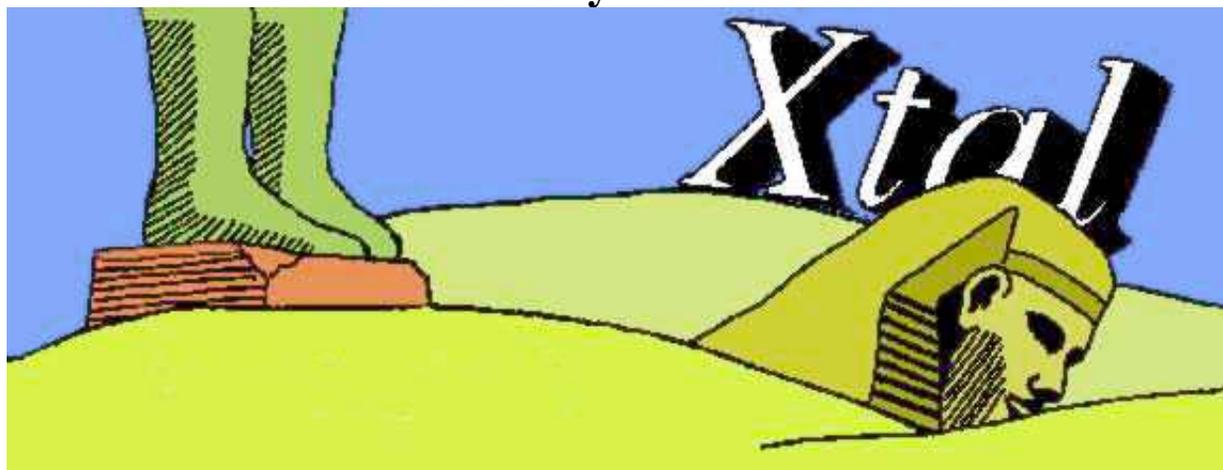


The Gnu Xtal System User's Manual



Edited by
Sydney Hall

Douglas du Boulay

Roeli Olthof-Hazekamp

The Gnu Xtal System User's Manual

Edited by Sydney Hall, Douglas du Boulay, and Roeli Olthof-Hazekamp

Xtal User's Guide version 12 for DocBook V3.0 Edition

Published 2002

Copyright © 1983-2001 The Xtal System of Crystallographic Software. The University of Western Australia.

All Rights Reserved

Copyright © 2002 The Gnu Xtal System of Crystallographic Software.

This is the user's manual for the *Gnu Xtal System* of crystallographic software, version 3.7.2.

Program descriptions prepared by contributing authors.

Compiled and collated by the editors.

<http://xtal.sourceforge.net>

Permission is granted to copy or distribute this manual provided that it is not modified and this copyright notice is retained intact.

Revision History

Revision 3.7.2 Feb 2003 Revised by: ddb

Twelfth edition. Docbook, sgml

Revision 3.7 Apr 2000 Revised by: srh, ddb

Eleventh edition. Html only

Revision 3.6 Apr 1997 Revised by: srh, ddb

Tenth edition. Linuxdoc - html only

Revision 3.5 Apr 1997 Revised by: srh

Ninth+1/2 edition

Revision 3.4 Feb 1996 Revised by: srh

Ninth edition. Last bound/published manual

Revision 3.2 Sept 1992 Revised by: srh

Eighth edition

Revision 3.0 Jun 1990 Revised by: srh

Seventh edition

Revision 2.6 Apr 1989 Revised by: srh

Sixth edition

Revision 2.4 Mar 1988 Revised by: srh

Fifth edition

Revision 2.2 Mar 1987 Revised by: srh

Fourth edition

Revision 2.0 Mar 1986 Revised by: srh

Third edition

Revision 1.5 Mar 1985 Revised by: srh

Second edition

Revision 1.0 Dec 1983 Revised by: srh

First edition of Xtal (derived from XRAY76)

Table of Contents

1. Preface	1
1.1. <i>Support and communications</i>	2
1.2. <i>Acknowledgement of Xtal programs</i>	2
2. Primer	5
2.1. <i>Xtal Files</i>	5
2.2. <i>System Terminology</i>	8
2.3. <i>Example Inputs</i>	10
3. Program controls	17
3.1. <i>System Commands</i>	17
3.2. <i>ABSORB</i> : Absorption corrections.....	18
3.3. <i>ADDDATM</i> : Load atom parameters.....	20
3.4. <i>ADDPAT</i> : Create powder pattern bdf.....	22
3.5. <i>ADDREF</i> : Load reflection data.....	23
3.6. <i>ATABLE</i> : Tabulate atomic parameters.....	26
3.7. <i>BAYEST</i> : Bayesian estimates of I and F	26
3.8. <i>BONDAT</i> : Generate atom coordinates.....	26
3.9. <i>BONDLA</i> : Bond lengths and angles.....	27
3.10. <i>BUNYIP</i> : Search for Additional Symmetry.....	28
3.11. <i>CBAZA</i> : Add crystal based azimuthal PSI angle.....	29
3.12. <i>CHARGE</i> : Atomic charges from difference density.....	30
3.13. <i>CIFIO</i> : Input and output CIF data.....	30
3.14. <i>CIFENT</i> : CIF entry of diffractometer data.....	31
3.15. <i>CONTRS</i> : Contour a map.....	31
3.16. <i>CREDOC</i> : Check cell choice.....	33
3.17. <i>CRILSQ</i> : Structure factor least-squares refinement 2.....	33
3.18. <i>CRISP</i> : <i>Well baked</i> structural solution.....	34
3.19. <i>CRYLSQ</i> : Structure factor least-squares refinement.....	36
3.20. <i>DIFDAT</i> : Process diffractometer data.....	39
3.21. <i>DYNAMO</i> : Molecular Dynamics structure factor calculation NEW	42
3.22. <i>EXPAND</i> : Expand archive atom sites NEW	43
3.23. <i>FC</i> : Calculated structure factors.....	44
3.24. <i>FOURR</i> : Fourier maps.....	45
3.25. <i>GENEV</i> : Normalized structure factors (E values).....	47
3.26. <i>GENSIN</i> : Generate triplets and quartets.....	49
3.27. <i>GENTAN</i> : Tangent phasing.....	51
3.28. <i>GIP</i> : Reciprocal space PIG.....	55
3.29. <i>LATCON</i> : Refine lattice parameters.....	55
3.30. <i>LISTFC</i> : List formatted reflection data.....	56
3.31. <i>LSABS</i> : Absorption corrections.....	57
3.32. <i>LSLS</i> : Structure factor least-squares refinement.....	58
3.33. <i>LSQPL</i> : Least-squares planes & lines.....	60
3.34. <i>LSRES</i> : Restraint editor.....	60
3.35. <i>MAPLST</i> : List map densities.....	61
3.36. <i>MAPXCH</i> : Import theoretical electron density maps NEW	61
3.37. <i>MODEL</i> : Build molecule from map.....	61
3.38. <i>MODHKL</i> : Modify reflection data.....	62
3.39. <i>NEWCEL</i> : Apply unit-cell change.....	63
3.40. <i>NEWMAN</i> : Newman projection of model.....	64
3.41. <i>ORTEP</i> : Molecular plot.....	65

3.42. <i>OUTSRC</i> : Export archive for external software NEW	75
3.43. <i>PARTN</i> : Hirshfield partitioning	75
3.44. <i>PATSEE</i> : Search for molecular fragment	75
3.45. <i>PEKPIK</i> : Search map for peaks	77
3.46. <i>PIG</i> : Display and manipulate model	78
3.47. <i>PLOTX</i> : Interface to graphics devices	79
3.48. <i>POWGEN</i> : Powder pattern graphics	81
3.49. <i>PREABS</i> : Interactive crystal shape manipulation	85
3.50. <i>PREPUB</i> : Pre-publication tests on CIF structural data	87
3.51. <i>PREVUE</i> : Modify text in plots	87
3.52. <i>REFCAL</i> : Process reflection data	87
3.53. <i>REFM90</i> : Convert SCFS-90 file to/from a BDF	88
3.54. <i>REGFE</i> : Analyse errors	88
3.55. <i>REGWT</i> : Analyse weights	90
3.56. <i>RFOURR</i> : Reverse Fourier transform	91
3.57. <i>RIGBOD</i> : Generate a Rigid Group	92
3.58. <i>RMAP</i> : Search for translated fragment	93
3.59. <i>RSCAN</i> : Analyse R-factors	94
3.60. <i>SCATOM</i> : Calculate pseudo-atom form factor. NEW	94
3.61. <i>SHAPE</i> : Patterson deconvolution for structure solution	94
3.62. <i>SHELIN</i> : Enter SHELX file	95
3.63. <i>SIMPEL</i> : Symbolic-addition phasing	95
3.64. <i>SLANT</i> : Slanted map	97
3.65. <i>SORTRF</i> : Sort/merge reflection data	99
3.66. <i>STARTX</i> : Create initial archive file	99
3.67. <i>SURFIN</i> : Render 3D electron density isosurfaces NEW	102
3.68. <i>VUBDF</i> : Dump archive file	102
3.69. <i>XTINCT</i> : Extinction from equivalent reflections	103
4. Reference Manual	105
4.1. <i>System Control Lines</i>	105
4.2. <i>ABSORB</i> : Absorption corrections	108
4.3. <i>ADDATM</i> : Load atom parameters	113
4.4. <i>ADDPAT</i> : Create powder pattern bdf	115
4.5. <i>ADDREF</i> : Load reflection data	119
4.6. <i>ATABLE</i> : Tabulate atomic parameters	125
4.7. <i>BAYEST</i> : Bayesian estimates of I and F 	126
4.8. <i>BONDAT</i> : Generate atom coordinates	128
4.9. <i>BONDLA</i> : Bond lengths and angles	131
4.10. <i>BUNYIP</i> : Search for Additional Symmetry	133
4.11. <i>CBAZA</i> : Add crystal based azimuthal PSI angle	134
4.12. <i>CHARGE</i> : Atomic charges from difference density	135
4.13. <i>CIFIO</i> : Input and output CIF data	137
4.14. <i>CIFENT</i> : CIF entry of diffractometer data	139
4.15. <i>CONTRS</i> : Contour a map	139
4.16. <i>CREDOC</i> : Check cell choice	144
4.17. <i>CRILSQ</i> : Structure factor least-squares refinement 2	147
4.18. <i>CRISP</i> : <i>Well baked</i> structural solution	149
4.19. <i>CRYLSQ</i> : Structure factor least-squares refinement	151
4.20. <i>DIFDAT</i> : Process diffractometer data	162
4.21. <i>DYNAMO</i> : Molecular Dynamics structure factor calculation NEW	168
4.22. <i>EXPAND</i> : EXPAND archive atom sites NEW	171

4.23. <i>FC</i> : Calculated structure factors.....	172
4.24. <i>FOURR</i> : Fourier maps	173
4.25. <i>GENEV</i> : Normalized structure factors (E values).....	178
4.26. <i>GENSIN</i> : Generate triplets and quartets	185
4.27. <i>GENTAN</i> : Tangent phasing	192
4.28. <i>GIP</i> : Reciprocal space PIG	203
4.29. <i>LATCON</i> : Refine lattice parameters.....	207
4.30. <i>LISTFC</i> : List formatted reflection data	210
4.31. <i>LSABS</i> : Absorption corrections.....	212
4.32. <i>LSLS</i> : Structure factor least-squares refinement	215
4.33. <i>LSQPL</i> : Least-squares planes & lines	224
4.34. <i>LSRES</i> : Restraint editor.....	226
4.35. <i>MAPLST</i> : List map densities.....	227
4.36. <i>MAPXCH</i> : Import theoretical electron density maps NEW	228
4.37. <i>MODEL</i> : Build molecule from map.....	230
4.38. <i>MODHKL</i> : Modify reflection data	233
4.39. <i>NEWCEL</i> : Apply unit-cell change	235
4.40. <i>NEWMAN</i> : Newman projection of model.....	236
4.41. <i>ORTEP</i> : Molecular plot.....	238
4.42. <i>OUTSRC</i> : Export archive for external software NEW	243
4.43. <i>PARTN</i> : Hirshfield partitioning.....	244
4.44. <i>PATSEE</i> : Search for molecular fragment	246
4.45. <i>PEKPIK</i> : Search map for peaks	250
4.46. <i>PIG</i> : Display and manipulate model	252
4.47. <i>PLOTX</i> : Interface to graphics devices.....	261
4.48. <i>POWGEN</i> : Powder pattern graphics	263
4.49. <i>PREABS</i> : Interactive crystal shape manipulation.....	265
4.50. <i>PREPUB</i> : Pre-publication tests on CIF structural data.....	270
4.51. <i>PREVUE</i> : Modify text in plots.....	272
4.52. <i>REFCAL</i> : Process reflection data	274
4.53. <i>REFM90</i> : Convert SCFS-90 file to/from a BDF	278
4.54. <i>REGFE</i> : Analyse errors	280
4.55. <i>REGWT</i> : Analyse weights	282
4.56. <i>RFOURR</i> : Reverse Fourier transform	286
4.57. <i>RIGBOD</i> : Generate a Rigid Group	289
4.58. <i>RMAP</i> : Search for translated fragment	292
4.59. <i>RSCAN</i> : Analyse R-factors	293
4.60. <i>SCATOM</i> : Calculate pseudo-atom form factor. NEW	294
4.61. <i>SHAPE</i> : Patterson deconvolution for structure solution	296
4.62. <i>SHELIN</i> : Enter SHELX file	299
4.63. <i>SIMPEL</i> : Symbolic-addition phasing.....	302
4.64. <i>SLANT</i> : Slanted map.....	309
4.65. <i>SORTRF</i> : Sort/merge reflection data.....	311
4.66. <i>STARTX</i> : Create initial archive file	315
4.67. <i>SURFIN</i> : Render 3D electron density isosurfaces NEW	318
4.68. <i>VUBDF</i> : Dump archive file.....	320
4.69. <i>XTINCT</i> : Extinction from equivalent reflections	322

A. Archive file map	327
A.1. 1 <i>hist</i> (F) File History	328
A.2. 2 <i>labl</i> (F) Label Information.....	328
A.3. 4 <i>cell</i> (F) Cell Constants.....	328
A.4. 5 <i>symm</i> (F) Symmetry Information	329
A.5. 6 <i>sgnm</i> (F) Space Group Names.....	330
A.6. 7 <i>twin</i> (F) Twin Laws	330
A.7. 8 <i>scat</i> (D) Atom-type Parameters.....	330
A.8. 9 <i>sequ</i> (F) Protein Atom Sequence Information.....	331
A.9. 10 <i>ddef</i> (F) Dataset Names	331
A.10. 11 <i>expl</i> (D) Experimental Parameters	331
A.11. 12 <i>dset</i> (D) General Data per Dataset.....	334
A.12. 13 <i>dict</i> (D) Macromolecular Residue Information.....	335
A.13. 14 <i>graf</i> (D) Molecular Graphing Information	335
A.14. 16 <i>atom</i> (D) Atom Parameters.....	336
A.15. 17 <i>cons</i> (D) Refinement Constraints	337
A.16. 18 <i>slav</i> (F) Protein Atom Link Information	337
A.17. 19 <i>mole</i> (D) Molecular Information.....	338
A.18. 20 <i>refl</i> (D) Reflection Information.....	338
A.19. 21 <i>thet</i> (F) Precise Angle Data for Cell Dimensions.....	341
A.20. 30 <i>sfls</i> (D) Structure Factor Least Squares Data.....	341
A.21. 31 <i>rfac</i> (F) Refinement History	342
A.22. 35 <i>dens</i> (D) Electron Density Data	342
A.23. 40 <i>bond</i> (D) Bond Distances	342
A.24. 41 <i>cont</i> (D) Contact Distances	343
A.25. 42 <i>angl</i> (D) Bond Angles	343
A.26. 43 <i>dihe</i> (D) Dihedral Angles	344
A.27. 44 <i>rigi</i> (D) Rigid Link Restraint Information.....	344
A.28. 65535 <i>endrecord</i> : (F) EOF Record	345
B. Symmap file	347

List of Figures

- 4-1. Screen shot of GIP 203
- 4-2. Screen shot of PIG 253
- 4-3. Screen shot of POWGEN 264
- 4-4. Screen shot of PREABS 266
- 4-5. Screen shot of PREVUE 272
- 4-6. Screen shot of SURFIN 318

Chapter 1. Preface

In most respects the structure, and the presentation, of the *Gnu Xtal System*¹ of Crystallographic Programs is little different to that of any recent versions. Previous users of this system will find that standard calculations are largely unchanged, though perhaps with the exception of STARTX.

In fact little change can be expected in the future. With the termination of actively funded maintenance any future advances of the suite lie in the hands of any volunteers with a penchant for hacking RATMAC/fortran77/C/tcl/tk code.

Xtal was originally developed with a goal toward being as versatile and comprehensive a suite of crystallographic software as was realistically possible. The cessation of actively funded development has dictated a more concessionary approach towards complementary software. To that end the program SHELIN has been modified slightly to read supplementary SHELX `.ins` files² and DIFDAT has been modified to read SHELX fixed format `.hkl` files (F² only). The program OUTSRC has been added to aid in creating VALRAY, XD and SHELX format input files from an Xtal archive. Those are probably the most useful modifications because they allow Xtal to play nicely with Louis Farrugia's WinGX suite in a MS Windows environment. In addition the program MAPXCH has been added which aims to read in Wien2k electron density maps and ABINIT valence density maps (both subject to limitations), for contouring and plotting in Xtal.

STARTX has now been extensively rewritten. All elemental data tables previously embedded within the source are now externalized in a CIF format elemental database. That database also contains the multiwavelength atomic dispersion and absorption correction factors calculated and compiled in S.Sasaki (1989), KEK Report, 88-14, 1-136 and S.Sasaki (1990), KEK Report, 90-16, 1-143. Consequently STARTX automatically retrieves these factors for wavelengths in the range 0.1 to 2.8 Angstroms. The CIF elemental table is however rather large and reading data from the CIF is now a significant fraction of the total computation time of typical multiprogram calculations. With a 1Ghz cpu This could take up to 10 seconds. In STARTX upd mode, reading of the `atomtype.cif` is eliminated where practical.

Xtal is becoming increasingly dependent on external libraries and software. In addition to tcl/tk, this version gratefully acknowledges the use of [CIFtbx Version 2.6.3](#) (Hall, S. R. and Bernstein, H. J., J. Appl. Cryst. (1996). 29, C 598-603). Furthermore, with the introduction of the new program SURFIN, Xtal now incorporates the [Togl](#) OpenGL module for Tcl/Tk.

The adoption of Tcl/Tk based graphics in Xtal3.7 certainly aided with the portability of the graphically oriented programs. There were however some shortcomings with the tcl/tk library version dependencies which hopefully have been resolved now in a more acceptable manner. Now by statically linking against a stubs library, the actual runtime library used can be specified by environment variables, rather than a specific library linked against at compilation time. Previously this wasn't such an important issue, but new improved tcl/tk libraries are continually being produced and these will hopefully all be compatible with the current version of Xtal.

A new program EXPAND has been added to aid in conversions of structures between different symmetries, and possibly different cells. It permits the expansion of the cell contents up to three or more unit cells, and then the contraction back to some unique set once the cell setting or symmetry have been changed.

Program documentation in the form of this manual is now encoded using the [DocBook](#) SGML dtd and rendered into online html form using the [DocBook dsssl stylesheets](#) and [OpenJade](#). This also provides a route to hardcopy printable versions of the manual via pdfjadetex. A HTML based document detailing how to program in RATMAC for Xtal is available separately, along with a HTML based manual for the now no longer supported XTAL programs.

The Xtal binary executable originally read only from the stdin filestream and wrote only to the stdout filestream. From version 3.7 arguments on the command line passed to the binary were assumed to be input `filename[.dat|.inp]` and output `filename[.lst]` filenames. Two command line options were then added; `-n` which supresses treatment of the first argument as an input filename, and `-p` which forces output listings to be printed to standard output filestream. In the event that only one input filename argument is provided and

in the absence of the `-p` optional argument, an output `filename[.lst]` is interpolated from the supplied input filename. This facilitates the use of xtal within a MS Windows environment (in particular WinGX v \geq 1.64.05), where filenames with the extensions `[.dat]` can be configured to activate the xtal binary directly (via double mouse clicks). For use in this manner it is necessary to have the following environment variables set globally within the users working environment:

variable	example	purpose
XTALHOME	C:\xtal372\bin\	path to xtal library files
XTALTCLLIB	C:\tcl\bin\tcl84.dll	path to Tcl dynamic link library(>=8.2) (.so for unix)
XTALTKLIB	C:\tcl\bin\tk84.dll	path to Tk dynamic link library (.so for unix)
POVEXE	C:\Program files\POV-Ray\bin\pvengine.exe	POV-Ray executable

Xtal behaviour under UNIX and Microsoft Windows environments should be largely identical. Unfortunately VMS support appears unlikely in the foreseeable future.

1.1. Support and communications

The Gnu Xtal System is distributed under the GNU General Public Licence as published by the Free Software Foundation, in the hope that it will be useful but WITHOUT ANY WARRANTY. See the GNU General Public License for more details.

Xtal now lives at SourceForge. Users are encouraged to report bugs etc using the resources that SourceForge provides. If you can isolate bugs and are bloody minded enough to fix them, please submit patches to the mailing list or the caretaker, or better yet, enlist as a developer and get write access to the CVS source tree. Active support is no longer provided by any of the past developers, coordinators, editors, nor yet by the interim caretaker, so get involved and support yourselves.

WWW server: <http://xtal.sourceforge.net>
The latest Xtal sources and executables: <http://sourceforge.net/projects/xtal/>
Mailing lists: <http://sourceforge.net/projects/xtal/>

1.2. Acknowledgement of Xtal programs

Xtal was developed by research workers supported by grants from a wide range of agencies. The names of the program authors are listed at the start of each program description and it is important that acknowledgement is given to the author(s) in any resulting publications and presentations.

Please use the following acknowledgement formats.

General use of the System:

Hall, S.R., du Boulay, D.J. & Olthof-Hazekamp, R. (2000) Eds. *Xtal3.7 System*. University of Western Australia.

Individual calculations:

<<author name(s)>> (1999) <<program name>> *Xtal3.7 System*. Eds. S.R. Hall, D.J. du Boulay & R. Olthof-Hazekamp. University of Western Australia.

Editor Contacts

Syd Hall

Doug du Boulay

Roeli Olthof-Hazekamp

Crystallography Centre

University of W Australia
Nedlands 6907
Australia
syd@crystal.uwa.edu.au

Materials and Structures
Laboratory

Tokyo Inst. of Technology
Nagatsuta, Midori Ku
Yokohama, Japan
ddb@R3401.msl.titech.ac.jp

Laboratorium voor Algemene
Chemie

Universitet Utrecht
3508 TB Utrecht
The Netherlands
roh@chem.ruu.nl

Twelfth Edition, February 2003

Notes

1. Thanks to Lachlan Cranswick for the name change suggestion.
2. Thanks to Louis Farrugia for testing and feedback.

Chapter 2. Primer

Author: Roeli Olthof-Hazekamp, Laboratorium voor Algemene Chemie, Padualaan 8, 3508 TB Utrecht, The Netherlands

This section provides an introduction to the terminology and input language of the *Xtal System*. It covers topics such as file structures and general terminology. It also shows some example inputs to guide new users.

2.1. *Xtal* Files

Most *Xtal* calculations are initiated, and controlled, from a input line file usually known as the `.dat` file. The results of a calculation are output to files and a screen device (as in graphics applications). *Xtal* calculations use various files to store intermediate and archival data. This strong dependence on *files* means that *Xtal* is considered a *file-driven archival* system, rather than an interactive menu-driven system. This is not strictly true because of the interactive graphics programs such as *PIG* and *PREVUE*, but for most calculations the file-driven approach has been adopted as the "*Xtal* philosophy" for the most efficient use of both computing and human resources.

2.1.1. File types

Xtal employs two basic file types: *line* (character) and *binary*.

Line files are composed of ascii text which may be manipulated with a normal text editor. The two principal line files are the line *inputfile* (*compid.dat*) and *print* file (*compid.lst*). The former contains data that determines which and how calculations are performed, and the latter is the printed results of these calculations. *Xtal* uses other line files for certain calculations. These include a *punch* file, *formatted sequential* files, *formatted direct-access* files, various ancillary database files (e.g. *cifmap*, *cifdic*, *symmap*) and an *error* message file. Later there will be a more detailed description of the format of line files.

Binary files are used extensively as working files (ie. intermediate files for exchanging within, and between, calculations), and to archive certain types of data. Binary files are more efficient than line files for certain operations because of their compactness and because they do not require decoding or encoding when data is extracted from or transferred to computer memory. A binary file is often referred to in *Xtal* documentation as a '**bdf**' (binary data file).

Xtal binary files are divided into two categories: *archive* and *auxilliary*. **Archive** bdf's are used as an archive record of accumulated crystal structure data. These files are updated as the analysis progresses. For example, *lratom:* and *lrrefl:* are stored in an archive file. The data structure for archive files is carefully defined for all programs (see the Archive Files description below and the *Archive* section at the end of this manual).

Auxilliary bdf's are used to transfer data from one calculation to another. They are created by a calculation for a specific data exchange purpose. Electron density maps are typical of data stored in an auxiliary file. The actual format of these files is similar to the archival files, but their contents are only known to the programs that write or read them. These are dicussed in detail later.

2.1.2. Filename syntax

The convention for filename construction for all *Xtal* files is normally `<compid>.<ext>` where *compid* is the compound identification code (maximum of six characters), and *ext* is the extension code (three characters). The *compid* code is defined by the user at the start of the input commands with the line "`compid compid`" (see *System*). [Note: If needed at your site, the filename construction can be changed via the system macro *filename:* in the supplied file *xmacro.r.*]

2.1.3. Archive Files

2.1.3.1. Binary

The concept of an *archive* bdf has been introduced above. Most *Xtal* programs employ two archive binary files; one for input and one for output. These have filename extensions `.aa1` and `.aa2`. At the start of an *Xtal* run the file `.aa1` is assumed to be the input archive file, and this file will be read by the first calculation requiring data from the archive file. If a calculation adds to, or changes archive data, this will initially be written to the file `aa2` so that it contains the latest archive data. At the close of this calculation the nucleus will reverse the roles of the two archive files so that `aa2` becomes the input file and `aa1` becomes the output file. This interchange process is repeated for every calculation that writes data to an archive file.

2.1.3.2. Master

Xtal provides a procedure for automatically saving the latest archive bdf. This is referred to as the *master file option*. When this is switched *on* the most recent archive binary file (`aa1` or `aa2`) is saved as an ascii file with the filename extension `aaa` (for details of the format read the CIFIO description if the `arc` file which is identical). Unlike the binary files `aa1` or `aa2`, the master file `aaa` may be read or edited using a normal text/screen editor, and is independent of machine or compiler types.

The master file option is set during installation via the definition of the the macro *masterfile:*. This option may also be set at run time using the **master** line (described in 2. below). If this option is switched off, the user is responsible for ensuring that at the start of a calculation the latest archive binary file is labelled `aa1`. If the master file option is switched on, the file `aaa` is automatically converted to the binary file `aa1` at the start of a calculation sequence and the latest binary file is automatically converted to the file `aaa` at the end of an *Xtal* run. Note that the `aaa` file is generated only when an *Xtal* run terminates with a **finish** line or an *Xtal* error. It will not be generated if a program terminates through a program execution error or an operating system fault. Note that if an `aaa` file does not exist at the start of a run, *Xtal* assumes that the file `aa1` is the input archive binary file.

The need for master archive files will depend on local applications and on the local computing environment. If you are an expert user, or there is a shortage of disc space, or there are very large files, the extra time and space needed for master file operations may be unacceptable. Where *Xtal* is being applied to many analyses simultaneously this option should be useful to ensure that the latest archive file is retained after each run. It is recommended that master file be made the default condition (i.e. set *masterfile:* to *yes:* on installation) and users switch it off with a **masterno** line if it is not required.

2.1.4. Auxiliary files

Unlike the archive files, auxiliary files are only used for the storage of intermediate data. Programs which write auxiliary files assign filename extensions to files according to their contents. Files with the extensions `xx*` are scratch files and may be deleted after a calculation run.

2.1.5. Format of line files

Xtal line files may be free format or fixed format. Character data is case insensitive (i.e. upper- or lower-case alphabetical characters are treated identically except for atom site labels).

2.1.5.1. Concept of a field

A *field* is a string of contiguous characters delimited by a blank. Every input line must start (in column 1) with a line identification code. This requirement may be overridden with the **setid** system command (see the **setid** description in [System](#)). The identification code constitutes field 0 of the input line and must terminate

with a blank. In free format input mode subsequent fields are numbered from left to right, 1, 2, 3, etc. If the system lines **field** and/or **order** are entered, the widths and/or the numbering of the input fields may be specified by the user (the **field** and **order** descriptions are in 2. below).

Note that the sequential numbering of fields should not be confused with the order that data is entered in a given program. For many calculations data may be entered in any order. This is clearly indicated in the *User's Guide* when no field numbers are attached. This is usually the case when numerical data follows control codes (e.g. see the **reset** line described in *System*).

2.1.5.2. Types of fields

There are three distinct types of input field; the *numerical* field, the *alphanumeric* field, and the *void* field. They are defined as follows:

Numerical field: a set of contiguous characters composed of the integers 0 to 9 and characters plus, minus and period. All numerical fields are entered and stored as real numbers, independent of whether a decimal point is entered. A number in 'scientific notation' is entered without the letter 'E' used by Fortran formatting routines. For example, the 0.000001 may be entered as 1.0-6. The numbers in the following fields are identical

```
3 3.0 30.-1 +3 +3. +.3+1 +30.-1 30-1 .03+2
```

Alphanumeric field: a set of contiguous characters containing at least one alphabetic character (a to z and A to Z). The character strings A123 and 123A are treated as alphanumeric fields. Internally the nucleus treats numerical and alphanumeric fields quite differently.

Void field: a field that contains only a blank. Internally this type of field is identified as a floating-point word with a value defined by the macro *voidflg*:. The default value for *voidflg*: is -4.0+20 -- *this value must not be used for any other purpose*.

2.1.5.3. Field delimiter character: *blank*

A field is delimited by any contiguous sequence of blanks.

2.1.5.4. Field skip control: *\$n*

The digraph *\$n* in an input line is interpreted as a signal to skip *n* fields. That is, this digraph causes the internal field pointer to be incremented by *n*. The digraph is not stored in the input buffer. For example, the following line will cause the number 5 to be placed in field 1 and the number 9 to be placed in field 7.

```
param 5 $5 9
```

2.1.5.5. Field position control: **n*

The digraph **n* in the input line is interpreted as a signal to set the internal field position pointer at *n*. Data immediately following this digraph will be placed in field *n* of the input buffer. For example, **10* places the next string encountered into field 10. Field position strings are not stored in the input buffer. Once set, subsequent fields are stored in *n+1, n+2*, etc. unless another field pointer directive is entered. For example,

```
map 2.6 *6 C $5 1 .31
```

will result in the number 2.6 being stored in field 1, the character c being stored in field 6, and the numbers 1. and 0.31 being stored in fields 12 and 13, respectively. All input fields except 1, 6, 12, and 13 will be set to void. Note that field position numbers do not have to be used in ascending order. The above data input could be entered as:

```
map *6 C *1 2.6 *12 1 .31
```

2.1.5.6. Input line delimiter character: ':'

The character colon (:) is used to signal the end of input fields. Note that a colon in column 1 of a line implies that the line is to be ignored. The colon allows additional unprocessed information to be placed on the input image. For example, the image containing the following line is processed only through to the field containing .035.

```
atom C12 .5 .6 .9 .035 :new site at -x, -y, z  
  
:pig :pig is disabled by the colon
```

2.2. System Terminology

2.2.1. *compid*: compound identification code

The compound identification code is a string of up to six alphanumeric characters entered as field 1 of the `compid` system command line. This code is usually referred to in program descriptions as the compound ID, or simply as the *compid*. The *compid* is inserted at the top of each printed page and used to construct the *Xtal* filenames. Note that if the *compid* is not defined its default value is XTAL. Users must be consistent about the definition of *compid* in order that the correct *Xtal* files will be opened (in other words, a `compid` line must always be entered, or never entered).

2.2.2. *rcode*: reflection status code

Reflection data in the *Xtal* system are assigned *rcodes* which indicate the reliability or condition of measurement. The *rcode* is a integer between 1 and 5.

1. measurement above the threshold of observability (e.g. > 3 sigma)
2. measurement below the threshold of observability (e.g. < 3 sigma)
3. measurement contains uncorrected systematic error
4. Friedel-related measurement is missing from a Friedel pair
5. measurement systematically absent for the space group in use

2.2.3. Atom labels

The rules for atom label construction are:

- An atom label may be up to 24 characters in length.
- An atom label may *not* contain imbedded blanks.
- The alphabetic characters in the atom label are case-sensitive. Upper and lower case characters are preserved.
- An atom label has up to eight sequential component codes numbered 0 to 7. Components 0 and 1 are concatenated; all *other* components are separated by an underline.
 - *Component 0* is the *atom type symbol* which is entered on the **CELCON** line of *STARTX*. This symbol may be up to 8 chars in length. All characters are permitted, except digits '0' to '9' may only precede a '+' or a '-' character to indicate an oxidation state. Normally, but not necessarily, the element symbol is used so

that STARTX can automatically assign scattering factors for the atom type. Typical atom type symbols are: C Cu S2- Cu&Fe Water

- *Component 1* is the *atom sequence code*. This may be any string of characters (excluding an underline) which *must* start with one or more digits. Some permitted sequence codes are: 1 3a1 23molA Examples of atom labels involving components 0 and 1 are: I C7 Cu2+4a C23molA 'C8%a' (this latter label will be plotted as C8 α - see below).

The atom label construction is flexible, visually decipherable and well suited to computer applications. The components can be easily identified and stripped in a single pass from left to right along the label string. Note that the underline separators are only used if higher order components exist. If intermediate components are not needed they may be omitted provided the underline separators are inserted. For example the label C233__ggg is acceptable and decodes as component 0: 'c', 1: '233', 2: ' ' and 3: 'ggg'. There is no requirement that the same number of components be used in each label.

The plot routine *PLOTX* supports upper and lower case characters, Greek characters, and superscripts and subscripts. Each Greek character is signaled by a preceding percent % character. For example the digraph '%b' will be plotted as 'β'; '%D' will be plotted as 'Δ'. A superscripted character is preceded by a caret ^, and a subscripted character by a tilde '~'. Italic characters can be initiated by the vertical bar | character.

2.2.4. *Frel, Fobs, Fcal*: structure factor data

All structure factor data obtained by experimental measurement are entered into the bdf through various data entry programs. This data may be raw intensities from a diffractometer file, or values of F^2 or $|F|$ obtained by some previous data treatment. These data should consist of at least a unique asymmetric set of reflections.

For example, if the quantities supplied to *ADDREF* are intensities, various systematic factors may be applied to reduce this to the *relative* measured structure factor $F(\text{relative})$, or, as it is usually referred to in *Xtal* programs, *Frel*. When F^2 is supplied, its square root is defined to be *Frel*. The *observed* (or measured) structure factor $F(\text{observed})$, or *Fobs*, is defined to be on the correct scale (e.g. in electrons) and is obtained from the product of *Frel* and the appropriate scale factor. Structure factor values which are *calculated* by the programs from atomic coordinates are referred to as $F(\text{calculated})$, or *Fcal*. The structure factor difference *Fdif* is obtained by subtracting *Fcal* from *Fobs*.

2.2.5. *k*: structure factor scale

The structure factor scales are used to place the values of *Frel* on the same scale as *Fcal*, as defined in the *International Tables, Vol. I*. That is, the scale factor is used to convert the value of *Frel* into *Fobs*.

The overall structure factor scale can be estimated with *GENEV* which uses the scattering factor and cell content information. The *enot* option can be entered to avoid placing E values on the binary file. In practice, these estimates have proved to be within 10% of the final scale factors for diffractometer data. In the case of film data, there may be several scale factors corresponding to data measured from different films. These scales are applied using scale group numbers.

These scale factors (whether overall, or for different groups of data) are usually refined by least squares calculations. During this refinement the quantity applied is the reciprocal of the scale factor, although all input is in terms of the actual *Frel* scale factor. Most programs which use the scale factors store and retrieve these quantities from the archive bdf. However, if needed, these values may be superseded by values from **scale** lines. All programs which calculate or refine scale factors will update the output bdf with the new values. The exception to this is the structure factor program *FC* and *RSCAN*.

In addition to the scale adjustments calculated by the least squares, programs such as *FC* and *RSCAN* calculate the quantity k' , the rescale factor $k' = \sum_h F_{\text{cal}} / \sum_h k F_{\text{rel}}$

If the scale k is correct, then k' will be 1.0. It should be noted that the practice of adjusting scale factors when not all atom data is known may be misleading. For example, a difference map for locating hydrogen atoms will tend to exhibit artificially lowered peaks because the scale factor does not yet contain the hydrogen atom contribution.

2.2.6. Atomic displacement parameters

A variety of atomic displacement parameters have been used in the crystallographic literature. *Xtal* uses the isotropic and anisotropic parameters U and U_{ij} exclusively. The isotropic displacement factor in the structure factor expression is

$$t = \exp(-8\pi^2 U s^2)$$

The anisotropic displacement factor in the structure factor expression is

$$t = \exp(-2\pi^2 (U_{11} h^2 a^2 + \dots + 2U_{23} k l b^* c^*))$$

2.3. Example Inputs

This section contains some typical input data sequences for crystallographic calculations. These trace the basic steps from the data reduction of the raw diffraction intensities to the preparation of the publication material. The *Xtal* system divides these different steps of an analysis into separate calculation models. Users must learn which module does what and how these can be chained together in the best way to suit the study. The archive *bdf*'s provide the main communication link between calculations with the auxiliary files supplying a specific data link between some calculations.

Usually several crystallographic calculations are performed together. The number of calculations in each run will vary according to the nature of the analysis and the expertise of the user. In fact, the length of the run is usually dictated by how often the user wishes to check how the calculations are proceeding. Between these runs the archive and auxiliary files retain the history and the accumulated knowledge of the calculation sequence.

Other example inputs are given with the program descriptions. More detailed examples are provided as the supplied test input files **p6122.dat**, **saly.dat**, **diam.dat**, **ags4.dat** and **lac1.dat**. Novice *Xtal* users are advised to print these files as an additional reference.

2.3.1. Step 1: Input primary data

The first step in any analysis is to enter the primary crystallographic information such as cell, symmetry and chemical formula. In *Xtal* this is usually done with the program *STARTX* which is responsible for creating and checking the initial archive file. [Note that all input command sequences in the **.dat** file must start with the `compid` declaration.]

```
compid exampl
title study of the compound C23 H28 O6 P212121 Z=4
STARTX
cell 17.076 16.604 7.425 90. 90. 90.
cellsd .004 .002 .007
sgname P 2ac 2ab : H-M notation P212121
celcon C 92
celcon O 24
celcon H 112
```

Note that it is possible start an analysis with CIF input as well (using *CIFIO*) because this automatically invokes the *STARTX* routine in the process.

2.3.2. Step 2: Input reflection data

The next step, after creating the initial archive file, is to input the reflection data. This may be done in a number of different ways using *DIFDAT*, *ADDREF*, *CIFIO* or *REFM90*. In these examples we will look at the two most common approaches involving *DIFDAT* and *ADDREF*.

If the reflection data are raw intensities or counts from a diffractometer then there needs to be a calculation that reads this data and scales it for conversion to structure factors. *DIFDAT* is usually used to process a diffractometer file.

Here is a control sequence for processing a standard Siemens ascii file labelled *compid.sie*. *DIFDAT* reads this file and stores the scaled intensities on the archive file. *ADDREF* reads the archive file and reduces the intensities into structure factors (note that *ADDREF* stores both $|F_{meas}|$ and $|F_{meas}|^2$ on the archive file). The other data such as diffractometer angles are not entered (ie. absorption corrections will not be applied). If they were on would place `all` on the `bdfin` line. [Note that the `compid` line is omitted from this sequence but if it were used as a separate run it would need to be present].

```
DIFDAT sie baln 3
genscl
ADDREF
reduce itof rlp4 12.2
bdfin hkl irel sigi
```

If one is entering reflection data that has already been scaled it may be entered as part of the `.dat` file as follows.

```
ADDREF
reduce itof rlp4 12.2
hklin hkl rcod irel sigi
hkl 0 0 2 1 21023 644
hkl 0 0 4 1 40 5.97
hkl 0 0 6 1 252 11.4
hkl 0 0 8 1 33.4 4.9
.....reflection data omitted for brevity
hkl 19 11 0 2 .214+00 .333+01
```

Other initial reflection processing calculations are provided by programs such as *SORTRF* and *ABSORB*. Absorption corrections to the intensity data, if required, should be applied before the *ADDREF* step. The *SORTRF* calculation is used to sort the reflection data into a specified order and to average multiply-observed data into an asymmetric set. Most *Xtal* calculations need to operate with a unique asymmetric set of reflections and atoms. For example:

```
SORTRF order hlk aver 1
```

Look carefully at the examples given in the program descriptions.

2.3.3. Step 3: Solving the structure

The next step is to determine the crystal structure. This can be done with a variety of ways. The most automatic approach is to use the new iterative solution program *CRISP*, which amalgamates many of the individual program steps described in the alternate approaches described below. The command needed is simply

```
CRISP
```

Another direct methods approach which does not involve the automatic iterative program CRISP, uses the programs *GENTAN* or *SIMPEL* or *PATSEE*. Here is a typical sequence for these approaches

```
GENTAN
FOURR emap
PEKPIK
PIG
```

For some centrosymmetric structures it may be more effective to run

```
SIMPEL
FOURR emap
PEKPIK
PIG
```

Or in difficult cases where a fragment of the structure is known one should try the sequence

```
GENEV
GENSIN
FOURR epat
PATSEE
```

Alternatively the user can apply the heavy atom method with a combination of the programs *GENEV*, *FOURR*, *PEKPIK* and *FC*. There are many combinations of these calculations, plus routines such as *MODEL* and *PIG*, which can be used to visualise the proposed models.

In these sequences the program *GENEV* calculates the E values needed for direct methods. *GENSIN* generates triplets and quartets and outputs them to the auxiliary file `.inv`. This file is read by subsequent phasing programs such as *GENTAN* (or *SIMPEL* or *PATSEE*). If rerun *GENTAN* for later phasing attempts the `.inv` file should be retained. *GENTAN* has a wide range of options for applying the multi-solution tangent phasing methods. The program *SIMPEL* can replace *GENTAN* in this sequence to determine phases using the symbolic addition approach.

The *FOURR* program inputs the phases estimated by *GENTAN* to produce an E map. This map is output to the file `map`. *PEKPIK* reads the `map` file and searches for the unique set of highest peaks in the asymmetric unit of the density map. These are output to the file `pek` which is in turn read by the programs *MODEL* and/or *PIG*. Both these programs search for interatomic connections between the peaks according a preset molecular geometry based on distance and angle criteria. *PIG* is an interactive graphics version of *MODEL* and has a number of a visualisation advantages. With *MODEL* special care must be taken with the specification of the parameters on the `limits` line as the default values may not be appropriate for the structure being studied. *MODEL* produces a table of bond lengths and angles, a printer plot of the connected peaks, and outputs peak sites as `ATOM` lines to the line file `.pch`. The `ATOM` lines may be used in subsequent calculations such as *BONDLA* or *ADDATM*. Changes to the atomic site information with *PIG* are automatically applied to the archive file.

2.3.3.1. Direct methods strategies.

In the event that solution programs such as *CRISP*, *GENTAN*, *SIMPEL* or *PATSEE* do provide a clear solution of the initial structure, it may be necessary to adjust the standard options used in these calculations. Here is some general advice on the optimal use of these routines. Note that in the above example input only a few command lines are needed. This is because the default controls are appropriate for most analyses. For some analyses additional control signals may be required. Here are some tips for the application of the direct methods programs.

- For the first attempt, use the default values for a structure solution sequence as shown in the above example. This has been shown to solve about 90% of small and medium sized structures automatically.
- For centrosymmetric structures SIMPEL may often be used in place of GENTAN and is about four times faster.

For structures not solved with the default options, try the following alternatives separately. Note that the judicious saving of the bdf's permits you to start calculations at the program in which the option needs to be changed. Do not always start with STARTX, or even GENEV!

- Increase the number of starting phases in GENTAN using the `select` line (particularly for low symmetry structures).
- Decrease Emin in GENSIN to 1.5 - 1.3 (particularly for triclinic).
- If there are sufficient triplets (i.e. >10 times the number of E's), increase the Emin to about 1.6 in GENTAN (or GENSIN).
- Increase the number of triplets and quartets by reducing Amin and Bmin to about 0.8 and 0.4 on the GENSIN lines `trip` and `quar`. Even lower values can be considered for large structures.
- If a fragment is found in the space group P1bar with a possible origin translation, redo STARTX as P1 and locate the origin using the fragment in FC, FOURR, etc. Then return to P1bar for the refinement.
- Exclude quartets from the GENTAN calculation using the `invar` line.
- For noncentrosymmetric structures, change the `magic` option to `permute` on the `select` line of GENTAN. Alternatively, use the `random` option instead of `permute`.
- Apply the `block` and/or `w2` options on the `refine` line of GENTAN.

When assessing a direct methods run for success or failure, do not rely wholly on the GENTAN or SIMPEL figure of merit estimates. For some structure types these will be unreliable. It is good practice to calculate the E maps for the top four (or even eight) phase sets and then look closely at the PEKPIK results. One or two dominant peaks usually signals an incorrect phase set (except if there one or two heavy atoms in the structure!). The best E maps contain a large number of medium height peaks and this may be used as a quite sensitive criterion for identifying the more correct phase sets.

There are many other combinations of options available in GENSIN and GENTAN. If the above alternatives fail you should look critically at the reflection data, both in terms of its precision and high angle limit, and at the possibility of incorrect space group symmetry. In some cases structures are solved automatically by recollecting the data with better precision and/or higher angle data. In cases where there are very high B-values it may be necessary to collect low-temperature data. It cannot be emphasised enough that these methods are very dependent on good E-values. GENEV works well with good data, but if the Wilson plot shows anomalies at high angles it is better to exclude this data from the structure solution (by setting Smax in GENEV). *Always* check the Wilson plots carefully if the default run fails.

2.3.4. Step 4: Refining atom parameters

Once a structure is determined the coordinate and displacement parameters need to be refined. A variety of programs are needed for a refinement sequence. Often the initial atomic coordinates are loaded onto the archive file by PEKPIK and FIG. In other instances all, or some, of the coordinates will need to be loaded with ADDATM.

In this simple example ADDATM is used to load atom data and *CRILSQ* or *CRYLSQ* to refine these parameters. The programs *RSCAN*, *PIG* and *BONDLA* are used to gauge the progress of the refinement and check the reflection data and current model.

```

ADDATM
scale 1.02033 1
atom      C1  .48236  -.11502  .93088  .0332  1.0  .00023  .00024  .00053
uij       C1  .00244  .00391  .01162  -.00008  .00009  -.00021
atom      O1  .47364  -.18767  .83988  .0385  1.0  .00017  .00015  .00039
uij       O1  .00378  .00323  .01594  -.00027  -.00053  .00071
.....atom data omitted for brevity
atom      H6  .75500  -.10800  .93600  .0570  1.0  .00000  .00000  .0000
atom      H7  .64700  -.14500  1.20600  .0680  1.0  .00000  .00000  .0000
atom      H81 .81360  -.11540  1.23320  .0740  1.0  .00000  .00000  .0000
atom      H82 .79070  -.20340  1.18740  .0740  1.0  .00000  .00000  .0000
atom      H83 .78040  -.17080  1.37730  .0740  1.0  .00000  .00000  .0000
CRYLSQ cy 2 ad fm fu 0.7 : CRILSQ could be used instead
noref H
RSCAN
PIG
BONDLA

```

This sequence emphasises the modularity of *Xtal* calculations. There are many other combinations involving many programs. For example, when atoms are still missing from the model (or one is searching for the H atom sites) the following combination is very useful.

```

CRYLSQ cy 2
noref H
FOURR fdif
PEKPIK
PIG input
peaks prad 1.

```

Here the program *PIG* is used to decide which peak sites are to become atom sites (and are labelled) and which should be rejected

2.3.5. Step 5: Checking molecular geometry

Periodically during an analysis the model and refined structure factors need be checked. The model can, of course, be monitored with *PIG*, but many other checking tools are provided in the *Xtal* system. These include *BUNYIP*, *LSQPL*, *BONDLA*, *REGFE* and *RSCAN*, to name a few.

Here is one sort of run which also inserts H atom sites from the non H atom geometry.

```

BONDLA cont dihe
atrad H 1. .5
FOURR fdif r2 0
PEKPIK punch
BONDAT
calcat tetchn 0.95 O5 C1 C2 H1 H11
calcat teterm 0.95 C1 O1 C11 H111 H112 H113
calcat tetchn 0.95 C1 C2 C3 H21 H22
calcat tetchn 0.95 O2 C21 C22 H211 H212
calcat trigon 0.95 C22 C23 C24 H23
calcat trigon 0.95 C23 C24 C25 H24
calcat trigon 0.95 C24 C25 C26 H25
calcat tetchn 0.95 C3 C4 C5 H41 H42
calcat tetchn 0.95 C4 C5 O5 H51 H52

```

```
calcat teterm 0.95 C6 C7 C8 H81 H82 H83
```

BONDAT calculates new coordinates for the H atoms based on the prescribed geometries of the refined non-hydrogen sites. Note that *PIG* could also have been used after *PEKPIK* for viewing and manipulating the atom sites, and for checking the geometry.

2.3.6. Step 6: Preparing for publication

When the structure is ready for publication various molecular plots, and tables need to be prepared. Here are some of the calculations that can be used for this purpose.

```
BONDLA
LSQPL
PLANE
DEFINE C22 C23 C24 C25 C26 C27
nondef O2 C21
PLANE
DEFINE C32 C33 C34 C35 C36 C37
nondef O3 C31
PIG
ORTEP mole acta
PREVUE
PLOTX post1 pre
LISTFC nofl wid 120 lin 64
ATABLE isou
CIFIO cifo
```

2.3.7. General Advice

The examples shown above are a very small sample of the combinations possible with *Xtal* programs. A novice user should experiment with input sequences using known data so that there will be a proper understanding of the various functions. And always remember that the archive bdfs *aa1* and *aa2* are interchanged with every update, so it is best to initially rely on the master file procedure which always reads and generates the latest *aaa* file. The *aaa* file is also readable with a text editor and this provides a new user with the opportunity to learn how the data is archived. However, only the most experienced users should ever attempt to change values in this file with a text editor, and then, only after keeping a copy of the unedited version!

Chapter 3. Program controls

This section of the manual contains information on how to prepare the input command file for calculations in the *Xtal System*. Detailed program descriptions are contained in the next section, referred to as the [Reference Manual](#).

3.1. System Commands

title specify the output print header
compid set the compound identification code
master set the master file switch and specify filenames
remark insert comments into the printed output
copybdf copy one binary data file to another
reset reset printer specifications
setid specify default line identifiers for subsequent linedata
field fix format of subsequent input lines
order reorder fields of input line data
finish end of the input line file
A complete description of these lines is given in the Primer Section:Reference Manual.

title	for printout header, columns 8 to 72		<i>blank</i>
compid	1 filename id (up to six characters)		<i>xtal</i>
master ^a	<i>Option</i>	<i>Code</i>	
	master file support ^b	no	switch master file support off
		yes	switch master file support on
	read different master	read	v copy master file <i>v</i> to binary
			<i>aa1</i>
	write different master	write	w copy final binary file to <i>w</i>

Notes:

- A **master** line, if entered, *must* precede the first program initiation line.
- Default master file support state is set on installation with the masterfile: macro.

remark for inline comments, columns 8 to 72 *blank*

copybdf^a 1 **ext1** filename extension of source bdf
 2 **ext2** filename extension of destination bdf

Notes:

- The **copybdf** line cannot be used prior to the **compid** line and *must* be used either *before* the first program initiation line or *after* an **end** line.

reset	<i>Option</i>	<i>Code</i>	
	list current parameters	(null)	
	reset parameters to installed default values	all	
	max characters per line of printer	cmax	n
	max lines per page of printer	lmax	n
	characters per inch across the page	cpin	n
	lines per inch down the page	lpin	n
	priority status of print lines (normally 3; max 5)	psta	n
	use private graphics display color map ^a	priv	

use common (shared) color map a **comm**

Notes:

- a. These commands are not currently supported for the Win32 version.

setid_a 1 string of up to 6 characters _b

Notes:

- a. Once the **setid** has been entered, all succeeding lines, except the system commands, are treated as having the identification code specified on the **setid** line, until another **setid** is read. When **setid** is in force, field 1 of a line starts in column 1 of the input line (this is field 0 when **setid** is not in force).

- b. A blank string resets to normal input.

field _a 1 rightmost character _b of field 1 for subsequent input lines
 2 rightmost character of field 2 for subsequent input lines
 3- etc

Notes:

- a. If an identification code is entered on an input line which has been formatted by a preceding **field** line, an **order** line will also need to be specified so that this field can be ignored or properly treated. All fields expected as input must be specified on the **field** line. A blank **field** line resets free format input.

- b. The leftmost character of each field is assumed to be one more than the rightmost column number of the previous field. The leftmost character of field 1 is column 1.

order _a 1 new field number for first input field
 2 new field number for second input field
 3- etc

Notes:

- a. A blank **order** line resets the order to 1, 2, 3,

finish end of all calculations. The *end-of-file* will cause the same action.

3.2. **ABSORB**: Absorption corrections

astar _a spherical absorption A* values
tbar _a spherical T values
diff _a diffractometer description
orient _a crystal orientation
orface _b microscope orientation
facept _b face described by three points
face _b face described by χ, ϕ , and distance
faceom _b face described by χ, ϕ, ω and distance
faceml _b face described by Miller indices and distance
grid _c Gaussian grid description

Notes:

- a. Optional according to mode: **diff** and **orient** are mandatory for Gaussian and analytical corrections and are in fact order dependent (the **diff** line MUST precede the **orient** line for correct operation)
- b. Only one of the methods **orface**, **facept**, **face** or **faceml** may be used to specify the faces of the crystal.
- c. Used only for the Gaussian integration method.

ABSORB	<i>Option</i>	<i>Code</i>		
	absorption coefficient in mm ⁻¹	mu	q	<i>auto</i> _a
	absorption method	gauss		Gaussian integration
		analyt		analytical method
		sphere		spherical crystal _b
		cylin		cylinder
		tables		user tables

application of absorption	napp		do not apply	
	irel		apply to I_{rel}	
	f2r1		apply to F_{2rel}	
	frel		apply to F_{rel}	
store T on bdf	ntbar		no	
	tbar		yes ^b	
print reflection data	print	n	print n hkl	99999
limit reflections processed	limit	n	process n hkl	99999
dataset specification	dset	n	dataset number	1
precision scale factor ^c	pscal	q	scale distance and μ	
crystal plot view on file ^{abs}	stereo		stereo view	<i>projection</i>
radius of crystal in mm	rad	q		

Notes:

- Default is to calculate μ from the cell contents on the bdf.
- If the **sphere** option is used, do not use the **tbar** option.
- This option is used to avoid rounding errors for small crystals.

astar ^a 1-19 values of A^* corresponding to $0^\circ, 5^\circ, 10^\circ, \dots, 90^\circ$

Notes:

- More than one **astar** line may be used.

tbar ^a 1-19 values of T corresponding to $0^\circ, 5^\circ, 10^\circ, \dots, 90^\circ$

Notes:

- More than one **tbar** line may be used.

diff ^a	1	sense of rotation of the 2θ circle	c	clockwise	
			a	anti-clockwise	
	2	sense of rotation of the χ circle	c	clockwise	
			a	anti-clockwise	
	3	sense of rotation of the ϕ circle	c	clockwise	
			a	anti-clockwise	
	4	sense of rotation of the ω circle	c	clockwise	
			a	anti-clockwise	
	5	sense of rotation of the ψ circle	c	clockwise	
			a	anti-clockwise	
	6	value of ϕ at $\chi = 90$ degrees			0
	7	minimum possible ω ($-180 < \omega < 180$)			-48
	8	maximum possible ω ($-180 < \omega < 180$)			58

Notes:

- See the Diffractometer description in [ABSORB](#) for the method to be used in determining the sense of rotation for each diffractometer circle.

orient 1-3 hkl indices of first orienting reflection
 4 angle χ of first orienting reflection ^a
 5 angle ϕ of first orienting reflection
 6-8 hkl indices of second orienting reflection
 9 angle χ of second orienting reflection
 10 angle ϕ of second orienting reflection

Notes:

- Angles in the bisecting mode. (For CAD see Ch. II of the CAD User's Manual.)

orface ^a 1-3 x, y, z coordinates of first point on orienting face
 4-6 x, y, z coordinates of second point on orienting face
 7-9 x, y, z coordinates of third point on orienting face

Notes:

a. If the **orface-facept** method is used for entering face data, two **orface** lines must be used. The first contains the coordinates of three points on a plane parallel to the first orienting reflection given on the **orient** line. The second **orface** line contains the coordinates of three points on a plane parallel to the second orienting reflection given on the **orient** line.

facept	1-3	x, y, z coordinates of first point on bounding face		
	4-6	x, y, z coordinates of second point on bounding face		
	7-9	x, y, z coordinates of third point on bounding face		
face	1	χ for bounding face (<i>angles in the bisecting mode</i>)		
	2	ϕ for bounding face		
	3	distance to origin from bounding face in mm		
faceom	1	χ for bounding face (<i>angles in the bisecting mode</i>)		
	2	ϕ for bounding face		
	3	ω for bounding face		
	4	distance to origin from bounding face in mm		
faceml	1-3	h, k, l indices of bounding face		
	4	distance to origin from bounding face in mm		
grid _a	1-3	number of grid points in x, y, z directions		
	4	method of exponentiation	<u>0</u> 1	library function table look-up

8

Notes:

a. The **grid** line is used only if the Gaussian method is selected. Only the intervals 6, 8, 10, 12, 14, 16, 24, or 32 are possible.

3.3. **ADDATM** : Load atom parameters

scale	set F_{rel} scale factors
extinc	set extinction factors
uov	set overall isotropic temperature factor
shift	displace coordinates by shift and/or reversal
invert	invert x, y, z coordinates
xabs	xabs parameter for CRYLSQ
graf	enter orientation matrix for ORTEP
grid _a	set Fourier grid divisions for atomg lines
stack	sort sites in the atom list
rename	rename atom labels in the atom list
delete	delete sites from the atom list
editu	change the U values for an atom or atoms
atom	atom parameters, fractional coordinates
atomg	atom parameters, grid coordinates
u _b	isotropic thermal displacement parameter U and σ U
uij _b	anisotropic thermal displacement parameters U_{ij}
suij _b	σ 's of anisotropic thermal displacement parameters U_{ij}
link _b	calculation method (if used)

Notes:

a. If entered, must follow an **atom** or **atomg** line with the same atom label.

b. Must precede any **atomg** lines entered.

ADDATM Option

Code

or **ani** to convert from isotropic to anisotropic

atom	1	atom label		
	2-4	fractional coordinates x, y, z		
	5	isotropic displacement parameter U	0.035	
	6	population parameter	1	
	7-9	σ 's of x, y, z	0	
	10	σ of isotropic displacement parameter U	0	
	11	σ of population parameter	0	

u	1	atom label		
	2	U		
	3	σ U	0	

uij	1	atom label		
	2-7	$U_{11}, U_{22}, U_{33}, U_{12}, U_{13}, U_{23}$		
suij	1	atom label		
	2-7	$\sigma U_{11}, \sigma U_{22}, \sigma U_{33}, \sigma U_{12}, \sigma U_{13}, \sigma U_{23}$	0	

atomg_a The **atomg** line is identical to the **atom** line except x, y, z are in grid coordinates.

Notes:

- a. The **atomg** line must be preceded by a **grid** line.

link_a	1	atom label				
	2	calculation type	<i>Code</i>	<i>N</i>	<i>M</i>	<i>Angles</i>
			linear	2	1	$\alpha_1 = 0^\circ$ or 180°
			trigon	3	1	
			tetchn	3	2	
			teterm	3	3	α_1
			genral	3	1	$\alpha_1\alpha_2$
			methyn	4	1	
			trterm	3	2	
			median	4	1	
	3	bond distance in Å				
	4...3+N+M	Labels of atoms used in the generation, followed by atom labels of generated atoms				
	..	angles $\alpha_1\alpha_2$				

Notes:

- a. Atom labels must match identically those in the bdf. The number of fields required depends on the calculation. The first N fields will be used for the generators and the next M fields will be used for the generated atoms. The final angle fields will be used only for the specified generation type. The parent atom of atoms built in this method should be related to the archive sites only by the identity symmetry operation. Auxilliary geometry specifying sites may however be symmetry related to the archive sites. In this case the symmetry operation required should be indicated as a variation on the the integral number n555 (where 1555 refers to the identity symop with no translations). **link** lines are only required for *riding model* least squares refinement in **CRILSQ** and should be automatically added for new H atoms built using **PIG** or **BONDAT**

3.4. **ADDPAT**: Create powder pattern bdf

label	data for the lrplab: record
backgr	not used
phases	phase names

wavlen instrumental parameters

ADDPAT	<i>Option</i>	<i>Code</i>		Default	
	input file extension	inp	ext	diffractometer file	pow
	output file extension	out	ext	suitable for POWGEN or RIETVD	pat
	input file type	typ			phi
	Philips APD ASCII files	phi			
	Philips PC-APD files	apd			
	Raw data files	raw			
	WYRIET input files	wyr			
	Universal X-ray data	uxd			
	?	tan			

label 1 character data for the lrlab: record

backgr 1 not used

phases 1 name of first phase
2 name of second phase
3... up to 10 phases

wavlen	1	Wavelength alpha1	1.540598
	2	Wavelength alpha2	0.0
	3	alpha2/alpha1 intensity ratio	0.0
	4	polarisation ratio	.915
	5	radiation type (xray ,electron, neutron or synchr)	xray
	6	fixed divergence slit (0/1=fix/var)	fixed
	7	data lp corrected (0/1=no/yes)	no

3.5. **ADDREF**: Load reflection data

reduce^a conversion factors
transf^a transform Miller indices
limits^a restrictions of reflection data
bdfin reflection items to be input from a bdf
hklin reflection items to be input from **hkl** lines
hklgen reflection items to be generated
remove reflection items to be excluded from output bdf
hkl reflection data as specified on **hklin** line
c^b continuation line

Notes:

a. These lines must precede those that follow. Optional control lines MUST be used in listed order

b. **c** (continuation) control lines may be used directly after **bdfin**, **hklin**, **remove**, and/or **hkl** lines to allow more data items to be entered than will fit on one line.

ADDREF ^a	<i>Option</i>	<i>Code</i>		
	data set number	dset	n	1
	interpolated form factors ^b	noff		do not put ff in <i>refl</i> record
		ffac		put form factor in <i>refl</i>
	data for all reflections	slist		list data for 10 refls
		list	n	list n reflections
	Bayesian treatment to $I < 3\sigma I$	bay		Bayesian estimate if $I < 3\sigma I$

systematically absent hkl	nobay rsabs sabs oabs	do not apply do not output to bdf include in output bdf only put absences in output bdf
insertion of max hkl	smax nmax	set max from input data preserve as on input bdf
save 1/Lp on bdf	nolp lpin	do not output to bdf ^c include in output bdf
Friedel related data	nofr frie	do not treat Friedel pairs treat Friedel pairs

Notes:

- Rejected reflections may be listed by entering the line **reset psta 4**
- See the **ADDREF** section of the ADDREF program description.
- If the **nolp** option is chosen, 1/Lp will be excluded from the output bdf unless it is copied from input by an explicit or implied **bdfin** instruction.

reduce	1	desired data conversion	nocon	no data conversion			
			itof2	intensity to F ₂			
			itof	intensity to F _{rel}			
			f2tof	F _{2rel} to F _{rel}			
			f2toi	F _{2rel} to intensity			
			ftof2	F _{rel} to F _{2rel}			
			ftoi	F _{rel} to intensity			
			2	1/Lp calculation method ^a	r1p1	powder method	
					r1p2	Eulerian cradle geometry and 2θ scan	
					r1p3	diffractometer with perp. monochromator	
r1p4	diffractometer with equatorial monochromator						
		none	set 1/Lp to 1.0 for all reflections				
	3	2θ angle of monochromator ^b		(in degrees) <i>assuming graphite from λ</i>			
	4	monochromator perfection factor		0.5			
	5	incident beam polarization factor before the monochromator		0.5			

Notes:

- For data conversion types **itof2**, **itof**, **f2toi**, and **ftoi**, the 1/Lp calculation method is automatically set to **r1p2**; otherwise, it is set to **none**.
- Fields 4 and 5 are only for 1/Lp calculation methods **r1p3** and **r1p4**.

transf ^{a b} 1-9 transformation matrix a₁₁, a₁₂, a₁₃, a₂₁, a₂₂, a₂₃, a₃₁, a₃₂, a₃₃

Notes:

- The transformation is: h'k'l' = hkl*a-matrix
- As of 3.7.2 the **transf** command was modified to reject reflections that transform to fractional indices. This can be used in conjunction with the **ADDREF oabs** to retain (untransformed) only those reflections which would have been rejected by the transformation. They can then be viewed with **GIP** for instance.

limits _a	1-3	maximum h , k , l output	<i>from field 4</i>
	4	maximum sinθ/λ	1/λ
	5	minimum d-spacing allowed	0
	6	maximum reflection type	5
	7	maximum scale group allowed	64
	8	minimum scale group allowed	1
	9	yes generate whole sphere	no

Notes:

a. The default for each quantity is such that, effectively, no restriction is made. If the **nmax** option has been specified in the **ADDRESS** line, the **limits** line quantities will be ignored.

bdfin	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	input archive bdf	file	ext	read bdf with extension ext aan
	<i>lrrefl</i> : archive items ^a	hkl		Miller indices
		frel		F_{rel}
		sigf		σF_{rel}
		fcal		calculated F
		acal		calculated A
		bcac		calculated B
		irel		relative intensity
		sigi		σI_{rel}
		f2r1		F_{2rel}
		sgf2		σF_{2rel}
		rcod		reflection code (rcode)
		scgp		scale group
		extf		extinction factor
		eval		E value
		absf		absorption correction factor
		tbar		mean beam path length
		pcyc		phase in cycles
		pdeg		phase in degrees
		rlp		1/Lp or 1/L for neutrons
		wgt1		least squares weight 1
		wgt2		least squares weight 2
		wgt3		least squares weight 3
		ifri		intensity of Friedel mate
		sfri		σI of Friedel mate
		f2fr		F_{2rel} of Friedel mate
		sf2f		σF_{2rel} of Friedel mate
		ffri		F_{rel} of Friedel mate
		sfri		σF_{rel} of Friedel mate
		rcdf		rcode of Friedel mate
		scgf		scale group of Friedel mate
		all		all items from logical record <i>lrrefl</i> : ^b

Notes:

a. ID numbers may also be used to specify items to be input. These are listed in the *Archive* section of the Reference Manual. Also see *ADDRESS*.

b. The option **all** will cause all items from logical record *lrrefl*: of this bdf to be read from the file, except $\sin\theta/\lambda$, epsilon, multiplicity, 1/Lp and form factors, which are calculated later in the program. This option may be used only on the **bdfin** line.

hklin ^a	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	input item specification for hkl lines			same options as bdfin line
	skip a field on the hkl lines	skip		

Notes:

a. Data on an **hkl** line must be in the same order as the corresponding fields on the **hklin** line, and likewise for **c** (continuation) lines.

hklgen ^a	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	items written to output archive bdf			same options as bdfin line

Notes:

a. Options must include **hkl** and either **frel**, **f2r1** or **irel**.

remove	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	items on hkl line that should not be placed on the bdf			same options as bdf line

hkl_a 1- data corresponding to each item, in order, on the **hkl** line

Notes:

- a. Exception: h, k, and l each take three fields on the **hkl** line, but are specified using the one mnemonic **hkl** on the **hkl** line. The use of continuation, **c**, lines with **hkl** and **hkl** input lines is allowed but there must be a one-to-one correspondence between the data on the **hkl**-**c** pair and the **hkl**-**c** pairs.

3.6. **ATABLE**: Tabulate atomic parameters

ATABLE	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	use brackets in the atom label e.g. C(44A)			yes
	do not change bdf atom label	nobr		

3.7. **BAYEST**: Bayesian estimates of |I| and |F|

BAYEST	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	number of $\sin^2 \theta/\lambda$ bins	nrange	n	50
	Bayesian correction to	int		intensity
		frel		F _{rel}
	list reflection data	npri		do not print hkl
		print		print hkl

3.8. **BONDAT**: Generate atom coordinates

site temporary atom sites needed for geometry calculations

calcat calculate atom from given atoms in specified configuration

BONDAT	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	update bdf	nu		no update of archive file
		uf		update archive file
	set U for new atoms	dp	u	0.035

site 1 site label_a

 2-4 fractional coordinates x, y, z

Notes:

- a. If label matches site on the bdf, these coordinates will be substituted for the calculation. Otherwise a new site will be added to the atom list. This site will not be added to the punch file.

calcat	1	calculation type	<i>Code</i>	<i>N</i>	<i>M</i>	<i>Angles</i>
			linear	2	1	
			trigon	3	1	
			tetchn	3	2	

teterm	3	3	α_1
sqrpln	3	2	
octhed	3	4	
hexgnl	3	3	
genral	3	1	$\alpha_1\alpha_2$
methyn	4	1	
trterm	3	2	
median	4	1	

- 2 bond distance in Å
 3- labels of atoms used in the generation,
 followed by atom labels of generated
 atoms ^a

Notes:

- a. Atom labels must match identically those in the bdf. The number of fields required depends on the calculation. The first N fields will be used for the generators and the next M fields will be used for the generated atoms. The final angle fields will be used only for the specified generation type. The coordinates of the generating atoms must be such that the atoms are connected, not related by a symmetry operation other than the identity.

3.9. *BONDLA*: Bond lengths and angles

atrad	atomic radii by atom type
addpek	add peak site to atom list from pek file
bond	calculate a specific bond distance
angle	calculate a specific bond angle
dihed	calculate a specific dihedral angle
nviron	calculate distances to a specific atom site
site	enter temporary atom sites needed for geometry

BONDLA	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	bond distances	<u>bond</u>		list all bond distances
	bond angles	<u>nbon</u> <u>angl</u>		no bond distances list all bond angles
	contact distances	<u>nang</u> <u>ncon</u> <u>cont</u>		no angles no contact distances list all contact distances
	dihedral angles	<u>ndih</u> <u>dihe</u>		no dihedral angles list all dihedral angles
	bond check	<u>chek</u>		for Acta Cryst CIF bond checking
	sort bond & contact distances	<u>nsor</u> <u>sort</u>		do not sort distances sort distances
	optimally connect input sites	<u>clus</u> <u>nclu</u>		recluster input sites use input sites
	use full symmetry to searches	<u>symm</u> <u>nsym</u>		apply full symmetry no symmetry
	do not output geom data to bdf	<u>nbd</u>		
	use brackets in pch output file	<u>brac</u>		<i>output</i> <i>tobdf</i> <i>no</i> <i>brackets</i>
	number of cells searched	<u>cell</u>	n	1 for 1 unit cell 2 for 27 cells 3 for 125 cells

atrad	1	atom-type symbol to which the radii apply(max 8 chars)	
	2	maximum radius to be used in identifying contacts	1.25
	3	maximum radius to be used in identifying bonds	0.9
	4	minimum radius to be used in identifying bonds	0.015
addpek	1	atom-type symbol assigned to selected peak sites (max 8 chars)	
	2	number of peaks assigned to atom-type	1
	3	maximum radius to be used in identifying contacts	1.25
	4	maximum radius to be used in identifying bonds	0.9
	5	minimum radius to be used in identifying bonds	0.015
	6	thermal displacement parameter assigned to output site	0.035
bond	1-2	atom labels of sites 1 and 2 (max 24 chars)	
	3-4	numbers _a of symops applied to sites 1 and 2	1
	5-6	codes of cell translations applied to sites 1 and 2	555

Notes:

- a. Symmetry matrix number as entered into STARTX.

angle	1-3	atom labels of sites 1, 2 and 3 (max 24 chars)	
	4-6	numbers _a of symops applied to sites 1, 2 and 3	1
	7-9	codes of cell translations applied to sites 1, 2 and 3	555

Notes:

- a. Symmetry matrix number as entered into STARTX.

dihed	1-4	atom labels of sites 1, 2, 3 and 4 (max 24 chars)	
	5-8	numbers _a of symops applied to sites 1, 2, 3 and 4	1
	9-12	codes of cell translations applied to sites 1, 2, 3 and 4	555

Notes:

- a. Symmetry matrix number as entered into STARTX.

nvirion	1	atom label of central site (max 24 chars)	
	2	number _a of symop applied to sites	1
	3	code of cell translations applied to site	555
	4	radius of sphere to be searched for other sites	3

Notes:

- a. Symmetry matrix number as entered into STARTX.

site	1	atom label of site _a (max 24 chars)	
	2-4	fractional coordinates x, y, z	
	5	U	0.035
	6	population	1
	7-9	$\sigma_x, \sigma_y, \sigma_z$	0.0

Notes:

- a. If label matches site on the bdf, these coordinates will be substituted for the calculation. Otherwise a new site will be added to the atom list. This site will not be added to the bdf.

3.10. *BUNYIP*: Search for Additional Symmetry

radii	set radius for atom types
assume	assume atom types correct
search	limit search to symmetry type

complx define molecular cluster
remove define atom label or atom type not to be included in search
align atom labels to be forced to be on the symmetry element

BUNYIP	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	coordinates used in search	<u>clust</u>		cluster coordinates
		<u>input</u>		use input coordinates
	print the midpoint table	<u>midpt</u>		
	set search tolerance	<u>tolc</u>	f	0.5A
	allow n sites outside tolerance	<u>allow</u>	n	0
	co-linearity/centricity threshold	<u>close</u>	t	0.25A
	quick search	<u>fast</u>	n	n atom matches 10
radii	1	atom-type code		
	2	atomic radius		
	3-	repeat as above		
assume	1-	atom types assumed correct in symm search		all except H
search	<i>Option</i>	<i>Code</i>		
	limit search to			
		<u>centre</u>		inversion centre
		<u>axis</u>		rotor or screw
		<u>plane</u>		mirror or glide
complx	1-2	pairs of atom labels of sites connecting a molecular complex		
	3-	atom label pairs for additional clusters		
remove	1-	atom label, or atom-type code of atoms excluded from the search		
align	1-	labels of sites forced to exist on a symmetry element (e.g. site on the mirror plane)		

3.11. **CBAZA**: Add crystal based azimuthal PSI angle

diff_a diffractometer description

orient crystal orientation

Notes:

a. **diff** and **orient** are mandatory.

CBAZA	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	data set	<u>dset</u>	n	1
	store direction cosines	<u>dcos</u>		
	store diff angles	<u>angl</u>		bisecting mode
	list reflections	<u>print</u>	n	20
diff _a	1	sense of rotation of the 2θ circle	<u>c</u>	clockwise
			a	anti-clockwise
	2	sense of rotation of the χ circle	<u>c</u>	clockwise
			a	anti-clockwise
	3	sense of rotation of the ϕ circle	<u>c</u>	clockwise
			a	anti-clockwise

4	sense of rotation of the ω circle	<u>c</u>	clockwise	
		<u>a</u>	anti-clockwise	
5	sense of rotation of the ψ circle	<u>c</u>	clockwise	
		<u>a</u>	anti-clockwise	
6	value of ϕ at $\chi = 90$ degrees			0
7	minimum possible ω ($-180 < \omega < 180$)			-48
8	maximum possible ω ($-180 < \omega < 180$)			58

Notes:

- a. See the Diffractometer Description in [ABSORB](#) for the method to be used in determining the sense of rotation for each diffractometer circle.

orient	1-3	hkl indices of first orienting reflection
	4	angle χ of first orienting reflection ^a
	5	angle ϕ of first orienting reflection
	6-8	hkl indices of second orienting reflection
	9	angle χ of second orienting reflection
	10	angle ϕ of second orienting reflection

Notes:

- a. Angles in the bisecting mode. (For CAD see Ch. II of the CAD User's Manual.)

3.12. **CHARGE**: Atomic charges from difference density

CHARGE	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	weighting function	<u>density</u>		promolecular density
		<u>potential</u>		promolecular potential
	inclusion region	<u>border</u>	n	max edge distance
	atom range	<u>contact</u>	n	max atom distance

3.13. **CIFIO**: Input and output CIF data

block data_block code definition

CIFIO	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	dictionary checks	<u>dict</u>		enter <code>cifdic</code> and apply checks
	processing mode	<u>cifin</u>		convert a <code>.cif</code> file to a <code>bdf</code>
		<u>cifout</u>		convert a <code>bdf</code> to a <code>.cif</code> file
		<u>arcout</u>		convert a <code>bdf</code> to an <code>.arc</code> file
		<u>arcin</u>		convert an <code>.arc</code> file to a <code>bdf</code>
	CIF struct. factors	<u>sfac</u>		output requested sf data to <code>cif</code> file
		<u>nsfac</u>		do not output sf data to <code>cif</code> file
		<u>scale</u>		scale measured sf data for <code>cif</code> file
		<u>nscal</u>		do not scale sf data for <code>cif</code> file

	<u>corr</u>		correct measured sf for extinction	
	<u>ncorr</u>		no correction of extinction effects	
data set number	<u>dset</u>	n		1
Acta Cryst.	<u>bcheck</u>		bond checking option a	

Notes:

- a. Not recommended for general use.

block 1 data_ block code written by `cifin` or read by `cifout`

3.14. *CIFENT*: CIF entry of diffractometer data

CIFENT	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	dictionary checks	<u>dict</u>		enter <code>cifdic</code> and apply checks
	processing mode	<u>cifin</u> <u>cifout</u> <u>arcout</u> <u>arcin</u> <u>sfac</u>		convert a .cif file to a bdf convert a bdf to a .cif file convert a bdf to an .arc file convert an .arc file to a bdf
	CIF struct. factors	<u>nsfac</u> <u>scale</u> <u>nscal</u> <u>corr</u> <u>ncorr</u>		output requested sf data to cif file do not output sf data to cif file scale measured sf data for cif file do not scale sf data for cif file correct measured sf for extinction no correction of extinction effects
	data set number	<u>dset</u>	n	1

3.15. *CONTRS*: Contour a map

LEVELS contour levels specification
lintyp assignment of line types to contour levels
dash specification of dashed line dash-gap sequence
dot specification of dotted line dash-gap sequence
color specify contour colors

CONTRS	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	type of input to expect	<u>fourr</u> <u>slant</u>		input from <code>map</code> file input from <code>sla</code> file
	vector scale factor	<u>scal</u>	n	1.0
	first layer to contour	<u>flayer</u>	n	<i>origin</i> <i>from bdf</i>

last layer to contour	llayer	n		<i>max.</i>
increment between layers	dlayer	n		<i>from bdf</i>
				<i>incr. from</i>
				<i>bdf</i>
superimpose layers	super		yes	
interpolation type	cubic		cubic interpolation	
	linear		linear interpolation	
curve smoothing	smooth		smoothing on curves	
	nsmooth		no smoothing	
resolution for smoothing	resol	n		0.10
mesh points as crosses	mesh		yes	
plot border of map	border		plot border	
	nborder		do not plot border	
plot atoms near plane	atoms	n	distance n from plane	
plot atom names	nnames		do not plot names	
off plane atoms	dist		plot italic distances	
draw the atomic bonds	draw			
rotate map 90 degrees	rotate		rotate clockwise	
vector function plot ^a	mod		modulus of vector	
	vect		3-component vector	
contour filling	filled		switch on ^{b c}	

Notes:

a. Note that these function require input from a **sla** file.

b. Contour filling is currently very, very rudimentary. The The zero level is a baseline and contours are filled downwards to -ive and then upwards to +ive. The upshot is that for re-entrant contour levels the contour will (a) be filled one color in the wrong direction and (b) be overwritten by the nearest local maxima/minima of the reentrant curve portion. You have been warned!

c. Contour filling will not be apparent unless PLOTX or PREVUE are invoked with the "color fill" option.

LEVELS	<i>Option</i>	<i>Code</i>	<i>Arg</i>	
	type of contour ^a	arith		arithmetic intervals
		geom		geometric intervals
		direct		explicit levels input
	number of contours ^{a b}	nc	n	number of contours
	starting contour level ^b	start	n	first level
	level incr. ratio ^b	incr	n	increment (max 90)
	include levels of -ve sign	nnegate		do not include
		negate		include
	include zero level	nzincl		do not include
		zincl		include

Notes:

a. These items are mandatory.

b. Codes **nc**, **start**, and **incr** are not allowed for the **direct** option - follow the **direct** flag on the input line with the levels desired.

The **levels** line can function in a cumulative manner for the **direct** option. The **negate** and **zincl** operations are not performed until all line input is complete.

lintyp	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	dashed line ^a	dashed	n m	
	dotted lines	dotted	n m	
	solid line	solid	n m	

Notes:

a. The **n** is the first line and **m** the last line to be assigned to this type. Both **n** and **m** must be input. The lines are assigned sequentially.

Options can be input in any order and in any combination of line types.

dash ^a	1	number of elements in the dash-gap sequence
	2	first dash length
	3	first gap length
	4	second dash length
	5	second gap length

Notes:

a. A maximum of 10 elements is allowed. The lengths are relative to a y-dimension (down the page) of 10 units.

dot	1-	same format as the dash line but applies to dotted line types		
color	1	temperature spectrum color	temp	
		scale		
		hue, saturation=1.0,	hue	
		brightness=1.0		
		greyscale color scheme	grey	
	2	Lower color limit		0.0
	3	Upper color limit		1.0
	4	fixed color for Zero contour level		auto
		or		
	1	explicit RGB entries in following fields	RGB	
	2 -	one color per contour level in hexadecimal notation black=000000, red=FF0000, green=00FF00, blue=0000FF		

3.16. **CREDOC**: Check cell choice

CREDOC	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	maximum obliquity (in degrees)	amax	d	3.0
	maximum cell index	mult	n	2

3.17. **CRILSQ**: Structure factor least-squares refinement 2

maxhkl	reflection limits to be used in ls refinement
twin	twinning operation in reciprocal space
constr	linear constraints between parameters
noref	shut off refinement of specified parameters

CRILSQ	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	number of refine cycles	cy	n	1
	dispersion	nd		do not apply dispersion
	overall tdp	ov		apply and refine overall tdp
	extinction	ex		apply and refine extinction parameter
	Flack param	ab		apply and refine Flack absolute structure parameter
	occupancy factor	ro		apply and refine occupancy factor
	reflection skip	sk	n	use every n-th reflection
	weighting scheme	wt	n	n = 1 unit weights n = 2 weight from bdf n = 3 weight = 1/variance

	contribute to refl. ls	mr	q	only if $w* dY < q$	
	contribute to refl. ls	m1		exclude rcode 2 reflections (included by default)	
	shift damping factor	sd	q		1.0
	terminate refinement	tc		if convergence reached	
	list	li	n	n reflections after last cycle	99999
		lr	q	poor reflections if $w* dY > q$	
		lm		matrix and vector after last cycle	
		lc	q	correlation matrix for elements $> q$	
	punch	pc		parameter lines to .pch file	
	save	sm		correlation matrix to .cmx file	
maxhkl	1-3	max h , k , l in refinement process		from bdf (see ADDREF)	
	4	min value of $\sin(\theta)$ applied		from bdf (see ADDREF)	
	5	max value of $\sin(\theta)$ applied		from bdf (see ADDREF)	
twin	1-9	matrix of symmetry operation in reciprocal space coded as: m11 m12 m13 m21 m22 m23 m31 m32 m33			
	10	number of twin fractions			
	11	volume ratio of twin fraction 1			
	12-	volume ratio of twin fraction 2, 3 ... a			

Notes:

- a. A maximum of 8 fractions are allowed, they should be in sequence and form a closed group (see [CRYLSQ](#))

constr	constraint equation (see CRYLSQ) Embedded blanks are allowed.			
noref	invariant parameter string (see CRYLSQ) No embedded blanks are allowed. The general parameters SKF, UOV, EXT, ABS RID and TWI can be shut off as well as the atom parameters X,Y,Z,U,U11,U22,U33,U12,U13,U23,UIJ and POP.			

3.18. [CRISP](#): Well baked structural solution

invar	control structure invariant relationships used
select	control starting phase selection process
refine	parameters controlling tangent refinement process
setfom	input figure-of-merit parameters
only	request specific phase sets from an extended run

CRISP	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	data set number	dset	n	1
	optimization level	optim	n	0-3 depth of study
	output listing	list	n	0-5 depth of study
	total phase sets	trial	n	random phase sets
	best phase sets	best	n	look in depth at
	minimum E value	emin	n	from .inv file
	max generators	rmax	n	from .inv file

	start from bdf phases	p bdf	progressively randomize with
	trial 1000 presets	xxxx	each set
	go for broke	zzzz	check best 10
			check best every 500
invar	1	invariant types to be used in the phasing process use triplet and quartet invariants all	
		use triplets only trip	
		use quartets only quar	
	2	minimum A of triplets used	<i>from file</i>
			<i>.inv</i>
	3	minimum B of quartets used	<i>from file</i>
			<i>.inv</i>
	4	maximum triplets allowed per generator	100
	5	maximum quartets allowed per generator	100
	6	reduce A and B values by this fraction of σ A and B	1
	7	control eligible quartets according to X-vector sum include quartets parxv with XVsums <LO or >HI include all quartets allxv include quartets with X-vectors in the data sphere insxv include only quartets with X-vectors out of sphere outxv use negative quartets actively active	
	8	lower limit for cross-vector sums - LO	<i>25% of</i>
	9	upper limit for cross-vector sums - HI	<i>max to 2.5</i> <i>75% of</i>
	10	invariant ψ applied to negative quartets (in degrees)	<i>max to 4.0</i> 180
			if no fragment ψ estimate
			otherwise
			0
select	9	starting weight for phases assigned in random mode	0.25
	10	integer used to seed random phase generator 1	137
	11	integer used to seed random phase generator 2	13804
refine	3	maximum number of refinement iterations	30
setfom	1	control flag for pretest of FOM values pretest FOM values after the 6 th pret tangent iteration do not pretest FOM values nopr	
	2	optimum RFOM value for CFOM/AMOS calculation	1.0
	3	optimum RFAC value for CFOM/AMOS calculation	0.25
	4	optimum PSIO value for CFOM/AMOS calculation	1.5

5	optimum NEGQ value for CFOM/AMOS calculation	60
6	weight of RFOM value for CFOM/AMOS calculation	1
7	weight of RFAC value for CFOM/AMOS calculation	1
8	weight of PSIO value for CFOM/AMOS calculation	2
9	weight of NEGQ value for CFOM/AMOS calculation	2
10	automatic termination of multi-solution process continue even if FOMs < <code>cont</code> optimal FOMs	yes
11	RFOM lower rejection limit for 014 and 024	0.25

only_a 1 - 35 request specific phase set numbers to look at in detail

Notes:

- a. Generally used after terminating a `CRISP ZZZZ list 3` run, having spotted some interesting phase sets that may be worth further in depth study.

3.19. *CRYLSQ*: Structure factor least-squares refinement

disper	dispersion factor
scale	scale factor
extinc	extinction parameters
maxhkl	reflection limits
xabs	absolute-structure parameter
select	select subset of atoms from the bdf
dptype	change atomic displacement parameter type
group	group or identical molecule refinement
atomgr _a	atoms per group
restr _b	establish conditions for positional parameters
constr _b	constrain one parameter to be a function of otherparameters
noref _b	shut off refinement of specified parameters
block	matrix blocking line

Notes:

- a. Must be preceded by a **group** line
- b. Order specific lines. **restr** lines precede **constr** which precede **noref** lines.

CRYLSQ	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	number of refine cycles	cy	n	1
	refinement coefficient	fr	F	
		f2	F₂	
		in	intensity	
		fh	F(heavy atom)	
	derivative matrices	bd	block-diagonal	
		fm	full-matrix	
		bl	specified in block lines	
	reflection weight	ws	1/σ ₂	
		wu	unit weights	
		ww	n bdf wt with id=190n	0

scale factor(s)	<u>us</u>		refine together	
	<u>ss</u>		refine separately	
dispersion	<u>ad</u>		apply but do not refine	
	<u>rd</u>		apply and refine	
extinction	<u>ax</u>		apply but do not refine	
	<u>rx</u>		apply and refine	
abs-struct. param	<u>aa</u>		apply but do not refine	
	<u>ra</u>		apply and refine	
thermal displacement type	<u>mx</u>		mixed iso and aniso (as in bdf)	
	<u>is</u>		isotropic	
	<u>an</u>		anisotropic	
	<u>ov</u>		overall	
population	<u>pp</u>		yes	
dataset number	<u>ds</u>	n	data set number	<i>bdf or 1</i>
reflection skip	<u>sk</u>	n	use every n^{th} reflection	
fixed atoms used	<u>ap</u>		apply values from prior run	
	<u>ep</u>		apply & store on output bdf	
type hkl as rcode2 ^a	<u>tl</u>	q	only if $Y < q \sigma Y$	
use in matrix	<u>mr</u>	q	only if $w \Delta Y < q$	
rcode2 use in matrix	<u>ml</u>		only if $Y_{\text{cal}} > Y_{\text{obs}}$	
shift damping factor	<u>fu</u>	q	for <u>bd</u>	0.8
			for <u>fm</u>	1.0
non positive-def. U	<u>tr</u>		make pos. def. and continue	
	<u>tw</u>		tell user and continue	
	<u>tt</u>		terminate calculation	
termination criterion	<u>tc</u>		stop if no convergence of R	
list reflections	<u>l1</u>		last cycle	<i>no</i>
	<u>l2</u>		every cycle	
list poor reflections	<u>lr</u>	q	if $w\Delta Y > q$	
atom lines to pch file	<u>p1</u>		last cycle	<i>no</i>
	<u>p2</u>		every cycle	<i>no</i>
list correl. matrix	<u>m1</u>	q	if $> q$ after the last cycle	
list deriv. matrices	<u>m3</u>		after last cycle only	
	<u>m4</u>		after every cycle	
save correl. matrix	<u>ms</u>		output to file <u>cmx</u>	
old atomic parameters	<u>ao</u>		output old params to the bdf	

Notes:

a. Applies for the current run only and does not affect the bdf.

disper 1 atom-type name
 2 real part of dispersion scattering factor
 3 imaginary part of dispersion scattering factor
 4-6 as in 1-3, or use separate **disper** lines

scale 1 F_{rel} scale factor
 2 scale group number 1
 3, 4 as in 1, 2, or use separate **scale** lines

extinc 1 extinction correction method
 Zachariasen zach
 Becker-Coppens, type I typ1
 Becker-Coppens, type II typ2
 Becker-Coppens, type I and II gen
 2 extinction type
 isotropic is

		anisotropic (not implemented)	an		
	3	mosaic distribution			
		Gaussian distribution	<u>gaus</u>		
		Lorentzian distribution	lore		
	4	isotropic extinction		(g)	0
		parameter(zach or typ1)			
	5	isotropic extinction		(ρ)	0
		parameter(typ2 or gen)			
	6	mean value of T in mm		<i>Xray</i>	0.3
				<i>neutron</i>	1.5
maxhkl	1-3	max h , k , l applied in refinement process			999
	4	min value of $\sin\theta/\lambda$ applied			0
	5	max value of $\sin\theta/\lambda$ applied			10
xabs	1	Flack absolute structure parameter			0.5
select	1-	atom label <i>or</i> atom label string ^a			
		Notes:			
		a. An atom label string is two atom labels separated by a slash '/'. The first label identifies the first atom, and the second label the last atom in a sequence of atoms.			
dptype	1	atom type symbol			
	2	thermal displacement type			
		overall U	ov		
		isotropic U	is		
		anisotropic U _{ij}	an		
	3-4	as in 1, 2, or use separate dptype lines			
group	1	group refinement number of molecule			0
	2-4	x, y, z fractional coordinate of centre of gravity, if fixed			
	5	Euler angle refinement signal		<i>not fixed</i>	
		Euler angles invariant	<u>0</u>		
		symmetry element is parallel or			
		perp. to a-axis	1		
		symmetry element is parallel or			
		perp. to b-axis	2		
		symmetry element is parallel or			
		perp. to c-axis	3		
	6	signal for the refinement of isotropic U			
		group isotropic U	<u>+1</u>		
		invariant U	0		
		individual isotropic U	-1		
	7	signal for population parameter refinement		<i>no constraints</i>	
		group population	+1		
		invariant population	<u>0</u>		
		individual population	-1		
atomgr	1	atom label string ^a			

Notes:

- a. An atom label string is two atom labels separated by a slash '/'. The first label identifies the first atom, and the second label the last atom in a sequence of atoms.

```

restr a    1      restraint class
              bond length           bond      (n=2 atoms)
              bond angle            angle     (n=3 atoms)
              dihedral angle        dihed    (n=4 atoms)
          2      atom labels of the n restrained
              atoms b
          n+2    ideal values of bond length,
              angle or dihedral angle
          n+3    standard deviation of ideal
              value

```

Notes:

- a. **restr** is an order specific line and cannot be preceded by **constr** or **noref** lines.
- b. These atom labels may each be followed by an optional field indicating a symmetry operation to be applied to the coordinates of the atom. This field has the form (n_mmm), where n is the symop number from *STARTX*, and mmm is the translation based on the number 555. The parentheses are mandatory.

```

constr a    1      constraints equation b (see CRYLSQ)

```

Notes:

- a. **constr** is an order specific line and cannot be preceded by **noref** lines.
- b. Embedded blanks are allowed. The order of the constraints is not arbitrary - reference atoms must occur higher in the atom list than subject atoms. Furthermore, for parameters belonging to the same atom, the reference parameters must occur earlier in the parameter list than the subject parameters. The parameters may be any of the following: SKF, UOV, EXT, DSP, X, Y, Z, U, U11, U22, U33, U12, U13, U23, POP, APP, and NEU.

```

noref      1      invariant parameter string a

```

Notes:

- a. No embedded blanks are allowed. See the **constr** line for the list of parameters and for special atom names.

```

block a    1      matrix block (same format as noref)

```

Notes:

- a. Any **block** lines must cover all selected parameters. When no parameter symbol is given, a new block is started.

3.20. *DIFDAT*: Process diffractometer data

```

transf a    transform Miller indices of input reflections
attenu a    attenuator factors
stands ab   specification of standard reflection signals
sthkls ab   alternative method of establishing standard reflections
fetch      definition of the order of the quantities in data lines
data       diffractometer data lines
genscl c d  generation of intensity scale factors
setscl cd  set intensity scale factors
discon c   definition of discontinuities in scale factors

```

Notes:

- a. Must precede following option lines.
- b. One or the other but not both may be used.
- c. One of these lines must be present if the data is to be scaled. Otherwise, scale factors are calculated but not applied.
- d. If a **setscl** line is input before a **genscl** line, the **genscl** line will be ignored.

DIFDAT Option	Code	Arg	Def
----------------------	------	-----	-----

input data mode	<u>dat</u>		data lines from input	
	bdf		from archive file	
	cad		CAD4 diffr. file cad	
	niv		Nicolet (Vax) ^a file nic	
	nid		Nicolet (DG) ^a file nic	
	sie		Siemens ascii file sie	
	uwa		P3 special binary file dct	
	tsu		Tsukuba diffr. file tsu	
	spe		special diffr. file dif	
	rd5		Rigaku Denki. file rd5	
	shx		SHELX F4 fixed format file	
			.hkl ^b	
	sh3		SHELX F3 fixed format file	
			.hkl ^b	
	rap ^b		Rigaku Rapid .f2plus	
	bru ^b		Bruker Smart file ^c raw	
filename <256 characters	file	n	change shx / rap input data filename	
Bruker raw file	cos		load direction cosines on to bdf	
Tsukuba .tsu file data ^d	smu	n	monitor count smoothing	1
			across 1<n<21 refl'ns	
Nicolet scan mode	2th		Nicolet θ - 2θ scan	
	ome		Nicolet ω scan	
add bdf refln data	<u>noadd</u>		do not add hkl from old bdf	
	add		add new hkl to old bdf ^e	
print reflection data	<u>noprint</u>		print no reflections	
	print	n	print n	<i>all</i>
print raw data	raw		print n lines, defined above	
scale intensities	nap		do not apply scale	
fate of standards	<u>include</u>		include on output bdf	
	<u>exclude</u>		exclude from output bdf	
backg. time each side	bgtime	p	time in seconds	10
min. scan time ^f	mint	t	time in seconds	10
observ. threshold ^g	obst	n	r _{code} =2 if I < n σ I	2
set intensity if rcode 2	<u>nobth</u>		store as calculated	
	obth	m	store as m σ I _h	
backg. imbalanced	baln	q	warn if (b1-b2) > q σ I	10
damp stds variation	damp	r	scaledamping factor	.5
instability option	inst	o m b	option, slope, intercept ⁱ	
instability calculation	nsc		calc. before scaling	<i>after</i>
instability calculation	nrw		do not use raw counts	
σ I options	est	f	use f*I as σ I	
	sig	s	incr. σ I with sI	.02
output diffract. angles	ang		put angles on the bdf	<i>no</i>
output Eulerian angles	eul		put angles on the bdf	<i>no</i>
deadtime correction	dtm	n	add n % to deadtime correction	

Notes:

- a. Outputs input data as **data** lines to the punch file
- b. data is stored as f_{2rel} and not $irel$
- c. The Bruker SMART .raw file contains primary and diffracted beam direction cosines, to represent the orientation of the crystal at reflection measurement time (generally anything other than the bisecting position). DIFDAT contains code to convert those direction cosines to an orientation specified in terms of the Crystal Based Azimuthal Psi Angle (l_{refl} : IDNUM n011), which should permit absorption correction using programs such as LSABS and LSLS (But NOT ABSORB). This requires the orientation matrix UB to be stored in STARTX. The azimuthal angle calculation has not been verified as correct!
- d. adjustments for bugged software. Generally not required.
- e. The **add** option copies unaltered reflections from the input logical record *lrefl*: and adds newly processed reflections to them. The input file must contain at least the quantities h, k, l, I, $\sigma(I)$, and rcode. All other quantities are discarded.
- f. If an encountered scan time is less than ϵ following the **mint** option, the reflection is rejected.
- g. Reflections with intensities greater than $n\sigma$ above background as defined by the **obst n** option will be classed as observed, rcode=1. All other reflections will be coded as unobserved with an rcode=2.
- h. The **obth** option allows the user to replace the value of an intensity with an rcode=2 by the quantity $n*\sigma(I)$, where n is the value specified in the **obst n** option. The default allows intensities to be stored just as they are calculated. This includes intensities calculated to have negative values.
- i. See the **DIFDAT** section in the Reference Manual for a description.

attenu a 1- attenuator factors (minimum of 1.0)

Notes:

- a. The first attenuator factor does not refer to the open attenuator position. The attenuator factors are used as multiplicative factors.

stands 1 character string that identifies standard or observed reflection

^a
2 integer assigned to reflection identified by string in field 1
for a normal reflection 1
for a standard 2
3, 4 same as 1 and 2 above

Notes:

- a. The observed reflection/standard reflection identifiers may be either a character string or a single decimal digit. (e.g., #ST or 1). A maximum of 11 pairs (1 observed and up to 10 standards) may appear on the **stands** line. This line may be used as an alternative to the **sthkls** lines. Note that this line is used only in conjunction with **data** lines.

sthkls a 1-3 hkl indices of first standard reflection
4-6 hkl indices of second standard reflection , and so on

Notes:

- a. A maximum of 10 standards may be defined. More than one **sthkls** line may be used. The first blank h field terminates the scan. These lines may be used as an alternative to a **stands** line. Note that this line is used only in conjunction with **data** lines.

fetch a 1- mnemonics defining the input fields on **data** lines

field to be skipped ^b	skp
signal reflections as standard or regular observation ^c	sig
h index	h
k index	k
l index	l
scan time of θ - 2θ scan ^d	sct
scan counts	scc
counts for background 1 ^e	bg1
counts for background 2	bg2
attenuator number ^f	atn
reflection sequence number ^g	rfn
I_{rel}	irl
σI_{rel}	sir
F_{2rel}	fsq
σF_{2rel}	sfs
F_{rel}	fo
σF_{rel}	sfo

2θ	tth
ω	omg
χ	chi
ϕ	phi

Notes:

- The values for **h**, **k** and **l**, and at least one and only one of the group **sec**, **ir1**, **fsq** and **fo** are mandatory.
- The **skp** option may be used more than once.
- If **sig** is not specified, the reflections to be treated as standards must be supplied on **sthkls** lines.
- If **set**, the scan time, is not present it is presumed equal to the variable **bgtime** which is supplied on the **DIFDAT** line.
- If **bg1** and **bg2** are not specified, they default to zero.
- If **atn** is not specified, the default is to assume no attenuator.
- If **rftn** is not available, the current reflection count will be used during processing.

data _a 1- reflection input quantities

Notes:

- These quantities are specified in the **fetch** line as described previously. In practice the use of a **setid** line or a **field** line followed by a **setid** line may be found useful. The intention is that the diffractometer output tape will be processed through local operating system utilities to form a coded file. This file will be edited using a text editor and then fed to the Xtal system in the input stream. (See examples in the Reference Manual.)

genscl _a 1 number of standard groups scale factors are smoothed 5

Notes:

- This line will be ignored if preceded by a **setscl** line. It causes the generation of scale factors based on observations of standard reflections. If **setscl** lines follow, they will replace specified group scale factors.

setscl 1 scale factor to supersede value for specified standard group
2 serial number _a of the last standard reflection of the group of standards for which this scale supersedes

Notes:

- The serial number refers to the diffractometer serial number, if that has been supplied, otherwise it refers to the sequence number developed as the reflections are being treated on input. Smoothing will take place over the scale factors unless prevented by **discon** input lines. The serial number given must be that of the LAST standard reflection of the group of standards used to establish the scale. Otherwise, the **setscl** line will be ignored.

discon _a 1- serial number of the last standard reflection of the group of standards used to establish the scale factor

Notes:

- A preliminary run of DIFDAT may have to be made to find discontinuities in the scale factors.

transf _a 1-9 transformation matrix a_{11} , a_{12} , a_{13} , a_{21} , a_{22} , a_{23} , a_{31} , a_{32} , a_{33}

Notes:

- The transformation is: $h'k'l' = hkl * \mathbf{a}$ -matrix

3.21. **DYNAMO**: Molecular Dynamics structure factor calculation **NEW**

scale	F_{rel} scale factor
transf	transformation matrix from archive unit cell to molecular dynamics cell
typinf	molecular dynamics data file atom type information
format	molecular dynamics data file format
maxhkl	maximum and minimum h , k , l and $\sin\theta/\lambda$
select _a	select a subset of atoms in the bdf for F_{cal}
omit _a	omit a subset of atoms in the bdf for F_{cal}

Notes:

- Either **select** lines may be used, or **omit** lines, but not both.

DYNAMO	Option	Code	Arg	Def
	store Fmd as	<u>frel</u>		
	molecular dynamics	<u>fcal</u>		
	temperature factor	<u>file</u>	name	absolute or relative path to file
				^a
		<u>mix</u>		mixed iso and aniso
		<u>iso</u>		isotropic U only
		<u>ani</u>		anisotropic Uij only
		<u>ove</u>		overall isotropic, all atoms
	anom. dispersion	<u>dis</u>		apply dispersion terms
		<u>nod</u>		do not apply dispersion
	extinction correction	<u>noex</u>		do not apply extinction
		<u>ext</u>		apply extinction correction
	sin/cos function	<u>nos</u>		calculate from library routine
		<u>sct</u>		use prestored table look-up
	dataset number	<u>dset</u>	n	use dataset n on bdf
	reflection control	<u>nlis</u>		produce no hkl listing
		<u>lis</u>	s	list hkl with $F_{obs}-F_{cal} > s$
		<u>nrej</u>		treat all reflections
		<u>rej</u>	t	reject hkl if rcode $\geq t$

Notes:

- a. **file** should be the last option on the command line

transf 1-9 transformation matrix T₁₁, T₁₂, T₁₃, T₂₁, T₂₂, T₂₃, T₃₁, T₃₂, T₃₃ ^a

Notes:

- a. Transformation is $\alpha' = T\alpha$, α is a column vector of the unit cell vectors **a**, **b**, **c**.

typinf_a 1- sequential atom type character code
 2 number of consecutive sites of atom type in MD file
 3-4... as for 1-2, up to 15 types per line, to a max of 40 sequences

Notes:

- a. Mandatory sub command line for the DYNAMO calculation.

format 1 mol. dynam. file format (3F8.5,24x,I5)

scale 1 F_{rel} scale factor
 2 scale group 1

maxhkl 1-3 max |h|, |k|, |l|
 4 min $\sin\theta/\lambda$ 0
 5 max $\sin\theta/\lambda$ 2

select_a 1- atom labels of atoms to be selected

Notes:

- a. Any number of atom names may be supplied; those given will be included in the DYNAMO calculation. If no **select** or **omit** lines are used, all atoms in the bdf will be used in the calculation.

omit 1- atom labels of atoms to be omitted

3.22. **EXPAND**: Expand archive atom sites **NEW**

EXPAND	Option	Code	Arg	Def
	expansion extent	<u>cell</u>	n	fill n cells
				1

contract to independents	not			
site overlap tolerance	tolc	d	separation d in Angstroms	0.1
move atoms within tolerance to	stolc	d	distance d in Angstroms	0.4
special sites				
atom name strip	nost		don't strip symm from name	

3.23. *FC*: Calculated structure factors

scale	F_{rel} scale factor
maxhkl	maximum and minimum h, k, l and $\sin\theta/\lambda$
select ^a	select a subset of atoms in the bdf for F_{cal}
omit ^a	omit a subset of atoms in the bdf for F_{cal}
archiv	direct addition or deletion of items in logical record <i>lrrefl</i> :

Notes:

- a. Either **select** lines may be used, or **omit** lines, but not both.

FC ^a	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	temperature factor	<u>mix</u> iso ani ove		mixed iso and aniso isotropic U only anisotropic Uij only overall isotropic, all atoms
	anom. dispersion	<u>dis</u> nod		apply dispersion terms do not apply dispersion
	extinction correction	noex <u>ext</u>		do not apply extinction apply extinction correction
	sin/cos function	<u>nos</u> sct		calculate from library routine use prestored table look-up
	dataset number	dset	n	use dataset n on bdf
	reflection control	<u>nlis</u> lis <u>nrej</u> rej	s t	produce no hkl listing list hkl with $F_{\text{obs}} - F_{\text{cal}} > s$ treat all reflections reject hkl if $\text{rcode} \geq t$
	output Sim weight	sim		weight stored as 1910 on

Notes:

- a. : Coordinates will be listed if **reset psta 4** line is entered before FC line.

scale	1	F_{rel} scale factor	
	2	scale group	1
maxhkl	1-3	max h , k , l	
	4	min $\sin\theta/\lambda$	0
	5	max $\sin\theta/\lambda$	2

select 1- atom labels of atoms to be selected ^a

Notes:

- a. Any number of atom names may be supplied; those given will be included in the FC calculation. If no **select** or **omit** lines are used, all atoms in the bdf will be used in the calculation.

omit 1- atom labels of atoms to be omitted

archiv 1- ID numbers of reflection data to be added or deleted on the bdf ^a

Notes:

a. By default, F_{cal} and the estimated phase are added to *lrrefl*, thus updating any previous values of F_{cal} and phase that may have been on the file. Also by default, all information from previous calculations related to F_{cal} for the dataset under consideration is deleted. Specifically, the following items are deleted if they are on the input data file; A(total), B(total), A(dispersion), B(dispersion), and A and B excluding any extinction correction. This option is used only in special cases.

Positive ID numbers signal additions to the file and negative ID numbers signal deletions. The allowed values for the ID numbers and their meanings may be found in the *lrrefl*: section of the Reference Manual.

