Technical Report
RAL-TR-2000-003

Selection, Rejection and Optimisation of Pyrolytic Graphite (PG) Crystal Analysers for Use on the New IRIS Graphite Analyser Bank

P J Marshall D S Sivia M A Adams and M T F Telling

27th January 2000

COUNCIL FOR THE CENTRAL LABORATORY OF THE RESEARCH COUNCILS

The Central Laboratory of the Research Councils
Library and Information Services
Rutherford Appleton Laboratory
Chilton
Didcot
Oxfordshire
OX11 0QX
Tel: 01235 445384 Fax: 01235 446403
E-mail library@rl.ac.uk

ISSN 1358-6254

Selection, Rejection and Optimisation of Pyrolytic Graphite (PG) Crystal Analysers for use on the new IRIS graphite analyser bank.

P.J.Marshall, D.S.Sivia, M.A.Adams and M.T.F.Telling

ISIS Facility,
Rutherford Appleton Laboratory,
Chilton,
Didcot,
Oxfordshire,
OX11 OQX

January 2000

ABSTRACT

This report discusses design problems incurred by equipping the IRIS high-resolution inelastic spectrometer at the ISIS pulsed neutron source, UK with a new 4212 piece pyrolytic graphite crystal analyser array. Of the 4212 graphite pieces required, approximately 2500 will be newly purchased PG crystals with the remainder comprising of the currently installed graphite analysers. The quality of the new analyser pieces, with respect to manufacturing specifications, is assessed, as is the optimum arrangement of new PG pieces amongst old to circumvent degradation of the spectrometer's current angular resolution. Techniques employed to achieve these criteria include accurate calliper measurements, FORTRAN programming and statistical analysis.

ACKNOWLEDGEMENTS

The authors wish to thank Dr. Colin Carlile (Instiut Laue Langevin, Grenoble, France) and Dr. John Tomkinson (ISIS Facility, Rutherford Appleton Laboratory) for fruitful discussion that enabled completion of this work. P.J.Marshall is grateful to the Nuffield Foundation for funding his time at the Rutherford Appleton Laboratory.

I. INTRODUCTION

IRIS is a time of flight (t.o.f) inverted-geometry crystal analyser spectrometer designed for high-resolution quasi-elastic and low-energy inelastic spectroscopy (Figure 1). It employs two large analysers (pyrolytic graphite and muscovite mica) oriented close to back-scattering geometry [1]. Using non-back scattering geometry avoids the loss in intensity caused by a beam modulation chopper when exact back scattering is employed.

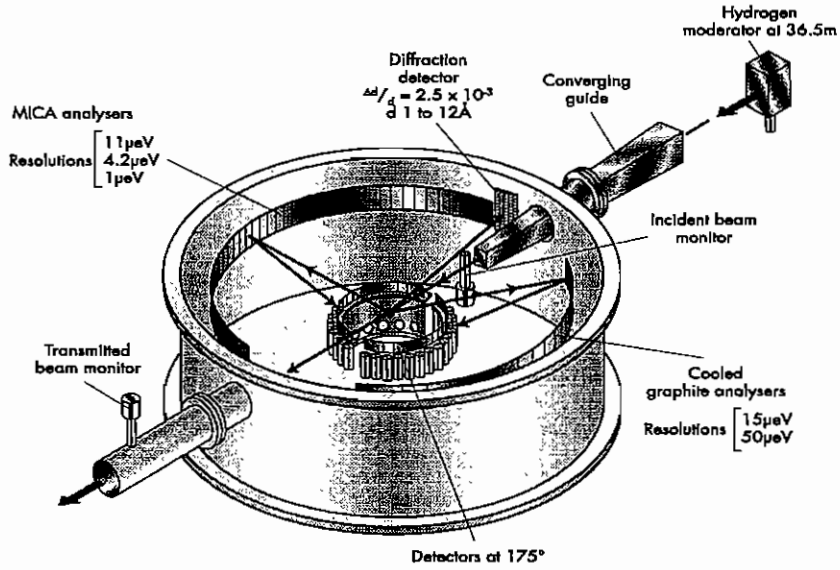

Figure 1 *The IRIS high-resolution spectrometer on the ISIS pulsed neutron source.*

The pyrolytic graphite (PG) analyser is set 0.85 meters from the sample position in the horizontal scattering plane and covers scattering angles from 15° to 165°. The present analyser consists of 1350 (6 rows by 225 columns) 2mm thick, cooled (~ 25K [1]) pyrolytic graphite pieces ($10mm^2$) with a mosaic spread (η) of 0.8° mounted on a spherically machined aluminium backing plate. The analysed beam is back scattered through 175°, slightly below the horizontal scattering plane, and

detected using a multi-detector composed of 51 scintillator detectors located approximately 0.6 meters from the analyser. However, while the use of pyrolytic graphite affords the possibility of two analysing reflections, 002 and 004, with analysing energies of 1.82meV and 7.28meV and resolution of 15μeV and 50μeV respectively, the analyser itself intercepts only a small percentage (~ 1.5%) [1] of the total scattered beam. In theory, the count rate of the IRIS spectrometer may be significantly improved by simply increasing the area of the analyser.

Considering the geometric and physical constraints (shielding etc.) of the instrument, it is possible to achieve a three fold increase in the area of the graphite analyser. Consequently, the current pyrolytic graphite analyser bank on IRIS is to be upgraded from an array 6 rows by 225 columns (1350 crystals) to one comprised of 4212 graphite pieces (18 rows, 234 columns). In addition, the graphite is to be cooled close to liquid helium temperature to further reduce thermal diffuse scattering [1] and thereby significantly improve the sensitivity of the spectrometer. For an instrument such as IRIS, with its analyser out of exact back-scattering geometry, optical aberration and variation in the time-of-flight of the analysed neutrons is introduced as one moves out from the horizontal scattering plane. To minimise such effects, the cross section profile of the analyser has been redesigned (see [2] or http://www-dienst.rl.ac.uk/library/1999/tr/raltr-1999044.pdf).

However, to actualise the specified upgrade, over two and a half thousand additional pyrolytic graphite pieces are required. In this report, we address the design problems incurred by equipping the IRIS spectrometer with a new array of pyrolytic graphite crystal analysers. The quality of the new PG crystals, with respect to manufacturing specifications, is assessed, as is the optimum arrangement of new PG pieces among old to circumvent degradation of the spectrometer's current angular resolution. Techniques employed to achieve these criteria include accurate calliper measurements, FORTRAN programming and statistical analysis.

III. PYROLYTIC GRAPHITE

In brief, pyrolytic graphite consists of a planar distribution of hexagonally arranged carbon atoms, the d-spacing between planes being approximately 3.35 Å. Each crystal piece is comprised of micro-crystallites ordered in a similar fashion to that of a Roman mosaic. Deviations from perfect ordering are common and give rise to a crystals mosaicity, or mosaic spread; mosaicity being assumed normally distributed and given in angular units. Indeed, neutron beam diffusion is more pronounced from crystals exhibiting a high mosaicity.

Of the 4212 pyrolytic graphite crystals required for the new analyser array approximately 2500 will be newly purchased PG pieces with the remainder comprising of currently installed crystal analysers. At present, the graphite pieces mounted on the current analyser bank are 2mm thick. These pieces will therefore be cleaved to produce 2700 1mm thick crystals. The cleaved crystals will be graded for thickness and only those whose width falls within the manufacturing specification for the new graphite pieces (1mm ± 0.1mm, see below) will be included for use on the new analyser. For clarity, new crystals will from now on be referred to as Type 1 while those presently installed will be designated Type 2. The manufacturing specifications for the newly purchased crystals were as follows:

- Individual pieces of graphite should be flat, square-faced, 10mm ± 0.1mm by 10mm ± 0.1mm of thickness 1mm ± 0.1mm. The front and back ($10mm^2$) surfaces of any individual piece should be parallel to within 0.5°. All face corners should be square to within 1°

- Each piece should contain 002 crystallographic planes. The normal to the 002 crystallographic plane should be within 0.5° of the front face of each piece. Neither the surfaces nor the edges should be friable when handled

- The distribution of mosaic blocks in each piece should be gaussian in shape with a FWHM $\eta = 0.7° \pm 0.2°$ (measured using X-rays, see Figure 2). No piece should have a $\eta > 1.0°$. The mean value of η over all pieces should be less 0.85°. The distribution of η over all pieces should not contain multiple peaks.

