A COLLECTION OF FORTRAN PROGRAMS
FOR CRYSTAL STRUCTURE ANALYSIS

H. G. Norment

Diffraction Branch
Optics Division

April 29, 1963

U. S. NAVAL RESEARCH LABORATORY
Washington, D.C.
CONTENTS

Preface iv
Problem Status iv
Authorization iv

1. UNIT CELL PARAMETERS AND ERRORS BY LEAST SQUARES 1
   1.1 Purpose 1
   1.2 Input 1
   1.3 Method of Calculation 2
   1.4 Output 3
   1.5 Limitations 3
   1.6 Special Subroutines Called 3
   1.7 Special Tape Requirements 3

2. QUASI NORMALIZATION OF STRUCTURE FACTORS 3
   2.1 Purpose 3
   2.2 Input 3
   2.3 Method of Calculation 4
   2.4 Output 4
   2.5 Limitations 5
   2.6 Special Subroutines Called 5
   2.7 Special Tape Requirements 5

3. RATIONAL DEPENDENCE 5
   3.1 Purpose 5
   3.2 Input 5
   3.3 Method of Calculation 6
   3.4 Output 7
   3.5 Limitations 7
   3.6 Special Subroutines Called 7
   3.7 Special Tape Requirements 7

4. SIGMA-2 LISTINGS 7
   4.1 Purpose 7
   4.2 Input 7
   4.3 Method of Calculation 8
   4.4 Limitations 8
   4.5 Special Subroutines Called 8
   4.6 Special Tape Requirements 9

5. TRIPLE PRODUCT SUMMATION FOR ORTHORHOMBIC CRYSTALS 9
   5.1 Purpose 9
   5.2 Input 9
   5.3 Method of Calculation 10
5.4 Output 11
5.5 Limitations 11
5.6 Special Subroutines Called 11
5.7 Special Tape Requirements 11

6. STRUCTURE FACTOR CALCULATION 11
6.1 Purpose 11
6.2 Input 11
6.3 Data Deck 15
6.4 Method of Calculation 15
6.5 Printed Output 16
6.6 Tape Output 17
6.7 Special Subroutines Called 17
6.8 Special Tape Requirements 17

7. INTERATOMIC DISTANCES AND ANGLES WITH INTERPOLATION FOR PEAK CENTER LOCATION 17
7.1 Purpose 17
7.2 Input 18
7.3 Data Deck 20
7.4 Method of Calculation 20
7.5 Output 20
7.6 Limitations 21
7.7 Special Subroutines Called 21
7.8 Special Tape Requirements 21

8. LEAST-SQUARES PLANE AND LINE FITTER 21
8.1 Purpose 21
8.2 Input 21
8.3 Data Deck 23
8.4 Method of Calculation 23
8.5 Output 23
8.6 Limitations 24
8.7 Special Subroutines Called 24
8.8 Special Tape Requirements 24

9. POINT TO PEAK DISTANCE CALCULATION 24
9.1 Purpose 24
9.2 Input 24
9.3 Data Deck 25
9.4 Method of Calculation 25
9.5 Output 26
9.6 Limitations 26
9.7 Special Subroutines Called 26
9.8 Special Tape Requirements 26

10. FORM FACTORS FOR THE BUSING LEAST-SQUARES REFINEMENT PROGRAM 26
10.1 Purpose 26
10.2 Input 26
PREFA CE

This report contains descriptions and operating instructions for a collection of general utility programs applicable to various phases of crystal structure analysis with computers. All coding is done in FORTRAN II language for 32K core IBM 704 and 7090 machines; the FORTRAN-language programs are completely listed in the Appendix.

The data input to many of the programs has been designed to be compatible with the output of the automatic data reduction program XRDDR (H.G. Norment, “An X-Ray Diffraction Data Reduction Program for the IBM 704 and 7090,” NRL Report 5739, Feb. 1962).

Where data input from special tape is required, the logical tape numbers are assigned symbolic designations near the beginnings of the FORTRAN programs. Thus, if it is required that these tape numbers be changed, it is sufficient to change only the one FORTRAN statement card and then recompile.

In the form presented in this report, the programs are written for use with the IBM 7090 IB Monitor system.

PROBLEM STATUS

This is a final report on one phase of the problem; work on other phases continues.

AUTHORIZATION

NRL Problem C07-03
Project RR 001-02-43-4805

1. UNIT CELL PARAMETERS AND ERRORS BY LEAST SQUARES

1.1 Purpose

Given a set of reflection data consisting of Miller indices and $\sin \theta$ or $\sin \theta/\lambda$ values, compute the least-squares reciprocal and real-space unit cell parameters and uncertainties for any crystal system.

1.2 Input

The data input is taken from cards immediately following the program deck. It consists of one control card and a deck of reflection cards.

A. Control Card:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-2</th>
<th>3-4</th>
<th>5-12</th>
<th>13-72</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>N1</td>
<td>LIST</td>
<td>YAM</td>
<td>Hollerith information</td>
</tr>
<tr>
<td>formats</td>
<td>12</td>
<td>12</td>
<td>E8.5</td>
<td>10A6</td>
</tr>
</tbody>
</table>

1. N1 specifies the crystal as follows:

<table>
<thead>
<tr>
<th>Crystal System</th>
<th>N1</th>
</tr>
</thead>
<tbody>
<tr>
<td>triclinic</td>
<td>1</td>
</tr>
<tr>
<td>monoclinic*</td>
<td>2</td>
</tr>
<tr>
<td>orthorhombic</td>
<td>3</td>
</tr>
<tr>
<td>tetragonal</td>
<td>4</td>
</tr>
<tr>
<td>hexagonal</td>
<td>5</td>
</tr>
<tr>
<td>cubic</td>
<td>6</td>
</tr>
</tbody>
</table>

2. If LIST = 0 or blank, the complete reflection data with observed and calculated $([\sin \theta]/\lambda)^2$ are printed out. If LIST $\neq 0$, only the cell parameter and errors are printed.

3. YAM is the wavelength of x radiation used, or else YAM is 0. If YAM = 0 or blank, the program assumes that $(\sin \theta)/\lambda$ values have been loaded. If YAM = $\lambda$, the program assumes that $\sin \theta$ values have been loaded.

B. Reflection Card:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-4</th>
<th>5-8</th>
<th>9-12</th>
<th>13-22</th>
<th>23-32</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>h</td>
<td>k</td>
<td>$\ell$</td>
<td>S</td>
<td>W</td>
</tr>
<tr>
<td>formats</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>E10.5</td>
<td>E10.5</td>
</tr>
</tbody>
</table>

1. $h,k,\ell$ are the Miller indices

2. S is either $\sin \theta$ or $(\sin \theta)/\lambda$ (see above)

3. W is a least-squares weighting factor. If W is not punched, the program assumes $W = (\sin \theta)/\lambda$.

*It is always assumed by the program that c is the unique axis; thus the first setting is used for the monoclinic system.
The reflection card deck is terminated with a blank card.

A second calculation may follow. The reflection deck of the last calculation is terminated with two blank cards.

1.3 Method of Calculation

Let \( S^2 = 4\sin^2 \theta / \lambda^2 \)

\[ H_{ij} = h_1 h_j, \quad i, j = 1,2,3 \]

\[ g^{ij} = a^1 a^j, \quad i, j = 1,2,3 \]

where \( h_1 \) and \( a^1 \) are respectively a Miller index and a reciprocal cell edge vector.

Then

\[ g^2 = H_{ij} g^{ij} \]

where the repetition of a subscript as a superscript in the same term of an expression denotes summation over that index.

The least squares criterion requires that

\[ F = [\sum (H_{ij} g^{ij} - S^2)^2] = \text{minimum}, \]

whence

\[ 1/2 \frac{\partial F}{\partial H_{mn}} = [\sum (H_{ij} g^{ij} - S^2) H_{nm}] = 0. \]

or

\[ [\sum H_{nm} H_{ij}] g^{ij} = [\sum S^2 H_{nm}], \quad n, m, i, j = 1,2,3, \]

where the brackets denote summation over all experimental observations, and \( W \) is a weighting factor, different in general for each observation.

After collecting terms, this becomes

\[ (2 - \delta_{nm}) (2 - \delta_{ij}) [\sum H_{nm} H_{ij}] g^{ij} = (2 - \delta_{nm}) [\sum S^2 H_{nm}] \]

where \( n, m, i, j = 1,2,3; \quad m > n; \quad j > i; \) and \( \delta_{ij} \) is the Kronecker delta. In matrix notation this becomes

\[ (h^{(nm)\langle ij \rangle}) \cdot (G^{(nm)}) = (D^{(nm)}) \]

where

\[ h^{(nm)\langle ij \rangle} = (2 - \delta_{nm}) (2 - \delta_{ij}) [\sum H_{nm} H_{ij}] \]

\[ G^{(nm)} = g^{nm} \]

\[ D^{(nm)} = (2 - \delta_{nm}) [\sum S^2 H_{nm}] \]

The \( G^{(nm)} \) are found by the usual matrix methods. The variances likewise are found by the usual methods, i.e.,

\[ \sigma^2 G^{(nm)} = \frac{F}{|h|} \cdot \frac{h^{(nm)(nm)}}{k - \ell} \]
where \( h^{(m)(n)} \) is the cofactor of \( h^{(m)(n)} \) element of \( |h| \), \( k \) is the number of terms in the bracket summation, and \( t \) is the order of \( |h| \).

The uncertainties in the unit cell parameters and volume are then found by error propagation methods. The equations are quite messy and will not be reproduced here.

1.4 Output

The output is self explanatory. It consists of the reciprocal cell parameters, the real cell parameters with uncertainties, the real cell volume with uncertainty, and, if desired, the observed and calculated S values are listed for each reflection.

1.5 Limitations

No more than 150 observations may be used.

1.6 Special Subroutines Called

In addition to the executive program, the following special subroutines are used:

- a. PARAM
- b. ERRPRM
- c. ERRREL
- d. OUTPRM.

Other nonlibrary programs used are:

- e. RECIP
- f. ARCSIN
- g. Modified versions of RW MATS and RW DET (SHARE distn. no. 635).

1.7 Special Tape Requirements

None

2. QUASI NORMALIZATION OF STRUCTURE FACTORS

2.1 Purpose

Prepare a BCD tape containing \( h, k, l, \) and \( F^2-1 \), similar to the one written by XRDDR, using for input the \( h, k, l, F^2 \) tape written by XRDDR in the Busing least-squares program format (1).

2.2 Input

The input consists of a card input, which is loaded following the program deck, and a tape input, which usually will consist of the data tape output from SUBROUTINE OUTPUT of XRDDR. The tape input is from logical tape 16.
A. Control Card:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
<th>31-40</th>
<th>41-50</th>
<th>51-60</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>NOIB</td>
<td>NASP</td>
<td>NF</td>
<td>BT1</td>
<td>EX</td>
<td>LIB</td>
</tr>
<tr>
<td>formats</td>
<td>I10</td>
<td>I10</td>
<td>E10.4</td>
<td>E10.4</td>
<td>I10</td>
<td></td>
</tr>
</tbody>
</table>

1. NOIB is the number of reflection records on logical tape 16.
2. NASP is the number of different atomic species in the crystal.
3. NF is the number of files on the input tape to be spaced over. This allows input to be taken from a library tape.
4. BT1 is the modified Wilson equation temperature factor B (as taken directly from the output of XRDDR).
5. EX is the exponent in the modified Wilson equation (labeled x in the output of XRDDR).
6. LIB is 1 if a library tape is used as input. It is blank otherwise. (It is assumed that the first record of a library tape is an identification record.)

Atomic Scattering Factor Cards:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-8</th>
<th>9-16</th>
<th>17-24</th>
<th>25-32</th>
<th>33-40</th>
<th>41-44</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>AS</td>
<td>AS1</td>
<td>BS</td>
<td>BS1</td>
<td>CS</td>
<td>AN</td>
</tr>
<tr>
<td>formats</td>
<td>F8.4</td>
<td>F8.4</td>
<td>F8.4</td>
<td>F8.4</td>
<td>F8.4</td>
<td>F4.0</td>
</tr>
</tbody>
</table>

The quantities AS, AS1, BS, BS1 and CS are the parameters λ, a, B, b, and c defined and listed by Forsyth and Wells for calculation of atomic scattering factors (3). AN is the number of atoms of the atomic species in the unit cell. There is one card for every different atomic species in the crystal. The deck consists of one control card followed by NASP atomic scattering factor cards.

B. Reflection Data Tape:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-9</th>
<th>10-18</th>
<th>19-27</th>
<th>28-36</th>
<th>37-54</th>
<th>55-63</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>h</td>
<td>k</td>
<td>λ</td>
<td>F²</td>
<td>blank</td>
<td>(sin θ)/λ</td>
</tr>
<tr>
<td>formats</td>
<td>F9.2</td>
<td>F9.2</td>
<td>F9.2</td>
<td>F9.2</td>
<td>16X</td>
<td>F9.6</td>
</tr>
</tbody>
</table>

2.3 Method of Calculation

F² values are converted to $\xi^2$ values according to the following equation

$$\xi^2 = F^2 \left( \exp \left[ \frac{\lambda (\sin \theta)^2}{\lambda} \right] \right)/ \left( \sum F^2 \right).$$

The reflections are read from the input tape and processed in groups of 500 (or less for the last group). Thus, any number of reflections may be processed in one computer run.

2.4 Output

Logical tape 15 is rewound at the beginning of the calculation. Output is BCD on tape 15. The program writes an END FILE on 15 and rewinds it when the calculation is finished.
2.5 Limitations

There can be no more than twenty different atomic species in the crystal.

2.6 Special Subroutines Called

A FAP coded subroutine FLSKPD is called which forward-spaces tape 16 the number of files specified by NF (see control card).

2.7 Special Tape Requirements

Logical tape 16 is used for BCD reflection input.

3. RATIONAL DEPENDENCE*

3.1 Purpose

To determine the extent of rational dependence, as defined by Hauptman and Karle (4), in a crystal structure using the normalized structure factor magnitudes.

3.2 Input

The input consists of a card input, which is loaded following the program deck, and a tape input, which usually will consist of the data tape output from program SF NORM of XRDDR. The tape input is from logical tape 12.

A. Card Input

There are two types of cards in the card input.

a. 72 columns of Hollerith characters (these characters are used to title the output).

b. Control Card:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
<th>31-40</th>
<th>41-50</th>
<th>51-60</th>
<th>61-70</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>NA</td>
<td>NINT</td>
<td>NFIN</td>
<td>MINN</td>
<td>AN</td>
<td>SDMIN</td>
<td>JPUT</td>
</tr>
<tr>
<td>formats</td>
<td>I10</td>
<td>I10</td>
<td>I10</td>
<td>I10</td>
<td>E10.4</td>
<td>E10.4</td>
<td>I10</td>
</tr>
</tbody>
</table>

1. NA is 1 for noncentrosymmetric crystals; it is 0 or blank for centrosymmetric crystals.

2. NINT is the initial modulus.

3. **FIN** is the final modulas.

4. **MINN** is the minimum number of reflections considered significant in a subset showing rational dependence effects. Results will not be output for subsets of reflections with fewer than **MINN** members.

5. **AN** is the minimum number of standard deviations which is considered a significant difference between the subset average and overall average of $E^2$.

6. **SDMIN** is the smallest deviation between the subset average and the overall average of $E^2$, where the deviation is considered to significantly show the presence of rational dependence effects.

7. **JPUT** is not 0 when complete listings of reflections are desired for each subset for which evidence of rational dependence has been found. **JPUT** is 0 when the reflection listings are not desired.

Additional control cards may follow if additional calculations are desired. The card deck is terminated with a blank card.

**B. Tape Input.**

This tape may be tape number 5 or 7 from program **SF NORM** of XRDDR:

1. The first record on the tape consists of the number of reflection records on the tape (format 17).

2. Reflection records:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-4</th>
<th>5-8</th>
<th>9-12</th>
<th>10-32</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>h</td>
<td>k</td>
<td>$\ell$</td>
<td>$E^2-1$</td>
</tr>
<tr>
<td>formats</td>
<td>I4</td>
<td>I4</td>
<td>I4</td>
<td>F20.5</td>
</tr>
</tbody>
</table>

**3.3 Method of Calculation**

A FORTRAN source program written by Block and Yannoni was obtained from Dr. Block of the National Bureau of Standards. The program has been modified with respect to reflection input; certain redundancies in the selection of subsets have been eliminated; a provision for specifying the minimum significant number of reflections in the subsets has been added; the criterion for accepting as significant a deviation of average $E^2$ from the overall value has been operationally improved by making it independent of the numbers of reflections in the subsets; and, finally, the output has been vastly expanded and improved. Otherwise the calculations are done essentially as coded by Block and Yannoni.

All reflections in a subset satisfy the relation

$$ah + bk + cl = n \quad (\text{mod } m)$$

where $m$ is a positive integer limited (by the program) to a maximum value of 14; $a$, $b$, and $c$ may have integral values ranging from -13 to +14, depending upon the value of $m$.

If the value of average $E^2$ for a subset differs significantly from the average over all observed $E^2$, then rational dependence is said to exist in the crystal structure.

The program automatically checks all possible subsets for rational dependence effects for all moduli (i.e., all $m$) between the limits specified on the control card.
3.4 Output

The output consists of three parts A, B, and C. A and B are always obtained, and C is or is not obtained at the option of the program user.

A. This is a leading one-page printout consisting of the control information, overall average $E^2$, and the total number of reflections on the input tape.

B. For each subset for which significant rational dependence effects are found the following is printed: $a, b, c, n, m$, average $E^2$, and the number of reflections in the subset.

C. If the control datum JPUT is not zero, data for all reflections in the subset defined by the printout of B above are listed, immediately following the B printout, in the form: $h, k, l, E^2 - 1$, and $E^2 - A$, where $A$ is the overall average $E^2$.

3.5 Limitations

The only limitation in addition to those already mentioned in section 3.3 is that the number of input reflections be less than 5000.

3.6 Special Subroutines Called

A subroutine OUTRAT, especially designed to write output for this program, is called.

3.7 Special Tape Requirements

Logical tape 12 is used for reflection input.

4. SIGMA-2 LISTINGS

4.1 Purpose

For each member of an ordered list of reflections, where the member is defined by the Miller index triple $h$ and normalized structure factor magnitude $E$, the program lists all pairs of reflections $h_i$, $E_i$, and $h_j$, $E_j$ from a similar ordered list of reflections which satisfy the relation

$$
\bar{h} = h_i + h_j.
$$

A pair of reflections $\bar{h}_i$ and $\bar{h}_j$ is said to be a $\Sigma_2$ interaction pair for the reflection $\bar{h}$ if the equation above is satisfied.

4.2 Input

The data input consists of one control card, which is positioned at the end of the program deck, and a list of $h$ and $E^2 - 1$, which is read in either from logical tape 11 or from cards. The output tapes from program SF NORM of XRDDR may be used.
A. Control Card:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>Z</td>
<td>IZ</td>
<td>NOIB</td>
</tr>
<tr>
<td>formats</td>
<td>E10.4</td>
<td>110</td>
<td>110</td>
</tr>
</tbody>
</table>

1. All reflections for which $E^2 - 1 < Z$ are rejected from consideration.
2. If IZ = 0 the reflection data are read from logical tape 11. If IZ ≠ 0 the reflection data are read from cards.
3. Lists of $\Sigma_2$ interaction pairs are tabulated for the NOIB reflections with largest $E^2 - 1$ values. If NOIB is negative, lists are prepared for the entire group of reflections for which $E^2 - 1 > Z$ (see method of computation below).

B. Reflection Data.

The reflection data is prefaced with one card (or record) NRT containing the number of reflections to be read. The format is 17.

The reflection records contain the following information:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-4</th>
<th>5-8</th>
<th>9-12</th>
<th>13-32</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>h</td>
<td>k</td>
<td>$\ell$</td>
<td>$E^2 - 1$</td>
</tr>
<tr>
<td>formats</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>F20.5</td>
</tr>
</tbody>
</table>

4.3 Method of Calculation

As each reflection is read into the computer it is accepted or rejected on the basis of whether or not the criteria on $h, k, \ell$ specified in REJVEC (see section 4.5) are satisfied. If the criteria are not satisfied, REJVEC sets $E^2 - 1 = -10$. Then, the reflection is tested on the basis of the magnitude of $E^2 - 1$, and all reflections for which $E^2 - 1 < Z$ are rejected. (The minimum physically meaningful value of $E^2 - 1$ is -1.)

