SHELXL-97 Quick Reference Manual

By George Sheldrick
Command Card Reference Guide for electron book readers

ACTA
AFIX
ANIS
ANIS names
BASF
BIND
BLOC
BOND
BUMP
CELL
CGLS
CHIV
CONF
CONN
DAMP
DANG
DEFS
DELU
DFIX
DISP
EADP
END
EQIV
EXTI
EXYZ
FEND

FLAT
FMAP
FRAG
FREE
FVAR
GRID
HFIX
HKLF
HOPE
HTAB
HTAB d a
ISOR
L.S.
LATT
LAUE
LIST
MERG
MOLE
MORE
MOVE
MPLA
NCSY
OMIT
OMIT atom names
OMIT $h\,k\,l$

PART
PLAN
REM
RESI
RTAB
SADI
SAME
SFAC elements
SFAC label
SHEL
SIMU
SIZE
SPEC
STIR
SUMP
SWAT
SYMM
TEMP
TIME
TITL
TWIN
UNIT
WGHT
WPDB
ZERR

ACTA 2thetafull[#]

A 'Crystallographic Information File' file *name.cif* is created in self-defining STAR format. This ASCII file is suitable for data archiving, network transmission, and (with suitable additions) for direct submission for publication. ACTA automatically sets the BOND, FMAP 2, PLAN and LIST 4 instructions, and may not be used with other FMAP or LIST instructions or with a positive OMIT s threshold. A warning message appears if the cell contents on the UNIT instruction are not consistent with the atom list, because they are used to calculate the density etc. which appears in the *.cif* output file.

2thetafull is used to specify the value of 2θ for which the program calculates the completeness of the data for the CIF output file as required by Acta Crystallographica. If no value is given, the program uses the maximum value of 2θ for the reflection data. If the data were collected to a specific limiting 2θ, or if a limit was imposed using SHEL, this would be a good choice. Otherwise the choice of 2thetafull is a difficult compromise; if it is too low, the paper will be rejected because the resolution of the data is not good enough; if it is higher, the lower completeness might lead to rejection by the automatic Acta rejection software! SHELXL calculates the completeness by counting reflections after merging Friedel opposites and eliminating systematic absences (and the reflection 0,0,0).

AFIX mn d[#] sof[11] U[10.08]

AFIX applies constraints and/or generates idealized coordinates for all atoms until the next AFIX instruction is read. The digits **mn** of the AFIX code control two logically quite separate operations. Although this is confusing for new users, it has been retained for upwards compatibility with SHELX-76, and because it provides a very concise notation. The *m* refers to geometrical operations which are performed before the first refinement cycle (Hydrogen atoms are idealized before every cycle), and *n* sets up constraints which are applied **throughout the least-squares refinement.** The **n** is always a single digit; **m** may be two, one or zero digits (the last corresponds to m = 0).

The options for idealizing Hydrogen atom positions depend on the connectivity table that is set up using CONN, BIND, FREE and PART; with experience, this can also be used to generate Hydrogen atoms attached to disordered groups and to atoms on special positions. The *d* determines the bond lengths in the idealized groups, and *sof* and *U* OVERRIDE the values in the atom list for all atoms until the next AFIX instruction. The **U** is not applied if the atom is already anisotropic, but is used if an isotropic atom is to be made anisotropic using ANIS. Any legal U value may be used, e.g. 31 (a free variable reference) or -1.2 (1.2 times Ueq of the preceding normal atom). Each AFIX instruction must be followed by the required number of Hydrogen or other atoms. The individual AFIX options are as follows; the default X-H distances depend on both the chemical environment and the temperature (to allow for librational effects) which is specified by means of the TEMP instruction.

m : Geometric Operations

m = 0 No action.

m = 1 Idealized tertiary C-H with all X-C-H angles equal. There must be three and only three other bonds in the connectivity table to the immediately preceding atom, which is assumed to be carbon. m = 1 is often combined with a riding model refinement (n = 3).

m = 2 Idealized secondary CH_2 with all X-C-H and Y-C-H angles equal, and H-C-H determined by X-C-Y (i.e. approximately tetrahedral, but widened if X-C-Y is much less than tetrahedral). This option is also suitable for riding refinement (n = 3).

m = 3 Idealized CH_3 group with tetrahedral angles. The group is staggered with respect to the shortest other bond to the atom to which the -CH_3 is attached. If there is no such bond (e.g. an acetonitrile solvent molecule) this method cannot be used (but m = 13 is still viable).

m = 4 Aromatic C-H or amide N-H with the Hydrogen atom on the external bisector of the X-C-Y or X-N-Y angle. m = 4 is suitable for a riding model refinement, i.e. AFIX 43 before the H atom.

m = 5 Next five non-Hydrogen atoms are fitted to a regular pentagon, default d = 1.42 Å.

m = 6 Next six non-Hydrogen atoms are fitted to a regular hexagon, default d = 1.39 Å.

m = 7 Identical to m = 6 (included for upwards compatibility from SHELX-76). In SHELX-76 only the first, third and fifth atoms of the six-membered ring were used as target atoms; in SHELXL this will still be the case if the other three are given zero coordinates, but the procedure is more general because any one, two or three atoms may be left out by giving them zero coordinates.

m = 8 Idealized OH group, with X-O-H angle tetrahedral. If the oxygen is attached to a saturated carbon, all three staggered positions are considered for the Hydrogen. If it is attached to an aromatic ring, both positions in the plane are considered. The final choice is based on forming the 'best' Hydrogen bond to a nitrogen, oxygen, chlorine or fluorine atom. The algorithm involves generating a potential position for such an atom by extrapolating the O-H vector, then finding the nearest N, O, F or Cl atom to this position, taking symmetry equivalents into account. If another atom that (according to the connectivity table) is bonded to the N, O, F or Cl atom, is nearer to the ideal position, the N, O, F or Cl atom is not considered. Note that m = 8 had a different effect in SHELX-76 (but was rarely employed).

m = 9 Idealized terminal $X=CH_2$ or $X=NH_2^+$ with the Hydrogen atoms in the plane of the nearest substituent on the atom X. Suitable for riding model refinement (AFIX 93 before the two H atoms).

m = 10 Idealized pentamethylcyclopentadienyl (Cp*). This AFIX must be followed by the 5 ring carbons and then the 5 methyl carbons in cyclic order, so that the first methyl group (atom 6) is attached to the first carbon (atom 1). The default d is 1.42 Å, with the C-CH$_3$ distance set to 1.063d. A variable-metric rigid group refinement (AFIX 109) would be appropriate, and would allow for librational shortening of the bonds. Hydrogen atoms (e.g. with AFIX 37 or 127) may be included after the corresponding carbon atoms, in which case AFIX 0 or 5 (in the case of a rigid group refinement) must be inserted before the next carbon atom.

m = 11 Idealized naphthalene group with equal bonds (default d = 1.39 A). The atoms should be numbered as a symmetrical figure of eight, starting with the alpha C and followed by the beta, so that the first six atoms (and also the last six) describe a hexagon in cyclic order. m = 11 is also appropriate for rigid group refinement (AFIX 116).

m = 12 Idealized disordered methyl group; as m = 3 but with two positions rotated from each other by 60 degrees. The corresponding occupation factors

should normally be set to add up to one, e.g. by giving them as 21 (i.e. 1*fv(2)) and -21 (1*(1-fv(2))). If HFIX is used to generate an AFIX instruction with m=12, the occupation factors are fixed at 0.5. AFIX 12n is suitable for a *para* methyl on a phenyl group with no *meta* substituents, and should be followed by 6 half Hydrogen atoms (first the three belonging to one - CH_3component, then the three belonging to the other, so that Hydrogens n and n+3 are opposite one another). The six Hydrogens should have the same PART number as the carbon to which they are attached (e.g. PART 0).

m = 13 Idealized CH_3 group with tetrahedral angles. If the coordinates of the first Hydrogen atom are non-zero, they define the torsion angle of the methyl group. Otherwise (or if the AFIX instruction is being generated via HFIX) a structure-factor calculation is performed (of course only once, even if many Hydrogens are involved) and the torsion angle is set that maximizes the sum of the electron density at the three calculated Hydrogen positions. Since even this is not an infallible method of getting the correct torsion angle, it should normally be combined with a rigid or rotating group refinement for the methyl group (e.g. mn = 137 before the first H). In subsequent least-squares cycles the group is re-idealized retaining the current torsion angle

m = 14 Idealized OH group, with X-O-H angle tetrahedral. If the coordinates of the Hydrogen atom are non-zero, they are used to define the torsion angle. Otherwise (or if HFIX was used to set up the AFIX instruction) the torsion angle is chosen which maximizes the electron density (see m = 13). Since this torsion angle is unlikely to be very accurate, the use of a rotating group refinement is recommended (i.e. AFIX 147 before the H atom).

m = 15 BH group in which the boron atom is bonded to either four or five other atoms as part of an polyhedral fragment. The Hydrogen atom is placed on the vector that represents the negative sum of the unit vectors along the four or five other bonds to the boron atom.

m = 16 Acetylenic C-H, with X-C-H linear. Usually refined with the riding model, i.e. AFIX 163.

m > 16 A group defined in a FRAG...FEND section with code = m is fitted, usually as a preliminary to rigid group refinement. The FRAG...FEND section MUST precede the corresponding AFIX instruction in the '.*ins*' file, but there may be any number of AFIX instructions with the same m corresponding to a single FRAG...FEND section.

When a group is fitted (m = 5, 6, 10 or 11, or m > 16), atoms with non-zero coordinates are used as target atoms with equal weight. Atoms with all three coordinates zero are ignored. Any three or more non-colinear atoms may be used as target atoms.

'Riding' (n = 3, 4) and 'rotating' (n = 7, 8) Hydrogen atoms, but not other idealized groups, are re-idealized (if m is 1, 2, 3, 4, 8, 9, 12, 13, 14, 15 or 16) before each refinement cycle (after the first cycle, the coordinates of the first Hydrogen of a group are always non-zero, so the torsion angle is retained on re-idealizing). For n = 4 and 8, the angles are re-idealized but the (refined) X-H bond length is retained, unless the Hydrogen coordinates are all zero, in which case d (on the AFIX instruction) or (if d is not given) a standard value which depends on the chemical environment and temperature (TEMP) is used instead.

n : geometric constraints

n = 0 No action.

n = 1 The coordinates, s.o.f. and U or U_{ij} are fixed.

n = 2 The s.o.f. and U (or U_{ij}) are fixed, but the coordinates are free to refine.

n = 3 The coordinates, but not the s.o.f. or U (or U_{ij}) 'ride' on the coordinates of the previous atom with n not equal to 3. The same shifts are applied to the coordinates of both atoms, and both contribute to the derivative calculation. The atom on which riding is performed may not itself be a riding atom, but it may be in a rigid group (m = 5, 6 or 9).

n = 4 This constraint is the same as n = 3 except that the X-H distance is free to refine. The X-H vector direction does not change. This constraint requires better quality reflection data than n = 3, but allows for variations in apparent X-H distances caused by libration and bonding effects. If there is more than one equivalent Hydrogen, the same shift is applied to each equivalent X-H distance (e.g. to all three C-H bonds in a methyl group). n = 4 may be combined with DFIX or SADI restraints (to restrain chemically equivalent X-H distances to be equal) or embedded inside a rigid (n = 6) group, in which case the next atom (if any) in the same rigid group must follow an explicit AFIX instruction with n = 5. Note that n = 4 had a different effect in SHELX-76.

n = 5 The next atom(s) are 'dependent' atoms in a rigid group. Note that this is automatically generated for the atoms following an n = 6 or n = 9 atom, so does not need to be included specifically unless m has to be changed (e.g. AFIX 35 before the first Hydrogen of a rigid methyl group with AFIX 6 or 9 before the preceding carbon).

n = 6 The next atom is the 'pivot atom' of a NEW rigid group, i.e. the other atoms in the rigid group rotate about this atom, and the same translational shifts are applied to all atoms in the rigid group.

n = 7 The following (usually Hydrogen) atoms (until the next AFIX with n not equal to 7) are allowed to ride on the immediately preceding atom X and rotate about the Y-X bond; X must be bonded to one and only one atom Y in the

connectivity list, ignoring the n = 7 atoms (which, if they are F rather than H, may be present in the connectivity list). The motion of the atoms of this 'rotating group' is a combination of riding motion (c.f. n = 3) on the atom X plus a tangential component perpendicular to the Y-X and X-H bonds, so that the X-H distances, Y-X-H and H-X-H angles remain unchanged. This constraint is intended for -OH, -CH$_3$ and possibly -CF$_3$ groups. X may be part of a rigid group, which may be resumed with an AFIX n = 5 following the n = 7 atoms.

n = 8 This constraint is similar to n = 7 except that the X-H distances may also vary, the same shifts being applied along all the X-H bonds. Thus only the Y-X-H and H-X-H angles are held constant; the relationship of n = 8 to n = 7 corresponds to that of n = 4 to n = 3. DFIX and SADI restraints may be useful for the X-H distances. This constraint is useful for -CF$_3$ groups or for -CH$_3$ groups with good data.

n = 9 The first (pivot) atom of a new 'variable metric' rigid group. Such a group retains its 'shape' but may shrink or expand uniformly. It is useful for C$_5$H$_5$ and BF$_4$ groups, which may show appreciable librational shortening of the bond lengths. Subsequent atoms of this type of rigid group should have n = 5, which is generated automatically by the program if no other AFIX instruction is inserted between the atoms. Riding atoms are not permitted inside this type of rigid group. Only the pivot atom coordinates may be fixed (by adding 10) or tied to free variables, and only the pivot atom may lie on a special position (for the automatic generation of special position constraints).

