1.1. Restraints and constraints

By I. David Brown and Ilia Guzei

Some notes on the philosophy followed in this dictionary
Restraints and constraints are handled in many different ways in different programs. The actual parameters restrained are determined by the way the restraint is expressed within the program, and different programs express the same restraint in different ways. In this dictionary the restraints are expressed in a way that is most natural to the description of the crystal structure which means that the parameters given here may not correspond to the parameters that were actually restrained. For example the rigid body constraint is expressed here by a single item, a flag which identifies which atoms belong to the same rigid body. For a rigid body the target parameters are the same as the refined parameters, allowing the target geometry of the rigid body to be found from the reported atomic coordinates. However, in order to apply this constraint in a refinement, either the coordinates of the atoms in some arbitrary coordinate system, or a sufficient number of geometric parameters of the body, must be specified. Since all these methods preserve the rigid body intact, it is only the result that needs to be reported, not the details of the method used. In this way the description of restraints and constraints has been kept as simple as possible.

A *restraint* is a condition used in the refinement of a crystal structure that requires one or more of the parameters of the refinement to lie within a certain range.

A *constraint* is a condition used in the refinement of a crystal structure that requires one or more parameters of the refinement to have a specific value or be exactly equal to another refined parameter.

The range of values allowed in a restraint is given in this dictionary by a target value and a weighting parameter, the latter being the expectation value of the difference between the refined value and the target. Constraints are indicated by setting the weighting parameter to zero.

The restraints and constraints used are many and varied, and not all can be given in the forms listed in this dictionary. For this reason a general item `_restr_special_details` has been provided so that a text description can be given for a restraint or constraint that cannot be reported in any other way.

This dictionary describes 11 restraints some of which are also constraints (the numbering refers to the order of presentation in the draft dictionary. In this formatted version categories are arranged alphabetically).

1. *Distance*

2. *Angle*

3. *Torsion angle*

4. *Distances restrained to be equal*

5. *Angles restrained to be equal*

6. *Torsion angles restrained to be equal*

7. *Atoms lie on a plane*

8. *Rigid bond*

9. *Minimum approach of two atoms*

10. *Restrained parameter*

11. *Rigid body*

12. *Other restraint or constraint*

Restraint 11. Rigid body
New item in the ATOM_SITE *category which defines rigid bodies.*

ATOM_SITE

`_atom_site_rigid_body_id`   (*char*)
All atoms having the same value of the id are taken as belonging to the same rigid body, and by using different `_*_rigid_body_id` values an unlimited number of rigid bodies can be defined. The coordinates of the atoms in the ATOM_SITE loop belonging to a given rigid body must form a single connected body. Because the body is rigid, the target geometry will be the same as the refined geometry. A program may refine the center of mass and orientation of the rigid body, but these refined parameters do not need to be reported since they can be extracted from the reported atomic coordinates using a user-chosen coordinate system. As this is a constraint, no weighting parameter is needed.
Appears in list containing `_atom_site_label`.   **[atom_site]**

Restraint 12. General description if further restraints

RESTR
This category is for describing restraints that cannot be described elsewhere.

`_restr_special_details`   (*char*)
Text descring any restraint or constraint that cannot be described using any of the the other restraint items.
                                          [restr]

Restraint 2. An angle is restrained to a predetermined value

RESTR_ANGLE
Items in this category define angles that were restrained in the final refinement.

```
loop_
_restr_angle_atom_site_label_1
_restr_angle_site_symmetry_1
_restr_angle_atom_site_label_2
_restr_angle_atom_site_label_3
_restr_angle_site_symmetry_3
_restr_angle_target
_restr_angle_target_weight_param
_restr_angle_diff
_restr_angle_details
C1 1_555  C2  C3 1_555 120  1   -0.3 'generated by JANA'
C2 1_555  C3  C4 3_455 120  1.5  0.5 ?
```

_restr_angle_atom_site_label_1
_restr_angle_atom_site_label_2
_restr_angle_atom_site_label_3   (*char*)
The **_atom_site_labels** used to define the angle. Atom 2 is at the apex of the angle.

Appears in list as essential element of loop structure. **Must** match parent data name
_atom_site_label.   **[restr_angle]**

_restr_angle_details   (*char*)
A free text description of the restraint.

Appears in list containing **_restr_angle_atom_site_label_**.   **[restr_angle]**

_restr_angle_diff   (*numb, su*)
The difference between the target and the refined angle

Appears in list containing **_restr_angle_atom_site_label_**.
The permitted range is $0 \rightarrow \infty$.   **[restr_angle]**

_restr_angle_site_symmetry_1
_restr_angle_site_symmetry_2
_restr_angle_site_symmetry_3   (*char*)
The symmetry transformation needed to generate the coordinates of the three atoms that define the angle. The symmetry code of each atom site as the symmetry-equivalent position number '*n*' and the cell translation number '*klm*'. These numbers are combined to form the code *n_klm*. The character string *n_klm* is composed as follows: *n* refers to a symmetry operation that is applied to the coordinates stored in **_atom_site_fract_x**, **_atom_site_fract_y** and **_atom_site_fract_z**. It should match a number given in **_space_group_symop_id** (formerly **_symmetry_equiv_pos_site_id**). *k*, *l* and *m* refer to the translations that are subsequently applied to the symmetry-transformed coordinates to generate the atom used in calculating the distance. These translations (x, y, z) are related to (k, l, m) by the relations $k = 5 + x$, $l = 5 + y$, $m = 5 + z$. By adding 5 to the translations, the use of negative numbers is avoided.

Appears in list containing **_restr_angle_atom_site_label_**. Where no value is given, the assumed value is '1_555'.

Examples: '.' (no symmetry or translation to site). , '4' (4th symmetry operation applied), '7_645' (7th symmetry position: $+a$ on x, $-b$ on y).   **[restr_angle]**

_restr_angle_target   (*numb*)
The expectation angle defined by the three atoms. This is the target angle for the restrained refinement.

Appears in list containing **_restr_angle_atom_site_label_**.
The permitted range is $0 \rightarrow 180$.   **[restr_angle]**

_restr_angle_target_weight_param   (*numb*)
Weighting parameter = sqrt(1/weight). It is the expectation value of the difference between the refined value and the target. If this parameter is set to zero, the angle will be constrained to refine to the target value. If this item is absent, its value will be taken as zero and the distance will be constrained

Appears in list containing **_restr_angle_atom_site_label_**.
The permitted range is $0 \rightarrow 180$. Where no value is given, the assumed value is '0'.   **[restr_angle]**

Restraint 1. A distance is restrained to a predetermined value.

