBL
      DIMENSION NAME(8),FMT(8),BUFFER(1024)
      DATA  (FMT(I),I=1,8) / "(A3,4X,A7,","1X,A10,A5,","10X,2(F2.0",
     .      ",1X),F6.5,","1X,F3.0,1X",",F2.0,1X,F","8.5,F8.3,A",
     .      "1)        " /
  900 FORMAT (8A10)
  901 FORMAT (I2,3I1,4F5.1,9F5.3,9X,A1)
  902 FORMAT (1H1,/////20X,"EEEEEE   SSSS  TTTTTT  PPPPP   MM    MM",25
     .      30("*"),/20X,"EE        SS       TT     PP  PP  MMM MMM",25
     .      "* J.A.R. BLAIS - JULY 1979   *",/20X,"EEEE      SSS
     .      "T    PPPPP   MM M MM",25X,"* PHYSICAL GEODESY SECTION
     .      /20X,"EE          SS     TT     PP       MM   MM",25X,
     .      "* GEODETIC SURVEY OF CANADA  *",/20X,"EEEEEE  SSSS
     .      "TT     PP       MM    MM",25X,30("*")//)
  905 FORMAT (//10X,"IDENTIFICATION OF PROJECT",10X,8A10/)
  910 FORMAT (//10X,"ALTERNATE INPUT FORMAT",5X,8A10)
  914 FORMAT (/10X,"THE GRID LATITUDE LIMITS ARE  ",F6.1,"  AND  ",F6.
     .      "  (+ NORTH)",//10X,"THE GRID LONGITUDE LIMITS ARE ",F6.
     .      "  AND  ",F6.1,"  (+ WEST)")
  915 FORMAT (/10X,"WRONG LIMITS FOR THE GRID")
  921 FORMAT (///10X,"THE LOCAL ORIGIN IN LATITUDE IS  ",F8.3,
     .      "   DEGREES   (+ NORTH)")
  922 FORMAT (/10X,"THE LOCAL ORIGIN IN LONGITUDE IS ",F8.3,
     .      "   DEGREES   (+ WEST)")
  924 FORMAT (/10X,"THE NUMBER OF GEODETIC STATIONS USED IS  ",I10)
  925 FORMAT (//10X,"TRANSFORMATION FROM NAD27 TO ATS77   ",A10)
  926 FORMAT (//10X,"TRANSFORMATION FROM ATS77 TO NAD27   ",A10)
  931 FORMAT (//10X,"THE SCALE FACTOR IN LATITUDE IS  ",F8.3)
  932 FORMAT (/10X,"THE SCALE FACTOR IN LONGITUDE IS ",F8.3)
  934 FORMAT (//10X,"THE NORMALIZED UNIT DISTANCE IN LATITUDE IS  ",
     .      F10.2,3X,"METRES")
  935 FORMAT (//10X,"THE NORMALIZED UNIT DISTANCE IN LONGITUDE IS ",
     .      F10.2,3X,"METRES")
  940 FORMAT (///10X,"THE COMPUTED COEFFICIENTS ARE "/)
  942 FORMAT (20X,I5,5X,2F20.10)
  945 FORMAT (1H1///5X,"GEODETIC  STATIONS",13X,"COORDINATES",13X,
     .      "DIFFERENCES",22X,"ESTIMATES",15X,"RESIDUALS")
  948 FORMAT (///10X,"THE DEGREE OF THE EXPANSION IS ",I5,
     .      ///10X,"THE NUMBER OF COEFFICIENTS IS ",I6)
  949 FORMAT (///10X,"***  WARNING  ***   THE MAXIMUM NUMBER OF ",
     .      "COEFFICIENTS IS ONLY ",I5)
  950 FORMAT (//5X,"NUMBER",5X,"NAME",15X,"LAT.",6X,"LONG.",8X,
     .      "D(LAT)",6X,"D(LONG)",8X,"E(LAT)",10X,"E(LONG)",6X,
     .      "R(LAT)",1X,"R(LONG)",2X,"LENGTH"/)
  951 FORMAT (34X,"(DEG)",6X,"(DEG)",8X,"(SEC)",7X,"(SEC)",4X,
     .      3(4X,"(SEC)",2X,"(SEC)"),6X,"(M)"/)
  952 FORMAT (4X,2(1X,A10),A5,1X,2(F8.3,2X),3X,2(F8.3,4X),3(2X,2F7.3),
     .      3X,F7.3)
  955 FORMAT (//10X,"THE RMS VALUE IN LATITUDE IS   ",F10.3," SECONDS
  956 FORMAT (//10X,"THE RMS VALUE IN LONGITUDE IS  ",F10.3," SECONDS
  960 FORMAT (1H1,///5X,"IDENTIFICATION OF PROJECT",10X,8A10,/////5X,
     .      "TRANSFORMATION OF OTHER POINTS",46X,"APPLIED CORRECTIONS
```

```
     .                  " IN SECONDS")
  961 FORMAT (//5X,"NUMBER",5X,"NAME",15X,"LATITUDE",13X,"LONGITUDE",
     .        19X,"LATITUDE",8X,"LONGITUDE"/)
  962 FORMAT (5X,A7,1X,A10,A5,2I5,F10.5,3X,2I5,F10.5,10X,2(2F7.3,2X))
  965 FORMAT (A3,4X,A7,1X,A10,A5,9X,2I3,F9.5,1X,2I3,1X,F8.5,F8.3,A1)
  970 FORMAT (///10X,"THE REQUESTED CENTRAL MEMORY IS  ",O6,"  OCTAL
     .        "WORDS")
  990 FORMAT (/////10X,"THE REQUESTED CENTRAL MEMORY IS TOO SMALL BY
     .        I10,"  DECIMAL WORDS")
      X0=Y0=U0=V0=SX=SY=SU=SV=0.0
      ICR = 1
      IT2 = 2
      IT3 = 3
      IT4 = 4
      IPR = 10
      ITP = 11
      ITQ = 12
      SGN = - 1.0
      CODE = 3H   4
      QODE = 3H  04
      SCRD = ASIN(1.0)/90.0
      CALL  JCMEM (IPG,ICOM,JCM)
      CALL  KOREA (JCM,ICOM)
      CALL  COMSAVE (JCM)
C*****IDENTIFICATION OF THE PROJECT
      READ (ICR,900)  (NAME(I),I=1,8)
      WRITE (IPR,902)
      WRITE (IPR,905)   (NAME(I),I=1,8)
      READ (ICR,901)   NS,K1,K2,K3,GW1,GW2,GN1,GN2,SGD,SCV,GRD,RCV,
     .                 SBN,SBS,SBI,RUI,RUE,KF
      IF (KF.EQ.1H )   GO TO 1
      READ (ICR,900)  (FMT(I),I=1,8)
      WRITE (IPR,910)  (FMT(I),I=1,8)
    1 IF (NS.EQ.99)   READ (ITP) NR,NP,K1,RUI,X0,Y0,U0,V0,SX,SY,SU,SV
      IF (SGD.LE.0.0)   SGD = 1.0
      IF (SCV.LE.0.0)   SCV = 1.0
      IF (GRD.LE.0.0)   GRD = 1.0
      IF (SBN.LE.0.0)   SBN = 0.21
      IF (SBS.LE.0.0)   SBS = 0.14
      IF (SBI.LE.0.0)   SBI = 0.07
      IF (RCV.LE.0.0)   RCV = 3.0
      IF (RUI.LE.0.0)   RUI = 0.1
      IF (RUE.LE.0.0)   RUE = 0.1
      SBL = 2.0*SBN/3.0
      IF (GW1.NE.GW2)   GO TO 2
      IF (GN1.NE.GN2)   GO TO 2
      WRITE(IPR,914) GN1,GN2,GW1,GW2
      WRITE (IPR,915)
      STOP
    2 IF (GW1.GT.GW2)   GO TO 3
      GW0 = GW1
      GW1 = GW2
      GW2 = GW0
    3 IF (GN1.LT.GN2)   GO TO 4
      GN0 = GN1
      GN1 = GN2
      GN2 = GN0
```

should be 9

100