Although there are many possible combinations of m and n, in practice only a small number is used extensively, as discussed in the section on Hydrogen atoms. Rigid group fitting and refinement (e.g. AFIX 66 followed by six atoms of a phenyl ring or AFIX 109 in front of a Cp* group) is particularly useful in the initial stages of refinement; atoms not found in the structure solution may be given zero coordinates, in which case they will be generated from the rigid group fit.

A rigid group or set of dependent Hydrogens must ALWAYS be followed by 'AFIX 0' (or another AFIX instruction). Leaving out 'AFIX 0' by mistake is a common cause of error; the program is able to detect and correct some obvious cases, but in many cases this is not logically possible.

ANIS *n*

The next **n** isotropic non-Hydrogen atoms are made anisotropic, generating appropriate special position constraints for the U_{ij} if required. Intervening atoms which are already anisotropic are not counted. A negative **n** has the same effect.

ANIS names

The named atoms are made anisotropic (if not already), generating the appropriate constraints for special positions. Note that names may include '$' followed by a scattering factor name (see SFAC); 'ANIS $CL' would make all chlorine atoms anisotropic. Since ANIS, like other instructions, applies to the current residue unless otherwise specified, ANIS_* $S would be required to make the sulfur atoms in all residues anisotropic (for example). ANIS MUST precede the atoms to which it is to be applied. ANIS on its own, with neither a number nor names as parameters, makes all FOLLOWING non-Hydrogen atoms (in all residues) anisotropic. The L.S. and CGLS instructions provide the option of delaying the conversion to anisotropic of all atoms specified by ANIS until a given number of least-squares cycles has been performed.

BASF scale factors

Relative batch scale factors are included in the least-squares refinement based on the batch numbers in the *.hkl* file. For batch number *BN*, the F_c^2 value is multiplied by the (*BN-1*)'th scale factor from the BASF instruction, as well as by the overall scale factor. For batch number one (or zero), F_c is multiplied by the overall scale factor, but not by a batch scale factor. The least-squares matrix will be singular if there are no reflections with *BN=1* (or zero), so the program considers this to be an error. Note that BASF scale factors, unlike the overall scale factor (see FVAR) are relative to F^2, not *F*. For twinned crystals, i.e. when either TWIN or HKLF 5 are employed, BASF specifies the fractional contributions of the various twin components. BASF parameters may also be used by the HOPE instruction. Except when they are used by HOPE, the program does not allow BASF parameters to become negative.

BIND atom1 atom2

The specified 'bond' (which may be of any length) is added to the connectivity list if it is not there already. Only one of the two atoms may be an equivalent atom (i.e. have the extension _$n).

BLOC n1 n2 atomnames

If **_n1_ or _n2_ are positive**, the x, y and z parameters of the named atoms are refined in cycle |n1| or |n2| respectively. **If _n1_ or _n2_ are negative**, the occupation and displacement parameters are refined in the cycle. Not more than two such cycle numbers may be specified on a single BLOC instruction, but the same atoms may be mentioned in any number of BLOC instructions. To refine both x, y and z as well as displacement parameters for an atom in the same block, n1 and n2 should specify the same cycle number, but with opposite signs. A BLOC instruction with no atom names refines all atoms (in residue 0) in the specified cycles. The pattern of blocks is repeated after the maximum block number has been reached if the number of L.S. refinement cycles is larger than the maximum BLOC |n1| or |n2|. If a cycle number less than the maximum |n1| or |n2| is not mentioned in any BLOC instruction, it is treated as full-matrix. The overall scale, batch/twin scale factors, extinction coefficient, SWAT g parameter, HOPE parameters and free variables (if present) are refined in every block. Riding (Hydrogen) atoms and atoms in rigid groups are included in the same blocks as the atoms on which they ride.

For example, a polypeptide consisting of 30 residues (residue numbers 1..30 set by RESI instructions) could be refined efficiently as follows (all non-Hydrogen atoms assumed anisotropic):

BLOC 1
BLOC -2 N_1 > O_16
BLOC -3 N_14 > O_30

which would ensure 3 roughly equally sized blocks of about 800 parameters each and some overlap between the two anisotropic blocks to avoid problems where they join. The geometric parameters would refine in cycles 1,4,7 .. and the anisotropic displacement parameters in the remaining cycles. In this example it is assumed that the first atom in each residue is N and the last is O. An alternative good blocking strategy would be to divide the structure into three overlapping blocks of xyz and U_{ij} parameters, and to add a fourth cycle in which all xyz but no U_{ij} values are refined (these four blocks would then also each contain about 800 parameters), i.e.:

BLOC 1 -1 N_1 > O_11
BLOC 2 -2 N_10 > O_21
BLOC 3 -3 N_20 > O_30
BLOC 4

A BLOC instruction with no parameters fixes all atomic parameters (xyz, sof and U or U_{ij}). Such a BLOC instruction takes priority over all other BLOC instructions, irrespective of their order in the *.ins* file.

BOND atomnames

BOND outputs bond lengths for all bonds (defined in the connectivity list) that involve two atoms named on the same BOND instruction. Angles are output for all pairs of such bonds involving a common atom. Numerical parameters on a BOND instruction are ignored, but not treated as errors (for compatibility with SHELX-76). A BOND instruction with no parameters outputs bond lengths (and the corresponding angles) for ALL bonds in the connectivity table, and 'BOND $H' on its own includes all bonds to Hydrogens as well (but since the Hydrogens are not included in the connectivity table, bonds involving symmetry equivalent Hydrogens are not included). Other element names may also be referenced globally by preceding them with a '$' on a BOND instruction. BOND is set automatically by ACTA, and the bond lengths and angles are written to the *.cif* file. Note that the best way to calculate B-H-B angles is with RTAB !

BUMP s [0.02]

'Anti-bumping' restraints are generated automatically for all distances involving two non-bonded C, N, O and S atoms (based on the SFAC type) that are shorter than the expected shortest non-bonded distances, allowing for the possibility of Hydrogen bonds. All pairs of atoms that are not connected by one, two or three bonds in the connectivity table are considered to be non-bonded for this purpose. Anti-bumping restraints are also generated for short contacts between Hydrogen atoms (if present) provided that the two Hydrogen atoms are not bonded to the same atom; this should help to avoid energetically unfavorable side-chain conformations. If the sum of occupancies of the two atoms is less than 1.1, no restraint is generated; also if the atoms have different PART numbers and neither of them is zero no restraint is generated.

The default esd s is the first DEFS parameter (0.02 if there is no DEFS instruction). If s is given a negative sign, the absolute value is used as an esd, and symmetry equivalent atoms in the connectivity array are considered too in deciding which atoms are connected and so should not have anti-bumping restraints applied. **Thus when s is positive** (the default action if s is not specified on the BUMP instruction**) short contacts between appropriate atoms in different asymmetric units ALWAYS result in anti-bumping restraints**. This will be the normal procedure for macromolecular refinements (where it helps to eliminate accidental contacts between molecules in low-resolution refinements), but in the (unusual) case of a crystallographic twofold axis running through (say) a disulfide bond it will be necessary to make s negative to prevent the generation of anti-bumping restraints that would break the bond. Refinement with anti-bumping restraints provides a solvent model with acceptable Hydrogen bonding distances that is consistent with the diffraction data. The anti-bumping restraints are regenerated before each refinement cycle. Anti-bumping restraints can also be added by hand using DFIX instructions with negative distances d.

CELL λ *a* *b* *c* α β γ

Wavelength and unit-cell dimensions in Å and degrees.

CGLS nls[0] nrf[0] nextra[0] maxvec[511]

As L.S., but the Konnert-Hendrickson conjugate-gradient algorithm is employed instead of the full-matrix approach. Although BLOC may be used with CGLS, in practice it is much better to refine all parameters at once. CGLS is much faster than L.S. for a large number of parameters, and so will be the method of choice for most macromolecular refinements. The convergence properties of CGLS are good in the early stages (especially if there are many restraints), but cannot compete with L.S. in the final stages for structures which are small enough for full-matrix refinement. The major disadvantage of CGLS is that it does not provide estimated standard deviations, so that when a large structure has been refined to convergence using CGLS it may be worth performing a blocked full-matrix refinement (L.S./BLOC) to obtain the standard deviations in quantities of interest (e.g. torsion angles, in which case only xyz blocks would be required).The other parameters have the same meaning as with L.S.; CGLS is entirely suitable for R_{free} tests (**negative nrf**), and since it requires much less memory than L.S. there will rarely be any reason to change **maxvec** from its default value.

The CGLS algorithm is based closely on the procedure described by Hendrickson & Konnert (1980). The structure-factor derivatives contribute only to the diagonal elements of the least-squares matrix, but all 'additional observational equations' (restraints) contribute in full to diagonal and off-diagonal terms, although neither the l.s. matrix A nor the Jacobean J are ever generated. The preconditioning recommended by Hendrickson & Konnert is used to speed up the convergence of the internal conjugate gradient iterations, and has the additional advantage of preventing the excessive damping of poorly determined parameters characteristic of other conjugate gradient algorithms (Tronrud,1992).

A further refinement in the CGLS approach is to save the parameter shifts from the previous CGLS cycle, and to use them to improve the estimated parameter shifts in the current cycle. Since this is only possible in the second and subsequent cycles, an initial shift multiplier of 0.7 is assumed in the first cycle. If the refinement proves to be unstable, this starting value can be reset using the first DAMP parameter.

In addition to this optimization of the CGLS shift multiplication factor, the individual parameter shifts are monitored each L.S. or CGLS cycle, and the shift multiplication factors are reduced (to a value between 0.5 and 1) for parameters that tend to oscillate. This applies only to refinements in which BLOC is not used. This produces an additional improvement in the convergence of the least-squares refinement, but (unlike Marquardt damping) has no effect on esds.

CHIV V[0] s[0.1] atomnames

The chiral volumes of the named atoms are restrained <u>to the value</u> **V** (in Å^3) with <u>standard deviation</u> *s*. The chiral volume is defined as the volume of the tetrahedron formed by the three bonds to each named atom, which must be bonded to three and only three non-Hydrogen atoms in the connectivity list; the (ASCII) alphabetical order of the atoms making these three bonds defines the sign of the chiral volume. Note that RTAB may be used to list chiral volumes defined in the same way but without restraining them. The chiral volume is positive for the alpha-carbon (CA) of an L-amino-acid if the usual names (N, CB and C) are used for the three non-Hydrogen atoms bonded to it. It is also possible to define a chiral volume when two substituents are chemically equivalent but have different names; this may be useful to ensure that CB of a valine retains a pyramidal geometry with the conventional labeling of CG1 and CG2. Note that 'CHIV 0' (or just CHIV since the default V is zero) may be used to impose a planarity restraint on an atom which is bonded to three other non-Hydrogen atoms, by making its chiral volume zero. CHIV restraints with zero and non-zero target values are listed separately in the restraints summary printer out after each refinement cycle.

