PyCrystalField: Software for Calculation, Analysis, and Fitting of Crystal Electric Field Hamiltonians

A. Scheie[1]

[1]*Institute for Quantum Matter and Department of Physics and Astronomy,*
Johns Hopkins University, Baltimore, MD 21218

(Dated: July 31, 2018)

We introduce PyCrystalField, a Python software package for calculating single-ion crystal electric field (CEF) Hamiltonians. This software calculate a CEF Hamiltonian *ab initio* from a point charge model for any transition or rare earth ion in either J basis or LS basis, calculate the energy spectrum and observables such as neutron spectrum and magnetization, and fit CEF Hamiltonians to any experimental data. Theory, implementation, and examples of its use are discussed.

I. INTRODUCTION

A material's electronic properties—which includes magnetic, optical, and electric qualities—are governed by electron orbitals. One of the key interactions which determines the ground state of an ion are the crystal electric field (CEF) interactions. These occur when Pauli exclusion and Coulomb repulsion from surrounding atoms shift the relative energies of valence electron orbitals. This can have dramatic effects on an ion's magnetism, causing cause single-ion anisotropies, quantum or classical spins, or even nonmagnetic singlets [1]. Unfortunately, CEF Hamiltonians are often very difficult to calculate. To aid in such calculations, we introduce PyCrystalField: a python library for calculating and fitting the CEF Hamiltonian of ions.

The theory behind CEF calculations has been worked out over 50 years ago [2, 3], but calculating the single-ion Hamiltonian remains a challenge for two reasons. First, the exact Hamiltonian cannot be calculated *ab initio*. One can use an approximate "point charge model", but these never reproduce measurements exactly [4, 5]. A quantitatively accurate CEF Hamiltonian must be fit to data (usually neutron scattering for heavy ions), and these fits must be done carefully. Second, the complexity of the CEF Hamiltonian depends upon the symmetry of the ion's environment, causing low-symmetry Hamiltonians to have over a dozen independent parameters. Because of this, determining a CEF Hamiltonian can be quite challenging.

Other software has been developed to do CEF calculations, such as FOCUS [6], McPhase [7], SIMPRE [8], or SPECTRE [9]. Useful as they are, these software are limited in either the type of ions they consider or the type of data they can fit. The ideal software would allow the user freedom to consider any ion in any environment to fit any kind of data. PyCrystalField achieves this.

II. THEORY

The general CEF Hamiltonian can be written as

$$\mathcal{H}_{CEF} = \sum_{n,m} B_n^m O_n^m \tag{1}$$

where O_n^m are the Stevens Operators [2, 3] and B_n^m are multiplicative factors called CEF parameters. These parameters can either be fit to experimental data or calculated from a point-charge model.

A. Point Charge Model

An approximate CEF Hamiltonian can be calculated by treating the surrounding ligands as point-charges and calculating the CEF Hamiltonian from Coulombic repulsion. PyCrystalField calculates B_n^m using the method outlined by Hutchings in ref. [3], where

$$B_n^m = -\gamma_{nm} q C_{nm} \langle r^n \rangle \theta_n \tag{2}$$

Here, γ_{nm} is a term calculated from the ligand environment expressed in terms of Tesseral Harmonics, q is the charge of the central ion (in units of e), C_{nm} are constants from spherical harmonics, $\langle r^n \rangle$ is the expectation value of the radial wavefunction (taken from ref. [4] for all rare-earth ions), and θ_n are constants associated with electron orbitals. Equation 2 is derived in Appendix A.

B. Neutron Cross Section

Neutron scattering is one of the easiest ways to see the energies and intensities of CEF transitions from one state to another. PyCrystalField calculates the powder-averaged neutron cross section with the dipole approximation

$$\frac{d^2\sigma}{d\Omega d\omega} = N(\gamma r_0)^2 \frac{k'}{k} F^2(\mathbf{Q}) e^{-2W(\mathbf{Q})}$$
$$p_n |\langle \Gamma_m | \hat{J}_\perp | \Gamma_n \rangle|^2 \delta(\hbar\omega + E_n - E_m) \tag{3}$$

[10], where $N(\gamma r_0)^2$ are normalization factors, k and k' are the incoming and outgoing neutron wavevectors, $F(\mathbf{Q})$ is the electronic form factor, $e^{-2W(\mathbf{Q})}$ is the Debye Waller factor, p_n is the Boltzmann weight, and $|\langle\Gamma_m|\hat{J}_\perp|\Gamma_n\rangle|^2 = \frac{2}{3}\sum_\alpha |\langle\Gamma_m|\hat{J}_\alpha|\Gamma_n\rangle|^2$ is computed from the inner product of total angular momentum J_α with the CEF eigenstates $|\Gamma_n\rangle$.