```
    4 IF (NS.NE.99)  NR = NS
      NF = NR + 1
      NC = 2*NF
      WRITE (IPR,948)  NR,NC
      WRITE (IPR,970) JCM
      KCM = JCM - LOCF(A(1))
      CKT = 10HNO
      IF (K1.EQ.1)  CKT = 10HYES
      WRITE (IPR,925)  CKT
      CKT = 10HNO
      IF (K1.EQ.2)  CKT = 10HYES
      WRITE (IPR,926)  CKT
      IF (NS.EQ.99)  GO TO 10
C*****LATITUDE IN DEGREES  (+ NORTH)
C*****LONGITUDE IN DEGREES (+ WEST)
      ITS = IT2
    5 READ (ITS,FMT) D0,D1,D2,D3,U1,U2,U3,V1,V2,V3,HH,HC
      IF (EOF(ITS).NE.0.0)  GO TO 8
      IF (KF.EQ.1HZ)  GO TO 6
      IF (D0.EQ.3H  4.OR.D0.EQ.3H 04)  GO TO 6
      IF (D0.EQ.3H  5.OR.D0.EQ.3H 05)  GO TO 6
      IF (D0.EQ.3H  6.OR.D0.EQ.3H 06)  GO TO 6
      IF (D0.EQ.3H 96)  GO TO 6
      GO TO 5
    6 IF (ITS.EQ.2)  GO TO 7
      IF (K1.EQ.2)  CALL  NADATS (U1,U2,U3,V1,V2,V3,HH)
      IF (K1.EQ.3)  CALL  ATSNAD (U1,U2,U3,V1,V2,V3,HH)
    7 X = SGN*(U1 + U2/60.0 + U3/3600.0)
      Y = SGN*(V1 + V2/60.0 + V3/3600.0)
      IF (ABS(X).LT.GN1)  GO TO 5
      IF (ABS(X).GT.GN2)  GO TO 5
      IF (ABS(Y).GT.GW1)  GO TO 5
      IF (ABS(Y).LT.GW2)  GO TO 5
      WRITE (ITP)  D1,D2,D3,X,Y
      GO TO 5
    8 IF (ITS.EQ.IT3)  GO TO 9
      ITS = IT3
      SGN = 1.0
      GO TO 5
    9 REWIND  ITP
      CALL  DATAST (ITP,ITQ,JCM)
      NP = INT (QP+0.1)
C*****LOCAL ORIGIN AS CENTRE OF GRAVITY OF DATA
   10 WRITE (IPR,921) X0
      WRITE (IPR,922) Y0
      WRITE (IPR,914) GN1,GN2,GW1,GW2
      WRITE (IPR,924) NP
      IF (NP.EQ.0)  STOP
      LCM = NF*(NF+1) + 4*NF + 5*NP - KCM
      IF (LCM.GT.0)  WRITE (IPR,990)  LCM
      IF (LCM.GT.0)  STOP
      N1 = 1
      N2 = 2*NP + 1
      N3 = 4*NP + 1
      N4 = 5*NP + 1
      N5 = N4 + 2*NF
      N6 = N5 + 2*NF
```

```
      NM = (NF*(NF+1))/2
      IF (NS.EQ.99)  GO TO 11
      CALL  LSQSLN (A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),NP,NR,NF,NM,I
C*****SCALE FACTORS FROM THE DISPERSION OF THE DATA
   11 WRITE (IPR,931) SX
      WRITE (IPR,932) SY
      RNX = RUI*111319.456/SX
      RNY = RUI*111319.456*COS(X0*SCRD)/SY
      WRITE (IPR,934) RNX
      WRITE (IPR,935) RNY
      IF (K2.EQ.0)  GO TO 12
      SGN = - SGD/COS(GN1*SCRD)
      ASW = (GW2-GW1)/SGN
      ASN = (GN2-GN1)/SGD
      CALL  PLOTS (BUFFER,1024)
      CALL  PLOT (0.0,-30.0,-3)
      CALL  PLOT (4.0,2.0,-3)
      CALL  SYMBOL (0.0,-2.0,SBN,NAME,0.0,80)
      CALL  SYMBOL (0.0,-1.0,SBL,"DEGREE OF ESTIMATION = ",0.0,23)
      CALL  NUMBER (5.0,-1.0,SBL,DR,0.0,-1)
      CALL SYMBOL  (0.0,-1.5,SBL,"VECTOR SCALE IN SECONDS PER INCH = ",
     .                0.0,35)
      CALL  NUMBER (5.0,-1.5,SBL,SCV,0.0,1)
      CALL  AXIS (0.0,0.0,27HLONGITUDE POSITIVE WESTWARD,-27,ASW,0.0,
     .                GW1,SGN)
      CALL  AXIS (0.0,0.0,27HLATITUDE POSITIVE NORTHWARD,27,ASN,90.0,
     .                GN1,SGD)
      CALL  AXIS (ASW,0.0,27HLATITUDE POSITIVE NORTHWARD,-27,ASN,90.0
     .                GN1,SGD)
200
      SGN = - SGD / COS(GN2*SCRD)
      GWN = (GW1+GW2-ASW*SGN) / 2.0
      CALL  AXIS (0.0,ASN,27HLONGITUDE POSITIVE WESTWARD,27,ASW,0.0,
     .                GWN,SGN)
   12 IF (NS.NE.99)  REWIND ITP
      WRITE (IPR,940)
      IF (NS.EQ.99)  READ (ITP) (A(I),I=1,N5)
      DO 14 I=1,NF
      II = I - 1
      C1 = A(N4+2*I-2)
      C2 = A(N4+2*I-1)
   14 WRITE (IPR,942) II,C1,C2
      RP = RQ = PU = PV = QU = QV = 0.0
      IF (NS.EQ.99)  GO TO 21
      WRITE (IPR,945)
      WRITE (IPR,950)
      WRITE (IPR,951)
      DO 20 I=1,NP
      READ (ITP) PT,PT1,PT2,X,Y,U,V,W
      CZ = CMPLX (SX*(X-X0),SY*(Y-Y0))
      FF = CMPLX (SU*(U-U0)-A(N2+2*I-2),SV*(V-V0)-A(N2+2*I-1))
      QQ = 0.0
      GG = CMPLX (0.0,0.0)
      DO 18 J=1,NP
      ZZ = CMPLX (A(N1+2*J-2),A(N1+2*J-1))
      EE = CMPLX (A(N2+2*J-2),A(N2+2*J-1))
      DD = (CZ-ZZ)*CONJG(CZ-ZZ) / (RUI*RUI)
      IF (DD.GT.675.0)  DD = 675.0
```