CONF atomnames

The named atoms define a chain of at least four atoms. CONF generates a list of torsion angles with esd's for all torsion angles defined by this chain. CONF is often used to specify an n-membered ring, in which case the first three atoms must be named twice (n+3 names in all). If no atoms are specified, all possible torsion angles not involving Hydrogen are generated from the connectivity array. The torsion angles generated by CONF are also written to the .cif file if an ACTA instruction is present. All torsion angles calculated by SHELXL follow the conventions defined by Allen & Rogers (1969).

CONN bmax[12] r[#] atomnames or CONN bmax[12]

The CONN instruction fine-tunes the generation of the connectivity table and is particularly useful when π-bonded ligands or metal ions are present in the structure. For the purposes of the connectivity table (which is always generated), bonds are all distances between non-Hydrogen atoms less than r1 + r2 + 0.5 Å, where r1 and r2 are the covalent radii of the atoms in question (taking PART into consideration as explained below). A shell of symmetry equivalent atoms is also generated, so that all unique bonds are represented at least once in the list. All bonds, including those to symmetry equivalent atoms, may be deleted or added using the FREE or BIND instructions.

Default values of r (identified by the scattering factor type) are stored in the program. These defaults may be changed (for both the connectivity table AND the PLAN -n output) by using the full form of the SFAC instruction. Alternatively the defaults may be overridden for the named atoms by specifying r on a CONN instruction, in which case r is used in the generation of the connectivity list but not by the PLAN instruction. '$' followed by an element name (the same as on a SFAC instruction) may also be employed on a CONN instruction (and also does not apply to PLAN). The second form of the CONN instruction may be used to change the maximum coordination number bmax for all atoms (which defaults to 12 if there is no CONN instruction).

If, after generating bonds as above and editing with FREE and BIND, there are more than bmax bonds to a given atom, the list is pruned so that only the bmax shortest are retained. A harmless side-effect of this pruning of the connectivity list is that symmetry operations may be stored and printed that are never actually used. Note that this option only removes one entry for a bond from the connectivity list, not both, except in the case of 'CONN 0' which ensures that there are no bonds to or from the named atoms. 'CONN 0' is frequently used to prevent the solvent water in macromolecular structures from making additional 'bonds' to the macromolecule which confuse the generation of idealized Hydrogen atoms etc. In some cases it will be necessary to use FREE to remove a 'bond' from a light atom to an alkali metal atom (for example) in order to generate Hydrogen atoms correctly. Refinements of macromolecules will often include BUMP and 'CONN 0 O_200 > LAST' (where the water happens to begin with residue 200). 'LAST' is used to indicate the last atom in the file, which saves trouble when adding extra waters.

The CONN instruction, like ANIS and HFIX, MUST precede the atoms to which it is to be applied. Repeated CONN instructions are allowed; the LAST relevant CONN preceding a particular atom is the one which is actually applied. CONN without atom names changes the default value of bmax for all following atoms. The following example illustrates the use of CONN:

CONN Fe 0
MPLA 5 C11 > C15 Fe
MPLA 5 C21 > C25 Fe
Fe .....
C11 .....
....
C25 .....

which would prevent bonds being generated from the iron atom to all 10 carbons in ferrocene. In this example, the distances of the iron atom from the two ring planes would be calculated instead.

DAMP damp[0.7] limse[15]

The DAMP parameters take different meanings for L.S. and CGLS refinements. For L.S., damp is usually left at the default value unless there is severe correlation, e.g. when trying to refine a pseudo-centrosymmetric structure, or refining with few data per parameter (e.g. from powder data). A value in the range 1-10000 might then be appropriate. The diagonal elements of the least-squares matrix are multiplied by (1+damp/1000) before inversion; this is a version of the Marquardt (1963) algorithm. A side-effect of damping is that the standard deviations of poorly determined parameters will be artificially reduced; it is recommended that a final least-squares cycle be performed with little or no damping in order to improve these estimated standard deviations. Theoretically, damping only serves to improve the convergence properties of the refinement, and can be gradually reduced as the refinement converges; it should not influence the final parameter values. However in practice damping also deals effectively with rounding error problems in the (single-precision) least-squares matrix algebra, which can present problems when the number of parameters is large and/or restraints are used (especially when the latter have small esd's), and so it may not prove possible to lift the damping entirely even for a well converged refinement.

Note the use of 'DAMP 0 0' to estimate esds but not apply shifts, e.g. when a final L.S. 1 job is performed after CGLS refinement.

For CGLS refinements, damp is the multiplicative shift factor applied in the first cycle. In subsequent CGLS cycles it is modified based on the experience in the previous cycles. If a refinement proves unstable in the first cycle, damp should be reduced from its default value of 0.7.

If the maximum shift/esd for a L.S. refinement (excluding the overall scale factor) is greater than limse, all the shifts are scaled down by the same numerical factor so that the maximum is equal to limse. If the maximum shift/esd is smaller than limse no action is taken. This helps to prevent excessive shifts in the early stages of refinement. limse is ignored in CGLS refinements.

DANG d s[0.04] atom pairs

This instruction is interpreted in exactly the same way as DFIX, but the default value of s is twice the value of the first DEFS parameter (i.e. 0.04 if no DEFS instruction is used). The DFIX and DANG instructions appear separately in the table of restraint statistics. DANG is usually used for 1,3 or `angle distances', i.e. distances between two atoms that are both bonded to the same atom. The distance between the first and second named atom, the third and fourth, fifth and sixth etc. (if present) is restrained to a target value d with an estimated standard deviation s. The value of d may refer to a 'free variable', otherwise it is considered to be fixed. Fixing **d** by adding 10 is not allowed, so the value may lie between 0 and 15.

DEFS sd[0.02] sf[0.1] su[0.01] ss[0.04] maxsof[1]

DEFS may be used to change the default effective standard deviations for the following DFIX, SAME, SADI, CHIV, FLAT, DELU and SIMU restraints, and is useful when these are to be varied systematically to establish the optimum values for a large structure (e.g. using R_{free}). sd is the default for s in the SADI and DFIX instructions, and also for s1 and s2 in the SAME instruction. sf is the default effective standard deviation for CHIV and FLAT, su is the default for both s1 and s2 in DELU, and ss is the default s for SIMU. The default st for SIMU is set to twice the default s.

maxsof is the maximum allowed value that an occupation factor can refine to; occupation factors that are fixed or tied to free variables are not restricted. It is possible to change this parameter (to say 1.1 to allow for Hydrogen atoms) when refining both occupation factors and U's for solvent water in proteins (a popular but suspect way of improving the R factor).

DELU s1[0.01] s2[0.01] atomnames

All bonds in the connectivity list connecting atoms on the same DELU instruction are subject to a 'rigid bond' restraint, i.e. the components of the (anisotropic) displacement parameters in the direction of the bond are restrained to be equal within an effective standard deviation $s1$. The same type of restraint is applied to 1,3-distances as defined by the connectivity list (atoms 1, 2 and 3 must all be defined on the same DELU instruction). If $s2$ is omitted it is given the same value as s1. A **zero** value for $s1$ or $s2$ switches off the corresponding restraint. If no atoms are specified, all non-Hydrogen atoms are assumed. DELU is ignored if (in the refinement cycle in question) one or both of the atoms concerned is isotropic; in this case a 'hard' restraint is inappropriate, but SIMU may be used in the usual way as a 'soft' restraint. DELU without atom names applies to all non-Hydrogen atoms (in the current residue); DELU_* without atoms applies to all non-Hydrogen atoms in all residues. SFAC element names may also be referenced, preceded by the symbol '$'. The default values of $s1$ and $s2$ may be changed by means of a preceding DEFS instruction.

DFIX d s[0.02] atom pairs

The distance between the first and second named atom, the third and fourth, fifth and sixth etc. (if present) is restrained to a target value d with an estimated standard deviation s. The value of d may refer to a 'free variable', otherwise it is considered to be fixed. Fixing d by adding 10 is not allowed, so the value may lie between 0 and 15.

If d is given a negative sign, the restraint is applied ONLY if the current distance between the two atoms is LESS than |d|. This is an 'anti-bumping' restraint, and may be used to prevent solvent (water) molecules from approaching too close to one another or to a macromolecule. Anti-bumping restraints may also be generated automatically using the BUMP instruction. The default value of s is 0.02. The default s may be changed by means of a preceding DEFS instruction.

DISP E f' f" [#] mu [#]

The DISP instruction allows the dispersion and (optionally) the absorption coefficient of a particular element E (the name may be optionally prefaced by '$') to be read in without having to use the full form of the SFAC instruction. It will typically be used for synchrotron data where the wavelength does not correspond to the values (for Cu, Mo and Ag radiation) for which these terms are stored in the program. All other terms on the SFAC instruction are independent of the wavelength, so its short form may then be used. DISP instructions, if present, MUST come between the last SFAC and the UNIT instruction.

EADP atomnames

The same isotropic or anisotropic displacement parameters are used for all the named atoms. The displacement parameters (and possibly free variable references) are taken from the named atom which precedes the others in the atom list, and the actual values, free variable references etc. given for the U_{ij} of the other atoms are ignored. The atoms involved must either be all isotropic or all anisotropic. An atom should not appear in more than one EADP instruction. 'Opposite' Fluorines of PF_6 or disordered -CF_3 groups are good candidates for EADP, e.g.

EADP F11 F14
EADP F12 F15
EADP F13 F16
C1 ....
PART 1
F11 ...... 21 ......
F12 ...... 21 ......
F13 ...... 21 ......
PART 2
F14 ...... -21 ......
F15 ...... -21 ......
F16 ...... -21 ......
PART 0

EADP applies an (exact) *constraint*. The SIMU instruction *restrains* the Uij components of neighboring atoms to be approximately equal with an appropriate (usually fairly large) esd.

END

END is used to terminate an 'include' file, and may also be included after HKLF in the *.ins* file (for compatibility with SHELX-76).

EQIV $n symmetry operation

Defines symmetry operation *$n* for referencing symmetry equivalent atoms on any instruction which allows atom names, by appending '_$n' (where n is an integer between 1 and 511 inclusive) to the atom name. Such a symmetry operation must be defined before it is used; it does not have to be an allowed operation of the space group, but the same notation is used as on the SYMM instruction. The same *$n* may not appear on two separate EQIV instructions. Thus:

EQIV $2 1-x, y, 1-z

CONF C1 C2 C2_$2 C1_$2

could be used to calculate a torsion angle across a crystallographic twofold axis (note that this may be required because CONF with no atom names only generates torsion angles automatically that involve the unique atom list and a one atom deep shell of symmetry equivalents). If the instruction *codeword* refers to a residue, this is applied to the named atoms before any symmetry operation specified with '_$n'. Thus:

RTAB_23 O..O OG_12 O_$3

would calculate the (Hydrogen bond) distance between OG_12 and (O_23)_$3, i.e. between OG in residue 12 and the equivalent obtained by applying the symmetry operation defined by EQIV $3 to the atom O in residue 23.

EXTI x[0]

An extinction parameter x is refined, where F_c is multiplied by:

$$k\left[1 + .001xF_c^2\lambda^3 \Big/ {\sin 2\theta}\right]^{-\frac{1}{4}}$$

where **k** is the overall scale factor. Note that it has been necessary to change this expression from SHELX-76 (which used an even cruder approximation) and XLS in SHELXTL version 4 (which used 0.002 instead of $0.001\lambda^3$). The wavelength dependence is needed for HKLF 2 (Laue) data. The program will print a warning if extinction (or SWAT - see below) may be worth refining, but it is not normally advisable to introduce it until all the non-Hydrogen atoms have been found. For twinned and powder data, the F_c^2 value used in the above expression is based on the total calculated intensity summed over all components rather than the individual contributions, which would be easier to justify theoretically (but makes little difference in practice). For the analysis of variance and .fcf output file, the F_o^2 values are brought onto the absolute scale of F_c^2 by dividing them by the scale factor(s) and the extinction factor. The above expression for the extinction is empirical and represents a compromise to cover both primary and secondary extinction; it has been shown to work well in practice but does not appear to correspond exactly to any of the expressions discussed in the literature. The article by Larson (1970) comes closest and should be consulted for further information.

EXYZ atomnames

The same x, y and z parameters are used for all the named atoms. This is useful when atoms of different elements share the same site, e.g. in minerals (in which case EADP will probably be used as well). The coordinates (and possibly free variable references) are taken from the named atom which precedes the others in the atom list, and the actual values, free variable references etc. given for the x, y and z of the other atoms are ignored. An atom should not appear in more than one EXYZ instruction.