In practice, the delta function $\delta(\hbar\omega + E_n - E_m)$ in eq. 3 has a finite width due to the limited energy resolution of the instrument and the finite lifetime of the excited states. PyCrystalField approximates the instrument resolution with a Gaussian profile and the finite lifetimes with a Lorentzian profile, making the effective profile a convolution of the two (PyCrystalField uses a Voigt profile for the sake of computational efficiency). PyCrystalField allows the user to specify an arbitrary resolution function giving the Gaussian FWHM as a function of ΔE. (A useful approximation for time-of-flight spectrometers can be found in ref. [11].) Lorentzian FHWM is provided by the user and is treated as ΔE-independent.

PyCrystalField can calculate 2D (intensity vs ΔE) and 3D (intensity vs ΔE and Q) data sets, as shown in Fig. 1. For 3D data sets, the ion and Debye-Waller factor must be specified in order to calculate Q-dependence. (At the time of writing, the 3D neutron data calculations are only available for rare-earth ions.)

C. Magnetization and Susceptibility

It can be useful to compute susceptibility and magnetization from the CEF Hamiltonian. To do this, PyCrystalField calculates magnetization non-perturbatively as $M_\alpha = g_J\langle J_\alpha\rangle$, where $\langle J_\alpha\rangle = \sum_i e^{\frac{-E_i}{k_B T}}\langle i|J_\alpha|i\rangle /Z$ and $|i\rangle$ are the eigenstates of the effective Hamiltonian $\mathcal{H} = \mathcal{H}_{CEF} + \mu_B g_J \mathbf{B}\cdot\mathbf{J}$, where $\mathbf{B}$ is magnetic field. Susceptibility $\chi_{\alpha,\beta} = \frac{\partial M_\alpha}{\partial B_\beta}$ is calculated via a numerical derivative of magnetization with respect to field.

The advantage of the non-perturbative approach is that it can be extended to large magnetic fields and high temperatures without sacrificing accuracy.

D. Intermediate Coupling Scheme

For the rare earth ions, the crystal fields are much weaker than spin-orbit coupling and are generally treated as a perturbation to the spin-orbit Hamiltonian, operating on an effective spin J. However, for transition ions (and certain rare earths) where the spin-orbit Hamiltonian $\mathcal{H}_{SOC} = \lambda S\cdot L$ is of the same magnitude as $\mathcal{H}_{CEF}$, it is necessary to treat both CEF and spin-orbit coupling non-perturbatively in what is called the "intermediate coupling scheme" where $\mathcal{H}_{CEF}$ acts only on orbital angular momentum L. [1]

PyCrystalField can do all calculations in either the weak coupling scheme (J basis) or the intermediate coupling scheme (LS basis), although in the latter case the ion's spin-orbit coupling λ must be provided by the user. The advantage of the intermediate scheme is that it can account for intermultiplet transitions, but the disadvantage is that the eigenkets are usually harder to interpret.

In calculating the intermediate-coupling neutron spectrum, $|\langle\Gamma_m|\hat{J}_\perp|\Gamma_n\rangle|^2 = |\langle\Gamma_m|\hat{L}_\perp + \hat{S}_\perp|\Gamma_n\rangle|^2$ and in calculating magnetization $M_\alpha = g_J\langle J_\alpha\rangle = \langle L_\alpha + g_e S_\alpha\rangle$. Details of calculating a point-charge model in the intermediate scheme are given in Appendix B.

