User's guide : calculating EELS with feff85.

implementation by Kevin Jorissen
this document by Kevin Jorissen
this version : 05-15-2006
contact me at kevin.jorissen@ua.ac.be

1. Introduction

In dipole approximation, XAS and EELS are almost the same, so it is only natural to use a code like FEFF for EELS, too. The necessary changes to make this possible were implemented in this version of the FEFF code (FEFF 8.5). More precisely :
* XAS spectra are converted to EELS spectra (different energy-dependence of the cross-section)
* Relevant experimental parameters can be used as input – eg., beam energy, collection angle, etc.
* We add relativistic effects [Schattschneider] to the cross section, which is currently of much interest in the EELS community.

We stay true to the FEFF tradition of making the calculation easy on the user : one input file ("*feff.inp*"), one command ("*feff*"). True to the concept of ab initio computation, we want only structural information and experiment parameters as input.
So, FEFF 8.5 calculations of EELS spectra will proceed very much like XAS calculations with older versions. The only difference is that you will use specific input CARDs for EELS (described in section 2) and that you will see new files in your calculation directory (see section 4).

Throughout this document, it is assumed that you are somewhat familiar with the basics of FEFF, and that you have access to a FEFF user's guide. Together with the program and this document, you should have received an example input file *feff.inp* for the calculation of the C K edge of graphite.
If you are a beginning FEFF user and want to avoid confusion over advanced details, you can start by reading sections 2.1, 2.3 (first paragraph only), 3.1, 4.2, and 8.

2. How to do EELS – input and the EELS cards.

FEFF has a very user-friendly setup. There is one input file, *feff.inp*, in which you specify the system of interest and what you wish to calculate for this system. The FEFF user's guide explains how to use CARD's for this purpose. Below we describe the new cards for EELS. Just like the XAS spectrum is divided into a near-edge region (XANES) and an extended region (EXAFS) for every core edge in the spectrum, EELS spectra have ELNES (near edge structure) and EXELFS (extended structure). The two regions generally need to be computed in a different way, and therefore each has its own input card.

2.1. The ELNES card.

To tell the code to calculate EELS, and give it all relevant parameters, you need to add either the ELNES card or the EXELFS card to your feff.inp input file. The first line of those cards contains the same parameters as the XANES resp. the EXAFS card, and the next lines contain additional information. For ELNES, it looks like this :

```
ELNES [xkmax  xkstep vixan]
E   [aver [cross [relat]]] # beam energy in keV; optional parameters
[kx ky kz     # beam direction in the crystal frame]
β α          # collection semiangle, convergence semiangle (in mrad)
nr na        # q-integration mesh : radial size, angular size
dx dy        # position of the detector (x,y angle in mrad)
```

The parameters between square brackets are optional. For EXELFS, it looks like this :

```
EXELFS xkmax
E   [aver [cross [relat]]] # beam energy in keV; optional parameters
[kx ky kz     # beam direction in the crystal frame]
β α          # collection semiangle, convergence semiangle (in mrad)
nr na        # q-integration mesh : radial size, angular size
dx dy        # position of the detector (x,y angle in mrad)
```

Meaning of these parameters :

xkmax maximal k-value for the calculation

xkstep step size of the upper part of the k-mesh

vixan step size of the lower part of the k-mesh

aver 1 to calculate orientation averaged spectrum (eg. polycristalline sample, working at magic angle …) ; 0 to use specific sample to beam orientation (default);

cross 1 to use cross terms for the cross section (eg. xy or yx ; default); 0 to use only direct terms (eg., atom coordinates entered in symmetric coordinate frame ; assumed as default if aver is set to 1);

relat 1 to use relativistic formula for the cross-section (default and always recommended) ; 0 to use nonrelativistic formula (for checking against old results ; does not save any time!);

E energy of the electron beam in keV (typical values are 100-400 keV);

kx,ky,kz wave vector **k** of the incoming electron in the crystal frame (i.e., the Carthesian coordinate system in which the atom positions of the ATOMS card are given) ; in arbitrary units (only the direction, not the size of k is used) and to be omitted if aver is set to 1;

β the collection semiangle of the EELS detector in mrad (typical values are of the order of 1 mrad);

α the convergence semiangle of the incoming beam in mrad (typical values are of the order of 1 mrad);

nr, na the cross section is integrated over the values of the impuls transfer vector **q** = **k** – **k'** allowed by α and β ; the integration grid consists of nr concentric circles sampling a disc of radius α + β ; circle i contains na * (2i-1) points, making for nr*nr*na points in total (this is a nonphysical parameter and should be converged ; typical would be, eg., 5,2 ; the integration is quite fast ; only for small values of nr is it necessary to increase na above 1);

dx,dy the position of the detector in the scattering plane, specified by angles in mrad along x and y axis (the same as used in the ATOMS card) (typical values are 0.0, 0.0).