FEND

This must immediately follow the last atom of a [FRAG](#) fragment.

FLAT s[0.1] four or more atoms

The named atoms are restrained to lie a common plane. This restraint is actually applied by restraining a sufficient number of tetrahedra involving the atoms in question to have (chiral) volumes of zero, using the same algorithm as CHIV. This way of applying a planarity restraint has good convergence properties because it does not fix the orientation of the plane in its current position. The value s should be given in Å^3 as for CHIV, but for comparison with other methods the *r.m.s.* deviation from the plane is also printed. The default values of s is set by the second DEFS parameter.

FMAP code[2] axis[#] nl[53]

The unique unit of the cell for performing the Fourier calculation is set up automatically unless specified by the user using FMAP and GRID; the value of axis must be non-zero to suppress the automatic selection. The program chooses a **53 x 53 x nl or 103 x 103 x nl** grid depending on the resolution of the data. The **axis** is 1, 2 or 3 to define the direction perpendicular to the layers. Dispersion corrections are applied (so that the resulting electron density is real) and Friedel opposites are merged after the least-squares refinement and analysis of variance but before calculating the Fourier synthesis. This will improve the map (and bring the maximum and minimum residual density closer to zero) compared with SHELX-76. In addition, since usually all the data are employed, reflections with σ (F) relatively large compared with F_c are weighted down. This should be better than the use of an arbitrary cutoff on $F_o/\sigma(F)$. The *r.m.s* fluctuation of the map relative to the mean density is also calculated; in the case of a difference map this gives an estimate of the 'noise level' and so may be used to decide whether individual peaks are significant. Usually FMAP 2 is employed to find missing atoms, but if a significant part of the structure is missing, FMAP 5 or 6 may be better. ACTA requires FMAP 2 so that the difference density is on an absolute scale.

If **code** is made negative, both positive and negative peaks are included in the list, sorted on the absolute value of the peak height. This is intended to be useful for neutron diffraction data.

code = 2: Difference electron density synthesis with coefficients (F_o-F_c) and phases ϕ(calc).

code = 3: Electron density synthesis with coefficients F_o and phases ϕ(calc).

code = 4: Electron density synthesis with coefficients ($2F_o$-F_c) and phases ϕ(calc). $F(000)$ is included in the Fourier summations for code = 3 and 4.

code = 5: Sim-weighted ($2F_o$-F_c) Fourier (Giacovazzo, 1992).

code = 6: Sim-weighted ($2F_o$-F_c) Fourier with coefficients sharpened by multiplying with $(E/F)^{1/2}$.

FRAG code[17] a[1] b[1] c[1] α [90] β[90] γ[90]

Enables a fragment to be input using a cell and coordinates taken from the literature. Orthogonal coordinates may also be input in this way. Such a fragment may be fitted to the set of atoms following an AFIX instruction with *m = code* (code must be greater than 16); there must be the same number of atoms in this set as there are following FRAG, and they must be in the same order. Only the coordinates of the FRAG fragment are actually used; atom names, *sfac* numbers, sof and U_{ij} are IGNORED. A FRAG fragment may be given anywhere between UNIT and HKLF or END, and must be terminated by a FEND instruction, but must precede any AFIX instruction which refers to it. This 'rigid fit' is often a preliminary to a rigid group refinement (AFIX with n = 6 or 9).

FREE atom1 atom2

The specified 'bond' is deleted from the connectivity list (if present). Only one of the two atoms may be an equivalent atom (i.e. have the extension _*$n*).

FVAR osf[1] free variables

The overall scale factor is followed by the values of the 'free variables' fv(2) ... The overall scale factor is given throughout as the square root of the scale factor which multiplies F_c^2 in the least-squares refinement [to make it similar to the scale factor in SHELX-76 which multiplied F_c], i.e. $osf^2 F_c^2$ is fitted to F_o^2.

SHELXL goes to some trouble to ensure that the initial value of the scale factor has very little influence. Firstly, if the initial scale is exactly 1.0, a quick structure factor summation with a small fraction of the total number of reflections is performed to estimate a new scale factor. If the values differ substantially then the new value is used. Secondly the scale factor is factored out of the least-squares algebra so that, although it is still refined, the only influence the previous value has is an indirect one via the weighting scheme and extinction correction.

Before calculating electron density maps and the analysis of variance, and writing the structure factor file (*name.fcf*), the observed F^2 values and esds are brought onto an absolute scale by dividing by the scale factor.

The **free variables** allow extra constraints to be applied to the atoms, e.g. for common site occupation factors or isotropic displacement parameters, and may be used in conjunction with the SUMP, DFIX and CHIV restraints. If there is more than one FVAR instruction, they are concatenated; they may appear anywhere between UNIT and HKLF (or END).

GRID sl[#] sa[#] sd[#] dl[#] da[#] dd[#]

Fourier grid, when not set automatically. Starting points and increments multiplied by 100. <u>s means starting value, *d* increment, *l* is the direction perpendicular to the layers, *a* is across the paper from left to right, and *d* is down the paper from top to bottom.</u> Note that the grid is 53 x 53 x nl points, i.e. twice as large as in SHELX-76, and that sl and dl need not be integral. The 103 x 103 x nl grid is only available when it is set automatically by the program.

HFIX mn U[#] d[#] atomnames

HFIX generates AFIX instructions and dummy Hydrogen atoms bonded to the named atoms, the AFIX parameters being as specified on the HFIX instruction. This is exactly equivalent to the corresponding editing of the atom list. The atom names may reference residues (by appending '_n' to the name, where n is the residue number), or SFAC names (preceded by a '$' sign). U may be any legal value for the isotropic temperature factor, e.g. 21 to tie a group of Hydrogen U value to free variable 2, or -1.5 to fix U at 1.5 times U(eq) of the preceding normal atom. HFIX MUST precede the atoms to which it is to be applied. If more than one HFIX instruction references a given atom, only the FIRST is applied. 'HFIX 0' is legal, and may be used to switch off following HFIX instructions for a given atom (which is useful if they involve '_*' or a global reference to a residue class).

HKLF n[0] s[1] r11...r33[1 0 0 0 1 0 0 0 1] wt[1] m[0]

The value **n** is <u>negative if reflection data follow</u>, otherwise they are read from the *.hkl* file. The data are read in FORMAT(3I4,2F8.2,I4) (except for |n| < 3) subject to FORTRAN-77 conventions. The data are terminated by a record with *h*, *k* and *l* all zero (except |n| = 1, which contains a terminator and a checksum). In the reflection formats given below, **BN** stands for batch number. If <u>**BN** is greater than one, F_c is multiplied by the (BN-1)'th coefficient</u> specified by means of <u>BASF</u> instructions (see below). If **BN** is zero or absent, it is reset to one. <u>The multiplicative scale **s** multiplies both F_o^2 and $\sigma(F_o^2)$</u> (or F_o and $\sigma(F_o)$ for n = 1 or 3). <u>The multiplicative weight **wt** multiplies all $1/\sigma^2$ values and **m** is an integer 'offset' needed to read 'condensed data' (HKLF 1)</u>; both are included for compatibility with SHELX-76. Negative n is also only retained for upwards compatibility; it is much better to keep the reflection data in the *name.hkl* file, otherwise the data can easily get lost when editing *name.res* to *name.ins* for the next job.

> **n = 1:** SHELX-76 condensed data (**BN** is set to one). 'Condensed data' impose unnecessary index restrictions and can introduce rounding errors; although they still have their uses (email!), SHELXL cannot generate condensed data and their use is discouraged.

> **n = 2:** *h k l* $F_o^2\sigma(F_o^2)$ BN [1] λ [#] in FORMAT(3I4,2F8.2,I4,F8.4) for refinement based on singlet reflections from Laue photographs. The data are assumed to be scaled for source intensity distribution and geometric factors and (if necessary) corrected for absorption. If λ is zero or absent the value from the CELL instruction is used. **n** = 2 switches off the merging of equivalent reflections BEFORE l.s. refinement (i.e. sets MERG 0); equivalents and measurements of the same reflections at different wavelengths are merged after least-squares refinement and the subsequent application of a dispersion correction, but before Fourier calculations.

The remaining options (n > 2) all require FORMAT(3I4,2F8.2,I4); other compatible formats (e.g. F8.0 or even I8) may be used for the floating point numbers provided that eight columns are used in all and a decimal point is present.

> **n = 3:** *h k l* $F_o\sigma(F_o)$ BN [1] (if **BN** is absent or zero it is set to 1).The use of data corresponding to this format is allowed but is NOT RECOMMENDED, since the generation of F_o and $\sigma(F_o)$ from F_o^2 and $\sigma(F_o^2)$ is a tricky statistical problem and could introduce bias.

n = 4: $h \, k \, l \, F_o^2 \, \sigma(F_o^2)$ BN [1] is the standard reflection data file. Since F_o^2 is obtained as the difference of the experimental peak and background counts, it may be positive or slightly negative. *BN* may be made negative (e.g. by SHELXPRO) to flag a reflection for inclusion in the R_{free} reference set (see CLGS and L.S. with a second parameter of -1).

n = 5: $h \, k \, l \, F_o^2 \, \sigma(F_o^2) \, \boldsymbol{m}$ where *m* is the twin component number. Each measured F_o^2 value is fitted to the sum of $k_{|m|}F_{c|m|}^2$ over all contributing components, multiplied by the overall scale factor. The value *m* should be given as positive for the last contributing component and negative for the remaining ones (if any). The values of F_o^2 and $\sigma(F_o^2)$ are taken from the last ('prime') reflection in a group, and may simply be set equal for each component, but the indices *h,k,l* will in general take on different values for each component. The starting values of the twin factors $k_2..k_{\text{max}(m)}$ are specified on BASF instruction(s); k_1 is given by one minus the sum of the other twin factors. Note that many simple forms of twinning can also be handled with HKLF 4 and a TWIN instruction to generate the indices of the remaining twin component(s); HKLF 5 is required if the reciprocal space lattices of the components cannot be superimposed exactly. HKLF 5 sets MERG 0, and may not be used with TWIN.

n = 6: $h \, k \, l \, F_o^2 \, \sigma(F_o^2) \, \boldsymbol{m}$ as for n = 5, there may be one or more sets of reflection indices corresponding to a single F_o^2 value. The last reflection in a group has a positive *m* value and the previous members of the group have negative *m*. The values of F_o^2 and $\sigma(F_o^2)$ are taken from the last ('prime') reflection in a group, and may simply be set to the same values for the others. The value *m* is here the reflection MULTIPLICITY, and is defined as the number of equivalent permutations of the given *h, k* and *l* values, not counting Friedel opposites. This is intended for fitting resolved powder data for high symmetry crystal systems. For example, in a powder diagram of a crystal in the higher cubic Laue class (m3m) the reflections 3 0 0 (with multiplicity 3) and 2 2 1 (multiplicity 12) would contribute to the same measured F_o^2. HKLF 6 sets MERG 0. HKLF 6 may not be used with BASF or TWIN.

THERE MAY ONLY BE ONE HKLF INSTRUCTION AND IT MUST COME LAST, except when HKLF -n is followed by reflection data in the *.ins* file, in which case the file is terminated by the end of the reflection data. Negative n is retained for compatibility with SHELX-76 but is not recommended!

HOPE nh [1]

Refines 12 anisotropic scaling parameter as suggested by Parkin, Moezzi & Hope (1995). The value *nh* points to the BASF parameter that stores the value of the first HOPE parameter; if *nh* is negative the 12 parameters are fixed at their current values. These parameters are highly correlated with the individual atomic anisotropic displacement parameters, and so are only useful for structures that are refined isotropically, e.g. macromolecules at moderate resolution. To some extent they can also model absorption errors. If HOPE is given without any parameters and there are no BASF instructions, the program will generate appropriate starting values. If BASF parameters are needed for twin refinement or as scale factors for different batches of data, *nh* should be given an absolute value greater than one.

HTAB dh[2.0]

The new HTAB instruction provides an analysis of the Hydrogen bonds. A search is made over all polar Hydrogens (i.e. Hydrogen bonded to electronegative elements) present in the structure, and Hydrogen bonds printed for which: **H***A < r(A)+*dh* and <DHA > 110º**. If it appears likely that the Hydrogens have been assigned wrongly (e.g. two -OH groups have been assigned to the same O***O vector) a suitable warning message appears. This output should be checked carefully, since the algorithms used by HFIX/AFIX to place Hydrogens are by no means infallible! To obtain esd's on the distances and angles involved in the Hydrogen bond, the second form of the HTAB instruction (and if necessary EQIV) should be used (see below); HTAB without atom names is used first to find the necessary symmetry transformations for EQIV.