```
      WW = A(N3+J-1)
      QW = WW*EXP(-DD)
      GG = (QQ*GG + QW*EE) / (QQ + QW)
   18 QQ = QQ + QW
      FU = REAL (FF) / SU + U0
      FV = AIMAG (FF) / SV + V0
      GU = REAL (GG) / SU
      GV = AIMAG (GG) / SV
      WU = WV = 1.0
      RU = RV = 0.0
      RU = FU + GU - U
      QU = (PU*QU + WU*RU*RU) / (PU + WU)
      PU = PU + WU
      RP = RP + RU
      RV = FV + GV - V
      QV = (PV*QV + WV*RV*RV) / (PV + WV)
      PV = PV + WV
      RQ = RQ + RV
      RR = SQRT (RU*RU + (RV*COS(X*SCRD))**2)*30.92209533
      WRITE (IPR,952) PT,PT1,PT2,X,Y,U,V,FU,GU,FV,GV,RU,RV,RR
      IF (K2.EQ.0)  GO TO 20
      SGN = - SGD / COS(X*SCRD)
      GWN = (GW1+GW2-ASW*SGN) / 2.0
      PW = (Y-GWN) / SGN
      PN = (X-GN1) / SGD
      QW = PW - V*COS(X*SCRD)/SCV
      QN = PN + U / SCV
      IF (K2.EQ.2)  GO TO 19
      CALL  SYMBOL (PW+SBI,PN-SBI,SBI,PT,0.0,10)
      PQ = (PW-QW)**2 + (PN-QN)**2
      IF (PQ.LT.RCV*RCV)  GO TO 19
      CALL  SYMBOL (PW,PN,SBS,14,0.0,-1)
      GO TO 20
   19 CALL  SYMBOL (PW,PN,SBS,2,0.0,-1)
      CALL  PLOT (PW,PN,3)
      CALL  PLOT (QW,QN,2)
   20 CONTINUE
   21 CONTINUE
      IF (K3.EQ.1)  CALL  GRID (A(N1),A(N2),A(N3),A(N4),NP,NF,RUI,RUE
      IF (K3.EQ.2)  CALL  GRID (A(N1),A(N2),A(N3),A(N4),NP,NF,RUI,-1,
      IF (NS.EQ.99)  GO TO 48
      REWIND IT2
      QU = SQRT (QU)
      QV = SQRT (QV)
      WRITE (IPR,955)  QU
      WRITE (IPR,956)  QV
   48 KPT = 0
      QQQ = 0.0000001
   50 READ (IT4,FMT)  D0,D1,D2,D3,U1,U2,U3,V1,V2,V3,HH,HC
      IF (EOF(IT4).NE.0.0)  GO TO 90
      IF (KF.EQ.1HZ)  GO TO 51
      IF (D0.EQ.3H  4.OR.D0.EQ.3H 04)  GO TO 51
      IF (D0.EQ.3H  5.OR.D0.EQ.3H 05)  GO TO 51
      IF (D0.EQ.3H  6.OR.D0.EQ.3H 06)  GO TO 51
      IF (D0.EQ.3H 96)  GO TO 51
      GO TO 50
   51 IF (K1.EQ.2)  CALL  NADATS (U1,U2,U3,V1,V2,V3,HH)
```

```
      IF (K1.EQ.3)  CALL  ATSNAD (U1,U2,U3,V1,V2,V3,HH)
      X = U1 + U2/60.0 + U3/3600.0
      Y = V1 + V2/60.0 + V3/3600.0
      IF (ABS(X).LT.GN1)  GO TO 50
      IF (ABS(X).GT.GN2)  GO TO 50
      IF (ABS(Y).GT.GW1)  GO TO 50
      IF (ABS(Y).LT.GW2)  GO TO 50
      CZ = CMPLX (SX*(X-X0),SY*(Y-Y0))
      IF (CABS(CZ).GT.(1.0+RUE))  GO TO 50
      KPT = KPT + 1
      IF (KPT.GT.1)  GO TO 52
      WRITE (IPR,960)  (NAME(I),I=1,8)
      WRITE (IPR,961)
   52 CF = CMPLX (A(N4),A(N4+1))
300   CZZ = CMPLX (1.0,0.0)
      DO 54 K=2,NF
      CZZ = CZ*CZZ
   54 CF = CF + CZZ*CMPLX (A(N4+2*K-2),A(N4+2*K-1))
      QQ = 0.0
      CZZ = CMPLX (0.0,0.0)
      DO 60 I=1,NP
      ZZ = CMPLX (A(N1+2*I-2),A(N1+2*I-1))
      EE = CMPLX (A(N2+2*I-2),A(N2+2*I-1))
      DD = (CZ-ZZ)*CONJG(CZ-ZZ) / (RUI*RUI)
      IF (DD.GT.675.0)  DD = 675.0
      WW = A(N3+I-1)
      QW = WW*EXP(-DD)
      CZZ = (QQ*CZZ + QW*EE) / (QQ + QW)
   60 QQ = QQ + QW
      FU = REAL (CF) / SU + U0
      FV = AIMAG (CF) / SV + V0
      GU = REAL (CZZ) / SU
      GV = AIMAG (CZZ) / SV
      XX = 3600.0*X + FU + GU
      YY = 3600.0*Y + FV + GV
      IU1 = IFIX ((XX+QQQ)/3600.0)
      IV1 = IFIX ((YY+QQQ)/3600.0)
      IU2 = IFIX ((XX+QQQ)/60.0) - IU1*60
      IV2 = IFIX ((YY+QQQ)/60.0) - IV1*60
      U3 = XX - 60*IU2 - 3600*IU1
      V3 = YY - 60*IV2 - 3600*IV1
      WRITE (IPP,962)  D1,D2,D3,IU1,IU2,U3,IV1,IV2,V3,FU,GU,FV,GV
      WRITE (IT2,965)  D0,D1,D2,D3,IU1,IU2,U3,IV1,IV2,V3,HH,HC
      IF (K2.EQ.0)  GO TO 50
      SGN = - SGD / COS(X*SCRD)
      GWN = (GW1+GW2-ASW*SGN) / 2.0
      PW = (Y-GWN) / SGN
      PN = (X-GN1) / SGD
      QW = PW - (FV+GV)*COS(X*SCRD)/SCV
      QN = PN + (FU+GU)/SCV
      IF (K2.EQ.2)  GO TO 62
      CALL SYMBOL (PW+SBI,PN-SBI,SBI,D1,0.0,10)
      PQ = (PW-QW)**2 + (PN-QN)**2
      IF (PQ.LT.RCV*RCV)  GO TO 62
      CALL SYMBOL (PW,PN,SBS,10,0.0,-1)
      GO TO 50
   62 CALL SYMBOL (PW,PN,SBS,1,0.0,-1)
```

