INIS

TRN AU8004694

AAEC/E488

AAEC/E488

AUSTRALIAN ATOMIC ENERGY COMMISSION

RESEARCH ESTABLISHMENT

LUCAS HEIGHTS

A FORTRAN IV LEAST SQUARES COMPUTER PROGRAM FOR THE PROFILE REFINEMENT OF CUBIC POWDER DIFFRACTION PATTERNS WITH CUBIC HARMONIC FUNCTIONS

by

J.C. TAYLOR

June 1980

ISBN 0 642 59689 1

AUSTRALIAN ATOMIC ENERGY COMMISSION
RESEARCH ESTABLISHMENT
LUCAS HEIGHTS

A FORTRAN IV LEAST SQUARES COMPUTER PROGRAM FOR THE PROFILE REFINEMENT OF CUBIC POWDER DIFFRACTION PATTERNS WITH CUBIC HARMONIC FUNCTIONS

by

J.C. TAYLOR

ABSTRACT

A cubic harmonic least squares Fortran IV program has been written for the refinement of cubic structures with disordered or rotating groups, e.g. MX_6, which cannot be refined satisfactorily with classical methods. The program has been used to refine neutron powder data for the plastic cubic phases of MoF_6, WF_6, SF_6 and Na_2UBr_6. The anion density is assumed to lie on the surface of a sphere of radius the M-X distance. The powder pattern intensities and relevant crystal data are input and the program refines the M-X distance, a scale factor, an overall Debye-Waller factor and the cubic harmonic coefficients by a full matrix, least squares treatment. The program is adaptable, with small changes in the structure factor expression, to other disordered cubic structures.

National Library of Australia card number and ISBN 0 642 59689 1

The following descriptors have been selected from the INIS Thesaurus to describe the subject content of this report for information retrieval purposes. For further details please refer to IAEA-INIS-12 (INIS: Manual for Indexing) and IAEA-INIS-13 (INIS: Thesaurus) published in Vienna by the International Atomic Energy Agency.

FORTRAN; COMPUTER CODES; CRYSTALS; CUBIC LATTICES; NEUTRON DIFFRACTION

CONTENTS

1. INTRODUCTION 1

2. MATHEMATICAL DETAILS 1

3. DATA INPUT 3
   3.1 JCL 3
   3.2 Control Cards 3
   3.3 Data Cards 4
   3.4 Parameter Selection Card 4
   3.5 Definitions 4
   3.6 Other Details 5
   3.7 Output 6

4. ACKNOWLEDGEMENTS 6

5. REFERENCES 6

APPENDIX A Program Listing 9

1. INTRODUCTION

Many disordered crystals showing cubic symmetry are only cubic on average. For example, cubic MoF_6, WF_6, SF_6, K_3ZrF_7 and CBr_4 have reorienting groups with anion density distributed spherically about the central cation. Diffraction studies of these phases assuming discrete spherical or ellipsoidal anions are not satisfactory because the disordered anion density is smeared out and curved. Although low R-factors may be obtained by classical refinements, the bond-lengths and temperature factors for the disordered atoms will be in error. A better approach is to use cubic harmonic functions which assume the rotational motions and which, furthermore, have very few least squares variables.

Cubic harmonic analyses have been made with single-crystal data [Seymour and Pryor 1970; More, Lefebre and Fouret 1977]. No computer program has been reported for profile refinement of powder diffraction patterns with these functions. The program described here was written for the analysis of neutron powder data for the plastic cubic phases of MoF_6, WF_6 and SF_6. For these compounds, the cubic harmonic refinements have worked well, and the results obtained with the program have been fully described in the literature [Levy et al. 1975; Levy, Taylor and Wilson 1976; Taylor and Waugh 1977]. As mentioned above, the program is only designed for smeared out cubic distributions, but would not work for anions well defined in angular position. Relevant details such as formulae and input data are given here for others wishing to use the method. A magnetic tape of the program in Fortran IV may be obtained from the author. A listing is given at Appendix A for completeness of documentation.

2. MATHEMATICAL DETAILS

For the MoF_6, WF_6 and SF_6 (MX_6) cases, with Mo, W or S at (000) and body-centred cubic positions, the anions lie on the surface of a sphere of radius c about the cation M. The normalised anion distribution

$$t_x(\underline{r}) = \frac{6}{4\pi c^2} \Sigma\, a_m K_m\, \delta(r-c) \quad ,$$

where

$$K_0 = 1,$$

$$K_1 = 0,$$

$$K_2 = (x^4+y^4+z^4)/(x^2+y^2+z^2)^2 - 3\frac{K_0}{5} ,$$

$$K_3 = x^2y^2z^2/(x^2+y^2+z^2)^3 + \frac{K_2}{22} - \frac{K_0}{105} ,$$

$$K_4 = \frac{x^8+y^8+z^8}{(x^2+y^2+z^2)^4} - 28\frac{K_3}{5} - 210\frac{K_2}{143} - \frac{K_0}{3} ,\text{etc.}$$

are the cubic harmonic functions [Von der Lage and Bethe 1947], a_m are amplitude coefficients, c is the M-X distance and $\delta(r-c)$ is the delta function. The program includes harmonics up to K_4. K_0 alone is for a spherically symmetric case, K_2 assumes peaks in <100> directions and K_3 in <111> directions.

