TOPAS-Academic, Version 9

Modulated Structures

October 1, 2026

Alan A. Coelho

working with Claude Opus 5.5, the AI model from Anthropic

www.topas-academic.com

1.... #INP_FROM_CIF - INGESTING CIF FILE

[#inp_from_cif $file]	**Example:** INP_FROM_CIF.INP

#inp_from_cif is a pre-processor command that in-place translates a CIF to INP format. Here's a before/after scenario:

```
XDD(ceo2)
   CuKa2(0.001)
   bkg @ 0 0
   #inp_from_cif ceo2.cif
      scale @ 0.001
      CS_L(100)
```

```
XDD(ceo2)
   CuKa2(0.001)
   bkg @  38.8156042` -18.0674089`
   str
      phase_name "CeO2"
      a 5.410278
      b 5.410278
      c 5.410278
      volume  158.365
      space_group "F_M_3_M"
      site Ce1 x 0     y 0    z 0    occ Ce+4  1 beq 0.200
      site O1  x 0.25  y 0.25 z 0.25 occ O-2   1 beq 0.43
      scale @  0.00571627568`
      CS_L(100)
```

move_to_site $site. A new keyword that moves scope of the loader to a site.

2...MODULATED STRUCTURES

[str]...
  [apply_rotation_matrix # # # # # # # # #] ' not modulation specific
  [p1_hkls] ' not modulation specific
  [mod_d_tolerance !E]
  [mod_qm $name]...
    [mod_qx E] [mod_qy E] [mod_qz E]
    [mod_m #]
    [mod_user_operators $operators]
    [mod_tau_from_cif $file]
    [mod_tau_symbol | space_group_ssg_name_IT $symbol]
    [mod_tau !E]...
    [mod_t0 E]
    [load_q_vector_type $type]
  [site $name]...
    [mod_x !E E E [qm !E]]... [mod_y !E E E [qm !E]]... [mod_z !E E E [qm !E]]...
    [mod_occ !E E E [qm !E]]...
    [mod_beq !E E E [qm !E]]...
    [mod_u11 !E E E [qm !E]]... to [mod_u23 !E E E [qm !E]]...
    [mod_mlx !E E E [qm !E]]... [mod_mly !E E E [qm !E]]... [mod_mlz !E E E [qm !E]]...
    [mod_crenel_width E [qm !E]]...
      [mod_crenel_center E]
      [mod_saw_x E] [mod_saw_y E] [mod_saw_z E]
      [mod_legendre_x !E E]... [mod_legendre_y !E E]... [mod_legendre_z !E E]...

macros

Get_site(m, s)	Copies site s from the str that is in the scope of prm m
Get_sites(m)	Copies sites from the str that is in the scope of prm m
Get_modulated_sites(m)	Copies modulated sites that is in the scope of prm m
Get_mod_s(x)	See text
Get_mod_c(x)	See text

Examples TEST_EXAMPLES\MOD\ : Incorrect derivatives often show up in poor convergence. Examples below tests convergence for a range of parameters; in all cases convergence is rapid.

Single crystal

ORG\MOD.INP	Wagner & Schönleber (2009) fitting to order 4 to a published structure using the observed data.
CE2O2MNSE2\MOD.INP CE2O2MNSE2\MANUAL-TWIN.INP	Wang *et al.* (2015) An incommensurate twin using apply_rotation_matrix, a crenel with Legendre displacements, extinction, and satellites to fourth order.

Powder

3-ARM-CREN.INP	Three modulation vectors, two crenels with sawtooths and a harmonic modulation vector; refines twelve parameters including all three wave vectors.
MSCIF-OUT.INP	writing the refined structure as msCIF.
Constant wavelength neutron	
OCC-MERGE.INP	Demonstrates that occ_merge derivatives are correct for modulated sites with mly, mod_mly, mod_x type parameters.
RIGID-LP-MOD_X.INP	A rigid body defined only in the m=0 str, with the satellite str reaching two of its sites through Get_modulated_sites. The satellite scale is deliberately unphysical, so the rigid body refines from satellite intensities. P21/c, and Mn1 and O1 are both in the rigid body and modulated.
EXAMPLE-1.INP	Moment modulation and a crenel together.
REFINE-ALL.INP	mod_u11..., mod_x..., mod_occ, mlx..., mg, u11.... 30 parameters
MLX-MOD_OCC-CRENEL.INP	mod_occ, mod_crenel
MLX-MOD_X-MOD_MLY.INP	mod_x, mod_mly
MLX-MOD_X-MOD_MLY-CRENEL.INP	mod_x, mlx, mod_mly, crenel
CRENEL-LEGENDRE.INP	mod_crenel_width , mod_crenel_center, mod_legendre_x, mod_saw_x. A crenel with a Legendre displacement, and the sawtooth it reduces to at order one.
MOD-MLX-CONSTRAINTS.INP	add_constraints on mod_mlx/y/z. A centrosymmetric site, where every displacive amplitude is forbidden and the moment is not.
Time-of-flight neutron	
TOF-MOD-1.INP	Satellites on a TOF axis, with the reflection lists.

2.1 .....Introduction

Modulated structures have been implemented in TOPAS with the assistance of Claude, Anthropic's AI assistant. It has contributed to the understanding of the mathematics and to implementation including the writing of INP files from CIF files.

A modulated structure is not periodic on the basic cell; the departure is a periodic function of position, and that function has its own period which in general is incommensurate with the cell. Reflections are indexed on the basic cell plus integer multiples of the modulation wave (MW) vectors:

$$\mathbf{Q} = \mathbf{H} + \sum_{a} m_a \, \mathbf{q}_a$$

where ***H*** = h a* + k b* + l c* is the Miller index of the basic structure, $\boldsymbol{q}_a$ is the a'th modulation wave vector and m_a is the corresponding satellite order. Reflections with all m_a=0 are the main reflections; the others are satellites. Modulated structures can comprise:

> rigid bodies

> and occ_merge
> and twins

Both commensurate and incommensurate modulations are described in the same way. A modulation wave vector with rational components repeats after a whole number of basic cells and could equally be described by a supercell, whereas one with irrational components never repeats; neither the input nor the calculation distinguishes the two.

van Smaalen (2007) gives a general account of modulated structures and of the superspace approach. Crenel and sawtooth functions, and their implementation in Jana2006, are described by Petříček et al. (2016). Wang et al. (2015) is a worked example of an incommensurately modulated structure refined from single crystal data.

2.2 .....Structure factors for modulated structures

Nomenclature	
MW	Modulation wave vector
MMW	Multiple modulation wave vectors
H	**H** = h **a*** + k **b*** + l **c***
q	the modulation wavevector, mod_qx, mod_qy, mod_qz, with components q_x, q_y, q_z
m	the satellite order, mod_m
n	the modulation harmonic
Q	**H** + m **q**, the satellite vector ($h + m\ q_x$, $k + m\ q_y$, $l + m\ q_z$)
$\mathbf{x}_{s,e,0}$	the unmodulated fractional coordinate of the atom of site s at equivalent position e, normalised into the unit cell. It comes from the position given by the site keyword in the INP file, as $\mathbf{R}_e\ \mathbf{x}_s + \mathbf{t}_e$. The 0 marks the unmodulated value, the actual position being $\mathbf{x}_{s,e,0} + \mathbf{u}_{s,e}$
$\mathbf{u}_{s,e}(v_{s,e})$	the modulation displacement function
$v_{s,e}$	$\mathbf{q} \cdot \mathbf{x}_{s,e,0}$, the modulation phase
D_m	the m^{th} Fourier coefficient of the phase factor the displacement produces
$\mathbf{C}_m$	the moment coefficient at order m, the convolution of the **P** with the D
$\mathbf{R}_e$	the space group rotation matrix
$\mathbf{t}_e$	the translation part of the equivalent position
f_s	the atomic scattering factor of site s
$f_{M,s}$	the magnetic scattering factor of site s
mod_x$_{sin,a}$, mod_x$_{cos,a}$ mod_x$_{c,a}$	the isin and icos of mod_x, and the same for mod_y and mod_z
mlx	the static moment along x, with mly and mlz likewise

mod_mlx$_{sin,a}$, mod_mlx$_{cos,a}$	the isin and icos of mod_mlx, and the same for mod_mly and mod_mlz
τ	mod_tau is the phase shift a symmetry operator applies to the modulation wave.

