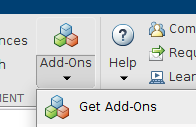
# Excitations Data Analysis Training Course – Worksheet 6

## SpinW Introduction

[SpinW](https://spinw.org/) is a Matlab program to calculate spin wave dispersion and inelastic neutron scattering spectra using linear spin wave theory.

You can install SpinW using the “Add-Ons” option:

Then search for SpinW and “Add” it.

On the IDAaaS system, you should use the

>> horace\_on

command instead.

SpinW is implemented as a Matlab class. To run a calculation you first have to create an object (instance) of this class:

>> object = spinw();

then set properties, such as the lattice parameters and atom positions, by calling methods of the class, e.g.:

>> object.genlattice('lat\_const',[4,4,4],'angled',[90,90,90],'spgr','P 4');

## SpinW I – Excercise 1: 1D chain

As an alternative to setting up a spin wave model from scratch, SpinW also has a function to set up some basic models called sw\_model. Type:

>> help sw\_model

to see how to use this. Generally all the SpinW commands have help text with the syntax of the command at the start of the help text. (As an alternative to “help <command>”, you can also use “doc <command>” which will display the help text in a separate window.)

1. Using sw\_model set up a model of an antiferromagnetic spin chain with nearest neighbour interaction *J*=1 meV. (Remember that positive values indicate antiferromagnetic exchange in the convention used by SpinW)
2. Plot the model – which crystallographic direction is the spin chain along? In which direction in Q would you expect the spin waves to be dispersive?

Hint: You can use “plot(object)” to plot a 3D model of the SpinW object.

1. Use the spinwave method to calculate the spin wave dispersion *along the chain*, and plot the result.

Hint: Type “help object.spinwave” or “help spinw/spinwave” to get the syntax for the spinwave method. You can use sw\_plotspec to plot the spectra object returned by spinwave. [N.B. a “method” is a function belonging to a class]

1. Use the sw\_egrid function on the calculated spectrum object to create an intensity spectrum and plot it with sw\_plotspec with the 'dE' option.
2. Now create a *ferromagnetic* chain using sw\_model and calculate and plot its neutron intensity spectrum. (Use the figure command to create a new figure before plotting the new spectrum otherwise the old [antiferromagnetic chain] spectrum will be erased and overplotted).
3. Aside from the difference in the dispersion, what is the difference between the neutron intensity between the ferromagnetic and antiferromagnetic chain? Why do you think this difference arises?

Hint: Remember that neutron scattering function is proportional to the spin-spin correlation function.

1. Use the powspec method to calculate a powder spectrum of the FM chain between 0 and 3Å-1 in 100 steps with 1000 random sampling points per Q bin. What feature of the dispersion correspond to what feature of the powder spectrum?
2. Repeat the powder spectrum calculation for the AFM chain – what differences do you notice?
3. Finally, you can also calculate the magnon density of states (DOS) by randomly sampling in the first Brillouin zone and summing the results over Q:

Qvec = rand(3, 1e5); % Generate 10000 random Q points between [0,1]

dosspec = swobj.spinwave(Qvec); % Calculates the dispersion

% Now take the trace of the Sab correlation matrix instead Sperp

dosspec = sw\_egrid(dosspec, 'component', 'Sxx+Syy+Szz');

% Now broaden the spectrum with a Gaussian of width 0.1meV

dosspec = sw\_instrument(dosspec, 'dE', 0.1);

magnon\_dos = sum(dosspec.swConv, 2);

% The energy vector is stored as bin boundaries in the spectrum

en = (dosspec.Evect(1:end-1) + dosspec.Evect(2:end))/2

figure; plot(en, magnon\_dos);

1. How does the DOS compare to summing the powder spectrum over Q? Why did we use the 'Sxx+Syy+Szz' component instead of the neutron 'Sperp' component which is the default used in powspec?

## SpinW I – Excercise 2: Triangular Lattice Antiferromagnet

1. Create an empty spinw object with:

>> swobj = spinw();

1. Use the genlattice method of this object to create a hexagonal lattice with parameters *a*=*b*=3 Å and c=5 Å (α=β=90˚, γ=120˚).
2. Use the addatom method to add a Ni2+ ion with S=1 at **r**=(0,0,0).
3. Use the gencoupling method to generate the bonds up to 5 Å distance.
4. Use the addmatrix method to add an exchange tensor (3x3 matrix) representing an antiferromagnetic Heisenberg exchange interaction of magnitude 1 meV, then use the addcoupling method to define the first (nearest neighbour bond) using this matrix (tensor).
5. Plot your model and see that it represents a triangular lattice antiferromagnet (you should be able to click on the atoms and bonds to get information on them).
6. Use the genmagstr method with the 'mode', 'helical' option, **k**=[⅓,⅓,0], **S**=[1;0;0] and **n**=[0,0,1] to create a 120˚ magnetic structure in the incommensurate representation.
7. Calculate and plot the inelastic neutron spin wave spectrum.
8. Add an easy-axis anisotropy in the *a-b* plane (perpendicular to [0,0,1]) with magnitude 0.1 meV using the addmatrix and addaniso methods and recalculate and re-plot the INS spectrum. What is the difference between this and the previous spectrum you plotted? From this can you infer what fluctuations are associated with the different magnon modes in the spectrum? (How many modes are there in the spectrum?)
9. Make a copy of swobj (the spinw object you just created – spinw objects are “handle” objects so with swobj2 = swobj, swobj2 is just another handle to the original object – operations on swobj2 actually operate on the same object as swobj. In Horace most objects are “value” objects where the = (equals) operator creates a new copy). Using genmagstr on the copied object with the option 'nExt', 0.1 and the same parameters as before, create the same 120˚ magnetic structure but in the super-cell representation instead of the rotating frame (incommensurate) representation. Plot the model to check that it still looks like the 120˚ structure.
10. Calculate and plot the INS spectrum – what’s the difference with the previous (incommensurate) case? (How many modes are plotted now?) Why is this?
11. Add an easy-axis anisotropy along the *a* axis with magnitude +1 meV. Recalculate the INS spectrum – you should get an error message. Why does this occur?