The anion contribution to the structure factor has the term $T_x(hkl) = 6\Sigma\, a_m B_m J_{2m}$ where the B_m are the same as the corresponding K_m with x, y and z replaced by h, k and l, and the J_{2m} are the spherical Bessel functions of order 2m. The T_x are the Fourier transforms of the K_m. The latter obey the recurrence formulae:

$$J_{n+1}(x) = \frac{n}{x} J_n(x) - \frac{d}{dx} J_n(x)$$

where

$$J_0(x) = \frac{\sin x}{x}, \quad J_1(x) = \frac{\sin x}{x^2} - \frac{\cos x}{x},$$

$$J_4(x) = \sin x\left(\frac{105}{x^5} - \frac{45}{x^3} + \frac{1}{x}\right) - \cos x\left(\frac{105}{x^4} - \frac{10}{x^2}\right), \text{ and}$$

$$J_6(x) = \sin x\left(\frac{10395}{x^7} - \frac{4725}{x^5} + \frac{210}{x^3} - \frac{1}{x}\right) - \cos x\left(\frac{10395}{x^6} - \frac{1260}{x^4} + \frac{21}{x^2}\right) ,\text{etc.}$$

Here $x = kc$, where $k = 4\pi \sin\theta/\lambda$ [Seymour 1970].

The structure factor expression is, for the neutron case,

$$F(hkl) = \exp(-B \sin^2\theta/\lambda^2) \left\{ b_m + b_x T_x(hkl) \right\},$$

where B is an isotropic Debye-Waller factor, and b_m and b_x are the scattering lengths of the cation and anion. The calculated intensity at a point in the powder pattern is

$$I_c = \sum_{hkl} S\, W_{hkl} \exp(-\,2B \sin^2\theta/\lambda^2) \left\{ b_m + b_x\, T_x(hkl) \right\}^2 ,$$

where S is the scale factor, and W a Gaussian weighting term which depends on the deviation of the 2θ value of the data point from the theoretical 2θ value of the (hkl) reflexion.

The profile-fitting method is described by Rietveld [1969]. The present problem was linearised by a Taylor expansion about the initial least squares parameters, and the full matrix, least squares procedure follows that of Hamilton [1964]. The program is set to run for a certain number of cycles. After a given cycle, not necessarily the last, the observed and calculated powder patterns are printed. The program occupies 72K of core. Thirteen cycles with six variables took 5 minutes and 34 seconds of CPU time on an IBM360/50H computer (CPU times are 10 times less on the IBM370 computer).

3. DATA INPUT

3.1 JCL

The program uses arrays and not discs for storage, so no special JCL cards are needed other than those for running a normal Fortran program.

3.2 Control Cards

Card 1 λ, a^*, b^*, c^*, $\cos\alpha^*$, $\cos\beta^*$, $\cos\gamma^*$ - 7F10.5

Card 2 h_{min}, k_{min}, l_{min}, h_{max}, k_{max}, l_{max} - 6I5

Card 3 R1, R2, R3, BBMO, BBF, OVB - 6F10.5

Card 4   XF, R4, ACELL, ICARD, ICC, T2CORR - 3F10.5, 2I2, F10.5

3.3 Data Cards

THI, II, IBI, WI   - F10.5, 2I10, F10.5

One card for every data point. Blank card at end of data. Regions with no reflexions are not punched up. (II-IBI) cannot be 1, or the program will stop (1 is a flag for superposed h,k,l). In this case, II or IBI should be changed by one count.

3.4 Parameter Selection Card

(KI(I), I=1,6)   - 7I1

1 for refine, 0 for no refinement. See below for description of the KI.

3.5 Definitions

λ : wavelength in Å

a^*, b^*, c^*, $\cos \alpha^*$, $\cos \beta^*$, $\cos \gamma^*$: reciprocal cell parameters

h_{min}, k_{min}, l_{min}, h_{max}, k_{max}, l_{max} : minimum and maximum values of h, k, l expected over the range of the pattern. The minimum values are 0 and the maximum values positive. Exceeding the maximum (hkl) in the pattern will not matter. The program sets h>k>l.

R1 : a scale factor

R2, R3 and R4 : the cubic harmonic coefficients a_2, a_3 and a_4. R1 alone gives a spherically symmetrical distribution.

BBMO : scattering length of cation M.

BBF : scattering length of anion X.

OVB : overall Debye-Waller factor in $Å^2$.

XF : initial X-F distance in Å.

ACELL : cubic unit cell edge in Å.

ICARD : intensities are printed, punched before the ICARD 'th' cycle.

ICC : number of least squares cycles.

T2CORR : 2θ zero angle in degrees.

THI : $2\theta_i$ of observation in degrees.

II, IBI : observed intensity, background intensity.

WI : machine halfwidth in degrees 2θ at the point.

KI(I), I=1,6 : parameter selection integers (1 or 0) for refinement of scale (R1), a_2, (R2), a_3, (R3), X-F distance, a_4 (R4) and overall B.

3.6 Other Details

Matrix inversion is carried out by a local subroutine SID [Robinson 1968]. Multiplicity data are coded explicitly for m3m. These would need changing for m3 [International Tables for X-ray Crystallography 1965]. The present program holds for MX_6 groups on the cubic lattice points. If ions at other sites should occur, the structure factor expression in the program would need an additional term, but the least squares part of the program would be unaffected. The program has been run, modified in this way, for the high temperature phase of Na_2UBr_6 [Gaune-Escarde et al. 1979]. For MX_4 groups, the normalisation constants would change from 6 to 4. Peak asymmetry corrections [Rietveld 1969], should be applied to the data where necessary before using the program. Note a $\sin\theta$ limit in the program just before statement 40 which may need changing. The number of pattern points (360 in the program) set by dimensions of LLH, etc. may also need changing.

