

Chapter 5.3.1

SHELXS-97

Structure Solution

George W. Sheldrick
Dept. of Structural Chemistry
Tammannstraße 4
37077 Göttingen, Germany
email: gsheldr@shelx.uni-ac.gwdg.de

SHELXS is primarily designed for the solution of 'small moiety' (1-200 unique atoms) structures from single crystal at atomic resolution, but is also useful for the location of heavy atoms from macromolecular isomorphous or anomalous ΔF data. SHELXS is general and efficient for all space groups in all settings, and there are no arbitrary limits to the size of problems which can be handled, except for the total memory available to the program. Instructions and data are taken from two standard (ASCII) text files, compatible to those used for SHELXL, so that input files can easily be transferred between different computers.

5.3.1.1 Program and file organization

The way of running SHELXS and the conventions for filenames will of course vary for different computers and operating systems, but the following general concept should be adhered to as much as possible. SHELXS may be run on-line by means of the command:

shelxs name

where *name* defines the first component of the filename for all files which correspond to a particular crystal structure. On some systems, *name* may not be longer than 8 characters. On UNIX systems, all filenames (including SHELXS) MUST be given in *lower case*. Batch operation will normally require the use of a short batch file containing the above command etc.

Before starting SHELXS, at least one file - *name.ins* - MUST have been prepared; it contains instructions, crystal and atom data etc. It will usually be necessary to prepare a *name.hkl* file as well which contains the reflection data; the format of this file (3I4,2F8.2) is the same as for all versions of SHELX. This file should be terminated by a record with all items zero. The reflection order is unimportant. This *.hkl* file is read each time the program is run; unlike SHELX-76, there is no facility for intermediate storage of binary data. This enhances computer independence and eliminates several possible sources of confusion. SHELXS requires a single set of input data, and ignores batch numbers, direction cosines or wavelengths if they are present at the end of each record in the *name.hkl* file.

A brief summary of the progress of the structure solution appears on the console (i.e. the standard FORTRAN output), and a full listing is written to a file *name.lst*, which can be printed or examined with a text editor. After structure solution a file *name.res* is written; this contains crystal data etc. as in the *name.ins* file, followed by potential atoms. It may be copied or edited to *name.ins* for structure refinement using SHELXL or partial structure expansion with SHELXS (Chapter 14).

Two mechanisms are provided for interaction with a SHELXS job which is already running. The first, which it is not possible to implement for all computer systems, applies to 'on-line' runs. If the <ctrl-l> key combination is hit, the job terminates almost immediately (but without the loss of output buffers etc. which can happen with <ctrl-C> etc.). If the <Esc> key is hit during direct methods, the program does not generate any further phase permutations but completes the current batch of phase refinement and then proceeds to *E*-Fourier recycling etc. If the <Esc> key is hit during Patterson interpretation, the program stops after completing the calculations for the current superposition vector. Otherwise <Esc> has no effect. On computer consoles with no <Esc> key, <F11> or <Ctrl-[> usually have the same effect.

The second mechanism requires the user to create the file *name.fin*; the program tries at regular intervals to delete this file, and if it succeeds it takes the same action as after <Esc>. The file is also deleted (if found) at the start of a job in case it has been accidentally left over from a previous job. This approach may be used with batch jobs, but may prove difficult to implement on certain systems. The output files are also 'flushed' at regular intervals (if permitted by the operating system) so that they can be examined whilst a batch job is running (if permitted).

The UNIX version of SHELXS is able to read the *.ins* and *.hkl* files in either UNIX or DOS format, and may be compiled under UNIX so as to write the *.res* file in DOS format (see the comments near the start of the program source), so that PC's can access such files via a shared disk without the need for conversion programs such as DOS2UNIX etc. However the compiled programs are supplied with this option switched off, i.e. they write standard UNIX format files. The *.lst* file is always in the local format for reasons of efficiency. The MSDOS program SPRINT supplied with SHELX can print from both MSDOS or UNIX formats.

5.3.1.2 The .ins instruction file

Three types of general calculation may be performed with SHELXS. The structure of the *.ins* file is extremely similar for all three (and the *.hkl* file is always the same). The *.ins* file always begins with the instructions TITL..UNIT in the order given below. There follows TREF (for direct methods), PATT (for Patterson interpretation) or TEXP plus atoms (for partial structure expansion). The final instruction is usually HKLF.