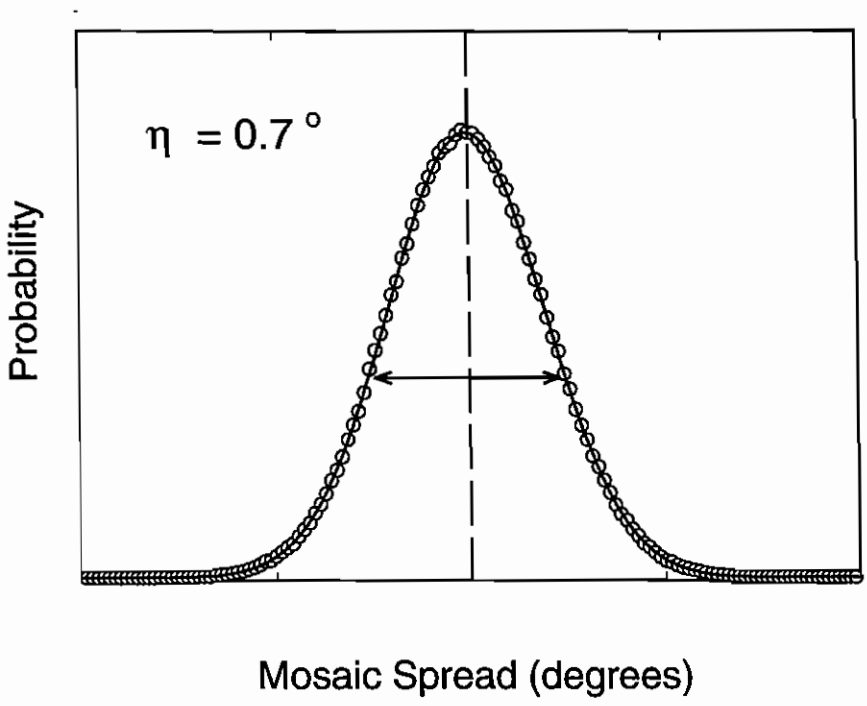

Figure 2 *The distribution of mosaic blocks in each Type 1 crystal analyser should be gaussian with a FWHM* $\eta = 0.7^\circ (\pm 0.2^\circ)$

IV. INSPECTION

While it was not possible to verify each specification it was important, as will now be discussed, to a) gauge the overall quality and dimensions of the new graphite pieces and b) ascertain the distribution of mosaic spread values over all Type 1 crystals.

4.1. OVERALL CRYSTAL QUALITY

The method of mounting pyrolytic graphite pieces on the presently installed analyser backing plate is illustrated below in Figure 3; each crystal is held firmly in position using four brass pins, the pitch distance between pin centers being 10mm ± 0.2mm.

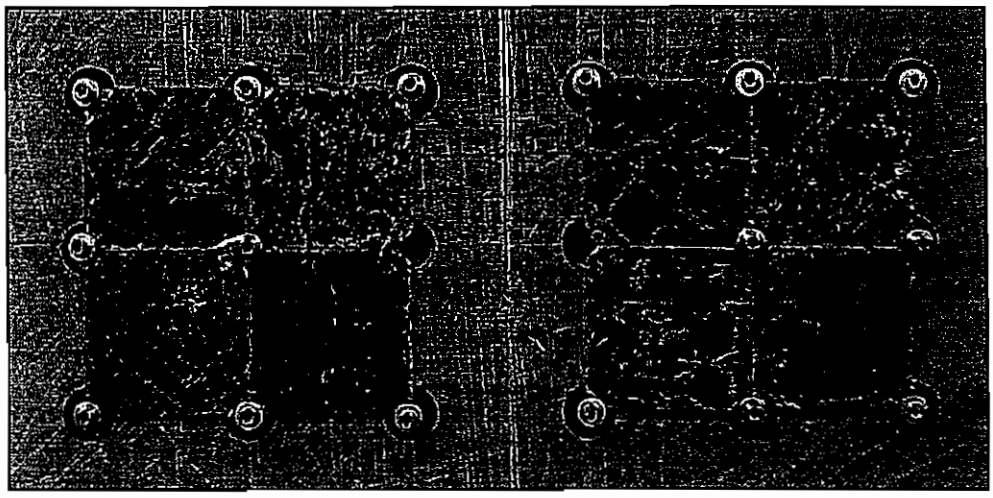

Figure 3 *The method of 'pinning' pyrolytic graphite crystals to the presently installed pyrolytic graphite analyser backing plate.*

Since this design method had proven effective, it was also to be used to mount the new 4212 crystal piece array. However, it was important to gauge dimensional tolerances for the front face edge lengths of the Type 1 crystals since the lower tolerance limit ascertained would govern an absolute pitch distance to be machined between pin centers so as not to reject crystal pieces on the grounds of being 'undersize'. Type 1 crystals with one, or both, front face edge lengths greater than the determined pitch distance were to be simply 'machined' to fit.

Upon arrival, a visual inspection of all Type 1 crystals was conducted, each crystal being carefully examined for physical damage and prominent surface imperfection. In addition, approximate dimensions were determined by comparing each crystal with a) other crystals and b) with the exact 1.1mm x 1.1mm square holes in the trays in which the crystals were transported and stored. Dimensions were estimated to within ±0.5mm, rather than the ±0.1mm error defined in the specification. Crystals were rejected on the grounds of being the wrong size, 'non-square', split or having uneven edges. Thirty-three crystals (~ 1.2%) were discounted in this manner. Examples of rejected crystals are shown below in Figure 4.

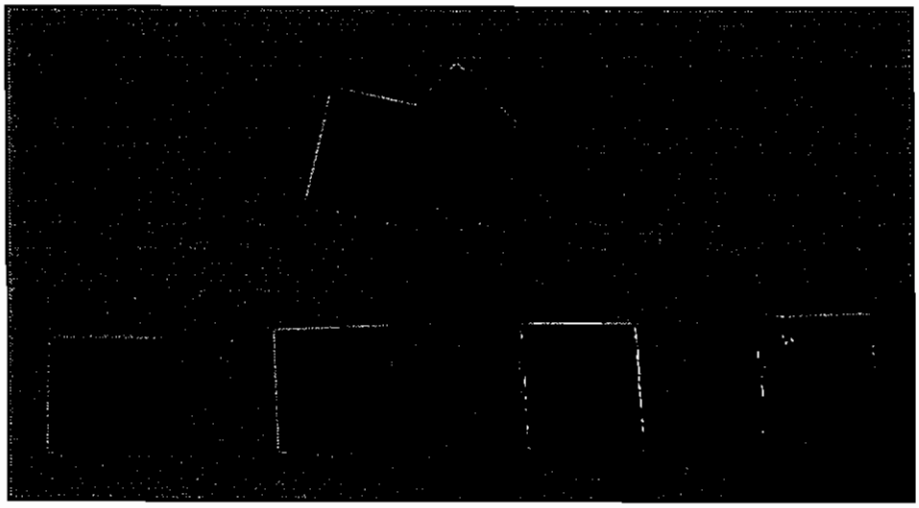

Figure 4 *Examples of rejected Type 1 crystals: Top: split. Bottom (left to right): too small, usable crystal (for comparison), 'non-square' and chipped.*

To determine dimensional tolerances, the Type 1 crystals were measured on two adjacent front face edges using digital callipers to an accuracy of ± 0.01 mm. Not all the crystals were measured. Instead, a small sample was tested with the resulting measurements being used to predict the distribution of front face edge lengths for the entire Type 1 crystal population.

In brief, the graphite pieces to be measured were selected at random using a simple FORTRAN program calling a random number generator subroutine; the thirty-three crystal pieces rejected during the visual inspection being excluded from this survey. The measured edge lengths were fed into FORTRAN program, 'Reliability' (Appendix 1), where upon Bayesian probability theory [3] was applied to the measured data set to determine the said edge length probability distribution. The probability distribution was updated every ten measurements and measuring stopped when the shortest 95 % confidence interval lay within 0.1 mm of the mean (as opposed to within 0.1 mm of 10.00 mm). One hundred crystals (4% of the entire Type 1 crystal population) were measured giving a total of two hundred measurements. The final distribution of crystal front face edge lengths for the entire Type 1 crystal population is shown below in Figure 5.