What follows is written for one modulation wave (MW) vector , subscript *a*. The structure factor for one atom at site *s* of equivalent position *e* displaced sinusoidally along one direction, at modulation phase $v_{s,e,a}$ is:

$$F(hklm) = \sum_{s,e} f_s \ \exp\left[2\pi i\,(\mathbf{H} + m_a\mathbf{q}_a)\cdot\left(\mathbf{x}_{s,e,0} + \mathbf{u}_{s,e,a}(v_{s,e,a})\right)\right]$$

Rearranging we get:

$$F(hklm) = \sum_{s,e} f_s \ e^{2\pi i\,\mathbf{H}\cdot\mathbf{r}_{s,e,0}}\, e^{2\pi i\, m_a v_{s,e,a}}\, e^{2\pi i\,\mathbf{H}\cdot\mathbf{u}_{s,e,a}(v_{s,e,a})}$$

Assuming the atom moves sinusoidally we get:

$$\mathbf{u}_{s,e,a}(v_{s,e,a}) = \mathbf{A}_a \ \sin(2\pi v_{s,e,a})$$

Using the Jacobi–Anger identity we get:

$$e^{i\,(\mathbf{H}\cdot\mathbf{A}_a)\sin(2\pi v_{s,e,a})} = \sum_{n=-\infty}^{\infty} J_n\,(\mathbf{H}\cdot\mathbf{A}_a)\, e^{i2\pi n v_{s,e,a}}$$

The final form of *F*(*hklm*) is:

$$F(hklm) = \sum_{s,e} f_s \ e^{i2\pi\,\mathbf{H}\cdot\mathbf{r}_{s,e,0}} \sum_{n=-\infty}^{\infty} J_n\,(\mathbf{H}\cdot\mathbf{A}_a)\, e^{i2\pi(m_a+n)v_{s,e,a}}$$

2.3 .....The space group operator, the moment, and the full F(hklm)

A real structure adds equivalent positions of the space group, a magnetic moment, and a modulation of that moment.

2.3.1..... The space group operator

An atom of site *s* at equivalent position *e* does not sit at $\mathbf{x}_{s,e,0}$, where $\mathbf{x}_{s,e,0}$ is the fractional atomic coordinates defined using the site keyword. Instead $\mathbf{x}_{s,e,0}$ is displaced by an amount that depends on where it sits along the modulation wave. Throughout, this chapter, subscript *s* alone marks a number as entered in the INP file, and the pair of subscripts *s*,*e* marks that site at that equivalent position. The modulation phase $v_{s,e}$, sometimes called the internal coordinate, is the phase of the modulation wave at the atom's basic position:

$v_{s,e,a} = \mathbf{q}_a \cdot \mathbf{x}_{s,e,0}$

$v_{s,a} = \mathbf{q}_a \cdot \mathbf{x}_s$

Where the subscript a corresponds to a modulation wave a. $v_{s,e,a}$ runs from 0 to 1 over one period of the modulation, every atom has one, and two atoms at different positions sit at different points on the same wave. An atom has one $v_{s,e,a}$ per modulation wave vector. Each equivalent position ($\mathbf{R}_e$, $\mathbf{t}_e$) of the space group puts a copy of site s at $\mathbf{R}_e\,\mathbf{x}_s + \mathbf{t}_e$. The superspace operator carries the modulation phase across at the same time, reversing it or leaving it alone and shifting it by $\tau_{e,a}$. That shift does not follow from $\mathbf{R}_e$ and $\mathbf{t}_e$; it is what mod_tau$_e$ supplies:

$$\begin{aligned}
\mathbf{x}_{s,e,0} &= \mathbf{R}_e\,\mathbf{x}_s + \mathbf{t}_e \\
v_{s,e,a} &= \mathbf{q}_a \cdot (\mathbf{R}_e\,\mathbf{x}_s + \mathbf{t}_e) \\
&= \mathbf{q}_a \cdot \mathbf{R}_e\,\mathbf{x}_s + \mathbf{q}_a \cdot \mathbf{t}_e \\
&= \mathbf{G}_{e,a} \cdot \mathbf{x}_s + \varepsilon_{e,a}\,v_s + \mathbf{q}_a \cdot \mathbf{t}_e
\end{aligned}$$

where,

$$\begin{aligned}
\mathbf{q}_a \cdot \mathbf{R}_e\,\mathbf{x}_s &= \mathbf{G}_{e,a} \cdot \mathbf{x}_s + \varepsilon_{e,a}\,v_s \\
\mathbf{q}_a\,\mathbf{R}_e &= \varepsilon_{e,a}\,\mathbf{q}_a + \mathbf{G}_{e,a}, \quad \varepsilon_{e,a} = \pm 1
\end{aligned}$$

$\mathbf{G}_{e,a}$ is a reciprocal lattice vector of integers and is zero unless a component of $\mathbf{q}_a$ is rational, so for an ordinary incommensurate $\mathbf{q}_a$ only $\varepsilon_{e,a}$ remains.

$\varepsilon_{e,a}$ reverses the modulation wave or leaves it alone, $\tau_{e,a}$ is the phase shift the operator applies to it, and $\mathbf{G}_{e,a}$ couples the modulation phase to the position. $\varepsilon_{e,a}$ and $\mathbf{G}_{e,a}$ follow from what the operator does to $\mathbf{q}_a$, which returns to $\pm\mathbf{q}_a$ only to within a whole reciprocal lattice vector:

Three symbols involved are:

> x_4 is the fourth axis. Every modulation function repeats with period 1 along it.

> $v_{s,e}$ is a value on x_4 where the atom of site s at equivalent position e sits. In four dimensions an atom is a line running along x_4 and the structure is a three-dimensional slice through it; $v_{s,e}$ is where the slice cuts that line.

> $v_{s,e}$ is the symmetry operator equivalent.

The modulation phase of an equivalent position is $\mathbf{q}_a \cdot \mathbf{x}_{s,e,0}$, where $\mathbf{x}_{s,e,0}$ is the position normalised back into the unit cell. $\tau_{e,a}$ is added on top of it and is not a restatement of $\mathbf{q}_a \cdot \mathbf{t}_e$. For the C centring operators of Cmma at $\mathbf{q}_a$ = (0.158, 0, 1/2), $\mathbf{q}_a \cdot \mathbf{t}_e$ is 0.079 whilst $\tau_{e,a}$ is 0 or 1/2.

Each equivalent position contributes one term to the structure factor. $\tau_{e,a}$ reaches it as a phase $2\pi\, m_a\, \varepsilon_{e,a}\, \tau_{e,a}$, and therefore multiplies that term by $\exp(2\pi i\, m_a\, \varepsilon_{e,a}\, \tau_{e,a})$. With $\tau_{e,a}$ of 1/2 the factor becomes $(-1)^m$, so the odd order satellites change sign whilst the even orders and the main reflections do not.

The phase carries the parent index $\mathbf{H}$ rather than the satellite vector $\mathbf{Q} = \mathbf{H} + \sum_a m_a\,\mathbf{q}_a$, because the $\sum_a m_a\,\mathbf{q}_a \cdot \mathbf{x}$ part cancels against the atom's own modulation phase:

$$\varphi_{s,e} \;=\; 2\pi \left[\mathbf{H} \cdot (\mathbf{R}_e\, \mathbf{x}_s \;+\; \mathbf{t}_e) \;+\; \sum_a m_a \left(\mathbf{G}_{e,a} \cdot \mathbf{x}_s \;+\; \varepsilon_{e,a}\, \tau_{e,a} \right) \right]$$

$\mathbf{R}_e\, \mathbf{x}_s + \mathbf{t}_e$ is normalised back into the unit cell, giving each atom one modulation phase $v_{s,e,a}$. Normalising matters because a lattice translation changes the phase of the modulation wave. $\mathbf{R}_e$ acts on the amplitudes; $\mathbf{t}_e$ does not.

2.3.2 ..... The displacive part, mod_x

A site may carry several harmonics. Each harmonic *n* has a sine and a cosine amplitude, and each is a vector with three components:

$$\mathbf{s}_{n,s,a} \;=\; \left(\mathrm{mod_x}_{sin,n,a}, \mathrm{mod_y}_{sin,n,a}, \mathrm{mod_z}_{sin,n,a} \right)$$
$$\mathbf{c}_{n,s,a} \;=\; \left(\mathrm{mod_x}_{cos,n,a}, \mathrm{mod_y}_{cos,n,a}, \mathrm{mod_z}_{cos,n,a} \right)$$

The displacement of an atom is the sum over all harmonics:

$$\mathbf{u}_{s,e,a}\left(v_{s,e,a}\right) \;=\; \sum_n \left[\mathbf{s}_{n,s,a}\; sin\; 2\pi n v_{s,e,a} \;+\; \mathbf{c}_{n,s,a}\; cos\; 2\pi n v_{s,e,a} \right]$$

D is defined by an integral over one modulation period, with *Q* the satellite vector of that same order:

$$D_{m,s,e,a} \;=\; \int_0^1 \exp\left(2\pi i \left(\mathbf{Q} \;\cdot\; \mathbf{u}_{s,e,a} \;+\; m\, v_{s,e,a} \right) \;-\; \frac{\mathrm{mod_beq}_a\left(v_{s,e,a}\right)}{4\, d^2} \right) dv_{s,e,a}$$

beq and u11 … u23 are constant in *v* and come out of the integral, multiplying the structure factor as usual. mod_beq and mod_u11 … mod_u23 stay inside and multiply the integrand, so they reach *D* as a convolution with their own coefficients rather than as a factor on *D*. The displacive term is imaginary and moves the phase; the mod_beq term is real and changes the magnitude. The rest of this section evaluates the integral in closed form.