The selected set of reflections is arranged in decreasing order of magnitude of $E^2 - 1$. Then lists of $\Sigma_2$ interaction pairs are calculated and tabulated for the top NOIB reflections in the selected set. If NOIB has been entered as a negative number, the lists are calculated for every member of the selected set.

4.4 Limitations

The selected set of reflections may not exceed 2400 reflections. There may not be more than 1000 $\Sigma_2$ interaction pairs for each $\bar{h}$.

4.5 Special Subroutines Called

a. REJVEC.

This subroutine imposes acceptance criteria on $h, k, \ell$ of each reflection. The program user may write his own program or use the dummy program already written. The quantities $N1, N2, N3,$ and $Q$ in the calling sequence are $h, k, \ell$, and $E^2 - 1$. 
b. SIGVEC.

A crystal-system-dependent subroutine which selects \( \mathbf{h} \) and \( \mathbf{h}' \) for each \( \mathbf{h} \). If SIGVEC for the triclinic case is used, and the reflection decks are always "blownup" so as to include all symmetry mates of all reflections, then this subroutine becomes noncrystal-system dependent. SIGVEC versions for triclinic, monoclinic, and orthorhombic crystals are available.

c. OUTSIG.

This subroutine prints out the list of \( \mathbf{h}, E_i \) and \( \mathbf{h}', E_j \) and the products \( E_i E_j E \) for each \( \mathbf{h}, E \).

4.6 Special Tape Requirements

Logical tape 11 is used for reflection input.

5. TRIPLE PRODUCT SUMMATION FOR ORTHORHOMBIC CRYSTALS

5.1 Purpose

These programs (one for centrosymmetric and one for noncentrosymmetric crystals) are designed to calculate the quantity \( E_1 E_2 E_3 \), as defined by Hauptman and Karle (5), or the quantity \( |E_1 E_2 E_3| \cos(\phi_1 + \phi_2 + \phi_3) \), as defined by Karle and Hauptman (6).

5.2 Input

The input consists of a card input, which is loaded following the program deck, and a tape input, which usually will consist of an output tape from program SF NORM of XRDDR.

A. Control Card:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
<th>31-40</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>NA</td>
<td>IM</td>
<td>JM</td>
<td>KM</td>
</tr>
<tr>
<td>formats</td>
<td>110</td>
<td>110</td>
<td>110</td>
<td>110</td>
</tr>
</tbody>
</table>

1. NA is the number of atoms (exclusive of hydrogen atoms) in the unit cell.
2. IM is the maximum value of Miller index \( h \).
3. JM is the maximum value of Miller index \( k \).
4. KM is the maximum value of Miller index \( l \).

Triple Product Selection Card:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
<th>31-40</th>
<th>41-50</th>
<th>51-60</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>( h_1 )</td>
<td>( k_1 )</td>
<td>( t_1 )</td>
<td>( h_2 )</td>
<td>( k_2 )</td>
<td>( t_2 )</td>
</tr>
<tr>
<td>formats</td>
<td>110</td>
<td>110</td>
<td>110</td>
<td>110</td>
<td>110</td>
<td>110</td>
</tr>
</tbody>
</table>
1. \( h_1, k_1, \ell_1 \) are the Miller indices of the reflection with normalized structure factor \( F_1 \) (see section 5.1).

2. \( h_2, k_2, \ell_2 \) are the Miller indices of the reflection with normalized structure factor \( F_2 \) (see section 5.1).

The deck consists of one control card followed by any number of triple product selection cards. Calculation is terminated by the second blank card read (the first blank card causes calculation of the triple for \( h_1, k_1, \ell_1 = 0 \)).

B. Reflection Data.

The first record on the reflection data input tape contains the number of reflection records to follow (format 17).

Each reflection record consists of:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-4</th>
<th>5-8</th>
<th>9-12</th>
<th>13-32</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>( h )</td>
<td>( k )</td>
<td>( \ell )</td>
<td>( F_2-1 )</td>
</tr>
<tr>
<td>formats</td>
<td>14</td>
<td>14</td>
<td>14</td>
<td>14 F 20.5</td>
</tr>
</tbody>
</table>

Tape input is from logical tape 9.

For many problems it may be necessary to recompile the source program in order that the dimension assignments on the triply indexed variable \( F \) be changed (see section 5.3).

5.3 Method of Calculation

The program stores each value of \((F_2^2 - 1)_{h,k,\ell}\) as the triply indexed FORTRAN variable \( F(h, k, \ell) \). The program then runs through the list of reflections in a triple nest of DO loops in order to select summands.

The orthorhombic symmetry operations on \( h, k, \ell \) are coded into the program.

Since the program uses \( h, k, \ell \) as indices, these quantities always must be positive. The program automatically rejects one- and two-dimensional data. Core size limits the maximum values of \( h, k, \ell \) rather severely. As listed in this report, the program accepts maximum values of \( h, k, \ell \) as large as 30, 10, 30. \( F \) is the only dimensioned variable in the programs, and most of the core is available to it; however it will frequently be necessary to recompile.

The only difference between the programs for centrosymmetric and noncentrosymmetric crystals is that for the former, the average is divided by \( N^{1/2}/8 \), whereas the latter is divided by \( N^{3/2}/2 \). (Note the third statement following statement number 8 in the FORTRAN listing given in Appendix A.)

Whereas only three-dimensional data are included in the summations, \( \tilde{h}_1, \tilde{h}_2, \tilde{h}_3 \) need not be three dimensional.

If none of the Miller indices of \( \tilde{h}_1, \tilde{h}_2, \tilde{h}_3 \) are zero, the program includes the correction term \((F_{h_1}^2 + F_{h_2}^2 + F_{h_3}^2 - 2)/N^{1/2}\) in the results.
5.4 Output

The output for each reflection includes:

a. \( h_3, h_1, h_2 \)
b. the number of terms in the sum.
c. the scaled triple product average without correction term.
d. the scaled triple product average with correction term.
e. the correction term.

5.5 Limitations

See section 5.3.

5.6 Special Subroutines Called

None.

5.7 Special Tape Requirements

Logical tape 9 is used for reflection input.

6. STRUCTURE FACTOR CALCULATION

6.1 Purpose

To provide a convenient means for calculation of structure factors and quasi-normalized structure factors for use during preliminary stages of structure determination. Reflection input is from the \( F^2 - 1 \) or \( F^2 \) output tapes of XRDDR. Isotropic temperature factors are used.

6.2 Input

The input may be divided into general and control information, and reflection data.

A. General and Control Information.

The cards in this part are included with the program deck.

1. 72 columns of Hollerith characters (format 12A6). These Hollerith characters are used to title the output.

2. Control Card No. 1:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
<th>31-40</th>
<th>41-50</th>
<th>51-50</th>
<th>61-70</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>IC</td>
<td>IE</td>
<td>NR</td>
<td>NF</td>
<td>IP</td>
<td>LIB</td>
<td>LINE</td>
</tr>
<tr>
<td>formats</td>
<td>I10</td>
<td>I10</td>
<td>I10</td>
<td>I10</td>
<td>I10</td>
<td>I10</td>
<td>I10</td>
</tr>
</tbody>
</table>
IC = 0 if crystal is centric; IC ≠ 0 if crystal is acentric.

IE = 0 if $F^2$ reflection data input is used; IE ≠ 0 if $E^2-1$ reflection data input is used.

NR is the number of input reflections (if $E^2-1$ reflection data is used, this may be left blank).

NF is the number of files to be spaced over if the reflection data are read from a library tape.

IP = 0: $E$ output mode is specified; IP ≠ 0: $F$ output mode is specified. IP is irrelevant for centric crystals (see section 6.5.B for a description of the modes).

LIB = 0: library tape not used for reflection input; LIB ≠ 0: library tape is used for reflection input.

Line = 0: reflection data is read from logical tape 10; LINE ≠ 0: reflection data is read from cards.

3. Cell Constant Card:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
<th>31-40</th>
<th>41-50</th>
<th>51-60</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>a</td>
<td>b</td>
<td>c</td>
<td>a</td>
<td>β</td>
<td>γ</td>
</tr>
<tr>
<td>formats</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
</tr>
</tbody>
</table>

$a, b, c, a, \beta, \gamma$ are the real-space unit cell parameters in units of angstroms and degrees.

4. Control Card No. 2:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
<th>31-40</th>
<th>31-50</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>BT</td>
<td>XT</td>
<td>SFC</td>
<td>KKT</td>
<td>IOUT</td>
</tr>
<tr>
<td>formats</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>110</td>
<td>110</td>
</tr>
</tbody>
</table>

BT and XT are the parameters $B$ and $X$ of the modified Wilson equation as taken from the output of XRDDR (if KKT = 1, these fields may be left blank).

SFC is a scale factor by which all $F_0$ are to be multiplied to bring them to the proper scale. It is irrelevant if $E_0$ data are input.

KKT has the value of 0, 1, or 2, which serves to select the desired structure factor normalization procedure as follows:

KKT = 0: apply exact normalization procedure to $F_c$; apply approximate normalization procedure to $F_0$ or $E_0$.

KKT = 1: apply exact normalization procedure to $F_c$; apply exact normalization procedure to $F_0$ or $E_0$.

KKT = 2: apply approximate normalization procedure to $F_c$; apply approximate normalization procedure to $F_0$ or $E_0$.

(see section 6.4 for explanations of exact and approximate normalization.)

If IOUT ≠ 0, a BCD output (logical tape 11) will be written (see section 6.6).
5. Scale factor cards.

If IE is 0 (i.e., F₀ reflection data are input), a deck of scale factor cards is input. Each card contains one scale factor (format E10.4). The deck is terminated with a blank card. The scale factors are indexed in the order loaded and are applied according to the procedure given in section 6.2.B under description of SCF.

6. Atomic scattering factor cards.

There is one card for each unique atomic species in the crystal:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-8</th>
<th>9-16</th>
<th>17-24</th>
<th>25-32</th>
<th>33-40</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>A</td>
<td>a</td>
<td>B</td>
<td>b</td>
<td>C</td>
</tr>
<tr>
<td>formats</td>
<td>F8.4</td>
<td>F8.4</td>
<td>F8.4</td>
<td>F8.4</td>
<td>F8.4</td>
</tr>
</tbody>
</table>

A, a, B, b, and C are the parameters used in the Vand, Eiland, Pepinsky approximation equation for calculation of atomic scattering factors as defined by Forsyth and Wells (3,7).

The deck of atomic scattering factor cards is terminated by a blank card.

The loading order of the atomic scattering factor cards is important and must be consistent with the quantities I on the coordinate cards (see section 6.2.A.7).

7. Coordinate cards.

There is one card for each atom in the asymmetric unit:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
<th>31-40</th>
<th>41-50</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>X</td>
<td>Y</td>
<td>Z</td>
<td>B</td>
<td>I</td>
</tr>
<tr>
<td>formats</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>I10</td>
</tr>
</tbody>
</table>

X, Y, and Z are the fractional atomic coordinates of one atom.*

B is the isotropic temperature factor.

I is an integer which relates the atom to one of the atomic scattering factor cards and thus specifies the atomic type. Each atomic scattering factor card is numbered 1, 2, 3,... in sequence as it is read into the computer. Thus, if the second scattering factor card to be read is for carbon, then I = 2 on all carbon coordinate cards.

The deck of coordinate cards is terminated with a blank card.

8. Transformation cards.

The atoms in the asymmetric unit are transformed according to the relations given in "International Tables for X-Ray Crystallography," Vol. 1, (N.F.M. Henry and K. Lonsdale, Birmingham: Kynoch Press, 1952) so as to fill out the unit cell. There is one card for each transformation.

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
<th>31-40</th>
<th>41-50</th>
<th>51-60</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>T1</td>
<td>T2</td>
<td>T3</td>
<td>U1</td>
<td>U2</td>
<td>U3</td>
</tr>
<tr>
<td>formats</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
</tr>
</tbody>
</table>

*If an atom is at the origin, X, Y, Z should be given the values 1, 1, 1 instead of 0, 0, 0.
The transformations are done as follows:

\[ X' = T_1 + U_1 \cdot X \]
\[ Y' = T_2 + U_2 \cdot Y \]
\[ Z' = T_3 + U_3 \cdot Z. \]

Example: Space group \( \text{P}2_1/\text{c} \)

Transformations: \( \bar{x}, \bar{y}, \bar{z}; x, 1/2 - y, 1/2 + z; \bar{x}, 1/2 + y, 1/2 - z. \)

Transformation cards

<table>
<thead>
<tr>
<th></th>
<th>T1</th>
<th>T2</th>
<th>T3</th>
<th>U1</th>
<th>U2</th>
<th>U3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>.0</td>
<td>.0</td>
<td>.0</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td></td>
<td>.0</td>
<td>.5</td>
<td>.5</td>
<td>1</td>
<td>-1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>.0</td>
<td>.5</td>
<td>.5</td>
<td>-1</td>
<td>1</td>
<td>-1</td>
</tr>
</tbody>
</table>

The deck of transformation cards is terminated with a blank card.

B. Reflection Data.

The reflection input may be included along with the program deck, following the general and control information cards, or be taken from logical tape 10. It may be taken directly from an XRDDR output tape or from a library tape, each file of which contains the data from an XRDDR input or output tape. In the latter case, the program assumes that each file is prefixed with one identification record.

The program calculates structure factors for each reflection in the reflection input.

If \( F_0^2 \) reflection input is used \((\sin \theta)/\lambda\) values may or may not be included (they are automatically included on the XRDDR \( F_0^2 \) output tapes). The format is the same as that used for input to the Busing least-squares program.

If \( E^2-1 \) reflection input is used, the first record must consist of the number of reflections to be read (format 17). No provision is made for input of \((\sin \theta)/\lambda\) data.

\( F^2 \) reflection data:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-9</th>
<th>10-18</th>
<th>19-27</th>
<th>28-36</th>
<th>37-45</th>
<th>46-54</th>
<th>55-64</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>h</td>
<td>k</td>
<td>( \ell )</td>
<td>( F_0 )</td>
<td>(blank)</td>
<td>SFC</td>
<td>((\sin \theta)/\lambda)</td>
</tr>
</tbody>
</table>

The quantity SCF is a floating point integer which relates the reflection to one of the scale factors described in section 6.2.A.5 above. The integer corresponds to the loading sequence number of the scale factor (i.e., the first scale factor loaded corresponds to SCF = 1.0, etc.). SCF may be punched on every card. Alternatively, the reflection input may be grouped into sets where all reflections in a set are multiplied by the same scale factor. In this case, it is sufficient to punch SCF in the first card only of each set. This is the same scaling procedure used in the Busing least-squares refinement program; however, in this program, the \( F_0 \) values are scaled.
**6.3 Data Deck**

The composition of the data deck is as follows:

a. Hollerith card.
b. control card no. 1.
c. cell constant card.
d. control card no. 2.
e. all scale factor cards (if \( IE \neq 0 \), there are no scale factor cards).
f. blank card.
g. all atomic scattering factor cards.
h. blank card.
i. all coordinate cards.
j. blank card.
k. all transformation cards (if any).
l. blank card.
m. reflection cards unless reflection input is from tape.
n. a card with 1 in column 7 or a blank card depending upon whether or not another calculation is to follow.

**6.4 Method of Calculation**

Structure factors are calculated from the total unit cell contents (rather than from the asymmetric unit contents) using the equations for the triclinic case. Nevertheless, coordinates may be entered for one asymmetric unit only, provided that transformations follow by which the asymmetric unit set is expanded to fill the unit cell. The transformations may be copied directly from the *International Tables for Crystallography,* Vol. I (1952). In this way, the program is completely general in its applicability while requiring the most simple type of input.

The quasi normalization of structure factors (and in the case where \( E^2-1 \) reflection input is used, the inverse normalization) may be done in either of two ways, designated as exact and approximate normalization.

1. Exact normalization:

\[
E^2 = \frac{F^2}{\sum_{i=1}^{N} f_{i}^2}
\]
2. Approximate normalization:

\[ g^2 = \frac{F^2 \exp \left[ BT(\sin^2 \theta / \lambda^2) \right]}{\sum_{i=1}^{N} f_{0,1}^2}, \]

where

\[ f_i = f_{0,1} \exp \left[ -B_i(\sin^2 \theta / \lambda^2) \right] \]

and \( N \) is the number of atoms in the unit cell.

6.5 Printed Output

A. Centric Crystals.

The output for each reflection is:

a. \( h,k,C \)

b. \( F_0 \)

c. \( F_c \)

d. \( E_0 \)

e. \( E_c \)

f. \( \Delta F \)

g. \( \Delta E \).

B. Acentric Crystals.

Either one of two modes of output may be selected:

1. F mode

a. \( h,k,C \)

b. \( E_0, E_c \)

c. \( F_0, F_c \)

d. \( A_0, B_0 \)

e. \( A_c, B_c \)

f. \( \Delta A, \Delta B \)

g. \( \Delta F \).

2. E mode

a. \( h,k,C \)

b. \( F_0, F_c \)

c. \( E_0, E_c \)

d. \( C_0, C_c \) where \( |E|^2 = C^2 + D^2 \)

e. \( D_0, D_c \)

f. \( \Delta C, \Delta D \)

g. \( \Delta E \).
In addition to the reflection output, the program prints $\Sigma|\Delta F|$, $\Sigma|F_0|$, and $R = \Sigma|\Delta F|/\Sigma|F_0|$ both including and excluding unobserved data.

6.6 Tape Output

If IOUT ≠ 0 the program writes a reflection output tape containing one of the following sets of data for each reflection (format 7F10.4):

a. Centrosymmetric crystals

\[ h, k, \ell, F_0, E_0, \Delta F, \Delta E \]

b. Noncentrosymmetric crystals

1. If IP = 0:
   \[ h, k, \ell, C_0, D_0, \Delta C, \Delta D \]

2. If IP ≠ 0:
   \[ h, k, \ell, A_0, B_0, \Delta A, \Delta B \]

where the observed quantities are scaled and given the signs of the corresponding calculated quantities.

6.7 Special Subroutines Called

a. OUTSFC. An output routine designed for these calculations only.

b. RECIP.

c. The FAP coded tape utility program written by P. Gum.*

d. INTG.

6.8 Special Tape Requirements

a. Logical 10 is used for reflection input.

b. Logical 11 is used for reflection output.

7. INTERATOMIC DISTANCES AND ANGLES WITH INTERPOLATION FOR PEAK CENTER LOCATION

7.1 Purpose

Given a basic set of atomic or peak coordinates and a set of symmetry transformations on these coordinates, calculate and list all interatomic distances $b$, where $0.7 \leq b \leq BMX$, and calculate all angles between connected pairs of distances for all possible combinations in any crystal.

*Diffraction Branch, Optics Division, NRL.
7.2 Input

The data input is from cards which are loaded immediately following the program deck. The cards are described in loading sequence as follows:

A. 72 columns of Hollerith characters (format 12A6). These Hollerith characters are used to title the output.

B. Cell Constant Card:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
<th>31-40</th>
<th>41-50</th>
<th>51-60</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>a</td>
<td>b</td>
<td>c</td>
<td>a</td>
<td>b</td>
<td>y</td>
</tr>
<tr>
<td>formats</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
</tr>
</tbody>
</table>

The quantities a, b, c, α, β, γ are the real-space unit cell parameters expressed in units of angstroms and degrees.

C. Bond Length Limit Card:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-20</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>BMX</td>
<td>AMX</td>
</tr>
<tr>
<td>formats</td>
<td>E10.4</td>
<td>E10.4</td>
</tr>
</tbody>
</table>

BMX.

Interatomic distances b for which 0.7 ≤ b ≤ BMX are considered acceptable and are tabulated for printing.

AMX.

Angles for all pairs of acceptable distances sharing a common atom, for which the distances of both members of the pair are less than or equal to AMX, are tabulated for printing.

[WARNING: When a large number of atoms or peaks are involved and BMX and AMX are both large (i.e., greater than about 3.0Å), the number of angles printed can be very large.]

D. Grid Interval Card:

If peak maxima coordinates are to be found by interpolation between grid points of a Fourier calculation result, this card contains the fractional grid increments in the directions of X, Y, and Z.

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>XINC</td>
<td>YINC</td>
<td>ZINC</td>
</tr>
<tr>
<td>formats</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
</tr>
</tbody>
</table>

For example, if the Fourier calculation was done in increments of 1/30, 1/60, and 1/120 along X, Y, and Z, respectively, then

XINC = .03333, YINC = .01666, and ZINC = .008333.