HTAB donor-atom acceptor-atom

The second form of the HTAB instruction is required to generate the esds and the CIF output records. The donor atom D and acceptor A should be specified; the program decides which of the Hydrogen atoms (if any) makes the most suitable Hydrogen bond linking them. Only the acceptor atom may specify a symmetry operation (_$n) because this standard CIF entry for publication in Acta Crystallographica requires this.

ISOR s[0.1] st[0.2] atomnames

The named atoms are *restrained* with effective *standard deviation s* so that their U_{ij} components approximate to isotropic behavior; however the corresponding isotropic U is free to vary. ISOR is often applied, perhaps together with SIMU, to allow anisotropic refinement of large organic molecules when the data are not adequate for unrestrained refinement of all the U_{ij}; in particular ISOR can be applied to solvent water for which DELU and SIMU are inappropriate. ISOR should in general be applied as a weak restraint, i.e. with relatively large sigmas, for the reasons discussed above (see SIMU); however it is also useful for preventing individual atoms from becoming 'non-positive-definite'. However it should not be used indiscriminately for this purpose without investigating whether there are reasons (e.g. disorder, wrong scattering factor type etc.) for the atom going n.p.d. If (according to the connectivity table, i.e. ignoring attached Hydrogens) the atom is terminal (or makes no bonds), *st* is used instead as the esd. If s but not st is specified, *st* is set to twice *s*. If no atoms are given, all non-Hydrogen atoms are understood. SFAC element names may also be referenced, preceded by '$'. The value of *s* or *st* may be set to zero to switch off the appropriate restraints. ISOR without atom names (or ISOR_* if residues are used) applies this restraint to all non-Hydrogen atoms. Note also the use of the keyword 'LAST' to indicate the last atom in the .ins file; an anisotropic refinement of a macromolecule will often include:

ISOR 0.1 O_201 > LAST

assuming that the solvent water starts with O_201 and continues until the end of the atom list. ISOR should in general be given a much larger esd (and hence lower weight) than DELU; whereas there is good evidence that DELU restraints should hold accurately for most covalently bonded systems, ISOR (and SIMU) are only rough approximations to reality.

L.S. nls[0] nrf[0] nextra[0] maxvec[511]

nls cycles of full-matrix least-squares refinement are performed, followed by a structure factor calculation. When L.S. (or CGLS) is combined with BLOC, each cycle involves refinement of a block of parameters which may be set up differently in different cycles. If no L.S. or CGLS instruction is given, 'L.S. 0' is assumed.

If ***nrf*** is positive, it is the number of these cycles that should be performed before applying ANIS. This two-stage refinement is particularly suitable for the early stages of least-squares refinement; experience indicates that it is not advisable to let everything go at once!

Negative ***nrf*** indicates which reflections should be ignored during the refinement but used instead for the calculation of free R-factors in the final structure factor summation; for example L.S. 4 -10 would ignore every 10th reflection for refinement purposes. It is desirable to use the same negative value of ***nrf*** throughout, so that the values of '$R1$(free)' and 'w$R2$(free)' are not biased by the 'memory' of the contribution of these reflections to earlier refinements. These independent R-factors (Brünger, 1992) may be used to calibrate the sigmas for the various classes of restraint, and provide a check as to whether the data are being 'over-refined' (primarily a problem for macromolecules with a poor data to parameter ratio). In SHELXL, these ignored reflections are not used for Fourier calculations.

nrf=-1 selects the R_{free} reference set that is flagged (with negative batch numbers) in the *.hkl* file (SHELXPRO may be used to do this). The division of the data into reference and working set is then independent of the space group and the MERG, OMIT and SHEL settings. However on merging reflections, to play it safe a reflection is retained in the reference set only if all equivalents have the R_{free} flag set. Thus if equivalents are present, it is a good idea to use the SHELXPRO option to set the R_{free} flag in thin shells, so that all equivalents of a particular unique reflection are either all in the reference set or all in the working set. ***nrf=-1*** is the recommended way of applying the R_{free} test in SHELXL.

nextra is the number of additional parameters which were derived from the data when performing empirical absorption corrections etc. It should be set to 44 for DIFABS [or 34 without the theta correction; Walker & D. Stuart (1983)]. It ensures that the standard deviations and GooF are estimated correctly; they would be underestimated if the number of extra parameters is not specified. ***nextra*** is zero (and so can be omitted) if extra information in the form of indexed crystal faces or psi-scan data was used to apply an absorption correction.

maxvec refers to the maximum number of reflections processed simultaneously in the rate-determining calculations. Usually the program maximizes all available memory to process as many reflections as possible simultaneously, subject to a maximum of ***maxvec***, which may not be larger than 511. For complicated reasons involving the handling of suppressed and 'R_{free}' reflections and input/output buffering, some blocks may be smaller than the maximum, especially if the facilities for refinement against twinned or powder data are being used. It may be desirable to set ***maxvec*** to a smaller number than 511 to prevent unnecessary disk transfers when large structures are refined on virtual memory systems with limited physical memory.

LATT N[1]

Lattice type N: 1=P, 2=I, 3=rhombohedral obverse on hexagonal axes, 4=F, 5=A, 6=B, 7=C. N must be made negative if the structure is non-centrosymmetric.

LAUE E

Wavelength-dependent values of f' and f" may be defined for an element **E** by means of the LAUE instruction, which is used in conjunction with the HKLF 2 reflection data format (in which the wavelength is given separately for each reflection). This is primarily intended for refinement of structures against Laue data collected using synchrotron radiation, but could also be used for refinement of a structure using data collected at different wavelengths for which some of the dispersion terms are significant (e.g. MAD data for macromolecules). There is no provision for handling overlapping reflection orders, and scaling for the source intensity distribution and Lp, absorption corrections etc. must have been performed before using SHELXL. A dummy wavelength of say 0.7 Å should be given on the CELL instruction, and the absorption coefficient estimated by the program should be ignored.

The element symbol may be preceded by '$' but this is optional; it must be followed by at least one blank or the end of the line. Any remaining information on the LAUE instruction line is ignored. The line immediately following the LAUE instruction is always ignored, and so may be used for headings. The following lines contain values of wavelength (in Å), f' and f" in FORMAT(F7.3,2F8.3); further information (e.g. the absorption coefficient μ) may follow on the same line but will be ignored. The wavelength values must be in ascending order and will be linearly interpolated; the wavelength intervals do not need to be equal (but it is more efficient if most of them are) and should indeed be smaller in the region of an absorption edge. This list is terminated by a record in which all three values are given as zero. There should only be one LAUE instruction for each element type; if a reflection wavelength is outside the range specified, the constant f' and f" values defined by the corresponding SFAC instruction are used instead.

A LAUE instruction must be preceded by (normal) SFAC and UNIT instructions referencing the elements in question, and by all atoms. Thus the LAUE instruction(s) are usually the last instructions before HKLF 2 (or -2) at the end of the *.ins* file (which facilitates editing). The +filename construction may conveniently be used to read long LAUE tables from 'include' files without echoing them.

LIST m[#] mult[1]

m = 0: No action.

m = 1: Write h,k,l, F_o, F_c and phase (in degrees) to *.fcf* in X-PLOR format. Only unique reflections after removing systematic absences, scaling [to an absolute scale of F(calc)], applying dispersion and extinction or <u>SWAT</u> corrections (if any), and merging equivalents including Friedel opposites are included. If F_o^2 was negative, F_o is set to zero. Reflections suppressed by <u>OMIT</u> or <u>SHEL</u> [or reserved for R(free)] are not included.

m = 2: List h,k,l, F_o, $\sigma(F_o)$ and phase angle in degrees in FORMAT(3I4,2F8.2,I4) for the reflection list as defined for m = 1.

m = 3: List h,k,l, F_o, $\sigma(F_o)$, A(real) and B(imag) in FORMAT(3I4,4F8.2), the reflections being processed exactly as for m = 2.

m = 4: List h,k,l, F_c^2, F_o^2, $\sigma(F_o^2)$ and a one-character status flag. F_o^2 are scaled to F_c^2 and possibly corrected for extinction, but no corrections have been made for dispersion and no further merging has been performed. FORMAT (3I4,2F12.2,F10.2,1X,A1) is employed. The status flag is 'o' (observed), 'x' [observed but suppressed using 'OMIT *hk l*', SHEL or reserved for R(free)], or '<' (F_o^2 is less than $t\sigma(F_o^2)$, where t is one half of the *F*-threshold <u>*s*</u> specified on an OMIT instruction).

m = 5: Write h,k,l, F_o, F_c, and ϕ(phase angle in degrees) in FORMAT(3I4,2F10.2,F7.2) for the reflection list as defined for ***m = 1***. Like the ***m = 1*** option, this is intended for input to some standard macromolecular FFT programs (such as W. Furey's PHASES program), thereby providing a possible route to a graphical display of the electron density.

m = 6: Write a free-format CIF file containing h,k,l, F_o^2, $\sigma(F_o^2)$, F_c and ϕ (phase angle in degrees) for the reflection list as defined for ***m = 1***. This is the <u>recommended format for the deposition of reflection data with the PDB</u>, and is also the format required for the generation of refinement statistics and electron density maps using SHELXPRO.

<u>For m = 4 only</u>, ***mult*** <u>is a constant multiplicative factor applied to all the quantities output</u> (except the reflection indices!), and may be used if there are scaling problems. For other *m* options ***mult*** is ignored. For m = 2,3 or 4 only a blank line is included at the end of the file as a terminator. The reflection list is written to the file *name.fcf*, which is in CIF format for ***m*** = 3, 4 or 6; however the actual reflections are always in fixed format except for ***m*** = 1 or 6. The program CIFTAB can - amongst other options - read the **m = 4** output and print $F_o/F_c/\sigma(F)$ tables in compact form on an HP-compatible laser printer. ***m*** = 4 <u>is the standard archive format for small-molecule structures</u>, ***m*** = 6 for macromolecules (with Friedel opposites averaged). Since the final refinement is normally performed on all data (including the R_{free} reference set) the LIST 6 output is not able to flag the R_{free} reflections.

MERG n[2]

If **n** is equal to 2 the reflections are sorted and merged before refinement; if the structure is non-centrosymmetric the Friedel opposites are not combined before refinement (necessary distinction from SHELXS). If **n** is 1 the indices are converted to a 'standard setting' in which *l* is maximized first, followed by *k*, and then *h*; if n is zero, the data are neither sorted nor converted to a standard setting. **n = 3 is the same as n = 2** except that Friedel opposites are also merged (this introduces small systematic errors and should only be used for good reason, e.g. to speed up the early stages of a refinement of a light atom structure before performing the final stages with MERG 2). Note that the reflections are always merged, and Friedel opposites combined, before performing Fourier calculations in SHELXL so that the (difference) electron density is real and correctly scaled. Even with n = 0 the program will change the reflection order within each data block to optimize the vectorization of the structure factor calculations (it is shuffled back into the MERG order for LIST 4 output). Note that MERG may not be used in conjunction with TWIN or HKLF 5 or 6. In SHELX-76, MERG 3 had a totally different meaning, namely the determination of inter-batch scale factors; in SHELXL, these may be included in the refinement using the BASF instruction.

MERG 4 averages all equivalents, including Friedel opposites, and sets all $\Delta f''$ values to zero; it is often used in refinement of macromolecules.

MOLE n

Forces the following atoms, and atoms or peaks that are bonded to them, into molecule n of the PLAN output. The value of n may not be greater than 99. $n = 99$ has a special meaning: the 'lineprinter plot' is suppressed for the following atoms, but the table of distances is still printed. This is sometimes useful for saving paper.

MORE m [1]

MORE sets the amount of (printer) output; *m* takes a value in the range 0 (least) to 3 (most verbose). MORE 0 also suppresses the echoing to the .lst file of any instructions or atoms which follow it (until the next MORE instruction).

MOVE dx[0] dy[0] dz[0] sign[1]

The coordinates of the atoms that follow this instruction are changed to: x = dx + $sign$ * x, y = dy + $sign$ * y, z = dz + $sign$ * z until superseded by a further MOVE.   MOVE should not be used at the same time as the specification of zero coordinates to indicate that an atom should not be used in fitting a fragment of known geometry (e.g. AFIX 66), because after the move the coordinates will no longer be zero!