> ### RESTR_DISTANCE
>
> Category of items that describes restraints applied to distances during the final refinement.
>
> ```
> loop_
>     _restr_distance_atom_site_label_1
>     _restr_distance_atom_site_label_2
>     _restr_distance_site_symmetry_2
>     _restr_distance_target
>     _restr_distance_target_weight_param
>     _restr_distance_diff
>     _restr_distance_details
>   C1   C2  1_555  1.524  0.04 -0.032 'generated by SHELX
> DFIX'
>   C2   C3  1_555  1.340  0.04  0.051 'generated by SHELX
> DFIX'
>   Na1  Ca1 1_555  0.0    0.0   0.0    'Na1 is constrained to
> occupy Ca site'
>   Fe1  Ca1 1_555  0.0    0.0   0.0  ?
>   Al1  Ca1 1_555  0.0    0.0   0.0  ?
> ```

_restr_distance_atom_site_label_1
_restr_distance_atom_site_label_2   (*char*)
The atom site labels of the two atoms defining the distance to be restrained

Appears in list as essential element of loop structure. **Must** match parent data name
_atom_site_label_.   **[restr_distance]**

_restr_distance_details   (*char*)
A free text description of the restraint.

Appears in list containing **_restr_distance_atom_site_label_**.
     [restr_distance]

_restr_distance_diff   (*numb, su*)
The difference between the target and the refined distance

Appears in list containing **_restr_distance_atom_site_label_**.
     [restr_distance]

_restr_distance_site_symmetry_1
_restr_distance_site_symmetry_2   (*char*)
The symmetry transformation needed to generate the coordinates of the two atoms that define the distance. The symmetry code of each atom site as the symmetry-equivalent position number '*n*' and the cell translation number '*klm*'. These numbers are combined to form the code *n_klm*. The character string *n_klm* is composed as follows: *n* refers to a symmetry operation that is applied to the coordinates stored in **_atom_site_fract_x**, **_atom_site_fract_y** and **_atom_site_fract_z**. It should match a number given in **_space_group_symop_id** (formerly **_symmetry_equiv_pos_site_id**). *k*, *l* and *m* refer to the translations that are subsequently applied to the symmetry-transformed coordinates to generate the atom used in calculating the distance. These translations (x, y, z) are related to (k, l, m) by the relations $k = 5 + x$, $l = 5 + y$, $m = 5 + z$. By adding 5 to the translations, the use of negative numbers is avoided.

Appears in list containing **_restr_distance_atom_site_label_**. Where no value is given, the assumed value is '1_55'.

Examples: '.' (no symmetry or translation to site). , '4' (4th symmetry operation applied), '7_645' (7th symmetry position: $+a$ on x, $-b$ on y).   **[restr_distance]**

_restr_distance_target   (*numb*)
The expectation distance between the two atoms. This is the target distance for the restrained refinement.

Appears in list containing **_restr_distance_atom_site_label_**.
     [restr_distance]

_restr_distance_target_weight_param   (*numb*)
Weighting parameter = $\sqrt{(1/weight)}$. It is the expectation value of the difference between the refined value and the target. If this parameter is set to zero, the distance will be constrained to refine to the target value. If this item is absent, its value will be taken as zero and the distance will be constrained

Appears in list containing **_restr_distance_atom_site_label_**. Where no value is given, the assumed value is '0'.   **[restr_distance]**

Restraint 9. Atoms cannot occupy the same position
This 'ant-bumping' restraint prevents two atoms from occupying the same site. It would normally be reported only for those distances in which this restraint was invoked.

RESTR_DISTANCE_MIN

Items in this category offer power law and exponential expressions for a function designed to prevent two atoms occupying the same location.

```
  loop_
_restr_distance_min_atom_site_label_1
_restr_distance_min_site_symmetry_1
_restr_distance_min_atom_site_label_2
_restr_distance_min_site_symmetry_2
_restr_distance_min_A
_restr_distance_min_B
_restr_distance_min_C
_restr_distance_min_E
_restr_distance_min_F
_restr_distance_min_G
_restr_distance_min_distance
_restr_distance_min_details
 O1 1_555 O2 1_555 0 . . 1 2.8 0.3 2.75(1) 'using the
exponential restraint'
 O2 1_555 O3 2_455 0 0 0 1 2.8 0   2.83(1) 'using the hard
sphere model'
```

_restr_distance_min_A *(numb)*
The weight associated with the difference between the refined distance D and the prescribed minimum distance (B or F) is given by the expression:

$$w = A(B/D)^C + E\exp((D-F)/G)$$

A and E are dimensionless weighting parameters. Either function could be used alone by setting A or E to zero. The default values of A and E are zero. If $A = 0$, B and C are undefined, if $E = 0$, F and G are undefined A hard sphere contact can be generated by setting $E = 1$, F =prescribed minimum distance and $G = 0$. In this case G (combined with a non zero E) should be treated as a flag indicating a hard sphere interaction with a target distance of F so as to avoid division by zero.

Appears in list containing **restr_distance_min_atomi_site_label_**. Where no value is given, the assumed value is '0'.　　　　**[restr_distance_min]**

_restr_distance_min_atom_site_label_1
_restr_distance_min_atom_site_label_2 *(char)*
_atom_site_labels for the two atoms that are to be kept apart.

Appears in list as essential element of loop structure. **Must** match parent data name **_atom_site_label**.　　　　**[restr_distance_min]**

_restr_distance_min_B *(numb)*
The weight associated with the difference between the refined distance D and the prescribed minimum distance (B or F) is given by the expression:

$$w = A(B/D)^C + E\exp((D-F)/G)$$

A and E are dimensionless weighting parameters. Either function could be used alone by setting A or E to zero. The default values of A and E are zero. If $A = 0$, B and C are undefined, if $E = 0$, F and G are undefined A hard sphere contact can be generated by setting $E = 1$, F =prescribed minimum distance and $G = 0$. In this case G (combined with a non zero E) should be treated as a flag indicating a hard sphere interaction with a target distance of F so as to avoid division by zero.

Appears in list containing **restr_distance_min_atomi_site_label_**.
The permitted range is $0 \rightarrow \infty$. Where no value is given, the assumed value is '0'.
　　　　[restr_distance_min]

_restr_distance_min_C *(numb)*
The weight associated with the difference between the refined distance D and the prescribed minimum distance (B or F) is given by the expression:

$$w = A(B/D)^C + E\exp((D-F)/G)$$

A and E are dimensionless weighting parameters. Either function could be used alone by setting A or E to zero. The default values of A and E are zero. If $A = 0$, B and C are undefined, if $E = 0$, F and G are undefined A hard sphere contact can be generated by setting $E = 1$, F =prescribed minimum distance and $G = 0$. In this case G (combined with a non zero E) should be treated as a flag indicating a hard sphere interaction with a target distance of F so as to avoid division by zero.

Appears in list containing **restr_distance_min_atomi_site_label_**.
　　　　[restr_distance_min]

_restr_distance_min_detail *(char)*
A text description of the restraint giving details not given elsewhere.

Appears in list containing **restr_distance_min_atomi_site_label_**.
　　　　[restr_distance_min]

_restr_distance_min_difference *(numb)*
The difference in Ångströms between the refined distance of approach of the two atoms and the target distance B or F: difference $= D - B$ or $D - F$.