If interpolation is not to be done, this card must be blank.
E. Coordinate Card:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>X</td>
<td>Y</td>
<td>Z</td>
</tr>
<tr>
<td>formats</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
</tr>
</tbody>
</table>

X, Y, and Z are the fractional coordinates of an atom or peak. There is one card for each atom or peak in the basic (untransformed) set. If interpolation is to be done, each coordinate card is followed by an interpolation card (see section 7.2.F below). If interpolation is not to be done, the interpolation cards are absent from the deck.

The coordinate card deck is terminated by a blank card.

F. Interpolation Card.

Interpolation cards are included in the coordinate deck, one following each coordinate card, if and only if card D is not blank (see section 7.2.D).

The fractional coordinates of the grid point closest to a peak maximum in a Fourier calculation result are given on the coordinate card. Relative to this central grid point, designate the Fourier summation value at nearby grid points by \( P(i, j, k) \), where \( i, j, \) and \( k \) have integral values representing the (signed) numbers of grid interval separations of the point from the central one in the directions of \( X, Y, \) and \( Z \), respectively.

The interpolation card contains the following:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
<th>31-40</th>
<th>41-50</th>
<th>51-60</th>
<th>61-70</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>( P(0,0,0) )</td>
<td>( P(-1,0,0) )</td>
<td>( P(1,0,0) )</td>
<td>( P(0,-1,0) )</td>
<td>( P(0,1,0) )</td>
<td>( P(0,0,-1) )</td>
<td>( P(0,0,1) )</td>
</tr>
<tr>
<td>formats</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
</tr>
</tbody>
</table>

If the user wishes to interpolate some, but not all, points, then values of \( P(0,0,0) = 100.0 \) and all other \( P = 1.0 \) may be entered for those points that are not to be interpolated.

Note that \( P(0,0,0) \) must be the largest \( P \) on the card. Also, all \( P \) must be greater than zero. Negative or zero \( P \) can be accommodated by adding a constant increment to all \( P \) such that the results are all positive.

G. Transformation Card:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
<th>31-40</th>
<th>41-50</th>
<th>51-60</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>( T_1 )</td>
<td>( T_2 )</td>
<td>( T_3 )</td>
<td>( U_1 )</td>
<td>( U_2 )</td>
<td>( U_3 )</td>
</tr>
<tr>
<td>formats</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
</tr>
</tbody>
</table>

There is one transformation card for every symmetry transformation to be applied to the basic set of coordinates. The transformations are applied according to the equations:

\[
X' = T_1 + U_1 \cdot X \\
Y' = T_2 + U_2 \cdot Y \\
Z' = T_3 + U_3 \cdot Z.
\]

The transformation card is terminated by a blank card.
7.3 Data Deck

The composition of the data deck is as follows:

a. Hollerith card.
b. cell constant card.
c. bond length limit card.
d. grid interval card, or blank if interpolation is not desired.
e. all coordinate and interpolation cards.
f. blank card.
g. all transformation cards (if any).
h. blank card.

7.4 Method of Calculation

The basic set of coordinates are loaded, interpolated (if desired), and then each transformation is applied to the basic set, generating a transformed set for each transformation. Thus, if there are N atoms in the basic set and there are M transformations, the complete set contains N(M + 1) atoms.

Interatomic distances less than or equal to BMX are tabulated for all pairs of atoms in the basic set and for all distances (≤ BMX) which cross the transformed set boundaries for the complete set.

Atom pairs closer together than 0.7Å are listed in a separate output and are rejected from further consideration.

Next, the list of accepted interatomic distances is scanned to find pairs of interatomic distances which share a common atom. Angles for all such pairs of distances are tabulated if both distances are less than or equal to AMX. Logical tape 9 is used for intermediate storage of angles.

Interpolation is done by fitting the seven Fourier grid points closest to the peak maximum to a Gaussian function and then finding the maximum of the function. The corrections, ΔX, ΔY, and ΔZ to be added to X, Y, and Z are found by solving the matrix

\[
\begin{pmatrix}
\hat{a} & \hat{a} & \hat{a} & \hat{a} & \hat{a} & \hat{a} & \hat{a} \\
\hat{b} & \hat{b} & \hat{b} & \hat{b} & \hat{b} & \hat{b} & \hat{b} \\
\hat{c} & \hat{c} & \hat{c} & \hat{c} & \hat{c} & \hat{c} & \hat{c}
\end{pmatrix}
\begin{pmatrix}
Δx \\
Δy \\
Δz
\end{pmatrix}
= \begin{pmatrix}
D_1 \\
D_2 \\
D_3
\end{pmatrix}
\]

where

\[
D_x = \frac{a^2}{2} \left[ \log \frac{P(1,0,0)}{P(-1,0,0)} \right] \left[ \log \frac{P(0,0,0)}{P(1,0,0)} + \log \frac{P(0,0,0)}{P(-1,0,0)} \right],
\]

etc.

7.5 Output

In the output, each atom is represented by two integers separated by a hyphen. The first integer specifies the transformation (numbered according to loading order) which has been applied, and the second integer specifies the atom in the basic set (numbered according to loading order) from which it was derived.
The output is in the following order:

a. Atomic coordinates for the complete set.
b. Interatomic distances.
c. Angles.
d. Interatomic distances less than 0.7Å.

7.6 Limitations

a. The complete set of atoms or peaks is limited to 500 in number.

b. If more than 2900 acceptable interatomic distances are found, BMX is decremented by 0.25Å and the calculation is begun again. This process is repeated until a total of 2900, or less, acceptable distances are found, or until BMX $< 2.0$. In the latter case, distances and angles are calculated for the basic set of atoms only. If more than 2900 acceptable distances are found in the basic set, the calculation is terminated without output of results.

c. No more than 200 distances less than 0.7Å can be accommodated.

d. There is no limit on the number of bond angles.

7.7 Special Subroutines Called

a. subroutine INTERP. This subroutine does the interpolation by Gaussian curve fitting.

b. subroutine MATS (SHARE distr. no. 635).

c. subroutine OUTBND: (distance and angle output).

d. function DOTPRD.

e. function ARCSIN.

7.8 Special Tape Requirements

Logical Tape 9 is used for intermediate storage.

8. LEAST-SQUARES PLANE AND LINE FITTER

8.1 Purpose

This program fits planes and/or lines to sets of points and calculates angles between planes and lines which have been fitted.

8.2 Input

a. 72 columns of Hollerith characters (format 12A6). These Hollerith characters are used to title the output.
b. Unit Cell Card:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
<th>31-40</th>
<th>41-50</th>
<th>51-60</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>a</td>
<td>b</td>
<td>c</td>
<td>a</td>
<td>β</td>
<td>γ</td>
</tr>
<tr>
<td>formats</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
</tr>
</tbody>
</table>

The quantities $a, b, c, \alpha, \beta, \gamma$ are the real-space unit cell parameters expressed in units of angstroms and degrees.

c. Plane or Line Specification Card:

<table>
<thead>
<tr>
<th>columns</th>
<th>10-3</th>
<th>4-6</th>
<th>7-12</th>
<th>13-72</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>N</td>
<td>NO</td>
<td>L</td>
<td>Hollerith</td>
</tr>
<tr>
<td>formats</td>
<td>13</td>
<td>13</td>
<td>16</td>
<td>10A6</td>
</tr>
</tbody>
</table>

N is the number of points to be fitted.

NO is the plane or line number.

L = 0 or blank if the points are to be fitted to a plane.

L = 1 if the points are to be fitted to a line.

The Hollerith characters in columns 13-72 are used as a page heading for the line or plane in the output.

d. Coordinate Card:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
<th>31-40</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>X</td>
<td>Y</td>
<td>Z</td>
<td>W</td>
</tr>
<tr>
<td>formats</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
</tr>
</tbody>
</table>

X, Y, and Z are fractional coordinates of a point to be fitted to a line or plane.

W is a least-squares weighting factor. If W is not punched, it is assumed to have a value of one by the program.

e. Angle specification card.

This card specifies a plane-plane, line-line, plane-line, or line-plane pair for which the dihedral angle is to be calculated.

<table>
<thead>
<tr>
<th>columns</th>
<th>1-3</th>
<th>4-6</th>
<th>7-9</th>
<th>10-12</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>NO1</td>
<td>L1</td>
<td>NO2</td>
<td>L2</td>
</tr>
<tr>
<td>formats</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
</tr>
</tbody>
</table>

NO1 is the number of the first plane or line.

L1 = 0 or blank if NO1 refers to a plane.

L1 = 1 if NO1 refers to a line.

NO2 is the number of the second plane or line.

L2 = 0 or blank if NO2 refers to a plane.

L2 = 1 if NO2 refers to a line.
The plane and line numbers NO are used to identify the different planes and lines both internally and in the output. The numbering may be different from the order of input. No two planes may have the same number, and no two lines may have the same number, but the numbering of lines and planes is independent. The number of any plane or line must not be greater than 50.

8.3 Data Deck

The composition of the data deck is as follows:

a. Hollerith card.
b. unit cell card.
c. plane or line specification card.
d. all coordinate cards for this plane or line (the number of coordinate cards must be the same as the N punched on the preceding plane or line specification card).
e. plane or line specification card.
f. all coordinate cards for this plane or line.
g. additional sets of specification and coordinate cards for as many planes and lines as are to be fitted.
h. blank card.
i. all angle specification cards, if any.
j. blank card.

8.4 Method of Calculation

The method described in detail by Shomaker, Waser, Marsh, and Bergman (8) is used.

The program is written in the fullest generality so as to be able to handle points defined in any three-dimensional coordinate system. However, no provision is made for symmetry transformation of point positions, so all points must be properly transformed before input.

For a given coordinate system, the program sequentially fits all desired planes and lines. It stores the appropriate unit vectors for the planes and lines by plane and line number. The program can then calculate dihedral angles between all pairs of planes, lines, or plane-line combinations that are specified.

8.5 Output

The output is largely self explanatory. Specifically, the following information is given:

a. equations of the planes and lines.
b. for planes, the coordinates of the centroid.
c. perpendicular distances of each point from the least-squares planes and lines.
d. average deviations, standard deviations, etc.
24 NAVAL RESEARCH LABORATORY

e. cosines of angles between plane-plane and line-line pairs.
f. sines of angles between line-plane pairs.

Each point is numbered in the output according to loading order. Thus, the Hollerith section of the plane or line specification card (see section 8.2.c) may be used, in part, to relate the points to atom numbers or other external designations.

8.6 Limitations

a. No more than 50 points may be fitted to any plane or line.
b. No less than three points may be fitted to a plane.
c. No less than two points may be fitted to a line.
d. There may be no more than 50 planes and no more than 50 lines.
e. The number of a line or plane may not be larger than 50.

8.7 Special Subroutines Called

a. subroutine RECIP.
b. subroutine MTXMUL.
c. function DOTPRD.

8.8 Special Tape Requirements

None.

9. POINT TO PEAK DISTANCE CALCULATION

9.1 Purpose

This program provides a means for calculating distances from one or several points to a set of points in any three-dimensional coordinate system.

9.2 Input

a. 72 columns of Hollerith characters (format 12A6). These characters are used to title the output.
b. Unit Cell Card:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
<th>31-40</th>
<th>41-50</th>
<th>51-60</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>a</td>
<td>b</td>
<td>c</td>
<td>a</td>
<td>β</td>
<td>γ</td>
</tr>
<tr>
<td>formats</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
</tr>
</tbody>
</table>

The quantities $a, b, c, \alpha, \beta, \gamma$ are the real-space unit cell parameters in units of angstroms and degrees.
c. Maximum distance card.

This card contains the single quantity BMX. All distances $d$ are tabulated for which $d \leq$ BMX (format E10.4).

d. Point Card:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>XC</td>
<td>YC</td>
<td>ZC</td>
</tr>
<tr>
<td>formats</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
</tr>
</tbody>
</table>

XC, YC, and ZC are the fractional coordinates of the point for which distances to the set of points, specified by the coordinate cards and their transformations, are to be calculated.

e. Coordinate card

These cards contain the fractional coordinates X, Y, and Z of the basic set of points, which gives rise to a complete set via the transformations (see card below), whose distances to point XC, YC, ZC are desired. The card layout is the same as that already given for the point card above.

f. Transformation Cards:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-30</th>
<th>21-30</th>
<th>31-40</th>
<th>41-50</th>
<th>51-60</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>T1</td>
<td>T2</td>
<td>T3</td>
<td>U1</td>
<td>U2</td>
<td>U3</td>
</tr>
<tr>
<td>formats</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
<td>E10.4</td>
</tr>
</tbody>
</table>

The function of these quantities is explained in section 7.2.G.

9.3 Data Deck

a. Hollerith card.
b. unit cell card.
c. maximum distance card.
d. point card.
e. all coordinate cards.
f. blank card.
g. all transformation cards.
h. blank card.
i. additional point cards for as many calculations as desired.
j. blank card.

9.4 Method of Calculation

This program is an abbreviation of the Interatomic Distance and Angle Program (section 7). It does not provide for interpolation or angle calculation.
9.5 Output

The output of distances is essentially the same as that described for the Interatomic Distance and Angle Program (section 7.5).

9.6 Limitations

a. The complete set of points must not exceed 500 in number.

b. A maximum of 5000 distances can be accommodated.

9.7 Special Subroutines Called

a. DOTPRD.

9.8 Special Tape Requirements

None.

10. FORM FACTORS FOR THE BUSING LEAST-SQUARES REFINEMENT PROGRAM

10.1 Purpose

Provide an automatic process for preparing form factor cards used for input to the Busing least-squares refinement program ORXLS (1).

10.2 Input

a. One blank card.

b. Form Factor Parameter Card:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-8</th>
<th>9-16</th>
<th>17-24</th>
<th>25-32</th>
<th>33-40</th>
<th>41-46</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>A</td>
<td>a</td>
<td>B</td>
<td>b</td>
<td>C</td>
<td>Hollerith</td>
</tr>
<tr>
<td>formats</td>
<td>F8.5</td>
<td>F8.5</td>
<td>F8.5</td>
<td>F8.5</td>
<td>F8.5</td>
<td>A8</td>
</tr>
</tbody>
</table>

The quantities A, a, B, b, C are the parameters, as defined by Forsyth and Wells (3), used in calculating the form factors. The six columns of Hollerith characters are used as identification on the output cards.

10.3 Data Deck

The composition of the data deck is as follows:

a. blank card.

b. as many form factor cards as desired.

c. blank card.
10.4 Method of Calculation

The analytical approximation of Vand, Eiland, and Pepinsky (7), is used as extended by the work of Forsyth and Wells (3).

The approximations are good only as far as \((\sin \theta)/\lambda \approx 1.40\) (the range of \(M_K\) radiation).

These approximations are not recommended for highly accurate work.

10.5 Output

The form factor card images are written on logical tape 10 for peripheral punching. For each atomic species, five cards are punched as follows:

a. a blank separator card.

b. four cards containing the form factor values in intervals of 0.05 in \((\sin \theta)/\lambda\) from 1.55 to 0, in that order.

Each card has an F in column 72. Columns 73-78 contain the Hollerith characters specified in the form factor parameter cards. Column 80 contains the sequence number of the card, i.e., 1,2,3, or 4.

Essentially the same information also is printed.

10.6 Special Subroutines Called

None.

10.7 Special Tape Requirements

a. Logical tape 10 for peripheral punching.

11. VARIANCE-COVARIANCE MATRIX AND ATOMIC COORDINATE INPUT FOR THE BUSING FUNCTION AND ERROR PROGRAM

11.1 Purpose

This program produces two BCD card decks containing, respectively:

a. The variance-covariance matrix for a set of atomic position coordinates, with all covariance elements given a value of zero.

b. Atomic coordinates.

These cards are ready for input to the Busing Function and Error Program ORXFE (2).

11.2 Input

a. Preliminary Card:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-9</th>
<th>10-19</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>N</td>
<td>SF</td>
</tr>
<tr>
<td>formats</td>
<td>I9</td>
<td>E10.5</td>
</tr>
</tbody>
</table>
N is the order of the variance-covariance matrix, i.e., \( N = 3 \times \) (number of atoms).

SF is a scale factor by which the matrix is to be multiplied. Specifically,
\[
SF = \frac{\Sigma \omega (F_0 - F_c)^2}{(m - n)}, \text{ or unity.}
\]

b. Unit Cell Card:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>a</td>
<td>b</td>
<td>c</td>
</tr>
<tr>
<td>formats</td>
<td>E10.5</td>
<td>E10.5</td>
<td>E10.5</td>
</tr>
</tbody>
</table>

If the standard deviations on the cards described below are in fractional (dimensionless) form, \( a, b, \) and \( c \) all have values of one.

If the standard deviations are in units of angstroms, then \( a, b, \) and \( c \) are the unit cell edge lengths in units of angstroms.

c. Standard Deviation Card:

<table>
<thead>
<tr>
<th>columns</th>
<th>1-10</th>
<th>11-20</th>
<th>21-30</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>( \sigma (X) )</td>
<td>( \sigma (Y) )</td>
<td>( \sigma (Z) )</td>
</tr>
<tr>
<td>formats</td>
<td>E10.5</td>
<td>E10.5</td>
<td>E10.5</td>
</tr>
</tbody>
</table>

The quantities \( \sigma (X), \) \( \sigma (Y), \) and \( \sigma (Z) \) are standard deviations or uncertainties of the atomic coordinates \( X, Y, \) and \( Z. \) These quantities may be dimensionless or may have units of angstroms (see section 11.2.b).

d. Coordinate card.

The layout of the coordinate cards is the same as for the standard deviation cards. Each coordinate card contains the fractional coordinates \( X, Y, \) and \( Z \) of an atom whose standard deviations are given in the corresponding card of the standard deviation deck.

11.3 Data Deck

a. preliminary card.
b. unit cell card.
c. all standard deviation cards.
d. all coordinate cards.

d. Coordinate card.

Output may be on logical tape 9 for peripheral punching or, if sense switch 1 is down, the cards are punched on-line.

Each matrix card is numbered sequentially and identified with the letters SD (format 8E9.4, 14, MH SD).

Each atomic coordinate card is numbered sequentially and identified with the letter P (format 8F9.6, 14, 24 P).

The two decks are separated by a blank card.
11.5 Limitations

The program will accommodate no more than 64 atoms.

11.6 Special Subroutines Called

None.

11.7 Special Tape Requirements

Logical tape 9 is used for peripheral punching.

12. OTHER FUNCTIONS AND SUBROUTINES

12.1 Function ARCSIN(X)

This program uses Hasting's Chebyshev approximation (sheet 39 of Ref. 9).

If |X| > 1, the program calls a subroutine ENDJOB which may be constructed by the user to suit his purposes.

The program was coded by B. A. Schoomer.*

12.2 Subroutine RECIP (AR, BR, CR, ALR, BER, GAR, AA, BB, CC, COSAL, COSBE, COSGA)

Given the sets of three translational and three angular parameters for a three-dimensional space lattice, this subroutine finds the parameters for the corresponding reciprocal space lattice.

Given Lattice: Reciprocal Lattice:

![Diagram of lattice transformation]

AR, BR, CR, ALR, BER, GAR → AA, BB, CC, AL, BE, GA

AR, BR, and CR may be given in any convenient units; ALR, BER, and GAR, are given in degrees.

*Diffraction Branch, Optics Division, NRL.
12.3 Function DOTPRD (U, V, W, X, Y, Z, B1, B2, B3, COSA, COSB, COSC)

This function calculates the scalar product of two vectors U and V where \( U = U(U, V, W) \) and \( V = V(X, Y, Z) \).

The vectors U and V are defined on a skew-vector basis B1, B2, B3

\[
\mathbf{U} = U_1 \mathbf{B}_1 + U_2 \mathbf{B}_2 + U_3 \mathbf{B}_3 \\
\mathbf{V} = V_1 \mathbf{B}_1 + V_2 \mathbf{B}_2 + V_3 \mathbf{B}_3
\]

such that

\[
\mathbf{U} = U_1 \mathbf{B}_1 + U_2 \mathbf{B}_2 + U_3 \mathbf{B}_3 \\
\mathbf{V} = V_1 \mathbf{B}_1 + V_2 \mathbf{B}_2 + V_3 \mathbf{B}_3
\]

and

\[
\mathbf{U} \cdot \mathbf{V} = U_1 V_1 (\mathbf{B}_1)^2 + U_2 V_2 (\mathbf{B}_2)^2 + U_3 V_3 (\mathbf{B}_3)^2 + (V_3 + Y V_2) B_2 B_3 \cos A + \\
(U_3 + W U_2) B_1 B_3 \cos B + (U_2 + V U_1) B_1 B_2 \cos C.
\]

12.4 Function SCAFAC (A, A1, B, B1, C, S)

This function calculates an atomic scattering factor using the analytic approximation of Vand, Eiland, and Pepinsky(6).

