# Excitations Data Analysis Training Course – Solutions 8

## Fitting the bcc-Fe dataset.

The practical today is just to fit the bcc-Fe dataset. For reference the function to calculate the ferromagnetic spin waves analytically is:

function y=Fe\_FM\_spinwaves(qh,qk,ql,en,pars,ff\_fun)

js=pars(1); delta=pars(2);

gam=pars(3); amp=pars(4);

sample\_temperature = 300;

omega0 = delta + (8\*js)\*(1-cos(pi\*qh).\*cos(pi\*qk).\*cos(pi\*ql));

Bose = en./ (1-exp(-11.602.\*en./sample\_temperature));

% Use DSHO model to give intensity:

y = amp .\* (Bose .\* (4.\*gam.\*omega0)./ ...

(pi.\*((en.^2-omega0.^2).^2 + 4.\*(gam.\*en).^2)));

% Scale by form factor function if it exists

if exist('ff\_fun', 'var')

y = y .\* ff\_fun(qh,qk,ql,en,[])';

end

to call it with the form factor included use something like:

ff\_fun = MagneticIons('Fe0').getFF\_calculator(w1);

mf = multifit\_sqw(w1);

mf = mf.set\_fun(@sqwfun);

mf = mf.set\_pin({[JS Delta Gamma Scale\_factor] ff\_fun});

otherwise it will calculate the intensity without the form factor. The model contains an exchange interaction (between nearest neighbours) JS, a gap Delta a width Gamma and a scaling factor.

1. Make the usual 1D cut of the bcc-iron dataset along [H H 0] and integrating from -1.1 to -0.9 in [-H,H,0], from -0.1 to 0.1 in [0,0,L] and from 100 to 120 meV. Plot it to see the expected peaks in the data (there should be ~5 clear peaks with a few smaller ones).
2. Set up a multifit\_sqw object to use the function above with the magnetic form factor. Include a linear background and fit your data. Try a few different fit runs with different starting parameters – what do you notice? Make a note of the most reasonable fitted parameters.

Answer: Depending on the starting parameters, the fitted parameters can vary wildly – by orders of magnitudes even though the fitted graph looks ok. It seems to that one cut is not enough to constrain the model (especially the exchange parameter JS).

1. Make a series of 1D cuts with the same integration range in Q as in step 1 but centred at 80, 100, 120 and 140 meV instead and put them into an array of sqw objects. Then fit them. What do you notice about the fitted parameters now?

Answer: The fitted parameters should be a bit more stable now and not change so much depending on initial values, but does still vary if you give it bad initial values. You should get something like: JS=35, Delta=5, Gamma=70, Scale=400.

1. Set up a SpinW model of ferromagnetic spin waves in bcc-iron – plot the structure and check you have a body-centred cubic structure with ferromagnetic spins on the iron sites, a nearest neighbour Heisenberg interaction and an easy-plane anisotropy perpendicular to z.
2. Make a copy of the multifit\_sqw object created in step 2 and modify it to use the horace\_sqw property of the Iron spinw object you just created instead of the handle to the model file. Set the 'mat' option to vary the nearest neighbour exchange interaction and the easy plane anisotropy. Also use the 'resfun','sho' option with the temperature set to 300K. Thus your input parameters should be something like [J1 K Gamma Temperature Scale\_factor]. Use the set\_free function to make sure that the Temperature parameter is not varied. Run the fit on a single cut and compare the fitted parameters to what you got with the analytical model. (You might also have to set 'hermit', false). What’s different and what’s the same? Why is this? Make a note of the fitted parameters for later comparison.

Answer: Using a good set of initial parameters (e.g. answer to step 3), you should get that the exchange is about a factor of S (=2.5 if you define it for Fe3+) smaller because now we are using a value of S different from one. You can change the model and use 'S',1 in addatom instead. If you set S=1, then it should be the same. The SIA will be much smaller than the gap (around an order of magnitude ) and Gamma should be about the same. The scale factor is also different because the intensity calculated by SpinW is not normalised unlike the analytic function.