Harmonic *n* contributes a sinusoid to $\mathbf{Q}\cdot\mathbf{u}_i$; u_n is the square of its amplitude:

$$w_{n,s,e,a} \;=\; \left(\mathbf{Q} \cdot \left(\mathbf{R}_e\, \mathbf{s}_{n,s,a} \right) \right)^2 \;+\; \left(\mathbf{Q} \cdot \left(\mathbf{R}_e\, \mathbf{c}_{n,s,a} \right) \right)^2$$

The Jacobi–Anger sum above then gives one series per harmonic,

$$D^{(n)}_{m,s,e,a} \;=\; G\left(|m|, w_{n,s,e,a}\right) \left(\mathbf{Q} \cdot \left(\mathbf{R}_e\, \mathbf{s}_{n,s,a} \right) \;+\; i\, \mathbf{Q} \cdot \left(\mathbf{R}_e\, \mathbf{c}_{n,s,a} \right) \right)^m \quad m \;\geq\; 0$$
$$D^{(n)}_{m,s,e,a} \;=\; (-1)^{|m|}\, G\left(|m|, w_{n,s,e,a}\right) \left(\mathbf{Q} \cdot \left(\mathbf{R}_e\, \mathbf{s}_{n,s,a} \right) \;-\; i\, \mathbf{Q} \cdot \left(\mathbf{R}_e\, \mathbf{c}_{n,s,a} \right) \right)^{|m|} \quad m \;<\; 0$$

where

$$G(m, w) \;=\; \frac{J_m\left(\sqrt{w}\right)}{\left(\sqrt{w}\right)^m}$$

The isin and the icos are kept together as one complex amplitude, and the phase of that complex number is what distinguishes a sine modulation from a cosine one. With one amplitude zero, D reduces to the Jn(**H**·**A**) above.

Harmonic series *n* at term *k* lands on order-*m* = *nk*. With more than one harmonic present their contributions add, so a reflection at order-*m* collects every combination of terms for which ∑ *nk* = *m*, the sum running over the harmonic series present. $D_{m,s,e,a}$ is therefore the convolution of the per harmonic series. The harmonics are folded in one at a time, which is also how the calculation proceeds: with $D_{m,s,e,a}$ the series for the harmonics so far, adding harmonic *n* replaces it by

$$D_{m,s,e,a} = \sum_k D_{m-n\,k,s,e,a}\, D^{(n)}_{k,s,e,a}$$

starting from $D_0 = 1$ with every other order-m zero. One harmonic at $n = 1$ leaves $D_m = D_m^{(1)}$, which is the case the sections above describe, and the superscript can be dropped. Two harmonics at $n = 1$ and $n = 2$ give, for the third satellite, $D_3^{(1)}D_0^{(2)} + D_1^{(1)}D_1^{(2)} + D_{-1}^{(1)}D_2^{(2)}$ and so on; every pair of terms whose weighted total is 3.

There is one *D* per order-m, and its subscript is whichever order-m is wanted; *m* for the reflection itself, *m* – *p* inside the convolution below. A harmonic series has a term list of its own, written D with a superscript, and that one is indexed by term rather than by order-m. Each site at each equivalent position carries its own series, and one with no modulation keeps that seed, so it reaches the main reflections only.

2.3.3 ..... The moment, mlx and mod_mlx

The moment enters the magnetic structure factor, not the nuclear one, and it is linear in its amplitudes; there is no Bessel function.

$$\mathbf{m}_{0,s} = (\mathrm{mlx}, \mathrm{mly}, \mathrm{mlz})$$
$$\mathbf{m}_{sin,n,s,a} = \left(\mathrm{mod_mlx}_{sin,n,a}, \mathrm{mod_mly}_{sin,n,a}, \mathrm{mod_mlz}_{sin,n,a}\right)$$
$$\mathbf{m}_{cos,n,s,a} = \left(\mathrm{mod_mlx}_{cos,n,a}, \mathrm{mod_mly}_{cos,n,a}, \mathrm{mod_mlz}_{cos,n,a}\right)$$

so the moment on the site is

$$\mathbf{m}_{s,e}(v_{s,e,a}) = \mathbf{m}_{0,s} + \sum_a \sum_n \left[\mathbf{m}_{sin,n,s,a}\, sin\, 2\pi n v_{s,e,a} + \mathbf{m}_{cos,n,s,a}\, cos\, 2\pi n v_{s,e,a}\right]$$

The rotated moment has one coefficient at order 0 and a pair at ±n for every harmonic:

$$\mathbf{P}_{0,s,e} = det(\mathbf{R}_e)\, \mathbf{R}_e\, \mathbf{m}_{0,s}$$
$$\mathbf{P}_{+n,s,e,a} = det(\mathbf{R}_e)\, \mathbf{R}_e\, \frac{\left(\mathbf{m}_{cos,n,s,a} - i\, \mathbf{m}_{sin,n,s,a}\right)}{2}$$
$$\mathbf{P}_{-n,s,e,a} = det(\mathbf{R}_e)\, \mathbf{R}_e\, \frac{\left(\mathbf{m}_{cos,n,s,a} + i\, \mathbf{m}_{sin,n,s,a}\right)}{2}$$

C is the same integral with the moment inside it:

$$\mathbf{C}_{m,s,e,a} = \int_0^1 \mathbf{m}_{s,e,a} \exp\left(2\pi i \left(\mathbf{Q} \cdot \mathbf{u}_{s,e,a} + m\, v_{s,e,a}\right) - \frac{\mathrm{mod_beq}_a\left(v_{s,e,a}\right)}{4\, d^2}\right) dv_{s,e,a}$$

and because the atom carrying the moment is itself displaced, the moment coefficient at order *m* is the convolution of the two series:

$$\mathbf{C}_{m,s,e,a} = \sum_p \mathbf{P}_{p,s,e,a}\, D_{m-p,s,e,a}$$

If the site has no displacive modulation then *D* reduces to $D_0 = 1$, the convolution collapses, and a moment modulated at harmonic *n* reaches orders 0 and ±*n* only, whereas a displacive modulation at the same harmonic reaches every multiple of *n*. That difference is the quickest way to tell the two apart in a pattern. When both are present the convolution spreads the moment across the displacive orders as well.

2.3.4 ..... The full F(hklm)

Collecting the operator, the displacive series and the moment, and summing over the atoms *i* and the equivalent positions (**R**, *t*), the nuclear structure factor is

$$F(hklm) = \sum_s f_s \sum_e e^{i\varphi} \prod_a D_{m,s,e,a}$$

and the magnetic structure factor, a vector with one component per axis, is

$$\mathbf{M}(hklm) = \sum_s f_{M,s} \sum_e e^{i\varphi} \sum_a \mathbf{C}_{m,s,e,a} \prod_{b \neq a} D_{m,s,e,b}$$

At $m = 0$ the satellite vector is **H**, D_0 is the Bessel reduction $J_0(\sqrt{u})$, C_0 is the static moment, and the expression collapses to the ordinary structure factor.

Two conventions decide signs rather than magnitudes. The satellite is taken as the coefficient of $\exp(-2\pi i\, m\, v_i)$, following msCIF, so the order used to index *D* is $-m$. And a reflection kept as its Friedel partner carries a modulation sign of −1, which multiplies *m* in **Q**, in the phase, and in the order indexing *D*; all three, or a row is evaluated at one satellite for its position and another for its coefficient.