The scattering factor \( f(s) \) is

\[
f(s) = A \exp(-A_1 S^2) + B \exp(-B_1 S^2) + C.
\]

The quantities A, A1, B, B1, and C are taken from the tables of Forsyth and Wells (2). S is \((\sin \theta)/\lambda\).

12.5 Subroutine MTXMUL (L, M, N, A, B, C)

This subroutine performs matrix multiplication according to the equation

\[
AB = C
\]

where

the number of rows in A is L
the number of columns in A is M
the number of columns in B is N.

(As listed in this report, the dimensions of all matrices are 3 x 3; however, these may be changed to any other values needed.)
12.6 Function INTG(A)

This program converts floating point integers to fixed point integers without truncation error. This is done in a manner such that the integer value zero is not printed with a leading minus sign.

REFERENCES

APPENDIX A

FORTRAN LISTINGS OF GENERAL UTILITY PROGRAMS APPLICABLE TO CRYSTAL STRUCTURE ANALYSIS

In the form presented in this appendix, the FORTRAN-language programs are written for use with the IBM 7090 IB Monitor system.

<table>
<thead>
<tr>
<th>Program</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Least-Squares Cell Parameter</td>
<td>33</td>
</tr>
<tr>
<td>Subroutine DET</td>
<td>34</td>
</tr>
<tr>
<td>Subroutine PARAM</td>
<td>35</td>
</tr>
<tr>
<td>Subroutine ERRPRM</td>
<td>36</td>
</tr>
<tr>
<td>Subroutine ERRREL</td>
<td>37</td>
</tr>
<tr>
<td>Subroutine OUTPRM</td>
<td>37</td>
</tr>
<tr>
<td>Quasi Normalization of Structure Factors</td>
<td>39</td>
</tr>
<tr>
<td>Rational Dependence</td>
<td>41</td>
</tr>
<tr>
<td>Subroutine OUTRAT</td>
<td>43</td>
</tr>
<tr>
<td>Sigma-2 Listings</td>
<td>44</td>
</tr>
<tr>
<td>Subroutine OUTSIG</td>
<td>45</td>
</tr>
<tr>
<td>Subroutine REJVEC</td>
<td>46</td>
</tr>
<tr>
<td>Subroutine SIGVEC, Triclinic</td>
<td>46</td>
</tr>
<tr>
<td>Subroutine SIGVEC, Monoclinic</td>
<td>47</td>
</tr>
<tr>
<td>Subroutine SIGVEC, Orthorhombic</td>
<td>48</td>
</tr>
<tr>
<td>Triple Product Summation, Centrosymmetric</td>
<td>48</td>
</tr>
<tr>
<td>Triple Product Summation, Noncentrosymmetric</td>
<td>50</td>
</tr>
<tr>
<td>Structure Factor Calculation</td>
<td>52</td>
</tr>
<tr>
<td>Subroutine OUTSFC</td>
<td>56</td>
</tr>
<tr>
<td>Interatomic Distance and Angle</td>
<td>57</td>
</tr>
<tr>
<td>Subroutine INTERP</td>
<td>60</td>
</tr>
<tr>
<td>Subroutine OUTBND</td>
<td>60</td>
</tr>
<tr>
<td>Least-Squares Line and Plane Fitter</td>
<td>63</td>
</tr>
<tr>
<td>Point to Peak Distance Calculation</td>
<td>67</td>
</tr>
<tr>
<td>Form Factors for Busing Least Squares</td>
<td>68</td>
</tr>
<tr>
<td>Variance-Covariance and Atomic Parameter Input for Busing Function and Error</td>
<td>69</td>
</tr>
</tbody>
</table>
Program (Cont'd)

Function ARCSIN(X) 70
Subroutine RECIP 71
Function DOTPROD 71
Function SCAFAC 71
Subroutine MTXMUL 71
Function INTG(A) 71

Least-Squares Cell Parameter

```plaintext
DIMENSION NAME(10), M(3,15), W(150), S(150), SC(150),
1 H(3,3,150), G(3,3), SG(3,3), BE(3,3), AL(3,3), B(3), A(3), SA(3), SAL(3,3)
2 )
169

ITP=5
JTP=6
50 MM=1
READ INPUT TAPE ITP*1000, N1, LIST, YAM*(NAME(I), I=1,10)
IF(N1)500, 500, 100
100 READ INPUT TAPE ITP*2000, M(1,MM), M(2,MM), M(3,MM), S(MM), W(MM)
IF(XABSF(M(1,MM)) + XABSF(M(2,MM)) + XABSF(M(3,MM))) > 140, 140, 110
110 S2(MM) = 4.0*(S(MM)**2)
IF(W(MM))120, 120, 130
120 W(MM) = S(MM)
130 MM = MM + 1
GO TO 100
140 MM = MM - 1
IF(YAM)145, 145, 141
141 YAM = YAM**2
DO 142 I = 1, MM
142 S2(I) = S2(I)/YAM
145 NN = 1
DO 147 I = 1, MM
147 H(1,I) = 0.0
GO TO(310, 240, 210, 180, 150)*N1
150 N = 1
DO 160 I = 1, MM
GO TO 325
180 N = 2
NN = 2
DO 190 I = 1, MM
190 H(I,I) = M(I,I)**2 + M(2,I)**2 + M(3,I)**2
GO TO 325
210 N = 2
DO 220 I = 1, MM
GO TO 325
240 N = 3
GO TO 315
290 N = 4
GO TO 315
310 N = 6
```

02691
Subroutine DET

C SHARE DISTRIBUTION NO. 695
SUBROUTINE DET(A,ALPHA,N,BET) 02691
DIMENSION A(6*6) 02691
BET=1. 02691
IF(N-1)300,200,200 02691
200 BET = A(1,1) 02691
GO TO 400 02691
300 A(1,1) = A(1,1) - ALPHA 02691
GO TO 400 02691
6 DO 15 I=2,N 02691
A(I,1) = A(I,1) - ALPHA 02691
15 CONTINUE 02691
70 IF(N-1)300,200,200 02691
300 CALL ERRREL(GSG,BEAALN,S,A,SAL,SVR,LL) 02691
VR = 1.0/V 02691
CALL OUTPRM(A,AL,NAME,EB,VR,S,A,SAL,SVR,LIST,MM,N,LL,F,SC,S2,W,M) 02691
1,N1,JTP) 02691
GO TO 50 02691
500 CALL EXIT 02691
1000 FORMAT(2I2,EB,5,10A6) 02691
2000 FORMAT(3I4,2E10.5) 02691
3000 FORMAT(1H1,3X,98H THERE ARE NOT ENOUGH DEGREES OF FREEDOM. UNIT02691
1 CELL PARAMETERS AND ERRORS CANNOT BE CALCULATED.) 02691
END 02691

Subroutine DET
Subroutine PARAM

SUBROUTINE PARAM(N,H,W,G,S,SC,H,LL) 02691
DIMENSION H(3,3,150),W(150),G(3,3),S(150),SC(150)
I A(6*6),B(6*6),A(6*6),X(6*6) 02691
IF(N-3)362,365,365 02691
362 LL = N 02691
GO TO 330 02691
365 LL = 3 02691
330 DO 360 I = 1,LL 02691
DO 360 J = 1,LL 02691
IF(I-J)350,340,350 02691
340 D(I,J) = 1.0 02691
GO TO 360 02691
350 D(I,J) = 0.0 02691
360 CONTINUE 02691
367 DO 440 I = 1,LL 02691
DO 440 J = 1,LL 02691
DO 440 K = 1,LL 02691
DO 440 L = K,LL 02691
IF(J-L)410,400,410 02691
400 B(I,J) = B(I,J) 02691
GO TO 420 02691
410 B(I,J*K) = B(I,J) 02691
420 A(I,J*K) = A(I,J*K) 02691
B(I,J) = 0.0 02691
GO TO 430 02691
430 DO 450 I = 1,MM 02691
A(I,J*K) = A(I,J*K) + W(I)*H(I,J)*H(K,L)*H(I,J)*H(K,L) 02691
450 B(I,J) = B(I,J) + W(I)*S2(I)*H(I,J)*H(K,L)*H(I,J) 02691
A(I,J*K) = A(I,J*K) 02691
GO TO 440 02691
440 AA(I,J*K) = A(I,J*K) 02691
B(I,J) = B(I,J) 02691
GO TO 460 02691
460 CALL MATS(AA,X,S,N+1) 02691
GO TO 480 02691
SUBROUTINE ERRPRM(W,H,F,SC,SG,MM,GS2,ALL,D)
DIMENSION H(3,3),SC(150),SG(393),G(393),S2(150),A(6,6),AA(6,6)
1 )W(150)*SK(6)*D(3,3)
F = 0.0
DO 490 L = 1,MM
SC(L) = 0.0
DO 480 I = 1,LL
DO 480 J = I,LL
SC(L) = SC(L) + H(I,J)*G(I,J)*(2*O-D(I,J))
480 F = F + W(L)*(SC(L) - S2(L))**2
V = MM - N
IF(N-1)500,505,505
500 SG(I,J) = F/V
GO TO 680
505 DO 506 I = 1,N
DO 506 J = 1,N
AA(I,J) = A(I,J)
CALL DET(AA,0,0,NTED)
N2 = N - 1
DO 640 K = 1,N
DO 570 I = 1,N
DO 570 J = 1,N
IF(K-I)520,570,540
510 II = I + J
GO TO 530
520 II = I - 1
530 IF(K-J)550,570,540
540 JJ = J
GO TO 560
550 JJ = J - 1
560 AA(II,JJ) = A(I,J)
570 CONTINUE
CALL DET(AA,0,0,N2*TED)
640 SK(K) = (F*TED)/(TED*V)
DO 670 I = 1,LL
DO 670 J = 1,LL
IF(I-J)660,670,660
650 SG(I,J) = SK(I)
GO TO 670
660 K = 1+I+J
SG(I,J) = SK(K)
670 CONTINUE
680 RETURN
END
Subroutine ERRREL

SUBROUTINE ERRREL(G, SG, BE, A, AL, N, SA, SAL, VS, SVR, LL)
DIMENSION G(3, 3), SG(393), BE(3, 3), A(3), AL(3, 3), SA(3), SAL(3, 3), U(3)
V = G(1, 1)* (G(2, 2)* G(3, 3) + 2*G(2, 3)* G(1, 3) + G(1, 2) - G(1, 1)* G(1, 1)* (G(2, 2) ** 2) - G(2, 2)* (G(1, 3) ** 2) - G(3, 3)* (G(1, 2) ** 2))
V = SQRTF(V)
SZ = 0.0
DO 5 I = 1, LL
5 U(I) = 0.0
IF(N-1)10, 20, 0
DO 10 SA(I) = (SG(I, 1) + (A(I)/G(I, 1)) ** 2) / 2.0
GO TO 380
10 SA(I) = (SG(I, 1) - (A(I)/G(I, 1)) ** 2) / 2.0
DO 310 J = 1, LL
DO 310 K = 1, LL
IF(I - J)100, 310, 100
100 IF(I - K)200, 310, 310
200 IF(J - K)300, 310, 310
300 U(I) = 2.0* SG(J, K) - (G(J, K)/(G(J, J)* G(K, K)) ** 2)* SG(K, K)* SG(J, J) + (G(J, J)* G(K, K)) ** 2)* U(I) + (G(J, J)* G(K, K)) * SQRTF(SA(I))
CONTINUE
310 CONTINUE
SZ = SZ/(2.0* U** 4))
D = 1.0
IF(N-4)330, 320, 330
320 D = -D
330 DO 370 I = 1, LL
DO 370 J = 1, LL
DO 370 K = 1, LL
IF(LL-2)335, 360, 335
335 IF(I-J)340, 370, 340
340 IF(I-K)350, 370, 350
350 IF(J-K)360, 370, 360
360 SA(I) = ((D*(G(J, K)/(G(J, J)* G(K, K)) ** 2)) * U(I) + (A(I)) ** 2)* SG(I, I)
1/(G(I, I)) ** 2) + 2.0* (SZ)) / 2.0
SA(J, K) = (I*0 - AL(J, K)) ** 2)* SG(I, I)* U(I) + (BE(I, J)) ** 2 + (BE(I, K)) ** 2 + (AL(J, J)) ** 2)* (1.0 - BE(I, J)) ** 2) / (1.0 - BE(I, K)) ** 2)) * (G(J, J) ** 2)
K ** 2)* (1.0 - BE(I, J)) ** 2)) + (AL(J, J)) ** 2)* (1.0 - BE(I, K)) ** 2)) * G(K, K) * U(K) / (2.0* U** 2))
CONTINUE
370 CONTINUE
380 SVR = SQRTF(SZ)
420 DO 430 I = 1, 3
430 SVR = SVR + SG(I, I)/ G(I, I) ** 2)
DO 435 I = 1, LL
435 SVR = SQRTF(SVR/(V** 2))
RETURN
END

Subroutine OUTPRM

SUBROUTINE OUTPRM(A, AL, NAME, B, EB, VS, VR, SA, SAL, SVR, LIST, MM, N, LL, F, SC)
DIMENSION NAME(10), M(3, 150), W(150), S2(150), SC(150), AL(3, 3), B(3)
1A(3), SA(3), SAL(3, 3), EB(3), ALD(3), SALD(3)
IF(N-4)95, 90, 95
90 DO 100 I = 1, 3
100 DO 100 J = 1, 3
100 L = 6 - I - J
60 ALD(I) = 90.0 - 57.295780* ARCSIN(AL(I, J))
70 D = SQRTF(SAL(I+J))
100 SALD(L) = 57.295780*ABSF(ABSF(Arcsin(AL(I+J)+D)) - ABSF(Arcsin(AL(I+J-D)))
95 WRITE OUTPUT TAPE JTP*1000*(NAME(I),I=1,10),MM
      DO 170 I = 1+N
      GO TO(101+102*96)*LL
      IF(N-4)*103+97*103
      GO TO(110+120*130*160)+I
      101 GO TO 110
      102 GO TO(110+125)*I
      103 GO TO(110+120*130*140*150*160)*I
      130 WRITE OUTPUT TAPE JTP*1100*(B(3)+A(3)+SA(3))
      GO TO 170
      120 WRITE OUTPUT TAPE JTP*1200*(B(2)+A(2)+SA(2))
      GO TO 170
      125 WRITE OUTPUT TAPE JTP*1100*(B(2)+A(2)+SA(2))
      GO TO 170
      110 WRITE OUTPUT TAPE JTP*1300*(B(1)+A(1)+SA(1))
      GO TO 170
      160 WRITE OUTPUT TAPE JTP*1400*(B(3)+A(3)+SA(3))
      GO TO 170
      150 WRITE OUTPUT TAPE JTP*1500*(B(2)+A(2)+SA(2))
      GO TO 170
      140 WRITE OUTPUT TAPE JTP*1600*(B(1)+A(1)+SA(1))
      GO TO 170
      170 CONTINUE
      WRITE OUTPUT TAPE JTP*1700*VR*SVR
      WRITE OUTPUT TAPE JTP*1800
      GO TO(500+510*520+530*540+550)+N1
      500 WRITE OUTPUT TAPE JTP*2400
      GO TO 175
      510 WRITE OUTPUT TAPE JTP*2500
      GO TO 175
      520 WRITE OUTPUT TAPE JTP*2600
      GO TO 175
      530 WRITE OUTPUT TAPE JTP*2700
      GO TO 175
      540 WRITE OUTPUT TAPE JTP*2800
      GO TO 175
      550 WRITE OUTPUT TAPE JTP*2900
      175 IF(LIST*260+180*260)
180 R = MM - N
      SD = SQRTF(F/R)
      SD2 = SD*2.5
      L1 = MM/20
      L2 = MM - L1*20
      L3 = 0
      185 L3 = L3+1
      IF(L1 - L3*200*190*190
190 K2 = L3*20
      K1 = K2 -19
      GO TO 220
195 L3 = L3+1
      200 IF(L2*260+260*210
      210 K1 = L1*20 +1
      K2 = K1 + L2 -1
      220 NM = 0
      WRITE OUTPUT TAPE JTP*2000*(L3*SD
      DO 250 I = K1*K2
      DEL = (S2(I)+SC(I))*SQRTF(W(I))
      IF(ND2-ABSF(DEL))230*230+240
      230 NM = 1
      WRITE OUTPUT TAPE JTP*2100*(M(I)+M(2)+M(3)+S2(I)+SC(I)+W(I))
240 WRITE OUTPUT TAPE JTP, 2200, M(1,1), M(1,2), M(3,1), S2(I), SC(I), W(I),
10 DEL
250 CONTINUE 02691
255 WRITE OUTPUT TAPE JTP, 2300 02691
256 IF(1.L.13) 2560 +I1059185 02691
260 RETURN 02691
1000 FORMAT(H1, /, H4, 36X, 45HUNIT CELL PARAMETERS WITH STANDARD
15//29X10A6//46X+23HNUMBER OF REFLECTIONS = 16//1) 02691
1100 FORMAT(100*6X, 6H +OR=*, F8.5/) 02691
1200 FORMAT(100*6X, 6H +OR=*, F8.5/) 02691
1300 FORMAT(100*6X, 6H +OR=*, F8.5/) 02691
1400 FORMAT(100*6X, 6H +OR=*, F8.5/) 02691
1500 FORMAT(100*6X, 6H +OR=*, F8.5/) 02691
1600 FORMAT(100*6X, 6H +OR=*, F8.5/) 02691
1700 FORMAT(100*6X, 6H +OR=*, F8.5/) 02691
1800 FORMAT(100*6X, 6H +OR=*, F8.5/) 02691
1900 FORMAT(100*6X, 6H +OR=*, F8.5/) 02691
2000 FORMAT(100*6X, 6H +OR=*, F8.5/) 02691
2100 FORMAT(100*6X, 6H +OR=*, F8.5/) 02691
2200 FORMAT(100*6X, 6H +OR=*, F8.5/) 02691
2300 FORMAT(100*6X, 6H +OR=*, F8.5/) 02691
2400 FORMAT(100*6X, 6H +OR=*, F8.5/) 02691
2500 FORMAT(100*6X, 6H +OR=*, F8.5/) 02691
2600 FORMAT(100*6X, 6H +OR=*, F8.5/) 02691
2700 FORMAT(100*6X, 6H +OR=*, F8.5/) 02691
2800 FORMAT(100*6X, 6H +OR=*, F8.5/) 02691
2900 FORMAT(100*6X, 6H +OR=*, F8.5/) 02691
END 02691

Quasi Normalization of Structure Factors

CSF Quasi Normal of Structure Factors Using F Tape Input
DIMENSION NME(12), AS(20), AS1(20), BS(20), BS1(20), CS(20), AN(20),
1 M1(500), H1(500), H2(500), H3(500), M1(500), M2(500), M3(500), AVEQ(500), S1(500)
1 ITP=5
1 ITAPE = 15
1 ITAPE = 16
1 REWIND ITAPE
1 REWIND JTAPE
1 READ INPUT TAPE ITP,
1 1000, NOIB, NASP, NF, BT1, EX, *LIB
1 READ INPUT TAPE ITP,
1 1 40, (AS(I), AS1(I), BS(I), BS1(I), CS(I), AN(I), I=1, NASP)
1 IF(NF) 5, 5+1
1 IF(NF) 5, 5+1
1 N1 = NOIB/500
1 WRITE OUTPUT TAPE JTAPE, 740, NOIB
1 200 N4 = NOIB - N1 = 500
1 IF(N4) 400, 400 + 210
1 200 N4 = 500