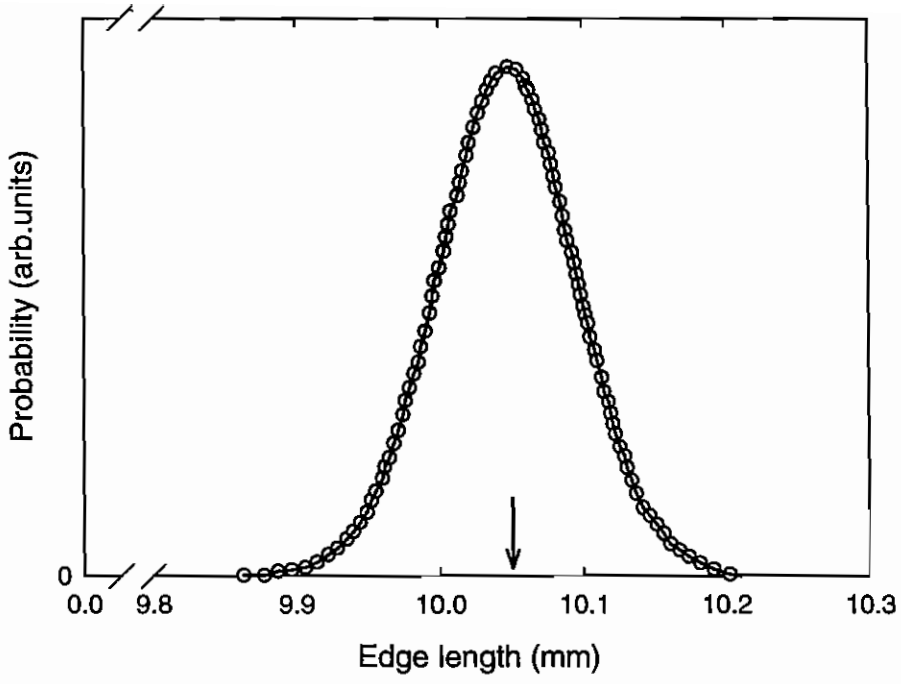

Figure 5 *The distribution of Type 1 pyrolytic graphite crystal front face edge length as determined using Bayesian probability theory[3]*

It is clear that the mean front face edge length is approximately 10.05 mm, indicating a systematic manufacturing error of 0.05 mm. However, the results also highlight that no Type 1 crystals longer than 10.2mm, or shorter than 9.85mm, are expected. Since the pitch distance between pin centres has a tolerance of 0.2mm all Type 1 are expected to fit and no further graphite pieces should need be rejected due to being 'undersize'.

4.2. MOSAICITY

Ideally, to maintain the current momentum transfer, or Q, resolution of the spectrometer, the mosaicity of the additional graphite crystals should match that of the presently installed PG pieces. However, while the mosaicity of the crystal pieces used on the current analyser is well defined, the new PG crystals purchased to complete the new array show a pronounced variation in their mosaic spread values.

The presently installed Type 2 graphite crystals, manufactured by Graphite Loraine, had their mosaic spread values determined using neutron techniques. Neutron methods provide a precise measurement of mosaicty since they are bulk measurement techniques. Consequently, the actual Type 2 crystals used on the presently installed analyser were selected such that $\eta = 0.8°$ for all crystals.

In contrast, however, the newly purchased Type 1 crystals, supplied by Atomgraph Corp. USA, had their mosaic spread values determined using X-ray diffraction. Being a less penetrating probe, X-rays only measure mosaicity at a crystal's surface. Consequently, mosaic spread valves for both the front and back faces of the Type 1 crystals were determined with the bulk mosaicity for each Type 1 graphite piece being assumed the average of these two values. In addition, and as a consequence of the surface measurement, X-rays measurements systematically return a mosaic-spread value approximately 0.1° less than that determined by neutrons. For ease of manipulation, therefore, all mosaic-spread values were deemed measured using X-rays. Consequently, all Type 2 crystals, with a known 'neutron' mosaicity of 0.8 °, were assumed to have an 'X-ray' mosaicity of 0.7 °.

To determine the average mosaicity of the entire Type 1 crystal population statistical analysis of the X-ray mosaic spread data supplied with each graphite crystal was performed. Using the front and back face X-ray mosaic spread values, FORTRAN program 'Average' (Appendix II.) was used to compute an 'average' mosaic spread value for each crystal along with a population mean and standard deviation. The frequency of each mosaic-spread value was also calculated to verify that the distribution of mosaic spreads was single peaked. Selected information about the mosaicity of the Type 1 crystals, determined using program 'Average' is given in Table 1 with the frequency distribution being illustrated in Figure 6.

Average of front faces :	0.7243	± 0.0948
Average of back faces :	0.7239	± 0.0937
The mean of two averages :	0.7241	± 0.0812

Table 1 *Average mosaic spread values determined using statistical analysis program 'Average'*

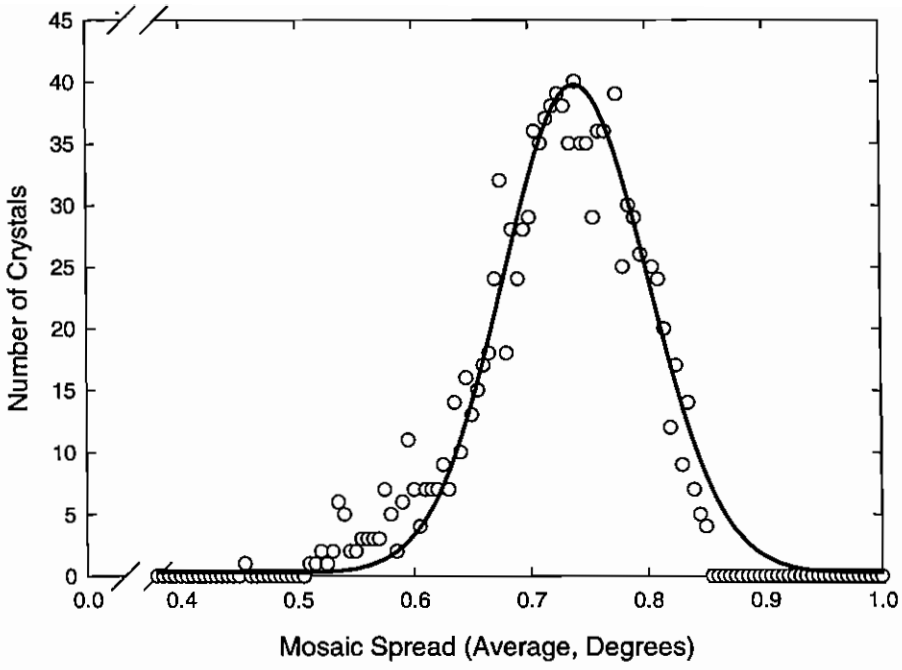

Figure 6 *Frequency distribution of the average mosaicity values determined for each Type 1 pyrolytic graphite crystals using FORTRAN program 'Average'. The solid line is the result of modelling the distribution using the gaussian form.*

The results show a single peaked distribution of mosaic spread values over all Type 1 graphite crystals with a mean value of η less than 0.85, as specified.

VI. Optimum Arrangement Of The Pyrolytic Graphite Crystals

The optimum arrangement of Type 1 and Type 2 crystal pieces on the new analyser backing plate is one which results in little to no variation in Q-resolution across the face of the analyser as seen from one detector to next. As previously mentioned, this is most readily achieved if all Type 1 and Type 2 crystals have equal mosaic spread values since it is the mosaicity of a crystal analyser that contributes the most to the angular resolution of a back-scattering spectrometer such as IRIS [4]. However, this is not the case. Before the pyrolytic graphite pieces could be mounted on the analyser backing plate, therefore, it was important to ascertain an arrangement of the Type 1 and Type 2 crystals that resulted in as little variation in

localised average mosaic spread as possible along the length of the analyser. On scales of the order of a few centimetres variation in mosaic spread is unavoidable. However, variations can be minimised on larger length scales. It was decided that variations in mosaic spread would be more obvious should the arrangement have structure built into it. For example, should each row of crystals have the same average mosaic spread, then the columns' average spread may vary greatly. Instead, a random pattern of crystals was taken as a starting point after which a Monte Carlo algorithm approach was assumed.