Direct Methods:	Patterson Interp.:	Partial Structure Exp.:
-----	-----	-----
TITL ...	TITL ...	TITL ...
CELL ...	CELL ...	CELL ...
ZERR ...	ZERR ...	ZERR ...
LATT ...	LATT ...	LATT ...
SYMM ...	SYMM ...	SYMM ...
SFAC ...	SFAC ...	SFAC ...
UNIT ...	UNIT ...	UNIT ...
TREF	PATT	TEXP
HKLF	HKLF	atoms
		HKLF

Although these standard settings should be appropriate for a wide range of circumstances, various parameters may be specified for TREF, PATT or TEXP, and further instructions may be included between UNIT and HKLF for 'fine tuning' in the case of difficult structures. The parameter summary printed out after the data reduction in every job should be consulted before this is attempted, since the default settings for parameters that are not specified depend on the space group, the size of the structure, and the parameters that are actually specified (this is sometimes referred to as 'artificial intelligence' !).

All instructions commence with a four (or less) letter word (which may be an atom name); numbers and other information follow in free format, separated by one or more spaces. Upper and lower case input may be freely mixed; with the exception of the text strings input using TITL it is all converted to upper case for internal use in SHELXS. The TITL, CELL, ZERR, LATT, SYMM, SFAC and UNIT instructions must be given in that order; all remaining instructions, atoms, etc. should come between UNIT and the last instruction, which is almost always HKLF (to read in reflection data).

Defaults are given in square brackets in this documentation; '#' indicates that the program will generate a suitable default value based on the rest of the available information. Continuation lines are flagged by '=' at the end of a line, the instruction being continued on the next line which must start with at least one space. Other lines beginning with one or more spaces are treated as comments, so blank lines may be added to improve readability. All characters following '!' or '=' in an instruction line are ignored, except after TITL or SYMM (for which

continuation lines are not allowed). AFIX, RESI and PART instructions may be present in the *.ins* file for compatibility with SHELXL but are ignored.

5.3.1.3 Instructions common to all modes of structure solution

TITL []

Title of up to 76 characters, to appear at suitable places in the output. The characters '!' and '=' may form part of the title. The title could include a chemical formula and/or space group, but one must be careful to update these if the UNIT or SYMM instructions are later changed !

CELL λ a b c α β γ

Wavelength and unit-cell dimensions in Angstroms and degrees.

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

Z value (number of formula units per cell) followed by the estimated errors in the unit-cell dimensions. This information is not actually required by SHELXS but is allowed for compatibility with SHELXL.

LATT N [1]

Lattice type: 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.

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 may NOT be input. If the structure is centrosymmetric, the origin MUST lie on a center of symmetry. Lattice centering 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. At least one SYMM instruction must be present unless the structure is triclinic.

SFAC elements

These element symbols 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 with certain SHELXL instructions but is ignored). The program uses absorption coefficients from International Tables for Crystallography (1991), Volume C. For organic structures the first two SFAC types should be C and H, in that order; the E-Fourier recycling generally assigns the first SFAC type (i.e. C) to peaks.

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

Scattering factor in the form of an exponential series, followed by real and imaginary corrections, linear absorption coefficient, covalent radius and atomic weight. Except for the atomic weight the format is the same as that used in SHELX-76. In addition, a 'label' consisting of up to 4 characters beginning with a letter (e.g. Ca2+) may be included before a1 (the first character may be a '\$', but this is not obligatory). 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. Not all numbers on this instruction are actually used by SHELXS, but the full data must be given for compatibility with SHELXL. For neutron data, c should be the scattering length (which may be negative) and a1..b4 will usually all be zero.

UNIT n1 n2 ...

Number of atoms of each type in the cell, in SFAC order.

REM

Followed by a comment on the same line. This comment is ignored by the program but is copied to the results file (*.res*). Note that comments beginning with one or more blanks are only copied to the *.res* file if the line is completely blank; REM comments are always copied.

MORE verbosity [1]

More sets the amount of (printer) output; verbosity takes a value in the range 0 (least) to 3 (most verbose).

TIME t [#]

If the time t (measured in seconds from the start of the job) is exceeded, SHELXS performs no further blocks of phase permutations (direct methods), but goes on to the final *E*-map recycling etc. In the case of Patterson interpretation, no further vector superpositions are performed after this time has expired. The default value of t is installation dependent, and is usually set to a little less than the maximum time allocation for a particular job class. Usually t is 'CPU time', but on some simpler computer systems (eg. MSDOS) the elapsed time has to be used instead.

OMIT s [4] 2 θ (lim) [180]

Thresholds for flagging reflections as 'unobserved'. Note that if no OMIT instruction is given, ALL reflections are treated as 'observed'. Internally in the program s is halved and applied to F_o^2 , so the test is roughly equivalent to suppressing all reflections with $F_o < s\sigma(F_o)$, as required for consistency with SHELX-76. Note that s may be set to 0 (to suppress reflections with negative F_o^2) or even to a negative threshold (to suppress very negative F_o^2) which has no equivalent in SHELX-76. If 2 θ (lim) is POSITIVE, it specifies a 2 θ value above which the data are treated as 'unobserved'; if it is negative, the absolute value is used as a lower 2 θ cutoff.