3.7 Output

The program first lists the input data, including the neutron pattern. The (hkl) are listed when they occur in the pattern with their (II-IBI), σ(II-IBI), Gaussian weights and multiplicities. Where more than one (hkl) term occurs, the (II-IBI) and σ(II-IBI) are given as one for the (hkl) preceding the last (hkl) of the superposed group. These data are followed by the weighted and unweighted R factors, old and new parameters, correlation coefficients and chi-squared values for each of the ICC cycles. The observed and calculated patterns are printed out where nominated in the cycle output.

4. ACKNOWLEDGEMENTS

I wish to thank Dr A.W. Pryor and Dr P. Sanger for helpful discussion.

5. REFERENCES

Gaune-Escarde, M., Bros, J., Hewat, A.W. and Taylor, J.C. [1979] - unpublished work.

Hamilton, W.C. [1964] - Statistics in Physical Science. Ronald Press, New York.

International Tables for X-ray Crystallography [1965]. Kynoch Press, Birmingham, UK. p.32.

Levy, J.H., Sanger, P.L., Taylor, J.C. and Wilson, P.W. [1975] - Acta Cryst., B31:1065.

Levy, J.H., Taylor, J.C. and Wilson, P.W. [1976] - J. Less Common Metals, 45:155.

More, M., Lefebre, J. and Fouret, R. [1977] - Acta Cryst., B33:3862.

Rietveld, H.M. [1969] - J. Appl. Cryst., 2:65.

Robinson, G.S. [1968] - AAEC/E190.

Seymour, R.S. [1970] - Ph.D. Thesis, University of New South Wales.

Seymour. R.S. and Pryor, A.W. [1970] - Acta Cryst., B26:1487.

Taylor, J.C. and Waugh, A.B. [1977] - J. Solid State Chem., 18:241.

Von der Lage, F.C. and Bethe, H.A. [1947] - Phys. Rev., 71:61.

APPENDIX A

PROGRAM LISTING

```
                    80/80 LISTING OF CARD INPUT.
C PROGRAM IS SET UP FOR NA2UBR6 CASE
      DIMENSION JJH(500),JJK(500),JJL(500),TTH2(500)
      DIMENSION IIH(50),IIK(50),IIL(50),SSS(50),IM(50)
      REAL*8 CC,CCC,CCCC,CCCD,D,DD,DDD,DDDD,B1,B2,B3,B4,B5,B6,B7
      REAL*8 B8,B9,B10,B11,D2,D3
      DIMENSION A(500,6),FF(500),PII(500),AB(6,500),AC(6,6)
      DIMENSION AD(6),AF(6),AG(500),AH(500),AL(6,6)
      DIMENSION LLH(500),LLK(500),LLL(500),FFF(500),SSSIG(500),
     * SSSS(500),STHI(500)
      DIMENSION SG(6),COR(6,6),KI(6)
      LL=1
      LLX=1
      IQ=1
      I=1
      READ(1,1) WAV,AS,BS,CS,CAS,CBS,CGS
    1 FORMAT(7F10.5)
      WRITE(3,2)WAV,AS,BS,CS,CAS,CBS,CGS
    2 FORMAT('OWAV=',F10.5,'A*=',F10.5,/'OB*=',F10.5,'C*=',F10.5,/
     1 'OCOSALF*=',F10.5,'COSBET*=',F10.5,/'OCOSGAM*=',F10.5)
      READ(1,3)JHMIN,JKMIN,JLMIN,JHMAX,JKMAX,JLMAX
    3 FORMAT(6I5)
      WRITE(3,4)JHMIN,JKMIN,JLMIN,JHMAX,JKMAX,JLMAX
    4 FORMAT('OJHMIN=',I5,'JKMIN=',I5,'JLMIN=',I5,/'OJHMAX=',I5,'JKMAX=*
     1 ',I5,'JLMAX=',I5)
      READ(1,910)R1,R2,R3,BBMO,BBF,OVB,BBNA
  910 FORMAT(7F10.5)
      READ(1,9101)XF,R4,ACELL,ICARD,ICC,T2CORR
 9101 FORMAT(3F10.5,2I2,F10.5)
      WRITE(3,911)R1,R2,R3,OVB,BBMO,BBF,XF,R4,ACELL,BBNA,T2CORR
  911 FORMAT(3X,                          'R1= ',F10.5,'R2= ',F10.5,/ 3X,
     * 'R3= ',F10.5,'OVERALL B = ',F10.5,/ 3X,
     *'SCATT FACTOR MO= ',F10.5,'SCATT FACTOR F= ',F10.5,
     * 'X-F STARTING= ',F10.5,'R4= ',F10.5,'ACELL= ',F10.5,/ 3X,
     * 'BBNA= ',F10.5,10X,'TWO THETA ZERO =',F12.5)
      IF(ICC.EQ.0)GO TO 9113
      WRITE(3,9112)ICARD
 9112 FORMAT(10X,'STRUCTURE FACTORS PRINTED,PUNCHED AFORE ',I2,' CYCLE')
 9113 CONTINUE
      RR1=R1
      RR2=R2
      RR3=R3
      RR4=R4
      XXF=XF
      OOVB=OVB
C SET UP ARRAY OF HKL AND 2 THETA POSSIBLE
  603 JH=JHMIN
      JK=JKMIN
      JL=JLMIN
   51 CONTINUE
   50 SINSQT=(WAV**2/4.)*(JH**2*AS**2+JK**2*BS**2+JL**2*CS**2
     1+2.*JK*JL*BS*CS*CAS+2.*JL*JH*CS*AS*CBS+2.*JH*JK*AS*BS*CGS)
      SINT=SQRT(SINSQT)
      IF(SINT-.600)40,505,505
   40 CONTINUE
      IF(SINT-.01)505,505,41
   41 CONTINUE
      T=FLOAT(JH+JK+JL)/2.
      IT=INT(T)
```