The optimum arrangement of Type 1 and Type 2 graphite crystals for the new analyser bank was determined using the FORTRAN program 'Gridassign' (Appendix III), the optimisation process being presented in detail in Figure 7. The program was designed to run until interrupted to allow for as many permutations, or different Type 1 crystal configurations, as possible; the 'best' and 'worst' crystal configurations being written to files 'Finalbest.dat' and 'Finalworst.dat' respectively. The simplest method of comparing and contrasting the two extremes was to produce colour maps to detail the variation of mosaicity across the face of the analyser bank. This was achieved using the plotting package PG PLOT and the results are presented in Figure 8. The results shown are the 'best' and 'worst' graphite crystal configurations generated after 10^7 program iterations.

Despite colour representation, however, it is difficult to distinguish the best arrangement from the worst. To further differentiate between the two results, therefore, a more mathematical approach was assumed. Program 'Test' (Appendix IV) was written to effectively divide the simulated crystal analyser array into equal vertical segments, the number of divisions (2^N where N= 1 to 6, see Figure 9) representing the number of times the array was divided into two. Average mosaic spread values for each division were then determined, subtracted from the overall analyser bank mean (i.e. mean of all graphite crystals, 0.71) and summed; the results being plotted against number of divisions in Figure 9. One can see that the so-called 'best' arrangement is indeed better since the smaller the magnitude of a column, the less the variation in mosaicity. Indeed, the largest column (2^6 or 64 divisions) is an approximate representation of the variation in average mosaic spread as would be 'seen' by each of the 51 scintillator detectors.

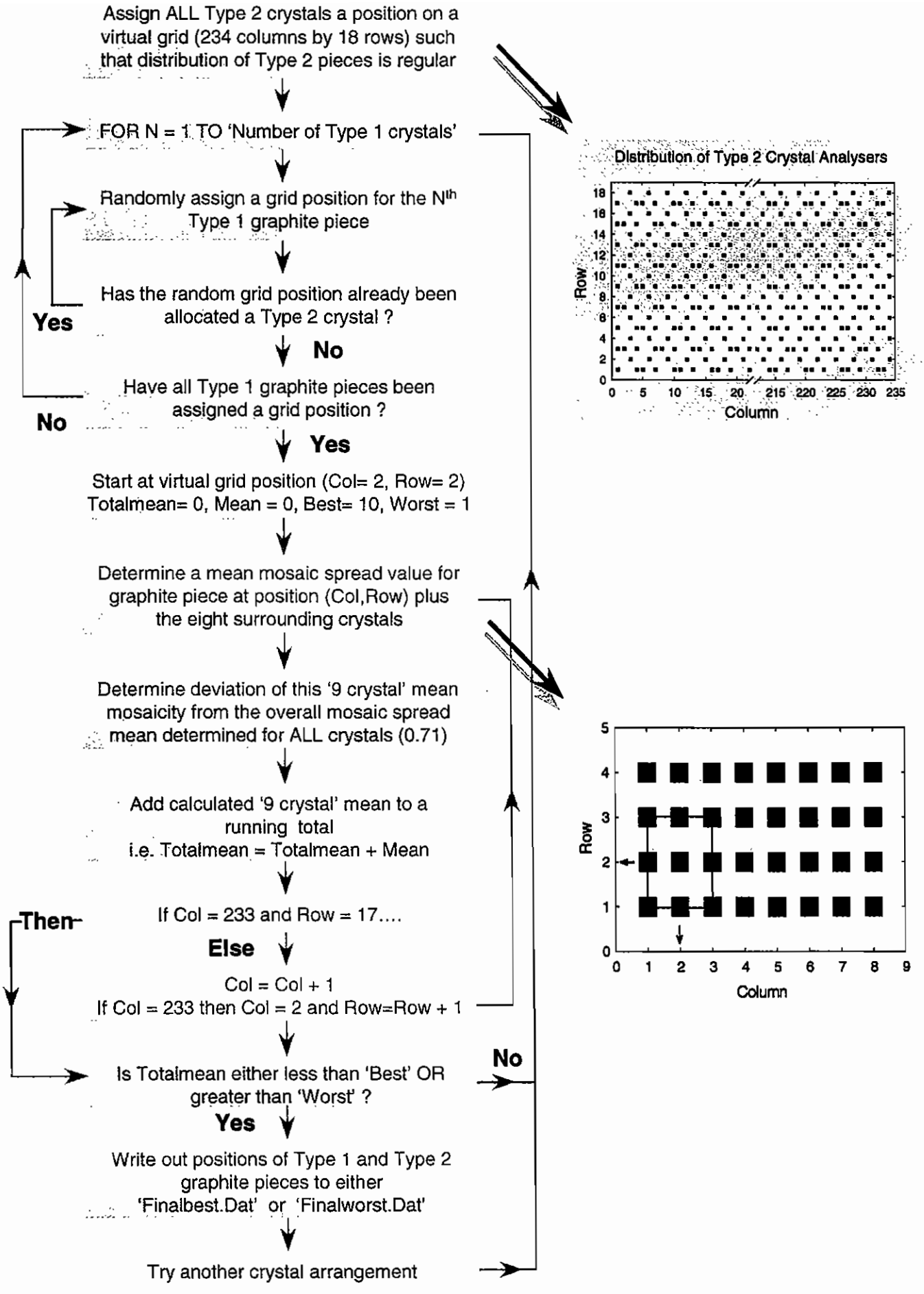

Figure 7 *The optimisation process – flow diagram of FORTRAN program 'Gridassign' (Appendix III)*

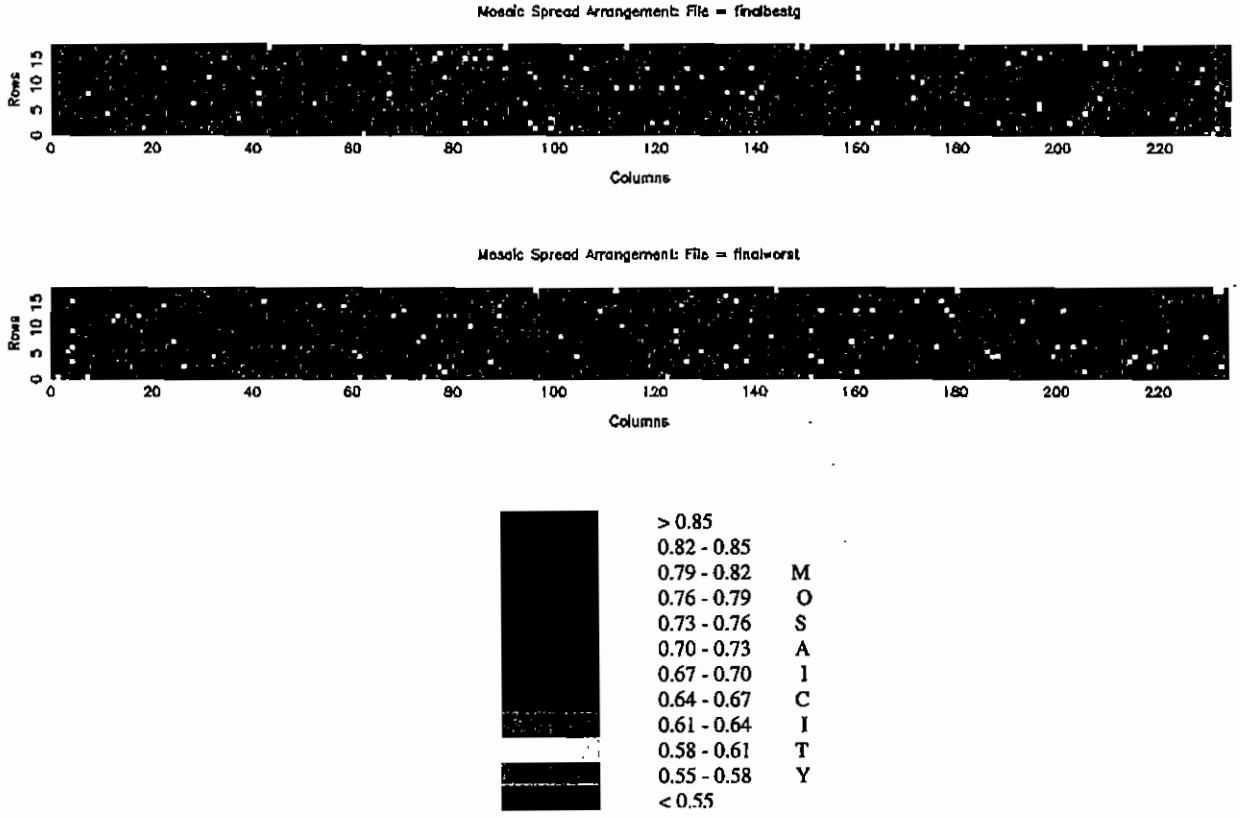

Figure 8 *'Best' (Top) and 'Worst' (Bottom) graphite crystal arrangements determined after 10^7 program iterations. The 'Totalmean' values for the two configurations are 1.35 and 2.17 respectively*