Xkmax, xkstep and vixan are exactly the same parameters as those used for XANES and EXAFS cards, and described in the FEFF user's guide.

The line giving beam orientation (kx,ky,kz) should only be present when an orientation sensitive spectrum is calculated. If an orientation averaged spectrum is to be calculated, that line should be omitted (or commented out).

As an example :

```
ELNES  # calculate elnes.
300          # beam energy in keV
0 1 0        # beam direction in the crystal frame
2.4 0.0      # collection semiangle, convergence semiangle (in mrad)
5 3          # q-integration mesh : radial size, angular size
0.0 0.0      # position of the detector (x,y angle in mrad)
```

simulates an experiment with a 300 keV beam hitting the sample along the y-axis. The detector is set in the forward direction and has a 2.4 mrad (semi-)opening ; the width of the incoming beam is 0 mrad. To do the integration over the detector aperture, 5*5*3=75 points are used. The calculation is relativistic and takes sample to beam orientation into account. Default settings are used for the energy/k – mesh.

Another example :

```
EXELFS 25    # calculate exelfs.
200 1 0      # beam energy in keV
0.4 0.5      # collection semiangle, convergence semiangle (in mrad)
10 1         # q-integration mesh : radial size, angular size
0.0 0.0      # position of the detector (x,y angle in mrad)
```

simulates an experiment with a 200 keV beam and an orientation averaged calculation (say the sample is isotropic). The detector is set in the forward direction (last line) and has a 0.4 mrad (semi-)opening ; the width of the incoming beam is 0.5 mrad. To do the integration over the detector aperture, 10*10*1=100 points are used. The calculation is relativistic and averages over sample to beam orientation. No cross terms are calculated (this option – the "0" on the second line could have been omitted, as it is the default for orientation averaged calculations). We use an energy grid up to k = 25 $Å^{-1}$.

2.2 Other EELS related cards.

Currently, there is only one additional new card : the MAGIC card.

```
MAGIC 20     # create a plot that shows the sp² ratio at 20 eV above
threshold.
```

The MAGIC card makes the code calculate cross sections as a function of collection angle at a particular energy loss. So, in addition to the energy-resolved output produced by the EELS card, the MAGIC card produces angle-resolved output.
The name MAGIC was chosen because the output of this card enables one to find the magic angle for a material very quickly : run the code for two different orientations, and see where the sp^2 - curves cross. This gives you the magic collection angle for the collection angle and EELS edge chosen in feff.inp.

2.3 Using traditional FEFF cards for EELS calculations.

* The EELS calculations use the machinery that is at the heart of FEFF to calculate spectra: either full multiple scattering, or path expansion (or both). So, the cards that we reviewed above must be combined with the appropriate card telling FEFF which of its algorithms to use.
For near edge structure (ELNES) we use :

FMS 7.0
ELNES …

to use the full multiple scattering.
For extended structure (EXELFS) one must use :

RPATH 8.0
EXELFS …

to use the path expansion.

Of course, the arguments of the FMS and RPATH cards may be set to different values than those given here. Additionally, one can specify other cards in the *feff.inp* file – see the FEFF user's guide for an exhaustive description.
Here, we only discuss a few issues relevant to EELS calculations in FEFF 8.5.

* For EELS calculations, the polarization vector is an internal variable that is set by the code itself. Therefore, the POLARIZATION card cannot be used. If the card is present, the code should give a warning and ignore it.
Similarly, the beam direction is given by the input parameters of an ELNES/EXELFS card. It is not allowed to use the ELLIPTICITY card when an EELS card is present. Again, if it is used, the code should complain and ignore it.

* One should obviously not combine ELNES or EXELFS with the XANES or EXAFS cards.

* The FPRIME and DANES cards are expected to work in combination with an ELNES/EXELFS card. However, it is not clear what the resulting output in *eels.dat* should mean. The combination of, say, DANES and ELNES could be used as a way of obtaining DANES for all polarizations with a single calculation, though, if one uses the *xmu.dat*-files produced by such a calculation (and ignores the probably meaningless *eels.dat*). (Note : please put the DANES card after the ELNES card, and make sure first lines of both cards are identical.)

* The CONTROL and PRINT cards do not have an additional field for the EELS module. Execution of the EELS module is governed by the presence of an ELNES/EXELFS card in feff.inp (or, on a lower level, by the corresponding switch in eels.inp). The output level can not be modified at this time, except through the MAGIC card.

