FROM RESEARCH TO INDUSTRY

www.cea.fr

Ab-initio Computation of Raman spectra

within the DFPT formalism coupled with the PAW method

Implementation in the ABINIT project

Lucas Baguet, Marc Torrent

CEA, DAM, DIF, F-91297, Arpajon, France

May 11$^{\text{th}}$, 2017

Table of Contents

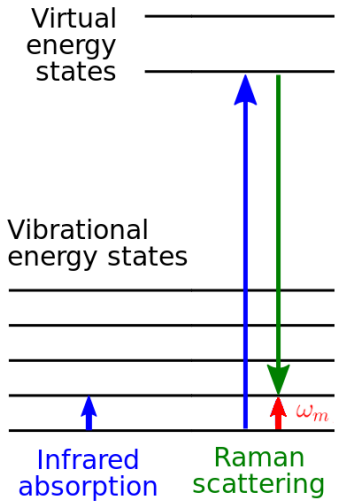
1. Theory of non-resonant Raman scattering
   - *Basics*
   - *Measured intensity*
   - *Raman intensity / Raman tensor*
   - *Derivative of the electric susceptiblity*
   - *Macroscopic properties from total energy derivatives*

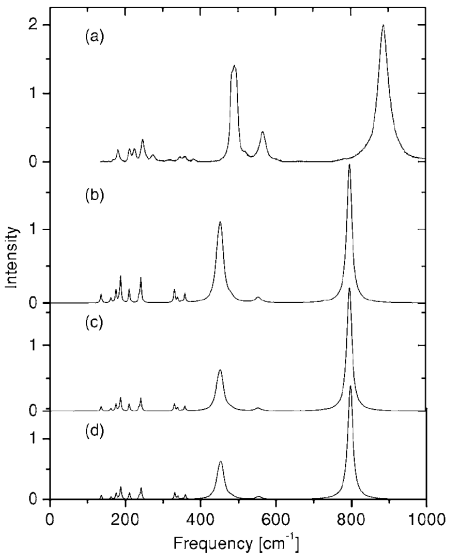
2. Implementation of 3^{rd} order DFPT in ABINIT
   - *Previous works / DFPT and PAW*
   - *$2n+1$ theorem with electric field perturbations*
   - *Work flow for computing third derivatives*
   - *The 1^{st} and 2^{nd} order Sternheimer equations*
   - *Test of the 2^{nd} order Sternheimer implementation*
   - *Expression of third derivatives*
   - *Structure of the nonlinear routine*
   - *Comparison of 3^{rd} derivatives of E*

3. Conclusion

1. Theory of non-resonant Raman scattering

Basics





Stokes mecanism : inelastic scattering of an *incident photon* interacting with a *phonon*.

$\Rightarrow$ Frequency shift between the incident and scattered light : $\Delta\omega = \omega_m$

We treat only *non resonant* scattering : $E_\gamma < E_{\text{gap}}$

$\Rightarrow$ Relevant only for insulators

Measured intensity $\approx$ sum of lorentzian functions :

$$I(\omega) \approx \sum_m^{\text{active}} \frac{I_m}{\pi} \frac{C_m}{(\omega - \omega_m)^2 + C_m^2}$$

Only *active* phonon modes contribute :

- They have a null wavevector ($\mathbf{q} = 0$)
- They follow selection rules depending on crystal symmetries and photon polarization

$$I(\omega) \approx \sum_{m}^{\text{active}} \frac{I_m}{\pi} \frac{C_m}{(\omega - \omega_m)^2 + C_m^2}$$

Peak properties :

- position : ω_m phonon frequency
  $\Rightarrow 2^{\text{nd}}$ derivatives of the total energy E
- intensity : I_m
  $\Rightarrow 3^{\text{rd}}$ derivatives of the total energy E
- width : C_m
  $\Rightarrow 3^{\text{rd}}$ and 4^{th} derivatives of the total energy E

In green : available in ABINIT (and other codes !) for norm conserving (NC) or projector augmented-wave (PAW) pseudo-potentials

In red : subject of the present work, implementation of new NC/PAW routines in ABINIT

For a single crystal :

$$I_m \propto \frac{(\omega_0 - \omega_m)^4}{\omega_m} |\mathbf{e}_S . \alpha^m . \mathbf{e}_I|^2$$

For polycrystals : mean over all possible angles between $\mathbf{e}_I$ and $\mathbf{e}_S$.