2.3.5 ..... Real and imaginary parts

Both D_m and C_m are complex:

$$D_{m,s,e} = D_{re,s,e} + i\, D_{im,s,e} \qquad \mathbf{C}_{m,s,e} = \mathbf{C}_{re,s,e} + i\, \mathbf{C}_{im,s,e}$$

and with the phase φ of the first section, $\exp(i\varphi) = \cos\varphi + i \sin\varphi$ gives

$$F_{re}(hklm) = \sum_s f_s \sum_e \left(D_{re,s,e}\cos\varphi - D_{im,s,e}\sin\varphi\right)$$

$$F_{im}(hklm) = \sum_s f_s \sum_e \left(D_{re,s,e}\sin\varphi + D_{im,s,e}\cos\varphi\right)$$

$$\mathbf{M}_{re}(hklm) = \sum_s f_{M,s} \sum_e \left(\mathbf{C}_{re,s,e}\cos\varphi - \mathbf{C}_{im,s,e}\sin\varphi\right)$$

$$\mathbf{M}_{im}(hklm) = \sum_s f_{M,s} \sum_e \left(\mathbf{C}_{re,s,e}\sin\varphi + \mathbf{C}_{im,s,e}\cos\varphi\right)$$

The intensity is as follows:

$$|F(hklm)|^2 = {F_{re}}^2 + {F_{im}}^2$$

Only the component of **M** perpendicular to **Q** scatters:

$$\mathbf{M}_\perp = \mathbf{M} - (\mathbf{M}\cdot\mathbf{Q})\,\mathbf{Q}\,/\,|\mathbf{Q}|^2$$

The nuclear and magnetic parts do not interfere for unpolarised neutrons, so the intensity is:

$$I(hklm) = |F(hklm)|^2 + |\mathbf{M}_\perp|^2$$

2.4 .....Generating the reflections of a modulated structure

A satellite is indexed by *h*, *k*, *l* and an order m_a for each modulation wave vector:

$$\mathbf{Q} = \mathbf{H} + \sum_a m_a\,\mathbf{q}_a\,, \quad \mathbf{H} = h\,\mathbf{a}^* + k\,\mathbf{b}^* + l\,\mathbf{c}^*$$

Reflections are merged (grouped) when **Q** and structure factors are the same; the latter is a function of $\mathbf{Q}\,\mathbf{R}_e$:

$$\mathbf{Q}\,\mathbf{R}_e = \left(\mathbf{H} + \sum_a m_a\,\mathbf{q}_a\right)\mathbf{R}_e = \mathbf{H}\,\mathbf{R}_e + \sum_a m_a\,(\mathbf{q}_a\,\mathbf{R}_e)$$

Everything depends on $\mathbf{q}_a\,\mathbf{R}_e$, which is $\varepsilon_{e,a}\,\mathbf{q}_a + \mathbf{G}_{e,a}$, so:

$$\mathbf{Q}\,\mathbf{R}_e = \left(\mathbf{H}\,\mathbf{R}_e + \sum_a m_a\,\mathbf{G}_{e,a}\right) + \sum_a \varepsilon_{e,a}\,m_a\,\mathbf{q}_a$$

The partner of (**H**, m_a) is the satellite with parent index $\mathbf{H}\,\mathbf{R}_e + m_a\,\mathbf{G}_{e,a}$ and order $\varepsilon_{e,a}\,m_a$; it is not ($\mathbf{H}\,\mathbf{R}_e$, m_a). $\mathbf{G}_{e,a}$ fixes the partner index and $\varepsilon_{e,a}$ fixes its order.

With more than one modulation wave vector the single order *m* becomes a set, one m_a for each modulation wave (MW), and **Q** runs over every combination of them. The hkl generator steps each m_a through the range requested for its MW and forms a reflection for each combination, so the number of satellites grows as the product of the per-MW ranges rather than with a single order. Each MW

keeps its own $\varepsilon_{e,a}$, $\mathbf{G}_{e,a}$ and $\tau_{e,a}$, so merging and the partner index follow the rule above applied MW by MW; a coupled operator differs only in sending one MW onto another before its shift and sign are applied.

2.4.1..... Defining mod_tau

A symmetry operator applies a phase shift of mod_tau$_e$ to the modulation wave; one value per equivalent position. It comes from the superspace group and is not refined. mod_tau$_e$ is not determined by the three-dimensional space group alone, though the possible series can be derived. Several superspace groups share the same three-dimensional group and differ only in these shifts; the trailing 0s0 of Cmme(a,0,1/2)0s0 is what identifies the mod_tau$_e$ sequence.

Measuring the modulation phase from an origin shifted by c, changes τ_e by c(1 − ε_e). At ε_e = +1 that factor is zero, so τ_e is the same whatever origin you measure the modulation phase from; they are a property of the superspace group rather than of the setting. At ε_e = −1 the factor is 2c, so those τ_e can be moved, and since there is one origin for the whole group, only one of them can be moved to zero. In other words, if every ε_e, other than for the identity $\mathbf{R}_e$ = I, is −1, every τ_e is fixed by the choice of origin rather than by the structure. That is the group of order two, P-1 with a general **q** being the case.

In summary $\mathbf{R}_e$, ε_e and $\mathbf{G}_e$ decide which reflections are equivalent, and τ_e adds a phase. Absences arise when τ_e changes the phases of the equivalent positions so that *F* is zero.

2.4.2..... Space group operators from the INP file

mod_user_operators reads the superspace operators from a string. Each operator comprises four algebraic parts, x1,x2,x3,x4, of which the first three are the ordinary space group operator; the constant part of the fourth is mod_tau. When a space_group is also defined, a check is made such that the operators of space_group match those of mod_user_operators; the result is identical to mod_tau_from_cif (see below). When no space_group is given the operators define the equivalent positions themselves and a space_group keyword is not needed. mod_user_operators are a dependent of mod_qm, and therefore the modulation vector can carry its own operators. Equivalent positions however are str dependent and therefore an exception is thrown if two mod_qm's of a particular str have different equivalent positions. mod_user_operators cannot be used with mod_tau or mod_tau_build_all. mod_tau_symbol used alongside mod_user_operators is ignored. Example usage:

```
mod_user_operators {
    x1,x2,x3,x4
    x1,-x2,-x3,x3+x4+1/2
    -x1+1/2,x2,-x3,-x4+1/2
    -x1+1/2,-x2,x3,-x3-x4
    -x1,-x2,-x3,-x4
    ...
}
```

2.4.3 ..... mod_tau from a CIF file

[mod_tau_from_cif $file] loads mod_tau from a CIF file. The superspace operators and the intrinsic translation of each operator is read, matched to the corresponding operator of the space group, and then written to mod_tau. Like mod_user_operators, mod_tau_from_cif can be defined without a space_group. mod_tau_from_cif suits the case where a modulated structure has been solved elsewhere and its CIF is at hand. If the file cannot be read, or an operator has no counterpart in the space group, the run stops and asks for mod_tau to be entered directly. See CE2O2MNSE2\MOD.INP, where we have:

```
   mod_tau_from_cif "ce2o2mnse2.cif"   ' reads the operators and sets mod_tau
```

2.4.4 ..... Deriving mod_tau

mod_tau can be used to explicitly write τ, one for each equivalent position of the space group *.sg file. mod_tau_symbol on the other hand, accepts full or part symbols, for example:

```
   mod_tau_symbol Cmme(a,0,1/2)0s0
   mod_tau_symbol 0s0
```

If there's only the trailing part and there's more than one possible series, then the possibilities are written to the OUT file as a grouped #list, and no refinement iterations are performed. Note: mod_tau_symbol cannot accept symbols such α; letters are therefore used where a, b and c correspond to alpha, beta and gamma; the examples use g for gamma, as in (0,0,g).

mod_tau_symbol "" writes every allowed series to the output file as a grouped #list. A pair is one superspace group in two settings of the internal origin a quarter apart; the two are identical on the operators with ε = +1 and differ by 1/2 on those with ε = −1. The comment on each line carries the group number and the setting. For Cmme(a,0,1/2) the sixteen series are eight groups in two settings each:

```
#list Mod_Tau_Series {
   { 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 }   ' Group 1 a
   { 0 0.5 0.5 0.5 0 0 0.5 0 0 0.5 0.5 0.5 0 0 0.5 0 }   ' Group 1 b
   { 0 0.5 0 0 0.5 0.5 0.5 0 0 0.5 0 0 0.5 0.5 0.5 0 }   ' Group 2 a
   { 0 0 0.5 0.5 0.5 0.5 0 0 0 0 0.5 0.5 0.5 0.5 0 0 }   ' Group 2 b
   ...
   { 0 0.5 0.5 0 0.5 0 0 0.5 0.5 0 0 0.5 0 0.5 0.5 0 }   ' Group 8 a
   { 0 0 0 0.5 0.5 0 0.5 0.5 0.5 0.5 0.5 0 0 0.5 0 0 }   ' Group 8 b
   }
num_runs #list_n Mod_Tau_Series
load mod_tau { Mod_Tau_Series(Run_Number) }
```

num_runs, #list_n, Mod_Tau_Series refines each series in turn.

mod_t0 can be used to refine the origin of the internal coordinate. The origin is free within a superspace group, and a correct series can be written with more than one origin; the pairs in the list above with the same Group number, belong to the same group at two origins a quarter apart. With the

amplitudes fixed and mod_t0 refined, the origin is rapidly found. For the Ce2O2MnSe2 example, the two degenerate settings are:

```
load mod_tau { 0 0 0.5 0.5 0.5 0.5 0 0 0 0 0.5 0.5 0.5 0.5 0 0 }
load mod_tau { 0 0.5 0 0 0.5 0.5 0.5 0 0 0.5 0 0 0.5 0.5 0.5 0 }
```