(Continued)

80/80 LISTING OF CARD INPUT.

```
      TT=FLOAT(IT)
      TTT=T-TT
      IF(TTT.GT..45.AND.TTT.LT..55)GO TO 505
      IF(JH.LT.JK)GO TO 505
       IF(JH.LT.JL) GO TO 505
      IF(JK.LT.JL)GO TO 505
   61 TH=ARSIN(SINT)*57.296
      TH2=2.*TH
       JJH(I)=JH
       JJK(I)=JK
       JJL(I)=JL
      TTH2(I)=TH2
      I=I+1
  505 CONTINUE
      IF(JL.GE.JLMAX)GO TO 501
  504 JL=JL+1
      GO TO 51
  501 IF(JK.GE.JKMAX)GO TO 502
      JL=JLMIN
      JK=JK+1
      GO TC 51
  502 IF(JH.GE.JHMAX)GO TO 503
      JL=JLMIN
      JK=JKMIN
      JH=JH+1
      GO TC 51
  503 CONTINUE
       IO=I-1
C READ PATTERN POINT,AND FIND CONTRIBUTING HKL 'S
  680 READ(1,610)THI,II,IBI,WI
        THI=THI+T2CORR
  610 FORMAT(F10.5,2I10,F10.5)
      WRITE(3,809)THI,II,IBI,WI
  809 FORMAT(10X,F10.5,2I10,F10.5)
      IF(II .EQ.0.)GO TO 670
      I1=II-IBI
       X=FLOAT(I1)
       KKK=II+2.*IBI
       Y=FLOAT(KKK)
        S1=SQRT(Y)
      J=1
       DO 620 I=1,IO
       JH=JJH(I)
        JK=JJK(I)
        JL=JJL(I)
       TH2=TTH2(I)
       P=1.55*WI
      Q=ABS(THI-TH2)
      IF(Q.GT.P)GO TO 630
C CODE IN MULTIPLICITIES
      IF(JH.NE.JK.AND.JK.NE.JL)M=48
      IF(JH.NE.JK.AND.JK.EQ.JL)M=24
      IF(JH.EQ.JK.AND.JK.NE.JL)M=24
      IF(JH.EQ.JK.AND.JK.EQ.JL)M=8
       IF(JL.EQ.0.AND.JH.NE.JK)M=24
      IF(JL.EQ.0.AND.JH.EQ.JK)M=12
      IF(JK.EQ.0.AND.JL.EQ.0)M=6
      QQ=Q**2
```

80/80 LISTING OF CARD INPUT.

```
      WW=WI**2
      PP=2.772588
      PPP=(PP*QQ)/WW
      AA=TH2/57.296
      AAA=SIN(AA)
      BB=AA/2.
      BBB=SIN(BB)
      S=EXP(-PPP)
      SS=(M*S)/(WI*AAA*BBB*10.)
        IIH(J)=JH
      IIK(J)=JK
       IIL(J)=JL
      SSS(J)=SS
       IM(J)=M
      J=J+1
  630 CONTINUE
  620 CONTINUE
      IIC=J-2
       IF(IIO.EQ.0)GO TO 690
       IF(IIO.LT.0)GO TO 695
      DO 650 K=1,IIO
      F=1.
      SIG=1.
      IQ=1
      LLH(LL)=IIH(K)
      LLK(LL)=IIK(K)
      LLL(LL)=IIL(K)
      FFF(LL)=F
      SSSIG(LL)=SIG
      SSSS(LL)=SSS(K)
      STHI(LL)=THI
      LL=LL+1
      WRITE(3,660)IIH(K),IIK(K),IIL(K),F,SIG,IQ,SSS(K),IM(K)
  640 FORMAT(1X,3I4,1X,2F9.3,I3,F9.4)
  660 FORMAT(10X,3I4,1X,2F9.3,I3,F9.3,I4)
  650 CONTINUE
  690 CONTINUE
       MM=J-1
      LLH(LL)=IIH(MM)
      LLK(LL)=IIK(MM)
      LLL(LL)=IIL(MM)
      FFF(LL)=X
      SSSIG(LL)=S1
      SSSS(LL)=SSS(MM)
      STHI(LL)=THI
      LL=LL+1
      LLX=LLX+1
      WRITE(3,660)IIH(MM),IIK(MM),IIL(MM),X,S1,IQ,SSS(MM),IM(MM)
  695 CONTINUE
      GO TO 680
  670 CONTINUE
      NOC=LLX-1
      NO=LL-1
      WRITE(3,950)NOO
  950 FORMAT(10X,'NO OF OBSERVATIONS IS ',I4)
      WRITE(3,951)NO
  951 FORMAT(10X,'NUMBER OF HKL CATA IS ',I4)
      ICY=1
```