3.24. *FOURR*: Fourier maps

scale	F_{rel} scale factors
maxhkl	max and min h, k, l and $\sin\theta/\lambda$
fzero	non-default F_{000} coefficient
print	print control parameters
map	Fourier map output
grid	axial divisions of Fourier map
layout	presentation of Fourier map
coeff	special summation coefficients

FOURR	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	<i>Map type</i>		<i>coeff.</i>	<i>phase</i> <i>symm</i>
	ΔF	<u>fdif</u>	ΔF	A&B, ϕ full
	Patterson	<u>patt</u>	F_2	none Laue
	E-Patt.	<u>epat</u>	$E * F$	none Laue
	Vector	<u>vect</u>	$E_2 - 1$	none Laue
	Emap	<u>emap</u>	$E \ x$	$\phi \ n$ full
	F_{obs}	<u>fobs</u>	kF_{rel}	A&B, ϕ full
	F_{cal}	<u>fcad</u>	F_{cal}	A&B, ϕ full
	Sim F_{obs}	<u>wfob</u>	wkF_{rel}	A&B, ϕ full
	Sim ΔF	<u>wdif</u>	$w\Delta F$	A&B, ϕ full
	elect. potential	<u>pdif</u>	$\Delta F/ S _2$	A&B, ϕ full
	elect. field	<u>edif</u>	$\Delta F S_i/ S _2$	A&B, ϕ full
	elect. gradient	<u>gdif</u>	$\Delta F(S_i S_j/ S _2 - 1/3)$	A&B, ϕ full
	Fourier symmetry	<u>gsym</u>	general space group symmetry	
		<u>laue</u>	^a Laue symmetry	
	map volume	<u>asym</u>	asymmetric unit	
		<u>full</u>	full unit cell (default for E-Patt.)	
	rejection criteria	<u>r1</u>	n reject $r_{\text{code1}} \ hkl \ \text{if} \ nF_{\text{obs}} > F_{\text{cal}}$	
		<u>r2</u>	n reject $r_{\text{code2}} \ hkl \ \text{if} \ nF_{\text{obs}} > F_{\text{cal}}$	
		<u>rf</u>	n reject if $F_{\text{obs}} > n$	
		<u>rs</u>	n reject if $n\sigma F_{\text{obs}} > F_{\text{obs}}$	
	summation method	<u>blsum</u>	Beevers-Lipson ^c	
		<u>ffsum</u>	Fast-Fourier with BL first sum	
	output map file ^d	<u>file</u>	output map to <u>map</u> file	
		<u>nofile</u>	suppress output on <u>map</u>	
	print map	<u>nprint</u>	do not print <u>map</u>	
		<u>print</u>	print <u>map</u>	
	E definition	<u>e1</u>	E-value 1 from bdf	
		<u>e2</u>	E-value 2 from bdf	
	E map phase set	<u>pset</u>	n use $\phi \ n$ from bdf	1

data set number	dset	n	use hkl dataset _n	1
list atom coords	ne		print atoms in theasym. unit	
	pe		print atoms in map limits	
	nw		print no atom coordinates	

Notes:

- The default is determined by the map. If a **coeff** line is entered this symmetry must be defined explicitly.
- If **n** is omitted, all **rcode2** reflections are rejected.
- Must be used for **pdif**, **edif** and **gdif** maps.
- No archive file is generated by FOURR unless a difference map is generated.

scale	1	F_{rel} scale factor		<i>from bdf or 1</i>
	2	scale group number		1
maxhkl	1-3	max h , k , l for accepted coefficient		
	4	min $\sin\theta/\lambda$ for accepted coefficient		0
	5	max $\sin\theta/\lambda$ for accepted coefficient		2
fzero ^a	1	forced value of (0, 0, 0) coefficient in Fourier		0

Notes:

- Do not enter an **fzero** line if automatic assignment is required.

print	1	map print-out format code		
		base-10 integers	normal	
		base-36 (0-9A-Z) code	packed	
		shade character in intervals defined in field 5	shaded	
	2	density scale for map print-out ^a	PATT, EPAT, VECT	1
			EMAP, FOBS, FCAL	10
			FDIF	100
	3	upper-bound for print suppression (HI)		10
	4	lower-bound for print suppression (LO)		-9999,
	5	density interval for shaded format output	for FDIF	-10 10
	6	calculate density scale so first-point density is this value		<i>no</i>
7	explicit page layout - print columns per map point ^b		4	
8	explicit page layout - print lines per map point ^b		2	
9	printer lines per inch		<i>default</i>	
10	printer columns per inch		<i>default</i>	
11	print character for rejected density ^c $LO < d < HI$		(.)	

Notes:

- Scaling of printed densities does not affect the values output to the **map** file.
- If a non-default choice for either field 7 or 8 is desired, both fields must be specified. For field 7 the equation is $(lines/point) * (col/inch) / (lines/inch)$. For field 8 the equation is $(col/point) * (lines/inch) / (col/inch)$. The **packed** and **shaded** printout is one-col/one-line per map point. **packed** output rejects negative densities.
- Enter the character bounded by = (e.g. **+=** for + character). **HI** and **LO** values of 1 and -1 cause all points to be printed. A map will be printed if the **print** line is present, even if the **nprint** option from the **FOURR** line is in effect.

map ^a	1	layer-to-layer axis of map		<i>longest axis</i>
		layer down a, b, or c axis	a b c	

		projection down a, b, or c axis	pa pb pc	
2-4		lower bound of map in fractions of a, b, c		<i>automatic</i>
5-7		upper bound of map in fractions of a, b, c		<i>automatic</i>
8		automatic map definition (points per Angstrom)		4

Notes:

- a. The **map** line should not be used if the **layout** line is entered, and vice versa.

grid	1-3	map point divisions along axes a, b, c		<i>automatic</i>
layout	1-3	map orientation codes for the axes a, b, c axis as page-to-page direction a layer axis as down page direction down axis as across page direction across specifies axis as for projection Fourier project		<i>automatic</i>
	4-6	number of map grid points along axes a, b, c		grid or <i>auto</i> ^b
	7-9	first grid point of map along axes a, b, c		<i>auto</i>
	10-12	grid increment of map along axes a, b, c		1

Notes:

- a. Either **layer** or **project** may be used but not both.

b. Layout parameters define explicitly the calculated map in terms of grid points. The map output to the **map** file will have these dimensions. The actual number of points output in each direction is defined by the width of the map (fields 4-6) divided by the grid increment (fields 10-12). For printing purposes, however, the grid across the page may be adjusted in order that the requested span will fit the output device.

coeff ^a	1	logical record <i>lrrefl</i> : ID no. for factor1, or constant scale1
	2	logical record <i>lrrefl</i> : ID no. for factor2, or constant scale2
	3	logical record <i>lrrefl</i> : ID no. for group number of scale3
	4	logical record <i>lrrefl</i> : ID no. for factor4, or constant scale4
	5	logical record <i>lrrefl</i> : ID no. for factor5, or constant scale5
	6	logical record <i>lrrefl</i> : ID no. for group number of scale6
	7	logical record <i>lrrefl</i> : ID no. for real component of structure factor
	8	logical record <i>lrrefl</i> : ID no. for imaginary comp. of structure factor
	9	logical record <i>lrrefl</i> : ID no. for structure factor phase in cycles

Notes:

- a. ALL values must be entered. Constants are entered as negative values. Enter -1 for unused fields 1, 2, 3, 4, 5, or 6. Enter 0 for unused fields 7, 8, or 9. ID numbers must include the dataset bias (n*1000).

3.25. *GENEV*: Normalized structure factors (E values)

frag ^a	fragment definition line
grid ^b	specify cell grid intervals

site _{a c} atom coordinates as cell fractions
siteg _{ac} atom coordinates as cell grid
sitea _{ac} atom coordinates as orthogonal Angstrom units
indexk hkl index rescaling parameters

Notes:

- Optional, but a **frag** line must precede each group of **site**, **siteg**, **sitea** lines.
- Optional, but a **grid** line must precede the first **siteg** line if entered.
- Only one method of supplying atom coordinates may be used.

GENEV	<i>Option</i>	<i>Code</i>	<i>Arg</i>		<i>Def</i>
	max $\sin\theta/\lambda$ used	smax	s	$\sin\theta/\lambda < s$	<i>bdf</i> value or 2.
	output $ E $'s to bdf	eout		yes	
		enot		no (only scale & U output)	
	bdf Frel scale	bscl		extract & apply as pre-scale	
		nscl		do not apply bdf scales	
	list estimated $ E $'s	list	e	yes if $ E > e$	
		nlist		no	
	fill missing weak data	fill		yes	
	Bayesian to weak data	baye		apply	
	use profile not linear	prof		calculate E2 with Debye profile	
	fix overall U	fixu	u	fix U - do not calculate	
	fix linear scale	fixk	k	fix Frel scale - do not calculate	
	define inflexion points	din1	p	set $\sin^2\theta/\lambda^2$ for inflex 1	0.15
		din2	q	set $\sin^2\theta/\lambda^2$ for inflex 2	0.26
	bdf expectation values	bexp		extract from bdf	
		nexp		estimate new values	
	data set number	dset	n		1
frag	1	fragment type indicator			
		type 1 random atom fragment	wilson		
		type 2 random group fragment	random		
		type 3 oriented group fragment	orient		
		type 4 positioned group fragment	positn		
	2	source of atomic coordinates	bdf	from site lines extract from binary data file	
	3	fragment multiplicity in asymmetric unit			1
	4-6	x, y, z fractional coordinates of type4 fragment origin			0

Notes:

- If the **random** option is used, only **sitea** lines are allowed.

grid 1-3 a, b, c axis grid divisions

site 1 atom label
 2-4 x, y, z fractional coordinates
 5 isotropic U (not used)
 6 population parameter 1

siteg *The **siteg** line is the same format as the **site** line except for grid coordinates.*

			or calculated assuming E=1.8 (centro); E=2.2(noncentro)	
	3	maximum unique quartets to be generated		<i>auto</i>
	4	maximum quartets per generator		500
	5	permitted cross-vectors for quartets all cross-vectors must be inside <u>insxv</u> data sphere all cross-vectors are permitted allxv one or more cross-vectors outxv outside data sphere ignore cross-vectors and notxv include all quartets		
	6	minimum single cross-vector E accepted		0
	7	maximum single cross-vector E accepted		100
	8	low cross-vector sum limit (max for negative quartets)		2.5
	9	high cross-vector sum limit (min for positive quartets)		4
gener	1	minimum E eligible as generator		<i>auto</i>
	2	number of generator reflections (max 10,000)		<i>auto</i>
	3	minimum $\sin\theta/\lambda$ eligible as generator		0
	4	maximum $\sin\theta/\lambda$ eligible as generator		2
	5	minimum E -to- σ E ratio eligible as generator		1
	6	maximum number of $\psi(0)$ reflections allowed		<i>auto</i>
psical	1	select reflections used in structure invariant generation select generators from hkl with <u>refine</u> input ϕ values allow all reflections to be extend generators		
	2	bdf ID number of ϕ used to estimate ψ of invariant		n750
	3	bdf ID number of F_{cal} used to estimate A and B		n751
print ^a	1	print invariant summary for N generators		<i>for all</i>
	2	first generator number of detailed listing		0
	3	last generator number of detailed listing		0
	4	maximum generator number listed in invariant blocks		100
	5	minimum A or B listed in invariant blocks		0
Notes:				
a. A detailed listing of invariants for each generator in the range defined by fields 2 and 3. Only invariants where all reflections satisfy the constraints defined by fields 4 and 5 are listed. See example in the GENSIN .				
change ^a	1-3	h, k, l of reflection to be changed		
	4	E value to replace value on input bdf		<i>not replaced</i>
	5	σ E to replace value on input bdf		<i>not replaced</i>

Notes:

- a. The **change** line is entered in the order of hkl on input bdf.

3.27. **GENTAN**: Tangent phasing

invar	control structure invariant relationships used
select	control starting phase selection process
phi	input specific starting phase
assign	assign starting phase by sequence number
refine	parameters controlling tangent refinement process
partsf	parameters for partial structure factor input
setfom	input figure-of-merit parameters
psets	specify phase sets to be output
dump	print detailed information on the phasing process
archiv	control items output to logical record <i>lrrefl</i> : of bdf

GENTAN	<i>Option</i>	<i>Code</i>	<i>Arg</i>		<i>Def</i>
	min E applied	emin	e	only E 's > e phased	
	max phases calc.	rmax	m	the m largest E 's phased	<i>from file</i>
					inv
	max phase sets	pset	s	calculate s permutations	64
	max phase sets bdf	pout	k	output k phase sets to the bdf	4
	list phase sets	plis	x	list top x phase sets (max 8)	0
	phase convergence	pacc		fix phase when converged	
		nacc		phase refined every cycle	
	bdf starting phases	nbdff		do not use bdf phases to start process	
		pbdf		use bdf phases to start	
	data set number	dset	n		1
invar	1			invariant types to be used in the phasing process use triplet and quartet invariants all	
				use triplets only	trip
				use quartets only	quar
	2			minimum A of triplets used	<i>from file</i>
					inv
	3			minimum B of quartets used	<i>from file</i>
					inv
	4			maximum triplets allowed per generator	100
	5			maximum quartets allowed per generator	100
	6			reduce A and B values by this fraction of σ A and B	1
	7			control eligible quartets according to X-vector sum	
				include quartets with	parxv
				XVsums<LO or >HI	
				include all quartets	allxv

		include quartets with X-vectors in the data sphere	insxv		
		include only quartets with X-vectors out of sphere	outxv		
		use negative quartets actively	active		
8		lower limit for cross-vector sums - LO			25% of max to 2.5
9		upper limit for cross-vector sums - HI			75% of max to 4.0
10		invariant ψ applied to negative quartets (in degrees)		if no fragment ψ estimate	180
				otherwise	0
select	1	method to select starting phases			
		permute phases in specified increments(<i>def: centro</i>)	perm		
		permute phases using magic integers(<i>def: noncentro</i>)	magic		
		assign random values to all but ODR phases	random		
2		number of initial starting phases selected (other than ODR, EDR)		if random	0
				if permute ,	1
				if magic	2
3		maximum starting phases selected with MAXEXT process		<i>number needed to connect 75% of field 4</i>	
4		maximum generators used in MAXEXT selection process			100
5		types of relationships used in selection processes			
		use all invariants except negative quartets	all		
		use only triplet relationships	trip		
		use only quartet relationships	quar		
6		criteria for MAXEXT selection process			
		test for maximum sum of(weighted α * generator)	gwa		
		test for maximum reflections phased in oneiteration	gen		
		test for maximum invariants involved in oneiteration	inv		
		test for maximum sum of(weighted α)	wal		
7		increment for permuting restricted phases in degrees			180
8		increment for permuting nonrestricted phases in degrees			90
9		starting weight for phases assigned in random mode			0.25

	10	integer used to seed random phase generator		137
phi	1-3 4	hkl indices of input phase phase (in degrees)		<i>restricted value or 45</i>
	5	starting phase status code permute starting phase per (permutable) origin defining phase odr (non-permutable) enantiomorph phase edr (permutable) fixed starting phase set (non-permutable) disable phase from refinement process null		
	6	fix input phase for n refinement iterations	for per or set for edr for odr	5 10 30
	7	permutation increment between phase sets (edr and per)	for restricted phases	180
	8	phase weight for refinement	otherwise if not null then else	90 1 0
assign ^a	1-	starting phase status codes followed by groups of sequence numbers		
Notes:				
a. Status codes are the same as defined for the phi line. Generator sequence numbers are in order of decreasing E value as listed by GENSIN. The status code of a series of starting phases precedes their generator sequence numbers (see <i>GENTAN</i> in the Reference Manual). Any number of assign lines may be entered.				
refine	1	phase extension and refinement procedure estimate new phases during tangent iteration cascade estimate new phases after tangent iteration block		
	2	tangent weighting scheme probabilistic weights w1 Hull-Irwin statistical weights w2 modified Hull-Irwin statistical weights w3		
	3	maximum number of refinement iterations		30
	4	minimum weight to phase extension eligibility		0.15
	5	initial α_2 threshold to extension eligibility		<i>auto</i>
	6	reduction fraction of α_2 threshold per cycle		0.7

	7	refinement cutoff as fraction of average α_2 shift	0.02
	8	fix odr phase for n refinement iterations	30
	9	fix edr phase for n refinement iterations	10
	10	fix perm phase for n refinement iterations	5
	11	fix set phase for n refinement iterations	5
	12	disable null phase for n refinement iterations	30
partsf	1	control application of fragment ψ estimate include ψ estimated from fragment data qpsi exclude ψ estimated from fragment data nqpi	
	2	control use of starting phases from input archive bdf do not use phases from input archive bdf no include FC phases according to 4, 5 and 6 below fc include GENTAN phases according to 3 and 4 below gt	
	3	phase set number of GENTAN phases to be entered	1
	4	minimum $ E $ of phases to be entered	<i>emin</i>
	5	fraction x of F_{cal} phases accepted if $F_{cal} > x * F_{rel}$	0.5
	6	F_{rel} scale applied before test in field 5	<i>bdf value</i>
setfom	1	control flag for pretest of FOM values pretest FOM values after the 6th tangent iteration pret do not pretest FOM values nopr	
	2	optimum RFOM value for CFOM/AMOS calculation	1.0
	3	optimum RFAC value for CFOM/AMOS calculation	0.25
	4	optimum PSI0 value for CFOM/AMOS calculation	1.5
	5	optimum NEGQ value for CFOM/AMOS calculation	60
	6	weight of RFOM value for CFOM/AMOS calculation	1
	7	weight of RFAC value for CFOM/AMOS calculation	1
	8	weight of PSI0 value for CFOM/AMOS calculation ^a	2

9	weight of NEGQ value for CFOM/AMOS calculation ^a	2
10	automatic termination of multi-solution process continue even if FOMs < <code>cont</code> optimal FOMs	yes
11	RFOM lower rejection limit for 014 and 024	0.25

Notes:

- a. The PSIO weight is scaled down if the number of PSIO triplets is less than 100. The reduction factor is the number of triplets per 100. The NEGQ weight is similarly adjusted according to the number of negative quartets.

psets ^a 1- specific phase set numbers to be output to the bdf

Notes:

- a. Used to select specific phase sets, independent of their FOM values, when repeating an *identical* GENTAN run.

dump	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	generator & selection list	<code>sel</code>		print start set data
	specify phase set limits	<code>pse</code>	<code>m n</code>	phase sets <code>m</code> to <code>n</code> 1 1
	generator set limits	<code>gen</code>	<code>t u</code>	generators <code>t</code> to <code>u</code> <i>all</i>
	print starting phases ^a	<code>sta</code>		starting phase set
	print phase iterations ^a	<code>ite</code>	<code>p q</code>	iterations <code>p</code> to <code>q</code> 1 99
	print invariants ^{a b}	<code>inv</code>		print each invariant
	print phase history map ^a	<code>map</code>		print phase est./iteration
	print FOM sums	<code>fom</code>		print sums used in FOM's
	terminate phasing process	<code>hal</code>	<code>t</code>	stop after phase set <code>t</code> 1

Notes:

- a. These print dumps are limited by `pset` and `gen` entries.
b. CAUTION! This printout can be enormous unless limited.

archiv ^a 1- bdf ID numbers to be added or deleted from logical record
lrreft:

Notes:

- a. Normally this line is not entered. The items added to the reflection record are controlled via `pout` of the **GENTAN** control line, not this line.

Caution: A maximum of 70 items may appear in the archive list and 64 are generated automatically in order to delete existing phase sets and weights. This means that a maximum of 6 user-generated deletes may be entered on this line. Negative identification numbers indicate the items to be deleted (if present) from the bdf.

3.28. *GIP*: Reciprocal space PIG

GIP	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	graphics display	<code>dbuf</code>		double buffering

3.29. *LATCON*: Refine lattice parameters

theta reflection line
skip skip reflection stored on the bdf

LATCON	<i>Option</i>	<i>Code</i>	<i>Arg</i>		<i>Def</i>
	crystal system	<u>tr</u>		triclinic	
		<u>mo</u>		monoclinic b-axis unique	
		<u>mc</u>		monoclinic c-axis unique	
		<u>mb</u>		monoclinic b-axis unique	
		<u>ma</u>		monoclinic a-axis unique	
		<u>or</u>		orthorhombic	
		<u>te</u>		tetragonal	
		<u>he</u>		hexagonal	
		<u>rh</u>		rhombohedral	
		<u>cu</u>		cubic	
	wavelength	<u>wl</u>	n	in Å	1.54183
	σ (wavelength)	<u>sw</u>	n	in Å	0
	2θ zero	<u>tz</u>	n	in degrees	0
	refine Q-values ^a	<u>rq</u>			
	refine 2θ zero ^a	<u>rz</u>	n	refine for n cycles	4
	input from bdf	<u>bi</u>		crystal system and hkl angles	
	output to bdf	<u>bo</u>		refined cell parameters	
		<u>ds</u>	n	dataset number	1

Notes:

a. Note that the **rq** and **rz** are mutually exclusive.

theta	1-3	h, k, l indices		
	4	2θ in degrees		
	5	$\sigma(2\theta)$		0.1
	6	wavelength (if different from LATCON line)		
skip	1-n	sequence number(s) of reflection(s) NOT to be used in refinement		

3.30. **LISTFC**: List formatted reflection data

output	specify items to be listed
format	specify formats of items to be listed
pscale	print scales of items to be listed
scale	structure factor scale
maxhkl	specify max h, k, l and min/max $\sin \theta/\lambda$ to be listed

LISTFC	<i>Option</i>	<i>Code</i>	<i>Arg</i>		<i>Def</i>
	hkl indices listed	<u>ind</u>	n	indices listed (1 or 3)	1
	hkl sort order ^a	<u>ord</u>	hkl	hkl chars in order slo/med/fast <i>bdf</i>	
	list rcode flags	<u>nfl</u>		rcode flags will not be listed	
		<u>fla</u>	f	four chars for rcodes 2345	-***
	output <i>pch</i> file	<u>pun</u>		punch data (80 chars max)	
		<u>npun</u>		no punch output	
	output line width ^b	<u>wid</u>	w	chars per output line	<i>local</i>
					<i>default</i>
	output lines/page ^b	<u>lin</u>	l	lines per page header	<i>local</i>
					<i>default</i>
	dataset number	<u>dse</u>	n		1
	print void items	<u>vbl</u>		print void items as blank	<i>no</i>

Notes:

- a. Only used if the SORTRF sort order code is not on the bdf. If data is unsorted, `ind 3is` is recommended because LISTFC does *not* sort reflection data!
- b. These parameters may be used to output one reflection per line for use with external software. To do this, set `wia` to the number of chars in one reflection and `lin` to a very large number.

output ^a 1-20 items output (as codes *or* bdf ID numbers)

Notes:

- a. The item specification for field 1 MUST be either `hkl` or 1. Alphabetic codes accepted are those defined for ADDREF `bdfin` (with the exception of `all` and Friedel reflection packets) and the addition of `f2c1` and `ica1` for outputting calculated intensity I and F_{2cal} and `ffrel`, `fsgf`, `ff2r1` and `fsgf2` for Friedel data. Typical codes are `fre1`, `fca1`, `sigf`, `pcyc` or `pdeg`. Each item must be specified in order of output. A maximum of 20 items may be entered.

NOTE that if the `output` line is not entered `hkl fre1 fca1 and sigf` will be listed.

format 1-20 format of output items as 'Inn' or 'Fnn.m' I3 for hkl, else I5

pscale 1-20 print scale of output items ^a 1 for hkl, else 10

Notes:

- a. Defaults only apply if `output` line not entered.

scale	1	F _{rel} scale	1.0
	2	scale group number	1
maxhkl	1-3	max h , k , l output	9999
	4	min $\sin\theta/\lambda$ output	0
	5	max $\sin\theta/\lambda$ output	2

3.31. LSABS: Absorption corrections

face input of bounding faces
sphere input of sphere description
shape crystal shape description
grid Gaussian grid description
sgrid Gaussian surface grid description
noabs skip calculation for selected reflections

LSABS	Option	Code	Arg	Def
	absorption coefficient in mm^{-1}	<code>mu</code>	<code>v</code>	<i>auto</i>
	data set number	<code>dset</code>	<code>n</code>	<i>1</i>
	absorption method	<code>anal</code>		Tompa method
		<code>gauss</code>		Gaussian integration
	list all reflection data	<code>list</code>	<code>n</code>	<i>all</i>

face	1-3	Miller indices of face	
	4	distance from origin to face in mm	
	5	esd of distance in mm	<i>5% of distance</i>
	6	weight of observation	<i>1/esd₂</i>

sphere	1	radius in mm	
	2	esd of radius in mm	<i>5% of radius</i>
	3	weight of observation	<i>1/esd₂</i>

shape	1	shape option code: (fixes grid z axis for Gaussian integration) plate shape	<code>plate</code>
--------------	---	--	--------------------

		needle shape for <code>plate</code> , Miller indices ($h k l$) of the plate for <code>needle</code> , zone axis $[u v w]$ of the needle direction	<code>needle</code>	
<code>grid</code>	1-3	number of grid points in the 3 dimensions		16
<code>sgrid</code>	1	number of grid points in the 2 dimensions		16
<code>noabs</code>	1	serial number of first reflection to be skipped		
	2	serial number of last reflection to be skipped		<i>field 1</i>

3.32. *L*SLS: Structure factor least-squares refinement

<code>maxhkl</code>	set maximum $ h , k , l $ and minimum/maximum values of $\sin\theta/\lambda$
<code>twinop</code>	twinning operation in reciprocal space
<code>noref</code>	keep one parameter (or a set of parameters) fixed
<code>ref</code>	turn on refinement of one (or a set of) parameter(s)
<code>constr</code>	linear constraint between parameters
<code>limsft</code>	shift limitation of one parameter

<i>L</i> SLS	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	number of cycles	<code>cy</code>	<code>n</code>	1
	atomic-displacement type	<code>mx</code> <code>an</code> <code>is</code> <code>ov</code>		mixed anisotropic isotropic overall
	weighting scheme	<code>ws</code> <code>wu</code> <code>ww</code>		1/esd ₂ unit weights weights from bdf
	refinement type	<code>in</code> <code>f2</code>		intensities $ F _2$
	scale factors	<code>us</code> <code>ms</code>		unit scale many scales
	crystal shape	<code>rs</code>		refine
	population parameters	<code>pp</code>		refine
	dispersion parameters	<code>nd</code> <code>ad</code> <code>rd</code>		no dispersion apply apply & refine
	extinction parameters	<code>nx</code> <code>ax</code> <code>rx</code>		no extinction apply apply & refine
	twinning parameter	<code>np</code> <code>ap</code> <code>rp</code>		no twins apply apply & refine
	data set number	<code>ds</code>	<code>n</code>	1
	skip reflections	<code>sk</code>	<code>n</code>	no skip
	damping factor	<code>fu</code>	<code>x</code>	1.0
	correlation between measured dist	<code>cd</code>	<code>x</code>	0.0
	save variance-covariance matrix ^a	<code>sm</code>		last cycle

list normal equations matrix	lm	n		
list variance-covariance matrix	lc		last cycle	
non-positive definite displ	tr		reset	
params				
	tp		print warning	
	ts		stop	
twinning parameter <0 or >1	er		reset	
	ep		print warning	
	es		stop	
crystal shape parameter <0	sr		reset distances	
	sp		print warning	
	ss		stop	
scale not positive	kr		reset	
	kp		print warning	
	ks		stop	
list correlations	pc	x	list corr ≥ x	
list poor reflections	lp	x	if weighted dif ≥ x	
list reflections on last cycle	ll	n		<i>all</i>
every cycle	le	n		<i>all</i>
list restraints on last cycle	tl	n		<i>all</i>
every cycle	te	n		<i>all</i>
refinement on restraints only	ro			
do not use reflection in matrix	cs	x	I/esd(I) = x	
punch parameters	pl		last cycle	
reject reflection	rj	x	if weighted dif = x	

Notes:

- Variance-covariance matrix is saved according to **REGFE** specifications.
- If **n** is greater than 0, list on every cycle, else only on the last cycle. 1 to list only inverted matrix and shift vector, 2 to list direct and inverted matrices and both vectors.

maxhk1	1-3	max h , max k , max l used in refinement
	4-5	min & max of sin θ/λ used in refinement
twinop	1-9	matrix of symmetry operation in reciprocal space coded as : m ₁₁ , m ₁₂ , m ₁₃ , m ₂₁ , m ₂₂ , m ₂₃ , m ₃₁ , m ₃₂ , m ₃₃
	10	Twinop number
	11	twin component 1
	12	twin component 2 (Friedel)

noref ^a 1- non-refined parameter(s)

Notes:

- General parameters are: scale- **skf**, crystal shape- **shp**, twinning parameter- **twi**, extinction parameter- **ext**, dispersion parameter- **dsp**, real part of dispersion- **re**, imaginary part of dispersion- **im**, overall displacement parameter- **uov**. The syntax to address a general parameter is: param(number). Atomic parameters are: **x**, **y**, **z**, **u**, **u11**, **u22**, **u33**, **u12**, **u13**, **u23**, **pop**. The syntax to address an atomic parameter is: param(atom name). Sets of parameter can be grouped using (param1/param2), (number1/number2), (atom1/atom2).

ref ^a 1- refined parameter(s)

Notes:

- See **noref** line for the description of the parameters.

constr 1 constraint written as :
subject parameter = constant μ factor * reference parameter
&± ... ^a

Notes:

- There can be more than one reference parameter. See **noref** line for the description of the parameters. Blanks are not accepted in the field.

limsft 1 shift-limited parameter ^a
 2 weight of pseudo-observation ^b

Notes:

a. See **noref** line for the description of the parameters.

b. If the weight is less than 1, it is assumed to be the Levenberg-Marquardt parameter.

3.33. *LSQPL*: Least-squares planes & lines

plane ^a initiates plane calculation
line_a initiates line calculation
define atoms which define a line or a plane
nondef atoms which do not define a line or a plane
site site coordinates and estimated standard deviations

Notes:

a. Either a **plane** or a **line** must be entered before the define line(s).

LSQPL	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
--------------	---------------	-------------	------------	------------

plane	1-3	fractional coordinates x, y, z of centre of symmetry	^a	
--------------	-----	--	--------------	--

Notes:

a. Fields 1-3 are optional, but **plane** must be present to initiate a plane calculation.

line	1-3	fractional coordinates x, y, z of centre of symmetry	^a	
-------------	-----	--	--------------	--

Notes:

a. Fields 1-3 are optional, but **line** must be present to initiate a line calculation.

define ^a	1-	atom labels extracted from the bdf, or site labels entered as lines		
----------------------------	----	---	--	--

Notes:

a. Defining atoms are used in the least-squares plane or line calculations.

nondef ^a	1-	atom labels extracted from the bdf, or site labels entered as lines		
----------------------------	----	---	--	--

Notes:

a. Non-defining atoms are used only to calculate the atom-to-plane or -line distances.

site	1	site label		
	2-4	fractional coordinates x, y, z		
	5-7	$\sigma_x, \sigma_y, \sigma_z$		

3.34. *LSRES*: Restraint editor

symeq symmetry-equivalent atom builder
restr input one restraint
delres delete one or more restraints

LSRES	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	mode of bdf usage	<u>upd</u>		update old restraints
		apr		create new restraints
	restraint listing	list	n	<i>all</i>

symeq	1	root atom name	
	2	symeq atom name	
	3-5	symmetry-equivalent position	
restr	1	restraint type code	
		bond length	dist
		bond angle	angl
		dihedral angle	dihe OR
			tors
		rigid bond	bond
	2-N	atom names ^a	
	N+1	restraint value	
	N+2	esd of restraint value	
	N+3	weight of restraint value	<i>1/esd₂</i>

Notes:

- a. 2 names required for **dist** and **bond**, 3 for **angl**, 4 for **dihe** and **tors**

delres	1	restraint type code	
		bond length	dist
		bond angle	angl
		dihedral angle	dihe OR
			tors
		rigid bond	bond
	2-N	atom names	

3.35. **MAPLST**: List map densities

MAPLST	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	input map filename extension		ext	map

3.36. **MAPXCH**: Import theoretical electron density maps **NEW**

uov set an overall temperature factor

MAPXCH	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	load external 3D map	abi2map w2k2map		from abinit _DEN file from Wien2k lapw5_3D case.tmp file
	path to file	inpfiler	name	
	Xtal map terminal treatment	noterm		switch off duplicated layer/line/point
	Lattice symmetry checking	no latch		switch off STARTX checks
uov	1	force an effective overall temperature factor on imported data		<i>.035</i>

3.37. **MODEL**: Build molecule from map

limits sets search parameters for Fourier peaks
connect defines the connectivity of expected atomic sites

MODEL	Option	Code	Arg	Def
	extend input site list	sext		extend symmetry sites ^a
		nose		no symmetry extension ₁
	use atom sites	atom		include atom sites on <i>pek bdf</i>
		noat		exclude atom sites on <i>pek bdf</i>
	subcluster search	conn		use connectivity priority
		none		no angle limits applied
		mean		use mean distances and angles

Notes:

a. If the number of molecules per cell is less than the number of symmetry equivalent positions, the input sites will automatically be expanded to include symmetrically equivalent sites (i.e. **sext**), otherwise **nose** is the default. The user may override these defaults with this option.

limits ^a	1	bond connections permitted for each atom (max 12)	4
	2	min bond radius for peak site (in Å)	0.5
	3	max bond radius for peak site (in Å)	1
	4	min angle allowed in subcluster (in degrees) ^b	85
	5	max angle allowed in subcluster (in degrees) ^b	145
	6	total number of sites (atom and peak) to be entered	<i>all</i>
	7	threshold for contact distances to be listed	2.4
	8	increment to atom radii on the <i>bdf</i>	0.1
	9	maximum number of interpretations printed per cluster	3
	10	number of projections to be plotted for each atom cluster	1 <i>for planar clusters,</i> 2 <i>for globular clusters</i>

Notes:

a. **WARNING**: The ability of MODEL to correctly connect peak (not atom) sites is very dependent on the values in fields 1-5 being appropriate for the structure under study.

b. See also subcluster search mode on the **MODEL** line.

connect 1- numeric group numbers or atom labels ^a 1

Notes:

a. Numeric fields specify the group number of the atom sites that follow. They are used if several atom fragments or molecules are expected in an asymmetric unit.

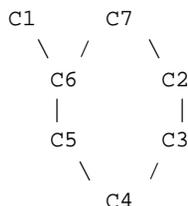
Atom labels specify the atoms connected in the group. Connected sites are entered as sequence of atom labels.

(i) an atom label is delimited by a blank

(ii) a sequence is delimited by a semicolon (;) or new **connect** line

(iii) the central atom is the first label in the sequence

(iv) redundant connections need not be entered



The connections of the preceding fragment would be entered as:

```
connect C2 C7; C2 C3; C3 C4; C4 C5; C5 C6; C6 C1
```

3.38. *MODHKL*: Modify reflection data

remove	globally remove an item from all packets
rcode	apply global changes to reflection codes
purge	remove reflection packets from output bdf
modify ^a	change items for a specific hkl
delete ^a	delete a packet for a specific hkl
modmul ^a	change items for a specific occurrence of an hkl
delmul ^a	delete a packet for a specific occurrence of an hkl

Notes:

a. These must follow any **remove**, **rcode** or **purge** lines.

MODHKL	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	list modified reflections	list		<i>no</i>
rcode	1	output rcode		
	2	multiplication factor 1 - condition 1		1
	3	item 1 logical record <i>lrrefl</i> : ID number - condition 1		<i>not used</i>
	4	multiplication factor 2 - condition 1		1
	5	item 2 logical record <i>lrrefl</i> : ID number - condition 1		<i>not used</i>
	6-9	equivalent to fields 2-5, but for condition 2		
purge	1	multiplication factor 1 - condition 1		1
	2	item 1 logical record <i>lrrefl</i> : ID number condition 1		<i>not used</i>
	3	multiplication factor 2 - condition 1		1
	4	item 2 logical record <i>lrrefl</i> : ID number condition 1		<i>not used</i>
	5-8	equivalent to fields 1-4, but for condition 2		
remove	1-	logical record <i>lrrefl</i> : ID numbers for items to be removed		
modify	1-3	hkl indices of reflection to be modified		
	4	logical record <i>lrrefl</i> : ID number of item to be modified		
	5	new value of item identified in field 4		
	6-7	as in 4-5, or use separate modify lines		
delete	1-3	hkl indices of reflection to be deleted		
modmul	1-3	hkl indices of reflection to be modified		
	4	occurrence number		<i>all occurrences</i>
	5	logical record <i>lrrefl</i> : ID number of item to be modified		
	6	new value of item identified in field 5		
	7-8	as in 5-6, or use separate modmul lines		
delmul	1-3	hkl indices of reflection to be deleted		
	4	occurrence number		<i>all occurrences</i>

3.39. *NEWCEL*: Apply unit-cell change

transf define transformation matrix

NEWCEL	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	non-integer transformed	stop		stop and complain
	Miller indices	remv		remove non-integer reflections

transf 1-9 transformation matrix $T_{11}, T_{12}, T_{13}, T_{21}, T_{22}, T_{23}, T_{31}, T_{32}, T_{33}$ a

Notes:

- a. Transformation is $\alpha' = T\alpha$, α is a column vector of the unit cell vectors **a**, **b**, **c**.

3.40. **NEWMAN**: Newman projection of model

atrad specify atom radius
select selects atoms from the bdf
bondin bond information
projax defines Newman projection axis
plotp plot data
ptitle a regular title of the plot

Notes:

- a. Must be preceded by a **plotp** line.

NEWMAN	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	min. bond distance	bxmn	q	0.25
	output plot file	nplt		no plot file
		plot	n	yes with n plots/frame
	plot width in inches	wdth	q	11

atrad 1 atom type code
 2 atomic radius in Angstroms
 3- as above for new atom types

select 1- labels of atoms to be selected from the bdf

bondin 1 label of central atom a
 2- labels of peripheral atoms

Notes:

- a. These labels may be followed by an optional symmetry operator n_mmm , where n is the STARTX symmetry number and mmm is the lattice translation based on 555.

projax 1 label of central atom 1 a
 2 label of central atom 2 a
 3-4 labels for next projections
 5-6 etc

Notes:

- a. These labels may be followed by an optional symmetry operator n_mmm , where n is the STARTX symmetry number and mmm is the lattice translation based on 555.

plotp 1 number of plots per page (1 or 4) a
 2 width of total plot a
 3 height of total plot ϵ 11
 4 height of chemical symbols 0.02 ϵ
 5 height of angle values 0.02 ϵ
 6 height of characters in subtitle b 0.02 ϵ
 7 height of characters in regular title 0.02 ϵ
 8- sequence numbers of Newman projections to be plotted c

Notes:

- a. See also **NEWMAN** line
 b. No subtitle or title is plotted when the values of fields 6 & 7 are zero. When a title is requested, the height of the total plot is increased by twice the height of the characters in the title.
 c. The sequence numbers are in order of **NEWMAN** plots.

ptitle 1 title string (max 72 characters) ^a *compid*

Notes:

- a. The string should be centred around columns 43-44.

3.41. *ORTEP*: Molecular plot

select ^a select atoms from bdf
radius radius of a sphere for specific atom site
sphere radius of a sphere for atom type
height height limits for colour classification of atoms
vsc ^{b c} vector search code
symbol ^d alphanumeric string to be drawn on the plot
genins ^d generate an automatic instruction sequence
plotp ^d plot parameters for the automatic sequence
ellips ^d ellipsoid parameters for the automatic sequence
bond ^d bond parameters for the automatic sequence
symop symmetry translation for **symm** or **csym** option
molorg origin to start **mole** or **cmol** option
svstar ^{e f} start of a saved sequence
svend ^{e f} end of a saved sequence
svexec ^e invoke a saved sequence
inst ^{b e} instruction codes
seq ^g modify instruction array
inst instructions used only in modifications

Notes:

- a. If the **select** line is not used, all atoms in the bdf will be loaded.
b. Required control line when ORTEP is run in *manual* mode.
c. Entry of any **vsc** lines will replace the automatically generated **vsc** lines for the atom search. (The **vsc** lines used to specify the cell will be still be added automatically.)
d. This control line must *not* be used when ORTEP is run in *manual* mode.
e. This control line must *not* be used when ORTEP is run in the *automatic* mode.
f. If ORTEP is run in the manual mode and the **svstar** line is used, it must be followed by at least one **inst** line and then the **svend** line.
g. In the *modification* mode, the **seq** and **inst** lines must be used together; **seq** first followed by **inst**.

CAUTION: ORTEP is very sensitive to the *order* of the input optional lines.

ORTEP	Option	Code	Arg	Def
	plot style	acta		plot full ellipsoids, bonds & labels
	atom labels	brac nhla ialab		extension in brackets no hydrogen atom labels only label symmetry indep. sites
	plot type	mole inpu cell coor symm csym cmol	n	plot connected molecule ^a plot input atoms ^b plot atoms in cell ^c atoms connected to atom n inpu at symmetry site(s) ^d symm with cell outline ^d mole with cell outline ^b
	bond radius ^e	arad		atom radii from bd for radii line

	vrad	search radii based on input
		vsc ^e
input mode	<u>auto</u>	automatic
	manu	manual
plot perspective	<u>infi</u>	no perspective, infinite view
	pers	perspective view
cut at cell edge	<u>full</u>	extend molecule past cell edge
	mcut	cut molecule cell edge
hydrogen sites	<u>sph</u>	make hydrogens spheres ^f
	sort	sort to bottom of atom list
	excl	exclude hydrogen sites
	same	same as non-hydrogen
overlap correction	<u>over</u>	correct overlap ^g
	nove	no overlap correction
generate CPK model	<u>nocp</u>	no ball model
	cplm	space-filling ball model
output PS300 file	ps300	file for use on PS300 series
no bdf orientation	npig	do not use PIG orientation ^h

Notes:

- The **molog** line must be used if there is more than one molecule or unconnected fragment in the asymmetric unit. Note that this option is critically dependent on the max. bond radius (see **bond**). The **vsc** lines may have to be used in some cases.
- For **inpu** mode all atoms appear as they are input. In other modes (**mole**, etc.) all duplicate atoms are ignored.
- For a cell containing a wide range of bond lengths, **vsc** lines will be needed to define the acceptable connections.
- The **symop** line *must* be entered for this option.
- The default is **arad** if no **vsc** lines are entered, otherwise the default is **vrad**.
- Default radius is 0.1 Å but may be set by **sphere** line.
- If **over** and **manu** are specified, an **inst 511** must be included. If no overlap is desired in **manu** mode, **nove** must be specified.
- The default is to apply the PIG orientation matrix if present on the bdf. It is recommended that the **npig** option is used if input angle rotations are entered on the **plotp** line.

select 1- atom label of each atom to be extracted from the bdf
take all atoms from the bdf **all**

radii 1 atom type label
2 atomic radius in Angstroms
3- as above

radius ^{a b} 1 atom label
2 radius of displayed sphere to replace U for this atom (in Å)

Notes:

- This radius is scaled by the 600 series instructions.
- The **radius** line sets the radius for a particular atom; The **sphere** line sets the radius for all atoms of that type.

sphere 1 atom type symbol
2 radius of displayed sphere to replace U for this atom type (in Å) ^a
3-4 as in fields 1-2, etc

Notes:

- Do not enter an **ellips** line.

height 1 lowest height limit for colour classification of atoms
2 2nd lowest
3 3rd lowest
...
8 greatest height limit for colour classification of atoms

vsc	1	atom number of the first atom of the origin ANR ₁	
	2	atom number of the last atom of the origin ANR ₁	
	3	atom number of the first atom of the target ANR ₁	
	4	atom number of the last atom of the target ANR ₁	
	5	bond type (any number from -5 to +7) ²	
	6	minimum bond length, in Å ₃	
	7	maximum bond length, in Å	
	8	radius of stick bond, in Å	
		<i>Normal Bond Plotting Format</i>	
	9	perspective bond label height, in inches	
	10	perpendicular displacement, in inches	
	11	nonperspective label height, in inches	
	12	perpendicular displacement, in inches	
	13	number of decimal places for one, two and three decimal places	-1, 0, 1
	14	increment for dotted or dashed lines ²	0.33
	15	dot length as fraction of the dot increment	0.15
		<i>Alternate Polyhedra Bond Plotting Format</i>	
	9	negative atom number of the first atom of polyhedra ANR	
	10	atom number of the last atom of polyhedra ANR	
	11	polyhedra dmin, in Å	
	12	polyhedra dmax, in Å	

1 *Caution:* Note that the checking order of the **vsc**'s requires that the origin ANR's include the largest range of atoms and the target ANR's the least. **2** *Magnitudes* 0, 1, 2, 3, 4, 5 form bonds with 0, 2, 3, 5, 9, 17 lines respectively. Negative integers are used for bonds terminating only at ellipsoids; positive integers are used for bonds terminating at ellipsoids or at the tangent cone with the apex at the viewpoint. The value +6 gives a single dashed line; +7 gives a single dotted line. **3** *Options* appearing in fields 5, 6, or 7 of the first **vsc** line may be overridden if options are present in fields 1, 2, or 3 of the **bond** line.

symbol text to be printed on plot (max. 72 characters)

genins 1- option control codes

calculate bond lengths and angles	cbla
list plotted atoms and bonds	list
draw atom labels ^a	symb
plot a stereoscopic pair	ster
plot atoms as a skeleton model	skel
do not place the title on the plot	not1

plot axes	axes n	n=1 add axes labels to cell plot	1
		n=2 unit length auxilliary axes	
		n=3 proportional length auxilliary axes	

Notes:

a. Default alignment of symbols is at the centre of the atoms. This may be changed using the 700 series **inst** lines.

plotp	1-2	x, y dimensions of the plot, in inches	11
	3	view distance, in inches	2.5 max. <i>dim</i>
	4	margin width, in inches	0.1 <i>x-dim</i>
	5	x-y axes of the plot plane possible axes are	
		ab	
		ba	
		ac	
		ca	
		bc	
		cb	
	6	plot scale, in inches per Angstrom	<i>automatic</i>
	7-9	rotation of the origin about x, y, z axes, in degrees ^a	0
	10	margin for retracing lines, in inches ^b	0
	11	margin for overlapped lines, in inches ^c	0

Notes:

a. The positive rotation direction is taken as anti-clockwise looking down the axis toward the origin. Note the **npig** option on the **exec** line.

b. See 303 and 700 **inst**.

c. See 511 **inst**

ellips	1	ellipsoid type	
		point	1
		spheroid ^a	2
		envelope	3
		principal ellipsoid and envelope	4
		principal ellipsoid, envelope, and axes	5
		ellipsoid with octant shading	6
	2	ellipsoid probability scale or radius of spheroid ^b	1.54 0.13
	3	x displacement of atom labels, in inches	<i>auto centre</i>
	4	y displacement of atom labels, in inches	<i>auto centre</i>
	5	height of atom labels, in inches	<i>auto</i>

Notes:

a. Note that this setting will cause *all* atoms to be treated as spheroid with a radius set in field 2. The **sphere** line enables each atom type to be plotted as spheres of specified radius.

b. The following table shows the probability that the ellipsoid encloses the atom versus the ellipsoid scale.

<i>prob(%)</i>	10	20	30	40	50	60	70	80	90	99
<i>scale</i>	0.76	1.0	1.19	1.37	1.54	1.72	1.91	2.15	2.50	3.37

bond ^a	1	bond type (any number from -5 to +5) ^b	1
	2	maximum bond length, in Å	2
	3	radius of stick bonds, in Å	0.03
	4	draw bond length values on plot	<i>no lengths</i>

draw bond length value on plot **draw**

^c

Notes:

a. Caution should be observed when using the **bond** line. Options appearing in fields 1, 2, or 3 of the **bond** line will override options appearing in fields 5, 7 or 8 of the first **vsc** line. This line should be used only if **vsc** lines are not used.

b. See comment 1 on **vsc** line.

c. This causes default values for fields 9-13 of the first **vsc** lines to be .2, 0, .2, 0.

symop ^a	1	symmetry operation number as stored on the bdf by STARTX	1
	2-4	additional unit translations along x, y, z	0

Notes:

a. This line is only entered when the **symm** or **csym** option is used.

molorg	1	atom label of origin atom for 'molecule' search ^a	
	2	symmetry operation number stored on the bdf by STARTX	1
	3-5	additional unit translations along x, y, z	0

Notes:

a. This line is only entered when the **molc** or **cmol** option is applied. More than one **molorg** may be entered. Used to start search in unconnected atom clusters.

svstar name of the sequence to be saved (6 chars max)

svend *There are no arguments for this line.*

svexec name of the saved sequence to be invoked (*to match svstar name*)

seq	1	sequencing function in the instruction array	
		replace specified instruction	replace
		insert before specified instruction	precede
	2	number of the instruction to be replaced or preceded	

inst

101-102,

105-106

1	print distances	101
	print distances and angles	102
	print distances for spheres of enclosure	105
	print distances for reiterative spheres of enclosure	106
2-7	origin ADR (atom designator run) ^a	
8-9	target ANR (atom number run) ^a	
10	maximum distance in Å	
11-12	not used	

- 13 serial number of first VSC applied ^b
- 14 serial number of last VSC applied ^b

Notes:

- a. See *ORTEP* for format of ADR's and ANR's.
- b. These fields only apply to instructions 105 and 106. If field 13 is 0 (not void), the VSC's will not be used and the distance in field 10 is used.

inst 103
 1 print thermal ellipsoids 103

inst 201-203
 1 start plot 201
 advance plot 202
 end plot 203
 2-3 shift plot origin in x, y in inches (202 only)

inst 301
 1 define plot dimensions 301
 2-3 plot limit x, y in inches
 4 view distance in inches
 5 border in inches

inst 302
 1 title rotation 302
 2 title rotation angle in degrees

inst 303
^a
 1 retrace displacement 303
 2 displacement parameter in inches

Notes:

- a. See **inst 700** series lines for retrace parameters.

inst 401, 410, 411
 1 add atoms to atom array 401
 delete all atoms from the atom array 410
 delete atoms from atoms array 411
 2-7 ADR (atom designator run) ^a

Notes:

- a. If only single atoms are added/deleted use only three fields for the appropriate atom designator code (ADC). Several single atoms per line may entered. ADR sequences are recognised by a negative atom number in ADC2.

inst 402-407, 412-417
 1 *add - delete* atoms within:
 sphere of enclosure 402 412
 box of enclosure 403 413
 triclinic box of enclosure 404 414

	expand sphere enclosure	405 415	
	reiterate convolute sphere enclosure	406 416	
	as 406 but sites related by celltranslation ignored	407	
2-7	origin ADR		
8-9	target ANR		
10-12	semi-dimensions of box in fract. coords(403-404 only)		0.5
13	serial number of the first VSC which applies ₁ (not403-404)		
14	serial number of the last VSC which applies ₁ (not403-404) for 405, 406, 415, 416		
10	max distance in Å		2
11-12	not used		
13	serial number of the first VSC which applies _a		
14	serial number of the last VSC which applies _a		

Notes:

- a. See the explanation for the `inst` line 101-102, 105-106.

inst 501

1	define the plot orientation	501	
2-4	ADC for origin atom		
5-10	ADR for vector V1		
11-16	ADR for vector V2		
17	orientation type		
	X=V1,Y=V1xV2,	0	Z=V1x(V1xV2)
	X=V1,Y=(V1xV2)xV1,	1	Z=V1xV2

inst**502, 503**

1	rotate plot reference system	502
	rotate plot working system	503
2	type of rotation	
	rotate about X	1
	rotate about Y	2
	rotate about Z	3
	rotate 120 degrees about body-diagonal	-1
	rotate 240 degrees about body-diagonal	-2
3	rotation angle in degrees <i>+ve direction is anti-clockwise, looking towards origin</i>	
4-5	same as fields 2-3, and so on	

inst 504

1	shift origin of projection axis	504
2-4	x, y, z shift in inches (scaled by the 600 series instructions)	

inst 511

a

1	apply overlap correction - hidden line removal	511	
2	overlap margin, in inches		0

Notes:

- a. This line must follow any **inst 600** series lines.

inst

**601-604,
611-613**

1	explicit centre and scale	601	
	explicit centre, auto scale	602	
	auto centre, explicit scale	603	
	auto centre and scale	604	
	increment centre and scale	611	
	increment centre, auto scale	612	
	auto centre, increment scale	613	
2	x centre or increment in inches		
	^a		
3	y centre or increment in inches		
	^a		
4	scale or scale increment ^a		
5	ellipsoid scale ^b		1.54

Notes:

- a. Value or increment is determined by instruction number.
b. See **ellips** line for the table of probability versus scale.

inst

**701-704,
711-714**

1	instruction number (700's list data; 710's do not)		
	plot atoms as shaded ellipsoids	701	711
	plot atoms as unshaded ellipsoids	702	712
	plot atoms as ellipsoid with bounding ellipses	703	713
	plot atoms as boundary ellipse only	704	714
2-5	not used		
6	symbol height in inches		
7	parallel offset for symbols in inches		
8	perpendicular offset for symbols in inches		
9	increase in radial dimension for retracing in inches ^a		
10	rate of increase in radial dimension with height ^a		
11-12	ANR applied to boundary retracing atoms ^a		

Notes:

- a. Fields 9-12 are used only if boundary retracing and plot constraints are needed as a result of an **inst 303** line being entered.

inst**705, 715a**

1	define ellipsoid parameters and list atoms define ellipsoid parameters and do not list atoms	705 715
2	ellipsoid type no ellipsoid components boundary ellipse only principal ellipses only boundary and principal ellipses	 0 1 3 4
3	back side of principal ellipses solid line omitted 4 dots 8 dots 16 dots 32 dots	 -1 0 3 4 5 6
4	forward principal axes omitted plotted plotted and shaded with 1 to n-1 lines	 0 1 2-n
5	reverse principal axes omitted plotted and shaded with 1 to ndashes	 0 1-n
6	symbol height in inches	
7	parallel offset for symbols in inches	
8	perpendicular offset for symbols in inches	
9	increase in radial dimension for retracing in inches	
10	rate of increase in radial dimension with height	
11-12	ANR for boundary retracing	

Notes:

- a. In fields 1-5 voids are not the same as 0. Fields 9-12 are used only if boundary retracing is applied as a result of an **inst 303** line.

inst**801, 811**

a

1	plot explicit stick bonds and list distances plot explicit stick bonds and do not list distances	801 811
2-7	ADR for bond	
8-13	ADR for next bond, and so on	

Notes:

- a. Up to 3 bonds can be specified per line. The bond type specified on the first **vsc** line applies to all **801** defined bonds. At least one **vsc** must be specified for the **801** to work.

inst**802, 803****812, 813**

1	instruction number (800's list bond data;810's do not)	
	plot generated stick bonds	802 812
	plot generated line bonds	803 813
2-7	ADR for origin atom	
8-13	ADR for target atoms	
14	bond type _a (any number -5 to +5)	
15	minimum bond distance in Å	
16	maximum bond distance in Å	
17	bond radius in Å _a	
18	perspective label height in inches _a	
19	perpendicular displacement in inches _a	

Notes:

a. See description of **vsc** line.**inst****901-909****913-916**

1	instruction number (900's for projection;910's for perspective)	
	plot chemical symbol	901
	plot regular title	902
	plot vector title	903 913
	plot bond labels (1 decimal place)	904 914
	plot bond labels (2 decimal places)	905 915
	plot bond labels (3 decimal places)	906 916
	plot auxilliary axes	907 917
	plot centred symbol, pen up	908
	plot centred symbol, pen down	909
2-4	ADC of atom 1 on vector baseline	
5-7	ADC of atom 2 on vector baseline	
8	X edge reset in inches	
9	Y edge reset in inches	
10	lettering height in inches	
11	parallel offset in inches	
12	perpendicular offset in inches	
13	centred symbol, 0-14 (see p. 67 of Johnson, 1970)	
	907 & 917	
8	X location of auxilliary axes	
9	Y location of auxilliary axes	
13	axes type	1 cell labels

		2	unit length aux axes
		3	proportional length aux axes
14	fractional dimension of aux axes		
15	maximum dimension of aux axes		

3.42. **OUTSRC**: Export archive for external software **NEW**

OUTSRC	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	desired target	shelx		create SHELX control file shx.ins
		valray		create SHELX reflection file shx.hkl
				create VALRAY control file val.dat
				create VALRAY reflection val.hkl
		xd		file create XD control file xd.ins
				create XD reflection file xd.hkl
	use data set	dset	n	from archive bdf 1
	override output basefilename		name	first unrecognized <i>Code</i>

3.43. **PARTN**: Hirshfield partitioning

SITE atom parameters, fractional coordinates

PARTN	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	weighting function	density		promolecular density
		potential		promolecular potential
	inclusion region	border	n	max edge distance (Å) 6
	atom range	contact	n	max atom distance (Å) 6
	partition type	fragment		single atom contribution
		entire		symmetry equiv. contribution
site	1			atom label
	2-4			fractional coordinates x, y, z if fragment partitioning

3.44. **PATSEE**: Search for molecular fragment

FRAG	definition of molecular fragment
site	atom sites in molecular fragment
zmatrix	geometry description matrix
rotate	rotational search parameters
transl	translational search parameters
triple	triple relationship parameters
shift	shift coordinates of following atom sites
spin	rotate coordinates of following atom sites about Cartesian axes

twist twist following atom sites about connected atom site

PATSEE	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	do not list bond distances and angles of fragments	ngeom		
FRAG	1	Use cartesian coordinates on site lines.	cart	
		or		
	1-3	cell lengths in Å a, b, c _a		
	4-6	cell angles in degrees α, β, γ _a		
		or		
	1-	labels of atoms to be extracted from <i>lratom</i> : of the input bdf		
		or		
	1	five membered ring	pent	
		benzene planar conformation	benz	
		benzene boat conformation	boat	
		benzene chair conformation	chair	

Notes:

- a. These are applied to the fractional coordinates of the atom sites that follow and must be entered if the cell differs from that on the bdf.

site	1	atom label		
	2-4	coordinates _a x, y, z		
	5	relative site weight in search (<i>max 1, min 0</i>)		1

Notes:

- a. Assumed fractional unless the preceding **frag** line contains the Cartesian code.

zmatrix	1	n1 sequence number		
	2	atom label		
	3	n2 sequence number		
	4	distance to atom n1		
	5	n3 sequence number		
	6	n3-n2-n1 angle		
	5	n4 sequence number		
	6	n1-n2-n3 n2-n3-n4 torsion angle		

rotate	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	number of rotations applied	ntry	n	<i>auto</i>
	number of solutions saved	nsol	s	5
	intra-mol. vector angle resolution	vres	a	10°
	intra-mol. vector d min/max	vlim q	p q	2. 6.A
	intra-mol. vectors for Rfom (ratio)	vfom	f	0.5
	intra-mol. vectors for pretest (ratio)	vtes	r	0.1
	Rfom acceptance threshold	fomt	t	0.4
	number of refinement searches	nref	m	1000
	random rotation start value	nran	r	5431
	rotation start positions about <i>a b c</i>	seta	a b c	0° 0° 0°
	rotation range about <i>a b c</i>	setr	a b c	<i>auto</i>
	list all intra-molecular vectors	vlis		<i>no</i>
	signal mirror in fragment	mirr		<i>no</i>

transl	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	number of translations applied	ntry	n	<i>auto</i>
	number of solutions saved	nsol	s	5
	inter-mol. vector d minimum	vmin	d	3.4 Å
	inter-mol. vector shift resolution	vres	q	0.25 Å
	Tfom acceptance threshold	fomt	t	0.3
	t3sum acceptance threshold	tprt	f	0.3
	translation origin as x/a y/b z/c ^a	sets	x y z	0 0 0
	translation range as x/a y/b z/c ^a	setr	x y z	<i>auto</i>
	list triplets used in t3sum	tlis		<i>no</i>
	exclude use of t3sum refinement	nots		<i>include</i>
	random translation start value	nran	r	5431

Notes:

- a. Six values may be entered; three for the rotation/translation fragment and optionally three for the translation-only fragment.

triple	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	minimum E value used	emin	e	<i>auto</i>
	minimum A of any triple	amin	a	<i>auto</i>
	max number of E's used	nume	n	100
	max number of triples used	numt	m	100

shift ^a 1-9 rotation matrix elements $r_{11}, r_{21}, r_{31}, \dots, r_{33}$
 10-12 translation vector t_1, t_2, t_3

Notes:

- a. The **shift** transformation matrix is applied to all atom sites in the next input fragment.

spin ^a 1-3 rotation angles (in degrees) about the Cartesian axes x, y, z

Notes:

- a. A positive value is an anticlockwise rotation as the axis comes towards you. The **spin** angles are applied to all atom sites in the next input fragment.

twist	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	minimum torsion angle (deg.)	tmin	a	0
	maximum torsion angle (deg.)	tmax	b	355
	increment torsion angle (deg.)	tinc	c	5
	torsion angle test method	test	i	1

3.45. **PEKPIK**: Search map for peaks

plimit specify peak search parameters

PEKPIK	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	input map file	bdf	e	file extension e
	output peaks as atom lines	npun		no output
	output peaks to bdf	punch		atom lines to pch
		sfil		write peaks to pek
		nfil		no output to pek
	use symmetry in peak search	sym		use all symops
		nsym		no symops
	search for peaks or holes	nhol		peaks only
		hole		holes only

plimit	1	min point density used in peak search ^a <i>The following defaults apply, according to FOURR map type.</i> COEF=1.0 PATT=10.0 EPAT=5.0 VECT=2.0 EMAP=2.0 FOBS=1.0 FDIF=0.25 FCAL=1.0	
	2	min separation of resolved peaks in Å	0.75
	3	max peaks output	50 or 2*(non-H atoms)

Notes:

- a. Automatically adjusted after searching first layer.

3.46. *PIG*: Display and manipulate model

peaks	input peak site parameters
radii	reset default atom radii
preset	reset default plot parameters
complx	define molecular complex
ortctl	ortep control line prefix

PIG	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	Displayed sites	<u>clus</u>		optimally clustered
		<u>input</u>		coordinates as input ^a
	add symmetry sites ^b	<u>sext</u>	<u>con</u>	extend connected input list
		<u>sext</u>	<u>all</u>	extend all site symmetry elements
		<u>nose</u>		do not add to input
	new built H atoms	<u>no link</u>		don't add link lines to ADDATM
	cluster packing	<u>pack</u>	<u>n</u>	n=0 no packing <i>n=1</i>
				n=1 smallest combined COG.
				n=2 closest to COG of largest cluster
				n=3 nearest contact to largest cluster
	create ORTEP diagram	<u>ort</u>		from screen contents at PIG
				exit
	BUNYIP data	<u>buny</u>		display symmetry from .bun
	CRISP data	<u>msol</u>		show structural solutions .sol
	cartesian coordinates	<u>labx</u>		full atom labels in .xyz file
	cartesian coordinates	<u>rotx</u>		rotated to current view .xyz file
	graphics display	<u>dbuf</u>		double buffering

Notes:

- a. input atom sites mode is forced for more than 500 atoms (clustering is switched off)
- b. The default is sext if Z (from the STARTX celmo1 line) is less than the number of symmetry positions. Otherwise nose is the default.

peaks	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	max number peaks used	<u>numb</u>	<u>n</u>	input <u>n</u> peaks
	peak height min/max	<u>plim</u>	<u>p q</u>	0 1000
	peak bond radius	<u>prad</u>	<u>r</u>	<i>max atom types</i>
	peak proximity cutoff	<u>plap</u>		in Å 0.5

preset	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	list Cartesian coordinates	cart		
	max bonds per atom	bond	n	10
	set atom type colour	colr	a n	a=atomtype; n=colour
		noco		no colour
	molecules in cell	full		molecules complete
		mcut		mols cut at cell edge
	extent of colour shading	shad	d	between 0.0 - 1.0
				0.5
radii	1	atom type symbol (must match atom type in bdf)		
	2	atom type radius in Å (supersedes bdf value)		
	3-4	same as for fields 1-2. etc		
complx	1-2	labels of two sites in separate molecules that should be as close as possible to define a molecular complex		
	3-	additional label pairs if needed		
ortctl ^a	1-	any valid ORTEP instruction line to add to or replace those automatically generated by PIG		

Notes:

- a. Up to 15 a maximum of 15 ortctl instructions can be used.

3.47. *PLOTX*: Interface to graphics devices

params	plot parameters		
color	plot colours		
frames	indicates which frames are to be plotted		
PLOTX	1	graphical output options ^a	
		screen display using X11	x11lib
		output A3 HPGL plot	hpg1 create .hp file ^b
		output A4 HPGL plot	hpgla4 create .hp file ^b
		output landscape PostScript file	post1 create .ps file
		output portrait PostScript file	postp create .ps file
		color landscape PostScript file	postlc create .ps file
		color portrait PostScript file	postpc create .ps file
		color landscape XFIG file	xfig1 create .fig file
		color portrait XFIG file	xfigp create .fig file
		output plot as Mathematica file	mat1 create .mat file
		screen display for os2	os2pm
	2	input plot file extension	
		generated by ORTEP	ort
		generated by ABSORB	abs
		generated by NEWMAN	nwm
		generated by CONTRS	con
		generated by PREVUE	pre
		generated by GIP	gip
		generated by POWGEN	ppp
	3 -	do not blank label backgrounds	nb1k
		use for pre xtal3.6 plot (.pre etc) files	old

retrace outline of filled curves	ret	
set postscript paper size	A3	
set postscript paper size	A4	
set postscript paper size	A5	
set postscript paper size US	let	
letter		
set postscript paper size US	leg	
legal		
set postscript paper size US	exe	
executive		
print to a specific path/filename	file	path 256 chars maximum

c

Notes:

- default filenames can be overridden with the **file path** option
- If the output device is a pen plotter then a **frames** line may be needed.
- Overrides the default extension. Should be used as the final option on the PLOTX command line as the filename may contain embedded white space as well as directory/file-system indicators.

params	1	x output plot dimension ₁		<i>defined at implementation</i>
	2	y output plot dimension ₁		<i>defined at implementation</i>
	3	plot scale (input to output)		<i>automatic scaling</i>
	4	x plot centre (input units)		<i>automatic centring</i>
	5	y plot centre (input units)		<i>automatic centring</i>
	6	plot border lines		<i>no</i>
		yes plot a border	bord	
	7	plot resolution _a		<i>defined at implementation</i>

Notes:

- The units of these parameters will depend on device and the method of installation. For example, if the plot dimensions are in inches, the plot resolution will be the smallest distance that can be resolved for this device in inches.

color _a colour/line style option followed by colour/line style indices

atom type colours; do not fill	nofill	n1...n10
atoms or bonds		
atom type colours; fill atoms or bonds	fill	n1...n10
atom type shades; fill atoms or bonds with black and shading	shade	n1...n10
instead of colour: 1, hollow; 2, solid; 3,pattern; 4, horizontal hatch; 5, vertical hatch; 6, horizontal and verticalhatch; 7, lower-left to upper-right hatch; 8, upper-left to lower-right hatch;9, lower-left to upper-right and upper-left to lower-right hatch		
atom outline line styles;draw atom lines in black and white lines with the following styles: 1, solid;2, dashed; 3, dotted; 4, dashed-dotted; 5, double	atlin	n1...n10
bond colour	bond	m
character colour	char	m
number colour	numb	m
contour colours	contur	n1...n5

one-colour plot

nocol

Notes:

a. the default colour indices are as follows:

nofill/fill/shade 2 3 4 5 6 7 2 3 4 5

atlin 1 1 1 1 1 1 1 1 1 1

bond 1 char 2 numb 3

contur 2 3 4 5 6

frames_a 1- frame sequence numbers to be displayed

Notes:

a. Frame numbers start at 1 and must be entered in increasing order. For example, the first contour map output by CONTRS will be frame 1 and so on. If the **frames** line is not entered all frames will be displayed. If a **frames** line is entered only the specified frames will be displayed.

3.48. *POWGEN*: Powder pattern graphics

```

-----PATTERN SPECIFIC -----
limits      set powder pattern limits and step size
wavlen      set wavelength and polarization ratio
zero        set zero point corrections
backgr      specify parameters for background function
peakf       define a learned (experimental) peak shape function
peaksh      enter experimental peak shape values
excltt      exclude zones of the powder pattern
-----PHASE SPECIFIC -----
phase       specify id and scale factor of a new phase
peak        select peak shape function for this phase
hwidth      specify halfwidth parameters
pasymm      specify asymmetry parameters
orient      specify preferred orientation parameters
select      select subset of atoms in bdf
-----PAGE LAYOUT -----
*layout     Set page size orientation and margins
-----BLOCK SPECIFIC -----
block       Set block position on multi graph page
blkbrd      Set block border attributes
graph       Set style of graph in this block
*exclr      Graphical treatment of exclusion regions
hklbar      Marker attributes for reflection positions
axis        Set attributes for axes
axslab      Label axes for graph
*tickmk     Control tick mark quantities
grid        Control background grid lines
gtitle      Control attributes of title text
titext      Set Title for this graph block
*legend     Control attributes of the graph legend
trace       Set attributes of traces to display
trctxt      set trace labels for this graph block

```

POWGEN	Option	Code	Arg	Action	Default
	graphical interface mode	ng		no graphical interface	yes
	input RIETVD powder file	pot		reads .pot file	.pot
	input ADDPAT powder file	pat		reads .pat file	.pot
	auxilliary powder file	ap	xxx	reads auxilliary powder file	none

apply dispersion in calc	nd		No application dispersion	yes
peak span in halfwidths	ps	n	reflection contribution width	3;n=12
file adjusted params	pc		create .pch file(s)	no
Points to skip in powder	sk	n	step size of n powder points	1
relative interface dimension a	sc	n	Set window to n% of screen	0.9
Set colormap parameters	cm	b s	0<brightness<1,0<saturation<1.5	1.
arrangement of multiple graphs	ho		Horizontally optimised	li
	vo		vertically optimised	
	li		literal placement	
	as		as specified	

Notes:

- a. See also the xtalrc file in the XTALHOME directory. It can be copied to the \$HOME/.xtalrc file and modified on a per user basis.

3.48.1. PATTERN SPECIFIC CONTROL LINES

limits	1	minimum two theta of pattern data processed	
	2	maximum two theta of pattern data processed	
	3	two theta step size of pattern data processed	
wavlen	1	Wavelength alpha1	<i>1.540598</i>
	2	wavelength alpha2	<i>0.0</i>
	3	alpha2/alpha1 intensity ratio	<i>0.0</i>
	4	polarisation ratio	<i>.915</i>
	5	radiation type (xray ,electron, neutron or synchr)	<i>xray</i>
	6	fixed divergence slit (0/1=fix/var)	<i>fixed</i>
	7	data lp corrected (0/1=no/yes)	<i>no</i>
zero	1-2	two theta zero correction factors z1 and z2in the correction expression $2\theta(\text{corrected}) = 2\theta(\text{measured}) + z1 + z2 * 2\theta$	
backgr	1-5	polynomial coefficients b(1) to b(5) to describe background as the expression $y_{bg} = b1 + b2 * 2t + b3 * (2t) ** 2 + \dots$ i.e. $y_{bg} = \sum(j) b(j) * (2\theta) ** (j-1)$ (values included in pattern file will be ignored)	
peakf	1	identification number of learned peak shape function to follow	<i>1</i>
peaksh	1	ri (values for the learned peak shape function)	
	2	fsi	
	3	fai	
	4	dfsi	
	5	dasi	
excltt	1	minimum two theta value to be excluded from refinement: zone 1	
	2	maximum two theta value to be excluded from refinement: zone 1	
	3-4	min/max two theta for exclusion zone 2	
	5-	min/max pairs for up to 10 exclusion zones	

3.48.2. PHASE SPECIFIC CONTROL LINES (one set per phase)

phase	1	compound ID of phase (to identify archive files)	<i>current compid</i>
	2	scale factor for this phase	<i>1.0</i>
peak	1	one of the following to define the peak shape v for pseudo-Voigt (x = variable) g for Gaussian (x=1) l for Lorentzian (x=0) n=number of learned peak shape function (defined with peakf)	<i>v</i>
	2	peak span value (in halfwidths) for this shape function	<i>12.</i>
	3	mixing parameter z in the pseudo-Voigt function: $f = z * \text{gaussian part} + (1-z) * \text{lorentzian part}$	<i>1.0</i>
hwidth	1-3	halfwidth parameters u,v,w used in the expression $\text{half width} = 0.5 * \sqrt{u + v * \tan(\theta) + w * (\tan(\theta)) ** 2}$	
pasymm	1-2	peak asymmetry correction parameters x and y used in the expression $\text{peak asymm} = x + y * 2\theta$	
orient	1	preferred orientation factor	
	2-4	preferred orientation vector (U,V,W)	
select	1-	atom label *or* atom type symbol *or* atom sequence (labels separated by a '/')	

3.48.3. PAGE DESCRIPTION CONTROL LINES (one set per graph)

layout ^a Specify page size, orientation and margins and choose scale units

Notes:

- a. This command line option is currently not supported

3.48.4. GRAPH SPECIFIC CONTROL LINES (one set per graph)

block ^{a b}	1	number 1 ...16	
	2	nodisp?	
	3	block x grid position (on 9x9)	<i>5</i>
	4	block y grid position (on 9x9)	<i>5</i>
	5	graph x origin position	
	6	graph y origin position	
	7	block width	
	8	block height	
	9	block aspect ratio	

Notes:

- a. should be the first line of a block specific instructions
b. If the number is absent on the first block line, all control lines up till the next block line are applied to all blocks

blkbrd 1 block border colour
2 block border line width

	3	block border line style		
	4	dash line style dash length		
	5	dash line style break length		
graph	1	specify type of graph for this block _a	<u>lin</u>	linear
			log	logarithmic
			sqrt	square root
	2	x origin in block		
	3	y origin in block		
	4	width of graph		
	5	height of graph		
	6	aspect ratio of graph		
	7	boxed		

Notes:

- a. only `lin` is supported

exclr _a	1	phaseid
	2	line
	3	artificial value to set y in this exclusion region

Notes:

- a. This command line option is currently not supported

hklbar _a	1	display hkl reflection position markers
	2-3	min,max 2theta of markers to display
	4-5	min,max intensity of markers to display

Notes:

- a. This command line option is currently not supported but is available from within the graphics interface

axis	1	x or y
	2-3	min, max values to plot for graph
	4	axstop/axisBOT
	5	axlabelsize
	6	axlabxpos
	7	axlabypos
	8	axlabangl
	9	ticlab top/bot/tab
	10	decimal places in text
	11	text centered on digit
	12	displacement of label by x
	13	displacement of label by y
	14	ticlabsize

axslab	1	x (abscissa) or y (ordinate)
	2	label for ordinate or abscissa

tickmk _a	1	x or y
	2	size of tick
	3	length
	4	line width
	5	smallest spacing

Notes:

- a. This command line option is currently not supported

grid	1	display x axis grid lines yes/no	yes
	2	display y axis grid lines yes/no	yes
	3	X axis frac	

	4	Y axis frac	
	5	Num horizontal grid lines	
	6	NUM vertical grid lines	
	7	colour	
	8	line style	
gtitle	1	x origin of graph title	
	2	y origin of graph title	
	3	angle of graph title text	
	4	size of graph title text	
	5	colour of graph title text	
titext	1	The graph title text	
legend ^a	1	display yes/no	<i>yes</i>
	2	boxed	
	3	shadowed	
	4	x orig of legend box	
	5	y orig of legend box	
	6	width of legend box	
	7	height of legend box	
	8	aspect ratio of legend box	
	9	text size of legend	

Notes:

- a. This command line option is currently not supported

trace	1	trace identifier ^a	
	2	trace color	
	3	x displacement of trace on graph	
	4	y displacement of trace on graph	
	5	auxilliary trace scale factor, for plotting	
	6	divide trace intensity by sigma	
	7	line style	
	7	line dash length	
	8	line break length	
	9	marker style	
	10	apply artificial level for low angle null data (yes/no)	
	11	y value to set low angle null data	
	12	apply artificial level for high angle null data (yes/no)	
	13	y value to set high angle null data	
	14	Y value to setfor intermediate null data	

Notes:

- a. valid trace identifiers are **TTH** **OBS** **SIG** **BAK** **CAL** **DEL** and for up to ten phases **PH1** **PH2** **PH3** **PH4** **PH5** **PH6** **PH7** **PH8** **PH9** **PH0**

trctxt	1	trace identifier (see above)	
	2	new label for trace	

3.49. **PREAMS**: Interactive crystal shape manipulation

diff ^a	diffractometer description
orient ^a	crystal orientation
orface ^b	microscope orientation

facept _b	face described by three points
face _b	face described by χ , ϕ , and distance
faceom _b	face described by χ , ϕ , ω and distance
faceml _b	face described by Miller indices and distance

Notes:

- diff** and **orient** are mandatory.
- Only one of the methods **orface-facept**, **face** or **faceml** may be used to specify the faces of the crystal.

PREABS	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	double buffered graphics _a	dbuf		
	precision scale factor _b	pscal	q	scale distance

Notes:

- On some screens and for large window sizes, copying the back buffer to the visible window adds a significant time lag which detracts from general performance. On other displays, particularly linux PCs this option can help quite a bit.
- This option is used to avoid rounding errors for small crystals.

diff _a	1	sense of rotation of the 2θ circle	<u>c</u>	clockwise	
			a	anti-clockwise	
	2	sense of rotation of the χ circle	<u>c</u>	clockwise	
			a	anti-clockwise	
	3	sense of rotation of the ϕ circle	<u>c</u>	clockwise	
			a	anti-clockwise	
	4	sense of rotation of the ω circle	<u>c</u>	clockwise	
			a	anti-clockwise	
	5	sense of rotation of the ψ circle	<u>c</u>	clockwise	
			a	anti-clockwise	
	6	value of ϕ at $\chi = 90$ degrees			0
	7	minimum possible ω ($-180 < \omega < 180$)			-48
	8	maximum possible ω ($-180 < \omega < 180$)			58

Notes:

- See the Diffractometer Description in the **ABSORB** for the method to be used in determining the sense of rotation for each diffractometer circle.

orient	1-3	hkl indices of first orienting reflection
	4	angle χ of first orienting reflection _a
	5	angle ϕ of first orienting reflection
	6-8	hkl indices of second orienting reflection
	9	angle χ of second orienting reflection
	10	angle ϕ of second orienting reflection

Notes:

- Angles in the bisecting mode. (For CAD see Ch. II of the CAD User's Manual.)

orface _a	1-3	x, y, z coordinates of first point on orienting face
	4-6	x, y, z coordinates of second point on orienting face
	7-9	x, y, z coordinates of third point on orienting face

Notes:

- If the **orface-facept** method is used for entering face data, two **orface** lines must be used. The first contains the coordinates of three points on a plane parallel to the first orienting reflection given on the **orient** line. The second **orface** line contains the coordinates of three points on a plane parallel to the second orienting reflection given on the **orient** line.

facept	1-3	x, y, z coordinates of first point on bounding face
	4-6	x, y, z coordinates of second point on bounding face
	7-9	x, y, z coordinates of third point on bounding face

face	1	χ for bounding face (<i>angles in the bisecting mode</i>)
	2	ϕ for bounding face
	3	distance to origin from bounding face in mm
faceom	1	χ for bounding face (<i>angles in the bisectingmode</i>)
	2	ϕ for bounding face
	3	ω for bounding face
	4	distance to origin from bounding face in mm
faceml	1-3	h, k, l indices of bounding face
	4	distance to origin from bounding face in mm

3.50. **PREPUB**: Pre-publication tests on CIF structural data.

PREPUB 1 - no additional parameters

3.51. **PREVUE**: Modify text in plots

PREVUE	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
1	input plot file extension	ort	generated by ORTEP	ort
		abs	generated by ABSORB	
		nwm	generated by NEWMAN	
		con	generated by CONTRS	
		gip	generated by GIP	
		ppp	generated by POWGEN	
2	use pre xtal3.6 plot (.abs etc) files	old		

3.52. **REFCAL**: Process reflection data

setscl set intensity scale factors
discon definition of discontinuities in scale factors

REFCAL	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>	
	instability options 0 o 3 user-fixed instability instability calculation	inst	o	calculate coefficients	0
		inst	4 m b	user-specified coefficients	
		aftr		after scaling references	
		befr		before scaling reference	
		appl		apply scales	
	scale factor use	naps		do not apply	
	range in scale smoothing	sclm	n		5
	"observed" threshold	obst	n	rcode= 2 if $ F_o < n \sigma F_o$	99°
	output data type	rawi		raw intensity	
		neti		nett intensity	
		fsqr		$ F_{obs} _2$	

		frel		F _{obs}	
	dataset number	dset	n	1	
	interpolated form factors	noff		do not insert	
		ffac		insert in reflection record	
	systematically absent hkl	sabs		include on output bdf	
		rsab		do not output to bdf	
	fate of reference	incl		include on output bdf	
	reflections	excl		exclude from output bdf	
	listing data	slis		list data for 10 reflections	
		list	n	list n reflections	<i>all</i>
setscl	1			scale factor to supersede value for the elapsed time in 2	
	2			elapsed time of the scale factor which is superseded by the value in 1	
discon	1			elapsed times of the scale factor sets which mark discontinuities	

3.53. **REFM90**: Convert SCFS-90 file to/from a BDF

REFM90	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
1	reformat mode	xtal		convert archive bdf to SCFS file
		scfs		convert SCFS file to archive bdf
2		pack		format of lines in the HKL
		fobs		PACK or
		fsqr		HKL section of the SCFS file
		int		produced in xtal reformat mode
3	SCFS sequence checking	calc		enabled
		chek		disabled
4	data set number	noch		output in xtal mode
				<i>1</i>

3.54. **REGFE**: Analyse errors

atrad atomic radii by atom type
plane plane names and parameters
line line names and parameters
functn define distance and angle calculations

REGFE	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	atom and plane output	lst		list
		no1		do not list
	covariance matrix	cov		use covariance matrix
		var		use variance terms only
	symops to atoms	sym		apply symops
		nsy		no

correlation matrix	<u>n</u> l <u>r</u>		do not list
	l <u>c</u> r		list ^a
use covariance cell terms	<u>c</u> v <u>a</u>		cell variance terms only
	<u>c</u> c <u>o</u>		cell covariance terms
bond length calculation	<u>l</u> e <u>n</u>		yes
	n <u>l</u> e		no
bond angle calculation	<u>a</u> n <u>g</u>		yes
	n <u>a</u> n		no
dihedral angle calculation	<u>d</u> i <u>h</u>		no
	d <u>i</u> h		yes
number of cells searched	<u>s</u> r	<u>n</u>	1 for 1 unit cell 2 for 27 cells 3 for 125 cells
			2
variance-covariance matrix	<u>n</u> l <u>v</u>		do not list matrix
	l <u>v</u> c		list matrix ^a
tables for publication	<u>t</u> a b		

Notes:

- a. When a listing of the variance-covariance matrix or the correlation matrix is required, set print priority to 4 by entering **reset psta 4**.

atrad	1	atom type symbol	
		use all atom types in the file ^a	a11
	2	maximum radius to be used in identifying contacts	1.25
	3	maximum radius to be used in identifying bond	0.9
	4	minimum radius to be used in identifying bonds	0.015

Notes:

- a. The mnemonic **a11** cannot be used when REGFE is run as a stand-alone program. In this mode each atom type must have an **atrad** line to establish the scattering factor type.

plane	1	plane ID (8 characters maximum)	
	2	data type	
		coefficients of plane equation in 0 fractional coordinates	
		coefficients of plane equation in 1 orthogonal coordinates	
		three atoms to define a plane	2
	3	coefficient A, or atom label	
	4	coefficient B, or atom label	
	5	coefficient C, or atom label	
	6	coefficient D	
	7-10	standard deviations of coefficient A, B, C and D	
line	1	line ID (8 characters maximum)	
	2	data type	
		coefficients of line equation in 0 fractional coordinates	
		coefficients of line equation in 1 orthogonal coordinates	
		three atoms to define a line	2
	3	coefficient A, or atom label	

4	coefficient B, or atom label
5	coefficient C, or atom label
6	coefficient X(0)
7	coefficient Y(0)
8	coefficient Z(0)

functn	1	Option	Code
		distance between two atoms	dis
		angle defined by three atoms	ang
		dihedral angle defined by four atoms	dih
		angle between two planes	pap
		angle between plane and line	pal
		distance from plane to atom	pda
		distance from line to atom	lda
		angle between two lines	lal
		plane equation (three atoms)	pla
		line equation (two atoms)	lin
		unit cell volume	vol
	2-	atom labels, plane ID's or line ID's for the above function ^a	

Notes:

- a. The plane ID or line ID may be any one from the predefined plane list or line list, or can be a group of atom labels which are adequate to define a plane or line. Different types of functions require different numbers and types of components. The sequence of components is important. For example, function PDA (distance from plane to atom) should be followed by a plane ID and an atom label, but not an atom label and a plane ID. A plane can also be defined by entering three atom labels and a line can be defined by entering two atom labels.

3.55. *REGWT*: Analyse weights

normal ^a	normal weighting scheme
regina ^a	Regina weighting scheme
maxhkl	set limits on h, k, l and s
archiv	addition and deletion of items in logical record <i>lrrefl</i> :
fgrid ²	specification for F_{rel} division of the analysis map
sgrid ^b	specification for s division of the analysis map

Notes:

- a. ^a Indicates only one control line with this symbol may be used.
- b. The **fgrid** and **sgrid** lines are mandatory for weight analysis. They must be the last two control lines and the **fgrid** line must precede the **sgrid** line.

REGWT	Option	Code	Arg	Def
	reflection listing	no1		do not list all reflections
		lst	n	list every n^{th} reflection
	reflection type	rel		use rcode=1 reflns only
		in1		use rcode=1, 2 reflections
	dataset selection	dse	n	use hkl dataset n
	contribution factor	cnt	q	exclude hkl if $\Delta F > q$
	analysis based on	fr1		relative F
		fr2		relative F_2
		irl		relative intensity
	type of run mode	reg		Regina weighting scheme
		nor		normal weighting scheme

	wta	m	weight analysis a	3
scale factor for grid map	wfc	k		1
dispersion correction	nod		no dispersion correction	
	dis		dispersion correction	
scale factor for R-map	rsc	k		100

Notes:

- a. $m=1, 2, 3, 4, 5$ or 6 for analysis of the weighting schemes: unit weight, $1/\sigma_2$, LSWT, LSW1, LSW2 or LSW3.

normal a	1	weighting scheme number (0 to 9)	
	2	mnemonic name of coefficients (a to r)	
	3	coefficient of the normal weighting function	0
	4-5	same as 2 and 3	

Notes:

- a. Be careful. Some values of the coefficients cannot be zero during the weight estimation. Read the *REGWT* section in the Reference Manual first.

regina	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	analysis map mode	indi		individual reflection mode
		grid		grid point mode
	power of the weight a	pow	n	2
	freedom factor of weight	fac	m	1
	var.-covariance correction	avc		apply
		noavc		do no apply
	grid weight	ngw		do not apply
		apw		apply

Notes:

- a. The highest power term in the weight function must be ≥ 0 and < 3 .

maxhkl	1-3	max h , k , l used in the weighting process	<i>from bdf</i>
	4-5	min and max values of $\sin \theta/\lambda$	<i>from bdf</i>

archiv	1-	ID numbers a of bdf items to be added/deleted from logical record <i>lrrefl</i> :
---------------	----	---

Notes:

- a. The only items which can be added or deleted are LSW1(n901), LSW2(n902) and LSW3(n903), where n is the data set number. **archiv** here is different from the one used in other programs. The previous weights located at LSWT, for instance, will be saved at location LSW1 if n901 is present in the **archiv** line. When -n901 is specified, the weight at LSW1 will be deleted from logical record *lrrefl* in the bdf. In both cases the old weight at LSWT(n900) will be replaced by the newly calculated weight.

fgrid a	1-	summation grid points along F_{rel} axis
----------------	----	--

Notes:

- a. More than one line of grid points for F_{rel} is allowed. The maximum value for a grid point for F_{rel} is 9999 and the value of the grid points must be in ascending order. If this line is used, at least one grid point must be given.

sgrid a	1-	summation grid points along the $\sin \theta/\lambda$ axis
----------------	----	--

Notes:

- a. More than one line of grid points for s is allowed. The maximum value for a grid point for s is 2.00 and the value of the grid points must be in ascending order. If this line is used, at least one grid point must be given.

3.56. *RFOURR*: Reverse Fourier transform

idnums	input and output structure factor coefficients
---------------	--

maxhk1 limits on h, k, l and $\sin\theta/\lambda$

RFOURR	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	data set number	ds	n	1
	print control	pr	n	<i>all</i>
	sort order s/m/f	hk1 h1k kh1 k1h l1k lkh		<i>from bdf</i>
	reflection signs	ph pk pl	positive h only positive k only positive l only	<i>from bdf</i> <i>from bdf</i> <i>from bdf</i>
	A and B threshold	th	t	phase if A or B > t 0.005
	F _{cal} threshold	ft	q	phase if F _{cal} > q F _o 0
maxhk1	1-3	max value h , k and l to be phased		<i>from bdf</i>
	4	min value of $\sin\theta/\lambda$ to be phased		<i>from bdf</i>
	5	max value of $\sin\theta/\lambda$ to be phased		<i>from bdf</i>
idnums ^a	1	bdf idnum of <i>input</i> Frel coefficient		701 or 304
	2	bdf idnum of <i>output</i> Fcal coefficient		800
	3	bdf idnum of <i>output</i> Acal coefficient		801
	4	bdf idnum of <i>output</i> Bcal coefficient		802
	5	bdf idnum of <i>output</i> phase coefficient		700

Notes:

- a. Note that the data set multiplier is *not* appended to the idnum's.

3.57. **RIGBOD**: Generate a Rigid Group

RIGID	start of a rigid body group
atomsy	coordinates of symmetry related real atoms
delete	delete real atoms in fitting procedure
insert	insert real atoms from bdf
cellid ^a	cell constants of ideal atoms
atomor ^b	orthogonal coordinates of ideal atoms
atomfr ^b	fractional coordinates of ideal atoms
atname	atom names for a second set of atoms

Notes:

- a. Must be present to establish the conversion factor for the fractional coordinates given for the ideal atoms on **atomfr** lines.
b. Indicates one or the other is optional.

RIGBOD	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	atom lines output to pch	p1 p2		fractional coordinates orthogonal coordinates
	data set number	ds	n	1
RIGID	1	isotropic thermal parameter U for all atoms of this group		
	2	population parameter for all atoms of this group		
	3	multiplicity for all atoms of this group		
atomsy	1	atom label		
	2-4	fractional coordinates x, y, z		

delete	1	atom label
	2-	as in 1, or use separate lines
insert	1	atom label
	2-	as in 1, or use separate lines
cellid	1-3	a, b, c cell lengths
	4-6	α, β, γ cell angles
atomor	1	atom label
	2-4	orthogonal coordinates x, y, z
atomfr	1	atom label
	2-4	fractional coordinates x, y, z
atname	1	atom label
	2-	as in 1 or use separate atname lines

3.58. *RMAP*: Search for translated fragment

scale	F_{rel} scale factors
maxhkl	limits to hkl and $\sin\theta/\lambda$
select	select a subset of atoms in the bdf for F_{cal}
fixatm	set atoms to be fixed and translated
bounds	set parameters of unit cell to be searched

RMAP	<i>Option</i>	<i>Code</i>	<i>Arg</i>		<i>Def</i>
	Rmap function weight	wei	w	0 for C or 1 for R	1
	thermal parameter type	ove		overall isotropic	
		iso		individual isotropic	
	control on rcode	rej	n	reject rcode = n	1
	printing of hkl & F_{cal}	lis	q	list if $F > \mathbf{q}$	999
		lis	-n	list first n hkl	0
	printing of RMAP	map		print map	<i>no</i>
	set F_{obs} threshold	thr	t	reject $F_{\text{obs}} < \mathbf{t}$	0
	selection of hkl	sel	n	if n =1 all hkl rejected	<i>all</i>
				if n =2 alternate hkl used	
	selection of small hkl	sth	m	applies to thr rejects	1
	source of trig functions	nos		use library sin/cos	
		sct		use sin/cos table	
	data set number	dse	n	use dataset n	1

scale	1	F_{rel} scale factor	
	2	scale group	1
	3	data set name	parent
maxhkl	1-3	max h , k , l	
	4	min $\sin\theta/\lambda$	0
	5	max $\sin\theta/\lambda$	2

select ^a 1 atom labels of atoms selected from the bdf

Notes:

a. See **fixatm** line below.

fixatm ^a	1	number of atoms to be treated as fixed	0
----------------------------	---	--	---

2 number of atoms to be used in translation search

Notes:

a. Note that atoms must be loaded into the bdf so that the first selected number is the fixed sub-group and the second the translated subset.

bounds	1	first x fractional coordinate for search	0
	2	last x fractional coordinate for search	1
	3	δ x fractional coordinate for search	0.3/cell length
	4-6	as above for y	
	7-9	as above for z	

3.59. *RSCAN*: Analyse R-factors

maxhkl limits for h, k, l and $\sin\theta/\lambda$
scale scale groups and scale factors

RSCAN	1	<i>max1</i> maximum for measured F range 1	19
	2	<i>int1</i> interval for measured F range 1	1
	3	<i>max2</i> maximum for measured F range 2	500
	4	<i>int2</i> interval for measured F range 2	10
	5	maximum for $(\sin\theta/\lambda)_2$ range	4
	6	interval size for $(\sin\theta/\lambda)_2$ range	0.02
	7	maximum for measured I range	1000000
	8	factor for computing measured I interval size	2
	9	data set number	1
maxhkl	1-3	max h , k , l to be used in summations	from bdf
	4	min $\sin\theta/\lambda$ to be used in summations	0
	5	max $\sin\theta/\lambda$ to be used in summations	from bdf
scale	1	F_{rel} scale factor	
	2	scale group	1

Multiple *scale* lines may be used.

3.60. *SCATOM*: Calculate pseudo-atom form factor. **NEW**

SCATOM	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
		iso		isotropic U only
		ani		anisotropic Uij only
		ove		overall isotropic, all atoms

3.61. *SHAPE*: Patterson deconvolution for structure solution

smf calculates symmetry minimum function
imf calculates image seeking minimum function

SHAPE	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
--------------	---------------	-------------	------------	------------

smf - no additional parameters

imf 1 - 3 x,y,z fractional co-ordinates of pivot position

3.62. *SHELIN*: Enter SHELX file

SHELIN ^a <i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
shelx control instructions	<u>new</u>		external file
	<u>old</u>		in line within <i>compid.dat</i>
population treatment	<u>pop</u>		force atom pop to 1.0
shelx source file(s) basename ^b	<u>file</u>	<u>path</u>	(256 chars max) instructions from <i>compid.ins</i>
			reflections from <i>compid.hkl</i>

Notes:

- See also the program *OUTSRC* for exporting an Xtal archive back to Shelx input format.
- Should be used as the last option on the **SHELIN** line as embedded blanks are permissible, as well as the DOS disk identifier ":" and directory separators.

The following SHELX lines are translated into XTAL commands ^a

TITL, CELL, LATT, SYMM, SFAC, UNIT	for STARTX
HKLF	for DIFDAT/ADDREF
FVAR	for ADDATM

Notes:

- The recognised data are: the standard SHELX hkl lines (two formats) following **HKLF**, and the standard atom lines following **FVAR**. The **CELL** to **UNIT** commands must always be input, and the **FVAR** command and atom lines, if entered, must be the last in the **SHELIN** sequence. All other SHELX lines will be ignored.

3.63. *SIMPEL*: Symbolic-addition phasing

invar structure invariant options

start selection of starting phases options

phase defining starting phase

print print options of different stages

sybad symbolic addition procedure options

setfom define figure of merit parameters

psets specify phase sets to be output

SIMPEL	1	number of generators eligible for starting set	25% of total
	2	number of generators eligible for divergence	50% of total
	3	number of generators eligible for phasing	100% of total
	4	minimum E value of input generators	from <i>bdf.inv</i>
	5	data set number of E 's	1
	6	number of phase sets output on output archive <i>bdf</i>	4

start	1	control invariants used in con/divergence process	
		use triplets only in starting process	<u>trip</u>

		use quartets only in starting process	quar	
		use triplets and quartets in starting process	all	
2		probability term used in convergence process		
		use (1) described in Step 1	<u>1</u>	
		use (2) described in Step 1	2	
3		threshold ALTHR to accept one invariant		<i>10th-largest</i> α
4		factor used to reduce [alpha] using σ [alpha];		1
5		number of symbols to be assigned in convergence process		<i>if no phase lines entered</i> 4 <i>otherwise</i> 0
invar	1	invariant types to be used in symbolic addition		
		use triplets only	<u>trip</u>	
		use quartets only	quar	
		use triplets and quartets	all	
2		minimum A of triplets used		<i>from bdf</i> <i>inv</i>
3		minimum B of quartets used		<i>from bdf</i> <i>inv</i>
4		maximum triplets allowed per generator		100
5		maximum quartets allowed per generator		100
6		low crossvector sum limit for quartets		<i>25% of</i> <i>max</i> <i>XV-sum (up</i> <i>to max of</i> <i>2.5)</i>
7		high cross vector sum limit for quartets		<i>75% of</i> <i>max</i> <i>XV-sum (to</i> <i>min of 4.0)</i>
8		ψ phase correction applied to negative quartets in degrees		<i>if no fragment ψ</i> 180 <i>otherwise</i> 0
print ^a	1-	print invariant statistics and sums	invar	
		print converge order of generators	converg	
		print extension of starting set into converge set	extend	
		print extension of starting set into diverge set	diverg	
		print symbolic addition extension process	sympad	

print application of accepted symbols to all |E|'s **correl**
 print figure-of-merit details **fomtest**
 print phase sets output to bdf **bdfout**

Notes:

- a. Entering no options implies that **all** print options are invoked.

symbad	1	probability factor used in symbolic addition	
		use sum of weighted α 's (see Step 3.)	wa
		use sum of α 's and invariant count	an
	2	acceptance procedure if multiple symbols generated	
	accept most probable indication	mult	
	reject all but single symbol indication	sing	
	3	no. of generators added in blocks to the symbolic process	100
	4	acceptance threshold of weight for wa option	0.3
setfom	1	number of top phase sets applied in FOM tests ^a	16
	2	optimum value for QFOM	0.75
	3	weight of QFOM in CFOM and AMOS	1
	4	optimum value of RFOM	1
	5	weight of RFOM in CFOM and AMOS	1
	6	optimum value of RFAC FOM	0.25
	7	weight of RFAC in CFOM and AMOS	1
	8	optimum value of PSIO FOM	0.75
	9	weight of PSIO in CFOM and AMOS	1
	10	optimum value of NEGQ FOM	60
	11	weight of NEGQ in CFOM and AMOS	1

Notes:

- a. Phase sets are presorted based on the max correlation with the symbolic phases.

phase	1-3	hkl indices	
	4	input phase (in degrees)	<i>assign as new symbolic phase</i>
	5	<i>Phase status option</i>	<i>Code</i>
		assign as origin defining phase	odr
		assign as enantiomorph phase	edr
		assign as symbolic phase ^a	sym
		assign as numeric phase ₁	set
		remove phase from phasing process	nu1

Notes:

- a. The default is **sym** if field 4 is void; **set** is default if field 4 is numeric.

psets 1- phase set numbers to be output to output archive bdf ^a *none*

Notes:

- a. These numbers are selected from a previous run using identical parameters.

3.64. SLANT: Slanted map

DEFPLN	specification of general plane
mapres	specification of grid interval in Å
layout	specification of Fourier map output
print	specification of print control parameters
site	atom positions in fractional coordinates
siteg	atom positions in grid coordinates

SLANT	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	output map to bdf	<u>nofile</u>		no output bdf
		file		output map to bdf .sla
	print slant map	<u>print</u>		print map
		nprint		do not print map
	interpolation type	<u>linear</u>		linear
		cubic		cubic

DEFPLN a	1	input coordinate signal	
		fractional coordinates	frac
		grid coordinates	grid
	2-4	x, y, z coordinates of origin	
		point	
	5-7	as for 2-4 for point(2)	
	8-10	as for 2-4 for point(3)	

Notes:

a. All items on the **defpln** line are mandatory.

mapres	1-3	grid interval width in a, b, c directions in Å/division	<i>auto</i>
layout	1-3	map orientation codes of the a, b, c axes	
		specifies axis as page-to-page direction	layer
		specifies axis as down page direction	down
		specifies axis as across page direction	across
	4-6	number of map grid points in the a, b, c directions	
	7-9	first grid point of map in the a, b, c directions	
	10-12	grid increment of map in the a, b, c directions	
print	1	density scale for map print out	1
	2	upper bound for print suppression	+1
	3	lower bound for print suppression	-1
	4	print columns per map point (max 5) a	4
	5	print lines per map point (max 3) b	2

Notes:

a. If a non-default choice for field 4 or 5 is made BOTH fields must be specified. For field 4 the equation is (lines/point)*(col/inch)/(lines/inch).

b. For field 5 the equation is (col/point)*(lines/inch)/(col/inch).

site	1	atom label
	2-4	fractional coordinates x, y, z

siteg as for **site** except grid coordinates are entered.

3.65. **SORTRF**: Sort/merge reflection data

label information added to logical record *label* of the bdf
merge items on input bdf to be averaged

SORTRF	<i>Option</i>	<i>Code</i>	<i>Arg</i>		<i>Def</i>
	sorting order	ord	hkl	slow/med/fast indices	hkl
	sorting type	sort	n	1 -high→0→+high index 2 as 1 with Friedels adjacent 3 as 1 with h = k = l groups	
	equivalent hkl data	nav clus aver	n	do not average equivalent hkl cluster equivalent hkl 1 standard average 2 Fisher test average	
	rcode threshold	rthr	r	rcode=2 if $r\sigma Y > Y$	r=0
	prior rcode values	ignore		do not use input rcodes	use
	culling for average 1	ncull cull	t	no culling cull Y if $ Y-Y > t \sigma Y$	3
	Friedel independence ^a	pakfr1 sepfr1		pack pairs in single packet ^b keep pairs as separate reflns ^c	
	averaged coefficients ^d	irel f2r1 frel		average I and σI average $ F_2 $ and $\sigma F_2 $ average $ F $ and $\sigma F $	
	print output data	print print nopr	-n n	print n problem hkl's print n hkl do not print	25
	print observations	mob	n	print n 100	
	σ 's for average 2 if hypothesis true	success	n	1 Poisson 2 equivalents 3 maximum	
	σ 's for average 2 if hypothesis is false	unsucce	n	1 Poisson 2 equivalents 3 maximum 4 reject reflection	
	dataset number	dset	n		1

Notes:

- Friedel related data is treated as non-equivalent data if one of these options is entered. Otherwise, Friedel pairs are considered to be symmetrically equivalent.
- Correct option if Friedel data is to be used in CRYLSQ for an **xabs** refinement.
- Correct option if Friedel data is to be used in LSLS for a **twinop** refinement.
- Only the specified coefficient type is merged. Other coefficients on the input bdf are not averaged, unless on a **merge** line. If the specified coefficient is not present on the bdf, the coefficients I, F₂ and F will be used in that order. I is the default.

label 1 character information added to the logical record *lrlabl*: of the bdf

merge 1-30 ident numbers of items input on the bdf to be averaged (max *only selected irel, f2rel or frel*)

3.66. STARTX: Create initial archive file

label	descriptive information
cell	unit cell parameters
cellsd	standard deviations in unit cell parameters
sgname ^a	space group name in explicit-origin <i>Hall</i> notation
spaceg	space group name in Hermann-Mauguin notation
lattice ^a	unit cell lattice type and centricity
symtry ^a	symmetry equivalent positions
celmol	number of molecules in the cell
celcon	contents of unit cell atomic species information
formfx ^b	scattering factor information, tabular
formgn ^b	scattering factor information, expansion coefficients
scale ^c	F _{rel} scale factors for data sets
maxhkl ^c	maximum $\sin\theta/\lambda$, h, k, and l for datasets
datdef ^c	definition of crystal data set (enantiomorph, etc.)
exper ^c	experimental conditions for data sets
crystal	crystal colour and dimensions
orient ^c	crystal orientation for data sets

Notes:

- a. Symmetry information must be entered by using either a **sgname** line or a **lattice** line and/or one or more **symtry** lines.
 b. **CELCON** lines must precede these lines. For a given atom type, *both* **formfx** and a **formgn** lines may *not* be used. However, a **formfx** line for atom type 1 and a **formgn** line for atom type 2 may be used. If neither is entered, stored expansion coefficients will be used by default for atoms specified on the **CELCON** lines (Cromer and Mann, 1968).
 c. These lines, if entered, must follow the **CELCON** lines.

STARTX	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	bdf creation	apr		create an <i>a priori</i> output archive bdf
		upd		output updated from input bdf
	stored form factors	sint	s	$\sin\theta/\lambda$ interpolation int. 0.02
		smax	s	$\sin\theta/\lambda$ maximum 2.0
	radiation type	xray		X-rays
		neut		neutrons
		elec		electrons
		lax		no symmetry checking at all

label	1	character information added to the logical record <i>lrlabl</i> : of the bdf	
cell	1-3	a, b, c cell dimensions (in Å)	
	4-6	α, β, γ cell angles (in degrees)	90
cellsd	1-3	$\sigma a, \sigma b, \sigma c$	0
	4-6	$\sigma\alpha, \sigma\beta, \sigma\gamma$	0
sgname	1-	space group in Hall notation ^a	

Notes:

- a. A table of these notations, together with the corresponding space group as found in the International Tables, Vol. A., is located in the *Symmetry* appendix. (NOTE that separate elements of the notation must be separated by space or underscore characters e.g. p_2_2 not p22.)

spaceg	1-	space group in Hermann-Mauguin notation ^a
---------------	----	--

Notes:

- a. Separate elements of the notation must be separated by space or underscore characters e.g. P 1 21 2 not P1211. Note also that because colons (:) denote specific settings they cannot be used as XTAL comment delimiters on **spaceg** lines.

lattice ^a	1	centricity code	
		centrosymmetric	c
		noncentrosymmetric	n
2		Bravais lattice code	
		primitive	p
		end-centred	a, b, c
		body-centred	i
		rhombohedral	r
3		face-centred	f
		no symmetry checks ^b	ncheck

Notes:

a. If this line is entered then the equivalent positions described by the centricity and Bravais lattice codes *must not* be input as **symtry** lines. This line must NOT be entered if an **sgname** line is used.

b. no symmetry checks will be applied to input **symtry** lines and the sgname code will not be automatically generated and stored.

This option is invoked if special symmetry is to be entered.

symtry ^a 1- equivalent positions entered as algebraic strings

Notes:

a. Symmetry equivalent positions are entered as listed in the International Tables, Vol I or A. If the **lattice** line is entered then the equivalent positions described by the centricity and Bravais lattice codes *must not* be input as **symtry** lines. The definition of an equivalent position is the x, y, and z operations for each coordinate separated by commas. Only one equivalent position may be defined on a **symtry** line. This line must NOT be entered if an **sgname** line is used. Xtal comments (preceded by a colon :) are *not* permitted.

celmol	1	number of molecules in the cell	<i>one molecule per asymmetric unit</i>
celcon ^a	1	atom type symbol (max 8 characters)	
	2	number of atoms of this type in the unit cell	
	3	atomic weight	<i>based on field 1</i>
	4	atomic number	<i>based on field 1</i>
	5	number of electrons in atom type	<i>based on field 1</i>
	6	atomic bond radius (in Å)	<i>based on field 1</i>
	7	atomic contact radius (in Å)	<i>based on field 1</i>
	8	real dispersion factor f' ^b	<i>based on field 1</i>
	9	imaginary dispersion factor f'' ^b	<i>based on field 1</i>
	10	neutron scattering factor (CAUTION ^c) ^d	<i>based on field 1</i>

Notes:

a. One **celcon** line must be entered for each atom type expected in the structure. If neither **formgn** nor **formfx** lines are used, scattering factors will be generated using information on **celcon** lines and tables from Cromer and Mann (1968).

b. The dispersion factors are automatically entered only for MoK and CuK $\alpha 1$ radiations. Actually, as of Xtal-3.7.2 dispersion factors in the wavelength range 0.1Å to 2.89Å can be automatically added based on the calculations of S. Sasaki (1989) KEK report, which you should acknowledge. The radiation wavelength is specified on the **exper** line.

c. If the neutron scattering factor is entered, ALL subsequent calculations will require that neutron data be processed. *Do not* enter unless this is the case.

d. If the command **STARTX neut** is invoked then the neutron scattering length is automatically added to the archive.

formfx ^a	1	atom type symbol (max 8 characters)
	2	$\sin\theta/\lambda$ for scattering factor
	3	scattering factor value
	4	comment about source of scattering factor (surround with '=')

Notes:

a. Not more than 200 **formfx** lines are allowed per atom type. Lines must be in order of increasing $\sin\theta/\lambda$. The first line must have a $\sin\theta/\lambda$ value of zero.

formgn ^a	1	atom type symbol (max 8 characters)
	2-10	Cromer-Mann coefficients a1, b1, a2, b2, a3, b3, a4, b4, c

Notes:

a. One **formgn** line is used per atom type.

scale	1	F _{rel} scale	
	2	scale group number	1
	3	data set number	1
	4-6	same as fields 1-3 for next scale; and so on	
maxhkl	1-3	max h , k , l to be stored on the bdf	
	4-5	min and max $\sin\theta/\lambda$ to be stored on the bdf	
	6	data set number	1
datdef	1	data set number	
	2	data set name (max 12 characters)	parent
exper	1	data set number	1
	2	weighted mean wavelength (in Å)	0.71069 (<i>X rays</i>) 1.0 (<i>neutrons</i>) 0.01 (<i>electrons</i>)
	3	wavelength line 1	
	4	wavelength line 2	
	5	wavelength line 3	
	6	relative weight line 1	
	7	relative weight line 2	
	8	relative weight line 3	
	9	measured density of crystal (mg/mm ³)	
	10	linear absorption coefficient	
	11	temperature of measurement (in Celsius)	
	12	sorting order for hkl as packed integer e.g. 213 is k-slow, l-fast	
	13	pressure of measurement (in GPa)	
	14	radiation type (1=neutron 2=electron 3=MoKa 4=CuKa 5=AgKa 6=synch)	
crystal	1	crystal colour (20 characters max)	
	2	crystal shape (20 characters max)	
	3	Maximum crystal dimension (mm)	
	4	Median crystal dimension (mm)	
	5	Minimum crystal dimension (mm)	
orient	1	data set number	1
	2-10	orientation matrix: UB ₁₁ , UB ₂₁ , UB ₃₁ ... , UB ₃₃	

3.67. **SURFIN**: Render 3D electron density isosurfaces **NEW**

SURFIN	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
---------------	---------------	-------------	------------	------------

3.68. **VUBDF**: Dump archive file

VUBDF	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	bdf device assignment	file	ext	aa1

physical record range	<code>prec</code>	<code>r s</code>	1 10000
logical record range	<code>lrec q</code>	<code>p q</code>	1 <i>en- drecord:</i>
packets per logical record	<code>npak w</code>	<code>v w</code>	1 100
remove format template	<code>xform</code>		format packets separately

3.69. *XTINCT*: Extinction from equivalent reflections

limits restrictions on reflections to be included in refinement

XTINCT	<i>Option</i>	<i>Code</i>	<i>Arg</i>	<i>Def</i>
	apply correction to	<code>frel</code>		F
		<code>f2r1</code>		F _{2rel}
		<code>irel</code>		I _{rel}
	extinction coefficient r*	<code>refx</code>		refine and apply
		<code>appx</code>	<code>r s</code>	apply input r* and σ r*
	treat Friedels as equiv.	<code>eqfr</code>		(non-centrosymmetric data) <i>no</i>
	refine separately for all equiv.s	<code>sall</code>		output to punchfile <i>no</i>
	include hkl if T equal	<code>incl</code>		<i>no</i>
	print reflections	<code>prin</code>	<code>n</code>	25
	apply extinction if r*<0	<code>negx</code>		<i>no</i>
	data set used	<code>dset</code>	<code>n</code>	1
	polarization factor	<code>polf</code>	<code>p</code>	range (0-1) 0.5
	mosaic distribution factor	<code>case</code>	<code>n</code>	2
limits	1	limit range of selected option:		
		F	<code>frel</code>	
		F _{2rel}	<code>f2r1</code>	
		I _{rel}	<code>irel</code>	
		sin θ / λ	<code>stol</code>	
	2	minimum value of field 1		0
	3	maximum value of field 1		<i>Inf</i>

Chapter 4. Reference Manual

This section contains the descriptions of programs in the *Xtal System*. These were prepared by the program authors who are listed at the head of each program. If you experience difficulties with a program, or require further information, you could, as a last resort, attempt to contact the author indicated for this purpose. However, since the source code is available and it is so well documented, that should be your first resource.