E. g-tensor

PyCrystalField can calculate the Lande g-tensor from an intermediate coupling CEF Hamiltonian. The g tensor is defined such that $M_\alpha(B_\beta) = g_{\beta\alpha}\langle J_\alpha\rangle = \langle L_\alpha + g_e S_\alpha\rangle$, where $\langle J_\alpha\rangle = \sum_i e^{\frac{-E_i}{k_B T}}\langle i|J_\alpha|i\rangle /Z$ and $|i\rangle$ are the eigenstates of the effective Hamiltonian $\mathcal{H} = \mathcal{H}_{CEF} + \mu_B g_J B_\beta J_\beta$, the magnetic field being in the β direction. Solving for g by writing $\langle J_\alpha\rangle = \langle L_\alpha + S_\alpha\rangle$,

$$g_{\alpha\beta} = \frac{\langle L_\alpha + g_e S_\alpha\rangle}{\langle L_\alpha + S_\alpha\rangle}. \tag{4}$$

which PyCrystalField uses to compute the g tensor from any intermediate coupling Hamiltonian.

III. IMPLEMENTATION

PyCrystalField is a collection of Python objects and functions which allow the user to build and fit CEF Hamiltonians with just a few lines of code. The ability to write scripts greatly streamlines the analysis process. It is available for download at https://github.com/asche1/PyCrystalField. Please report bugs to scheie@jhu.edu.

A. Building CEF Hamiltonian

The user may build a CEF Hamiltonian in two ways: (a) specifying the CEF parameters B_n^m, or (b) defining the locations of point-charges to build a point-charge model. PyCrystalField can build a point-charge CEF Hamiltonian for any ion in any environment, but shortcuts have been built-in for rare-earth 3+ ions.

For magnetic rare-earth 3+ ions with orbital moments (Pr^{3+}, Nd^{3+}, Pm^{3+}, Sm^{3+}, Tb^{3+}, Dy^{3+}, Ho^{3+}, Er^{3+}, Tm^{3+}, and Yb^{3+}) the ground state J, L, S, and $\langle r^n\rangle$ values are automatically read from internal tables by PyCrystalField. For other ions, L, S, and $\langle r^n\rangle$ must be provided by the user, with the radial integral in units of $(a_0)^n$. Because nearly all other ions have significant J multiplet mixing, calculations in the J basis are only available for rare-earth ions.

Table I. Calculated CEF parameters for a Yb^{3+} ion inside a cube of O^{2-} ions, by hand via the method in Hutchings [3], and then by PyCrystalField.

B_n^m (meV)	By Hand	PyCrystalField
B_4^0	0.1515035	0.1515035
B_4^4	0.7575173	0.7575173
B_6^0	0.001604	0.001604
B_6^4	-0.0336841	-0.0336841

B.   Fitting

PyCrystalField allows complete freedom in fitting data. The user must provide a global χ^2 function and PyCrystalField will minimize χ^2 by varying user-specified variables. The user may fit any set of variables to the data: B_n^m, point-charge location, effective charge of the point-charges, width of neutron scattering peaks, etc. In this way, the fit can be as constrained or as free as necessary. The user may fit to any calculated value: neutron data, magnetization, a list of eigenvalues, or susceptibility. In this way, any variable may be fit to any kind of data.

IV.   DISCUSSION

A.   Examples

In this section we present and discuss three examples to show the performance and capabilities of PyCrystalField.

1.   Comparison to Manual Calculation: Yb^{3+} inside a cube of O^{2-} ions

To confirm the accuracy of PyCrystalField, we compare its results to a result calculated by hand. We computed the weak-coupling point-charge CEF Hamiltonian for a Yb^{3+} ion surrounded by a cube of O^{2-} ions at a distance of $\sqrt{3}$ Å from the central ion. This system is easily calculable by hand, thanks to Hutchings (ref. [3]).