```
      CALL PLOT (PW,PN,3)
      CALL PLOT (QW,GN,2)
      GO TO 50
   90 IF (K2.NE.0)  CALL  PLOT (15.0,0.0,999)
      IF (NS.EQ.99)  GO TO 100
      REWIND ITP
      WRITE (ITP) NR,NP,K1,RUI,X0,Y0,U0,V0,SX,SY,SU,SV
      WRITE (ITP) (A(I),I=1,N5)
  100 REWIND IT2
      REWIND IT3
      REWIND ITP
      REWIND ITQ
      CALL  COMREST
      STOP
      END
```

355

OUTLINE

The program ESTPM estimates positional corrections at any location within the periphery of control stations with known positional corrections. This estimation approach is carried out at the datum, regional and local level, and the results can be displayed graphically by the program.

At the datum level, the transformation from NAD27 to ATS77 and its inverse are optional in ESTPM. These use the linearized formulas of Vening-Meinesz for differential changes in the semi-major axis, the flattening and the location of the ellipsoidal centre. Other similar transformations could easily be added with corresponding optional values for the input parameter KI.

At the regional level, a complex polynomial model is used to represent the regional trends in the given correction vectors at the control stations. For a given degree, the complex coefficients of the polynomial are estimated by least squares. There also is provision for situations where the input degree is too high for the input observational information in estimating only the lower-order coefficients numerically feasible with the higher-order ones set to zero.

At the local level, the remaining residual corrections at the control stations can be practically eliminated through interpolation with an appropriately calibrated covariance function. The covariance function used in ESTPM is the inverse exponential of the squared distance, which corresponds to a Gaussian stochastic process. The calibration of the covariance function is done using the input normalized unit distance in terms of the spatial dispersion of the control stations. By default, this normalized unit distance is defined as one tenth of the radial dispersion of the control stations about their centre of gravity.

The system SORT/MERGE is used to fin the matching stations on the input file TAPE2 and TAPE3. Out of those matching stations, only the ones within the input grid limits are kept for further process ing. Note that the grid limits are alwa required for this purpose even when n grid is wanted.

The optional graphical output is generate using the standard CDC Fortran calls to CALCOMP routines for PLOT, AXIS, SYMBOL, and so on. These are well documented in the CDC manuals. The interactive use of the program may necessitate mino changes to those CALL statements to conform with the interactive system software.

MAIN VARIABLES

NAME: array of 8 words (i.e., 80 characters) corresponding to the identification of the project.

NS : integer variable corresponding to the degree (NR) of the complex polynomial or 99 when the coefficient information is on TAPE11.

K1 : option variable for datum transformations.

K2 : option variable for plotting.

K3 : option variable for the grid.

GW1 : Western limit in longitude.

GW2 : Eastern limit in longitude.

GN1 : Southern limit in latitude.

GN2 : Northern limit in latitude.

SGD : scale in degrees of latitude per inch.

SCV : scale of vectors in seconds per inch.

GRD : spacing of grid in degrees.

RCV : maximum vector length in inche

SBN : size of title letters in inches.

SBS : size of symbols in inches.

SBI : size of point numbers in inches

RUI : normalized unit distance in term of radial dispersion of control stations with respect to their centre of gravity for the covariance funct

RUE : normalized unit distance in ter of radial dispersion of control stations with respect to their centre of gravity for extension beyond th peripheral control stations.

KF : control variable for data formats.

FMT : eight-word array containing the data format used.

BUFFER: 1024 word array for PLOTS routines.

JCM : central memory requested on the job submission card.

KCM : number of words available in the blank common area.

NP : number of control stations used in the modelling.

VARIABLE IN BLANK COMMON

A : array of variable length for dynamic dimensioning purposes.

VARIABLES IN COMMON /REF/

- as defined in the subroutine DATAS.

VARIABLES IN COMMON /GDL/

- as defined in the subroutine GRID.

```
1        SUBROUTINE  NADATS (U1,U2,U3,V1,V2,V3,HE)
   C*****TRANSFORMATION FROM NAD27 TO ATS77
   C*****INPUT AND OUTPUT LATITUDES ARE POSITIVE NORTHWARD
   C*****INPUT AND OUTPUT LONGITUDES ARE POSITIVE WESTWARD
   C*****U1,U2,U3 = LATITUDE IN DEG., MIN., SEC.
   C*****V1,V2,V3 = LONGITUDE IN DEG., MIN., SEC.
   C*****HE = ELLIPSOIDAL HEIGHT IN METRES
   C*****REF.  HEISKANEN AND MORITZ = PHYSICAL GEODESY  P.207, EQN.5-55
         SCRD = ASIN(1.0) / 90.0
         AAA = 6378135.0
         FFF = 1.0 / 298.257
         AA = 6378206.4
         BB = 6356583.8
         FF = (AA-BB)/AA
         DA = AAA - AA
         DF = FFF - FF
         DX = 15.0
         DY = - 165.0
         DZ = - 175.0
         UU = (U1 + U2/60.0 + U3/3600.0)*SCRD
         VV = 4.0*ASIN(1.0) - (V1 + V2/60.0 + V3/3600.0)*SCRD
         SU = SIN (UU)
         SV = SIN (VV)
         CU = COS (UU)
         CV = COS (VV)
         DU = (SU*CV*DX + SU*SV*DY - CU*DZ + 2.0*AA*SU*CU*DF) / AA
         DV = (SV*DX - CV*DY) / (AA*CU)
         DH = - CU*CV*DX - CU*SV*DY - SU*DZ - DA + AA*SU*SU*DF
         UU = UU + DU
         VV = 4.0*ASIN(1.0) - VV - DV
         U1 = AINT(UU/SCRD)
         U2 = AINT(UU*60.0/SCRD) - 60.0*U1
         U3 = UU*3600.0/SCRD - 3600.0*U1 - 60.0*U2
         V1 = AINT(VV/SCRD)
         V2 = AINT(VV*60.0/SCRD) - 60.0*V1
         V3 = VV*3600.0/SCRD - 3600.0*V1 - 60.0*V2
         HE = HE + DH
         RETURN
         END
```

OUTLINE

The subroutine NADATS is a linearized trans formation of geodetic coordinates from the NAD27 reference system to the ATS77 reference system, as used in the framework test adjustments of May '76 and October '77, respectively [e.g., Beattie et al, 1978]. The formulation is that of Vening-Meinesz for differential changes in the semi-major axis, flattening and location of ellipsoidal centre [see Heiskanen and Moritz, 1967, page 207, equation 5-55].

The input-output latitudes are assumed positive Northward; the input-output longitudes are assumed positive Westward and the heights are ellipsoidal.

ARGUMENTS

U1 : degrees of latitude
U2 : minutes of latitude
U3 : seconds of latitude
V1 : degrees of longitude
V2 : minutes of longitude
V3 : seconds of longitude
HE : ellipsoidal height in metres.

REFERENCES

BEATTIE, D.S.; BLAIS, J.A.R. and PINCH, M.C. (1978): *Test Adjustments of the Canadian Primary Horizontal Network*. Collected Papers (1978), Geodetic Survey of Canada.

HEISKANEN, W.A. and MORITZ, H. (1967): *Physical Geodesy*. W.H. Freeman and Co.