4.1. System Control Lines

The line input file is used to control how and when calculations are performed. Three different types of input command lines are recognised in an *Xtal* input file. **System command** lines signal certain nucleus functions; **program initiation** lines initialize specific calculations; and **program data** lines are used to enter data for a calculation.

4.1.1. System command lines

The system command lines are:

title	specify the output print header
compid	set the compound identification code
master	set master file switch and specify filenames
remark	insert comments into the printed output
copybdf	copy a binary file
reset	reset printer parameters
setid	specify a default line identifier
field	fix the format of input lines
order	reorder fields of input line data
finish	end of the input line file

4.1.1.1. title

This line contains a character string to be inserted at the top of each printed page. Only one `title` line is active at a time, but any number may be entered during a sequence of calculations. `STARTX` stores the current `title` image in the archive file record `lrlabl`:

Field 1- character string, columns 8 to 72. Default string is blank.

4.1.1.2. compid

This line is used to set the compound identification code for the data under study. See the *compid* discussion below in 3.

Field 1 an alphanumerical string of up to six characters.

4.1.1.3. master

This line is used to control the use of the master archive file. The default master file condition is set on installation with the *masterfile*: macro. This line enables the user to specify master file support for the current run. A `master` line, if entered, *must* precede the first program initiation line. The following free format controls may be used:

- `no` switch master file support off; no master files are read or written

- **yes** switch master file support on; master files read & written from `aaa`
- **read v** copy a master archive file with extension `v` to the binary file `aa1`
- **write w** copy the latest binary file to a master archive file with extension `w`

4.1.1.4. remark

This line contains a character string to be output to the print file. Unlike the `title` line, it is output only once and is output as soon as it is encountered in the input. An example follows at the end of this section.

Field 1 - character string, columns 8 to 72. Default string is blank.

4.1.1.5. copybdf

This line is used to copy one binary data file (bdf) to another. The syntax of this command is: '`copybdf <ext1> <ext2>`'. The binary file with extension `ext1` is copied to a binary file with extension `ext2`. The extension codes are user choice, and need not be any of the extensions described above. The purpose of this command is enable the user to save or retrieve a binary file during an *Xtal* run. Note, however, that a `copybdf` line can only be used after the `compid` line or an `end` line.

If the codes `a` or `b` are used as extension codes, they will be interpreted as the current input or output archive binary file, respectively. In other words, the extension `aa1` or `aa2` will be substituted where appropriate. Typical uses of this command follow:

```
copybdf b tem      :save current output archive binary file to extension tem
copybdf ort or4    :save ORTEP output - used to save successive ORTEP's
copybdf tem a      :replace current input archive file with extension tem
```

4.1.1.6. reset

This line resets certain *Xtal* input/output functions. The prime use of this command is to redefine the printer characteristics for a given run. This is particularly useful in a networked computing environment. The syntax of the `reset` command consists of a parameter code followed by a numerical parameter.

```
cmax n           max characters per line of printer
lmax n           max lines per page of printer
cpin n           characters per inch across the page
lpin n           lines per inch down the page
psta n           priority status of print lines (normally 3; max 5)
```

4.1.1.7. setid

This line is used to preset the identification code of the lines which follow. This may be used to read non-*Xtal* formatted lines. Once the `setid` has been entered, all succeeding lines, *except the system commands*, are treated as having the identification code specified on the `setid` line, until another `setid` is read. When `setid` is in force, the field 1 of a line starts in column 1 of the input line (this is field 0 when `setid` is not in force).

Field 1 1 character string of up to 6 characters. A blank string resets to normal input.

4.1.1.8. field

This line specifies the format of fields for input lines that follow. The input numbers specify the *rightmost character* of successive fields. The leftmost character of each field is assumed to be rightmost character number of the previous field, +1. The leftmost character of field 1 is character 1. If an identification code is entered on an input line which has been formatted by a preceding `field` line, an `order` line will need to be also specified so that this field can be ignored or properly treated. *All* fields expected as input *must* be specified on the `field` line. A blank `field` line is used to reset to free format input.

<i>Field</i>	1	number of the rightmost character for field 1 of following
	2	number of the rightmost character for field 2 of following input lines
	3-	etc.

4.1.1.9. order

This line specifies the order of fields for input lines that follow. This enables the input fields of either fixed or free format lines to be re-ordered to match the requirements of the inputting program. This facility is particularly useful for reading non-*Xtal* lines in which data must be reordered or ignored to meet *Xtal* program requirements. A blank `order` line resets the order to sequential.

<i>Field</i>	1	new field number for first input field
	2	new field number for second input field
	3-	etc.

4.1.1.10. finish

This line signals the end of all calculations. It is optional since the *end-of-file* will cause the same action. The **finish** line is particularly useful to prevent processing beyond a certain line in the input line file.

4.1.2. Examples of System Commands

The first example shows the simplest use of system command lines.

```
title This line appears at the top of each printed page
remark These lines appear only once in the printout
reset cmax 80 :reset printer line width
master yes read zzz :use the master file compid.zzz
```

The next example is more complicated. It shows how a set of input lines containing the atom information may be input as if they were `atom` lines (these data lines are used by many programs -- see [ADDATM](#)). The blank `setid` and `order` lines are necessary to restore line identification and field ordering to default values.

```
order 2 3 4 1 5 :change input order to x y z label tdp
setid atom :set the input line id to 'atom'
0.1645 0.2295 0.3721 Cu1 0.035
0.3978 0.3342 0.3788 Cu2 0.030
0.4564 0.5000 0.6123 01 0.021
setid :switch off the 'atom' line id
order :switch off the changed input order
```

4.1.3. Program initiation lines

A 'program initiation' line is used to initiate an *Xtal* calculation. These lines contain the program identifier code which is used to identify each program in the *Xtal* documentation and in the nucleus routine AA00. For example, a line containing **FC** (as characters 1-2) initiates the structure factor calculation. The program initiation line may also contain input parameters. Any additional input data is entered on separate 'program data' lines. These are described below.

4.1.4. Program data lines

A 'program data' line is used to enter data under the control of a specific program. Each data line starts with a code which uniquely identifies its function. This code is used by the program to identify the data. The order that data lines are entered may be arbitrary or it may be fixed. The entry rules for each data line are given in the *Program controls* section. If a program data line is mandatory its name is capitalised, whereas optional data lines are in lower case. Note also that the requirement to input some data lines may be dependent on the presence of others.

The majority of input lines, and parameters, used by *Xtal* programs are optional. The default for each program are set at their most commonly used values. In the *Program controls* section of this manual default values are indicated either with a dotted underline or in red (e.g. **sinv**), or as italicised numbers or characters on the right hand side of the line (e.g. 1000). Unless the user needs to specify a different value, these parameters or lines need not be entered.

The last data line for a given calculation is followed by one of the following: an **end** line; another *program initiation* line; a **finish** line; or an *end-of-file*.

4.2. **ABSORB**: Absorption corrections

Authors: George Davenport, Nick Spadaccini and James Stewart

Contact: Nick Spadaccini, Computer Science Department, University of Western Australia, Nedlands, WA 6907, Australia

ABSORB provides the following methods of correcting diffraction data for absorption effects: the Gaussian integration method devised by Busing and Levy (1957), the analytical method of de Meulenaer and Tompa (1977); corrections for spherical crystals; corrections for cylindrical crystals; and corrections derived from tables supplied by the user. Options include the choice of method, calculation of the mean path length T_{bar} , and the application of the absorption correction to data in the bdf.

4.2.1. Application

There are four steps in the determination of absorption corrections:

1. Careful measurement of the dimensions of the crystal.
2. Preparation of a description of the orientation of the crystal relative to its mounting on the diffractometer.
3. Determination, by recalculation, of the diffractometer angles and beam vectors for each reflection.
4. Calculation of the absorption correction for each reflection.

4.2.1.1. Analytical Method

In this method the convex polyhedral crystal is subdivided into tetrahedra. This method has the advantage that the absorption correction for a tetrahedron can be calculated analytically. The total absorption correction for the crystal is obtained by summation over all the tetrahedra which make up the polyhedron that is the crystal. The algorithm used is that given by Alcock (1974). For each tetrahedron the volume and transmittance are calculated.

4.2.1.2. Gaussian Method

In this method the convex polyhedral crystal is subdivided by an arbitrary grid set up inside the polyhedron. The path lengths of rays from the source to each point and out along the diffraction direction are calculated. A Gaussian integration over all the path lengths for all the grid points gives the absorption correction for each reflection. The methodology is essentially that of Busing and Levy (1957).

4.2.1.3. Spherical Method

Absorption corrections for spherical crystals of radius r are selected by entering `sphere r` on the **ABSORB** line. The values of A^* , and T if requested, are derived from the algorithm of Tibballs (1982,1990) which is reportedly accurate to better than 1% for $\mu r < 2.5$.

4.2.1.4. Cylindrical Method

Absorption corrections for scattering in the equatorial planes of a cylindrical crystal of radius r are selected by entering `cylinder r` on the **ABSORB** line. Note that corrections for scattering from planes in the general orientation of a cylindrical specimen are not possible. The values of A^* , and T if requested, are again derived from the algorithm of Tibballs (1982,1990) which is reportedly accurate to better than 1% for $\mu r < 2.5$.

4.2.1.5. Tables Method

Values of A^* and T may be supplied by the user by selecting the table option on the **ABSORB** line. One or more `astar` lines containing values of the absorption corrections A^* at intervals of 5° must follow the **ABSORB** line. If the `tbar` option is requested on the **ABSORB** line, one or more `tbar` lines containing values of T at intervals of 5° must also be included. Four point interpolation routines are used in both cases.

4.2.2. Crystal Descriptions

To specify the shape of a crystal, it is necessary to describe the faces of the crystal. The following three methods are available in the **ABSORB** program. Note that one and only one method may be used to describe the crystal.

4.2.2.1. facem1

Faces are described by giving, for each face, the Miller indices and the perpendicular distance from the face to an arbitrary origin within the crystal. The origin must be the same for all faces of the crystal. The Miller indices are transformed into the crystal Cartesian system using the B matrix (Busing and Levy, 1967), and the resulting vector is normalized, producing the normal to the face.

$$\mathbf{n} = \mathbf{B} \mathbf{h} / |\mathbf{B} \mathbf{h}|$$

$$n_1 x + n_2 y + n_3 z = d \text{ (equation of face)}$$

4.2.2.2. facept

Each face is defined by the coordinates of 3 points on the face in an arbitrary Cartesian system (it must be right-handed, orthogonal and normalized). The unit length in this system is 1.0 mm. The coordinate system origin must not be on an edge or face.

4.2.2.3. face

Each crystal face is specified by the χ and ϕ angles required to place it in the diffracting position, plus the distance from the crystal face to an arbitrary origin inside the specimen. This kind of measurement can be readily done on diffractometers which have the viewing telescope mounted perpendicular to the χ circle. The faces need have no relationship to the two orienting reflections.

4.2.3. Diffractometer Description

The ABSORB program is designed to accept input data from any 4-circle diffractometer, and to calculate absorption corrections for that diffractometer. The motions of any diffractometer can be described by four parameters; the direction of rotation of the 2θ , χ , ϕ , and ω circles. In the bisecting mode this reduces to 3-circles ($\omega=\theta$). For ψ scans (rotations about the scattering vector) the sense of rotation about the normal to the diffraction plane must be included. Described below is the method which must be followed.

1. Set all circles to zero.
2. Look downward from above the diffractometer and note the directions of rotation of the 2θ and ω circles as clockwise (C) or anti-clockwise (A).
3. Look along the direction of the primary beam, from the x-ray source toward the goniometer head. Note the direction of rotation of the χ circle as clockwise or anti-clockwise.
4. Look downward from above the diffractometer and note the direction of rotation of the ϕ circle as clockwise or anti-clockwise.
5. Look down the scattering vector note the direction of motion during the ψ rotation as clockwise or anti-clockwise.

The following table lists the parameters for a number of diffractometer systems, both real and constructs used in papers. This information has been supplied by Jacek Grochowski.

System					Senses of Rotation	
	2θ	χ	ϕ	ω	ψ	ψ_a
Siemens (Syntex, Nicolet)	A	A	C	A		
CAD-4	C	C	C	C		
Philips PW 1100	A	C	A	A		C
Busing and Levy (1967)	C	C	C	C		C
Chidambaram (1980)	A	A	A	A		
International Tables, Vol. IV	C	A	A	A		A
Picker FACS-1	C	A	A	C		
Picker with Finger refinement	C	C	C	C		

Notes:

a. At the present time the ABSORB program does not handle any system where the sense of rotation is not uniquely definable. This occurs for the sense of rotation of ψ for the Syntax after Sparks.

The formulae of Busing and Levy can be modified using the equations which follow (BL = Busing and Levy, C = clockwise, A = anti-clockwise). For any angle τ ($\theta, \chi, \phi, \omega, \psi$):

$$\tau(\text{BL}) = \tau(\text{C}) = -\tau(\text{A})$$

By introducing the parameters $S(\theta)$, $S(\chi)$, $S(\phi)$, $S(\omega)$ and $S(\psi)$, the senses of rotation of the circles of a diffractometer are defined as:

$S = +1$ for clockwise rotation $S = -1$ for anticlockwise rotation

Three equations are produced which convert the angles used on a diffractometer, (DIFF), to equivalent angles in the Busing and Levy system.

$$\tau(\text{BL}) = S(\tau) * \tau(\text{DIFF})$$

The program will assume the bisecting mode unless the angle ψ is stored on the bdf ([lrrefl](#), IDN=1011). If, in the ψ scan mode, ω does not fall within the diffractometer limits (specified in Fields 7-8 on the **diff** line) the equivalent solution is used

$$\chi = -\chi, \phi = \pi + \phi, \omega = \pi + \omega$$

4.2.3.1. Coordinate Systems

A number of different coordinate systems are used in determining absorption corrections. These are summarized as follows: ($\mathbf{v}(\text{system})$ represents a column vector in a given coordinate system.)

1. The reciprocal lattice system may be non-Cartesian but is not necessarily. In this system a vector $\mathbf{v}(\text{hkl})$ is represented as a linear combination of the right-handed reciprocal lattice vectors.
2. The crystal system is a right-handed Cartesian system obtained through the transformation of the hkl system $\mathbf{v}(\text{crystal}) = \mathbf{B} \mathbf{v}(\text{hkl})$, where \mathbf{B} is the transformation matrix defined using the direct and reciprocal lattice parameters. The \mathbf{B} matrix is listed below.

$$\begin{array}{lll} a^* & b^* \cos\gamma^* & c^* \cos\beta^* \\ 0 & b^* \sin\gamma^* & -c^* \sin\beta^* \cos\alpha \\ 0 & 0 & 1/c \end{array}$$

Calculation of the absorption corrections is done using the coordinates of the crystal system.

3. The ϕ system is a right-handed Cartesian system which is rigidly attached to the ϕ shaft of the diffractometer. The definition of the axes is completely arbitrary. For two examples see Busing and Levy (1967) and Coppens (1970). The ABSORB program uses the definition of Busing and Levy, i.e., with all circles set to zero, the x-axis is along the scattering vector, the y axis is in the direction of the primary beam, and the z-axis points upward. The ϕ system is related to the crystal system by the rotation matrix \mathbf{U} as follows:

$$\mathbf{v}(\phi) = \mathbf{U} \mathbf{v}(\text{crystal})$$

4.2.4. Plot Of Crystal Shape

Two views of the crystal shape are provided as options. These are a projection of the crystal shape along the three cell directions or a stereoscopic view of the crystal tilted 15° in z and -10° in y. These options are

selected by entering `proj` or `stereo` on the **ABSORB** line. All plot information is stored on the plot bdf `abs` and the program **PLOTX** must be used to display the selected view on a graphical device.

4.2.5. File Assignments

- Reads reflection data from the input archive `bdf`
- Writes absorption corrections to the output archive `bdf`
- Writes crystal shape commands to the plot bdf `abs`
- Optionally, writes `HKL` lines to the line file `pch`

4.2.6. Examples

```
remark SEE ALCOCK (1974) AND FLACK AND VINCENT (1980)
ABSORB gauss
diff c c c c c 0
orient 1 0 0 -30.66 83.32 0 1 0 23.82 34.93
faceml 1 0 0 0.10
faceml 0 1 1 0.15
faceml 0 -2 1 0.05
faceml -3 0 1 0.03
faceml 1 1 -4 0.13
grid 8 8 8 0
```

```
ABSORB analyt : Alcock's Analytical Test
diff c c c c c 0
orient 1 0 0 -30.66 83.32 0 10 23.82 34.93
faceml 1 0 0 2.5
faceml -1 0 0 2.5
faceml 0 0 -1 10
faceml 0 1 0 3.75
faceml 0 -1 0 3.75
faceml 0 -1 1 0.883875
faceml 0 1 1 0.883875
```

4.2.7. References

- Alcock N.W. 1970. *The Analytical Method for Absorption Correction*. Crystallographic Computing. F.R Ahmed, S.R. Hall, and C.P. Huber, eds., Munksgaard. Copenhagen: 271-278.
- Alcock, N.W. et al. 1972. *An Improvement in the Algorithm for Absorption Correction by the Analytical Method*. Acta Cryst. **A28**, 440-444.
- Alcock, N.W. 1974. *Absorption and Extinction Corrections: Calculation Methods and Standard Tests*. Acta Cryst. **A30**, 332-335.
- Busing, W.R. and Levy, H.A. 1957. *High-Speed Computation of the Absorption Correction for Single Crystal Diffraction Measurements*. Acta Cryst. **10**, 180-182.

- Busing, W.R. and Levy, H.A. 1967. *Angle Calculations for 3- and 4- Circle X-ray and Neutron Diffractometers*. Acta Cryst. **22**, 457-464.
- Cahen D. and Ibers, J.A. 1972. *Absorption Corrections: Procedures for Checking Crystal Shape , Crystal Orientation, and Computer Absorption Programs*. J.Appl. Cryst. **5**, 298-299.
- Chidambaram, R. 1980. *Absorption Corrections for Single-Crystal X-ray and Neutron Data*. Computing in Crystallography. R. Diamond, S. Ramaseshan, and K. Venkatesan, eds., Indian Academy of Sciences. Bangalore, India: 2.01.
- Coppens, P. 1970. *The Evaluation of Absorption and Extinction in Single-Crystal Structure Analysis*. Crystallographic Computing. F.R. Ahmed, S.R. Hall, and C.P. Huber, eds., Munksgaard. Copenhagen: 255.
- Coppens, P. & Hamilton, W.C. 1970. *Anisotropic Extinction Corrections in the Zachariasen Approximation*. Acta Cryst. **A26**, 71-83.
- de Meulenaer, J. and Tompa, H. 1965. *The Absorption Correction in Crystal Structure Analysis* . Acta Cryst.**19**, 1014-1018.
- Flack, H.D. and Vincent, M.G. 1980. *Absorption and Extinction Corrections: Standard Tests*. Acta Cryst. **A36**, 682-686.
- Gabe, E.J. 1980. *Diffractometer Control with Minicomputers*. Computing in Crystallography. R. Diamond, S. Ramaseshan, and K. Venkatesan, eds., The Indian Academy of Sciences. Bangalore, India: 1.01.
- Grochowski, J. Personal communication.
- Tibballs, J.E. 1982 *The Rapid Computation of Mean Path Lengths for Spheres and Cylinders*. Acta Cryst. **A38**, 161-163.
- Tibballs, J.E. Personal communication.

4.3. **ADDATM**: Load atom parameters

Authors: James Stewart, Nick Spadaccini and Syd Hall

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

ADDATM places atom and general parameters on the archive bdf. Atoms in special positions are detected and the symmetry constraints on variables are stored in the bdf.

4.3.1. Calculations Performed

ADDATM provides for the *ab initio* addition of data to the logical record *lratom*: on the bdf, and can also edit existing data on the input bdf. ADDATM also detects atoms in special positions and generates the appropriate constraints for positional and thermal parameters.

The parameters loaded by ADDATM are of two types; general structural parameters and atom specific parameters.

4.3.1.1. General Parameters

Three general structure parameters are loaded by ADDATM as a function of data set. F-relative scale factors for each scale group with a `scale` line; overall isotropic thermal parameters with `uov`; and extinction parameters with `extinct`.

4.3.1.2. Atom Specific Parameters

Specific atom parameters are placed in the bdf record, *lratom:*. The following list of atom parameters are loaded using the control lines **atom**, **link**, **u**, **uij** and **suij**:

- atomic coordinates,
- H atom linkage calculation
- atomic thermal displacement parameters,
- population parameter,
- anomalous population parameter,
- and standard deviations for those parameters.

4.3.2. Atoms In Special Positions

ADDATM identifies atoms in special positions and places them into *lratom:* with the correct atom multiplicity set. The constraint information required by refinement programs is stored in *lrcons:* of the bdf. Special positions are detected by transforming the input x, y, and z over all the symmetry operations and testing if any position is within 0.0005 of the unit cell edge of the parent atomic coordinates.

4.3.2.1. Special position constraints

Special position constraints are determined from the application of atom parameters to the 4x4 symmetry matrices (i.e. rotation plus translation operations). This provides the constraints and interdependence of atomic coordinates and thermal parameters due to symmetry. The printed output of the atom parameters indicates that a given atom is in a special position by placing symbols instead of the decimal fractional coordinates entered in the data input stream. The notation is the same as that which is listed in the International Tables for Xray Crystallography, Vol. I. Thus in space group P6₁22 an atom entered as shown:

```
atom s1 .1313 .2626 .25
```

will be printed as:

```
:          x          y          z          mul
S1          .1313      2X          1/4          .50
```

If print priority is set to 4 with a '**reset psta 4**' line each parameter constraint will be printed as it is transferred to *lrcons:* of the bdf.

4.3.2.2. Changing current atom list in archive file

ADDATM also provides updating and editing facilities. The "update" mode is switched on by entering **upd** on the **ADDATM** line. This causes existing atom sites from the input archive bdf to be placed into the atom list. Input **atom/linkatomg/u/uij/suij** lines will either modify existing atom site data if atom labels match, or be added to the list if labels do not match an existing site label. Note that all **u** lines must still be preceded by an **atom** line with the same atom label.

The **stack**, **rename**, **delete** and **editu** controls apply in both *ab initio* and update modes. These lines must be entered after any **atom/atomg/u/uij/suij** lines have been entered. **stack** lines are used to sort the sites

in the atom list. The first entry will determine the sites at the top of the list. Up to 100 entries are permitted on any number of **stack** lines. The **rename** line is used to change atom labels. Atom-type symbols may NOT be entered. **delete** lines are used to delete sites from the atom list. An atom label is used to delete a single atom and an atom-type symbol is used to remove all atoms of that type. **editu** lines are used to change the U values of atom sites or atom types. Individual isotropic values may be entered or atoms converted between isotropic and anisotropic.

The **invert** line to change the enantiomorph-polarity of the structure by inversion in a point. The algorithm is described in Appendix B of Bernardinelli and Flack (1985). A suitable point of inversion is found (in some cases the origin cannot be used). In the case of the 22 space groups forming the 11 enantiomorphic pairs, the space group has to be changed into the opposite member of the pair. No user intervention is required in this procedure.

The **link** line adds geometrical atom site recalculation information to the archive. This is only required for riding atom model refinement using CRILSQ! Existing link information may be purged from the archive by running **ADDATM upd plink**. Link information is automatically generated by both PIG and CALCAT programs. In the case of PIG this link information may be omitted by invoking the control line **PIG nolink** whenever H atoms are to be built from within PIG.

4.3.3. File Assignments

- Reads current input archive bdf
- Writes atom parameters to output archive bdf

4.3.4. Example

```
ADDATM
atom S .202 .798 .91667 2.0 .5
uij S .009162 .009162 .001205 .04581 0 0
atom O .498 .498 .66667 2.0
atom Cl .488 .096 .038 2.5
```

The atoms **s**, **o** and **cl** are loaded onto the bdf with mixed thermal parameters.

4.3.5. References

- Cruickshank, D.W.J. 1956. *The Determination of the Anisotropic Thermal Motion of Atoms in Crystals*. Acta Cryst. **9**, 747.
- Henry, Norman F. and Lonsdale, Kathleen. 1965. Eds. *International Tables for X-ray Crystallography Vol. 1*. Birmingham, England: Kynoch Press.
- Bernardinelli, G. and Flack, H.D. 1985. *Acta Cryst.* **A41**, 500-511.

4.4. **ADDPAT**: Create powder pattern bdf

Authors: Gert Kruger, Syd Hall

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

ADDPAT generates a powder diffraction pattern binary file from a variety of instrumental data files.

4.4.1. Description

Powder patterns can be read in from:

- Philips APD ASCII files
- PC-APD files
- Raw data files
- WYRIET input files
- UXD (Universal X-ray Data) input files

4.4.2. File assignments

- Reads powder pattern data from powder files (default ext POW)
- Writes powder pattern binary data files (ext PAT)

4.4.3. Data Formats

4.4.3.1. RAW DATA FILE as of 12-DEC-91 16:23

```
PW1700 Automated Powder Diffractometer System 1
<newline>
Listed RD file name      : [300,7]BARKH3.RD
Sample identification    : kaapse hoop barb/sti
Measurement date/time   : 11-DEC-91 11:12
Generator settings      : 40 kV, 40 mA
Co alpha1,2 wavelengths : 1.78896, 1.79284 Ang
Step size, sample time  : 0.020 deg, 15.00 s, 750.00 s/deg
Monochromator used      : Yes
Divergence slit         : Fixed (1/2 deg)
<newline>
Analysis program number : 7
File angle range        : 10.000 - 80.000 deg
Range in D spacing      : 1.39156 - 10.2630 Ang
Maximum intensity       : 473895. cts, 31593.0 cps
<newline>
```

Angle	Counts				
10.0000	437.	396.	416.	454.	433.
10.1000	433.	458.	462.	433.	493.
10.2000	449.	458.	433.	445.	480.
10.3000	428.	424.	384.	412.	454.
10.4000	400.	441.	437.	433.	388.
10.5000	462.	441.	437.	462.	471.

10.6000	441.	412.	412.	437.	462.
10.7000	437.	408.	424.	441.	420.
10.8000	445.	502.	433.	416.	428.
10.9000	454.	396.	428.	420.	433.

4.4.3.2. APD

```

SampleIdent,qzstd8,/
Title1,J Gutzmer,/
Title2,PC-APD, Diffraction software,/
DiffType,PW1710,/
DiffNumber,1,/
Anode,Fe,/
LabdaAlpha1, 1.93604,/
LabdaAlpha2, 1.93998,/
RatioAlpha21, 0.50000,/
DivergenceSlit,Fixed,1/2,/
ReceivingSlit,0.2,/
MonochromatorUsed,YES ,/
GeneratorVoltage, 40,/
TubeCurrent, 40,/
FileDateTime, 3-mar-1994 18:04,/
DataAngleRange, 3.0000, 130.0000,/
ScanStepSize, 0.020,/
ScanType,STEP,/
ScanStepTime, 5.00,/
RawScan
  681,    640,    708,    692,    630,    625,    620,    615
  615,    595,    581,    576,    595,    595,    581,    581
  586,    586,    552,    529,    524,    552,    534,    538

```

4.4.3.3. RAW

12.46	2.	1.	12.48	1.	1.	12.50	1.	1.	12.52	1.	1.
12.54	3.	1.	12.56	3.	1.	12.58	3.	1.	12.60	5.	2.
12.62	5.	2.	12.64	-2.	2.	12.66	-1.	2.	12.68	3.	2.
12.70	3.	2.	12.72	5.	2.	12.74	9.	2.	12.76	5.	2.
12.78	10.	3.	12.80	12.	3.	12.82	8.	3.	12.84	9.	3.
12.86	14.	4.	12.88	26.	4.	12.90	22.	5.	12.92	35.	6.

4.4.3.4. WYR

20.120	.020	60.100	PS999RF	24AP90	QUARTZ,	6-7-86	T=10	20.120
30	34	44	37	35	25	29	46	20.260
42	34	40	35	21	39	43	30	20.420
48	44	52	44	50	46	47	56	20.580
64	62	86	79	103	124	197	257	20.740
484	831	1558	2478	3169	3185	2534	1828	20.900
1217	661	263	173	128	79	88	68	21.060

4.4.3.5. UXD

```
; A:\AH-a26-9d.RAW(Diffrac Plus V1.01 file) converted by XCH V1.0
_FILEVERSION=2
_SAMPLE='[Co(NH3)6][Ce(dipic)3] RAW'
+_SAMPLE='Line 1Line 2'
_SITE='SRCAMMPChemistry Department, University of Western Australia'
_USER='SRCAMMP'
_GONIOMETER_CODE=0
; D5000 Theta/2Theta; Standard
_SAMPLE_CHANGER_CODE=0
_ATTACHMENTS_CODE=0
_GONIOMETER_RADIUS=217.500000
_FIXED_DIVSLIT=2.000000
_FIXED_SAMPLESLIT=2.000000
_SOLLER_SLITS='N'
_FIXED_DETSLIT=1.000000
_MONOCHROMATOR=0
; None
_SOLLER_SLITS_2='Y'
_THIN_FILM='N'
_BETA_FILTER='N'
_FIXED_ANTISLIT=1.000000
_RUNTIME=2752.000000
_WL_UNIT='A'
_WL1=1.540560
_WL2=1.544390
_WL3=1.392220
_WLRATIO=0.500000
_ANODE='Cu'
; (Data for Range number 1)
_DRIVE='COUPLED'
_STEPTIME=2.000000
_STEPSIZE=0.040000
_STEPMODE='C'
_START=5.000000
_THETA=2.500000
_2THETA=5.000000
_PHI=0.000000
_DIVERGENCE=1.000000
_ANTISCATTER=1.000000
_DETECTOR=1
; S.C.
_HV=950.000000
_GAIN=2.000000
_LLD=0.500000
_ULD=2.000000
_DETECTORSLIT='out'
_AUX1=0.000000
_AUX2=0.000000
_AUX3=0.000000
_TIMESTARTED=2.703000
_TEMPERATURE=299.839996
_TEMP_RATE=0.000000
_TEMP_DELAY=0.000000
_KV=40
_MA=35
_RANGE_WL=1.540560
_3DPLANE=1
_2THETACOUNTS
5.0000 121
5.0400 105
5.0800 124
5.1200 105
```

```

5.1600 99
...
59.8000 37
59.8400 35
59.8800 38
59.9200 34
59.9600 42
60.0000 37

```

4.4.4. Examples

```

title Philips data
compid philip
master no
ADDPAT inp bpf out plt
end
finish

```

```

title MnO2 full house
compid mno2
master no
ADDPAT
finish

```

```

title Test using Edingtonite structure ETH
compid ed
master no
ADDPAT type raw

```

4.5. *ADDREF*: Load reflection data

Authors: George Davenport and Syd Hall

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

ADDREF adds measured reflection data to the archive *bdf* and converts them into different structure factor coefficients. *ADDREF* can also create reflection data, expand or contract an existing reflection record, and merge reflection records. Options include the insertion of interpolated form factors into the reflection record, the removal of systematically absent reflections, and the calculation and application of the Lorentz-polarization factor for a range of geometries.

4.5.1. Overview

Most crystallographic calculations require reflection information to be present in the bdf. The basic reflection data includes the Miller indices, $\sin\theta/\lambda$, reflection multiplicity, the symmetry reinforcement factor (epsilon), phase restriction codes and $|F|$ relative. Other quantities such as $\sigma|F|$, reflection weights, the Lorentz-polarization factor, scale group number, and interpolated scattering factors may be added as the user requires.

Up to five sources of input may be used (input lines and/or up to four binary data files) or dummy reflection records may be generated from the cell dimensions within specified limits of h, k, l, and $\sin\theta/\lambda$. Generated data is useful for calculating structure factors when only the atomic parameters are available. When more than one input source is used, it is vital that each source have the same set of reflections and refer to the same compound. This constitutes a *lrrefl*: merging feature. Also worth noting is that the input archive bdf must contain all cell, symmetry, and cell content information pertinent to the compound. In other words, the input archive bdf must contain the data generated by *STARTX* or any subsequent calculation.

The sources of reflection data, as well as the specific items, are specified using combinations of **bdfin** and **hklin** lines. One **hklin** line is allowed, and one **bdfin** line per input bdf is allowed. These lines can be in any order, but an important *caveat* applies. That is, if a data item is specified as coming from two different sources (e.g. if the Miller indices are given on an **hklin** line and also on a **bdfin** line), the item will be taken from the source specified *first*.

For convenience, a **remove** line has also been provided. For cases in which a large number of items are to be taken from an input bdf, it may be easier to specify those items which are not to be output than those items which are to be output. The **remove** line may thus be used in conjunction with a **bdfin** line (using the **a11** option) to trim unneeded items from the reflection record (see Examples).

Items on **bdfin**, **hklin**, **remove**, and **c** (continuation) lines may be specified in two ways. A set of four-letter mnemonics are available for the most commonly used quantities. For more involved applications, items can be specified using their identification numbers as listed in the BDF section at the back of the manual.

The printed output contains a list of items from each source, along with the status of each item. The status refers to possible user errors, such as a duplicate request for an item or the absence of an item on the given source. A list of data for each reflection is given, for either a specified number of reflections or all reflections, depending on the user's choice. A list of items contained in the reflection record of the output bdf is given, along with the maxima and minima of the data for those items. In addition, the maximum magnitudes of the Miller indices and the maximum and minimum values of $\sin\theta/\lambda$ for the output bdf are stored in *lrdsset*: of the output bdf for use by other programs.

ADDREF tests each reflection to see if it is systematically absent under the space group symmetry given. Systematically absent reflections are either rejected from the file or marked with an rcode of 5 and these must not be included in the bdf as observed reflections. Inclusion of such reflections will cause the Fourier transform to show incorrect symmetry. To ensure that no

duplicate reflections are present on the bdf, the program *SORTRF* (with **average** specified) should be run after (or, in some cases, before) ADDREF.

Reflection status codes (rcode)

The values of *rcode* which are recognised for most XTAL calculations are as follows:

1. "observed" reflection (ie. $Y \geq n\sigma Y$)
2. "less than" reflection (ie. $Y < n\sigma Y$)
3. unreliable reflection not used in refinement or R factors
4. Friedel-related value is missing when data stored as Friedel pairs
5. systematically absent

Note that rcodes are assigned by *SORTRF* and retained by ADDREF.

4.5.2. Calculations Performed

Generate hkl data

Occasionally it is necessary to produce a file containing 'dummy' reflection data. Measured diffraction data may not be available for input but reflections are needed in order to calculate structure factors with programs such as *FC*. Reflection data may be generated by entering the **hklgen** line. It should be emphasised that **ADDREF** only limits the data to the boundaries of the defined limits (see **ADDREF** line). **ADDREF** will remove systematically absent reflections (due to symmetry) but for crystal classes higher than orthorhombic it will generate redundant reflections. These are easily removed from the bdf by applying the sort/merge routine **SORTRF** immediately following **ADDREF**. Using **SORTRF** to average the generated data (option **aver 1**) will ensure that only a unique set of reflections remain. *It is good policy always to run SORTRF after ADDREF, especially if the reflection generation feature is used.*

The minimum contents of the *lrrefl*: record are the packed hkl word, $\sin\theta/\lambda$, the packed phase-code/multiplicity/epsilon word and structure factor coefficient (this can be **Frel**, **Frel** squared or **Irel**). The input **hklgen** line *must* contain the codes **hkl** and either **frel**, **f2r1** or **irel**. This line may also contain any of the other option codes if they are needed, but remember that all values, except those calculated by **ADDREF** (e.g. **r1p** or the interpolated form factors), will be stored on the bdf as 0. Note that it is possible to generate the Friedel equivalent reflections for noncentrosymmetric space groups by entering the **frie** code on the **ADDREF** line.

When running **SORTRF** after **ADDREF** make sure that, in addition to specifying the sort order code, the **aver 1** and the merged coefficient (**frel**, **f2r1** or **irel**) are also entered. Note that if Friedel equivalent data is to be preserved the **pakfr1** or **sepfr1** code must be entered on the **SORTRF** line.

4.5.2.1. Form Factor usage (*Important!*)

The *lrscat*: are present in the *Archive* (stored by **STARTX**). These may be used to calculate interpolated form factor values which are stored in each reflection record. This option is invoked by the **ffac** option on the **ADDREF** line. This provides more precise structure factors with programs such as *FC*, *CRYLSQ* etc. The disadvantage is that the size of the archive bdf is increased substantially and the extra precision is often unwarranted for routine structure analyses. This option should be used for precision analyses or when *R-factors of less than 0.04 are anticipated.*

4.5.2.2. Data Reduction

ADDREF provides many options for reducing the data to be stored in the output archive bdf. It is a "two pass" program. During the first pass, statistical information on the reflection data is calculated and written to a scratch file. During the second pass, reflection data processing is completed and the results are written to the output archive bdf.

The following calculations may be performed:

- transform Miller indices
- generate hkl data
- calculate $\sin\theta/\lambda$
- calculate the multiplicity and reinforcement factor, and phase restriction code
- determine if reflection is systematically absent for this space group
- apply Bayesian statistics to weak data
- interpolate the scattering factors and store reflection packets
- obtain max/min h, k, l, $\sin\theta/\lambda$, and other quantities

- calculate and apply 1/Lp for all geometries and radiation
- convert input function of F to requested output function of F
- calculate the standard deviation in the chosen F function

4.5.2.3. Lorentz-polarization corrections

Algorithms for the common 1/Lp factors are available. In the equations below θ refers to the reflection diffraction angle, and θ_m to the diffractometer angle for the monochromator crystal.

$$\begin{aligned} \text{Lorentz factor} &= 1/(2\sin 2\theta) && \text{for single crystal} \\ &= 1/(2\sin 2\theta \sin \theta) && \text{for powder} \end{aligned}$$

The formulae used for polarization are those described by Azaroff (1955), Hope (1971), and Vincent & Flack (1980). The general expression for polarization of a *twice-diffracted* beam is

$$P = K_i(1-B)(\cos^2\rho \cos^2 2\theta + \sin^2\rho) + B(\sin^2\rho \cos^2 2\theta + \cos^2\rho) / [(1-B)K_i + B]$$

where K_i , the polarization ratio is $K_k = \cos^2 2\theta_m$ for an ideal mozaic crystal and $K_d = |\cos 2\theta_m|$ for an ideal crystal. B is the fraction of the intensity with the electric field parallel to the plane of the monochromator (Azaroff's notation is E_π). This direction is given by the cross product of the vector in the direction of the source beam and the normal to the monochromator crystal plane. In a standard X-ray diffractometer the source beam is unpolarized and B=0.5. θ_m is the monochromator angle and ρ is the angle between 2 planes of diffraction (i.e. planes, containing the incident and reflected rays of the monochromator and the sample).

The general expression for the integrated polarization of *mosaic/perfect* crystal is

$$P_i = (1-C) P_k + C P_d,$$

where C is the monochromator perfection factor (the fraction of the monochromator crystal considered to be perfect) and P_k and P_d the kinematic and dynamical components of the polarization. **RLP** is the reciprocal of Lp.

X-ray powder, no monochromator

$$\mathbf{RLP1} = 2\sin \theta \sin 2\theta / (1 + \cos^2 2\theta)$$

X-ray single crystal, no monochromator

$$\mathbf{RLP2} = 2\sin 2\theta / (1 + \cos^2 2\theta)$$

X-ray single crystal, with monochromator and perfection factor, perpendicular setting

In the perpendicular monochromator setting, the rotation axis of the monochromator crystal is perpendicular to the normal to the equatorial plane of the diffractometer (ie. 2θ axis), such that the plane of the incident beam and the beam reflected by the monochromator is perpendicular to the plane of the beam reflected by the monochromator and the beam reflected by the crystal under study. Rho, as defined by Azaroff, is 90° . If the source beam incident on the monochromator is unpolarized then B=0.5. For a perfectly polarized beam B=0. for this setting. *This is the CAD4 setting.*

$$T1 = (1 - C) ((1-B)\cos^2 2\theta_m + B\cos^2 2\theta) / (B + (1-B) \cos^2 2\theta_m)$$

$$T2 = C(B\cos^2 2\theta + (1-B)\cos 2\theta_m) / (B + (1-B)\cos 2\theta_m)$$

$$\mathbf{RLP3} = \sin 2\theta / (T1 + T2)$$

X-ray single crystal, with monochromator and perfection factor, parallel setting

For the equatorial, or normal, monochromator setting, the rotation axis of the monochromator is parallel to the normal to the equatorial plane of the diffractometer (ie. 2θ axis) such that the incident beam, the beam

reflected by the monochromator and the beam reflected by the crystal under study all lie in the same plane. If the source beam incident on the monochromator is unpolarized then $B=0.5$. For a perfectly polarized beam $B=1$. for this setting. *This is the Nicolet setting.*

$$T1 = (1 - C) (B + (1-B)\cos^2 2\theta \cos^2 2\theta_m) / (B + (1-B)\cos^2 2\theta_m)$$

$$T2 = C(B + (1-B)\cos^2 2\theta \cos 2\theta_m) / (B + (1-B)\cos 2\theta_m)$$

$$\mathbf{RLP4} = \sin 2\theta / (T1 + T2)$$

Neutron powder (no polarization)

$$\mathbf{RLP1} = 2\sin \theta \sin 2\theta$$

Neutron single crystal

$$\mathbf{RLP2} = 2\sin 2\theta$$

4.5.2.4. Calculation of $\sigma(F)$ from Intensity data

The calculation of $\sigma(F)$ is straightforward except where F is close to zero or negative. To avoid the asymptotic form of this conversion, the following expression is used in **ADDREF**.

$$\sigma F = (\mathbf{RLP} \sigma I) / (F + \{F^2 + \mathbf{RLP} \sigma I\}^{1/2})$$

4.5.3. File Assignments

1. Reads *lrcell*: and symmetry data from the input archive bdf
2. Writes updated file to the output archive bdf
3. Optionally, reads reflection data form specified (on **bdfin**) bdf

4.5.4. Examples

```

title CREATION OF AB INITIO REFLECTION RECORD
ADDREF dset 1 ffac list
reduce itof rlp2
hklin skip hkl rcod irel sigi absf eval
remove irel sigi
hkl p6122 0 1 1 1 22004.8 4043.4 1.0 3.0
hkl p6122 0 1 4 1 387.4 205.5 1.0 1.0
hkl p6122 0 1 5 1 6735.0 1110.5 1.0 3.0
:.....reflection data omitted for brevity
hkl p6122 2 2 7 1 358.1 98.2 1.0 0.3
hkl p6122 2 2 8 1 384.3 78.1 1.0 0.3
hkl p6122 2 3 3 1 2275.6 247.0 1.0 2.0

```

The **ADDREF** line specifies that interpolated form factors are to be inserted in *lrrefl*:, and that all reflections are to be listed. The **reduce** line indicates that relative intensities and their sigmas are to be converted to relative F 's and $\sigma(F)$'s; the L_p factor is to be calculated by method 2 (spectrometer with Eulerian cradle geometry and 2θ scan). The **hklin** line specifies the items which are found on **hkl** lines. Note that the user has labeled the **hkl** lines with the compound identification, so **skip** must be used to ignore this data. The **remove** line indicates that **irel** and **sigi** are to be excluded from the output bdf (they are used, however, in the conversion to **fre1** and **sigf**, so they must be specified on the **hklin** and **hkl** lines.

```
title Contraction of the reflection record
ADDREF
bdfin all
remove absf tbar extf
```

To decrease the size of a large archive bdf, first check to see which items are in *lrrefl*: of the file. Then check (carefully) to see which items are no longer necessary to keep in the bdf. Use the **remove** line to get rid of these, as in the procedure above. The input archive bdf is read. The absorption correction, extinction correction, and mean radiation path length will be removed from *lrrefl*: of the bdf by the lines above.

```
title Merging of reflection data
ADDREF
bdfin file a all
bdfin file ddd absf eval
```

Suppose the input archive bdf contains reflection information about a compound and the bdf with extension DDD contains a bdf with different reflection information about the same compound. The preceding sequence of lines will merge certain items from both bdf's and output them to the output archive bdf.

```
title EXAMPLE USE OF CONTINUATION LINES
ADDREF dset 1
hklin hkl frel sigf fcal 1000 1001 1002
c 1003 rcod tbar
hkl 1 1 1 40 5 56 289 33 256 4 0.1
```

The preceding example illustrates the use of **c** lines. Note that some items are specified using ID numbers and some are specified using four-letter mnemonics.

```
title Generate reflection data to sin(theta)/lambda=.5
ADDREF
limits *4 0.5
hklgen hkl frel
title Complete data preparation sequence
STARTX
cell 11.52 11.21 4.92 90 90.833 90 288.0
cellsd .012 .011 .005 0.0 .0005 0.0
sgname -p 2yab :p21/a
celcon o 12
celcon c 28
celcon h 24
DIFDAT cad
attenu 5.
genscl 3
SORTRF order khl aver 1 cull 1.5 print 1500 pakfrl
ADDREF dset 1 list 7 ffac lpin friedel
reduce itof rlp2 xray
bdfin hkl irel sigi rcod ifri sfri rcdf
remove irel sigi ifri sfri
```

In this example a run of *STARTX*, *DIFDAT*, *SORTRF*, and *ADDREF* is combined to produce a bdf with F values in logical record *lrrefl*: for a unique asymmetric set of reflections including the Friedel related pairs.

4.5.5. References

- Azaroff, L.V. 1955. *Polarization Correction for Crystal-Monochromatized X-radiation*. Acta Cryst. **8**, 701.
- Hope, H. 1977. *Polarization Factor for Graphite X-ray Monochrometers*. Acta Cryst. **A27**, 392.
- Iwasaki, Hitoshi and Ito, Tetsuzo. 1977. *Values of Epsilon for Obtaining Normalized Structure Factors*. Acta Cryst. **A33**, 227-229.
- Karle, I. 1969. *General Procedure for Phase Determination*. Crystallographic Computing. F.R. Ahmed, Sydney R. Hall, C.P. Huber, eds., Munksgaard. Copenhagen: 19-25.
- Kasper, John S. and Lonsdale, Kathleen. 1959. Eds. *International Tables for X-ray Crystallography Vol. II*. Birmingham, England: Kynoch Press.
- Larson, A.C. 1969. *The Inclusion of Secondary Extinction in Least-Squares Refinement of Crystal Structures*. Crystallographic Computing.
- F.R. Ahmed, S.R. Hall, C.P. Huber, eds., Munksgaard. Copenhagen: 291-294. Rollett, J.S. 1965. *Computing Techniques in Crystallography*. Elmsford, NY: Pergamon Press.
- Stewart, J.M. and Karle, J. 1976. *The Calculation of Epsilon Associated with Normalized Structure Factors, E*. Acta Cryst. **A32**, 1005-1007.
- Stewart, J.M. and Karle, J. 1977. *Two Papers on the Calculation of Epsilon for Obtaining Normalized Structure Factors*. Acta Cryst. **A33**, 519.
- Vincent, M.G. and Flack, H.D. 1980. *On the Polarization Factor for Crystal Monochromated X-radiation I Assessment of Errors*. Acta Cryst. **A36**, 610.

4.6. **ATABLE**: Tabulate atomic parameters

Authors: Lutz Englehardt and Syd Hall

Contact: Syd Hall, Crystallography Centre,

University of Western Australia, Nedlands, WA 6907, Australia

ATABLE tabulates atomic coordinates and thermal parameters suitable for publication. Atoms are listed in the same order as stored on the bdf. The precision of each parameter type is automatically determined and the standard deviations, if available, are appended.

Up to five tables are automatically output to the print file and the pch file.

Table 1 Atomic coordinates and isotropic thermal parameter U , or U_{eq} for anisotropically refined atoms (flagged with *) for all atoms, where

$$U_{eq} = (1/3) \sum_i \sum_j U_{ij} a_i^* a_j^* a_i a_j$$

Table 2 Identical to Table 1 but for non-hydrogen atoms only.

Table 3 Identical to Table 1 but for hydrogen atoms only.

Table 4 Thermal displacement parameters U_{ij} and U for all atoms.

Table 5 Anisotropic thermal displacement parameters U_{ij} .

Tables 2 and 3 will be listed only if there are hydrogen atoms. Tables 4 and 5 will be listed only if there are anisotropic thermal parameters. Table 5 will be listed only if there are isotropic thermal parameters.

4.6.1. File Assignments

- Reads atom data from the input archive bdf
- Writes tabulated data to the line file pch

4.7. *BAYEST*: Bayesian estimates of |I| and |F|

Author: Nick Spadaccini, Computer Science Department, University of Western Australia, Nedlands, WA 6907, Australia

BAYEST determines a posterior intensity and/or structure factor modulus and the associated standard deviations. The program is useful in the treatment of weak or negative intensities. The calculation is based on the work of French and Wilson (1978), and the user should refer to the original paper for details of the method.

4.7.1. Introduction

Intensity measurements which are negative are generally either excluded from the data set or are reset to zero. Both these practices result in biased determinations of the structure. This bias may be overcome by including all reflections in a refinement based on F^2 . However, for refinements based on $F(\text{rel})$ it is essential to obtain the best possible estimates to the structure factor modulus and its standard deviation.

A more adequate treatment of negative or weak intensities is required. Setting the structure factor to zero ignores the information present in the observation of a negative intensity (i.e. it is weak). Assuming the intensity distribution of data follows the Wilson distribution, and that the true intensity is constrained to be non-negative, a posterior estimate of I or $F(\text{rel})$ and their standard deviations can be made.

4.7.2. Intensity Measurements And Bayes Theorem

When prior information is known, say from a physical model, about an experimental measurement one has a starting point from which to infer a better estimate of the observation from the data. A useful method of employing prior knowledge is in the Bayesian approach to inference where the probability distribution is regarded as a measure of information and represents our degree of belief rather than just relative frequencies. Our interest lies in determining the best estimate of the intensity, denoted J . The experimentally measured intensity I and its variance $\sigma^2(I)$ are subject to counting and statistical error. What must be determined is the probability distribution of J given the observation I . The prior probability distribution of J , $p(J)$, is assumed to be the appropriate Wilson distribution. The likelihood distribution for I , $p(I | J)$, is assumed to be normal with mean J and variance $\sigma^2(I)$. Bayes theorem states that the posterior probability distribution of J given the observation I is:

$$p(J | I) = k p(J) p(I | J) \text{ where } p(J) = W(J, \Sigma) \text{ and } p(I | J) = N(J, \sigma^2(I))$$

The posterior estimates for I and its esd are given by the following expectation values:

$$E(J) = \int J p(J | I) dJ$$

$$E(\sigma^2(J)) = \int (J - E(J))^2 p(J | I) dJ$$

The posterior estimates of $F(\text{rel})$ and $\sigma^2(F)$ may be derived from $E(J)$ and $E(\sigma^2(J))$ or from the expectation values:

$$E(F) = \int F p(J | I) dJ$$

$$E(\sigma^2(F)) = \int (F - E(F))^2 p(J | I) dJ$$

4.7.3. The Wilson Parameter (S)

The prior distribution of J assumed requires knowledge of the Wilson parameter, Σ . Assuming the data collected contains a fair number of observations, Σ is determined from the mean value of F^2 in shells of s^2 (where $s = \sin\theta/\lambda$). The user may specify the number of shells the data is to be divided into through the `nrange` option in the `BAYEST` line. The default is 50 shells.

The program searches the bdf for items in a hierarchial manner. The priority item is F^2 . If this is present, the distribution of mean F^2 versus s^2 is calculated and the appropriate values of Σ determined. If I and L_p corrections are found, F^2 is generated and the distribution determined. If any of these items are present their esds must also be present.

The user can apply the correction to F(rel) only, leaving the other items as they were input. Otherwise all items present are corrected. The value of F(rel) may be corrected in one of the two following ways.

1. Requesting the `int` option and having F(rel) present on the bdf. In this case the corrected F(rel) is derived from the corrected I or F^2 .
2. Explicitly requesting F(rel) to be corrected only. In this case the corrected F(rel) and $\sigma(F)$ are derived from the expectation values $E(F)$ and $E(\sigma^2(F))$.

Note, however, that the two cases are not identical since $E(F)$ does not equal $(E(J))^{1/2}$.

The number of shells should be chosen so that there are sufficient reflections in each shell so that a reasonable determination of the distribution may be made (150-200 or reflections per shell).

4.7.4. Input Output

BAYEST is a two pass operation. The first pass determines the data set distribution and the second pass applies the Bayesian corrections. Items which **MUST** be present on the bdf are the maximum and minimum values of s, s for each reflection, and either of intensity and L_p , or F^2 . No expansion to the bdf is made. The items input are also output, though the user has the option to apply the correction as desired.

4.7.5. File Assignments

- Reads reflection data from the input archive bdf
- Writes corrected reflection data to the output archive bdf

4.7.6. Example

```
BAYEST nrange 35 frel
```

In this run the data is divided into 35 groups and the correction is applied to F(rel) only leaving the other items uncorrected. The printing is suppressed. Note that F(rel) and $\sigma(F)$ must be present in the input bdf. BAYEST does not expand the bdf by generating F(rel) from I or F^2 . That is left to `ADDREF`.

4.7.7. References

- French, S. and Wilson, K. 1978. Cryst. A34, 517-525.

4.8. *BONDAT*: Generate atom coordinates

Authors: Ruth Doherty, Jim Stewart and Howard Flack

Contact: Howard Flack, Laboratoire de Cristallographie, University of Geneva, 24 quai Ernest Ansermet, CH-1211 Geneve 4, Switzerland

BONDAT generates the coordinates of atoms whose positions can be inferred from the geometry of previously placed atoms.

4.8.1. Calculations Performed

BONDAT generates inferred atomic coordinates from known atomic coordinates. All the known atomic coordinates of a structure are loaded into memory. Once loaded, a series of input lines denoted by **CALCAT** are supplied. There are nine types of **CALCAT** input lines described in detail below.

Each line generates a different configuration of inferred atomic coordinates from known coordinates. The generation is carried out based upon matches of the character strings which are the atom identities. One feature of the program is that, once generated, new atoms become part of the list of atoms in memory and can therefore be used in further generations. Caution should be exercised when using this feature to extend chains because of the danger of the magnification of errors in the placement of the "known" atoms.

All the atoms which are used for generation must have coordinates which are geometrically correct for the generation to be carried out. The generating atoms must be located so that they form a connected set. There are no tests made in the BONDAT program to assure that this restriction is met. The use of BONDLA following BONDAT will serve as a check on the correctness of the locations of the generating atom coordinates.

In archive update mode, BONDAT adds the **CALCAT** generation instructions as well as the new atom sites to the archive. This information can be used by the least squares routine *CRILSQ* to implement *riding atom* model refinement.

4.8.2. Calculation Types

The geometric arrangements of the nine types of calculations permitted are:

4.8.2.1. linear

Given atoms P1 and P2, atom P3 will be generated at a distance D from P2, with a bond angle P1-P2-P3 of 0.0° . Note that P3 lies in the same direction from P2 as does P1. This calculation type is useful in the generation of hydrogen-bonded hydrogen atoms.

4.8.2.2. trigon

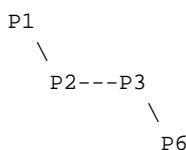
Given atoms P1, P2, and P3 with bond angle ϕ , atom P4 will be generated at a distance D from P2, with bond angles P1-P2-P4 and P3-P2-P4 of $(360-\phi)/2^\circ$. If $\phi = 120^\circ$, the geometry about P2 will be trigonal planar.

4.8.2.3. tetchn

Given atoms P1, P2, and P3, atoms P4 and P5 will be generated at a distance D from P2 with a bond angle P4-P2-P5 of 109.5° . The plane P1-P2-P3 will be perpendicular to the plane P4-P2-P5. If the bond angle P1-P2-P3 is about 109° , the resultant geometry about P2 will be tetrahedral. This type of calculation can be used to produce the hydrogen atoms of a methylene group (-CH₂-) in a tetrahedral chain.

4.8.2.4. teterm

Given three atoms P1, P2, and P3 with bond angle ϕ , three atoms will be generated, all at a distance D from P3. Initially, P6 will lie in the plane of P1-P2-P3 with a bond angle P2-P3-P6 of 109.47° and a dihedral angle P1-P2-P3-P6 of 180° (an anti-periplanar arrangement). From this initial arrangement it is possible to rotate P6 counter-clockwise by α_1° about the P3-P2 bond.



P4 and P5 will form an angle of 109.5° about P3, and the plane P4-P3-P5 will be perpendicular to the plane P2-P3-P6. This calculation type may be used to generate the hydrogen atoms on a methyl group at the end of a tetrahedral chain (tetrahedral terminal).

4.8.2.5. trterm

Given three atoms P1, P2 and P3 with bond angle ϕ , BONDAT will generate two atoms P4 and P5, attached to P3 at a distance D, and lying in the plane of P1, P2 and P3. The angle P4-P3-P2 will be ϕ and P5-P3-P2 will be $(360-\phi)/2^\circ$. Thus if ϕ is 120° , this calculation leads to a terminal sp² arrangement.

4.8.2.6. median

Given four atoms P1, P2, P3 and P4 such that P1 is attached to P2, P3 and P4, atom P5 will be generated at a distance D from P1 to complete an approximately tetrahedral arrangement of four atoms about P1 with the three angles P5-P1-P2, P5-P1-P3 and P5-P1-P4 all equal. **median** is a better choice than **methyn** for cases where the angles subtended at P1 by P2, P3 and P4 are far from the tetrahedral angle.

4.8.2.7. sqrpln

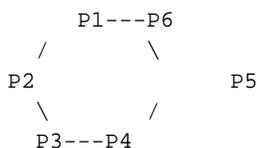
Given three atoms P1, P2, and P3 with bond angle ϕ , atoms P4 and P5 will be generated, each at a distance D from P2. All five atoms will lie in a plane and the bond angle P4-P2-P5 will be ϕ . The bond angles P1-P2-P4 and P3-P2-P5 will be $(180-\phi)$. If ϕ is approximately 90° , the resultant geometry about P2 is square planar.

4.8.2.8. octhed

Given three atoms P1, P2, and P3, atoms P4, P5, P6, and P7 will be generated such that P4 and P5 bear the same relationship to the input atoms as in the **sqrpln** case. P6 and P7 will each be at a distance D from P2. The straight line P6-P2-P7 will be perpendicular to the plane of the other atoms. If the angle P1-P2-P3 is about 90° , the resultant geometry about P2 will be octahedral.

4.8.2.9. hexgnl

Given three atoms P1, P2, and P3, atoms P4, P5, and P6 will be generated such that all six atoms lie in a plane and form a six membered ring. The distances P1-P6 and P3-P4 will be D, and the distance P2-P5 will be 2*D. If the angle P1-P2-P3 is ϕ , then the angles P2-P1-P6 and P2-P3-P4 be $180-(\phi/2)$. If the distances P1-P2 and P2-P3 are approximately equal and the angle P1-P2-P3 is about 120° , the geometry of the six atoms will be hexagonal.



4.8.2.10. genral

Given three atoms P1, P2, and P3, atom P4 will be generated at a distance D from P2. To understand the position of P4 relative to the generating atoms, consider a new Cartesian coordinate system (x', y', z') in which the $x'y'$ plane coincides with the plane of the three generating atoms. The $+z'$ direction is defined by the cross product $(P1-P2) \times (P3-P2)$. The $+y'$ direction is defined by the vector $P3-P2$, and $+x'$ lies in the direction which gives a right-handed system with y' and z' . The projection of the vector $P4-P2$ onto the $x'y'$ plane forms an angle α_1 with the vector $P3-P2$. The cross product $(P3-P2) \times (P4-P2)$ lies in the $+z$ direction. The angle α_2 is the angle between the vector $P4-P2$ and the $x'y'$ plane. If α_2 is between 0 and 180° , P4 will lie above the $x'y'$ plane (i.e., in the $+z$ direction); if α_2 is between 180° and 360° (or 0° and -180°), P4 will lie below the $x'y'$ plane. As an example, consider a chain of three carbon with two hydrogen atoms attached to C2 (viz., H21 and H22) are to be generated using **genra1** calculations, two **CALCAT** lines are needed, one for each hydrogen. The plane defined by C2, H21 and H22 must bisect the C1-C2-C3 angle, so for both H21 and H22 the angle α_1 is 125.25° (i.e., $(360-109.5)/2$). The bond angle H21-C2-H22 will be 109.5° (so that the arrangement around C2 will be tetrahedral) and the plane defined by C1, C2 and C3 will bisect that angle. Therefore, α_2 will be $+54.75$ for one of the hydrogen atoms and -54.75 for the other. Although the generation of hydrogen atoms in a tetrahedral chain can be accomplished more easily with a **tetchn** calculation, the results will be the same if the two **genra1** calculations described here are used instead.

4.8.2.11. methyn

Given four atoms P1, P2, P3, and P4 such that P1 is attached to P2, P3 and P4 with bond angles about P1 all about 109° , atom P5 will be generated at a distance D from P1 to complete the tetrahedral arrangement of four atoms about P1. This calculation type can be used to generate methynyl hydrogen atoms.

4.8.3. General Information

The cell parameters and atomic coordinates must be supplied from the bdf. Generated atoms are added to the end of the atom list in memory if they were not already present in the bdf and may be specified as input for any subsequent **CALCAT** request. If a generated atom was read in from the bdf, the coordinates read in from the bdf are superseded by the newly generated coordinates.

A summary is written after all the **CALCAT** lines have been handled. Listed in the summary, for each generated atom, are the atom name, the generated coordinates, the atom type pointer (referring to *lrscat*: in the bdf), the name of the atoms used in the generation and the bond distance.

Caution: it is important to remember that no check is made to determine whether the set of atoms used in the generation of any new atom(s) is a connected set. If the coordinates in the file for the set of atoms used in some

generation are not the coordinates of a connected set, but rather the symmetrically equivalent coordinates of points that are not connected, the coordinates of the generated atom may not be the desired ones.

4.8.4. File Assignments

- Optionally reads input archive bdf
- Optionally writes output archive bdf
- Optionally writes atom lines to file pch
- In update mode, spawns reentrant ADDATM input file .ad1

4.8.5. Examples

```
BONDAT
CALCAT trigon 1.0 c2 c3 c4 h3
CALCAT trigon 1.0 c3 c4 c5 h4
CALCAT trigon 1.0 c4 c5 c6 h5
CALCAT trigon 1.0 c5 c6 c1 h6
```

The coordinates of all of the hydrogen atoms attached to the ring in salicylic acid may be generated by BONDAT. The input shown above will cause BONDAT to read the coordinates of all of the atoms in the bdf and use them in the calculations. The hydrogen atoms will be located at a distance of 1.0 Angstrom from the carbon atoms to which they are attached and will have the same atom numbers as the carbon atoms to which they are attached (e.g. H3 is attached to c3).

```
BONDAT
CALCAT linear 2.0 s1 c1 h1
CALCAT tetchn 1.0 s1 c3 c4 h31 h32
CALCAT octhed 2.0 c1 s1 c3 p1 p2 p3 p4
CALCAT trigon 1.0 c1 s1 o1 c2
CALCAT hexgnl 1.0 c1 s2 c2 he1 he2 he3
```

4.9. *BONDLA*: Bond lengths and angles

Authors: Wolfgang Dreissig, Ruth Doherty, Jim Stewart & Syd Hall

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

BONDLA provides information about the geometric arrangement of the atoms in the unit cell. This includes bond distances and angles, contact distances, and dihedral angles. The search includes space group and adjacent cell symmetry. The atoms are grouped in connected sets and the groups are optimally clustered.

4.9.1. Atom Connectivity

BONDLA classifies interatomic distances as bonds or contacts or neither. This is determined by the distance relative to the sum of the radii of the atoms involved. The radii that are required for each atom type are:

rcon contact radius
rmin minimum bond radius
rmax maximum bond radius

Radii for each atom type are either read from the bdf, or supplied on **atrad** lines (these supersede the values read from the bdf). Default values are used for any atom types for which one or more of the radii have not been specified. The classification of interatomic distances, $d(A-B)$, is made as follows:

	<i>if</i>	$d <$	$rmin(A) + rmin(B)$	
<i>or</i>	<i>if</i>	$d >$	$rcon(A) + rcon(B)$	not reported
	<i>if</i>	$d \geq$	$rmin(A) + rmin(B)$	
<i>and</i>	<i>if</i>	$d \leq$	$rmax(A) + rmax(B)$	considered a <i>bond</i>
	<i>if</i>	$d >$	$rmax(A) + rmax(B)$	
<i>and</i>	<i>if</i>	$d \leq$	$rcon(A) + rcon(B)$	considered a <i>contact</i>

All input atom sites are grouped for optimal connectivity and the connected groups are clustered about the body centre of the cell, unless the user specifies that the input sites should be used by entering the 'no-cluster' option **nocl** on the **BONDLA** line. Note that the clustered sites are *only* used for the *automatic* generation of bond angles, distances, contact distances and dihedral angles (as specified by the **bond**, **angle**, **cont** and **dih** options), *not* for the distances and angles requested by **bond**, **angle**, **dihed** and **nviron** lines. For these requests, the user has the option of specifying the symmetry operation and the cell translation to be applied to the input *coordinates*. Note that the **nviron** line produces the distances to all atom sites within a specified maximum of the entered atom site.

4.9.2. Printed Output

Control parameters and atom radii are listed first. Check these to ensure that the distance and angle search has been correctly done. The coordinates of the *input* atom sites are then listed. These will be followed by a list of the coordinates of the 'clustered' atom sites, *only if* these are any differences with the input coordinates. These will be followed by any symmetry operations needed to form the connected sites or in the subsequent bond distance searches.

The distances generated from the options **bond** and **cont**, and the **nviron** line will be listed *in the order of generation* unless the **sort** option is entered. This option will cause these distances to be sorted *in order of ascending magnitude*.

The option **dih** will cause the dihedral angle involving all possible connections to be produced. The number of dihedral angles can be large even for modest structures. It may be preferable to use **dihed** lines to request only the angles of interest. The dihedral angle table is not redundant. If an entry appears for the bonded atoms A-B, a second entry will not be given for the pair B-A. Dihedral angles about the pairs A-B will not be included in the table if the coordinates of atom B bonded to atom A are not the coordinates of atom B found for the connected set. This situation can arise only if the connected set of atoms lies on a plane or centre of symmetry. In these cases **dihed** lines may need to be used.

The space group symmetry and cell translation operations used to transform atom sites are listed under the heading **sym trn**. These **sym** numbers refer to the symmetry matrices shown and in the order of input into the

program *STARTX* (see the *STARTX* listing). The `trn` code refers to cell translations where the code 555 refers to a displacement of 0,0,0 (e.g. the code 645 refers to a displacement of +1,-1,0).

4.9.3. Line File Output

Two types of data are output to the line file `pch`. The first is the coordinates and standard deviations of the *connected* atom sites. The second is the distance and angle data in a format suitable for publication purposes. Note that brackets will be inserted into the atom labels if the `brac` option is entered (e.g. the atom name `C23` is listed as `C(23)`).

4.9.4. File Assignments

- Reads *lrcell*:, symmetry and *lratom*: data from the input archive `bdf`
- Outputs `ATOM` lines, distance and angle data to the line file `pch`

4.9.5. Examples

BONDLA

Bond distances and angles of connected sites are listed.

```
BONDLA cont dih
bond C1 C2 2 3 555 566
dihed C2 O4 C5 C7 1 1 3 6
nviron C1 *4 5.0
```

Lists of all distances and angles are generated. In addition, specified bond distances and a dihedral angle are calculated.

4.10. *BUNYIP*: Search for Additional Symmetry

Authors: James Hester and Syd Hall

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

BUNYIP checks for consistent symmetry relationships within the model which are not described by the space group.

BUNYIP (Hester and Hall, 1996) scans the asymmetric set of atom coordinates and looks for the midpoints of any two sites that coincide with a centre (indicating the presence of an inversion), an axis (indicating a 2-fold rotation or screw) or a plane (indicating a mirror or a glide). If a large proportion of the possible midpoint combinations coincide with these symmetry elements within a specified tolerance, the coordinates of the centre, the equation of the line or plane is output along with the pairs of atoms that satisfy this requirement. In the case of the line or plane, additional data on the translation of the connected sites is listed. This provides information about a possible screw or glide translation parallel to the rotation axis or mirror plane.

For inversion centre:

The fractional coordinates x , y , z of the centre are output.

For rotation axis or screw:

The equation of the axis vector is

$$Ax + By + Cz$$

(e.g., the axis along c has C=1.0 and A=0.0, B=0.0)

For mirror plane or glide:

The equation of the vector normal to the plane is

$$Ax + By + Cz = D$$

(e.g., the mirror perpendicular to b has B=1.0; if the mirror is at $y=1/4$, then $D=c/4$ A)

The search is performed in Cartesian coordinates (orthogonal Angstroms) and no assumptions are made about the existing symmetry or cell, apart, that is, from aligning different molecules in the asymmetric unit into the optimally packed cluster. All coordinates are output in fractional units but the values of the deviations (all symmetries), translations (screws and glides) and the value of D (mirror plane equation) are in Angstroms.

Optimal clustering is an important aspect of the midpoint search, especially if molecular complexes of non-identical molecules are involved. It may be necessary in some cases to use the **comp1x** line to ensure that complexed molecules are "linked" identically. This may be required because the search for symmetry does not involve the application of known space group transformations, apart from in the preliminary clustering.

4.10.1. File Assignments

- Read input from input archive bdf
- Writes a file containing the related atom sites and the symmetry on **.bun**

4.10.2. Examples

BUNYIP

4.10.3. References

- Hester J.R. and Hall S.R. *J.Appl. Cryst.* (1996). **29**, 474-478

4.11. **CBAZA**: Add crystal based azimuthal PSI angle

Authors: Doug du Boulay

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

CBAZA requires diffractometer and crystal orientation settings, in conjunction with the diffractometer based PSI angle to calculate and store the Flack and Schwarzenbach, Crystal based Azimuthal PSI angle. This quantity is a prerequisite for LSABS and LSLs anisotropic extinction refinement, but has previously been unavailable to those who do not use the DIFRAC and CIFENT reflection data entry method.

Warning: the *correctness* of this program has not been established!

4.11.1. Operation

This program requires a specification of the diffractometer and crystal orientation following the methods of ABSORB. For each reflection on the archive the Azimuthal PSI angle (measured with respect to reference vectors (k-l,l-h,h-k) or (h,-h,0) if h==k==l) is calculated. The diffractometer specific psi angle from lrrfl: item 1011 is read, if it exists for non bisecting mode reflections, and the calculated crystal based angle is written back as item 1011

4.11.2. References

- Schwarzenbach, D and Flack, H.D. 1989. *J. Appl. Cryst.* **22**, 601-605.

4.12. *CHARGE*: Atomic charges from difference density

Author: Nick Spadaccini, Computer Science Department, University of Western Australia, Nedlands, 6907 WA, Australia

CHARGE calculates a charge associated with an atom from the difference density. The difference density ($\Delta\rho_{mol}$) is partitioned according to the Hirshfeld method.

4.12.1. Hirshfeld Partitioning

The Hirshfeld method apportions the electron density among the atoms by the appropriate weighting. The weights are related by the atomic contribution to the promolecular density,

$$w_A(\mathbf{r}) = \frac{\rho_{atm}^A(\mathbf{r})}{\rho_{pro}(\mathbf{r})}$$

The fragment of the deformation density apportioned to atom A is,

$$\Delta\rho_{frag}^A(\mathbf{r}) = w_A(\mathbf{r}) \Delta\rho_{mol}(\mathbf{r})$$

The net atomic charge Q_A , is derived from the integration of the difference density fragment,

$$Q_A = \int_{input} \Delta\rho_{frag}^A(\mathbf{r}) d\mathbf{v}$$

An alternative scheme is based on the atomic contributions to the total promolecular potential V_{pro} defined as the sum of the electronic and nuclear contributions.

4.12.2. Density And Potential Profiles

The promolecular density or potential is the sum of the atomic densities or potentials. These latter values are derived from the Clementi and Roetti atomic wavefunctions. Associated with each atom type are the

parameters A_k , n_k and z_k for $k=1,\dots,m$ such that the density is,

$$\rho_{\text{atm}}(r) = \sum_{k=1}^m A_k r^{n_k} e^{-z_k r}$$

and the potential is

$$V_{\text{atm}}(r) = 4\pi \sum_{k=1}^m A_k \left(\frac{\gamma(n_k+3, z_k r)}{r z_k^{n_k+3}} + \frac{(n_k+1)!}{z_k^{n_k+2}} - \frac{\gamma(n_k+2, z_k r)}{z_k^{n_k+2}} \right) + \frac{Z_A}{r}$$

the last term is the nuclear contribution and $\gamma(n,x)$ is the Incomplete Gamma Function.

The density profiles (e/bohr³) and potential profiles (e/bohr) are stored at 44 discrete values of r (bohrs) for the points,

$$r_1 = 0. ; r_{k+1} = 1.15(r_k + .01); 0. \leq r \leq 31.2$$

The divisions are chosen so that the density of points is greatest in the region of steepest gradient. The density or potential value at any general point is linearly interpolated from the profile.

4.12.3. Density, Distances And Errors

The difference density ($\Delta\rho_{\text{mol}}$) must be input from the FOURR file `map`. CHARGE partitions this density into atomic contributions and integrates over the input region to obtain a charge. If the input map is the asymmetric unit the values obtained are total atomic charges for atoms at general positions or fractions governed by the site symmetry for atoms at special positions. The user must determine the fraction of an atom present in the input map and calculate the total charge accordingly.

The user may specify the effective range of atom contributions in two ways. The `border` option in the program initiation line determines the region beyond the input map for which atom contributions are included. The default value of 6Å implies that any atom within this distance of the input map edges is included. Also the distance beyond which an atom contribution is zero may be set by the `contact` option.

Estimates of $\sigma(Q)$ are determined for spherical regions of various radii following the method of Davis and Maslen. The estimates are derived from the values of $\sigma(F)$, the errors in the structure factor amplitudes used to derive the difference density. For each atom the program outputs a value of $\sigma(Q)$ for a radius set within the program. The absolute radii are generated from relative radii, derived from the density profiles and stored in the database. These values are rescaled so that the total volume of the atoms in the cell equals the cell volume. The radii used are listed.

Since values of atomic radii are not unique the variation of $\sigma(Q)$ with r is also output so that the user may determine $\sigma(Q)$ at an alternative radius if desired. The scheme assumes a centrosymmetric structure so that phase errors are not included.

4.12.4. File Assignments

- Reads `lrcell`: and symmetry data from the input archive `bdf`
- Reads difference density from a `map` file
- Reads profile data base from the file designated by the macro `profile`:

4.12.5. Examples

CHARGE

The weights here are determined from the promolecular density. The contributions of atoms up to 6Å beyond the input map edges are included in the calculation. However, an atom will not contribute to the density if the point is greater than 6Å away.

CHARGE poten bord 4 cont 4.5

Here the weights are determined from the promolecular potential. Atoms up to 4Å from the input map edges are included in the calculation. The effective range of a atom is set to 4.5Å.

4.12.6. References

- Clementi, E and Roetti, C. 1974. *Atomic Data and Nuclear Data Tables* **14**, 177-478.
- Davis, C.L. and Maslen, E.N. 1978. *Acta Cryst.* **A34**, 743-746.
- Hirshfeld, F. 1977. *Israel Journal of Chemistry* **16**, 198-201.

4.13. *CIFIO*: Input and output CIF data

Author: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia.

CIFIO reads and writes CIF data to and from the Xtal binary or ascii archive file representations.

4.13.1. Purpose

The Crystallographic Information File (CIF) has been developed for electronic submissions to IUCr journals and databases. *CIFIO* converts an Xtal archive file into CIF format and vice versa. The data to be extracted from the bdf is specified by the user by entering data names into a request file `cifreq`. A template `cifreq` file is supplied with the system but this can be copied and modified according to individual needs. *CIFIO* will only recognise requests for data items which have been defined by the core CIF dictionary which is supplied as the file `cifdic`. However, users may construct their own dictionary `cifdic.loc` to handle local data (DO NOT edit the `cifdic` file).

The generated `.cif` file will contain information on all requested data items *that are currently present on the archive bdf*. The unsatisfied requests will be flagged with a question mark '?'. The unavailability of a specific data item may be that data has not been archived on the bdf (as a result of a prior calculation) or that Xtal does not generate this information. The user may use a standard text editor to add or amend a `.cif` file, but caution must be exercised at all times in order not to corrupt the structure of the file. This may be checked by reading the `.cif` file with *CIFIO* with the dictionary checking facility on (ie. enter "`cifio cifin dict`")

CIFIO uses the mapping file `cifmap` which is supplied with the system. This file contains the data that enables each CIF data name to be mapped to the Xtal bdf items. The `cifmap` file should never be changed and a single copy is usually kept in a globally-accessible read-only area of your directory.

CIFIO has four modes: read and write a CIF, read and write and ascii image of the Xtal archive file.

4.13.2. CIF Input

This is the default mode. It will cause the **cif** file to be read and converted into a sequence of line commands in the file **.new**. The file **.new** is automatically read and used to execute a sequence of Xtal calculations needed to create an archive bdf containing the data items in the CIF. If the **dict** option is set the input cif file will be checked against the dictionary files *cifdic* and *cifdic.loc* (optional).

4.13.3. CIF Output

Entering **cifout** will cause CIFIO to output the data values listed in the file *cifreq*. (supplied for *Acta Crystallographica C* submissions) to the file **.cif**. The following controls apply when generating a CIF.

dset n

Specifies which data set is output to **cif**. The default is data set 1.

qchr OR **nqch**

This signals how the character data output by CIFIO is encapsulated. If **qchr** is entered character strings are bounded by single quotes. For example the description of the extinction method could appear as 'Zachariasen Gaussian'. If **nqch** is entered bounding quotes are not used but imbedded blanks are replaced with an underline. The above example would appear as Zachariasen_Gaussian. **nqch** is the default.

sfac OR **nsfa**

The outputting of requested structure factor data is treated specially by CIFIO. This is because a user may wish to use a standard **CIFREQ** file for all CIFIO runs but only generate structure factor data in specific instances where they are required (note that in the future structure factor data may not be an automatic requirement for submission to IUCr journals). Entering **sfac** will cause the requested structure factor data as defined by the **_refln_** data names to output to the **CIF** file. Entering **nsfa**, or as default, these data item requests will be ignored.

scal OR **nsca**

The measured structure factor data (**Frel**, **F²rel** and **Irel**) on the bdf is unscaled. These options signal CIFIO to scale, or not to scale, the measured data which is output to the **CIF** file. The default is **scal**.

corr OR **ncor**

The measured structure factor data (**Frel**, **F²rel** and **Irel**) on the bdf is not corrected for extinction effects. These options signal CIFIO to correct, or not to correct, the measured data which is output to the **CIF** file. The default is **corr**.

4.13.4. arc File Output

The **cif** file discussed above is designed for data communication external to the Xtal System. It contains only selected items that are required by the recipient journal, database or laboratory. CIFIO is able to read a standard CIF (which was not necessarily generated by CIFIO) and create input commands which enable this data to be stored on an Xtal bdf. However, the data on a CIF is users choice (for example, no structure factors) and this limits the universality of recreating a bdf. In other words, even an CIFIO generated **cif** file is not the exact text image of the data on the archive bdf and cannot be used therefore to produce an identical copy of the original bdf.

This is where CIFIO serves a second important archiving function. The **arcout** signal causes an **arc** archive file to be created. This is also a text file in STAR format. The **arc** file differs from the a **cif** file in that it contains *every* data item on the bdf and can be used to recreate an identical copy of that bdf.

The **arc** file has a number of important applications. First, it provides for a machine independent archive file which can be stored and reused with any machine and any future version of the Xtal System. Second, it can be

sent to any other site by electronic means and converted directly into an identical bdf. Third, its contents are completely self-defined and it is suitable as a communication file with other software. Fourth, because ARC is a text file, data can be examined, added to, deleted or modified with a standard text editor. The modified arc can be used to create a new modified bdf (*it goes without saying that such editing must be done with extreme caution*). No additional control options are used to output an arc file.

4.13.5. arc File Input

Setting `arcin` on the `CIFIO` line will cause the arc file to be read and converted to an output archive bdf. This file will be identical to the bdf that was used to create arc.

4.13.6. File Assignments

- Optionally reads the input archive bdf (`cifout` and `arcout` modes)
- Optionally writes an output archive bdf (`cifin` and `arcin` modes)
- Optionally reads a cif line file (`cifin` mode)
- Optionally reads an arc line file (`arcin` mode)
- Optionally writes a cif line file (`cifout` mode)
- Optionally writes an arc line file (`arcout` mode)
- Optionally reads the `cifreq` and line file (`cifout` mode)
- Optionally reads the `cifmap` and line file (`cifout` and `cifin` modes)

4.13.7. Examples

```
CIFIO dict
```

Read the `.cif` file and validate the data items against the dictionaries `cifdic` and `cifdic.loc`.

```
CIFIO cifo sfac ncor
```

Create a `cif` file containing data items from the input archive bdf which are specified in the line file `cifreq`. Structure factor data should be included if requested in `cifreq`, and the measured structure factor data will be scaled but no extinction corrections applied.

4.14. *CIFENT*: CIF entry of diffractometer data

Author: Howard Flack

CIFENT is based largely on *CIFIO*, but with modifications to read *DIFRAC* diffractometer CIF files into the *XTAL* archive.

4.15. **CONTRS**: Contour a map

Author: Mark Spackman, Department of Chemistry, University of New England, Armidale, NSW, Australia.

CONTRS, used in conjunction with *PLOTX*, contours density maps output by either *FOURR* or *SLANT*. Contour plot instructions are written to the `con` file.

4.15.1. Introduction

A regular grid consists of two sets of equi-spaced lines parallel to two axes. The axes need not be orthogonal, nor do the spacings have to be equal for the two sets of gridlines. The intersection points of these gridlines are the mesh-points, and a regular grid of data therefore consists of functional values at all the mesh points of a finite grid. The smallest parallelogram made by four mesh-points is referred to as a pixel, and the mesh-size in each axial direction is the length of the sides of a pixel parallel to each axis.

Output bdfs produced by *FOURR* and *SLANT* contain regular grids of electron density or similar type of data, which are more readily interpreted when suitably contoured. *CONTRS* allows for the contouring of any layer of such data, or even the contouring of successive layers from the input bdf.

CONTRS is specifically aimed towards production of smooth contours. The various steps in construction of the final plot can be summarized as input of the appropriate layer of data, search for intersections of the contour with all gridlines, chaining of points together to form curves, drawing of the curves with optional smoothing, addition of a title, optional drawing of a border, mesh points and atoms in the vicinity of the plane.

4.15.2. Layer Input

Selection of layers is controlled by the *CONTRS* input line. The layer numbers correspond precisely with those output by *FOURR* or *SLANT*. The absence of any layer specification will cause all layers of the input bdf to be contoured. The resulting contour map will have the same orientation as the printed map unless rotation is requested. A single layer is stored in memory at any one time and the outermost loop in *CONTRS* is thus over the layers requested.

4.15.3. Specification Of Levels

All contour levels are specified by the *LEVELS* input line. Three possibilities are allowed for: *arithmic* (constant interval between levels), *geometric* (constant ratio between levels) and *direct* (input of levels explicitly).

For the *arith* and *geom* options, it is necessary to specify a starting level (*start*), a constant increment or ratio between levels (*incr*), and the number of levels to construct from these conditions. For arithmetic intervals, the levels would then be *start*, *start+incr*, *start+2*incr*, ... *start+(nc-1)*incr*, where *nc* levels were requested. For the geometric case the levels would be *start*, *start*incr*, *start*incr**2*, etc.

In the case of *direct* level input, the *direct* flag on the input line is simply followed by the values of the desired levels. If not all such levels can be accommodated on a single *LEVELS* input line, then additional *LEVELS* lines can be input until all levels are entered.

A deficiency of the *geom* option is that it cannot be used by itself to construct a list of levels which span zero and change sign (e.g. 1.0, 0.5, 0.25, 0.0, -0.25). The *negate* flag on the *LEVELS* line will double the number of levels by including all those of opposite sign. In addition the zero level can be explicitly included via the *zinc1* flag. *negate* and *zinc1* function after all line input has been processed, and hence caution should be exercised when using these options with the *arith* or *direct* options.

A maximum of 30 levels is allowed by *CONTRS*, but it is usually found that when the number of levels exceeds 15 the maps tend to become cluttered.

4.15.4. Search For Points On The Contour

Each gridline of the layer of data (across and down) is scanned to find all intersections of the contour with gridlines of the layer. These intersections are stored in the order found as coordinates relative to the grid axes.

The current contour value is subtracted from the function value at each mesh-point in turn. Should this result in a numerical zero, a small amount is added to the difference, which has a negligible effect on the resulting map, but insures that no intersections occur at mesh points. Adjacent mesh points exhibiting opposite signs for this difference clearly must bracket an intersection.

The precise location of the intersection may be determined by either linear or cubic interpolation. For the linear case, only two adjacent mesh points are needed, but the cubic requires four neighbouring points which are generally chosen so that the intersection lies between the two central points. For pixels on the edges of the grid this is not possible, and the closest four adjacent points are chosen.

The interpolation over the grid is not a time consuming step in contouring, and cubic interpolation, being only slightly slower and far more accurate, is the program default. Linear interpolation should rarely be necessary, unless as a convenient check that the cubic algorithm is finding all of the points.

4.15.5. Chaining Of The Points

Two essential assumptions are made by CONTRS in chaining together successive points to form a curve. The first is that all intersection points lie on the edges of pixels, and never at the mesh-points. This is guaranteed by the minor perturbation of the difference function at the mesh-points described earlier. The second assumption is that only one intersection is found on any edge of a pixel. This is necessarily true with linear interpolation, but is not necessarily the case for cubic interpolation. However, it is extremely unlikely that this will ever happen if adequate mesh-sizes are employed (typically < 0.20 Angstroms), and has never been observed by the author in hundreds of maps.

With these assumptions it is a relatively straightforward matter to search for linkages from one point to another. CONTRS simplifies matters by constructing curves which begin on the edge of the map first. Such open curves must end on an edge also, and searching for them first avoids the creation of curves which are obviously incomplete.

The procedure for chaining is quite simple. The unchained list of points is repeatedly searched from bottom to top until a linkage is found, in which case it is swapped with the point at the bottom of the unchained list and the search is repeated with one fewer point. If no linkage is found, a curve is considered to be complete, and its end point flagged for subsequent curve drawing. Notice here that a contour will comprise several closed and open curves on a typical map.

For linkage purposes no use is made of the previous course of the curve. Rather the obvious requirement, that if a contour enters one edge of a pixel it must exit via one of the other three, is the only criterion used. This means that there are either one, two or three possible links from each point. If several links are found, it is not obvious which is the appropriate one, except that CONTRS avoids the case where curves would cross each other. In such cases CONTRS makes a decision based on which choice leads to the maximum separation between curves inside the pixel, since this avoids the presence of absurdly narrow goose-necked contours. It should be borne in mind, however, that this decision is quite arbitrary and based on aesthetics, where information about the function inside the pixel is missing.

Chaining is the most time consuming part of the contouring procedure and, since it is a sorting process, takes increasingly longer as the number of points per contour increases. Thus for grids with large numbers of mesh points along both axes, and for zero contours on a difference map, several thousand points may have to be searched repeatedly. However in most cases only several hundred points are obtained and the chaining procedure is very fast.

4.15.6. Drawing Of The Curves

At completion of chaining, the list of unchained points has been transformed into a list of contiguous points on curves, with stored pointers to the end of each curve. The drawing of straight lines between contiguous points on the curves is a trivial exercise, and where a large number of curves are to be drawn or appearance is not important, this is a sensible option.

Drawing smooth curves is also straightforward. CONTRS employs an isotropic four point smoothing algorithm due to Midgley (1979), which makes the direction of the curve at any data point that of a circle through the point and the two adjacent points. This algorithm can be described as a cubic parametric representation in two dimensions of the curve between points r_i and r_{i+1} ,

$$r_t = r_i + At + Bt^2 + Ct^3$$

where t ranges from 0 to 1 and the constants A , B and C are determined by the adjacent points on the curve. Separate conditions apply to the terminal points of closed and open curves.

The number of discrete increments in t is governed by the resolution requested on the CONTRS input line. This resolution is closely related to the longest straight line segment used in constructing smooth curves, and is obviously relative to the internal dimensions of the contour map. CONTRS scales all maps to 10 units in the y (down the page) direction, and hence the default resolution of 0.10 is 1 of this dimension. Increasing this value, to 0.20 for example, will reduce the number of plot instructions written to the `con` file, will hardly be noticed on a high-resolution dot-matrix printer, but any high quality plotting device will display obvious straight line segments.

4.15.7. Types Of Lines Drawn

CONTRS allows for the drawing of both solid lines and dashed lines with regular dash-gap sequences. The assignment of line type to contour level is performed by the `lintyp` input line, although a default of positive contours solid, zero dashed and negative levels dotted is generally applicable. A distinction is made internally between `solid`, `dashed` and `dotted` lines. Dash and gap lengths are given relative to a y -dimension of 10 units as above, and defaults for these are 0.2 for dash and gap lengths of `dashed` lines, and 0.08 for dash and gap lengths of `dotted` lines. Alterations to these characteristics are possible via the `dash` and `dot` input lines, which allow for up to 10 different elements in such a sequence, enabling the production of fancy chain-dotted lines if desired.

4.15.8. Border And Mesh Points

The border of the contour map can be optionally deleted. The border produced by CONTRS is quite independent of that produced by `PLOTX`, and in fact the border should be suppressed in `PLOTX` to avoid any confusion. Mesh-point locations can be optionally plotted as small crosses on top of the final map.

4.15.9. Atoms In The Vicinity Of The Map

Atoms within a specified neighbourhood of the map will be plotted as large crosses. Atom names of up to four characters can be optionally plotted beneath the crosses. The specification of the neighbourhood is done via the inclusion of the required distance after the `atoms` flag on the CONTRS control line. The distance is interpreted as being in Angstroms and parallel to the layer axis. Thus the total volume of space within which atoms are generated by CONTRS is the parallelepiped formed by the top and bottom layers plus and minus the specified distance respectively. If no distance is specified a default value of half the distance between layers is used. Off plane atoms are plotted in italics.

The `dist` includes the off plane atom distances as a second italic text string. A check is made for exact overlap of two atoms at equal and opposite distances which are printed as a single atom label with the distance string

"+/-xx.xxx". CONTRS flags the out-of-plane atom labels for printing in italics by preceding the label text string with the vertical bar "|".

4.15.10. Printed Output

There are four sections of printed output produced by CONTRS. The first two sections echo control and contour levels input.

The third section lists contour levels found for each layer, and tabulates along with the contour level the line type, the total number of points found for the level, the number of points purged and the number of individual curves comprising the contour. The number of points purged should always be zero except in very special cases (e.g., a single isolated point at the centre of a peak or hollow), or perhaps where the chaining algorithm is failing. This column is an excellent diagnostic and the presence of any non-zero entries should be carefully investigated.

The fourth section summarizes atoms found within the specified vicinity of the contour map, layer by layer, giving the full name, the number of the symmetry operation which generated the particular atom, the distances in Angstroms (parallel to the border axes) from the top left corner of the map to the atom projected onto the plane, and the distance in Angstroms (parallel to the layer axis) of the atom from the layer.

4.15.11. Monochrome Graphics Output

On conventional monochrome graphics devices (e.g., dot matrix printer or vector pen plotters)CONTRS will produce the map and additional output appropriately scaled to the device by *PLOTX*. Two lines of title are added below the contour map. The first line gives the compound ID, the lengths of the x-axis and y-axis (across and down the map) in Angstroms, the angle between axes at the lower left of the map and the layer number(s). The second line reproduces the title submitted via the `title` control line if one was included in the input stream.

4.15.12. Colour Graphics Output

CONTRS is currently capable of producing colour graphics output, provided the appropriate device dependent software is incorporated in *PLOTX*. CONTRS supports six separate colours, and these are assigned to the six groups of output: negative contours, zero contours, positive contours, the border and mesh points, the crosses for atomic sites, the titles and atom names. This colour coding is completely transparent to the user and will not affect output on monochrome devices. Assignment of specific colours to each colour group is performed via *PLOTX*.

4.15.13. File Assignments

- Reads *lrcell:*, *lrsymm:* and *lratom:* data from the input archive bdf ListItem>
- Optionally reads FOURR map points from a `map` file
- Optionally reads SLANT map points from a `sla` file
- Writes contour plot instructions to the `con` file

4.15.14. Examples

```
CONTRS flay 16 llay 16 atoms 2.0
LEVELS arith nc 7 st -0.3 in 0.1
```

In this example, layer 16 of the Fourier map is to be contoured. The levels requested are -0.3, -0.2, -0.1, 0.0, 0.1, 0.2, 0.3, appropriate to a difference Fourier map. All atoms within 2.0 Angstroms of the map will be plotted.

```
CONTRS slant nn
LEVELS direct -0.3 -0.2 -0.1 negate zincl
LINTYP do 1 3 da 4 4 so 5 7
dash 4 0.2 0.2 0.1 0.1
```

In this example, *SLANT* input is specified. The **LEVELS** line will produce identical levels to the previous example. The **lintyp** line specifies explicitly what the default assignment of the line types would be. The **dash** line alters the default dashed line to a chain-dotted line (long dash, short dash, long dash, etc.). Since, in general, *SLANT* writes only a single layer of data in the `s1a` file, *CONTRS* will produce only a single map. Atom sites, but not names, will also be plotted.

```
CONTRS linear nsmooth
LEVELS arith nc 10 start 0.0 incr 1.0
```

In this case all layers are to be contoured sequentially, with linear interpolation and no smoothing. The levels requested are appropriate to a Fourier of the total electron density.

4.15.15. References

- Midgley, J.E. 1979. *Isotropic Four-Point Interpolation*. Computer Graphics and Image Processing **11**, 192-196

4.16. *CREDOC*: Check cell choice

Authors: Y Le Page & H D Flack

Contact: H D Flack, Laboratoire de Cristallographie, University of Geneva, CH-121 1 Genève 4, Switzerland.

CREDOC undertakes the calculation of cell reduction using the algorithm of Le Page (1982). The essential objective of cell reduction is the recognition of the metric (lattice) symmetry of the input unit cell. Once this metric symmetry has been identified, the basis vectors of the lattice may be chosen to lie along the main symmetry direction in the conventional way.

4.16.1. Cell Reduction

The Le Page algorithm is based on the search and identification of twofold axes of the translation lattice. The search is carried out by systematically generating pairs of direct and reciprocal lattice rows. In the normal case, the maximum values of the indices of these direct and reciprocal lattice rows may be limited to a value of two by working from a primitive Burger reduced cell generated from the input cell. For each direct-reciprocal row pair, the vector product is used to calculate the angle between the rows and the scalar product is also evaluated. The two rows are parallel or nearly so when the angle between them is zero or close to zero, and parallel rows are a twofold axis of the whole translation lattice if their scalar product has a value of 1 or 2. Direct-reciprocal row pairs satisfying these conditions (i.e. having an inter-row angle less than a specified value) are stored in a list containing possible twofold axes.

The point symmetries of the seven crystal systems have characteristic distributions of twofold axes in space. Hence comparison of the distribution in space of the twofold axes found in the systematic search with the well-known distribution of the twofold axes of the seven crystal systems allows the lattice symmetry to be identified. This process places the lattice in a conventional orientation and the choice of rows to be selected as edges of the conventional cell is guided by the coincidence of lattice rows with pre-determined symmetry axes. The search for the crystal system takes place in the order cubic, hexagonal, rhombohedral, tetragonal, orthorhombic, monoclinic and triclinic. Each time a solution is found, all the twofold axes with an inter-row angle greater than or equal to the largest one used in the match are deleted from the list of twofold axes and then another solution is sought.

The first input parameter allows the maximum acceptable angular deviation between the direct and reciprocal lattice rows defining a twofold axis to be set. The angular deviation of each direct-reciprocal row pair as calculated from its vector product is compared against the maximum-acceptable value and the pair is retained if the angle is less than the input value. True twofold axes have values of the angular deviation very close to zero. Hence large values of the angular deviation will find cells having pseudo-symmetric lattices. The second input parameter fixes both the maximum acceptable value of the scalar product of the direct and reciprocal row pairs and the maximum value of the indices in the direct-reciprocal row search. Scalar products of 1 and 2 identify twofold axes causing all lattice points to find a rotation-related partner. Values of the scalar product larger than two identify twofold axes which cause only subsets of the lattice nodes to be superimposed. Consequently supercells of higher lattice symmetry can be identified. This is an undocumented and very interesting option which existed in the original Le Page programme.

4.16.2. Twin-Law Evaluation

The (pseudo)-merohedral twin laws are evaluated by the algorithm of Flack (1987). The basis of this method is a coset decomposition of the point symmetry of the declared crystal space group with respect to the lattice symmetry determined from the cell reduction process. The programme produces a list of twinning operations (described both in direct and reciprocal space) which make the lattices nodes of the twin-related lattices superimpose. The algorithm is arranged in such a way that for a centrosymmetric space group all of the twin laws are generated, whereas for a non-centrosymmetric space group each twin law listed (and saved in the *ltrwin:*) corresponds to two twin laws i.e. the one listed and one related to it through an inversion in the origin.

4.16.3. File Assignments

- Reads data from the input archive bdf.
- Writes data to the output archive bdf.

4.16.4. Example

```
STARTX
cell      5.657      5.657      6.828      90. 90.      120.
sgname    R 3
celcon    C 8
CREDUC
```

The output will be as follows:

```
Maximum acceptable obliquity: 3.0 degrees
Maximum multiple cell index: 2
Input cell:
```

```

5.657          5.657          6.828          90.000          90.000
Lattice mode: R
A Buerger cell:
3.981          3.981          3.981          90.555          90.555
The input-to-Buerger cell matrix:
-.3333          .3333
-.3333          -.6667          .3333
.6667          .3333          3333
Possible 2-fold axes:
      Rows
Direct          Reciprocal          Products          Dot          Vector
0 0 1          0 0 1          1          .788
0 1 -1          0 1 -1          2          .000
0 1 0          0 1 0          1          .788
0 1 1          0 1 1          2          .788
1 -1 0          1 -1 0          2          .000
1 0 -1          1 0 -1          2          .000
1 0 0          1 0 0          1          .788
1 0 1          1 0 1          2          .788
1 1 0          1 1 0          2          .788
Pseudo cubic          P          Max delta          .788
A = -.3 -.7 .3          3.9809          Alpha = 90.555          A* = .000 -1.000 1.000
B = -.7 -.3 -.3          3.9809          Beta = 89.445          B* = 1.000 .000 -1.000
C = .3 -.3 -.3          3.9809          Gamma = 89.445          C* = 1.000 -1.000 -1.000
Metrically hexagonal          R          Max delta          .000
A = -1.0 -1.0 .0          5.6570          Alpha = 90.000          A* = .000 -1.000 .000
B = 1.0 .0 .0          5.6570          Beta = 90 .000          B* = 1.000 -1.000 .000
C = .0 .0 1.0          6.8280          Gamma = 120.000          C* = .000 .000 1.000
(Pseudo-)Merohedral Twinning Operations
=====
If the space group is non-centrosymmetric each operation below
impliestwo - i.e. the given operation multiplied by +1 and -1
Twin x = ( 1.00 .00 .00)x :          Twin h = ( 1.00 .00 .00)h
Twin y = ( .00 1.00 .00)y ;          Twin k = ( .00 1.00 .00)k
Twin z = ( .00 .00 1.00)z ;          Twin l = ( -.00 .00 1.00)l
Twin x = (-1.00 .00 .00)x ;          Twin h = (-1.00 -1.00 .00)h
Twin y = (-1.00 1.00 .00)y ;          Twin k = ( .00 1.00 .00)k
Twin z = ( .00 .00 -1.00)z ;          Twin l = ( .00 .00 -1.00)l

```

In the above example a structure in space group $R\bar{3}$ is input on hexagonal axes. The input cell is first reduced using the Burger algorithm. Next a systematic search is taken over all pairs of direct and reciprocal rows up to a maximum index of two. Those pairs which both have an angular deviation of less than 3° and a scalar product of 1 or 2 are retained. These directions are (pseudo) twofold axes. The distribution of the twofold axes in space allows the metric (lattice) symmetry to be identified. The 9 (pseudo) twofold axes have the correct distribution for a lattice of cubic symmetry but as the angle between some of the direct and reciprocal rows is as large as 0.788° , this is only a pseudo-cubic primitive cell. Having eliminated the 6 pseudo twofold axes with an angular deviation of 0.788° , the programme considers the remaining three pure twofold axes and finds that they have the correct spatial distribution for the trigonal system on rhombohedral axes. As a conventional choice a triply-primitive cell (in the obverse setting) on hexagonal axes is generated. In all cases of pseudo or metrically-exact cell, the matrices describing the relation between the input and reduced direct and reciprocal cells are output. Finally the twin laws are calculated and the twin-law transformation matrices are output both for direct and reciprocal space. In the example, the first twin law is the identity operation, the second being a twofold rotation about $[2\ 1\ 0]$. As the declared space group is non-centrosymmetric, these two twin operations imply two further inversion-related ones: *i.e.* inversion in the origin and reflection in $(1\ 0\ 0)$.

4.16.5. References

- M.J., Burger (1960). *Z. Kristallogr.* **113**, 52-56.
- H.D. Flack (1987). *Acta Cryst.* **A43**,564-568.
- Y. Le Page (1982). *J. Appl. Cryst.* **15**, 255-259.

4.17. *CRILSQ*: Structure factor least-squares refinement 2

Author: Roeli Olthof-Hazekamp, Laboratorium voor Algemene Chemie, Padualaan 8, 3508 TB Utrecht, The Netherlands

CRILSQ is a general purpose atomic parameter refinement program. The program can be used for any space group and any setting. It adjusts the scale, extinction, positional, thermal displacement and site occupancy factors by minimizing the difference between observed and calculated structure factors. Refinement is based on F^2 . Derivatives are only accumulated in full matrix mode. Provision is made for the control of parameters of atoms in special positions and refinement of the Flack absolute parameter XABS. The riding atom model and twin refinement are also supported.

4.17.1. Introduction

A summary of the method of establishing least-squares equations which are used in *CRILSQ* can be read in the Program Description of *CRYLSQ*. A number of the features of *CRYLSQ* are not implemented. The differences between the two programs are indicated below. New features are the refinement of twin fractions and the calculation of the riding atom model. Most parameters must have been written to the bdf by preceding programs. Their default settings are shown below. 'Apply and refine' flags may be set in the *CRILSQ* line. Refinement may be shut off by *NOREF* lines.

4.17.2. Calculations Performed

4.17.2.1. Least-squares Procedure

$Y(o)$ and $Y(c)$ are 'observed' and 'calculated' values of F^2 .

4.17.3. General Structure Parameters

4.17.3.1. F relative Scale Factor

A unit *Frel* scale factor, set by *ADDATM*, is applied and refined.

4.17.3.2. Overall Thermal Displacement Factor

The overall isotropic displacement parameter may be set by *ADDATM*. It may be applied and refined by setting 'ov' in the *CRILSQ* line. In that case no individual atom displacement parameters are taken into account.

4.17.3.3. Dispersion Parameters

The dispersion parameters are set by STARTX. They are applied, unless a 'nd' flag is present in the CRILSQ line.

4.17.3.4. Extinction Parameter

The Zacchariasen formalism for extinction is implemented. Use ADDATM for input of the isotropic extinction parameter (g) and 'ex' as 'apply and refine' flag.

4.17.3.5. Absolute-Structure Parameter

The Flack absolute configuration parameter, if used, must be set by ADDATM. Including 'ab' in the CRILSQ line flags the 'apply and refine' instruction.

4.17.4. Atom Specific Parameters

4.17.4.1. Positional Parameters

By default all positional parameters are refined.

4.17.4.2. Thermal Displacement Parameters

The mode of the thermal displacement parameters is set by ADDATM. An 'editu' line in that program may convert the TDP mode.

4.17.4.3. Occupancy Factor (Population Parameter)

The refinement flag is turned on by 'oc' in the CRILSQ line.

4.17.5. Blocking

Refinement of all parameters in one block (full matrix refinement) is the only option in this program.

4.17.6. Invariant Parameters

The NOREF line may be used for the general parameters SKF, UOV, EXT, ABS and TWI and for the atom specific parameters X, Y, Z, U, U11, U22, U33, U12, U13, U23, UIJ, POP.

4.17.7. Dependent Parameters

The form of the CONSTR lines is the same as in CRYLSQ. The order of the different CONSTR lines is not important.

4.17.8. Riding Atom Model

Hydrogen atom positions should be precalculated in other programs, such as PIG or BONDAT or manually inserted using ADDATM `link` lines. These programs add H atom link information on to the archive bdf which is read by CRILSQ and used to invoke the riding model. Only H atoms for which linkage has been generated will be treated using the riding model, all other atoms will be refined in the usual manner. The positional parameters of the generated atom is recalculated after each refinement cycle. The isotropic thermal displacement parameter is 1.2 times Uequivalent of the atom to which the new atom is attached.

4.17.9. Twin Refinement

The twin laws for (pseudo-) merohedral twins will be read from the bdf if placed there by CREDUC, or may be entered by use of 'twin' lines. Up to 8 volume fractions may be refined. Their sum is constrained to 1.0. A 'noref twin' line will apply the volume ratio's, but not refine them. Twin laws should form a sequential, closed group in the sense that: twinop 1 = A1 =I twinop 2 = A2 and A2*A2 = I twinop 3 = A3 and A3*A3 = A2 twinop 4 = A4 and A4*A4 = A2(T)

4.17.10. Final notes

Not implemented (yet) are 'Rigid Groups And Identical Molecules', 'Restrained Parameters' and 'Fixed Atom Contribution'. The other features are identical to those in CRYLSQ.

4.18. *CRISP*: Well baked structural solution

[Caveat: This program should not be confused with the program CRISP marketed by Calidris for image processing of electron micrographs]

Authors: Doug du Boulay & Syd Hall, Crystallography Centre, University of Western Australia, Nedlands, WA 6907, Australia

CRISP combines tangent direct methods with iterative real to reciprocal space transformation methods to determine organic or organometallic molecular structure. This program is numerically very intensive and combines Fourier calculations with block diagonal least squares structural refinement as a basis for excluding or incorporating density peaks as real atomic sites in the model. It relies on accurate atomic resolution reflection data, and an extremely good guess at the actual cell contents.

