Bandstructure (spectral-function) unfolding in ONETEP

As described in the Supplementary Information in the work of Constantinescu and Hine[1], spectral function unfolding is the means through which one can study the bandstructure of the primitive cell from simulations involving (often complicated) supercells. For details theoretical descriptions, one should visit the aforementioned article[1].

In this documentation, I will give a short explanation of the procedures required to unfold the bandstructure in a ONETEP "Properties" calculation. We strongly suggest that one reads the entire document carefully before attempting any calculations, as there are some stringent requirements along the way. For further details, please contact Gabriel Constantinescu, the code author. Essentially, all one will need is the following group of keywords and blocks:

```
BSUNFLD_CALCULATE : T
BSUNFLD_TRANSFORMATION : t11 t12 t13 t21 t22 t23 t31 t32 t33

%block SPECIES_BSUNFLD_GROUPS
Species_name-1 Species_name-2 ... Species_name-N
%endblock SPECIES_BSUNFLD_GROUPS

BS_PERTURBATIVE_SOC : F
BSUNFLD_NUM_ATOMS_PRIM : nat_prim
BSUNFLD_RESTART : F
BSUNFLD_NUM_EIGENVALUES : num_eigvl

%block BSUNFLD_KPOINT_PATH
 fraction_G1_kpt-1 fraction_G2_kpt-1 fraction_G3_kpt-1
 fraction_G1_kpt-2 fraction_G2_kpt-2 fraction_G3_kpt-2
 ...
 fraction_G1_kpt-N fraction_G2_kpt-N fraction_G3_kpt-N
%endblock BSUNFLD_KPOINT_PATH

BSUNFLD_NUM_KPTS_PATH : num_kpts_path
```

For each in turn:

- BSUNFLD_CALCULATE setting this to True will enable the calculation of the unfolded spectral function. Default: False

- BSUNFLD_TRANSFORMATION is a list of 9 integers representing the flattened transformation matrix from the primitive cell to the supercell: $\mathbf{T} = \begin{bmatrix} t11 & t12 & t13 \\ t21 & t22 & t23 \\ t31 & t32 & t33 \end{bmatrix}$, where $\mathbf{T} \begin{bmatrix} \mathbf{a_1} \\ \mathbf{a_2} \\ \mathbf{a_3} \end{bmatrix} = \begin{bmatrix} \mathbf{A_1} \\ \mathbf{A_2} \\ \mathbf{A_3} \end{bmatrix}$, with $\mathbf{a_i}$ ($\mathbf{A_i}$) being the lattice vectors of the primitive cell (supercell). For instance, BSUNFLD_TRANSFORMATION: 4 0 0 0 4 0 0 0 1 would correspond to an implicit 4x4x1 supercell. Default: 4 0 0 0 4 0 0 0 1, corresponding to a 1x1x1 supercell.

- The block SPECIES_BSUNFLD_GROUPS contains the atoms on which we are projecting; one needs to specify the ids of the atoms species (can be different from the chemical symbol) on which the spectral function is projected. Currently, only one group at a time is allowed. Example:
  ```
  %block SPECIES_BSUNFLD_GROUPS
  Mo Se1
  %endblock SPECIES_BSUNFLD_GROUPS
  ```
  this projects onto the $MoSe_2$ monolayer in a $MoSe_2/WSe_2$ heterostructure, where the Se atoms belonging to the $MoSe_2$ monolayer have been denoted as *Se1*.

- The logical keyword BS_PERTURBATIVE_SOC controls the inclusion (True) or exclusion (False) of perturbative spin-orbit coupling in our calculation. Note that if set to True, the eigenvalues are no longer spin-degenerate and one will have twice the number of states. Since this option essentially quadruples the size of the Hamiltonian and overlap matrix, the calculation will be significantly slower. Default: False.

- The integer keyword BSUNFLD_NUM_ATOMS_PRIM indicates the number of atoms in the primitive cell onto which you are unfolding. This brings us to a major requirement of the code: while the projected atoms can be split across the list of all input atoms, they former must be grouped by primitive cells, always maintaining the same order of the atoms in the primitive cells!. Example: if atoms At1 and At2 form a primitive cell, the input ordering "At3 At1 At2 At4 At1 At2 A3" is correct, while "At3 At1 At2 At4 At2 At3 At1" or "At3 At1 At2 At4 At2 At1 At3" are incorrect.

- The logical keyword BSUNFLD_RESTART controls whether one wishes to reuse previously calculated values for the spectral function. The restart procedure works as

follows: the code writes a restart file ("unfolded_specfunc_red.dat") where it stores the calculated values for the unfolded spectral function at unique k-points of the monolayer - the values that have not been calculated yet are filled with zeros. If this file is present and the restart keyword is set to true, the code will read in the previously calculated values and skip them in the current computations. The final output file ("unfolded_specfunc.dat") will only be written once all the k-points have been dealt with.

Each line in final output file or the restart file contains 8 numbers: the eigenvalue (in eV), the real part of the unfolded spectral function (at that k-point and eigenvalue), the imaginary part of the unfolded spectral function, the eigenvalue count (from 1 to 2· BSUNFLD_NUM_EIGENVALUES, the x, y, and z component of the current primitive-cell k-point ($\mathring{A}^{-1}$), and the index of the k-point (from 1 to the total number of considered k-points).

After the final output file has been obtained, one can use a discretisation script (should be found on the ONETEP webside, in the utilities section), in order to obtain a file that is ready to plot with gnuplot.

- The integer keyword BSUNFLD_NUM_EIGENVALUES controls the number of eigenvalues (above and below the Fermi level) for which the spectral function is calculated. If set to negative values, BSUNFLD_NUM_EIGENVALUES will internally be set to the number of nondegenerate occupied eigenstates. Thus, in total, for each k-point, the spectral function will be calculated at 2· BSUNFLD_NUM_EIGENVALUES.
- The block BSUNFLD_KPOINT_PATH indicates the fractional coordinates of the different k-points that mark endpoints of desired paths through the primitive-cell Brillouin zone. The fractional coordinates are with respect to the reciprocal lattice vectors of implied the primitive cell, not the simulation cell (supercell).
- BSUNFLD_NUM_KPTS_PATH represents the number of k-points calculated along each path from the BSUNFLD_KPOINT_PATH block. This number includes the endpoints of each path. Default: 2 (the endpoints only)

---

[1] Constantinescu, G. C.; Hine, N. D. M. *Phys. Rev. B* **2015**, *91*, 195416