- ω_0/ω_m : light / phonon frequency
- $\mathbf{e}_I/\mathbf{e}_S$: incident / scattered photon polarization direction
- α^m : Raman tensor, depending on crystal properties

Raman tensor :

$$\alpha_{ij}^m = \sqrt{\Omega_0} \sum_{\kappa,\beta} \frac{d\chi_{ij}}{d\tau_{\kappa\beta}} u_m(\kappa\beta)$$

- Ω_0 : crystal volume
- $u_m(\kappa\beta)$: eigendisplacement m of atom κ along direction β
- $\frac{d\chi_{ij}}{d\tau_{\kappa\beta}}$: derivative of the electric susceptiblity with respect to the displacement of atom κ along direction β (at $\tau_{\kappa\beta} = 0$)

$$\frac{d\chi_{ij}}{d\tau_{\kappa\beta}} = \frac{\partial\chi_{ij}}{\partial\tau_{\kappa\beta}} + \sum_k \frac{\partial\chi_{ij}}{\partial\mathcal{E}_k}\frac{\partial\mathcal{E}_k}{\partial\tau_{\kappa\beta}} \qquad \chi_{ij}(\mathcal{E}) = \chi^{(1)}_{ij} + \sum_k \chi^{(2)}_{ijk}\mathcal{E}_k + O(\mathcal{E}^2)$$

For transverse optical modes (TO) :

$$\alpha^{m_{TO}}_{ij} = \sqrt{\Omega_0}\sum_{\kappa,\beta}\frac{\partial\chi^{(1)}_{ij}}{\partial\tau_{\kappa\beta}}u_m(\kappa\beta)$$

For longitudinal optical modes (LO) :

$$\alpha^{m_{LO}}_{ij} = \sqrt{\Omega_0}\sum_{\kappa,\beta}\left(\frac{\partial\chi^{(1)}_{ij}}{\partial\tau_{\kappa\beta}} - \frac{8\pi}{\Omega_0}\sum_k \chi^{(2)}_{ijk}\sum_l \epsilon^{-1}_{kl}Z^*_{\kappa\beta,l}\right)u_m(\kappa\beta)$$

- ϵ^{-1}_{ij} : inverse of the dielectric tensor ($\epsilon_{ij} = 4\pi\chi^{(1)}_{ij} - 1$)
- $Z^*_{\kappa\beta,l}$: Born effective charges tensor

Notation :

$$X^{(\lambda_1)} \equiv \left.\frac{dX}{d\lambda_1}\right|_{\lambda_1=0} \qquad X^{(\lambda_1\lambda_2)} \equiv \left.\frac{d^2X}{d\lambda_1 d\lambda_2}\right|_{\lambda_1=\lambda_2=0} \qquad \dots$$

Here we consider derivatives of the total energy E with respect to :

- $\tau_{\kappa\alpha}$: atomic displacement ($\mathbf{q} = 0$)
- $\mathcal{E}_i$: uniform electric field

2$^{\text{nd}}$ derivatives :

$$E^{(\tau_{\kappa\alpha}\tau_{\kappa'\beta})} = C_{\kappa\alpha,\kappa'\beta} \Rightarrow D_{\kappa\alpha,\kappa'\beta}(\mathbf{q}=0) \Rightarrow \{\omega_m, u_m(\kappa,\beta)\}$$

$$E^{(\mathcal{E}_i\mathcal{E}_j)} = -\Omega_0\chi^{(1)}_{ij} \qquad E^{(\tau_{\kappa\beta}\mathcal{E}_i)} = -Z^*_{\kappa\beta,i}$$

3$^{\text{rd}}$ derivatives :

$$E^{(\tau_{\kappa\beta}\mathcal{E}_i\mathcal{E}_j)} = -\Omega_0\frac{\partial\chi^{(1)}_{ij}}{\partial\tau_{\kappa\beta}} \qquad E^{(\mathcal{E}_i\mathcal{E}_j\mathcal{E}_k)} = -2\Omega_0\chi^{(2)}_{ijk}$$

2. Implementation of 3rd order DFPT in ABINIT

Previous works / DFPT and PAW

In ABINIT, the PEAD (Perturbation Expansion After Discretization) for 3rd derivatives has been implemented by Veithen *et al* (2005).