MPLA na atomnames

A least-squares plane is calculated through the first *na* of the named atoms, and <u>the equation of the plane and the deviations of all the named atoms from the plane are listed</u> with estimated standard deviations (from the full covariance matrix). The angle to the previous least-squares plane (if any) is also calculated, but some approximations are involved in estimating its esd. The value of *na* must be at least 3. <u>If *na* is omitted the plane is fitted to all the atoms specified</u>.

NCSY DN sd[0.1] su[0.05] atoms

The NCSY instruction applies local non-crystallographic symmetry restraints. In contrast to the widely used global NCS constraints, these do not save any CPU time but do not require the definition (and refinement) of a matrix transformation and mask. They are also very flexible, and can accommodate rotation of the molecule about hinges etc. Since for macromolecules at modest resolution the 1,2- and 1,3-distances are normally restrained to fixed target values by DFIX and DANG restraints, the NCS restraints are generated for equivalent 1,4-distances (if *sd* is non-zero or absent) and equivalent isotropic U-values (if *su* is non-zero or absent). The default *sd* is set to five times the first DEFS parameter, and the default *su* is equal to the fourth DEFS parameter.

For each atom the program attempts to find an `equivalent' atom with the same name but with a residue number *DN* greater than the residue number of the named atom. If *sd* is greater than zero, the connectivity array is used to find 1,4-distances for which both atoms are specified in the same NCSY instruction; a SADI restraint is then created to make the distance equivalent to the same distance between the equivalent atoms. This is not quite the same as restraining torsion angles to be the same, because + and - gauche would have the same distance; however it is chemically plausible that equivalent side-chain conformations could differ in this way. If *su* is greater than zero (or absent), a SIMU restraint is generated to make the U-values approximately equal for each pair of `equivalent' atoms, provided that both are isotropic. NCS restraints should be used whenever possible for isotropic (protein) refinement at modest resolution, since they increase the effective data to parameter ratio and so have a similar effect to that of increasing the resolution of the data. They are also very easy to set up; for example, to apply three-fold NCS restraints to a protein structure containing three equivalent chains numbered 1001-1109, 2001-2109 and 3001-3109, the following two instructions are all that is required:

> **NCSY 1000 N_1001 > OT2_1109**
> **NCSY 2000 N_1001 > OT2_1109**

The atom list may easily be modified to leave out particular loops, residues or side-chains. This is not only easier than specifying a transformation matrix and mask: it also will correspond more closely to reality, because the restraints are more flexible than constraints and also act *locally* rather than *globally*.

OMIT s[-2] 2θ (lim)[180]

If *s* is given as negative, all reflections with $F_o^2 < 0.5\sigma(F_o^2)$ are replaced by $0.5\sigma(F_o^2)$; thus if no OMIT instruction is given the default action is to replace all F_o^2 values less than $-\sigma(F_o^2)$ by $-\sigma(F_o^2)$. If *s* is positive it is interpreted as a threshold for flagging reflections as 'unobserved'. Unobserved data are not used for least-squares refinement or Fourier calculations, but are retained for the calculation of *R*-indices based on all data, and may also appear (flagged with an asterisk) in the list of reflections for which F_o^2 and F_c^2 disagree significantly. Internally in the program *s* is halved and applied to F_o^2, so for positive F_o^2 the test is roughly equivalent to suppressing all reflections with $F_o < \sigma(F_o)$, as required for consistency with SHELX-76. Note that *s* may be set to 0 or (as in the default setting) to a negative threshold (to modify very negative F_o^2). An OMIT instruction with a positive *s* value is NOT ALLOWED in combination with ACTA, because it may introduce a bias in the final refined parameters; individual aberrant reflections may still be suppressed using OMIT *h k l*, even when ACTA is used.

2θ(lim) defines a limiting 2θ above which reflections are totally ignored; they are rejected immediately on reading in. This facility may be used to save computer time in the early stages of structure refinement, and is also sometimes useful for macromolecules. The SHEL command may also be used to ignore reflections above or below particular limiting resolution values.

OMIT followed by atom names but no numbers may be used to calculate an 'omit map'.

OMIT atomnames

The named atoms are retained in the atom list but ignored in the structure factor calculation and least-squares refinement. This instruction may be used, together with L.S. 0 and FMAP 2, to create an 'OMIT map' to get a clearer picture of disordered regions of the structure; this concept will be familiar to macromolecular crystallographers. In particular, 'OMIT $H' can be used to check the Hydrogen atom assignment of -OH groups etc. If an actual peak is present within 0.31Å of the calculated Hydrogen atom position, the electron density appears in the 'Peak' column of the output created by PLAN with a negative first parameter. OMIT_* $H must be used for this if residues are employed.

OMIT *h k l*

The reflection *h,k,l* (the indices refer to the standard setting after data reduction, and correspond to those in the list of `disagreeable reflections' after refinement) is ignored completely. Since there may be perfectly justified reasons for ignoring individual reflections (e.g. when a reflection is truncated by the beam stop) this form of OMIT is allowed with ACTA; however it should not be used indiscriminately. If MERG N with non-zero N is employed (or the (default) MERG 2 is assumed), all reflections which would generate the final indices *h,k,l* are ignored; if MERG 0 is specified, the indices must match those in the input *.hkl* file exactly.

PART n sof

The following atoms belong to PART n of a disordered group. The automatic bond generation ignores bonds between atoms with different PART numbers, unless one of them is zero (the value before the first PART instruction). If a site occupation factor (**sof**) is specified on the PART instruction, it overrides the value on the following atom instructions (even if set via an AFIX instruction) until a further PART instruction, e.g. 'PART 0', is encountered).

If **n** is negative, the generation of special position constraints is suppressed and bonds to symmetry generated atoms with the same or a different non-zero PART number are excluded; this is suitable for a solvent molecule disordered on a special position of higher symmetry than the molecule can take (e.g. a toluene molecule on an inversion center). A PART instruction remains in force until a further PART instruction is read; 'PART 0' should be used to continue with the non-disordered part of the structure.

Some care is necessary in generating Hydrogen atoms where disordered groups are involved. If the Hydrogen atoms are assigned a PART number, then even if the atom to which they are attached has no part number (i.e. PART 0) the above rules may be used by the program to work out the correct connectivity for calculating the Hydrogen atom positions. HFIX Hydrogens are assigned the PART number of the atom to which they are attached. If the Hydrogens and the atom to which they are attached belong to PART zero but the latter is bonded to atoms with non-zero PART, the LOWEST of these non-zero PART numbers is assumed to be the major component and is used to calculate the Hydrogen positions. In general, if the same residue numbers and names and the same atom names but different PART numbers are used for different disorder components in a macromolecule, HFIX will generate Hydrogen atoms correctly without any special action being required. For example the use of HFIX with the following disordered serine residue:

HFIX_Ser 33 N
HFIX_Ser 13 CA
HFIX_Ser 23 CB
HFIX_Ser 83 CG
:
RESI 32 Ser
N .....
CA .....
C .....
O .....
PART 1
CB 1 ... ... 21 ...
OG 4 ... ... 21 ...
PART 2
CB 1 ... ... -21 ...
OG 4 ... ... -21 ...
PART 0

would set up the AFIX Hydrogens as if the following had been input. Note that only one, fully occupied, Hydrogen is attached to CA; for this reason, and also to prevent small inconsistencies in the DFIX and DANG restraints, the disorder should be traced back one more atom than can be resolved (i.e. CB should be split even if it does not look as though this would be necessary in an electron density map):

RESI 32 Ser
N .....
AFIX 43
H0 2 ... ... 11 -1.2
AFIX 0
CA .....
AFIX 13
HA 2 ... ... 11 -1.2
AFIX 0
C .....
O .....
PART 1
CB 1 ... ... 21 ...
AFIX 23
HB1 2 ... ... 21 -1.2
HB2 2 ... ... 21 -1.2
AFIX 0
OG 4 ... ... 21 ...
AFIX 83
HG 2 ... ... 21 -1.5
AFIX 0
PART 2
CB 1 ... ... -21 ...
AFIX 13
HB1 2 ... ... -21 -1.2
HB2 2 ... ... -21 -1.2
AFIX 0
OG 4 ... ... -21 ...
AFIX 83
HG 2 ... ... -21 -1.5
AFIX 0
PART 0

where free variable 2 is the occupation factor for PART 1 (say 0.7) and the occupation factor of the second component is tied to 1-fv(2) (i.e. 0.3). The value for this free variable is set on the FVAR instruction and is free to refine. If there were more than two components, a linear free variable restraint (SUMP) could be used to restrain the sum of occupation factors to unity. The addition of disorder components after including Hydrogen atoms will require some hand editing and so is less efficient, but the auxiliary program SHELXPRO can be persuaded to do most of the work

PLAN npeaks[20] d1[#] d2[#]

If **npeaks** is positive a Fourier peak list is printed and written to the .res file; if it is negative molecule assembly and line printer plots are also performed. Distances involving peaks which are less than r1+r2+**d1** (the covalent radii r are defined via SFAC; 1 and 2 refer to the two atoms concerned) are printed and used to define 'molecules' for the line printer plots. Distances involving atoms and/or peaks which are less than r1+r2+|**d2**| are considered to be 'non-bonded interactions'; however distances in which both atoms are Hydrogen or at least one is carbon (recognized by SFAC label 'C') are ignored. These non-bonded interactions are ignored when defining molecules, but the corresponding atoms and distances are included in the line printer output. Thus an atom or peak may appear in more than one map, or more than once on the same map. A table of the appropriate coordinates and symmetry transformations appears at the end of each molecule.

Negative **d2** includes Hydrogen atoms in the line printer plots, otherwise they are left out (but included in the distance tables). For the purposes of the PLAN instruction, a Hydrogen atom is one with a radius of less than 0.4 Å. Peaks are assigned the radius of SFAC type 1, which is usually set to carbon. Peaks appear on the printout as numbers, but in the .res file they are given names beginning with 'Q' and followed by the same numbers. Peak heights are also written to the .res file (after the sof and dummy U values) in electrons $Å^{-3}$. See also MOLE for forcing molecules (and their environments) to be printed separately.

A default **npeaks** of +20 is set by FMAP; to obtain line printer plots, an explicit PLAN instruction with negative **npeaks** is required. If **npeaks** is positive the nearest unique atoms to each peak are tabulated, together with the corresponding distances. A table of shortest distances between peaks is also produced. For macromolecules and for users of the Siemens' SHELXTL system **npeaks** will almost always be positive! If **npeaks** is positive **d1** and **d2** have a different meaning. The default of **d1** is then -1 and causes the full peaklist to appear in the .res file. If it is positive (say 2.3) then the full peaklist is still printed in the .lst file, but only suitable candidates for (full occupancy) water molecules appear in the .res file (with SFAC 4 and U set to 0.75). The water molecules must be less than 4 Å from an atom which begins with 'O', 'N' or 'W', and may not be nearer than **d2** (default 3.0) from any atom which does not begin with 'O', 'N', 'W' or 'H', and may not be nearer than **d1** to any 'O', 'N' or 'W' atom or to other potential waters which have larger peak heights. This facility is intended for extending the water structure of proteins in connection with BUMP and SWAT. To include the waters in the next refinement job, their names need to be changed and they need to be moved to before the HKLF instruction at the end of the atom list in the new .ins file. This can be performed automatically using SHELXPRO. It is recommended that the last water be called 'LAST' on the ISOR and CONN instructions so that its name does not need to be updated each job.

The heights and positions of the highest (difference) electron density maximum and the deepest minimum are output irrespective of the PLAN parameters.

REM

If REM begins the line and is followed by a comment on the same line then this comment is copied to the results file (*.res*). A line beginning with at least one blank may also be used as a comment, but such comments are only copied to the *.res* file if the line is completely blank; REM comments are always copied. Comments may also be included on the same line as any instruction following the character '!', and are copied to the *.res* file (except in the case of atoms and FVAR, EXTI, SWAT and BASF instructions).

RESI class[ ] number[0] alias

Until the next RESI instruction, all atoms are considered to be in the specified 'residue', which may be defined by a *class* (up to four characters, beginning with a letter) or *number* (up to four digits) or both. The same atom names may be employed in different residues, enabling them to be referenced globally or selectively. The residue number should be unique to a particular residue, but the class may be used to refer to a class of similar residues, e.g. a particular type of amino acid in a polypeptide.