Appears in list containing **restr_distance_min_atomi_site_label_**.
　　　　[restr_distance_min]

_restr_distance_min_distance *(numb)*
The refined distance, D, between the two atoms.

Appears in list containing **restr_distance_min_atomi_site_label_**.
The permitted range is $0 \rightarrow \infty$.　　　　**[restr_distance_min]**

_restr_distance_min_E *(numb)*
The weight associated with the difference between the refined distance D and the prescribed minimum distance (B or F) is given by the expression:

$$w = A(B/D)^C + E\exp((D-F)/G)$$

A and E are dimensionless weighting parameters. Either function could be used alone by setting A or E to zero. The default values of A and E are zero. If $A = 0$, B and C are undefined, if $E = 0$, F and G are undefined A hard sphere contact can be generated by setting $E = 1$, F =prescribed minimum distance and $G = 0$. In this case G (combined with a non zero E) should be treated as a flag indicating a hard sphere interaction with a target distance of F so as to avoid division by zero.

Appears in list containing **restr_distance_min_atomi_site_label_**. Where no value is given, the assumed value is '0'.　　　　**[restr_distance_min]**

_restr_distance_min_F *(numb)*
The weight associated with the difference between the refined distance D and the prescribed minimum distance (B or F) is given by the expression:

$$w = A(B/D)^C + E\exp((D-F)/G)$$

A and E are dimensionless weighting parameters. Either function could be used alone by setting A or E to zero. The default values of A and E are zero. If $A = 0$, B and C are undefined, if $E = 0$, F and G are undefined A hard sphere contact can be generated by setting $E = 1$, F =prescribed minimum distance and $G = 0$. In this case G (combined with a non zero E) should be treated as a flag indicating a hard sphere interaction with a target distance of F so as to avoid division by zero.

Appears in list containing **restr_distance_min_atomi_site_label_**.
The permitted range is $0 \rightarrow \infty$.　　　　**[restr_distance_min]**

_restr_distance_min_G   (*numb*)

The weight associated with the difference between the refined distance *D* and the prescribed minimum distance (*B* or *F*) is given by the expression:

$$w = A(B/D)^C + E \exp((D - F)/G)$$

A and *E* are dimensionless weighting parameters. Either function could be used alone by setting *A* or *E* to zero. The default values of *A* and *E* are zero. If $A = 0$, *B* and *C* are undefined, if $E = 0$, *F* and *G* are undefined A hard sphere contact can be generated by setting $E = 1$, *F* =prescribed minimum distance and $G = 0$. In this case *G* (combined with a non zero *E*) should be treated as a flag indicating a hard sphere interaction with a target distance of *F* so as to avoid division by zero.

Appears in list containing **restr_distance_min_atomi_site_label_**.

[restr_distance_min]

_restr_distance_min_site_symmetry_1
_restr_distance_min_site_symmetry_2   (*char*)

The site symmetries of the two atoms to be kept apart. The symmetry code of each atom site as the symmetry-equivalent position number '*n*' and the cell translation number '*klm*'. These numbers are combined to form the code *n_klm*. The character string *n_klm* is composed as follows: *n* refers to a symmetry operation that is applied to the coordinates stored in **_atom_site_fract_x**, **_atom_site_fract_y** and **_atom_site_fract_z**. It should match a number given in **_space_group_symop_id** (formerly **_symmetry_equiv_pos_site_id**). *k*, *l* and *m* refer to the translations that are subsequently applied to the symmetry-transformed coordinates to generate the atom used in calculating the distance. These translations (x, y, z) are related to (k, l, m) by the relations $k = 5 + x$, $l = 5 + y$, $m = 5 + z$. By adding 5 to the translations, the use of negative numbers is avoided.

Appears in list containing **restr_distance_min_atomi_site_label_**.

[restr_distance_min]

Restraint 5. Several angles are restrained to be equal
This restraint contains two categories. The first defines the angles to be restrained and assigns them to different classes that are restrained independently. The second category defines the properties of each class.

RESTR_EQUAL_ANGLE

Items in this category list the atoms defining the angles that are restrained to be equal in the final refinement.

```
loop_
_restr_equal_angle_atom_site_label_1
_restr_equal_angle_site_symmetry_1
_restr_equal_angle_atom_site_label_2   # Atom 2 is at the
apex of the angle
_restr_equal_angle_site_symmetry_2
_restr_equal_angle_atom_site_label_3
_restr_equal_angle_site_symmetry_3
_restr_equal_angle_class_id
_restr_equal_angle_details
C1 1_555  C2  1_555  C3 1_555 1  'Benzene ring with mirror
symmetry'
C2 1_555  C3  1_555  C4 2_655 2  ?
C4 1_555  C5  1_555  C6 1_555 2  ?
C5 1_555  C6  1_555  C1 1_555 1  ?
```

_restr_equal_angle_atom_site_label_1
_restr_equal_angle_atom_site_label_2
_restr_equal_angle_atom_site_label_3   (*char*)

The **_atom_site_labels** of the three atoms that define one of the angles to be restrained to be equal to other angles in the same class.

Appears in list as essential element of loop structure. **Must** match parent data name **_atom_site_label**.   **[restr_equal_angle]**

_restr_equal_angle_class_id   (*char*)

A character string identifying the class of equal angles to which this angle belongs.

Appears in list containing **_restr_equal_angle_atom_site_label_**. Where no value is given, the assumed value is '1'.   **[restr_equal_angle]**

_restr_equal_angle_details   (*char*)

A text description giving details of an angle in a class of angles that are restrained to be equal.

Appears in list containing **_restr_equal_angle_atom_site_label_**.

[restr_equal_angle]

_restr_equal_angle_site_symmetry_label_1
_restr_equal_angle_site_symmetry_label_2
_restr_equal_angle_site_symmetry_label_3   (*char*)

The site symmetries of the three atoms that define an angle to be restrained to be equal to other angles in the same class. The symmetry code of each atom site as the symmetry-equivalent position number '*n*' and the cell translation number '*klm*'. These numbers are combined to form the code *n_klm*. The character string *n_klm* is composed as follows: *n* refers to a symmetry operation that is applied to the coordinates stored in **_atom_site_fract_x**, **_atom_site_fract_y** and **_atom_site_fract_z**. It should match a number given in **_space_group_symop_id** (formerly **_symmetry_equiv_pos_site_id**). *k*, *l* and *m* refer to the translations that are subsequently applied to the symmetry-transformed coordinates to generate the atom used in calculating the distance. These translations (x, y, z) are related to (k, l, m) by the relations $k = 5 + x$, $l = 5 + y$, $m = 5 + z$. By adding 5 to the translations, the use of negative numbers is avoided.