NAVAL RESEARCH LABORATORY 39
210 READ INPUT TAPE ITAPE, 1100, (H1(I), H2(I), H3(I), AVEQ(I), SI(I), I = 1, N4)
   DO 300 I = 1, N4
   M1(I) = INTG(H1(I))
   M2(I) = INTG(H2(I))
   M3(I) = INTG(H3(I))
   SI(I) = SI(I)**2
   SIG = 0.0
   DO 250 J = 1, NASP
   SIG = SIG + AN(J)*AS(J)*EXPF(-AS1(J)*SI(I)) + BS(J)*EXPF(-BS1(J)*SI(I))
   SIG = SIG + CS(J)*EXPF(BT1*(SI(I)**2))/SIG
   AVEQ(I) = AVEQ(I) + EXPF(BT1*(SI(I)**2))/SIG
   300 WRITE OUTPUT TAPE JTAPE, 680, (M1(I), M2(I), M3(I), AVEQ(I), I = 1, N4)
   GO TO 100
   400 REWIND ITAPE
   END FILE JTAPE
   REWIND JTAPE
   PRINT 2000
   CALL EXIT
1000 FORMAT(3I10, 2E10.4, 4I10)
400 FORMAT(5F8.4, 4F4.0)
1100 FORMAT(4F9.2, 2E18X, F9.6)
680 FORMAT(3I14+F20.5)
2000 FORMAT(16H JOB FINISHED,//////)
740 FORMAT(I7)
1200 FORMAT(12A6)
END
Rational Dependence

CRATDEP RATIONAL DEPENDANCE (AFTER BLOCK-YANNONI)

DIMENSION NAME(12), NA(.29), KA(.28), KB(.28),
EC(.29), HK(.5000), ML(.5000), NH(.5000), ESQ(.5000), MR(.5000)
COMMON MH(.MK, ML, ESQ, HK, KA, KB, KC, J, F, AD, IT, NAME, N, J, K, MA, JTP, 
1AVESO, JPUT, NCTR, NO

KA(1) = 0
KA(2) = 1
KA(3) = -1
KA(4) = 2
KA(5) = -2
KA(6) = 3
KA(7) = -3
KA(8) = 4
KA(9) = -4
KA(10) = 5
KA(11) = -5
KA(12) = 6
KA(13) = -6
KA(14) = 7
KA(15) = -7
KA(16) = 8
KA(17) = -8
KA(18) = 9
KA(19) = -9
KA(20) = 10
KA(21) = -10
KA(22) = 11
KA(23) = -11
KA(24) = 12
KA(25) = -12
KA(26) = 13
KA(27) = -13
KA(28) = 14

ITP=5
JTP=6
KTP=12
REWIND KTP
50 READ INPUT TAPE ITP,8,(NAME(I),I=1,12)
51 READ INPUT TAPE ITP,4,NA,NINT,NFIN,MINN,AN,SDMIN,JPUT
60 READ INPUT TAPE KTP,5,NO
61 READ INPUT TAPE KTP,6,(MH(J),MK(J),ML(J),ESQ(J),J=1,NO)
REWIND KTP
SUMESO=0.0
DO 62 I=1,NO
ESOM(I)=ESOM(I)+1.0
62 SUMESO=SUMESQ+ESOM(I)
BNO=NO
AVESO=SUMESO/BNO
70 MA=NINT
NCTR=-1
WRITE OUTPUT TAPE JTP,15,(NAME(J),J=1,12),NINT,NFIN,AVESO,AN,MINN,
1NO,SDMIN
80 IF(NA)83,90,90
83 AN=1.414214*AN
90 DO 95 J=1,28
92 KB(J)=KA(J)
95 KC(J)=KA(J)
180 DO 440 N=1,MA
190 DO 440 J=1,MA
200 DO 440 K=1,MA
N=N
J=J
K=K
IF(KA(N)) 440,2000,202
42  

NAVAL RESEARCH LABORATORY

2000 IF(KB(J))44C,2010,202
2010 IF(KC(K))44C,440,202
202 MAD=MA+1
   DO 207 LD=1,MAD
   NF(LD)=0
207 ECL(LD)=0.
2100 DO 2400 L=1,N0
   KSUM=KA(N)*MH(L)+KB(J)*MK(L)+KC(K)*ML(L)
   MT=XMDF(KSUM,MA)
   IF(MT)2200,2300*2300
2200 MT=MT+MA
2300 MT=MT+1
   MR(L)=MT
   EC(MT)=ESQ(M)+EC(MT)
2400 NF(MT)=NF(MT)+1
300 PMA=MA
301 PMA=PMA/2.
302 MAN=MA+1
303 MAN=MAN+1
304 IT=MA+2
310 DO 430 M=1,MAN
311 IT=IT-1
1315 IF(IT-NI)1320,1325
1320 STOP 0066
1321 DNF=NF(M)
   JNF=NF(M)
1322 BEC=EC(M)
1323 GO TO 1350
1325 JNF=NF(M)+NF(IT)
1330 DNF=JNF
1340 BEC=EC(M)+EC(IT)
1350 IF(JNF-MINN)430,430,1355
1355 AVESQG=BEC/DNF
1360 CHEK=AN/SGRTF(DNF)
   IF(CHEK=SDMIN)1365,1370,1370
1365 CHEK=SDMIN
1370 PHEK=AVESQ+CHEK
1380 PMHEK=AVESQ-CHEK
360 IF(AVESQG=PHEK)370,380,380
370 IF(AVESQG=PMHEK)380,380,430
380 IF(MA-3)400,400,3193
3193 IR=0
3194 CAT=2.0
3200 XMA=MA
3210 XLA=XMA/CAT
3215 IZ=0
3220 MX=XLA
3230 YMX=MX
3240 IF(YMX=XLA)3350,3250,3350
3250 XKE=KA(N)
3260 AAD=XKE/CAT
3270 MAC=AAD
3280 ABD=MAC
3290 IF(ABD-AAD)3350,3300,3350
3300 IZ=IZ+1
3310 GO TO (3320,3330,3340,430),IZ
3320 XKE=KB(J)
3325 GO TO 3260
3330 XKE=KC(K)
3335 GO TO 3260
3340 XKE=M-1
3345 GO TO 3260
3350 IR=IR+1
3355 IF(MA-IR-3)400,400,3360
3360 DO 2400 L=1,N0
3365 IF(CAT-14*01) 3210*3210*400
400 CALL OUTRAT
430 CONTINUE
500 IF(SENSE SWITCH 2) 510*440
440 CONTINUE
450 MA-MA+1
460 IF(NFIN-MA) 470*180*180
470 READ INPUT TAPE ITP.*4*N.INT.*NFIN*MINN*AN*SDMIN*JPUT
IF(NINT)500*500*70
510 CALL EXIT
510 WRITE OUTPUT TAPE JTP.*9*(NAME(L2) + L2=1*12)
WRITE OUTPUT TAPE JTP.*7*KA(J)+KB(J)+KC(K)+MA
WRITE OUTPUT TAPE JTP.*16
GO TO 500
8 FORMAT (12A6)
4 FORMAT (4116*2E13*4*I10)
5 FORMAT (17)
6 FORMAT(314*F20.5)
7 FORMAT(6H0=13V6X93HB=13,6X,6HIAODE=16)
9 FORMAT (1H6) 10 FORMAT(1H2.19X.12A6/20X,16HINITIAL MODULAS=16*10X+15HFINAL MODU
1LAS=16/2OX.19HDETAVERAGE E**2=-F9.5/20X. 13HDEVIATIONS OFF7
2*3. 66H STANDARD DEVIATIONS FROM AVE E**2 ARE CONSIDERED SIGN
IFICANT/20X. 40H ONLY SUBGROUPS CONTAINING NO FEWER THAN 16,
4 59H REFLECTIONS ARE CONSIDERED SIGNIFICANT/20X. 10THERE ARE 17
5. 50H REFLECTIONS ON THE INPUT TAPE/20X,5INTHE MINIMUM ACCEPTED
6DEVIAITION FROM AVE. E**2 IS F5.3)
16 FORMAT(1H0=19X.47HCALCULATION TERMINATED BY SENSE SWITCH CONTROL.)
END

Subroutine OUTRAT

DIMENSION NAME(12), NF(29), KA(29), KB(29),
KC(28), EC(29), MK(500), ML(500), MH(500), ESQM(500), MR(500),
2LH(40), LH(40), LL(40), ESQL(40)
COMMON MH,MK,ML,ESQM,MR,KA,KB,EC,JN,NAME,N,J,K,MA,JTP,
1AVESQ,GJNF,JPUT,JCTR...N,KET=M-1
IF(JPUT)75*50=75
50 NCTR=NCTR+1
IF(NCTR=9(NCTR/9))960*55=60
55 WRITE OUTPUT TAPE JTP.*9*(NAME(L2)+L2=1*12)
60 WRITE OUTPUT TAPE JTP.*8*KA(J)+KB(J)+KC(K)+MA+KET=A
AVESQ+JNF
GO TO 350
75 NP=0
L4=0
DO 220 L9=1*NO
L9=L9
IF(MR(L9)=M)200*210*200
200 IF(MR(L9)=IT)220*210*220
210 L4=L4+1
LHL(L4)=MH(L9)
LKL(L4)=MK(L9)
LL(L4)=ML(L9)
ESQL(L4)=ESQM(L9)
IF(40-L4)230*230*220
220 CONTINUE
230 J1=L4/40
J2=L4-J1*40
J3=J2-20
IF(J3=J4+1241*243
241 J4=J2
GO TO 245
NAVAL RESEARCH LABORATORY

243 J4=J2-2*J3
GO TO 247
245 IF (J1)300=300+246
246 J5=J1
GO TO 250
247 J5=J1+1
250 DO 290 L1=1, J5
NP=NP+1
WRITE OUTPUT TAPE JTP,99,(NAME(L2),L2=1,12)
WRITE OUTPUT TAPE JTP,7,KA(N),KB(J),KC(K),MA,KET, NP,AVESOG,JNF
L5=L1 *40=39
IF (J1-L1)260=270,270
260 L6=L5+J3=1
GO TO 280
270 L6=L5+19
280 DO 290 L2=L5+L6
A0=ESOL(L2)-1.0
AQA=ESOL(L2)-AVESOG
AO1=ESOL(L2+20)-1.0
AQA1=ESOL(L2+20)-AVESOG
WRITE OUTPUT TAPE JTP,10
290 WRITE OUTPUT TAPE JTP,11,LH(L2),LK(L2),LL(L2),A0,AQA, LH(L2+20), LK(L2+20), LL(L2+20), A01,AQA1
300 IF (J4)360=360+360
310 IF (J3)320=320+315
315 L5=L6+1
GO TO 335
320 NP=NP+1
WRITE OUTPUT TAPE JTP,99,(NAME(L2),L2=1,12)
WRITE OUTPUT TAPE JTP,7,KA(N),KB(J),KC(K),MA,KET, NP,AVESOG,JNF
330 L5=L6+21
335 L6=L5+J4=1
DO 340 L2=L5+L6
A0=ESOL(L2)-1.0
AQA=ESOL(L2)-AVESOG
340 WRITE OUTPUT TAPE JTP,12,LH(L2),LK(L2),LL(L2),A0,AQA
350 RETURN
360 L4=0
IF (NO=L9)350,350,320
7 FORMAT(6H0A =13,6X,3HE =13,6X,3HC =13,6X,6HM,1 =16,8X,11HRE,1 =16/ 2//2160H H K L E**2 = 1 E**2 - A 3)
8 FORMAT(6H0A =13,6X,3HE =13,6X,3HC =13,6X,6HM,1 =16,8X,11HRE,1 =16/ 2//2214X,7XF9*5,7X9F9*5915X)
9 FORMAT (1H1,19X,12A6)
10 FORMAT (1H )
11 FORMAT (2(1X,3I4,7X,F9.5,7X,F9.5,15X))
12 FORMAT (1H0,3I4,7X,F9.5,7X,F9.5)
END

Sigma-2 Listings

CS1GEXC SIGMA-2 INTERACTION EXC
DIMENSION M1(2400),M2(2400),M3(2400),AV(2400),K1(2400),K2(2400),K3
12400),QAV(2400)
COMMON M1,M2,M3,AV
ITP=5
IP =11
READ INPUT TAPE ITP, 1
1002=2,I2,NO1B
IF(I2) 2, 1,2
02691
Rewind IP
READ INPUT TAPE IP,1001,NRT
GO TO 3

REWIND IP
READ INPUT TAPE ITP,1001,NRT

DO 25 J=1,NRT
20 GO TO(5+6),MM
5 READ INPUT TAPE ITP,
1 1000,N1,N2,N3,O
GO TO 7
6 READ INPUT TAPE IP,1000,N1,N2,N3,O
7 CALL REJVEC(N1,N2,N3,O)
IF(Q-Z)259.10,10
10 L=L+1
M1(L)=N1
M2(L)=N2
M3(L)=N3
AV(L)=SORTF(1.0+Q)
25 CONTINUE
30 DO 160 J=1,L
40 DO 70 K=1,L
50 IF(AV(K))70,60*60
60 BAV=AV(K)
GO TO 80
70 CONTINUE
80 DO 110 I=K,L
90 IF(AV(I)-BAV)110,100*100
100 BAV=AV(I)
I=I
110 CONTINUE
K1(J)=M1(I)
K2(J)=M2(I)
K3(J)=M3(I)
OAV(J)=AV(I)
AV(I)=1.0
160 CONTINUE
IF(NOIB)165,180,170
165 NOIB=L
170 CALL SIGVEC(NOIB,K1,K2,K3,OAV,L)
180 CALL EXIT
1000 FORMAT(3I4,F20.5)
1001 FORMAT(I7)
1002 FORMAT(E10.4,2I10)

END

Subroutine OUTSIG

SUBROUTINE OUTSIG(M1,M2,M3,O,N1,N2,N3),
Q0,N1,N2,N3)
DIMENSION K1(2400),K2(2400),K3(2400),
OAV(2400),M1(1000),M2(1000),M3(1000),OQ(1000),
N1(1000),N2(1000),N3(1000),OQ(1000)
COMMON M1,M2,M3,O,N1,N2,N3,OQ
JTP=6
IF(N)25,25,50
25 K=1
WRITE OUTPUT TAPE JTP,
1 1000,K1(I),K2(I),K3(I),OAV(I)
GO TO 130
50 J=M/20
IF(J)120,120,100
Subroutine REJVEC

C DUMMY
100 RETURN
END

SUBROUTINE REJVEC(N1,N2,N3,Q)
C REJECTS ALL REFLECTIVITY VALUES FOR WHICH ANY OF HKL EQUAL ZERO
IF(N1)10,30,10
10 IF(N2)20,30,20
20 IF(N3)40,30,40
30 Q=-10.
40 RETURN
END

Subroutine SIGVEC, Triclinic

C TRICLINIC
SUBROUTINE SIGVEC(NOIB,K1,K2,K3,GAV,L)
DIMENSION K1(2400),K2(2400),K3(2400),
GAV(2400),M1(1000),M2(1000),M3(1000),Q(1000),
2N1(1000),N2(1000),N3(1000),Q1(1000),Q2(1000),Q3(1000),
COMMON M1,M2,M3,Q,N1,N2,N3,Q1,Q
DO 170 I=1,NOIB
N=0
DO 160 J=1,L
IF(K1(J)+K1(K)-K1(I)) 10,40,10
10 IF(K2(J)+K2(K)-K2(I)) 20,50,20
20 IF(K2(J)-K2(K)+K2(I)) 30,60,30
30 IF(K3(J)+K3(K)-K3(I)) 40,70,40
40 IF(K3(J)-K3(K)+K3(I)) 50,80,50
50 IF(K3(J)+K3(K)+K3(I)) 60,90,60
60 IF(K3(J)-K3(K)-K3(I)) 70,100,70
70 IF(K3(J)+K3(K)-K3(I)) 80,110,80
80 IF(K3(J)+K3(K)+K3(I)) 90,120,90
90 IF(K3(J)-K3(K)+K3(I)) 100,130,100
100 IF(K3(J)-K3(K)-K3(I)) 110,140,110
110 IF(K3(J)+K3(K)-K3(I)) 120,150,120
120 IF(K3(J)+K3(K)+K3(I)) 130,160,130
130 RETURN
1000 FORMAT(1H1,17X*9HH K L,9X,1HE913X95HPAGE 12///16X,314,4X,F8.5
1///70H I=I) K(I) L(I) E(I) H(I) K(J) L(J) E(J)
2(E(I)*E(J))
2000 FORMAT(1H0,14,215,4XF8.5,1X,3I5 .4XF8.5,4XF10.5)
END
110 IF(K3(J)-K3(K)+K3(I))160*,150*160
150 N=N+1
M1(N)=K1(J)
M2(N)=K2(J)
M3(N)=K3(J)
Q(N) = QAV(J)
N1(N)=K1(K)
N2(N)=K2(K)
N3(N)=K3(K)
Q1(N)= QAV(K)
QQ(N)= QAV(J)* QAV(K)* QAV(I)
160 CONTINUE

CALL OUTSIG(M1,M2,M3,Q1,N1,N2,N3,QQ,N)
1 K1+K2+K3+I=QAV)
RETURN
END

Subroutine SIGVEC, Monoclinic

C MONOCLINIC, SECOND SETTING
SUBROUTINE SIGVEC(NOIB,K1,K2,K3,QAV,L)
DIMENSION K1(2400),K2(2400),K3(2400),
1QAV(2400),M1(1000),M2(1000),M3(1000),Q1(1000),
2N1(1000),N2(1000),N3(1000),Q1(1000),QQ(1000)
COMMON M1,M2,M3,Q,N1,N2,N3,Q1,QQ
DO 160 J=1,L
DO 160 K=1,L
IF(XABS(F(K2(J)+K2(K))-XABS(K2(I)))) 10, 20, 10
10 IF(XABS(K1(J)-K1(K))-60, 30, 30
30 IF(K1(J)-K1(K)+K1(I)) 40, 70, 40
40 IF(K1(J)+K1(K)-K1(I)) 50, 80, 50
50 IF(K1(K)) 160, 130, 160
60 M=1
IF(K3(J)+K3(K)-K3(I)) 100, 150, 100
70 M=2
IF(K3(J)+K3(K)+K3(I)) 100, 150, 100
80 IF(K3(J)-K3(K)-K3(I)) 146, 150, 146
90 IF(K3(J)+K3(K)+K3(I)) 160, 150, 160
100 IF(K1(J)) 120, 110, 120
110 GO TO(140*145)*M
120 IF(K1(K)) 160, 130, 160
130 GO TO(145*140)*M
140 IF(K3(J)-K3(K)+K3(I)) 160, 150, 160
145 IF(K3(J)+K3(K)-K3(I)) 160, 150, 160
150 IF(K1(K)) 1160, 90, 160
150 M=M+1
M1(N)=K1(J)
M2(N)=K2(J)
M3(N)=K3(J)
Q(N) = QAV(J)
N1(N)=K1(K)
N2(N)=K2(K)
N3(N)=K3(K)
Q1(N)= QAV(K)
QQ(N)= QAV(J)* QAV(K)* QAV(I)
160 CONTINUE

CALL OUTSIG(M1,M2,M3,Q,N1,N2,N3,QQ,N)
1 K1+K2+K3+I=QAV)
RETURN
END
Subroutine SIGVEC, Orthorhombic

Orthorhombic
SUBROUTINE SIGVEC(NO1B,K1,K2,K3,QAV,L)
DIMENSION K1(2400),K2(2400),K3(2400),
1QAV(2400),M1(1000),M2(1000),M3(1000),Q(1000),
2N1(1000),N2(1000),N3(1000),Q1(1000),QQ(1000)
COMMON M1,M2,M3,Q,N1,N2,N3,Q1,QQ
DO 170 I=1,NOIB
N=0
DO 160 J=1,1
DO 160 K=1,1
IF(XABS((K1(J)+K1(K)))-XABS(K1(I)))>100.1,100,100
100 IF(XABS((K1(J)-K1(K)))-XABS(K1(I)))>100.1,100,100
110 IF(XABS((K2(J)+K2(K)))-XABS(K2(I)))>120.1,120,120
120 IF(XABS((K2(J)-K2(K)))-XABS(K2(I)))>120.1,120,120
130 IF(XABS((K3(J)+K3(K)))-XABS(K3(I)))>140.1,140,140
140 IF(XABS((K3(J)-K3(K)))-XABS(K3(I)))>150.1,150,150
150 N=N+1
M1(N)=K1(J)
M2(N)=K2(J)
M3(N)=K3(J)
Q(N)=QAV(J)
N1(N)=K1(I)
N2(N)=K2(I)
N3(N)=K3(I)
Q1(N)=QAV(K)
QQ(N)=QAV(J)*QAV(K)*QAV(I)
160 CONTINUE
170 CALL OUTSIG(M1,M2,M3,Q,N1,N2,N3,Q1,QQ,N)
1 K1,K2,K3,I,QAV)
RETURN
END