4.18.1. Solution Method

CRISP uses the GENSIN structural invariants and generators as a basis for phasing generator reflections using tangent direct methods. Random starting phases are assigned to all generator reflections and, in GENTAN terminology, phases are refined in BLOCK mode, for up to 30 cycles. By default CRISP refines 64 random phase sets, logging the best 12 phase sets (according to their refinement figs of merit) for possible further analysis. By default only the first (i.e. best) of those 12 is examined further for molecular geometry.

CRISP relies on the user to specify (in **STARTX**) a very good approximation of the expected unit cell contents using the **celcon** lines. This is quite crucial, especially so for heavy atoms. Hydrogen atoms are essentially ignored. Generally it is thought unlikely that no information about atomic contents is available. In addition BDF archive should contain Frel values and E values.

The phases and E values of the phased generators are used to calculate a fast Fourier transform emap at 4 point per angstrom resolution. The Emap is searched for the expected number of map peaks, which are ranked according to peak height. Typically a volume larger than the asymmetric unit is generated, but symmetry related peaks are identified and rejected.

The ranked peak list is used as a basis for assigning atom types. If heavy atoms are expected, then those atoms are broken up into different ranges according to their atomic number. In the first instance, only the largest Z value atoms are assigned to the top peaks. (One shortcoming of this algorithm is that the top map peaks do not necessarily correspond to the heaviest atoms. A second shortcoming is that this algorithm does not take into account the fractional scattering power of the expected atoms, so for instance if there is one S atom and 200 C atoms the likelihood of successful structure determination is very low if only 1 S atom is assigned in the first instance)

With the top peaks assigned to the heaviest atoms, the scale factor and overall temperature factor are least squares refined (5 cycles), initialized with values determined from GENEV. If the new refined values are beyond certain limits, it is an indication that the atom type assignment maybe incorrect, or relatively incomplete and the scale and overall temperature factor revert to their GENEV values.

Subsequently the atom site x, y, z and individual isotropic temperature factors are refined using 5 cycles of diagonal least squares (using Frel-Fcal). Positional constraints are applied to sites on symmetry positions. The refined temperature factors are used as to critically examine the assigned atomic types. If the temperature factor is too high an atom type may be demoted to a lower atomic number atom or rejected altogether, whereas if it is too low it can be promoted to the next higher atomic number amongst the expected atom types. Atom type promotions are not invoked until later iterations when C atoms are being actively sought, that is unless the refined temperature factor for an individual site becomes ≤ 0 .

Also in later iterations additional peaks beyond the number expected based on the celcon lines are added, in batches and rerefined. Between the addition and rerefine, the scale factor and overall temperature factor are refined also.

If no more peak sites are or can be added, the current structural solution is used as a basis for calculating new structure factor phases for use with Frel values for Fourier transformation to an Frel map. During early iterations, this may be only for reflections with $E > 1.0$ whereas for later iterations all reflections with $F > 2\sigma$ are used. The frel map is known to exhibit significant series termination ripple, predominantly around the heavy atoms, but these spurious map peaks are relatively localised and are largely excluded from any downstream processing based on geometrical (bond-radii) constraints. The advantage of using an Fmap at this stage is that the peaks recovered from this new map have peak heights much more in keeping with the atomic numbers of the expected atomic types. In addition, the new peak list is more readily matched with the results of previous iterations. At this stage the next lower level of heavy atoms (if any) is assigned and the refinements of scales, coordinates and temperature factors is repeated. If the last atomic number range has already been invoked then a final iteration is made for solution comparison purposes.

After a number of interative cycles corresponding to the number of atom ranges + an extra comparison cycle, it is hoped that the peak assignment and refinement will have converged to the correct molecular solution. The resultant solution is saved to a **.sol** file. In default mode, if the solution converged (same solution in final and previous cycle) then CRISP will terminate and the molecular viewing program PIG is automatically invoked to view the solution. Ideally only one solution is required, and that solution is most likely to come from the phase set with the best figure of merit.

NOTE: To actually use a particular solution, you have to explicitly ensure that a bdf will be written out in **PIG**, and ensure that the solution you require is currently active as you exit the PIG program. This will write out the current atom sites to the **.add** file for automatic reentrant streaming into ADDATM.

4.18.2. Control options

When a non-default number of **best** phase sets are requested for examination, subsequent solutions are appended to the **.sol** file. The automatic invocation of **PIG** at CRISP termination permits the the interactive **NEXT**, **PREVIOUS** controls to be used to step forward and back through the requested solutions (up to 200). When PIG is used in this mode, by default IT WILL NOT write an output archive file containing the solved structure. This must be requested explicitly from within PIG.

The `optim n` control card on the CRISP line controls the level of optimization to which the required structural solutions should be studied. At the zeroth level of optimization `n=0` only the initial emap peaks are written to the `.sol` file. The number of peaks is still governed by the `STARTX celcon` lines.

At the first level of optimization `n=1`, the `STARTX celcon` contents are imposed as a whole, on the emap peak list and refined. There is no attempt at an iterative solution.

At optimization level `n=2`, the `STARTX celcon` contents are iteratively imposed on successive peak lists, with successively lighter heavy atom ranges added at each iteration. There is no attempt to look beyond the input cell contents for extra unexpected sites.

The final optimization level, `n=3`, is identical to `n=2`, except that the residual peaks in the peaklist are trialed to test if for unexpected atom type numbers (but only for the C atom range). This additional testing is quite time consuming.

4.18.3. Caveats

CRISP has been found to work reasonably well for centrosymmetric structures but less well for acentric structures. For some structures with strong pseudo symmetry, direct methods cannot be used effectively and other techniques such as patterson searches may prove more useful. CRISP automatically invokes GENSIN and GENEV if the structure invariant file `.inv` is missing. Sometimes it may be necessary to use more than the default number of invariants and in that case GENSIN should be run manually with a reduced Emin value or an explicitly larger number of triplets should be requested.

4.18.4. File Assignments

- Reads reflection data from the input archive `bdf`
- Reads invariants from the invariant file `.inv`
- Writes structural solutions to direct access file `.sol`

4.19. *CRYLSQ*: Structure factor least-squares refinement

Author: Roeli Olthof-Hazekamp, Laboratorium voor Algemene Chemie, Padualaan 8, 3508 TB Utrecht, The Netherlands

CRYLSQ is a general purpose atomic parameter refinement program. The program can be used for any space group and any setting. It adjusts the scale, extinction, positional, thermal displacement, and site occupancy factors by minimizing the difference between observed and calculated structure factors. Refinement may be based on F , F^2 or I . Provision is made for the control of parameters of atoms in special positions; and the refinement of rigid groups or identical molecules; refinement of the Flack absolute structure parameter XABS; and restraints involving bond distances, angles and dihedral angles. Derivatives are accumulated as full matrix, blocked matrix, or diagonal matrix at the user's discretion.

4.19.1. Calculations Performed

The following equations give a summary of the method of establishing the least-squares equations which are used in the CRYLSQ program. The user is referred to a standard crystallographic text for the background to these calculations (Buerger, 1960). The methodology follows closely that which has been used by Busing and Levy (1959), among others. The main enhancements are concerned with the automation of the handling of special positions, matrix blocking, rigid groups, and identical molecules.

Least-squares Procedure

In the least squares procedure the function Q is minimized:

$$Q = \sum_h [w(Y_o - Y_c)^2] = \sum_h [w \Delta^2] \text{ (summed over all hkl)} \quad (1)$$

where Y_o is the 'observed' value of $|F|$ or F^2 or I ; Y_c is the calculated value of $|F|$ or F^2 or I and w is the least squares weight. The calculated structure factor F is:

$$F_c = A_c + i B_c \quad (2)$$

$$A_c = \sum_n [\sum_s \{ f1 \cos\phi - f2 \sin\phi \}] \quad (2a)$$

$$B_c = \sum_n [\sum_s \{ f1 \sin\phi + f2 \cos\phi \}] \quad (2b)$$

where \sum_n is the sum over the atoms; \sum_s is the sum over the equivalent positions. The structure factor components are:

$$f1 = (f + f') K M P \exp(U) \text{ and } f2 = f'' K M P \exp(U);$$

$$\cos\phi = \cos 2\pi(hx + ky + lz) \text{ and } \sin\phi = \sin 2\pi(hx + ky + lz),$$

where K is the reciprocal scale factor; M is the multiplicity; P is the population parameter; $\exp(U)$ is the thermal displacement term; f is the atomic scattering factor; f' is the real part of dispersion factor and f'' is the imaginary part of dispersion factor.

$$\text{Minimization of } Q \text{ gives for each parameter } p_i: Q/dp_i = \sum_r [w \Delta dY_c / dp_i] = 0 \quad (3)$$

Since the structure factor is not a linear function of the parameters, the function must be expanded in a Taylor series: $Y_c = Y_c(0) + \sum_j [\epsilon_j dY_c / dp_j] + H$ (4) where \sum_j is the sum over the parameters; $Y_c(0)$ is the structure factor calculated from start parameters; ϵ_j is the difference between real and start value of parameter p_j and H is the higher order terms (neglected in CRYLSQ).

The normal equations are obtained by substituting equation (4) in equation (3): $\sum_r [wY_o - Y_c(0) - \sum_j \epsilon_j Y_c/dp_j] dY_c / dp_i = 0$ (5) which can be written as $\sum_r [w \sum_j \{ dY_c/dp_i \epsilon_j dY_c/dp_j \}] = \sum_r [w \sum_j \{ dY_c/dp_i \Delta \}]$ (5a) $\sum_j [A_{ij} \epsilon_j] = B_i \Delta$ (5b) or in matrix notation $A \epsilon = B \Delta$ (5c) where

$$A_{ij} = \sum_r [w dY_c/dp_i dY_c/dp_j] \text{ and } B_i = \sum_r [w dY_c/dp_i \Delta]$$

The formulae of the various derivatives are given in the section on derivatives. The shifts, ϵ_j , are calculated from this equation. Because the normal equations are approximated functions, the calculations do not lead to the true values of the parameters. However, the shifts can be used to correct the starting values. The new parameters can be used as starting values in a subsequent refinement cycle.

4.19.2. General Structure Parameters

4.19.2.1. F relative Scale Factor(s)

F-relative scale factor(s) are refined as reciprocal scale factor(s) SKF. When groups of reflections have different scale factors, these can be refined separately or as a unit.

4.19.2.2. Pseudo-overall Thermal Displacement Parameter

The overall displacement parameter is used to calculate a measure of the correlation between scale factor(s) and displacement parameters. The scale factors and displacement parameters of a structure are always correlated. If they are refined in separate blocks, incorrect shifts will result unless the correlation is taken into account. By refining the pseudo-UOV in the same block as the scale factor, a compensation is made for the correlation. Individual displacement parameters must also be adjusted based upon the shift in the pseudo-UOV. They are corrected with:

$$BCORR = \epsilon(UOV) - \epsilon'(UOV) \quad (6)$$

where $\epsilon(\text{UOV})$ is the shift obtained from an n by n matrix with SKF and pseudo-UOV, and $\epsilon'(\text{UOV})$ is the shift obtained from a 1 by 1 matrix with the pseudo-UOV only. The $\epsilon(\text{pseudo-UOV})$ is not applied to the overall thermal displacement parameter, when there are no atoms with an overall value to be refined.

4.19.2.3. Dispersion Parameters

Atomic scattering factor dispersion factors are applied in the calculation of the structure factors when this is requested in the **CRYLSQ** line by the **ad** specification. The values are present in logical record *lrscat*: of the bdf or are given in **disper** lines. When specified in both ways, the **disper** line values are used.

Refinement of all dispersion parameters is turned on by **rd** in the **CRYLSQ** line. Using **noref** lines, in combination with **rd** in the **CRYLSQ** line, turns off the refinement of the indicated parameters. In the blocked refinement modes, the dispersion factors are refined in the same block as the scale factor(s).

4.19.2.4. Extinction Parameters

Isotropic extinction parameters may be applied or refined in CRYLSQ. The formalism of Zachariasen can be used, employing the expressions of Larson (1970). Zachariasen introduced two crystal types:

Type I extinction is dominated by the mosaic spread (sometimes referred to as *secondary* extinction). The extinction is parametrised by the refinable quantity g which is related to the mosaic distribution function.

Type II extinction is dominated by the particle size (sometimes referred to as *primary* extinction). The correction is parametrised by the refinable quantity ρ .

The correction y applied to each reflection intensity is $y = (1 + 2x)^{-1/2}$

Details are given in Larson (1970). The above formula does not differentiate between Type I or II explicitly, although values of g or ρ may be inferred by the user, as explained below.

For more severe secondary extinction and for primary extinction the formalism of Becker and Coppens (1974a) may also be used. The Becker-Coppens expression for y involves a higher order expansion given by where $i=s$ (secondary) or $i=p$ (primary). The expressions for a_i , b_i and c_i for primary extinction and for secondary extinction using either Gaussian or Lorentzian distributions are given in Becker and Coppens (1974a). Type I or II may be refined in that formalism. A third type denoted General which represents a mixture of the two types is a further option in which both g and ρ are refined simultaneously. Their high correlation often leads to convergence problems.

For *Type II and General*, *primary extinction* is refined simultaneously since any secondary extinction affected by particle size will have some primary component and be dependent on it. For neutron diffraction the total extinction correction y is

$$y = y_p(x_p) \cdot y_s(y_p(x_p) \cdot x_s)$$

The expression in the x-ray case is functionally similar but includes polarisation terms due to the specimen and the monochromator crystal, as given by Becker and Coppens (1974b) in an appendix.

The primary (type II) extinction variables are:

x_p is not applicable	for ZACH
$x_p = 0$	for TYP1
$x_p = \gamma_p F_c^2 \rho^2$	for TYP2 and GEN

The secondary (type I) extinction variables are:

$x_s = r^* F_c^2 \delta$	for ZACH
$x_s = y_p(x_p) \gamma_s F_c^2 \psi$	

where

$\psi = g / \sin 2\theta$	for TYP1
$\psi = \rho$	for TYP2
$\psi = \rho / (1 + (\rho \sin 2\theta / g)^2)^{1/2}$	for GEN with a Gaussian distribution
$\psi = \rho / (1 + \rho \sin 2\theta / g)$	for GEN with a Lorentzian distribution

These expressions indicate the strong dependence of the extinction correction on F_c . For given g or ρ , different values of y can be obtained depending on the structure factor model. This is a consequence of the Zachariasen or Becker and Coppens formalism for the *posteriori* correction of intensities for extinction.

For the Zachariasen correction, r^* is equal to the final value of EXT printed by CRYLSQ *multiplied by 10⁴*. That is, if EXT=.2 then $r^* = 2000$. If one assumes that the extinction is Type I then $r^* = g$. If one assumes a Type II extinction then $r^* = \rho$.

The values of g and ρ obtained are dimensionless and relate to the physical properties of the crystal in the following manner,

$\Delta_{1/2} = (\ln 2 / 2\pi)^{1/2} / g$ radians	(= .33214 / EXT x 10 ⁴) for the Gaussian distribution
$\Delta_{1/2} = 1 / (2\pi g)$ radians	(= .15915 / EXT x 10 ⁴) for the Lorentzian distribution, where the half width $\Delta_{1/2}$ of the distribution indicates the mosaic spread
$r = \rho \lambda$	(= RHO x 10 ⁴ x λ) where r is the size of the mosaic blocks and λ the wavelength

4.19.2.5. Absolute-Structure Parameter

The absolute-structure (twin) parameter **xabs** may be used to define the absolute configuration or polarity of untwinned crystals (Flack, 1983). The crystal is described as an inversion twin. $1-x$ and x are the fractions of the structure and its inverse in the macroscopic sample. The structure factor equation is given by

$$|F(h,k,l,\mathbf{xabs})|^2 = (1-x) |F(h,k,l)|^2 + x |F(-h,-k,-l)|^2$$

xabs converges in 2 to 4 cycles, depending on the anomalous dispersion contribution, to values between 0. and 1., within 3 e.s.d.'s. Friedel pair measurements must be present in the bdf for all or part of the reflections. Note that the **packfr1** option needs to be used in SORTRF to pack pairs into a single packet.

4.19.3. Atom Specific Parameters

4.19.3.1. Positional Parameters

All positional parameters for selected atoms are refined unless a **noREF** condition is present to direct that a parameter be held invariant.

4.19.3.2. Thermal Displacement Parameters

The program can be used to refine the values of overall, isotropic, anisotropic or mixed thermal displacement parameters. In the mixed mode these parameters are set in accordance with the specification stored in the bdf. For one or more atom types the mode can be changed by the use of a **dptype** line. In the program, use is made of **u** or **u.i.j**. On the printed output, U values (multiplied by 100) are displayed. After correction, the displacement parameters are tested to be sure of their physical significance. If an isotropic thermal displace-

ment parameter is negative or an anisotropic displacement tensor is not positive definite, a warning is printed. When a displacement parameter is found to be non-positive definite one of three options can be specified in the **CRYLSQ** line -- continue with calculated values (**tw**), reset displacement to positive definite and continue (**tx**), or stop (**tt**).

4.19.3.3. Treatment of Atoms With **uov**

When one or more atoms has an overall thermal displacement parameter and the use of a full matrix is specified, derivatives are calculated with the formulae that is used for the **uov**, but summation is over only those atoms with **uov**. When other block types are used, the summation for **UOV** is over all atoms.

4.19.3.4. Population Parameter

The term population parameter is used to describe the occupancy of an atom site. It is independent of atom multiplicity for the site. Population parameters are applied in the calculation only when they are present in logical record *lratom*: of the bdf. Their refinement flag is turned on by the **pp** code in the **CRYLSQ** line. **noref** lines can be used to turn off refinement of indicated population parameters.

4.19.4. Solution Of The Normal Equations

The normal equations are solved by the method of Choleski. **A** is a positive definite, symmetric matrix. It is decomposed to an upper triangle **U** and a lower triangle **L**. Matrix **L** is the transpose of **U**, so only enough memory to hold half the matrix is required.

$$\mathbf{A} \epsilon = \mathbf{L} \mathbf{U} \epsilon = \mathbf{B} \quad (7)$$

From the decomposed triangle **U** the shifts can be calculated easily.

4.19.5. Estimated Standard Deviations

The estimated standard deviations are derived by the following expression where *n* is the number of reflections and *m* is the number of variables:

$$\sigma_i = \{A_{ii}^{-1} \sum_r [w \Delta^2]/(n - m)\}^{1/2} \quad (8)$$

For the calculation of standard deviations, the inverted matrix is required. It is obtained by inverting the triangular matrix. Because the inversion is a rather time consuming, the standard deviations are calculated only during the last cycle.

4.19.6. Blocking

There are several options for blocking the matrix of normal equations, **A**.

1. All parameters may be refined in one block. This is referred to as full matrix and is specified by **fm** in the **CRYLSQ** line.
2. The matrix may be blocked so that there is one block for the parameters of each atom. This is referred to as block diagonal and is specified by **bd** in the **CRYLSQ** line.
3. The matrix may be blocked in such a way that each block contains the parameters for more than one atom. The option is specified by **b1** in the **CRYLSQ** line. When this specification is made details of the structure of the blocks must be given in **block** lines.
4. The matrix may be divided into blocks containing any arbitrary parameters. This option is indicated by **b1** in the **CRYLSQ** line. Details of the structure of the blocks must be given in **block** lines.

4.19.7. Invariant Parameters

Parameters can be held invariant by two methods:

4.19.7.1. Method 1:

By commands from logical record *lrcons*: of the bdf. These commands are generated by the program ADDATM and specify atoms in special position. See the documentation on ADDATM for details.

4.19.7.2. Method 2:

By a **noref** line. The general form for the invariant parameter set is:

```
(P1/P2)(R1/R2) or (P1,P2)(R1,R2) or any combination of these
P1/P2 parameter symbols P1 to P2 inclusive (see lists below)
P1,P2 parameters P1 and P2 only
R1/R2 atom name's R1 to R2 inclusive
R1,R2 atom name's R1 and R2 only
```

If only an atom name or atom type is present, then all the parameters for that atom or atom type are held invariant. If only a parameter is specified, then that parameter is held invariant for all atoms. More than one invariant parameter set may be input on a **noref** line. In addition to the individual atom parameters the following general parameters may also be specified invariant:

skf	scale factor(s)
uov	overall thermal displacement parameter
ext	extinction parameter(s)
dsp	dispersion factors

The individual atom parameters are: (See the program *ADDATM* for definitions) **x y z u u11 u22 u33 u12 u13 u23 pop app neu**

There exist atom types Y and U. In these cases, parameter symbols and atom types must be present.

```
noref y(c3)           : y invariant for atom c3
noref (x,z)(c2,c5)   : x and z invariant for atoms c2 and c5
noref (x/u11)(c1/c3) : x to u11 invariant from atom c1 to c3
noref u              : u invariant for all atoms
noref c5            : all parameters invariant for atom c5
noref u(h) c        : u for h atoms and all c parameters invariant
```

4.19.8. Dependent Parameters

Parameters may be constrained to be relative to other parameters by: the space group symmetry requirements of an atom sited at a special position, the requirements of definition of the origin in polar space groups or use of the **constr** option as specified by the user. *ADDATM* will detect atom parameters constrained by space group symmetry or polar space group requirements and store the information in logical record *lrcons*: of the bdf.

The general form of a constraint equation on the **constr** line is:

$$\text{par}(s)(\text{atom}(s)) = Q + \text{fct}(1)*\text{par}(1)(\text{atom}(1)) + \dots + \text{fct}(n)*\text{par}(n)(\text{atom}(n))$$

where

par(s)	= subject (dependent) parameter
atom(s)	= subject (dependent) atom name
Q	= constant term
par(1,2,...,n)	= reference (independent) parameter(s)
atom(1,2,...,n)	= reference (independent) atom name(s)
fct(j)	= multiplication factor for corresponding parameter
par(j)	

Note that the subject parameter or atom must not precede the reference parameter or atom in the parameter or atom list. That is,

$y(\text{Cu1})=0.0+2.0*x(\text{Cu1})$	is correct
$x(\text{Cu1})=0.0+0.5*y(\text{Cu1})$	is not correct
$x(\text{Cu2})=0.5+1.0*x(\text{Cu1})$	is correct
$x(\text{Cu1})=0.5+1.0*x(\text{Cu2})$	is not correct

Two examples follow:

```
constr x(c1)=0.5+1.0*y(o2)-0.5*z(c3)
constr pop(a11)=1.0-1.0*pop(Mg2)
```

Though the first example is unlikely it illustrates the general form of a **constr** line input. The use of the **constr** line restricts the value of the x parameter of the atom c1 to be equal to the y parameter of o2 minus half of the z parameter of c3 plus a constant term. The second example is common in mineral structures where stoichiometric disorder occurs at atom sites. The total population of the two atom types at the site must sum to a population of one.

If the input parameters do not satisfy the above equation prior to refinement, they are modified to do so, except when the constant value Q is given as the letter Q. The applied shifts always satisfy the constraint equation.

4.19.8.1. Mathematical Treatment of the Dependent Parameters

In polar space groups the position of the origin in one or more directions is not fixed by symmetry elements. Coordinates in a polar direction can be considered as a special case of constraints. One polar coordinate depends on all others. (The factors are negative 1.) In the block diagonal approximation polarity does not have to be considered, because it will not lead to singularity of the matrices.

The contribution of the dependent parameters to the derivatives of the reference parameters is expressed in the chain rule where $\sum(i)$ is the sum over all parameters dependent on p(j), including the derivative of p(j) itself (Finger, 1969). $dF/dp_j = \sum_i [fct(i) dF/dp_i]$ (9)

The standard deviation of the dependent parameter is calculated according to the approximate formula: $\sigma_{p_i} = \{ \sum_j [\sum_k [fct(j) fct(k) COV(j,k)]] \sum_r [w\Delta^2]/(n-m) \}^{1/2}$ (10)

where COV(j,k) is the covariance of the jth and kth reference parameters.

4.19.9. Rigid Groups And Identical Molecules

The total number of parameters in a structure can be reduced if some subset of the atoms can be considered as a rigid group. Idealized coordinates for rigid groups can be calculated and written to the bdf either by the program *RIGBOD* or by the program *BONDAT*. Another method of reducing the total number of parameters is when a structure contains two or more fragments consisting of "identical" points. These are referred to as identical molecules. Using the identical molecule, two or more groups of atoms in a crystal structure can be forced to maintain the same intermolecular structure during the refinement.

In both cases use is made of the Euler parameters: three coordinates of the centre of gravity and three angles that relate the principal axes of the group to the cell axes respectively, to the principal axes of the first identical molecule. For the Euler angles the Goldstein definition from classical mechanics is used. The calculation of the principal axes is accomplished by the diagonalization of the inertia tensor by the method of Jacobi.

For a non-degenerate inertia tensor three non-collinear atoms are needed in the rigid group. Groups positioned on a symmetry element lead to dependent or invariant Euler parameters. Information on fixed parameters must be given in the **group** line; dependency must be given in **constr** lines for the first atom(s) of the group(s).

Thermal displacement parameters and population parameters can be refined either for the individual atoms or for the groups (for group isotropic displacement parameters only). Dependencies of these parameters for different groups can be maintained by the use of **constr** lines. When there is only one displacement parameter or one population parameter for a group, one **constr** line will suffice (for the first atoms of the groups).

4.19.10. Restrained Parameters

The use of stereochemical data as individual observational equations was first proposed by Waser (1963). In CRYLSQ these equations are referred to as *restraints*. The atom model is restricted to a realistic range of geometries. The weights are inversely related to the standard deviation of the distribution of possible values. The extra term to be minimized by the least squares process is

$$\sum_n w_n (\Phi_n - \Phi_n \text{obs})^2$$

where Φ_n is either the bond distance, bond angle, or dihedral angle; and

$$w_n = \sum_r [w \Delta^2] / (n-m) * 1 / \sigma^2 \phi_n$$

4.19.11. Weighting

The process of refinement of a crystal structure utilizing a nonlinear least-squares procedure requires some attention to be given to the weight assigned to each experimental observation. In the XTAL system, the quantity stored in the binary data file is the square root of the reflection weight (this convention has arisen because it saves many thousands of multiplication operations during the building of the matrix and vector of the normal equations). The practice follows that established in ORXLS (Busing and Levy, 1959) and is common to many least squares programs.

The derivation of the values of the weights is a matter of considerable controversy in the crystallographic literature. For this reason, a number of methods of dealing with the problem of establishing values suitable to the user are provided. The weights to be applied may be specified in three different ways: a weight of 1.0 for all observations (**wu**), a weight of $1/\sigma^2 F$ (**ws**), or the weights supplied in the bdf (**ww**).

4.19.12. Partial Shift Factor

In all block modes a partial shift factor of 0.8 is automatically applied. The factor may be changed by using the code **fu q** in the **CRYLSQ** line. The factor which is used will depend on the correlation between atoms in different blocks. When correlations are high, a small shift factor should be applied to damp the shifts.

4.19.13. Reflection Classification

Poorly measured data points can be treated in a number of ways. A broad category, referred to as less-thans (rcode=2) may be suppressed from the data set by the `tl` `q` option in the `CRYLSQ` line where the threshold of observability is defined in terms of multiples of $\sigma(F, F^2, \text{ or } I)$.

In general, less-thans are not contributed to the least-squares matrix. This can be overridden by the `m1` option on the `CRYLSQ` line which forces all less-thans with a F_c greater than F_o to contribute to the matrix. Only those reflections considered to be more accurately measured (where the $\text{weight} * \Delta(F)$ is less than some user defined quantity `q`) can contribute to the matrix by the `mr` `q` option in the `CRYLSQ` line. When this reclassification is specified, it is only applied during the current calculation. It will not be written to the bdf.

4.19.14. Scattering Factors

Scattering factors are entered from the bdf. If interpolated scattering factors are present on the reflection data record `lrrefl:` (as inserted by `ADDREF`), these will be used. Otherwise, scattering factors are extracted from the form factor tables on `lrscat:`.

WARNING: The scattering factors from the form factor tables are considered sufficiently reliable for routine analyses with R-factors above 0.03. For more accurate analyses the interpolated form factors must be used.

4.19.15. Space Group Symmetry

The CRYLSQ program is general with respect to symmetry. The symmetry operations stored in the bdf are used to generate all the symmetrically equivalent points for the asymmetric set of reflections and atoms supplied. It is important to note that there must be no systematically absent reflections coded as observed reflections (rcode=1 or 2) included in the data supplied.

4.19.16. Fixed Atom Contribution

The number of unique atoms which is used for the calculations may be limited. The structure factor calculation and refinement procedure can be carried out with a selected set of atoms (use the `select` line). Fixed contributions to the structure factors for the remaining atoms must have been calculated in a former structure factor calculation with `FC` or CRYLSQ. By using `ep` in the `CRYLSQ` line the fixed contributions are applied as partial contributions to the structure factor sums and they are stored in `lrrefl:` (IDN 1810 etc.). In subsequent runs, with the same selected atom set, the same fixed contributions may again be applied (`ap` in the `CRYLSQ` line). In this way it is possible to carry out blocks of refinements of various parts of the structure to achieve overall refinement of the complete structure.

4.19.17. Initial Refinement

During the initial steps of refinement it is not always necessary to use all reflections. The number of reflections can be limited by:

1. Defining a $\sin\theta/\lambda$ range in a `maxhkl` line.
2. Assignment of maximum h, k and l values in a `maxhkl` line.
3. Skipping of reflections with the `sk` option in the `CRYLSQ` line.

4.19.18. Derivatives

The following formulae give the equations used in the calculation of the derivatives of $F(\text{cal})$ required in the building of the least-squares matrices.

The structure factor equation (2) can be written as:

$$F = \sum_n [(A/F)(A' - B'') + (B/F)(B' + A'')] = \sum_n F_n b$$

$$\text{where } A' = \sum_s [f1 \cos \phi] \quad B' = \sum_s [f1 \sin \phi]$$

$$\text{and } A'' = \sum_s [f2 \cos \phi] \quad B'' = \sum_s [f2 \sin \phi]$$

The factors A/F and B/F are introduced to relate the phases of the partial derivatives with the phase of F . The formulae for the derivatives are:

4.19.18.1. Relative reciprocal scale factor (k)

$$dF/dk = F$$

4.19.18.2. Overall thermal displacement parameter (u_{OV})

$$dF/dU_{\text{OV}} = -s^2 \sum_n F_n$$

where n' is the atoms with overall thermal displacement parameter. When u_{OV} is used for the correlation correction, the summation is over all atoms.

4.19.18.3. Dispersion factors (f' and f'')

$$dF/df' = \sum_n [f1 / (f+f') ((A/F)A' + (B/F)B')]$$

$$dF/df'' = \sum_n [f2 / f'' ((-A/F)B'' + (B/F)A'')]$$

where n' is the atoms with specified f' and f'' .

4.19.18.4. Extinction factors

$$dF/dg = -0.5 \cdot \text{DELT} \cdot F^3 \cdot (1 + 2 \cdot g \cdot \text{DELT} \cdot (F^2))^{-5/4}$$

where the derivatives are reported by Becker & Coppens (1974).

4.19.18.5. Atomic positional parameters (x,y,z)

$$dF/dx = 2\pi h [(-A/F)(B' + A'') + (B/F)(A' - B'')]$$

4.19.18.6. Atomic thermal displacement parameters (U or U_{ij} in terms of B and β_{ij})

$$dF/dB_n = -s^2 F_n$$

$$dF/d(\beta_{ij})_n = h^2 [(-A/F)(A' - B'') - (B/F)(B' + A'')]$$

4.19.18.7. Population parameter (POP)

$$dF/dP_n = 1/P F_n$$

4.19.19. File Assignments

- Reads atom and reflection data the input archive `bdf`
- Writes refined atom and reflection data the output `bdf`
- Optionally writes correlation matrix to file `cmx`
- Optionally outputs `atom` lines to the line file `PCH`

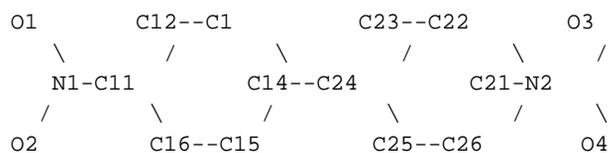
4.19.20. Examples

```
CRYLSQ
```

The input above causes all the defaults to be taken by CRYLSQ. All positional and thermal parameters of all atoms on the `bdf` will be refined in a single cycle of block diagonal refinement based on $F(\text{rel})$. The thermal displacement parameter type used in the calculation for each atom will be read from logical record *lratom*: of the `bdf`.

```
CRYLSQ f2 cy 3 is bl
block N1 N2
```

The compound used in this example is 4,4'-dinitrodiphenyl.



The `bdf` for this example has been formed so that the first atom in *lratom*: is N1. It is followed by O1, O2 and carbon atoms C11 through C16. After these atoms comes N2 and then the remaining atoms. The `block` line requests that the matrix be blocked in two pieces, each block containing the atoms of one of the rings. Such a blocking scheme can be useful at early stages of refinement or when there is insufficient storage to hold the entire matrix. The information on the `CRYLSQ` line specifies three cycles of refinement on F^2 with isotropic thermal displacement parameters for all atoms.

```
CRYLSQ is
group
atomgr N1/C16
group
atomgr N2/C26
constr u(N2)=0.0+1.0*u(N1)
```

Both parts of the molecule are refined as rigid groups. There is only one isotropic thermal displacement parameter for all 18 atoms.

```
ADDRESS dset 1 extf
reduce f2tof rlp2 neut
hklin hkl f2rl sgf2
hkl 0 0 2 5.227 1.303
hkl 0 0 4 19.679 2.195
```

.....reflection data omitted for brevity

```
CRYLSQ asn nu f2 ws rx cy 3 fm
scale 0.2295 1
extinc *4 .11726
noref C1/H10
CRYLSQ asn nu f2 ws l1 cy 2 fm rx
```

This example shows a run in which a set of neutron data with a Zachariasen's secondary extinction correction is refined. The [ADDREF](#) run shows the setting up of the reflection extinction factors. Two runs of CRYLSQ are shown. In the first one all parameters are held constant except the overall scale and g. In the second run all parameters are allowed to refine.

4.19.21. References

- Becker, P.J. and Coppens, P. 1974. *Extinction within the Limit of Validity of the Darwin Transfer Equations. I. Refinement of Extinction in Spherical Crystals of SrF₂ and LiF*. Acta Cryst. **A30**, 148.
- Buerger, M.J. 1960. *Crystal Structure Analysis*. New York: John Wiley.
- Busing, W.R. and Levy, H.A. 1959. ORXLS, *A Crystallographic Least-squares Refinement Program for the IBM 704*. ORNL Report ORNL-CF 59-4-37
- Finger, L.W. 1969. *The Inclusion of Secondary Extinction in Least-Squares Refinement of Single-crystal X-ray Data*. Yearbook 67. Carnegie Inst. Washington: 216.
- Flack, H.D. 1983. *On Enantiomorph-Polarity Estimation*. Acta Cryst. **A39**, 876.
- Larson, A.C. 1969. *The Inclusion of Secondary Extinction in Least-Squares Refinement of Crystal Structures*. Crystallographic Computing. F.R. Ahmed, S.R. Hall, C.P. Huber, eds., Munksgaard. Copenhagen: 291-294.
- Waser, J. 1963. *Least-Squares Refinements with Subsidiary Conditions*. Acta Cryst. **15**, 1091.
- Zachariasen, W.H. 1967. *A General Theory of X-Ray Diffraction in Crystals*. Acta Cryst. **23**, 558.

4.20. [DIFDAT](#): Process diffractometer data

Authors: Jim Stewart, Rina Merom, Jim Holden, Ruth Doherty, Syd Hall, Ted Maslen and James Hester

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

DIFDAT reads and scales diffractometer data (either raw counts, intensities or structure factors) according to standard reflections, and outputs either an archive bdf or HKL lines. The format of the input diffractometer data may be specified by input control lines.

4.20.1. Overview

A DIFDAT calculation reads an existing [Archive](#) containing cell and symmetry information, and reads intensity data from a diffractometer generated file, or from line images. The diffraction data may be input as raw intensities, net intensities, F² or F data.

The nature of the input data is specified in the **DIFDAT** line (as **cad**, **dat**, etc.). DIFDAT processes this data, and converts it into the quantities needed in the scaling process. The processed data is output as an archive bdf suitable for input to [SORTRF](#) (for sorting and averaging) or [ADDREF](#) (for data reduction). *Note that DIFDAT requires that there is an input archive bdf which has been previously created by [STARTX](#), etc.* Note also that it

is possible to process the diffractometer data in a non-conventional setting and subsequently transform data to a conventional set of axes (see Example 4). The following methods may be used to enter data.

4.20.2. Input diffractometer modes

dat mode (data lines)

Enter the diffraction data from **data** lines in the input stream. The contents of **data** lines are specified with a **fetch** line. See later section on data lines and the third example.

bdf mode

The diffractometer data is read from the archive bdf. It is possible to add this data to the archive bdf with CIFIO.

sie mode (file .sie)

Enter Siemens diffractometer data from a text file `.sie`. This file is either generated directly by the diffractometer software or is a binary file which has been converted to an ascii text file. The program `siecon.f` is available from the Xtal FTP site (see [Preface](#)) for this purpose.

cad mode (file .cad)

Enter Enraf-Nonius CAD4 diffractometer data from a text file `.cad`.

niv or nid modes (file .nic)

Enter the older Nicolet/Syntex/Siemens diffractometer binary data formats from a file `.nic`. Specifying the Siemens/Nicolet/Syntex file is complicated because it has a binary format and is therefore dependent on the controlling processor. For a Nicolet controlled by a Data General computer, the code `nid` is entered on the **DIFDAT** line. For a Siemens controlled by a VAX computer, the code `niv` is used. For a Nicolet connected to any other machine, changes will need to be made to the routine DD32. This option also outputs the intensity data as **data** lines on `pch` (for any subsequent modification of input data).

tsu mode (file .tsu)

Enter the standard data file collected at the Tsukuba Photon Factory on the file `.tsu`.

rd5 mode (file .rd5)

Enter AFC5 and AFC6 Rigaku diffractometer files on the file `.rd5`.

4.20.3. Derived intensity values

When raw data is entered the nett intensity is calculated according to the following scheme:

Attenuator	= value selected from the attenuator table
Timescale	= (ScanTime/2) / BackgroundTime
Background	= TimeScale*(Background1 + Background2)
Relative Intensity	= ScanCounts*Attenuator - Background
σI	= Attenuator*(ScanCounts + TimeScale*Background)
	= TotalCounts

4.20.4. Treatment Of Standards

If raw diffractometer data is read, the decoded quantities are first placed on the intermediate bdf. This file is then searched to separate the 'normal' measurements from the 'standard' measurements. Normal refers to a measurement made in the course of scanning all reflections, and standard to a group of reference reflections

for which measurement is made at regular intervals.

No more than 10 different reflections may be designated as standards. There may be as many measurements of the groups of standards as can be held in memory using four words per stored reflection per group measurement. The method of identifying normal and standard reflections will vary according to the diffractometer, and the method of processing data discussed above. If the reflection is coded as a standard, the σI , the total counts, a signal, and the sequence number of the measurement are stored. This operation requires that the reflections designated as standards appear sequentially and in groups with the same number of standards at each encounter. The sequence numbers of the standards are stored for use in forming the scale factors which will be used to compensate all reflections for any drift in the standards over the data gathering process. In the case where no sequence numbers are available from the diffractometer, they will be generated from the count of the reflection input lines.

Two methods are available to specify which reflections are to be used as standards. The first is to specify the coded information in the input which may be used to identify a standard reflection. The second is to give an ordered list of h, k, and l values for the reflections to be treated as standards.

4.20.4.1. Fate of the Standards

This option is specified in the **DIFDAT** line. The user may specify whether the standard reflections are to be used for scaling only or used for scaling and then placed in the output file as observed reflections for later merging with all the other normal observations.

4.20.5. Calculation Of Sigmas

If the input data has been partially preprocessed so that σI , σF^2 , or σF is not known, the user may specify an average error to be applied to I, F^2 , or F to give an estimate of the σI for further calculation. During the course of the data translation and intensity calculations, the standards measurements are stored in memory. These standards are used for two purposes; first is the establishment of a set of scale factors based on the counts of all the standards, second is the preparation of an *instability* factor. This factor is based on the spread and trend of the measurement of the standard reflections and used in the estimation of sigmas.

4.20.5.1. Instability Factor Calculation

The instability factor is calculated from variations in the intensities of the standard reflections. There are two quite distinct approaches to estimating the instability factor. This factor may be estimated either *before* or *after* the standard reflections are scaled. In the first approach, specified by the **nsc** option in the **DIFDAT** line, the instability factor (and therefore the estimated sigmas) contains all the unscaled variations in standards that occur during the course of measurement. This means that "slow changing" variations in standards, such as due to crystal degradation, will be included in the instability factor.

In the second approach, which is the default, the instability factor will only include those "fast changing" variations in standards which have not been removed by the overall rescaling process.

4.20.5.2. Application of Instability Factors

The form of the instability factor is specified as the **inst** option on the **DIFDAT** line. This option determines how the instability factor is used to modify the σI based on counting statistics for the single observation of a reflection. No attempt will be made to calculate the values of slopes or intercepts if the standards have been measured less than seven times during the data gathering procedure.

For *all* the instability factor options, the average value of the total counts for each measured standard reflection is calculated as follows.

$$\sum (\text{total counts of } j\text{th standard})$$

Average total count of jth standard = $\frac{\text{no. of observations of jth standard}}{\text{-----}}$

From this average value and the individual measurements of each standard the external variance may be calculated for each standard. This gives an independent measure of the variance over and above the variance based on counting statistics.

external variance = $\frac{\text{NTO} * \sum[(\text{TC} - \text{ATC})^2/\text{TC}]}{(\text{NTO}-1) * \sum [1/\text{TC}]}$

where TC is the total counts of the jth standard, ATC is the average total counts of the jth standard and NTO is the number of times the jth standard was observed.

4.20.5.3. Instability options (*inst* on the **DIFDAT** line)

Option *Meaning*

0 Only the counting statistics are used in calculating $\sigma(I)$, the square root of the total counts taken for a given observation. Default if no *inst* option is specified.

1 Only the slope of a line through the origin is calculated.
The customary least squares fit of a set of data to a line of the form of $y = mx + b$ is carried out. In this analysis the variable x is the square of the average total counts for each standard reflection and the variable y is the difference between the external variance and the average total counts. The sums are over the different standard reflections.

$$m = \frac{\sum xy}{\sum x^2}$$

2 Both the slope and intercept of the line are determined.
The values of x and y are the same as in option 1 and the sums are over the N standards.

$$\text{slope} = N * \frac{[\sum x \sum (y/N) - \sum xy]}{[(\sum x)^2 - \sum x^2]}$$

$$\text{intercept} = \frac{[\sum y - \text{slope} * \sum x]}{N}$$

3 The slope, determined in option 2, is used and the intercept value is left as zero.

4 Same as option 3 except that the user supplies the values of the slope and intercept in the **DIFDAT** line behind option *inst*. *m* is the second field beyond *inst*. *b* is the third field beyond *inst*.

4.20.5.4. Application of the Instability Factors

If any of the options for the application of the instability factors is chosen, the σI on the output *bdf* will be modified. The modification consists of scaling the σI calculated from the counting statistics according to the method shown below. For further information see Stout and Jensen (1968). The option will have been specified by the user in the first field beyond the *inst* option of the **DIFDAT** line and the values of m and b will be as defined above.

$$\text{new } \sigma I = [(\sigma I)^2 + |b| + |m| * (\text{total count})^2]^{1/2}$$

4.20.6. Reflection Codes (Rcodes)

During reflection processing estimated I values are compared with the estimated σI , based on intensity counting statistics. When $I > n * \sigma I$ a reflection is considered to be "observed" and is assigned an rcode=1. If $I < n * \sigma I$ a reflection is considered to be a "less than" and is assigned an rcode=2. The value of n may be specified in the **DIFDAT** line by the use of the **obst** nspecification. The value of n=2 is the default. If n=0 is specified, all reflections will be given rcode=1 values.

If the option **obth** is specified, the $n\sigma I$ value will be stored for the value of I. This option is a controversial one in that it imposes a feature reminiscent of the threshold of observability characteristic of photographic intensity measurements. The default, **nobth**, causes the calculated values of I to be stored in the output file. This means that negative as well as positive values are stored for reflections with weak intensities.

4.20.7. Data Lines

DIFDAT was originally designed on the assumption that the diffractometer output will be similar to that produced by a Picker FACS (see the Examples). As such diffractometer lines are assumed to start with the line identifier **data**. If this is not the case it will be necessary to preset the input line identifiers by entering **setid data** and follow the last data line with a blank **setid** (see *System*). The **fetch** line is used to specify the order and the type of data entered on the **data** line. Fields on the **data** lines must be separated by blanks. If they are not, it will be necessary to read these lines in fixed format mode using the **field** line (see CONTROLS section).

The alternative method of entering the diffractometer data involves the implementation of local routine (e.g. DD32) to read the data lines. The first two examples show the input used in this approach; the last two when **data** lines are entered.

4.20.8. Scale Factors

The option **nap** on the **DIFDAT** line, and the two input lines **setscl** and **genscl**, control the application of scale factors to the intensity data. The **nap** option suppresses the application of any scale factors to the intensities.

The **setscl** line specifies a scale factor which will will supercede that for a group of standards. The **genscl** line signals that the scales will be generated from the standard reflection groups and smoothed over a specified number of scale groups. If a **genscl** line is not entered, a scale factor must be entered for each standard group using **setscl** lines. If a **GENSCL** line is entered, the **setscl** lines will be used to override one or more of the generated scale factors.

genscl controls the smoothing of scale factors over a specified number of standard groups. Statistical fluctuations can occur due to natural counting errors or too few standards being observed. The default smoothing range is five groups forward and five groups backward.

It is possible that the scale discontinuities are not statistical. This occurs if there is a change in the radiation source or a degradation of crystal. In this case, **discon** lines may be used to point to the standard groups at which the scale discontinuities are real. The smoothing function will not span these groups. Note that the serial numbers of the last member of a group of standards are used in the **setscl** and **discon** input lines.

DIFDAT assumes that data starts with a group of standards and ends with a group of standards. This arrangement is not mandatory but warning messages are written if this condition is not met and a careful assessment of the scale factors should be made.

4.20.9. Output Data

DIFDAT is a two pass program. The data is first processed for the estimation of scale and instability factors. It is then reread and the scaled intensities are written to the output archive bdf. The data written into logical record *lrrefl*: are the packed hkl (IDN=1), the net intensity (IDN=1300), σI (IDN=1301), and rcode (IDN=1308).

Printed output of reflection data is specified on the **DIFDAT** line with the options (**print** and **raw**). In the first pass the raw intensity data is printed and in the second the scaled reflection data is printed. There is provision for a sample of reflection data to be printed during each pass. Each reflection line is screened for unequal backgrounds and this condition is noted in the optional printed output.

4.20.10. File Assignments

- Reads *lrcell*: and symmetry data from the input archive bdf
- Writes reflection data to the output archive bdf
- Optionally reads diffractometer data from separate files, data lines or the bdf.

4.20.11. Examples

Process a 61 byte-record CAD4 diffractometer file *cad*.

```
DIFDAT cad pri 50 excl inst 1 obst 3
attenu 19.14
genscl 2
```

Process a standard Siemens diffractometer text file *sie*.

```
DIFDAT sie pri 50 excl inst 1 obst 2
genscl 3
```

Process data on **data** lines

```
DIFDAT pri 50 eul inst 1 obst 2.0
stands 1 1 2 2
fetch rfn h k l tth omg phi chi irl sir sig
data 1 0 0 2 13.56 6.78 95.68 112.63 6574 135
.....data lines omitted for brevity
data 1354 12 7 -9 87.42 43.71 15.93 74.14 761 48
genscl 3
```

Process data for non-conventional setting - full calculation sequence.

```
title IO11 in non-conventional setting Pnca of Pbcn
STARTX
CELL 10.664 15.838 26.086
sgname -p 2a 2n :pnca
CELCON c 10
DIFDAT cad excl inst 1
atten 16.8
genscl 2
SORTRF hkl merge 1 cut
COPYBDF a tem :store merged refln data on tem
```

```
title I011 in conventional setting Pbcn
STARTX
CELL 15.838 26.086 10.664
cellsd .004 .006 .002
sgname -p 2n 2ab      :pbcn
CELCON c 120
CELCON o 40
CELCON h 200
ADDREF
reduce itof rlp3
transf 0 1 0 0 0 1 1 0 0
bdfin file tem hkl irel sigi rcod
```

This is an important example because it shows a full Xtal run from *STARTX* to *ADDREF* involving the DIFDAT calculation. It also demonstrates how to process diffractometer data collected with non-conventional axial settings, and to subsequently transform it the conventional space group setting. The components of this run are: generate an archive bdf for the non-conventional space group; process the CAD4 data in this space group; sort and merge the intensity data in this space group; save the current archive bdf containing the merged data on file `tem`; create a new archive bdf with the conventional space group settings; reduce the reflection data by inputting the intensity data from the file `tem` (see the `bdfin` line) and transform the indices to match the conventional axes. The output archive file contains an unique set of Frel values for the space group Pbcn.

4.20.12. References

- Stout , G. and Jensen, L. 1968. *X-ray Structure Determination: a Practical Guide*. New York: MacMillan Publishing Co.

4.21. **DYNAMO**: Molecular Dynamics structure factor calculation **NEW**

Authors: Doug du Boulay

Contact: D. du Boulay, Materials and Structures Laboratory, Tokyo Institute of Technology, Nagatsuta, Midori Ku.

DYNAMO calculates structure factors from molecular dynamics data such as that from the program *MX-DORTO* (K. Kawamura). In MD simulations atomic nuclei positions are recorded at pico second intervals, typically for the order of nanosecond duration simulation experiments. These data sets can be used to calculate the average structure factors that one might expect to see in equivalent diffraction experiment averaged over the timescales involved and also over the spatial extent of an entire crystal. The "experimental" MD structure factors can then be compared with structure factors calculated in the standard crystallographic manner, incorporating temperature factors to model the time and spatially averaged electron density.

4.21.1. Process

Structural crystallography is ordinarily focussed on the most minimal set of independent atom sites arranged in the asymmetric unit or, the arrangement of a symmetry extended subset of sites contained within one unit cell. Consequently crystallographic software like Xtal has generally been optimised to work with the most primitive set of unique atomic information. In contrast, molecular dynamics (MD)simulation studies dictate an approach wherein all atoms in the entire crystal are assumed to evolve fully independently, over time scales of the order of femto to pico seconds. Practical limitations involving storage space and evaluation times place

severe restrictions on this ideal so generally it is sufficient to adopt "very large" unit cells with no internal symmetry (i.e. *PI*) but subject to periodic boundary conditions.

To compare MD simulation data with charge density experiments we need to convert the MD atomic site information into structure factors and thereby into charge densities. **DYNAMO** can assist with the first step and **CRYLSQ** in conjunction with **FOURR** enables the second.

The results of MD simulations are generally large data files consisting of the instantaneous positions of atomic nuclei for each step. At each instant an MD atom effective scattering contribution should thus be very well accounted for by the elemental form factors of Cromer and Mann (19??) determined for isotropic noninteracting and non-vibrating atoms. By summing the form factors of all the independent MD atoms across the entire MD cell, and by further averaging of the MD cells across repeated MD time steps very well averaged, effective scattering factors can be determined. This is a convenient way of converting very extensive MD data sets into a relatively simple representation in the form of time and spatially averaged structure factors, exactly the same kind of data measured experimentally in single crystal X-ray scattering experiments.

If the MD effective structure factors F_{md} are treated as experimentally determined structure factors, then an effective scattering model can be refined to model the MD data using standard least squares programs, such as **CRYLSQ**. In the refinement however there is absolutely no need to refine the scale factor because the MD structure factor data have all been calculated on an absolute scale. There is also no extinction and no dispersion because the MD data do not incorporate such effects, by default. It should therefore, be sufficient to refine only the first and second moments of the density, i.e. the mean atomic positions and anisotropic temperature factors. Any residual features in the charge density after Fourier transforming the structure factor differences therefore reflect real deviations from the averaged, harmonic model and, not chemical or bonding redistributions. An anharmonic structure factor model may or may not account for such features, but in any case can not be undertaken within Xtal.

4.21.2. Calculation Performed

For extensive MD data sets the calculations performed by **DYNAMO** can be rather time consuming. As a small measure of optimization, all reflections on the bdf are loaded into memory simultaneously. Unfortunately this puts limits on the number of reflections for which MD structure factors can be evaluated. As each atom site in each requested frame is read, its contribution to the effective MD cell scattering power is accumulated for every reflection stored in memory. After each frame a running average is performed on the MD structure factors. Finally the MD structure factors are normalised by the determinant of the cell transformation matrix to put them on the same scale as the archive BDF sites. Subsequently, if atom sites exist on the archive, then normal F_{cal} structure factors are calculated, and relative scales compared (though not applied to archive). In principle, if your archive model sites and molecular dynamics sites are consistent then the scale factor should be 1.0.

The F_{md} values can be stored as either F_{rel} , or F_{cal} , in case it is desired to compare the F_{md} values with some experimental structure factors already existent as F_{rel} on the archive. In that case it may be necessary to apply dispersion and extinction corrections to the F_{md} values to match the experimental F_{rel} terms.

4.21.3. File Assignments

- Reads atom and reflection data from the input archive bdf
- Reads MD data file
- Writes calculated structure factors to the output archive bdf

4.21.4. Example

```

STARTX      : spinel
cell      8.2228 8.2228 8.2228 90 90 90
cellsd    0.0001 0.0001 0.0001 0 0 0
spaceg    F_d_-3_m:2
celmol    8
celcon    Li  8
celcon    Mn 16
celcon    O   32
exper    *2 0.60470

ADDATM
UOV              .03500                PARENT
ATOM  Li1        .12500  .12500  .12500 $1 1.000 .00000 .00000 .00000
U      Li1        .01295  .00095
ATOM  Mn         .50000  .50000  .50000 $1 1.000 .00000 .00000 .00000
UIJ   Mn         .00521  .00521  .00521 -.00083 -.00083 -.00083
SUIJ  Mn         .00005  .00005  .00005 .00004 .00004 .00004
ATOM  O          .26526  .26526  .26526 $1 1.000 .00008 .00008 .00008
UIJ   O          .01921  .01921  .01921 -.00461 -.00461 -.00461
SUIJ  O          .00021  .00021  .00021 .00021 .00021 .00021

ADDRF ffac      : use interpolated form factors in DYNAM/FC
limits *4 1.0   : *9 yes
hklgen hkl frel sigf : generate reflection hkl values for DYNAM
end

SORTRF frel aver 1 : remove symmetry redundant reflections
end

: Be wary of rounding errors in accumulating averages over many
: frames. <F(n-1)>(n-1)/n + F(n)/n Doesn't scale well.
DYNAMO frel list file fort.81      : save Fmd as Frel, list all refs , read file
frames 1 100                       : start from frame 1 read 100 frames
transf 4 0 0 0 4 0 0 0 4          : change cell from 8x8x8 -> 32x32x32
typinf 0 2048 Mn 512 Mn 512 Li 512 : order matches fort.81 order
end

: improve the fcal model by refining aniso Uij
CRYLSQ cy 20 wu fr pl tl 0. ll : no dispersion/extinction needed
noref skf

FOURR fdif
grid 96 96 96

In this example reflections in an asymmetric unit of reciprocal space are generated for the 8x8x8 cell entered in STARTX. In DYNAM those hkl values are transformed to the 32x32x32 unit cell setting and MD structure factors are calculated before being transformed back to the original 8x8x8 cell. This is followed by a normal structure factor calculation within DYNAM, for magnitude comparison purposes.

:STARTX and ADDATM as for example 1
:followed by ...

EXPAND cell 3 : generate 5x5x5 cells worth of redundant sites

STARTX upd      : throw away all symmetry
sgname P1

NEWCELL          : transform archive cell from 8x8x8 to 32x32x32
transf 4 0 0 0 4 0 0 0 4

```

```

STARTX upd          : override wierd centred symmetry
sgname P1

EXPAND not          : contract to 32x32x32 cell P1 independents

ADDATM upd         : update atomic constraints

ADDREF ffac        : use interpolated form factors in DYNAM/FC
limits *4 0.7      : *9 yes  : too many reflections for DYNAM to handle
hklgen hkl frel sigf : generate empty hkl packets
end

SORTRF frel aver 1 : remove redundant reflection indices

DYNAMO frel list file fort.81 : save Fmd as Frel, list all refs , read file
frames 1 10          : start from frame 1 read 10 frames
transf 1 0 0 0 1 0 0 0 1 : No change from MD cell to archive bdf
typinf 0 2048 Mn 512 Mn 512 Li 512 : order matches fort.81 order

FOURR fdif pset 1   : use MD phases via pset 1 flag
grid 192 192 192

```

Here structure factors are computed for the 32x32x32 P1 cell. There is no need for Friedel pairs as there isn't any dispersion. To correctly do the Fourier transformation, the MD phases are stored on the archive (in addition to the normal structure factor phases - which are nearly all zero owing to the imposed symmetry of the original 8x8x8 model). The MD phases are accessed in FOURR via the `pset 1` flag. Alternatively, the MD structure factors (now Frel values) and the whole unit cell could then be transformed back to the 8x8x8 F_d-3_m:2 cell for structure factor averaging and model comparisons.

4.21.5. References

- Henry, Norman and Lonsdale, Kathleen, 1965. Eds. *International Tables for X-Ray Crystallography Vol. I*. Birmingham, England: Kynoch Press.

4.22. **EXPAND**: EXPAND archive atom sites **NEW**

Authors: Doug du Boulay

Contact: D. du Boulay, Materials and Structures Laboratory, Tokyo Institute of Technology, Nagatsuta, Midori Ku.

The program EXPAND can be used to expand the list of independent atom sites on the archive bdf (as introduced using ADDATM) to a complete list of all dependent sites in the current symmetry defined unit cell, or even multiple unit cells. This is not a desirable end result of itself, but it is very useful for converting between different symmetries and/or unit cell transformations. When followed by the EXPAND not option, the redundant symmetry related sites are culled from the archive to restore normality.

4.22.1. Calculation Performed

The asymmetric set of atomic sites is read from the input archive bdf. By default, those sites are expanded to include all sites in the unit cell. To avoid duplicate name errors in **ADDATM upd** mode the atom names are amended with cell translation and symmetry operator strings. This can increase the atom name string length beyond 8 chars, the standard limit in small molecule crystallography. Fortunately Xtal handles up to 24 character atom names, so it is not a problem, though not pleasing aesthetically.

In any case, by following any **EXPAND cell n** command with a matching **EXPAND not**, the extraneous atom name appendages are culled in favour of simpler, more compact site names.

EXPAND also transforms the anisotropic temperature factors, but it does not touch the constraint record *lr-cons:*, so it is always necessary to run **ADDATM upd** to fix any broken atom site constraints after running EXPAND.

4.22.2. File Assignments

- Reads atom data from the input archive bdf
- Writes atom data to the output archive bdf

4.22.3. Example

```
EXPAND cell 1
```

4.23. *FC*: Calculated structure factors

Authors: Syd Hall, Jim Stewart, Richard Alden & Jim Holden

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

FC calculates structure factors when refinement of atomic parameters is not required.

4.23.1. Calculation Performed

The asymmetric set of atomic sites is entered from the input archive bdf, and is used to generate symmetrically related sites for all atoms in the unit cell. The structure factor calculation is applied to these sites. The coordinates of all sites in a cell may be listed by entering the line "reset psta 4" prior to the FC line.

Overall isotropic, individual isotropic, individual anisotropic, or mixed individual thermal displacement parameters may be applied in FC. Dispersion factors and isotropic extinction corrections may also be applied provided prerequisite data is present on the input bdf.

If requested FC will calculate, and store on the bdf, the Sim (1960) weight.

$$w(\text{Sim}) = I_1(f) / I_0(f) \text{ where } f = 2 F_o F_c / \sum f^2$$

4.23.2. File Assignments

- Reads atom and reflection data from the input archive bdf
- Writes calculated structure factors to the output archive bdf

4.23.3. Example

```
FC disp list 7.5
scale 1.013
```

In this example, structure factors are calculated using all atoms in the bdf. Dispersion corrections will be applied. All reflections for which F(rel) and F(cal) differ by more than 7.5 will be listed. In addition, for this calculation, the F relative scale factor is to be set to 1.013 for scale group 1. When the updated bdf is written it will include F(cal), the estimated phase and the contributions to A and B from dispersion.

4.23.4. References

- Henry, Norman and Lonsdale, Kathleen, 1965. Eds. *International Tables for X-Ray Crystallography Vol. I*. Birmingham, England: Kynoch Press.
- Sim, G.A. (1960) *Acta Cryst.* **13**, 511.

4.24. **FOURR**: Fourier maps

Authors: Jim Stewart, Jim Holden, Ruth Doherty & Syd Hall

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

FOURR is a general purpose reciprocal-space to real-space crystallographic Fourier transform. It uses the Beevers-Lipson (BL) algorithm for the first sum and either the BL or Fast Fourier Transform (FFT) algorithm for the second and third sum directions. The summation is general with respect to space group setting and axial display. A wide range of resolutions may be specified. The default is 1/4 Angstrom between grid points.

The Fourier density values are stored in the `map` file for use by succeeding calculations such as **PEKPIK** and **CONTRS**. For Patterson maps the densities will be in electrons squared per cubic Angstrom provided that a reasonably good value of the F(rel) scale is available. For E maps the E coefficients are automatically scaled by the ratio F(000)/E(000) so that the calculated density will closely approximate $e/\text{\AA}^3$. In the same way the vector map (E^2-1) is automatically scaled so that the sharpened densities approximate $e^2/\text{\AA}^3$. It is these densities that are output to the `map` file.

Since the magnitude of these densities is generally unsuitable for printing as integer numbers, a special print scale factor PRINTSCL is applied to the densities before they are output on the line printer or visual display unit screen. A wide range of formats and options are available for displaying the printed output. The defaults are shown in the description of the `print` line. The PRINTSCL default is a function of the type of coefficient specified. Those involving F, for example, require, a PRINTSCL of 10.0 to produce sensible integers for printing on a 4 column per Fourier grid point basis.

Other coefficients require other PRINTSCL values. Defaults are provided for the major coefficient types F(rel), F²(rel), F(cal), E, and F(diff).

4.24.1. Shaded And Packed Printed Output

A Fourier may be output in a compact form with the `shaded` option of the `print` line. Layers are printed with one column and one line per grid point. By judicious choice of the PRINTSCL constant and SCLINT density interval a shaded mini-Fourier can be produced which shows the structure on an approximately 1/4 scale when compared to the default maps. The peaks are represented by higher density print characters. Zero is represented by period, and the negative regions by a different set of characters. The actual coordinates of the peaks may be determined by the use of `PEKPIK` which processes the Fourier transform as it is stored on the `map` file.

An alternative column output is possible where base-36 arithmetic is used to give densities 0 through 9 and A through Z. This kind of printing is referred to as `packed`. No negatives are printed when this option is specified on the `print` line.

4.24.2. Limits Of The Transform

Ideally the limits of the Fourier transform should only encompass one asymmetric unit as close to the origin as possible. The default limits for the parallelepiped produced by the FOURR program attempt to achieve this goal. For the higher symmetry space groups it will often be necessary to produce more than an asymmetric portion of the cell since the limits set are always an integral number of grid points along the axial directions. When there is redundancy, the `PEKPIK` program will screen out the symmetrically equivalent points which have been produced. An algorithm has been set up which uses the space group symmetry operators to seek an asymmetric portion of the unit cell. These limits are those which are set when the default limits are utilized. The `layout` or `map` lines allow the user control of the limits of the map if it is desired to control these limits for other reasons. It should be noted that for small regions the BL algorithm may be the best choice since only the map specified will be calculated. It is necessary when using the FFT algorithm always to calculate a full cell span whether or not only a small fraction of the cell is written to the `map` file or to the line display device.

4.24.3. Summation Order

The summation order is usually determined by the axial lengths of the unit cell. When the long axis is summed first, the calculation will be fastest and the least computer memory will be required. Thus the default will be long axis layer to layer, next longest axis down each layer, and shortest axis across each page. Both the `map` and `layout` input lines allow the user a means to override these defaults by specifying the desired orientation of the map on the printed pages.

4.24.4. Coefficients Used In Summations

FOURR performs Fourier transforms on a wide range of coefficients and phase types. The coefficients and phases must be functions of the quantities stored in the input `bdf`. The most common coefficient types are selected by setting a signal in the FOURR input line. These are:

Patterson map

Uses scaled $F(\text{rel})^2$ as coefficients and exhibits the Laue symmetry of the space group.

Patterson E map

Uses $F(\text{rel}) * E$ as coefficients and exhibits the Laue symmetry of the space group. This is the recommended map for use with `PATSEE` (The default layout will produce a map for the full cell - this may be overridden with `map` and `overlay` lines).

Vector map

Uses (E^2-1) as coefficients and exhibits the Laue symmetry of the space group. E-values are calculated by *GENEV*.

E map

Uses E values as coefficients and phases from one of the phase sets generated by *GENTAN* or *SIMPEL*. The map has the symmetry of the space group.

Fobs map

Uses scaled F(rel) values as coefficients and phases from programs such as *FC* or *CRYLSQ*. The map produced has the symmetry of the space group.

 ΔF map

Uses $(K \text{ Frel} - \text{Fcal})$ as coefficients and phases the last structure factor calculation. K is the appropriate scale factor to convert Frel to absolute. The map has the symmetry of the space group.

Fcal map

Uses F(cal) values as coefficients and phases from programs such as *FC* or *CRYLSQ*. The map produced has the symmetry of the space group.

wFob map

Uses scaled F(rel) values multiplied by the Sim weights as coefficients and phases from the program *FC*. The map produced has the symmetry of the space group.

wdif map

Uses $(K \text{ F(rel)} - \text{Fcal})$ values multiplied by the Sim weights as coefficients and phases from the program *FC*. The map produced has the symmetry of the space group.

pdif map

Calculates the electrostatic potential from $(K \text{ F(rel)} - \text{Fcal})$.

edif map

Calculates the electric field vector (3 components) from $(K \text{ F(rel)} - \text{Fcal})$.

gdif map

Calculates the electrostatic field gradient (5 components) from $(K \text{ F(rel)} - \text{Fcal})$.

In addition to the pre-set transforms described above, the FOURR program uses of a **coeff** line to specify coefficients from constants and from quantities held in the bdf. This allows almost any type of transform to be undertaken with relative ease. In addition, the program BFOURR can be used to store very complex Fourier coefficients on the bdf, which may be employed in FOURR through the use of a **coeff** input line. This is discussed below.

4.24.5. Explicit Definition Of Fourier Coefficients

In FOURR the user may explicitly define Fourier coefficients and phases using the **coeff** line. This is done with the identification (ID) numbers of items in the *lrrefl:* packet of the input bdf (for details see the BDF section at the back of this manual), and explicit constants. A coefficient is specified as a series of integers. Positive integers represent the ID numbers, negative integers represent positive constant scales, and zeros signal that a particular component is to be excluded from the construction of the Fourier coefficient. The general form of coefficient expression is as follows:

Coefficient = (factor1 * factor2 * *scale3* - factor4) * factor5 * *scale6*

where the components factor1, factor2, factor4 and factor5 may be specified in fields 1, 2, 4 and 5 of the **coeff** line as ID numbers or constants. The components scale3 and scale6 must be specified as either the ID number of the *scale group*, or as a constant scale value.

The coefficient phase is defined using fields 7 and 8, or field 9. Fields 7 and 8 specify the real and imaginary components of the structure factor in terms of their ID numbers. Field 9 specifies the ID number of the structure factor phase in cycles. If both are specified and are present in the bdf the phase defined by field 9 will be used. These fields are applied to the Fourier expression as follows:

$$\cos\phi = \text{factor7} / ((\text{factor7})^2 + (\text{factor8})^2) \text{ or } \cos\phi = \cos(2\pi \text{ factor9})$$

$$\sin\phi = \text{factor8} / ((\text{factor7})^2 + (\text{factor8})^2) \text{ or } \sin\phi = \sin(2\pi \text{ factor9})$$

To illustrate the application of these coefficients in more general cases the six standard transforms specified via the FOURR line are shown as they would be defined on a **coeff** line. In the table which follows, n represents the dataset number.

	<i>fac1</i>	<i>fac2</i>	<i>scl3</i>	<i>fac4</i>	<i>fac5</i>	<i>scl6</i>	<i>fac7</i>	<i>fac8</i>	<i>fac9</i>
PATT	n304	n304	n309	0	-1	n309	0	0	0
EPAT ^a	n304	n600	n309	0	-1	-1	0	0	0
VECTOR _a	n600	n600	-1	-1	-1	-1	0	0	0
EMAP _a	n600	n632	-1	0	-1	-1	0	0	n631
FOBS	n701	-1	n309	0	-1	-1	n801	n802	n700
FDIF	n701	-1	n309	n800	-1	-1	n801	n802	n700
FCAL	n800	-1	-1	0	-1	-1	n801	n802	n700
WFOB	n701	-1	n309	0	n910	-1	n801	n802	n700
WDIF	n701	-1	n309	n800	n910	-1	n801	n802	n700

Notes:

- a. The ID numbers for |E|, phase and weight are not fixed values but vary according to the phase set selected by the user with **FOURR** line controls.

It will be necessary to read the BDF section at the back of this manual, and in particular the part concerning the *lrrefl*: ID numbers, to understand properly how to use the **coeff** line. It is important to stress that to permit maximum flexibility there is very little internal checking of parameters input on the **coeff** line.

4.24.6. Output

Output of the Fourier densities may be as a printed map or as a `map` file. The bdf serves as input to programs such as *SLANT*, *PEKPIK*, *CONTRS*. The quantities stored in this file include the generated transform, the description of the map, cell dimensions, and space group symmetry operations. The printed output is identical to the map output to the file but on a different scale, as described above. If the number of points in the third sum direction is more than can be printed across a page, the printed output may be abbreviated to accommodate the number of print columns on the printer. If it is abbreviated, the method leaves out alternate points thereby reducing the resolution of the printed map. All points are written to the `map` file.

The printed output may be scaled so that a distance on the page will have an Angstrom equivalent. No attempt is made, however, to compensate for the effect of an interaxial angle in the printed layer. The program *CONTRS* will produce a file for plotting of undistorted contour maps.

4.24.7. Limits Of Resolution

The output map of FOURR is divided into specific densities on a three dimensional grid. The grid interval, or rather the number of grid points along each cell dimension, defines the map resolution. Typically this resolution needs to be 3 or 4 points per Angstrom. In FOURR the default resolution is set at 0.25 Angstroms.

This may be changed using a **grid** line. In conjunction with this option, it should be noted that the **maxhkl** line provides a means to reject reflections that do not contribute to the sums for the resolution specified by the given grid. In the case of the FFT method, the resolution specified will force the rejection of reflections beyond the resolution limit set or cause the inclusion of zero for those beyond the limit. An attempt to calculate a FFT with grid intervals of less resolution than the maximum h, k, l data allowed will result in an error termination.

4.24.8. The F(0,0,0) Term

The F(0,0,0) term is an important part of the Fourier calculation. The *default* F(0,0,0) term is applied for the following options:

COEF	PATT	EPAT	VECT	EMAP	FOBS	FDIF	FCAL	WFOB	WDIF
0.	0.	0.	0.	E000	F000	0.	F000	F000	0.

These values may be replaced using the **fzero** input line.

4.24.9. File Assignments

- Reads Fourier coefficients from the input archive **bdf**
- Writes map points to file **map**.

4.24.10. Examples

```
FOURR patt ffsum
map *5 .5
```

This use of the Fourier program will result in a Patterson map being produced. The algorithm which is used for the second and third sums will be the FFT. The scale of the map will be four points per Angstrom. The summation will be made over the long axis first, so that there will be a number of pages corresponding to one layer every 1/4 Angstrom. If there is an interaxial angle in the plane of the "page" there will be a distortion due to the orthogonal print positions. The **map** input line forces the printing and limits the printing to 1/2 the cell in the first sum direction. The scale will default to one electron squared per Angstrom cubed for a Patterson map.

```
FOURR patt
print *3 10 -100
map c 0 0 0 .25 1 .25
```

In this case, a Patterson function is to be computed. However, more control over the orientation, scale, and extent of the transform is desired. The algorithm used will be BL. The **print** line supplies directives to print the map, to scale to one electron squared per Angstrom cubed - assuming the F(rel) scale factor is reasonably close to one. If GENEV has been run, an estimate of the F(rel) scale factor will be known. Furthermore, the **print** line has a specification to suppress printing of all values between 10 and -100. This means that the map will have decimal points placed in this region of densities. On the **map** line the **c** designates that the c crystallographic axis is to go from page to page. The next 6 fields are the minimum and maximum fractional coordinates of the map which is to be generated 0 to .25 along a; 0 to 1 along b, and 0 to .25 along c.

```
FOURR fobs
print *2 100.
```

```
grid 12 12 12
layout down across layer 13 13 13 0 0 0 1 1 1
```

In this case, it is desired to control the actual grid of the Fourier map scaled to 100 electrons per Angstrom cubed. A **grid** line is used to force twelfths along all axial directions. The **layout** line specifies the number of points to be calculated, the starting point, and the step or increment along all of the axial directions. The BL method is specified so that 13 points can be printed in every direction and if maximum h, k, or l should be greater than 12 the computation will still be done. Finally, the summation order is specified as c first - page to page, a second - down the pages, and b third - across the pages. This method of use can be very useful for checking that the map has the expected symmetry. This example, in fact, is a most useful check since twelfths or twenty-fourths will be divisible by the fractions of all symmetry operations. It is always a useful exercise to check to see that the points for a general position of the space group are all the same to within rounding. If it is determined that the space group symmetry is violated it is probably because systematically absent reflections have been entered in the bdf as if they were observed.

```
FOURR fobs ffsum
maxhkl 62 45 30
grid 126 96 64
layout layer down across 32 96 64 0 0 0 1 1 1
```

The compound used in this example is cytochrome c peroxidase. The crystal belongs to space group $P2_12_12_1$, with cell dimensions of 107Å, 77Å, and 51Å. For a cell this large the preferred algorithm is FFT. In order to obtain a map with the asymmetric portion of the cell, the limits may be set in either one of two ways:

Xmin	Xmax	Ymin	Ymax	Zmin	Zmax
0.0	1.0	0.0	0.5	0.0	0.5
0.0	0.25	0.0	1.0	0.0	1.0

In either case, only 1/4 of the total map is calculated. However, the second case is preferable for two reasons: It induces calculation of fewer points along the largest axis and it takes advantage of the fact that the FFT will necessarily produce all of the coefficients for the points from 0 to 1 along y and z whether they are requested or not. The **layout** line therefore specifies that only 32 points should be calculated in the x-direction. Since there are to be 126 grid points between 0 and 1 along x, this will give the desired result.

4.24.11. References

- Henry, Norman and Lonsdale, Kathleen. 1965. Eds. *International Tables for X-ray Crystallography Vol. I.*, Birmingham, England: Kynoch Press.

4.25. *GENEV*: Normalized structure factors (E values)

Authors: Syd Hall and V. Subramanian

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

GENEV calculates normalized structure factors (E values) from the measured structure factors, and calculates the scale and overall U. Unique aspects of *GENEV* include the estimation of |E| errors, the application of

simple Bayesian statistics to weak data, the compensation for missing weak data, a facility for user-specified index rescale groups, and the calculation of group structure factor phases for use in subsequent phasing processes.

4.25.1. Introduction

Structure factor normalization converts measured $|F|$ values into "point atoms at rest" coefficients known as $|E|$ values. The normalization process used in GENEV is based on a Wilson plot procedure (Wilson, 1942). This approach also provides a reasonable estimate of the overall thermal displacement parameter and overall structure factor scale.

A general introduction to normalization procedures based on the Wilson plot can be found in the crystallographic texts such as Stout and Jensen (1968), or Luger (1980). For more detailed information on structure factor normalization the user should refer to the three papers by Subramanian and Hall (1982) and Hall and Subramanian (1982a,b). These studies form the basis for GENEV and provide detailed definitions and descriptions of the scaling functions and expectation expressions discussed here.

Central to the normalization approach of GENEV is the calculation of two separate normalized structure factors $|E1|$ and $|E2|$. $|E1|$ is calculated using the linear scale $K \exp(Bs^2)$, the random-atom expectation value, and an overall rescale term. This combination of parameters has been shown to provide a consistently good estimate of $|E|$ (S & H, 1982). Inflection point least-squares (H & S, 1982a) is used to obtain an estimate of the overall B which is largely independent of the Debye scattering effects. The basic process for calculating $|E1|$ is fixed and cannot be varied by the user.

$|E2|$ is calculated using scaling options selected by the user. These include the choice of linear or profile scale; random atom, random fragment, oriented fragment or positioned fragment $|F^2|$ expectation values; overall or index rescaling terms (see S & H for detailed description of these options). The control lines may be used to determine what form the calculation of $|E2|$ should take. If no control lines are entered, $|E2|$ will be calculated identically to $|E1|$ except that index rescaling is the default rather than overall rescaling. Note that if GENEV is run just for calculating the scale or the overall U, the `enot` option should be used to prevent $|E|$'s from being output to the bdf. The default is `eout`.

Some of the options detailed below apply to both $|E1|$ and $|E2|$. The `fill` and `baye` control line for example provides for the adjustment of weak data. It should also be emphasized here that group structure factor phases calculated from fragment information entered via `frag` and `atom` lines may be applied in subsequent phasing processes to both $|E1|$ and $|E2|$. Details of this are given below.

4.25.2. Treatment Of Weak Data

One of the factors determining the reliability of $|E|$ values is the precision of the measured structure factors. This is true not only for the large structure factors but also for the weak reflections that often make up the majority of the data. In particular the precision of the weak data affects the reliability of B and K estimated by the Wilson plot least-squares process. GENEV provides two methods of treating weak data; one applies Bayesian statistics to $|F|$ and $\sigma|F|$ values if $|F|$ is less than $6\sigma|F|$, and the other compensates the Wilson plot for weak data omitted from the input bdf.

4.25.2.1. Application of Limited Bayesian Statistics

The expected Bayesian distribution of net intensities, and their associated standard deviations, have been reported by French and Wilson (1978). A limited Bayesian treatment of structure factor data is applied by entering `baye` on the GENEV line. This assumes that the mean intensity for all shells of reciprocal space is $20\sigma I$. The procedure used in GENEV to do this requires the presence of both the net intensity and the Lorentz-polarization factor on the input bdf (S & H, 1982).

4.25.2.2. Compensation for Missing Weak Data

The Wilson plot process relies on a relatively uniform distribution of data with s^3 ($s = \sin\theta/\lambda$). The omission of weak data from this calculation tends to give rise to erroneously low values of B and to inaccurate estimates of $|E|$. In GENEV the `fix` option on the GENEV line forces each shell of reciprocal space to have the correct population. The values of $|F|$ used for the added reflections are based on half the minimum $|F|$ of each range.

4.25.2.3. Expected $|E|$ Distribution and Statistics

Expected mean of $\langle E \rangle$	$\langle E_2 \rangle$	$\langle E_2-1 \rangle$	$\langle E_2-1 _2 \rangle$	$\langle E_2-1 _3 \rangle$	
For random P -1	.798	1.000	.968	2.000	8.000
For random P 1	.886	1.000	.736	1.000	2.000

Percent of $ E >$	0.0	1.0	1.2	1.4	1.6	1.8	2.0	2.50	3.00
Total With									
For random P	100.	31.7	23.0	16.1	11.0	7.2	4.6	1.20	0.27
For random P 1	100.	36.8	23.7	14.1	7.7	3.9	1.8	0.19	0.01

4.25.3. Application Of Scaling Options

GENEV provides for two basic scaling approaches: The linear scale $K \exp(8\rho^2 U s^2)$ (this is the default) and the profile scale (`prof`). In addition there are two rescaling procedures; overall (default) and index (using `indexk`). The definitions and the properties of these scales is detailed by S & H (1982).

4.25.3.1. Linear Scale (applied to E1; default for E2)

The application of the linear scale first requires the evaluation of the overall scale, K, and the overall thermal parameter U from the Wilson plot. The particular form of the linear scale used to calculate $|E_1|$ is based on U and K values estimated using a inflection-point least-squares procedure (see below). Use of the linear scale is optional for $|E_2|$ and, if applied in the default mode, will be identical to that used for $|E_1|$. Entering the `frag` line will cause an independent linear scale to be evaluated for $|E_2|$.

4.25.3.2. Profile Scale (`prof` option for E2)

If a `prof` is entered on the GENEV line the profile scale is used to evaluate $|E_2|$. The profile scale is an interpolated curve based on 41 overlapped Wilson plot averages. Some degree of caution should be exercised in using this option because of the tendency of the radially-dependent structural contributions to $|E_2|$ to be reduced. It may, however, be useful for reducing the dominant features such as occur in "chicken-wire" structures.

4.25.3.3. Overall Rescaling (applied to E1; default for E2)

The rescaling options in GENEV are used to insure that the overall mean $|E|^2$ is precisely one. The simplest and most effective way of achieving this is by summing the $|E|$ values determined using linear or profile scales,

and then applying the inverse of the average $|E^2|$. This is referred to as overall rescaling. It is mandatory for $|E1|$ and optional for $|E2|$.

4.25.3.4. Index Rescaling (`indexk` option for E2)

With index rescaling, different groups of reflections defined by a particular combination of hkl indices are rescaled so that the mean value of $|E^2|$ is one. This option may be applied only to $|E2|$. The conditions for each index group may be specified on the `indexk` line with the 15 parameters $p_1, m_1, n_1, p_2, \dots, n_3$. A reflection belongs to a particular index group provided its indices jointly satisfy the following three equations:

$$[h p_1 + k q_1 + l r_1] \bmod(m_1) = n_1$$

$$[h p_2 + k q_2 + l r_2] \bmod(m_2) = n_2$$

$$[h p_3 + k q_3 + l r_3] \bmod(m_3) = n_3$$

Each group may be specified by a separate `indexk` line or, in the case of the even-odd parity groups, with a single blank `indexk` line. Only the specified index groups will be scaled separately, the remainder will be scaled together. Judicious use of the index parameters will permit single reflections to be scaled in this way. Specific scale values may also be entered on the `indexk` line for this purpose. No attempt will be made to make the mean $|E^2| = 1$. in this case.

The index rescaling option must also be used carefully. As with the profile scale it can have the overall effect of reducing the structural content of the $|E|$ values. The study of S & H (1982) showed that, in general, it provided less reliable $|E|$ s than the overall rescaling option. Index rescaling can, however, be useful in the study of superstructure or hypersymmetry, since it ensures that groupings of reflections are given similar weight in the phasing process.

4.25.4. Choice Of Expectation Values

A squared normalized structure factor is the ratio of its scaled intensity to its expectation value. The expectation value for an intensity (or rather $|F^2|$) depends on what is known about the structure. If only the atomic contents of the unit cell are known, then the best estimate of $\langle F^2 \rangle$ is the random-atom approximation (see S & H, 1982, for definition). Using the random-atom $\langle F^2 \rangle$ in the normalization process provides $|E|$ values that will reflect how well the $|F|$ values conform to those expected for a random structure. Significant departures of individual $|E^2|$ values from 1.0 (the overall mean) indicate whether a reflection is sensitive to the non-random aspects of the structure. The larger the departure from 1.0, the more important that reflection will be to a phasing process designed to investigate the non-random aspects of a structure. This is the basis for most structure invariant procedures.

If the coordinates of a structure are known (i.e. refined) then the value of $\langle F^2 \rangle$ is simply calculated $|F^2|$, assuming atoms-at-rest. Application of this expectation value in the normalization procedure will result in all $|E^2|$ values being close to 1.0 (assuming of course, good data and a well-refined structure). Obviously $|E|$ values determined in this way are of very limited use in direct methods since all reflections have equal weight. Those that are most sensitive to the non-random aspects of the structure cannot be identified.

Contrasting the application of random-atom and refined-atom expectation values illustrates a very important aspect of the normalization process. Structure information used in the expectation value will reduce that particular contribution in the resulting $|E|$ values. In other words, the departures of $|E^2|$ values from unity reflect the differences due to structural information not used in evaluating the expectation value. In general, therefore, the higher-order expectation values, as provided with fragment information of type 2, 3, and 4 (see below), often have deleterious effects on the calculation of $|E|$ values. There will, however, be situations when selective attenuation of structural information from $|E|$ values, via the application of high-order expectation values, is extremely useful. The reduction of the dominant effects of a heavy atom or planarity are two obvious examples. In general, however, it is strongly recommended (S & H, 1982) that the random-atom expectation value

be used in the initial stages of a structure solution, even when additional structure information is known (note well the comments in the next section).

4.25.4.1. Application of Known Structure Information

The general problem of applying known structural information to the structure invariant process is described by Main (1976). The definitions of the different categories of structural information as used by GENEV have been detailed by S & H (1982). These are treated in GENEV as the following categories,

type 1 is for random atoms	(wilson on fragline)
type 2 is for random fragments	(random on fragline)
type 3 is for oriented fragments	(orient on fragline)
type 4 is for positioned fragments	(positn on frag line)

While fragment information of type 2, 3, and 4 may not provide $|E_2|$ values that are more reliable than $|E_1|$ values based on random-atom expectation values, it should always be included in the GENEV calculation when available. This is because the group structure factors which are calculated as part of the evaluation of the expectation value of F^2 , $\langle F^2 \rangle$, provide phase information that is extremely useful in subsequent stages of the phasing process. This phase information can be applied to $|E_1|$, as well as $|E_2|$, in later calculations.

4.25.5. Inflection-point least-squares

For a random-atom structure, the Wilson plot is a straight line defined by the overall thermal displacement parameter of the constituent atoms, and the overall scale of the measure structure factors. For a real structure, a Wilson plot will often show significant systematic deviations from this line due principally to the short-range interatomic distances in the structure. The scattering effects of translational symmetry on the radial distribution of intensities is known as Debye scattering. For the majority of light-to-medium atom structures, the gross effects of Debye scattering are very similar. For instance the nodes, antinodes and inflection-points of a Debye scattering curves calculated for interatomic distances ranging from 1.30 to 1.55Å in a 6-membered ring molecule are quite similar (H & S, 1982a). This means that for many structures inflection-points (the points where the Debye curve crosses the linear mean line) provide a means of finding a reliable linear fit to the Wilson plot, independent of the extent of the Debye scattering effects and the s^2 truncation of the data.

GENEV uses the Wilson plot ratios for the 5 ranges clustered about the two cardinal inflection-points. The s squared default values for these inflection-points are set at 0.15 and 0.26 Å⁻², but these may be changed for non-typical structures with the **ain1** and **ain2** input on the GENEV line. In addition to the clusters of 5 points fixing the two inflection-points, the five largest Wilson plot ratios are used to fix the low angle part of the least-squares line. The user should always check that the assumptions embodied in the inflection-point least-squares process are valid for each structure. The Wilson plot points used for this purpose are shown in the printed graph as at signs (@). It is recommended that GENEV be rerun with specified values of U and K (using the **fixu** and **fixk** options) or different inflection-points (using **ain1** and **ain2** values), if the least-squares fit is unsatisfactory.

4.25.6. Estimation Of $|E(hkl)|$ Errors

GENEV provides an estimate of the $|E(hkl)|$ errors using a procedure described by H & S (1982b). The principal source of error in $|E|$ values arises from inaccuracies in the measured structure factors. It follows that the legitimacy of the errors estimated in GENEV will depend on the precision of the σF values entered on the bdf. The second most important contributor to the $|E(hkl)|$ errors arises from fitting the linear or profile scaling functions to the Wilson plot (see below). The effect of Debye scattering on the Wilson plot has already been discussed, and this is taken into account when estimating the errors. The errors estimated for $|E_1|$ and $|E_2|$

are placed in the bdf for use in subsequent calculations. The error distribution for a typical structure is listed below.

4.25.6.1. Typical Error Statistics (for an R-factor of .04)

<i>Error for</i>	0.00	1.00	1.20	1.40	1.60	1.8	2.0	2.5	3.0
<i>/E/s ></i>									
<i> E from</i>	.25	.60	.70	.80	.90	1.0	1.1	1.3	1.5
linear K									
<i> E from</i>	.30	.65	.75	.85	.90	1.0	1.1	1.3	1.5
profile K									
<i>Error at</i>	.00	.05	.10	.125	.150	.175	.200	.225	.250
<i>mean s₂=</i>									
<i> E from</i>	.15	.17	.19	.21	.22	.27	.32	.35	.38
linear K									
<i> E from</i>	.20	.20	.25	.36	.26	.30	.31	.33	.37
profile K									

4.25.7. BDF Output Of Genev Items

The user may decide which GENEV items are output to *lrrefl:* of the bdf. If an **archiv** line is not entered, the following items are automatically output to the bdf.

Scales K	(item 100-)	in record <i>lrexpl:</i>
Overall U	(item 2)	in record <i>lrdset:</i>
Fragment count	(item 10)	in record <i>lrdset:</i>
E1	(item 1600)	in record <i>lrrefl:</i>
E2	(item 1601)
σ E1	(item 1602)
σ E2	(item 1603)
Group structure factor 1	(item 1606)
Group structure factor 2	(item 1607)
.....
Group structure factor N	(item 1605+N)

If any **archiv** lines are entered, the items 1600 to 1630 must be named explicitly to be output to the bdf. Particular care must be taken if fragment information is used in GENEV calculation. The number of group structure factors (items 1606 on) is equal to the number of fragments, except for type 3 fragments where there is one group structure factor for each point group. The user must also check if any extra type 1 fragments have been added by GENEV to balance the cell content. Subsequent calculations that use the group structure factor phases require that the correct number be present. It is important to note that if GENEV items 1600 to 1630 are present on the input bdf, they will not be transferred to the output bdf. These are purged from *lrrefl:* before new GENEV items are appended. Additional items, other than 1600 - 1630, may also be deleted from *lrrefl:* using the **archiv** lines. A maximum of 30 items may be added or deleted in this way.

4.25.8. Some Normalization Tips

Tip 1 Check that $\sin\theta/\lambda$ maximum on the **GENEV** line is as accurate as possible. The default value comes from

the bdf, otherwise it is set to 1.0. If *ADDREF* has been used with the two pass option, the accurate value will be stored.

Tip 2 Assess the precision of the data. If some weak data are missing from the input bdf, use the *fix11* option to compensate for this. If the weak data have not been processed with Bayesian statistics (e.g. negative $|F^2|$ have been set to 0), then the *baye* option can be used to apply a limited Bayesian treatment to $|F|$ and σF values.

Tip 3 Check what scaling and expectation options should be applied in the calculation of $|E2|$. These are fixed for $|E1|$.

Tip 4 If the values of either U or K need to be fixed, use the *fixu* and/or *fixk* options. These will apply only to $|E2|$.

Tip 5 The default rescaling mode for $|E2|$ is index rescaling applied to the eight parity groups, provided that *frag*, *prof*, or *fixu/k* options are not used. If any of these lines are entered, the default rescaling mode for $|E2|$ becomes 'overall'. Index rescaling may be specified explicitly with the *indexk* line(s) but care should be taken in selecting index groups appropriate to the problem.

Tip 6 Known structure information is entered using the *frag*, *site*, *sitea*, and *siteg* lines. Fragment information is entered for the asymmetric unit, as opposed to the *celcon* information which is entered for the whole cell. *sitea* lines containing coordinates in orthogonal Angstroms must be used for type 2 fragments. A *grid* line must precede the first *siteg* line entered. The *frag* line may be used to move the origin of atom coordinates that follow. This is sometimes useful for converting from type 3 to type 4 input.

Tip 7 The user may output the $|F^2|$ expectation values used in the GENEV calculation as items 1604 and 1605 in *lrrefl*. This can be efficient for very large structures when GENEV is repeated for different normalization parameters but the same fragment information. This is specified by putting *bexp* in the *GENEV* line and leaving out the *frag* and *site* lines.

Tip 8 Cell content information is extracted from the input bdf (if entered through the program *STARTX*). This may be replaced by entering *celcon* lines.

Tip 9 Always check that the items to be used in subsequent calculations (e.g. *GENSIN*, *GENTAN*, and *FOURR*) are to be output. In most cases the default items will be sufficient, but in special cases the *archiv* line(s) can be used to add the items required. It is good archival practice to regularly check the contents of *lrrefl*: and remove any items that are no longer needed.

4.25.9. File Assignments

- Reads structure factors from the input archive bdf
- Writes normalized structure factors (E's) to the output archive bdf

4.25.10. Examples

```
GENEV smax 0.52           :s max of all data is .52
```

$|E1|$ will be calculated with linear scale, random-atom expectation value, and overall rescale. $|E2|$ will be the same except for index rescaling using hkl parity groups. No $|E|$ values will be listed and $|E1|$ and $\sigma|E1|$ will be output on the bdf.

```
GENEV list 1.5
frag oriented           :specify type 3 fragment
```

```

site br1 .5 .5 0 *7 .5           :bromine in special position
site c1 .73 .57 .333
site n3 -.15 .44 .62
archiv 1600 1602 1606 1607      :add |E1|, s|E1|, gsf1, gsf2

```

|E2| will be calculated with a linear scale, overall rescale and an expectation value derived from the type 3 fragment of atoms Br1, C1, and N3 and the remaining atoms (i.e. balance of cell contents) as a type 1 fragment.

```

GENEV smax .33           fixu .04 baye fill
indexk                   :use index rescale (parity hkl) for |E2|
archiv 1601 1603 -1800    :add |E2|, sigma|E2|, delete |Fc| from bdf

```

|E2| will be calculated with linear scale (with $u=0.04$), random-atom expectation value, and index rescaling with parity groups. All input |F|s and σ |F|s are treated with limited Bayesian statistics and the Wilson plot is adjusted for missing data. Only |E2| and σ |E2| are added to the bdf; |Fc| is removed.

```

GENEV dset 3 bexp prof
indexk 1 1 1 4 1         :set index scale group 1
indexk 1 1 1 3 *16.5     :set index scale group 2 and set scale

```

|E2| will be calculated with a profile scale, random-atom expectation value, and index rescaling based on the groups $(h+k+l)\bmod 4=1$, $(h+k+l)\bmod 4=3$, and the remainder. The scale of the second index groups will be fixed at 0.5. In this example the two |E| estimates will be output without their error values.

4.25.11. References

- French, S. and Wilson, K. 1978. *On the Treatment of Negative Intensity Observations*. Acta Cryst. **A34**, 517-525.
- Hall, S.R. and Subramanian V. 1982a. *Normalized Structure Factors. II. Estimating a Reliable Value of B*. Acta Cryst., **A38**, 590-598.
- Hall, S.R. and Subramanian, V. 1982b. *Normalized Structure Factors. III. Estimation of Errors*. Acta Cryst. **A38**, 598-608.
- Luger, P. 1980. *Modern X-ray Analysis on Single Crystals*. New York: de Gruyter.
- Main, P. 1976. *Recent Developments in the MULTAN System - The Use of Molecular Structure..* Crystallographic Computing Techniques, F.R. Ahmed, K. Huml, B. Sedlacek, eds., Munksgaard. Copenhagen: 97-105.
- Stout, G.H. and Jensen, L.H. 1968. *X-ray Structure Determination: A Practical Guide*. New York: MacMillan.
- Subramanian, V. and Hall, S.R. 1982. *Normalized Structure Factors. I. Choice of Scaling Function*. Acta Cryst. **A38**, 577-590.
- Wilson, A.J.C. 1942. *Determination of Absolute from Relative X-ray Intensity*. Nature **150**, 151-152.

4.26. **GENSIN**: Generate triplets and quartets

Authors: V. Subramanian and Syd Hall

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907 Australia

*GENSIN generates triplet and quartet structure invariant relationships. Group structure factor information entered from the input bdf (from **GENEV**) can be used to enhance the conditional probabilities of invariant relationships and to improve the estimate of their expected phase value.*

4.26.1. Introduction

Most direct methods approaches use the *triplet* structure invariant relationship

$$\psi_3 = \phi(h_1) + \phi(h_2) + \phi(h_3)$$

$$\text{where } h_1 + h_2 + h_3 = 0 \quad (1)$$

The value of ψ_3 is conditional on the probability expression

$$A = 2 a |E(h_1) E(h_2) E(h_3)| \text{ where } a = \sum_N f^3 / (\sum_N f^2)^{3/2} \quad (2)$$

N is the number of atoms in the unit cell. For uniform-atom structures a tends to $1/N^{1/2}$. The probability distribution of ψ_3 , conditional on A , (Cochran, 1955) is

$$P(\psi_3 | A) = \exp(A \cos \psi_3) 2\pi I_0(A) \quad (3)$$

The importance of the magnitude of A (and therefore the E -values and N) may be seen from the variation of $P(\psi_3 | A)$ for the typical values of A .

ψ_3	0°	45°	90°	135°	180°
$P(\psi_3 3.)$.65	.27	.03	.004	.002
$P(\psi_3 1.)$.34	.25	.13	.06	.05

Direct methods procedures depend critically on being able to predict the value of ψ_3 and on being able to apply triplet relationships as a series of equations. For large values of A , ψ_3 has a high probability of being 0. and this makes large- A triplets particularly important for these processes.

For the *quartet* structure invariant relationship

$$\psi_4 = \phi(h_1) + \phi(h_2) + \phi(h_3) + \phi(h_4) \text{ where } h_1 + h_2 + h_3 + h_4 = 0 \quad (4)$$

the value of ψ_4 depends principally on the probability factor

$$B = 2\mathbf{b} |E(h_1) E(h_2) E(h_3) E(h_4)| \text{ where } \mathbf{b} = \sum_N f^4 / (\sum_N f^2)^2 \quad (5)$$

For uniform-atom structures \mathbf{b} tends to $1/N$. The probability distribution of ψ_4 , ignoring the cross-vectors magnitudes $E(h_1+h_2)$, $E(h_1+h_3)$, and $E(h_1+h_4)$ (referred to later as E_{12}, E_{13}, E_{14}), may be written as follows (Hauptman, 1976)

$$P(\psi_4 | B) = \exp(B \cos \psi_4) / 2\pi I_0(B) \quad (6)$$

This function is similar to $P(\psi_3 | A)$, except that the value of B will tend to be much smaller for large structures. The probability distribution of ψ_4 is dependent on more than the principal vectors that go to make up B and is more correctly

$$P(\psi_4 | B, E_{12}, E_{13}, E_{14}) = \exp(-B \cos \psi_4) / 2\pi I_0(Z) \quad (7)$$

where Z is a function of the seven vectors $E(h_1)$ to E_{14} . An important property of the probability expression (7) is that if the cross-vectors are large, then expression (6) holds, but if the cross-vectors are *small* then expression (7) approximates as

$$P(\psi_4 | B, E_{12}, E_{13}, E_{14}) = \exp(-B \cos \psi_4) / 2\pi I_0(2B) \quad (8)$$

The importance of this result may be illustrated for a fixed value of B (for example, 3.) and for small and large cross-vectors (XV)

ψ_4	0°	45°	90°	135°	180°
P(ψ_4 3., <i>large XV</i>)	.65	.27	.03	.004	.002
P(ψ_4 3., <i>small XV</i>)	.0001	.0003	.002	.005	.007

Because of the dependence of ψ_4 on the magnitude of the cross-vectors, quartets are usually grouped into two classes according to the sum of cross-vector magnitudes.

$$XVsum = |E_{12}| + |E_{13}| + |E_{14}| \quad (9)$$

If XVsum is greater than a certain threshold (e.g. XSHI), a quartet is referred to as a *positive quartet* (because $\cos\psi_4$ should be positive). If it is less than a lower threshold (e.g. XSLO), a quartet is known as a *negative quartet* (because $\cos\psi_4$ should be negative). GENSIN estimates the value of XVSUM and various procedures can be adopted by the user to control the generation of quartets using this sum.

4.26.2. Application Of Group Structure Factors

The normalization program, *GENEV*, converts known structural information into one or more group structure factors G(h) for each reflection. These group structure factors are used in *GENEV* to calculate an expectation value for $F^2(h)$ where M depends on the number of molecular fragments and the nature of the fragment information (see *GENEV* for details).

$$\langle F^2(h) \rangle = \sum_M G^2(h) \quad (10)$$

GENEV outputs the group structure factor values as the magnitude G(h) and the phase g(h). Knowledge of the structure influences the values expected for ψ . If the atomic parameters are known to a certain precision, then G(h) and g(h) values (which in this instance are the same as F(h) and $\phi(h)$) may be used to predict ψ to the same precision. The group structure factor is in fact an important component in the conditional probability expressions for ψ_3 and ψ_4 .

4.26.2.1. Triplets

Main (1976) modifies the value of A to the form:

$$A' = 2a' |E(h_1) E(h_2) E(h_3)| \quad (11)$$

where

$$a' = \sum_M G(h_1, h_2, h_3) / [\langle F^2(h_1) \rangle \langle F^2(h_2) \rangle \langle F^2(h_3) \rangle]^{1/2} \quad (12)$$

The correspondence between a' and a in expression (3) is quite apparent when the only knowledge of a structure is its atomic content. Then each $\langle F^2(h) \rangle = \sum f^2(s)$; each $G(h_1, h_2, h_3) = \sum f^3(s)$ and $a' = a$. With increasing structural information the value of a' may differ substantially from a . The most important term in expression (12) is the joint group structure factor $G(h_1, h_2, h_3)$ which is calculated directly for each invariant from the known structural information (see Main, 1976). This calculation is, however, a time-consuming task even for small structures. A more efficient approach involving a minimal loss in precision is the use of the individual group structure factors of the form (Hall, 1978):

$$a' = \sum_M [G(h_1) G(h_2) G(h_3)] / [\langle F^2(h_1) \rangle \langle F^2(h_2) \rangle \langle F^2(h_3) \rangle]^{1/2} \quad (13)$$

4.26.2.2. Quartets

This approximation applied to quartet relationships gives

$$B' = 2b' |E(h_1) E(h_2) E(h_3) E(h_4)| \quad (14)$$

$$\text{where } b' = \frac{\sum_M [G(h_1) G(h_2) G(h_3) G(h_4)]}{[\langle F_2(h_1) \rangle \langle F_2(h_2) \rangle \langle F_2(h_3) \rangle \langle F_2(h_4) \rangle]^{1/2}} \quad (15)$$

In this way fragment information is used to predict the distribution of ψ_3 and ψ_4 according to $P(\psi_3|A')$ and $P(\psi_3|B', \text{XVsum})$. Most importantly, however, the probability terms A' and B' from (13) and (14) are complex

$$A' = |A'| \exp(i \Psi_3) \text{ and } B' = |B'| \exp(i \Psi_4) \quad (17)$$

with the phase values Ψ_3 and Ψ_4 , which are estimates of ψ_3 and ψ_4 , respectively. The reliability of Ψ_3 and Ψ_4 as phase estimates depends on the precision of the structural information, and the magnitude of A' and B' , respectively. For random-atom structures (i.e., fragment information type-1 in *GENEV*) $A' = A$ and $B' = B$, and $\Psi_3 = \Psi_4 = 0$. If random-fragment (type-2) information is used in *GENEV*, the Ψ_3 and Ψ_4 values are also assumed to be zero (this is a limitation of using approximation (15) instead of (14)). For type-3 and type-4 fragments information the values of Ψ_3 and Ψ_4 may be non-zero. In the following description and input line formats Ψ_3 and Ψ_4 values are referred to as the fragment ψ estimates "QPSI".

4.26.3. Origin And Enantiomorph Definition

In addition to generating structure invariants, GENSIN provides the conditions for the origin and enantiomorph definition of the cell. Fixing the origin and enantiomorph is a necessary first step in the *GENTAN* phase extension process. It is performed automatically; the user may, however, override this procedure using phases selected according to the definitions output by GENSIN. The conditions for specifying the origin in terms of structure factor seminvariant phases is detailed by Hauptman and Karle (1956) and Karle and Hauptman (1956). Application procedures for applying these conditions are described by Stewart and Hall (1971), Luger (1980), and Hall (1983).

It should be noted that for GENSIN and *GENTAN* the seminvariant vector conditions are always in terms of the input indices. It is therefore unnecessary to transform centred indices to primitive indices for the purposes of origin specification. Details of the seminvariants vectors for centred space groups are described by Hall (1982).

The origin of a cell is fixed by specifying the structure factor phases of p linearly-independent reflections. The value of p ranges from 0 to 3, and is determined by the space group symmetry.

Any reciprocal lattice vector \mathbf{h} is a linear combination of p origin defining vectors $h(1) \dots h(p)$

$$\mathbf{h} = \sum_p n_j \mathbf{h}_j \quad (18)$$

where n_j is any integer value. This relationship may be expressed as the vector transformation

$$\mathbf{h} = \mathbf{n} \mathbf{H} \quad (19)$$

where \mathbf{n} is the set of integers $n(1), \dots, n(p)$ and \mathbf{H} is the set or origin defining reflections $h(1), \dots, h(p)$. The linear relationship of a reflection \mathbf{h} to the set of origin defining reflections \mathbf{H} is given by

$$\mathbf{n} = \mathbf{h} \mathbf{H}^{-1} \quad (20)$$

A reflection vector \mathbf{h} may also be transformed into the seminvariant indices \mathbf{u} by the operations

$$\mathbf{u} = \mathbf{h}' \pmod{\mathbf{m}} = (u, v, w) \quad (21)$$

$$\text{and } \mathbf{h}' = \mathbf{V} \mathbf{h} \quad (22)$$

where \mathbf{V} is the seminvariant vector matrix and \mathbf{m} the seminvariant moduli (Hauptman and Karle, 1956). A necessary requirement of any set of origin defining reflections is that the matrix of seminvariant indices \mathbf{U}

$$\mathbf{U} = (\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_p) \quad (23)$$

has the magnitude

$$|\mathbf{U}| = +1 \text{ or } -1. \quad (24)$$

The linear relationship of any structure factor phase $\phi(\mathbf{h})$ may be expressed in terms of the \mathbf{u} from (21) as:

$$\mathbf{n}' = \mathbf{u} \mathbf{U}^{-1}. \quad (25)$$

If \mathbf{Q} is defined as the set of origin defining phases

$$\mathbf{Q} = (\phi(\mathbf{h}_1), \phi(\mathbf{h}_2), \dots, \phi(\mathbf{h}_p)) \quad (26)$$

The seminvariant phase due to linear relationship of \mathbf{h} to \mathbf{H} may be derived from the linear combination of these phases (Hauptman and Karle, 1956)

$$q(\mathbf{h}) = \mathbf{n}' \mathbf{Q}. \quad (27)$$

If seminvariant phase $q(\mathbf{h})$ is equal, modulus π , to the phase of vector \mathbf{h} , then the value of $\phi(\mathbf{h})$ is independent of the enantiomorphous structure. If $q(\mathbf{h})$ is significantly different (modulus π) to $\phi(\mathbf{h})$, then the enantiomorph may be specified by fixing $\phi(\mathbf{h})$ at one of its two possible values.

For space groups where $\phi(\mathbf{h})$ is restricted to one of two values (e.g. for $P2_12_12_1$ $\phi(\mathbf{u}\mathbf{u}0) = \pi/2$ and $\phi(\mathbf{g}\mathbf{u}0) = 0, \pi$), the calculation of $q(\mathbf{h})$ from integer set \mathbf{n}' and its application to the phase set \mathbf{Q} provides a straightforward approach to identifying the formal requirements of enantiomorphic discrimination (Hall, 1983).

If all restricted phases have values of $\phi(\mathbf{h}) = q(\mathbf{h})$, then a reflection with a non-restricted phase value must be used to specify the enantiomorph. For these space groups, $q(\mathbf{h})$ indicates the range of values $\phi(\mathbf{h})$ should have to separate satisfactorily the enantiomorphs. Typically a $\phi(\mathbf{h})$ value would be permuted in a multisolution process, to a series of values for $q(\mathbf{h}) + \pi/4$ to $q(\mathbf{h}) + 3\pi/4$ in increments of $\pi/4$. Differences between $\phi(\mathbf{h})$ and $q(\mathbf{h})$ of less than $\pi/6$ will not provide strong enantiomorphic discrimination and are likely to lead to instability in the phasing process. For examples of enantiomorphic discrimination see Hall (1983).

4.26.4. Notes On GENSIN Parameters

4.26.4.1. Number of Invariants

In the default mode GENSIN generates both triplet and quartet structure invariant relationships. Invariant types may be specified by the user with the **trip** and **quar** control lines. The number of structure invariants generated is determined by a range of parameters, including the number of generators, the magnitude of the E values, and the magnitude of the A and B thresholds. In default mode, the maximum number of invariants for either type is set at 2000.