```
      SUBROUTINE  NADATS (U1,U2,U3,V1,V2,V3,HE)
C*****TRANSFORMATION FROM NAD27 TO ATS77
C*****INPUT AND OUTPUT LATITUDES ARE POSITIVE NORTHWARD
C*****INPUT AND OUTPUT LONGITUDES ARE POSITIVE WESTWARD
C*****U1,U2,U3 = LATITUDE IN DEG., MIN., SEC.
C*****V1,V2,V3 = LONGITUDE IN DEG., MIN., SEC.
C*****HE = ELLIPSOIDAL HEIGHT IN METRES
C*****REF.  HEISKANEN AND MORITZ = PHYSICAL GEODESY  P.207, EQN.5-55
      SCRD = ASIN(1.0) / 90.0
      AAA = 6378135.0
      FFF = 1.0 / 298.257
      AA = 6378206.4
      BB = 6356583.8
      FF = (AA-BB)/AA
      DA = AAA - AA
      DF = FFF - FF
      DX = 15.0
      DY = - 165.0
      DZ = - 175.0
      UU = (U1 + U2/60.0 + U3/3600.0)*SCRD
      VV = 4.0*ASIN(1.0) - (V1 + V2/60.0 + V3/3600.0)*SCRD
      SU = SIN (UU)
      SV = SIN (VV)
      CU = COS (UU)
      CV = COS (VV)
      DU = (SU*CV*DX + SU*SV*DY - CU*DZ + 2.0*AA*SU*CU*DF) / AA
      DV = (SV*DX - CV*DY) / (AA*CU)
      DH = - CU*CV*DX - CU*SV*DY - SU*DZ - DA + AA*SU*SU*DF
      UU = UU + DU
      VV = 4.0*ASIN(1.0) - VV - DV
      U1 = AINT(UU/SCRD)
      U2 = AINT(UU*60.0/SCRD) - 60.0*U1
      U3 = UU*3600.0/SCRD - 3600.0*U1 - 60.0*U2
      V1 = AINT(VV/SCRD)
      V2 = AINT(VV*60.0/SCRD) - 60.0*V1
      V3 = VV*3600.0/SCRD - 3600.0*V1 - 60.0*V2
      HE = HE + DH
      RETURN
      END
```

OUTLINE

The subroutine ATSNAD is a linearized tran formation of geodetic coordinates from the ATS77 referelce system to the NAD27 refe rence system, as used in the framework test adjustments of October '77 and May '76 respectively [e.g., Beattie et al., 1978]. The formulation is that of Vening-Meinesz for differential changes in the semi-majo axis, flattening and location of ellipsoida centre [see Heiskanen and Moritz, 1967: page 207, equation 5-55]

The input-output latitudes are assumed positive Northward; the input-output longitudes are assumed positive Westward and the heights are ellipsoidal.

ARGUMENTS

U1 : degrees of latitude
U2 : minutes of latitude
U3 : seconds of latitude
V1 : degrees of longitude
V2 : minutes of longitude
V3 : seconds of longitude
HE : ellipsoidal height in metres.

REFERENCES

BEATTIE, D.S.; BLAIS, J.A.R. and PINCH, M.C. (1978
Test Adjustments of the Canadian Primary Horizontal Network.
Collected Papers (1978), Geodetic Survey of Cana

HEISKANEN, W.A. and MORITZ, H. (1967): Physical Geodesy.
W. H. Freeman and Co

```
      SUBROUTINE  DATAST (ITP,ITQ,JCM)
      COMMON /REF/ X0,Y0,U0,V0,SX,SY,SU,SV,QP
      COMMON /SPAN/ XA,XB,YA,YB
      CALL  COMREST
      CALL  SMSORT (50)
      CALL  SMFILE ("SORT","BINARY",ITP,"REWIND")
      CALL  SMFILE ("OUTPUT","BINARY",ITQ,"REWIND")
      CALL  SMKEY (1,1,10,0,"LOGICAL")
      CALL  SMEND
      CALL  COMSAVE (JCM)
      QP = 0.0
      W = 1.0
      QT = BLK = 10H
    1 READ (ITQ)  PT,PT1,PT2,X,Y
      IF (EOF(ITQ).NE.0.0)  PT = BLK
      IF (PT.EQ.BLK)  GO TO 10
      IF (PT.NE.QT)  GO TO 10
    2 IF (X.GT.0.0)  GO TO 5
      X1 = (Q1*X1+X) / (Q1+W)
      Y1 = (Q1*Y1+Y) / (Q1+W)
      Q1 = Q1 + W
      GO TO 1
    5 X2 = (Q2*X2+X) / (Q2+W)
      Y2 = (Q2*Y2+Y) / (Q2+W)
      Q2 = Q2 + W
      Q2 = Q2 + W          repeat ??!!!
      GO TO 1
   10 IF (QT.EQ.BLK)  GO TO 12
      IF (Q1*Q2.LT.0.5)  GO TO 12
      X1 = - X1
      Y1 = - Y1
      U = 3600.0*(X2-X1)
      V = 3600.0*(Y2-Y1)
      WRITE (ITP) QT,QT1,QT2,X1,Y1,U,V,W
      QQ = QP + W
      X0 = (QP*X0+X1) / QQ
      Y0 = (QP*Y0+Y1) / QQ
      U0 = (QP*U0+U) / QQ
      V0 = (QP*V0+V) / QQ
      QP = QQ
      IF (QP.GT.1.5)  GO TO 20
      XA = XB = X1
      YA = YB = Y1
   20 XA = AMIN1 (XA,X1)
      XB = AMAX1 (XB,X1)
      YA = AMIN1 (YA,Y1)
      YB = AMAX1 (YB,Y1)
      SU = AMAX1 (SU,U*U)
      SV = AMAX1 (SV,V*V)
   12 IF (PT.EQ.BLK)  GO TO 40
      QT = PT
      QT1 = PT1
      QT2 = PT2
      X1 = Y1 = Q1 = 0.0
      X2 = Y2 = Q2 = 0.0
      GO TO 2
   40 IF (QP.GT.0.5)  GO TO 50
      SX=SY=SU=SV=0.0
```