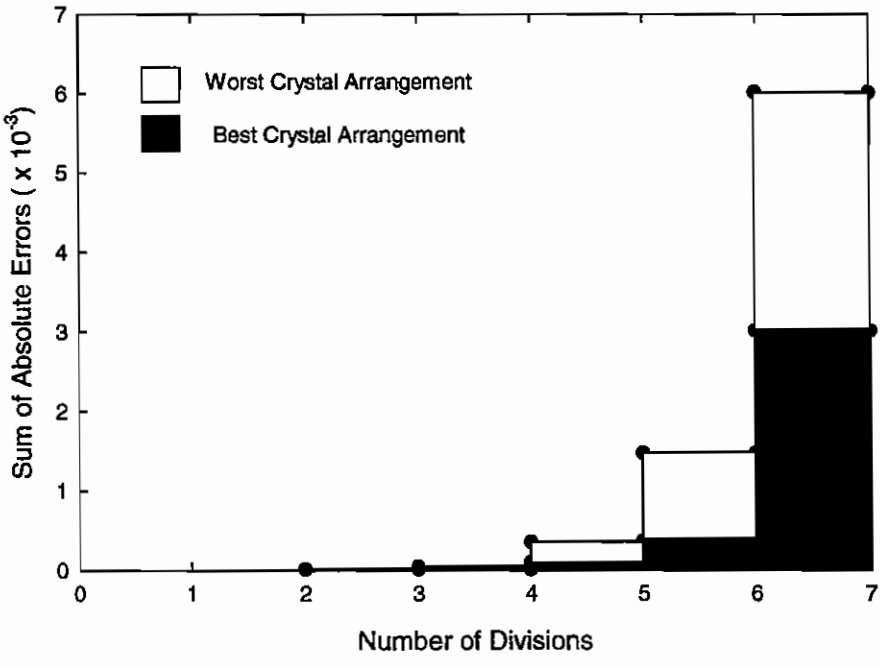

Figure 9 *Comparison of the 'best' and 'worst' graphite crystal configurations as ascertained using FORTRAN program 'Test' (Appendix IV), see text.*

VI. DISCUSSION

Design problems incurred by equipping the IRIS spectrometer with a new 4212 piece pyrolytic graphite (PG) crystal analyser array have been discussed. Of the 4212 pyrolytic graphite pieces required approximately 2500 will be newly purchased PG crystals (Type 1) with the remainder comprising of the currently installed graphite analysers (Type 2). Techniques employed to circumvent these problems included accurate calliper measurements, FORTRAN programming and statistical analysis.

The quality of the new PG crystals, with respect to initial manufacturing specifications, has been successfully ascertained. All Type 1 graphite pieces, which passed an initial visual inspection for prominent physical imperfection, appear suitable for inclusion on the new analyser array.

To limit pronounced variation in mosaicity across the face of the analyser bank, and hence potential degradation of current instrument angular resolution, Monte Carlo methods have been used to generate the optimum arrangement of Type 1 and Type 2 graphite crystals. Comparison between the 'best' and 'worst' crystal arrangements, generated after 10 million Monte Carlo iterations, show the 'best' to be indeed the better. This arrangement is therefore to be used as a blueprint from which to assemble the actual pyrolytic graphite crystal array.

VII. REFERENCES

[1] C.J.Carlile and M.A.Adams, Physica **B 182** (1992) 431-440

[2] M.T.F.Telling and S.I.Campbell, *'The Optimisation of Analyser Geometry for...IRIS on the ISIS Pulsed Source'*, Rutherford Appleton Laboratory Report: RAL-TR-1999-044, 1999

[3] D.S.Sivia, in *Los Alamos Science* (1990), **19**, 180-206

[4] R.Crevecoeur, I.de Schepper, L.de Graaf, W.Montfrooij, E.Svensson and C.J.Carlile, *Nuclear Instruments and Methods in Physics Research A***356** (1995) 415