- Mixing of DFPT and Berry phase formalisms
- Available only for NC pseudos, **difficult to adapt to PAW**

The advantages of PAW pseudopotentials are :

- cutoff energy is reduced (as with *ultra-soft* pseudopotentials)
- electronic properties around ionic cores are well described
- precision similar to *all-electron* methods
- one can use LDA+U formalism straightforwardly

$\Rightarrow$ To get rid of the Berry phase formalism, we need to compute third derivatives of E in a **"full DFPT"** way (like in the work of Miwa, 2011).

2. Implementation of 3rd order DFPT in ABINIT

$2n+1$ *theorem with electric field perturbations*

From the $2n+1$ theorem, to compute 3rd derivatives of the energy requires only ground state wave functions $|\Psi^{(0)}_{n\mathbf{k}}\rangle$ and its first derivatives :

$$|\Psi^{(\tau_{\kappa\beta})}_{n\mathbf{k}}\rangle \qquad |\Psi^{(\mathcal{E}_i)}_{n\mathbf{k}}\rangle$$

$\Rightarrow$ They are obtained solving 1st order Sternheimer equations.

However, the electric field perturbation brings a difficulty :

$$V(\mathcal{E}) = \mathcal{E}.\mathbf{r} = \mathcal{E}.\text{“ } i\nabla_{\mathbf{k}} \text{ ”}$$

For 2nd derivatives of E, one also needs : $|\Psi^{(k_i)}_{n\mathbf{k}}\rangle$

and for 3rd derivatives : $|\Psi^{(k_i\mathcal{E}_j)}_{n\mathbf{k}}\rangle$, $|\Psi^{(k_ik_j)}_{n\mathbf{k}}\rangle$

$\Rightarrow$ We need to solve two 2nd order Sternheimer equations !

Note : in PEAD, $\nabla_{\mathbf{k}}$ is discretized $\Rightarrow$ no need of $|\Psi^{(k_i\mathcal{E}_j)}_{n\mathbf{k}}\rangle, |\Psi^{(k_ik_j)}_{n\mathbf{k}}\rangle$

2. Implementation of 3rd order DFPT in ABINIT

Work flow for computing third derivatives

Previous implementation using PEAD :

- Dataset 1 : ground state
  $\Rightarrow \Psi^{(0)}_{n\mathbf{k}}$, $\epsilon^{(0)}_{n\mathbf{k}}$, $n^{(0)}(\mathbf{r})$, . . .
- Dataset 2 : 1st order Sternheimer (and 2de derivatives of E)
  rfddk $= 1$, *rfphon* $= 1$, *rfelfd* $= 1$
  $\Rightarrow \Psi^{(k_i)}_{n\mathbf{k}}$, $\Psi^{(\tau_{\kappa\beta})}_{n\mathbf{k}}$, $\Psi^{(\mathcal{E}_i)}_{n\mathbf{k}}$
- Dataset 3 : 3rd derivatives of E
  optdriver=5
  d3e_pert1_phon=1, d3e_pert1_elfd=1

2. Implementation of 3rd order DFPT in ABINIT

Work flow for computing third derivatives

New "full DFPT" implementation :