(Continued)

```
                    80/80 LISTING OF CARD INPUT.
9122 READ(1,912)(KI(I),I=1,6),KI1
 912  FORMAT(7I1)
      INCY=0
      R1=RR1
      R2=RR2
      R3=RR3
      R4=RR4
      XF=XXF
      OVB=OOVB
 900  INCY=INCY+1
      IF(ICARD.NE.ICY)GO TO 9002
      WRITE(3,9001)INCY
9001  FORMAT(6X,'OBSERVED AND CALCULATED INTENSITIES BEFORE CYCLE ',I3)
      WRITE(3,901)
 901  FORMAT(15X,'H',5X,'K',5X,'L',5X,'OBS INTENS',5X,'CALC INTENS ',5X,
     *'DELTA   I')
9002  CONTINUE
      V1=0.
      V2=0.
      V3=0.
      V4=0.
      AA1=0.
      AA2=0.
       AA3=0.
       AA4=0.
      AA5=0.
      AA6=0.
       FF1=0.
      JJ=1
      DO 800 I=1,NO
      D=LLH(I)**2+LLK(I)**2+LLL(I)**2
      DD=D**2
      DDD=LLH(I)**4+LLK(I)**4+LLL(I)**4
      DDDD=(LLH(I)**2)*(LLK(I)**2)*(LLL(I)**2)
       D2=LLH(I)**8+LLK(I)**8+LLL(I)**8
      D3=D**4
      C=(2.*3.1416*XF)/ACELL
      CC=DSQRT(D)
      CCC=C*CC
      CCCC=DSIN(CCC)
       CCCD=DCOS(CCC)
      THETA=STHI(I)/(2.*57.296)
      STHA=SIN(THETA)
      STHAL=STHA/WAV
      SL2=STHAL**2
      SL22=-OVB*SL2*2.
      SL222=EXP(SL22)
      EEE=BBMO
      FFF1=6.*BBF
      B1= CCCC/CCC
      B2=   (DDD/DD-0.6D0)
      B3=((1.D0+(-45.D0+105.D0/CCC**2)/CCC**2)/CCC)*CCCC
      B4=(-(-10.D0+105.D0/CCC**2)/CCC**2)*CCCD
      B5=   ( DDDD/D**3 +(DDD/DD-0.6D0)/22.D0- 1.D0/105.D0 )
      B6=(-1.D0+(210.D0-(4725.D0-10395.D0/CCC**2)/CCC**2)/CCC**2)
     * *CCCC/CCC
      B7=-((21.D0-(1260.D0-10395.D0/CCC**2)/CCC**2)*CCCD)/CCC**2
      B8=D2/D3-28.D0*B5/5.D0-210.D0*B2/143.D0-1.D0/3.D0
```

80/80 LISTING OF CARD INPUT.

```
      B11=1.D0/(CCC**2)
      B9=CCCC/CCC*(1.D0+B11*(-630.D0+B11*(51975.D0+B11*(-945945.D0
     * +2027025.D0*B11))))
      B10=CCCD*B11*(36.D0+B11*(-6930.DC+B11*(270270.D0-2027025.D0*B11)))
      D1=B1+R2*B2*(B3+B4)+R3*B5*(B6+B7)+R4*B8*(B9+B10)
C THE FOLLOWING 14 CARDS MODIFY THE S.F.FOR THE NA CONTRIBUTION
      JX=LLH(I)
       JY=LLK(I)
        JZ=LLL(I)
      IF(JX.EQ.1.AND.JY.EQ.1)EEE1=0.00*BBNA
      IF(JX.EQ.2.AND.JY.EQ.0)EEE1=-2.0*BBNA
      IF(JX.EQ.2.AND.JY.EQ.1)EEE1=0.0*BBNA
      IF(JX.EQ.2.AND.JY.EQ.2.AND.JZ.EQ.0)EEE1=2.0 *BBNA
      IF(JX.EQ.3.AND.JY.EQ.1)EEE1=0.00*BBNA
      IF(JX.EQ.2.AND.JY.EQ.2.AND.JZ.EQ.2)EEE1=-2.*BBNA
      IF(JX.EQ.3.AND.JY.EQ.2)EEE1=0.0*BBNA
      IF(JX.EQ.4.AND.JY.EQ.0)EEE1=2.*BBNA
      IF(JX.EQ.3.AND.JY.EQ.3)EEE1=0.00*BBNA
      IF(JX.EQ.4.AND.JY.EQ.1)EEE1=0.00*BBNA
      IF(JX.EQ.4.AND.JY.EQ.2)EEE1=-2.0*BBNA
      E=(EEE+EEE1+FFF1*D1)
      F1=R1*SSSS(I)*(E**2)*SL222
      A1=SSSS(I)*(E**2)*SL222
      A2=2.*R1*SSSS(I)*E*FFF1*B2*(B3+B4)*SL222
      A3=2.*R1*SSSS(I)*E*FFF1*B5*(B6+B7)*SL222
       CCC1=2.*3.14159*XF*CC      /ACELL
      C1=(CCC1/XF)*COS(CCC1)
      C2=(CCC1/XF)*SIN(CCC1)
      C3=C1/CCC1-C2/(CCC1**2)
      C4=C1/(CCC1**3)-3.*C2/(CCC1**4)
      C5=C1/(CCC1**5)-5.*C2/(CCC1**6)
      C6=-C2/(CCC1**2)-2.*C1/(CCC1**3)
      C7=-C2/(CCC1**4)-4.*C1/(CCC1**5)
      C8=C1/(CCC1**7)-7.*C2/(CCC1**8)
      C9=-C2/(CCC1**6)-6.*C1/(CCC1**7)
      C10=C1/(CCC1**9)-9.*C2/(CCC1**10)
      C11=-C2/(CCC1**8)-8.*C1/(CCC1**9)
      A4=2.*R1*SSSS(I)*SL222*E*FFF1*(C3+R2*B2*(C3-45.*C4+105.*C5+10.*C6
     *      -105.*C7)+R3*B5*(-C3+210.*C4-4725.*C5+10395.*C8-21.*C6
     *  +1260.*C7-10395.*C9) +R4*B8*(C3-630.*C4+51975.*C5-945945.*C8
     $ +2027025.*C10+36.*C6-6930.*C7+270270.*C9-2027025.*C11))
      A5=2.*R1*SSSS(I)*E*FFF1*B8*(B9+B10)*SL222
       A6=-R1*SSSS(I)*E**2*2.*SL2*SL222
      G=FFF(I)
      IF(G.GT.1.001 .OR.G.LT.0.999 )GO TO 801
      AA1=AA1+A1
      AA2=AA2+A2
      AA3=AA3+A3
       AA4=AA4+A4
      AA5=AA5+A5
      AA6=AA6+A6
      FF1=FF1+F1
       IF(ICARD.NE.ICY)GO TO 8011
      WRITE(3,415)LLH(I),LLK(I),LLL(I)
  415 FORMAT(10X,'SUPERPOSED REFLEXION',3I4)
 8011 CONTINUE
      GO TO 800
  801 AA1=AA1+A1
```