4.26.4.2. Number of Generators

The reflections used in the generation process are selected from the largest E_n , where n designates the E-type (1 or 2) output by *GENEV*. The default value of n is 1. The number of generators is controlled by the user via the **gener** line, or set automatically according to the algorithm

$$\text{MAXGEN} = \max(10000, 150 + \text{NNHA} * (4 + \text{ICNT} + 1/\text{NEQP})),$$

where $\text{ICNT}=1$ if centrosymmetric and $\text{ICNT}=0$ if noncentrosymmetric, NNHA is the number of non-hydrogen atoms in the molecule, and NEQP is the number of general equivalent positions.

4.26.4.3. Psi(0) Triplets

Triplet invariants are always generated for up to 100 of the smallest E-values independent of the TRIP parameters. These are used subsequently in the ψ (zero) figure-of-merit tests in *GENTAN*, and are referred to as $\psi(0)$ triplets (Cochran and Douglas, 1957).

4.26.4.4. Estimation of Psi From bdf Phases

GENSIN provides, via the `psical` line, the facility to calculate ψ from phases and structure factor values stored on the bdf. The ϕ and $|F|$ values on the bdf may be from a previous *GENTAN* run, an *FC* calculation on a partial structure, or the back transform of modified density for protein structures. The bdf *lrrefl*: ID numbers for ϕ and F are assumed to be n750 and n751 unless otherwise specified on the `psical` line.

4.26.4.5. Application of Fragment QPSI

The inclusion of fragment QPSI values (see the `GENSIN` line and description above) in the invariant generation process will modify the calculated A and G values, and therefore change the number and nature of invariants generated. This is quite independent of the E used (i.e., E1 or E2). It is recommended that E1, based on random-atom expectation values, are always used, except in special circumstances (see Subramanian and Hall, 1982; Hall and Subramanian, 1982a,b). This is because the QPSI values contain the group structure phase information of the fragment, and E1 has been shown to provide more reliable structure invariant relationships. As a rule-of-thumb E2 should be treated as a second option that the user can invoke in cases of severe non-randomness (e.g., hypersymmetry, super-structures or very dominant heavy atoms).

4.26.4.6. Adjusting |E| Values

The `change` control line is available for modifying the magnitude of specific E or $\sigma(E)$ values. `change` can be used to enhance or to suppress a particular E in the generation process by increasing or decreasing its E value. It is also useful for running "replica" tests against other software and different machines. Please note that `change` lines must be entered in the order of reflection data on the input bdf, and be the last control lines entered.

4.26.4.7. Types of Quartet Invariants

The nature of quartet invariants generated by GENSIN is determined largely by the cross-vector magnitudes. The types and magnitudes of cross-vectors permitted during the generation process are controlled via the parameters IXVF, XVMN, XVMX, XSLO, and XSHI on the `quar` line. The XSLO and XSHI parameters apply only to quartets with cross-vectors inside the data sphere. The cross-vector limits XVMN and XVMX are applied to individual cross-vector E values. If an individual cross-vector E lies outside the range XVMN to XVMX the quartet is rejected. In the default mode, the sum of the cross-vector magnitudes XVsum (eqn (9)) is calculated for all cross-vectors inside the data sphere with XVsum < XSLO and XVsum > XSHI.

Quartets generated by GENSIN are used in several different ways in the subsequent calculations. Typically quartets are divided into three categories: those with cross-vector sums above an upper threshold XSHI (known as positive quartets), those below a lower threshold XSLO (known as negative quartets), and those in between. The last category is often not used in the phasing process because of the unpredictability of the ψ_4 values. The user can specify these upper and lower thresholds, XSHI and XSLO, with the `quar` line for both the GENSIN and *GENTAN* calculations.

It is usual to use only quartets with cross-vectors inside the data sphere. The XVsum is then able to be calculated and a prediction made about the value of ψ_4 . There are, however, some drawbacks to this approach. When XVsum is greater than XSHI it is probable that the quartet generated will in fact be equivalent to a combination of three triplet invariants also generated by the GENSIN process. The phase relationships provided by positive

quartets tend therefore to reinforce, rather than add to, those provided by the triplets. The phase "pathways" provided by quartets will, of course, be different to those of triplets but the generators they connect will, in effect, be the same.

Quartets with $XSLO < XVsum < XSHI$ are usually not redundant to triplets but are less useful for the reasons already discussed.

Negative quartets with $XVSUM$ less than $XSLO$ provide completely different phase information to triplets but are very few in number. For this reason they are used in GENTAN for a figure-of-merit parameter.

In contrast, quartet invariants with one or more cross-vectors outside the data sphere, provide relationships that cannot be represented by a combination of triplets. These quartets provide new phase pathways and, as such, could prove crucial in particularly difficult solutions. The disadvantage of these "extra-terrestrial" quartets is the lack of cross-vector information and, therefore, the inability to predict the value of ψ_4 . However, it may be assumed that ψ_4 for these quartets has a distribution based on B (just as ψ_3 is a function of A) and an overall reliability comparable to that of triplets (ignoring the relative magnitudes of A and B). See Examples 3 and 4.

4.26.5. File Assignments

- Reads E values from the input archive bdf
- Writes structure invariant relationships to the file `inv`

4.26.6. Examples

GENSIN

Generate triplet and quartet invariants to a maximum of 2000 using type-1 E values. Only quartets with cross-vectors inside the data sphere will be output. QPSI values will be applied if fragment information is on the input bdf. Print invariant totals for all generators.

```
GENSIN nqpi           :do not use QPSI information
gener *2 300         :use top 300 E values
trip yes 1.5 3000    :set max A and max triplets
quar no             :do not generate quartets
print *2 1 50 100    :print SI for gens 1-50 to N of 100
```

Generate maximum of 3000 triplets with A values greater than 1.5 from 300 generators. Available QPSI values are not applied. Structure invariants for the top 50 generators are printed provided all generator numbers are ≤ 100 .

```
GENSIN smax .45      :exclude all s values >.45
quar yes 0.75 *8 1. 5. :for Q4 B>.75 and XVsum>5 or <1.
change 1 7 3 3.2      :make E = 3.2
change 2 3 -4 2.75    :make E = 2.75
```

Generate triplets and quartets for generators selected from Es with $s < .45$. Quartets will be accepted if $B > .75$ and has a cross-vector sum (all cross-vectors inside the data sphere) 5. or 1. The E value of reflections 1, 7, 3 and 2, 3, -4 will be modified on input.

```
GENSIN
quar *5 outxv           :generate quartets with outside cross-vectors
```

Generate triplets and quartets. The quartets must have at least one of their three cross-vectors outside the data sphere.

```
GENSIN
quar *5 outxv -5. 0.    :all cross-vectors outside sphere
```

Generate triplets and quartets. The quartets must have all three cross-vectors outside the data sphere.

4.26.7. References

- Cochran, W. and Douglas, D. 1957. *The Use of a High-speed Digital Computer for the Direct Determination of Crystal Structures*. Proc. Roy. Soc. **A243**, 281.
- Hall, S.R. 1981. *A Procedure for Random-access to Reflection Data*. J. Appl. Cryst. **14**, 214-215.
- Hall, S.R. 1982. *Seminvariant Vectors for Centred Space Groups*. Acta Cryst. **A38**, 874-875.
- Hall, S.R. 1983. *A Procedure for Identifying Enantiomorph-Defining Phases* Acta Cryst. **A39**, 22-26.
- Hall, S.R. and Subramanian, V. 1982a. *Normalized Structure Factors. II. Estimating a Reliable Value of B*. Acta. Cryst. **A38**, 590-598.
- Hall S.R. and Subramanian, V. 1982b. *Normalized Structure Factors. III. Estimation of Errors*. Acta Cryst. **A38**, 598-608.
- Hauptman, H. 1976. *Some Recent Advances in the Probabilistic Theory of the Structure Invariants*. Crystallographic Computing Techniques. F.R. Ahmed, K. Huml, B. Sedlacek, eds., Munksgaard. Copenhagen: 129-130.
- Hauptman, H. and Karle, J. 1956. *Structure Invariants and Seminvariants for Noncentrosymmetric Space Groups*. Acta. Cryst. **9**, 45-55.
- Karle, J. and Hauptman, H. 1956. *Theory of Phase Determination for the Four Types of Non-centrosymmetric Space Groups 1P222, 2P222, 3P12, 3P22*. Acta. Cryst. **9**, 635-654.
- Luger, P. 1980. *Modern X-ray Analysis on Single Crystals*. de Gruyter: New York.
- Main, P. 1976. *Recent Developments in the MULTAN System - The Use of Molecular Structure*. Crystallographic Computing Techniques F. R. Ahmed, K. Huml, B. Sedlacek, eds., Munksgaard. Copenhagen: 97-105.
- Stewart, R.F. and Hall, S.R. 1971. *X-ray Diffraction . Determination of Organic Structures by Physical Methods*. F.C. Nachod and J.J. Zuckerman, eds., Academic Press: New York, 74-132.
- Subramanian, V. and Hall, S.R. 1982. *Normalized Structure Factors. I. Choice of Scaling Function*. Acta. Cryst. **A38**, 577-590.

4.27. **GENTAN**: Tangent phasing

Author: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands WA 6907, Australia

GENTAN uses triplet and/or quartet structure invariant relationships in a general tangent formula to propagate and refine structure factor phases. The phases required to define the cell origin and enantiomorph are specified automatically, or may be selected (wholly or partly) by the user.

4.27.1. Tangent Formula

The tangent formula approach of Karle and Hauptman (1956) is the most widely used procedure for the extension and refinement of structure factor phases. Users who are relatively unfamiliar with this field are well advised to read summaries of these methods by Karle and Karle (1966) and Stewart and Hall (1971). Practical guidelines and background information on the application of the tangent formula is detailed in the proceedings of the 1975 IUCr Computing School (Ahmed, 1976).

The tangent formula is a straightforward computational approach to summing phases estimated with the *triplet invariant* relationship

$$\psi_3 = \phi(h_1) + \phi(h_2) + \phi(h_3), \text{ where } h_1 + h_2 + h_3 = 0 \quad (1)$$

The conditions for which ψ_3 has a value close to zero have been discussed in the *GENSIN* introduction. If there are **m** triplet relationships then the cyclic nature of the contributing phases requires that the mean value of $\phi(h_i)$ has the trigonometric form

$$\tan \phi(h_1) = \sum_m A_i \sin\{-\phi(h_2) - \phi(h_3)\}_i / \sum_m A_i \cos\{-\phi(h_2) - \phi(h_3)\}_i \quad (2)$$

Because contributing phases can vary in reliability, a weighted tangent formula is required to average the phases as follows,

$$\tan \phi(h_1) = \sum_m A_i w_i \sin\{-\phi(h_2) - \phi(h_3)\}_i / \sum_m A_i w_i \cos\{-\phi(h_2) - \phi(h_3)\}_i \quad (3)$$

The quantity w_i is a measure of the joint reliability of the two phases contributing to the *i*'th triplet invariant. This approach can be applied to any order of structure invariant. A *general* expression for structure invariants of order **n** is

$$\psi_n = \sum_n \phi_i$$

$$\text{provided that } \sum_n h_i = 0 \quad (4)$$

It follows from equation (3) that the general tangent formula suitable for application to these invariants has the form shown in equation (5) (Hall, 1978). X is A, B, . . . according to the order of the invariant.

$$\tan \phi(h_1) = \sum_m w_i X_i \sin \{ \psi_n - \sum_{n-1} \phi_j \}_i / \sum_m w_i X_i \cos \{ \psi_n - \sum_{n-1} \phi_j \}_i \quad (5)$$

The general tangent formula also permits the use of non-zero ψ values. This is particularly important if oriented or positioned fragment information is used to estimate ψ from the group structure factors (see the *GENEV* and *GENSIN* documentation and the section on partial structure information below). In strongly non-random structures ψ can depart significantly from zero even for invariants with large X values.

It has also been discussed in *GENSIN* that Ψ may be expected to be closer to π than 0 for negative quartets (see the *GENSIN* documentation for definition). The negative quartets are used by *GENTAN* for FOM purposes only, unless the `active` option is entered on the `invar` line (then negative quartets are used actively and *not* for the FOM).

4.27.2. Starting the tangent process

The tangent phasing process is usually initiated with a few "known" phases. From these phases additional values are determined through the application of the tangent formula to connecting invariant relationships. In turn, new phases are used to expand the phasing process further.

The tangent refinement of a phase set stops when the estimated phases converge to a constant α^2 value. Convergence occurs when the refined phases are self-consistent with the structure invariant relationships and the refinement constraints applied (e.g., weighting scheme). Due to the pyramidal nature of the phasing process (i.e., a few phases determine many), the final phase set is also strongly dependent on the starting phases. To a large extent the success or failure of the multisolution method is determined by the actual values of the starting phases. This is why a great deal of the computational effort goes into the selection of starting reflections.

Where do the "known" starting phases come from? Some may be specified directly in order to fix the origin in the cell. These are known as origin defining reflections (ODR). ODR phases alone are usually insufficient to reliably start the phasing process. The larger the number of starting phases, the less dependence there is on the reliability of a few invariant relationships and the higher the likelihood of obtaining a correct solution.

Where do the additional "known" phases come from? Because phase values are usually not known before a solution, one way is to assign trial phase values to a limited number of reflections and permute these in a series of separate tangent refinements. Permuted starting phases form the basis for the multisolution approach to the tangent phasing process.

4.27.2.1. Selecting Origin Defining Reflections (ODR)

The first step in the GENTAN calculation is to specify sufficient phases to define uniquely the cell origin. The formal conditions required to do this have been detailed in the GENSIN documentation. The specification of phases to fix the cell origin is to be done automatically unless the user intervenes via the **phi** or **assign** control lines. All ODR phases entered are checked for validity by the program. If they are incorrect or insufficient, additional or new phases will be automatically selected to satisfy the origin fixing requirements.

4.27.2.2. **permute** and **magic**: Selecting starting phases

The selection of starting phases which are involved in reliable structure invariant relationships is critical to the success of these methods. In GENTAN, the generator reflections are sorted by a convergence-type process (Woolfson, 1976) that maximizes the connections through structure invariant relationships between ODR phases and additional starting phases. This is called the MAXCON(nection) procedure. If the structure is non-centrosymmetric this procedure is also used to specify one or more EDR (enantiomorph defining reflection) phases to fix the enantiomorphic form of the structure. This is done automatically by the MAXCON procedure unless the EDR phase is selected manually by the user. Additional phases are specified in the MAXCON procedure as requested (see field 2, **select** line). The MAXCON procedure sorts generator reflections in order of descending connectivity and this sort order is used in all subsequent operations. The sorted order of generators is referred to as the phase path. The origin, enantiomorph, and any other MAXCON-selected phases are at the beginning of this path.

In addition to the starting phases selected for maximum connectivity, phases are specified to optimize the rate of "phase extension" to the remaining generators. This is referred to as the MAXEXT(ension) procedure. Extra phases are often needed in the starting set to accelerate phase propagation along the sorted phase path, rather than maximizing the connections between the initial starting set. It is important to emphasise that the MAXCON approach insures that there are strong links between the initial starting set and other strong generators, while the MAXEXT procedure provides additional phases which enhance the rate at which new phases are generated in one pass down the phase path. The criteria used in the MAXEXT procedure to measure the rate of "phases extension" is specified in field 6 of the **select** control line.

Both the MAXCON and MAXEXT procedures use all available triplet and quartet invariant relationships unless instructed otherwise (see field 5, **select** line). This option is useful if the automatically selected starting phases fail to provide a solution. Note that field 1 of the **invar** line has precedence over this option.

The permutation of phases assigned to starting reflections is performed in several ways. In the **permute** mode (see field 1, **select** line), each starting phase (other than the ODRs) is assigned a value according to whether it is restricted or unrestricted. Restricted phases can have two values (separated by π) and unrestricted phases

are usually assigned four different values (separated by $\pi/2$). In this mode therefore, the total number of phase sets tested is increased by a factor of two, or four, for each restricted, or unrestricted, phase specified. The number of phase sets increases very rapidly with the number of unrestricted starting phases.

An alternative approach to phase permutation is available using the magic integer procedure of White and Woolfson (1975). The **magic** option (field 1, **select** line) permutes the unrestricted starting phases so that the number of phase sets increases at a much lower rate. If a large number of unrestricted phases is needed to start an analysis, the magic integer permutation approach can provide a considerable reduction in computing time. The **magic** permutation procedure does increase the initial rms error of phases but for large analyses this is usually a worthwhile tradeoff for benefits of more unrestricted starting phases.

4.27.2.3. **random**: Specifying random starting phases

The inherent fragility of a phasing process started with a limited set of known phases and extended sequentially to all other phases has already been discussed. The success of this procedure hinges on the reliability of a few individual structure invariants. This is particularly critical in the early stages of the extension process when new phase estimates are often determined from one or two relationships. An incorrect phase estimate at this stage will frequently cause the phasing procedure to fail.

The random phase approach of Yao Jia-xing (1981) is an alternative to using a limited set of permuted phases. In the **random** mode (field 1, **select** line) random phases are assigned to all generators except the origin defining reflections. The use of random starting phases lessens the dependence on a small number of critical relationships by insuring that all invariants are immediately involved in the phase refinement process. Weights of each refined phase are used to filter the phase extension process. When the weight $w(h)$ of phase estimate $\phi(h)$ exceeds its starting value (default is 0.25 - see field 9, **select** line), this phase replaces the random starting value. This mode is particularly useful for large structures in low symmetry where there are few relationships among generators, and for strongly non-random structures where invariant relationships with large probability factors may still be suspect. Strong enantiomorph definition is also possible through the application of random starting phases.

In the **random** start mode, tangent refinements are performed on different random starting sets until either a correct solution is identified, or the phase set maximum is reached. The user may specify a random number generator "seed" (field 10, **select** line) and in this way insure that repeat runs employ different random starting phases. Alternatively, the default seed will insure that the same random phases will be generated, if this procedure is desired in a re-run.

4.27.3. Tangent phase extension and refinement

The extension and refinement of phases is simultaneous in the tangent process. When a structure invariant relationship contains only one unknown phase then this phase is estimated from the relationship. This is referred to as a phase extension. If all phases in an invariant are known, then each member phase is determined by a combination of the others. When a phase occurs in more than one such invariant, the estimate of its value is averaged or refined. Repeated averaging of phases in this way is referred to as phase refinement. As discussed earlier, the generator reflections and related invariants are sorted into an optimal phase path during the selection of starting phases. Reflections and invariants are subsequently processed in this sequence and one pass down this list is known as a single refinement iteration. The maximum number of refinement iterations may be specified by the user via the **refine** control line (field 3). The default value is 30.

The reliability of each phase estimate $\phi(h)$ is gauged by the calculated value of $\alpha(h)$ (see equation (9) below). Phases are accepted for phase extension if the value of $\alpha^2(h)$ is above a threshold (see field 5, **refine** line). This threshold is reduced with each iteration (see field 7) thus insuring that the most reliable phases are used in the early iterations.

Only the top sorted generators (those used in the MAXEXT selection process (see field 4, **select** line)) are phased in the early iterations. When the average value of $\alpha^2(\mathbf{h})$ changes less than a specified percentage (see field 7, **refine** line) additional generators (50, or 25% of total, whichever is greater) are added to the phasing process. The weight of each phase, is also used to control phase extension and refinement. A weight threshold is set for the duration of the refinement (see field 4, **refine** line) and is used to reject less reliable phases.

There are two different methods for propagating phases in GENTAN; **cascade** and **block**. These two modes differ only in the point in the tangent iteration where the phase is estimated. The difference in procedure can, however, have a profound effect on phase convergence and stability.

4.27.3.1. **cascade: Phase estimation DURING an iteration**

In the **cascade** mode phases are evaluated in sequential order along the phase path, and accepted or rejected immediately on estimation. Each phase estimate is made separately and tested for acceptability, based upon $\alpha^2(\mathbf{h})$ and w , before proceeding to the next phases estimate. An accepted phase is then available for phasing reflections further down the phase path. In this way phase information cascades down the phase path and provides the most rapid phase extension possible per iteration.

The two main characteristics of this process are that it is very dependent on the order of the phase path, and that phase values change during an iteration. This is important because three or four phases are dependent on a common triplet or quartet. If the ψ of this invariant changes substantially during the course of a single iteration the phase estimates tend to be dominated by phases near the end of the phase path. As these phases are usually of lower reliability, some instability in the refinement process can result. This may show itself as phase oscillations from iteration to iteration. This will be a problem if the value of ψ is strongly non-zero; as may be the case in heavy-atom or strongly planar structures.

4.27.3.2. **block: Phase estimation AFTER a tangent iteration**

The **block** procedure differs from the **cascade** method in that known phase values remain fixed during a tangent iteration. New phase estimates are only calculated at the completion of an iteration. This insures that phase estimates are made from fixed phase values, and the ψ values of invariants remain constant for the entire iteration. Phases are therefore determined as a block defined by the current length of the phase path. This procedure will not propagate phases as rapidly as the **cascade** mode but it is less susceptible to phase instability when values of ψ depart significantly from zero. It is the recommended procedure for strongly non-random structures.

4.27.4. Weighting

The tangent formula weight is defined by the joint probabilities of the phases contributing to the RHS of equation (5). If the individual weight of a phase is defined as

$$w(\mathbf{h}) = 1 / \sigma^2\phi(\mathbf{h}) \quad (6)$$

and the joint tangent weight as

$$w = \prod_n w_j(\mathbf{h}) / \sum_n w_j(\mathbf{h}) \quad (7)$$

The variance of an unrestricted phase ($0-2\pi$) is given by the following expression (Karle and Karle, 1966) where I_0 and I_k are modified Bessel functions dependent on $\alpha(\mathbf{h})$,

$$\sigma^2\phi(\mathbf{h}) = \pi^2/3 + 4/I_0 \sum_{\text{inf}} [(-1)^k I_k/k^2] \quad (8)$$

$\alpha(\mathbf{h})$ is a measure of agreement between contributing estimates of $\phi(\mathbf{h})$ within the tangent formula and is defined in the following equation where T and B are the tangent formula numerator (top) and denominator (bottom), respectively:

$$\alpha(\mathbf{h}) = (T^2 + B^2)^{1/2} \quad (9)$$

It follows that the variance of $\phi(h_j)$ may be calculated from equations (8) and (9). Computationally, however, this is prohibitive so in practice it is necessary to approximate $w(h)$. GENTAN provides for three different weighting schemes: probabilistic, Hull-Irwin statistical, and modified statistical.

4.27.4.1. w1: Probabilistic weights

For unrestricted phases the weight $w(h)$ based on equation (8) is approximately linear with respect to $\alpha(h)$ (see p92-94, Stewart and Hall, 1971). It is possible, therefore, to approximate $w(h)$ as $K\alpha(h)$ where K is set a fixed fraction. This approach has the disadvantage, however, of not providing weights normalized about 1.0. This normalization is important for the correct evaluation of $\alpha(h)$ from equation (9). In addition, for very large structures, all $\alpha(h)$ values will be small and weights based only on $\alpha(h)$ alone must also be small. The converse is true for small structures. For this reason, GENTAN uses a modified probabilistic weight which is essentially structure-independent and has values restrained to the range of 0 to 1.

4.27.4.1.1. Acentric structures

$$w_1(h) = \min \{ 1., \alpha(h)/\alpha' \} \quad (10)$$

where $\alpha' = \min (5., X)$ and X is mean A (triplet) or B (quartet).

4.27.4.1.2. Centric structures

The value of $w(h)$ calculated from equation (8) is not valid for restricted phases because it assumes a continuous phase distribution. For restricted phases the probability that $\cos \phi(h)$ is positive is given by the following expression:

$$P^+ \{ \cos \phi(h) \} = 1/2 + 1/2 \tanh \alpha (h)/2 \quad (11)$$

It follows that weight of an individual phase has the form

$$w(h) = \tanh \alpha(h)/2 \quad (12)$$

Although $w(h)$ calculated from equation (12) will lie in the range 0.0 to 1.0, it has the same deficiencies of unrestricted weights based solely on $\alpha(h)$. For this reason GENTAN uses the relative weight expression for restricted phases.

$$w_1(h) = \tanh \alpha(h) / \alpha' \quad (13)$$

where $\alpha' = \min(2., X)$ and $X = \text{mean of A (triplets) or B (quartets)}$

4.27.4.2. w2: Hull-Irwin Statistical Weights

The probabilistic weight w_1 is not suitable for application to some structural types. It tends to increase rapidly to 1.0 even for a small number of invariant relationships. As a consequence, it can be insensitive to significant variations in phase agreement. In addition, the w_1 weighting scheme has no provision for over-correlated phase sets which are characterized by unexpectedly high $\alpha(h)$ values.

To overcome these deficiencies, Hull and Irwin (1978) have suggested a weighting scheme based on the ratio of $\alpha(h)$ and the expectation value $\langle \alpha(h) \rangle$. It has the general functional form

$$w_2(h) = f \{ [\alpha(h) / \langle \alpha(h) \rangle]^2 \} \quad (14)$$

$$\langle \alpha(h) \rangle = \sum_m X_j P_j \quad (15)$$

where $P_j = I_1(X_j)/I_0(X_j)$ for *nonrestricted* ψ ; (16)

$P_j = \tanh(X_j)$ for *restricted* ψ (17)

The precise functional form of $f\{\}$ is given by Hull and Irwin.

The w_2 weighting scheme has several important properties. First, it depends on the individual expectation value $\langle\alpha(h)\rangle$ calculated from the actual number of invariants involved in the current estimate of $\phi(h)$. Secondly, the value of weight w_2 decreases if the phase agreement exceeds that expected for that stage of the refinement. In this way, it reduces the contribution of over-correlated phase estimates and enhances values that are close to that expected. Thirdly, w_2 takes into account the importance of phase correlation effects to weighting for different structure analyses (see the plot of w_2 above). At the same time the overall magnitude of w_2 is essentially independent of structure size, and the scaling procedures used in w_1 are not needed.

There is, however, a fundamental limitation to w_2 which is due to its implicit dependence on a reliable estimate of the α expectation value. For strongly non-random structures the estimates for $\langle\alpha(h)\rangle$, based on equations (16) and (17), may be inaccurate and this can lead to incorrect weights. In these cases the w_1 weights, which are based solely on phase agreement (i.e., $\alpha(h)$), will tend to be more reliable.

4.27.4.3. w_3 : Modified H-I Statistical Weights

This weighting scheme is identical to w_2 except that it is a function of x rather than x^2 . That is,

$$w_3(h) = f\{\alpha(h) / \langle\alpha(h)\rangle\} \quad (18)$$

w_3 is not as sensitive as w_2 to variations in the ratio of $\alpha(h)$ and $\langle\alpha(h)\rangle$. In some structures this is advantageous, particularly when invariant relationships are sparse and there is a tendency for phase oscillations. The weights due to w_3 are more heavily damped than w_2 but not as insensitive as w_1 to variations in phase agreement. w_3 may, for this reason, be considered a compromise between w_1 and w_2 .

4.27.5. Identifying the correct phase set

It is very desirable in multiresolution tangent methods to have some method of detecting "correct" phase sets prior to computing a Fourier transform (i.e., E-map). An a priori assessment of phase sets is made in GENTAN using two 'measure of success' parameters CFOM and AMOS. These parameters are based principally on the four figures-of-merit, RFOM, RFAC, PSI0, and NEGQ.

4.27.5.1. Relative Figure-of-Merit (RFOM)

This parameter is the inverse of the ABSFOM parameter of the MULTAN program (Main et al., 1980) and has the form

$$\text{RFOM} = [\sum_h \langle\alpha(h)\rangle - \sum_h \alpha_r] / [\sum_h \alpha(h) - \sum_h \alpha_r] \quad (19)$$

where all α 's are the mean values for the calculated α , the expected α (see equation (15)) and the random α (i.e., if all phases were randomly distributed). For a correct phase set the value of $\alpha(h)$ should approach that of $\langle\alpha(h)\rangle$ and RFOM should tend to 1.0. Incorrect phase sets will deviate significantly from 1.0, random phases towards 2.0, and overcorrelated phases towards 0.0. In general, however, phase sets with small RFOMs are more likely to be correct than those with larger RFOMs. The actual range of RFOMs for a given GENTAN run will vary according to the validity of the estimate of $\langle\alpha(h)\rangle$. For this reason RFOM tends to be less reliable for strongly non-random structures.

4.27.5.2. R-factor Figure-of-Merit (RFAC)

The RFAC parameter is similar to the residual FOM calculated in MULTAN (Main et al., 1980) except for a scale that takes into account the relative dominance of heavy atoms in the structure.

$$\text{RFAC} = \sum_h |\alpha(h) - \langle\alpha(h)\rangle| / \sum_h \langle\alpha(h)\rangle \quad (20)$$

RFAC is a minimum when there is close correspondence between the refined $\alpha(h)$ and the expected $\langle\alpha(h)\rangle$. In this respect it is very similar to the R-factor of Karle and Karle (1966). RFAC is, like RFOM, dependent on the reliable estimate of $\langle\alpha(h)\rangle$.

4.27.5.3. Psi(zero) Figure-of-Merit (PSI0)

The $\psi(0)$ triplet invariants of Cochran and Douglas (1957) provide a sensitive figure-of-merit which is largely independent of the triplet and quartet invariants used in the tangent refinement. A $\psi(0)$ triplet relates two generator reflections (with $|E| > E_{MIN}$) to a third which is selected to have an $|E|$ -value as close as possible to zero (see the *GENSIN* documentation). The phases estimated from a series of $\psi(0)$ triplets are expected to be random when the contributing phases from the other two large- $|E|$ reflections are correct. When this is the case the resulting values of α^2 are significantly lower than if the distribution of contributing phases was biased or incorrect. These invariants are used to form the figure-of-merit

$$PSI0 = \sum_k \alpha(k) / \sum_k \langle \alpha(k) \rangle \text{ for } \psi(0) \text{ triplets (21)}$$

PSI0 should be smallest for the correct phase sets. PSI0 is, along with NEGQ, one of the most sensitive and independent methods of measuring the relative likelihood of success.

4.27.5.4. Negative Quartet Figure-of-Merit (NEGQ)

Quartet structure invariant relationships are classified according to the magnitude of their crossvector $|E|$ values. When the crossvector sum is very low there is a high probability that the invariant phase ψ will tend to have a value of π rather than 0. These invariants are referred to as negative quartets. In GENTAN negative quartets are usually not used applied to the tangent refinement process but are retained as a test of the phase sets (unless the `active` code is entered in the `invar` line). The negative quartets are considered independent because, unlike the positive quartets, they cannot be represented by a series of triplet invariants. The negative quartets provide, therefore, a separate estimate of the phases. A direct comparison of these phases provides the basis for the figure-of-merit,

$$NEGQ = \sum |\phi(h) - \Phi(h)| / n \text{ for } n \text{ negative quartets (22)}$$

where $\phi(h)$ is the tangent refined phase, $\Phi(h)$ is the phase estimated from negative quartets alone. Correct phase sets should have low values of NEGQ ranging from 0° for centrosymmetric structures, to 20-60° for noncentrosymmetric structures. Note that if fragment QPSI values are used, the value of ϕ is automatically set to 0° and the NEGQ test will remain valid. This FOM is a very powerful discriminator of phase sets provided that sufficient negative quartets are available.

4.27.5.5. Combined Figure-of-Merit (CFOM)

The combined FOM is a scaled sum of the four FOM parameters RFOM, RFAC, PSI0 and NEGQ.

$$CFOM = \sum [WFOM_i (FOMMAX_i - FOM_i) / (FOMMAX_i - FOMMIN_i)] \text{ } i=1,4 \text{ (23)}$$

The FOM weights $WFOM_i$ may be specified on the `setfom` control line. These values are subsequently scaled so that the maximum value of CFOM is 1.0. It is important to stress that CFOM is a relative parameter and serves mainly to highlight which is the best combination of FOMs for a given run. It does not indicate whether these FOMs will provide a solution.

4.27.5.6. Absolute Measure-of-Success Parameter (AMOS)

The AMOS parameter is a structure-independent gauge of the correctness of a phase set. It uses pre-defined estimates of the optimal values for the FOM parameters RFOM, RFAC, PSI0 and NEGQ. OPTFOM values may be user defined (see `setfom` line). Rejection values for the four FOM parameters are derived from the OPTFOM values as $REJFOM=2*OPTFOM$. The default values are as follows,

	<i>RFOM</i>	<i>RFAC</i>	<i>PSI0</i>	<i>NEGQ</i>
OPTFOM	1.0	0.25	.75	60.
REJFOM	2.0	0.5	1.5	120.

The absolute measure-of-success parameter is calculated from all active FOMs as

$$\text{AMOS} = \sum [\text{WFOM}_i (\text{REJFOM}_i - \text{FOM}_i) / \text{OPTFOM}_i] \quad i=1,4 \quad (24)$$

where the WFOM values are scaled so that AMOS ranges from 0 to 100. In addition to being used to sort phase sets in order of correctness, the AMOS values provide a realistic gauge of the correctness of phase sets. As a rule of thumb, AMOS values can be interpreted in the following way:

AMOS

100-81%	<i>high probability of being correct set</i>
80-61%	<i>good chance of being correct set</i>
60-41%	<i>possibility of being correct set</i>
40-21%	<i>low probability of being correct set</i>
20-0%	<i>unlikely to be correct set</i>

These classifications are only approximations. The predictability of optimal FOM values can be perturbed by a variety of structure dependent factors and by the FOM weighting. Nevertheless, the AMOS value provides the user with a concise overview of the phase sets.

4.27.6. Rejection Of Phase Sets

Phase sets must satisfy certain criteria before being considered for possible output to the bdf. Each phase set is tested at three stages in the tangent process and is rejected if the FOM values and other parameters are outside acceptable limits. In this way time is not spent on phase sets that have little or no chance of being correct - a very desirable feature for a multisolution procedure.

The first rejection test is made following the sixth tangent iteration and involves only the top block of sorted phase estimates. This is referred to as the PRETEST of FOMs and the rejection criteria 011, 012, and 014 are applied. The user may disable the PRETEST of FOMs with the `setfom` line (field 1). The second rejection test is made after the last tangent iteration and a phase set is rejected according to the criteria 021, 022, 023, 024, and 025. The final rejection test occurs during the sorting of phase sets and the calculation of the AMOS value. Phase sets are rejected if criteria 036 and 037 are not satisfied.

PRETEST Rejection Criteria

011	reject if $\text{RFOM} > 2 * \text{REJFOM}(1)$
012	reject if $\text{RFAC} > \text{REJFOM}(2)$ and $\text{RFOM} > \text{REJFOM}(1)$
014	reject if $\text{RFOM} < 0.25$

Last Iteration Rejection Criteria

021	reject if $\text{RFOM} > \text{REJFOM}(1)$
022	reject if $\text{RFAC} > \text{REJFOM}(2)$
023	reject if $\text{PSI0} > \text{REJFOM}(3)$
024	reject if $\text{RFOM} < 0.25$
025	reject if convergence limit not reached after maximum refinement iterations (see <code>refine</code> line)

Final Rejection Criteria

036 _a	reject if $ \text{av.}\phi - \langle \text{av.}\phi \rangle $ exceeds 45°
037	reject if $\text{AMOS} < 5\%$

Notes:

- The value of $\text{av.}\phi$ is 90° for centrosymmetric structures and $150-180^\circ$ for noncentrosymmetric structures. This test rejects "all-plus catastrophe" phase sets.

4.27.7. Automatic Termination

Tests for phase correctness are made at the same time as the second rejection test. These are made on the basis of the optimum FOM values, OPTFOM. If all tests are satisfied, the tangent phasing process is terminated and the program enters the sort mode. In this way GENTAN insures that computing time is not wasted on generating further phase sets when a correct set of phases has already been calculated. This is particularly important when the `random` start option is invoked. Tangent cycling is terminated

```
if  RFOM  <  OPTFOM(1)
and  RFAC  <  OPTFOM(2)
and  PSIO  <  OPTFOM(3)
and  NEGQ  <  OPTFOM(4)
and  av.  $\phi$   >  45°
```

If inadvertant termination occurs, the user can adjust the OPTFOM values used in the above criteria, or switch off this test entirely (field 10, `setfom` line).

4.27.8. Application Of Partial Structure Data

4.27.8.1. Psi Calculated from Fragment Information

If structure information of types 3 and 4 (oriented and positioned) is entered into the GENEV calculation, and the `qpsi` option is invoked in GENSIN, then the phase estimate of the invariant may be used in the extension and refinement process. This can provide a significant improvement to the phasing process, particularly for planar and heavy-atom structures, and provides a valuable second line of attack for less tractable problems. The rule of thumb is: "if oriented or positional structural information is available, use it!"

4.27.8.2. Applying Structure Factor Phases as Starting Phases

An alternative approach to partial structure information is to use calculated structure factor phases as input starting phases (Karle, 1976). This approach has the advantage over the group structure factor method of not requiring the repeat of *GENEV* and *GENSIN* calculations. In practice this method is limited because it fails to take into account the expected change to the ψ values which is available from knowledge of the structure. Because of this, strongly non-random structural features tend to dominate the phases. Nevertheless, careful application of the parameters on the `partsf` line can provide an important alternative to applying known structural information.

4.27.9. File Assignments

- Reads $|E|$ values from input archive `bdf`
- Writes estimated phases to the output archive `bdf`
- Reads structure invariants from file `inv`

4.27.10. Examples

GENTAN

This run automatically selects starting phases. Phases are extended and refined in the `block` mode using weight scheme 2. The top four phase sets are written to the output bdf. All FOM rejection and cycle termination tests will be applied.

```
GENTAN pout 10           :output the top 10 phase sets
invar trip              :use triplet invariants only
select magic            :use magic integer permuted phases
assign odr 7 23 5 per 1 17 :assign ODR/permute phases
refine block w3         :block mode with mod H-I weights
setfom nopr             :do not pretest FOM values
```

```
GENTAN pset 128         :permit 128 phase sets
invar *7 allxv          :include all quartets
refine cascade w2 30    :max iterations to 30
phi 5 2 -3 0. odr       :define ODR
phi 1 5 7 0. odr        :define ODR
phi 3 2 2 0. odr        :define ODR
select random           :select all phases with random values
archiv -1601 -1603     :delete items 1601 and 1603 from bdf
```

4.27.11. References

- Ahmed, F.R. and Hall, S.R. 1976. *Computer Application of the Symbolic Addition and Tangent Procedure*. Crystallographic Computing Techniques, Eds. F.R. Ahmed, K. Huml, B. Sedlacek Munksgaard: Copenhagen, 71-84.
- Cochran, W. and Douglas, A.S. 1955. *The Use of a High Speed Digital Computer for the Direct Determination of Crystal Structures I*. Proc. Roy. Soc., A277, 486-500.
- Hall, S.R. 1978. Paper 15.1-2, *Collected Abstracts. 11th IUCr Congress*, Warsaw.
- Hull, S.E. and Irwin, M.J. 1978. *On the Application of Phase Relationships to Complex Structures. XIV. The Additional Use of Statistical Information in Tangent-Formula Refinement*. Acta Cryst., **A34**, 863-870.
- Karle, J. 1976. *Structures and Use of the Tangent Formula and Translation Functions*. Crystallographic Computing Techniques. Eds. F.R. Ahmed, K. Huml, B. Sedlacek. Munksgaard: Copenhagen, 155-164.
- Karle, J. and Hauptman, H. 1956. *Theory of Phase Determination for the Four Types of Non-Centrosymmetric Space Groups 1P222, 2P222, 3P12, 3P22*. Acta Cryst., **9**, 635.
- Karle, J. and Karle, I.L. 1966. *The Symbolic Addition Procedure for Phase Determination for Centrosymmetric and Noncentrosymmetric Crystals*. Acta Cryst., **21**, 849.
- Main, P., Fiske, S.J., Hull, S.E., Lessinger, L., Germain, G., Declercq, J.P. and Woolfson, M.M. 1980. *MULTAN-80 Program Writeup*, Dept. of Physics, University of York, York, England.
- Stewart, R.F. and Hall, S.R. 1971. *X-ray Diffraction: Determination of Organic Structures by Physical Methods*. Eds. F.C. Nachod and J.J. Zuckerman. Academic Press: New York, 74-132.
- White, P.S. and Woolfson, M.M. 1975. *The Application of Phase Relationships to Complex Structures VII. Magic Integers*, Acta Cryst., **A31**, 53-56.

- Woolfson, M.M. 1976. *Doing Without Symbols - MULTAN*. Crystallographic Computing Techniques. Eds. F.R. Ahmed, K. Huml, B. Sedlacek. Munksgaard: Copenhagen, 85-96.
- Yao Jai-xing. 1981. *On the Application of Phase Relationships to Complex Structures XVIII. RANTAN - Random MULTAN*, Acta Cryst., **A37**, 642-644.

4.28. **GIP: Reciprocal space PIG**

Author: Doug du Boulay, Crystallography Centre, University of Western Australia, Nedlands WA 6907, Australia

GIP represents all reflection data on the bdf in terms of the scattering sphere. The scattering vector of each unique measured reflexion is represented as points in reciprocal space and the color and size of those points can be used to represent many reflection properties stored on the XTAL lrrefl: archive record. This representation and the desired quantities can be rotated viewed and adjusted interactively. Duplicated reflexions are ignored.

4.28.1. Overview

GIP is essentially a data viewing or educational tool. It reads an archive bdf and optionally renders a series of screen dumps to a **.gip** file for label manipulation or plotting with PREVUE/PLOTX. It was originally conceived as means of identifying systematic trends in measured data, via plotting of derived quantities such as $(I_{rel} - \langle I_{equiv} \rangle) / \sigma$ i.e. systematic deviations amongst groups of equivalents in selective regions of reciprocal space. Reflection color and size can be used to show two properties simultaneously.

4.28.2. General functionality

GIP has limited command line controls, but numerous interactive controls in the graphical interface. Double buffering of the graphics display window can be invoked with the **GIP dbuf** control line, though the copying of back buffer to display typically causes significant slowing of the rendering, particularly for large windows.

The scattering sphere may be rotated by left mouse button click-drag events and magnified or reduced by up/down middle mouse button events, (typically emulated on two button mice by pressing left and right simultaneously)

An animation button allows the sphere to be set spinning by rotation button drags, with angular velocity dictated by the length of the drag from mouse click until mouse release.

4.28.3. General control button function

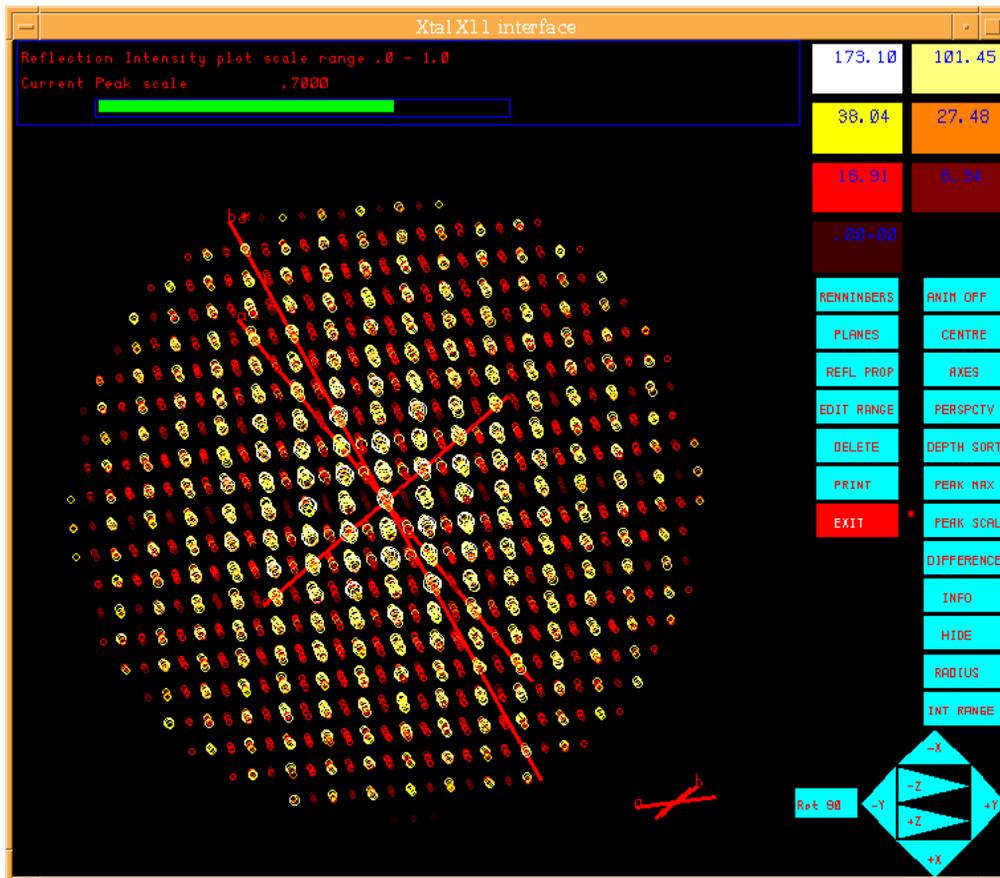
4.28.3.1. Anim On/Off

Switches free rotating animation on or off. If animation is on, a left-mouse-click-drag event invokes the animation. A single left-mouse-click-release with no mouse movement stops the animation.

4.28.3.2. CENTRE

Translations of the sphere centre away from the rotation centre are currently disabled, so the centre control essentially has no effect at all.

Figure 4-1. Screen shot of GIP



4.28.3.3. AXES

Reciprocal lattice axes are currently drawn through the centre of the diffraction sphere. Occasionally this can obscure reflections in axial planes. Selecting the **AXES** control moves the axes to a more extraneous position.

4.28.3.4. PERSPECTIVE

Switching on perspective can highlight symmetry axes of reciprocal space. Perspective works in two ways, to reduce the size of reflection markers due to viewing distance, and to cause apparent convergence of lattice points with viewing distance.

4.28.3.5. DEPTH SORT

Two modes of reciprocal space viewing are possible. The default method ignores viewing distance of the reciprocal lattice vectors and simply plots all reflections in planes from Hmin up to Hmax. The second method attempts to draw the sphere from furthest reflection, to closest by partitioning reciprocal space into thin slices according to depth and plotting all points which fall within the range of each slice. This method is significantly slower and occasionally misses some reflections, while drawing others more than once. Not recommended for general use unless you want to print the current view, in which case depth sorting is an advantage.

4.28.3.6. PEAK MAX

This command activates a slider control in the message window enabling the global reflection scale factor to be adjusted from 0.0 to 0.5, which scales the size of all reflection markers up or down by the same amount. Use the right mouse button to adjust the value of the slider. Reselect **PEAK MAX** to cancel this mode.

4.28.3.7. PEAK SCAL

The **PEAK SCAL** command activates a slider control in the message window scaling the relative difference in size of strong and weak reflections. A value of 1.0 gives the largest differences and anything less than 0.5 is indistinguishable from "no difference at all". Use the right mouse button to adjust the value of the slider. Reselect **PEAK SCAL** to cancel this mode.

4.28.3.8. DIFFERENCE

Depending on the contents of the archive `lrrefl`: reflection record and the stage of data processing at which GIP is invoked, up to four options maybe available when selecting **DIFFERENCE** mode, - namely:

- $(I_{rel} - \langle I_{equiv} \rangle) / \sigma$
- $(F_{rel} - \langle F_{equiv} \rangle) / \sigma$
- $(F_{rel} - F_{cal})$
- $(F_{rel} - F_{cal}) / \sigma$

If there is only one option available owing to `bdf` contents, it will be invoked immediately without going through the intermediate menu selection.

4.28.3.9. INFO

In **INFO** mode, the right mouse button can be used by click-drag events to view archive information associated with a given reflection in reciprocal space. Reselect **INFO** to cancel this mode. Note that **INFO** has not been configured to operate correctly when **PERSPECTIVE** is on.

4.28.3.10. HIDE

The **HIDE** control can be used to hide or redisplay all reflections belonging to a given colour class. Select **HIDE** and then select the color range (from the top-right color range boxes) to be hidden or redisplayed. Hiding all **RCODE 1** reflections for instance can highlight systematically absent **rcode 2** reflections, which may be useful in identifying space group symmetry.

4.28.3.11. RADIUS

This command activates a slider control in the message window enabling the minimum and maximum $\sin \theta / \lambda$ values to be adjusted to highlight a particular shell of reciprocal space. Reselect the **RADIUS** button to cancel this mode.

4.28.3.12. INT RANGE

This command activates a slider control in the message window enabling adjustment of the minimum and maximum intensity range to be displayed. Reselect the **INT RANGE** control to cancel this mode.

4.28.3.13. EWALD

This command is analogous to **INFO** mode in that it begins by permitting the user to identify a reflection using right-mouse-click-drag events. Having identified a suitable reflection, the Ewald multiple scattering sphere for that reflection at the diffractometer measurement angle is illustrated by reselecting the **EWALD** control. The veracity of this representation relies on the presence of the crystal based azimuthal ψ angle to abstract away from crystal orientation and diffractometer specifics. Reflections lying on or near the surface of this sphere can be identified by invoking **INFO** mode using right-click-drag events to identify reflections. Alternatively,

middle-mouse-click-drag events can be used to rotate the Ewald sphere about the scattering vector for the selected reflection. Ewald mode can be cancelled by reselecting the **EWALD** control

4.28.3.14. PLANES

Selecting the **PLANES** control button offers up a selection of six different modes for identifying specific reciprocal lattice planes for display.

- **0k1** request specific kl plane for given h
- **h01** request specific hl plane for given k
- **hk0** request specific hk plane for given l
- **GENRL** specify a general plane through hkl and normal to the hkl scattering vector.
- **CENTR** specify a general plane through the origin and normal to the hkl scattering vector.
- **3PSEL** identify three reflections by right-mouse select. Reselect **PLANES** and the specified plane will be singled out for display.

PLANES mode can be switched off by reselecting the **PLANES** control button.

4.28.3.15. REFL PROP

Selecting this control button reveals a selection of controls in the message panel, for choosing between the different **lrrefl: bdf** items which are supported for display as colour or size properties. Choose one of the options or cancel the change by reselecting the **REFL PROP** control.

4.28.3.16. EDIT RANGE

This control displays a multiway slider bar in the message window for enabling relatively coarse grained adjustments to the colour ranges. Colour range changes do not take effect immediately, but only after the **EDIT RANGE** control has been reselected to exit edit mode.

4.28.3.17. DELETE

This control has no functionality.

4.28.3.18. PRINT

The **PRINT** control adds a dump of the current screen view to the **.gip** file. Multiple prints can be invoked per GIP session and they accumulate as multiple frames for display, editing and printing using **PLOTX** and **PREVUE**.

4.28.3.19. EXIT

The function of this control is rather obvious. Given the non-critical nature of this program, there is no safety net for accidental exit.

4.28.4. Known shortcomings

- Problems identifying reflections in perspective mode
- The Ewald sphere is missing a tolerance parameter to specify distance from the sphere surface

4.28.5. References

- Schwarzenbach, D. & Flack, H. D. (1989). *J. Appl. Cryst.* **22**, 601-605.
- Busing, W.R. and Levy, H.A. (1967). *Angle Calculations for 3- and 4- Circle X-ray and Neutron Diffractometers.* *Acta Cryst.* **22**, 457-464.
- Tanaka, K. & Saito, Y. (1975). *Acta Cryst.* **A31**, 841-845.

4.29. **LATCON**: Refine lattice parameters

Authors: Dieter Schwarzenbach and Geoff King

Contact: Geoff King, Fysica-Chemische Geologie, K.U. Leuven, B-3001 Heverlee, Belgium

LATCON is a rewrite of Dieter Schwarzenbach's program of the same name in the XRAY76 system. It determines the 'best' lattice parameters and their standard deviations from the 2θ diffraction angles of a set of indexed reflections, taking into account any constraints imposed by the crystal system and allowing for systematic errors in the zero values.

Given a set of indexed reflections with their diffraction angles 2θ , the program uses the least-squares method to determine the best values of the reciprocal and direct metric tensors and lattice parameters with their estimated standard deviations (esd's) together with the variance-covariance and correlation matrices. The 2θ values may have been obtained by powder or single-crystal measurements. Different schemes are available for allowing for systematic errors.

In principle there are two methods of refining lattice constants by minimizing

$$R = \sum w (2\theta_{\text{obs}} - 2\theta_{\text{calc}})^2$$

The first is to express 2θ as a non-linear function of the lattice constants. This leads to a non-linear least-squares procedure which requires starting values and several cycles of refinement. The second, used here, takes advantage of the fact that $\sin^2 \theta$ is a linear function of the reciprocal metric tensor giving a linear least-squares problem which can be solved directly without the need for starting values and without iteration. The much more complicated task consists then in computing the esd's of the best lattice constants from those of the reciprocal metric tensor elements.

The following concepts are useful:

Metric tensor (M)	Reciprocal metric tensor (M*)	Vector (v*)
a ₂ a.*b a.*c	a* ₂ a*.*b* a*.*c*	h
a.b b ₂ b.c	a*.*b* b ₂ b*.*c*	k
a.c b.c c ₂	a*.*c* b*.*c* c ₂	l

All symbols containing a "*" refer to reciprocal space. The matrix \mathbf{M}^* is the inverse of the matrix \mathbf{M} . The vector $\mathbf{v}^{*\dagger}$ is the row vector (h k l), i.e. \mathbf{v}^* transposed. Therefore,

$$\sin^2 \theta = (\lambda^2/4) (\mathbf{v}^{*\dagger} \mathbf{M}^* \mathbf{v}^*)$$

The variance-covariance matrix **COV** (see e.g. ref. 1) is calculated from the inverse \mathbf{N}^{-1} of the normal-equation matrix \mathbf{N} .

$$\text{COV} = e\mathbf{N}^{-1}, e = R/(n-k)$$

With R defined above, n = number of observations, k = number of parameters refined, $n-k$ is the number of degrees of freedom and e the variance of an observation with unit weight, so \sqrt{e} , the error of fit, is the corresponding σ . If the weights are the reciprocals of real variances, the expectation value for the error of fit is unity and R should be distributed as χ^2 . The diagonal terms of **COV** are the variances whose square roots are the standard deviations (esd's), σ . The ij th term of the correlation matrix **COR** is given by

$$\text{COR}_{ij} = \text{COV}_{ij} / (\sigma_i \sigma_j)$$

The law of propagation of errors expresses the variance-covariance matrix **COVB** of derived quantities $b_1 = \text{function1}(a_1, a_2, \dots)$, $b_2 = \text{function2}(a_1, a_2, \dots)$, etc in terms of **COVA** of a_1, a_2, \dots . Denoting the derivative of b_k with respect to a_l by $\partial b_k / \partial a_l$ and the matrix of derivatives as \mathbf{D} , we obtain

$$\text{COVB}_{ij} = \sum_{k,l} \partial b_i / \partial a_k \partial b_j / \partial a_l \text{COVA}_{kl}$$

$$\text{COVB} = \mathbf{D} \text{COVA} \mathbf{D}^\dagger$$

The program proceeds by the following steps:

1. The esd's (i.e. σ 's) of the 2θ values are used to compute the σ 's of $\sin^2 \theta$

$$\text{i.e. } Q = 1/d^2 = 4 \sin^2 \theta / \lambda^2$$

where d is the interplanar spacing. According to the error propagation law,

$$\sigma(Q) = (2/\lambda^2) \sin 2\theta \sigma(2\theta) \text{ The weight is } 1/\sigma^2.$$

2. With these weights the linear least squares problem is set up, the symmetry constraints are introduced and the tensor \mathbf{M}^* is determined together with **COV** and **COR**. These matrices are complete and include the elements which are determined by the crystal symmetry and which must not be refined. Thus the variances, covariances and correlations of invariant terms are zero. For a dependent term $s = k \times t$ we obtain

$$\text{variance}(s) = k^2 \text{variance}(t); \sigma(s) = k \sigma(t)$$

$$\text{covariance}(s,t) = k \text{variance}(t); \text{correlation}(s,t) = 1$$

3. The reciprocal lattice constants a^* , b^* , c^* , $\cos \alpha^*$, $\cos \beta^*$ and $\cos \gamma^*$ are determined together with their **COV** and **COR** matrices, using the error propagation law. The σ 's of these quantities are again the square roots of the diagonal terms of **COR**; the σ of the angles in degrees are derived from those of their cosines.
4. The matrix of the reciprocal metric tensor \mathbf{M}^* is inverted to give \mathbf{M} , again with the appropriate **COV** and **COR** matrices. This requires the evaluation of all 36 derivatives of the matrix elements with respect to the reciprocal matrix elements e.g.

$$\partial a^* b / \partial a^{*2}, \partial b^2 / \partial c^{*2} \text{ etc.}$$

5. The direct lattice constants are obtained as in step 3. If all reflections have been measured with the same wavelength, the **COR** matrix and therefore the σ 's may be corrected for the uncertainty of the wavelength,

$\sigma(\lambda)$, which is especially interesting in neutron diffraction.- This must not, of course, involve the least-squares weighting. Since the relative error in Q due to an error in the wavelength is

$$\sigma(Q)/Q = 2\sigma(\lambda)/\lambda$$

The variances of \mathbf{M}^* after refinement are increased by $\sigma^2(Q)$ x tensor element. The covariances are derived so as to leave the correlation coefficients unchanged.

4.29.1. Systematic Errors

Two mutually exclusive schemes for taking account of systematic errors may be used in the refinement. Both involve refinement of an additional parameter D_k .

1. This scheme is included when the mnemonic **rq** is used in the **LATCON** line. It has been shown (2) that errors in the measured d-values (Δd) caused by absorption and spherical eccentricity of the specimen are approximately proportional to

$$\Delta d = k d \cos^2\theta = - 2D_k d \cos^2\theta$$

where $d=1/\sin\theta$. Thus ΔQ is equal to the derivative of Q with respect to d times Δd .

$$\Delta Q = D_k (4/\lambda^2) \sin^2 2\theta$$

D_k is refined as an additional parameter. The computed Q and 2θ values listed in the output are no longer the TRUE values but they include the correction term. This correction can be used if the position of the primary beam is well-known, i.e. if 2θ has been determined from corresponding reflections at plus and minus 2θ or with respect to an internal standard. Note that many high orders must be included in the refinement since the error approximation is valid for high scattering angles and corresponds to an extrapolation of d to $\cos^2\theta$ zero. Furthermore, the relative error $\Delta Q/Q$ in Q is proportional to $\cos^2\theta$ which has non-negligible values at quite high angles. Limiting measurements to low orders leads to very high correlations between D_k and the metric tensor elements so that the correction may be meaningless.

2. Sometimes the position of the primary beam may not be accurately known, as in the case when the diffractometer does not allow measurements at negative 2θ values,. Here we must set

$$2\theta_{\text{true}} = 2\theta_{\text{obs}} - 2D_k$$

where $2D_k$ may be called the true zero value of 2θ . The refinement of D_k results in a non-linear least-squares procedure for this parameter and several cycles may be necessary to achieve convergence. Experience shows, however, that convergence is quite fast. The maximum number of cycles must be input after the mnemonic **rz** but a shift of less than 0.002° halts the refinement. An initial value of 2θ zero may also be given (and will be held constant unless **rz** refinement is invoked). Linearisation of the expression for Q leads to

$$\Delta Q = D_k (4/\lambda^2) \sin 2\theta$$

$$Q_{\text{true}} = \mathbf{V}^{*\dagger} \mathbf{M}^* \mathbf{V}^* + D_k (4/\lambda^2) \sin 2\theta$$

and D_k is refined as an additional parameter. The true Q and 2θ values are listed in the output. Inclusion of many high-order reflections is desirable but the correlation of D_k with the metric tensor is less severe than in scheme 1 since $\Delta Q/Q$ is proportional to $\cot\theta$, which decreases faster than $\cos^2\theta$ and has large values at low orders.

LATCON should give exactly the same results as a non-linear least-squares procedure. The errors in the variances and covariances derived from the error propagation law, which is due to the series expansion involved, are very small for σ 's of a few percent or less.

LATCON can also use the angle data and the crystal system stored on the archive bdf. The option line **skip** enables reflections to be omitted. Options also exist for outputting the refined cell dimensions.

4.29.2. File Assignments

- Writes CELL and CELLSD lines to the line file pch

4.29.3. References

- D. Schwarzenbach et al. 1989. *Statistical Descriptors in Crystallography* (Report of the IUCr subcommittee). Acta Cryst., **A45**, 63-75.
- L.V. Azaroff and M.J. Buerger. 1958. *The Powder Method in X-ray Crystallography* (pp. 239-244). McGraw Hill, New York.

4.30. *LISTFC*: List formatted reflection data

Author: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

LISTFC lists reflection data from the bdf in packed or unpacked format.

4.30.1. Description

LISTFC formats reflection data from the bdf in packed (only the fastest-varying Miller index is listed **ind 1**) or unpacked (all indices are listed **ind 3**) formats. LISTFC does not sort the reflection data so that the listed order is determined by the order of the data on the bdf. The sort order must be entered (**ord hkl**) if the program has not been run. *SORTRF* must be run again if a different list order is required.

In packed mode (the default **ind 1**) the first item listed is the fastest-varying hkl index. Reflections are listed in blocks with medium-varying and slow-varying indices constant. Each block is separated by a header line describing these indices.

The unpacked mode (**ind 3**) is particularly useful for listing data which is not sorted or contains Friedel indices. It may also be used list the reflection data for input to other applications. The repetition of reflection columns on a "page" is controlled with the **wid** and **lin** options. If only one reflection is needed per line the **wid** should be set to a value just greater than the columns needed for data in one line. If no headers are needed then **lin** should be set to a value greater than the total number of reflections. The reflection data may also be output to the punch line file using the **pun** option.

4.30.2. Control Lines

4.30.2.1. output

The **output** line is used to specify which items are to be listed, and in what order. Commonly listed items may be specified using the alphabetic codes **hk1**, **fre1**, **fca1**, **sigf** and **phi**. Alternatively, items may be specified using the directory ID numbers listed in the BDF section at the back of this manual. Use of the ID numbers enables the user to list any item present in *lrrefl*: of the bdf. *The only restriction to specifying items is that hk1 must be first.* The number of items specified may not exceed ten (10). Note that the reflection code (i.e., RCODE) may be output as a flag via a control parameter on the **LISTFC** line (field 4). If the RCODE flag is requested, then this is considered by the program as an additional item. The flag symbols may be changed by use of field 8 of the **LISTFC** line. When the **output** line is not entered the default items are: **hk1 fre1 fca1 sigf**.

4.30.2.2. format

The **format** line is used to specify the dimension and the nature (real or integer) of the numbers listed. The order of the input formats is assumed to be identical to the items on the **output** line. Note that only one format code is input for the Miller indices whether it is packed mode or not.

Two types of formats are permitted: the integer format of the general form 'Inn' and the floating-point format of the format 'Fnn.m'. As with normal usage, the 'nn' integer specifies the maximum number of digits permissible for the output number, and 'm' is the number of digits to the right of the decimal point. 'nn' also specifies the number of characters reserved for an item and therefore the relative spacing of the numbers.

When the **format** line is not entered, the default formats to match the default **output** line are: I5 I5 I5 I3.

4.30.2.3. pscale

The **pscale** line is used to scale items before printing. The scales entered on this line are in the identical order to the **output** line. These scales are particularly useful when 'I' formats are used to make the listing as compact as possible. For example, an item such as **fre1** with the value 'xy.z' can be scaled up by 10.0 and listed as the integer 'xyz', thus saving one digit for each occurrence on the line. These scales may also be used to convert an item stored on the bdf into a value suitable for publication. For instance, the structure factor phase is stored on the bdf as cycles (1 cycle = two pi). The user may wish to list the phase in millicycles by specifying a scale of 1000., or in degrees by specifying a scale 360.

If the **pscale** line is not entered, the default scales are set to comply with default **output** line at: 1. 10. 10. 10.

4.30.2.4. scale

The **scale** line specifies the structure factor scale(s) which are applied to the **fre1** and **sigf** items. This line need only be entered if the scale(s) stored on the bdf are incorrect.

4.30.2.5. maxhk1

The **maxhk1** line enables the user to list only that data on the bdf that has been used in the refinement process. If **maxhk1** is not entered all reflections will be output.

4.30.3. File Assignments

- Read reflection data from the input archive bdf

4.30.4. Examples

LISTFC

Standard default options will provide a packed listing of hkl frel fcal sigf(without rcode flags). The width of printed reflection is 18 characters, so that 6 reflections will be listed on each line of 120 characters. The RCODE for each reflection will be translated into a print character rcode=1 a blank; rcode=2 a '<' character; rcode=3 an 'e'.

```
LISTFC ind 3 wid 40 lin 5000 pun
output hkl frel sigf 1600 1631      :list |E| and est. phase
format i5 f6.1 f5.1 f5.1 i5        :36 chr wide; 1 refl/line
pscale 1. 1. 1. 1. 360.            :convert phase to degrees
maxhkl *5 0.4                       :list only s >0.4
```

```
LISTFC ind 1
output hkl frel 1314 sigf 1315      :list |F| Friedel pairs
format i5 f6.1 f6.1 f5.1 f5.1
```

4.31. *LSABS*: Absorption corrections

Authors: Eric Blanc & Dieter Schwarzenbach

Contact: Eric Blanc, Institut de Cristallographie, University of Lausanne, 1015 Lausanne, Switzerland

LSABS calculates transmission factors T , absorption-weighted mean-path lengths T , and derivatives, $\partial T/\partial d_m$

4.31.1. Overview

The calculation may be carried out either with an analytical method similar to the one of de Meulenaer & Tompa (1965) or by Gaussian grid integration (Busing & Levy, 1957). A major innovation in this program is the calculation of the derivatives of transmission with respect to crystal shape $\partial T/\partial d_m$, where d_m defines the distance of the m^{th} crystal face from an origin inside the crystal. These derivatives may be used during least-squares refinement to obtain improved estimates of the crystal shape and consequently an improved absorption correction. The program uses crystal-based azimuthal angles (Schwarzenbach & Flack, 1989, 1992) to define the diffraction geometry and is thus independent of the particular instrument (diffractometer, camera, *etc.*) used to collect the intensity data.

4.31.2. Methods of Absorption Correction

LSABS offers two methods of integration, the analytical method and Gaussian integration. The analytical method is more efficient than the Gaussian method in calculating the derivatives. For this reason, execution times for the two methods are comparable in the case of a coarse Gaussian grid, and in favour of the analytical method in the case of a fine Gaussian grid.

4.31.3. Analytical method

The analytical method proposed by de Meulenaer & Tompa (1965) describes the crystal shape by the set of its vertices, whereas **LSABS** defines it by the set of its edges. This technique facilitates an unambiguous identification of the vertices and minimizes rounding errors.

4.31.4. Gaussian grid integration

The implementation of the numerical integration of Gauss-Legendre offers a choice of grids with 6, 8, 10, 12, 14, 16, 24, or 32 intervals for each dimension. For needle- and plate- shaped crystals the orientation of the grid may be selected to lie along one of the principal directions. For a needle, the z-axis of the grid is chosen along the needle axis identified by its zone indices $[u\ v\ w]$. For a plate, the z-axis of the grid is chosen along the plate normal identified by the face indices $(h\ k\ l)$. The calculation of the derivatives $\partial T/\partial d_m$ requires evaluation of surface integrals. Consequently a surface grid must be specified in addition to the 3-dimensional grid for the volume integral.

4.31.5. The Linear Absorption Coefficient μ

The linear absorption coefficient μ is calculated by **LSABS** using the chemical formula stored on the bdf and the X-ray cross sections stored in the program. For the moment, the available X-ray cross sections are those for $\text{CuK}\alpha$, $\text{MoK}\alpha$ or $\text{AgK}\alpha$ radiation taken from Table 2.1B of International Tables for X-ray Crystallography Vol IV (Ibers & Hamilton, 1974) for the elements from hydrogen to plutonium. For other wavelengths and elements the value of μ may be entered on the **LSABS** line.

4.31.6. Input and Output Data Specification

The M plane faces of a polyhedral crystal are defined by their Miller indices $(h_m\ k_m\ l_m)$ relative to the lattice base \mathbf{a} , \mathbf{b} , \mathbf{c} , and by the perpendicular distances d_m from a given point inside the crystal to each face ($m = 1, \dots, M$). For the natural faces of a crystal, $(h_m\ k_m\ l_m)$ are integers. d_m should have been carefully measured and an estimated standard deviation of d_m should be available. The I Bragg reflections are identified by their Miller indices $h_1\ k_1\ l_1$ and an instrument-independent crystal-based azimuthal angle ψ_1 as defined by Schwarzenbach & Flack (1989, 1992). The above data must be available on the bdf for **LSABS** to function correctly.

By default the net intensities of the reflections (Schwarzenbach & Flack, 1991) are not modified by **LSABS**: the calculated quantities $A=1/T$, $\langle T \rangle$ (i.e. T_{bar}) and $\partial T/\partial d_m$ are simply stored on the bdf.

The recommended method of placing raw diffractometer information including crystal face indices and distance data on the bdf is first to transform a diffractometer-specific file into a CIF using DIFRAC (Flack, Blanc & Schwarzenbach, 1992). This file may then be read with **CIFENT** (same input lines as **CIFIO**) to create a bdf. This process compactly loads all of the necessary raw experimental data onto a bdf in a diffractometer-independent way.

4.31.7. File Assignments

- Reads data from the input archive bdf.
- Writes data to the output archive bdf.

4.31.8. Examples

```

title      Create an absorption-corrected |Fo| bdf
CIFENT     cfin
STARTX     upd
sgname     -C 2YC
REFCAL     neti
LSABS      gauss
shape      needle 1 2 0
grid       10 10 24
sgrid      16
REFCAL     frel
SORTRF     aver 1  frel

```

The above example shows the use of *STARTX*, *REFCAL*, *LSABS*, and *SORTRF* to create a bdf with $|F_o|$ data from diffractometer data produced in CIF format by the DIFRAC program. *STARTX* calculates the direct and reciprocal cell metrics, loads the scattering factor tables, atomic radii, and checks the symmetry information - the *sgname* line is necessary in *STARTX* as this information is not stored with the raw diffractometer data. *REFCAL* calculates the reflection information. *LSABS* performs a Gaussian grid integration and applies the absorption correction to the net intensities in the bdf. As the crystal is needle-shaped, the Gaussian grid is oriented with the needle direction [1 2 0] parallel to the grid z axis. A finer grid is defined along the needle axis than in the other directions. A second pass through *REFCAL* is needed to convert the net intensities to $|F_o|$ corrected for absorption. Finally the absorption-corrected data are sorted and averaged leaving a bdf ready for structure solution and conventional $|F_o|$ refinement.

```

title      Create a net intensity bdf
CIFENT     cfin
STARTX     upd
sgname     -C 2YC
REFCAL     neti inst 1 incl
LSABS
SORTRF

```

A bdf with net intensity data is created. *LSABS* performs an analytical absorption correction and stores the resulting values in the bdf. The bdf is suitable for the refinement of the crystal shape and decay parameters.

4.31.9. References

- Blanc, E., Schwarzenbach, D. & Flack, H.D. (1991). *J. Appl. Cryst.* **24**, 1035-1041
- Busing, W. R. & Levy, H. A. (1957). *Acta Cryst.* **10**, 180-182.
- de Meulenener, J. & Tompa, H. 1965. *Acta Cryst.* **19**, 1014-1018.
- Flack, H.D., Blanc, E. & Schwarzenbach, D. (1992) *J. Appl. Cryst.* **25**, 455-459.

- Ibers, J.A. & Hamilton, W.C. (1974) Eds. *International Tables for X-rays Crystallography Vol. IV*, Birmingham, England: Kynock Press
- Schwarzenbach, D. & Flack, H. D. (1989). *J. Appl. Cryst.* **22**, 601-605.
- Schwarzenbach, D. & Flack, H. D. (1991). *Acta Cryst.* **A47**, 134-137.
- Schwarzenbach, D. & Flack, H. D. (1992). *J. Appl. Cryst.* **25**, 69.

4.32. LSLS: Structure factor least-squares refinement

Authors: Eric Blanc & Dieter Schwarzenbach

Contact: Eric Blanc, Institut de Cristallographie, University of Lausanne, 1015 Lausanne, Switzerland

LSLS is a least-squares refinement program that follows the recommendations of the IUCr Subcommittee on Statistical Descriptors (Schwarzenbach et al., 1989). All structural parameters may be refined, with scale, Becker & Coppens extinction, dispersion, crystal shape parameters and the volume ratio of the twins. The refinement is a full matrix refinement based on net intensities or $|F|^2$. Restraints on distances, bond angles, dihedral angles and rigid links are included. Various weighting schemes are provided, and variances of adjusted parameters are computed according to any scheme.

4.32.1. Calculations Performed

The least-squares procedure conveniently furnishes estimators for crystallographic parameters which minimize (1)

$$\Delta = \sum_k w_k (Y_k^{\text{obs}} - Y_k^{\text{calc}})^2, \quad (1)$$

where Y^{obs} is an observation, such as a measured intensity, a restraint or a face distance of the crystal. Y^{calc} is the corresponding value computed by the model and is a function of the variables $\{v_i\}$. The weights w_k are in general any non-negative real numbers independent of the observations. They form a diagonal weighting matrix \mathbf{W} .

Apart from when Y^{calc} is linear in the variables $\{v_i\}$, minimisation of (1) with respect to v_i leads to a system of equations which is in general not solvable. The solution is then approximated by iterations of the linearized problem around the n^{th} refinement step. The vector of shifts from the n^{th} to the $n+1^{\text{th}}$ step is given by the linear system (2), where \mathbf{A} is the design matrix with $A_{ki} = \partial Y_k^{\text{calc}} / \partial v_i$ and \mathbf{N} the normal equations matrix $\mathbf{N} = \mathbf{A}^T \mathbf{W} \mathbf{A}$.

$$\sum_k w_k [Y_k^{\text{obs}} - Y_k^{\text{calc}}|_n] \frac{\partial Y_k^{\text{calc}}}{\partial v_i} \Big|_n = \sum_k w_k \frac{\partial Y_k^{\text{calc}}}{\partial v_i} \Big|_n \frac{\partial Y_k^{\text{calc}}}{\partial v_j} \Big|_n [v_j|_{n+1} - v_j|_n]$$

or $\mathbf{A}^T \mathbf{W} \delta \mathbf{o} = \mathbf{A}^T \mathbf{W} \mathbf{A} \delta \mathbf{v}$ (2)

$$\delta \mathbf{v} = \mathbf{N}^{-1} (\mathbf{A}^T \mathbf{W} \delta \mathbf{o})$$

is then an unbiased estimator of $(\mathbf{v}_{\text{solution}} - \mathbf{v}_n)$ whatever weights are chosen. It may be shown that an unbiased estimate of the variance-covariance matrix \mathbf{V}_v of the parameters \mathbf{v} is obtained through

$$\mathbf{V}_v = \mathbf{N}^{-1} (\mathbf{A}^T \mathbf{W} \mathbf{V}_o \mathbf{W} \mathbf{A}) \mathbf{N}^{-1} \quad (3)$$

by using an unbiased estimate of the variance-covariance matrix of the observations \mathbf{V}_o . Weights $\mathbf{W} = \mathbf{V}_o^{-1}$ i.e. $w_k = 1/\sigma^2$ where σ^2 is the variance (not an estimate) of the k^{th} observation lead to a minimum-variance estimate of \mathbf{v} and \mathbf{V}_v , and a simplified form of (3), $\mathbf{V}_v = \mathbf{N}^{-1}$, where no extra calculation is necessary to obtain \mathbf{V}_v from \mathbf{N}^{-1} . Following the recommendation of the IUCr Statistical Descriptors Subcommittee (Schwarzenbach *et al.*, 1988), the estimated variance-covariance matrix of the parameters is not scaled by the global goodness-of-fit parameter S ,

$$S = \sqrt{\frac{\Delta}{E(\Delta)}} \quad \text{with } E(\Delta) = \text{trace}(\mathbf{V}_o \mathbf{W}) - \text{trace} \mathbf{N}^{-1} (\mathbf{A}^T \mathbf{W} \mathbf{V}_o \mathbf{W} \mathbf{A}). \quad (4)$$

\mathbf{V}_v is computed on the last cycle. Several types of crystallographic observations can be compared with the model: intensities I^{obs} , crystal shape parameters d^{obs} and restraints R^{obs} . Intensities are calculated by the formula

$$I_k^{\text{obs}} \approx I_k^{\text{calc}} = K_k L_k P_k T_k(\{d\}) y(\{e\}) \sum_i x_i (A_i^2 + B_i^2), \quad (5)$$

where K_k is a scale factor, L_k is the Lorentz factor of the k^{th} reflection, and P_k its polarization factor. T_k is the transmission factor depending on the crystal shape parameters $\{d\}$, y is the Becker & Coppens extinction $\{e\}$ and x_i is the volume ratio of the i^{th} twin. The real and imaginary structure factor components are given by

$$A_i = \sum_{\text{atoms}} m p \sum_s t ((f+f') \cos \varphi_i - f'' \sin \varphi_i). \quad (6a)$$

$$B_i = \sum_{\text{atoms}} m p \sum_s t ((f+f') \sin \varphi_i + f'' \cos \varphi_i). \quad (6b)$$

where the sum over s means over the symmetry-equivalent positions, m is the multiplicity factor and p the population parameter, t is the displacement factor, f is the atom form factor, f' the real part of anomalous dispersion and f'' the imaginary part of anomalous dispersion. The angle φ_i is given by $\varphi_i = 2\pi (h_i x + k_i y + l_i z)$ where h_i , k_i and l_i are the indices of the reflection $(h k l)$ for the i^{th} twin.

4.32.2. General Parameters

The general parameters detailed in this section can be refined. Apart from the overall displacement factor and the scale factor(s) which are always refined unless an appropriate **noREF** line is inserted, refinement of these parameters must be turned on in the **LSLS** line, and not via **ref** lines. Some of these parameters are tested to see if they are physically meaningful. If one parameter takes a non-physical value, the program prints a warning and either (a) stops, or (b) resets the parameter to the closest physical value, or (c) continues, depending on the user's choice.

4.32.3. Scale Factor(s)

The refined scale factor (**skf**) is not the structure scale factor commonly used in Xtal. It places the square of the structure factor on the same scale as the primary observation, such that this latter value remains untouched by any data-reduction procedure during the refinement. When sets of reflections have different scales, they can be refined separately or as a single variable. In the latter case, constraints are automatically inserted so that $\text{skf}(n) = \text{constant} * \text{skf}(1)$.

4.32.4. Crystal Shape Parameters

The distances (**shp**) from an arbitrary origin inside the crystal to faces can be adjusted to optimize crystal shape and transmission factors. The measured values for these parameters are used as restraints,

$$d_k^{\text{obs}} = d_k^{\text{calc}} \equiv d_k,$$

where d_k^{obs} is the measured value for the k^{th} distance, d_k^{calc} is the calculated distance, which is identical to the parameter d_k .

WARNING: To fix the origin of the crystal, the distances to 3 non-coplanar faces must be kept invariant. The program does not choose these faces automatically. They should be as mutually perpendicular as possible, and sufficiently large to have well-characterized Miller indices. However, when the invariant faces have been chosen, weights on these observations are automatically set to zero.

To take into account systematic errors due to magnification effect, refined distances can be constrained, or a correlation value between observations can be used. In the latter case, the variance-covariance matrix \mathbf{V}_o is no longer diagonal, but the weighting matrix remains diagonal even in the case of $1/\text{esd}^2$ weights.

4.32.5. Extinction

Becker & Coppens extinction parameters can be applied or refined, either isotropically or anisotropically. Following Zachariasen (1967), one can distinguish two types of secondary extinction. Type I refers to crystals in which the extinction is dominated by the mosaic distribution (described by g), whereas effective particle size (related to the parameter ρ) is the critical parameter describing extinction in type II crystals. For anisotropic extinction, g and ρ become tensorial quantities as defined according Thornley and Nelmes (1974).

Several extinction refinement options are available: g and ρ can be refined separately choosing type I or type II respectively. In this case, primary extinction is neglected. g and ρ can also be refined simultaneously (but only g can be tensorial) allowing primary extinction to be taken into account. Either a Lorentzian or a Gaussian distribution of the mosaic spread g can be selected.

A short summary of used expressions is given below. Full details can be found in Becker & Coppens (1974a, 1974b, 1975) and Zachariasen (1967).

$\psi = \frac{g}{\sin 2\theta}$ for type I, $\psi = \rho$ for type II, and for general extinction :

$$\psi = \frac{g \rho}{1 + \rho \sin 2\theta} \quad (\text{Lorentz}), \quad \psi = \frac{g \rho}{\sqrt{1 + \rho^2 \sin^2 2\theta}} \quad (\text{Gauss})$$

$$\gamma_p = \frac{3}{2} \frac{\lambda^4}{v^2} a^2 \quad \text{and} \quad \gamma_s = \frac{\lambda^4}{v^2} a^2 \bar{T}$$

then $x_p = \gamma_p F_c^2 \rho^2$ and $x_s = \gamma_s F_c^2 \psi^2$ and finally :

$$y = y_p(x_p) y_s(y_p(x_p) x_s) \quad \text{with} \quad y_i(x_i) = \sqrt{1 + c_i x_i + \frac{a_i x_i^2}{1 + b_i x_i}} \quad i = p, s$$

The parameters g and ρ are related respectively to the half-width of the mosaic spread distribution $\Delta_{1/2}$ expressed in radians and to the size of the mosaic blocks r expressed in cm by

$$\Delta_{1/2} = \frac{\sqrt{\ln 2 / 2\pi}}{g} \quad (\text{Gaussian}) \quad \Delta_{1/2} = \frac{1}{2\pi g} \quad (\text{Lorentzian})$$

$$r = \rho \lambda.$$

The values of g and ρ given in the program listing are multiplied by 10^4 .

4.32.6. Twinning parameter(s)

(Pseudo-) merohedral twin laws may be determined automatically and entered onto the bdf by *CREDOC* for use by LSLs. Twin laws may also be entered directly in LSLs by use of **twinop** lines. By default one twin law *viz.* the identity operation is assumed. For centrosymmetric space groups all twin laws are generated by *CREDOC* or must be entered by way of **twinop** lines. For non-centrosymmetric space groups, each twin law generated by *CREDOC* or entered by way of a **twinop** line corresponds to a pair of twin laws: *viz.* the one indicated and the one related to it through a center of inversion. The two laws of the inversion-related pair in non-centrosymmetric space groups have serial numbers i and $i+n$ where n is the number twin matrices on the bdf or **twinop** lines given in LSLs.

Twinning (**twi**) is treated by refining the volume fractions x_i of each twin component and automatically constraining their sum to be 1.0.

For non-centrosymmetric structures, attention has to be paid to two points. *Primo*, anomalous dispersion should be applied. *Secundo*, the bdf should be prepared with each reflection and its Friedel opposite in separate packets: **seprf1** on the **SORTRF** line.

The Becker and Coppens extinction theory used in LSLs does not take twinning into account, implying that the coupling constant between the incident and diffracted beams is forced to be symmetric under the lattice-point group symmetry.

The following example illustrates the way to handle twins in the non-centrosymmetric space group *P23*. *CREDOC* generated 2 (pairs of) twin laws:

$$\begin{array}{llll} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} & \text{for (1) Identity} & \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} & \text{and (3) Centre} \\ & & & \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix} \end{array}$$

$\begin{pmatrix} 0 & -1 & 0 \\ -1 & 0 & 0 \end{pmatrix}$	for (2) 2-fold axis $\begin{pmatrix} 0 & -1 & 0 \\ -1 & 0 & 0 \end{pmatrix}$	and (4) Mirror $\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$
$\begin{pmatrix} 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & -1 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 1 \end{pmatrix}$

In order to refine only the enantiomorph-polarity parameter in this case starting with a value of 0.5, the following lines are necessary:

```
twinop      *10 1 0.5 0.5
noref      twi(2) twi(4)
```

4.32.7. Dispersion and Neutron-Scattering-Factor Parameters

Assuming that refinement of dispersion parameters has been turned on by `rd` on the `LSLS` line, dispersion parameters can be addressed individually by `re` or `im` for the real or the imaginary parts, or as a set with `dsp`.

Neutron scattering factor refinement is achieved by turning on dispersion refinement and keeping imaginary parts invariant.

4.32.8. Overall Displacement Parameter

The overall displacement parameter is automatically refined if at least one atom's displacement parameter type is set to overall. It can be held invariant by using `noref uov`.

4.32.9. Atomic Parameters

4.32.9.1. Positional Parameters

Atomic positional parameters are always refined unless the atom is in a special position or they are explicitly constrained. `noref <atom>` fixes all atomic parameters of the named atom.

4.32.9.2. Displacement Parameters

The program can be used to refine overall, isotropic or anisotropic displacement parameters. In mixed mode, the displacement-parameter type of an atom is set in accordance to the value on the `bdf`. However it can be modified by the use of `ref` and `noref` lines. After refinement, the mode used is written on the output `bdf`. If the displacement factor is non-positive definite, the program will either stop (`ts`), or reset it to a positive value (`tr`), or continue with non-physical values (`tp`).

4.32.10. Invariant Parameters

`ADDATM` generates symmetry constraints for atoms on special positions which are written on logical record `lrcons`. See the documentation of `ADDATM` for details.

A `noref` line can fix any parameter. The general form of the arguments is:

$(p1/p2)(a1/a2)$ fixes from atomic parameter $p1$ to atomic parameter $p2$ of all atoms in list between $a1$ and $a2$ inclusive. $(p1/p2)$ can be replaced by $p1$ if only one parameter is to be fixed, and $a1$ can replace $(a1/a2)$ if there is only one target atom.

Atomic parameters are : **x**, **y**, **z**, **u**, **u11**, **u22**, **u33**, **u12**, **u13**, **u23**, **pop**

a1/a2 fixes all atomic parameters from atom a1 to atom a2. a1/a1 is abbreviated a1.

When an atom name is identical to an atom type name, then a1 and a2 will hold for atom types instead of individual atoms.

g1(n1/n2) fixes general parameters g1, which can be **uov**, **skf**, **twi**, **shp**, **ext**, **dsp**, **re** or **im**, from n1 to n2. If only one parameter is kept invariant, n1 can be used for n1/n2.

For dispersion refinement, **dsp** fixes real and imaginary parts and it is equivalent to (**re/im**).

The syntax of the **ref** line is identical to that of **noref**. When mixed displacement mode has been selected on the **LSLS** line, **ref/noref** lines can be used to change atomic displacement types. **ref U** converts the displacement type to isotropic, while **ref U11/U23** set the displacement type to anisotropic.

```
ref      U(N)   (U11/U23)(N2)
ref      U(N1)  U(N3/N6)
```

These examples show how to set to isotropic the displacement parameter type for all 6 nitrogen atoms, except for atom N2. There is a difference in the resulting displacement parameters values between these two examples. In the first example, all nitrogens are first changed to isotropic, and then the displacement factor of atom N2 is reset to an anisotropic tensor. In the second example, the displacement parameter of atom N2 is not changed at all.

WARNING If atom types Y and U appear in the atom type list, atomic parameters y and u must be lowercase.

4.32.11. Constrained Parameters

As for invariant parameters, symmetry constraints are read from the input bdf logical record *lrcons*. Linear constraints between parameters can also be introduced by **constr** lines :

$p = Q + f(1)*r(1) + \dots + f(n)*r(n)$ where :
 p = object (dependent) parameter (same format as ref/noref lines)
 Q = constant term
 f(i) = ith multiplicative term
 r(i) = ith reference (independent) parameter (same format as ref/noref lines).

The specification of the parameters is the same as in **ref/noref** lines, but object and reference parameters must be uniquely determined. It should be noted that the constant terms Q and f(i) are mandatory, and blanks are not allowed.

WARNING Origin fixing is carried out by shift-limiting restraints, following the algorithm of Flack & Schwarzenbach (1988). Additional constraints should not be used in LSLS in polar space-groups.

The normal-equations matrix is constructed for independent variables only to minimize storage size and inversion time. The constraints modify the derivative vector, which leads to an unconstrained least-squares problem. The dependent parameters p^o contribute to the derivatives vector of the reference parameters p^r , which become:

$$\frac{dY}{dp^r} = \frac{\partial Y}{\partial p^r} + \sum_i \left[\frac{dp_i^o}{dp^r} \frac{\partial Y}{\partial p_i^o} \right] = \frac{\partial Y}{\partial p^r} + \sum_i f_i \frac{\partial Y}{\partial p_i^o} \quad (8)$$

The dependent parameter estimated standard deviation is then given by (9) :

$$\sigma^2(p^0) = \sum_i f_i^2 \sigma^2(p_i^r) + 2 \sum_{i < j} f_i f_j \text{cov}(p_i^r, p_j^r). \quad (9)$$

4.32.12. Restraints

Restraints Φ are additional stereochemical pseudo-observations used to restrict interatomic geometry. For a discussion of the relations between rigid bodies and restraints, see Didisheim & Schwarzenbach (1987). Distance, bond-angle, dihedral-angle and rigid-bond restraints are provided. If \mathbf{M} and \mathbf{M}^{-1} are respectively the direct and reciprocal metric tensors, \mathbf{x}_i is the position of atom i expressed in the crystal coordinates, β_i the displacement tensor of atom i expressed in the same basis, $\mathbf{n}_1 = (\mathbf{x}_2 - \mathbf{x}_1) \wedge (\mathbf{x}_3 - \mathbf{x}_2)$, $\mathbf{n}_2 = (\mathbf{x}_3 - \mathbf{x}_2) \wedge (\mathbf{x}_4 - \mathbf{x}_3)$ and V the volume of the cell, then restraints are given by (10) :

$$\begin{aligned} \text{distance} \quad \Phi &= \sqrt{(\mathbf{x}_2 - \mathbf{x}_1)^T \mathbf{M} (\mathbf{x}_2 - \mathbf{x}_1)} & (10a) \\ \text{bond angle} \quad \Phi &= \text{Arc cos} \frac{(\mathbf{x}_1 - \mathbf{x}_2)^T \mathbf{M} (\mathbf{x}_3 - \mathbf{x}_2)}{\sqrt{(\mathbf{x}_1 - \mathbf{x}_2)^T \mathbf{M} (\mathbf{x}_1 - \mathbf{x}_2)} \sqrt{(\mathbf{x}_3 - \mathbf{x}_2)^T \mathbf{M} (\mathbf{x}_3 - \mathbf{x}_2)}} & (10b) \end{aligned}$$

$$\begin{aligned} \text{dihedral angle} \quad \Phi &= \text{Arc cos} \frac{\mathbf{n}_1^T \mathbf{M}^{-1} \mathbf{n}_2}{\sqrt{\mathbf{n}_1^T \mathbf{M}^{-1} \mathbf{n}_1} \sqrt{\mathbf{n}_2^T \mathbf{M}^{-1} \mathbf{n}_2}} & \text{if } |\cos \Phi| \leq |\sin \Phi| \\ &= \text{Arc sin } V \frac{\sqrt{(\mathbf{x}_3 - \mathbf{x}_2)^T \mathbf{M} (\mathbf{x}_3 - \mathbf{x}_2)} \mathbf{n}_1^T (\mathbf{x}_4 - \mathbf{x}_3)}{\sqrt{\mathbf{n}_1^T \mathbf{M}^{-1} \mathbf{n}_1} \sqrt{\mathbf{n}_2^T \mathbf{M}^{-1} \mathbf{n}_2}} & \text{if } |\cos \Phi| > |\sin \Phi| \end{aligned} \quad (10c)$$

$$\text{rigid link} \quad \Phi = \frac{(\mathbf{x}_2 - \mathbf{x}_1)^T \mathbf{M} (\beta_2 - \beta_1) \mathbf{M} (\mathbf{x}_2 - \mathbf{x}_1)}{(\mathbf{x}_2 - \mathbf{x}_1)^T \mathbf{M} (\mathbf{x}_2 - \mathbf{x}_1)}. \quad (10d)$$

When the normal-equations matrix is ill-defined, it is possible to force it into a positive-definite form, which is easily invertible. The usual way to do this is to use shift-limiting restraints which increase the diagonal element N_{ii} of the variable v_i leading to the degeneracy. Two possibilities are available in LSLs: (a) when the weight w of the restraint is greater than 1, the restraint equation adds the quantity $w(v_i^{\text{obs}} - v_i^{\text{calc}})^2$ to the minimized function (1), so the diagonal element becomes $N_{ii} + w$. This type of shift-limiting restraint may then be viewed as a pseudo-observation whose value is taken to be the variable's value in cycle n , and the restraint imposes that v_i in cycle $n+1$ should be "close to" its former value. (b) If w is smaller than 1, it is interpreted as a Levenberg-Marquardt parameter (see Press, Flannery, Teukolsky & Vetterling (1986)). There is no pseudo-observational justification to this method where the diagonal element is multiplied by a w independent of the N_{ii} . It should be pointed out that tests to determine the optimal Levenberg-Marquardt parameter are not carried out by LSLs.

4.32.13. Weights

LSLS provides 3 different weighting schemes: the usual minimal variance $w = 1/\text{esd}^2$, unit weights $w = 1$,

or weights read from the bdf (idn 1900 is taken as the square root of the weight on Y); this latter possibility allows the user to define his/her own weighting scheme. In any case, the goodness of fit and the parameter variance-covariance matrix are computed according to (3) and (4).

Weights on $|F|$ are computed from weights on the refined quantity Y (either $|F|^2$ or I) by (11) where LPT is set to 1.0 if $Y = |F|^2$. These $|F|$ weights are used in the calculation of the weighted R-factor.

$$w_{|F|} = \frac{L P T K}{\sqrt{Y^2 + 1/(2 w)} \cdot Y} \quad \text{if } I > 0 \quad \text{and } w_{|F|} = \sqrt{2 w} L P T K \quad \text{if } Y \leq 0 \quad (11)$$

4.32.14. Reflection Classification

A reflection can be omitted from the refinement under several conditions: if it is flagged as suffering from extinction (rcode 3), if its weight is 0, if its weighted difference $w^{1/2} |Y^{\text{obs}} - Y^{\text{calc}}|$ is greater than some user defined value, or if Y/esd_Y is less than a user defined threshold. These unused reflections do not contribute to the matrices, Durbin-Watson statistic, and goodness-of-fit. Less-than reflections are those for which Y/esd_Y less than the cutoff value, and those with rcode = 2. In the latter case, the reflections are discarded only if Y^{obs} is greater than or equal to Y^{calc} . The status code is never updated by LSLS.

The meaning of the printed status codes is the following: **F** is for reflections without their Friedel mate, **A** is for systematically absent reflections. Reflections flagged with these codes are always used in the refinement. **E** is for a reflection suffering from extinction, **L** for less-than reflections, i.e. which Y/esd_Y smaller than the threshold, **w** for reflections with a weight of zero, and **R** for a rejected reflection for which $w^{1/2} |Y^{\text{obs}} - Y^{\text{calc}}|$ is greater than the cutoff. Reflections marked with these codes are always excluded from the refinement. **u** is for unobserved reflections; it corresponds to rcode = 2. The asterisk ***** marks reflections which do not contribute to the matrix.

4.32.15. R-Factors

The unweighted and weighted R-factors are printed by the program. The unweighted R-factor is defined as:

M.I.A.

and the weighted R-factor by:

M.I.A.

For the 4 R-factors printed by the program:

- (1) marked "Of contributing reflections" is calculated with R , $Y = |F|$ and $M = \text{NREF}$.
- (2) marked "Of weighted contributing reflections" is calculated with wR , $Y = |F|$ and $M = \text{NREF}$.
- (3) marked "Of Observations" is calculated with R ; for refinement on $|F|^2$, $Y = |F|^2$; for refinement on I , $Y = I$; and $M = \text{NOBS}$.

(4) marked "Of Weighted Observations" is calculated with wR ; for refinement on $|F|^2$, $Y = |F|^2$; for refinement on I , $Y = I$; and $M = \text{NOBS}$.

4.32.16. File Assignments

- Reads parameter- and observation- information from the input archive `bdf`
- Writes refined parameters on output archive `bdf`
- Optionally writes variance-covariance matrix to file `cmx`
- Optionally writes parameters to file `pch`

4.32.17. Examples

```

title      Gd3Ru4Al12
compid     gra
CIFENT     cifin
STARTX     upd
sgname     -P 6C 2C
REFCAL     neti inst 1 excl
LSABS
ADDATM
extinc     typ1 is gaus 0.005
atom       Al1  0.1622  0.3244  0.5763  0.008
atom       Gd   0.1928  0.3856  0.25    0.007
atom       Ru1  0.5     0      0      0.007
atom       Al3  0.3333  0.6667  0.0119 0.008
atom       Al4  0      0      0.25   0.008
atom       Ru2  0      0      0      0.007
atom       Al2  0.5637  0.1274  0.25   0.008  0.95
atom       Ru3  0.5637  0.1274  0.25   0.008  0.05
LSLS       cy 5 ad ax pp tr rs sm
noref      shp(1) shp(3) shp(5)
noref      pop
ref        pop(Ru3) pop(Al2)
constr     x(Ru3)=0.0+1.0*x(Al2)
constr     u(Ru3)=0.0+1.0*u(Al2)
constr     pop(Ru3)=1.0-1.0*pop(Al2)
REGFE      sr 1 nan tab
finish

```

To refine the crystal shape, the absorption program LSABS is run just after the conversion from raw data to net intensities. No averaging is done, and the refinement is made on all intensities. 3 faces are kept invariant. Ru3 and Al2 are constrained to the same position, and the same displacement parameter. The stereochemical disorder of these atoms is taken into account constraining the total population of the site to be equal to 1. The variance-covariance matrix is saved to be used in the calculation of distances by *REGFE*.

```

compid     coanp
CIFENT     cifin
STARTX     upd
lattice    n p
symtry     x,y,z
symtry     -x,y,1/2+z
symtry     -x,1/2+y,-z

```

```
symtry      x,1/2+y,1/2-z
NEWCEL
transf      0 0 1  1 0 0  0 1 0
CREDUC
REFCAL      excl fsqr ffac
SORTRF      ord lkh aver 1 sepfrl f2r1
ADDATM
Atomic parameters omitted here for brevity
LSLS        cy 1 f2 cs 3.0
noref       C N O H
LSLS        cy 5 f2 cs 3.0 rp ep
FOURR
MAPLST
finish
```

In this example, the space group is changed by *NEWCEL*, then twins are searched (*CREDUC*), the data are converted to $|F|^2$ and averaged in separate packet for Friedel pairs. The scale factor is refined alone for one cycle, and then all structural parameters are refined, including enantiomorph (absolute structure) parameter. When the initial value of the scale factor is far from the real one, it can be useful to refine the scale factor alone in a first refinement step.

4.32.18. References

- Becker, P.J. & Coppens, P. (1974). Acta Cryst. A30, 129-147.
- Becker, P.J. & Coppens, P. (1974). Acta Cryst. A30, 148-153.
- Becker, P.J. & Coppens, P. (1975). Acta Cryst. A31, 417-425.
- Bernardinelli, G. & Flack, H.D. (1985). Acta Cryst. A41, 500-511.
- Didisheim, J.-J. & Schwarzenbach, D. (1987). Acta Cryst. A43, 226-232.
- Flack, H.D. (1983). Acta Cryst. A39, 876-881.
- Flack, H.D. & Schwarzenbach, D. (1988). Acta Cryst. A44, 499-506.
- Press, W.H., Flannery, B.P., Teukolsky, S.A. & Vetterling, W.T. (1986). Numerical Recipes. Cambridge University Press. Cambridge.
- Thornley, F.R. & Nelmes, R.J. (1974). Acta Cryst. A30, 748-757.
- Schwarzenbach, D. et al. (1989). Acta Cryst. A45, 63-75.
- Zachariasen, W.H. (1967). Acta Cryst. 23, 558-564.

4.33. *LSQPL*: Least-squares planes & lines

Authors: George Davenport and Howard Flack

Contact: Howard Flack, Laboratoire de Cristallographie, University of Geneva, CH-1211 Geneve 4, Switzerland.

LSQPL calculates the least-squares plane or line through a set of specified atoms and the angles between sets of least-squares planes and lines.

4.33.1. Calculations Performed

LSQPL calculates the equations of least-squares lines and planes. The method of calculation is that of Schomaker, Waser, Marsh and Bergman (1959) with the exception of the method used for obtaining the roots of the cubic equation in Lagrangian multipliers. Schomaker *et al.* used an iterative technique, whereas LSQPL solves for the roots directly by the general solution equations. LSQPL also calculates the values and their estimated standard deviations of the distances of atoms from the least-squares planes or lines, and of the angles between planes and lines. The least-squares plane or line equation is output both in fractional and orthogonal Angstrom coordinates parallel to a^* , b' , c (Rollett, 1965). Cell and symmetry data are taken from the bdf. The bdf also provides atom coordinates which may be supplemented or updated with values read from the line input on **site** lines. The atoms used in the calculations may be supplemented by those related by a centre of symmetry at a position indicated on the **plane** or **line** lines. LSQPL also permits the selection of defining atoms to be used in the least-squares plane calculations (**define** line) and non-defining atoms to be used only to calculate the atom-to-plane or -line distances (**nondef** line).

4.33.2. Choice of Method

The 2.6 version of this program used the algorithm of Ito (1981). The algorithm of Ito works well for the calculation of least-squares planes. However (see Flack, 1990), when applied to the calculation of best-lines through a set of atoms, it either finds the best-plane or has poor convergence properties. The much older algorithm of Schomaker *et al.* (1959), used in the X-RAY 76 version of LSQPL by Roger Chastain and Wilson de Camp, has the considerable advantage of applying to both least-squares lines and planes. Of the original Davenport LSQPL XTAL program the input/output sections have been retained and the calculation section rewritten by Flack. For the calculation of the variance-covariance matrix of the plane normal/line direction, it should be pointed out that the formula $\mathbf{M} = \mathbf{C}^{-1}$ given by Hamilton (1961,1964) [equation 15 of Hamilton (1961) and equation 22, section 5-8 of Hamilton (1964)] are incorrect. Hamilton's derivation is based on taking expectation values of $(m_i - \mu_i)(m_j - \mu_j)$ assuming that the three m_i are independent random variables. In fact \mathbf{m} should be constrained to be a unit vector, leading to only two independent random variables, as clearly indicated in the work of Waser, Marsh and Cordes (1973). The variance-covariance matrix of \mathbf{m} should thus be evaluated by elimination of one of the m_i contributing to \mathbf{C} before inversion followed by augmentation (by propagation of errors), after inversion, for the eliminated variable. A difficulty in the calculation of estimated standard deviations of interplanar or interlinear angles is a $\sin 2\theta$ appearing in the denominator of the expression for converting $\text{variance}(\cos\theta)$ into $\text{variance}(\theta)$ [Equation 16 of Waser, Marsh & Cordes (1973) and equation 26 of Ito (1981)]. For values of θ close to 0, this leads to very large e.s.d.s or division by zero. In LSQPL the problem is resolved by using a finite difference rather than a differential formula:

$$\sigma(\theta) = | \arccos[\cos\theta \pm \sigma(\cos\theta)] - \theta |$$

4.33.3. File Assignments

Reads existing atom and symmetry data from the input archive bdf.

4.33.4. Example

```
LSQPL
PLANE
DEFINE N11 C12 C13 C14 C15 C16
nondef Ni
LINE
DEFINE Ni N1 C1 S1
```

4.33.5. References

- Flack, H.D. (1990). *Acta Cryst.* In preparation.
- Hamilton, W.C. (1961). *Acta Cryst.* **14**, 185-189.
- Hamilton, W.C. (1964). *Statistics in Physical Science*. The Ronald Press Company: New York.
- Ito, T. (1981). *Acta Cryst.* **A37**, 621-624.
- Rollett, J.S. (1965). *Computing Methods in Crystallography*. Pergamon Press: New York.
- Schomaker, V., Waser, J., Marsh, R.E. & Bergman, G. (1959). *Acta Cryst.* **12**, 600-604.
- Waser, J., Marsh, R.E. & Cordes, A.W. (1973). *Acta Cryst.* **B29**, 2703-2708.

4.34. *LSRES*: Restraint editor

Author: Eric Blanc

Contact: Eric Blanc, Institut de Cristallographie, University of Lausanne, 1015 Lausanne, Switzerland

*LSRES creates or updates the restraints logical records (*lrbond:*, *lrangl:*, *lrdihe:*, *lrrigi:*) on the archive bdf. It also computes the values of these restraints.*

4.34.1. Calculations Performed

LSRES provides for the creation or modification of the logical records *lrbond:* (distance restraints), *lrangl:* (bond angle restraints), *lrdihe:* (dihedral angle restraints) and *lrrigi:* (rigid bond restraints). The atoms must be on the input bdf, and for each rigid bond restraint, at least one atomic displacement factor must be anisotropic, and neither may be overall. All restraints that are output on the archive bdf are computed and tested to make sure two atoms do not occupy the same position.

LSRES accepts restraints between symmetry-equivalent atoms as defined by **symeq** line. Introduction of atoms outside the unit cell is achieved by translations greater than 1.

In update mode, when an atom name is used it is taken to apply to all symmetry-equivalent atoms based on the named atom, whereas a symmetry-equivalent atom name applies only to the atom generated by the particular symmetry operation. This means that if the user gives atom names instead of symmetry-equivalent names, *LSRES* will select the first matching restraint (**restr** line) or all matching restraints (**delres** line). Besides, for the **delres** line only, the "wild field" **all** is for any atom in any symmetry-equivalent position. For example:

```
delres dist C1 : delete distance restraints containing atom C1
delres dist C1 all : delete distance restraints
containing atom C1 as first field
delres dist all C1 : delete distance restraints containing atom C1 as second field
```

This can be useful to delete bond angle and dihedral angle restraints.

4.34.2. File Assignments

- Read atoms and restraints from the input archive bdf.

- Write restraints to the output archive bdf.

4.34.3. Examples

```
LSRES  apr
restr  dist  C31  C71  1.458  0.0001
restr  bond  C31  C71  0        0.0001
restr  dist  C32  C72  1.458  0.0001
restr  bond  C32  C72  0        0.0001
```

Introduction of 4 restraints, 2 bond distances, and 2 rigid links.

```
LSRES  upd
symeq  C32  C325  -z  -x  -y
symeq  C72  C725  -z  -x  -y
delres dist
delres bond C71
restr  bond  C32  C325  0        0.0001
restr  bond  C72  C725  0        0.0001
```

Creates symmetry-equivalent atoms to C32 & C72, delete all distances restraints, delete all rigid bond restraints on atom C71 and all its symmetry-equivalent atoms, and insert restraints between C32 and a symmetry-equivalent atom, and C72 and a symmetry-equivalent atom.

4.35. *MAPLST*: List map densities

Author: James Hester, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

MAPLST outputs the electron density values of a "map" file as ACSII text. This output is intended specifically for use with the *Mathematica(R)* package but is useful for any program requiring the map data as text.

4.35.1. Purpose

MAPLST outputs a *FOURR* (or *SLANT*, etc.) map bdf as 80 column text ascii data. The output line file (`asc`) consists of:

1. header decriptor line
2. line containing the 3 integers which are the number of sections (i.e. layers) in the map, the number of rows in a section and the number of density points in a row.
3. density value for each point as floating point numbers. These are listed continuously and are separated only by blanks.

4.35.2. File Assignments

Reads the map densities from the specified bdf (default is `map`) Writes the map densities to the line file `asc`.

4.35.3. Example

```
MAPLST sla
```

Output the map densities from the *SLANT* map file to asc.

4.36. *MAPXCH*: Import theoretical electron density maps **NEW**

Authors: Doug du Boulay, Materials and Structures Laboratory, Tokyo Institute of Technology, Nagatsuta, Midori-Ku, Yokohama Japan.

MAPXCH currently imports a abinitio (DFT) 3D electron density map from the open source *ABINIT* project or from *Wien2k* lapw5-3D into an Xtal archive file.

4.36.1. Purpose

To compare theoretical and experimental charge densities it is useful to know that all densities are on the same scale, with the same contour intervals and the same map orientations, so that topographic features can be reliably compared. To this end *MAPXCH* was written to import the electron density maps of the completely unrelated and unaffiliated external density functional theory programs *Wien2k* and *ABINIT* into *Xtal*, to exploit the relatively reliable contouring machinery that already exists herein (amongst other things).

4.36.2. Using *Wien2k* data

At last checking, the *Wien2k* DFT code included one program, lapw5, intended for determining the electron density on a planar grid of values for contouring in external software. *Florent Boucher* kindly made available (to existing *Wien2k* users) a modified version of lapw5, heretofore named lapw53D, with which it was possible to compute the electron density on a 3D grid of values, in much the same manner as an existng planar lapw5 calculation.