- Dataset 1 : ground state
  $\Rightarrow \Psi^{(0)}_{n\mathbf{k}}$, $\epsilon^{(0)}_{n\mathbf{k}}$, $n^{(0)}(\mathbf{r})$, . . .
- Dataset 2 : 1st order Sternheimer (and 2de derivatives of E)
  rfddk $= 1$, *rfphon* $= 1$, *rfelfd* $= 1$
  $\Rightarrow \Psi^{(k_i)}_{n\mathbf{k}}$, $\Psi^{(\tau_{\kappa\beta})}_{n\mathbf{k}}$, $\Psi^{(\mathcal{E}_i)}_{n\mathbf{k}}$
- Dataset 3 : 2nd order Sternheimer
  rf2_dkdk $= 1$
  $\Rightarrow \Psi^{(k_i k_j)}_{n\mathbf{k}}$
- Dataset 4 : 2nd order Sternheimer
  rf2_dkde $= 1$
  $\Rightarrow \Psi^{(k_i E_j)}_{n\mathbf{k}}$
- Dataset 5 : 3rd derivatives of E
  optdriver=5, usepead=0 (default : 1)
  d3e_pert1_phon=1, d3e_pert1_elfd=1

2. Implementation of 3rd order DFPT in ABINIT

The 1st and 2nd order Sternheimer equations

The Sternheimer equations have the form $Ax = b$ where $A^\dagger = A \Rightarrow$ Solved with a conjugate gradient algorithm.

$$1^{\text{st}} \text{ order : } \left(P^c\right)^\dagger \left(\tilde{H}^{(0)} - \epsilon_{n\mathbf{k}} S^{(0)}\right) P^c |\Psi_{n\mathbf{k}}^{(\lambda_1)}\rangle = -\left(P^c\right)^\dagger \left(\tilde{H}^{(\lambda_1)} - \epsilon_{n\mathbf{k}} S^{(\lambda_1)}\right) |\Psi_{n\mathbf{k}}^{(0)}\rangle$$

$$\begin{aligned} 2^{\text{nd}} \text{ order : } \left(P^c\right)^\dagger \left(\tilde{H}^{(0)} - \epsilon_{n\mathbf{k}} S^{(0)}\right) P^c |\Psi_{n\mathbf{k}}^{(\lambda_1\lambda_2)}\rangle &= -\left(P^c\right)^\dagger \left(\tilde{H}^{(\lambda_1\lambda_2)} - \epsilon_{n\mathbf{k}} S^{(\lambda_1\lambda_2)}\right) |\Psi_{n\mathbf{k}}^{(0)}\rangle \\ &- \left(P^c\right)^\dagger \left(\tilde{H}^{(\lambda_1)} - \epsilon_{n\mathbf{k}} S^{(\lambda_1)}\right) |\Psi_{n\mathbf{k}}^{(\lambda_2)}\rangle - \left(P^c\right)^\dagger \left(\tilde{H}^{(\lambda_2)} - \epsilon_{n\mathbf{k}} S^{(\lambda_2)}\right) |\Psi_{n\mathbf{k}}^{(\lambda_1)}\rangle \\ &+ \sum_m^{\text{occ}} \Lambda_{mn}^{(\lambda_1)} \left(P^c\right)^\dagger \left(S^{(\lambda_2)} |\Psi_{m\mathbf{k}}^{(0)}\rangle + S^{(0)} |\Psi_{m\mathbf{k}}^{(\lambda_2)}\rangle\right) + \sum_m^{\text{occ}} \Lambda_{mn}^{(\lambda_2)} \left(P^c\right)^\dagger \left(S^{(\lambda_1)} |\Psi_{m\mathbf{k}}^{(0)}\rangle + S^{(0)} |\Psi_{m\mathbf{k}}^{(\lambda_1)}\rangle\right) \end{aligned}$$

The Sternheimer equations (1st and 2nd orders) are solved in the same routine (*respfn*) :

call *respfn*
 call *dfpt_looppert*
  do *ipert* = 1, *mpert* ← "*rf2_dkdk*" : ipert=natom+10 ($k_i k_j$) , "*rf2_dkde*" : ipert=natom+11 ($k_i \mathcal{E}_j$)
   call *dfpt_scfcv*
    do *istep* = 1, *nstep* ← for ipert=natom+10/+11 : nstep=1 (as for natom+1 (k_i))
     call *dfpt_vtorho*
      do *isppol* = 1, *nsppol*
       do *ikpt* = 1, *nkpt*
        call *dfpt_vtowfk*
         if (ipert==natom+10 or +11) call *rf2_init* ← Compute and store *b* for every bands
         do iband=1,nband
          call *dfpt_cgwf* ← Solve $Ax = b$ for one band (if ipert/=natom+10/+11 : *b* is computed on the fly)