```
      ENDFILE  ITP
      RETURN
   50 IF (XA.NE.XB)  GO TO 52
      SX = 1.0
      GO TO 53
   52 SX = 1.0 / AMAX1(XB-X0,X0-XA)
   53 IF (YA.NE.YB)  GO TO 54
      SY = 1.0
      GO TO 55
   54 SY = 1.0 / AMAX1(YB-Y0,Y0-YA)
   55 IF (SU.GT.U0*U0)  GO TO 56
      SU = 1.0
      GO TO 57
   56 SU = 1.0 / SQRT(SU-U0*U0)
   57 IF (SV.GT.V0*V0)  GO TO 58
      SV = 1.0
      GO TO 59
   58 SV = 1.0 / SQRT(SV-V0*V0)
   59 REWIND  ITP
      REWIND  ITP
      RETURN
      END
```

OUTLINE

The subroutine DATAST finds the control stations from the common input stations on TAPE2 and TAPE3, and their associated positional differences. The optional datum transformations of the coordinates have already been done in the main program. A number of numerical quantities such as centre of gravity and various dispersion parameters.

The sorting of the contents of ITP is done using SORT/MERGE with access in Fortran. The station numbers or character names are used to find the repeated ones and hence evaluate the corresponding positional differences for the control stations.

ARGUMENTS

ITP : logical file name for input to SORT and transfer of observations back to the main program.
ITQ : logical file name for output of SORT.
JCM : central memory requested on job submission card.

VARIABLES IN COMMON /REF/

X0 : latitude of centre of gravity of control station
Y0 : longitude of centre of gravity of control station
U0 : mean positional difference in latitude.
V0 : mean positional difference in longitude.
SX : scale factor in latitude
SY : scale factor in longitude.
SU : scale factor for positional differences in latitude
SV : scale factor for positional differences in longitude.

QP : number of control stations.

VARIABLES IN COMMON /SPAN/

XA : minimum latitude of control stations
XB : maximum latitude of control stations
YA : minimum longitude of control stations
YB : maximum longitude of control stations

```
      SUBROUTINE  LSQSLN (ZZ,EE,WW,CC,DD,AA,NP,NR,NF,NM,ITP)
      COMMON /REF/ X0,Y0,U0,V0,SX,SY,SU,SV,QP
      COMMON /SPAN/ X1,X2,Y1,Y2
      COMPLEX ZZ(NP),EE(NP),CC(NF),DD(NF),AA(NM),CF
      DIMENSION  WW(NP)
      DO 1 I=1,NF
    1 DD(I) = CMPLX (0.0,0.0)
      DO 2 J=1,NM
    2 AA(J) = CMPLX (0.0,0.0)
      DO 5 I=1,NP
      READ (ITP) PT,PT1,PT2,X,Y,U,V,W
      ZZ(I) = CMPLX (SX*(X-X0),SY*(Y-Y0))
      EE(I) = CMPLX (SU*(U-U0),SV*(V-V0))
      WW(I) = W
      CC(1) = CMPLX (1.0,0.0)
      DD(1) = DD(1) + WW(I)*EE(I)
      AA(1) = AA(1) + WW(I)
      DO 3 J=2,NF
      J1 = KFN(1,J,NF)
      CC(J) = ZZ(I)*CC(J-1)
    3 AA(J1) = AA(J1) + WW(I)*CC(J)
      IF (NF.EQ.1)  GO TO 5
      DO 4 K=2,NF
      DD(K) = DD(K) + WW(I)*EE(I)*CONJG(CC(K))
      DO 4 L=K,NF
      KL = KFN(K,L,NF)
    4 AA(KL) = AA(KL) + WW(I)*CC(L)*CONJG(CC(K))
    5 CONTINUE
      CALL  CXSLN (AA,DD,CC,NM,NF,NRT)
      DO 10 I=1,NP
      CF = CC(1)
      IF (NR.EQ.0)  GO TO 10
      DD(1) = CMPLX (1.0,0.0)
      DO 8 J=2,NF
      DD(J) = ZZ(I)*DD(J-1)
    8 CF = CF + CC(J)*DD(J)
   10 EE(I) = EE(I) - CF
      RETURN
      END
```

OUTLINE

The subroutine LSQSLN forms the normal equations directly from the contents of ITP, calls in the subroutine CXSLN to solve them and finally, evaluates the residuals at the control stations. The observational information on ITP was prepared by the subroutine DATAST.

In the normal equations, only the upper triangular part of the matrix is used and it is stored in vectorial form. The external function KFN is used to get the linear address corresponding to row and column indices in the matrix.

ARGUMENTS

ZZ : complex vector of control station position
EE : complex vector of given positional differences at input and residual positional differences at output.
WW: real vector of statistical weights.
CC: complex vector of coefficients for the complex polynomial.
DD: complex vector used for the right-hand side of the normal equations.
AA: complex vector for the upper triangular part of the matrix in the normal equations.
NP: number of control stations.
NR: degree of complex polynomial.
NF: number of coefficients for the polynomial (i.e., NF=NR+1)
NM: number of elements in AA (i.e., NM=(NF+1)NF/2).
ITP: logical unit for observational information.

VARIABLES IN COMMON /REF/

- as defined in the subroutine DATAST

VARIABLES IN COMMON /SPAN/

- as defined in the subroutine DATAST