APPENDIX I

```
      PROGRAM RELIABILITY
*     --------------------
*
      Bayesian probability distribution program: computes probability
      distribution of graphite crystal lengths given two prior
      limits.

      REAL XP(1000),YP(1000),XD(1000),XYG(41,41),XG(41),YG(41)
      DATA NP,ND,NX,NY /400,1000,41,41/
*
      CALL DATIN(XD,ND,X0,X0SIG,SIG0)
      CALL PRIOR(XP,NP,X0,SIG0)
      WRITE(*,*)' Thinking ...'
      CALL POSTR(XD,ND,X0,X0SIG,SIG0,XYG,XG,YG,NX,NY,XP(1),XP(NP))
      CALL INFER(XP,NP,XG,YG,XYG,NX,NY,YP)
      CALL PGBEGIN(0,'?',1,1)
      CALL PGSCH(1.0)
      CALL PGENV(XP(1),XP(NP),0.0,1.07,0,0)
      CALL PGLINE(NP,XP,YP)
      CALL PGLABEL('Length \fil\fn (mm)','prob(\fil\fn|{data})',' ')

      CALL PGEND
      CALL CONFID(XP,YP,NP)
      END
*
*
*
*
      SUBROUTINE DATIN(X,N,X0,X0SIG,SIG0)
*     -----------------------------------
*
      REAL       X(*)
      CHARACTER*60 FILNAM
*
      XSUM=0.0
      XXSUM=0.0
      WRITE(*,*)
    1 WRITE(*,100)
  100 FORMAT(' Input>  Data filename ?  : ',$)
      READ(*,200,ERR=1) FILNAM
  200 FORMAT(A)
      OPEN(UNIT=1,FILE=FILNAM,STATUS='OLD',FORM='FORMATTED',READONLY,
     *     ERR=1)
      DO 10 I=1,N
        READ(1,*,ERR=2,END=2) X(I)
        XSUM=XSUM+X(I)
        XXSUM=XXSUM+X(I)*X(I)
   10 CONTINUE
    2 CLOSE(UNIT=1)
      N=I-1
      WRITE(*,*)
      WRITE(*,110) N
  110 FORMAT(' No. of data read in = ',I5)
      IF (N.LT.2) STOP Need at least 2 data for sensible calculation!
      XNORM=1.0/FLOAT(N)
      X0=XSUM*XNORM
      SIG0=SQRT((XXSUM-X0*XSUM)*XNORM)
      X0SIG=SIG0*SQRT(XNORM)
      WRITE(*,*)
      WRITE(*,120) X0,X0SIG
```

```
      WRITE(*,130) SIG0,X0SIG*0.7071
 120  FORMAT(' Mean                =   ',F7.3'  +/-  ',F7.3'  mm')
 130  FORMAT(' Standard deviation =   ',F7.3'  +/-  ',F7.3'  mm')
      END
*
*
      SUBROUTINE PRIOR(X,N,X0,SIG0)
*     ----------------------------
*
      REAL X(*)
*
      WRITE(*,*)
   1  WRITE(*,100)
 100  FORMAT(' Prior range>  Xmin & Xmax ?  : ',$)
      READ(*,*,ERR=1) XMIN,XMAX
      IF (XMIN.GE.XMAX) GOTO 1
      IF (XMIN.GT.(X0+10.0*SIG0)) GOTO 1
      IF (XMAX.LT.(X0-10.0*SIG0)) GOTO 1
      CALL XINIT(X,XMIN,XMAX,N)
      END
*
*
      SUBROUTINE XINIT(X,XMIN,XMAX,N)
*     ------------------------------
*
      REAL X(*)
*
      DX=(XMAX-XMIN)/FLOAT(N-1)
      X(1)=XMIN
      DO 10 I=2,N
  10    X(I)=X(I-1)+DX
      END
*
*
      SUBROUTINE POSTR(XD,ND,X0,X0SIG,SIG0,XYG,XG,YG,NX,NY,XMIN,XMAX)
*     --------------------------------------------------------------
*
      REAL XD(*),XYG(NX,NY),XG(*),YG(*)
*
      XGMIN=X0-5.0*X0SIG
      XGMAX=X0+5.0*X0SIG
      IF (XGMIN.LT.XMIN) XGMIN=XMIN
      IF (XGMAX.GT.XMAX) XGMAX=XMAX
      YGMIN=SIG0-5.0*0.7071*X0SIG
      YGMAX=SIG0+5.0*0.7071*X0SIG
      IF (YGMIN.LE.1.0E-10) YGMIN=YGMAX/1000.0
      CALL XINIT(XG,XGMIN,XGMAX,NX)
      CALL XINIT(YG,YGMIN,YGMAX,NY)
      ZMIN=1.0E+20
      DO 30 J=1,NY
        SIG2=1.0/(2.0*YG(J)*YG(J))
        SIGLOG=FLOAT(ND)*LOG(YG(J))
        DO 20 I=1,NX
          XGI=XG(I)
          SUM=0.0
          DO 10 K=1,ND
            DIF=XD(K)-XGI
            SUM=SUM+DIF*DIF
  10      CONTINUE
          Z=SUM*SIG2+SIGLOG
          XYG(I,J)=Z
          IF (Z.LT.ZMIN) ZMIN=Z
  20    CONTINUE
  30  CONTINUE
      DO 50 J=1,NY
        DO 40 I=1,NX
```

```
          PRBLOG=XYG(I,J)-ZMIN
          XYG(I,J)=EXP(-PRBLOG)
 40     CONTINUE
 50   CONTINUE
      END
*
*
      SUBROUTINE INFER(XP,NP,XG,YG,XYG,NX,NY,YP)
*     ------------------------------------------
*
      REAL XP(*),XYG(NX,NY),XG(*),YG(*),YP(*)
*
      PMAX=-1.0E+20
      DO 30 K=1,NP
        SUM=0.0
        DO 20 J=1,NY
          SIGNRM=1.0/YG(J)
          SIG2=0.5*SIGNRM*SIGNRM
          DO 10 I=1,NX
            DIF=XP(K)-XG(I)
            SUM=SUM+XYG(I,J)*EXP(-SIG2*DIF*DIF)*SIGNRM
 10       CONTINUE
 20     CONTINUE
        YP(K)=SUM
        IF (SUM.GT.PMAX) PMAX=SUM
 30   CONTINUE
      PNORM=1.0/PMAX
      DO 40 K=1,NP
 40     YP(K)=YP(K)*PNORM
      END
*
*
      SUBROUTINE CONFID(X,Y,N)
*     ------------------------
*
      REAL     X(*),Y(*)
      LOGICAL ANOTHR
*
      YMAX=0.0
      YSUM=0.0
      DO 10 I=1,N
        IF (Y(I).GT.YMAX) THEN
          YMAX=Y(I)
          I0=I
        ENDIF
        YSUM=YSUM+Y(I)
 10   CONTINUE
      YNORM=100.0/YSUM
      DO 20 I=1,N
 20     Y(I)=Y(I)*YNORM
  1   WRITE(*,*)
      WRITE(*,100)
100   FORMAT(' Reliability>  Confidence level ?  (%) : ',$)
      READ(*,*,ERR=1) PVAL
      IF (PVAL.LT.0.0 .OR. PVAL.GT.100.0) GOTO 1
      CALL INTRVL(X,Y,N,I0,PVAL)
      WRITE(*,*)
      CALL LOGQYN(' >  Try another confidence-interval ?','Y',ANOTHR)
      IF (ANOTHR) GOTO 1
      END
*
*
      SUBROUTINE INTRVL(X,Y,N,I0,PVAL)
*     --------------------------------
*
      REAL X(*),Y(*)
```

```
*
      DX2=(X(2)-X(1))/2.0
      IMIN=I0
      IMAX=I0
      P=Y(I0)
      GOTO 2
    1 IF (IMIN.EQ.1) THEN
        IF (IMAX.EQ.N) GOTO 3
        IMAX=IMAX+1
        P=P+Y(IMAX)
      ELSEIF (IMAX.EQ.N) THEN
        IMIN=IMIN-1
        P=P+Y(IMIN)
      ELSE
        IF (Y(IMIN-1).GT.Y(IMAX+1)) THEN
          IMIN=IMIN-1
          P=P+Y(IMIN)
        ELSE
          IMAX=IMAX+1
          P=P+Y(IMAX)
        ENDIF
      ENDIF
    2 IF (P.LT.PVAL) GOTO 1
    3 XMIN=X(IMIN)-DX2
      XMAX=x(IMAX)+DX2
      IF (XMIN.LT.X(1)) XMIN=X(1)
      IF (XMAX.GT.X(N)) XMAX=X(N)
      WRITE(*,*)
      WRITE(*,100) PVAL,XMIN,XMAX
  100 FORMAT(' Shortest ',F4.1'% confidence interval = ',F7.2' to ',
     *       F7.2' mm')
      END
```