Appears in list containing **_restr_equal_angle_atom_site_label_**. Where no value is given, the assumed value is '1_555'.

```
Examples: '.' (no symmetry or translation to site). , '4' (4th
symmetry operation applied), '7_645' (7th symmetry position: +a
on x, −b on y).
```
[restr_equal_angle]

RESTR_EQUAL_ANGLE_CLASS

Items in this category give details of the target angles for each class of angles that are restrained to be equal in the final refinement.

```
loop_
_restr_equal_angle_class_class_id
_restr_equal_angle_class_target_weight_param
_restr_equal_angle_class_average
_restr_equal_angle_class_esd
_restr_equal_angle_class_diff_max
_restr_equal_angle_class_details
1  0.50  123.52  0.32 0.62  ?
2  0.50  118.23  0.52 1.43  ?
```

_restr_equal_angle_class_average   (*numb*)

The average angle in the class of angles restrained to be the same after refinement.

Appears in list.

The permitted range is $0 \rightarrow \infty$.   **[restr_equal_angle_class]**

_restr_equal_angle_class_class_id   (*char*)

A character string that identifies the class of angles whose properties are described.

Appears in list as essential element of loop structure. **Must** match parent data name **_restr_equal_angle_class_id**. Where no value is given, the assumed value is '1'.

[restr_equal_angle_class]

_restr_equal_angle_class_details   (*char*)

A text description giving details of the class of angles that are restrained to be equal.

Appears in list.   **[restr_equal_angle_class]**

_restr_equal_angle_class_diff_max   (*numb, su*)

The maximum deviation of an angle in the class from the class average after refinement.

Appears in list.

The permitted range is $0 \rightarrow \infty$.   **[restr_equal_angle_class]**

_restr_equal_angle_class_esd   (*numb*)

The actual estimated standard deviation of the angles in the class from their average after refinement. This number is expected to be similar to the value set for **_restr_equal_angle_class_target_weight_param**

Appears in list.

The permitted range is $0 \rightarrow \infty$.   **[restr_equal_angle_class]**

_restr_equal_angle_class_target_weight_param
(*numb*)

The weighting parameter $= \sqrt{(1/weight)}$. The expectation value of the estimated standard deviation of the angles in the class from their average after refinement. This value determines the weight assigned to the restraint. If it is zero the angles are constrained to be equal. The default value is zero.

Appears in list.

The permitted range is $0 \rightarrow \infty$. Where no value is given, the assumed value is '0'.

          [restr_equal_angle_class]

Restraint 4. Several distances are restrained to be equal

This restraint contains two categories. The first defines the distances to be restrained and assigns them to different classes that are restrained independently. The second category defines the properties of each class.

RESTR_EQUAL_DISTANCE

Items in this category list the atoms defining the distances that are restrained to be equal in the final refinement.

```
loop_
 _restr_equal_distance_atom_site_label_1
 _restr_equal_distance_site_symmetry_1
 _restr_equal_distance_atom_site_label_2
 _restr_equal_distance_site_symmetry_2
 _restr_equal_distance_class_id
 _restr_equal_distance_details
 C1 1_555  C2  1_555  1  'C1-C2 and C3-C4 are restrained to
be equal'
 C2 1_555  C3  1_555  2  'C2-C3, C4-C5 and C5-C6 are
restrained to be equal'
 C3 1_555  C4  2_655  1  ?
 C4 1_555  C5  1_555  2  ?
 C5 1_555  C6  1_555  2  ?
```

_restr_equal_distance_atom_site_label_1
_restr_equal_distance_atom_site_label_2   (*char*)

The **_atom_site_labels** of the two atoms that define one of the distances to be restrained to be equal to other distances in the same class.

Appears in list as essential element of loop structure. **Must** match parent data name **_atom_site_label**.   **[restr_equal_distance]**

_restr_equal_distance_class_id   (*char*)

A character string identifying the class of equal distances to which this distance belongs.

Appears in list containing **_restr_equal_distance_atom_site_label_**. Where no value is given, the assumed value is '1'.   **[restr_equal_distance]**

_restr_equal_distance_details   (*char*)

A text description giving details of a distance in a class of distances that are restrained to be equal.

Appears in list containing **_restr_equal_distance_atom_site_label_**.

          [restr_equal_distance]

_restr_equal_distance_site_symmetry_label_1
_restr_equal_distance_site_symmetry_label_2
(*char*)

The site symmetries of the two atoms that define a distance to be restrained to be equal to other distances in the same class. The symmetry code of each atom site as the symmetry-equivalent position number '*n*' and the cell translation number '*klm*'. These numbers are combined to form the code *n_klm*. The character string *n_klm* is composed as follows: *n* refers to a symmetry operation that is applied to the coordinates stored in **_atom_site_fract_x**, **_atom_site_fract_y** and **_atom_site_fract_z**. It should match a number given in **_space_group_symop_id** (formerly **_symmetry_equiv_pos_site_id**). *k*, *l* and *m* refer to the translations that are subsequently applied to the symmetry-transformed coordinates to generate the atom used in calculating the distance. These translations (x, y, z) are related to (k, l, m) by the relations $k = 5 + x$, $l = 5 + y$, $m = 5 + z$. By adding 5 to the translations, the use of negative numbers is avoided.

Appears in list containing **_restr_equal_distance_atom_site_label_**. Where no value is given, the assumed value is '1₅55'.

Examples: '**.**' (no symmetry or translation to site). , '4' (4th symmetry operation applied), '7_645' (7th symmetry position: +*a* on *x*, −*b* on *y*).   **[restr_equal_distance]**

RESTR_EQUAL_DISTANCE_CLASS

Items in this category give details of the target distances for each class of distances that are restrained to be equal in the final refinement.

```
loop_
 _restr_equal_distance_class_class_id
 _restr_equal_distance_class_target_weight_param
 _restr_equal_distance_class_average
 _restr_equal_distance_class_esd
 _restr_equal_distance_class_diff_max
 _restr_equal_distance_class_details
      1  0.04   1.534  0.032  0.053    ?
      2  0.04   1.338  0.052  0.103    ?
```

_restr_equal_distance_class_average   (*numb*)

The average distance in the class of distances restrained to be the same after refinement.

Appears in list.

The permitted range is $0 \rightarrow \infty$.   **[restr_equal_distance_class]**

_restr_equal_distance_class_class_id   (*char*)

A character string that identifies the class of distances whose properties are described.

Appears in list as essential element of loop structure. **Must** match parent data name **_restr_equal_distance_class_id**. Where no value is given, the assumed value is '1'.   **[restr_equal_distance_class]**

_restr_equal_distance_class_details   (*char*)

A text description giving details of the class of distances that are restrained to be equal.

Appears in list.   **[restr_equal_distance_class]**

_restr_equal_distance_class_diff_max   (*numb, su*)

The maximum deviation of a distance in the class from the class average after refinement.

Appears in list.