These agree on every operator whose ε is +1 and differ by a half on those whose ε is −1. Refining mod_t0 results in identical fits. These two possible mod_tau series are degenerate only if mod_t0 can be varied such that the structure is the same. What mod_t0 cannot do is change the symmetry; it cannot turn one superspace group into another. On a tetragonal P4(0,0,g) test carrying a quarter intrinsic translation, the correct series refined to Rwp = 0.000, while an all-zero series stuck at Rwp = 97.8. mod_t0 therefore distinguished a series that was merely self-consistent from one that was crystallographically correct; it was central to checking the derivation across the trigonal, tetragonal and hexagonal groups.

space_group_ssg_name_IT is a synonym of mod_tau_symbol; the name matches msCIF. A symbol on its own defaults to the standard centring and carries no translations along the internal coordinate. space_group_ssg_name_IT does not normally include a shift. In TOPAS therefore, a shift can be included in the symbol string by appending one character for each centring translation, s for a half or 0 for a plain centring. C, A, B and I have one centring translation and take one character, F has three, and P has none. Quotes are needed once a centring string is present. For example:

```
space_group_ssg_name_IT "Cmma(a,0,1/2)0s0"     ' C centred, standard centring
space_group_ssg_name_IT "Cmma(a,0,1/2)0s0 s"   ' C centred, a half on the centring
space_group_ssg_name_IT "Immm(0,0,g)000 s"     ' I centred, one centring character
space_group_ssg_name_IT "Fmmm(0,0,g)000 0s0"   ' F a half on the middle centring
```

2.4.5 ..... Trigonal, tetragonal and hexagonal groups

The examples above are orthorhombic, where every operator is its own inverse and each shift along the internal coordinate is a half or nothing. mod_tau_symbol also covers the trigonal, tetragonal and hexagonal groups, where an axis can have order three, four or six and its shift can be a third, a quarter or a sixth. The symbol keeps the one-line msCIF form, so it can be copied from a published file.

The characters after the modulation vector are read left to right, one for each direction of the point-group symbol. An axis and the mirror perpendicular to it, written *N/m*, takes two characters, while a rotoinversion such *as -3 or -6* takes one. A direction written as *1* carries no operator and its character is 0. The character set is *0* for no shift, *s*, *t*, *q* and *h* for a half, a third, a quarter and a sixth, and *-t*, *-q* and *-h* for the complements two thirds, three quarters and five sixths. For example:

```
mod_tau_symbol "P4(0,0,g)q"         ' the fourfold axis carries a quarter
mod_tau_symbol "P321(0,0,g)0s0"     ' a twofold on the second direction, a half
mod_tau_symbol "P31m(0,0,g)00s"     ' a mirror on the third direction
mod_tau_symbol "R-3(0,0,g)s"        ' a rotoinversion takes a single character
mod_tau_symbol "P6/mmm(0,0,g)00ss" ' 6/m is two characters, then a half on each mirror set
```

The derivation is for one modulation vector such that every operator maps to itself. The cubic groups have no such vector, since a threefold axis results in more than one direction. A cubic symbol therefore cannot be resolved, and mod_tau needs to be entered directly. A cubic modulation instead requires a star of several vectors in a higher superspace.

A star of that kind is entered through mod_user_operators rather than a symbol. Each additional MW carries its own internal coordinate, so where a single MW uses x4, a (3+2)D operator is read as x1,x2,x3,x4,x5, with one more part for each further MW. An operator may then send one MW onto another rather than back onto itself, coupling the waves, and the kernel refines directly on such coupled modulation waves.

2.4.6 ..... How the reflections are generated numerically

In determining the range of hkls present, the hkl generator considers the magnitude of *m* **q**; a satellite peak inside the observed *d* range can have a parent hkl that is far outside the observed *d* range. For powder data the range can be extended using extra_X_left and extra_X_right, this allows the peak tails to contribute the calculated pattern. The example CE2O2MNSE2\MOD.INP refines the single crystal data of Wang et al. (2015) against the published model, with one xdd_scr per order. The number of hkl groups generated, matches the published counts within a few reflections; the differences come from the observability cut. With more than one MW the generator runs over every combination of the per-MW orders and tests the magnitude of their combined contribution to **Q**, so a mixed satellite whose individual orders are small can still have a parent hkl well outside the observed range.

2.5 .....Special position constraints on modulation amplitudes

A site on a special position has its modulation amplitudes restricted by the site symmetry. add_constraints derives those restrictions and writes them into the output file; it is used as follows:

```
site Ce1  x 0 y 0.25 z 0.13346  occ Ce+3 1 beq 0.3
   mod_y  1 add_constraints
   mod_x  2 add_constraints

   ' the output file then contains
   mod_y  1  @ 0.001000 = -Get(isin)*Tan(Pi*(Get(z))); : 0
   mod_x  2  @ 0.001000 = -Get(isin)*Tan(2*Pi*(Get(z))); : 0
```

The keyword is replaced in the output file by the constraints. The sin and cos amplitudes are either refined, fixed to zero, or tied to another amplitude by an equation.

For each operator of the site symmetry group, with ε the sign the operator gives the modulation vector and τ its shift of the internal coordinate, the modulation function satisfies $\mathbf{u}(\varepsilon\, x_4 + \tau) = \mathbf{D}\,\mathbf{u}(x_4)$, where **D** is the representation of the operator for the quantity modulated: 1 for a scalar, **R** for a displacement, $\det(\mathbf{R})\mathbf{R}$ for a moment, and $\mathbf{U} \rightarrow \mathbf{R}\,\mathbf{U}\,\mathbf{R}^{\mathrm{T}}$ for the displacement tensor. The τ here is the intrinsic shift of the operator plus $\mathbf{H}\cdot\mathbf{r}$, where **H** is the reciprocal lattice vector in $\mathbf{qR} = \varepsilon\mathbf{q} + \mathbf{H}$ and **r** is the site. That second term depends on the site, so it is not the mod_tau of the operator. Applying this for every operator of the site symmetry group and averaging the results leaves only the amplitudes the symmetry permits; everything else is either zero or tied to what remains by an equation in the site

coordinates. The multiplier depends on τ and so on the site, and is written as an equation rather than a number because a number would go stale if a coordinate refines.

add_constraints derives constrains for mod_x/y/z, mod_occ, mod_beq, mod_mlx/y/z, and for mod_u11 to mod_u23. The derivation is the same in each case and only the representation of the quantity differs. add_constraints on any other keyword is an error.

The example MOD-MLX-CONSTRAINTS.INP shows this on a centrosymmetric site, where every displacive amplitude is forbidden and the moment is not, because inversion reverses a displacement and leaves a moment alone.

Get_mod(mod_keyword) written within a mod equation *mod_eqn*, returns the mod parameter with the same harmonic and mod_qm as *mod_eqn*. For example,

```
macro Get_mod_s(w) { Get(Get_mod(w), isin) }
macro Get_mod_c(w) { Get(Get_mod(w), icos) }
site ...
   mod_x  1  @ 0.001000      = -0.123 Get(isin);
   mod_y  2  @ 0.001000      = -0.456 Get_mod_s(mod_z);
   mod_z  2  @ = Get(icos);  = -0.567;
   mod_z  3  @ 0.001000      @ 0.001000
```

The Get(Get_mod(mod_z), isin) will get the isin parameter with the value of -0.567 as it checks that the harmonic and modulation wave mod_qm are the same as that of the equation where Get_mod is defined. This allows for constraints across axes.

2.6 .....Extinction

There is no separate extinction keyword. Secondary extinction is applied as a per reflection scale through scale_pks, with I_no_scale_pks giving the calculated intensity before scale_pks is applied:

```
prm extinction 0.001 min 1e-10 max 100
scale_pks = 1 / Sqrt(1 + 1e-5 extinction * I_no_scale_pks);
```

2.7.....Twinning and rotated structures

apply_rotation_matrix is typically used for twinning, see CE2O2MNSE2-MOD.INP. If the twinned individual is the same structure but in a different orientation, then Get_modulated_sites or Get_sites can be used together with apply_rotation_matrix.

apply_rotation_matrix takes the nine integer elements of a 3×3 matrix **T** and applies it to the str. It is not specific to modulated structures. **T** is applied to the symmetry operators of the space group, $\mathbf{R}_e$ → $\mathbf{TR}_e$ and t → $\mathbf{Tt}_e$, and to the modulation wave vector, **q** → **Tq**. Fractional coordinates and modulation parameters, dependent on $\mathbf{R}_e$, are internally rotated whilst all parameters stay as they were written. A magnetic operator keeps its time reversal sign. The determinant of **T** must be +1 or −1. Anything else is reported as “Determinant of rotation matrix should be +-1”.

mod_tau needs no attention. $\tau(\mathbf{T}\mathbf{R}_e\mathbf{T}^{-1}) = \tau(\mathbf{R}_e)$, and the **q** and the operators are rotated together. The sign ε_e of mod_tau is formed from $\mathbf{q}\cdot\mathbf{R}_e$, and $(\mathbf{T}\mathbf{q})\cdot(\mathbf{T}\mathbf{R}_e) = \mathbf{q}\cdot\mathbf{R}_e$ only because both carry **T**. Rotating one without the other flips ε_e for some operators, which changes the sign of mod_tau at $m \neq 0$ and is invisible at $m = 0$, where mod_tau is inert.