```
      SUBROUTINE  CXSLN (AA,AB,CT,NM,NT,NRT)
      COMPLEX  AA(NM),AB(NT),CT(NT),PP,QQ
C*****FORWARD REDUCTION
      QQ = CSQRT (AA(1))
      DO 1 I=1,NT
    1 AA(I) = AA(I) / QQ
      AB(1) = AB(1) / QQ
      I1 = 0
      IF (NT.EQ.1)  GO TO 7
      DO 6 I=2,NT
      I1 = I - 1
      I2 = I + 1
      PP = CMPLX (0.0,0.0)
      DO 2 L=1,NT
    2 CT(L) = CMPLX (0.0,0.0)
      DO 4 K=1,I1
      KI = KFN(K,I,NT)
      DO 3 J=I,NT
      KJ = KFN(K,J,NT)
    3 CT(J) = CT(J) + CONJG(AA(KI))*AA(KJ)
    4 PP = PP + CONJG(AA(KI))*AB(K)
      II = KFN(I,I,NT)
      IF (REAL(AA(II)-CT(I)).LE.0.001)  GO TO 7
      QQ = CSQRT (AA(II)-CT(I))
      AA(II) = QQ
      IF (I.EQ.NT)  GO TO 6
      DO 5 M=I2,NT
      IM = KFN(I,M,NT)
    5 AA(IM) = AA(IM)/QQ - CT(M)/QQ
    6 AB(I) = AB(I)/QQ - PP/QQ
C*****BACKWARD SUBSTITUTION
    7 DO 8 I=1,NT
    8 CT(I) = CMPLX (0.0,0.0)
      IF (I1.EQ.(NT-1))  I1 = NT
      NRT = I1
      II = KFN(I1,I1,NT)
      CT(I1) = AB(I1) / AA(II)
      IF (I1.EQ.1)  RETURN
      I2 = I1 - 1
      DO 10 I=1,I2
      J = I2 - I + 1
      JJ = KFN(J,J,NT)
      CT(J) = AB(J) / AA(JJ)
      JP1 = J + 1
      DO 9 K=JP1,I1
      JK = KFN(J,K,NT)
    9 CT(J) = CT(J) - AA(JK)*CT(K)/AA(JJ)
   10 CONTINUE
      RETURN
      END
```

OUTLINE

The subroutine CXSLN solves the complex normal equations using Choleski's square root algorithm in its complex form [e.g., Householder, 1964: page]. As the upper triangular part of the matrix is kept in vectorial form, the function KFN is used to get the linear address corresponding to row and column indices in the matrix.

A numerical check is carried out on the reduced diagonal elements to the effect that the reduction procedure would be stopped if they became inferior to 10^{-3} in magnitude. The back substitution would then be initiated from that diagonal element, with all the higher-order coefficien set to zero [e.g., Blais, 1979].

ARGUMENTS

AA : complex vector for the upper triangul part of the matrix in the normal equations.

AB : complex vector used for the right-har side of the normal equations.

CT : complex vector of coefficients for the complex polynomial.

NM : number of elements in AA.

NT : number of coefficients for the polynomial

NRT: number of evaluated coefficients, i.e., NRT $\leq$ NT.

REFERENCES

BLAIS, J.A.R. (1979): Least-Squares Estimation in Geodetic Horizontal Control Densification
Presented at the Canadian Geophysical Union Meeting at U.N.B.

HOUSEHOLDER, A.S. (1964): The Theory of Matrices in Numerical Analysis
Blaisdell Publishing Company.

```
      SUBROUTINE  GRID (ZZ,EE,WW,CFN,NP,NF,RUI,RUE)
      COMMON /REF/ X0,Y0,U0,V0,SX,SY,SU,SV
      COMMON /GDL/ GW1,GW2,GN1,GN2,GRD,SGD,SCV,RCV,SBL
      COMPLEX ZZ(NP),EE(NP),CFN(NF),CZ,CZZ,CF
      DIMENSION WW(NP)
      SCRD = ASIN(1.0) / 90.0
      IF (GW1-GRD.LT.GW2)  RETURN
      GN = GN1 + GRD
      IF (GN.GT.GN2)  RETURN
      SGN = - SGD / COS(GN1*SCRD)
      ASW = (GW2-GW1) / SGN
    1 GW = GW1 + GRD
    2 GW = GW - GRD
      SGN = - SGD / COS(GN*SCRD)
      GWN = (GW1+GW2-ASW*SGN) / 2.0
      PW = (GW - GWN) / SGN
      PN = (GN - GN1) / SGD
      CZ = CMPLX (SX*(GN-X0),SY*(GW-Y0))
      CF = CFN(1)
      CZZ = CMPLX (1.0,0.0)
      DO 3 I=2,NF
      CZZ = CZ*CZZ
    3 CF = CF + CFN(I)*CZZ
      QQ = 0.0
      CZZ = CMPLX (0.0,0.0)
      DO 5 I=1,NP
      DD = (CZ-ZZ(I))*CONJG(CZ-ZZ(I)) / (RUI*RUI)
      IF (DD.GT.675.0)  DD = 675.0
      QW = WW(I)*EXP(-DD)
      CZZ = (QQ*CZZ + QW*EE(I)) / (QQ + QW)
    5 QQ = QQ + QW
      CF = CF + CZZ
      EU = REAL(CF)/SU + U0
      EV = AIMAG(CF)/SV + V0
      IF (CABS(CZ).GT.(1.0+RUE))  EU = EV = 0.0
      QW = PW - EV*COS(GN*SCRD)/SCV
      QN = PN + EU/SCV
      PQ = (PW-QW)**2 + (PN-QN)**2
      IF (PQ.LT.RCV*RCV)  GO TO 8
      CALL SYMBOL(PW,PN,SBL,11,0.0,-1)
      GO TO 10
    8 CALL SYMBOL(PW,PN,SBL,3,0.0,-1)
      CALL  PLOT (PW,PN,3)
      CALL  PLOT (QW,QN,2)
   10 IF (GW-GRD.GE.GW2)  GO TO 2
      GN = GN + GRD
      IF (GN.LT.GN2)  GO TO 1
      RETURN
      END
```

OUTLINE

The subroutine GRID generates the grid points and estimates the correction vectors at those locations. The plotting routines SYMBOL and PLOT are also called in to display the results graphically when the plotting options are chosen. The estimation of the correction vectors at the grid points is carried out in exactly the same manner as for the stations on TAPE4 in the main program

ARGUMENTS

ZZ: complex vector of control station positio
EE: complex vector of residual positional differences at the control stations.
WW: real vector of statistical weights.
CFN: complex vector of coefficients for the polynomial.
NP: number of control stations.
NF: number of coefficients for the polynomial.
RUI: normalized unit distance for the covariance function.
RUE: normalized unit distance for extension beyond peripheral control stations.

VARIABLES IN COMMON /REF/

- as defined in the subroutine DATAST

VARIABLES IN COMMON /GDL/

GW1: Western limit in longitude.
GW2: Eastern limit in longitude.

GN1: Southern limit in latitude.
GN2: northern limit in latitude.
GRD: spacing of grid in degrees.
SGD: scale of grid in degrees of latitude per inch.
SCV: scale of vectors in seconds per inch.
RCV: maximum vector length in inches
SBL: size of symbols in inches.

```
      FUNCTION  KFN (I,J,N)
      KFN = (I-1)*N - (I*I-I)/2 + J
      RETURN
      END
```

OUTLINE

The function KFN computes a vector address for each double row and column index in the upper triangular part of a square matrix.

ARGUMENTS

I: row index in upper triangular matrix
J: column index in upper triangular matrix
N: order of the matrix.

```
                           IDENT   JCMEM
                           ENTRY    JCMEM
                           USE     //
       1       A           BSS     1              SET A AS FIRST WORD IN BLANK COMMON
                           USE     0
000000         STATUS      VFD     30/-1,29/0     SET -1 IN BITS 30-59 REST 0
000000         JCMEM       DATA    0
                           SA3     A1             SAVE ADDRESS OF F.P.  1
15                         MEMORY    CM,STATUS,R
                           SA2     STATUS         GET THE RETURNED CM VALUE IN BITS 30-59
36                         AX2     30             SHIFT TO STANDARD INTEGER FORMAT
                           SX7     A                PUT ADDRESS OF BLANK COMMON IN F.P. 1
0                          SA7     X3             STORE VALUE
                           SA3     A3+1           SET ADDRESS TO F.P.  2
27                         IX7     X2-X7          CALCULATE SIZE OF BLANK COMMON
                           SX7     X7-1
)                          SA7     X3             STORE IN F.P.  2
                           SA3      A3+1          SET ADDRESS OF F.P.  3
22                         BX7      X2            STORE MAX FL IN F.P.  3
  53730                    SA7     X3
                           EQ      JCMEM
                           END

)200B CM  STORAGE USED                  50 STATEMENTS          4 SYMBOLS
          MODEL 74 ASSEMBLY          0.148 SECONDS             9 REFERENCES
```

OUTLINE

The compass routine JCMEM is used to find the central memory requested on the job submission card. It also returns the number of words used by the program in central memory and the blank common area available

ARGUMENTS

IPG : size of program load.
ICOM : size of blank common area.
JCM : total space available.

REFERENCE

BEATTIE, D.S. (1978): Documentation of Program GANET. Geodetic Survey of Canada.