Triple Product Summation, Centrosymmetric

CORTHP ORTHO RHOMBIC TRIP PROD PROG FOR CENTRO CRYST
DIMENSION E(30,10,30)
ITP=5
JTP=6
KTP=9
REWIND KTP
B=1.0E-21
READ INPUT TAPE ITP,1000,NA,IM,JM,KM
1000 FORMAT(6I10)
DO 75 I=1,IM
DO 75 J=1,JM
DO 75 K=1,KM
75 E(I,J,K)=B
READ INPUT TAPE KTP,2000,NOIB
2000 FORMAT(17)
DO 8 NO=1,NOIB
READ INPUT TAPE KTP,2100,I,J,K,A
2100 FORMAT(3I4,F20.5)
IF(I)=1
1 I=1
2 IF(J)=3,J=4
3 J=J
4 IF(K)=5,K=6
5 K=K
6 E(I,J,K)=A
8 CONTINUE
REWIND KTP
AN=NA
SF=SQRTF(AN**3)/8.0
JJJ=0
MN=0
C START NEW TP CALC.
9 READ INPUT TAPE ITP+1000*M1*M2*M3*N1*N2*N3
JJJ=JJJ+1
I1=XABSF(M1)
I2=XABSF(M2)
I3=XABSF(M3)
J1=XABSF(N1)
J2=XABSF(N2)
J3=XABSF(N3)
L1=M1+N1
L2=M2+N2
L3=M3+N3
K1=XABSF(L1)
K2=XABSF(L2)
K3=XABSF(L3)
IF(I1+I2+I3+J1+J2+J3)10,10,50
10 MN=MN+1
EEE=1.0
COR=0.0
GO TO (55+630)*MN
50 IF(M1)11,2,11
11 IF(M2)12,20,12
12 IF(M3)13,20,13
13 IF(N1)14,20,14
14 IF(N2)15,20,15
15 IF(N3)16,20,16
17 IF(L1)18,20,18
18 IF(L2)19,20,19
19 IF(L3)25,20,25
20 EEE=1.0
COR=0.0
GO TO 55
25 EEE=SQRTF((E(I1+I2+I3)+1.0)*(E(J1+J2+J3)+1.0)*(E(K1+K2+K3)+1.0))
COR=(E[I1+I2+I3]+E[J1+J2+J3]+E[K1+K2+K3]+1.0)/SQRTF(AN)
C CALCULATE TRIPLE AVERAGES
55 RN=0.0
TP=0.0
DO 600 I=1,IM
DO 600 J=1,JN
DO 600 K=1,KN
IF(E(I+J+K)-B)90,600,900
90 L=I
M=J
N=K
NNN=0
100 IH=XABSF(L1+L)
IK=XABSF(L2+M)
IL=XABSF(L3+N)
IF(IM)110,500+110
110 IF(IM=IH)500+120
120 IF(IK)130,500+130
130 IF(JM=IK)500+140
140 IF(IL)150,500+150
150 IF(KM=IL)500+160
160 IF(E(IM+IK+IL)-B)165,500+165
165 KM=XABSF(M1+L)
KK=XABSF(M2+M)
KL=XABSF(M3+N)
TRIPLE PRODUCT SUMMATION, NONCENTROSYMMETRIC

**DIMENSION** E(I0*10*30)

**DO 75 I=1,IM**

**DO 75 J=1,JM**

**DO 75 K=1,KM**

**READ INPUT TAPE KTP*2000,NOIB**
2000 FORMAT(17)
    DO 8 NO=1,NOIB
    READ INPUT TAPE KTP*2100*I*J*K*A
2100 FORMAT(314*F20.5)
    IF(I)1,8,2
1 I=I
2 IF(J)3,8,4
3 J=J
4 IF(K)5,8,6
5 K=K
6 E(I+J+K)=A
8 CONTINUE
    REWIND KTP
    AN=NA
    SF=SQRTP(AN**3)/2.0
    JJJ=0
    MN=0
C START NEW TP CALC+
    9 READ INPUT TAPE ITP*1000+M1*M2*M3+K1+K2+K3
    JJJ=MJ+1
    I1=XABSF(M1)
    I2=XABSF(M2)
    I3=XABSF(M3)
    J1=XABSF(N1)
    J2=XABSF(N2)
    J3=XABSF(N3)
    L1=M1+K1
    L2=M2+K2
    L3=M3+K3
    K1=XABSF(L1)
    K2=XABSF(L2)
    K3=XABSF(L3)
    IF(I1+I2+I3+J1+J2+J3)10+10+50
10 MN=MN+1
    EEE=1.0
    COR=0.0
    GO TO(55+630)+MN
50 IF(M1)11,20,11
11 IF(M2)12,20,12
12 IF(M3)13,20,13
13 IF(M1)14,20,14
14 IF(M2)15,20,15
15 IF(M3)16,20,16
16 IF(L1)17,20,17
17 IF(L2)18,20,18
18 IF(L3)19,20,19
19 IF(L3)20,20,20
20 EEE=1.0
    COR=0.0
    GO TO 55
25 EEE=SQRTP((E(I1+I2+I3)+1.0)*(E(J1+J2+J3)+1.0)*(E(K1+K2+K3)+1.0))
    COR=7(E(I1+I2+I3)+E(J1+J2+J3)+E(K1+K2+K3)+1.0)/SQRTP(AN)
C CALCULATE TRIPLE AVERAGES
55 RN=0.0
    TP=0.0
    DO 600 I=1,IN
    DO 600 J=1,JM
    DO 600 K=1,KN
    IF(E(I+J+K)=8)90,600,90
600 L=I
    M=J
    N=K
    NNN=0
100 HH=XABSF(L+I)
    IX=XABSF(L+M)
    IL=XABSF(L+K)
    IF(HH)110,500,110
Structure Factor Calculation

CSTRFAc  STRUCTURE FACTOR EXECUTIVE PROGRAM
DIMENSION NAME(12), X(3,500), HI(3,128), F1(20), FA1(20), FB(20), FB1(120), FC(20), AA(20), IA(500), TB(500), AF(500), AF1(500), BF(500), CF(500), SI(128), E(128), F(128), A1(128), FA(128), P(500), GYP(500), FK(128), C(128), SF(128), EN(128), B(128), D(128), DF(128), DE(4,128), CO(128), DO(128), DC(128), DD(128), AO(128), BO(128), DA(128), DB(1), SCF2(100)
110 IF (IM-IH).GT.500 .AND. (I-H).LE.120
120 IF (IK).GT.500 .AND. (I-K).LE.130
130 IF (JM-IK).GT.1500 .AND. (J-K).LE.140
140 IF (IL).GT.150 .AND. (I-L).LE.150
150 IF (KM-IL).GT.1600 .AND. (K-M).LE.160
160 IF (E(IH,IK,IL)).GT.165 .AND. (E(IH,IK,IL)).LE.500
165 KH*XABSFL(M1+L)
   KK*XABSFL(M2+M)
   KL*XABSFL(M3+N)
167 IF (KM-IL).GT.170 .AND. (K-M).LE.170
170 IF (IJK).GT.180 .AND. (I-J).LE.180
190 IF (KL).GT.210 .AND. (K-L).LE.210
200 IF (E(KH,KK,KL)).GT.220 .AND. (E(KH,KK,KL)).LE.500
230 RN=RN+1
   TP=TP-E(J)E(I)E(K)E(H)E(K)E(I)E(L)
500 NNN=NNN+1
   GO TO(510,520,530,540,550,600),NNN
510 L=L-1
   M=M-1
   N=N-1
   GO TO 100
520 L=L-1
   GO TO 100
530 M=M-1
   GO TO 100
540 M=M-1
   N=N-1
   GO TO 100
600 CONTINUE
   TP=TP*SF/RN+COR
   NR=RN
   TPC=TP/EEE
   IF (XMODFI).GT.1620 .AND. (XMODFI).LE.620
610 WRITE OUTPUT TAPE JTP*1100
620 IF (MN).GT.625 .AND. (MN).LE.621
621 WRITE OUTPUT TAPE JTP*1300+M1+M2+M3+N1+N2+N3+NR+TP+TPC+COR
   PRINT 1300+M1+M2+M3+N1+N2+N3+NR+TP+TPC+COR
   GO TO 9
625 WRITE OUTPUT TAPE JTP*1200+M1+M2+M3+N1+N2+N3+NR+TP+TPC+COR
   PRINT 1200+M1+M2+M3+N1+N2+N3+NR+TP+TPC+COR
   GO TO 9
630 CALL EXIT
1100 FORMAT(1H1.1OX(75H H K L H K L NO*CONT+ TRIP*PRO
   10x-COEINV) COR*TERM//)
1200 FORMAT(1H1.1OX(12H H K L H K L H K L H K L H K L H K L
   10x-COEINV) COR*TERM//)
1300 FORMAT(1H1.1OX(12H H K L H K L H K L H K L H K L H K L
   10x-COEINV) COR*TERM//)
END
50 KT=10
   IT=5
   JT=6
   LT=11
   REWIND KT
   REWIND LT
   NASP=0
   I=0
   READ INPUT TAPE IT,1000,(NAME(I),I=1,12)
   READ INPUT TAPE IT,1100,IC,IE,NR,NF,IP,LIB,LINE
   READ INPUT TAPE IT,1200,A1,A2,A3,BE,GA
   READ INPUT TAPE IT,1300,BT,XT,SCF,KKT,OUT
60 I=I+1
   READ INPUT TAPE IT,1600,SCF2(I)
   IF(SCF2(I))60,160,60
   NASP=NASP+1
   READ INPUT TAPE IT,1400,F1(NASP),F1(NASP),FB(NASP),FB1(NASP),
   FC(NASP)
   IF(F1(NASP))100,110,100
110 K=0
120 K=K+1
   READ INPUT TAPE IT,1500,X(1,K),X(2,K),X(3,K),TB(K),IA(K)
   IF(ABS(F1(1,K))+ABS(F1(2,K))+ABS(F1(3,K)))130,130,120
130 K=K-1
   NC=K
140 READ INPUT TAPE IT,1600,T1,T2,T3,U1,U2,U3
   IF(ABS(F1(T1))+ABS(F1(T2))+ABS(F1(T3))+ABS(U1)+ABS(U2)+ABS(U3))170,170,
   1 150
150 DO 160 I=1,NC
   K=K+1
   X(1,K)=T1+U1*X(1,I)
   X(2,K)=T2+U2*X(2,I)
   X(3,K)=T3+U3*X(3,I)
   IA(K)=IA(I)
   TB(K)=TB(I)
   GO TO 140
160 DO 170 J=1,NC
   SDF=0.0
   SDF0=0.0
   DO 180 I=1,K
      J=IA(I)
      AF(I)=F1(J)
      BF(I)=FB(J)
      CF(I)=FC(J)
      IF(AF(I))180,185,180
180 NN=1
      GO TO 200
190 NN=2
200 CALL RECIP(A1,A2,A3,AB,GA,AB(1,1),AB(2,2),AB(3,3),AB(2,3),AB(1,3),
   AB(1,2))
   DO 230 I=1,3
      J=IA(I)
      IF(1-J)220,210,220
210 DO 230 J=1,3
      I=IA(J)
      CALL FLSKPD(KT)
220 CONTINUE
      IF(NF)240,240,231
230 CONTINUE
      CALL FLSKPD(KT)
231 DO 230 I=1,NS
      CALL FLSKPD(KT)
240 CONTINUE
      CALL FLSKPD(KT)
241 CONTINUE
      CALL FLSKPD(KT)
242 READ INPUT TAPE KT,1000,NME(I),I=1,12
243 GO TO(244,240),NN
IF(LINE)2402*2403*2403
2402 READ INPUT TAPE KT*2100*NR
  GO TO 2401
2403 READ INPUT TAPE IT*2100*NR
2401 N0=0
  N1=NR/128
245 IF(N1=NO)470*250*260
250 N2=NR-N1*128
  IF(N2)470*470*270
260 N2=128
270 GO TO(280*310)*NN
280 IF(LINE)281*282*281
281 READ INPUT TAPE IT*1700*(H(I)*H(2*1)*H(3*1)*E(I)*I=1*N2)
  GO TO 283
282 READ INPUT TAPE KT*1700*(H(I)*H(2*1)*H(3*1)*E(I)*I=1*N2)
283 DO 300 I=1,N2
  E(I)=SORT(F(I)+A01)*SCF
  SI(I)=0.*0
  DO 290 L=1,3
    SI(I)=H(L)*H(M)*AA(L,M)+SI(I)
  300 SI(I)=SI(I)/4.*0
  GO TO 330
310 IF(LINE)311*312*311
311 DO 2311 I=1,N2
  READ INPUT TAPE IT*1800*, H(I)*H(2*1)*H(3*1)*F(I)*SCF*SI(I)
  IF(SCF)2314*2314*2313
2313 JK=INTG(SCF)
2314 F(I)=SORT(F(I))*SCF2(JK)
2311 CONTINUE
  GO TO 313
312 DO 3311 I=1,N2
  READ INPUT TAPE KT*1800*, H(I)*H(2*1)*H(3*1)*F(I)*SCF*SI(I)
  IF(SCF)3314*3314*3313
3313 JK=INTG(SCF)
3314 F(I)=SORT(F(I))*SCF2(JK)
3311 CONTINUE
313 IF(SI(I))314*314*317
314 DO 316 I=1,N2
  SI(I)=0.*0
  DO 315 L=1,3
    SI(I)=H(L)*H(M)+SI(I)
  316 SI(I)=SI(I)/4.*0
  GO TO 330
317 DO 320 I=1,N2
320 SI(I)=SI(I)+2
330 DO 450 I=1,N2
  A(I)=0.*0
  SIGK=0.*0
  FK(I)=0.*0
  DO 350 J=1,K
    FA(J)=AF(J)*EXP(-AF1(J)*SI(I))+BF(J)*EXP(-BF1(J)*SI(I))+CF(J)
    SIGK=SIGK+FA(J)**2
    FK(I)=FK(I)+FA(J)*EXP(-TB(J)*SI(I))**2
    P(J)=0.*0
  350 P(J)=P(J)+H(L)*X(L)*J
  P(J)=P(J)*6.2831853
340 FY(J)=FA(J)*EXP(-TB(J)*SI(I))
350 A(I)=A(I)+GY(J)*COSF(P(J)) 
  FBX=EXP(B(T*(SI(I)+XT))/2.*SRTF(SIGK)
  FX=1.*0/SRTF(FK(I))
  IF(KIK+1)351*352*359
351 FK(I)=FFAX
352 FK(I)=FFAX
353 FK(I)=FFAX
354 FK(I)=FFAX
355 C(I)=A(I)*FK(I)
IF(IC)370,360,370
360 SF(I)=A(I)
EN(I)=C(I)
GO TO 390
370 B(I)=0
DO 380 J=1,K
380 B(I)=B(I)+GYP(J)*SINF(P(J))
D(I)=B(I)*FK(I)
SF(I)=SORTF(A(I)**2+B(I)**2)
EN(I)=SF(I)*FK(I)
390 GO TO(400+10)*NN
400 F(I)=E(I)/FJ(I)
GO TO 420
410 E(I)=F(I)*FJ(I)
420 DF(I)=SIGNF(F(I)*SF(I))-SF(I)
SDF=SAFE+ABSF(DF(I))
SFO=SFO+F(I)
DE(I)=SIGNF(E(I)*EN(I))-EN(I)
IF(F(I)422,423,422
422 SDF=SDF+ABSF(DF(I))
423 IF(IC)425,424,425
424 F(I)=SIGNF(F(I)*SF(I))
E(I)=SIGNF(E(I)*EN(I))
GO TO 450
425 IF(IP)440,430,440
430 TAR=E(I)/ABSDF(EN(I))
CO(I)=C(I)*TAR
DO(I)=D(I)*TAR
DC(I)=SIGNF(CO(I)+C(I))-C(I)
DD(I)=SIGNF(ROI(I)+D(I))-D(I)
GO TO 450
440 RAT=F(I)/ABSDF(SF(I))
AO(I)=A(I)*RAT
BO(I)=B(I)*RAT
DA(I)=SIGNF(AO(I)+A(I))-A(I)
DB(I)=SIGNF(BO(I)+B(I))-B(I)
450 CONTINUE
460 IF(OUT)62,465,62
62 IF(IC)66,65,66
65 WRITE OUTPUT TAPE LT*1200+(H(I)+H(2)+H(3)+I)*F(I)+E(I)+DF(I),
1DE(I),I=1,N2
GO TO 465
66 IF(IP)68,67,68
67 WRITE OUTPUT TAPE LT*1200+(H(I)+H(2)+H(3)+I)*CO(I)+DO(I)+DC(I),
1DD(I),I=1,N2
GO TO 465
68 WRITE OUTPUT TAPE LT*1200+(H(I)+H(2)+H(3)+I)*AO(I)+BO(I)+DA(I),
1DB(I),I=1,N2
465 CALL OUTSF(I)=1C*H+FSF*S+A*B+C*E*DF*NO+A*BO+DA*DC*CO*DO*DC,
1DD*NAME*IP*DE)
NO=NO+1
GO TO 245
470 REWIND KT
END FILE LT
REWIND LT
R=SDF/SFO
R1=SDF/SFO
WRITE OUTPUT TAPE JT*1900*SFO;SDF;SFO*R1
READ INPUT TAPE IT*2100*I1J
IF(I1J)50,480,50
480 PRINT 2000
CALL EXIT
1000 FORMAT(12A6)
1100 FORMAT(7110)
1200 FORMAT(3E10.4,2I10)
1300 FORMAT(5F8.4)
1400 FORMAT(4E10.4,110)
1500 FORMAT(4E10.4)
1600 FORMAT(3F8.0,4F2.0)
1900 FORMAT(1H15,*9X,1H15SIGMA F(O)=F10.3,8X
1.15SIGMA /DEL F/ =F10.3,8X
2* THESE QUANTITIES DO NOT INCLUDE CONTRIBUTIONS FROM REFLECTIONS/
30X*33H OBSERVED WITH ZERO INTENSITIES
2000 FORMAT(1H15,*9X,2I10) THIS JOB IS FINISHED/*/////////)
2100 FORMAT(17)

Subroutine OUTSFC

SUBROUTINE OUTSFC(N2,IC,H,F, SF,A,B,E,EN,C,D,DF,NO,AO,BO,DA, DB, CO,
1 DO,DC,DD,NAME,IP, DE)
DIMENSION H(3,128),F(128),A(128),B(128),E(128),EN(128),C(128),D(128),
1 DC(128), DD(128), NAME(12), DE(128), M(3,128), SF(128)
JT=6
N4=6*NO
DO 50 I=1,N2
DO 50 J=1,3
50 M(J,I)=INTG(H(J,I))
IF(128-N2)
110 N1=N2+22
N3=N2-N1+22
IF(N1)
115 N1=5
110 N1=5
N3=18
120 DO 160 I=1,N1
N4=N4+1
J1=(I-1)*22+1
J2=J1+21
WRITE OUTPUT TAPE JT*1000*(NAME(K),K=1,12),N4
IF(1C)
125 IF(IP)
130 WRITE OUTPUT TAPE JT*1200
WRITE OUTPUT TAPE JT*1400*(M(1,J),M(2,J),M(3,J),F(J),SF(J),E(J),
1 EN(J),C(J),D(J),DD(J),DE(J),J=1,J2)
GO TO 160
140 WRITE OUTPUT TAPE JT*1100
WRITE OUTPUT TAPE JT*1300*(M(1,J),M(2,J),M(3,J),E(J),EN(J),F(J),
1 SF(J),AO(J),BO(J),A(J),B(J),DA(J),DB(J),DF(J),J=1,J2)
GO TO 160
150 WRITE OUTPUT TAPE JT*1500
WRITE OUTPUT TAPE JT*1600*(M(1,J),M(2,J),M(3,J),F(J),SF(J),E(J),
1 EN(J),DF(J),DE(J),J=1,J2)
160 CONTINUE
165 IF(N3)210,210+170
170 J1=N1+2+1
N4=N4+1
WRITE OUTPUT TAPE JT*1000*(NAME(K),K=1,12)/N4
IF(IC)175,220,175
175 IF(IP)190,180,190
180 WRITE OUTPUT TAPE JT*1200
WRITE OUTPUT TAPE JT*1400*(M(1,J),M(2,J),M(3,J),F(J),SF(J),E(J),
1 EN(J),CO(J),DO(J),C(J),D(J),DE(J),J=J1,N2)
GO TO 210
190 WRITE OUTPUT TAPE JT*1100
WRITE OUTPUT TAPE JT*1300*(M(1,J),M(2,J),M(3,J),E(J),EN(J),F(J),
1 SF(J),A0(J),B0(J),A(J),B(J),DA(J),DB(J),DF(J),J=J1,N2)
GO TO 210
200 WRITE OUTPUT TAPE JT*1500
WRITE OUTPUT TAPE JT*1600*(M(1,J),M(2,J),M(3,J),F(J),SF(J),E(J),
1 EN(J),DF(J),DE(J),J=J1,N2)
210 RETURN
1000 FORMAT(1H1,12A6//10X,4HPAGE=14/) 02693
1100 FORMAT(120H0 H K L 02693
1E0) F(0) F(C) 02693
1A(0) B(O) A(C) B(C) DEL A DEL B DEL F 02693
2) 02693
1200 FORMAT(120H0 H K L F(0) F(C) E(O) E(C) 02693
1A(0) B(O) A(C) B(C) DEL A DEL B DEL F 02693
2) 02693
1300 FORMAT(1H0.313*2F10.4*9F10.3) 02693
1400 FORMAT(1H0.313*2F10.4*9F10.4) 02693
1500 FORMAT(186HO.313*2F10.4*9F10.4) 02693
1600 FORMAT(1HO.313*2F10.3*9F10.3*9F10.4) 02693
END

