

25. MACROMOLECULAR CRYSTALLOGRAPHY PROGRAMS

```

evaluate ( $crystal_lattice.space_group = "P2(1)2(1)2(1)" )
evaluate ( $crystal_lattice.unit_cell.a   = 61.76 )
evaluate ( $crystal_lattice.unit_cell.b   = 40.73 )
evaluate ( $crystal_lattice.unit_cell.c   = 26.74 )
evaluate ( $crystal_lattice.unit_cell.alpha = 90 )
evaluate ( $crystal_lattice.unit_cell.beta = 90 )
evaluate ( $crystal_lattice.unit_cell.gamma = 90 )

```

(a)

```

define (
  &crystal_lattice.space_group = P2(1)2(1)2(1) ;
  &crystal_lattice.unit_cell.a  = 61.76 ;
  &crystal_lattice.unit_cell.b  = 40.73 ;
  &crystal_lattice.unit_cell.c  = 26.74 ;
  &crystal_lattice.unit_cell.alpha = 90 ;
  &crystal_lattice.unit_cell.beta  = 90 ;
  &crystal_lattice.unit_cell.gamma = 90 ;
)

```

(b)

Fig. 25.2.3.2. Examples of compound symbols and compound parameters.

(a) The 'evaluate' statement is used to define typed symbols (strings, numbers and logicals). Symbol names are in bold. (b) The 'define' statement is used to define untyped parameters. Each parameter entry is terminated by a semicolon. The compound base name 'crystal_lattice' has a number of sub-levels, such as 'space_group' and the 'unit_cell' parameters. 'unit_cell' is itself base to a number of sub-levels, such as 'a' and 'alpha'. Parameter names are in bold.

```

group type = hl object = pa object = pb
object = pc object = pd end
(25.2.3.9)

```

where 'pa', 'pb', 'pc' and 'pd' refer to the individual arrays. This group statement indicates to CNS that the specified arrays need to be transformed together when reflection indices are changed, e.g. during expansion of the diffraction data to space group *P1*.

25.2.3.3. Symbols and parameters

The CNS language supports two types of data elements which may be used to store and retrieve information. *Symbols* are typed variables, such as numbers, character strings of restricted length and logicals. *Parameters* are untyped data elements of arbitrary length that may contain collections of CNS commands, numbers, strings, or symbols.

Symbols are denoted by a dollar sign (\$), and parameters by an ampersand (&). Symbols and parameters may contain a single data element, or they may be a *compound* data structure of arbitrary complexity. The hierarchy of these data structures is denoted using a period (.). Figs. 25.2.3.2(a) and (b) demonstrate how crystal-lattice information can be stored in compound symbols and parameters, respectively. The information stored in symbols or parameters can be retrieved by simply referring to them within a CNS command: the symbol or parameter name is substituted by its content. Symbol substitution of portions of the compound names (e.g. '&crystal_lattice.unit_cell.\$para') allows one to carry out conditional and iterative operations on data structures, such as matrix multiplication.

25.2.3.4. Statistical functions

The CNS language contains a number of statistical operations, such as binwise averages and summations. The resolution bins are defined by a central facility in CNS.

Fig. 25.2.3.3 shows how σ_A , σ_Δ and D (Read, 1986, 1990) are computed from the observed structure factors ('fobs') and the calculated model structure factors ('fcalc') using the CNS statistical operations. The first five operations are performed for the reflections in the test set, while the last three operations expand the results to all reflections. The 'norm' function computes normalized structure-factor amplitudes for the specified arguments. The 'sigacv' function

```

do (eobs=norm(amplitude(fobs))) ( test_set )
do (ecalc=norm(amplitude(fcalc))) ( test_set )
do (sigmaA=sigacv(eobs,ecalc)) ( test_set )
do (sigmaD=sqrt(save(amplitude(fobs))^2 (1-sigmaA^2))) ( test_set )
do (D= sigmaA * sqrt(save(amplitude(fobs))^2 /
save(amplitude(fcalc)^2))) ( test_set )

do (sigmaA=sum(sigmaA*test) / max(1,sum(test))) ( all )
do (sigmaD=sum(sigmaD*test) / max(1,sum(test))) ( all )
do (D=sum(D*test) / max(1,sum(test))) ( all )

```

Fig. 25.2.3.3. Example for statistical operations provided by the CNS language. 'norm', 'sigacv', 'save' and 'sum' are functions that are computed internally by the CNS program. Binwise operations are in italics ('sigacv', 'save' and 'sum'). The result for a particular bin is stored in all elements belonging to the bin. The σ_A ('sigmaA') parameters are computed in binwise resolution shells. The σ_Δ ('sigmaD') and D parameters are then computed from σ_A and binwise averages involving $|\mathbf{F}_o|^2$ and $|\mathbf{F}_c|^2$. The binwise results are expanded to all reflections by the last three statements. 'test' is an array that is 1 for all reflections in the test set and 0 otherwise. 'sum' is a binwise operation on all reflections with the same partitioning used for the test set.

evaluates σ_A from the normalized structure factors. The 'save' function computes the statistical average

$$\text{save}(f) = \frac{\sum_{hkl} f_{hkl}(w/\varepsilon)}{\sum_{hkl} w}, \quad (25.2.3.10)$$

where w is 1 and 2 for centric and acentric reflections, respectively, and ε is the statistical weight. The averages are computed binwise, and the result for a particular bin is stored in all selected reflections belonging to the bin.

25.2.3.5. Symbolic target function

One of the key innovative features of CNS is the ability to symbolically define target functions and their first derivatives for crystallographic searches and refinement. This allows one conveniently to implement new crystallographic methodologies as they are being developed.

The power of symbolic target functions is illustrated by two examples. In the first example, a target function is defined for simultaneous heavy-atom parameter refinement of three derivatives. The sites for each of the three derivatives can be disjoint or identical, depending on the particular situation. For simplicity, the Blow & Crick (1959) approach is used, although maximum-likelihood targets are also possible (see below). The heavy-atom sites are refined against the target

$$\sum_{hkl} \frac{(|\mathbf{F}_{h_1} + \mathbf{F}_p| - |\mathbf{F}_{ph_1}|)^2}{2v_1} + \frac{(|\mathbf{F}_{h_2} + \mathbf{F}_p| - |\mathbf{F}_{ph_2}|)^2}{2v_2} + \frac{(|\mathbf{F}_{h_3} + \mathbf{F}_p| - |\mathbf{F}_{ph_3}|)^2}{2v_3}. \quad (25.2.3.11)$$

\mathbf{F}_{h_1} , \mathbf{F}_{h_2} and \mathbf{F}_{h_3} are complex structure factors corresponding to the three sets of heavy-atom sites, \mathbf{F}_p represents the structure factors of the native crystal, $|\mathbf{F}_{ph_1}|$, $|\mathbf{F}_{ph_2}|$ and $|\mathbf{F}_{ph_3}|$ are the structure-factor amplitudes of the derivatives, and v_1 , v_2 and v_3 are the variances of the three lack-of-closure expressions. The corresponding target expression and its first derivatives with respect to the calculated structure factors are shown in Fig. 25.2.3.4(a). The derivatives of the target function with respect to each of the three associated structure-factor arrays are specified with the 'dtarget' expressions. The 'tselection' statement specifies the selected subset of reflections to be used in the target function (e.g. excluding outliers), and the 'cvselection' statement specifies a subset of reflections to be used for cross-validation (Brünger, 1992b) (i.e. the subset is not used