1. Run the fit the array of cuts and compare it with the previous fit results.

Answer: The SpinW fitted parameters with Fe3+ (S=5/2) should be around J1=-14, K=0.5, Gamma=70, Scale=140.

Solution script:

%% Make a cut of the data and fit it with the analytical model

sqw\_file = '/mnt/ceph/auxiliary/excitations/iron.sqw';

proj = projaxes([1,1,0], [-1,1,0], 'type', 'rrr')

w1 = cut\_sqw(sqw\_file, proj, [-3,0.05,3], [-1.1,-0.9], [-0.1, 0.1], ...

[100,120]);

JS = 20; Delta=5; Gamma=5; Scale\_factor = 10;

ff\_fun = MagneticIons('Fe0').getFF\_calculator(w1);

mf = multifit\_sqw(w1);

mf = mf.set\_fun(@fedisp);

mf = mf.set\_pin({[JS Delta Gamma Scale\_factor] ff\_fun});

mf = mf.set\_bfun(@linear\_bg);

mf = mf.set\_bpin([0.1 0]);

[wfit, parfit] = mf.fit()

plot(w1); pl(wfit)

%% Make a set of 1D cuts and fit them to the analytical model

E\_cen = [80 100 120 140];

for ii = 1:numel(E\_cen)

cuts(ii) = cut\_sqw(sqw\_file, proj, [-3,0.05,3], [-1.1,-0.9], ...

[-0.1, 0.1], [-10,10]+E\_cen(ii));

end

JS = 50; Delta=1; Gamma=5; Scale\_factor = 10;

mf = multifit\_sqw(cuts);

mf = mf.set\_fun(@fedisp);

mf = mf.set\_pin({[JS Delta Gamma Scale\_factor] ff\_fun});

mf = mf.set\_bfun(@linear\_bg);

mf = mf.set\_bpin([0.1 0]);

[wfit, parfit] = mf.fit()

for ii = 1:numel(E\_cen)

plot(cuts(ii)); pl(wfit(ii)); keep\_figure;

end

%% Set up a SpinW model and fit it to a single dataset

spgr = 'I m -3 m'; a = 2.87;

fe = spinw();

fe.genlattice('lat\_const', [a a a], 'angled', [90 90 90], 'spgr', spgr)

fe.addatom('r', [0 0 0], 'S', 2.5, 'label', 'MFe3')

fe.gencoupling();

fe.addmatrix('label', 'J1', 'value', -1);

fe.addcoupling('mat', 'J1', 'bond', 1);

fe.addmatrix('label', 'K', 'value', diag([0 0 0.1]));

fe.addaniso('K');

fe.genmagstr('mode', 'direct', 'S', [0 0;

0 0;

1 1]);

JS = 50; Delta=1; Gamma=5; Temperature = 300; Scale\_factor = 10;

mf = multifit\_sqw(w1);

mf = mf.set\_fun(@fe.horace\_sqw);

mf = mf.set\_pin({[JS Delta Gamma Temperature Scale\_factor], ...

'mat', {'J1', 'K(3,3)'}, 'resfun', 'sho', ...

'formfact', true, 'hermit', false});

mf = mf.set\_free([1 1 1 0 1]);

mf = mf.set\_bfun(@linear\_bg);

mf = mf.set\_bpin([0.1 0]);

[wfit, parfit] = mf.fit()

plot(w1); pl(wfit)

%% Fit the SpinW model to a set of multiple cuts.

mf = multifit\_sqw(cuts);

mf = mf.set\_fun(@fe.horace\_sqw);

mf = mf.set\_pin({[JS Delta Gamma Temperature Scale\_factor], ...

'mat', {'J1', 'K(3,3)'}, 'resfun', 'sho', ...

'formfact', true, 'hermit', false});

mf = mf.set\_free([1 1 1 0 1]);

mf = mf.set\_bfun(@linear\_bg);

mf = mf.set\_bpin([0.1 0]);

[wfit, parfit] = mf.fit()

for ii = 1:numel(E\_cen)

plot(cuts(ii)); pl(wfit(ii)); keep\_figure;

end