Interatomic Distance and Angle

CHNDANG  BOND DISTANCE AND ANGLE PROGRAM

DIMENSION NAME(12),N(500),X(500),Y(500)/
1Z(500),NS1(200),NS2(200),SV(200),N1(2900),N2(2900)/
2D(2900),E(2900),E(2900),E3(2900),ANG(2000)/
COMMON N1,N2,D*E1+E2+E3
EQUIVALENCEx(M1,N),(M2*X),(M3*Y),(ANG,Z) 02693
COSDF(X)=COSF(3.14159*X/180a) 02693
JTP=5
ITP=6
KTP=9
REWIND KTP
READ INPUT TAPE ITP*1000*(NAME(11),I=1,12)/
READ INPUT TAPE ITP*1200*A,B,C,AL,BE,GA
READ INPUT TAPE ITP*1500*BMX,AMX
READ INPUT TAPE ITP*1100*XINC,YINC,ZINC
BMX=BMX#2
CAL=COSDF(AL)
CBE=COSDF(RE)
GLA=COSDF(GA) 02693
I=0
IF(XINC+YINC+ZINC>100,100+107
100 I=I+1
READ INPUT TAPE ITP*1100*X(I),Y(I),Z(I) 02693
105 N(I)=I-1
IF(ABS(F(I))+ABS(F(Y(I))+ABS(F(Z(I))))>1100,110+100
107 CALL InTerP(I,N,X,Y,Z,ITP,XINC,YINC,ZINC,A,B,C,AL,CAEB,GA)
READ INPUT TAPE ITP,1200,T1,T2,T3,U1,U2,U3
IF(ABS(T1)+ABS(T2)+ABS(T3)+ABS(U1)+ABS(U2)+ABS(U3))140*140=0
DO 120 J=1,I
I=I+1
N(J)=J-1
X(J)=T1+U1*X(J)
Y(J)=T2+U2*Y(J)
Z(J)=T3+U3*Z(J)
GO TO 115
120 DO 130 J=1,I
I=I+1
N(J)=J-1
X(J)=T1+U1*X(J)
Y(J)=T2+U2*Y(J)
Z(J)=T3+U3*Z(J)
GO TO 115
130 DO 140 J=1,I
I=I+1
N(J)=J-1
X(J)=T1+U1*X(J)
Y(J)=T2+U2*Y(J)
Z(J)=T3+U3*Z(J)
GO TO 115
140 DO 145 J=1,I
I=I+1
WRITE OUTPUT TAPE JTP,1700,L1,L2,NAME(K),K=1,12
145 WRITE OUTPUT TAPE JTP,1700,L1,L2,NAME(K),K=1,12
141 WRITE OUTPUT TAPE JTP,1800,LNAME(K),K=1,12
142 L1=N(J)/I
L2=N(J)-L1*I+1
150 DO 190 J=1,I
I=J+1
DO 190 K=1,I
IF(N(J)-N(K))155,157,159
157 V1=X(J)-X(K)
V2=Y(J)-Y(K)
V3=Z(J)-Z(K)
VEC=ABS(dotprod(V1,V2,V3,V1,V2,V3))
SV=SQRT(VEC)
GO TO 190
155 IF(N(J))150,157,159
150 CONTINUE
160 IF(VEC=49)170,180,180
160 CONTINUE
170 J=JJ+1
NS(J)=N(J)
NV(J)=N(K)
SV(J)=SQRT(VF)
GO TO 190
180 KK=KK+1
N1(KK)=N(J)
N2(KK)=N(K)
D(KK)=SQRT(VEC)
E1(KK)=V1
E2(KK)=V2
E3(KK)=V3
IF(D(KK)=99)190,190,190
190 CONTINUE
LL=0
GO TO 322
IF(D(J)-AMX)195,195,322
195 KK=J+1
DO 320 K=KK1,KK
IF(D(K)-AMX)1200,200,320
200 IF(N1(J)=N1(K))220,210,220
210 M=1
ZD=1,0
GO TO 276
220 IF(N1(J)=N2(K))240,230,240
230 M=2
ZD=1,0
GO TO 276
240 IF(N2(J)=N1(K))260,250,260
250 M=3
ZD=1,0
GO TO 276
IF(N2(J)-N2(K))^2 > 320*270*320
M=4
ZD=1.0
LL=LL+1
COANG=ZD*DOTPRD(E1(J),E2(J),E3(J),E1(K))
1E2(K),E3(K)=A,B,C,CAL*CEBE(CGA)/(D(J)*D(K))
ANGCO=1.0+ABSF(COANG)
IF(ANGCO>127792779277)
WRITE OUTPUT TAPE JTP,1600,N1(J),N2(J),N1(K),N2(K),ANGCO
COANG=1.0
ANG(LL)=90.00-57.295780*ARCSIN(COANG)
GO TO(280,290,300,310)
M1(LL)=N2(J)
M2(LL)=N1(J)
M3(LL)=N2(K)
GO TO 315
280 M1(LL)=N2(J)
M2(LL)=N1(J)
M3(LL)=N2(K)
GO TO 315
290 M1(LL)=N2(J)
M2(LL)=N1(J)
M3(LL)=N1(K)
GO TO 315
300 M1(LL)=N1(J)
M2(LL)=N2(J)
M3(LL)=N2(K)
GO TO 315
310 M1(LL)=N1(J)
M2(LL)=N2(J)
M3(LL)=N1(K)
315 IF(2000-LL)360,360,320
360 WRITE TAPE KTP,(M1(MN),M2(MN),M3(MN),ANG(MN),MN=1,2000)
LL=0
NM=NM+1
320 CONTINUE
322 CONTINUE
IF(LL)323,323,1323
1323 CALL OUTBND(I,J,J,J,K,K,L,L,S1,S2,SV)
1N1,N2,N3,D,E1,E2,E3,ANG,M1,M2,M3,
2NAME,JTP,NM,KTP)
325 CALL EXIT
330 BMX=(SORTF(BMX)-25)**2
IF(BMX=4.0)335,335,370
335 IF(I/I-1),340,340,350
340 WRITE OUTPUT TAPE JTP,1300
GO TO 325
350 WRITE OUTPUT TAPE JTP,1400
I=1
370 KK=0
JJ=0
GO TO 150
1000 FORMAT(12A6)
1100 FORMAT(E10.4)
1200 FORMAT(E10.4)
1300 FORMAT(1H1,8X,71HMORE THAN 2900 2 A. OR LESS BOND DISTANCES MAY
1E BEEN FOUND AMONG THE///9X,71HUNTRANSLATED SET OF ATOMS. THIS PROBLEM IS TOO LARGE. NO RESULTS WILL///9X,10MBE ISSUED.)
1400 FORMAT(1H1,8X,71HMORE THAN 2900 2 A. OR LESS BOND DISTANCES MAY
1E BEEN FOUND. VECTORS///9X,58H WILL BE CONSIDERED FOR THE UNTRANSLATED SET OF ATOMS ONLY.)
END
Subroutine INTERP

```
SUBROUTINE INTERP(I, INXYZ, TPXINCYINCAL, A, B, C, CAL, CBE, CGA)
DIMENSION X(500), Y(500), Z(500), N(500), P(7), E(3, 3), G(3), H(3)
E(1, 1) = 1.0
E(1, 2) = B * CGA * YINC / (XINC * A)
E(1, 3) = C * CBE * ZINC / (XINC * A)
E(2, 1) = A * CGA * XINC / (YINC * B)
E(2, 2) = 1.0
E(2, 3) = C * CAL * ZINC / (YINC * B)
E(3, 1) = A * CBE * XINC / (ZINC * C)
E(3, 2) = B * CAL * YINC / (ZINC * C)
E(3, 3) = 1.0

100 I = 1
READ INPUT TAPE ITP, 1000, X(I), Y(I), Z(I)
IF (ABSF(X(I)) + ABSF(Y(I)) + ABSF(Z(I))) 130, 130, 110

110 READ INPUT TAPE ITP, 1000, (P(J), J = 1, 7)

L = 0
DO 120 J = 1, 7
L = L + 2
G(J, 4) = LOGF(P(1) / P(L)) / LOGF(P(1) / P(L + 1))
G(J + 4) = (G(J + 4) - 1.0) / (2.0 * (G(J + 4) + 1.0))
DO 120 K = 1, 3

120 G(J + K) = E(J, K)
CALL MATSCGEH, 01391)
X(I) = X(I) + H(1) * XINC
Y(I) = Y(I) + H(2) * YINC
Z(I) = Z(I) + H(3) * ZINC
N(I) = I - 1
GO TO 100

130 RETURN
1000 FORMAT (7E10.4)
END
```

Subroutine OUTBND

```
SUBROUTINE OUTBND(I, J, K, L, M, N, S1, S2, SV, N1, N2, N3, E1, E2, E3, ANG, M1, M2, M3, NAME, JTP, KM, KTP)
    REWIND KTP

50 I = IK / 44
IF (IK) 185, 85, 75
75 IF (I1) 120, 120, 100
85 WRITE OUTPUT TAPE JTP, 8000
GO TO 250
100 DO 110 J = 1, 11
WRITE OUTPUT TAPE JTP, 1000, (NAME(K), K = 1, 12), J
I3 = J * 44 - 22
I2 = I3 - 21
DO 110 L = I2, I3
L1 = N1(L) / I
M11 = N1(L) - L1 * I + 1
L2 = N2(L) / I
M22 = N2(L) - L2 * I + 1
L11 = N1(L + 22) / I
M111 = N1(L + 22) - L11 * I + 1
L22 = N2(L + 22) / I
M222 = N2(L + 22) - L22 * I + 1
```

02693

02693
   1D(L+22)
   IF(KK-11*44) 170*170*120 02693
120 14=11*44+1 02693
   15=KK-14-21 02693
   IF(15) 121*122*122 02693
121 15=0 02693
122 16=14+15-1 02693
   17=16-1 02693
   19=11*44+1 02693
   WRITE OUTPUT TAPE JTP*1000*(NAME(K)*K=1*12)*19
   IF(15) 150*150*130 02693
130 DO 140 L=16*I6 02693
   L1=N1(L)/I 02693
   M11=N1(L*L1*I+1 02693
   L2=N2(L)/I 02693
   M22=N2(L-L2*I+1 02693
   LL1=N1(L+22)/I 02693
   MM1=N1(L+22)*L1*1+1 02693
   LL2=N2(L+22)/I 02693
   MM2=N2(L+22)*L2*1+1 02693
140 WRITE OUTPUT TAPE JTP*2000*L1*M11*L2*M22*D(L)*LL1*MM1*
   1LL2*MM2*D(L+22) 02693
150 DO 160 L=17*I8 02693
   L1=N1(L)/I 02693
   M11=N1(L*L1*I+1 02693
   L2=N2(L)/I 02693
   M22=N2(L-L2*I+1 02693
160 WRITE OUTPUT TAPE JTP*3000*L1*M11*L2*M22*D(L) 02693
170 GO TO(175*270)*IJK 02693
175 IF(NM) 250*250*176 02693
   LL=LM 02693
   IF(LL) 250*250*177 02693
176 IF(J1) 1200*200*180 02693
180 DO 190 K=1*J1 02693
   WRITE OUTPUT TAPE JTP*4000*(NAME(L)*L=1*12)*K
   J3=K*4#22 02693
   J2=J3-21 02693
   DO 190 N=12*J3 02693
   L1=M1(N)/I 02693
   L2=M2(N-L1*I+1 02693
   L3=M3(N)/I 02693
   L4=M2(N-L3*I+1 02693
   L5=M3(N)/I 02693
   L6=M3(N-L5*I+1 02693
   ML1=M1(N+22)/I 02693
   ML2=M1(N+22)-ML1*I+1 02693
   ML3=M2(N+22)/I 02693
   ML4=M2(N+22)-ML3*I+1 02693
   ML5=M3(N+22)/I 02693
   ML6=M3(N+22)-ML5*I+1 02693
190 WRITE OUTPUT TAPE JTP*5000*L1*L2*L3*L4*L5*L6*ANG(N)
   J1L1*ML2*ML3*ML4*ML5*ML6*ANG(N+22) 02693
   IF(LL=J1*44) 175*175*200 02693
200 J4=J1*44+1 02693
   J5=LL-J4-21 02693
   IF(J5) 201*202*202 02693
201 J5=0
WRITE OUTPUT TAPE JTP,4000,(NAME(L)L+L=1+12),J9

WRITE OUTPUT TAPE JTP,5000,L1,L2,L3,L4,L5,L6,ANG(N),ML1,ML2,ML3,ML4,ML5,ML6,ANG(N+22) Go to 175

WRITE OUTPUT TAPE JTP,6000,L1,L2,L3,L4,L5,L6,ANG(N)

Go to 75

WRITE OUTPUT TAPE JTP,9000

REWIND KTP

RETURN

1000 FORMAT(1H1,6X,90MBOND DISTANCES(ANGSTROMS) FOR 12A6,7H PAGE 12/02693

1/2(11X,32HATOM ATOM DISTANCE) 1000 FORMm1H1,6X,30MBOND ANGLES (DEGREES) FOR 12A6,7H PAGE 12/02693

1/2(11X,32HATOM ATOM ANGLE) 11X)

5000 FORMAT(1H0,17X,12+1H=12,4X,F8.5) 26X,12+1H=12,4X,F8.5)

3000 FORMAT(1H0,17X,12+1H=12,4X,F8.5) 4000 FORMAT(1H1,6X,30HATOM ANGLES (DEGREES) FOR 12A6,7H PAGE 12/02693

1/2(11X,32HATOM ATOM ANGLE,11X)

7000 FORMAT(1H1,10X,33HATOM ANGLES INVOLVING THESE BONDS HAVE NOT BEEN CALC02693

1LULATED.)

8000 FORMAT(1H1,10X,64HNO BOND DISTANCES BETWEEN 0.7 AND 4.0 ANGSTROMS 02693

1HAVE BEEN FOUND.1

9000 FORMAT(1H1,10X,58HNO BOND DISTANCES LESS THAN 0.7 ANGSTROMS HAVE BEEN FOUND.)

END
NAVAL RESEARCH LABORATORY

Least-Squares Line and Plane Fitter

CLSOPLN LEAST-SQUARES LINE AND PLANE FITTER

DIMENSION ZZ(12),ZY(10),X(350),WXS(3),WXXS(3),WT(50),
1ADJ(3),A(3),G1(3),XMK(3),B(3),VMK(3),VM(3),VMI(3),
2VMI(3),C(3),DI(50),DEL(50),DEL2(50),XBAR(3),VMK(3),VM(3),
3P(3),UMK(3),BV(3)

4*Z12(3)

ITP=5
JTP=6
Z34=1.0E-12
MM=10

READ INPUT TAPE ITP,1000,(ZZ(I),I=1,12)
READ INPUT TAPE ITP,1003,A1,A2,A3,AL,BE,GA
WRITE OUTPUT TAPE JTP,1020,ZZ(I),I=1,12
CALL RECIP(A1,A2,A3,AL,BE,GA)

C SET UP MATRIX G
G(1,1)=A1**2
G(1,2)=A1*A2*COS(A*3.14159/180.0)
G(1,3)=A1*A3*COS(B*3.14159/180.0)
G(2,1)+G(1,2)
G(2,2)=B2**2
G(2,3)=B2*B3*COSA
G(3,1)=G(1,1)
G(3,2)=G(2,2)
G(3,3)=B3**2
G(1,1)*A1**2
G(1,3)=A1*A3*COSF(GA*3.14159/180.0)
G(2,1)*G(1,1)
G(2,2)=A2**2
G(2,3)=A2*A3*COSF(AL*3.14159/180.0)
G(3,1)=G(1,1)
G(3,2)=G(2,1)
G(3,3)=A3**2

READ INPUT TAPE ITP,1002,N,NO,(ZY(I),I=1,10)

C COORDINATES OF CENTROID
DO 25 I=1,N
IF(WT(I),25,25)
22 WT(I)=1.0
25 CONTINUE
DO 30 I=1,N
DO 30 I=1,N
WXS(I)=0.
DO 30 I=1,N
30 WXS(I)=WXS(I)+WT(J)*X(I,J)
WS=0.
DO 40 I=1,N
40 WS=WS+WT(I)
DO 50 I=1,N
50 XBAR(I)=WXS(I)/WS

C SET UP MATRIX A
DO 60 I=1,N
DO 60 I=1,N
WXXS(I,J)=0.
DO 60 K=1,N
55 WXXS(I,J)=WXXS(I,J)+WT(K)*X(I,K)*X(J,K)
60 A(I,J)=WXXS(I,J)-XBAR(I)*XBAR(J)*WS

C EVALUATE MATRIX B FOR LINE
61 CALL MTXMUL(3,3,A,G1*A+B)
GO TO 75

C SET UP ADJOINT OF A
65 ADJ(1,1) = A(2,2) * A(3,3) - A(2,3) * A(3,2)
ADJ(2,1) = A(3,1) * A(2,3) - A(2,1) * A(3,3)
ADJ(3,1) = A(1,2) * A(3,3) - A(1,3) * A(3,2)
ADJ(1,2) = A(2,3) * A(1,1) - A(2,1) * A(1,3)
ADJ(2,2) = A(1,1) * A(3,3) - A(1,3) * A(3,1)
ADJ(3,2) = A(1,3) * A(2,1) - A(1,1) * A(2,3)
ADJ(1,3) = A(3,1) * A(1,2) - A(3,2) * A(1,1)
ADJ(2,3) = A(3,2) * A(1,1) - A(3,1) * A(1,2)
ADJ(3,3) = A(2,1) * A(1,2) - A(2,2) * A(1,1)

C EVALUATE DETERMINANT OF A
CALL MTXMUL(3,3,3,ADJ,A,C)
DETNAA=0.
DO 70 I=1,3
70 DETNA=DETNA+C(I,I)
DETNAA=DETNAA/3.0

C EVALUATE MATRIX B FOR PLANE
CALL MTXMUL(3,3,3,ADJ,G+B)

75 DO 80 I=1,3
80 BV(I)=0.
DO 90 I=1,3
90 BV(I)=BV(I)+B(J,I)**2
BIGBV=BV(I)
KK=1
DO 110 I=2,3
IF(BV(I)-BIGBV)110,110,100
100 KK=I
BIGBV=BV(I)
110 CONTINUE
VMN=SQRTF(BIGBV)
DO 120 I=1,3
120 VM(I,I)=B(I,KK)/VMN
NNN=0

C EVALUATE VM BY ITERATION
125 CALL MTXMUL(3,3,1,B+VM*VM)
NNN=NNN+1
IF(MM-NNN)3,3,4
3 MM=MM+10
Z34=Z34+100.
4 IF(40-NNN)11,12
1 WRITE OUTPUT TAPE JTP,1021,NO
GO TO 235
2 I=1
VMN=SQRTF(VMI(I,1)**2+VMI(2,1)**2+VMI(3,1)**2)
DO 126 I=1,3
126 VMI(I,I)=VMI(I,1)/VMN
DO 140 I=1,3
Z12(I)=VMI(I,1)/VMI(I,1)-1.0
IF(ABS((VMI(I,1)/VMI(I,1))-1.0)-Z34 140,140,130
130 IJ=2
GO TO 150
140 CONTINUE
150 DO 160 I=1,3
VMI(I,1)=VMI(I,1)
160 VMI(I,1)=VMI(I,1)
GO TO(170*125)*IJ