All parameters associated with a site turns with **T**, each by its own rule. Fractional coordinates and displacement amplitudes are polar vectors and take **T**. A magnetic moment is axial and takes det(**T**) **T**, so an improper **T** reverses a moment where it leaves a displacement alone. An anisotropic displacement parameter **T** U $\mathbf{T}^{\mathrm{T}}$. Occupancies, beq, and the crenel width and centre are scalars and do not turn at all. The modulation wave vector takes **T q**.

x, y, z turns and so do isin and icos of every harmonic of mod_x/y/z, mod_saw_x/y/z, and mod_legendre_x/y/z. The moment mlx/y/z and its modulation mod_mlx/y/z turn as axial vectors. mod_u11 to mod_u23 turn as tensors. occ, beq, mod_occ, mod_beq, mod_crenel_width and mod_crenel_center do not. The four-fold used in CE2O2MNSE2\MOD.INP has:

```
apply_rotation_matrix  0 -1 0 1 0 0 0 0 1
```

where (x, y, z) becomes (−y, x, z), and:

```
mod_y n  isin icos   becomes   mod_x n  with both signs reversed
mod_x n  isin icos   becomes   mod_y n  unchanged
mod_z n  isin icos   stays as it is
```

Space group operators carry **T**, not the parameters. Each equivalent position of the space group becomes $\mathbf{T}(\mathbf{R}_e\,\mathbf{x}_s + \mathbf{t}_e)$. A refinement therefore moves the parameter as written, not its rotated image. Hkl generation is still str dependent with operators $\mathbf{T}\,\mathbf{R}_e\,\mathbf{T}^{-1}$ and a wave vector of **T q.** A twin str with its parameters rotated by hand is a lot of work as seen in CE2O2MNSE2\MANUAL-TWIN.INP.

A twin law sends **q** back onto itself with ε ± 1 and a whole reciprocal lattice vector, or it sends it elsewhere. In the first case the twin's satellites fall on the same reciprocal points as the parent's and the two overlap. In the elsewhere case, reflections indexed on the parent's **q** has no reflections, and the str therefore has no twin. A message is displayed if this occurs; if there's only one str the refinement aborts. The set of vectors **q** defined by the point group is called a star.

2.8 .....R(F), and how reflections are grouped

R(*F*) is defined as:

$$R(F) = \frac{\sum\left(\left|\sqrt{Y_{calc}} - \sqrt{Y_{obs}}\right|\right)}{\sum\sqrt{Y_{obs}}}$$

Equivalent reflections are merged by default; TOPAS sums the observed intensities and fits against that. dont_merge_equivalent_reflections fits reflection as loaded; except that a satellite and its Friedel partner are always merged for modulated data. num_highest_I_values_to_keep applies to single crystal data.

2.9 .....Input to modulated structures and move_to_site

A modulated str with *m*=0 comprises both modulated and unmodulated sites. For *m*≠0 only modulated sites can be included. Satellite peaks for +*m* and -*m* will be at different 2*θ* positions for a particular hkl. 2*θ* dependent peak shapes and intensities, such as Lorentz polarization etc., are correctly applied. Site dependent harmonic keywords are in the form:

keyword harmonic sine_parameter cosine_parameter

For example,

mod_x 2 sine_prm_name 0.1 cosine_prm_name 0.1

The sine and cosine parameters are called **isin** and **icos**. The following template describes a modulated structure with m = 0, 1 and 2:

```
xdd ...
   str mod_m !m0_global 0 local !m0_local 0...
      site S1 ...
      site S2 ... mod_x ... ' modulated site
      site S3 ... mod_x ... ' modulated site
   str mod_m 1 ...
   str mod_m 2 ...
   for strs { ...
      if Prm_There(m0_local) == 0 {
         Get_modulated_sites(m0_global)
      }
   }
```

or, alternatively:

```
xdd ...
   str mod_m $m0 0 ...       ‘ m0 has local scope
   str mod_m m1 1 ...
   str mod_m m2 2 ...
   for strs { ...
      if Prm_There(m0) {
         site S1 ...        ’ unmodulated sites
      }
      site S2 ... mod_x ... ' modulated site
      site S3 ... mod_x ... ' modulated site
   }
```

In the first format, all sites unmodulated and modulated are defined in the *m*=0 str. In the second format, sites are defined with the for loop. The first format allows the use of the @ character as parameter names. Example RIGID-LP-MOD_X.INP follows the first format. Here, a modulated structure is described using two str's. For this to make physical sense, the sites in the satellite str's (*m*≠0) must be identical to the sites in the *m*=0 str. The Get_modulated_sites macro (see TOPAS.INC) creates sites in the *m*=1 and *m*=2 str from *m*=0 str, with all parameters pointing to those in the *m*=0 str. Get_modulated_sites identifies the *m*=0 str from the m0_global parameter which is defined in the scope of the *m*=0 str.

The move_to_site keyword can be use to place the mod parameters at the end of a str as follows:

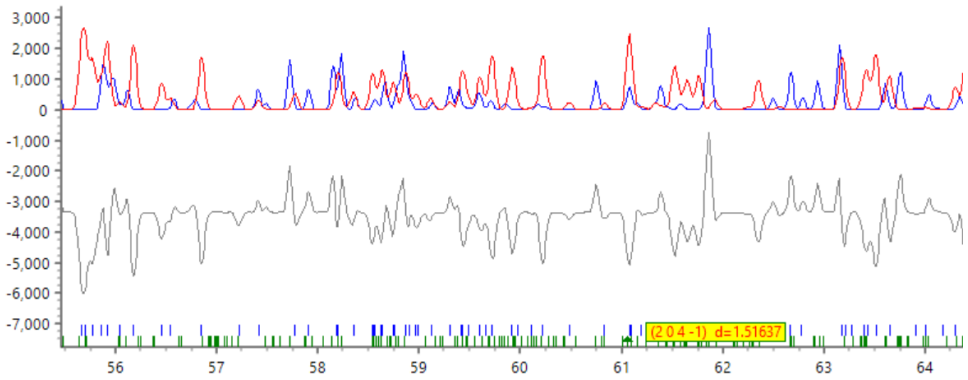
```
site S1 ...
site S2 ...
move_to_site S1 mod_x ...
move_to_site S2 mod_y ...
```

2.10.....Rigid bodies with modulated data

rigid bodies can be defined in the *m*=0 str; *m*≠0 strs can then point to site parameters in the *m*=0 str using the Get_modulated_sites macro. RIGID-LP-MOD_X.INP example, without CREATE__ is as follows:

```
r_f 0 ' Single crystal R(F) goodness of fit indicator

xdd rigid-lp-mod_x.xy  r_f 0 ' xdd dependent r_f
   neutron_data
   lam ymin_on_ymax 0.0001 la 1 lo 1 lg 0.3

   str phase_name m0 mod_m !m0 0 ' the handle - position in the tree
      CS_G(@, 100)
      scale sc 10
      prm ro 1.5 min 1 max 3
      prm ax 0.1      prm bx 0.1
      prm cx 0.1      prm dx 0.1

      site Mn1  x 0.30  y 0.20  z 0.15  occ Mn+3 1 beq 0.4
         mod_x 1  = ax; = bx;
      site O1   x 0.532 y 0.20  z 0.15  occ O-2 1 beq 0.6
         mod_x 1  = cx; = dx;
      site O2   x 0.068 y 0.20  z 0.15  occ O-2 1 beq 0.6
      site O3   x 0.30  y 0.432 z 0.15  occ O-2 1 beq 0.6
      site O4   x 0.30  y 0.032 z 0.15  occ O-2 1 beq 0.6
      site O5   x 0.30  y 0.20  z 0.382 occ O-2 1 beq 0.6
      site O6   x 0.30  y 0.20  z 0.082 occ O-2 1 beq 0.6

      site Fe1  x @ 0.10 y @ 0.42 z @ 0.75 occ Fe 1 beq 0.6
      view_structure
      rigid
         Octahedra(Mn1, O1, O2, O3, O4, O5, O6, ro)
         Translate_with_site_start_values(Mn1, tx, ty, tz)

   str phase_name m1 mod_m 1 ' Satellites
      scale @ 1000 ' Deliberately made unphysical for testing reasons
      CS_G(@, 200)
      Get_modulated_sites(m0)

   for strs {
      space_group P21/c
      mod_tau_symbol s
      a a 8.42  b b 7.32  c c 6.18
      mod_qx qx 0.11
   }
```

The refined parameters are highlighted. This example shows that the derivatives of the rigid body parameters are calculated correctly for the satellite str, even though the rigid body is defined in the m=0 str. The structures share one scale, since they are one crystal. The example also has separate peak shapes for m=0 and m=1, and starting values far from the optimum, and it still converges. The graphic below shows the hkl tick marks at iteration zero, with m displayed on the highlighted mark.