According to Hutchings Eq. (6.11), a point-charge cubic environment has a Hamiltonian of the form

$$H = B_4^0(O_4^0 + 5O_4^4) + B_6^0(O_6^0 - 21O_6^4) \quad (5)$$

where for eight-fold coordination (i.e., a cube of ligands) $B_4^0 = \frac{7}{18}\frac{|e|q}{d^5}\beta_J\langle r^4\rangle$ and $B_6^0 = -\frac{1}{9}\frac{|e|q}{d^7}\gamma_J\langle r^6\rangle$ (Hutchings Table XIV), and for Yb^{3+} $\beta_J = \frac{-2}{3\cdot5\cdot7\cdot11}$ and $\gamma_J = \frac{2^2}{3^3\cdot7\cdot11\cdot13}$ (Hutchings Table VI). The resulting CEF parameters are given in Table I. PyCrystalField's point-charge calculation agrees with the result perfectly.
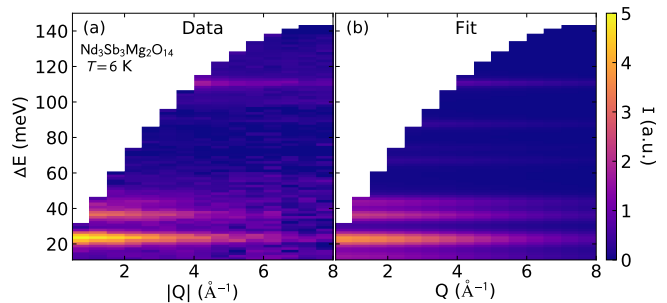

Figure 1. Sample of $Nd_3Sb_3Mg_2O_{14}$ scattering data at 6K (a) and fit to the data (b). A total of nine neutron data sets (with $E_i = 150$, 80, and 40 meV, and $T = 6$, 100, and 200 K) were fit to generate the scattering pattern in panel (b).

2.   Fits to Neutron Data: $Nd_3Sb_3Mg_2O_{14}$

PyCrystalField is useful in fitting very complex Hamiltonians to complex data sets. One example of this is the CEF fits to neutron data for $Nd_3Sb_3Mg_2O_{14}$ found in ref. [12]. A sample data set and fit are shown in Fig. 1.

In this case, a low-symmetry Hamiltonian with 13 independent CEF parameters was fit simultaneously to nine 3-D (I vs Q and ΔE) neutron scattering data sets at different temperatures and incident energies. The starting parameters were calculated from a point-charge model, and then effective ligand charges were fit to the data as an intermediate step to constrain the data. The result is an accurate fit to an otherwise underdetermined problem.

3.   Ab-initio Ni^{2+} Energy Levels

PyCrystalField can also be used to calculate the ground state energy splitting of an ion in a particular ligand environment. If the ligand positions are well-known, a point-charge model is accurate enough to predict the general character of the ion's energy spectrum. This is similar to a comparison with a Tanabe-Sugano diagram [1], but with far more quantitative accuracy.

An example of this is the calculation of the electron orbital energies of $Ni_2Mo_3O_8$, which has Ni^{2+} ions surrounded by distorted octahedral and tetrahedral coordination of O^{2-} ions [13]. PyCrystalField allowed for *ab initio* calculation of the energy spectrum of the valence orbitals in the intermediate coupling scheme. These calculations were used to qualitatively predict magnetic exchanges, single-ion anisotropies, and temperature-dependence in g-factor within $Ni_2Mo_3O_8$.

B.   Uses and Limitations

Although Crystal Field theory is very useful, it has pitfalls that users should be aware of. First, the point-charge model is imperfect because the ligand electrons

generally have valence p-orbitals, which are anisotropic and cause the effective charge to differ from the idealized point-charge model. Second, the success of fits involving many CEF parameters is highly dependent upon the starting values. There are various approaches to get around this, such as using the point-charge model to generate starting parameters or using a Monte Carlo method to generate many sets of starting parameters and fitting from each one. (In the latter case, a comparison to a point-charge model is still helpful to ensure that the CEF parameters are physically sensible.) Third, it is not possible to translate a set of CEF parameters from the weak-coupling scheme to the intermediate scheme or vice verse. This is because the parameters θ will differ (see Appendix B), requiring a re-calculation or re-fitting of the CEF parameters.

V. CONCLUSION

PyCrystalField is a general-purpose crystal field calculation package which allows the user to consider any type of data, any ion, and any ligand environment. It also allows the user to calculate observable quantities such as neutron scattering and bulk magnetization and susceptibility. It is accurate in reproducing simple point-charge calculations, it can fit highly asymmetric Hamiltonians to complex data sets, and it can make qualitative ground state predictions from a point-charge model.