C NORMALIZE VECTOR VM
170 CALL MTXMUL(1,3,3,VM*G+VMI)
CALL MTXMUL(1,3,1,VMI,VM+VMI)
ORM=SQRTF(VMJ(1,1))
DO 180 I=1,3
180 VMI(I,1)=VMI(I,1)/ORM
IF(I)185,185,500
500 DO 510 I=1,3
   P(I)=0.0
   DO 510 J=1,3
510 P(I)=VM(J+1)*G(J+I)+P(I)
   DL2=0.0
   DO 530 I=1,N
   DEL2(I)=0.0
   DO 520 J=1,3
   DO 520 K=1,3
520 DEL2(I)=DEL2(I)+((XBAR(J)-X(J+1))*(XBAR(K)-X(K+1))*(G1(J,K)-
   VM(J+1)*VM(K+1))
530 DL2=DL2+DEL2(I)
   WRITE OUTPUT TAPE JTP,1025,NON,(ZY(I),I=1,10),XBAR(I)*P(I),XBAR(2
   1),P(2)*XBAR(3)*P(3)*(I+WT(I)+DEL2(I))*I=1,N)
   SATN=N-2
   IF(SATN)550,550,540
540 STND=SQRT(DL2/SATN)
   WRITE OUTPUT TAPE JTP,1032,STND
550 DO 560 I=1,3
560 VMK(I+NO)=VM(I+1)
   GO TO 235
C EVALUATE EIGEN VALUE FOR PLANE
185 CALL MTXMUL(3*3*1*8*VM*VMJ)
   DNEG=0.
   DO 190 I=1,3
190 DNEG=DNEG+VMJ(I+1)/VM(I+1)
   EIGEN=3.0*DETNA/DNEG
C EVALUATE D(PLANE TO ORIGIN DISTANCE)
   D=0.
   DO 200 I=1,3
200 D=D+VM(I+1)*XBAR(I)
C EVALUATE D*DELTA D*(DELTA D)**2 FOR EACH POINT
   DL2=0.0
   DO 220 I=1,N
   DI(I)=0.0
   DO 210 J=1,3
210 DI(I)=DI(I)+VM(J+1)*X(J+1)
   DEL(I)=DI(I)-D
   DEL2(I)=DEL(I)**2
220 DL2=DL2+WT(I)*DEL2(I)
   DO 230 I=1,3
230 VMK(I+NO)=VM(I+1)
C OUTPUT ONE
   WRITE OUTPUT TAPE JTP,1005,NOSN,(ZY(I),I=1,10),VM(1,1),VM(2,1),VM(3
   1),XBAR(1),XBAR(2),XBAR(3)*(I+WT(I)+DI(I)+DEL(I)+DEL2(I))*I=1,N
2)
   WRITE OUTPUT TAPE JTP,1012,DL2,EIGEN
   AVED=0.
   DO 231 I=1,N
231 AVED=AVED+DI(I)
   ATN=N
   AVED=AVED/ATN
   SATN=N-3
   IF(SATN)232,232,233
232 WRITE OUTPUT TAPE JTP,1018,AVED
   GO TO 235
233 STND=SQRT(DL2/SATN)
   WRITE OUTPUT TAPE JTP,1019,AVED,STND
C INTERROGATE NEXT CARD
235 READ INPUT TAPE ITP,1002,NOSN,(ZY(I),I=1,10)
   IF(N)240,240,245
240 READ INPUT TAPE ITP,1013,NOLNOSN
   IF(N)270,270,245
C OUTPUT TWO
245 WRITE OUTPUT TAPE JTP*1014
250 READ INPUT TAPE JTP*1013*N*L1+N0L2
IF(L1)1270*270*260
260 IF(L2)1275*275*300
275 IF(L2)1280*280*340
280 DO 290 I=1,3
XMK(I)=VMK(I*N)
290 YMK(I)=VMK(I*N)
LLL=1
GO TO 350
300 IF(L1)310*310*330
310 DO 320 I=1,3
XMK(I)=UMK(I*N)
320 YMK(I)=UMK(I*N)
LLL=2
GO TO 350
330 DO 335 I=1,3
XMK(I)=UMK(I*N)
335 YMK(I)=UMK(I*N)
LLL=3
GO TO 350
340 DO 345 I=1,3
XMK(I)=VMK(I*N)
345 YMK(I)=UMK(I*N)
LLL=4
350 DTPD=DOTPRD(XMK(1),XMK(2),XMK(3),YMK(1),YMK(2),YMK(3),B1,B2,B3,
1COSA+COSB+COSC)
WRITE OUTPUT TAPE JTP*1006
GO TO(360*370*380*390)*LLL
360 WRITE OUTPUT TAPE JTP*1016*N*NO*DTPD
GO TO 250
370 WRITE OUTPUT TAPE JTP*1036*N*NO*DTPD
GO TO 250
380 WRITE OUTPUT TAPE JTP*1037*N*NO*DTPD
GO TO 250
390 WRITE OUTPUT TAPE JTP*1036*N*NO*DTPD
GO TO 250
270 PRINT 1017
CALL EXIT
1000 FORMAT(12A6)
1002 FORMAT(213916.F6)
1003 FORMAT(16E10.4)
1005 FORMAT(120,12HPLANE NUM:14,110,6H ATOMS:7X,10A6:///12X,90HEQU
2ATION OF PLANE IS M1*X + M2*Y + M3*Z = D WHERE D IS THE ORI
3GIN TO PLANE DISTANCE/1HO=10X,3HM1=E12,5*7H M2=E12,5*7H M3=E
412,5*6H D=E12,5:///12X,32HCORDINATES OF CENTROID = E12,5
5*5X3Z M1=12,5*3HZ =12,5:///15X*72HATOM NUMBER WEIGHT
6 D DELTA D (DELTA D)**2/(1HO=17X,136X=E10*4*,
75X=E12,5*5X=E12,5*4X=E12,5))
1006 FORMAT(1HO)
1012 FORMAT(///,15X,21HSUM WT*(DELTA D)**2 =E12,5*20X*13HEIGEN VALUE =E
112,5//)
1013 FORMAT(413)
1014 FORMAT(1H1,29X,40HDHEDRAL ANGLES BETWEEN PLANES AND LINES///,29X,
134HANGLE BETWEEN PLANES P1 AND P2 = A/1HO=28X*33HANGLE BETWEEN LIN
2ES L1 AND L2 = B/1HO=28X*36HANGLE BETWEEN LINE L AND PLANE P = C)
1016 FORMAT(1HO=28X*4HP1 =13,10H P2 =13,17H COSINE(A) =E12,5)
1017 FORMAT(1HO=19X,13HJOB FINISHED.........//)
1018 FORMAT(1HO=14X,11HVERAGE D =E12,5)
1019 FORMAT(1HO=19X,11HVERAGE D =E12,5*15X*20HSTANDARD DEVIATION =E12,
15)
1020 FORMAT (1H1,35X,12A6)
1021 FORMAT(1H1,7H A SATISFACTORY LEAST SQUARES FIT CANNOT BE FOUN
1D FOR SET NUMBER 13*38H. EXAMINE INPUT PARAMETERS FOR ERROR. )
NAVAL RESEARCH LABORATORY

1425 FORMAT(1H1,8X$1HLINE NUMBER$14$I10.6$ATOMS$7X$10A6$///$12X$40M$THE
PARAMETRIC EQUATIONS OF THE LINE ARE$7/37X4$4X$E12.5$5H$+$E12.5
2$4H$T$3$7X4$4H$E12.5$5H$+$E12.5$4H$T$3$7X4$4H$E12.5$5H
3$E12.5$4H$T$///$12X$40M$THE SQUARE DEVIATIONS FROM THE LINE$D**
42$ARE//22X$11$ATOM NUMBER$11X$6$WEIGHT$14X$4$HD**2$/1$H0$24X$13$1
54X$E12.5$6X$E12.5$1)

1032 FORMAT(///$12X$21H$STANDARD$DEVIATION$=E12.5$)
1035 FORMAT(1H0$28X$4H$1=13.10H$4L2=13.17H$COSINE(B)=E12.5$)
1036 FORMAT(1H0$28X$4H$P=13.17H$SINE(C)=E12.5$)
1037 FORMAT(1H0$28X$4H$P=13.17H$SINE(C)=E12.5$)

END

Point to Peak Distance Calculation

ATDIST DETERMINES DISTANCES OF ATOMS FROM ANY POINT IN UNIT CELL

DIMENSION NAME(12),N(500),X(500),Y(500),
1Z(500),NSI(2001),MS2(200),SV(200),NI(5000),N2(5000),
2V(111),NINT(100),AV(100),AV(200),EDM(100),NTB(100),B(5000),TT(111),
3TV(11),DIS(11),DI(100,11),
4COSDF(X)=COSF(3.14159*X/180.),
ITP=5
JTP=6
READ INPUT TAPE ITP,1000,(NAME(I),I=1,12)
READ INPUT TAPE ITP,1200,A,B,C,AL,AB,GA
READ INPUT TAPE ITP,1100,BMX
READ INPUT TAPE ITP,1200,XC,YC,ZC
BMX=BMX+2
I=0

100 I=I+1
READ INPUT TAPE ITP,1100,X(I),Y(I),Z(I)
105 N(I)=I-1
IF(ABSF(X(I))+ABSF(Y(I))+ABSF(Z(I)))>110,110,100
110 JJ=0
KK=0

115 READ INPUT TAPE ITP,1200,T1,T2,T3,U1,U2,U3
IF(ABSF(T1)+ABSF(T2)+ABSF(T3)+ABSF(U1)+ABSF(U2)+ABSF(U3))>140,140,1
120 DO 130 J=1,1
111 N(J)=I-1
X(I)=T1+U1*X(J)
Y(I)=T2+U2*Y(J)
130 Z(I)=T3+U3*Z(J)
GO TO 115
140 CAL=COSDF(AL)
CBE=COSDF(BE)
CGA=COSDF(GA)
150 DO190 J=1,11
157 V1=X(J)-XC
V2=Y(J)-YC
V3=Z(J)-ZC
VEC=ABSF(V1*X2+V2*Y2+V3*Z2)
160 IF(VEC-BMX<100,180,190)
170 KK=KK+1
N(KK)=N(J)
180 D(KK)=SQRT(VEC)

Point to Peak Distance Calculation

ATDIST DETERMINES DISTANCES OF ATOMS FROM ANY POINT IN UNIT CELL

DIMENSION NAME(12),N(500),X(500),Y(500),
1Z(500),NSI(2001),MS2(200),SV(200),NI(5000),N2(5000),
2V(111),NINT(100),AV(100),AV(200),EDM(100),NTB(100),B(5000),TT(111),
3TV(11),DIS(11),DI(100,11),
4COSDF(X)=COSF(3.14159*X/180.),
ITP=5
JTP=6
READ INPUT TAPE ITP,1000,(NAME(I),I=1,12)
READ INPUT TAPE ITP,1200,A,B,C,AL,AB,GA
READ INPUT TAPE ITP,1100,BMX
READ INPUT TAPE ITP,1200,XC,YC,ZC
BMX=BMX+2
I=0

100 I=I+1
READ INPUT TAPE ITP,1100,X(I),Y(I),Z(I)
105 N(I)=I-1
IF(ABSF(X(I))+ABSF(Y(I))+ABSF(Z(I)))>110,110,100
110 JJ=0
KK=0

115 READ INPUT TAPE ITP,1200,T1,T2,T3,U1,U2,U3
IF(ABSF(T1)+ABSF(T2)+ABSF(T3)+ABSF(U1)+ABSF(U2)+ABSF(U3))>140,140,1
120 DO 130 J=1,1
111 N(J)=I-1
X(I)=T1+U1*X(J)
Y(I)=T2+U2*Y(J)
130 Z(I)=T3+U3*Z(J)
GO TO 115
140 CAL=COSDF(AL)
CBE=COSDF(BE)
CGA=COSDF(GA)
150 DO190 J=1,11
157 V1=X(J)-XC
V2=Y(J)-YC
V3=Z(J)-ZC
VEC=ABSF(V1*X2+V2*Y2+V3*Z2)
160 IF(VEC-BMX<100,180,190)
170 KK=KK+1
N(KK)=N(J)
180 D(KK)=SQRT(VEC)
Form Factors for Busing Least Squares

CFORMBUS FORM FACTORS FOR Busing LEAST-SQ. REFIN. PROGRAM
DIMENSION S(33), F(32), NC(25), BLANK(12)
ITP=5
JTP=6
KT=10
REWIND KT
READ INPUT TAPE ITP, 250, (BLANK(I), I=1, 12)
NC(1)=1
NC(9)=2
NC(17)=3
NC(25)=4
5 READ INPUT TAPE ITP,
 1 100*A*A1*B+B1*C+ATOM
   IF(A)30, 30. 10
 10 S(1)=1.55
   DO 20 I=1, 32
20 S(I+1)=S(I)**05
   WRITE OUTPUT TAPE JTP, 200, ATOM
   WRITE OUTPUT TAPE JTP, 210
   WRITE OUTPUT TAPE JTP, 220
   WRITE OUTPUT TAPE JTP, 210
   WRITE OUTPUT TAPE JTP,
   1 230*(S(I)*F(I)+S(I+16)*F(I+16)+I=1, 16)
   WRITE OUTPUT TAPET, 250, (BLANK(I), I=1, 12)
   WRITE OUTPUT TAPET, 240, (F(I)*F(I), I=1, F(I+2), F(I+3), F(I+4),
   F(I+5), F(I+6), F(I+7), ATOM, NC(I), I=1, 25, 8)
   GO TO 5
30 END FILE KT
REWIND KT
CALL EXIT
100 FORMAT(5F8*5, A6)
200 FORMAT(1H1, 9X, A6)
210 FORMAT(1H0)
220 FORMAT(1H0, 9X, THSIN(I)/L*13X.11HFORM FACTOR*29X.7HSIN(I)/L*13X.11HFORM FACTOR)
IRM FACTOR)
230 FORMAT(1H0, 9X, F5*2, 15X, F10*6, 30X, F5*2, 15X, F10*6)
240 FORMAT(7F9*4, F8*4, 1H*F, A6*12)
250 FORMAT(12A6)
END
Variance-Covariance and Atomic Parameter Input
for Busing Function and Error

**CBUSVAR**
VARIANCE MATRIX INPUT FOR BUSING ERROR FUNCTION PROGRAM

```plaintext
DIMENSION A(18528), B(192)
ITP=5
JTP=6
I5=9
REWIND
READ INPUT TAPE ITP
1 100*N*SF
READ INPUT TAPE ITP
1 200*A1*A2*A3
DO 10 I=1,N*3
READ INPUT TAPE ITP
1 M=200*B(I)*B(I+1)*B(I+2)
5 B(I)=B(I)/A1**2
B(I+1)=B(I+1)/A2**2
10 B(I+2)=B(I+2)/A3**2
SF=SF**2
M=0
DO 40 I=1,N
40 M=M+1
A(M)=0
GO TO 40
30 M=M+1
A(M)=B(I)
40 CONTINUE
MN=M/8
IF(M<8*MN)45*41
41 L=M+1
MN=MN*8+8
DO 43 I=L*M
43 A(I)=0
45 IF(SENSE SWITCH 1)46*47
46 MM=1
PUNCH 600
PUNCH 100*N*SF
GO TO 48
47 MM=2
WRITE OUTPUT TAPE I5+100*N*SF
48 K=0
DO 55 I=1,M*8
K=K+1
GO TO(51*50)*MM
50 WRITE OUTPUT TAPE I5+300*A(I)*A(I+1)*A(I+2)*A(I+3)*A(I+4)*A(I+5)*A(I+6)*A(I+7)
1A(I+6)*A(I+7)*K
GO TO 55
51 PUNCH 700*A(I)*A(I+1)*A(I+2)*A(I+3)*A(I+4)*A(I+5)*A(I+6)*A(I+7)
55 CONTINUE
READ INPUT TAPE ITP
1 L=N+1
1J=N/8
IF(N-IJ*8)>80*60
60 L=N+1
N=N+1
GO TO 70
60 L=N+1
N=N+1
GO TO 55
70 I=0
GO TO(81*82)*MM
81 PUNCH 600
```

---

NAVAL RESEARCH LABORATORY

69
FUNCTION ARCSIN(X)

A = 1.5707963
B = 0.17453396
C = 0.08768571
D = 0.00582169
E = 0.00039216
F = 0.00002916
G = 0.00000238
H = 0.00000013

10 A = 1.5707963
20 B = 0.17453396
30 C = 0.08768571
40 D = 0.00582169
50 E = 0.00039216
60 F = 0.00002916
70 G = 0.00000238
80 H = 0.00000013

100 YY = X
110 IF(YY) 120 120
120 Y = ABSF(YY)
130 IF(Y-1.0) 140 280
140 Z = SQRTF(1.0-Y)
150 Z = A + Y * (B + Y * (C + Y * (D + Y * (E + Y * (F + Y * (G + HY))))))
160 IF(YY) 170 190
170 ARCSIN= -Z
180 RETURN
190 ARCSIN = Z
200 RETURN
210 ARCSIN = 0.0
220 RETURN
230 IF(YY) 240 260
240 ARCSIN = A
250 RETURN
260 ARCSIN = -A
270 RETURN
280 CALL ENOJOB
290 END
Subroutine RECIP

```fortran
SUBROUTINE RECIP(AR, BR, CR, AL, ER, R, GA, AA, BB, CC, CSAL, COSB, COSC, COSGA)
  AL = ALR * 14159/150 * 0
  BR = BRD * 14159/150 * 0
  CR = CRG * 14159/150 * 0
  CSAL = COSF(AL)
  COSB = COSF(BR)
  COSC = COSF(CC)
  COSGA = COSF(GA)
  VR = AR + CR * SQRTF(1.0 - CSAL * COSB * COSC + COSG)
  SINAR = SINF(AL)
  SINC = SINF(BR)
  SINGA = SINF(GA)
  AA = BR * CR * SINC / VR
  BB = AR * CR * SINAR / VR
  CC = AR * CR * SINC / VR
  COSB = (COSB * COSC + COSG) / (SINC * SINGA)
  COSB = (COSB * COSC + COSG) / (SINC * SINGA)
  COSB = (COSB * COSC + COSG) / (SINC * SINGA)
  RETURN
END
```

Function DOTPROD

```fortran
FUNCTION DOTPROD(U, V, W, X, Y, Z, B1, B2, B3, COSA, COSB)
  RETURN
END
```

Function SCAFAC

```fortran
FUNCTION SCAFAC(A, B, C)
  SCAFAC = A * EXP(-A1 * (S**2)) + B * EXP(-B1 * (S**2)) + C
  RETURN
END
```

Subroutine MTXMUL

```fortran
SUBROUTINE MTXMUL(M, N, A, B, C)
  DIMENSION A(3*3), B(3*3), C(3*3)
  DO 10 I = 1, M
    DO 10 J = 1, N
      10 C(I, J) = 0
      DO 20 I = 1, M
        DO 20 J = 1, N
          DO 20 K = 1, M
            20 C(I, J) = C(I, J) + A(I, K) * B(K, J)
      RETURN
END
```

Function INTG(A)

```fortran
FUNCTION INTG(A)
  IF (A) GT 20 THEN 10
    10 INTG = 0
    GO TO 20
  END(2.2, 2.2, 2.2)
```
This report contains descriptions and operating instructions for a collection of general utility programs applicable to various phases of crystal structure analysis with computers. All coding is done in FORTRAN II language for 32K core IBM 704 and 7090 machines; the FORTRAN-language programs are completely listed in the Appendix.

The data input to many of the programs has been designed to be compatible with the output of the

1. Mathematical computer programming
3. X-ray diffraction analysis - Data - Processing

I. FORTRAN
II. Norment, H. G.
UNCLASSIFIED


In the form presented in this report, the programs are written for use with the IBM 7090 IBM Monitor system.

UNCLASSIFIED


In the form presented in this report, the programs are written for use with the IBM 7090 IBM Monitor system.

UNCLASSIFIED


In the form presented in this report, the programs are written for use with the IBM 7090 IBM Monitor system.