2.11.....Modulation wave vectors

mod_qm defines one modulation wave vector, or modulation vector. mod_qx/y/z are its components in reciprocal lattice units. mod_m is the satellite order generated by the str. A str generates one order per modulation vetor, so a pattern comprising main reflections and satellites is described by several str's; one with mod_m=0 for the main reflections, one with mod_m comprising hkls for m=1 and m=-1, for the first order satellites, and so on, with their sites tied together by equations. Positive and negative orders are different reflections at different d-spacings; both are needed and each str has both for a particular m. Each str has its own peak shapes; angle dependence for the peak shapes as well as angle dependence for intensity parameters, such as f_0, are automatically corrected.

mod_qm defines one modulation wave vector. mod_qx/y/z are its components in reciprocal lattice units, and mod_m is the satellite order the str generates. A str generates one order per modulation wave vector, so a pattern of main reflections and satellites needs several strs: one with mod_m=0 for the main reflections, one with mod_m=1 for the first order satellites, and so on, with their sites tied together by equations. Positive and negative orders are different reflections at different d spacings; each str generates both for its own m. Each str has its own peak shapes, and the angle dependence of the peak shapes and of the intensity parameters is applied automatically.

Satellites are generated for $h \geq 0$ only. The Friedel partner of a satellite at $Q=h+m\,q$ lies at $-Q$, which is the reflection $-h$ at order $-m$, and so belongs to the str carrying the opposite order; that is why both are needed. For $h > 0$, the partner is not generated and the reflection carries a multiplicity of two. For h=0 the partner is generated, by the other str, and the multiplicity is one. At m=0, the two have the same d-spacing and merge, so main reflections carry a multiplicity of two in the ordinary way. A refinement holding main reflections and satellites weights them correctly against each other, which is what determines the modulation amplitudes.

When the structure has a single modulation vector, mod_qm may be omitted and mod_qx/y/z and mod_m written directly under str. The modulation vector then takes the identifier 0, and the site dependent qm may be omitted as it defaults to 0.

Defining mod_tau is optional, a default value of zero is given for each equivalent position of the space group. The values are rational, usually 0 or 1/2. They belong to the str and is applied to all sites. A τ of 1/2 on a centring operator makes the odd order satellites present where a zero τ extinguishes them; the even orders are unchanged either way.

load_q_vector_type associates a modulation vector with a modulation index column of a single crystal reflection file; the types are refln_modulation_index_1 which is the upwards.

load_q_vector_type associates a modulation vector with the column of a single crystal CIF reflection file that holds its satellite order. The value is the column name without its leading underscore, for example refln_modulation_index_1, refln_modulation_index_2 or refln_index_m_1; the default is refln_modulation_index_1. With several modulation vectors, each mod_qm must name its own column, otherwise an error is thrown. A file without the named column is loaded whole.

mod_d_tolerance sets the *d*-spacing tolerance for deciding when two generated satellites are the same reflection. p1_hkls generates all reflections and is a check for merged reflections. A structure defined with or without p1_hkls must give the same Ycalc.

2.12.....Modulation functions

Each modulated quantity is a Fourier series in the modulation phase $v_{s,e}$ of its modulation vector. For a quantity $u_{s,e}$:

$$u_{s,e,a}(v_{s,e,a}) = u_{0,s} + \sum_{n}\left[\mathrm{isin}_{n,s,a}\ \sin(2\pi n v_{s,e,a}) + \mathrm{icos}_{n,s,a}\ \cos(2\pi n v_{s,e,a})\right]$$

where u_0 is the ordinary x, occ, beq etc..., and *n* is the harmonic number, counting from 1.

Which order-*m* a parameter reaches, as a function of its harmonic, depends on the quantity modulated. mod_occ and the moment components mod_mlx/y/z enter the structure factor linearly, so harmonic *n* contributes only at order-*m* = ±*n*. A first harmonic occupancy on a str with mod_m = 2 therefore contributes nothing on its own, and a warning is given. mod_x/y/z, mod_beq and mod_u11 to mod_u23 enter through an exponential, so they multiply as functions of the modulation phase and therefore convolve in order-*m*. Harmonic *n* then contributes to every multiple of *n*, and a set of them reaches every multiple of their greatest common divisor: mod_x 2 with mod_x 3 gives every order-*m*. That convolution can also carry an occupancy or moment to an order-*m* it could not reach alone, which is tested before the warning is given. A crenel reaches every order-*m*, so a site carrying one is exempt from all of this.

qm is a site dependent number used to identify the modulation vector the harmonic belongs to, and may be omitted when there is only one modulation vector. Any number of harmonics may be given for a keyword on one modulation vector. The same harmonic of the same keyword on two different modulation vectors is two waves and is allowed; the same harmonic of the same keyword twice on one modulation vector is an error.

2.13 .....Modulated displacement parameters

mod_beq modulates the isotropic displacement parameter and mod_u11 ... mod_u23 the anisotropic tensor. A Debye-Waller exponent is linear in U, so both modulate the exponent by a scalar sinusoid and both are folded into the modulation the same way. What differs is the scale; mod_beq enters through $-1/(4\,d^2)$, and the tensor components through the rotated tensor contracted with the six monomials of **Q**.

mod_beq and mod_u11 ... mod_u23 are mutually exclusive on any one modulation vector, the same as the static beq and u11 ... u23 are for a site; both modulate the same exponent and would simply add. On different modulation vectors they belong to different modulation phases and may both be used. Static u11 ... u23 may be used on a modulated site and are evaluated at the satellite **Q** rather than at the parent reflection.

2.14 .....Crenel and sawtooth functions

A crenel occupancy is 1 over a window of the modulation phase and 0 outside. It describes a site that is present over part of the modulation period, something a Fourier series represents poorly. A sawtooth is a displacement that varies linearly with v across that same window, and the two are normally used together.

mod_crenel_center is the centre of the window, v_0, and mod_crenel_width its width Δ as a fraction of the modulation period, with $0 < \Delta \leq 1$. mod_saw_x/y/z are the components of the sawtooth displacement amplitude in fractional coordinates, traversed linearly across the window. Both are described by a single Fourier coefficient rather than by a series. For satellite order m:

$$P_{m,s,e} = \Delta_s \ \mathrm{sinc}(\sigma_e \, \mathbf{Q} \cdot \mathbf{A}_s - m\Delta_s) \ \ \exp\left(-2\pi i m \sigma_e v_{0,s}\right)$$

where $\mathrm{sinc}(x) = \sin(\pi x)/(\pi x)$, **A** is the sawtooth amplitude and σ is the sign the space group operator gives the modulation phase. The displacement inside the window is linear in v, so the integral over the window is elementary.

At m=0 the coefficient is Δ, so a crenel of width Δ with no sawtooth scatters exactly as an unmodulated site of occupancy Δ. And at Δ=1 the coefficient is zero for every $m \neq 0$, so an atom present over the whole period generates no satellites at all.

A crenel is the whole modulation of its modulation vector, not a harmonic of it, and one is allowed per modulation vector per site. It may carry mod_x/y/z for a displacement defined over the full period, and either mod_saw_x/y/z or mod_legendre_x/y/z for one defined across the window. The sawtooth is the first Legendre order, so a modulation vector takes one or the other, not both. A crenel cannot carry mod_occ as that is what the crenel already is. And mod_beq, mod_ml* and mod_u* are refused on that modulation vector.

Legendre polynomials give a second way to describe the displacement inside a crenel window. mod_legendre_x/y/z are accepted only on a site that has a mod_crenel_width on the same modulation vector, and the order must be 1 or more.

Without a window there is no window coordinate, and over a full period mod_x/y/z are already the right keywords. The polynomials are taken in $x = 2(v - v_0)/\Delta$, which runs from −1 to 1 across the window.

The first order is the sawtooth. A displacement linear in v across the window is $(A/2)P_1(x)$, so mod_saw_x A and mod_legendre_x 1 A/2 describe the same structure by two different routes; the closed form above, and numerical quadrature over the window. CRENEL-LEGENDRE.INP computes both and writes two identical calculated patterns. Because they are exactly degenerate rather than merely correlated, the two cannot be given on one modulation vector.

Orders of two and above have no sawtooth equivalent, and they are the reason for the keyword. Sine and cosine are orthogonal over a full period but not over part of one, so on a crenel site the ordinary harmonic amplitudes come out strongly correlated and refine poorly. Legendre polynomials are orthogonal on the window by construction, which leaves their amplitudes independent.