The permitted range is $0 \rightarrow \infty$.   **[restr_equal_distance_class]**

_restr_equal_distance_class_esd   (*numb*)

The actual estimated standard deviation of the distances in the class from their average after refinement. This number is expected to be similar to the value set for **_restr_equal_distance_class_target_weight_param**

Appears in list.

The permitted range is $0 \rightarrow \infty$.   **[restr_equal_distance_class]**

5

_restr_equal_distance_class_target_weight_param
(*numb*)

The weighting parameter = $\sqrt{(1/weight)}$. The expectation value of the estimated standard deviation of the distances in the class from their average after refinement. This value determines the weight assigned to the restraint. If it is zero the distances are constrained to be equal. The default value is zero.

Appears in list.

The permitted range is $0 \rightarrow \infty$. Where no value is given, the assumed value is '0'.

       [restr_equal_distance_class]

Restraint 6. Several torsion angles are restrained to be equal
This restraint contains two categories. The first defines the torsion angles to be restrained and assigns them to different classes that are restrained independently. The second category defines the properties of each class.

RESTR_EQUAL_TORSION

Items in this category list the atoms defining the torsion angles that are restrained to be equal in the final refinement.

```
loop_
_restr_equal_torsion_atom_site_label_1
_restr_equal_torsion_site_symmetry_1
_restr_equal_torsion_atom_site_label_2
_restr_equal_torsion_site_symmetry_2
_restr_equal_torsion_atom_site_label_3
_restr_equal_torsion_site_symmetry_3
_restr_equal_torsion_atom_site_label_4
_restr_equal_torsion_site_symmetry_4
_restr_equal_torsion_class_id
_restr_equal_torsion_details
C1 1_555  C2  1_555  C3 1_555  C4 1_555 1 ?
C5 1_555  C6  1_555  C1 1_555  C2 1_555 1 ?
```

_restr_equal_torsion_atom_site_label_1
_restr_equal_torsion_atom_site_label_2
_restr_equal_torsion_atom_site_label_3
_restr_equal_torsion_atom_site_label_4   (*char*)

The **_atom_site_labels** of the four atoms that define one of the torsion angles to be restrained to be equal to other torsion angles in the same class. The torsion angle is the dihedral angle between the plane defined by atoms 1, 2 and 3, and the plane defined atoms 2, 3 and 4.

Appears in list as essential element of loop structure. **Must** match parent data name **_atom_site_label**.   **[restr_equal_torsion]**

_restr_equal_torsion_class_id   (*char*)

A character string identifying the class of equal torsion angles to which this torsion angle belongs.

Appears in list containing **_restr_equal_torsion_atom_site_label_**. Where no value is given, the assumed value is '1'.   **[restr_equal_torsion]**

_restr_equal_torsion_details   (*char*)

A text description giving details of a torsion angle in a class of torsion angles that are restrained to be equal.

Appears in list containing **_restr_equal_torsion_atom_site_label_**.

       [restr_equal_torsion]

_restr_equal_torsion_site_symmetry_label_1
_restr_equal_torsion_site_symmetry_label_2
_restr_equal_torsion_site_symmetry_label_3
_restr_equal_torsion_site_symmetry_label_4 (*char*)

The site symmetries of the four atoms that define a torsion angle to be restrained to be equal to other torsion angles in the same class. The symmetry code of each atom site as the symmetry-equivalent position number '*n*' and the cell translation number '*klm*'. These numbers are combined to form the code *n_klm*. The character string *n_klm* is composed as follows: *n* refers to a symmetry operation that is applied to the coordinates stored in **_atom_site_fract_x**, **_atom_site_fract_y** and **_atom_site_fract_z**. It should

match a number given in **_space_group_symop_id** (formerly **_symmetry_equiv_pos_site_id**). *k*, *l* and *m* refer to the translations that are subsequently applied to the symmetry-transformed coordinates to generate the atom used in calculating the distance. These translations (x, y, z) are related to (k, l, m) by the relations $k = 5 + x$, $l = 5 + y$, $m = 5 + z$. By adding 5 to the translations, the use of negative numbers is avoided.

Appears in list containing **_restr_equal_torsion_atom_site_label_**. Where no value is given, the assumed value is '1₅55'.

Examples: '.' (no symmetry or translation to site). , '4' (4th symmetry operation applied), '7_645' (7th symmetry position: $+a$ on x, $-b$ on y).   **[restr_equal_torsion]**

RESTR_EQUAL_TORSION_CLASS

Items in this category give details of the target torsion nagles for each class of torsion angles that are restrained to be equal in the final refinement.

```
loop_
_restr_equal_torsion_class_class_id
_restr_equal_torsion_class_target_weight_param
_restr_equal_torsion_class_average
_restr_equal_torsion_class_esd
_restr_equal_torsion_class_diff_max
_restr_equal_torsion_class_details

1  0.50  123.52  0.32  0.62 ?
```

_restr_equal_torsion_class_average   (*numb*)

The average torsion angle in the class of torsion angles restrained to be the same after refinement.

Appears in list.

The permitted range is $0 \rightarrow \infty$.   **[restr_equal_torsion_class]**

_restr_equal_torsion_class_class_id   (*char*)

A character string that identifies the class of torsion angles whose properties are described.

Appears in list as essential element of loop structure. **Must** match parent data name **_restr_equal_torsion_class_id**. Where no value is given, the assumed value is '1'.   **[restr_equal_torsion_class]**

_restr_equal_torsion_class_details   (*char*)

A text description giving details of the class of torsion angles that are restrained to be equal.

Appears in list.   **[restr_equal_torsion_class]**

_restr_equal_torsion_class_diff_max   (*numb, su*)

The maximum deviation of a torsion angle in the class from the class average after refinement.

Appears in list.

The permitted range is $0 \rightarrow \infty$.   **[restr_equal_torsion_class]**

_restr_equal_torsion_class_esd   (*numb*)

The actual estimated standard deviation of the torsion angles in the class from their average after refinement. This number is expected to be similar to the value set for **_restr_equal_torsion_class_target_weight_param**

Appears in list.

The permitted range is $0 \rightarrow \infty$.   **[restr_equal_torsion_class]**

_restr_equal_torsion_class_target_weight_param
(*numb*)

The weighting parameter = $\sqrt{(1/weight)}$. This is the expectation value of the estimated standard deviation (given in **_restr_equal_torsion_class_esd**) of the torsion angles in the class from their average after refinement. This value determines the weight assigned to the target. If it is zero the torsion angles are constrained to be equal. The default value is zero.

Appears in list.

The permitted range is $0 \rightarrow \infty$. Where no value is given, the assumed value is '0'.

       [restr_equal_torsion_class]

Restraint 10. The sum of a given parameter of the specified atoms is restrained

This restraint, which will normally be used to restrain the total occupancy of an atom site, can be used to restrain the value of $\sum$(over the specified atoms)$\{parameter*coefficient\}$ where parameter will usually be the occupancy (but other allowed quantities such as x, y and z will be specified in the enumeration list) and coefficient is a user defined number with a default value of 1.0. Two loops are needed to describe this restraint.