3. How to do ELNES – running the code.

3.1 A full run.
Prepare the *feff.inp* file as described in section 2, and run feff! If you address the modules individually, then do not forget to add module *eels* to your calling sequence. A full calculation should consist of the steps
rdinp
pot
screen
xsph
ldos
fms
path
genfmt

ff2x
so2conv
eels
Your installation contains a script 'feff' so that you can run this whole sequence by just giving the command *feff*. The modules so2conv and screen and ldos are optional.

3.2 Changing something.
If you change something in the calculation of the material properties – such as atom positions, or the FMS radius … - you need to rerun all the traditional modules of FEFF (i.e., the whole sequence described above).

If you change from orientation averaged to orientation sensitive calculation, or from an orientation sensitive calculation without cross terms to one with cross terms, then you need to rerun rdinp and the sequence from fms until eels.

If you only change an experimental parameter – i.e., most of the parameters in the EELS and MAGIC cards – then you only have to rerun module *eels*, which is very fast (fast enough to couple it to fitting software, if you like). To do this, you either :
* change the ELNES/EXELFS card in *feff.inp*, run *rdinp*, run *eels* ;
* change the ELNES/EXELFS card and put CONTROL to 0 0 0 0 0 0 in *feff.inp*, and run *feff*
* edit *eels.inp* directly – but beware, this is a formatted file! – and run *eels*

4.3 Different behaviour.
When the EELS card is present, certain modules will behave different from what you're used to. Module *fms* will take a little longer, repeat some of its output, and produce a larger output file. Module *path* will take longer and produce more output files. Same goes for module *genfmt* and module *ff2x*.

All in all, the time increase should be quite modest, since most of the time goes into 1/ calculating self-consistent potentials in module 1, which is not affected at all ; and, 2/ inverting the matrix for FMS, which is still done only once – only the postprocessing in module *fms* takes longer.
A test on an ELNES calculation for a 100 atom FMS cluster in graphite showed negligible computation time increase compared to the corresponding XANES calculation (2 seconds longer on a total of about 9 minutes).

The basic novelty is in the treatment of the polarization. In non-EELS calculations, FEFF takes a polarization from the input (POLARIZATION and ELLIPTICITY cards), sets up a specific polarization matrix for this experimental situation, and calculates one corresponding spectrum. For EELS, however, we calculate a spectrum for every element of the polarization tensor (xx, xy, xz, yz, yy, …, zz). This gives us the whole absorption tensor (which is symmetric and has six independent components). The main computational effort is in setting up that tensor (modules rdinp through so2conv). Now, an EELS spectrum can be calculated in a second or less for specific experimental conditions (collection angle, orientation, beam energy, …).
A standard EELS calculation computes the whole absorption tensor. However, if the user asks for an orientation averaged calculation, only one element (corresponding to the trace) is calculated ; and if the user chooses not to have any cross terms, only xx, yy, and zz elements of the absorption tensor are calculated.

This explains the remarks made in section 4.2.

A last important difference to old calculations is that we disable the normalization of the spectrum at 50 eV above threshold. This was often annoying when several spectra need to be combined.

4. How to do EELS – files.

4.1 Input files.

* The file *feff.inp* contains all parameters related to EELS calculation and is the preferred way for non-expert users to set those parameters. After each change, module *rdinp* must be run to update *eels.inp*.
* The file *eels.inp* is read by the EELS module *eels* and determines what the code will actually do. Expert users can tweak this file directly. Basically, the file contains everything that's in the EELS and MAGIC cards. Of particular interest is the very first parameter, which determines whether *eels* is executed (=1) or not (=0), and also the parameters on the next line, which select the components (1-9) of the sigma tensor that will be calculated.
These parameters are very important because **other modules** will check for the presence of the eels.inp file and the values of these parameters to determine their course of action! People who have done EELS and then want to do something else in the same working directory may want to set the execution switch to 0 (or comment the EELS card in feff.inp and rerun module *rdinp*, which amounts to the same) to make sure none of the other modules bother about doing something special for EELS.

4.2 Output files.

* The file **eels.dat** contains the EELS spectrum. Its first column contains energy loss in eV, the second column the total spectrum, and the next columns contain the contribution to the total spectrum from each of the nine components of the cross section tensor.
* The file **magic.dat** is only written if the MAGIC card is present in *feff.inp* (and the corresponding switch is set to 1 in *eels.inp*). It contains the collection angle in rad, the pi to sigma ratio, the pi and sigma components of the spectrum, and the total spectrum ; all as a function of collection angle, evaluated at the energy loss set by the MAGIC card. Caution : this has not been tested in the presence of cross terms. E-mail the author in case of doubt.
* The file **logeels.dat** contains reports on the execution of the program. In particular, it contains a summary of the input options used. Most of the information in the file is also written to the screen during program execution.