PyCrystalField simplifies difficult CEF calculations in a way that makes them practical for more of the scientific community. Understanding the single-ion properties in a material is often critical to understanding the behavior of the whole, and CEF calculations are a critical part of this.

ACKNOWLEDGEMENTS

[1] A. Abragam and B. Bleaney, *Electron Paramagnetic Resonance of Transition Ions*, 1st ed. (Clarendon Press, Oxford, 1970).

[2] K. W. H. Stevens, Proceedings of the Physical Society. Section A **65**, 209 (1952).

[3] M. Hutchings, Solid State Physics, **16**, 227 (1964).

[4] S. Edvardsson and M. Klintenberg, Journal of Alloys and Compounds **275-277**, 230 (1998).

[5] J. Mesot and A. Furrer, "The crystal field as a local probe in rare earth based high-temperature superconductors," in *Neutron Scattering in Layered Copper-Oxide Superconductors*, edited by A. Furrer (Springer Netherlands, Dordrecht, 1998) pp. 335–374.

[6] P. Fabi, *FOCUS: an interactive crystal electric field parameter fitting package using neutron scattering data*, Tech. Rep. RAL-TR-95-023 (Rutherford Appleton Laboratory, 1995).

[7] M. Rotter, "Mcphase," (2013).

[8] J. J. Baldovi, S. Cardona-Serra, J. M. Clemente-Juan, E. Coronado, A. Gaita-arino, and A. Palii, Journal of Computational Chemistry **34**, 1961.

[9] A. Boothroyd, "Spectre – a program for calculating spectroscopic properties of rare earth ions in crystals," (2014).

[10] A. Furrer, J. Mesot, and T. Strässle, *Neutron scattering in condensed matter physics* (World Scientific, 2009).

[11] D. L. Abernathy, M. B. Stone, M. J. Loguillo, M. S. Lucas, O. Delaire, X. Tang, J. Y. Y. Lin, and B. Fultz, Review of Scientific Instruments **83**, 015114 (2012).

[12] A. Scheie, M. Sanders, J. Krizan, A. Christianson, R. Cava, and C. Broholm, to be submitted.

[13] J. Morey, A. Scheie, J. Sheckelton, C. M. Brown, and T. McQueen, ArXiv:1804.04776 (2018).

[14] B. Bleaney and K. W. H. Stevens, Reports on Progress in Physics **16**, 108 (1953).

Appendix A: Calculation of Point-charge CEF Parameters

We begin with Hutchings [3] Eq. 2.7:

$$V(r,\theta,\phi) = \sum_{n=0}^{\infty} \sum_{\alpha} r^n \gamma_{n\alpha} Z_{n\alpha}(\theta,\phi)$$

where $Z_{n\alpha}$ are tesseral harmonics, and

$$\gamma_{n\alpha} = \sum_{j=1}^{k} \frac{4\pi}{(2n+1)} q_j \frac{Z_{n\alpha}(\theta_j,\phi_j)}{R_j^{n+1}} \qquad (A1)$$

, summing over k ligands surrounding the central ion.

Now, we recognize that (according to Hutchings Eq. 5.3):

$$V(x,y,z) = \sum_{mn} A_n^m \frac{1}{-|e|} f_{nm}^c(x,y,z)$$

and according to Eq. (5.5), the Hamiltonian can be written as

$$\mathcal{H} = -|q| \sum_{i} V_i(x_i,y_i,z_i) = \sum_{i} \sum_{mn} A_n^m f_{nm}^c(x,y,z)$$

, summing over electrons and where $q = Ze$. Alternatively, we can write the Hamiltonian in terms of Stevens Operators (also following eq. 5.5):

$$\mathcal{H} = -|Ze| \sum_{i} \sum_{n,m} r^n \gamma_{nm} Z_{nm}(\theta_i,\phi_i)$$

$$= \sum_{i} \sum_{n,m} A_n^m f_{nm}^c(x_i,y_i,z_i) = \sum_{n,m} \underbrace{[A_n^m \langle r^n \rangle \theta_n]}_{B_n^m} O_n^m$$

$$\mathcal{H} = \sum_{n,m} B_n^m O_n^m$$

where θ_n is a multiplicative factor which is dependent on the ion ($\theta_2 = \alpha_J$; $\theta_4 = \beta_J$; $\theta_6 = \gamma_J$; see Table VI in Hutchings). Now, we solve the equations. We can look up $\langle r^n \rangle \theta_n$, we just need to find A_n^m.