Residues may be referenced by any instruction that allows atom names; the reference takes the form of the character '_' followed by either the residue class or number without intervening spaces. If an instruction *codeword* is followed immediately by a residue number, all atom names referred to in the instruction are assumed to belong to that residue unless they are themselves immediately followed by '_' and a residue number, which is then used instead.

RTAB codename atomnames

Chiral volumes (one *atomname*), bonds (two), angles (three) and torsion angles (four *atomnames*) are tabulated compactly against residue name and number. *codename* is used to identify the quantity being printed; it must begin with a letter and not be longer than 4 characters (e.g. 'Psi' or 'omeg'). There may not be more than 4 atom names. It is assumed that the atoms have the same names in all the required residues. For chiral volumes only, the necessary bonds must be present in the connectivity list (the same conventions are employed as for CHIV). Since the atoms do not themselves have to be in the same residue (it is sufficient that the names match), the residue name (if any) is printed as that of the first named atom for distances, the second for angles, and the third in the case of torsion angles. The latter should be consistent with generally accepted conventions for proteins. A typical application of RTAB for small-molecule structures is the tabulation of Hydrogen-bonded distances and angles (with esd's) since these will not usually appear in the tables created automatically by BOND.

If RTAB refers to more than one residue (e.g. RTAB_*), it is ignored for those residues in which not all the required atoms can be found (e.g. some of the main chain torsional angles for the terminal residues in a protein).

RTAB_4 Ang N H0 O_11

would cause the calculation of an angle N_4 - H0_4 - O_11, where the first two atoms are in residue 4 and the third is in residue 11.

If the instruction *codeword* is followed immediately by a residue class, the instruction is effectively duplicated for all residues of that class. '_*' may be used to match all residue classes; this includes the default class ' ' (residue number 0) which applies until the first RESI instruction is encountered. Thus:

MPLA_phe CB > CZ

would calculate least-squares planes through atoms CB to CZ inclusive of all residues of class 'phe' (phenylalanine). In the special case of HFIX, only the FIRST instruction which applies to a given atom is applied. Thus:

HFIX_1 33 N
HFIX_* 43 N

would add Hydrogens to the N-terminal nitrogen (residue 1) of a polypeptide to generate a (protonated) $-NH_3^+$ group, but all other (amide) Nitrogens would become -NH-.

Individual atom names in an instruction may be followed by '_' and a residue number, but not by '_*' or '_' and a residue class. If an atom name is not followed by a residue number, the current residue is assumed (unless overridden by a global residue number or class appended to the instruction *codeword*). The symbols '_+' meaning 'the next residue' and '_-' meaning 'the

preceding residue'(i.e. residues number n+1 and n-1 if the current residue number is n) may be appended to atom names but not to instruction codenames. Thus the instruction:

RTAB_* Omeg CA_+ N_+ C CA

could be used to calculate all the peptide ωtorsion angles in a protein or polypeptide. If (as at the C-terminus in this example) some or all of the named atoms cannot be found for a particular residue, the instruction is simply ignored for that residue.

'_$n' does not refer to a residue; it uses the symmetry operation $n defined by a preceding 'EQIV $n' instruction to generate an equivalent of the named atom (see [EQIV](#)). The value of *alias* specifies an alternative value of the residue number so that cyclic chains of residues may be created; for a cyclic pentapeptide (residue numbers 2,3,..6) it could be set to 1 for residue 6 and to 7 for residue 2. If more than one RESI instruction refers to the same number, alias only needs to be specified once. alias is referenced only by the _+ and _- operations (see above), and a value used for alias may not be used as a residue number on a RESI instruction. Note that if there is more than one cyclic peptide in the asymmetric unit, it is a good idea to leave a gap of TWO residue numbers between them. E.g. a cyclic pentapeptide with two molecules in the asymmetric unit would be numbered 2 to 6 and 9 to 13, with aliases 7 on RESI 2, 1 on RESI 6, 14 on RESI 9 and 8 on RESI 13. It will generally be found convenient for applying restraints etc. to use the same names for atoms in identical residues. Since SHELXL does not recognize chain ID's (used in PDB format) it is normal to add a constant to the residue numbers to denote a different chain (e.g. chain A could be 1001 to 1234 and chain B 2001 to 2234). The auxiliary program SHELXPRO provides extensive facilities for handling residues.

SADI s[0.02] atom pairs

The distances between the first and second named atoms, the third and fourth, fifth and sixth etc. (if present) are restrained to be equal with an effective standard deviation s. The SAME and SADI restraints are analyzed together by the program to find redundant and implied restraints. The same effect as is obtained using SADI can also be produced by using DFIX with d tied to a free variable, but the latter costs one more least-squares parameter (but in turn produces a value and esd for this parameter). The default effective standard deviations for SADI may be changed by means of a DEFS instruction before the instruction in question.

SAME s1[0.02] s2[0.02] atomnames

The list of atoms (which may include the symbol '>' meaning all intervening non-Hydrogen atoms in a forward direction, or '<' meaning all intervening non-Hydrogen atoms in a backward direction) is compared with the same number of atoms which follow the SAME instruction. All bonds in the connectivity list for which both atoms are present in the SAME list are restrained to be the same length as those between the corresponding following atoms (with an effective standard deviation *s1*). The same applies to 1,3 distances (defined by two bonds in the connectivity list which share a common atom), with standard deviation *s2*. The default value of *s1* is taken from the first DEFS parameter; the default value of *s2* is twice this. The value of *s1* or *s2* may be set to zero to switch off the corresponding restraints. The program automatically sets up the n*(n-1)/2 restraint equations required when n interatomic distances should be equal. This ensures optimum efficiency and avoids arbitrary unequal weights. Only the minimum set of restraints needs to be specified in the *.ins* file; redundant restraints are ignored by the program, provided that they have the same sigma values as the unique set of restraints. See also SADI and NCSY for closely related restraints.

The position of a SAME instruction in the input file is critical. This creates problems for programs such as SHELXPRO that provide a user interface to SHELXL, and for protein refinements SADI is to be preferred (e.g. to apply 4m local symmetry to a heme group); normally for proteins most of the 1,2- and 1,3-distances will be restrained to target values using DFIX and DANG respectively anyway. However SAME provides an elegant way of specifying that chemically identical but crystallographically independent molecules have the same 1,2 and 1,3 distances, e.g.

**C1A
:
C19A
SAME C1A > C19A
C1B
:
C19B
SAME C1A > C19A
C1C
:
C19C**

etc. This requires just n-1 SAME instructions for n equivalent molecules. In a more complicated example, assume that a structure contains several toluene solvent molecules that have been assigned the same atom names (in the same order!) and the same residue name (Tol) but different residue numbers, then one SAME instruction suffices:

SAME_Tol C1 > C7

This instruction may be inserted anywhere except after the last Tol residue; the program applies it as if it were inserted before the next atom that matches C1_Tol . This is convenient for proteins with repeated non-standard residues, since one command suffices to apply suitable restraints, and no target values are needed, for compatibility with SHELXPRO tis SAME instruction has to be placed before the FVAR instruction. This is an exception to the usual rule that the action of a SAME instruction is position dependent; but it might be best to put it before a toluene residue with good geometry, since the connectivity table for this residue will be used to define the 1,2- and 1,3-distances. In this case it would also be reasonable to impose local two-fold symmetry for each phenyl ring, so a further SAME instruction could be added immediately before one toluene residue (the ring is assumed to be labeled cyclically C1 .. C6 followed by the methyl group C7 which is attached to C1):

SAME C1 C6 < C2 C7

which is equivalent to:

SAME C1 C6 C5 C4 C3 C2 C7

Note that these two SAME restraints are all that is required, however many PHE residues are present; the program will generate all indirectly implied 1,2 and 1,3 equal-distance restraints! In this case it would also be sensible to restrain the atoms of each tolune molecule to be coplanar by a FLAT restraint:

FLAT_Tol C1 > C7

SFAC elements

Element symbols which define the order of scattering factors to be employed by the program. The first 94 elements of the periodic system are recognized. The element name may be preceded by '$' but this is not obligatory (the '$' character is allowed for logical consistency but is ignored). The program uses the neutral atom scattering factors, f', f'' and absorption coefficients from International Tables for Crystallography, Volume C (1992), Ed. A.J.C. Wilson, Kluwer Academic Publishers, Dordrecht: Tables 6.1.1.4(pp. 500-502), 4.2.6.8 (pp. 219-222) and 4.2.4.2 (pp. 193-199) respectively. The covalent radii stored in the program are based on experience rather than taken from a specific source, and are deliberately overestimated for elements which tend to have variable coordination numbers so that 'bonds' are not missed, at the cost of generating the occasional 'non-bond'. The default radii (not those set for individual atoms by CONN) are printed before the connectivity table.

SFAC label a1 b1 a2 b2 a3 b3 a4 b4 c f' f'' mu r wt

Scattering factor in the form of an exponential series, followed by real and imaginary dispersion terms, linear absorption coefficient, covalent radius and atomic weight. Except for the 'label' and atomic weight the format is the same as that used in SHELX-76. label consists of up to 4 characters beginning with a letter (e.g. Ca2+) and should be included before a1; for consistency the first label character may be a '$', but this is ignored (note however that the '$', if used, counts as one of the four characters, leaving only three for the rest of the label). The two SFAC formats may be used in the same *.ins* file; the order of the SFAC instructions (and the order of element names in the first type of SFAC instruction) define the scattering factor numbers which are referenced by atom instructions. The units of mu should be barns/atom, as in Table 4.2.4.2 of International Tables, Volume C (see above). For neutrons this format should be used, with a1...b4 set to zero.

Hydrogen atoms are treated specially by SHELXL; they are recognized by having the scattering factor number that corresponds to 'H' on the SFAC instruction. For X-ray structures that contain both D and H, e.g. because the crystals were grown from a deuterated solvent in an N.M.R. tube (a common source of good crystals!), both H and D should be included on the SFAC and UNIT instructions, but all the H and D atoms should employ the 'H' scattering factor number. In this way the density will be calculated correctly, but the D atoms may be idealized using HFIX etc.

SHEL lowres[infinite] highres[0]

Reflections outside the specified resolution range in Å are ignored completely. This instruction may be useful for macromolecules.

SIMU s[0.04] st[0.08] dmax[1.7] atomnames

Atoms closer than **dmax** are restrained with effective standard deviation *s* to have the same U_{ij} components. If (according to the connectivity table, i.e. ignoring attached Hydrogens) one or both of the two atoms involved is terminal (or not bonded at all), *st* is used instead as the esd. If *s* but not *st* is specified, *st* is set to twice *s*. If no atoms are given, all non-Hydrogen atoms are understood. SIMU_* with no atoms applies to all non-Hydrogen atoms in all residues. SFAC element names may also be referenced, preceded by '$'. The interatomic distance for testing against **dmax** is calculated from the atom coordinates without using the connectivity table (though the latter is used for deciding if an atom is terminal or makes no bonds).

Note that SIMU should in general be given a much larger esd (and hence lower weight) than DELU; whereas there is good evidence that DELU restraints should hold accurately for most covalently bonded systems, SIMU (and ISOR) are only rough approximations to reality. The valued of *s* or *st* may be set to zero to switch off the appropriate restraints.

SIMU is intended for use for larger structures with poorer resolution and data to parameter ratios than are required for full unrestrained anisotropic refinement. It is based on the observation that the U_{ij} values on neighboring atoms in larger molecules tend to be both similar and (when the resolution is poor) significantly correlated with one another. By applying a very weak restraint of this type, we allow a gradual increase and change in direction of the anisotropic displacement parameters as we go out along a side-chain, and we restrain the motion of atoms perpendicular to a planar group (which DELU cannot influence). The use of a distance criterion directly rather than via the connectivity table enables the restraints to be applied automatically to partially overlapping disordered atoms, for which it is an excellent approach. The value of **dmax** can be set so that coordination distances to metal ions etc. are excluded. Terminal atoms tend to show the largest deviations from equal U_{ij}'s and so *st* should be set higher than *s* (or made equal to zero to switch off the restraints altogether). SIMU restraints are NOT recommended for SMALL molecules and ions, especially if free rotation or torsion is possible (e.g. C_5H_5-groups, AsF_6-ions). For larger molecular fragments, the effective rotation angles are smaller, and the assumption of equal U_{ij} for neighboring atoms is more appropriate: both translation and libration of a large fragment will result in relatively similar U_{ij} components on adjacent atoms. SIMU may be combined with ISOR, which applies a further soft but quite different restraint on the U_{ij} components. SIMU may also be used when one or both of the atoms concerned is isotropic, in which case experience indicates that a larger esd (say 0.1 Å^2) is appropriate. The default value of *s* may be changed by a preceding DEFS instruction (*st* is then set to twice *s*).