RESTR_PARAMETER

This restraint, which will normally be used to restrain the total occupancy of an atom site, can be used to restrain the value of $\sum$(over the specified atoms)$\{parameter*coefficient\}$ where *parameter* will usually be the value of the occupancy (but other allowed quantities such as *x*, *y* and *z* can be specified in the enumeration) and *coefficient* is a user defined number with a default value of 1.0. The restraint requires two loops, the first defines the atoms in each sum (class) and the second describes the properties of the class.

These examples should be read in conjunction with the examples given for the loop restr_parameter_class.

In Class 1 the sites O1, O1a and O1b refer to separate sites over which oxygen is disordered, but which in total contain 0.8 oxygen atoms (as defined by **_restr_parameter_class_target).**

```
loop_
 _restr_parameter_id
 _restr_parameter_class_id
 _restr_parameter_atom_site_label
 _restr_parameter_atom_coefficient
1 1 O1    1
2 1 O1a   1
3 1 O1b   1
```

Class 2 consists of a mixture of Na, K, Ca and Al atoms on the same site with the total occupancy set to 1.0 as defined in **_restr_parameter_class_target.**

```
loop_
 _restr_parameter_id
 _restr_parameter_class_id
 _restr_parameter_atom_site_label
 _restr_parameter_atom_coefficient
a 2 Na1   1
b 2 K1    1
c 2 Ca1   1
d 2 Al1   1
```

Class 3 consists of the same atoms as Class 2, but by using coefficients equal to the ionic charge, the total charge on the site is restrained to 2.0. Note that the parameter restrained is still the occupancy but the use of coefficients transforms the restraint from occupancy to formal charge.

```
loop_
 _restr_parameter_id
 _restr_parameter_class_id
 _restr_parameter_atom_site_label
 _restr_parameter_atom_coefficient
1 3 Na1   1
2 3 K1    1
3 3 Ca1   2
4 3 Al1   3
```

In Class 4 the y coordinate of O2 is restrained to be close to a pseudo-mirror plane at y = 0.5 (see **_restr_parameter_class_target).** *This example may not have much practical use, but is included to show what can be done with this definition.*

```
loop_
 _restr_parameter_id
 _restr_parameter_class_id
 _restr_parameter_atom_site_label
 _restr_parameter_atom_coefficient
1 4 O2    1
```

In Class 5 the positions of O3 and O4 are correlated in such a way that these atoms are displaced equal distances from the plane x = 0 (i.e., x(O3)+x(O4) = 0)

```
loop_
 _restr_parameter_id
 _restr_parameter_class_id
 _restr_parameter_atom_site_label
 _restr_parameter_atom_coefficient
1 5 O3    1
2 5 O4    1
```

_restr_parameter_atom_coefficient   (*numb*)

A parameter that scales the qantity being restrained. It can be used to convert a constraint on occupance to a constraint on ionic charge.

Appears in list containing **_restr_parameter_id**. Where no value is given, the assumed value is '1'.   **[restr_parameter]**

_restr_parameter_atom_site_label   (*char*)

The **_atom_site_label** for an atom in this class.

Appears in list containing **_restr_parameter_id**. **Must** match parent data name **_atom_site_label**.   **[restr_parameter]**

_restr_parameter_class_id   (*char*)

The identifier of the class of restraint applied to the atoms

Appears in list containing **_restr_parameter_id**. Where no value is given, the assumed value is '1'.   **[restr_parameter]**

_restr_parameter_id   (*char*)

A unique identifer for each line in a loop

Appears in list as essential element of loop structure.   **[restr_parameter]**

RESTR_PARAMETER_CLASS

This restraint, which will normally be used to restrain the total occupancy of an atom site, can be used to restrain the value of SUM(over the specified atoms)PARAMETER*COEFFICIENT where PARAMETER will usually be the value of the occupancy (but other allowed quantities such as x, y and z can be specified in the enumeration) and COEFFICIENT is a user defined number with a default value of 1.0. The restraint requires two loops, the first defines the atoms in each sum (class) and the second describes the properties of the class.

This example gives the properties of the five classes given as examples in restr$_p$arameter.

*The **_*_class_id** is the **_list_reference** and is the same as **_*_class_id** in the restr$_p$arameterloop*

*The **_*_parameter_type** indicates the atomic parameter that is restrained. **_*_target** is the target value for the sum of the product of **_restr_parameter_atom_coeffcient** and the value of the **_*_parameter_type** for the atom. **_*_target_weight_sum** gives the expectation value of the difference between the target and the refined value of sum(coefficient*parameter) **_*_details** gives a description of the restraint applied*

```
loop_
_restr_parameter_class_class_id
_restr_parameter_class_parameter_type
_restr_parameter_class_target
_restr_parameter_class_target_weight_param
 _restr_parameter_class_details
1 occupancy   0.8   0.01    'total occupation is 0.8'
2 occupancy   1.0   0.001   'total occupation is 1.0'
3 occupancy   2.0   0.01    'total charge is 2.0'
4 position_y  0.5   0.002   'keep close to pseudo-mirror
plane'
5 position_x  0     0.01    'correlate position of O3 and O4'
```


_restr_parameter_class_class_id   (*char*)

The class id of the restraint described in this category

Appears in list as essential element of loop structure. **Must** match parent data name **_restr_parameter_class_id**. Where no value is given, the assumed value is '1'.   **[restr_parameter_class]**

_restr_parameter_class_details   (*char*)

A text description of the restraint.

Appears in list containing **_restr_parameter_class_class_id**.   **[restr_parameter_class]**

_restr_parameter_class_parameter_type   (*char*)

A flag that indicates the nature of the value that is being restrained.

Appears in list containing **_restr_parameter_class_class_id**.

The data value must be one of the following:

   occupancy **atom_site_occupancy**
   position_x **_atom_site_fract_x**
   position_y **_atom_site_fract_y**
   position_z **_atom_site_fract_z**

       [restr_parameter_class]

_restr_parameter_class_target   (*numb*)

The target value for the sum of values of the appropriate parameter type multiplied by the **_*_target_weight_param**.

Appears in list containing **_restr_parameter_class_class_id**.   **[restr_parameter_class]**

_restr_parameter_class_weight_param   (*numb*)

Weighting parameter $= \sqrt{(1/weight)}$. It is the expectation value of the difference between the refined value of the $\sum(parameter*coefficient)$ and the **_*_target**. If this parameter is set to zero, the sum will be constrained to refine to the target value. If this item is absent, its value will be taken and zero and the distance will be constrained

Appears in list containing **_restr_parameter_class_class_id**. Where no value is given, the assumed value is '0'.   **[restr_parameter_class]**

Restraint 7. A group of atoms is restrained to lie on a plane
This restraint contains two categories. The first defines the atoms that define each plane, each plane being composed of atoms from the same class. The second category describes the properties of each plane (class).