The program lapw53D requires a default filename mapping file - lapw53D.def with the contents:

```
5 , 'case.in53D', 'old', 'formatted', 0
6 , 'case.output5', 'unknown', 'formatted', 0
8 , 'case.struct', 'old', 'formatted', 0
9 , 'case.clmval', 'old', 'formatted', 0
10, 'case.3dmap', 'unknown', 'unformatted', 0
11, 'case.clmvaldn', 'unknown', 'formatted', 0
12, 'case.sigma', 'unknown', 'formatted', 0
20, 'case.rho_onedim', 'unknown', 'formatted', 0
21, 'case.rho', 'unknown', 'formatted', 0
```

For use with *Xtal* we need a lapw53D control file case.in53D with the contents:

```

          0          0          0      10000000
10000000          0          0      10000000
          0      10000000          0      10000000
          0          0      10000000      10000000
3 2 3
97 97 49
DIFFADD
ANG VAL NODEBUG
NONORTHO
```

Note especially the grid points 97 97 49 which will vary on a study by study basis, but if they are too large, the whole map cannot be squeezed into the F77 fixed length *Xtal* memory. For *Xtal* it is necessary that the

unduplicated grid dimensions be divisible by 2, 3 and/or 5 and for trigonal symmetry divisibility by 3 is essential.

So, having run your Wien2k DFT SCF calculations to convergence, run `lapw53D` to calculate the 3D grid data points which are stored in the raw binary `case.3dmap` file.

Then, run Xtal with the command `MAPXCH w2k2map inpfile case` to read both the `case.struct` file to obtain unit cell, symmetry and atom information and, also the `case.3dmap` to transform the 3D gridded Wien2k density map into the Xtal map file as a full cell map. After the Xtal archive is created `STARTX` and `ADDATM` are run in update mode to check the symmetry and atom site information, then the archive is copied to the `.map` file.

4.36.3. Using ABINIT data

To obtain a valence density output from the program ABINIT it is sufficient to add the command

```
prtden 1
```

to your ABINIT control file. Then at successful completion of the job all cell, symmetry and atom site information is written, along with the resultant valence density, to the file `case_DEN`. In contrast to Xtal, ABINIT plays hard and fast with the symmetry and cell in order to minimize cell volumes and optimize computation times. As a result some deductive logic is required to transform the 3D map data and cell to a more standard crystallographic setting. This can be lossy and the MAPXCH code is experimental.

One problem with the ABINIT `case_DEN` file is that the entire electron density map is written as a single Fortran90 record of double precision data. Reading it within Xtal stresses the memory limitations, severely limiting the size of map that can be accommodated.

As with the Wien2k data input method, after creating the Xtal archive it is passed to `STARTX` and `ADDATM` to check the symmetry and atom sites, before being copied to a `.map` file for contouring within xtal.

4.36.4. File Assignments

- Optionally read ABINIT `case_DEN` file
- Optionally read Wien2k `case.struct` and `case.3dmap` files
- Initializes an xtal archive `bdf` and creates an xtal `map` file.

4.36.5. Examples

```
MAPXCH w2k2map inpfile P63 : read Wien2k P63.tmp map file
UOV 0.22
```

Read a Wien2k 3D map file, presumably a difference electron density map, straight into an xtal `.map` file

```
MAPXCH abi2map noterm abifile sin_o_DEN : read ABINIT map
UOV 0.22 : spawns STARTX and ADDATM
ADDATM upd: : reoverride builtin value
UOV 0.00
```

```
ADDREF ffac : generate dummy reflection data
reduce nocon
limits *9 yes : whole sphere
hklgen hkl frel sigf
```

Chapter 4. Reference Manual

```
SORTRF aver 1 frel :      remove redundant reflections

RFOURR : pr 100 :      reverse fourier transform valence density

MODHKL :                purge unphased reflections
purge -4.+19 $1 1 1801

FC frt noex nod list :  Determine Frel as Sqrt((Acore+Aval)^2+(Bcore+Bval)^2)

CRYLSQ ov cy 10 wu :    optimize overall temp
noref skf
noref Si
noref N

FOURR :                  Calculate a difference map
grid 60 60 24
```

This example reads an ABINIT valence electron density map into an xtal input .aan archive file for updating using STARTX and ADDATM before copying to the standard .map file. Dummy reflections are determined (assuming $\lambda=0.5\text{\AA}$) and a reverse Fourier transform calculates the phase as well as A and B. The **Fc frt** option calculates an effective Frel by adding a pseudo core to the valence electrons ¹. Then total density, promolecular structure factors are computed in CRYLSQ and a difference Fourier map is evaluated.

```
MAPXCH abi2map noterm abifile sin_o_DEN :  read ABINIT map
UOV 0.00

ADDATM upd
UOV 0.000

STARTX upd      :          Add valence density form factors
celcon Si
celcon N
setid formfx
Si 0.  4.
Si 0.00836484041  3.99153447
: ... truncated for brevity. Valence form factors from SCATOM
N 1.98792839  4.64325094
setid
exper 1  0.9 :          change the wavelength
end

ADDREF ffac :          generate dummy reflection data
reduce nocon
limits *4 1.0  *9 yes :  whole sphere
hklgen hkl  frel sigf

SORTRF aver 1 frel :      remove redundant reflections

RFOURR th 0.000001 : pr 100 :  reverse fourier transform valence density
idnums $1 304 801 802 700  :

MODHKL :                purge unphased reflections
purge -4.+19 $1 1 1801
FC noex nod list :      calculate Fc for pure valence electrons

FOURR :                  Calculate a valence density difference map
grid 60 60 24
```

Calculate a true ABINIT *difference valence density map*.

4.37. **MODEL**: Build molecule from map

Authors: Ruth Doherty and Sydney Hall

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

MODEL searches peak or atom sites for a connected molecular model. The model interpretation is on the basis of bond lengths and angles, or if information concerning the connectivity sequence of a fragment or molecule is supplied, MODEL attempts to match a canonical description of the input fragment to the connected peak and atom sites. Connected sites are plotted on the printer and a comprehensive list of distances and angles is produced.

4.37.1. Clusters And Subclusters

The first step in the interpretation of a molecular model is to determine which sites are connected. Sites are considered connected if their bond radii overlap. The bond radii of atom sites is extracted from the archive bdf. The bond radius for peak sites is input on the **limits** line. The search for these connections includes all possible symmetry transformations of the sites. Sites connected in this way are placed in the same group and labelled as a *cluster*.

Maximum and minimum bond angles represent a different type of constraint on the site coordination. Within a cluster of sites there may be large number of different ways that the angle constraints can be satisfied. Sites that satisfy a particular interpretation of angle constraints are grouped in a *subcluster*. One of three different approaches must be selected (on the **MODEL** line) for identifying sites that belong to the same subcluster. They are:

4.37.1.1. No subcluster search (**none**)

In this mode no angle limits are applied to the connected sites. All sites within the specified site radii will be included in the model.

4.37.1.2. Connected subcluster search (**conn**)

In this mode sites are accepted into a subcluster by testing the angles in the *order of decreasing peak height and bond connectivity* (the number of bonds to each site). This referred to as the maximum connectivity approach.

4.37.1.3. Weighted mean subcluster search (**mean**)

A more geometric approach to angle constraints is based on weighting sites according to bond lengths and angles they form. Peak sites are assigned weights based on proximity of a connecting bond length to *mean* accepted value $(\text{bondmax} - \text{bondmin})/2$ and bond angles to the mean $(\text{angmax} - \text{angmin})/2$. This can be an important approach for very regular (e.g. 'chickenwire') or highly-coordinated structures. It is not suitable, however, for structures with a range of coordination geometries (e.g. heavy atoms or 5- & 6-membered rings.).

4.37.2. Search Parameters

The ability of MODEL to identify the correct set of connected sites is strongly dependent on the bond length and angle constraints discussed above. When only peak sites are input the default values (see the **limits** line) will permit a range of geometries. In such cases the limits will be adequate for a molecular search provided the peak sites are well-defined and there are not too many spurious peaks in the map. If possible the user should specify limits to suit the stereochemistry of the structure. Be careful - if you are too restrictive, some legitimate peaks may be rejected; if you are too permissive, spurious peaks will complicate the interpretation of the subcluster.

In addition to the bond length and angle modes for site selection, the user may specify a known fragment as a template (see the **connect** line). This approach places stringent stereochemical constraints on the selection of connected sites and should be used whenever possible. It should be emphasised, however, that the fitting of the input model to the peak sites is strongly dependent on the sites that are connected in rings. Special care must be taken that the input ring sites are correct - otherwise the fitting process will probably fail. The reliability of the non-ring atoms is less critical.

4.37.3. Selecting The Best Model

Interpretation of the connected peaks depends on the 'quality' of the Fourier map, the appropriateness of the bond length and angle constraints and the availability of reliable stereochemical information. The 'best' model is selected from two figures-of-merit values.

The first FOM is based on the sum of the subcluster connections, weighted according to peak height. This is a reasonable measure provided the bond length and angle constraints are appropriate for the structure. The expected value for the FOM is very dependent on the number of peak sites entered compared to the number of non-Hydrogen sites in the structure.

A second FOM relies on the stereochemical information input on **connect** lines. If this is used a FOM value is calculated based on the fit of the input model to subcluster peaks. This is a more sensitive FOM than the first and has an optimal value around 2.0. It should be emphasised that, just as the fitting process is strongly influenced by sites connected in rings, this FOM is especially sensitive to matching atoms connected in rings.

4.37.4. Output Information

MODEL outputs a range of numerical and graphical information about the modelling process. Here is a summary of the different sections.

1. The control parameters are listed as a record of the constraints used in the search process. It must be emphasised again that the MODEL calculation is as effective as the constraints are appropriate to the structure being processed.
2. If **connect** lines are input, the atomic coordination of each site is listed. Check to make certain that the specified model is as you intended - this is a common source of error.
3. Each interpretation of a cluster of peak sites satisfying the FOM limits is listed. This listing includes the calculated FOM values, the number of subclusters and the peak connectivity table. The table lists the atom and peak sites in decreasing order of peak height, with information on subcluster allocation, connected peaks and sites that have been assigned from the input model. This table must be referred to when connecting up the projected peak positions output in 4. below. Note that the atom labels stored on the bdf are truncated to 4 characters in MODEL to aid their presentation on the printer plot of the molecule as described in 4.
4. A graphical representation of the peak clusters projected down one or more axes is provided. The number of projection axes can be specified by the user (see the **limits** line). The default is for one projection perpendicular to the best least-squares plane through sites connected in rings (or if no atoms are in rings, all sites are used), and one projection down an axis in the least-squares plane. The second projection is not output if the cluster is approximately planar. The projected peak sites are shown to scale (default is 2.5 Å/cm) and as the peak sequence number preceded by a special site marker. For peaks connected in rings the site marker is an asterisk (*); for peaks satisfying the subcluster constraints, as a plus (+); and for peaks in the cluster but not within the subcluster, as a period (.). In addition to the graphing of peak clusters, projections are also output for all peak sites. This gives a composite picture of the peak positions independent of clustering and subclustering selections.

5. At the conclusion of the graph outputs, the peak sites are listed with the allocated cluster number and, if model sites were input, the assigned atom sites.
6. A table of contact distances less than a specified limit (see field 7 of the **limits** line) calculated using the coordinates from list 5. The default limit is 2.4Å.
7. A table of bond distances for the connected sites (based on the atom radii and the peak radius from the **limits** line) calculated using the coordinates from list 5 and the symmetry equivalent positions.
8. A table of bond angles for the peaks listed in list 5 for all asymmetric units.
9. A table of bond lengths for twinned peaks related only by symmetry operations.

4.37.5. File Assignments

- Reads symmetry, atoms and peak sites from bdf `pek`
- Writes connected atom sites to line file `pch`

4.37.6. Examples

MODEL

This example shows the fully defaulted run. All default search conditions apply. No molecular connections (as defined by **connect** lines) are used in the FOM assessment.

```
MODEL conn
limits 4 0.6 0.9 100 140
connect c1 o1 o2 c6; c7 c2 c6 03; c5 c4 c6; c3 c4 c2
```

The compound described in this input is salicylic acid. If **PEKPIK** has been run to select and sort the top peaks from a Fourier map onto file `pek`, MODEL will then sort the peaks into clusters of peaks that are bonded to one another. Then an analysis of the peaks in terms of some structure will be attempted. Only those peaks which are bonded at a distance of between 1.2 and 1.8 Angstroms and which form angles of between 100° and 140° with the other peaks will be considered in the interpretation. An attempt is made to assign the atom names, given in the **connect** line, to the cluster, or clusters, of peaks which have met the bonding criteria. Up to three interpretations will be attempted (by default). For each interpretation, a plot will be made of the peaks in the cluster projected on the least squares plane of the cluster. After the interpretation has been completed, a list of all the inter-peak distances up to 2.0 Angstroms will be produced.

4.38. **MODHKL**: Modify reflection data

Author: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

MODHKL edits specific reflections in the bdf, global changes to reflection codes, global deletion of reflection items or the removal of reflection packets from the bdf.

4.38.1. Description

During the course of a structure analysis, it may be necessary to make small changes to the reflection data. This program facilitates easy modification of reflection data stored in a bdf. Complicated changes such as the inclusion of new information for every reflection, changes in cell parameters, or expansion of the number of items stored for every reflection will require the use of programs such as *ADDREF*.

MODHKL may be used to make four different types of changes to the reflection data. In all cases the items accessed in a bdf reflection packet are identified by their ID numbers listed in the BDF section at the back of the manual. The user may modify any existing data in *lrrefl*: on the bdf. It is possible to reduce the size of packets by eliminating items. However, expansion of the packets is not possible.

4.38.2. Global HKL Options

4.38.2.1. rcode

The **rcode** line enables the global change of *reflection codes* according to two entered conditions. Each condition on the **rcode** line is composed of four input fields in the order: factor1, item1, factor2, item2. The factors 1 and 2 are floating-point multipliers, and items 1 and 2 are the bdf directory numbers for the specified items. Values are applied for each condition using the following expression to give a *TRUE* or a *FALSE* result:

$\text{factor1} * \text{item1} > \text{factor2} * \text{item2}$

If both conditions are specified, then the new rcode will replace the old rcode only if **BOTH** conditions are *TRUE*. If only condition 1 is specified, then condition 2 will be ignored and the rcode will be changed when condition 1 is *TRUE*. When the new rcode is already equal to the old rcode, the reflection is **NOT** listed as changed even if the conditions are true. The **rcode** line may be used only if an rcode already exists on the bdf. *Only one rcode line may be entered.*

4.38.2.2. remove

The **remove** line enables the global removal of quantities from the reflection packets of a bdf. The fields of the **remove** line(s) must contain valid reflection packet directory numbers for the items to be removed from all packets. These numbers must not be used again in the run. For example, if **purge** lines follow, the same directory numbers cannot appear in these lines.

4.38.2.3. purge

The **purge** line is used to remove reflection packets from the bdf according to the specified conditions 1 and 2. The general form of these conditions is identical to the **rcode** line. If both conditions are entered, then both must be true before a reflection is purged. If only one condition is entered, then a reflection is purged if this condition is true. The **purge** line acts to remove reflections while the **remove** line removes items for every reflection whether some **purge** condition is set or not. It is not possible to set a purge condition based upon an item which is specified to be removed. *Only one purge line may be entered.*

4.38.2.4. Specific modify and delete Options

The **modify** and **delete** lines are used to modify and to delete specific reflections. **modmul** and **delmul** are equivalent commands suitable for editing macromolecule data where there are multiple occurrences of the same reflection. These have been sorted on hkl but not averaged.

The input lines **modify**, **modmul**, **delete**, and **delmul** **must** be entered in the same sort order as the reflections on the input bdf. If this condition is not met an error will occur at the end of processing.

The difference between `modify` and `modmul` (and between `delete` and `delmul`) is that there exists one additional field containing the occurrence number. The occurrence number is the count of which measurement for a given hkl is to be treated; one for the first, two for the second, etc. With use of these lines it is possible to remove reflection packets from the bdf or to modify individual data stored on the bdf. If the occurrence number is omitted, all occurrence will be processed.

Note that neither the hkl nor the multiplicity-epsilon items may be changed by use of MODHKL. They are "packed" floating point words and are outside the scope of this program.

4.38.3. File Assignments

- Reads reflection data from the input archive bdf
- Writes modified reflection data to the output archive bdf

4.38.4. Examples

```
MODHKL
purge 1. 2 0.44           :delete hkl if s >.44
rcode 2 3. 1305 1. 1304   :make rcode=2 if 3sigmaF > F
```

```
MODHKL
modify 0 4 5 1301 143.2 1300 95632.   :modify sI and I
modify 3 5 1 1304 12.6 1305 8.3       :modify F and sF
modify 3 5 1 1308 2                   :modify rcode of same hkl
```

```
MODHKL
remove 2314 2315 2304 2305
delmul 1 1 1 2
```

In this example the items for the F(rel) and σ F(rel) and their Friedel mates will be removed from all packets. In addition, the second occurrence of the (1,1,1) reflection will be dropped from the bdf.

4.39. *NEWCEL*: Apply unit-cell change

Contact Author: H D Flack, Laboratoire de Cristallographie, University of Geneva, CH-1211 Geneve 4, Switzerland

NEWCEL transforms all data in a bdf for a change of unit cell.

4.39.1. Description

NEWCEL undertakes the transformation of data in a bdf due to a change in the base vectors of the unit cell. The transformation is specified as $\alpha' = \mathbf{T}\alpha$ where α is a column vector of the unit cell vectors **a**, **b**, **c** and the transformation matrix **T** is given on the **transf** line by rows. The following quantities in the bdf are transformed:

- the cell dimensions and cell volume;
- the symmetry operators and lattice type (centring);
- the number of atoms of each type per cell;
- the Miller indices *h, k, l* of the crystal faces and of the reference reflections;
- the minimum and maximum Miller indices *h, k, l* of the measured reflections;
- the fractional coordinates and anisotropic displacement parameters of the atoms;
- the Miller indices *h, k, l*;
- the crystal-based azimuthal angles, the multiplicity and the symmetry reinforcement parameter ϵ of the intensity reflections;
- the Miller indices *h, k, l* of the reflection used to refine the cell dimensions.

The programme uses the automatic streaming technique to run [STARTX](#) and [ADDATM](#) to ensure that the information in the bdf is made consistent. Also if *h, k, l, θ* data are available on the bdf, [LATCON](#) is automatically run to refine the transformed cell dimensions.

4.39.2. File Assignments

- Reads data from the input archive bdf.
- Writes data to the output archive bdf.

4.40. [NEWMAN](#): Newman projection of model

Author: Roeli Olthof-Hazekamp, Laboratorium voor Algemene Chemie, Padualaan 8, 3508 TB Utrecht, The Netherlands

NEWMAN calculates stereochemical Newman projections about any bond axis. Projections are output as a diagram on the line printer and/or a plotter. Structure data can be supplied by input lines or the bdf. The program is a RATMAC version of the NEWMAN program, originally written in ALGOL60 by H. Schenk, later in FORTRANIV by N.P. Brandenburg.

4.40.1. Introduction

Newman projections are stereospecific, graphic representations of the torsion angles around a particular bond. These projections are much easier to interpret than angle data. Cell and atom information is read from the bdf. Orthogonal atom coordinates are calculated. A bonding matrix is evaluated, containing all bonds with distances within predefined limits. For each particular bond a Newman projection is constructed: all connected bonds are projected on a plane perpendicular to the central bond. Distances and angles are output, as well as a survey of fractional and projection coordinates and a list of dihedral angles. The projection is printed and

optionally plotted. A **select** line may be given to select only those atoms, which are used for the Newman projections.

Bonds between atoms are determined from the atomic radii values on the archive bdf. These radii may be changed by entering an **atrad** line. A minimum of two bonds must be present to be considered for a Newman calculation and/or plot. The user may also give the bonds in advance. The **bondin** line then contains the name of the central atom, followed by the names of the peripheral atoms.

All bonds are investigated to determine whether they can be assigned as central bonds for a Newman projection. The criterion is that at least one peripheral atoms is bonded to either of the two central atoms. **projax** lines may be supplied to define the two central atoms. A projection perpendicular to two arbitrary, not necessarily bonded, atoms can be requested in this way.

4.40.2. Plot Information

Atom positions, related to the original atom positions by internal symmetry, may be used in the Newman plots. They must be defined in **bondin** lines. Note that once one **bondin** line is used, the other bonds are not derived; all bonds must be defined with **bondin** lines. If **projax** lines are used, the same definitions for the internal symmetry must be given.

A Newman diagram is always printed. A command file for the program *PLOTX* can be written to `NWN`. Each plot can contain 1 or 4 diagrams. Plot dimensions as well as heights of characters and numbers can be specified either globally in the **NEWMAN** line or in **plotp** lines. A title is plotted at the top of the plot, defaulting to the compound identifier, but the user may specify his own title in a **ptitle** line.

4.40.3. File Assignments

- Optionally reads atom data from the input archive bdf
- Optionally writes plot commands to the file `nwm`

4.40.4. Examples

```
NEWMAN
projax C1 C2
```

A Newman projection along the C1-C2 bond is printed. Data are from the bdf.

```
NEWMAN plot 4
bondin C1 C2 C6 C10 C11
bondin C2 C1 C3 H1 H2
bondin C3 C2 C4
bondin C4 C3 C5
plotp *7 .30
ptitle just a test
```

Four diagrams in one plot are written to the command file. Only the specified bonds are placed in the bonding matrix. The heights of the characters in the title is 0.30 .

4.40.5. References

- Brandenburg, N.P. 1974. *J. Appl. Cryst.* 7, 410.
- Schenk, H., Brandenburg, N.P., van Sante, B., Kragten, E.Y. and Loopstra, B.O. 1985.

4.41. *ORTEP*: Molecular plot

Authors: George Davenport, Syd Hall and Wolfgang Dreissig

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

ORTEP produces, when used in conjunction with [PLOTX](#), crystal structure illustrations suitable for publication. This is an improved fversion of the ORTEP2 program (Johnson, 1970) with a wide range of automatic options.

4.41.1. Modes Of Operation

ORTEP provides for a wide range of calculations involving atomic coordinates and thermal parameters. Its basic function is to generate information about how a crystal structure may be viewed in terms of its constituent atoms and their thermal motion. Atoms with anisotropic displacement parameters may be viewed as displacement ellipsoids. The task of defining how the crystal structure is to be viewed and how the atoms are connected together has been simplified considerably over the original version of ORTEP (Johnson, 1970).

In this version of ORTEP three basic processes are available for setting up the sequence of instructions needed to generate the desired view of atoms, molecules or cell. The first is an *automatic mode* in which a few input options are used to connect all of the needed atoms and to define the viewing information. The second is the *manual mode* where all aspects of the plotted atoms are specified in detail by the user. The third is a *combination of the automatic and manual modes* where the user selectively modifies generated instructions.

4.41.2. Automatic Mode

In automatic mode, the user inputs a small number of parameters on the optional control lines. Based on these parameters, ORTEP selects a sequence of instructions from a fixed subset of those instructions available in the manual mode. This procedure can be used in most cases to achieve the desired results. For more complex or non-standard plots, the sequence can be modified by the user with any of the instructions available in the automatic mode, thus making the automatic mode as powerful as the manual mode. Note that ORTEP now has access to the atomic radii values on the archive bdf and if the option `arad` is set (it is the default if no VSC line are entered) then these values are used to determine if bonds are to be drawn between atom sites. If the option `vrad` is set or VSC lines are entered then bonding distances are determined by the max and min distances on these lines.

Automatically generated plots are specified on the **ORTEP** line as follows:

acta- **Impose Acta C style on ellipsoid plot**

Plot molecule with shaded ellipsoids (type 6) and filled bonds (type +5) with atom labels afixed (as per "`geninsymb`").

inpu- **Plot the input atoms**

Plot the atom coordinates as entered from bdf without symmetry transformations (unless independently specified by `inst` lines).

mole- **Plot connected molecule (default)**

Plot the atom coordinates that have been transformed to form the closest connected cluster. Note that the connectivity is applied with the 407 instruction using either atomic radii (if option `arad` is in force) or specified vsc distances (if option `vrad` is set).

cell- Plot sites with a unit cell

Plot the atom coordinates in which a least one part of the connected molecule is inside the boundaries define by one unit cell. The cell outline is drawn. If the `full` option is specified complete molecules will be plotted (default). If the `mcut` is entered molecules will be cut at the cell boundary.

coord n- Plot sites connected to atom n

Plot the atom coordinates where the atom is connected the atom with the input sequence number 'n'.

symm- Plot input sites with symmetry applied

Plot the input atom coordinates generated by the application of the symmetry equivalent positions specified by the `symop` line(s).

csym- Plot symm sites plus a cell outline

Plot the same coordinates as in the `symm` option plus a cell outline. Used in association with the `symop` line(s).

cmol- Plot mole sites plus a cell outline

Plot the same coordinates as in the `mole` option plus a cell outline. Note the possible use of `molorg` line(s).

4.41.3. Semi-automatic Mode

The automatic use of ORTEP involves the modification of automatically generated instructions. When modifying the instruction sequence, it is necessary to make a preliminary run in order to obtain the sequential numbering of the instructions. Then, replacements, insertions, and deletions may be made as follows:

Insertions before automatic instruction 'n':

```
seq precede n      : insert the following instructions
inst ...          : instruction to be inserted
inst ...          : instruction to be inserted
```

All instructions between the `seq` line and the next non-`inst` line will be inserted before the nth instruction in the unmodified instruction list.

Replace automatic instruction 'm':

```
seq replace m
inst ...        : instruction to be used instead
inst ...        : instruction to be used instead
```

The mth instruction in the unmodified instruction list will be replaced by all instructions between the `seq` line and the next non-`inst` line.

Deletions:

```
seq replace n
```

The nth instruction in the unmodified instruction list will be replaced by nothing, i.e., it will be deleted.

4.41.4. Manual Mode

For users with special requirements, the manual mode offers more flexibility. Each instruction is input by the user.

A description of each manual instruction is provided in the User Guide. It is not possible to provide a detailed description of each instructions in this writeup and if this necessary the user is referred to the original ORTEP writeup (Johnson, 1970). It will suffice here to summarize the function of each instruction category. They are:

Instruction *Function*

101-102	print distances, angles, and thermal eigenvectors
103	print thermal eigenvectors and direction cosines
105-106	same as 101-102 but with convoluting sphere of enclosure
201-202	define start plot, advance plot, end plot
301-303	specify plot dimensions, title rotation, retrace
401-407	define atoms to loaded into the working list for later plotting
411-417	define atoms to deleted from the working list
501-503	define plot orientation: axes, rotations, projection axis translation
511	implement overlap (hidden line) correction
601-604	define plot position and scaling: set origin, set scale
611-614	define plot incremental position and scaling
701-715	specify thermal ellipsoid parameters, draw atom symbols.
801-813	bond plotting parameters
901-916	plot labels and titles

Saved sequences

In some situations, for example plotting a stereoscopic pair, it is necessary to repeat a long series of instructions. By using the saved sequence feature, the long series of instructions needs to be input once only. To store the saved sequence use the following lines:

```
svstar seq1      : start of saved sequence called seq1
inst .....      : the instructions which make up the
inst .....      : saved sequence seq1
svend           : end of the saved sequence seq1
```

To invoke the saved sequence use a control line with the name of the saved sequence use the following line:

```
svexec seq1
```

The saved sequence feature is available only for the manual mode.

4.41.5. Terminology And Definitions

ANC - Atom Number Code

An atom number code is the sequence number of an atom in the list of reference atoms.

ANR - Atom Number Run

An atom number run is a series of atoms contained in the list of reference atoms. The ANR is specified by designating the atom number codes of the first and last atoms in the run (i.e. ANC1 - ANC2).

ADC - Atom Designator Code

An atom designator code specifies an atom position. It has three components; the atom number, the translation number, and the symmetry equivalent position number. The translation number is a three digit number, where each digit corresponds to a lattice translation along each cell direction with respect to the origin as 555.

ADR - Atom Designator Run

An atom designator run is specified by two atom designator codes. Let ADC1 have components A1, T1, S1 and ADC2 have components A2, T2, S2. The ADR then consists of all atom positions which have $A1 \leq \text{atom number} \leq A2$, $T1 \leq \text{translation number} \leq T2$, and $S1 \leq \text{symmetry number} \leq S2$.

VSC - Vector Search Codes

Vector search codes consist mainly of two atom number runs (origin ANR and target ANR) and a distance range. VSC's place constraints on the search for interatomic vectors. In contrast to previous versions of ORTEP all VSC's are input before any instructions are read in this version. The instructions which use VSC's are 101, 102, 402, 405-407, 412, 415, 416, 801-803, and 811-813.

Height-colouring

It is possible to use the height of each atom above or below the plane of the drawing to determine its colour. This height may be found in the atom listing produced by "**genins list**" as the z Cartesian coordinate in inches. The limits for the height classification are introduced on a "**height**" line. An atom will be assigned to a colour of 1 plus the index of the largest height limit smaller than the z coordinate of the atom. Thus if the height limits are arranged in ascending order, and an atom has a z coordinate larger than limit M but smaller than limit M+1, the atom will be assigned colour M+1. If N limits are given (max 8), limits 0 and N+1 always have values of -Infinity and +Infinity respectively.

4.41.6. File Assignments

- Reads *lrcell*: and symmetry data from the archive file
- Writes a plot command file on `ort`
- Writes a PS300 plot file to `or3`

4.41.7. Examples

```
ORTEP mole acta
```

This is an Acta C style ellipsoid plot of a single molecule. Overlap will be applied. Hydrogens will be treated as spheres of fixed radius. There will be no perspective.

```
ORTEP cell pers
```

As above, except that the contents of the unit cell will be generated with a perspective view.

```
COMPID ICEANE
ORTEP molecule sphere over      :define type of plot
genins cbla list                :calculate bond lengths
plotp 11 11 24 1 ab *7 70 20   :plot dimensions and axes
ellips 6 1.00                   :type of ellipsoid to be plotted
```

Drawing of the molecule iceane (Hamon, et al., 1977). One iceane molecule is to be plotted. Bond lengths and angles are to be calculated up to a distance of 2.0 Angstroms. A full listing of atom and bond search output is to be printed. Plot dimensions are 11 by 11 inches with a 1 inch margin. The view distance is 24 inches. The A and B axes are to be taken as the X and Y axes, respectively, of the plot plane. The plot is to be rotated 70° about the X axis and 20° about the Y axis. The ellipsoids are to be type 6 (with octant shading) with probability scale 1.00 (i.e. probability that the ellipsoid encloses the atom is 20 percent).

```

title AKERMANNITE - 1.85
ORTEP manu inpu
radius or .01
vsc 1 1 1 1 1 .10 1.00 .005
vsc 1 1 2 6 1 .10 2.00 .03
vsc 3 3 1 6 3 .10 1.85 .03
symbol
svstar ak
inst 511 0
inst 702
inst 801 1 555 1 1 565 1 1 555 1 1 556 1 1 555 1 1 655 1
inst 801 1 666 1 1 665 1 1 666 1 1 656 1 1 666 1 1 566 1
inst 801 1 655 1 1 656 1 1 655 1 1 665 1 1 656 1 1 556 1
inst 801 1 565 1 1 665 1 1 565 1 1 566 1 1 556 1 1 566 1
inst 802
inst 902 *8 0 0 .25 3 -5 .5 1
svend
inst 201
inst 301 12 12 30 .5
inst 401 1 555 1 -1 666 1
inst 401 1 555 3 -1 556 3
inst 401 2 555 1 -2 565 1
inst 401 2 565 2 -2 665 2
inst 401 2 655 2
inst 401 2 546 3 -2 556 3
inst 401 2 456 4 -2 556 4
inst 401 2 566 4
inst 401 3 554 1 -3 655 1
inst 401 3 564 2 -3 665 2
inst 401 3 546 3 -3 557 3
inst 401 3 556 4 -3 567 4
inst 402 1 555 1 1 666 1 4 6 2.0
inst 402 1 555 3 1 556 3 4 6 2.0
inst 402 3 554 1 3 655 1 4 6 1.85
inst 402 3 564 2 3 665 2 4 6 1.85
inst 402 3 546 3 3 557 3 4 6 1.85
inst 402 3 556 4 3 567 4 4 6 1.85
inst 501 7 555 1 1 555 1 1 565 1 1 555 1 1 655 1 1
inst 502 1 -75 2 5
inst 601 6 6 .85 1.54      inst 503 2 2.7
svexec ak
inst 202 15
inst 503 2 -2.7
svexec ak
inst 203

```

This is an example of an essentially manual run. The **vsc** lines set the bonds to be drawn in the 800 series instruction. Notice that the .005 in the first **vsc** line is a value which causes a narrow line to be plotted for the lines connecting the atoms given in the 801 instructions. In this case, this line outlines the unit cell. In the **symbol** line, the name is placed to the right to cause it to be centred on the plot. **svstar** marks the start of the instructions that are to be saved for use. This placement of the 511 through 902 instructions is distinctly different from that order used for Johnson's version of ORTEP. The **svend** line marks the end of the saved

instructions. The next `inst` lines are executed as encountered. The `svexec` line causes the saved (name `ak`) instructions to be carried out. The reason they are done twice is to produce the stereo pair. If there was no stereo plot, the 511-902 would not need to be saved and could have been placed in order.

4.41.8. References

- Johnson, C.K., Guerdon, J.F., Richard, P., Whitlow, S., and Hall, S.R. 1972. *ORTEP. The X-RAY system of Crystallographic Programs*. TR-192, 283p.
- Johnson, C.K. 1970. *ORTEP: A FORTRAN Thermal-Ellipsoid Plot Program for Crystal Structure Illustrations.*, Report ORNL-3794, 135 pages.
- Johnson, C.K. 1971. *Supplementary Instructions for the Oak Ridge Thermal Ellipsoid Plot Program ORTEP-II.*, Oak Ridge National Laboratory., ORNL-3794 supp., 10 pages.

4.42. *OUTSRC*: Export archive for external software NEW

Authors: Doug du Boulay, Materials and Structures Laboratory, Tokyo Institute of Technology, Nagatsuta, Midori-Ku, Yokohama Japan.

OUTSRC provides an interface from Xtal to other external crystallographic software programs such as XD, SHELX and VALRAY. An instruction file and associated reflection file is created, if [lrrefl](#): exists on the archive bdf.

4.42.1. Purpose

Often it is quite useful to trial different structural solution programs when Xtal methods aren't successful. At other times, it could be useful to streamline the route to electronic multipole or anharmonic atomic displacement refinement programs, if the data warrant it. *OUTSRC* was intended to facilitate such migrations to some extent.

It hasn't been tested so thoroughly, and at least for VALRAY input it is rather incomplete.

4.42.2. File Assignments

- Reads xtal archive file
- Writes hkl data to a `hkl` file
- Optionally writes a SHELX `.ins` file
- Optionally writes a XD `.mas` file
- Optionally writes a VALRAY `.dat` file

4.42.3. Examples

```
OUTSRC shelx ../mystruct
: create ../mystruct.ins and ../mystruct.hkl
: comments and options after the filename will be treated as part of the filename.
```

4.43. *PARTN*: Hirshfeld partitioning

Author: Nick Spadaccini, Computer Science Department, University of Western Australia, Nedlands 6907, Australia

PARTN partitions an input density into either the contribution of a particular atom (**fragment**) or of an atom and its symmetry equivalents (**entire**). The Hirshfeld method of partitioning is employed.

4.43.1. Hirshfeld Partitioning

The Hirshfeld (1977) method apportions the electron density among the atoms by the appropriate weighting. The weights are related by the atomic contribution to the promolecular density

$$w_A(\mathbf{r}) = \frac{\rho_{\text{atm}}^A(\mathbf{r})}{\rho_{\text{pro}}(\mathbf{r})}$$

The fragment of the density apportioned to atom A is

$$\rho_{\text{frag}}^A(\mathbf{r}) = w_A(\mathbf{r}) \rho_{\text{mol}}(\mathbf{r})$$

An alternative scheme is based on the atomic contributions to the total promolecular potential V_{pro} defined as the sum of the electronic and nuclear contributions.

4.43.2. Density And Potential Profiles

The promolecular density or potential is the sum of the atomic densities or potentials. These latter values are derived from the Clementi and Roetti (1974) atomic wavefunctions. Associated with each atom type are the parameters A_k , n_k and z_k for $k=1, \dots, m$ such that the density is

$$\rho_{\text{atm}}(\mathbf{r}) = \sum_{k=1}^m A_k r^{n_k} e^{-z_k r}$$

and the potential is

$$V_{\text{atom}}(r) = 4\pi \sum_{k=1}^m A_k \left(\frac{\gamma(n_k+3, z_k r)}{r z_k^{n_k+3}} + \frac{(n+1)!}{z_k^{n_k+2}} - \frac{\gamma(n_k+2, z_k r)}{z_k^{n_k+2}} \right) + \frac{Z_A}{r}$$

the last term is the nuclear contribution and $\gamma(n,x)$ is the Incomplete Gamma Function.

The density profiles (e/bohr³) and potential profiles (e/bohr) are stored at 44 discrete values of r (bohrs) for the points,

$$r_1 = 0. ; r_{k+1} = 1.15(r_k + .01) ; 0 \leq r \leq 31.2$$

The divisions are chosen so that the density of points is greatest in the region of steepest gradient. The density or potential value at any general point is linearly interpolated from the profile.

4.43.3. Program Requirements

The structural data is read from the input archive `bdf`. The density must be input on file `map`. PARTN partitions this density into the atomic contributions. The density ascribed to the fragment atom or the entire atom specified on the `site` line is output to file `par`.

4.43.4. Contribution Distances

The user may specify the effective range of atom contributions in two ways. The `border` option in the program initiation line determines the region beyond the input map for which atom contributions are included. The default value of 6Å implies that any atom within this distance of the input map edges is included. The distance beyond which an atom contribution is zero may also be set by the `contact` option.

4.43.5. File Assignments

- Reads atom data from the input archive `bdf`
- Reads density map from the file `map`
- Reads `profdb` data base from a file defined by the macro *profile*:
- Writes partitioned density map to the file `par`

4.43.6. Examples

```
PARTN
SITE O12 .1234 -.1234 .7890
```

This example employs all the default settings. The weights are determined from the promolecular density. The contributions of atoms up to 6Å beyond the input map edges are included in the calculation. However, an atom will not contribute to the density if the point is greater than 6Å away. The partitioned density will include the contribution of the `O12` atom at (.1234,-.1234,.7890) only.

```
PARTN poten bord 4 cont 4.5 entire
SITE Culab
```

Here the weights are determined from the promolecular potential. Atoms up to 4Å from the input map edges are included in the calculation. The effective range of an atom is set to 4.5Å. The partition density will include the contributions of all the `CuLab` atoms in the input range.

4.43.7. References

- Clementi, E. & Roetti, C. 1974. *Atomic Data and Nuclear Data Tables*. **14**, 177-478.
- Hirshfeld, F. 1977. *Israel Journal of Chemistry*. **16**, 198-201.

4.44. *PATSEE*: Search for molecular fragment

Authors: Ernst Egert and Syd Hall

Contact: Ernst Egert, Institut für Organische Chemie, Universität Frankfurt, Niederurseler Hang, D-6000 Frankfurt am Main, FRG.

PATSEE searches for a fragment of known geometry in the unit cell using an integrated Patterson and direct methods procedure. This program, which is valid and efficient for all space groups, is based on the standalone program written by Ernst Egert for the SHELX system. The rotation search is applicable to a fragment of any size and allows one torsional degree of freedom. The translation search may locate up to two independent search models of any size (including single atoms), taking into account known atoms at fixed positions, if any. The principles of this method are detailed by Egert & Sheldrick (1985), Acta Cryst. A41, 262-268.

4.44.1. Introduction

The choice of strategy for the solution of a crystal structure at atomic resolution is usually determined by the presence or absence of heavy atoms. Thus it is common practice to solve light-atom structures with direct methods and those containing heavy atoms with Patterson techniques. If this strategy fails, it may be advisable to resort to the corresponding alternative method; direct methods may well reveal the positions of heavy atoms, and the Patterson function can be interpreted even for purely light-atom structures, such as those of organic molecules, provided that part of the molecular geometry is known. This so-called Patterson search has been shown by various authors to be a powerful tool for solving difficult crystal structures; its great strength is that it employs chemical information directly, and so can compensate for mediocre precision and resolution of the X-ray data. *PATSEE* combines the merits of both Patterson and direct methods - in a manner that is generally applicable, efficient, automatic and easy to use - and thus to exploit all the *a priori* available information in order to solve large problem structures.

4.44.2. Preparation of the Search

Generally, a Patterson search in vector space consists of the following stages: (1) definition of a search model; (2) calculation and storage of the Patterson function; (3) rotation search, and (4) translation search. It is a serial technique, with the last two stages crucially dependent on the accuracy of the preceding ones. Thus the first step is by no means trivial; this is especially true for a procedure such as this where the fragments are taken as rigid and no model refinement is attempted (with the exception of one torsional degree of freedom between rigid groups). Usually a small well-defined search model is more appropriate than a larger one containing several incorrect atoms. The model is defined by atomic coordinates in a given coordinate system; these will normally be either fractional (taken from a related crystal structure) or Cartesian (e.g. from a force-field calculation).

The triplet structure invariant relationships which are required for the translation search (if applied) are calculated prior to the PATSEE run using *GENSIN* (which gets its E-values from *GENEV*). The **lac1.dat** example given below shows a typical input sequence for the *GENEV*, *GENSIN*, *FOURRF*, PATSEE calculations. Note that **lac1** is one of the Xtal test decks so that reference to the **lac1** listing will be useful in guiding initial PATSEE applications.

The Patterson map is generated by the program *FOURR*. For almost all purposes, we recommend using E.F as the coefficient (**epat full**); these lead to a sharper map than F^2 (**patt full**) but generate fewer ripples than E^2-1 (**vect full**).

The region around the origin of the Patterson function is dominated by *intra*-molecular vectors, which depend on the orientation but *not* on the position of the fragment. Thus the full six-dimensional search can be split into two three-dimensional searches, a *rotation* and a *translation* search (depending on the space group, the latter may be of even lower dimensionality).

The atom fragment information used in the searches is entered in the following way. Atom site coordinates may be entered as fractional or Cartesian according to the option on the preceding **FRAG** line. Each group of atom sites must be preceded by a **FRAG** line, or in the case of sites loaded from the bdf, the **FRAG** line(s) contain the labels of the atom sites to be used in the search. The position of each fragment of the atom sites in the input stream determines how they are used in PATSEE. A fragment that precedes the **rotate** line will be *fixed* (i.e. the vectors between these atoms will be neither rotated nor translated but they will be used in the figure of merit calculations). Fragment(s) that follow the **rotate** line but precede the **transl** line will be *rotated and translated*. Fragments that follow the **transl** line will be *translated only*. The **shift**, **spin** and **twist** lines are used to modify the atom sites in a fragment. If a **shift** or a **spin** line will transform or rotate, respectively, the atom sites of the *next* fragment. The **twist** line serves a different function -- it enables two parts of a fragment to be rotated about a connecting bond (and searches are applied for each twist settings). The **twist** line must be positioned in the input stream between the two atom sites which will be twisted with respect to each other.

4.44.3. ROTATE - Rotation Search

The first step in the rotation search is to set up the *intra*-molecular vector set to be used for the search, i.e. to express the model geometry (which should always be checked thoroughly - see the **geom** option on the **PATSEE** line) in the form of discrete vectors with associated weights. Of the $N(N-1)/2$ intramolecular vectors, the short (e.g. $d < p \text{ \AA}$, see **vlim**) and long (e.g. $d > q \text{ \AA}$, see **vlim**) values are immediately eliminated. Since the inner sphere around the Patterson origin shows some vector density everywhere, the short vectors provide little angular discrimination and are normally not very useful for determining the orientation of the fragment. However, they may be important for molecules (e.g. those consisting of fused aromatic rings) that are characterized by a few short vectors with high weights. An upper limit for the vector length is also advisable because very long vectors, though quite characteristic of the search model, suffer most from uncertainties in the geometry and could easily miss the corresponding maximum in the Patterson map. Close vectors (see **vres**) are replaced by a weighted average vector with the combined weight. In order to save computing time, low-weight vectors may be omitted from the figure of merit assessment (see **vform**).

Any orientation of a rigid fragment relative to a fixed coordinate system can be described by three angles corresponding to successive rotations about properly chosen axes. (There are various definitions of the Eulerian angles. For computational reasons, we prefer successive rotations about the *a*, *b* and *c* axes, in that order.) The asymmetric unit of angular space depends on both the Laue group and the model symmetry. Instead of scanning the respective range of angles by specifying rotation increments, we have chosen to generate random orientations (see **nran**). The optimum number of orientations (see **ntry**) to be tried depends on the size and the shape of the search fragment, the Laue group and the Patterson grid intervals. PATSEE usually generates 10000-600000 angle triplets, which corresponds to mean rotation increments of about 7° ; this is normally sufficient for the coarse location of the maxima.

For each orientation, the correlation between the rotated intramolecular vector set and the Patterson function is measured by a *product* function (note that this is a different approach to non-Xtal versions of PATSEE). The weight of each vector \mathbf{w}_i is thus multiplied with the nearest Patterson grid value \mathbf{p}_i . The rotation figure of merit (*Rfom*) is $\sum \mathbf{w}_i \mathbf{p}_i$ for a specified sample (see fraction \mathbf{f} of \mathbf{vfofom}) of the largest weighted vectors. Note that these vectors are sorted into a test list in the order of descending $w_i d_i$ values, where d_i is the vector length. This is important because a pretest requires that the top \mathbf{t} (see \mathbf{vtes}) of the vector list has a *Rfom* which is positive or the orientation is immediately rejected. The sorting is necessary to avoid the dominance of this test by short heavily weighted vectors. If the final normalized *Rfom* is $< \mathbf{p}$ (see \mathbf{fomt}) the orientation is also rejected.

Before an orientation is placed in the short list of best solutions, it must pass two tests. The 'overlap test' ensure that no close interatomic contacts arise from the application of the lattice translations present and the 'equivalence test' compares the orientation in question with those already stored. Two orientations are regarded as similar when all pairs of equivalent atoms are close to each other; in that case only the better one is kept.

In order to improve the performance of the subsequent translation search, the best solutions are 'refined' by a restricted and finer rotation search. The maximum within each promising region of angular space is found by testing up to \mathbf{n} (see \mathbf{nref}) additional random points, which corresponds to a mean rotation increment of less than 2° .

Users may also specify the starting orientation of a fragment (see \mathbf{seta}) and the range or the random rotations (see \mathbf{setr}). This is normally only used, however, if a rotation search is repeated over a target set of orientations.

If the search model has one torsional degree of freedom the rotation searches are repeated for each distinct geometry using the \mathbf{twist} option. This specified by a range of possible torsion angles and an appropriate increment. Invocation of \mathbf{twist} causes a merged list of best solutions is set up. At the end of the rotation search, a small number of promising orientations are passed over to the translation search. It is our experience that the correct one is usually present among the best two or three for reasonably sized fragments.

4.44.4. TRANSL - Translation Search

In procedures to position a fragment of known geometry in the unit cell, the translation search has usually proved to be less reliable than the rotation search. This is because the 'cross' (i.e. *inter*-molecular) vectors used to locate a fragment with respect to the origin suffer from errors in both the model geometry and orientation amplified by the symmetry elements; in addition, model vectors with very high weight are less likely than in the rotation search.

The phases calculated from the coordinates of an oriented model are a continuous function of the shift vector $\Delta\mathbf{r}$. When the fragment is moved through the unit cell keeping its orientation fixed:

$$\mathbf{F}_h = \mathbf{F}_{h_0} \cdot \exp 2 \pi \mathbf{h} \Delta\mathbf{r}$$

since all atomic displacements $\Delta\mathbf{r}$ are the same. So the scattering contributions from the atoms of the search model have to be summed only once for each orientation and reflection to yield a structure factor \mathbf{F}_{h_0} for the starting position; subsequently, the structure factor \mathbf{F}_h for any position is readily obtained by multiplication with a simple phase factor. For the true structure, the individual phases of the strongest reflections are linked by various statistical phase relations; amongst these, the three-phase structure invariants have proved to be especially useful. The search fragment is usually incomplete and may also be not very accurate. Nevertheless, if its scattering power is significant, the triple-phase relations should hold at least approximately for the correct solution, in the sense that the distribution of the phase sums is far from being random.

These considerations led us to the development of a novel strategy for a Patterson translation search, which exploits in an integrated fashion the information contained in the sharpened Patterson function, the three-phase structure invariants and allowed intermolecular distances. In short, we have chosen the optimization of a weighted sum of cosine invariants as our refinement procedure, with the Patterson correlation and R indices as additional figures of merit, and the minimum intermolecular distance as a possible rejection criterion. This method is computationally efficient, especially for larger structures, because the refinement is based on phase relations derived from a relatively small number of large E magnitudes (say, >1.8). Only when an

acceptable solution has been found by this 'direct search' is it necessary to calculate the time-consuming Patterson correlation.

Since, in order to save computing time, relatively few phase relations are employed for the refinement, they have to be selected carefully. Normally only the 40-60 most probable and translation-sensitive three-phase structure invariants are used for a translation search. It is advisable to apply a 2θ limit to the E values before searching for phase relations, since high-order reflections may be influenced considerably by errors in the model. However, if the cut-off is too severe, the accuracy of the phase-refinement procedure suffers. It seems that a nominal resolution of about 1 Å is the best compromise (`smax 0.5` in *GENEV*).

Then random positions are generated for the rotated search fragment(s); it is our experience that about one translation try per cubic Ångstrom is sufficient in order to have a good chance of locating one search model correctly (see `ntry`). Since the number of tries rises as a high power of the number of independent fragments, it is unreasonable to search for more than two fragments simultaneously. However, any number of fixed fragments (obtained from a previous search or a heavy-atom Patterson interpretation, for example) may be added and, in fact, are quite valuable provided their size or scattering power is large enough.

Taking the limited range of the subsequent refinement into account, only those random positions that are fairly close to physically reasonable solutions are worth refining; thus all positions that give rise to short inter-molecular distances (say $d < p \text{ Å}$) are immediately rejected (see `vmin`). The refinement procedure consists of two cycles during which the translation parameters are refined by optimizing *t3sum*, which measures the triple-phase consistency.

$$t3sum = \sum E_1 E_2 E_3 \cos(\phi_1 + \phi_2 + \phi_3)$$

t3sum is expected to be large and positive for the correct solution. At the end of the second cycle, only positions with $t3sum > f$ (see `tprt`) are regarded as possible solutions.

For solutions that have survived these tests, the *Tfom* value is calculated identically as for *Rfom* but now for the inter-molecular vectors. A small number of best solutions (according to both *t3sum* and *Tfom*) are stored provided that they pass various tests for possible equivalence (allowed origin shift or lattice translation). Although the true position of the search fragment is usually recognizable at this stage, R indices *Re1* and *Re2* based on E magnitudes have proved very useful in distinguishing further between correct and false solutions.

4.44.5. Combined Solutions

Finally, the solutions are sorted according to a combined figure of merit:

$$Cfom = (Rfom \cdot Tfom \cdot t3sum^{1/2}) / (10 \cdot Re1 \cdot Re2)$$

For all solutions printed, a Patterson sum function is calculated as a measure of fit/misfit for each individual atom, taking all vectors (intra- and inter-molecular) into account, this enables identification of possible wrong atoms and thus model correction.

4.44.6. Summary

The procedure described differs from other Patterson translation functions in that the oriented model is placed with respect to all symmetry elements of the space group simultaneously. Tests with known structures have indicated that this routine is able to locate very large fragments (of more than 300 atoms), in which case the distance tests sometimes preclude the majority of trial positions, as well as single atoms even when the latter are not very heavy (e.g. phosphorus or sulphur in large organic structures). Above all, the variety of different criteria employed to judge solutions should make this combination of Patterson and direct methods a powerful structure-solving strategy, if chemical information is available. One would expect that a position that is in agreement simultaneously with packing criteria (`dmin`), the Patterson function (*Tfom*), triple-phase relations (*t3sum*) and E values (*Re*) is probably correct, and our experience shows that this is indeed the case.

4.44.7. File Assignments

- Reads symmetry & atom data from the input archive bdf
- Writes new atom sites to the output archive bdf
- Reads Patterson map from map file
- Reads E's and triplets from inv file
- Writes new atom sites to file pch

4.44.8. Example

```
compid lac1
GENEV smax 0.5 list 1.8           :calculate the E values
GENSIN                          :calculate the structure invariants
gener 1.8quar noprint 100 1 100
FOURR  epat full                 :calculate the E.F Patterson map
PATSEE
geom
rotate vfom .8
frag  9.946 29.966 11.189 90 90 90
setid site
C1      0.43394 0.51778 0.51693 :Diastereoisomer coordinates
C2      0.46733 0.54875 0.62187 :Acta Cryst,C39,95 (1983).
C3      0.48235 0.52338 0.73773
C4      0.57311 0.48221 0.72824
C5      0.54921 0.45318 0.61956
C6      0.65325 0.41691 0.60175
C7      0.63235 0.38980 0.51082
C8      0.50339 0.39796 0.44173
C9      0.49348 0.44807 0.39978
C10     0.53945 0.48047 0.50109
C11     0.56162 0.45583 0.27891
C12     0.51547 0.42331 0.18395
C13     0.54556 0.37553 0.22118
C14     0.46982 0.36608 0.34010
C15     0.48559 0.31558 0.35594
C16     0.47342 0.29772 0.22711
C17     0.48286 0.33807 0.14279
C18     0.69738 0.36838 0.23249
C19     0.67749 0.50191 0.47513
C20     0.55074 0.326640.02137
O1      0.54488 0.55020 0.82750
N1      0.41764 0.42882 0.62876
N2      0.39747 0.39645 0.53877
setid
transl
finish
```

This is the **lac1** test deck. It is the standard test for PATSEE. Use the **lac1.dat** listing as a guide for other applications of PATSEE.

4.45. *PEKPIK*: Search map for peaks

Authors: Syd Hall and Ruth Doherty

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

PEKPIK searches a Fourier map for a unique asymmetric set of the maxima of the density which is printed and/or written to a file.

4.45.1. Algorithm

The peak search algorithm involves a layer by layer search of the map. In the discussion which follows, reference is made to the form of the Fourier map as it would be output by the program FOURR. The first summation direction produces layers, referred to as pages. The second summation direction produces lines down the page, while the third summation direction produces the lines across the page (see *FOURR*). The (x,y) referred to here are not necessarily the crystallographic x,y. As each layer is searched, the value of the function at every grid point is compared to each adjacent grid point within the layer. When the value of the function at a particular grid point, e.g. (x,y), is found to be larger than at any of the grid points surrounding it, an interpolation may be done to determine the exact coordinates of the maximum within the layer and the maximum value of the function within the layer. If (x,y) lies at a corner of the layer (i.e. the first or last grid point on the first or last line of the layer) no interpolation is done and the maximum is taken to be on the grid point. If (x,y) is a point on the first or last line, but not at a corner, a one-dimensional interpolation is done in the third sum direction (across the page). If (x,y) is the first or last point on any line but the first or last line, a one-dimensional interpolation is done in the second sum direction (down the page). For any other point, a two-dimensional interpolation is done.

After a layer has been completely searched, the peaks from that layer are saved temporarily in the layer-to-layer storage table. As the next layer is being searched, each new peak is compared to those from the previous layer. If two peaks are within the resolution and if the peak is larger in the current layer, both the old and the new coordinates are saved. When a peak has been found in three successive layers, with the density going through a maximum in the second layer, an interpolation is done in the first sum direction (page to page) to find the most exact coordinates and density for the peak. No interpolation can be done in the first sum direction for peaks which occur within half a grid point of the first or last layer or which are not above the cutoff in three successive layers.

Before a peak is saved in the permanent list, a check is made to determine whether or not it is symmetrically related to any of the peaks which have already been saved. This insures that the final list will contain a unique set of peaks. If the map is a Patterson map, the Laue symmetry, rather than the space group symmetry, is used.

If a new peak is found to be symmetrically related to a peak which has already been saved, only one of the two will be saved. The more reliable of the two peaks will be saved. The reliability of a peak depends on its position in the map. The most reliable peaks are those at interior points of the map. Their high reliability is a result of the three-dimensional interpolation. Less reliable are peaks on a face of the map (two-dimensional interpolation), followed by peaks on an edge (one-dimensional interpolation). The least reliable peaks are those at corners of the map. For these, no interpolation can be done.

After the map has been searched completely, the coordinates of any atoms that may be present in the bdf are compared with the coordinates of all the peaks which have been saved. A list of the atoms which do not coincide with any peaks is produced. Finally, the highest peaks are listed in order of decreasing density. Each peak is labeled with either the name of the atom which occupies the same position or, if it coincides with no atom, a sequence number. The peaks are listed and, optionally, output in `atom` line images on the file `pek` and the punch file. The number of highest peaks listed may be specified by the user. If a peak has been identified with an atom name, that name will appear in the first field of the atom line. Otherwise, the atom name will be `c` followed by the sequence number assigned to the peak.

The program may also be used to search for holes. When this option is chosen the map is read in with every point being changed from plus to minus. This means that the peaks listed at the end of the run are actually holes in the entering map. It is necessary to do two separate runs to find both peaks and holes in a map.

The three fields on the `plimit` line are as follows:

1. **Minimum point density used in peak search:** Reasonable default values are assumed according to the map type. This cutoff is checked after the first layer. If more than one third of all grid points in the first layer have density values exceeding the cutoff, the cutoff is incremented by one quarter of the average density in that layer. The adjusted value is printed.
2. **Resolution (in Angstroms):** This value sets the separation of the maxima in the map which will establish two discrete peaks. This parameter should be set to no less than $\lambda/(2\sin\theta_{\max})$. For example, if all the reflections within the copper sphere of reflection were measured, the resolution might be set to 0.75 Angstroms.
3. **Total number of unique peaks to be saved, sorted, and listed:** The coordinates and value of the function at the maximum will be stored on file `pek` in order of decreasing peak density. If the number of peaks specified is too large for the available memory, the number of peaks will be adjusted to accomodate as many peaks as possible for the memory available. The default value for this parameter is twice the number of non-hydrogen atoms in the asymmetric unit.

4.45.2. File Assignments

- Reads symmetry and atom data from input archive `bdf`
- Reads map density from specified `bdf` (default is `map`)
- Writes sorted peak sites to `bdf pek`
- Optionally, writes peak sites to line file `pch`

4.45.3. Examples

```
PEKPIK
plimit *3 20
```

This is a typical run to analyze the first attempt at extracting an image of the molecule from an electron density map. The number of top peaks to be listed has been set somewhat higher than the number of non-hydrogen atoms in salicylic acid in order to allow for spurious peaks. The cutoff to be used in the search for peaks will be the value set in `FOURR` for the upper limit of print suppression.

```
PEKPIK bdf sla
```

Peaks here will be extracted from a Fourier map output from [SLANT](#).

4.46. *PIG*: Display and manipulate model

Authors: Ursula Bartsch, Syd Hall and Doug du Boulay

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

PIG is a Portable Interactive Graphics routine for displaying, selecting and manipulating atom and peak sites.

4.46.1. Description

PIG extracts atom data from the input archive `bdf` and/or the peaks data from the `pek` `bdf` or the sites from a `BUNYIP` `.bun` file. At the conclusion of a *PIG* session the results of the site manipulations are stored in the *lratom*: record of the output archive `bdf` and these may be used directly for subsequent calculations. A list of `atom` lines is also output to the file `add`. Note that atom labels are truncated to eight characters in *PIG*. *PIG* may be used to display both the molecule and cell. Sites may be rotated, deleted, added, relabelled and retyped. Bond distances, angles and torsion angles may be displayed. The orientation of the the displayed sites is stored on the output archive `bdf` for use by programs such as *ORTEP*. The saved orientation matrix is in *lrgraf*: of the output `bdf` and will be used by *ORTEP* as the default plot orientation, unless `npig` is entered (on the *ORTEP* line).

The molecular view can be rotated by LEFT-mouse-click-drag-release events. The magnification can be altered by MIDDLE-mouse-click-drag-release events (based on up/down translations) (NOTE that the MIDDLE mouse button is emulated by simultaneous LEFT and RIGHT mouse buttons on two button mice). The molecule can be translated by RIGHT-mouse-click-drag-release events.

PIG can be used as a WYSIWYG (what you see is what you get) interface to *POVRAY* for raytraced 3D images or *ORTEP* for traditional 2D thermal ellipsoid plots. Provided that a `bdf` is written at exit (containing the molecular view orientation), every site visible on screen at the time of exit will be displayed in the subsequent *ORTEP*/*PREVUE*. In addition, if labels are switched on, then the *ORTEP* will be labelled, and if the current view is a unit cell, then *ORTEP* plots the unit cell outline. Up to 15 *ORTEP* control lines may be specified in *PIG*, (prefaced with `ORTCTL`) and these are substituted for or added to the resultant *ORTEP* control file commands.

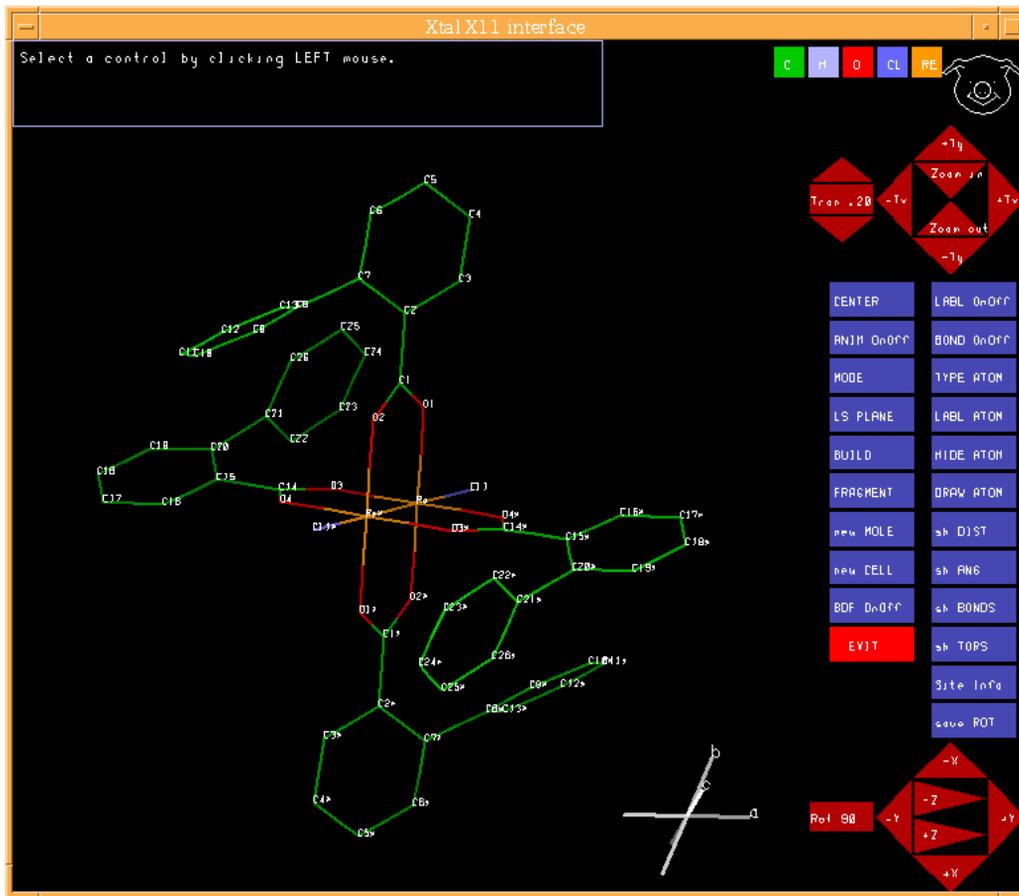
PIG can be used to generate a molecular information file for conversion of XTAL molecular data for use in other programs that do not yet read Crystallographic Information Files (CIFs). Two formats are provided. The Protein Data Bank (`.pdb`) format, containing space group, cell parameters, symmetry independent sites in cartesian coordinates, anisotropic temperature factors and bond connections. The second format is the Minnesota Supercomputer Centre XMOL `.xyz` file which contains the cartesian coordinates and atom types of all atoms on screen at *PIG* exit time (analogous to *ORTEP* exit mode). The `.xyz` file maybe useful as a means of generating symmetry related atom sites for use in other software packages such as *RASMOL*. If the `.xyz` file is relabelled as a `.spf` file it can also be used by *PLATON* (Ton Spek 1980).

More recently any H atoms built in *PIG*, iether automatically or manually (with the exception of `free sites` or peak assignments) add constraint linkage information to the archive file. This permits the recalculation of H atoms when invoking the CRILSQ riding model refinement. However, the H atom linkage and labelling is dependent on parent atom coordination. Any attempts to relabel parent atoms or H atoms after building will render the output linkage incorrect. In the event that a *riding model* refinement is subsequently attempted, it will be necessary for users to iether edit the `.add` file manually and include it as part of an *ADDATM* run, or rerun *ADDATM* in update mode and overwrite existing linkage lines on the archive with their own versions as described in *ADDATM*.

4.46.2. Interactive Controls

The display is controlled at two levels. (1) Input control lines are used to define the starting parameters for the atom and peak sites to be used, the atom radii, the display colours, the type of display (molecule or cell) and

Figure 4-2. Screen shot of PIG



so on. These values are applied prior to displaying the sites. (2) Control is via the interactive cursor and the menu buttons. *The user is guided by the instructions provided in the message panel.*

4.46.2.1. Rotation Control

Rotation control is provided by 6 buttons in the lower RH corner. The box immediately to the left of this enables the user to switch from course (90°) to fine (adjustable via up down arrows) rotations. These controls are in addition to using the left mouse button.

4.46.2.2. Zoom Control

Zoom and translation control is provided by 6 buttons in the upper RH corner, or alternatively by middle and right mouse button click-drag events.

4.46.2.3. LABEL onoff

A toggle switch to display or not display the site labels.

4.46.2.4. BOND onoff

Selectively hide or display interatomic bond lines. In this mode bonds may be hidden or displayed by clicking on selected labels. The actual change to the display status of the line bond is not effected until the pig icon is clicked (top right hand corner). Note that site markers are NOT hidden with this command, only the connecting bonds (see also the **HIDE atoms** panel).

4.46.2.5. TYPE atoms

Clicking the cursor on this panel places PIG in type-atoms mode. In this mode the user must first select which atom type by clicking on one of the atom type boxes displayed in the upper right hand area of the display. Then click on each of the sites (to be specific, on each site *label*) that is to be changed to the selected atom type. The atom type of selected sites will not be changed until the pig icon is clicked. Note that the site *labels* are *not* changed on the display, only the *colours* of the bonds. However, PIG will automatically attach the new atom type code to the atom label when it is output to the **pch** file and the archive bdf. To change the displayed site label, see the **LABL atom** command described below.

4.46.2.6. LABL atom

Clicking the cursor on this panel activates the label-atom mode. The user must click on the atom label to be changed and then enter the new label on the keyboard. Duplicate labels will be rejected when exiting from this mode. When labeling is complete click on the pig's nose to exit.

4.46.2.7. HIDE atoms

Clicking the cursor on this panel places PIG in the *hide atoms mode*. In this mode the user may select any number of sites (by clicking on the appropriate site labels) which are to be removed from the display. The actual removal of site markers, labels and attached bond lines does not take place until the pig icon is clicked.

4.46.2.8. DRAW atoms

Clicking the cursor on this panel will cause all hidden atom sites to be returned to the display. It reverses the **HIDE atoms** effect.

4.46.2.9. show DISTANCE

Clicking the cursor on this panel places PIG in the *show distance mode*. A distance will be displayed for two selected sites (by clicking on the appropriate site labels). These sites need not be bonded (this may be used on a 'cell' display as well as a 'molecule' display). This mode is exited by clicking on the PIG icon.

4.46.2.10. show ANGLE

Clicking the cursor on this panel places PIG in the *show angle mode*. An angle and two connecting distances will be displayed for three selected sites (by clicking on the appropriate site labels). The second selected site is assumed to be the central atom. These sites need not be bonded (this may be used on a 'cell' display as well as a 'molecule' display). This mode is exited by clicking on the PIG icon.

4.46.2.11. show BOND DISTANCES

Clicking the cursor on this panel places PIG in the *show bond distances mode*. All bond distances connected to a site will be displayed by clicking on the site label. This mode is exited by clicking on the PIG icon.

4.46.2.12. show TORSION ANGLE

Clicking the cursor on this panel places PIG in the *show torsion angle mode*. The torsion angle and distances will be displayed for the four selected sites (by clicking on the appropriate site labels). These sites need not be bonded (this may be used on a 'cell' display as well as a 'molecule' display). This mode is exited by clicking on the PIG icon.

4.46.2.13. show INFO

Using this mode the user can examine the fractional coordinates of all sites currently displayed, as well as determining the ORTEP/BONDLA symmetry and translation operations required to generate dependent sites, and the peak heights of peaks displayed from PEKPIK data.

4.46.2.14. save ROTATION matrix

Clicking the cursor on this panel causes the currently displayed orientation to be saved in a rotation matrix. On exiting PIG this matrix is stored in the output archive bdf and may be used by ORTEP to produce a plot with the same orientation. Note that doing a LS PLANE calculation will override the save operation.

4.46.2.15. ANIM OnOff

This control activates (by default) or deactivates the animated spin mode. With animation mode activated, using the left mouse button to rotate the molecule via a click-drag-release, imparts an angular momentum to the molecule, proportional to the length and direction of the effective drag displacement vector. To stop a freely spinning molecule simply click and release the mouse button without moving it. Reselect the ANIM control to cancel this mode.

4.46.2.16. CENTRE

The CENTRE control permits re-centring of the molecule, either about the collective centre of the current molecule, or, about a specific atom site by selecting its label.

4.46.2.17. RG STEREO

Red, green stereo mode. Technically this is an *anaglyph*. Two images of the current molecule are drawn, one of which is rotated by 3 degrees about the molecular centre. By wearing spectacles with one red and one green lens, a 3 dimensional effect may be apparent. Note that if the lens colors are reversed, a freely rotating molecule will appear to rotate backwards, which could be cause for confusion.

4.46.2.18. PEAKS

This control button is only present if (1) the **PEAKS** command line was invoked from the control file (2) a .pek file exists, or (3) PIG was invoked with the `msol` multi-solution command. It is a two state button, which switches on or off the display of peaks from a **PEKPIK** .pek file. However, when peaks are switched ON, a scrollbar facility allows the user to adjust the lower peak height threshold at which peaks will be displayed. This is useful if there are many insignificant electron density peaks obscuring the larger peaks in which you are interested. The **PEAKS** button must be reselected to confirm that the current threshold is what the user requires.

4.46.2.19. LS PLANE

Clicking the cursor on this panel places PIG in the *least squares plane mode*. The user must then select *at least* three sites (by clicking on the appropriate site labels) to define the least squares plane and then click on the pig icon. This will cause the plane to be calculated, and the atoms sites to be displayed with the plane perpendicular to the z axis. PIG remains in the least squares plane mode and the perpendicular distance of any site from this plane may be displayed by clicking on the site label. This mode is exited by clicking on the pig icon again. Note that the orientation of the plane is automatically stored in the rotation matrix, for subsequent storing on the output archive bdf.

4.46.2.20. FRAGMENT

The FRAGMENT mode has been enhanced over previous versions of PIG. There are now four FRAGMENT choices - `select`, `deselect`, `site connect` and `site reject`. The additional modes were designed primarily as tools for controlling display contents for generating ORTEPS and the .xyz file for export to other programs such as RASMOL.

The `select` and `deselect` modes enable all sites within a LEFT mouse button click-drag-release event to be either exclusively retained or exclusively deleted from the currently displayed molecule. The selected region is highlighted by a rectangular border.

`site connect` and `site reject` modes permit the current molecule to be either extended by adding all bonded sites to a specific atom by selecting its atom label, or by deleting the selected label from the current fragment.

It is important to note that deselecting an atom site or group of atoms using the FRAGMENT modes is quite different from HIDEing an atom in HIDE mode. Using the HIDE button, hidden atoms are irretrievably lost at the next NEW MOLE or EXIT, with potentially disastrous loss of cell contents on the archive file. Whereas, by using the fragment delete modes, the atom sites are merely hidden from display. This is why the FRAGMENT mode is preferred as a means of specifying ORTEP plots.

To return to the full molecular display click on the **new MOLECULE** panel.

4.46.2.21. BUILD

Click on this panel when new sites need to be added to the atom display list. If new H atoms are built, on exit they are assigned Uiso values corresponding to 1.5 or 1.25 times the Uiso value of their bonded C/N/O parent (if present), dependent on whether it is a methyl group or otherwise, respectively. Because of this it is more generally useful to have refined the vibrational motion of the heavier atoms before building H atoms.

Five additional panels will be displayed in the message area. Clicking on one of these panels places PIG into a particular 'build atom site' mode. For each mode follow closely the instructions displayed in the message area. The modes are:

- **Linear** Requests a new site which is linearly aligned with two other sites in the structure. First select an atom type for the new site. Then click on the two sites aligned with the new site; the furthest site first and then the site to which the new site will be bonded. The new site will automatically be assigned a site label composed of the label of the bonded site with the character 'a' appended.
- **Trigonal** Requests a new site which is trigonally aligned with three other sites in the structure. First select an atom type for the new site. Then click on the three sites which form an apex to which the new site will be attached; the apex site is clicked second in the sequence. The new site will automatically be assigned a site label composed of the label of the bonded site with the character 'a' appended.
- **Tetrahedral** Requests one, two or three new sites which are tetrahedrally bonded with neighbouring sites in the structure. First select the atom type for the new site(s). If one or two sites are sought, click on the three sites which form an apex to which the new site(s) will be attached; the apex site is clicked second in the sequence. If three tetrahedrally related sites are sought (e.g. methyl hydrogens) click on the two sites which form the fourth member of the tetrahedral bonding; the furthest site first and then the site to which the new sites will be bonded. The display will be then aligned so that the bond of the two clicked sites will point directly at the viewer and be surrounded by a circle. The position of the three new sites around this bond is selected by clicking the position of the first at a point on this circle. All new site(s) will automatically be assigned a site label composed of the label of the bonded site with the characters 'a', 'b' or 'c' appended.
- **Free site** Requests a new site which is not geometrically related, but is bonded, to another site in the structure. First select an atom type for the new site. Click on the site to which the new site will be bonded. Then click on the screen to determine the orientation (relative x and y coordinate; z will be set at the value of the bonded site). Note that the display usually needs to be carefully oriented before this mode is used. The new

site will automatically be assigned a site label composed of the label of the bonded site with the character 'a' appended.

- **Auto H atom** In this mode PIG searches every independent C atom in its atom array and examines its connectivity, based on current radii values. If the C atom is underbonded its bondlengths and angles are assessed for linear, trigonal and tetrahedral H atom bonding. Where there are no ambiguities, these H atoms are automatically built and labelled according to their parent C atoms label. No H atoms are built on parent O atom or N atoms because the number (for N) and orientation (for O) of such H atoms are in general not well defined. In general, bond angles below 116 degrees are treated as tetrahedral, and C-C bond lengths below 1.46 Angstroms are considered double bonds.

4.46.2.22. RADII

Use the RADII button to set the atomic radii for bonds to be calculated in future. Formerly the radii button was subsumed with the **new MOLE** command, but it is now accessible from the main menu to assist with fragment extension. In **RADII** mode selecting an atom type box will display the current radius, and reselecting that atom type enables a new value to entered via the keyboard (closed with a return carriage).

4.46.2.23. HIDE TYPE

This control is used to temporarily hide specific atom types from view, such as hydrogen atoms in cluttered diagrams. When invoked, selecting an atom-type panel immediately switches the display state of all sites of that type on or off. Select PIG to cancel this mode. Atom types hidden in this manner are not lost and may be redrawn later by repeating the same procedure.

4.46.2.24. new MOLECULE

Clicking this panel causes the currently displayed sites (not the hidden sites) to be included in the calculation of a new connected molecule and its display. Hidden atoms (from Hide Atom) will be removed from the site list and will not be available for output to the archive bdf. Conditions for the connected molecule search are set by the PRESET line and by the three panel buttons that appear in the message area. These buttons control the following:

- **SYM-EXT on/off** - a toggle switch to include/exclude atom sites related by symmetry which are within the bonding distance. This switch should be on if a molecule contains a symmetry element but may need to be switched off when peaks are active.
- **Clusterng** - A three position switch to enable/disable the reclustering of sites, or revert to the input site list (losing all changes) when the molecule is redrawn. The cluster mechanism attempts to optimally connect all input sites to the first atom site in the input list. For very large molecules clustering is time consuming. If a molecule contains more than 500 atoms, clustering is automatically switched off.
- **Cluster Packing**- A four position switch governing the treatment of multiple disconnected clusters. The first mode (**Pack Off**) switches off any extra packing treatment. The three additional packing modes provide variations on the optimal packing of secondary clusters about the primary aggregate molecule. **Pack close** attempts to arrange clusters in order to minimize distance between the centres of gravity of all clusters. **Pack centre** attempts to minimize the distances between the largest cluster center of gravity and other clusters **Pack contact** attempts to find the shortest contact between clusters - to accentuate hydrogen bonding for example.

Clicking the New Molecule panel a second time will start the display process using the current display parameters.

4.46.2.25. new CELL

Clicking the cursor on this panel causes the currently displayed sites (not the hidden sites) to be included in the calculation of a new cell display. The **mcut** option allows for all molecules to be cut at the cell boundary of the first unit cell, but otherwise all sites within +/-1 cell along all three axes are calculated and added to the display view. Hidden atoms (using **HIDE ATOM**) will be removed from the site list and will not be available for output to the archive bdf, or the **add** file. Note that the current display limit is 10000 atoms. If nine unit cells result in more sites than the current limit, then PIG will terminate with an error message.

4.46.2.26. POV IMG

This mode provides the user with limited control over the creation of **compid.pov** files containing a three dimensional description of the current display for rendering using the freely available **Persistence of Vision Ray Tracer**(TM) (Pov-Ray). XTAL is in no way associated with POV-Ray(TM) and makes no warranties as to that programs reliability, usefulness, or suitability for any purpose whatsoever. Consequently no support is offered or will be given towards installation of such external software. That being said, POV-Ray is a very powerful visualization tool and its use is recommended. In addition (under unix), if the environment variable **POVEXE** is set to the location of the POV-Ray(TM) executable, POV-Ray(TM) will be automatically invoked as a background process and the current scene rendered, on screen and also to the file **compid.ppm**.

In **POV IMG** mode the user can adjust several options by selecting the appropriate controls in the PIG message window. When satisfied, the user should reselect **POV IMG** to create the .pov file and render it - or choose **PIG** to cancel the mode. In addition, if labels are switched on in PIG then the output image will be labelled, and if the unit cell is being drawn, then it too will be rendered in the POV-Ray(TM) image.

- **CPK model** This control sets the desired molecular representation to either close packed overlapping spheres (**CPK**), Ball and stick model, Rods or thermal ellipsoids.
- **400x300** This is the smallest of five preset image dimensions. Rendering time increases significantly with image dimension.
- **No Octants** This is intended for ellipsoid plots where the principle axes may not be obvious.
- **Prob 50%** This sets the default probability level for rendering thermal ellipsoid plots.

Users should note that the ASCII text **compid.pov** file generated by XTAL includes important header information from the **xtalpov** file located in the directory specified by the **XTALHOME** environment variable. However, if a user has an alternate **xtalpov** file in the current working directory, it will be used by default. In this way image renderings can be customised by the user, setting different backgrounds, adjusting perspective, scaling and atom colors. Also the xtal.pov files generated by XTAL are sufficiently well commented that they can be edited by users to delete selective atoms, their labels or bonds, and redisplayed by manually running POV-Ray(TM) from the command line.

4.46.2.27. BDF on/off

Clicking on this panel toggles the output mode of the archive bdf between on and off. The default mode is on. If the mode is on, the displayed atom sites (including the retained peak sites and labels), and the orientation matrix of the displayed sites, will be stored on the archive file by the program **ADDATM**. Independently of the bdf mode, a file .add is output containing all displayed atom sites and the display orientation.

4.46.2.28. EXIT

Clicking on this panel will commence the exiting process from PIG. First the display will be redrawn in the currently stored orientation. If the user wishes to sort the output atom list based on atomic number and labels, there are three possible label SORT modes. The user can choose one mode and continue, or ignore and continue by reselecting the **EXIT** button. At this point the user has the option to proceed with an **ORTEP/PREVUE** process based on the current screen display, (assuming that a new `badf` will be written) or to complete the exiting process by clicking on the **EXIT** button again. An accidental `exit>` can be canceled by selecting the **PIG** icon.

4.46.2.29. NEXT/PREV

When PIG is automatically invoked after CRISP, as a structural solution viewer, (run as **PIG msol** from the command line) two additional control buttons are invoked between the message panel and the atom-type buttons. These buttons permit the stepping forwards and backwards through structural solutions loaded from the `.sol` file produced by CRISP.

4.46.3. Sort modes

The three atomlist sort modes all sort sites according to atomic number, but they differ in their label sorting algorithms. The method to choose for sorting labels is largely dictated by the structure and the site labelling scheme adopted. It is inevitable that the atom sites written to the `.add` file will require additional manual ordering for publication purposes, but the three modes provided should provide a helpful starting point. The different schemes are best illustrated by way of example.

sort left sorts with the leftmost label character most significant

atom F2	.28990	.19669	1.00840	.035
atom F3	.28330	.25008	.79120	.035
atom F32	.35080	.42862	.47520	.035
atom F33	.48430	.37205	.31320	.035
atom F35	1.12030	.28096	.45740	.035
atom F36	1.01950	.33188	.62620	.035
atom F5	1.02790	.17466	.67020	.035
atom F6	1.08261	.12219	.87690	.035

sort right sorts with the rightmost label character most significant

atom F2	.28990	.19669	1.00840	.035
atom F32	.35080	.42862	.47520	.035
atom F3	.28330	.25008	.79120	.035
atom F33	.48430	.37205	.31320	.035
atom F5	1.02790	.17466	.67020	.035
atom F35	1.12030	.28096	.45740	.035
atom F6	1.08261	.12219	.87690	.035
atom F36	1.01950	.33188	.62620	.035

sort numeric sorts labels into numeric order

atom F2	.28990	.19669	1.00840	.035	1.00
atom F3	.28330	.25008	.79120	.035	1.00
atom F5	1.02790	.17466	.67020	.035	1.00
atom F6	1.08261	.12219	.87690	.035	1.00
atom F32	.35080	.42862	.47520	.035	1.00
atom F33	.48430	.37205	.31320	.035	1.00
atom F35	1.12030	.28096	.45740	.035	1.00
atom F36	1.01950	.33188	.62620	.035	1.00

4.46.4. File Assignments

- Reads atom data from the input archive `bdf`
- Optionally reads the *BUNYIP* file `bun` if the `buny` option entered. Optionally reads peak data from `bdf` `pek`
- Optionally reads the *CRISP* file `sol` if the `msol` option is entered
- Writes final connected atom sites to line file `add` and thereby to the output archive `bdf`
- Writes final cartesian coordinates to the line file `pdb`
- Writes display cartesian coordinates to the line file `xyz`
- Writes POV-Ray description file `pov`

4.46.5. Examples

PIG

Enter atom data from archive `bdf`

```
PIG
peaks prad 1.
radii Zr 1.7 Cl 1.2
```

Enter atom and peak sites from file `pek`. Set the peak site radius to 1.0 Angstrom, and the atom type radii for Zn and Cl atoms to 1.7 and 1.2 Angstroms, respectively.

```
PIG ort          : invoke PIG with a view to ORTEP generation
radii Zr 1.7 Cl 1.2 : set bonding radii
ortctl exec ialab nhla : override automatic ORTEP line option
ortctl vsc 1 1 2 2 5 1.5 1.6 : Add an extra vsc instruction
ortctl genins axes 2 : override automatic genins line
ortctl ellipse 6 3.37 : override ellipse line
finish
```

Use PIG as a dedicated ORTEP plot specifier with additional explicit ORTEP controls to fine tune the output. Subsequently run PREVUE to edit label positions and potentially invoke PLOTX from PREVUE to generate Postscript output.

4.47. **PLOTX**: Interface to graphics devices

Authors: Syd Hall, Roeli Olthof-Hazekamp, Howard Flack, James Hester & Ursula Bartsch

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

*PLOTX converts graphical information generated by the programs such as **CONTRS** and **ORTEP** into plot commands for specific devices.*

4.47.1. Function

PLOTX generates data appropriate for the different graphical protocols (e.g. X11, HPGL, PostScript) for use with local plotting devices. A number of new options for setting the postscript papersize have been added.

4.47.2. Scaling And Centring

The positive x-axis of a plot is taken as horizontal from left to right, and the positive y-axis is taken as vertical from bottom to top. Default frame dimensions and resolutions for the different devices are defined for each device at implementation time. They may be changed with the **params** line. High resolution values (i.e. small values entered in field 7) may result in smoother plots but will run slowly. To magnify or contract a plot, or to translate a plot within the given frame, the user may define the scale factor and/or plot centre manually.

4.47.3. Colour Graphics

The plot command **bdf** generated by some programs contains subject indices that are used to produce colour plots. This facility allows different subjects, such as atoms, bonds, numbers, characters and contours, to be plotted in different colours. Default colours are set within PLOTX but the user can specify his own colours using the **color** line. For atoms up to 16 different colours are allowed. Options available on the **color** line are:

- **fill** causes closed areas such as those bounding atoms or bonds to be filled with a specified colour or pattern. The boundary line colour index may be different to the fill colour. This is the default for **vdw** only.
- **nofill** draws atoms without colour filling.
- **bond** signals that the integer specifying the colour index of the bonds follows.
- **char** signals that the integer specifying the colour index of character strings follows (e.g. atom names and title).
- **numb** signals that the integer specifying the colour index of numbers follows (e.g. bond distances).
- **contur** signals that the integers specifying the colour indices of contours follow.
 - **n1** is the colour of the negative contour levels
 - **n2** is the zero contour level colour
 - **n3** is the colour of the positive contour levels
 - **n4** is the atom site marker colour
 - **n5** is the character string colour.
- **noco1** causes 1 colour to be plotted. This overrides the default colours for **vdw**.

4.47.4. Postscript output

The format of postscript output has been modified significantly to make it more amenable to manual editing. It has also been modified to conform to the ADOBE postscript Document Structuring Conventions for multi-page documents. This permits the page selection methods of postscript previewers to be used to step through multi-page plots, such as multi-layered contour diagrams. To support these modifications a postscript prolog file **xtalps** is supplied with xtal and is included in every postscript output plot. For including single page XTAL postscript images in other image/manipulation packages it may be necessary to edit the first line of the .ps file to indicate that it is to be treated as an encapsulated image. The generated .ps file contains instructions for doing this.

4.47.5. XFIG3.2 output

XFIG is free software for interactively editing vector graphics under X11. **XFIG3.2** can be obtained from <http://www-epb.lbl.gov/xfig/>. Xtal is *not* associated with the XFIG project in any way, we merely recommend their software as a useful additional graphical editing tool for post processing Xtal plot files.

4.47.6. File Assignments

- Reads plot commands from a specified file (default is `ort`)
- Writes a plotter output to locally defined devices or files

4.47.7. Examples

```
PLOTX
```

This produces an *ORTEP* plot on the graphics screen in X11 protocol.

```
PLOTX post1 nwm
```

A *NEWMAN* plot from file `nwm` is converted to a landscape PostScript file.

4.48. *POWGEN*: Powder pattern graphics

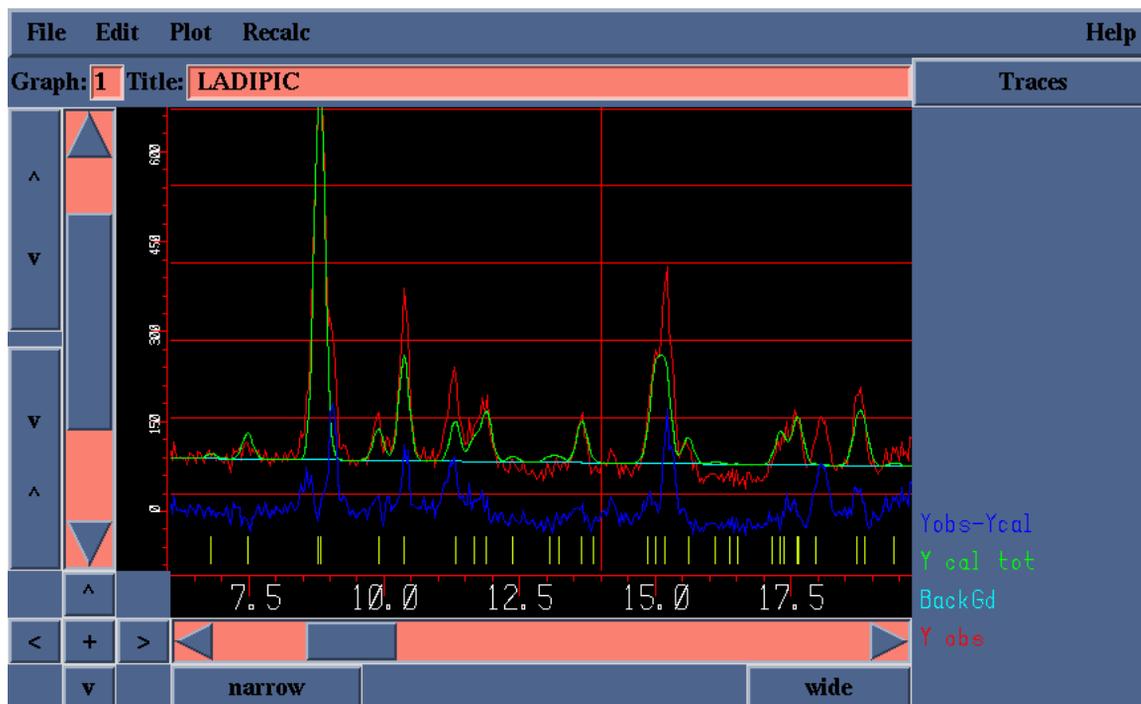
Authors: Doug du Boulay

Contact: Doug du Boulay, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

Warning: This is alpha software!

POWGEN optionally reads a binary pattern archive created with *ADDPAT* and calculates the expected pattern. These patterns can be viewed and specific factors controlling the calculation manually adjusted to improve fit. The plot range and various graphics rendering attributes can be adjusted and the plots dumped to a plot file for editing or printing. Multiple plots can be generated on one page and multiple pages can be printed.

Figure 4-3. Screen shot of POWGEN



4.48.1. Overview

POWGEN is a graphical interface program for displaying and manipulating powder diffraction pattern plots. It was designed to permit the user to dump an image of the manipulated page to a .ppp file for previewing with PREVUE or plotting with PLOTX. Multiple pages can be dumped and multiple powder plots can be displayed on one page. The graphical interface was designed exclusively using Tcl/Tk graphics functions, but if desired, the on screen graphics mode can be turned off and a complete description of ONE single page can be specified from the POWGEN control lines. That allows for efficient plotting once a plot format has been designed. In interactive mode the user can manipulate many of the attributes of either, the current graph block, or the layout of the whole page.

POWGEN requires as input, either of a .pat file produced by ADDPAT, or a .POT file produced by RIETVD. A check is made for the .POT file first, defaulting to the .PAT file if it exists. If the .PAT file is obtained, it is envisaged that a powder profile could be calculated and manipulated to a certain extent on screen. That requires atom site and hkl data on the input bdf so ADDATM and ADDREF are prerequisites in this mode. Many of the RIETVD commands also apply to POWGEN.

Page layouts can be either rigorously enforced to users specifications, or optimised to fit the page in a number of ways. The page layout is specified via the ARANGE option, where relative graph block positions are specified on a 9x9 grid, with (5,5) defining the location of a one graph block page. Typically EXAMPP will initiate with one graph invoked but with others being added or deleted as the user requires.

4.48.2. File Assignments

- Reads atom site and hkl data from the input archive file(s).
- Reads measured powder pattern intensities from file PAT.
- Reads calculated powder pattern intensities from file POT.
- Writes powder pattern plot data to file PPP.

- Writes card images to punch file PCH (not implemented yet).

4.48.3. Examples

```

POWGEN pat ho cm 0.5 1.0
:
phase ed 0.00065      : set the scale factor for this phase
peak voight 10. 0.5  : span(in halfwidths) mixing func
hwidth 0.008 0.01 0.0 :
pasymm 0.0 0.0       : assume symmetric peak
:
block $2 5 5 :          This block is centered in layout (5 5)
:                    this block has no numeric label
:                    So apply these settings to all subsequent blocks
axis x 10 72
axis y -1000 10000
trace obs 1 0.0 0.10 1.0 no $2 1
trace cal 2 0.0 0.05 1.0 no $2 1
trace del 3 0.0 0.0 1.0 no $2 1
:
block 2 5 6 :          This graph block is below previous block
titext This is graph number 2
axis x 10 30 :          plot range 2 theta = 10 to 30 degrees
axis y -1000 6000 :     Intensity range
axslab x two theta angle
axslab y Intensity (counts per sec)
trace obs 1 0.0 0.1 1.0 no $2 1 : plot observed
trace cal 2 0.0 0.0 1.0 no $2 1 : plot calculated
trace del 3 0.0 0.2 1.0 no $2 1 : plot difference (delta)
:
:                    If print is requested a .ppp plot file is written
PREVUE ppp
PLOTX xfigl ppp :      produce .fig file for line editing/printing in XFIG
finish

```

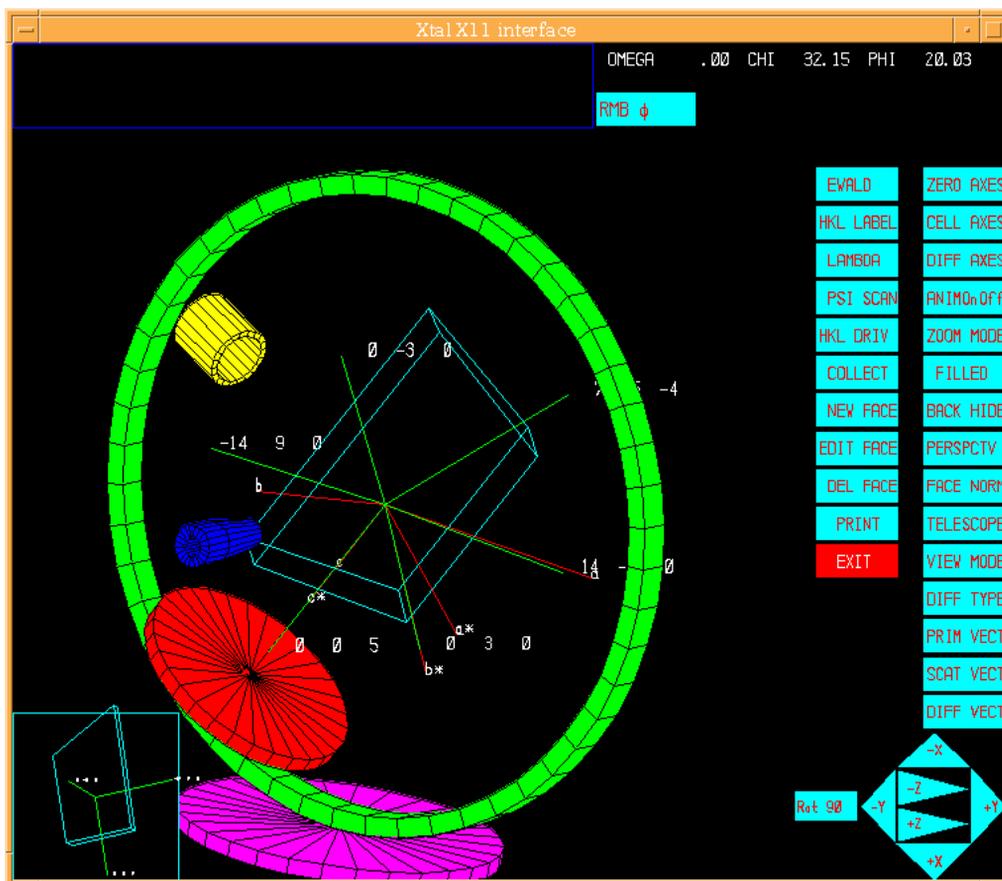
4.49. *PREABS*: Interactive crystal shape manipulation

Authors: Doug du Boulay

Contact: Doug du Boulay, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

PREABS is a screen-based interactive X11 display routine for checking and viewing the crystal shape description required for *ABSORB*. The diffractometer axes may also be rotated.

Figure 4-4. Screen shot of PREABS



4.49.1. Overview

PREABS is a work in progress. It is intended to permit graphical editing of crystal size and shape in order to accurately model real crystals of ill defined shape. Unfortunately the editing aspect of this program has not been completed. It does however, permit visualization of crystal models constructed via the usual ABSORB face description methods. The crystal model is displayed, embedded within the ABSORB, internal, equivalent diffractometer specification, corresponding to the Busing and Levy (1967) system. The overall view of the crystal and diffractometer can be rotated with left-mouse-click-drag events, and by switching the right-mouse-button mode, the crystal can be rotated about specific diffractometer circle axes. The position of the telescope is adjustable and an image of the crystal through the telescope can be obtained.

4.49.2. General control functionality

4.49.2.1. ZERO AXES

This control reset all diffractometer angles to zero

4.49.2.2. CELL AXES

Switches on/or off both the reciprocal and real space axes on the model

4.49.2.3. DIFF AXES

Switches the diffractometer coordinate systems on or off. This is a four mode switch, since both the XTAL coordinate system and the actual diffractometer coordinate system may be different. Setting 0 switches both

axes systems off. Setting 1 switches the XTAL coords on, setting 2 switches the real coords on, while the final setting displays both sets of axes. Drawing the real diffractometer axes requires the presence of the original UB matrix used to measure the reflection data on the xtal archive.

4.49.2.4. ANIM On/Off

Activating the `ANIM` option switches on animation. In this mode, and left-mouse-click-drag-release events are converted to angular momentum, and the diffractometer as a whole is set spinning. To stop a spinning image, simply left-click-release with no mouse motion. Reselecting `ANIM`, switches this mode off.

4.49.2.5. ZOOM MODE

Currently has no functionality

4.49.2.6. FILLED

Switches filling of the front display surfaces of the diffractometer circles on or off. This can result in significantly slower rendering.

4.49.2.7. BACK HIDE

This control switches on or off the display of diffractometer faces which face away from the viewing direction.

4.49.2.8. PERSPECTIVE

Currently has no functionality

4.49.2.9. FACE NORM

This control switches on or off the crystal face normals and their associated approximate miller indices.

4.49.2.10. TELESCOPE

Activating the `TELESCOPE` mode, temporarily changes the functionality of the right mouse button, such that by right-mouse-click-drag-release events, the position of the telescope can be dragged to an orientation corresponding to that on the actual measurement diffractometer. The telescope can then be locked to move synchronously with the omega axis, if that is where it is mounted, or else it can be locked rigidly with respect to the incident beam collimator. These settings are diffractometer specific, but the user must choose between the two `COLLIMATOR` or `W_CIRCLE` options available in the message display panel before exiting from telescope adjustment mode.

4.49.2.11. VIEW MODE

Three viewing modes are available. The default view is an overall view of the whole diffractometer. The second view has an additional reduced image of the crystal in the lower left corner, as seen through the telescope. The third view is a larger view of the crystal through the telescope. Note that the telescope view has greatly reduced functionality over the default diffractometer view.

4.49.2.12. DIFF TYPE

Currently has no functionality

4.49.2.13. PRIM VECTOR

Currently has no functionality.

4.49.2.14. SCAT VECTOR

SCAT VECTOR is a three state button for displaying scattering vectors of reflections lying on or near the Ewald sphere surface. In the first mode, only those reflections lying in the equatorial plane of the diffractometer are displayed, whereas in the second mode all reflections on the surface of the Ewald sphere are represented by scattering vectors. In the third mode scattering vectors are switched off.

4.49.2.15. DIFF VECTOR

DIFF VECTOR is a three state button for displaying the diffracted beams scattered by crystal in its current orientation i.e. those reflections lying on or near the Ewald sphere surface. In the first mode, only those diffraction vectors of reflections lying in the equatorial plane of the diffractometer are displayed, whereas in the second mode all reflections on the surface of the Ewald sphere are represented. In the third mode diffraction vectors are switched off.

4.49.2.16. EWALD

EWALD is a three state button for displaying the currently active reflections lying near the surface of the EWALD sphere as reciprocal space points, rather than vectors. The first mode displays just those near active reflections in the equatorial plane, the second displays all reflections near the Ewald surface, and the third mode disables the reciprocal lattice point display. It is important to note that the tolerance for points being displayed on this surface is essentially $\pm 1/2h, k$ or l , which is in fact very much higher than you could expect in a real experiment, so that many more reflections appear active than is actually the likely, given their distance from the sphere surface.

4.49.2.17. HKL LABEL

This is another three state control for switching on/off the labelling of reciprocal lattice points lying near the Ewald sphere surface.

4.49.2.18. LAMBDA

Currently has no functionality.

4.49.2.19. PSI SCAN

Currently has no functionality.

4.49.2.20. HKL DRIVE

Currently has no functionality.

4.49.2.21. COLLECT

Currently has no functionality.

4.49.2.22. NEW FACE

Currently has no functionality.

4.49.2.23. EDIT FACE

Currently has no functionality.

4.49.2.24. DELETE FACE

Currently has no functionality.

4.49.2.25. PRINT

Currently has no functionality.

4.49.2.26. EXIT

Terminates the current PREABS process immediately.

4.49.2.27. RMB ?

This controls the default functionality of the Right-Mouse-Button. By default this is set to ω but successive selections change it to χ , ϕ and back to ω . In any particular mode, rotations about the corresponding diffractometer axis are calculated by representing that particular axis circle as a sphere centred on the crystal and projecting forward out of the screen, but constrained to rotate about its axis. Any right-mouse-motion is mapped to rotations of this sphere about the axis in question.

4.49.3. Known shortcomings

- Crystal image doesn't necessarily reflect the true orientation on the measurement diffractometer - which could be confusing.
- Non closed crystal shapes can cause disturbing image views.
- Much of the intended functionality has not been written yet.

4.49.4. File Assignments

- Reads hkl data from the input archive file(s).

4.49.5. References

- Busing, W.R. and Levy, H.A. 1967. *Angle Calculations for 3- and 4- Circle X-ray and Neutron Diffractometers*. *Acta Cryst.* **22**, 457-464.
- Tanaka, K. & Saito, Y. (1975). *Acta Cryst.* **A31**, 841-845.

4.50. *PREPUB*: Pre-publication tests on CIF structural data.

Authors: Doug du Boulay & Syd Hall

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

PREPUB performs a subset of the IUCr data validation series of CIF tests on all data blocks of a supplied CIF containing structural data.

4.50.1. Description

A series of checks are made on a CIF containing one or more data blocks. The following tests are applied:

1. Test that the data organisation of the CIF conforms to the STAR syntax. Mismatched tags and values are identified, as well as unclosed text blocks, and whether items are missing from a looped list.
2. Check that the data items in the CIF are defined in the standard CoreCIF dictionary.
3. **SYMMG01 DVtest:** Check the `_symmetry_space_group_number` value matches that expected for the `_symmetry_space_group_name_H-M` entry.
4. **SYMMG02 DVtest:** Check that the `_symmetry_equiv_pos_as_xyz` values are consistent with the `_symmetry_space_group_name_H-M` entry.
5. **CELLZ01 DVtest:** Check that the combination of the `_cell_formula_units_Z` number and the `_chemical_formula_sum` entry matches the atomic content derived from the atomic site information in the `_atom_site` list.
6. **CHEMW03 DVtest:** Check that the `_chemical_formula_weight` value matches that calculated from the atomic site information in the `_atom_site` list.
7. **REFLT03 DVtest:** Check that the calculated number of reflections in the diffraction sphere out to the `_diffrn_reflns_theta_max` value is consistent with the number reported in the CIF by `_reflns_number_total`. Checks are made on the `_diffrn_reflns_limit_max` and `_diffrn_reflns_limit_min` values, and the presence of Friedel pairs.
8. **STRVAL02 DVtest:** Check that the `_refine_ls_abs_structure_Flack` value is in a sensible range.

4.50.2. Examples

A typical input sequence is as follows, assuming a CIF filename is `banana.cif`

```
compid banana
prepub
```

This results in an output, such as:


```

TEST3: Check Friedels for noncentro structure
      Estimate of Friedel pairs measured      894
      Fraction of Friedel pairs measured    0.651
      Are heavy atom types Z>Si present      yes
REFLT_03: OK
STRVAL_02: Check absolute structure measures
-----
From the CIF: _refine_ls_abs_structure_Flack      0.390
From the CIF: _refine_ls_abs_structure_Flack_su  0.070
ALERT: Flack test results are ambiguous.
Time  h  m  s      CPU secs      Total CPU secs      Memory words
     12:29:53      9.26126          9.40571          201001

```

4.51. *PREVUE*: Modify text in plots

Authors: Victor Streltsov, Doug du Boulay & Syd Hall

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

PREVUE is a screen-based interactive display routine for manipulating text on plots. It outputs a plot file *pre*. The *pre* file can be read by *PLOTX* to produce hardcopy plots.

4.51.1. Description

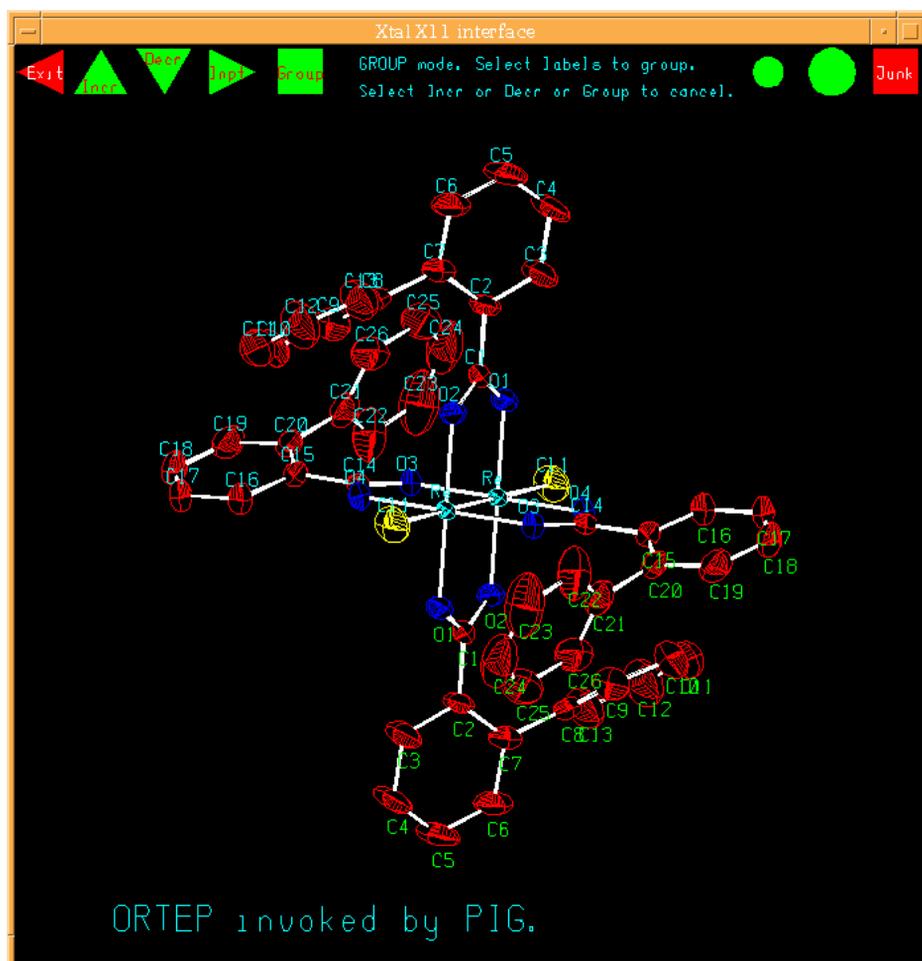
PREVUE enables the user to manipulate text strings in a plot. The plot itself cannot be altered, but may be enlarged or reduced in size during the text manipulation (the output plot size is unaffected by this resizing).

A character string may be moved by clicking on it (which causes it to disappear from the screen) and then clicking on the screen where the string is to be positioned (where it becomes visible). The string is repositioned with the bottom left corner at the mouse release position.

Eight control buttons are displayed at the top of the plot screen.

- The red **JUNK** button is the rubbish bin for unwanted text. A string moved into this box will be deleted from the output plot.
- The two green circular **ZOOM** buttons are for enlarging and reducing the plot. These are for display convenience only and do not affect the output plot data.
- The two green triangular **INCR** and **DECR** buttons are used to increase and decrease the font size of all character strings by 10%.
- The square green **GROUP** button, allows selective labels to be moved collectively, or preferentially resized. Selecting **GROUP** enables a new graphics mode whereby labels can be selectively grouped. Selecting a non-selected label, adds it to the group and reselecting it deletes it from the group. Grouped labels change color and remain grouped until the **GROUP** button is reselected. Right mouse button clicking anywhere in the display window, which is not occupied by a text label, combined with a drag and release, **EXCLUSIVE OR**'s the labels contained within the rectangular region indicated. Left mouse button clicking anywhere in the display window, which is not occupied by a text label, combined with a drag and release, moves the position of all grouped text labels collectively. Pressing a mouse button over the **Junk** button permanently deletes all currently grouped labels. There is no undo. Selecting **INCR/DECR** buttons, resizes only the grouped labels. The **GROUP** mode is de-activated by reselecting **GROUP**.
- The green **INPT** button is used to insert a new character string. After clicking on the position for the input string this is entered on the keyboard, terminated by a carriage return.