```
                           IDENT   KOREA
                           ENTRY   KOREA
      1        STATUS      BSS     1
                           USE     //
               A           BSS     0
                           USE     0
)00000         KOREA       DATA    0
                           SA3     X1
                           BX6     X3
)                          LX6     30
                           SA6     STATUS
10                         SA3     A1
                           MEMORY  CM,STATUS,R,,1
                           SA2     STATUS
36                         AX2     30
 10722                     BX7     X2
                           SA7     X3                 STORE FIELD LENGTH
)U C                       SX5     A
 37775                     IX7     X7-X5
                           SX7     X7-1
)C0001                     SA3     A3+1
                           SA7     X3                 STORE KOMMON SIZE
 +                         EQ      KOREA
                           END

-)200B CM  STORAGE USED             53 STATEMENTS          4 SYMBOLS
           MODEL 74 ASSEMBLY     0.154 SECONDS            10 REFERENCES
```

OUTLINE

The compass routine KOREA is used to reset the central memory to the maximum available, i.e., the job control card value obtained with JCMEM. It also returns the amount of space available for the blank common.

ARGUMENTS

JCM : total space available in central memory
ICOM: space available for the blank common

REFERENCE

BEATTIE, D.S. (1978): Documentation of Program GANET. Geodetic Survey of Canada.

```
                        IDENT   COMSAVE
                        ENTRY   COMSAVE
                        ENTRY   COMREST
       1      COMSAVE   BSSZ    1                    SAVE DABA TO FL
                        SB5     B0                   0 MEANS NO FL ARG
 ; +                    ZR      X1,NOFL              IF NO ARG
 53210                  SA2     X1                   FETCH FL VALUE
                        SB5     X2                   SAVE DESIRED FL
              NOFL      BSS     0
                        SA1     65B                  RA+65B
10                      SX6     X1
                        SA6     CMMACTV
 )00000 +               PL      X6,COMSAVE           IF CMM NOT ACTIVE
                        BX6     -X6
56 +                    SA6     DABA
                        SA6     OUT
 )00051 +               SA6     FIRST                SET FET POINTERS FOR WRITE
                        SA2     X6                   GET DYNAMIC AREA HEADER
                        SX7     X2                   FIELD LENGTH AS KNOWN BY CMM
 )00055 +               SA7     FL
                        SA7     LIMIT
7777776                 SX7     X7-1                 FL-1
                        SA7     IN
 )                      SA3     X7                   GET WORD AT FL-1
 10733                  BX7     X3                   SAVE IT (CANT BE WRITTEN)
                        SA7     FLSAVE
 )00024                 SX2     24B
                        RJ      CIO                  WRITE DABA TO FL TO SAVE FILE
                        SX2     50B
 )00033 +               RJ      CIO                  REWIND SAVE FILE
                        EQ      B5,B0,COMSAVE        RETURN IF NO FL ARG
50                      SX2     B5                   DESIRED FL
                        RJ      CMEM                 ISSUE CMM MEM REQUEST FOR FL
                        EQ      COMSAVE              RETURN


       1      COMREST   BSSZ    1                    RESTORE DABA TO FL
                        SA2     CMMACTV
 )00020 +               PL      X2,COMREST           RETURN IF NOTHING SAVED
                        MX6     0
                        SA6     A2                   INDICATE SAVE FILE IS OUT-OF-DATE
 )00055 +               SA2     FL                   CMM FL
                        RJ      CMEM                 ADJUST FL TO WHAT CMM WANTS
                        SA2     FIRST
 )                      SX6     X2
                        SA6     IN                   RESET FET POINTERS
)000053 +               SA6     OUT
                        SX2     10B
 )00033 +               RJ      CIO                  READ SAVE FILE TO DABA TO FL
                        SX2     50B
)000033 +               RJ      CIO                  REWIND SAVE FILE
                        SA2     FLSAVE
 2                      BX6     X2
                        SA3     FL
-777776                 SA6     X3-1                 RESTORE WORD AT FL-1
                        EQ      COMREST              RETURN
```

```
      1    CIO      BSSZ   1                   ISSUE CIO FUNCTION IN X2
                    SA3    FET
                    MX0    42
 11603              BX6    X0*X3               CLEAR OLD FUNCTION
                    BX6    X6+X2               OR IN NEW FUNCTION
                    SA6    A3                  TO FET
 31117              SYSTEM CIO,RECALL,FET    CALL CIO FOR FUNCTION
                    EQ     CIO                 RETURN


      1    CMEM     BSSZ   1                   ISSUE CMM MEMORY REQUEST
                    BX6    X2
                    LX6    30
)000004             SX2    4
                    BX6    X6+X2               FL+CMM FLAG
    +               SA6    MEMREQ
                    MEMORY CM,MEMREQ,RECALL
                    EQ     CMEM                RETURN


 00000     FET      VFD    42/7LZZZZZUF,18/0
      1    FIRST    BSSZ   1
      1    IN       BSSZ   1
      1    OUT      BSSZ   1
      1    LIMIT    BSSZ   1


      1    FL       BSSZ   1                   FIELD LENGTH KNOWN TO CMM
      1    DABA     BSSZ   1                   DYNAMIC AREA BASE ADDRESS
      1    FLSAVE   BSSZ   1                   WORD AT FL-1 SAVED
      1    MEMREQ   BSSZ   1                   ARG FOR MEM
      1    CMMACTV  BSSZ   1                   NEGATIVE IF CMM ACTIVE (DABA TO FL
                    END

..200B CM  STORAGE USED          130 STATEMENTS          16 SYMBOLS
           MODEL 74 ASSEMBLY    0.305 SECONDS            52 REFERENCES
```

OUTLINE

The compass routines COMSAVE and COMREST ha been prepared by B. Fillmore of the Computer Science Division of E.M.R. for programs which extend the blank common area when CMM is operational because of call to SORT/MERGE or Record Manager under the CDC operating system NOS/BE.

The routine COMSAVE is called prior to overextending the blank common area in order to save the vital CMM information.

The routine COMREST is called when through with the blank common area prior to executing some CMM-invoking function.

ARGUMENT

JCM: job central memory as requested on the job submission card.