RESTR_PLANE

Items in this category define the atoms that make up each class of plane.

```
loop_
_restr_plane_id
_restr_plane_atom_site_label
_restr_plane_site_symmetry
_restr_plane_class_id
_restr_plane_target_weight_param
_restr_plane_displacement
_restr_plane_details
1 c1 1_555 1 0.02  0.002(1) 'c1 to c4 lie on one plane'
2 c2 1_555 1 0.02 -0.003(2) ?
3 c3 1_555 1 0.02 -0.002(1) ?
4 c4 1_555 1 0.02  0.002(2) ?
5 c1 2_655 2 0.003  0.004(1) 'c1, c5, c6 and c7 lie on one
plane'
6 c5 1_555 2 0.003 -0.002(2) ?
7 c6 1_555 2 0.003  0.002(3) ?
8 c7 1_555 2 0.003 -0.002(2) ?
```

_restr_plane_atom_site_label   (*char*)

An **_atom_site_label** of one of the atoms that form the plane

Appears in list as essential element of loop structure. containing **?**. **Must** match parent data name **_atom_site_label**.   **[restr_plane]**

_restr_plane_class_id   (*char*)

A character string that identifies the plane to which this atom is constrained.

Appears in list as essential element of loop structure. containing **?**. Where no value is given, the assumed value is '1'.   **[restr_plane]**

_restr_plane_details   (*char*)

A text string giving details not described elsewhere.

Appears in list containing **?**.   **[restr_plane]**

_restr_plane_displacement   (*numb, su*)

The distance between this atom and the best plane through all the atoms.

Appears in list containing **?**.

The permitted range is $0 \rightarrow \infty$.   **[restr_plane]**

_restr_plane_id   (*char*)

A unique identifier for each line in the list.

Appears in list as essential element of loop structure. Where no value is given, the assumed value is '1'.   **[restr_plane]**

_restr_plane_site_symmetry   (*char*)

The site symmetries of the atom that helps to define the plane to which it to be restrained. The symmetry code of each atom site as the symmetry-equivalent position number '*n*' and the cell translation number '*klm*'. These numbers are combined to form the code *n_klm*. The character string *n_klm* is composed as follows: *n* refers to a symmetry operation that is applied to the coordinates stored in **_atom_site_fract_x**, **_atom_site_fract_y** and **_atom_site_fract_z**. It should match a number given in **_space_group_symop_id** (formerly **_symmetry_equiv_pos_site_id**). *k*, *l* and *m* refer to the translations that are subsequently applied to the symmetry-transformed coordinates to generate the atom used in calculating the distance. These translations (x, y, z) are related to (k, l, m) by the relations $k = 5 + x$, $l = 5 + y$, $m = 5 + z$. By adding 5 to the translations, the use of negative numbers is avoided.

Appears in list as essential element of loop structure. containing **?**. Where no value is given, the assumed value is '1_555'.

Examples: '.' (no symmetry or translation to site). , '4' (4th symmetry operation applied), '7_645' (7th symmetry position: +*a* on *x*, −*b* on *y*).   **[restr_plane]**

_restr_plane_target_weight_param   (*numb*)

The weighting parameter $= \sqrt{(1/weight)}$. The expectation value of the distance in Ångströms between this atom and the best plane through all the atoms of the class.

Appears in list.

The permitted range is $0 \rightarrow \infty$. Where no value is given, the assumed value is '0.'.   **[restr_plane]**

RESTR_PLANE_CLASS

Items in thiscategory describe the properties of the different groups of atoms that are restrained to form a plain.

```
loop_
_restr_plane_class_class_id
_restr_plane_class_displacement_esd
_restr_plane_class_displacement_max_atom_site_label
_restr_plane_class_displacement_max_site_symmetry
_restr_plane_class_displacement_max
_resrt_plane_class_details

1  0.032  c2 1_555  0.094    'displacements for plane 1'
2  0.0021 c1 2_655  0.010    'displacements for plane 2'
```

_restr_plane_class_class_id   (*char*)

The class identifier for the plane whose properties are described

Appears in list as essential element of loop structure. **Must** match parent data name **_restr_plane_class_id**. Where no value is given, the assumed value is '1'.   **[restr_plane_class]**

_restr_plane_class_details   (*char*)

Text describing any special features of the restraint.

Appears in list containing **_restr_plane_class_class_id**.   **[restr_plane_class]**

_restr_plane_class_displacement_esd   (*char*)

The observed estimated standard deviation of the atoms from this plane.

Appears in list containing **_restr_plane_class_class_id**.

The permitted range is $0 \rightarrow \infty$.   **[restr_plane_class]**

_restr_plane_class_displacement_max   (*numb, su*)

The distance in Ångström from the plane to the atom furthest removed from the plane.

Appears in list containing **_restr_plane_class_class_id**.

The permitted range is $0 \rightarrow \infty$.   **[restr_plane_class]**

_restr_plane_class_displacement_max_atom_site_label
(*char*)

The **_atom_site_label** of the atom that lies furthest from the plane defined by this class

Appears in list containing **_restr_plane_class_class_id**. **Must** match parent data name **_atom_site_label**.   **[restr_plane_class]**

_restr_plane_class_displacement_max_site_symmetry
(*char*)

The site symmetries of the atom that helps to define the plane to which it to be restrained. The symmetry code of each atom site as the symmetry-equivalent position number '*n*' and the cell translation number '*klm*'. These numbers are combined to form the code *n_klm*. The character string *n_klm* is composed as follows: *n* refers to a symmetry operation that is applied to the coordinates stored in **_atom_site_fract_x**, **_atom_site_fract_y** and **_atom_site_fract_z**. It should match a number given in **_space_group_symop_id** (formerly **_symmetry_equiv_pos_site_id**). *k*, *l* and *m* refer to the translations that are subsequently applied to the symmetry-transformed coordinates to generate the atom used in calculating the distance. These translations (x, y, z) are related to (k, l, m) by the relations $k = 5 + x$, $l = 5 + y$, $m = 5 + z$. By adding 5 to the translations, the use of negative numbers is avoided.

Appears in list containing **_restr_plane_class_class_id**. Where no value is given, the assumed value is '1_555'.

Examples: '.' (no symmetry or translation to site). , '4' (4th symmetry operation applied), '7_645' (7th symmetry position: +*a* on *x*, −*b* on *y*).   **[restr_plane_class]**

Restraint 3. A torsion angle is constrained to a predetermined value

---

RESTR_TORSION

Items in this category define torsion angles that were restrained in the final refinement.