Because $A_n^m f_{nm}^c(x_i,y_i,z_i) = -|e| r^n \gamma_{nm} Z_{nm}(\theta_i,\phi_i)$, we should be able to find A_n^m. Now it turns out that, according to Eq. 5.4 in Hutchings, $C_{nm} \times f_{nm}^c(x_i,y_i,z_i) = r^n Z_{nm}^c(\theta_i,\phi_i)$, where C_{nm} is a multiplicative factor in front of the Tesseral harmonics. Therefore,

$$A_n^m = -\gamma_{nm}^c |Ze| C_{nm}$$

. A closed-form expression for the constants C is very hard to derive, so I just used Mathematica to generate the prefactors to the spherical harmonics. They can be found in "TessHarmConsts.txt".

In the end, we can find the crystal field parameters B_n^m with the formula

$$B_n^m = A_n^m \langle r^n \rangle \theta_n$$

$$B_n^m = -\gamma_{nm} |Ze| C_{nm} \langle r^n \rangle \theta_n \qquad (A2)$$

We know $|e|$, Z (ionization of central ion), and the constants C_{nm}, we can look up θ_n in Hutchings, and we can look up $\langle r^n \rangle$ in the literature.

Units

The last trick here is the units. We want the Hamiltonian in units of meV. $\gamma_{n\alpha}$ is in units of $\frac{e}{\text{Å}^{n+1}}$, C_{nm} and θ_n are unitless, and $\langle r^n \rangle$ is in units of $(a_0)^n$. This means that B_n^m in the equation written above come out in units of $\frac{e^2}{\text{Å}}$. We want to convert this to meV.

To do this, we first recognize that we have to rewrite Hutchings eq. (II.2) with the proper prefactor for Coulomb's law: $W = \sum_i \frac{1}{4\pi\epsilon_0} q_i V_i$. Thus, our Hamiltonian becomes

$$\mathcal{H}_{CEF} = \sum_{nm} B(exp)_n^m O_n^m = \sum_{nm} \frac{1}{4\pi\epsilon_0} B(calc)_n^m O_n^m.$$

Now $\epsilon_0 = \frac{e^2}{2\alpha hc}$, so we can directly plug this into the equation:

$$B(exp)_n^m = \frac{1}{4\pi\epsilon_0} B(calc)_n^m = \frac{-1}{4\pi\epsilon_0} \gamma_{nm} |Ze| C_{nm} \langle r^n \rangle \theta_n$$

$$= \frac{-2\alpha hc}{4\pi e^2} \left(\gamma_{nm} C_{nm} \langle r^n \rangle \theta_n \right) e^2 Z \frac{a_0^n}{\text{Å}^{n+1}}$$

Plugging in the values, we get the equation used by PyCrystalField:

$$B_n^m = 1.440 \times 10^4 \, (0.5292)^n \, Z \left(\gamma_{nm} C_{nm} \langle r^n \rangle \theta_n \right) \text{meV} \qquad (A3)$$

Appendix B: θ_n in the Intermediate Scheme

When calculating crystal field levels from the point charge model, the θ_n constants are listed in Stevens (1952) [2] for the ground states of all the Rare Earth Ions in the J basis. But if we want to look at the state of the rare earth ion in the LS basis (intermediate coupling scheme), we must recalculate θ_n for each state.

General formulae for first two constants, $\theta_2 = \alpha$ and $\theta_4 = \beta$ are listed in ref. [14]. However, because they assume that such constants are only necessary for $3d$ group ions, they do not list θ_6. Therefore, we must derive it on our own, following the method of Stevens [2]. We take as an example the Sm^{3+} ion, which has five electrons in its f orbital valence shell and has a ground state of Sm^{3+} has $S = 5/2$, $L = 5$, and $J = 5/2$.