(Continued)

80/80 LISTING OF CARD INPUT.

```
      AA2=AA2+A2
      AA3=AA3+A3
      AA4=AA4+A4
      AA5=AA5+A5
      AA6=AA6+A6
      FF1=FF1+F1
      FF11=FFF(I)-FF1
      IF(ICARD.NE.ICY)GO TO 9022
      WRITE(3,902)LLH(I),LLK(I),LLL(I),FFF(I),FF1,FF11
  902 FORMAT(10X,3I4,3F12.5)
 9022 CONTINUE
      IF(ICARD.NE.ICY)GO TO 940
  420 FORMAT(2F10.3)
  940 V1=V1+FFF(I)**2/SSSIG(I)**2
      V2=V2+FF11**2/SSSIG(I)**2
      V3=V3+ABS(FF11)
      V4=V4+ABS(FFF(I))
      A(JJ,1)=AA1
      A(JJ,2)=AA2
      A(JJ,3)=AA3
      A(JJ,4)=AA4
      A(JJ,5)=AA5
      A(JJ,6)=AA6
      FF(JJ)=FF11
      PII(JJ)=SSSIG(I)**2
      JJ=JJ+1
      AA1=0.
      AA2=0.
      AA3=0.
      AA4=0.
      AA5=0.
      AA6=0.
      FF1=0.
  800 CONTINUE
  941 FORMAT(10X,F10.5)
C WE HAVE STORED A,F,AND WEIGHT MATRICES ,A HAVING ALL DERIVATIVES
C WE NOW CALCULATE A'P  WITH NON-VARIABLES EXCLUDED
      DO 913 I=1,NOO
      K=1
      DO914 J=1,6
      IF(KI(J).EQ.0) GO TO 914
      AB(K,I)=A(I,J)/PII(I)
      K=K+1
  914 CONTINUE
  913 CONTINUE
      NN=K-1
C NEXT MULTIPLY A'P BY A AND CALL RESULT AC(3,3)
      DO 915 K=1,6
       DO915 L=1,6
      AC(K,L)=0.
  915 CONTINUE
      DO 916 I=1,NOO
      DO 917 K=1,NN
      KK=1
      DO 917  J=1,6
      IF(KI(J).EQ.0)GO TO 917
      AC(K,KK)=AB(K,I)*A(I,J)+AC(K,KK)
      KK=KK+1
```