OMIT h k l

The reflection *h k l* is flagged as 'unobserved' in the list of merged reflections after data reduction. It will not be used directly in phase refinement or Fourier calculations, but is retained for statistical purposes and as a possible cross-term in a negative quartet. Thus if it is known that a strong reflection has been included accidentally in the *.hkl* file with a very small intensity (e.g. because it was cut off by the beam stop), it is advisable to delete it from the *.hkl* file rather than using OMIT (which is intended for imprecisely measured data rather than blunders).

ESEL Emin [1.2] Emax [5] dU [.005] renorm [.7] axis [0]

Emin sets the minimum *E*-value for the list of largest *E*-values which the program normally retains in memory; it should be set so as to give more than enough reflections for TREF etc. It is also the threshold used for tangent expansion and 'peak-list optimisation'. It is advisable to reduce Emin to about 1.0 for triclinic structures and pseudosymmetry problems. If Emin is negative, acentric triclinic data are generated for use in *all* calculations. The other parameters control the normalisation of the *E*-values:

$$\text{new}(E) = \text{old}(E) \cdot \exp[8dU (\pi \sin\theta/\lambda)^2] / [\text{old}(E)^{-4} + E_{\text{max}}^{-4}]^{0.25}$$

renorm is a factor to control the parity group renormalisation; 0.0 implies no renormalisation, 1.0 sets full renormalisation, i.e. the mean value of E^2 becomes unity for each parity group.

If axis is 1, 2 or 3, an additional similar renormalisation is applied for groups defined by the absolute value of the *h*, *k* or *l* index respectively. If axis is set to zero, no such additional renormalisation is applied.

EGEN d(min) d(max)

All missing reflections in the resolution range *d*(min) to *d*(max) Å (the order of *d*(min) and *d*(max) is unimportant) are generated on a statistical basis, assuming that they were skipped during the data collection because a prescan indicated that they were weak. These reflections will then be flagged as 'unobserved', but improve the estimation of the remaining *E*-values and enable an increased number of negative quartets to be identified. *d*(min) should be safely inside the resolution limit of the data and *d*(max) should be set so that there is no danger of regenerating strong reflections (as weak) which were cut off by the beam stop etc.

LIST m [0]

m = 1 and *m* = 2 write *h*, *k*, *l*, *A* and *B* lists to the *name.res* file, where *A* and *B* are the real and imaginary parts of a point atom structure factor respectively. If *m* = 1 the list corresponds to the phased *E*-values for the 'best' direct methods solution, before partial structure expansion (if any). If *m* = 2 the list is produced after the final cycle of partial structure expansion, and corresponds to weighted *E*-values

used for the final Fourier synthesis. These options enable other Fourier programs to be used, e.g. for graphical display of 3D-Fouriers for data to less than atomic resolution.

After data reduction and merging equivalent reflections, a list of h , k , l , F_o and $\sigma(F_o)$ (for $m = 3$) or h , k , l , F_o^2 and $\sigma(F_o^2)$ (for $m = 4$) is written to the *name.res* file. This provides a useful input file for programs such as DIRDIF and MULTAN, which do not include sort/merge and rejection of systematic absences etc. SHELXS always averages Friedel opposites. In all four cases the output format is (3I4,2F8.2), and the list is terminated by a dummy reflection 0,0,0.

FMAP code [#] axis [#] nl [#]

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 program chooses a 53 x 53 x nl or 103 x 103 x nl grid depending on the resolution of the data, provided sufficient memory is available in the latter case.

code = 1 (F^2 -Patterson), 3 (Patterson with coefficients input using HKLF 7; negative coefficients are allowed), 4 (*E*-map without peak-list optimisation, e.g. because the peaks correspond to unequal atoms), 5 (Fourier with *A* and *B* coefficients input using HKLF 3), 6 (*EF* Patterson), code > 6 (*E*-map followed by [code-6] cycles peak-list optimization). Note that the peak-list optimization assigns very strong peaks to heavy atoms (if specified by SFAC) and all remaining peaks to scattering factor type 1, so for many structures this should be specified as carbon on a SFAC instruction. FMAP 4 may be used with atoms but without TEXP etc. for an *E*-map based on calculated phases.

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

Fourier grid, when not set automatically. Starting points and increments are multiplied by 100. *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. 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 (see above).