We know that

$$
\begin{aligned}
V_6^0 &= \Sigma(231z^6 - 315z^4r^2 + 105z^2r^4 - 5r^6) \\
&= \gamma_{6,0}C_{6,0}\langle r^6 \rangle \theta_6 O_6^0
\end{aligned} \tag{B1}
$$

where the Stevens operator

$$
\begin{aligned}
O_6^0 =& 231L_z^6 - (315X - 735)L_z^4 + (105X^2 - 525X + 294)L_z^2 \\
& - 5X^3 + 40X^2 - 60X
\end{aligned} \tag{B2}
$$

and $X = L(L+1)$. Now to calculate θ_6, we pick an eigenstate (in this case, $|L = 5, S = 5/2, m_l = 5, m_s = -5/2\rangle$) and calculate the expectation value in terms of the Stevens Operators, and then calculate it in terms of the individual electron wave functions. Then we set the two results equal to each other to find the multiplicative factor necessary to make the Stevens Operator result match the wave function integral.

Stevens Operators: This is a straightforward calculation from eq. B2. Letting $G_6^0 = \langle r^6 \rangle \theta_6 O_6^0$,

$$
\begin{aligned}
\langle L = 5,\, m_l = 5|G_6^0|L = 5,\, m_l = 5\rangle \\
= \theta_6 \langle r^6 \rangle 37800
\end{aligned} \tag{B3}
$$

Individual electron wave functions: To carry out this calculation we write $|m_l = 5, m_s = -\frac{5}{2}\rangle$ as a product state of the individual electron wave functions in the valence shell, which have $l = 3$, $m = 3, 2, 1, 0, -1$ (adding up to $L = 5$). Now we just recompute V_6^0 in the basis of individual electrons: $V_6^0 = \gamma O_6^0$, where $m_l = m$ and $L = l$:

$$
\begin{aligned}
\langle L = 5,\, m_l = 5|V_6^0|L = 5,\, m_l = 5\rangle \\
= \{3, 2, 1, 0, -1\}V_6^0\{3, 2, 1, 0, -1\} \\
= \gamma(180 - 1080 + 2700 - 3600 + 2700) = 900\gamma. \tag{B4}
\end{aligned}
$$

We introduced γ to account for the unknown scaling factor. (The similarity to $\gamma_{6,0}$ is unfortunate, because the γs are unrelated. Nevertheless, we follow the notation of Stevens in this appendix.) So now we can relate eq. B3 to eq. B4 so $\beta \langle r^6 \rangle = \frac{900}{37800}\gamma = \frac{1}{42}\gamma$. Now we find γ by integrating the wave functions themselves. Let's pick the state $l = 3$, $m = 3$:

$$
\begin{aligned}
\langle l = 3, m = 3|V_6^0|l = 3, m = 3\rangle = 180\gamma \\
= \int_0^{2\pi}\int_0^\pi Y_3^{3*}(231z^6 - 315z^4r^2 + 105z^2r^4 - 5r^6)Y_3^3 \sin\theta d\theta d\phi \\
= \frac{-80}{429}r^6. \tag{B5}
\end{aligned}
$$

This means that $\gamma = -\frac{80}{180 \cdot 429}r^6 = -\frac{4}{3861}r^6$, so that

$$
\theta_6 \langle r^6 \rangle = -\frac{1}{42}\gamma = -\frac{4}{42 \cdot 3861}r^6 \tag{B6}
$$

and θ_6 for Sm^{3+} is

$$
\theta_6 = -\frac{4}{42 \cdot 3861} \tag{B7}
$$

For each ion, we have to calculate $\langle L,\, m_l|G_6^0|L,\, m_l\rangle$ and then calulate $\{3, 2...\}V_6^0\{3, 2...\}$. Fortunately, the result $\gamma_6 = \frac{4}{3861}r^6$ holds for all rare earth ions. Thus, we arrive at the general formula for θ_6:

$$
\theta_6 = \frac{\{3, 2...\}V_6^0\{3, 2...\}}{\langle L,\, m_l|O_6^0|L,\, m_l\rangle}\frac{4}{3861}. \tag{B8}
$$