80/80 LISTING OF CARD INPUT.

```
  917 CONTINUE
  916 CONTINUE
CWE NOW HAVE A'PA    WE FORM A'PF WHERE F(N,1) IS NOT WEIGHTED
C A'P IS AB(3,I) BY F(N,1)  THIS IS A 3X1 MATRIX  CALLED AD
      DO 812 J=1,NN
       AC(J)=0.
  812 CONTINUE
      DO 8121 I=1,NOO
      DO 8121 J=1,NN
      AD(J)=AD(J)+AB(J,I)*FF(I)
 8121 CONTINUE
C WE NOW FORM INVERSE OF A'PA ,I.E.OF AC(NN,NN)
       CALL SID(AC,-1,6,NN,NN,DET)
C WE MULTIPLY INVERSE,STILL CALLED AC BY AD(NN,1)
C TO GET AF(3,1) WHICH IS ANSWER FOR SHIFTS
      DO 821 J=1,NN
      AF(J)=0.
  821 CONTINUE
      DO 820 J=1,NN
      DO 820 I=1,NN
      AF(J)=AF(J)+AC(J,I)*AD(I)
  820 CONTINUE
      DO 943 J=1,NN
      WRITE(3,941)AF(J)
  943 CONTINUE
C  FOR ERRORS MULT A(NOO,NN)BY AF(NN) GIVING AG(NOO)
      DO 8301 J=1,NOO
      AG(J)=0.
 8301 CONTINUE
      DO 830 I=1,NOO
      KK=1
      DO 831 J=1,6
      IF(KI(J).EQ.0)GO TO 831
      AG(I)=AG(I)+A(I,J)*AF(KK)
      KK=KK+1
  831 CONTINUE
  830 CONTINUE
C SUBTRACT AG FROM F  CALL THIS AH,AND WEIGHT IT
      DO 832 I=1,NOO
      AH(I)=(FF(I)-AG(I))
  832 CONTINUE
C MULT BY AH AGAIN ,CALL IT AK
       AK=0.
      DO 833 I=1,NOO
      AK=AK+AH(I)**2/PII(I)
  833 CONTINUE
      SIGSQ=AK/(NOO-NN)
C CALCULATE MOMENT    MATRIX OF PARAMETERS
C CALL IT AL
      DO 840 I=1,NN
      DO 840 J=1,NN
      AL(I,J)=SIGSQ*AC(I,J)
  840   CONTINUF
      DO 8401 I=1,NN
      SG(I)=SQRT(AL(I,I))
 8401 CONTINUE
C THESE ARE ERRORS ON SHIFTS
C CALC CORRELATION COEFFICIENTS
```

(Continued)

80/80 LISTING OF CARD INPUT.

```
      DO 8402 I=1,NN
      DO 8402 J=1,NN
      COR(I,J)=AL(I,J)/(S G(I)*S G(J))
 8402 CONTINUE
C R-FACTORS
      V6=V2/V1
      RW=SQRT(V6)
      R=V3/V4
      WRITE(3,903)RW,R
  903 FORMAT(6X,'WEIGHTED R IS ',F10.5,'UNWEIGHTED R IS ',F 10.5)
      WRITE(3,8403)
 8403 FORMAT(10X,'PARAMETER VARIED',5X,'OLD',5X,'SHIFT',5X,
     * 'NEW',5X,'SIGMA')
      KK=1
      DO 920 J=1,6
      IF(KI(J).EQ.0)GO TO 920
      GO TO (921,922,923,924,925,961),J
  921 R11=R1
      R1=R1+AF(KK)
      WRITE(3,926)R11,AF(KK),R1,S G(KK)
  926 FORMAT(10X,'R1',4F10.5)
      KK=KK+1
      GO TO 920
  924 XF1=XF
            XF=XF+AF(KK)
      WRITE(3,927) XF1,AF(KK),XF,S G(KK)
  927 FORMAT(10X,'X-F DISTANCE' ,4F10.5)
      KK=KK+1
      GO TO 920
  922 R22=R2
      R2=R2+AF(KK)
      WRITE(3,928)R22,AF(KK),R2,S G(KK)
  928 FORMAT(10X,'R2',4F10.5)
      KK=KK+1
      GO TO 920
  923 R33=R3
      R3=R3+AF(KK)
      WRITE(3,929)R33,AF(KK),R3,S G(KK)
  929 FORMAT(10X,'R3',4F10.5)
      KK=KK+1
      GO TO 920
  925 R44=R4
      R4=R4+AF(KK)
      WRITE(3,930)R44,AF(KK),R4,S G(KK)
  930 FORMAT(10X,'R4',4F10.5)
      KK=KK+1
      GO TO 920
  961 OVB1=OVB
      OVB=OVB+AF(KK)
       WRITE(3,9611)OVB1,AF(KK),OVB,SG(KK)
 9611 FORMAT(10X,'OVERALLB ', 4F10.5)
      KK=KK+1
      GO TO 920
  920 CONTINUE
      WRITE(3,931)
  931 FORMAT(10X,'CORRELATION COEFF. 11,12,...21,22,...ETC')
       DO 932 I=1,NN
      DO 932 J=1,NN
```

```
                                 80/80 LISTING OF CARD INPUT.
      WRITE(3,933)COR(I,J)
  933 FORMAT(3(10X,5F10.5/))
  932 CONTINUE
      WRITE(3,934)SIGSQ
  934 FORMAT(6X,'CHISQUARED=   ',F10.5)
         ICY=ICY+1
      IF(INCY.LT.ICC)GO TO 900
      IF(INCY.EQ.ICC)GO TO 9122
  936 CONTINUE
      STOP
      END
1.909     .113636   .113636   .113636
   0     0     0     4     4     4
15.        6.        0.        .85       .68       17.       .0
2.8       0.        8.800     1520.265
16.45          232       215.62
16.50          234       215.62
16.55          245       215.62
16.6           258       215.62
16.65          262       215.62
16.70          292       215.62
16.75          292       215.62
C MORE PATTERN DATA CARDS ...

111101
```

For the MX_6 case, remove BBNA from read statement with format 910, and the 14 cards modifying the S.F. formula.