APPENDIX II

```
      PROGRAM AVERAGE
*     -----------------
*
      This program reads in the mosaic spreads (front and back face)
      of the graphite crystals from ATOMGRAPH and calculates the
      arithmetic mean of the front faces, back faces and average of
      the two. The values are stored in a data file called SPREAD.DAT
      in single column format: Front mosaic spread, back mosaic
      spread etc for each crystal. The order of the crystals follows
      their position in the trays in the following way:


*
*           11    12    13
*
*           21    22    23
*
*           31    32    33
*

*     For each position their are two values of mosaic spread, front
      and back. There are 281 trays, the final tray being one third
      full. If we ignore final tray then the total number of crystals
      is 2520.
*

      INTEGER
      ITRAY(2,3,3,280),FOCCUF(60:200),FOCCUB(60:200),FOCCUA(60:200)
      INTEGER LOOP, IFVAL, IBVAL, IAVAL
      REAL FSUM, BSUM, ASUM, FMEAN, BMEAN, AMEAN, FERRSQ, BERRSQ
      REAL AERR, AERRSQ, FESQSU, BESQSU, AESQSU, FDEV, BDEV, ADEV
      REAL FMIN, BMIN, AVER, AMIN, FMAX, BMAX, AMAX
      REAL OCCUF(30:100), OCCUB(30:100), OCCUA(30:100)
      REAL*8 AVE(3,3,280)
*
*     Read data into four-d array corresponding to real physical
      position in tray stack.
*
      OPEN(UNIT=7,FILE='NAMESPREAD.DAT',STATUS='OLD')
      READ(7,5) ((((ITRAY(I,J,K,L), I=1,2), J=1,3), K=1,3), L=1,280)
 5    FORMAT(10X,F5.3)
*
*
*     Initialise SUMS, MINIMUMS and MAXIMUMS
*
      FSUM = 0.0
      BSUM = 0.0
      ASUM = 0.0
      FESQSU = 0.0
      BESQSU = 0.0
      AESQSU = 0.0
      FMIN = 150.0
      BMIN = 150.0
      AMIN = 150.0
      FMAX = 0.0
      BMAX = 0.0
      AMAX = 0.0
*
      DO  LOOP=60,200
```

```
            FOCCUF(LOOP) = 0
            FOCCUB(LOOP) = 0
            FOCCUA(LOOP) = 0
      ENDDO
*
*     Calculate arithmetic mean and standard deviation for 1)Front
      faces, 2)Back faces, 3)Average of front and back faces.
*
      DO  L=1,280
            DO  K=1,3
                  DO  J=1,3
                  FSUM = FSUM + ITRAY(1,J,K,L)
                  BSUM = BSUM + ITRAY(2,J,K,L)
                  ASUM = ASUM + ((ITRAY(1,J,K,L)+ITRAY(2,J,K,L))/2.0)
                  AVE(J,K,L) =((ITRAY(1,J,K,L)+ITRAY(2,J,K,L))/200.0)
                  ENDDO
            ENDDO
      ENDDO
*
      FMEAN = FSUM/2520
      BMEAN = BSUM/2520
      AMEAN = ASUM/2520
      DO  L=1,280
         DO  K=1,3
            DO  J=1,3
            FERRSQ = (ITRAY(1,J,K,L)-FMEAN)*(ITRAY(1,J,K,L)-FMEAN)
            BERRSQ = (ITRAY(2,J,K,L)-BMEAN)*(ITRAY(2,J,K,L)-BMEAN)
            AERR = ((ITRAY(1,J,K,L)+ITRAY(2,J,K,L))/2)-AMEAN
            AERRSQ = AERR*AERR
            FESQSU = FESQSU+FERRSQ
            BESQSU = BESQSU+BERRSQ
            AESQSU = AESQSU+AERRSQ
            ENDDO
         ENDDO
      ENDDO
*
      FDEV = SQRT(FESQSU/2520)
      BDEV = SQRT(BESQSU/2520)
      ADEV = SQRT(AESQSU/2520)
*
*
*     Calculate minimum, maximums and modal values
*
      DO  L=1,280
            DO  K=1,3
                  DO  J=1,3
*
                  IF (ITRAY(1,J,K,L).LT.FMIN) THEN
                        FMIN = ITRAY(1,J,K,L)
                  END IF
                  IF (ITRAY(1,J,K,L).GT.FMAX) THEN
                        FMAX = ITRAY(1,J,K,L)
                  END IF
*
                  IF (ITRAY(2,J,K,L).LT.BMIN) THEN
                        BMIN = ITRAY(2,J,K,L)
                  END IF
                  IF (ITRAY(2,J,K,L).GT.BMAX) THEN
                        BMAX = ITRAY(2,J,K,L)
                  END IF
*
      AVER = ((ITRAY(1,J,K,L)+ITRAY(2,J,K,L))/2.0)
                  IF (AVER.LT.AMIN) THEN
                        AMIN = AVER
                  END IF
                  IF (AVER.GT.AMAX) THEN
```

```
                    AMAX = AVER
                END IF
*
*
                ENDDO
            ENDDO
        ENDDO
*
*       Calculate frequencies of occurrence for plotting purposes
*
*
        DO  L=1,280
            DO  K=1,3
                DO  J=1,3
                IFVAL = ITRAY(1,J,K,L)
                IBVAL = ITRAY(2,J,K,L)
                IAVAL = IFVAL+IBVAL

*       Don't divide by two yet - keep as an integer for array
        subscript. Times by two to obtain same array subscripts for
        IFVAL, IBVAL and IAVAL

                IFVAL = IFVAL*2
                IBVAL = IBVAL*2
*
                FOCCUF(IFVAL) = FOCCUF(IFVAL)+1
                FOCCUB(IBVAL) = FOCCUB(IBVAL)+1
                FOCCUA(IAVAL) = FOCCUA(IAVAL)+1
                ENDDO
            ENDDO
        ENDDO
*
*       Write out values:
*
        WRITE(*,*) 'Average of front faces = ', FMEAN/100
        WRITE(*,*) 'The minimum front face value = ', FMIN/100
        WRITE(*,*) 'The maximum front face value = ', FMAX/100
        WRITE(*,*) 'The front face standard deviation = ', FDEV/100
        WRITE(*,*) 'Average of back faces = ', BMEAN/100
        WRITE(*,*) 'The minimum back face value = ', BMIN/100
        WRITE(*,*) 'The maximum back face value = ', BMAX/100
        WRITE(*,*) 'The back face standard deviation = ', BDEV/100
        WRITE(*,*) 'The mean of average of two faces = ', AMEAN/100
        WRITE(*,*) 'The minimum average value = ', AMIN/100
        WRITE(*,*) 'The maximum average value = ', AMAX/100
        WRITE(*,*) 'The standard deviation of averages = ', ADEV/100
*
        OPEN(UNIT=9, FILE='FREQUENCIES.DAT', STATUS='NEW')
*
        WRITE(9,5)
*
 5      FORMAT(1X,'MOSAIC',5X,'FRONT',5X,'BACK',3X,'AVERAGE')
*
        DO  LOOP=60,200
*
        WRITE(9,15) LOOP/200.0, FOCCUF(LOOP), FOCCUB(LOOP),FOCCUA(LOOP)
        ENDDO
*
 15     FORMAT(2X,F5.3,5X,I4,5X,I4,5X,I4)
*
        OPEN(UNIT=7,FILE='MEANSPREADS.DAT',STATUS='NEW')
*
        N = 0
        DO  L=1,280
            DO  K=1,3
                DO  J=1,3
```

```
                IF(AVE(J,K,L).GT.0.520) THEN
                WRITE(7,25) AVE(J,K,L)
                N = N+1
                IF(N.EQ.2486) GO TO 20
                ELSE
                END IF
                ENDDO
          ENDDO
20    ENDDO
25    FORMAT(F5.3)
      END
```

APPENDIX III

```
      PROGRAM GRIDASSIGN
*     --------------------
*
            Program revised 24/9/99 for 4212 crystals (2457 new and
      1755 old crystals) "GRIDASSIGN" is a Fortran 77 program,
      designed to arrange 4212 pyrolytic graphite crystals in their
      "ideal" positions on the new IRIS analyser bank. There are two
      types of crystal, 1mm thick and 2mm thick. There are 2457 (new)
      and 1755 (old) crystals. Each crystal has a particular mosaic
      spread; the purpose of this program is to arrange the crystals
      in such a way that the mosaic spread of the whole bank is as
      evenly distributed as possible. This is achieved by placing the
      crystals on to the 234 x 18 grid at random, and calculating the
      average mosaic spread of consecutive blocks of nine crystals.
      The deviation from the mean, 0.72, is then worked out and
      squared, and all the errors across the whole grid are summed.
      This final value is used as a measure of the evenness of the
      distribution; the arrangement with the smallest value is stored
      and the process is repeated.
*
      The mosaic spread data is stored in a file called
      NAMESPREAD.DAT,in order of their positions in the trays, e.g.
      tray#1,row#1,column#1.  Rather than keep track of each
      individual 1mm crystal, it is just their mosaic spreads that
      are manipulated. When the final arrangement of crystals is
      reached, a separate search program is used to give individual
      crystals a real position in the analyser bank.
*
*
*
      PARAMETER(MAXITER=10000000)
      REAL  SUM, MEAN, COUNT, J
      REAL*8  ERR, ERRSQ, SUMERRSQ, GMAX, GMIN

*     my arrangement of old crystals uses 1755 hence 2457 new
      crystals needed

      REAL  X(2457), GRID(234,18), BEST(234,18), WORST(234,18)
      INTEGER  LIST(234,18), LISTSEED(MAXITER)
      INTEGER  MSEED, ISEED,P
*
      DATA  (LISTSEED(I), I=1,MAXITER)/MAXITER*0/
      GMIN = 10.0
      GMAX = 1.0
      COUNT = 1
      MSEED = 314159
      P=0
*
*
*     Set both GRID and LIST arrays to zero:
*
 10   DO  I=1,234
            DO  J=1,18
            LIST(I,J) = 0
            GRID(I,J) = 0
            ENDDO
      ENDDO

*     write (*,*) 'clear list and grid OK'
```

```
      SUMERRSQ = 0
      MEAN = 0
      SUM = 0
      ISEED = 0

*     write (*,*) 'set sumerrsq to zero OK'
*
*     Mark the positions of the OLD crystals in the array LIST:
*

                flag=1
                DO  J=1,18
                        IF(MOD(J,2.0).gt.0) THEN
                        if (flag.eq.0) then
                        flag=1
                        else
                        flag=0
                        endif
                        DO C=0,77
                        Y=1+(C*3)
                        LIST(Y,J) = 2
*                        WRITE (8,*) Y,J,flag
*                        write (*,*) Y,J,flag
                        IF(MOD(C,2.0).gt.0) THEN
                        if (flag.eq.0) then
                        LIST(Y+1,J) = 2
*                        write (8,*) Y+1,J,flag
*                        write (*,*) Y+1,J,flag
                        else
                        LIST(Y-2,J) = 2
*                        write (8,*) Y-2,J,flag
*                        write (*,*) Y-2,J,flag
                        endif
                        endif
                        ENDDO
                ELSE
                        DO C=0,77
                        Y=1+(C*3)
                        LIST(Y+2,J) = 2
*                        WRITE (8,*) Y+2,J
                        ENDDO
                ENDIF


      ENDDO

*     write (*,*) 'set position of old crystals OK'

*
*     Read in mosaic spread data:
*
      OPEN(UNIT=7,FILE='NAMESPREAD.DAT',STATUS='OLD')
      READ(7,5,END=20) (X(I), I=1,2457)
 5    FORMAT(10X,F5.3)

*     write (*,*) 'reads in mosaic spread data OK'
*
*     Place the 2mm thick crystals on to the GRID - these are assumed
      to have the average mosaic spread:
*
 20   DO  I=1,234
            DO  J=1,18
            IF(LIST(I,J).EQ.2) THEN
            GRID(I,J) = 0.700
            END IF
            ENDDO
```

```
      ENDDO

*     write (*,*) 'places old crystals on grid (assign 0.7) OK'
*
*     Now assign each 1mm thick crystal spread a position on the GRID
      at random and mark it's place in the LIST:
*

      if (count.eq.99999) then
      count=1
      p=p+1
      write (*,*) P
      endif


 29   ISEED = NINT((RAN(MSEED))*(MAXITER-1))
           if (iseed.eq.0) then
           go to 29
           end if
 30   IF(LISTSEED(ISEED).EQ.1) THEN
 31    ISEED = NINT((RAN(MSEED))*(MAXITER-1))
           if (iseed.eq.0) then
           go to 31
             end if
       GO TO 30
      END IF
      LISTSEED(ISEED) = 1
*
      DO  N=1,2457
           IRAN = NINT((RAN(ISEED))*234)
           JRAN = NINT((RAN(ISEED))*18)
 40        IF((LIST(IRAN,JRAN).GT.0).OR.(IRAN.EQ.0).OR.(JRAN.EQ.0))
     &           THEN
           IRAN = NINT((RAN(ISEED))*234)
           JRAN = NINT((RAN(ISEED))*18)
           GO TO 40
           END IF
           GRID(IRAN,JRAN) = X(N)
           LIST(IRAN,JRAN) = 1
      ENDDO

*     write (*,*) 'assigns 1mm crystals a place on grid OK'

*
*
*     The grid is now tested for evenness of the distribution of
      mosaic spreads. This is done by summing the squares of the
      deviations from the means of blocks of 9 crystals on the grid.
*
      DO  I=2,233
           DO  J=2,17
*
*     Calculate the mean of 9 squares:
*
                SUM = 0
                DO M=I-1,I+1
                     DO N=J-1,J+1
                     SUM = SUM + GRID(M,N)
                     ENDDO
                ENDDO
                MEAN = SUM/9.0

*     write (*,*) ' mean of 9 squares tested OK'