PLAN npeaks [#] d1 [0.5] d2 [1.5]

If *npeaks* is positive it is the number of highest unique Fourier peaks which are written to the *.res* and *.lst* files; the remaining parameters are ignored. If *npeaks* is given as negative, the program attempts to arrange the peaks into unique molecules taking the space group symmetry into account, and to 'plot' a projection of each such molecule on the printer (i.e. the *.lst* file). Distances involving peaks which are less than $r_1+r_2+d_1$ (the covalent radii *r* are defined via SFAC; 1 and 2 refer to the two atoms concerned) are considered to be 'bonds' for purposes of the molecule assembly and tables. Distances involving atoms and/or peaks which are less than $r_1+r_2+d_2$ are considered to be 'non-bonded interactions'. Such

interactions are ignored when defining molecules, but the corresponding atoms and distances are included in the line-printer output. Thus an atom may appear in more than one map, or more than once on the same map. Negative d2 includes hydrogen atoms in these non-bonds, otherwise they are ignored (the absolute value of d2 is used in the test). Peaks are always 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.

To simplify interpretation of the lineprinter plots, extra symmetry-generated atoms are added, so that atoms or peaks may appear more than once. A table of the appropriate coordinates and symmetry transformations appears at the end of the output. See also MOLE for forcing molecules (and their environments) to be printed separately.

MOLE n [#]

Forces the following atoms, and atoms or peaks that are bonded to them, into molecule n of the PLAN output. n may not be greater than 99.

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

Before running SHELXS, a reflection data file *name.hkl* must usually be prepared. The HKLF command tells the program which format has been chosen for this file, and allows the indices to be reorientated using a 3x3 matrix r11..r33 (which should have a positive determinant). n is negative if reflection data follow, otherwise they are read from the .hkl file. The data are read in fixed format 3I4,2F8.2 (except for n = 1) 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 checksum). If batch numbers, direction cosines or wavelengths are present in the .hkl file (e.g. for use with SHELXL) they will be ignored. The multiplicative scale s multiplies both F^2 and $\sigma(F^2)$ (or F and $\sigma(F)$ for n = 1 or 3). The multiplicative weight wt multiplies all $1/\sigma^2$ values and m is an integer 'offset' needed to read 'condensed data' (HKLF 1); both are included only for compatibility with SHELX-76. Usually simply 'HKLF 4' is all that will be required.

n = 1: SHELX-76 condensed data. Although now obsolete this format is both ASCII and compact, and contains a checksum, so is sometimes used for network transmission and testing purposes.

n = 3: $h k l F_o \sigma(F_o)$ or $h k l A B$ depending on FMAP setting. In the first case the sign of F_o is ignored (for use with macromolecular ΔF data). This format should NOT be used for routine structure determination purposes because the approximation(s) required for the derivation of F and $\sigma(F_o)$ degrade the quality of the data.

$n = 4: h k l F^2 \sigma(F^2)$. The recommended format for nearly all purposes (for macromolecular isomorphous or anomalous ΔF HKLF 3 is suitable).

$n = 7: h k l E$ or $h k l P$ (Patterson coefficient) depending on FMAP.

There may only be one HKLF instruction and it must come last !

END

This is the last instruction in the rare cases when the .ins file is not terminated by the HKLF instruction.

5.3.1.4 Instructions for writing and reading files for the program PATSEE

SPIN phi1 [0] phi2 [0] phi3 [0]

The following fragment (which should begin with a FRAG instruction) is rotated by the specified angles (in radians). This instruction is used to reinput angles from Patterson search programs (in particular PATSEE).

FRAG code [#] a [1] b [1] c [1] alpha [90] beta [90] gamma [90]

FRAG enables the PATSEE search fragment to be read in using the original cell or orthogonal coordinates. This instruction will usually be preceded by SPIN and MOVE commands to give the rotation angles and translation (same conventions as for PATSEE), and followed by a list of atoms. FRAG, SPIN and MOVE instructions remain in force until superseded by another instruction of the same type. code is ignored by SHELXS but is included for compatibility with PATSEE and SHELXL (where it is used for different purposes).

PSEE m [200] 2 θ (max) [#]

The largest $|m|$ E -values and the complete Patterson map are dumped into the *name.res* file in fixed format for use by Patterson search programs (in particular PATSEE) etc. $2\theta(\max)$ should be used to limit the resolution of the E -values generated; the default value uses $\sin\theta = \lambda/2$. The $2\theta(\max)$ value is also written to the .res file, so it is possible to restrict the resolution of the E -values actually used by PATSEE to a lower $2\theta(\max)$ by editing this file without rerunning SHELXS; of course the E -values with higher 2θ than the value used in SHELXS were not written to the .res file and so cannot be recovered in this way. When m is negative a 'super-sharp' Patterson with coefficients $\sqrt{E^3F}$ is used; if m is positive a standard sharpened Patterson with coefficients (EF) is employed. The resulting *name.res* file must be renamed *name.inp* (or *name.pat* if the search fragment and encoded Patterson are to be read from separate files) for use by PATSEE. After a PSEE instruction, UNIT is followed by the strongest E -values and the full Patterson map in this output file (which may be rather long !).