4.3 Files of other modules.

Modules *xsph, fms, path, genfmt, ff2x* and *so2conv* are affected by the presence of an active *eels.inp* file and may behave accordingly :

4.3.1. Module 2 (ffmod2 or XSPH).

If an ELNES/EXELFS card is present, *xsph* sets the polarization tensor to the unit matrix before proceeding. A message is written to log2.dat.

4.3.2 Module 3 (ffmod3 or FMS).

If an ELNES/EXELFS card is present, *fms* adds to fms.bin all the requested components of the sigma tensor instead of just one. This is fully compatible with non-EELS calculations.

4.3.3 Module 4 (ffmod4 or PATH).

For each polarization component, a separate list.dat file is written (i.e., list.dat, list02.dat, etc.). The format of the files is unchanged.

4.3.4 Module 5 (ffmod5 or GENFMT).
For each polarization component, the corresponding listnn.dat is written and a separate feff.bin file is written (i.e., feff.bin, feff02.bin, etc.). The format of the files is unchanged.
4.3.5. Module 6 (ffmod6 or FF2X).
The module reads the large fms.bin and all the feffnn.bin files, and produces a xmunn.dat file containing the corresponding component of the sigma tensor (xmu.dat, xmu02.dat, …, xmu09.dat). Those files have the traditional xmu.dat – format. Similarly, chinn.dat – files are produced.
4.3.6 Module 8 (SO2CONV).
Again, a loop over all components of the sigma tensor : each xmu.dat file is opened and altered individually.

5. Guarantees?

This is a new software development. As such, bugs and various annoying little (or big) problems may occur. Contact the author if necessary.

The code was mostly tested for ELNES calculations on linux pc's with the ifort compiler. We believe it should also work on other systems. However, if you experience problems, or if you are unsure about an EXELFS calculation, I encourage you to contact the author.

Compatibility with the use of other input cards and with files from older FEFF versions has been attempted, but may not have been achieved in all cases. Please report problems.

6. Compilation :

* If you have received a single file, just compile it as you always have.
* If you have received a small number of files, one for each module, you compile them as usual, but now do one extra module. That is, you'll receive a file *eels_tot.f* that should be compiled to an executable *eels*.
* If you have received a whole directory tree with lots and lots of files, there should be a Makefile in which you can set compilation parameters (most importantly, the name of your compiler). Do this and run "make".

The source code for the *eels* module contains some fortran90. If you do not have a fortran90 compiler, you can obtain a fortran77 compatible version from us. Alternatively, if you have received the version with the Makefile (item #3 in the above list), you can set the parameter EELSDIR in the Makefile (to 'EELS' for f90, and to 'EELS77' for f77).

If you experience any problems, we'd appreciate your feedback and the opportunity to help you out.

7. Programming details in a nutshell.

A separate Programmer's Guide is available in the documentation of your FEFF 8.5 distribution. Here is a concise summary.

A central quantity in FEFF is the polarization tensor (ptz). In regular FEFF calculations, it is calculated from the polarization vector e (evec in the code) and the beam direction k (xivec in the code), and then the spectrum corresponding to that physical situation is calculated. Ptz is processed internally.
For EELS calculations, we want access to all 9 polarizations (6 of which are independent), giving us 9 partial spectra that we can sum, with weights depending on experimental conditions, to very quickly (as sample information is precomputed into those 9 components) assemble a physical EELS spectrum.
The modifications to the code fall into three categories :
- minor things, such as adding new cards, according to FEFF tradition ;
- modifying existing routines that handle the polarization tensor so that the full tensor is preserved ;
- adding a new module, called *eels*, for calculating the spectrum out of the 9 partial spectra.
More details on implementation are available in a separate Programmer's guide.

8. Questions? Remarks? Bugs?

Once more : let me know! kevin.jorissen@ua.ac.be

9. References.

* Relativistic Electron Energy Loss Spectra calculated in the Real Space Multiple Scattering approach, K. Jorissen, J.J. Rehr. In preparation.
This paper will be the official reference for EELS calculations with FEFF.

* Practical aspects of electron energy-loss spectroscopy (EELS) calculations using FEFF8 M.S. Moreno, K. Jorissen and J.J. Rehr, Micron, 2006. *(review article - in print)*
This paper contains practical hints on how to use FEFF8.2 for EELS calculations.

* Real-Space multiple-scattering calculation and interpretation of x-ray-absorption near-edge structure, A.L. Ankudinov, B. Ravel, J.J. Rehr, S.D. Conradson, Phys. Rev. B 58 (12) 7565, 1998.
The main reference for FEFF8.

* Anisotropic relativistic cross sections for inelastic electron scattering, and the magic angle, P. Schattschneider, C. Hebert, H. Franco, B. Jouffrey, Phys. Rev. B 72 045142, 2005.
A good reference for relativistic EELS theory.