Figure 4-5. Screen shot of PREVUE



- The red **EXIT** button steps to the next plot file frame if more than one exists. If there are no more frames present on the input file PREVUE provides the option to automatically invoke PLOTX to generate hardcopy postscript or HPGL output. If either of these output formats is desired, a menu of format controls for each format is invoked. Note that only A3, A4 and landscape mode are available options for HPGL. When choices are made choose the red **OK** control to terminate PREVUE and continue.

The output file `pre` will contain all changes to the text data. Note that PREVUE will not read a `pre` file unless it is renamed.

4.51.2. File assignments

- Reads the input plot file (default is `ort`)
- Writes output plot file `pre`

4.52. **REFCAL**: Process reflection data

Author: H.D. Flack, Laboratoire de Cristallographie, University of Geneva 4, CH-1211 Geneve 4, Switzerland

REFCAL treats reflection raw intensity and background, net intensity $|F_o|^2$ or $|F_o|$ data

4.52.1. Overview

REFCAL processes intensity data coming from single crystal diffractometer measurements. The programme uses data already placed on the input bdf in the form (a) raw counts and associated experimental details, (b) net intensities and associated experimental details, (c) $|F_o|^2$ or (d) $|F_o|$ values. The reference reflections can be analysed both for long-term drift and to determine the short term stability of the measurements. The input data are transformed in a downhill cascade manner: raw intensities --> net intensities > $|F_o|^2$ > $|F_o|$ and output according to option. The Lorentz-polarization factor, $\sin\theta/\lambda$, rcode, reflection multiplicity, symmetry reinforcement factor ϵ and phase restriction code may be calculated, interpolated scattering factors may be inserted into the bdf and systematically-absent reflections may be marked or removed.

4.52.2. Data Input, Requirements and Transformations

The recommended method of placing raw diffractometer data on the bdf is first to transform a diffractometer-specific file into a CIF or SCFS90 (Standard Crystallographic File Structure 90, Brown, 1991) format using *Difrac* (Flack, Blanc Schwarzenbach, 1991). The `cif` file may be read by *CIFIO* or the `sfi` file by *REFM90* to create an Xtal bdf. This process compactly loads all of the necessary raw experimental data onto a bdf in a diffractometer-independent way which preserves all essential information concerning the data-gathering process. In a preliminary scan through the bdf, REFCAL determines whether the data in the input bdf are complete and in which of the forms (raw counts, net intensities, $|F_o|^2$ or $|F_o|$) they are present.

When REFCAL is run the input bdf must also contain all cell, symmetry and cell content information pertinent to the compound. This implies that the input bdf must contain the data generated by *STARTX*.

When using the input of raw net intensity data, the input bdf should contain the following information concerning the experimental method of intensity measurement: type of radiation used, mean wavelength of radiation, temperature of intensity measurements, minimum and maximum $\sin\theta/\lambda$ of intensity measurement, total number of reflections measured, minimum and maximum values of h , k and l used in data collection, Miller indices of reference reflections, crystal shape information as either (a) Miller indices of faces and their distances and

e.s.d.s from centre, or (b) the radius of a spherical crystal or (c) the radius and length of a cylindrical crystal, detector deadtime and e.s.d., wavelength of β filter absorption edge, for each attenuator filter: its reciprocal transmission factor with e.s.d and index; incident beam polarization characteristics *viz* polarization ratio, and e.s.d., dihedral angle between diffraction planes of the sample and monochromator, incident beam half-width, wavelength and intensity weights of spectral line components.

For raw intensities, the data for each reflection should contain the following information. h,k,l , peak count, high- and low-angle background counts, Flack, Blanc & Schwarzenbach coefficients, crystal-based azimuthal angle, elapsed time of measurement, attenuator filter index, scale factor index, reference reflection index, background scanning mode, total scan width, scan type, total horizontal and vertical detector aperture.

For net intensities, the data for each reflection should contain the following information h,k,l , net intensity and its e.s.d., crystal-based azimuthal angle, elapsed time of measurement, attenuator filter index, scale factor index, reference reflection index, background scanning mode, total scan width, scan type, total horizontal and vertical detector aperture. The net intensity is that derived from the raw peak counts by subtraction only of the background but without *any* other correction factor being applied to it.

For $|F_o|^2$ input, the data for each reflection should contain h,k,l , $|F_o|^2$ and its e.s.d. $|F_o|^2$ may be obtained from the net intensity by applying *all* systematic corrections. For $|F_o|$ input, the data for each reflection should contain h,k,l , $|F_o|$ and its e.s.d.

4.52.3. Treatment of Reference Reflections

REFCAL can undertake an analysis of reference reflections. These are specified reflections whose intensities are remeasured at regular intervals. These measurements are used for two purposes: (a) The establishment of a set of scale factors based on the counts of the reference reflections to compensate all reflections for any drift over the data gathering process, (b) The calculation of instability constants which are based on the spread and perhaps the trend of the measurement of the reference reflections.

During the course of the data treatment, the sets of interspersed reference reflection measurements (in the form of reference index, net intensity, its e.s.d., elapsed time of measurement and set index) are stored in memory. Each set contains at most one measurement of each reference reflection although all references need not be present in each set. Reference reflections separated by non-reference reflections in the sequence of measurements belong to different sets of reference reflections.

No more than 30 different reflections may be designated as reference reflections. Under option the user may specify whether the reference reflections, once they have been used in the scaling and stability calculation, should be removed from the bdf or kept as observed reflections for later use (such as merging) with the other intensity measurements.

4.52.4. Calculation and application of scale factors

The scales will be generated from the reference reflection sets and smoothed over a specified number of scale factors. The default smoothing range is five scale factors forward and five scale factors backward. This smoothing is necessary due to the statistical counting fluctuations in the measured intensities. No smoothing will occur if the smoothing range is set to 0. The `app1/naps` options on the `REFCAL` line control the application of the scale factors to all intensity data in the form of $|F_o|^2$, or $|F_o|$. Raw counts and net intensities are NOT modified.

Two input lines, `setscl` and `discon` are available to control the calculation of scale factors. `setscl` specifies scale factors. Scale factors which have been calculated automatically will have their calculated value overwritten by the `setscl` value.

`discon` provides a way of indicating the position of an abrupt discontinuity in the scale factor values. Abrupt scale discontinuities may occur if there is a change in the radiation source or a degradation in the crystal. A scale factor, indicated by its time on the `discon` line, will be understood to be the last member of a set of

scale factors over which smoothing takes place. The scale factor following the one indicated on the **discon** line will be taken as the first of another set used for smoothing. The smoothing function will not span the two sets. It may be necessary to make a preliminary run of REFCAL to find the discontinuities.

4.52.5. Calculation and application of instability constants

From the average value of the net intensity and the individual measurements, the external variance may be calculated for each reference reflection. This gives a measure of the variance over and above the variance based on counting statistics alone. The model used for the total variance of a reflection intensity is given by:

$$(\text{Total e.s.d.})^2 = (\text{Counting e.s.d.})^2 + |b| + |m| \cdot (\text{Net intensity})^2.$$

In REFCAL different estimates (as specified below) of the instability coefficients b and m are obtained by least-squares analysis, using all reference reflections, of the individual difference between the external and the counting-statistics variances against the square of the individual average net intensities. b and m may be estimated either before or after the reference reflections are scaled. In the first approach, specified by the **befr** option in the **REFCAL** line, the values of b and m obtained are subject to all the variations in reference reflections that occur during the course of measurement such as crystal degradation. In the second approach, specified by the default **after** option on the **REFCAL** line, b and m will be subject only to those fast changing variations in the reference intensities which have not been removed by the overall rescaling process.

The calculation of the instability coefficients b and m is controlled by the **inst** option on the **REFCAL** line with the value of \circ as follows:

- \circ Meaning of \circ
- 0 Fix $b = m = 0.0$. *i.e.* only counting statistics are used in calculating the e.s.d of an intensity.
- 1 Fix $b = 0.0$ and let m be the only variable. *i.e.* a straight line through the origin is fitted.
- 2 Let both b and m be variable. *i.e.* a general straight line is fitted.
- 3 Both slope m and intercept b are fitted but the intercept b is reset to zero.
- 4 The user supplies both m and b .

No attempt will be made to calculate the values of instability constants if the reference reflections have been measured less than seven times during the data gathering procedure. Raw counts and net intensities are NOT modified by the instability calculation.

4.52.6. Reflection Status Codes (rcodes)

Xtal uses a system of reflection status codes described earlier in the Primer Section. Each reflection is tested against its e.s.d. Those reflections which show a $|F_o|$ more than $n \cdot [\text{e.s.d.}(|F_o|)]$ are coded as $rcode = 1$ (observed) and those with less than this value are coded with $rcode = 2$ (less-thans). The value of n may be specified in the **REFCAL** line by use of the **obst** specification. Zero is allowed and means that all reflections will be given an $rcode = 1$. Each reflection is tested to see if under the specified space group it would be systematically absent. If absent, the reflection is either rejected or included in the bdf marked with an $rcode$ of 5.

4.52.7. Derivation of Intensities and "Observed" Structure Factors

The reflection data are transformed in a downhill cascade manner; raw intensities --> net intensities --> $|F_o|^2$ --> $|F_o|$ starting at the form of the input data. The form of the reflection data output is indicated by the **rawi**, **neti**, **fsqr** or **fre1** options on the **REFCAL** line.

The net intensity is calculated from the raw intensity data according to the scheme for reducing raw peak and background counts described in Flack, Blanc & Schwarzenbach (1991) with the coefficients cn_p ; cn_b , cn_h and cn_l available on the bdf. The net intensity is that derived from the raw peak counts by subtraction only of the background but without *any* other correction factor being applied to it.

$|F_o|$ is obtained from the net intensity by applying *all* systematic corrections. $|F_o|^2$ is obtained from $|F_o|$ by treating negative $|F_o|^2$ values as zero. The calculation of the e.s.d. ($|F_o|$) is carried out using the following expression:

$$\sigma(|F_o|) = [|F_o|^2 + \sigma(|F_o|^2)]^{1/2} - |F_o|$$

4.52.8. Lorentz-Polarization Factors

The Lorentz-polarization factor for each reflection is calculated from the values of the polarization ratio K and the dihedral angle ρ , the angle between the diffraction planes of the monochromator and the sample (*diffraction plane* is the plane containing the incident and diffracted ray directions). Let θ the Bragg angle of the sample. For single crystal equatorial-plane diffractometer measurements, the Lorentz factor = $L = 1/(2 \sin 2\theta)$. The expression for polarization of a twice-diffracted X-ray beam is given by:

$$P = [\cos^2 \rho + K \sin^2 \rho + (\sin^2 \rho + K \cos^2 \rho) \cos^2 2\theta] / (1 + K).$$

For diffraction by neutrons, there is no polarization. See Azaroff (1955) and Hope (1977).

4.52.9. Atomic Scattering Factor Interpolation

The scattering factor tables present on the input bdf (as stored by *STARTX*) may be used to calculate interpolated atomic scattering factor values for each reflection. A four-point interpolation procedure is used (Rollett, 1965). The interpolated values are stored with each reflection. This option is invoked by the `FFAC` signal on the `REFCAL` line. The advantage of interpolated atomic scattering factors is that they provide for more precise structure factor calculations with programmes such as *FC* and *CRYLSQ*. The disadvantage is that the size of the bdf is increased substantially and the extra precision is usually unwarranted for routine structure analyses.

****WARNING**** Users must use this option if *R*-factors of less than 0.04 are anticipated.

4.52.10. Symmetry Reinforcement Factor ϵ

The symmetry reinforcement factor ϵ is the integer multiplicity of a reflection intensity due to the symmetry-generated coincidence of identical diffraction vectors (i.e. direction and phase). The value of ϵ is required for the calculation of *E* values, normalized structure factors. In `REFCAL` this factor is calculated for each reflection and stored in the binary data file for use by other programmes in the system. The method is described by Stewart and Karle (1976), Iwasaki and Ito (1977); and Stewart and Karle (1977).

4.52.11. Examples

```
title    Create a bdf from an SCFS 90 archive file
REFM90   scfs
STARTX   upd
REFCAL   frel
ADDATM   upd
```

The above example shows the use of *REFM90*, *STARTX*, `REFCAL` and *ADDATM* to create a bdf with $|F_o|$ data from an archived data set in SCFS 90 format. *REFM90* is necessary to read the SCFS file; *STARTX* to calculate the direct and reciprocal cell metrics, load the scattering factor tables, atomic radii, and to check the symmetry

information; REFCAL to calculate the reflection information as detailed above and *ADDATM* to generate the symmetry constraints.

```
title      Form a bdf from an SCFS 90 diffractometer file
REFM90     scfs
STARTX     upd
sgname     -C 2YC
REFCAL
```

The above example shows the use of *REFM90*, *STARTX*, and REFCAL to load a bdf with net intensity data from diffractometer data produced in SCFS 90 format by the DIFRAC programme. The *sgname* line is necessary in *STARTX* as this information is not stored with the raw diffractometer data. Even if symmetry information were present in the SCFS file, it is still necessary to run *STARTX* in update mode to calculate and store the cell metrics, find the atomic radii, *etc.* and to check the symmetry information.

4.52.12. References

- Azaroff, L.V. (1955). *Acta Cryst.* **8**, 701.
- Brown, I.D. (1990). *Acta Cryst.* In preparation.
- Flack, H.D., Blanc, E & Schwarzenbach, D. (1991) *J. Appl. Cryst.* **24**, In preparation.
- Hope, H. (1977). *Acta Cryst.* **A27**, 392.
- Iwasaki, H. & Ito, T. (1977). *Acta Cryst.* **A33**, 227-229.
- Rollett, J.S. (1965). *Computing Techniques In Crystallography*. Elmsford, NY: Pergamon Press.
- Schwarzenbach, D & Flack, H.D. (1989). *J. Appl. Cryst.* **22**, 601-605.
- Stewart, J.M. & Karle, J. (1976). *Acta Cryst.* **A32**, 1005-1007.
- Stewart, J.M. & Karle, J. (1977). *Acta Cryst.* **A33**, 519.

4.53. *REFM90*: Convert SCFS-90 file to/from a BDF

Author: Howard Flack, Laboratoire de Cristallographie, University of Geneva, CH-1211 Geneve 4, Switzerland

REFM90 converts an XTAL archive bdf to an SCFS format file (SCFS-90, Brown, 1983, 1985, 1988, 1990) and a SCFS-90 file to an XTAL archive bdf.

4.53.1. Introduction

The Standard Crystallographic File Structure (Brown, 1983, 1985, 1988, 1990) is the current (July 1990) standard approved by the IUCr Data and Computing Commissions as the standard format for exchanging crystallographic data. REFM90 is a program for two-way conversion between an archive bdf and the SCFS-90 format.

A file of crystallographic data in a standard format has many potential uses. It should be a very convenient means of submitting short format papers for publication in *Acta Cryst. C*, for archiving crystallographic information, for transferring a series of measurements made on a single-crystal diffractometer, for transfer of data between computers in a multi-host environment, for running programmes external to an integrated system

such as XTAL, for communicating crystallographic data between laboratories by way of e-mail, floppy disks or magnetic tapes or for the preparation of material for deposit in electronic or hard-copy form. In REFM90 data are exchanged between the two file formats without any checking of their integrity and without any change of their potential information content.

Much of the code of REFM90 is taken or inspired from the programs REF87 and REFORM. REFM90 adheres *strictly* to the SCFS-90 standard. Superseded SCFS sections can neither be read nor produced by REFM90. Also, in those parts of the SCFS-90 standard where there is an indicated recommendation or preference for one format of the data over another, **ONLY** the recommended or preferred form can be interpreted by REFM90. REF() does not treat or use the SCFS-90 sections designed for macromolecular crystallography.

4.53.2. Description

4.53.2.1. 1. XTAL to SCFS Mode

The information of one single data set specified by its Data Set Number is output in SCFS-90 format. The data set to be output is selected on the **REFM90** line. The SCFS-90 sections output are as follows: TITLE; CELL DIMENSIONS (CID's: CELL, ERRS, VOL, PHYS); SGNAME (CID: LATT); SYMMETRY (CID: SYOP); FORM FACTOR (CID: TABL); FORMULA (CID: FORL); CONDITIONS (CID's: INT, ABS, STAT); CRYSTAL (CID: SCAL); ATOMS (CID: ATCO, ATCE, UALL, UIJ, UIJE); either HKL PACK OR HKL with CID: FOBS, FSQR, INT OR CALC according to the choice made on the **REFM90** line.

4.53.2.2. 2. SCFS to XTAL Mode

In reading an SCFS-90 file a completely new archive bdf is created. The same sections and CID's that can be produced in mode (1) can be interpreted in mode (2). This mode just dumps the data from the SCFS file into the archive bdf. It is *essential* to run **STARTX** immediately after REFM90 in SCFS to XTAL mode in order to check the integrity of the data and to assure that the archive bdf is completely formed.

4.53.3. File Assignments

	<i>SCFS-to-XTAL</i>	<i>XTAL-to-SCFS</i>
Reads	sfi	archive bdf
Writes	archive bdf	scf

4.53.4. Examples

```
REFM90                               :Convert XTAL bdf to SCFS87 file
```

```
REFM90 SCFS                           :Convert SCFS90 file to XTAL bdf
```

References

- Brown, I.D.(1983). *Acta Cryst.* **A39**, 216-224.
- Brown, I.D.(1985). *Acta Cryst.* **A41**, 399.

- Brown, I.D.(1988). *Acta Cryst.* **A44**, 232.
- Brown, I.D.(1990). *Acta Cryst.* **A46**, In press.

4.54. **REGFE**: Analyse errors

Authors: H. Wang, R.J. Barton and B.E. Robertson

Contact: Bev Robertson, Department of Chemistry, University of Regina, Regina, Saskatchewan, Canada S4S 0A2

REGFE calculates crystallographic functions and their estimated standard deviations (Busing et al., 1964). In the calculation of standard deviations, both the variances and the covariances of the parameters (Hamilton, 1964; Sands, 1982) are taken into account. REGFE may be used to create tables of bond lengths, bond angles and dihedral angles for publication and output them to the 'punch' file.

4.54.1. Introduction

REGFE calculates the values of derived quantities which are functions of the atomic coordinates and the crystal unit cell parameters and their associated standard deviations. These quantities are bond lengths, bond angles, dihedral angles, distances of atoms from planes and lines, and angles between planes and lines. The calculations of bond lengths, angles and dihedral angles of molecules or clusters can be performed, based on atom connectivity.

In least-squares structure refinement, the parameters of the crystal structure are not fully independent of each other; i.e., they are correlated. For any function of these parameters the covariances of the parameters should be included in the calculation of their standard deviation.

There are two variance-covariance (v-c) matrices which are involved in the calculation of the standard deviation; the matrix, C, of the cell parameters and the matrix, V, of the atom parameters. We assume that the cell parameters are fully independent of the atom parameters and the covariance terms involving a cell parameter and an atom parameter are set to zero.

The variance-covariance matrix of the atom parameters is derived from the inverse matrix of the normal matrix in least-squares refinement. If the weights for the observations (F, F² or I) are correct, V is the inverse of the normal matrix. However, it is assumed that any deviations of the goodness-of-fit, S, from its expected value represent a scaling error in the variances of the observations, and thus can be corrected by multiplying V by S²; i.e.

$$V = \mathbf{A}^{-1}S^2$$

where **A** is the normal matrix of the least-squares refinement, given by

$$A_{ij} = \sum_r [w \frac{dF_{cal}}{dp_i} \frac{dF_{cal}}{dp_j}]$$

and S, the goodness-of-fit is given by

$$S^2 = \sum_r [w\Delta^2] / (n-m)$$

The symbols and notation in these equations have been defined in the CRYLSQ section.

The diagonal terms of the v-c matrix are the variances of the parameters and the off-diagonal terms are the covariances of the parameters. The correlation matrix, N, of the atom parameters is a normalized variance-covariance matrix. The elements of the correlation matrix are defined as:

$$N_{ij} = V_{ij} / [V_{ii} V_{jj}] \quad (i,j = 1,2,3,\dots,m)$$

The correlation matrix of the atom parameters can be created and saved in the bdf **cmx** by entering the **ms** option code in CRYLSQ. Thus at least one cycle of least-squares refinement by CRYLSQ is required before running

REGFE. If the correlation matrix from least-squares structural refinement is a blocked matrix, the parameters in different blocks are considered to be uncorrelated and the covariances of parameters in two different blocks will be set to zero. If the correlation matrix is not provided, only the variances of the parameters will be used and all the covariances of the parameters will be set to zero.

4.54.2. Standard Deviation Calculation

For a given crystallographic function of the atom parameters a_1, a_2, a_3, \dots , and cell parameters c_1, c_2, c_3, \dots

$$F = f(a_1, a_2, a_3, \dots, c_1, c_2, c_3, \dots, c_6)$$

the variance (which is the square of the standard deviation, sigma) of the function can be obtained by

$$\sigma^2 = \sigma^2c + \sigma^2a$$

The first term in the right hand side of the equation is the contribution to the variance of the function from the variances of the unit cell parameters:

$$\sigma^2c = \sum_{ij} [C_{ij} dF/dc_i dF/dc_j]$$

and the second term is the contribution to the variance of the function from the variances of the atom parameters:

$$\sigma^2a = \sum_{ij} [V_{ij} dF/da_i dF/da_j]$$

4.54.3. Atom Connectivity

The atom connectivity is established according to the atomic radii data. The symmetry operations of the space group may be applied for the establishment of the atom connection table. The number (1, 27, 125) of unit cells can also be specified for atom searching. The values of the atomic radii must be provided from the input `bdf` or `atrad` lines. The values of the atomic radii in the `bdf` can be modified by entering `atrad` lines. The min allowed radius is 0.015 Å.

4.54.4. plane, line and atom Lines

In the function and error calculation, a plane can be represented by either a plane ID, which is predefined in a `plane` line, or by three atom IDs which define the plane. A line can be represented by either a line ID, which is predefined in a `line` line, or by two atom IDs which define the line. The equation of a plane can be created by entering the coefficients of the plane equation in either an orthogonal coordinate system or a fractional coordinate system, or by three atom IDs. Similarly, the equation of a line can be defined by entering the coefficients of a line equation in either an orthogonal coordinate system or a fractional coordinate system, or by two atom IDs. The forms of the equations of a plane and a line in fractional coordinate system are given by:

$$Ax + By + Cz = D$$

and

$$(x - x_0)/A = (y - y_0)/B = (z - z_0)/C$$

respectively. However, when the coefficients of the equations are input on lines, only the variance of the coefficients will be used in the standard deviation calculations.

4.54.5. File Assignments

- Optionally reads structure data from input archive `bdf`
- Optionally reads correlation matrix from `bdf` `cmx`
- Optionally outputs tables to line file `pch`

4.54.6. Examples

```
REGFE covar dihe
atrad zn 2.00 1.70 1.0
atrad cl 2.00 1.60 0.5
```

In this example, the atomic radii of Zn and Cl are modified and the bond lengths, bond angles, and dihedral angles of the molecule will be calculated based on the atom connectivity. The correlation matrix will be input from `cmx` and used in the standard deviation calculations.

```
reset psta 4
REGFE covar nlen nang lvc
plane pln1 0 12.345 -8.456 3.004 5.842
plane pln2 0 12.462 3.145 8.593 -4.053
plane pln3 2 c01 c02 c03
plane pln4 2 c04 c05 c06
plane plna 1 0.6258 0.2449 0.7405 5.005
functn vol
functn dis c01 c02
functn pap pln3 c04 c05 c06
functn dih cl01 zn01 o01 n01
functn ang c01 c02 c03
functn pda pln3 c04
functn pap c01 c02 c03 pln4
```

In this example, the calculation of the bond lengths, angles and dihedral angles is suppressed. The covariance of the atom parameter will be used in the standard deviation calculation and the variance-covariance matrix will be listed in the output.

4.54.7. References

- Busing, W.R., Martin, K.O. and Levy, H.A. 1964. *ORFFE--Oak Ridge National Laboratory*, Report ORNL-TM-306.
- Hamilton, W.C. 1964. *Statistics in Physical Science*, The Ronald Press Company: New York.
- Sands, D.E. 1982. *Vectors and Tensors in Crystallography*, Addison-Wesley: New York.

4.55. **REGWT**: Analyse weights

Authors: B.E. Robertson & H. Wang

Contact: Bev Robertson, Faculty of Science, University of Regina, Regina, Saskatchewan, Canada S4S 0A2

REGWT (Wang et al. 1985) analyzes an existing weighting scheme by calculating the average value of $w\Delta^2X$ within ranges of $|F|$, F^2 or I , and $\sin\theta/\lambda (=s)$. REGWT also estimates a modification to the variance of Δ so that $w\Delta^2X$ does not show trends with respect to s or $|F(\text{rel})|$ (Stewart et al. 1976) and applies either the estimated modification to the variance or weights (Prince, 1983).

4.55.1. Introduction

In weighted least-squares refinement the quantity minimized is $\sum w(X) \Delta^2X$ where X is one of the quantities $|F|$, F^2 or I , and ΔX is $|X(\text{rel})-X(\text{cal})|$. The weight, $w(X)$, reflects the accuracy of the measurement of the intensity of a reflection. A properly estimated weight will lead to an accurate crystal structure. The correct value of the weight is the reciprocal of the variance, σ^2X . Usually the contribution of counting statistics to σX is easily calculated, but the contribution from various other sources of error is not.

REGWT provides information to assist in the modification of weights to account for other sources of error. It is equivalent to WTANAL and WTLSQ in XRAY76. REGWT contains an additional feature which allows the calculation of coefficients for the weight modification expression in order to modify the weights. The program does not differentiate between random and systematic error. The procedure is described in detail elsewhere, (Wang and Robertson, 1985). In this program description, w refers to the weight rather than the square root of the weight as it does in some other program descriptions in this manual.

REGWT is used to examine existing weights based on the distribution of $w(X)\Delta^2X$. The data is divided into blocks bounded by both intervals of $X(\text{rel})$ and s . By choosing *either* $X(\text{rel})$ or s the analysis becomes one-dimensional with respect to $X(\text{obs})$ or s . For each grid point, the number of data, the average value of $\{w(X)\Delta^2X\}^{1/2}$ and weighted R index are given.

4.55.2. Normal Weighting Schemes

The weighting schemes used here are based on the program WTLSSQ of the XRAY76 system. Nine weight functions and an opportunity for manipulation of the location of the weights in the bdf are specified on the **normal** line. The weight functions are specified by a weighting scheme number (0-9) and the coefficients (A to I) corresponding to those functions. The default values of these coefficients are all zero. The full weighting schemes are described in detail below.

Scheme 0: Retrieve the old least-squares weight from one of the least-squares weight locations n901, n902, n903 and place it in the least-squares weight location n900, which is used in least-squares refinement in the logical record *lrrefl*: when the bdf weights are specified.

A = 1 : from LSW1(n901) to LSWT(n900)

B = 2 : from LSW2(n902) to LSWT(n900)

C = 3 : from LSW3(n903) to LSWT(n900)

Scheme 1: $W = 1 / (A + B\sigma^2X + C/(\text{oldweight}) + D*X(\text{rel}) + E*X(\text{rel})^2 + G*X(\text{rel})^H + I*\sin\theta)$

Scheme 2: $W = 1 / (A + B*X(\text{rel}) + C*X(\text{rel})^2/\text{oldweight} + D*X(\text{rel})^6 + E*\sin\theta)$

Scheme 3: $W = X * Y$ If $A*X(\text{rel})$ is greater than $|X(\text{cal})|$ then weight is 0.000000001. Let $B = \sin\theta$ limit, then if $\sin\theta$ is greater than B , $X = 1$, else $X = \sin\theta/B$ *B should not be zero*. Let $C = X(\text{rel})$ limit, then if $X(\text{rel})$ is less than C , $Y = 1$, else $Y = C/X(\text{rel})$.

Scheme 4: If $X(\text{rel})$ is less than or equal to A , then $W = (X(\text{rel})/A)^2$, else $W = (A/X(\text{rel}))^2$. Reflections with $X(\text{rel}) = A$ are given maximum weight.

Scheme 5: If $X(\text{rel})$ is less than or equal to A , then $W = 1$, else $W = (A/X(\text{rel}))^2$. Small reflections are given constant weight. In Hughes' original treatment, $A = 4 X(\text{min})$.

Scheme 6: $W = 1 / (1 + ((X(\text{rel}) - B)/A)^2)$ Reflections with $X(\text{rel}) = B$ are given maximum weight, but if A is very large, the weights are constant.

Scheme 7: $W = 1 / (A + X(\text{rel}) + B * X(\text{rel})^2 + C * X(\text{rel})^3)$ Cruickshank suggests that $A = 2 X(\text{min})$, $B = 2 / X(\text{max})$, and $C = 0$ are useful. A larger value of C , say $C = 5 / X(\text{max})^2$, may help down-weight large $X(\text{rel})$ subject to extinction. These values may prove useful in schemes 1 and 2.

Scheme 8: $W = A / \max(\sigma X(\text{rel}), (B * X(\text{rel}) + C), (D * G + E))$ where G is $X(\text{rel})(\text{max})$. For a complete description of this (Univ. of Washington) scheme, see the general section for the program DATRDN of the XRAY76 system.

Scheme 9: $W = A$ This weighting scheme produces constant weights

4.55.3. Estimation Of Weighting Modifications

If the variances of the structure factor amplitudes are correctly estimated, their average value should correspond to $\Delta^2 X$. (The variance will be assumed to represent error in both the experiment and the model; i.e. in $|X(\text{rel})|$ and $|X(\text{cal})|$.) The process of least squares allows some adjustment of the model to errors in the experiment, so that

$$\langle \Delta^2 X / \sigma^2 X \rangle = (N-M) / N$$

rather than unity. N is the number of reflections used for least-squares refinement and M is the number of least-squares variables. $\langle A \rangle$ is the average value of A . The quantity $(N-M)/N$ will be referred to in what follows as the "freedom factor". The calculated variance, $\sigma^2 X(\text{cal})$, is the variance obtained from a knowledge of the experimental conditions and is usually based entirely on counting statistics. A quantity $\sigma^2 X(\text{mod})$, the modified variance, may be added to $\sigma^2 X(\text{cal})$ to give an improved value of $\sigma^2 X$ or $1/w(X)$. The correction, $\sigma^2 X(\text{mod})$, commonly called the "ignorance factor", may be entered by the user as some function such as

$$E * |F(\text{rel})|^2 + I * \sin \theta$$

(see Scheme 1, normal weighting schemes) or it may be estimated automatically. An approximate expression for $\sigma^2 X(\text{mod})$ is

$$\sigma^2 X(\text{mod}) = \{ (\Delta^2 X + VC) - \sigma^2 X(\text{cal}) \} / (\text{freedom factor}) \quad (1)$$

where VC is a correction term including variance (VAR) and covariance (COV) terms.

$$VC = \frac{\Delta^2 X * \text{VAR}(\sigma^2 X) + 2 * \text{COV}(\Delta^2 X, \sigma^2 X)}{\langle \sigma^2 X \rangle^2}$$

This term results from replacing $\langle \Delta^2 X / \sigma^2 X \rangle$ by $\langle \Delta^2 X \rangle / \langle \sigma^2 X \rangle$, and the user may choose to not use it. Also, the "freedom factor" may be replaced by unity. The independent variables are normalized by dividing by $|X(\text{rel})|_{\text{max}}$ and s_{max} . The new independent variables are:

$$V(X) = |X(\text{rel})| / |X(\text{rel})|_{\text{max}} \quad \text{and} \quad V(S) = s / s_{\text{max}}$$

The correction, $\sigma^2 X(\text{mod})$, is estimated by least-squares fitting of the following expression to equation (1)

$$\langle \sigma^2 X(\text{mod}) \rangle = \sum_p (\sum_q \{ [A(p-q, q)] [V(X)^{p-q}] [V(S)^q] \}) \quad (2)$$

If $p=0$, a constant term is determined.

If $p=1$, the coefficients in the expression of the form

$$A(0,0) + A(1,0)*V(F) + A(0,1)*V(S)$$

are determined; etc. The coefficients $A(p,q)$ can then be used to calculate $\sigma^2 X(\text{mod})$ for an individual reflection. The weight for a reflection is then calculated as:

$$1/w(X) = \sigma^2 X = \sigma^2 X(\text{cal}) + \sigma^2 X(\text{mod}) \quad (3)$$

The structure is refined by normal least squares using the modified weights and new values of $\Delta^2 X$ are created. If the individual reflection option is chosen, the variance and covariance correction is not used or calculated. The process is iterated until the coefficients do not change. The change in the standard deviations of least-squares variables on the first iteration will probably be between 0 and 40% of their initial value with non-modified weights. The improvement on the second iteration will typically be 10% of the improvement of the previous one. A third iteration would seldom seem justified. The quantity $\sigma^2 X$ is required for calculating the variance and covariance correction (VC), but is not well known until after first iteration. Therefore $\sigma^2 X(\text{cal})$ must be used instead of $\sigma^2 X$ in the first iteration, if the VC correction is applied.

In the REGWT calculation, the value of $\sigma^2 X(\text{cal})$ in (3) above is obtained from IDN 1900 in logical record *lrrefl*: on the bdf. If IDN 1900 is empty, the value of $\sigma^2 X(\text{cal})$ is then obtained from IDN 130n ($n=1, 3$ or 5 for I, F² or F, respectively). After the modified weight $w(X)$ is calculated, its value will be stored in IDN 1900, replacing the old value of $\sigma^2 X(\text{cal})$.

The square root of the number of reflections in a grid point is used to weight the grid points when fitting equation (2) to equation (1). Not surprisingly, grid points with high $V(X)$ and high $V(S)$ may be empty and others may have few reflections. The option also exists to not average but treat every reflection as a separate grid point. Since averaging to form grid points involves first order differences, the influence of reflections whose contribution to the average in equation (1) deviate far from the mean will be enhanced if individual reflections are used to find the $A(p,q)$. If the individual reflection option is chosen, the variance and covariance correction is not used or calculated.

4.55.4. File Assignments

- Reads reflection data from the input archive bdf
- Optionally writes reflection data to the output archive bdf

4.55.5. Examples

```
REGWT lst wta
maxhkl 12 12 12 0.0 0.824
fgrid 2.3 4.3 7.6 12.4
fgrid 13.4 17.5 20.5 25.5 30.5 35.5
fgrid 45.0 55.5 65. 80. 100. 99999.
sgrid .1 .15 .2 .25 .3 .35 .4 .45 .5 .55 .6 .65
sgrid .7 .75 .80 .85
```

In this example, the weight-analysis process is called to analyze the weights which have been assigned for each reflection. The analysis maps are specified as 16x16 by the **fgrid** and **sgrid** lines. The program does not update the archive bdf. The reflection data will be printed.

```
REGWT nolist wfc 10.0 rel cnt 99.0
regina pow 2 ind fac 1. avc
archiv 1902 -1903
fgrid 3.0 4.0 5.0 6.0 7.0 8.0
fgrid 9.50 12.0 17.0 21.0 27.0 30.0 50.0
fgrid 70.0 999.0
sgrid 0.150 0.200 0.250 0.300 0.350 0.400 0.450
sgrid 0.500 0.515 0.530 0.550 1.000
```

The weight modification function with 2 as the highest power is specified. The estimation of the weights is based on the individual reflection mode. The variance and covariance correction is applied and the least-squares freedom factor is 1. The weight in location 1903 will be deleted from the binary data file and the old weight will be stored in the weight location 1902. Only the observed reflections are used excluding the reflections for which the value of $\sigma(F)$ is greater than 99.0. The output analysis map (15x12) will be scaled by the factor 10.

```
REGWT nor
normal 0 b 2
archiv -1903
```

A normal weighting scheme is indicated in the REGWT line. The scheme number is specified as 0, which transfers the weight saved at location 1902 to location 1900 where it will be used for weighted least-squares refinement. The old weights at location 1903, if they exist, will be erased from the bdf.

References

- Wang, Hong and Robertson, B.E. 1985. *Structure and Statistics in Crystallography*. Ed. A.J.C. Wilson. Adenine Press: New York.
- Stewart, J.M., Machin, P.A., Dickinson, C.W., Ammon, H.L., Heck, H. and Flack, H.D. 1976. *The XRAY76 SYSTEM*. Tech. Report. TR446, Computer Science Center, University of Maryland.
- Prince, E. 1982. *Mathematical Techniques in Crystallography and Materials Science*. Springer-Verlag: New York.
- Johnston, R.L. 1982. *Numerical Methods -- A software approach*. John Wiley: New York.

4.56. **RFOURR**: Reverse Fourier transform

Authors: Doug Collins, Jim Stewart & Jim Holden

Contact: Jim Stewart, Department of Chemistry, University of Maryland, College Park, MD 20742, USA

*RFOURR transforms a complete unit cell of sampled points in direct space (e.g. an electron density map) to a unique asymmetric set of structure factors in reciprocal space. The resolution of the map in direct space determines the extent of the reflections for which the structure factor components A, B and the phase will be computed. The input map may be prepared by use of **FOURR** or by any of the electron density programs of the XTAL system.*

4.56.1. Purpose

RFOURR is a reverse Fourier transform program which produces structure factor magnitudes and phases from an input electron density map. The electron density map may be prepared in a number of ways. The fastest, and least useful, is by means of the XTAL program *FOURR*. In this case the electron density is simply transformed back into the A and B which produced the original map. On the other hand FFOURR will often be the source of the electron density map which will be filtered by some means to improve the quality of the map.

4.56.2. Grid Restrictions

RFOURR uses a set of (fast) Winograd Fourier transform subroutines (Silverman, 1977) to carry out the transformation. The subroutines are restricted to certain grid parameters. These are given in Table 1. When the input map is generated these restrictions must be observed. In addition the input map must contain the density of the whole unit cell. For example, if the cell direction is divided into 30 grids then points 0 through to point 29 only must be present in the input map.

Table 1: Grid sizes allowed for in RFOURR

2	3	4	5	6	7	8	9	10
12	14	15	16	18	20	21	24	28
30	35	36	40	42	45	48	56	60
63	70	72	80	84	90	105	112	120
126	140	144	168	180	210	240	252	280
315	336	360	420	504	560	630	720	840
1008	1260	1680	2520	5040				

4.56.3. Sort Order Of Map And Data

The last restriction is that the summation order of the map must be consistent with the sort order of the reflections on the bdf. The algorithm in RFOURR depends upon the sort order of the reflections and the summation order of the input density map. Table 2 gives the relationship between the two parameters in terms of the instructions used for *SORTRF* and *FOURR*.

Table 2: Reflection Order Related to Map Directions

<i>Reflection order</i> <i>slow/med/fast</i>	<i>Map directions</i>		
	b	c	
hk1	across	down	layer
h1k	across	layer	down
kh1	down	across	layer
k1h	layer	across	down
1hk	down	layer	across
1kh	layer	down	across

If the sort order is not established by the *SORTRF* program it may be specified on the **RFOURR** line. In addition to the sort order another related parameter is required. They are a set of signals which indicate whether the miller indices on the bdf are negative. This information is stored on the bdf by *SORTRF* or may be explicitly stated on the **RFOURR** line. The `sort 3` option must be used in *SORTRF*. *Note that the slow changing index must have no negative indices.* That is for triclinic the hemisphere of data is always +++, -+-, +-+, --- when a **1hk** or **1kh** sort has been specified. For monoclinic only +++ and -+- would be present. For orthorhombic only

+++ would be present on the file.

The most efficient order for a Fourier transform is over the short axis first (across), medium length axis next (down) and finally the long axis (layer). This produces the fewest groups of reflections and results in the most efficient Fourier transformations.

4.56.4. Symmetry Considerations And Algorithm

RFOURR depends upon the input map being of the entire unit cell (no more, no less) with all symmetry equivalent points having identical electron density. The XTAL system stores only the unique asymmetric set of reflections in the bdf. The following algorithm is employed. As each line of the input map is loaded it is transformed, $\rho(x_1, x_2, x_3) \rightarrow G(x_1, x_2, h_3)$. The second sum coefficients G are stored until one layer, all x_2 for a given x_1 , has been treated. The layer is transformed for all h_3 values to produce $P(x_1, h_2, h_3)$ coefficients of the third sum direction. This is the massive array required by the FFT algorithm. The third direction summation is driven by the input bdf, resulting in only the phases of the unique asymmetric set of reflections within the resolution of the input map being calculated. In this summation the $F(h_1, h_2, h_3)$ is produced and stored on the output bdf as $A(h_1, h_2, h_3)$, $B(h_1, h_2, h_3)$ and $\alpha(h_1, h_2, h_3)$.

4.56.5. Control Of Input & Output Items

The RFOURR calculation produces four structure factor coefficients: the calculated magnitude (CAL); the phase (PHI) and the complex components (A & B). These coefficients are inserted into the *lrrefl*: packets of the output bdf. In the simplest case these coefficients CAL, A, B and PHI will be the standard structure factor items represented by the ID numbers n800, n801, n802 and n700, respectively. In other cases, the ID numbers will be determined by the type of density map, and the nature of the filtering that it has undergone. For the non-standard case the user may define the ID numbers of the output items with the *idnums* line.

The ID number of the input coefficient (OBS) may also be defined on this line. The default is Frel with dispersion removed (n701), or if this is not present, Frel (n304). Note that the input coefficient is only important if the results are to be printed (see the *pr* option on the **RFOURR** line) and the calculated R -factor (see below) is to be meaningful.

Map Resolution And Precision

An R factor $\sum ||OBS| - CAL| / \sum |OBS|$ is calculated during the transformation. This R factor is useful if the input density has been filtered. For an input map prepared by *FOURR* the result is of academic interest since it should be zero (OBS \rightarrow density \rightarrow OBS results in no differences in OBS). If, however, there is a reduction in resolution by restricting the fineness of the grid during FOURR then the resulting OBS produced by the back transform will be distorted and a higher R value will result.

When RFOURR is applied the resolution of the input density map will determine the extent of the reflections for which phases are determined. In order that every reflection be phased it is necessary that the input map have a resolution of at least $2h_{\max}$, $2k_{\max}$ and $2l_{\max}$ grid points respectively in the x,y and z directions. Anything less will result in the high order reflections being left unphased. Output items for these reflections are set to *voidflg*:

To avoid the indeterminacy associated with phasing structure factors where *both* A and B are very small, the user may define a threshold via the *th* option on the **RFOURR** line. The default value is 0.005. Items for reflections below the threshold, are set to *voidflg*. Similarly, if a reflection is outside the limits defined by the *maxhk1* line, or the values from *lrdset*: of the input bdf, the output items will be set to *voidflg*:

4.56.6. File Assignments

- Reads reflection data from the input archive bdf

- Writes output items to the output archive bdf
- Reads density map from the bdf `map`

4.56.7. Examples

```
RFOURR lhk
```

This is a standard run. The inclusion of `lhk` is only required if *SORTRF* has not been used to sort the reflection data on the bdf. If SORTRF has not been applied the `lhk` indicates the sort order l slow; k fast is how the data is supplied (the program will fail if the reflection data is not sorted -- you are strongly advised to use *SORTRF*). The output coefficients are Fcal, Acal, Bcal and ϕ . No reflection data is printed.

```
RFOURR ds 2 th .5 print 100
idnums 900 950 951 952 953
```

This is a special run using reflections from dataset 2. The input comparison coefficient has the idnum 2900; the output coefficients CAL, A, B and PHI are stored as 2950, 2951, 2952 and 2953. The first 100 reflections are printed and only reflections with A and B > 0.5 will be phased. The sort order of the reflection data is extracted from *lrdset*: and checked against the map orientation.

4.56.8. References

- Silverman, H.K. 1977. *Introduction to Programming the Winograd Fourier Transform Algorithm (WFTA)*. IEEE Trans. Acoust., Speech, Signal Processing, 25, 152-165

4.57. *RIGBOD*: Generate a Rigid Group

Author: Roeli Olthof-Hazekamp, Laboratorium voor Algemene Chemie, Padualaan 8, 3508 TB Utrecht, The Netherlands

RIGBOD converts coordinates of an idealized group of atoms into fractional coordinates properly located in a given unit cell. The idealized fractional coordinates are rotated and translated within the unit cell to coincide with a poorly resolved, unrefined, or even partial structure, derived from the electron density.

4.57.1. Calculations Performed

An idealized set of coordinates must be supplied for the rigid group to be placed in logical record *lratom*: of the bdf. Typically the rigid group will be molecular fragments with no permitted internal rotations, e.g. phenyl rings. There is no check made on the group presented so the user may give any set of coordinates desired.

Coordinates of atoms may be presented for calculation either as fractional coordinates or orthogonal Angstrom coordinates. The latter are useful when the rigid group is calculated by hand from a model. The former may be altered by hand from the coordinates obtained from a Fourier map or from use of another program such as *BONDAT*. Whichever source is used for the ideal group, it is necessary to specify how some atoms of the ideal group are related to atoms of the "rough" or "natural" structure as far as it is known.

The atoms used in the calculation will fall into two subgroups; real and ideal. The real atoms are atoms of the partially or fully solved structure while the ideal atoms are atoms of the idealized group. The coordinates of the atoms of both subgroups must be in the same region of the unit cell and must be converging toward a connected set. The term "connected set" is defined in the *BONDLA* program. Provisions have been made for modifying coordinates of atoms in a partially or fully solved structure to assure that this condition is met. The program *BONDLA* will search a set of atoms for a connected asymmetric set. If the idealized group lies in a special position in the unit cell so that some of the atoms will be related by a symmetry operation, provision has been made to specify that this condition exists for the appropriate atoms.

An approximate fit of the ideal group to the real group is obtained by using the following formulae.

4.57.1.1. Orthogonal coordinates (matrix notation):

$$\mathbf{x}' = \mathbf{A} \mathbf{x}$$

where \mathbf{A} is the matrix to transform fractional coordinates to orthogonal coordinates (*lrcell*., packet 4; relative to orthogonal axes a^* , b^* , and c^*); \mathbf{x}' are the orthogonal coordinates; and \mathbf{x} are the fractional coordinates.

4.57.1.2. Matrices for group-based unitary systems EI(ideal) and ER(real):

- The first 3 atoms will be used; they must not be co-linear.
- \mathbf{u} is vector from atom 1 to atom 2
- \mathbf{v} is vector from atom 1 to atom 3
- \mathbf{w} is perpendicular to \mathbf{u} and \mathbf{v} ($\mathbf{u} \times \mathbf{v}$)

- $\mathbf{E1} = \mathbf{u}$ (normalized)
- $\mathbf{E2} = \mathbf{w} \times \mathbf{u}$ (normalized)
- $\mathbf{E3} = \mathbf{w}$ (normalized)

Both systems are normalized.

4.57.1.3. Rotation matrix (ideal to real system)

$$\mathbf{R} = \mathbf{ER}^\dagger \mathbf{EI}$$

where \mathbf{R} is the rotation matrix and \mathbf{ER}^\dagger is the transpose of \mathbf{ER} .

4.57.1.4. The idealized coordinates

$$\mathbf{x}'(\text{idealized}) = \mathbf{R} \mathbf{x}'(\text{ideal}) + \mathbf{x}'(\text{atom 1})$$

The idealized coordinates are shifted so that the sum of the x , y , and z derivatives between idealized and real atoms is zero.

4.57.1.5. The idealized fractional coordinates

$$\mathbf{x}(\text{idealized}) = \mathbf{A}^{-1}\mathbf{x}'(\text{idealized})$$

where \mathbf{A}^{-1} = matrix to transform orthogonal coordinates to fractional coordinates (record *lrcell*:, packet 5)

The atoms of the *real* subgroup are loaded from the archive bdf. When groups of atoms are in special positions, pseudo-atoms may be added to the real subgroup by means of **atomsy** input lines. These atoms are used to get the proper orientation of the rigid group.

The atoms of the *ideal* subgroup must be loaded from input lines. Ideal atoms loaded as **atomor** lines have orthogonal Angstrom coordinates referred to an arbitrary origin. Ideal atoms may also be supplied as fractional coordinates on **atomfr** input lines. A **cellid** line must precede the **atomfr** lines and contain the unit cell parameters which define the **atomfr** coordinates. The origin is again arbitrary.

The correspondence between atoms in the two subgroups is established through the labels of atom sites from the bdf, **atomsy**, **atomor**, and/or **atomfr** input lines. Any atoms in the ideal subgroup not identified with ones in the real subgroup are assumed to be undetermined. These atoms will be appended to the real subgroup at the end of the fitting process if **atname** lines have been supplied to establish atom labels in the real subgroup.

If a structure possesses many copies of the idealized group, it may be replicated as many times as required in various locations in the unit cell by use of **atname** input lines. These lines allow the renaming of atoms in the idealized subgroup so they will match others in the real subgroup. The number of atoms specified in **atname** lines must, in each instance, be the same as the number of members in the ideal subgroup.

The **rigid** line permits the establishment of an isotropic thermal displacement parameter, population parameter, and multiplicity factor for the members of the generated group(s). This program has been adapted from the RIGBOD program of XRAY76 which was written by N.W. Alcock and adapted by Schwaba (1976).

4.57.2. File Assignments

- Reads atom data from the input archive bdf
- Outputs **atom** lines to the line file **pch**

4.57.3. Example

```
RIGBOD p1
RIGID
atomor C1  1.212 0.70 0.0
atomor C2  0.0  1.40 0.0
atomor C3 -1.212 0.70 0.0
atomor C4 -1.212 -0.70 0.0
atomor C5  0.0 -1.40 0.0
atomor C6  1.212 -0.70 0.0
RIGID
atname C11 C12 C13 C14 C15 C16
```

Two phenyl groups are generated, the first one fitting atoms C1 to C6, the second one fitting atoms C11 to C16. Real atoms are read from the bdf and lines with fractional coordinates are written to the punch file.

4.57.4. References

- Alcock, N.W. and Schwaba. 1976. *RIGBOD The X-ray System of Crystallographic Programs*. TR-446, 180A University of Maryland, Computer Science Center.

4.58. **RMAP**: Search for translated fragment

Authors: William Furey and Jim Stewart

Contact: Jim Stewart, Department of Chemistry, University Maryland, College Park, MD 20742 USA

RMAP calculates R factors or correlation coefficients as a function of the translated position of a molecular fragment in the unit cell. In addition to the translated fragment there may also be a fixed fragment. Control is provided for selecting reflections by resolution magnitude of Fobs, or numbers. The step size and range the unit cell to be used in the search also may be specified.

4.58.1. Introduction

RMAP is used to calculate maps analogous to electron density maps where each point in the map is a function of $1/R$, where R is the conventional R factor ($\sum \Delta F / \sum F_{obs}$). The use of $1/R$ forces the coordinate of best agreement to be on a maximum. In addition to $1/R$ maps it is possible to calculate correlation maps (referred to as C maps), or a linear function of $1/R$ and the correlation. The correlation C function is

$$C = \sum (F_{obs} F_{cal}) / \{ \sum (F_{obs})^2 \sum (F_{cal})^2 \}^{1/2} \text{ at each grid point.}$$

The advantage of C is the fact that it is independent of the Frel scale factor. It is not as "sharp" a function as $1/R$, however. It also requires more storage for the additional sums. The maps will show a maximum at the optimum translational position of a fragment in the unit cell. A grid, just like the grid in a regular Fourier transform, is set up in a specified region of the unit cell and structure factors are calculated for a selected subset of reflections at each of the grid points. The values of the calculated structure factors are used to calculate R or C or a linear combination of R and C, at each grid point. A linear combination which gives numbers on a convenient scale has been defined as:

$$D = \{10(1-W)C+1\} \{ (W/R)+1 \}$$

The values of D are written to a MAP file in the same format as that produced by *FOURR*. MAP may be used with *PEKPIK* to search for the maxima. The highest maximum is reported in the RMAP output. The factor W is a factorial weight. If $W=1$, only R contributes to D, while if $W=0$, only the correlation coefficient contributes to D. When $W=1$ is used, 1/3 the amount of storage is required.

It is not usually necessary, or desirable, to include all reflections in the search procedure. Much time can be saved if only the strong reflections, at an appropriate level of resolution, are used in the promotion of D values. It may be necessary to include a small number of weak reflections to control the scale factor. Perhaps a better method for this is to use *GENEV* with all data to establish a physically realistic Frel scale factor before using RMAP. It is useful to start with low resolution data and a coarse grid. The program is set to use 0.3 Angstrom steps over the whole unit cell with all reflections included in the calculation as default parameters.

Initially when no positional information is available the number of atoms in the fixed fragment will be zero and the calculation will be made only on the translated fragment. After a fragment has been positioned, the first fragment may be declared as fixed and any additional fragment atoms can be designated as translated. This feature allows molecules to "grow" and allows use of the program where there is more than one molecule in the asymmetric portion of the unit cell.

4.58.2. Algorithm

In the initial pass over the bdf data the contributions of all the atoms to the structure factor components A and B, as a function of the symmetry operations, are saved as separate quantities. The selected reflections are written to a scratch file. On the second pass the structure factors are calculated as a function of all the grid points. This is accomplished by using the sums of A and B at each symmetry point and the trigonometric identities:

$$\cos(x+y) = \cos x \cos y - \sin x \sin y$$

$$\sin(x+y) = \cos x \sin y + \sin x \cos y$$

From these identities each step y can be applied to each geometrical A and B part of the structure factor ($\cos x$, $\sin x$) and the complete structure factor at the step formed by summing only over the number of symmetry operations and not over the number of atoms in the fragment.

4.58.3. Data Required

The input bdf must have F_{rel} or F_{rel}^2 , and a set of atomic coordinates with an overall or with individual isotropic thermal displacement parameters. The atoms must be loaded (with *ADDATM*) as one or two subsets. The first subset, which may be null, contains the atoms of a fragment which is to remain fixed during the search. The second subset (which must not be null) consists of the atoms to be translated during the search. *Atom sites are listed by entering*

```
reset psta 4
```

4.58.4. File Assignments

- Reads reflections and atom fragments from the input archive bdf
- Writes a pseudo electron density map to the bdf_{map}

4.58.5. Example

```
RMAP iso map weight 0.1
bounds 0 .125 .0078 0 .125 .0078 0 .125 .0078
PEKPIK
plimit *3 12 80
```

A compound with cell dimensions $a=20.515$, $b=21.31$, $c=17.128$ is searched over 1/512 of the unit cell near the origin, in steps of 1/128 of each cell edge. The resulting RMAP is then searched for maxima. The D function calculated is weighted by 0.1 in terms of 1/R and by 0.9 in terms of C.

4.59. *RSCAN*: Analyse R-factors

Authors: George Davenport and Syd Hall

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

RSCAN analyzes structure factor residuals and ratios in terms of Miller indices, measured structure factors, measured intensities (if present on the bdf), and $(\sin\theta/\lambda)^2$. *RSCAN* calculates multiplicative corrections to scale factors, the thermal displacement parameter shift, and the overall R-factor. It is an adaptation of the XRAY program RLIST (Hall, 1972).

4.59.1. Program Calculations

Data are accumulated separately for reflections with `rcode=1` and `rcode=2`, for each of the six quantities h , k , l , measured F , s^2 , and measured I . Parameters on the **RSCAN** line are used to determine the range and the interval size for each quantity.

4.59.1.1. h, k, l

Each interval is a single integer value. The range is from $-N$ to $+N$, as specified on the **maxhk1** line.

4.59.1.2. $F(\text{rel})$

The range is divided into two sub-ranges. The first goes from 0 to $max1$ with intervals of uniform size $int1$. The second goes from $max1$ to $max2$ with intervals of size $int2$. The defaults for $max1$, $max2$, $int1$ and $int2$ are 19.0, 500.0, 1.0, and 10.0, respectively.

4.59.1.3. $\sin\theta/\lambda$ (s)

The range extends from a value of 0 to the maximum value with intervals of uniform size.

4.59.1.4. $I(\text{rel})$

The range extends from 1 to a maximum. The interval boundaries are determined by the *factor* on the **RSCAN** line. The boundaries are 1.0, *factor*, $factor^2$, ... , up to the specified maximum. Note that R-factors for $I(\text{rel})$ will be calculated only if the $I(\text{rel})$ is present in the bdf. Note also that *factor* must be greater than 1.0.

4.59.1.5. Calculated Quantities

For `rcode=1` and for `rcode=2`, for each of the six quantities, for each interval, the following quantities are determined. All reflections with `rcode>2` are ignored.

- Number of reflections within that interval
- Mean value of $F(\text{rel})$ for reflections within that interval
- Mean value of $F(\text{cal})$ for reflections within that interval
- Mean value of $F(\text{cal})/F(\text{rel})$ for reflections within that interval
- Mean value of $|F(\text{rel})-F(\text{cal})|$ for reflections within that interval

4.59.2. File Assignments

Reads reflection data from the input archive bdf

4.59.3. References

- Hall, S.R. 1972. *RLIST. The X-RAY System of Crystallographic Programs for any Computer TR-192, 223.*
- Wilson, A.J.C. 1942. *Determination of Absolute from Relative X-ray Intensity Data.* Nature 150, 151.

4.60. **SCATOM**: Calculate pseudo-atom form factor. **NEW**

Authors: Doug du Boulay

Contact: Doug du Boulay, Materials and Structures Laboratory, Tokyo Institute of Technology, Nagatsuta, Midori Ku, Yokohama, Japan.

SCATOM takes X-ray scattering structure factors from a single element crystal structure and deconvolutes them into the elemental form factor on a grid of $\sin\theta/\lambda$ values.

4.60.1. Overview

SCATOM was written to try and obtain form factors accounting for the valence density of neutral, noninteracting atoms. Those form factors could allow the calculation of general theoretical valence density structure factors and thereby difference valence density maps, which highlight the changes to the calculated valence density arising from chemical interactions. Such maps ought to be comparable to experimentally determined difference (total) density maps.

As yet, it hasn't proven so useful.

4.60.2. Purpose

Abinitio density functional calculations of valence electron density using plane wave solutions and pseudo-potential approximations are frequently seen in the literature. These valence densities are not experimental observables in elastic X-ray scattering experiments because the X-rays are scattered by the total electron density. It is therefore quite useful, for comparative purposes, if the valence densities can be transformed to difference valence densities for direct comparison with experimental charge density results.

The difference valence density can be calculated either by subtracting the standard crystallographic promolecular density² from the theoretically calculated total density directly in real space, or alternatively by Fourier transforming the reciprocal space structure factor differences. Being a crystallographic data analysis program Xtal is well suited to the latter option.

There are two different approaches to calculating the difference density. One is to add the neglected core electron density contribution back to the theoretical valence density and then subtract the promolecular density from the total theoretical density, in the manner of a standard crystallographic difference density analysis. The second approach, which this program could assist towards, is to use the theoretical valence density of a lattice of superposed but noninteracting equivalent neutral pseudo-atoms, i.e. a valence density analogue of the promolecule, and subtract that from the compound whose valence density was actually being calculated, either in real space or Fourier transformed from reciprocal space.

In the former approach there is some degree of uncertainty as to exactly what form the core density should take. Generally it will resemble that of an inert gas, depending strongly on the pseudo-potentials actually used. However, the radial density distribution could differ markedly from that approximation, and it is probably not easily characterised.

The alternate approach is to use exactly the same machinery used to calculate the abinitio valence density being studied, to calculate individually, the spherically symmetric neutral pseudo-atom valence densities to be subtracted from those in those actually calculated abinitio. If the valence density of a single isolated atom is reverse Fourier transformed in to structure factors, then this program **SCATOM** can calculate the $\sin\theta/\lambda$ form factor curve of a single pseudo-atom required to reconstruct the promolecular-valence density of the extended lattice.

4.60.3. Implementation

4.60.4. Examples

Calculate the pseudo-atom form factor of Si

```
compid si      : using a silicon lattice

MAPXCH abi2map noterm abifile si_o_DEN : read ABINIT map
UOV 0.22
:
:           The file si_o_DEN contains the spherically
:           symmetric non-interacting neutral atom
:           valence density calculated based on a
:           Troullier-Martins pseudo-potential

ADDATM upd
UOV 0.0000 : force to zero for SCATOM

ADDRF      : Generate dummy reflection data
reduce nocon
:limits *9 yes : for whole sphere - assuming MAPXCH wavelength?
hklgen hkl  frel sigf

SORTRF aver 1 frel : Remove redundant reflections

RFOURR th 0.000001 : Reverse fourier transform valence density
idnums $1 304 801 802 700

MODHKL : Purge unphased reflections
purge -4.+19 $1 1 1801

SCATOM ove : Calculate Si-pseudo-atom form factor.
finish
```

4.61. *SHAPE*: Patterson deconvolution for structure solution

Author: Michael A. Estermann

Contact: Michael A. Estermann, Laboratorium für Kristallographie, Eidgenössische Technische Hochschule, ETH-Zentrum. CH-8092 Zurich, Switzerland.

SHAPE deconvolutes a Patterson map by utilising Patterson superposition methods in a voxel-wise manner and writes the deconvoluted map to the "map" file.

4.61.1. Overview

In general, Patterson methods favour a small number of "outstanding" scatterers which dominate the vector map. Typically this is one or more heavy atoms in the presence of light atoms. In fact, the heavy-atom technique for structure solution is rather robust with incomplete and resolution-limited diffraction data such as from macromolecular single-crystals or polycrystalline samples, where limits in attainable resolution (min d-spacing) or the number of intensities suffering from reflection overlap can seriously degrade the quality of the diffraction data.

In fact, the position of one or more heavy atoms can often be solved by hand from the Patterson map. But the identification and analysis of the interatomic vectors by hand becomes increasingly difficult with Patterson maps of lesser quality and structures of higher complexity. Interpreting the cross-word puzzle of interatomic vectors can become cumbersome and may not discriminate sufficiently between atomic positions. In contrast, *SHAPE* utilises all the possible atomic positions for a given map, and not just the identifiable Patterson peaks. The program *SHAPE* offers an automatic and objective way to quantify the likelihood of a single atom sitting on a particular position by means of the symmetry minimum function.

4.61.2. Improving the interpretability of the patterson map

Limits of attainable resolution (min d-spacing) will affect both the resolution of atomic peaks in the crystal unit cell as well as peaks in the Patterson cell. The width of Patterson peaks will also increase with increasing thermal motion. Consequently, peak positions are less reliable or may be lost altogether. The interpretability of the Patterson map can be improved by either sharpening (Patterson, 1935), origin removal (Karle and Hauptmann, 1964) or maximum entropy image reconstruction techniques (David, 1990).

In the XTAL system, it is strongly recommended to use the "epat" option when calculating the Patterson map with the program FOURR. The "epat" option offers a good compromise between (a) sharpening the Patterson and (b) keeping the Fourier series truncation effect under control.

4.61.3. Patterson superposition methods

In the early 1950's, systematic Patterson vector-search techniques for structure solution were developed which were based on the superposition of shifted Patterson maps. The symmetry minimum function (SMF) is a natural extension of these methods (Kraut, 1961; Simpson, 1965). The SMF includes the space group symmetry and uses the entire Patterson map rather than just the identifiable peaks. The SMF can actually be justified on rigid Bayesian and statistical grounds (Bricogne, 1992). The program SHAPE is a computer implementation of the ideas of Kraut (1961) and Simpson (1965) in direct space.

For the applications of SHAPE to structure solution from powder diffraction data, see, for instance, Estermann (1995) and Hofman et al. (1995).

4.61.4. The symmetry minimum function, automatically evaluating harker vectors

The symmetry minimum function is defined as

$$SMF(r) = \min_{i=1}^p (1/m(i)) P(r - C(i)*r)$$

where a trial atomic position r in the crystal cell is ranked by comparing the heights of all unique Harker vectors $H(r) = r - C(i)*r$, ($i=1, \dots, p$) in the Patterson cell (the origin vector is not included). The symmetry operator of the space group is given by $C(i)$ where $C(i)*r = R(i)*r + T(i)$ is the operator applied to a position r , $R(i)$ is the rotational and $T(i)$ the translational part of the symmetry operator. The multiplicity of the Harker vectors is given by $m(i)$. For every grid point in the input map, the co-ordinate r is calculated and the value $SMF(r)$ derived. The minimum function in the equation for the SMF ensures a continuously low value for any position r unless all Harker vectors $H(r)$ are above background.

The enantiomorph structure and the structure shifted by a permissible origin shift have exactly the same vector set (homometric structures). Consequently, all the vector sets of these structures are correct solutions of the Patterson map and are present in the SMF. Therefore the maxima in the SMF are single-site solutions, since the different maxima may not relate to the same origin. Therefore the maxima cannot be used directly as input for a structure refinement.

In the next step, a single image will be selected from the Patterson map with the help of the image seeking minimum function.

4.61.5. The image seeking minimum function, automatically evaluating cross-vectors

To retrieve a single image of the structure from the Patterson map, it is necessary to search the Patterson map for vectors between atoms which are not related by symmetry, the so-called cross-vectors. A maxima from the SMF map at position r' is selected as an origin fixing atom. This fixed pivot position r' , and its

symmetry related copies $C(i)*r'$, are then tested against all other possible atomic positions r in the crystal cell by analysing the height of all cross vectors $r - C(i)*r'$ ($i=1, \dots$, number of symops) with the minimum function:

$$IMF(r) = \min_{i=1}^n P(r - C(i)*r')$$

It is named the image seeking minimum function (IMF) because it locates a single image out of all the ambiguous origin-shifted images. One of the two enantiomorphic structures may be eliminated by using a second pivotal position.

4.61.6. Hints

The size of the input Patterson map defines the size of the output map. Even, if only a partial map is used, e.g. an asymmetric unit (ASU), the backmapping of vectors outside the ASU is correctly done. However, the ASU of the Patterson map is not necessarily the same as the one of the image seeking minimum function. It is therefore strongly recommended to use the "full" option in the program FOURR in order to work with a complete unit cell.

If the symmetry of the output map does not agree with the symmetry of the space group then it is probably due to an unsuitable grid division. Preferably use multiples of 2, 3, 4, 6 as grid divisions in the program FOURR.

4.61.7. File Assignments

- Reads a Patterson map from file `map`
- Writes the map calculated with symmetry minimum function to the file `smf`
- Writes the map calculated with the image seeking minimum function to the file `imf`

4.61.8. Example

Deconvolution with the symmetry minimum function, Harker vectors

```
: calculate an |E*F| Patterson map, full cell, add I(000) term
FOURR epat full
grid 48 48 152
layout layer down across
fzero 5332.
end
: deconvolute voxel-wise with the symmetry minimum function
: inspecting Harker sections only
: single-sites
SHAPE
smf
end
: copy the deconvoluted map "smf" onto standard input "map" for PEKPIK
COPYBDF smf map
: search for maxima in the SMF map
: for a large number of maxima, adjust the parameters in PEKPIK
PEKPIK punch
plimit *2 0.1 20 0.4
end
```

Deconvolution with the image seeking minimum function, cross vectors

```

: calculate an |E*F| Patterson map, full cell, add I(000) term
FOURR epat full
grid 48 48 152
layout layer down across
fzero 5332.
end
: select a strong peak from the previous SHAPE run (smf option)
: inspecting all possible cross-vectors for a given atom position
: selects a single image from the Patterson map
SHAPE
imf .7500 .0104 .7434 : pivot selected from PEKPIK output
end
: copy the deconvoluted map "imf" onto standard input "map" for PEKPIK
COPYBDF imf map
: search for maxima in the IMF map
: PEKPIK maxima can be used as input for
PEKPIK punch
plimit *2 0.1 20 0.4
end

```

4.61.9. References

- Bricogne, G. (1992). Molecular Replacement. Proceedings of the CCP4 Study Weekend, compiled by E. J. Dodson, S. Gover, and W. Wolf, pp. 62-75. Daresbury Laboratory Publications.
- David, W. I. F. (1990). Nature (London), 346, 731-734.
- Estermann, M. A. (1995). Nucl. Instr. and Meth. in Phys. Res. A 354, 126-133.
- Hofmann, M., Schweda, E., Strähle, J., Laval, J.P., Frit, B. and Estermann, M. A. (1995). J. Solid State Chem. 114, 73-78.
- J. Kraut. (1961) Acta Crystallogr. 14, 1146-1152.
- Karle, J. and Hauptmann, H. (1964). Acta Cryst. 17, 392-396.
- Patterson, A. L. (1934). Phys. Rev. 46, 372-376.
- Patterson, A. L. (1935). Z. Kristallogr. 90, 517-542.
- Simpson, P. G., Dobrott, R. D. & Lipscomb, W. (1965) Acta Crystallogr. 18, 169- 179.
- Wrinch, D. M. (1939). Philos. Mag. 27, 98-122.

4.62. *SHELIN*: Enter SHELX file

Author: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

SHELIN reads a standard Shelx (Sheldrick, 1986) input file and extracts those commands and data required to perform the standard data initialization calculations *STARTX*, *ADDREF* and *ADDATM*. Any other XTAL calculations using the archive bdf generated by these routines may directly follow the *SHELIN* input. As well as the archive bdf, *SHELIN* generates a line file for subsequent applications.

4.62.1. Recent Modifications

SHELIN originally worked by including the `SHELX.ins` instruction file inline within the Xtal `.dat` input stream. To aid and facilitate data exchange, this behaviour is now activated only by use of the `old` command line option. The new default behaviour is to read external `.ins` and `.hkl` files, although thereafter the behaviour is largely as documented below. Any SHELX dummy Q atom sites are now written as peaks to a `compid.pek` file. These can then be viewed within the Xtal molecular graphics program PIG.

4.62.2. Shelx Commands And Data

All Shelx commands and data may be input into SHELIN but only the following commands are used to generate XTAL input data. Shelx commands and data not listed below will be ignored.

```
TITL          (optional; the TITL header may be omitted)
CELL
LATT          (optional if lattice is primitive noncentrosymmetric)
SYMM          (any number; SYMM X,Y,Z is not entered)
SFAC          (any number; but no Cromer-Mann coefficients allowed)
UNIT          (any number)
```

The lines `CELL` to `UNIT` *must* be entered first, and in this order. They are used to initialize the `STARTX` calculation.

```
HKLF -3 code
h k l F  $\sigma$ F          (reflection data in either format)
```

The `HKLF` line is used to initialize the [ADDREF](#) calculation. If `code` is blank the format of the reflection data is assumed to be '3I4,2F8.2'. If `code` is nonblank the format is assumed to be '3I3,F7.2,F6.2'. The reflection data input sequence is terminated by a reflection with $h=k=l=0$ or $h=k=l=99$.

```
FVAR K          atom-name sfacN x y z ....          (std. atom line(s))
```

The `FVAR` line is used to initialize the [ADDATM](#) calculation. The parameter `K` is the Frel scale factor. Both formats of the standard atom lines may be entered in any order (either with an isotropic `U` (one line) or with anisotropic `U`'s (two lines)). If anisotropic `U`'s are entered, it is assumed that `U11` and `U22` are on the first line followed by an '='. An `END` line must follow the last atom line, *and* this `END` must be the last line in the Shelx sequence.

```
END
```

The use of the `END` line in a Shelx sequence has a special meaning for SHELIN. It is assumed to close the SHELIN sequence. Be careful to remove any superfluous `END` lines from the SHELIN file.

4.62.3. Function

SHELIN is a relatively special program within the XTAL System in that it both *creates* the input data for [STARTX](#) (and optionally [ADDREF](#) and/or [ADDATM](#)), and then directly *initiates* the execution of these routines.

The first stage of SHELIN is to read the SHELX commands and data from the standard line file and converts these into the equivalent XTAL control and data lines. These are output to the formatted file `shx`. Only the selected SHELX commands and data lines described above are interpreted.

The second stage of SHELIN is to redirect the line input stream from the file `shx`. This means that after the `END` line terminating the SHELIN lines, the nucleus starts to input the generated XTAL lines from `shx` and these are used to control the XTAL calculations `STARTX`, `ADDREF` and `ADDATM`.

The second last line in the `shx` file is the command `'reset inp <ioinp:>'`. This command causes the line input stream to be returned to normal line input, so that additional XTAL commands may be executed. It should also be noted that the lines in `shx` may also be used in subsequent XTAL applications without having to use the SHELIN sequence.

4.62.4. File Assignments

- A SHELIN run sequence writes and reads archive `bdf`'s
- Reads a SHELX `ins` input file
- Optionally reads a SHELX `hkl` reflection file
- Writes XTAL input lines to the line file `shx`
- Optionally writes dummy sites to the Xtal binary `pek` file

4.62.5. Example

```

SHELIN
TITL IK 9/4 PHOTOREAKTIVES BUTENOLID RG PBCN 8/88
CELL 1.54179 15.180 7.413 22.607 90 90 90
SYMM .5-X, .5 -Y, .5+Z
SYMM -X, Y, .5-Z
SYMM .5+X, .5-Y, -Z
SFAC C H O
UNIT 128 128 16
BOND 1
FMAP 2          L.S. 4
HKL  -3
2 0 0 24.99 -7.66 1
4 0 0 20.51 -6.52 1
6 0 0 23.06 3.47 1
8 0 0 19.46 2.48 1
10 0 0 15.21 1.70 1
.....
.....
6 8 0 -1.95 -0.50 1
8 8 0 2.28 0.45 1
99 99 99 0.00 0.00 0
FVAR 0.81554
O1 3 0.13876 0.08798 0.50278 11.00000 0.04336 0.06483 =
0.04712 -0.00264 0.00342 -0.00309
O2 3 0.13828 0.14520 0.60009 11.00000 0.05341 0.09107 =
0.05675 -0.01351 0.01645 -0.00162
C1 1 0.18085 0.12821 0.55575 11.00000 0.05144 0.04701 =
0.05208 -0.00446 0.00460 -0.00039
.....
.....
H16A 2 0.44180 0.18390 0.52360 11.00000 0.05000

```

```
H16B 2 0.45100 0.33370 0.57250 11.00000 0.05000
END   PIG
```

The output line file SHX for the above example follows:

```
STARTX          CELL 15.18000 7.41300 22.60700 90.00000 90.00000 90.00000
LATTICE C P
SYMTRY X,Y,Z
SYMTRY 1/2-X,1/2-Y,1/2+Z
SYMTRY -X,Y,1/2-Z
SYMTRY 1/2+X,1/2-Y,-Z
CELCON C 128
CELCON H 128
CELCON O 16
DATDEF PARENT
EXPER 1 1.54179
ADDRF
HKLIN HKL FREL SIGF RCOD
HKL 2 0 0 24.990 7.660 2
HKL 4 0 0 20.510 6.520 2
HKL 6 0 0 23.060 3.470 1
HKL 8 0 0 19.460 2.480 1
HKL 10 0 0 15.210 1.700 1
.....
.....
HKL 6 8 0 1.950 .500 2
HKL 8 8 0 2.280 .450 1
ADDATM          SCALE 1 .81554
ATOM O1 .13876 .08798 .50278 3.42357 1.00000
UIJ O1 .04336 .06483 .04712 -.00309 .00342 -.00264
ATOM O2 .13828 .14520 .60009 4.21708 1.00000
UIJ O2 .05341 .09107 .05675 -.00162 .01645 -.01351
ATOM C1 .18085 .12821 .55575 4.06154 1.00000
UIJ C1 .05144 .04701 .05208 -.00039 .00460 -.00446
.....
.....
ATOM H16A .44180 .18390 .52360 3.94784 1.00000
ATOM H16B .45100 .33370 .57250 3.94784 1.00000
RESET INP 5
FINISH
```

4.63. *SIMPEL*: Symbolic-addition phasing

Authors: Henk Schenk and Syd Hall

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

SIMPEL applies the symbolic addition procedure to triplet and/or quartet structure invariant relationships to determine structure factor phases from normalized structure factors. The program is space group independent, and the origin and enantiomorph is specified automatically, or may be selected by the user. *SIMPEL* contains a wide range of phase selection and extension options. Different procedures for accepting, propagating, and evaluating symbol phases are provided and a variety of figure-of-merit tests are available to identify the correct symbol phases. Up to 16 phase sets may be output for subsequent E-map calculations.

4.63.1. The Algorithm

Symbols have been used with structure invariant relationships to determine structure factor phases from the beginning of direct methods. Symbolic addition procedures (SAP) compete with and complement the other main direct-methods approach, the multi-solution procedure. SAP defines and extends phases as symbols which are evaluated at the conclusion of the process. No permutation of phase sets is involved, as in the multisolution procedure, so that the process is fast.

There is considerable background literature on the symbolic addition procedure. In particular we refer to Karle & Karle (1966), Karle (1974) and Schenk (1980). References specific to the SIMPEL approach to symbolic addition can be found in Overbeek & Schenk (1978), Schenk (1983) and Schenk & Kiers (1984).

4.63.2. Step 1. Convergence Procedure: selecting the starting set

The first step in the symbolic addition procedure is the specification of the origin/enantiomorph defining phases, and the selection of symbolic phases. These phases, referred to as the starting set, must be those which will most reliably propagate to all other large E-values via the triplet and quartet structure invariants. This is the pivotal step in the phasing process and considerable care is taken to ensure that the best possible starting set is selected. The starting phases are selected from the largest $|E|$ -values (25% of total, or user option on the **SIMPEL** line) using a convergence type procedure as described by Germain and Woolfson (1970). Convergence rejection is based on either:

(1) α^2 for centrosymmetric and noncentrosymmetric

or (2) $\alpha \tanh(\alpha/2)$ for centrosymmetric or $\alpha I_1(\alpha)/I_0(\alpha)$ for noncentrosymmetric

See the *GENSIN* writeup for the full α definition. The result of this procedure is a set of reflections which form an optimal choice for a starting set. Within this set origin (and enantiomorph, if required) defining reflections are assigned phases first, and then several symbols are assigned amongst the remaining reflections of the set (maximum is 8). The assigned phases are applied to $|E|$ -values above the convergence limit using triplet and/or quartet invariants (user option on **start** line). Generated phases which have consistent values (no conflicts) are accepted and set at a weight of 1.0 for the rest of the phasing procedure.

4.63.3. Step 2. Divergence Procedure: test phase propagation

A divergence or accessibility procedure is used to test if the starting set of phases selected in Step 1 will propagate satisfactorily within the divergence set of generators (default is 50% of total). The procedure checks if the phases of all $|E|$ -values can be accessed via sufficiently reliable invariant phase relationships. No test for (symbolic) phase consistency is made in this procedure. New phases are accepted as accessible if the sum of the α s of each invariant exceeds a threshold value (ALTHR starts as 10th largest α of input invariants, or user option on the **start** line). The threshold value ALT(m) is varied according to the number of invariants, m, used to derive a phase. For example, if for one invariant ALT(1) = ALTHR, for two invariants ALT(2) = 1.3*ALT(1), for three invariants ALT(3) = 1.3*ALT(2) and so on. For each next cycle the threshold value is reduced by a factor of 0.9. This ensures that the most probable phases are accessed first.

If more than 10 of the $|E|$ -values in the divergence set remain unphased, additional symbol phases are assigned to unaccessed reflections. This enables any unconnected groups of related reflections to enter easily into the phasing process.

4.63.4. Step 3. Symbolic Addition Process

Once the starting set of phases has been fixed by the convergence and divergence processes, they are used to phase all remaining $|E|$ -values. The starting set is composed of numeric phases assigned to specify the cell origin (and enantiomorph), and symbol phases to ensure complete phase extension. The starting set also contains reliable numeric and symbolic phases which were derived in the earlier processes. All these phases,

assigned and derived, are now considered to be active phase propagators and have been assigned a weight of 1.0.

The symbolic addition process uses these phases and the structure invariant relationships to derive the phases of the other $|E|$ -values. Before accepting a newly derived phase into the list of known phases (and thus successively using this reflection to derive additional phases) it must be carefully checked for reliability. If an incorrect phase is accepted, it can lead to a failure of the whole phasing process.

Two different mechanisms are used in SIMPEL to test if a new phase is suitable as an active phase propagator. These are the "probability threshold" test and the "multisymbol acceptance" test. In each case there are two separate procedures for performing these tests - each has different properties that may suit the solution of particular structural types.

4.63.4.1. I. Probability Threshold Test

A new phase is derived by substituting known phases into one, or more, structure invariant relationships. Each of these relationships has an α value which is a measure of the probability that the phase relationships have a value of zero (base module 2π). If α is high, there is a high probability this is true; if it is not, the relationship must be used with caution, or in conjunction with other relationships. The sum of these α 's is therefore an important test in gauging the probable reliability of a new phase. In SIMPEL the sum of α 's may be applied in two different ways. They are:

- (i) *Weighted Alpha method (wa)*: Each phase in the active phase list is assigned a weight according to its expected reliability. These weights range from WMIN to 1.0 and are calculated for centrosymmetric phases as

$$W = \tanh \alpha_c / \text{ALTHR}$$

and for non-centrosymmetric phases as

$$W = \min(1., \alpha_c / \text{ALTHR}) \text{ where}$$

$$\alpha_c = [\{ \sum W_k \alpha_k \sin \phi_k \}^2 + \{ \sum W_k \alpha_k \cos \phi_k \}^2]^{1/2} \text{ (summed over m).}$$

ALTHR is the threshold α value specified by the user, or set automatically as the 10th largest α of input invariants. This is the same ALT used in divergence process. W_k is the combined weight of the component phases in the invariant derived from

$$W_k = \prod W_j / \sum W_j, j=1 \text{ to } 2 \text{ (triplets), or } 3 \text{ (quartets).}$$

A phase is accepted if its calculated weight W_k is above the minimum weight WMIN. WMIN is an input option or is set automatically to 0.3. In this method phase acceptance is a relatively smooth and continuous process. Each new phase given an associated reliability index; an index which is used to determine the reliability of subsequent phases (i.e. the history of prior determinations has a bearing on future phase estimates). The calculation of W_k is slower than the alternative but this tends to be offset by the more rapid propagation of phases.

- (ii) *Alpha Ninv method (an)*: Another test for phase acceptance is available in SIMPEL based on the same procedure described above in Step 2. The weights of all active phases are assumed to be 1.0. The calculated α of a new phase (see α_c definition above) is tested against ALT(m), where m is the number of invariants used to calculate α_c and the new phase. The values of ALT(m) are preset as described in Step 2 above. This procedure is relatively simple and fast and is based on a relatively demanding criterion for acceptance. It is discontinuous (i.e. it accepts or rejects - nothing in between) and therefore requires more phasing cycles than (i). It also does not use the relative reliability of the active phases. This may be particularly important for non-restricted phases.

4.63.4.2. II. Multisymbol Acceptance Test

In the symbolic addition process a second phase acceptance test is applied when more than one structure invariant is used to derive a new phase (i.e. $m > 1$). This test has two separate modes of operation.

- (i) *Accept multiple symbol indications* (**mult**): If two or more symbol combinations are generated for new phase (e.g. say, -A and +AD), the phase is still accepted provided the strongest indication (i.e. largest α_c) satisfies acceptance test I. This mode assumes that certain symbol combinations are equivalent (i.e. will reduce to the same numeric phase) and promotes a new phase to the active propagation role provided it satisfies the probability acceptance criteria. This is the default mode.
- (ii) *Reject multiple symbol indications* (**sing**): In this mode a new phase is rejected if more than one symbol combination is derived. This is a more demanding requirement than in (i) and means that fewer phases are promoted to the active list to assume the role of phase propagators. In this mode the symbolic addition process requires more cycles and the statistics available to the subsequent figure-of-merit tests are fewer in number.

It should be noted that in previous versions of SIMPEL the only available options are I(ii) and II(ii). These are the most conservative options in accepting new phases, and have been used successfully in the past. The strong point of the alternative acceptance criteria I(i) and II(i) is discussed above and in view of this these are currently set as the default modes. Users are advised that if the defaults fail to provide a solution the more conservative combination of I(ii) and II(ii) should be applied.

4.63.5. Step 4. Correlation of Symbol Phases

In Step 3. the starting phase set is expanded into a larger list of known phases containing numeric and symbolic values. In this step a final symbolic addition cycle is applied so that all phase estimates can be tabulated as symbol correlation statistics (Schenk, 1971). Only phase estimates that involve more than one symbol combination will contribute to these statistics. For example if a given phase is estimated from ten different invariants to be:

phase	m	$\sum \alpha$
-	5	20
+A	3	15
-AD	2	10

where m is the number of invariants, this would lead to the correlation statistics

phase = 0 (+)	frequency	Product of $\sum \alpha$
-A	1	300
+AD	1	200
-D	1	150

This process assumes that different symbol indications for the same reflection are in fact equal and may therefore be correlated. The statistics above are consistent with symbols A and B having the value of 180° .

The symbol correlation table is then used to test the plausibility of numeric values for each symbol. Symbols assigned to restricted phases are assigned their two possible values (e.g. $p/2$ and $3p/2$) and symbols assigned to unrestricted phases are tested for the numeric values in the range 0 to $2p$ in intervals of $p/4$. The correlation statistics are used to calculate a correlation factor QFAC (Schenk, 1971) that has a maximum value of 100 if the numeric phases agree exactly (and -100 if they disagree exactly!). In space groups with translational symmetry (non-symmorphic) cofactors greater than 50 are good, and >70 are excellent. However, in the other

space groups the QFAC is less indicative.

The last part of this step orders the phase sets in descending magnitude of cofactor. Only the top phase sets (16, or specified by user) will enter into the more exhaustive figure-of-merit tests in the next step.

4.63.6. Step 5. Figure-of-Merit Tests: Identifying the correct phase set

The previous step selected, and ordered, the numeric phase combinations that have the best chance of being correct. In this step each of these combinations is applied in a separate symbolic addition cycle to provide the agreement statistics needed to calculate various figures-of-merit. A figure-of-merit is intended to discriminate between a 'good' phase set and a 'bad' phase set (i.e. one that may provide a correct solution from one that will not). Not all FOM's of the original SIMPEL versions are implemented at this time, but will be considered for future development. On the other hand, several FOM's have been added that are not present in the other SIMPEL versions.

4.63.6.1. Correlation factor Figure-of-Merit (QFOM)

This figure-of-merit is a reformulation of QFAC calculated in Step 4. It is

$$QFOM = 1.5 - QFAC/100.$$

In accordance with all other FOM values, the best QFOM is the lowest. It has an active range from 0.5 to 2.5, and any value below 1.1 is considered good, and above 1.5 is considered unlikely. QFOM is, of course, correlated to the symbol extension process and cannot be considered an independent phase set discriminator in the same sense as the FOM tests PSIO and NEGQ. Caution must therefore be exercised in interpreting small differences in QFOM values.

4.63.6.2. Relative Figure-of-Merit (RFOM)

This parameter is the inverse of the CFOM parameter of the MULTAN program (Main et al., 1980) and has the form

$$RFOM = (\sum \langle \alpha \rangle - \sum \alpha_r) / (\sum \alpha_c - \sum \alpha_r) \text{ (summed over all h)}$$

where $\langle \alpha \rangle$ is the expected α of a phase, and α_r is the α if all phases were randomly distributed. For a correct phase set the value of α_c should approach that of $\langle \alpha \rangle$ and RFOM should tend to 1.0. Incorrect phase sets will deviate significantly from 1.0, random phases towards 2.0, and overcorrelated phases towards 0.0. In general, however, phase sets with small RFOM's are more likely to be correct than those with large RFOM's. The range of RFOM's will vary according to the validity of the estimate of $\langle \alpha \rangle$. For this reason RFOM tends to be less reliable for strongly non-random structures.

4.63.6.3. R-factor Figure-of-Merit (RFAC)

The RFAC parameter is similar to the residual FOM calculated in MULTAN (Main et al., 1980) except for a scale that takes into account the relative dominance of heavy atoms in the structure.

$$RFAC = \sum \{ |\alpha_c - \langle \alpha \rangle| \} / \sum \langle \alpha \rangle \text{ (summed over all h)}$$

RFAC is a minimum when there is close correspondence between α_c and $\langle \alpha \rangle$. In this respect it is very similar to the R-factor of Karle and Karle (1966). RFAC is, like RFOM, dependent on the reliability of $\langle \alpha \rangle$.

4.63.6.4. PSI0 Figure-of-Merit (PSI0)

PSI0 triplet invariants (Cochran and Douglas, 1955) provide a figure-of-merit which is largely independent of the triplet and quartet invariants used in the tangent refinement. A PSI0 triplet relates two strong reflections (with $|E| > E_{MIN}$) to a third which has an $|E|$ -value as close as possible to zero (see the *GENSIN* writeup). The phases estimated from a series of PSI0 triplets are expected to be random when the contributing phases from the other two large- $|E|$ reflections are correct. When this is the case the resulting values of α_c are significantly lower than if the distribution of contributing phases was biased or incorrect. These invariants are used to form the figure-of-merit

$$PSI0 = \sum \alpha_c / \sum \langle \alpha \rangle \quad (\text{summed over psi0 triplets}).$$

PSI0 should be smallest for the correct phase set. PSI0 is, along with NEGQ, one of the most independent methods of measuring the relative likelihood of success.

4.63.6.5. Negative Quartet Figure-of-Merit (NEGQ)

Quartet structure invariant relationships are classified according to the magnitude of their crossvector $|E|$ values. When the crossvector $|E|$'s are small there is a high probability that the phase invariant has a value close to π rather than 0 (Hauptman, 1974; Schenk, 1974). These invariants are referred to as negative quartets. In SIMPEL negative quartets are not used in the phasing process but are retained as a test of the phase sets. The negative quartets are considered independent because, unlike the positive quartets, they cannot be represented by a series of triplet invariants. They provide, therefore, a separate estimate of the phases. A direct comparison of these phases provides the basis for the figure-of-merit (Schenk, 1974).

$$NEGQ = \sum [\alpha_c |\phi_k - \Phi_k|] / \sum \alpha_c \quad (\text{summed over all k neg. quartets})$$

where ϕ_k is the phase estimated from triplets and positive quartets, and Φ_k is the phase estimated from negative quartets alone. Correct phase sets should have low values of NEGQ ranging from 0 for centrosymmetric structures, to 20-60° for non-centrosymmetric structures. Note that if fragment QPSI values are used the value of ψ is automatically set to 0 and the NEGQ test will remain valid. This FOM is a very powerful discriminator of phase sets provided that sufficient negative quartets are available.

4.63.6.6. Combined Figure-of-Merit (CFOM)

The combined FOM is a scaled sum of the FOM parameters QFOM, RFOM, RFAC, PSI0 and NEGQ.

$$CFOM = \sum [WFOM_i (FOMMAX_i - FOM_i) / (FOMMAX_i - FOMMIN_i)] \quad (i=1 \text{ to } 5)$$

The weights WFOM may be specified on the SETFOM control line. These values are subsequently scaled so that the maximum value of CFOM is 1.0. It is important to emphasise that CFOM is a relative parameter and serves only to highlight which is the best combination of FOM's for a given run. It does not indicate if a given FOM will provide a solution.

4.63.6.7. Absolute Measure-of-Success Parameter (AMOS)

The AMOS parameter is a structure-independent gauge of the correctness of a phase set. It uses pre-defined estimates of the optimal values for the FOM parameters QFOM, RFOM, RFAC, PSI0 and NEGQ. OPTFOM values may be user defined (see `setfom` line). Rejection values for the four FOM parameters are derived from the OPTFOM values as $REJFOM = 3 * OPTFOM$. The default values are as follows:

	<i>QFOM</i>	<i>RFOM</i>	<i>RFAC</i>	<i>PSI0</i>	<i>NEGQ</i>
OPTFOM	0.75	1.0	0.25	.75	60°
REJFOM	2.25	3.0	0.75	2.25	180°

The absolute measure-of-success parameter is calculated from all active FOMs as

$$AMOS = \sum [WFOM_i (REJF_i - FOM_i / OPTFOM_i)] \quad (i=1 \text{ to } 5)$$

where the WFOM values are scaled so that AMOS ranges from 0 to 100. In addition to being used to sort phase sets in order of correctness, the AMOS values provide a realistic gauge of the correctness of phase sets. As a rule of thumb, they can be interpreted in the following way:

AMOS	
100-81	<i>high probability of being correct set</i>
80-61	<i>good chance of being correct set</i>
60-41	<i>possibility of being correct set</i>
40-21	<i>low probability of being correct set</i>
20-0	<i>very unlikely to be correct set</i>

These classifications are only approximations. The predictability of optimal FOM values can be perturbed by a variety of structure dependent factors and by the FOM weighting.

4.63.6.8. Rejection of Phase Sets

Phase sets must satisfy certain criteria before being considered for possible output to the binary file for subsequent E-map calculations.

FOM Rejection Criteria	Message
Reject if QFAC > REJFOM(1)	REJECT1
Reject if RFOM > REJFOM(2)	REJECT2
Reject if RFAC > REJFOM(3)	REJECT3
Reject if PSI0 > REJFOM(4)	REJECT4
Reject if NegQ > REJFOM(5)	REJECT5
Reject if $ \text{av.}\phi - \langle \text{av.}\phi \rangle > 45^\circ$	REJECT10

The value of $\langle \text{av.}\phi \rangle$ is 90° for centrosymmetric structures and $150-180^\circ$ for non-centrosymmetric structures. This test avoids the "all-plus catastrophe" phase set.

4.63.7. File Assignments

- Reads |E| values from the input archive bdf
- Writes the estimated phases to the output archive bdf
- Reads structure invariant relationships from bdf `inv`

4.63.8. Examples

SIMPEL

This is the standard run in which all defaults will be applied to the converge, diverge, symbolic addition and FOM testing processes. All |E|-values used in the *GENSIN* program, and all invariants entered on `inv`, will be applied in the phasing process. Phase extension will be based on weighted α s.

```

SIMPEL 60 200 *6 8           :set con/divergence limits
invar all 1. 0.7             :use all invariants with new a/b limits
start trip                   :use only triplets in converge process
symbol an single            :accept single symbols based on altm
print bdfout                :print phase sets output to bdf

SIMPEL
symbol wa single            :use weighted alphas ; reject multisymbols
phase 5 7 11 0 nul         :suppress from process
phase 1 2 7                :assign as symbolic phase

```

4.63.9. References

- Cochran, W. and Douglas, A.S. 1955. *Proc. Roy. Soc.* A277, 486-500.
- Germain G., Main, P. and Woolfson, M.M. 1970. *Acta Cryst.* B26, 274-285.
- Hauptman, H.A. 1974. *Acta Cryst.* A30, 472-476.
- Karle, J. and Karle, I.L. 1966. *Acta Cryst.* 21, 849.
- Karle, J. 1974. *International Tables*, Vol. IV, section 6, 337.
- Main, P. 1980. *Multan-80* York England: University of York. Overbeek, A.R. & Schenk, H. 1978. *Computing in Crystallography*, University Press, Delft, p108-112.
- Schenk, H. 1971. *Acta Cryst.* B27, 2037-2039.
- Schenk, H. 1974. *Acta Cryst.* A30, 477-481.
- Schenk, H. 1980. *Computing in Crystallography*, eds. R. Diamond, S. Rameseshan and K. Venkatesan, Indian Academy of Sciences, Bangalore, p701-722.

4.64. *SLANT*: Slanted map

Authors: Nick Spadaccini & Richard Alden

Contact: Nick Spadaccini, Computer Science Department, University of Western Australia, Nedlands 6907, Australia

*SLANT produces, using interpolative methods, a Fourier map in any general plane from the Fourier map values generated by another program (usually *FOURR*). Three non-collinear grid points define a set of coordinate axes *a(calc)*, *b(calc)* and *c(calc)* relative to which the required map point values are interpolated in three dimensions.*

4.64.1. Input And Output Requirements

The Fourier map entered from the `map bdf` must contain enough points so that every point in the output plane is represented. The simplest way to ensure this is to input at least the map of the asymmetric unit of the cell. Any map type may be used (e.g. Patterson, E map, difference map). The computed map may be printed or output to the `bdf s1a`, with the same structure as `map`, for use with programs such as *CONTR*s. The optional output is printed in base-10 integer format (`normal`) with the map density scale, upper and lower print suppression limits and map format under user control. The output options are invoked on the *SLANT* line.

Note that all input map points are stored in the *QX* array. Depending on the size of the unit cell, and on the resolution desired, this program requires a very large memory allocation.

4.64.2. General Plane Specifications

There are two coordinate systems involved in the program. The first is the crystallographic coordinate system defined by the crystal symmetry. The other may be thought of as the calculated coordinate system and is determined relative to the general slant plane. The calculated coordinate system is orthogonal with two vectors in the general plane and the third normal to the plane. All information concerning the definition of the general plane is in terms of crystallographic coordinates and the layout of the computed map is specified in terms of calculated grid coordinates. Since the calculated coordinate vectors have no absolute length, the grid interval can be explicitly specified in angstroms per division. If not specified the program takes the average of the input map resolution as the output grid interval.

4.64.3. The Slant Plane Coordinate System

Three non-collinear points which define the required plane are specified on the **defp1n** line in either grid or fractional crystallographic coordinates. The two in plane vectors defined by these three points are used to generate an orthogonal coordinate system which characteristically defines its origin to be the first supplied point, the vector between the first and second points is defined to be *a(calc)*, the normal to the plane is *b(calc)* and the vector product of these is *c(calc)*. The grid interval widths in angstroms for each direction in the calculated coordinate system may be defined on the **mapres** line.

4.64.4. Map Orientation

The **layout** line specifies the desired orientation of the maps relative to the calculated coordinate system. The three vectors *a(calc)*, *b(calc)* and *c(calc)* must be specified as either the page to page, down page or across page directions. The **layout** line allows entry of the number of grid points, the origin, and the stepping interval for each of the three directions in the map. With the absence of a **layout** line the program will produce an output of the plane containing the three points supplied on the **defp1n** line. The number of grid points in the *a(calc)* and *c(calc)* directions is determined so that all three points just fit within the map.

4.64.5. Interpolation Routines

Linear interpolation from the input Fourier map to produce the slant plane map leads to the shortest calculation times. Each map point on the required plane is expressed in terms of the original crystallographic grid coordinates. The program is written to select 8 of the supplied map points closest to this point and to interpolate linearly. Assume that the required map point is at $(x+pX, y+qY, z+rZ)$ in crystallographic grid coordinates, where *X*, *Y* and *Z* are the grid intervals. The grid point less than $x+pX$ is designated as 1 and that greater than $x+pX$ is designated as 2. The same definition holds for $y+qY$ and $z+rZ$. The trivariate interpolation formula with the origin at 111 is as follows:

$$v(x,1) = 1 - p$$

$$v(x,2) = p$$

and similarly for

$$v(y,j) \text{ and } v(z,k) \text{ where } j, k = 1, 2$$

$$F(x+pX, y+qY, z+rZ) = \sum_i \sum_j \sum_k F_{ijk} v(x,i) v(y,j) v(z,k)$$

A slower, but more accurate, method of interpolation uses the cubic algorithm. It may be found useful or necessary when the input Fourier grid is at low resolution. The interpolation algorithm to be used is specified on the **SLANT** line. The cubic method is estimated to be more accurate by a factor of ten than the linear method.

It is, however, a factor of five slower. The use of either method is much faster than the calculation of an exact general plane Fourier transform. It should be noted that a cubic interpolation using a coarser grid resolution is comparable to a linear interpolation at roughly twice the resolution and thus requires much less memory to store the input map. The cubic algorithm uses the nearest 64 points and uses the following formula. The same notation is used again, but with the origin at 222:

$$v(x,1) = -p^3/6 + p^2/2 - p/3$$

$$v(x,2) = p^3/2 - p^2 - p/2 + 1$$

$$v(x,3) = -p^3/2 + p^2/2 + p \quad v(x,4) = p^3/6 - p/6$$

and similarly for $v(y,j)$ and $v(z,k)$ $j, k = 1, 2, 3, 4$

$$F(x+pX,y+qY,z+rZ) = \sum_i \sum_j \sum_k F_{ijk} v(x,i) v(y,j) v(z,k)$$

4.64.6. Transformed Positional Parameters

By the use of **site** or **siteg** lines the user may obtain positional parameters relative to the slant plane in terms of calculated grid coordinates.

4.64.7. File Assignments

- Reads Fourier map from the bdf `map`
- Optionally writes the slant map to the bdf `sla`

4.64.8. Examples

```
SLANT file nprint cubic
defpln frac 0 0 0 .56 .24 .31 .74 .86 .21
mapres .15 .15 .15
layout across layer down 21 1 21 -10 0 -10 1 1 1
```

In this example the map is not printed but is output to bdf `sla`. Cubic interpolation is used. The lines **defpln**, **mapres** and **layout** refer to the general plane coordinate axes a-calc, b-calc, c-calc. These axes are defined by the points, in fractional coordinates, given on the **defpln** line. The layer direction is b-calc, with c-calc down the page. Only one layer is to be calculated with 21x21 points passing through the origin, starting at -10 in both directions. Given the resolution of 0.15 Å/div., the output map is 3.x3.Å with origin at the centre and defined as (0,0,0)

```
SLANT nfile print linear
defpln grid 10 20 30 17.5 1.5 13.3 18.5 54.6 12.3
```

The output map is printed only. Linear interpolation is used. The slant plane is defined by the points on **defpln**. These refer to the *original* grid coordinates used in **FOURR**. The average resolution used in **FOURR** will be applied. The output plane will be large enough to contain the three points that define the plane.

4.65. **SORTRF**: Sort/merge reflection data

Authors: Syd Hall, Nick Spadaccini and Jim Stewart

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia.

SORTRF sorts and, optionally, averages the reflection data on an archive bdf.

4.65.1. Method

SORTRF applies two important processes to reflection data on the bdf. Reflections may be sorted by their Miller indices so that they are in an order that is optimal for subsequent calculations. According to the sort algorithm selected reflections which are symmetrically equivalent may be placed together or treated independently. This is detailed in the next section. The second function of SORTRF is to average reflection data which are symmetrically equivalent. That is, multiple observations of reflections and their Friedel mates are reduced to a single packet with an averaged value stored for the reflection. The averaged values for the Friedel related reflection¹ may be treated and stored independently if requested.

1 Note that SORTRF should be applied before ADDREF if the intensity data, rather than the Frel data, is to be averaged. Note also that systematically absent data are not removed by SORTRF - this is done in ADDREF.

4.65.2. Sorting Reflection Data

The sort *order* of reflections is the measure that each Miller index changes from reflection to reflection. The input option `order smf` on the SORTRF line determines the sort order. The parameter `smf` is the combination of characters `h`, `k`, and `l`. `s` designates the slowest varying index; `f` designates the fastest varying index. Thus the entry of `order kh1` specifies that `k` will be slowest varying and `l` fastest varying.

Three sort algorithms are available for the treatment of equivalent reflections. These are referred to as sort *types*.

4.65.2.1. Sort type 1: *Sort only mode*

In this mode data is sorted so that each index, within the constraints of the specified sort order, will start with its most negative index and finish with its most positive index. No account will be taken of symmetry equivalent reflections and no transformation of the input indeices will take place. This is strictly a *sort only* mode. This is the default provided `aver` or `c1us` options are not set.

4.65.2.2. Sort type 2: *adjacent Friedel mode*

This sort type is identical to 1 except that the Miller indices of equivalent reflections are transformed to have matching (and most positive) values. If the `pakfr1` or `sepfr1` flags are entered then the Friedel equivalent reflections will be treated independently and will be immediately adjacent in the sort list. If the flags `pakfr1` or `sepfr1` are *not* entered the Friedel related data will be treated identically to symmetrically equivalent data. This is the default mode if the `aver` or `c1us` options are set.

4.65.2.3. Sort type 3: *group magnitudes mode*

This sort type is identical to 1 except that the Miller indices of equivalent reflections are transformed to have matching (and most positive) values. In addition all reflections with the the same index *magnitudes* are adjacent in the sort list (note that reflections with matching index magnitudes are not necessarily symmetrically equivalent). The treatment of Friedel data is identical to a type 2 sort. This mode is intended for speeding up a Fourier calculation which uses the Beevers-Lipson algorithm. It is not suitable for listing purposes and it is not recommended except for very large data sets on a relatively slow computer.

4.65.3. Averaging Reflection Data

In addition to sorting data, SORTRF may be used to *average* or *cluster* multiple observations and symmetrically equivalent reflection data. The SORTRF options for this are `noav`, `aver` and `c1us`. `noav` is the default and specifies that no averaging of equivalent data will take place. `c1us` causes equivalent reflections with their original untransformed indices to be adjacent in the sorted list and that no averaging take place.

`aver` will cause the specified coefficients (Frel, F²rel or Irel) of equivalent reflections to be averaged. If not specified the averaged coefficient is assumed first to be Irel; if these data are not present Frel will be averaged.

Note that only the specified coefficients are averaged; other coefficients are not averaged. The two averaging processes available in SORTRF are described below.

4.65.3.1. `aver 1`: standard average with culling

This algorithm provides for a standard average to be calculated, with the provision for culling data which exceeds a certain deviation from the initial mean. In the following treatment, the intensity I may be replaced by $|F^2|$ or $|F|$ according to what is specified on the SORTRF line. The initial mean intensity is calculated as:

$$I = \sum [I_j] / N \text{ where } N \text{ is the number of equivalents.}$$

The σI is calculated from the individual I_j e.s.d.'s (estimated usually from counting statistics), plus the deviations of the contributing intensities from the initial mean. That is,

$$\sigma I = \max((\sum I_j)^{1/2} / N, [N \sum I_j^2 - (\sum I_j)^2 / (N(N-1))]^{1/2}, 0.0001)$$

If the cull `t` option is specified any individual intensity that deviated from the initial mean by greater than $t\sigma I$ is culled before the final mean I is calculated.

4.65.3.2. `aver 2`: Fisher Test

The second averaging process consists of finding the arithmetic average of reflection intensity is:

$$I = \sum I_j / N \text{ where } N \text{ is the number of equivalents.}$$

Two measures are made of the variance of the average intensity I . One is based on Poisson statistics, and the other is based on the scatter of equivalents,

$$(\sigma^2 I)_p = \sum \sigma^2 I_j / N^2$$

$$(\sigma^2 I)_E = \sum (I_j - I)^2 / \{N(N-1)\}$$

Since both measures have different degrees of freedom, N and $N-1$ respectively, a test can be made for the hypothesis that they are equal to the 1 significance level of the Fisher distribution. This is referred to as the Fisher test (Hamilton, 1964). Reflections whose measures of variance prove equal (the hypothesis is successful) may have the adjusted σI for the average intensity optionally set to one of the following values by the `success` option:

i. $\sigma I = (\sigma^2 I)_p^{1/2}$

ii. $\sigma I = (\sigma^2 I)_E^{1/2}$

iii. $\sigma I = \max(i, ii)$

Reflections whose measures of variance prove unequal (the hypothesis is unsuccessful) may have the adjusted σI for the average intensity optionally set to the above, or rejected (`rcode=2`, `σI =voidflg:`) by the `unsuccess` option on the SORTRF line.

Note that this option will assign an r_{code} of 2 to a reflection if, and only if, the tests fail to satisfy the hypothesis relating to its variances and if this is explicitly requested by the user in the `unsuccess` option. All reflections which satisfy the hypothesis will have an r_{code} of 1.

4.65.4. Agreement Factors

Two agreement factors are provided to measure the quality of the averaged data:

$$R_1 = \frac{\sum |I_j - I|}{\sum I_j} \text{ and } R_2 = \frac{\sum \sigma I_j}{\sum I_j}.$$

4.65.5. Treatment Of Friedel Pairs

Attention must be drawn to the necessity of calculating a $|F_{rel}|$ corrected for the effects of dispersion prior to use in an electron density calculation with *FOURR*. The dispersion corrections to $|F_{rel}|$ for reflection and anti-reflection are not equal. With Friedel data packed into the same packet of the bdf, as is necessary for absolute structure refinement, a dispersion-free $|F_{rel}|$ can be correctly calculated from the $|F_{rel}|$'s for the reflection and anti-reflection with *CRYLSQ*. However, this is only possible if the $|F_{rel}|$'s of Friedel pairs are not averaged! The average $|F_{rel}|$ is made up of (weighted) contributions from m reflections and n anti-reflections. The weights and the values of m and n vary from one reflection to another, and a proper calculation of the dispersion correction to $|F_{rel}|$ needs to know all of these values. It means that electron density calculations made with packed pairs, and from averaged pairs will almost certainly differ. The electron density calculated from the packed data is correct.

Friedel related reflections will be treated as symmetrically equivalent data unless either the `pakfr1` or `sepfr1` options are entered. If these options are specified Friedel pairs are treated as independent data and are not averaged. If `pakfr1` is entered the Friedel data is placed in the same *lrrefl*: packet on the bdf.

The packed Friedel data is needed if absolute structure parameters are to be refined by CRYLSQ.

If the `sepfr1` option is used the Friedel pair data is stored in separate but adjacent *lrrefl*: packets.

Note that invocation of either option will automatically cause the data to be averaged. If no `aver` option is stated then mode 1 is assumed.