SIZE dx dy dz

The values **dx, dy** and **dz** are the three principal dimensions of the crystal in **mm**, as usually quoted in publications. This information is written to the *.cif* file. If a SIZE instruction is present in the *.ins* file, SHELXL uses it to write the estimated minimum and maximum transmission to the *.cif* file. This should give order of magnitude estimates that should be replaced by the values from the actual absorption correction if these were applied. The empirical SHELXL estimates take into account that most of the diffraction from strongly absorbing crystals takes place at the edges and corners; these estimates of the actual absorption of the crystal may be a little smaller than those from psi-scan and other semi-empirical routines that include absorption by the mounting fibre and glue or oil.

SPEC del[0.2]

All following atoms (until the next SPEC instruction) are considered to lie on special positions (for the purpose of automatic constraint generation) if they lie within *del* (Å) of a special position. The coordinates of such an atom are also adjusted so that it lies exactly on the special position.

STIR sres step[0.01]

The STIR instruction allows a stepwise improvement in the resolution. In the first refinement cycle, the high-resolution limit (i.e. lowest d) is set at *sres*, in the next cycle to (*sres-step*), in the next (*sres-2*step*) etc. This continues until the limit of the data or the SHEL limit is reached, after which any remaining cycles to complete the number specified by CGLS or L.S. are completed with a constant resolution range. By starting at lower resolution and then gradually improving it, the radius of convergence for models with significant coordinate errors should be increased. This may be regarded as a primitive form of 'simulated annealing'; it could be useful in the early stages of refinement of molecular replacement solutions, or for getting rid of bias for R_{free} tests (in cases where the solution of the structure was - possibly of necessity - based on all the data).

SUMP c sigma c1 m1 c2 m2 ...

The linear restraint: $c = c1*fv(m1) + c2*fv(m2) +$ ... is applied to the specified free variables. This enables more than two atoms to be assigned to a particular site, with the sum of site occupation factors restrained to be a constant. It also enables linear relations to be imposed between distances used on DFIX restraints, for example to restrain a group of atoms to be collinear. sigma is the effective standard deviation. By way of example, assume that a special position on a four-fold axis is occupied by a mixture of Sodium, Calcium, Aluminum and Potassium cations so that the average charge is +2 and the site is fully occupied. The necessary restraints and constraints could be set up as follows (the program will take care of the special position constraints on the coordinates and U_{ij} of course):

```
SUMP 1.0 0.01 1.0 2 1.0 3 1.0 4 1.0 5        ! site fully occupied
SUMP 2.0 0.01 1.0 2 2.0 3 3.0 4 1.0 5         ! mean charge = +2
EXYZ Na1 Ca1 Al1 K1                         ! common x, y and z coordinates
EADP Na1 Ca1 Al1 K1                         ! common U or Uij
FVAR ... 0.20 0.30 0.35 0.15                ! starting values for free variables 2..5
...
Na1 ... ... ... ... 20.25 ... ! 0.25 * fv(2)     [the 0.25 is required for
Ca1 ... ... ... ... 30.25 ... ! 0.25 * fv(3)     a special position on a
Al1 ... ... ... ... 40.25 ... ! 0.25 * fv(4)     four-fold axis, i.e. site
K1 ... ... ... ... 50.25 ... ! 0.25 * fv(5)     symmetry 4]
```

This particular refinement would probably still be rather unstable, but the situation could be improved considerably by adding weak SUMP restraints for the elemental analysis. Such SUMP restraints may be used when elements are distributed over several sites in minerals so that the elemental composition corresponds (within suitable standard deviations) to an experimental chemical analysis.

SUMP may also be applied to BASF, EXTI and BASF parameters, including parameters used to describe twinning (TWIN) and anisotropic scaling (HOPE). The parameters are counted in the order overall scale and free variables, EXTI, then BASF.

SWAT g[0] U[2]

The SWAT option allows two variables g and U to be refined in order to model diffuse solvent using Babinet's principle (Moews & Kretsinger, 1975; the same formula is employed in the program TNT, but the implementation is somewhat different). The calculated intensity is modified as follows:

$$F_c^{\,2}(new) = F_c^{\,2}(old) \times \left[1 - g e^{-8\pi^2 U \left(\sin\theta / \lambda\right)^2}\right]$$

A large value of U ensures that only the low theta $F_c^{\,2}$ values are affected. Subtracting the term in g in this way from the occupied regions of the structure is equivalent to adding a corresponding diffuse scattering term in the (empty) solvent regions in its effect on all calculated $F_c^{\,2}$ values except F(000). For proteins g usually refines to a value between 0.7 and unity, and U usually refines to a value between 2 and 5; for small molecules without significant diffuse solvent regions g should refine to zero. Since g and U are correlated, it is better to start the diffuse solvent refinement by giving SWAT with no parameters; the program will then invent suitable starting values. Note that a different formula was employed in SHELXL-93, and so parameter values from SHELXL-93 may well be unsuitable starting values for the new version.

Since both extinction and diffraction from diffuse solvent tend to affect primarily the strong reflections at low diffraction angle, they tend to show the same symptoms in the analysis of variance, and so a combined warning message is printed. It will however be obvious from the type of structural problem which of the two should be applied. The program does not permit the simultaneous refinement of SWAT and EXTI.

SYMM symmetry operation

Symmetry operators, i.e. coordinates of the general positions as given in International Tables. The operator *x, y, z* is always assumed, so MUST NOT be input. If the structure is centrosymmetric, *the origin MUST lie on a center of symmetry*. Lattice centering and the presence of an inversion center should be indicated by LATT, not SYMM. The symmetry operators may be specified using decimal or fractional numbers, e.g. 0.5-x, 0.5+y, -z or Y-X, -X, Z+1/6; the three components are separated by commas.

TEMP T[20]

The TEMP card sets the temperature T of the data collection in degrees Celsius. This is reported to the *.cif* file and used to set the default isotropic U values for all atoms. TEMP must come before all atoms in the *.ins* file. TEMP also sets the default X-H bond lengths (see AFIX) which depend slightly on the temperature because of librational effects. The default C-H bond lengths and default U-values are rounded to two decimal places so that they may be quoted more easily.

TIME t [#]

If the time t (measured in seconds from the start of the job) is exceeded, SHELXL performs no further least-squares cycles, but goes on to the final structure factor calculation followed by bond lengths, Fourier calculations etc. The default value of t is installation dependent, and is either set to 'infinity' or to a little less than the maximum time allocation for a particular class of job. Usually t is 'CPU time', but some operating systems (e.g. MSDOS) the elapsed time may have to be used instead.

TITL [ ]

Title of up to 76 characters, to appear at suitable places in the output, is on this card. The characters '!' and '=', if present, are part of the title and are not specially interpreted.

TWIN 3x3 matrix [-1 0 0 0 -1 0 0 0 -1] n[2]

The value n is the number of twin components (2 or greater) and the matrix is applied (iteratively if |n| > 2) to generate the indices of the twin components from the input reflection indices, which apply to the first (prime) component.   If a transformation matrix is also given on the HKLF instruction, it is applied first before the (iterative) application of the TWIN matrix. This method of defining twinning allows the standard HKLF 4 format to be used for the *.hkl* file, but can only be used when the reciprocal lattices for all twinned components are metrically superimposable. In other cases HKLF 5 format must be used. The F_o^2 values are fitted to the sum of $k_{m*}F_{cm}^2$ multiplied by the overall scale factor, where k_1 is one minus the sum of k_2, k_3, .. and the starting values for the remaining twin fractions k_2, k_3, .. are specified on a BASF instruction. Only one TWIN instruction is allowed. If BASF is omitted the TWIN factors are all assumed to be equal (i.e. 'perfect' twinning).

If the racemic twinning is present at the same time as normal twinning, n should be doubled (because there are twice as many components as before) and given a negative sign (to indicate to the program that the inversion operator is to be applied multiplicatively with the specified TWIN matrix). The number of BASF parameters, if any, should be increased from m-1 to 2m-1 where m is the original number of components (equal to the new |n| divided by 2). The TWIN matrix is applied m-1 times to generate components 2 ... m from the prime reflection (component 1); components m+1 ... 2m are then generated as the Friedel opposites of components 1 ... m.

UNIT n1 n2 ...

The UNIT card lists the number of atoms of each type in the unit-cell, in SFAC order.

WGHT a[0.1] b[0] c[0] d[0] e[0] f[.33333]

The weighting scheme is defined as follows:

$$w = \xi / \left[\sigma^2 \left(F_o{}^2 \right) + (a \times P^2) + (b \times P) + d + e \sin\theta \right]$$

where P = [**f** * Maximum of (0 or $F_o{}^2$) + (1-**f**) * $F_c{}^2$]. It is possible for the experimental $F_o{}^2$ value to be negative because the background is higher than the peak; such negative values are replaced by 0 to avoid possibly dividing by a very small or even negative number in the expression for **w**. For twinned and powder data, the $F_c{}^2$ value used in the expression for P is the total calculated intensity obtained as a sum over all components. The value **ξ** is 1 when **c** is zero, $exp[\mathbf{c} \times (\sin\theta/\lambda)^2]$ when **c** is positive, and $1 - exp[\mathbf{c} \times (\sin\theta/\lambda)^2]$ when **c** is negative.

The use of P rather than (say) $F_o{}^2$ reduces statistical bias (Wilson 1976). The weighting scheme is NOT refined if **a** is negative (contrast SHELX-76). The parameters can be set by trial and error so that the variance shows no marked systematic trends with the magnitude of $F_c{}^2$ or of resolution; the program suggests a suitable WGHT instruction after the analysis of variance. This scheme is chosen to give a flat analysis of variance in terms of $F_c{}^2$, but does not take the resolution dependence into account. It is usually advisable to retain default weights (WGHT 0.1) until all atoms have been found and the refinement is essentially complete, when the scheme suggested by the program can be used for the next refinement job by replacing the WGHT instruction (if any) by the one output by the program towards the end of the *.res* file. This procedure is adequate for most routine refinements.

It may be desirable to use a scheme which does not give a flat analysis of variance to emphasize particular features in the refinement; for example **c** = +10 or -10 would weight up data at higher 2θ, e.g. to perform a 'high-angle' refinement (uncontaminated by Hydrogen atoms which contribute little at higher diffraction angle) prior to a difference electron density synthesis (FMAP 2) to locate the Hydrogens. The exponential weights which are obtained when **c** is positive were advocated by Dunitz & Seiler (1973). Weighting up the high angle reflections will in general give X-ray atomic coordinates which are closer to those from neutron diffraction.

Refinement against F^2 requires different weights to refinement against F; in particular, making all the weights equal ('unit weights'), although useful in the initial stages of refinement against F, is NEVER a sensible option for F^2. If the program suspects that an unsuitable WGHT instruction has been accidentally retained for a structure which had been refined previously with SHELX-76 or the XLS program in version 4 of the SHELXTL system, it will output a warning message.

WPDB n[1]

Writes the refined coordinates to a *.pdb* file in the Protein Data Base format. If *n* is positive Hydrogen atoms are omitted; if /*n*/ is 1 all atoms are converted to isotropic and ATOM statements generated and if /*n*/ is 2 ANISOU statements are also generated (but the equivalent B value is still used on the ATOM statement). The atom names and residue classes and numbers should conform to PDB conventions. This provides a direct link to X-PLOR and other programs which use (more of less) the official (Brookhaven) dialect of the PDB format. Note that SHELXPRO can be used to extend the PDB output file to include refinement details etc. (from the .lst file) for deposition with the PDB, and also to modify disordered residues so that they can be interpreted by programs such as O that cannot read the full standard PDB format.

ZERR Z esd(a) esd(b) esd(c) esd(α) esd(β) esd(γ)

The **Z** value (number of formula units per cell) followed by the estimated standard deviations in the unit-cell dimensions. The **Z** is only required for the CIF output; the cell *esds* contribute to the estimated *esds* in bond lengths etc. after full-matrix refinement.