An example of a torsion angle restrained to 90+/-1 degree with a refined difference of 0.97 degrees.

```
_restr_torsion_atom_site_label_1
   _restr_torsion_site_symmetry_1
   _restr_torsion_atom_site_label_2
   _restr_torsion_site_symmetry_2
   _restr_torsion_atom_site_label_3
   _restr_torsion_site_symmetry_3
   _restr_torsion_atom_site_label_4
   _restr_torsion_site_symmetry_4
   _restr_torsion_angle
   _restr_torsion_weight_param
   _restr_torsion_diff
   _restr_torsion_details
 Na1 1_555  Na1 2_555  O1 2_555  H101 1_555  90  1  0.97  ?
```

_restr_torsion_angle_target   (*numb*)

The angle in degrees to which the torsion angle is restrained. The torsion angle is the dihedral angle between the plane defined by atoms 1, 2 and 3, and the plane defined by atoms 2, 3 and 4.

Appears in list containing **_restr_torsion_atom_site_label_**.

The permitted range is $-180 \rightarrow 180$.   **[restr_torsion]**

_restr_torsion_atom_site_label_1
_restr_torsion_atom_site_label_2
_restr_torsion_atom_site_label_3
_restr_torsion_atom_site_label_4   (*char*)

The **_atom_site_labels** of the atoms in the sequence in which they are linked by the bonds whose torsion angle is to be restrained.

Appears in list as essential element of loop structure. **Must** match parent data name **_atom_site_label_**.   **[restr_torsion]**

_restr_torsion_details   (*numb*)

A free text description of the restraint.

Appears in list containing **_restr_torsion_atom_site_label_**.   **[restr_torsion]**

_restr_torsion_diff   (*numb, su*)

The difference between the target and the refined torsion angle.

Appears in list containing **_restr_torsion_atom_site_label_**.   **[restr_torsion]**

_restr_torsion_site_symmetry_1
_restr_torsion_site_symmetry_2
_restr_torsion_site_symmetry_3
_restr_torsion_site_symmetry_4   (*char*)

The symmetry transformation needed to generate the coordinates of the four atoms that define the torsion angle. The symmetry code of each atom site as the symmetry-equivalent position number 'n' and the cell translation number 'klm'. These numbers are combined to form the code n_klm. The character string n_klm is composed as follows: n refers to a symmetry operation that is applied to the coordinates stored in **_atom_site_fract_x**, **_atom_site_fract_y** and **_atom_site_fract_z**. It should match a number given in **_space_group_symop_id** (formerly **_symmetry_equiv_pos_site_id**). k, l and m refer to the translations that are subsequently applied to the symmetry-transformed coordinates to generate the atom used in calculating the distance. These translations (x, y, z) are related to (k, l, m) by the relations $k = 5 + x$, $l = 5 + y$, $m = 5 + z$. By adding 5 to the translations, the use of negative numbers is avoided.

Appears in list containing **_restr_torsion_atom_site_label_**. Where no value is given, the assumed value is '1_555'.

Examples: '.' (no symmetry or translation to site)., '4' (4th symmetry operation applied), '7_645' (7th symmetry position: $+a$ on x, $-b$ on y).   **[restr_torsion]**

_restr_torsion_weight_param   (*numb*)

Weighting parameter = $\sqrt{(1/weight)}$. It is the expectation value of the difference between the refined value and the target. If this parameter is set to zero, the angle will be constrained to refine to the target value. If this item is absent, its value will be taken as zero and the distance will be constrained

Appears in list containing **_restr_torsion_atom_site_label_**.

The permitted range is $0 \rightarrow \infty$. Where no value is given, the assumed value is '0'.   **[restr_torsion]**

Restraint 8. ADPs are restrained for a rigid bond

This 'rigid bond' restraint restrains the anisotropic displacement parameters of two atoms so that they are equal within a certain **_weight_param** *along the direction of the vector joining the atoms.*

┌───┐

RESTR_U_RIGID

The items in this category restrains the anisotropic displacement parameters of two atoms to be equal within a certain **_weight_param** along the direction of the vector joining the atoms.

```
loop_
_restr_U_rigid_atom_site_label_1
_restr_U_rigid_site_symmetry_1
_restr_U_rigid_atom_site_label_2
_restr_U_rigid_site_symmetry_2
_restr_U_rigid_target_weight_param
_restr_U_rigid_U_parallel
_restr_U_rigid_diff
_restr_U_rigid_details
C1 1_555 C2 2_655 0.001 0.0023(2) 0.0006 'C1-C2 is a rigid
bond'
```

└───┘

_restr_U_rigid_atom_site_label_1
_restr_U_rigid_atom_site_label_2   (*char*)

The **_atom_site_labels** of the two atoms that define the rigid bond.

Appears in list as essential element of loop structure. **Must** match parent data name **_atom_site_label**.   **[restr_U_rigid]**

_restr_U_rigid_details   (*char*)

Free text information about the rigid bond restraint.

Appears in list containing **restr_U_rigid_atom_site_label_**.   **[restr_U_rigid]**

_restr_U_rigid_diff   (*numb, su*)

The difference between the components along the bond direction of the atomic displacement parameters, U, of the two atoms.

Appears in list containing **restr_U_rigid_atom_site_label_**.   **[restr_U_rigid]**

_restr_U_rigid_site_symmetry_1
_restr_U_rigid_site_symmetry_2   (*char*)

The site symmetries of the two atoms that define the rigid bond. The symmetry code of each atom site as the symmetry-equivalent position number 'n' and the cell translation number 'klm'. These numbers are combined to form the code n_klm. The character string n_klm is composed as follows: n refers to a symmetry operation that is applied to the coordinates stored in **_atom_site_fract_x**, **_atom_site_fract_y** and **_atom_site_fract_z**. It should match a number given in **_space_group_symop_id** (formerly **_symmetry_equiv_pos_site_id**). k, l and m refer to the translations that are subsequently applied to the symmetry-transformed coordinates to generate the atom used in calculating the distance. These translations (x, y, z) are related to (k, l, m) by the relations $k = 5 + x$, $l = 5 + y$, $m = 5 + z$. By adding 5 to the translations, the use of negative numbers is avoided.

Appears in list containing **restr_U_rigid_atom_site_label_**. Where no value is given, the assumed value is '1_555'.

Examples: '.' (no symmetry or translation to site)., '4' (4th symmetry operation applied), '7_645' (7th symmetry position: $+a$ on x, $-b$ on y).   **[restr_U_rigid]**

_restr_U_rigid_target_weight_param   (*numb*)

The weighting parameter = $\sqrt{(1/weight)}$. The expectation value of the difference between the components of the atomic displacement parameter, U, along the bond direction of the two atoms that define the bond. This number is used to assign a weight during refinement. A value of zero causes the restraint to become a constraint. This item as a default value of zero.

Appears in list containing **restr_U_rigid_atom_site_label_**. Where no value is given, the assumed value is '0'.   **[restr_U_rigid]**

_restr_U_rigid_U_parallel   (*numb*)

The average value of the components parallel to the bond of the atomic displacement parameters of the two atoms that define the bond.

Appears in list containing **restr_U_rigid_atom_site_label_**. Where no value is given, the assumed value is '0'.   **[restr_U_rigid]**