

Notes for MULTIPLETY: A Python wrapper for Fortran-based multiplet calculations

(Dated: July 10, 2017)

MULTIPLETY provides a python wrapper for performing simple atomic calculations of RIXS spectra based Cowan's and the Racetrack codes. The approach is primarily for internal use at Brookhaven National Laboratory and largely mirrors what was implemented by Stavitski and de Groot [1], but with the flexibility to choose the x-ray polarization.

I. THEORY PRIMER

The multiplet method starts by computing atomic wavefunctions via the Hartree-Fock method using Cowan's codes [2]. Crystal fields are then introduced by mixing the atomic functions to generate the appropriate symmetry orbitals using Theo Thole's Racetrack code. *L*-edge RIXS involves two dipolar transitions via an intermediate state with a *p* core hole for example $2p^6 3d^8 \rightarrow 2p^5 3d^9 2p^6 3d^8 *$ where $*$ denotes an orbital *dd*-transition [3]. The matrix elements for these transitions are calculated for dipole operators appropriate for the incident and final x-ray polarization and RIXS is then computed using the Kramers-Heisenberg equation. The effects of electron-electron interactions are incorporated using the Slater-Condon parameters. F_{dd} and F_{pd} set the percentage reductions in the Coulomb repulsion between two *d* electrons and *p*-*d* electrons respectively [4, 5]. G_{pd} is the percentage reduction in the Coulomb exchange.

II. PRACTICALITIES

A typical experimental RIXS geometry is shown in Fig. 1. Vertically (σ) or horizontally (π) polarized x-rays are incident at θ_i on the sample, which has a *c*-axis surface normal. The incident x-ray electric field (\vec{e}) is always parallel to *b* for σ polarization. On the other hand, π incident x-ray polarization probes a mixture of $\vec{e} \parallel c$ and $\vec{e} \parallel a$ contributions. Such a geometry is handled via `generate_polarization($\theta_i, 2\theta$)`. Other geometries can be handled by constructing and passing `absorption.pol` and `emission.pol` to `calculate_rixs` by hand.

The core of the code happens in `atomic_calculation.py`, which creates the input files and runs the Fortran executables:

1. `rcn` and `rcn2` which calculate the atomic radial wavefunctions
2. `rcg` which calculates the atomic energy levels and spectra
3. `racetrack`: introduces the crystal field and magnetic exchange. Note that the states are given using Butler notation (see `atomic_calculation.py`).
4. `calc_exc`, which organizes the excitation energies calculated by `racetrack`.
5. `cleanup`, which moves all files to the selected folder.

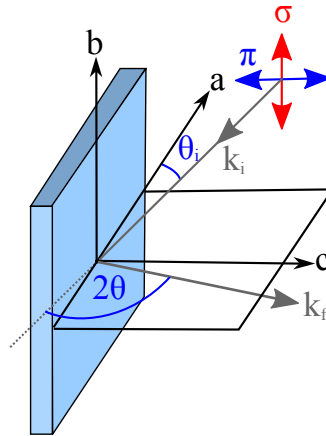


FIG. 1. RIXS scattering geometry. Linearly polarized x-rays (σ or π) are incident upon the sample at an angle θ_i . The scattered photons are collected at 2θ and measured as a function of final energy.

One RIXS process involves running this twice to compute the absorption (initial to intermediate states) transition and then the emission (intermediate to final state) processes. `process_rixs` reads the transition matrices `abs_racer.out.ora` and `emi_racer.out.ora` created in the first step, applies the x-ray polarization and Boltzmann factors and put all the transition into the Kramer-Heisenberg formula to calculate the spectrum.

-
- [1] Eli Stavitski and Frank M.F. de Groot, “The CTM4XAS program for EELS and XAS spectral shape analysis of transition metal *L*-edges,” [Micron](#) **41**, 687 – 694 (2010).
 - [2] R. D. Cowan, *The Theory of Atomic Structure and Spectra* (University of California Press, Berkeley, 1981).
 - [3] Luuk J. P. Ament, Michel van Veenendaal, Thomas P. Devereaux, John P. Hill, and Jeroen van den Brink, “Resonant inelastic x-ray scattering studies of elementary excitations,” [Rev. Mod. Phys.](#) **83**, 705–767 (2011).
 - [4] Frank De Groot and Akio Kotani, *Core level spectroscopy of solids* (CRC press, 2008).
 - [5] Frank de Groot, “Multiplet effects in x-ray spectroscopy,” [Coordination Chemistry Reviews](#) **249**, 31 – 63 (2005).