Hint: Think about what the axial anistropy term you just added does to the magnetic ground state vs what the current magnetic ground state (spin structure) is set to.

1. Use the 'hermit', false option to spinwave to force the calculation and plot the result – what’s the maximum energy of imaginary modes? If the maximum energy of the imaginary mode is small (~0.01-0.02 [a few percent] of the maximum energy of all the modes) then you can ignore the warning and the calculate spectrum is probably accurate enough. But if it is much larger then you should not trust the calculation.
2. Use the optmagsteep method to refine the magnetic structure, and recalculate and replot the INS spectrum. Are the imaginary modes still there? What are the main differences you see now, compared to previously? Can you explain what the modes are now?

## SpinW I – Exercise 3: BiFeO3

In this exercise we’re going to look at BiFeO3 a distorted G-type (cubic) antiferromagnet, which has a long period (small ***k***) cycloid magnetic structure as the result of the Dzyaloshinskii-Moriya interaction. The spin Hamiltonian and INS data is described in this paper:

<https://web.ornl.gov/~okapon/manybodytheory/publication/pdf/PhysRevLett_109_067205.pdf>

1. Construct a spinw object from a CIF description of the structure of BiFeO3 using:

>> bfo = spinw('BiFeO3.cif');

The CIF can be found in here:

<https://github.com/pace-neutrons/edatc/matlab_scripts/BiFeO3.cif>

1. Plot the Fe-O octahedra using swplot.plotchem. Type bfo.unit\_cell first to check what the atom labels are. (They should be 'Fe Fe' and 'O O' for the iron and oxygen atoms). Can you see how this structure relates to the cubic perovskite structure?
2. Set the spin magnitude and label of the Fe3+ ions to S=5/2 and 'MFe3':

bfo.unit\_cell.S(2) = 2.5; % Check that Fe is the 2nd atom.

bfo.unit\_cell.label{2} = 'MFe3'; % So SpinW knows the formfactor

1. Use gencoupling, addmatrix and addcoupling to define the nearest (J1=4.5 meV) and next nearest (J2=0.2 meV) neighbour couplings. (Note that the PRL paper uses the measured moment at 200K for the spin magnitude S value whilst we have used the theoretical S=5/2, and so there is a factor of 1.44 in the calculation as noted on page 4 of the PRL. Plot the structure again and check it looks like fig 1 of the PRL.
2. Use addmatrix to define a DM interaction (read its help text) and the use addcoupling to add an addition nearest neighbour DM interaction (SpinW can accommodate up to three difference exchange tensors per bond; the limit of 3 is just a hard-coded limit to save memory and can be extended (reprogrammed) if there is demand). The DM vector is along [1, -1, 0] in the hexagonal unit cell we are using, which is equivalent to the cubic [1,1,-2] direction which the PRL paper uses. Use the same magnitude DM as in the paper (D=0.1623 meV).
3. Use addmatrix and addaniso to add an easy-axis anisotropy along the (hexagonal) *c* axis of magnitude K=0.0068 meV (remember the SpinW convention of negative values for easy-axis and positive values for easy-plane anisotropy).
4. Use optmagk to optimise the incommensurate propagation vector *k*, both using the 'kbase' option and without it (the propogation vector is expected to be along [1;1;0]). Run the calculations a few times and compare the ground state energy E of the different approaches as well as the inferred *k* vector. What do you notice?

Hint: optmagk using a global minimisation algorithm, the particle-swarm optimizer to determine the *k*-vector which yields the smallest eigenvalue of the Fourier transform of the Hamiltonian, in a space which can have many small minima.

1. What value do you find for the propagation vector (δ δ 0)? Does this match the experimental value δ=0.0045 found in the PRL paper? Note that the propogation vector determined by SpinW here assumes a “harmonic” cycloid – that is that the ordered structure can be described by a single phase term exp(i*k*.*r*). In BiFeO3 when the SIA is included, the cycloid structure is no longer harmonic, but instead acquires additional modulations because the spins prefer to align along the easy-axis direction and so “bunches” up at those points. Given then, what value of D should you choose to get the experimental propagation vector?

Hint: Use figure 5 of the PRL paper.

1. Calculate and plot the spin wave dispersion around the (111) Bragg peak (say from (0.98,0.98,1) to (1.02,1.02,1)) – what do you notice? What if you ran an optmagsteep optimisation? What changed and why?