(Continued)

```
      SUBROUTINE SID(SIDW,LSID,NROW,MSID,NSID,SIDET)
C     SUBROUTINE TO SOLVE INVERSE AND DETERMINANT OF M*M MATRIX
C     WITH SOLUTION MATRIX M*(N-M) IF REQD. USES GAUSS-JORDAN METHOD.
C     EG. TO SOLVE ((A))(X)=(B), A=3 X 3, B=3 X 1,
C     THEN STORE (A1),(A2),(A3),(B) IN NEIGHBOURING COLS.
C     RESULT     (R1),(R2),(R3),(X).
C     IF LSID=+1 THEN ONLY DET AND (X) SENSIBLE.
C     IF LSID=0 THEN ONLY DET SENSIBLE.
C     IF LSID=-1 THEN INVERSE WILL ALSO BE IN ((R))
C     NOTE THAT THE ORIGINAL MATRIX IS DESTROYED
C     AS THE RESULT OCCUPIES LOCATIONS PREVIOUSLY ((A)).
C     'NROW' MUST BE THE NO. OF ROWS IN THE DIMENSIONED VARIABLE FOR
C     WHICH 'SIDW' IS A SUBSTITUTE.
C     IF MSID IS MORE THAN 50, CHANGE IFSID(50), ETC.
C     TO THE NUMBER OF ROWS REQD.
C     DIMENSION IFSID(50),ILSID(50),IGSID(50), SIDW(50,50)
      DIMENSION IFSID(50),ILSID(50),IGSID(50),SIDW(2)
      SIDET=1.
      ISID1=MSID-1
      ISIDR=NSID-MSID
      IF(ISIDR)8802,8851,8851
 8851 DO 8822 KSID=1,MSID
      ILSID(KSID)=0
 8822 IGSID(KSID)=KSID
      DO 8803 KSID=1,MSID
      IF(LSID)8852,8853,8853
 8852 KSID1=1
      GO TO 8854
 8853 KSID1=KSID+1
 8854 RSID=0.
      DO 8804 ISID=1,MSID
      IF(ILSID(ISID))8804,8805,8804
C8805 WSID=SIDW(ISID,KSID)
 8805 JRSID=ISID+(KSID-1)*NROW
      WSID=SIDW(JRSID)
      XSID=WSID
      IF(XSID)8806,8807,8807
 8806 XSID=-XSID
 8807 IF(XSID-RSID)8804,8808,8808
 8808 RSID=XSID
      PSID=WSID
      KFSID=ISID
 8804 CONTINUE
      IFSID(KSID)=KFSID
      ILSID(KFSID)=KFSID
      SIDET=SIDET*PSID
      IF(SIDET)8810,8802,8810
 8810 DO 8815 ISID=1,MSID
```

```
      JRSID=ISID+(KSID-1)*NROW
      IF(ISID-KFSID)8855,8856,8855
C8856 SIDW(ISID,KSID)=1./PSID
 8856 SIDW(JRSID)=1./PSID
      GO TO 8815
C8855 SIDW(ISID,KSID)=-SIDW(ISID,KSID)/PSID
 8855 SIDW(JRSID)=-SIDW(JRSID)/PSID
 8815 CONTINUE
      IF(KSID1-NSID)8870,8870,8803
 8870 DO8825JSID=KSID1,NSID
      IF(JSID-KSID)8858,8825,8858
C8858 WSID=SIDW(KFSID,JSID)
 8858 JRSID=KFSID+(JSID-1)*NROW
      WSID=SIDW(JRSID)
      IF(WSID)8821,8825,8821
 8821 DO 8820 ISID=1,MSID
      JRSID=ISID+(JSID-1)*NROW
      IF(ISID-KFSID)8823,8824,8823
C8824 SIDW(ISID,JSID)=WSID/PSID
 8824 SIDW(JRSID)=WSID/PSID
      GO TO 8820
C8823 SIDW(ISID,JSID)=SIDW(ISID,JSID)+WSID*SIDW(ISID,KSID)
 8823 JDSID=ISID+(KSID-1)*NROW
      SIDW(JRSID)=SIDW(JRSID)+WSID*SIDW(JDSID)
 8820 CONTINUE
 8825 CONTINUE
 8803 CONTINUE
      DO 8840 KSID=1,ISID1
      KFSID=IFSID(KSID)
      KLSID=ILSID(KFSID)
      KGSID=IGSID(KSID)
      IF(KFSID-KGSID)8841,8840,8841
 8841 IF(LSID)8842,8843,8844
 8844 IF(ISIDR)8802,8843,8846
 8842 DO 8861 ISID=1,MSID
      JRSID=ISID+(KFSID-1)*NROW
      JDSID=ISID+(KGSID-1)*NROW
C     RSID=SIDW(ISID,KFSID)
      RSID=SIDW(JRSID)
C     WSID=SIDW(ISID,KGSID)
      WSID=SIDW(JDSID)
C     SIDW(ISID,KFSID)=WSID
      SIDW(JRSID)=WSID
C8861 SIDW(ISID,KGSID)=RSID
 8861 SIDW(JDSID)=RSID
 8846 DO 8860 JSID=1,NSID
      JRSID=KSID+(JSID-1)*NROW
      JDSID=KLSID+(JSID-1)*NROW
C     RSID=SIDW(KSID,JSID)
      RSID=SIDW(JRSID)
C     WSID=SIDW(KLSID,JSID)
      WSID=SIDW(JDSID)
C     SIDW(KSID,JSID)=WSID
      SIDW(JRSID)=WSID
C8860 SIDW(KLSID,JSID)=RSID
 8860 SIDW(JDSID)=RSID
 8843 ILSID(KFSID)=KSID
      ILSID(KGSID)=KLSID
      IGSID(KLSID)=IGSID(KSID)
      IGSID(KSID)=KFSID
      SIDET=-SIDET
 8840 CONTINUE
 8802 RETURN
C     END OF SUBROUTINE
      END
```