A Legendre coefficient changes the crenel's integrand rather than multiplying the result; it sits inside the same integral as v_o and Δ. That integral has no closed form beyond first order, it is therefore evaluated numerically when a Legendre coefficient is present, taking the derivatives from the same integral. Without one the closed form is used.

2.15 .....Refinement

All modulation amplitudes, the crenel and sawtooth parameters, and the modulation wave vector components have analytic derivatives for peak intensities. For powder data, refining mod_qx/y/z, displaces the satellites peaks in 2θ; the peak shift is the dominant part of that derivative.

The origin of the modulation phase v is arbitrary, and separately so for each modulation vector. Shifting it changes every isin and icos on that modulation vector whilst leaving the intensities unchanged. With a single modulated site only the amplitude of each harmonic is determinable but not its phase. A second site, modulated differently, fixes the origin by interference. mod_crenel_center is a phase origin rather than an amplitude, so refining it on a structure with one modulated site does nothing.

2.16 .....A worked example — Ce2O2MnSe2

CE2O2MNSE2\MOD.INP refines the incommensurately modulated structure of Ce2O2MnSe2 against single crystal data, with satellites to fourth order and a twin. The main reflections and each satellite are separate xdd_scr blocks with each order viewed graphically. The m=0 str is where the sites are defined; other structures pick up sites using Get_modulated_sites; the structure is written once with all parameters constrained to the base str. This allows the '@' to be used where the sites are defined. Get_modulated_sites also allow for easy inclusion of rigid bodies even across a twin. scale_0 identifies the m=0 str.

Mn1 is a crenel. It occupies half of the modulation period, centred at v = 0.75, and its displacement inside that window is given by Legendre orders 1 and 3 rather than by a sawtooth. The other three sites are ordinary Fourier displacements to the fourth harmonic. The twin is a second m=0 str with

apply_rotation_matrix and its own scale and extinction. The INP file comprises the keywords apply_rotation_matrix, mod_tau_from_cif, and mod_tau_symbol:

```
r_f 6.22 ' global r_f
xdd_scr ce2o2mnse2.hkl
   local !xdd_m 0
   str
      phase_name base
      scale scale_0 0.09003
      site Ce1  x 0.00000   y 0.25000   z 0.13346  occ Ce+3 1 beq = b + 0.3245;
         mod_y  1   0.021490  -0.009590
         mod_x  2  -0.000620   0.000700
         mod_z  2  -0.000140  -0.000130
         mod_y  3  -0.000590   0.001830
         mod_x  4   0.000050   0.000400
         mod_z  4   0.001200  -0.000130
      site Se1  x 0.00000   y 0.25000   z 0.67756  occ Se   1 beq = b + 0.5180;
         mod_y  1   0.011400   0.018340
         mod_x  2   0.001280  -0.002600
         mod_z  2  -0.000300  -0.000130
         mod_y  3   0.008200  -0.000810
         mod_z  4  -0.001000   0.000800
      site Mn1  x 0.25000   y 0.00000   z 0.50000  occ Mn+2 1 beq = b + 0.6317;
         mod_crenel_center 0.7500
            mod_crenel_width  0.5000
         mod_legendre_x 1  -0.034100
         mod_legendre_x 3  -0.016200
      site O1   x 0.25000   y 0.00000   z 0.00000  occ O-2  1 beq = b + 0.0900;
         mod_y  1   0.022900   0.000000
         mod_z  1   0.000000  -0.005000
         mod_x  2  -0.001400   0.000000
         mod_y  3  -0.002300   0.000000
         mod_z  3   0.000000   0.003900
         mod_x  4   0.002800   0.000000

                     r_f  6.0150751
xdd_scr ce2o2mnse2.hkl r_f  5.90432369 local xdd_m 1 str
xdd_scr ce2o2mnse2.hkl r_f  11.1528392 local xdd_m 2 str
xdd_scr ce2o2mnse2.hkl r_f  12.6226363 local xdd_m 3 str
xdd_scr ce2o2mnse2.hkl r_f  26.4932873 local xdd_m 4 str

prm b 0.4268 ' added to each site's fixed beq

for xdds {
```

```
   if xdd_m { ' m != 0
      str
         phase_name = Concat(String(Satellite_m=), xdd_m);
         scale scale_satellites 0.17741
         Get_modulated_sites(scale_0)
   } else { ' m = 0
      str
         phase_name twin
         apply_rotation_matrix 0 -1 0 1 0 0 0 0 1
         Get_sites(scale_0)
         scale = scale_0; 'constrained to base str
   }
   for strs {
      a 5.6799  b 5.6790  c 9.0952
      i_on_error_ratio_tolerance 3
      space_group Cmma
      ' mod_tau_symbol Cmme(a,0,1/2)0s0
      mod_tau_from_cif "ce2o2mnse2.cif" ' Read directly from CIF file
      mod_qx 0.15800 mod_qz 0.5
      mod_m = xdd_m;
      Extinction(extinction, 2.50697744`)
   }
}
```

The logic comprises one if statement and two for loops; this is trivial considering the complexity of the problem. Setting phase_same in the manner shown is useful for identifying the plots graphically. The optional mod_tau is the phase shift in the modulation wave; if used, one needs to be defined for each equivalent positions in the SG\CMMA.SG file.

2.17 .....A published model — org\mod.inp

ORG\MOD.INP, Wagner & Schönleber (2009), is a test on a published data using the observed reflection file. TOPAS takes _refln_F_meas and squares it and honours _refln_observed_status; 24813 reflections survive, spread over orders −4 to +4. Every |mod_m| order is a separate xdd_scr with its own str. The structure is defined once and applied to all orders by for xdds / for strs. With 5 refined parameters on 24813 reflections, $R(F)$ returns 3.69% against the published number of 3.63%.

2.18 .....msCIF output

msCIF output is realized using the following macros:

[Out_CIF_STR(file)] [Out_msCIF(file)]

These macros outputs wave vectors, the Fourier components of the displacements, the occupancy and the displacement parameters, and the crenel and sawtooth special functions. See MSCIF-OUT.INP.

There are three differences from the dictionary; TOPAS gives a harmonic number where msCIF names a wave vector, so mod_x 2 is written as the vector 2q and referred to by its sequence id. And msCIF gives the sawtooth its own centre and width, separate from the crenel's; in TOPAS the sawtooth is defined over the crenel window, so the crenel values are written for both. _space_group_ssg_name is

not written; TOPAS holds the three-dimensional space group together with the modulation wave vectors, it does not derive a superspace group symbol.

The reflections themselves can be written as a plain table with phase_out. Get(mod_qms) returns the satellite order on each modulation vector for the reflection currently being written; one integer per modulation vector, in the order the mod_qm blocks were declared. mod_qm is a string and is therefore formatted with *%s*:

```
phase_out file
   load out_record out_fmt out_eqn {
      "\n%4.0f" = H;
      " %4.0f"  = K;
      " %4.0f"  = L;
      " %s"     = Get(mod_qms);
      " %14.6g" = I_no_scale_pks;
   }
```

Each satellite's order carries the sign of its modulation index, so a satellite and its Friedel pair are distinguishable. The order tuple corresponds directly to the values stored in the msCIF fields _refln_index_m_1 and _refln_index_m_2.

2.19 .....Parameter limits

Table 2-1. Parameter limits for modulation parameters.

Parameter	Min	Max
mod_qx, mod_qy, mod_qz, mod_crenel_center	None	None
mod_x, mod_y, mod_z, mod_legendre_x, mod_legendre_y, mod_legendre_z, mod_occ, mod_saw_x, mod_saw_y, mod_saw_z	-1	1
mod_beq, mod_mlx, mod_mly, mod_mlz	-20	20
mod_u12, mod_u13, mod_u23	-10	10
mod_u11, mod_u22, mod_u33, mod_crenel_width	1e-5	20

2.20 ....References

Petříček, V., Eigner, V., Dušek, M. & Čejchan, A. (2016). Z. Kristallogr. 231, 301-312. "Discontinuous modulation functions and their application for analysis of modulated structures with the computing system JANA2006"

van Smaalen, S. (2007). Incommensurate Crystallography, IUCr Monographs on Crystallography 21, Oxford University Press.

Wagner, T. & Schönleber, A. (2009). Acta Cryst. B65, 249-268. "A non-mathematical introduction to the superspace description of modulated structures"

Wang, C., Ainsworth, C. M., Gui, D. Y., McCabe, E. E., Tucker, M. G., Evans, I. R. & Evans, J. S. O. (2015). Chem. Mater. 27, 3121-3134. “Infinitely Adaptive Transition Metal Oxychalcogenides: The Modulated Structures of Ce2O2MnSe2 and (Ce0.78La0.22)2O2MnSe2”