4.65.6. File Assignments

- Reads reflection data from the input archive bdf
- Writes sorted and/or averaged data to the output archive bdf

4.65.7. Examples

```
SORTRF order lkh
```

Data will be sorted with h changing most rapidly, and k next most rapidly.

```
SORTRF aver 1 cull 2.5 print 512
```

During averaging measurements $>2.5 \sigma I$ from the initial mean are culled and a new average computed. The first 512 reflections will be printed.

```
SORTRF aver 2 succe 3 unsuc 4 print -9999
```

This example uses the Fisher average. Those reflections satisfying the hypothesis will have σI based on the maximum variance. Those reflections failing to satisfy the hypothesis will be rejected. The first 9999 reflections which fail to satisfy the hypothesis are printed.

4.65.8. References

- Hamilton, W.C. 1964. *Statistics in Physical Science*. Ronald Press: New York

4.66. *STARTX*: Create initial archive file

Authors: Jim Stewart, Jim Holden, Ruth Doherty, George Davenport, Syd Hall and Howard Flack

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

STARTX creates or updates an *Archive* (bdf) with all of the cell, symmetry, crystal and structure information that is known prior to the solution of the structure, exclusive of reflection data. Reflection data is added by subsequent calculations.

STARTX starts, or updates (in `upd` mode) the *Archive* by storing the initial information about the file history, any text descriptive of the nature of the structure, unit cell parameters, the unit cell symmetry, the types and number of atoms contained in the unit cell, and various experimental data. The bdf is in essence a repository for known information about the structure under study, and will be available to all subsequent calculations. Other calculations will read this file and add, supersede or delete information as requested.

4.66.1. Unit Cell Data

Cell information is supplied by the `cell` and `cellsd` input lines. This information is stored in the *lrcell*: (the macro name that identifies the *logical record* containing *cell* data) record of the bdf, along with the calculated reciprocal cell dimensions and transformation matrices used to transform to orthogonal coordinates (see the *Archive* at the back of the manual for details).

4.66.2. Symmetry Data

STARTX provides several approaches to entering symmetry information. The most concise way is to use the explicit-origin space group notation (Hall, 1981) with the `sgname` line. This notation is listed in the supplied `symmap` file. Space group symmetry can also be defined by entering a `lattice` and/or `symtry` or `spaceg` lines. For example the space group $A2/a$ (15:c1 z-unique) may be entered as follows:

```
sgname      -a_2a
```

or

```
spaceg     a_1_1_2/a
```

or

```
lattice      c      a
symtry       x,y,z
symtry       1/2-x,-y,z
```

or

```

symtry          x,      y, z
symtry          -x,     -y, -z
symtry  1/2-x,   -y, z
symtry          1/2+x,   y, -z
symtry          -x, 1/2-y, z
symtry          x, 1/2+y, -z
symtry          1/2+x, 1/2+y, -z
symtry          1/2-x, 1/2-y, z

```

The number of molecules per cell is entered with the **celmol** line. The default value is one molecule per asymmetric cell unit. If this is not correct for your structure, use a **celmol** line - as this greatly assists routines such as *PEKPIK*, *MODEL* and *PIG*.

The contents of the crystal unit cell is with **celcon** input lines and using internal data (such as the dispersion values) stored for each atom type. A **celcon** line *must* be entered for each atom species expected in the structure. The number of atoms of this type expected in the unit cell should also be entered. If the scattering factor name entered is the 'standard' element name, then other atom data such as the atomic weight, the atomic number, the number of electrons, the usual atomic bond radius, the usual Van der Waals radius, and anomalous dispersion scattering terms will be supplied by STARTX automatically. *NOTE that if the neutron scattering length is entered all subsequent calculations will assume that neutron data is being processed.*

STARTX reads tables stored in the file `atomtype.cif` for the automatic generation of atomic scattering factors. The use of these tables is dependent upon the value of the atomic number and the number of electrons derived from the scattering factor name *or* specified explicitly in the **celcon** input lines. For example, a Cu^+ atom could be defined as either Cu^+ (or CU^+ or cu^+ or Cu1^+ -- the scattering factor symbol is not case sensitive whereas the atom label which incorporates this symbol is), or CU with fields 3 and 4 set at 29 and 28, respectively.

STARTX also reads `atomtype.cif` for data to enable scattering factors to be automatically generated for Xrays, electrons and neutrons. Automatic generation is overridden by data entered on `formgn`, `formfx` and `celcon` lines. Scattering factors may be generated for s values out to 6.0 using the method of Fox et al. (1989) A.C. A45, 786. The Stewart, Davidson & Simpson H curve is now generated from the Int Tab Vol 4 coefficients in Table 2.2B, as do all scattering factors. Neutron scattering lengths come from Int Tab Vol 4 Table 2.6. Scattering factors may be printed by entering the line **reset psta 4**

In a departure from rationality STARTX will permit dummy atom types entered with an atomic number of 0 and a voidflg: or 0 molar weight to be treated as valid types. Use this feature to shoot yourself in the foot in some later calculation like *CRYLSQ* or *FC*.

The nature of the radiation may be specified on the *STARTX* line. Here are the input options:

Radiation type	xrays	generate xray scattering factors
	neutrons	generate neutron scattering lengths
	electrons	generate electron scattering factors
Scatt factor limits	smax s	max s value of generated sf <2.0>
	sinc t	incr s used for generated sf<0.02>

A scattering factor symbol (also referred to as the atom type symbol) consists of a string of up to eight characters. If the user enters the usual international chemical symbols on the **celcon** line, these will be used to generate appropriate scattering factor data without the need to enter the atomic number or number of electrons (see last section also).

4.66.3. File Assignments

- Optionally reads data from the input archive bdf
- Writes the new or updated file to the output archive bdf

4.66.4. Examples

```
STARTX           :the P6122 test in simplest form
CELL 8.53 8.53 20.37 90 90 120          sgname P 61
2 (0 0 -1)
CELCON s 6
CELCON o 6
CELCON c 12
```

```
title    BODY-CENTRED ACENTRIC CUBIC TEST CASE, I 4BAR 3D
STARTX   CELL 11 11 11 90 90 90
sgname I -4bd 2ab 3
CELCON s 48
CELCON o 36
CELCON c 16
exper 1 1.5418 *9 1.234
maxhkl 12 8 4 .001 .9
```

```
STARTX
label p6122 test 21 june 1982
CELL 8.53 8.53 20.37 90 90 120
cellsd .01 .01 .01 .01 .01 .01
lattice n p           :alternative is sgname p 61 2 (0 0 -1)
symtry x,y,z
symtry -x,-y,1/2+z
symtry y,x,1/3-z
symtry -y,-x,5/6-z
symtry -y,x-y,1/3+z
symtry y,y-x,5/6+z
symtry -x,y-x,2/3-z
symtry x,x-y,1/6-z
symtry y-x,-x,2/3+z
symtry x-y,x,1/6+z
symtry x-y,-y,-z
symtry y-x,y,1/2-z
CELCON s 6 *8 .350 .869
CELCON o 6
CELCON c 12
```

4.66.5. References

- Hall, S.R. 1981. *Acta Cryst.* **A37**, 517-525.
- Mott, N.F. and Massey, H.S.W. (1965) *The Theory of Atomic Collisions*. pp86-112 Oxford Clarendon Press
- Xray Form Factors: International Tables Volume C Table 2.2B
- Neutron Form Factors: International Tables Vol C Tables 2.6 pp 270-271
- Electron Form Factors: International Tables Vol C Tables 2.4.6 A neutral International Tables Vol C Tables 2.4.6 B ionized
- Modified Mott $s < 0.1$: Peng, L.-M. and Cowley, J.M. (1988) *Acta Cryst.* **A44**, 1-5
- Modified Mott $s > 2.0$: Fox, A.G., O'Keefe, M.A. and Tabbernor, M.A. (1989) *Acta Cryst.* **A45**, 786-793.

4.67. **SURFIN**: Render 3D electron density isosurfaces **NEW**

Authors: Doug du Boulay Materials and Structures Laboratory, Tokyo Institute of Technology, Nagatsuta, Midori-Ku, Yokohama Japan.

*SURFIN is an OpenGL based subprogram intended for viewing 3D electron density isosurfaces. It builds on the **ogl** library as a tcl/tk based interface to the OpenGL library and incorporates a public domain version of the "marching cubes" algorithm to identify isosurfaces from gridded data.*

4.67.1. Purpose

SURFIN was cobbled together with chewing gum, as an example program to prove that 3D rendering of isosurfaces within the Xtal tcl/tk library infrastructure was possible. As such it was a success. Unfortunately its functionality leaves a great deal to be desired...

4.67.2. Functionality

SURFIN renders the electron density contents of the xtal .map. The graphical interface is a little misleading, in that a great deal of the implied functionality has actually not been implemented.

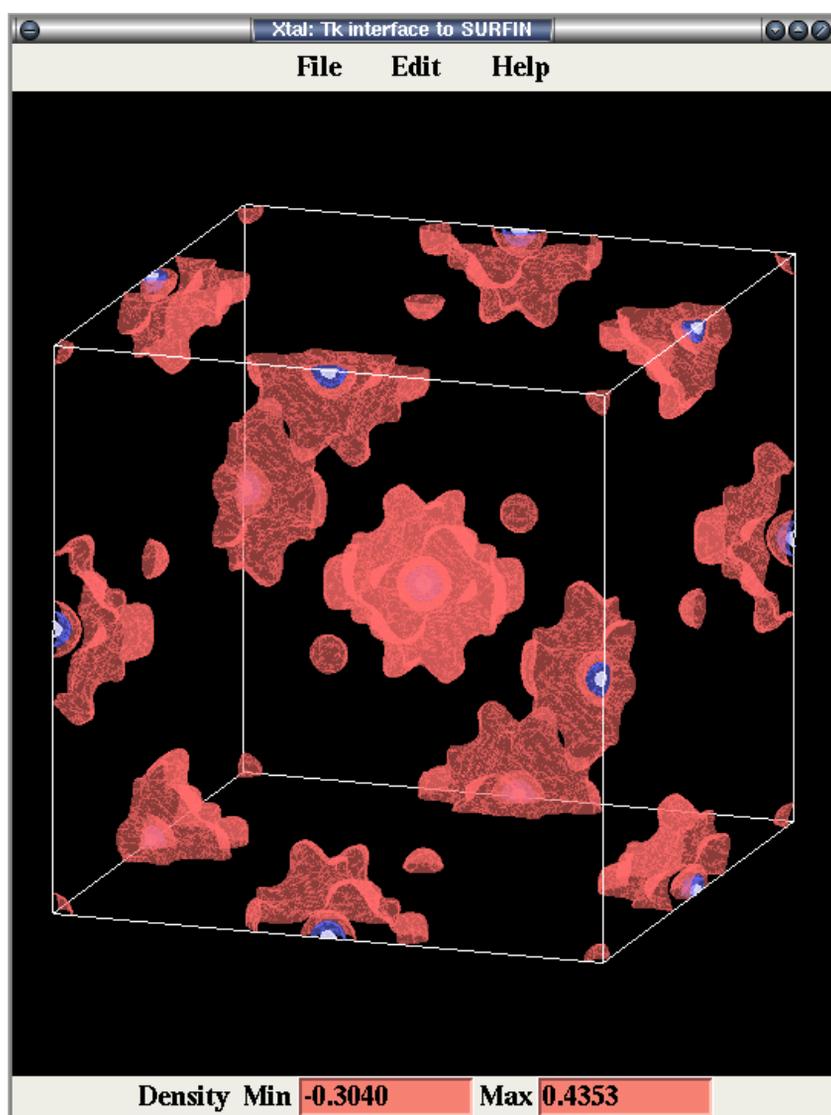
Currently contour lighting can be adjusted. The desired contour levels and colours can be customized via a tcl script file contained in the working directory. By default some relative values are adopted to span the range of the minimum and maximum difference electron densities.

Currently there is no functionality for viewing and manipulating atomic sites and bonds, lighting is basically not implemented and surface normals are not calculated correctly. Apart from that its perfect.

4.67.3. File Assignments

- Reads electron density from the .map file.
- Optionally reads ./tclsurfin.rc from the local directory.

Figure 4-6. Screen shot of SURFIN



4.67.4. Examples

```
# file: ./tclsurfin.rc    Sourced from surfin.tcl
# manually set some initial default values
#
#set contourSettings(Dvalues) [list -.5 -.2 .2 .5 ] ;# contour levels
set contourSettings(Dvalues) [list -4. -2. 2. 4. ] ;# contour levels
#set contourSettings(Dvalues) [list -40. -15. 15. 40.] ;# contour levels
set contourSettings(DrelVal) [list -0.8 -.3 .3 .8 ] ;# fraction of range
set contourSettings(FixedVals) 1 ;# 0/1 apply relVal or Dvalues
#
# Specify a colour gradient scheme for +ive and -ive contours
# The 4th coordinate is Opacity/transparency
set contourSettings(DmaxNeg) [list 1. 1. 1. 1.0 ] ; # Maximum negative colour
set contourSettings(DminNeg) [list 1. 0. 0. 0.0 ] ; # Minimum negative colour
set contourSettings(DmaxPos) [list 1. 1. 1. 1.0 ] ; # Maximum positive colour
set contourSettings(DminPos) [list 0. 0. 1. 0.0 ] ; # Minimum positive colour
#
```

4.68. *VUBDF*: Dump archive file

Author: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

VUBDF prints the contents of any *bdf* (archive or auxilliary) in terms of physical records, logical records and packets. No knowledge of the structure of the *bdf* or the logical records is assumed. *VUBDF* identifies each word in the *bdf* as containing character, integer, packed, real or void information and lists it accordingly.

4.68.1. Output Controls

VUBDF can be used to list a part or all of the contents of a *binary* data file. If no parameters are entered, *VUBDF* lists up to 100 packets of all logical records on the archive *bdf*.

VUBDF was originally written as a "developer's tool" and its use may require some knowledge of the structure of the *bdf*, as described in the *BDF* chapter at the back of this manual. It is, however, easy to apply and will be useful to any anyone requiring detailed information about the data stored on the *bdf*. See also the **arcout** option of the program *CIFIO*.

Here is a description of the optional control parameters. They may be entered in any order.

4.68.1.1. file extension

This parameter is used to specify the filename extension code of the *bdf* to be listed. The default input file is the archive *bdf*. The codes **a** and **b** refer to the input and output archive *bdfs*, respectively.

4.68.1.2. prec rmin rmax

These parameters specify the range of physical records, *rmin* to *rmax*, to be listed. The default values of *rmin* and *rmax* are 1 and 10000, respectively, provided the **lrec** parameter (see below) is not specified. Specification of the **lrec** parameter overrides the **prec** parameters. If *rmax* is omitted it is assumed equal to *rmin*.

lrec lmin lmax These parameters specify the range of logical records, *lmin* to *lmax*, to be listed. The default values of *lmin* and *lmax* are 1 and *endrecord*., respectively. Specification of this parameter overrides the **prec** parameters. If *lmax* is omitted it is assumed to be equal to *lmin*. See the *Archive* section of the manual.

4.68.1.3. `npak pmin pmax`

These parameters specify the range of packets, `pmin` to `pmax`, listed from each logical record. The default values of `pmin` and `pmax` are 1 and 100, respectively. If `pmax` is omitted then it is assumed to be equal to `pmin`.

4.68.1.4. `xform`

This code switches off the format checking routine that assumes the format of each packet in a logical record is the same. This is important when examining some auxiliary bdfs (e.g. plot command files such as `ort` and `con`).

4.68.2. Output Formats

The start of each physical record or logical record is identified with a header line that lists the three 'lead words' containing the logical record length, logical record type and packet size. The words within this logical record are numbered sequentially according to their position in the *physical* record.

Each word in the bdf is then listed as either a character string, integer number, packed-integer, real number or *voidflg*:. These words are listed in their packets. The start of each packet is identified by a '>' immediately following the word numbers. The formats of the different words types are as follows:

4.68.2.1. Character strings

A word is listed as a string of characters (length *mxchwd*:) if, and only if, *every* byte in a word is one of the following characters `A-Z 0-9 : * / + - . , () < > []`. Bytes representing other characters (including `a-z`) will cause the word to be treated as one of the other types. Note that the character 'blank' will be listed as a tilde `~` (the tilde is *not* present on the bdf). *It is occasionally possible that the bits of a real number can satisfy the character test and be listed as characters.*

4.68.2.2. Integer number

A word is listed as an integer when the real number has a *remainder*, modulo 1.0, of exactly zero. It listed in the format `I5`.

4.68.2.3. Real number

A word is listed as a floating-point number provided it has a value between *-large*: and *-small*: or between *+small*: and *+large*:. The real number is listed in the format `F9.5` if it is less than 1.0, otherwise in `F9.3`.

4.68.2.4. Void number

A word is recognised as a 'void number' (i.e. value unknown) if it contains a real number equal to *voidflg*:. The void number is listed as the 'at' character `@`.

4.68.2.5. Packed integer

A word is listed as a packed integer if it is not identified as any of the above word types. This word is output as a hexadecimal integer (i.e. radix 16) of 8 digits length. No attempt is made to unpack these integers since the size of the bit slices is unknown. Note that some packed integer words may be interpreted as real numbers.

4.68.3. File Assignments

Reads a specified file (default is the input archive bdf)

4.68.4. Examples

```
VUBDF
```

List contents of the input archive bdf.

```
VUBDF file b lrec 8
```

List contents of the logical record 8 of the output archive bdf.

```
VUBDF file ort prec 23 25 xform
```

List the contents of physical records 23 to 25 from the bdf ORT.

4.69. *XTINCT*: Extinction from equivalent reflections

Authors: Nick Spadaccini and Douglas du Boulay

Contact: Syd Hall, Crystallography Centre, University of Western Australia, Nedlands 6907, Australia

This program calculates the isotropic Zachariasen extinction coefficient r^ by correlating the intensity differences of symmetry equivalent reflections with the diffraction path lengths through a nonspherical crystal. The method is optimal for complete spheres of intensity data, collected from crystals of higher symmetry.*

4.69.1. Introduction

The greater accuracy required for high precision density studies has led to the development of XTINCT. These studies require a particularly careful treatment of the correction factors for absorption and extinction, both of which depend on the size and shape of the crystal. Extinction coefficients determined from structure factor least squares processes (such as *CRYLSQ*) are approximate because of correlation with other intensity dependent errors.

XTINCT determines the isotropic extinction correction factor from the measured intensity variations of symmetrically equivalent reflections. This application relies on a non-spherical crystal in which the path lengths of symmetrically equivalent reflections are different. The precision of the method depends on several factors: the measured precision of the intensity data, the crystal shape (non-spherical but well-defined faces so that accurate path lengths can be calculated), the space group symmetry (the higher the better) and the extent of the data measured (a full sphere is highly desirable).

The correction for secondary (Type I) extinction is based on the theory of Zachariasen (1967) and the application procedure of Larson (1970). The corrected measured structure factor is $F_m^{(cor)}$.

$$F_m^{(cor)} = F_m / y^{1/2}$$

where

$$y = (1 + 2r^* |F_k|^2 Qg(\theta))^{-1/2}$$

and

$$Q = (e^4/(m^2c^4V^2)) (\lambda^3/\sin 2\theta) ((p+(1-p)\cos^4 2\theta)/(p+(1-p)\cos^2 2\theta))T$$

The function $g(\theta)$ adapts the formalism to several special cases of extinction. This is necessary to ensure that the parameter r^* is independent of scattering angle.

case 1 - Type II, primary extinction $g(\theta) = \sin 2\theta$

case 2 - Type I, secondary extinction $g(\theta) = 1$

Parameter p specifies the fraction of total intensity incident on the crystal specimen, that is polarised perpendicularly to the diffraction plane of that specimen. In the instance of neutron diffraction Q reduces simply to

$$k^2(\lambda^3/(V^2\sin 2\theta))T,$$

where the neutron scattering length $k=10^{-14}\text{m}$.

The secondary extinction coefficient r^* is determined by minimising the statistical variation of the intensities for symmetry equivalent reflections. The minimum variance is found when the first derivative of the variance quadratic is zero. The derivative is taken with respect to the extinction parameter r^* , which in the isotopic case is a scalar variable. In practice the quadratic is approximated by a Taylor expansion about r^* truncated at the second order. The zero point of the first-order differential equation is found analytically using an iterative method which terminates when the shift in r^* at each iteration becomes less than 0.0001 of $\sigma(r^*)$. The weighting of each set of equivalents is given by $1/\sigma^2$ so that the stronger reflection intensities dominate the refinement. In accordance with the theory, the stronger reflections more accurately reflect the effect of extinction.

4.69.2. Application

XTINCT works best when applied to a full sphere of measured intensities. This provides as many equivalent reflections as possible. The strong dependence of the refinement on path length variations between equivalents restricts its application to non-spherical crystals. The procedure is applied after absorption corrections and applied to clustered F squared data. Here are the calculation steps needed for the application of XTINCT. This is when extinction is applied to $|F|^2$ data. See example 4 for a typical input file which parallels these steps.

1. A full sphere of intensity data should be collected from a single non-spherical crystal and processed with the *STARTX* and *DIFDAT* steps.
2. The intensity data is then processed using *SORTRF aver 2* to calculate the merge R-factor R_{int} . *The output of this run is NOT saved.*
3. The intensity data from (1) is then corrected for absorption using the program *ABSORB*. The *tbar* option must be set so that the mean path lengths of the individual reflections are stored on the bdf. An analytical absorption calculation is recommended for the most precise path lengths.
4. The output bdf from *ABSORB* is processed with *SORTRF aver 2* to again calculate the merge R-factor R_{int} . *The output of this run is NOT saved.* This is to check the effect of the absorption corrections on the equivalent data.
5. The output bdf from *ABSORB* is processed with *SORTRF clus* to cluster equivalent reflections into sequential groups. *seprfl* is required to group Friedels separately, if a non-centrosymmetric structure. This output is saved.
6. The clustered intensities from *SORTRF* are converted to structure factors squared with *ADDREF nobay* and *reduce itof2*.
7. Then use *GENEV enot* to determine the scale factor between observed and calculated structure factors. This is essential to place the extinction coefficient r^* calculated in the next step on an absolute scale.
8. Apply the XTINCT program (a description of the input options is below).

9. Apply **ADDREF nobay** and **reduce f2toi** to convert the F squared to I.
10. Apply **SORTRF aver 2** for the fourth time to yield the final R-factor R_{int} . This provides the final measure of the effect of correcting for extinction and absorption.
11. Finally apply **ADDREF nobay** and **reduce itof** to reduce the intensities to F.

4.69.3. XTINCT Options

In the default mode XTINCT calculates, refines and applies the extinction coefficient r^* to the F^2 data. This data is output to the bdf along with the value of r^* (on the absolute scale supplied from the prior *GENEV* run). 25 reflections will be printed.

Options on the **XTINCT** line and on the **limits** line provide for various controls on this calculation. The coefficient r^* may be determined within specific regions of the data using the **limits** control line (though the resulting r^* is then applied to all data). The value of r^* may also be determined separately for each set of equivalents by entering the **sa11** option (the resulting values are output to the **pch** file). A value of r^* may be input and applied using the **appx** option. Because sets of equivalent reflections with identical t bars can result in singularities, or negative r^* , these sets are automatically skipped in the refinement. They may, however, be included using the **incl** option. Similarly, large anomalous dispersion difference between Friedel related reflections (for non-centrosymmetric spacegroups) can effect the refinement of r^* and these will normally be treated separately, but will be treated as symmetrically equivalent reflections if the option **eqfr** is entered. The polarisation factor p , if unavailable on the bdf and not included as an option on the title line, takes the default value of unpolarised radiation (0.5).

4.69.4. File Assignments

- Reads reflection data from the input archive bdf.
- Outputs corrected (or uncorrected) data on the output archive bdf.
- Optionally outputs a **pch** file.

4.69.5. Examples

```
XTINCT
```

This is the standard run which refines and applies the extinction coefficient r^* to the F^2 data. 25 reflections will be printed and the scaled r^* value will be stored on the output bdf.

```
XTINCT print eqfr  
limits f2r1 1000 100000
```

The extinction r^* will be refined on all sets of equivalents for which the average F^2 lies between 1000 and 100000. All reflections will be printed, and the Friedel related intensities will be treated as equivalent reflections.

```
XTINCT appx 153 22 print 40
```

The unscaled extinction correction r^* of 153 and a sigma r^* of 22 will be applied to the data. The first 40 reflections on the bdf will be printed.

```

SORTRF  aver 2
end      : important - do not use a copybdf without preceding end
copybdf b a  : <<<<<<<<<<<<<<<<<<< do not save output bdf
ABSORB  analyt scal 7604.19 irel tbar print 20
diff    a a c a
orient  2  0  0  29.91 27.79  0  2  0 331.86 99.84
faceml  -1  1  0  .00318
faceml   1 -1  0  .00318
faceml   1  1  0  .00364
faceml  -1 -1  0  .00364
faceml   0  0  1  .00635
faceml   0  0 -1  .00635
faceml  -5 -8 -2  .00440
faceml   5  0 -1  .00440
SORTRF  aver 2
end      : important - do not use a copybdf without preceding end
copybdf b a  : <<<<<<<<<<<<<<<<<<< do not save output bdf
SORTRF  clus
ADDRF   nobay lpin
reduce  itof2 rlp4
bdfin   all
remove  f2rl sgf2
GENEV   enot
XTINCT  irel
SORTRF  aver 2 print -9999
ADDRF   nobay ffac
reduce  itof
bdfin   all

```

4.69.6. References

- Becker, P.J. Coppens, P. (1974) *Acta Cryst* **A30**, 129-147.
- Olthof-Hazekamp, R. (1991) *Xtal3.4 Manual*, CRYLSQ.
- Larson, A.C. (1970) p292 *Crystallographic Computing*: Munksgaard.
- Maslen, E.N. Spadaccini, N. (1992) *Acta Cryst.* **A49**, 661-667.
- Spadaccini, N. (1989) *Proceedings of CRYSTAL XVI*, Lorne SCA Meeting.
- Zachariasen, W. H. (1967) *Acta Cryst* **23**, 558-564.

Notes

1. Don't do this. It assumes the missing core electron density is something resembling an inert gas atom which is a very bad approximation.
2. The promolecule is the electron density of neutral, spherical atoms, vibrating but otherwise non interacting, i.e. the standard crystallographic model used for calculating structure factors.

Appendix A. Archive file map

Data in an *Xtal* archive file is separated into blocks referred to as *logical records*. The names of these records *lrhist:*, *lrrefl:*, *lratom:*, etc. indicate the nature of the data stored therein. The length of a logical record varies depending to the type of data it holds, the nature of crystal structure being processed, and the current state of the crystallographic analysis. A logical record is subdivided into a sequence of packets, each representing a logical subset of the data. Data for a reflection measurement, or an atom site, are typical record packets. There are two types of records: fixed format (F) and directory (D) in which the first packet contains the ID numbers of data values that follow.

The system names for logical records are **lr<name>:**. However, in the text archive files (.aaa or .arc) the logical records are identified by the data names starting with <name>_.

<i>logical record</i>	<i>number</i>	<i>Type</i>	<i>Contents</i>
<i>lrhist:</i>	1	F	compound ID and calculation history
<i>lrlabel:</i>	2	F	title and labels inserted by the user
<i>lrcell:</i>	4	F	cell and cell transformation data
<i>lrsymm:</i>	5	F	space group symmetry data
<i>lrsym:</i>	6	F	space group symbols and names
<i>ltrwin:</i>	7	F	twin laws
<i>lrscat:</i>	8	D	scattering factor, atom type information
<i>lrsequ:</i>	9	F	residue sequence information
<i>lrdddef:</i>	10	F	dataset name strings
<i>lrexpl:</i>	11	D	experimental parameters per dataset
<i>lrdset:</i>	12	D	general parameters per dataset
<i>lrdict:</i>	13	D	macromolecular residue information
<i>lrgraf:</i>	14	D	molecule orientation information
<i>lratom:</i>	16	D	atom site parameters
<i>lrcons:</i>	17	D	atom constraints information
<i>lrslav:</i>	18	F	macromolecular atom link information
<i>lrmole:</i>	19	D	molecular formula and cell content data
<i>lrrefl:</i>	20	D	reflection data
<i>lrthet:</i>	21	F	precise angle data for cell dimensions
<i>lrsfls:</i>	30	D	sfls refinement results
<i>lrrfac:</i>	31	F	sfls R-factor refinement history
<i>lrdens:</i>	35	D	electron density data
<i>lrbond:</i>	40	D	bond distances
<i>lrcont:</i>	41	D	contact distances
<i>lrangl:</i>	42	D	bond angles
<i>lrldihe:</i>	43	D	dihedral angles
<i>lrrigi:</i>	44	D	rigid link constraints
<i>endrecord</i>	5535	F	terminal archive record

Only the contents of archive files are described here. Information about the contents of *auxiliary* binary data files must be obtained from the source file of the generating program.

Each logical record is listed as:

A *header line* containing:

- the *default* logical record type number,
- the logical record name,
- the logical record format (F for fixed format; D for directory access) and

- a synopsis of the information stored.

A *second line* with the packet size information.

For *fixed format records*, the column headed **SEQ** gives the packet indices for items listed. For *directory records*, the column headed **IDN** gives the identification (ID) numbers of the items that may appear in the logical record. Remember that these are the numbers which are stored in the first packet of a directory record.

A.1. 1 *hist* (F) File History

Packet Size - $50 \cdot I$ words divided into segments of I words each. *I* is the number of words needed to contain 24 characters.

Description

Segment 1 Contains the compound identification (ID) code
 Segment 2 Contains the update history of the file: the name of the program writing the file; the date on which the file was written as dd/mm/yy; and the time at which the file was written as hh:mm:ss. After 48 updates, segment 3 is lost and the latest one is added to the end of the record, as in a LIFO stack. Segment 2, showing the originating program name (and the date and time, if present), is always retained.

A.2. 2 *labl* (F) Label Information

Packet Size - Number of words needed to hold 80 characters

SEQ	Description
Packet 1	Contains date and time of creation of the bdf.
Packet 2	Contains title in force at time of creation.
Packet 3-	This, and subsequent packets, contain information entered on <i>STARTX</i> lines.

A.3. 4 *cell*(F) Cell Constants

Packet Size - 9 words

SEQ	Description
<i>Packet 1</i>	
+1	a cell dimension in Angstroms
+2	b cell dimension in Angstroms
+3	c cell dimension in Angstroms
+4	$\cos\alpha$
+5	$\cos\beta$
+6	$\cos\gamma$
+7	α in cycles ($2\pi = 1.0000$)
+8	β in cycles ($2\pi = 1.0000$)
+9	γ in cycles ($2\pi = 1.0000$)
<i>Packet 2</i>	
+1...+9	Estimated standard deviations of the quantities of packet 1
<i>Packet 3</i>	
+1...+9	Reciprocal cell constants, in same order as packet 1

Packet 4

Transformation matrix from fractional coordinates to orthogonal Angstrom coordinates as given by Rollett (Oxford Computing School, 1965), p23. Crystal axes x, y, z; orthogonal axes X, Y, Z are defined with Z along c, with Y in the b-c plane.

+1 $t_{11} = a \sin\beta \sin\gamma^*$
 +2 $t_{21} = -a \sin\beta \cos\gamma^*$
 +3 $t_{31} = a \cos\beta$
 +4 $t_{12} = 0$
 +5 $t_{22} = b \sin\alpha$
 +6 $t_{32} = b \cos\alpha$
 +7 $t_{13} = 0 \quad t_{11} \quad t_{12} \quad t_{13} \quad x \quad X$
 +8 $t_{23} = 0 \quad t_{21} \quad t_{22} \quad t_{23} \quad y = Y$
 +9 $t_{33} = c \quad t_{31} \quad t_{32} \quad t_{33} \quad z \quad Z$

Packet 5

+1...+9 Transformation matrix from Miller indices to orthogonal pseudoMiller indices

Packet 6

Miscellaneous information.

+1 cell volume
 +2 e.s.d. of cell volume
 +3 last MIR cycle number, initialized to zero
 +4 last least squares cycle number, initialized to zero
 +5 goodness of fit parameter (chi squared), initialized to 100

Packet 7

Metric tensor

+1 a^2
 +2 $a b \cos\gamma$
 +3 $a c \cos\beta$
 +4 $a b \cos\gamma$
 +5 b^2
 +6 $b c \cos\alpha$
 +7 $a c \cos\beta$
 +8 $b c \cos\alpha$
 +9 c^2

Packet 8

Inverse metric tensor.

+1...+9 In above order for reciprocal cell parameters

Packet 9-10

Protein data base orthogonal to fractional 'scale' transformation as

$s_{11} \quad s_{12} \quad s_{13} \quad u_1$
 $s_{21} \quad s_{22} \quad s_{23} \quad u_2$
 $s_{31} \quad s_{32} \quad s_{33} \quad u_3$, with the last 6 words as 0.0.

*Packet**11-12*

Protein data base transformation from fractional coordinates to orthogonal (see packets 9-10).

A.4. 5 *symm* (F) Symmetry Information

Packet Size - 12 words

SEQ**Description***Packet 1*

Contains miscellaneous information.

+1 code to indicate lattice type as:
 lattice type P I R F A B C
 noncentro cell 1 2 3 4 5 6 7
 centro cell 8 9 10 11 12 13 14
 centrosymmetric / noncentrosymmetric indicator 0 / 1

+3	total number of symmetry equivalent positions
+4	number of distinct rotation matrices and translation vectors exclusive of lattice translations and center (N)
+5	number of rotation matrices of identical pattern of zeros
+6	cell multiplicity factor to place A and B parts of the structure factor on the scale of <i>International Tables (Vol. 1)</i> .
+7	number of matrices, M, stored for finding neighbour lattice interactions in CEDAR
+8	number of molecules in the cell (<i>see also lrmole</i> .)
+9	crystal and Bravais systems (tri=1; mon=2; ort=3; tet=4; trg=5; rho=6; hex=7; cub=8)
+10	cell unique axis number (1=a, 2=b, 3=c, 4=none)
Packet 2	Rotation matrix and translation vector for 1st equivalent position.
+1...+12	r11, r21, r31, r12, r22, r32, r13, r23, r33, t1, t2, t3
Packet 3 to N+1	for the remaining N equivalent positions. The maximum value of N is 24. Matrices involving an inversion centre or non-primitive translations are excluded.
Packet N+2 to M+1	additional equivalent positions used by CEDAR.

A.5. 6 *sgnm* (F) Space Group Names

Packet Size - Words to hold 40 characters

Packet 1	Hall space group symbol
Packet 2	Hermann-Mauguin space group symbol
Packet 3	Space group number from International Tables (<i>see Symmetry</i>)

A.6. 7 *twin* (F) Twin Laws

Packet Size - 13 words

SEQ	Description
+1...+9	Rotation matrix to relate Miller indices of a reflection to those of its twin-related reflection. $r_{11}, r_{12}, r_{13}, r_{21}, r_{22}, r_{23}, r_{31}, r_{32}, r_{33}$ $h' = (r_{11} \ r_{12} \ r_{13}) h$ $k' = (r_{21} \ r_{22} \ r_{23}) k$ $l' = (r_{31} \ r_{32} \ r_{33}) l$ <i>Note that if the space group is non-centrosymmetric, each packet corresponds to two twin laws, i.e. the given matrix multiplied by +1 and by -1.</i>
+10	twin volume ratio
+11	esd of twin volume ratio
+12	volume ratio of the inverse (if non-centrosymmetric)
+13	esd of volume ratio (if non-centrosymmetric)

A.7. 8 *scat* (D) Atom-type Parameters

Packet Size - Variable (one packet per atom type)

IDN	Description
1	number of atoms of this type per unit cell
2	atomic weight
3	atomic number
4	number of electrons in neutral atoms or ions
5	atomic bond radius in Angstroms
6	atomic contact radius in Angstroms
7	flag if interpolated scat. factors are stored in <i>refl</i> 0/1 = no/yes
9	effective spin quantum number
10	neutron scattering length in cm*10 ⁻¹²
11	atom type symbol ₁ (8 characters)
12-20	Cromer-Mann coefficients a ₁ , b ₁ , a ₂ , b ₂ , a ₃ , b ₃ , a ₄ , b ₄ , c
21-45	real part of dispersion scattering factor for data sets 1 to 25
61-85	imaginary part of dispersion scattering factor for data sets 1 to 25
95	source of scattering data ₁ (40 characters)
98	mass percentage of element from chemical analysis
99	e.s.d. of mass percentage in IDN 98
100	atomic scattering factor ₂ at s = 0.00
101	atomic scattering factor at s = 0.01
102	atomic scattering factor at s = 0.02
...
299	atomic scattering factor ₃ at s= 1.99

1 This ID number will be repeated for each word required to contain the character string.

2 Scattering factors at s-intervals of 0.01 need NOT be present. The fineness of the table is optional.

3 Additional scattering factors for a given atom type are stored in 300-499, 500-699, 700-899, ...

A.8. 9 *sequ* (F) Protein Atom Sequence Information

Packet Size - words needed to hold 8 characters

SEQ	Description
+1	8 character containing residue type (4 char) and sequence position (4char). Contains a packet for each residue in the sequence.

A.9. 10 *ddef* (F) Dataset Names

Packet Size - words needed to hold 12 characters

SEQ	Description
+1	Contains string of 12 characters used to describe the dataset. The order of the packets corresponds to dataset number. That is, the first packet contains the description of dataset 1, etc. Datasets may be defined as isomorphs, graphs of partial structures, or residues.

A.10.11 *expl* (D) Experimental Parameters

Packet Size - Variable (one packet per data set)

IDN	Description
1	data set key (1 designates dataset 1, etc.)
2	wavelength (weighted mean) in Angstroms
3	wavelength line1 in Angstroms
4	wavelength line2 in Angstroms
5	wavelength line3 in Angstroms
6	relative weight wavelength line1
7	relative weight wavelength line2
8	relative weight wavelength line3
10	linear absorption coefficient in 1/mm
11	sort mode applied by SORTRF
12	sorting order of hkl as slow/med/fast (<i>312 is l slow, k fast</i>)
13-15	a, b, c cell dimensions in Angstroms
16-18	$\cos\alpha$, $\cos\beta$, $\cos\gamma$
19-21	α , β , γ in cycles ($2\pi = 1.0000$)
22	cell volume
23-25	e.s.d. of a, b, c in 13-15
26-28	e.s.d. of α , β , γ in 19-21
29	e.s.d. of cell volume in 22
31	sense of rotation of chi circle; +1 for clockwise, -1 anticlockwise
32	sense of rotation of phi circle ₁
33	sense of rotation of theta or two theta circle
34	sense of rotation of omega circle
35	reflection scanning method used during data collection 1= theta two theta 2= Wyckoff 3= omega
36	hkl permitted sign pattern as packed word (0=-; 1=+)
40	diffractometer type (1 char) E=Enraf Nonius, G=GE-XRD, H=Hilger Watts, P=Picker, S=Siemens (Nicolet/Syntex) T=Stoe, W=Philips, O=other.
41-49	orientation matrix UB (11, 12, 13, ..., 33)
50	monochromator two theta angle (in degrees)
51	monochromator perfection factor (0 to 1.)
52	polarization ratio for incident beam on crystal $P_{\text{perp}}/P_{\text{total}}$
53	e.s.d. of polarization ratio
54	refinement weight of polarization ratio
55	dihedral angle between monochromator and diffraction planes
56	incident beam half width and intensity
57	polarization factor of source beam (0.0 to 1.0)
81	temperature of cell and intensity measurement in Kelvin
82	pressure of cell and intensity measurement in GPa
85	experimentally measured crystal density
86	e.s.d. of 85
87	calculated crystal density
88	e.s.d. of 87
89	wavelength of radiation used in cell refinement
90	number of hkl used in cell refinement
91	min theta for cell determination
92	max theta for cell determination
93	radiation used in cell and intensity measurement as an element symbol or source (20 chars)

94	type of radiation used in cell and intensity measurement (1=neutrons,2=electrons, 3=X-ray Mo K α , 4=X-ray Cu K α , 5=X-ray Ag K α ,6=X-ray synchrotron)
100	number of scale groups
101..100+n	F relative scale factor for scale group n (max=64)
201..200+n	scale group numbers associated with the scale factors
300	Number of crystal faces
301-303	h, k, l of crystal face 1
304	distance of crystal face 1 from centre in mm.
305	angle phi of crystal face 1
306	angle chi of crystal face 1
307	angle kappa of crystal face 1
308	e.s.d. of distance (IDN 304)
309	weight of distance (IDN 304) in refinement
311-319	as for 301-309 for crystal face 2, up to 20 faces
501-502	min and max h for measured intensities before averaging
503-504	min and max k for measured intensities before averaging
505-506	min and max l for measured intensities before averaging
507-508	min and max θ for measured intensities in degrees
511	intensity measurement method (40 chars)
512	radiation monochomator description (40 chars)
513	absorption method applied (40 chars)
514	intensity averaging process (40 chars)
515	intensity reduction process (40 chars)
521	total number of reflections measured
522	number of reflections after averaging of equivalents
523	averaged coefficient Y (1=F; 2=F ₂ ; 3=I)
524	R-factor of averaged coefficients (sum of Y diff)/(sum of Y)
525	(sum of sigma(Y))/(sum of Y)
526	absorption transmission T min
527	absorption transmission T max
531-535	criteria used for rcodes 1 to 5 (40 chars each)
536-540	number of reflections assigned rcodes 1 to 5
541	radius of crystal sphere/cylinder in mm
542	e.s.d. of radius
543	refinement weight of crystal radius
544	length of crystal cylinder in mm
545	e.s.d. of crystal length
546	refinement weight of crystal length
547	counter dead time in microseconds
548	e.s.d. dead time
549	refinement weight of dead time
550	wavelength of beta-filter absorption edge
551	crystal size max in mm
552	crystal size mean in mm
553	crystal size min in mm
560	crystal colour (20 characters)
561	crystal shape (20 characters)
601	number of standard reflections

602	number of reflections measured between standards
603	standards decay ratio
604	time (in minutes) between standards
610-612	h, k, l of standard reflection 1
613-615	h, k, l of standard reflection 2, up to 30 standards
701	number of attenuator filters
702	attenuator filter index for filter 1
703	reciprocal transmission factor (> 1.0) for filter 1
704	e.s.d. in rec. transmission factor for filter 1
705	refinement weight of rec. transmission for filter 1
706-709	like 702-705 for filter 2

1 See program *ABSORB* for the definition of angles.

A.11. 12 *dset* (D) General Data per Dataset

Packet Size - Variable (one packet per data set)

IDN	Description
1	dataset key (1 designates dataset 1, etc.)
2	overall thermal displacement parameter, UOV, in A_2
3	Flack X-absolute configuration parameter (<i>see also lrsfls:</i>)
4	Rogers absolute configuration parameter (<i>see also lrsfls:</i>)
10	packed word of GENEV fragment types
11-13	max h , k , l
14-15	min and max $\sin(\theta)/\lambda$
16	minimum value of E1 used for psi(0) calculation
17	minimum value of E2 used for psi(0) calculation
18	maximum number of reflections used in E calculations
30	relative scale, data set to parent
31	relative thermal parameter, ΔU , data set to parent
32	closure error
33	closure error, anomalous
34	number of $\sin(\theta)/\lambda$ bins
35	number of F squared bins
36	Matthews k value from intercept
37	slope of Matthews plot
61-69	hkl transformation matrix used to convert measured hkl to the indices of the unique data set (t11, t21,, t33)
71-72	min and max of h for unique data set
73-74	min and max of k for unique data set
75-76	min and max of l for unique data set
77-78	min and max of θ for unique data set in degrees
79	high resolution d ($2/\sin\theta(\max)$)
80	low resolution d ($2/\sin\theta(\min)$)
81	number of reflections used in refinement
82-86	numbers of reflections with rcodes 1 to 5 used in refinement
87	threshold factor applied to $\sigma(Y)$ to classify rcode2 reflections
88	coefficient Y used in threshold test (1=F; 2=F ₂ ; 3=I)

89	N(Laue symmetry) (in PAKFRL mode =total number of reflection packets)
90	N(Crystal class) - N(Laue symmetry) (in PAKFRL mode =total number of reflection packets - rcode 4 reflections - item 89)
91	_reflns_Friedel_coverage = item 90/ item 89
101	extinction type (0=Zach.; 1=iso Becker-Coppens 1; 2=iso B-C 2; 3=isoB-C 1+2; 4=aniso B-C 1; 5=aniso B-C 2; 6= aniso B-C 1+2)
102	distribution (0=Gaussian; 1=Lorentzian)
103	isotropic extinction type1 parameter g (see also <i>lrsfls</i> : 34)
104	isotropic extinction type2 parameter ρ (see also <i>lrsfls</i> : 35)
105-110	anisotropic extinction type1 parameters
111-116	anisotropic extinction type2 parameters
117	mean Tbar (in mm)
200-249	closure errors for shells (bins)
300-349	anomalous closure errors for shells

A.12. 13 *dict*(D) Macromolecular Residue Information

Packet Size - Variable

IDN	Description
1	residue name (4 characters)
2	atom name (4 characters, first part is scattering factor name)
3	relative back chain bond
4	relative furthest forward chain bond
5	group KEY (1=start group; 0=same group as back chain group; -1=no group; -9=dummy atom)
6	rotational angle energy value
7	number of rotational barriers
8	angle of minimum energy
9	backward bond length
10	backward bond angle
11	backward dihedral angle
12	energy associated with rotation about bond
13	energy associated with bond angle distortion
14	energy associated with dihedral distortion
15	energy associated with second dihedral distortion
16	value of second dihedral angle
17	Lennard-Jones coefficient A
18	Lennard-Jones coefficient B
19	Lennard-Jones coefficient C
20	special position KEY (-3=water oxygen; -2= - charged atom; +2=+charged atom; +1=other polar group; 0=non-polar atom)

A.13. 14 *graf*(D) Molecular Graphing Information

Packet Size - Variable

IDN	Description
21-29	Orientation matrix of cell axes to viewing axes. Generated by PIGand used by ORTEP

A.14. 16 atom (D) Atom Parameters

Packet Size - Variable

IDN	Description
1-3	x, y, z parameters in fractions of unit cell
4	individual isotropic thermal parameter U
5-10	anisotropic thermal parameters U11, U22, U33, U12, U13, U23
11	population parameter
12	anomalous population parameter
13	neutron scattering factor
14	atom label (24 characters)
15	residue code (4 characters) char 1: remoteness indicator char 2: branch designator char 3: sub-branch designator char 4: alternate location indicator
16	sequence code (4 characters) chars 1-3: least significant digits of residue sequence number char 4: residue code
17	data set number
18	chain code (4 characters) chars 1-3: chain identifier char 4: most significant digit of residue sequence number
19	residue name (4 characters)
20	atom serial number for Protein Data Bank
21	atom multiplicity for atoms in special positions
22	scattering factor pointer as a packet sequence number of <i>scat</i>
23	thermal displacement parameter type (0=overall; 1=iso; 2=aniso)
24	atom-group key for group refinements
25	model-refinement key for refining different model
26	number of valence electrons - added in MAPXCH
50	specified coordinates flag (0=no; 1= geometry fixed; 2=dummy)
51	atom label of attached atom (for geometry fixed hydrogen sites)
52	number of attached hydrogen atoms <i>not</i> present in <i>atom record</i>
55	refinement restrictions associated with this site (8 characters). String may be made up of the following flags: S special position constraint G rigid group refinement R riding atom site attached to non-riding atom site D distance or angle restraint T thermal displacement constraint U Uiso or Uij restraint due to rigid body P partial occupancy restraint
56	special information about this site (40 characters)
57	description of constraints to site parameters (40 characters)
58	description of restraints to site parameters (40 characters)
101-103	sigma x, sigma y, sigma z
104	sigma Uiso
105-110	sigma of U11, U22, U33, U12, U13, U23
111	sigma of population parameter

112	sigma of anomolous population parameter
113	sigma of neutron scattering
210	atom number in dictionary <i>dict</i>
212	pointer to atom number in <i>atom</i> of the back chain
213	pointer to atom number in <i>atom</i> of the furthest forward chain
214	mobility indicator
215	KEY for group/non-group/slave
300	6-char code for geometric constraint (same as CALCAT)
301	Bond distance to parent atom site
302	Alpha 1
303	Alpha 2
310	24-char label of sites used in the geometric constraint
311	Encoded symop/translation (if not 1 555)
312-317	as for 310 and 311 for three additional site specifiers
320-323	24-char label of sites in the same rigid group (eg. methyl)

A.15. 17 cons (D) Refinement Constraints

Packet Size - Variable

IDN	Description
1	atom label ₁ of <i>subject</i> atom in <i>cons</i>
2	parameter number of <i>subject</i> atom; 1 to 9 for x to U23
3	multiplication factor of <i>subject</i> parameter
4	constant, Q, in constraint equation
5	constraint classification key ₂
6	site multiplicity of <i>subject</i> atom
11	atom label ₁ of the <i>reference</i> atom 1
12	parameter number of <i>reference</i> atom 1; 1 to 9 for x to U23, 0 for invariant
13	multiplication factor for <i>reference</i> parameter of atom 1
21	atom label of <i>reference</i> atom 2
22	parameter number of <i>reference</i> atom 2
23	multiplication factor for <i>reference</i> parameter of atom 2
..
n1-n3	atom label, parameter, and multiplicity factor for atom n

1 The general form of the constraint equation is: $p(s)*f(s) = q + p(r1)*f(r1) + p(r2)*f(r2) + \dots + p(rn)*f(rn)$ The ID numbers 1, 11, 21, etc. may appear in more than one word if more than one real word is required to store 24 chars.

2 Constraint classification KEY = 1 CONSTRAINED Constraint classification KEY = 3 INVARIANT Constraint classification KEY = 4 Polar Axis Direction For a constraint of KEY = 4, the axis is stored in ID 2 as 1=a, 2=b, or 3=c. The rest of the packet is null.

A.16. 18 slav (F) Protein Atom Link Information

Packet Size - 2 words

Description

This record contains the number of the master atom and slave atom as used in the refinement of macromolecular structures. It contains cross linkage information which supplements the sequence information in *sequ*. For example, information about disulfide linkage, metal cluster linkage, or closed rings such as phenyls, etc., will be found here.

A.17. 19 *mole* (D) Molecular Information

Packet Size - Variable (one packet per data set)

IDN	Description
1	data set number (1 for data set 1, etc.)
11	molecular formula as char string (72 chars)
12	calculated molecular or formula mass in Daltons
13	molecular or formula Z (see also <i>lrsymm</i> .)
14	measured molecular or formula mass in Daltons
15	e.s.d. in formula mass
21	calculated density (see also <i>lrexpl</i> :87)
22	F(000)
23	E(000)

A.18. 20 *reft* (D) Reflection Information

Packet Size - Variable

IDN	Description
1 - 999	crystal-specific information
1000 -	dataset 1 information
1999	
2000 -	dataset 2 information
2999	
.....
n000 -	dataset n information
n999	

IDN	Crystal Specific
1	Miller indices packed word, with bit pattern 29-21 20-12 11-3 2-0 h k l <i>sign code</i> hkl +++ +-+ +-- +--- -+- -+- --- --- <i>sign code</i> 0 1 2 3 4 5 6 7
2	sin(theta)/lambda
3	reflection symmetry data, with bit pattern 15-12 11-5 4-0 <i>restricted phase</i> epsilon <i>symm.</i> hkl multiplicity <i>code</i> reinforce. factor <i>restr. phase code</i> is 0=no restriction; 1=0/180; 2=30/210; 3=45/225; 4=60/240; 5=90/270; 6=120/300; 7=135/315; 8=150/330.
501	interpolated scattering factor for atom type1
502	interpolated scattering factor for atom type2
338	

... ..
 510 interpolated scattering factor of atom type 10
 511-699 interpolated scattering factors for many other atom types or special scattering factor calculations

The 700 numbers are used to store estimated phase sets for the "native" structure or "parent" substance.

700 Figure of merit; weight of the 'best' Fourier coefficient
 701 cos(alpha) for the 'best' Fourier coefficient
 702 sin(alpha) for the 'best' Fourier coefficient
 703 cos(alpha), most probable
 704 sin(alpha), most probable
 705-709 alternate phase set 2

 795-799 alternate phase set 20

Overview for dataset n

n000-n199 measurement parameters
 n200-n299 reduction parameters
 n300-n499 reduced structure factor data
 n500-n599 Hendrickson coefficient data
 n600-n699 normalized structure factor data
 n700-n799 structure factor phase data
 n800-n899 refined structure factor data
 n900-n999 refinement parameters

IDN Dataset 1

1000 total gross counts
 1001 total background counts
 1002 ratio of scan to background time
 1003 net counts
 1004 sigma(net counts)
 1005 phi diffractometer angle in degrees
 1006 chi or kappa
 1007 omega
 1008 2-theta
 1009 2-theta scan range
 1010 omega scan range
 1011 psi azimuthal scan angle in degrees
 1012 reciprocal scan speed (actual scan speed in degrees/min is 20.1166 divided by this value)
 1013 number of contiguous points used in determining the intensity of the reflection by a modified Wyckoff step scan
 1014 the time in seconds from the beginning of intensity data collection to the end of the current measurement
 1015 goniometer status code (*=no problems; C=position results in collision; D=peak is miscentered; X=position is not accessible with machine geometry)
 1016 intensity status code (*=normal intensity; W=too weak to measure S=too strong without attenuator; T=too strong even with attenuator)
 1017 SPD scan speed in degrees/min
 1018 RFT type of reflection; 0=regular observation; 1=standard reference; 2=Phi spin for absorption; 3=special standard reference
 1020 SCT scan time min
 1021 SCC scan counts
 1022 BGT background time min

1023	BGL low angle background counts
1024	BGH high angle background counts
1025	EMR
1026	ATN attenuator factor (0 means no attenuator)
1027	RFN reflection sequence #
1028	RSB
1029	RMB
1031	calculated maximum step intensity
1032	calculated offset in maximum step intensity
1033	calculated peak width parameter
1034	calculated background at step scan center
1035	calculated background slope
1041-1045	sigmas for 1031-1035
1046	background scan mode (0=stationary, otherwise scanned)
1047	Cp
1048	Cb Flack et al. JAC (1992) 25, 455-495.
1049	Cl
1050	Ch
1051	SCNTP Scan type (Omega-SCNTP*theta) i.e. SCNTP = 0 means an omegascan and SCNTP=2 means an omega-2theta scan
1052	detector horizontal aperture in degrees
1053	detector vertical aperture in degrees
1100	numbers of steps in a step scan
1101-1199	step counts (actual number given in 1100)
1200	absorption weighted mean path length Tbar (mm)
1201	absorption correction factor to I(rel)
1202	extinction correction factor to I(rel)
1203	thermal diffuse scattering correction factor to I(rel)
1204	1/Lp factor
1205	I(rel) scale factor to scale counts to I(rel)
1210	wavelength in Angstroms for this reflection
1210+n	derivative of the transmission factor A with respect to the distance from face n
1300	relative intensity I(rel)
1301	sigma I(rel)
1302	relative F squared F ₂ (rel)
1303	sigma F ₂ (rel)
1304	relative F F(rel)
1305	sigma F(rel)
1308	rancode reflection status key (1=obs; 2=less-than; 3=extinction,4=Friedel mate is missing from Friedel pair; 5=systematically absent)
1309	scale group number
1310-1319	identical to 1300-1309 but for Friedel related reflection
1320	RFO
1321	RST
1322	RPS
1501-1504	A, B, C, D Hendrickson coeff. Phase prob distr. (isomorphous)
1505-1508	A, B, C, D Hendrickson coeff. Phase prob distr. (anomalous)
1510	calculated F from density modification methods
1511	calculated phase (in cycles) from density modification methods
1600	normalized structure factor, E1
1601	normalized structure factor, E2
1602	standard deviation of E1
1603	standard deviation of E2
1604	expectation value for F ₂ ; assuming random atoms
1605	expectation value for F ₂ ; from fragment information
1606-1630	group structure factors in sequence designated by <i>cons</i>

1631	structure factor phase estimate 1 (in cycles)
1632	weight of structure factor phase estimate 1
1633	structure factor phase estimate 2 (in cycles)
....
1694	weight of structure factor phase estimate 32
1700	calculated structure factor phase (in cycles) from 1801 to 1802
1701	relative F corrected for dispersion and extinction (i.e. dispersion and extinction effects removed)
1702	scaled corrected relative F 1701
1703	scaled relative sigma F 1305
1705	least squares refinement status (1=incl.; 2=excl.)
1800	calculated F no dispersion or extinction terms applied
1801	calculated A i.e. 1800 x cosine of 1700;
1802	calculated B i.e. 1800 x sine of 1700;
1803	calculated real dispersion component A(d)
1804	calculated imag dispersion component B(d)
1805	1801 with dispersion terms included (no extinction terms)
1806	1802 with dispersion terms included (no extinction terms)
1807	translation function coefficient
1810-1817	partial structure factor values in order of 1800-1807
1820	calculated F with dispersion and extinction applied and scaled
1900	square root of least squares weight last used
1901	square root of least squares weight1
1902	square root of least squares weight2
1903	square root of least squares weight3
1910	Sim weight for partial structure

A.19. 21 *thet*(F) Precise Angle Data for Cell Dimensions

Packet Size - 6 words

SEQ	Description
+1	data set number
+2	h
+3	k
+4	l
+5	2 θ in degrees
+6	wavelength index (0 = mean wavelength, 1 = α_1 , 2 = α_2)

A.20. 30 *sfls* (D) Structure Factor Least Squares Data

Packet Size - Variable

IDN	Description
1	data set number
11	refinement process description (40 chars)
12	SFLS weighting scheme description (40 chars)
13	SFLS extinction process description (40 chars)
14	SFLS matrix description (40 chars)
15	SFLS treatment of hydrogen atoms (40 chars)

16	SFLS treatment of absolute structure (40 chars)
17	SFLS extinction equation (40 chars)
21	R-value based on F diff for all data
22	R-value based on F diff for rcode1 data only
25	wR-value based on weighted Y diff for all data
26	wR-value based on weighted Y diff for rcode1 data
27	SFLS goodness of fit parameter for all data
28	SFLS goodness of fit parameter for rcode1 data
29	max SFLS shift/error
30	mean SFLS shift/error
31	Roger's abs. configuration measure NU
32	Flack's abs. configuration measure x
33	refined extinction parameter g or ρ
34	sigma g or ρ
35	sigma for Roger's abs. configuration measure NU
36	sigma for Flack's abs. configuration measure x
41	structure factor coefficient refined (F=1; F ₂ =2; I=3)
42	number of SFLS parameters refined
43	number of SFLS reflections contributed to the matrix
44	number of SFLS restraints applied
45	number of SFLS constraints applied

A.21. 31 *rfac* (F) Refinement History

Packet Size - Words needed for 40 characters, plus five words.

SEQ	Description
+1	Current count of SFLS cycles
+2	R factor $\sum F / \sum F $ for rcode 1 reflections
+3	R factor $\sum wF / \sum w F $ for rcode 1 reflections
+4	R factor $\sum wF^2 / \sum w F ^2$ for rcode 1 reflections
+5	Goodness of fit $\sum wF^2 / (N_{ref}-N_{var})$ for rcode 1 reflections
+6	Title used during least squares run (40 characters)

A.22. 35 *dens*(D) Electron Density Data

Packet Size - Variable

IDN	Description
11	min density in final difference map
12	max density in final difference map

A.23. 40 *bond*(D) Bond Distances

Packet Size - Variable

IDN	Description
-----	-------------

11	atom 1 label
12-14	atom 1 fractional coords x,y,z
15	atom 1 symop number (as in STARTX)
16	atom 1 packed cell translations for origin at 555
21	atom 2 label
22-24	atom 2 fractional coords x,y,z
25	atom 2 symop number
26	atom 2 packed cell translations for origin at 555
91	distance in Angstroms (calculated)
92	e.s.d. of distance (calculated)
93	distance in Angstroms (observed)
94	e.s.d. of distance (observed)
95	weight of restraint
101-112	atom 1 symmetry elements r ₁₁ r ₂₁ r ₃₁ r ₁₂ r ₂₂ r ₂₃ r ₃₁ r ₃₂ r ₃₃ t ₁ t ₂ t ₃
201-212	atom 2 symmetry elements

A.24. 41 *cont* (D) Contact Distances

Packet Size - Variable

IDN	Description
11	atom 1 label
12-14	atom 1 fractional coords x,y,z
15	atom 1 symop number
16	atom 1 packed cell translations for origin at 555
21	atom 2 label
22-24	atom 2 fractional coords x,y,z
25	atom 2 symop number
26	atom 2 packed cell translations for origin at 555
91	distance in Angstroms
92	e.s.d. of distance
101-112	atom 1 symmetry elements r ₁₁ r ₂₁ r ₃₁ r ₁₂ r ₂₂ r ₂₃ r ₃₁ r ₃₂ r ₃₃ t ₁ t ₂ t ₃
201-212	atom 2 symmetry elements

A.25. 42 *angl* (D) Bond Angles

Packet Size - Variable

IDN	Description
11	atom 1 label
12-14	atom 1 fractional coords x,y,z
15	atom 1 symop number
16	atom 1 packed cell translations for origin at 555
21	atom 2 label (at the apex)
22-24	atom 2 fractional coords x,y,z
25	atom 2 symop number
26	atom 2 packed cell translations for origin at 555
31	atom 3 label
32-34	atom 3 fractional coords x,y,z

35	atom 3 symop number
36	atom 3 packed cell translations for origin at 555
91	angle in degrees (calculated)
92	e.s.d. of angle (calculated)
93	angle in degrees (observed)
94	e.s.d. of angle (observed)
95	weight of restraint
101-112	atom 1 symmetry elements r_{11} r_{21} r_{31} r_{12} r_{22} r_{23} r_{31} r_{32} r_{33} t_1 t_2 t_3
201-212	atom 2 symmetry elements
301-312	atom 3 symmetry elements

A.26. 43 *dihe* (D) Dihedral Angles

Packet Size - Variable

IDN	Description
11	atom 1 label
12-14	atom 1 fractional coords x,y,z
15	atom 1 symop number
16	atom 1 packed cell translations for origin at 555
21	atom 2 label
22-24	atom 2 fractional coords x,y,z
25	atom 2 symop number
26	atom 2 packed cell translations for origin at 555
31	atom 3 label
32-34	atom 3 fractional coords x,y,z
35	atom 3 symop number
36	atom 3 packed cell translations for origin at 555
41	atom 4 label
42-44	atom 4 fractional coords x,y,z
45	atom 4 symop number
46	atom 4 packed cell translations for origin at 555
91	angle in degrees (calculated)
92	e.s.d. of angle (calculated)
93	angle in degrees (observed)
94	e.s.d. of angle (observed)
95	weight of restraint
101-112	atom 1 symmetry elements r_{11} r_{21} r_{31} r_{12} r_{22} r_{23} r_{31} r_{32} r_{33} t_1 t_2 t_3
201-212	atom 2 symmetry elements
301-312	atom 3 symmetry elements
401-412	atom 4 symmetry elements

A.27. 44 *rigi*(D) Rigid Link Restraint Information

Packet Size - Variable

IDN	Description
11	atom 1 label
12-14	atom 1 fractional coords x,y,z

15	atom 1 symop number
16	atom 1 packed cell translations for origin at 555
21	atom 2 label
22-24	atom 2 fractional coords x,y,z
25	atom 2 symop number
26	atom 2 packed cell translations for origin at 555
91	calculated rigid link
92	esd of calculated rigid link
93	observed rigid link (O)
94	variance of observed rigid link
95	rigid link weight
101-112	atom 1 symmetry elements r_{11} r_{21} r_{31} r_{12} r_{22} r_{23} r_{31} r_{32} r_{33} t_1 t_2 t_3
201-212	atom 2 symmetry elements

A.28. 65535 endrecord: (F) EOF Record

Packet Size - 0

Appendix B. Symmap file

```
#
# table of ralf w. grosse-kunstleve, eth, zuerich.
#
# version june 1995
#   updated   september 29 1995
#   updated   july       9 1997
# last updated july      24 1998
data_notation
```

```
loop_
_monoclinic_extension # cf. _symmetry_space_group_id
_monoclinic_axis      # cf. it vol. a 1983 sec. 2.16.
_monoclinic_setting   # cf. it vol. a 1983 tab. 2.16.1.
_monoclinic_cellchoice # cf. it vol. a 1983 sec. 2.16.(i) & fig. 2.6.4.
```

b	b	abc	1
b1	b	abc	1
b2	b	abc	2
b3	b	abc	3
-b	b	c-ba	1
-b1	b	c-ba	1
-b2	b	c-ba	2
-b3	b	c-ba	3
c	c	abc	1
c1	c	abc	1
c2	c	abc	2
c3	c	abc	3
-c	c	ba-c	1
-c1	c	ba-c	1
-c2	c	ba-c	2
-c3	c	ba-c	3
a	a	abc	1
a1	a	abc	1
a2	a	abc	2
a3	a	abc	3
-a	a	-acb	1
-a1	a	-acb	1
-a2	a	-acb	2
-a3	a	-acb	3

```
loop_
_symmetry_space_group_id
_symmetry_space_group_name_sch
_symmetry_space_group_name_h-m # recognised iucr cif data names
_symmetry_space_group_name_hall # recognised iucr cif data names
```

1	c1^1	p_1	p_1	#deprecated Hall symbols
2	ci^1	p_-1	-p_1	
3:b	c2^1	p_1_2_1	p_2y	
3:b	c2^1	p_2	p_2y	

3:c	c2^1	p_1_1_2	p_2	
3:a	c2^1	p_2_1_1	p_2x	
4:b	c2^2	p_1_21_1	p_2yb	
4:b	c2^2	p_1_21_1	p_2y1	
4:b	c2^2	p_21	p_2yb	
4:c	c2^2	p_1_1_21	p_2c	
4:c	c2^2	p_1_1_21	p_21	
4:a	c2^2	p_21_1_1	p_2xa	
4:a	c2^2	p_21_1_1	p_2x1	
5:b1	c2^3	c_1_2_1	c_2y	
5:b1	c2^3	c_2	c_2y	
5:b2	c2^3	a_1_2_1	a_2y	
5:b3	c2^3	i_1_2_1	i_2y	
5:c1	c2^3	a_1_1_2	a_2	
5:c2	c2^3	b_1_1_2	b_2	
5:c3	c2^3	i_1_1_2	i_2	
5:a1	c2^3	b_2_1_1	b_2x	
5:a2	c2^3	c_2_1_1	c_2x	
5:a3	c2^3	i_2_1_1	i_2x	
6:b	cs^1	p_1_m_1	p_-2y	
6:b	cs^1	p_m	p_-2y	
6:c	cs^1	p_1_1_m	p_-2	
6:a	cs^1	p_m_1_1	p_-2x	
7:b1	cs^2	p_1_c_1	p_-2yc	
7:b1	cs^2	p_c	p_-2yc	
7:b2	cs^2	p_1_n_1	p_-2yac	
7:b2	cs^2	p_n	p_-2yac	
7:b3	cs^2	p_1_a_1	p_-2ya	
7:b3	cs^2	p_a	p_-2ya	
7:c1	cs^2	p_1_1_a	p_-2a	
7:c2	cs^2	p_1_1_n	p_-2ab	
7:c3	cs^2	p_1_1_b	p_-2b	
7:a1	cs^2	p_b_1_1	p_-2xb	
7:a2	cs^2	p_n_1_1	p_-2xbc	
7:a3	cs^2	p_c_1_1	p_-2xc	
8:b1	cs^3	c_1_m_1	c_-2y	
8:b1	cs^3	c_m	c_-2y	
8:b2	cs^3	a_1_m_1	a_-2y	
8:b3	cs^3	i_1_m_1	i_-2y	
8:b3	cs^3	i_m	i_-2y	
8:c1	cs^3	a_1_1_m	a_-2	
8:c2	cs^3	b_1_1_m	b_-2	
8:c3	cs^3	i_1_1_m	i_-2	
8:a1	cs^3	b_m_1_1	b_-2x	
8:a2	cs^3	c_m_1_1	c_-2x	
8:a3	cs^3	i_m_1_1	i_-2x	
9:b1	cs^4	c_1_c_1	c_-2yc	
9:b1	cs^4	c_c	c_-2yc	
9:b2	cs^4	a_1_n_1	a_-2yab	# a_-2yac
9:b3	cs^4	i_1_a_1	i_-2ya	
9:-b1	cs^4	a_1_a_1	a_-2ya	
9:-b2	cs^4	c_1_n_1	c_-2yac	# c_-2ybc
9:-b3	cs^4	i_1_c_1	i_-2yc	
9:c1	cs^4	a_1_1_a	a_-2a	
9:c2	cs^4	b_1_1_n	b_-2ab	# b_-2bc
9:c3	cs^4	i_1_1_b	i_-2b	

9:-c1	cs ⁴	b_1_1_b	b_-2b	
9:-c2	cs ⁴	a_1_1_n	a_-2ab	# a_-2ac
9:-c3	cs ⁴	i_1_1_a	i_-2a	
9:a1	cs ⁴	b_b_1_1	b_-2xb	
9:a2	cs ⁴	c_n_1_1	c_-2xac	# c_-2xbc
9:a3	cs ⁴	i_c_1_1	i_-2xc	
9:-a1	cs ⁴	c_c_1_1	c_-2xc	
9:-a2	cs ⁴	b_n_1_1	b_-2xab	# b_-2xbc
9:-a3	cs ⁴	i_b_1_1	i_-2xb	
10:b	c2h ¹	p_1_2/m_1	-p_2y	
10:b	c2h ¹	p_2/m	-p_2y	
10:c	c2h ¹	p_1_1_2/m	-p_2	
10:a	c2h ¹	p_2/m_1_1	-p_2x	
11:b	c2h ²	p_1_21/m_1	-p_2yb	
11:b	c2h ²	p_1_21/m_1	-p_2y1	
11:b	c2h ²	p_21/m	-p_2yb	
11:c	c2h ²	p_1_1_21/m	-p_2c	
11:c	c2h ²	p_1_1_21/m	-p_21	
11:a	c2h ²	p_21/m_1_1	-p_2xa	
11:a	c2h ²	p_21/m_1_1	-p_2x1	
12:b1	c2h ³	c_1_2/m_1	-c_2y	
12:b1	c2h ³	c_2/m	-c_2y	
12:b2	c2h ³	a_1_2/m_1	-a_2y	
12:b3	c2h ³	i_1_2/m_1	-i_2y	
12:b3	c2h ³	i_2/m	-i_2y	
12:c1	c2h ³	a_1_1_2/m	-a_2	
12:c2	c2h ³	b_1_1_2/m	-b_2	
12:c3	c2h ³	i_1_1_2/m	-i_2	
12:a1	c2h ³	b_2/m_1_1	-b_2x	
12:a2	c2h ³	c_2/m_1_1	-c_2x	
12:a3	c2h ³	i_2/m_1_1	-i_2x	
13:b1	c2h ⁴	p_1_2/c_1	-p_2yc	
13:b1	c2h ⁴	p_2/c	-p_2yc	
13:b2	c2h ⁴	p_1_2/n_1	-p_2yac	
13:b2	c2h ⁴	p_2/n	-p_2yac	
13:b3	c2h ⁴	p_1_2/a_1	-p_2ya	
13:b3	c2h ⁴	p_2/a	-p_2ya	
13:c1	c2h ⁴	p_1_1_2/a	-p_2a	
13:c2	c2h ⁴	p_1_1_2/n	-p_2ab	
13:c3	c2h ⁴	p_1_1_2/b	-p_2b	
13:a1	c2h ⁴	p_2/b_1_1	-p_2xb	
13:a2	c2h ⁴	p_2/n_1_1	-p_2xbc	
13:a3	c2h ⁴	p_2/c_1_1	-p_2xc	
14:b1	c2h ⁵	p_1_21/c_1	-p_2ybc	
14:b1	c2h ⁵	p_21/c	-p_2ybc	
14:b2	c2h ⁵	p_1_21/n_1	-p_2yn	
14:b2	c2h ⁵	p_21/n	-p_2yn	
14:b3	c2h ⁵	p_1_21/a_1	-p_2yab	
14:b3	c2h ⁵	p_21/a	-p_2yab	
14:c1	c2h ⁵	p_1_1_21/a	-p_2ac	
14:c2	c2h ⁵	p_1_1_21/n	-p_2n	
14:c3	c2h ⁵	p_1_1_21/b	-p_2bc	
14:a1	c2h ⁵	p_21/b_1_1	-p_2xab	
14:a2	c2h ⁵	p_21/n_1_1	-p_2xn	
14:a3	c2h ⁵	p_21/c_1_1	-p_2xac	
15:b1	c2h ⁶	c_1_2/c_1	-c_2yc	

Appendix B. Symmetry

15:b1	c2h ⁶	c_2/c	-c_2yc	
15:b2	c2h ⁶	a_1_2/n_1	-a_2yab	# -a_2yac
15:b3	c2h ⁶	i_1_2/a_1	-i_2ya	
15:b3	c2h ⁶	i_2/a	-i_2ya	
15:-b1	c2h ⁶	a_1_2/a_1	-a_2ya	
15:-b2	c2h ⁶	c_1_2/n_1	-c_2yac	# -c_2ybc
15:-b2	c2h ⁶	c_2/n	-c_2yac	# -c_2ybc
15:-b3	c2h ⁶	i_1_2/c_1	-i_2yc	
15:-b3	c2h ⁶	i_2/c	-i_2yc	
15:c1	c2h ⁶	a_1_1_2/a	-a_2a	
15:c2	c2h ⁶	b_1_1_2/n	-b_2ab	# -b_2bc
15:c3	c2h ⁶	i_1_1_2/b	-i_2b	
15:-c1	c2h ⁶	b_1_1_2/b	-b_2b	
15:-c2	c2h ⁶	a_1_1_2/n	-a_2ab	# -a_2ac
15:-c3	c2h ⁶	i_1_1_2/a	-i_2a	
15:a1	c2h ⁶	b_2/b_1_1	-b_2xb	
15:a2	c2h ⁶	c_2/n_1_1	-c_2xac	# -c_2xbc
15:a3	c2h ⁶	i_2/c_1_1	-i_2xc	
15:-a1	c2h ⁶	c_2/c_1_1	-c_2xc	
15:-a2	c2h ⁶	b_2/n_1_1	-b_2xab	# -b_2xbc
15:-a3	c2h ⁶	i_2/b_1_1	-i_2xb	
16	d2 ¹	p_2_2_2	p_2_2	
17	d2 ²	p_2_2_21	p_2c_2	
17	d2 ²	p_2_2_21	p_21_2	
17:cab	d2 ²	p_21_2_2	p_2a_2a	
17:bca	d2 ²	p_2_21_2	p_2_2b	
18	d2 ³	p_21_21_2	p_2_2ab	
18:cab	d2 ³	p_2_21_21	p_2bc_2	
18:bca	d2 ³	p_21_2_21	p_2ac_2ac	
19	d2 ⁴	p_21_21_21	p_2ac_2ab	
20	d2 ⁵	c_2_2_21	c_2c_2	
20	d2 ⁵	c_2_2_21	c_21_2	
20:cab	d2 ⁵	a_21_2_2	a_2a_2a	
20:cab	d2 ⁵	a_21_2_2	a_2a_21	
20:bca	d2 ⁵	b_2_21_2	b_2_2b	
21	d2 ⁶	c_2_2_2	c_2_2	
21:cab	d2 ⁶	a_2_2_2	a_2_2	
21:bca	d2 ⁶	b_2_2_2	b_2_2	
22	d2 ⁷	f_2_2_2	f_2_2	
23	d2 ⁸	i_2_2_2	i_2_2	
24	d2 ⁹	i_21_21_21	i_2b_2c	
25	c2v ¹	p_m_m_2	p_2_-2	
25:cab	c2v ¹	p_2_m_m	p_-2_2	
25:bca	c2v ¹	p_m_2_m	p_-2_-2	
26	c2v ²	p_m_c_21	p_2c_-2	
26	c2v ²	p_m_c_21	p_21_-2	
26:ba-c	c2v ²	p_c_m_21	p_2c_-2c	
26:ba-c	c2v ²	p_c_m_21	p_21_-2c	
26:cab	c2v ²	p_21_m_a	p_-2a_2a	
26:-cba	c2v ²	p_21_a_m	p_-2_2a	
26:bca	c2v ²	p_b_21_m	p_-2_-2b	
26:a-cb	c2v ²	p_m_21_b	p_-2b_-2	
27	c2v ³	p_c_c_2	p_2_-2c	
27:cab	c2v ³	p_2_a_a	p_-2a_2	
27:bca	c2v ³	p_b_2_b	p_-2b_-2b	
28	c2v ⁴	p_m_a_2	p_2_-2a	
28	c2v ⁴	p_m_a_2	p_2_-21	

28:ba-c	$c2v^4$	$p_{b_m_2}$	p_{2_-2b}
28:cab	$c2v^4$	$p_{2_m_b}$	p_{-2b_2}
28:-cba	$c2v^4$	$p_{2_c_m}$	p_{-2c_2}
28:-cba	$c2v^4$	$p_{2_c_m}$	p_{-21_2}
28:bca	$c2v^4$	$p_{c_2_m}$	p_{-2c_-2c}
28:a-cb	$c2v^4$	$p_{m_2_a}$	p_{-2a_-2a}
29	$c2v^5$	$p_{c_a_21}$	p_{2c_-2ac}
29:ba-c	$c2v^5$	$p_{b_c_21}$	p_{2c_-2b}
29:cab	$c2v^5$	$p_{21_a_b}$	p_{-2b_2a}
29:-cba	$c2v^5$	$p_{21_c_a}$	p_{-2ac_2a}
29:bca	$c2v^5$	$p_{c_21_b}$	p_{-2bc_-2c}
29:a-cb	$c2v^5$	$p_{b_21_a}$	p_{-2a_-2ab}
30	$c2v^6$	$p_{n_c_2}$	p_{2_-2bc}
30:ba-c	$c2v^6$	$p_{c_n_2}$	p_{2_-2ac}
30:cab	$c2v^6$	$p_{2_n_a}$	p_{-2ac_2}
30:-cba	$c2v^6$	$p_{2_a_n}$	p_{-2ab_2}
30:bca	$c2v^6$	$p_{b_2_n}$	p_{-2ab_-2ab}
30:a-cb	$c2v^6$	$p_{n_2_b}$	p_{-2bc_-2bc}
31	$c2v^7$	$p_{m_n_21}$	p_{2ac_-2}
31:ba-c	$c2v^7$	$p_{n_m_21}$	p_{2bc_-2bc}
31:cab	$c2v^7$	$p_{21_m_n}$	p_{-2ab_2ab}
31:-cba	$c2v^7$	$p_{21_n_m}$	p_{-2_2ac}
31:bca	$c2v^7$	$p_{n_21_m}$	p_{-2_-2bc}
31:a-cb	$c2v^7$	$p_{m_21_n}$	p_{-2ab_-2}
32	$c2v^8$	$p_{b_a_2}$	p_{2_-2ab}
32:cab	$c2v^8$	$p_{2_c_b}$	p_{-2bc_2}
32:bca	$c2v^8$	$p_{c_2_a}$	p_{-2ac_-2ac}
33	$c2v^9$	$p_{n_a_21}$	p_{2c_-2n}
33	$c2v^9$	$p_{n_a_21}$	p_{21_-2n}
33:ba-c	$c2v^9$	$p_{b_n_21}$	p_{2c_-2ab}
33:ba-c	$c2v^9$	$p_{b_n_21}$	p_{21_-2ab}
33:cab	$c2v^9$	$p_{21_n_b}$	p_{-2bc_2a}
33:cab	$c2v^9$	$p_{21_n_b}$	p_{-2bc_21}
33:-cba	$c2v^9$	$p_{21_c_n}$	p_{-2n_2a}
33:-cba	$c2v^9$	$p_{21_c_n}$	p_{-2n_21}
33:bca	$c2v^9$	$p_{c_21_n}$	p_{-2n_-2ac}
33:a-cb	$c2v^9$	$p_{n_21_a}$	p_{-2ac_-2n}
34	$c2v^{10}$	$p_{n_n_2}$	p_{2_-2n}
34:cab	$c2v^{10}$	$p_{2_n_n}$	p_{-2n_2}
34:bca	$c2v^{10}$	$p_{n_2_n}$	p_{-2n_-2n}
35	$c2v^{11}$	$c_{m_m_2}$	c_{2_-2}
35:cab	$c2v^{11}$	$a_{2_m_m}$	a_{-2_2}
35:bca	$c2v^{11}$	$b_{m_2_m}$	b_{-2_-2}
36	$c2v^{12}$	$c_{m_c_21}$	c_{2c_-2}
36	$c2v^{12}$	$c_{m_c_21}$	c_{21_-2}
36:ba-c	$c2v^{12}$	$c_{c_m_21}$	c_{2c_-2c}
36:ba-c	$c2v^{12}$	$c_{c_m_21}$	c_{21_-2c}
36:cab	$c2v^{12}$	$a_{21_m_a}$	a_{-2a_2a}
36:cab	$c2v^{12}$	$a_{21_m_a}$	a_{-2a_21}
36:-cba	$c2v^{12}$	$a_{21_a_m}$	a_{-2_2a}
36:-cba	$c2v^{12}$	$a_{21_a_m}$	a_{-2_21}
36:bca	$c2v^{12}$	$b_{b_21_m}$	b_{-2_-2b}
36:a-cb	$c2v^{12}$	$b_{m_21_b}$	b_{-2b_-2}
37	$c2v^{13}$	$c_{c_c_2}$	c_{2_-2c}
37:cab	$c2v^{13}$	$a_{2_a_a}$	a_{-2a_2}
37:bca	$c2v^{13}$	$b_{b_2_b}$	b_{-2b_-2b}

38	$c2v^{14}$	$a_m m_2$	a_{2-2}	
38:ba-c	$c2v^{14}$	$b_m m_2$	b_{2-2}	
38:cab	$c2v^{14}$	$b_2 m m$	b_{-2_2}	
38:-cba	$c2v^{14}$	$c_2 m m$	c_{-2_2}	
38:bca	$c2v^{14}$	$c_m 2 m$	c_{-2_-2}	
38:a-cb	$c2v^{14}$	$a_m 2 m$	a_{-2_-2}	
39	$c2v^{15}$	$a_b m_2$	a_{2-2c}	
39:ba-c	$c2v^{15}$	$b_m a_2$	b_{2-2a}	# b_{2-2c}
39:cab	$c2v^{15}$	$b_2 c_m$	b_{-2a_2}	# b_{-2c_2}
39:-cba	$c2v^{15}$	$c_2 m_b$	c_{-2a_2}	# c_{-2b_2}
39:bca	$c2v^{15}$	$c_m 2_a$	c_{-2a_-2a}	# c_{-2b_-2b}
39:a-cb	$c2v^{15}$	$a_c 2_m$	a_{-2c_-2c}	
40	$c2v^{16}$	$a_m a_2$	a_{2-2a}	
40:ba-c	$c2v^{16}$	$b_b m_2$	b_{2-2b}	
40:cab	$c2v^{16}$	$b_2 m_b$	b_{-2b_2}	
40:-cba	$c2v^{16}$	$c_2 c_m$	c_{-2c_2}	
40:bca	$c2v^{16}$	$c_c 2_m$	c_{-2c_-2c}	
40:a-cb	$c2v^{16}$	$a_m 2_a$	a_{-2a_-2a}	
41	$c2v^{17}$	$a_b a_2$	a_{2-2ab}	# a_{2-2ac}
41:ba-c	$c2v^{17}$	$b_b a_2$	b_{2-2ab}	# b_{2-2bc}
41:cab	$c2v^{17}$	$b_2 c_b$	b_{-2ab_2}	# b_{-2bc_2}
41:-cba	$c2v^{17}$	$c_2 c_b$	c_{-2ac_2}	# c_{-2bc_2}
41:bca	$c2v^{17}$	$c_c 2_a$	c_{-2ac_-2ac}	# c_{-2bc_-2bc}
41:a-cb	$c2v^{17}$	$a_c 2_a$	a_{-2ab_-2ab}	# a_{-2ac_-2ac}
42	$c2v^{18}$	$f_m m_2$	f_{2-2}	
42:cab	$c2v^{18}$	$f_2 m m$	f_{-2_2}	
42:bca	$c2v^{18}$	$f_m 2_m$	f_{-2_-2}	
43	$c2v^{19}$	$f_d d_2$	f_{2-2d}	
43:cab	$c2v^{19}$	$f_2 d_d$	f_{-2d_2}	
43:bca	$c2v^{19}$	$f_d 2_d$	f_{-2d_-2d}	
44	$c2v^{20}$	$i_m m_2$	i_{2-2}	
44:cab	$c2v^{20}$	$i_2 m m$	i_{-2_2}	
44:bca	$c2v^{20}$	$i_m 2_m$	i_{-2_-2}	
45	$c2v^{21}$	$i_b a_2$	i_{2-2c}	
45:cab	$c2v^{21}$	$i_2 c_b$	i_{-2a_2}	
45:bca	$c2v^{21}$	$i_c 2_a$	i_{-2b_-2b}	
46	$c2v^{22}$	$i_m a_2$	i_{2-2a}	
46:ba-c	$c2v^{22}$	$i_b m_2$	i_{2-2b}	
46:cab	$c2v^{22}$	$i_2 m_b$	i_{-2b_2}	
46:-cba	$c2v^{22}$	$i_2 c_m$	i_{-2c_2}	
46:bca	$c2v^{22}$	$i_c 2_m$	i_{-2c_-2c}	
46:a-cb	$c2v^{22}$	$i_m 2_a$	i_{-2a_-2a}	
47	$d2h^1$	$p_m m m$	$-p_{2_2}$	
48:1	$d2h^2$	$p_n n n:1$	p_{2_2-1n}	
48:2	$d2h^2$	$p_n n n:2$	$-p_{2ab_2bc}$	
49	$d2h^3$	$p_c c_m$	$-p_{2_2c}$	
49:cab	$d2h^3$	$p_m a_a$	$-p_{2a_2}$	
49:bca	$d2h^3$	$p_b m_b$	$-p_{2b_2b}$	
50:1	$d2h^4$	$p_b a_n:1$	p_{2_2-1ab}	
50:2	$d2h^4$	$p_b a_n:2$	$-p_{2ab_2b}$	
50:1cab	$d2h^4$	$p_n c_b:1$	p_{2_2-1bc}	
50:2cab	$d2h^4$	$p_n c_b:2$	$-p_{2b_2bc}$	
50:1bca	$d2h^4$	$p_c n a:1$	p_{2_2-1ac}	
50:2bca	$d2h^4$	$p_c n a:2$	$-p_{2a_2c}$	
51	$d2h^5$	$p_m m_a$	$-p_{2a_2a}$	
51:ba-c	$d2h^5$	$p_m m_b$	$-p_{2b_2}$	
51:cab	$d2h^5$	$p_b m m$	$-p_{2_2b}$	
51:-cba	$d2h^5$	$p_c m m$	$-p_{2c_2c}$	

51:bca	d2h ⁵	p_m_c_m	-p_2c_2
51:a-cb	d2h ⁵	p_m_a_m	-p_2_2a
52	d2h ⁶	p_n_n_a	-p_2a_2bc
52:ba-c	d2h ⁶	p_n_n_b	-p_2b_2n
52:cab	d2h ⁶	p_b_n_n	-p_2n_2b
52:-cba	d2h ⁶	p_c_n_n	-p_2ab_2c
52:bca	d2h ⁶	p_n_c_n	-p_2ab_2n
52:a-cb	d2h ⁶	p_n_a_n	-p_2n_2bc
53	d2h ⁷	p_m_n_a	-p_2ac_2
53:ba-c	d2h ⁷	p_n_m_b	-p_2bc_2bc
53:cab	d2h ⁷	p_b_m_n	-p_2ab_2ab
53:-cba	d2h ⁷	p_c_n_m	-p_2_2ac
53:bca	d2h ⁷	p_n_c_m	-p_2_2bc
53:a-cb	d2h ⁷	p_m_a_n	-p_2ab_2
54	d2h ⁸	p_c_c_a	-p_2a_2ac
54:ba-c	d2h ⁸	p_c_c_b	-p_2b_2c
54:cab	d2h ⁸	p_b_a_a	-p_2a_2b
54:-cba	d2h ⁸	p_c_a_a	-p_2ac_2c
54:bca	d2h ⁸	p_b_c_b	-p_2bc_2b
54:a-cb	d2h ⁸	p_b_a_b	-p_2b_2ab
55	d2h ⁹	p_b_a_m	-p_2_2ab
55:cab	d2h ⁹	p_m_c_b	-p_2bc_2
55:bca	d2h ⁹	p_c_m_a	-p_2ac_2ac
56	d2h ¹⁰	p_c_c_n	-p_2ab_2ac
56:cab	d2h ¹⁰	p_n_a_a	-p_2ac_2bc
56:bca	d2h ¹⁰	p_b_n_b	-p_2bc_2ab
57	d2h ¹¹	p_b_c_m	-p_2c_2b
57:ba-c	d2h ¹¹	p_c_a_m	-p_2c_2ac
57:cab	d2h ¹¹	p_m_c_a	-p_2ac_2a
57:-cba	d2h ¹¹	p_m_a_b	-p_2b_2a
57:bca	d2h ¹¹	p_b_m_a	-p_2a_2ab
57:a-cb	d2h ¹¹	p_c_m_b	-p_2bc_2c
58	d2h ¹²	p_n_n_m	-p_2_2n
58:cab	d2h ¹²	p_m_n_n	-p_2n_2
58:bca	d2h ¹²	p_n_m_n	-p_2n_2n
59:1	d2h ¹³	p_m_m_n:1	p_2_2ab_-1ab
59:2	d2h ¹³	p_m_m_n:2	-p_2ab_2a
59:1cab	d2h ¹³	p_n_m_m:1	p_2bc_2_-1bc
59:2cab	d2h ¹³	p_n_m_m:2	-p_2c_2bc
59:1bca	d2h ¹³	p_m_n_m:1	p_2ac_2ac_-1ac
59:2bca	d2h ¹³	p_m_n_m:2	-p_2c_2a
60	d2h ¹⁴	p_b_c_n	-p_2n_2ab
60:ba-c	d2h ¹⁴	p_c_a_n	-p_2n_2c
60:cab	d2h ¹⁴	p_n_c_a	-p_2a_2n
60:-cba	d2h ¹⁴	p_n_a_b	-p_2bc_2n
60:bca	d2h ¹⁴	p_b_n_a	-p_2ac_2b
60:a-cb	d2h ¹⁴	p_c_n_b	-p_2b_2ac
61	d2h ¹⁵	p_b_c_a	-p_2ac_2ab
61:ba-c	d2h ¹⁵	p_c_a_b	-p_2bc_2ac
62	d2h ¹⁶	p_n_m_a	-p_2ac_2n
62:ba-c	d2h ¹⁶	p_m_n_b	-p_2bc_2a
62:cab	d2h ¹⁶	p_b_n_m	-p_2c_2ab
62:-cba	d2h ¹⁶	p_c_m_n	-p_2n_2ac
62:bca	d2h ¹⁶	p_m_c_n	-p_2n_2a
62:a-cb	d2h ¹⁶	p_n_a_m	-p_2c_2n

63	d2h ¹⁷	c_m_c_m	-c_2c_2	
63:ba-c	d2h ¹⁷	c_c_m_m	-c_2c_2c	
63:cab	d2h ¹⁷	a_m_m_a	-a_2a_2a	
63:-cba	d2h ¹⁷	a_m_a_m	-a_2_2a	
63:bca	d2h ¹⁷	b_b_m_m	-b_2_2b	
63:a-cb	d2h ¹⁷	b_m_m_b	-b_2b_2	
64	d2h ¹⁸	c_m_c_a	-c_2ac_2	# -c_2bc_2
64:ba-c	d2h ¹⁸	c_c_m_b	-c_2ac_2ac	# -c_2bc_2bc
64:cab	d2h ¹⁸	a_b_m_a	-a_2ab_2ab	# -a_2ac_2ac
64:-cba	d2h ¹⁸	a_c_a_m	-a_2_2ab	# -a_2_2ac
64:bca	d2h ¹⁸	b_b_c_m	-b_2_2ab	# -b_2_2bc
64:a-cb	d2h ¹⁸	b_m_a_b	-b_2ab_2	# -b_2bc_2
65	d2h ¹⁹	c_m_m_m	-c_2_2	
65:cab	d2h ¹⁹	a_m_m_m	-a_2_2	
65:bca	d2h ¹⁹	b_m_m_m	-b_2_2	
66	d2h ²⁰	c_c_c_m	-c_2_2c	
66:cab	d2h ²⁰	a_m_a_a	-a_2a_2	
66:bca	d2h ²⁰	b_b_m_b	-b_2b_2b	
67	d2h ²¹	c_m_m_a	-c_2a_2	# -c_2b_2
67:ba-c	d2h ²¹	c_m_m_b	-c_2a_2a	# -c_2b_2b
67:cab	d2h ²¹	a_b_m_m	-a_2b_2b	# -a_2c_2c
67:-cba	d2h ²¹	a_c_m_m	-a_2_2c	
67:bca	d2h ²¹	b_m_c_m	-b_2_2a	# -b_2_2c
67:a-cb	d2h ²¹	b_m_a_m	-b_2a_2	# -b_2c_2
68:1	d2h ²²	c_c_c_a:1	c_2_2_-1ac	# c_2_2_-1bc
68:2	d2h ²²	c_c_c_a:2	-c_2a_2ac	# -c_2b_2bc
68:1ba-c	d2h ²²	c_c_c_b:1	c_2_2_-1ac	# c_2_2_-1bc
68:2ba-c	d2h ²²	c_c_c_b:2	-c_2a_2c	# -c_2b_2c
68:1cab	d2h ²²	a_b_a_a:1	a_2_2_-1ab	# a_2_2_-1ac
68:2cab	d2h ²²	a_b_a_a:2	-a_2a_2c	
68:1-cba	d2h ²²	a_c_a_a:1	a_2_2_-1ab	# a_2_2_-1ac
68:2-cba	d2h ²²	a_c_a_a:2	-a_2ab_2b	# -a_2ac_2c
68:1bca	d2h ²²	b_b_c_b:1	b_2_2_-1ab	# b_2_2_-1bc
68:2bca	d2h ²²	b_b_c_b:2	-b_2ab_2b	# -b_2bc_2b
68:1a-cb	d2h ²²	b_b_a_b:1	b_2_2_-1ab	# b_2_2_-1bc
68:2a-cb	d2h ²²	b_b_a_b:2	-b_2b_2ab	# -b_2b_2bc
69	d2h ²³	f_m_m_m	-f_2_2	
70:1	d2h ²⁴	f_d_d_d:1	f_2_2_-1d	
70:2	d2h ²⁴	f_d_d_d:2	-f_2uv_2vw	
71	d2h ²⁵	i_m_m_m	-i_2_2	
72	d2h ²⁶	i_b_a_m	-i_2_2c	
72:cab	d2h ²⁶	i_m_c_b	-i_2a_2	
72:bca	d2h ²⁶	i_c_m_a	-i_2b_2b	
73	d2h ²⁷	i_b_c_a	-i_2b_2c	
73:ba-c	d2h ²⁷	i_c_a_b	-i_2a_2b	
74	d2h ²⁸	i_m_m_a	-i_2b_2	
74:ba-c	d2h ²⁸	i_m_m_b	-i_2a_2a	
74:cab	d2h ²⁸	i_b_m_m	-i_2c_2c	
74:-cba	d2h ²⁸	i_c_m_m	-i_2_2b	
74:bca	d2h ²⁸	i_m_c_m	-i_2_2a	
74:a-cb	d2h ²⁸	i_m_a_m	-i_2c_2	
75	c4 ¹	p_4	p_4	
76	c4 ²	p_41	p_4w	
76	c4 ²	p_41	p_41	
77	c4 ³	p_42	p_4c	
77	c4 ³	p_42	p_42	
78	c4 ⁴	p_43	p_4cw	
78	c4 ⁴	p_43	p_43	
79	c4 ⁵	i_4	i_4	

80	c4^6	i_41	i_4bw
81	s4^1	p_-4	p_-4
82	s4^2	i_-4	i_-4
83	c4h^1	p_4/m	-p_4
84	c4h^2	p_42/m	-p_4c
84	c4h^2	p_42/m	-p_42
85:1	c4h^3	p_4/n:1	p_4ab_-1ab
85:2	c4h^3	p_4/n:2	-p_4a
86:1	c4h^4	p_42/n:1	p_4n_-1n
86:2	c4h^4	p_42/n:2	-p_4bc
87	c4h^5	i_4/m	-i_4
88:1	c4h^6	i_41/a:1	i_4bw_-1bw
88:2	c4h^6	i_41/a:2	-i_4ad
89	d4^1	p_4_2_2	p_4_2
90	d4^2	p_4_21_2	p_4ab_2ab
91	d4^3	p_41_2_2	p_4w_2c
91	d4^3	p_41_2_2	p_41_2c
92	d4^4	p_41_21_2	p_4abw_2nw
93	d4^5	p_42_2_2	p_4c_2
93	d4^5	p_42_2_2	p_42_2
94	d4^6	p_42_21_2	p_4n_2n
95	d4^7	p_43_2_2	p_4cw_2c
95	d4^7	p_43_2_2	p_43_2c
96	d4^8	p_43_21_2	p_4nw_2abw
97	d4^9	i_4_2_2	i_4_2
98	d4^10	i_41_2_2	i_4bw_2bw
99	c4v^1	p_4_m_m	p_4_-2
100	c4v^2	p_4_b_m	p_4_-2ab
101	c4v^3	p_42_c_m	p_4c_-2c
101	c4v^3	p_42_c_m	p_42_-2c
102	c4v^4	p_42_n_m	p_4n_-2n
103	c4v^5	p_4_c_c	p_4_-2c
104	c4v^6	p_4_n_c	p_4_-2n
105	c4v^7	p_42_m_c	p_4c_-2
105	c4v^7	p_42_m_c	p_42_-2
106	c4v^8	p_42_b_c	p_4c_-2ab
106	c4v^8	p_42_b_c	p_42_-2ab
107	c4v^9	i_4_m_m	i_4_-2
108	c4v^10	i_4_c_m	i_4_-2c
109	c4v^11	i_41_m_d	i_4bw_-2
110	c4v^12	i_41_c_d	i_4bw_-2c
111	d2d^1	p_-4_2_m	p_-4_2
112	d2d^2	p_-4_2_c	p_-4_2c
113	d2d^3	p_-4_21_m	p_-4_2ab
114	d2d^4	p_-4_21_c	p_-4_2n
115	d2d^5	p_-4_m_2	p_-4_-2
116	d2d^6	p_-4_c_2	p_-4_-2c
117	d2d^7	p_-4_b_2	p_-4_-2ab
118	d2d^8	p_-4_n_2	p_-4_-2n
119	d2d^9	i_-4_m_2	i_-4_-2
120	d2d^10	i_-4_c_2	i_-4_-2c
121	d2d^11	i_-4_2_m	i_-4_2
122	d2d^12	i_-4_2_d	i_-4_2bw
123	d4h^1	p_4/m_m_m	-p_4_2
124	d4h^2	p_4/m_c_c	-p_4_2c
125:1	d4h^3	p_4/n_b_m:1	p_4_2_-1ab

125:2	d4h ³	p_4/n_b_m:2	-p_4a_2b	
126:1	d4h ⁴	p_4/n_n_c:1	p_4_2_-1n	
126:2	d4h ⁴	p_4/n_n_c:2	-p_4a_2bc	
127	d4h ⁵	p_4/m_b_m	-p_4_2ab	
128	d4h ⁶	p_4/m_n_c	-p_4_2n	
129:1	d4h ⁷	p_4/n_m_m:1	p_4ab_2ab_-1ab	
129:2	d4h ⁷	p_4/n_m_m:2	-p_4a_2a	
130:1	d4h ⁸	p_4/n_c_c:1	p_4ab_2n_-1ab	
130:2	d4h ⁸	p_4/n_c_c:2	-p_4a_2ac	
131	d4h ⁹	p_42/m_m_c	-p_4c_2	
132	d4h ¹⁰	p_42/m_c_m	-p_4c_2c	
133:1	d4h ¹¹	p_42/n_b_c:1	p_4n_2c_-1n	
133:2	d4h ¹¹	p_42/n_b_c:2	-p_4ac_2b	
134:1	d4h ¹²	p_42/n_n_m:1	p_4n_2_-1n	
134:2	d4h ¹²	p_42/n_n_m:2	-p_4ac_2bc	
135	d4h ¹³	p_42/m_b_c	-p_4c_2ab	
135	d4h ¹³	p_42/m_b_c	-p_42_2ab	
136	d4h ¹⁴	p_42/m_n_m	-p_4n_2n	
137:1	d4h ¹⁵	p_42/n_m_c:1	p_4n_2n_-1n	
137:2	d4h ¹⁵	p_42/n_m_c:2	-p_4ac_2a	
138:1	d4h ¹⁶	p_42/n_c_m:1	p_4n_2ab_-1n	
138:2	d4h ¹⁶	p_42/n_c_m:2	-p_4ac_2ac	
139	d4h ¹⁷	i_4/m_m_m	-i_4_2	
140	d4h ¹⁸	i_4/m_c_m	-i_4_2c	
141:1	d4h ¹⁹	i_41/a_m_d:1	i_4bw_2bw_-1bw	
141:2	d4h ¹⁹	i_41/a_m_d:2	-i_4bd_2	
142:1	d4h ²⁰	i_41/a_c_d:1	i_4bw_2aw_-1bw	
142:2	d4h ²⁰	i_41/a_c_d:2	-i_4bd_2c	
143	c3 ¹	p_3	p_3	
144	c3 ²	p_31	p_31	
145	c3 ³	p_32	p_32	
146:h	c3 ⁴	r_3:h	r_3	
146:r	c3 ⁴	r_3:r	p_3*	
147	c3i ¹	p_-3	-p_3	
148:h	c3i ²	r_-3:h	-r_3	
148:r	c3i ²	r_-3:r	-p_3*	
149	d3 ¹	p_3_1_2	p_3_2	
150	d3 ²	p_3_2_1	p_3_2"	
151	d3 ³	p_31_1_2	p_31_2_(0_0_4)	# p_31_2c_(0_0_1)
152	d3 ⁴	p_31_2_1	p_31_2"	
153	d3 ⁵	p_32_1_2	p_32_2_(0_0_2)	# p_32_2c_(0_0_-1)
154	d3 ⁶	p_32_2_1	p_32_2"	
155:h	d3 ⁷	r_3_2:h	r_3_2"	
155:r	d3 ⁷	r_3_2:r	p_3*_2	
156	c3v ¹	p_3_m_1	p_3_-2"	
157	c3v ²	p_3_1_m	p_3_-2	
158	c3v ³	p_3_c_1	p_3_-2"c	
159	c3v ⁴	p_3_1_c	p_3_-2c	
160:h	c3v ⁵	r_3_m:h	r_3_-2"	
160:r	c3v ⁵	r_3_m:r	p_3*__-2	
161:h	c3v ⁶	r_3_c:h	r_3_-2"c	
161:r	c3v ⁶	r_3_c:r	p_3*__-2n	
162	d3d ¹	p_-3_1_m	-p_3_2	
163	d3d ²	p_-3_1_c	-p_3_2c	
164	d3d ³	p_-3_m_1	-p_3_2"	
165	d3d ⁴	p_-3_c_1	-p_3_2"c	

166:h	d3d ⁵	r_-3_m:h	-r_3_2"	
166:r	d3d ⁵	r_-3_m:r	-p_3*_2	
167:h	d3d ⁶	r_-3_c:h	-r_3_2"c	
167:r	d3d ⁶	r_-3_c:r	-p_3*_2n	
168	c6 ¹	p_6	p_6	
169	c6 ²	p_61	p_61	
170	c6 ³	p_65	p_65	
171	c6 ⁴	p_62	p_62	
172	c6 ⁵	p_64	p_64	
173	c6 ⁶	p_63	p_6c	
173	c6 ⁶	p_63	p_63	
174	c3h ¹	p_-6	p_-6	
175	c6h ¹	p_6/m	-p_6	
176	c6h ²	p_63/m	-p_6c	
176	c6h ²	p_63/m	-p_63	
177	d6 ¹	p_6_2_2	p_6_2	
178	d6 ²	p_61_2_2	p_61_2_(0_0_5)	# p_61_2_(0_0_-1)
179	d6 ³	p_65_2_2	p_65_2_(0_0_1)	
180	d6 ⁴	p_62_2_2	p_62_2_(0_0_4)	# p_62_2c_(0_0_1)
181	d6 ⁵	p_64_2_2	p_64_2_(0_0_2)	# p_64_2c_(0_0_-1)
182	d6 ⁶	p_63_2_2	p_6c_2c	
182	d6 ⁶	p_63_2_2	p_63_2c	
183	c6v ¹	p_6_m_m	p_6_-2	
184	c6v ²	p_6_c_c	p_6_-2c	
185	c6v ³	p_63_c_m	p_6c_-2	
185	c6v ³	p_63_c_m	p_63_-2	
186	c6v ⁴	p_63_m_c	p_6c_-2c	
186	c6v ⁴	p_63_m_c	p_63_-2c	
187	d3h ¹	p_-6_m_2	p_-6_2	
188	d3h ²	p_-6_c_2	p_-6c_2	
189	d3h ³	p_-6_2_m	p_-6_-2	
190	d3h ⁴	p_-6_2_c	p_-6c_-2c	
191	d6h ¹	p_6/m_m_m	-p_6_2	
192	d6h ²	p_6/m_c_c	-p_6_2c	
193	d6h ³	p_63/m_c_m	-p_6c_2	
193	d6h ³	p_63/m_c_m	-p_63_2	
194	d6h ⁴	p_63/m_m_c	-p_6c_2c	
194	d6h ⁴	p_63/m_m_c	-p_63_2c	
195	t ¹	p_2_3	p_2_2_3	
196	t ²	f_2_3	f_2_2_3	
197	t ³	i_2_3	i_2_2_3	
198	t ⁴	p_21_3	p_2ac_2ab_3	
199	t ⁵	i_21_3	i_2b_2c_3	
200	th ¹	p_m_-3	-p_2_2_3	
201:1	th ²	p_n_-3:1	p_2_2_3_-1n	
201:2	th ²	p_n_-3:2	-p_2ab_2bc_3	
202	th ³	f_m_-3	-f_2_2_3	
203:1	th ⁴	f_d_-3:1	f_2_2_3_-1d	
203:2	th ⁴	f_d_-3:2	-f_2uv_2vw_3	
204	th ⁵	i_m_-3	-i_2_2_3	
205	th ⁶	p_a_-3	-p_2ac_2ab_3	
206	th ⁷	i_a_-3	-i_2b_2c_3	
207	o ¹	p_4_3_2	p_4_2_3	
208	o ²	p_42_3_2	p_4n_2_3	
209	o ³	f_4_3_2	f_4_2_3	
210	o ⁴	f_41_3_2	f_4d_2_3	

Appendix B. Symmetry

211	o^5	$i_{4_3_2}$	$i_{4_2_3}$	
212	o^6	$p_{43_3_2}$	$p_{4acd_2ab_3}$	
213	o^7	$p_{41_3_2}$	$p_{4bd_2ab_3}$	
214	o^8	$i_{41_3_2}$	$i_{4bd_2c_3}$	
215	td^1	$p_{-4_3_m}$	$p_{-4_2_3}$	
216	td^2	$f_{-4_3_m}$	$f_{-4_2_3}$	
217	td^3	$i_{-4_3_m}$	$i_{-4_2_3}$	
218	td^4	$p_{-4_3_n}$	$p_{-4n_2_3}$	
219	td^5	$f_{-4_3_c}$	$f_{-4a_2_3}$	$\# f_{-4c_2_3}$
220	td^6	$i_{-4_3_d}$	$i_{-4bd_2c_3}$	
221	oh^1	$p_{m_-3_m}$	$-p_{4_2_3}$	
222:1	oh^2	$p_{n_-3_n:1}$	$p_{4_2_3_-1n}$	
222:2	oh^2	$p_{n_-3_n:2}$	$-p_{4a_2bc_3}$	
223	oh^3	$p_{m_-3_n}$	$-p_{4n_2_3}$	
224:1	oh^4	$p_{n_-3_m:1}$	$p_{4n_2_3_-1n}$	
224:2	oh^4	$p_{n_-3_m:2}$	$-p_{4bc_2bc_3}$	
225	oh^5	$f_{m_-3_m}$	$-f_{4_2_3}$	
226	oh^6	$f_{m_-3_c}$	$-f_{4a_2_3}$	$\# -f_{4c_2_3}$
227:1	oh^7	$f_{d_-3_m:1}$	$f_{4d_2_3_-1d}$	
227:2	oh^7	$f_{d_-3_m:2}$	$-f_{4vw_2vw_3}$	
228:1	oh^8	$f_{d_-3_c:1}$	$f_{4d_2_3_-1ad}$	$\# f_{4d_2_3_-1cd}$
228:2	oh^8	$f_{d_-3_c:2}$	$-f_{4ud_2vw_3}$	$\# -f_{4cvw_2vw_3}$
229	oh^9	$i_{m_-3_m}$	$-i_{4_2_3}$	
230	oh^{10}	$i_{a_-3_d}$	$-i_{4bd_2c_3}$	