*
```

```
*     Now work out the deviation from the overall mean,0.72, and sum
      the squares of these errors.
*
                  ERR = MEAN - 0.72
                  ERRSQ = ERR*ERR
                  SUMERRSQ = SUMERRSQ +ERRSQ

            ENDDO
      ENDDO

*     write (*,*) ' derivation from mean calculated OK'
*
*     If the value of SUMERRSQ is the smallest so far, store the GRID
      in a separate array. Also keep the GRID with the largest value
      for comparison.
*
      IF(SUMERRSQ.LT.GMIN) THEN
         GMIN = SUMERRSQ
         DO  J=1,18
            DO  I=1,234
            BEST(I,J) = GRID(I,J)
            ENDDO
         ENDDO
      ELSE IF(SUMERRSQ.GT.GMAX) THEN
         GMAX = SUMERRSQ
         DO  J=1,18
            DO  I=1,234
            WORST(I,J) = GRID(I,J)
            ENDDO
         ENDDO
      END IF

*     write (*,*) 'gmin and gmax compared OK'

*
*     Every  25000  iterations,  write  out  the  best  and  worst
      arrangements into two datafiles
*
      COUNT = COUNT + 1.0
      IF(MOD(COUNT,25000.0).EQ.0) THEN
            OPEN (UNIT=8,FILE='BESTGRID1.DAT',STATUS='NEW')
            OPEN (UNIT=9,FILE='WORSTGRID1.DAT',STATUS='NEW')
            WRITE (8,15) ((BEST(I,J), J=1,18), I=1,234)
            WRITE (9,15) ((WORST(I,J), J=1,18), I=1,234)
            CLOSE (8)
            CLOSE (9)
      ENDIF
*
 15   FORMAT(1X,16F4.3)
*
      WRITE(*,25) 'Iteration number', NINT(COUNT), GMIN ,GMAX
 25   FORMAT ('+',A16,I6,3X,F8.5,1X,F8.5)

*     write (*,*) 'best and worst grids written out every 5K counts'

      GO TO 10
*
*     This program is designed to run continually, achieving a more
      preferable  arrangement  of  crystals  each  time  until  it  is
      interrupted.
*
      END
```

IV. APPENDIX 4

```
      PROGRAM TEST
*      --------------
*
*     Prints out average horizontal mosaic spreads at decreasing
      length intervals, to check arrangements of crystals.
*
      REAL  GRID(239,16),MEAN(128),ERROR(128)
      REAL  SUM(0:6),X(7),Y(7)
      INTEGER  IDUMMY(0:128)
      CHARACTER*80  FILNAM,TITLE
*
*     Read in data:
*
      WRITE(*,*) 'Enter Filename: '
      READ(*,5) FILNAM
 5    FORMAT(A10)
      OPEN(UNIT=7,FILE=FILNAM,STATUS='OLD',FORM='FORMATTED',READONLY)
      READ(7,15) ((GRID(I,J), J=1,16), I=1,239)
 15   FORMAT(1X,16F4.3)
*
*     Calculate average mosaic spread of whole array, two half
      arrays, four quarters etc. and sum squared deviations from
      overall mean.
*
      J = 0
 20   I = 2**J
      CALL AVERAGE(GRID,239,16,I,MEAN,IDUMMY)
      SUM(J) = 0.0
      DO  L=1,I
            ERROR(L) = (MEAN(L)-0.7174)**2
            SUM(J) = SUM(J) + ERROR(L)
      ENDDO
*
*     List sum values.
*
      WRITE(*,25) SUM(J)
 25   FORMAT(F8.6)
      WRITE(*,*)
      J = J+1
        IF(I.LT.64) GO TO 20
*
*
      DO  M=0,6
            X(M) = M
            Y(M) = SUM(M)
      ENDDO
*
*     Draw bar chart of values.
*
      WRITE(TITLE,35) FILNAM
 35   FORMAT('Deviations from Overall Mean at Successive Bank
     & Divisions - File: ',A10)
*
      CALL PGBEGIN(0,'?',1,1)
      CALL PGENV(0.,7.,0.,0.007,0,1)
      CALL PGLABEL('No. of Divisions','Sum of Absolute Errors',
     &  TITLE)
      CALL PGSCI(1)
      DO  M=0,6
            X1 = M
```

```
            X2 = M+1
            Y1 = 0.0
            Y2 = SUM(M)
            CALL PGRECT(X1,X2,Y1,Y2)
      ENDDO
      CALL PGEND
      END
*
*
*
*
      SUBROUTINE AVERAGE(ARRAY,IX,IY,N,MEAN,K)
*     ----------------------------------------
*
      REAL  ARRAY(IX,IY), MEAN(N)
      INTEGER  ICOUNT, K(0:N)
*
      K(0) = 1
      DO  M=1,N
            K(M) = (IX/N)*M
      ENDDO
*
      DO  M=1,N
            SUM = 0.0
            ICOUNT = 0
            DO  I=K(M-1),K(M)
                  DO  J=1,IY
                  SUM = SUM+ARRAY(I,J)
                  ICOUNT = ICOUNT+1
                  ENDDO
            ENDDO
            MEAN(M) = SUM/FLOAT(ICOUNT)
      ENDDO
      END
```