We compare our derivatives of wavefunctions to ones obtained with a finite difference method on **k** (FDM,**k**) :

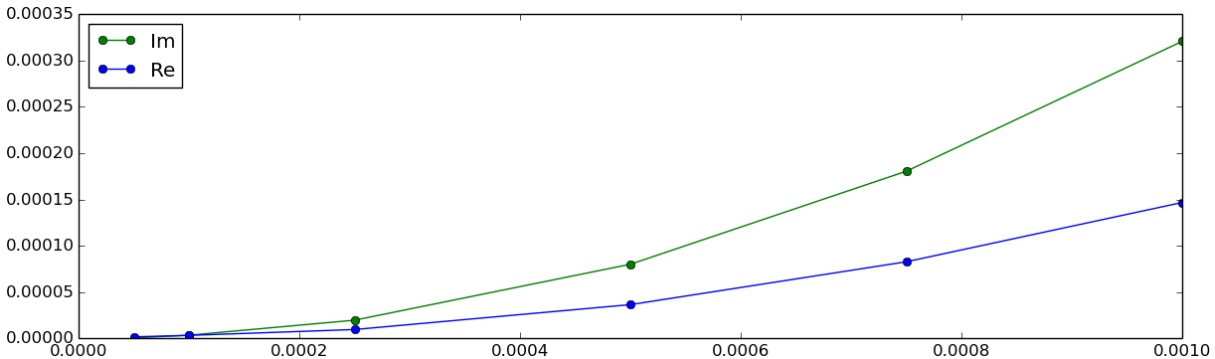
$$|\Delta\Psi^{(k_i)}_{n\mathbf{k}}\rangle \equiv \frac{|\Psi^{(0)}_{n\mathbf{k}+\Delta\mathbf{k}_i}\rangle - |\Psi^{(0)}_{n\mathbf{k}-\Delta\mathbf{k}_i}\rangle}{2\Delta k}$$

$\Rightarrow$ *gauge-dependent* quantities : $|\Delta\Psi^{(k_i)}_{n\mathbf{k}}\rangle$ is ill-defined.

Solution

To compare the derivative of *density matrices* instead.

$X \equiv \rho^{(k_i\mathcal{E}_j)}_{\mathbf{k}}(\mathbf{G},\mathbf{G}')$ $\qquad |X^{\mathrm{DFPT}} - X^{\mathrm{FDM},\mathbf{k}}|/|X^{\mathrm{DFPT}}|$



Δk (FDM parameter)

2. Implementation of 3rd order DFPT in ABINIT

Expression of third derivatives

Now we have to implement third order derivative of E, for both norm conserving and PAW pseudo potentials.

For two electric fields and one atomic displacement $(\lambda_1\lambda_2\lambda_3) = (\tau_{\kappa\beta}\mathcal{E}_i\mathcal{E}_j)$:

$$\begin{aligned}
\tilde{E}^{(\lambda_1\lambda_2\lambda_3)} = & \sum_{\mathbf{k}\in\mathcal{B}}\sum_{n}^{\mathrm{occ}}\left(\langle\psi_{n\mathbf{k}}^{(\lambda_1)}|H^{(\lambda_2)} - \epsilon_{n\mathbf{k}}^{(0)}S^{(\lambda_2)}|\psi_{n\mathbf{k}}^{(\lambda_3)}\rangle + \langle\psi_{n\mathbf{k}}^{(0)}|H_{\mathrm{KV}}^{(\lambda_1\lambda_2)}|\psi_{n\mathbf{k}}^{(\lambda_3)}\rangle + \langle\psi_{n\mathbf{k}}^{(\lambda_1)}|H_{\mathrm{KV}}^{(\lambda_2\lambda_3)}|\psi_{n\mathbf{k}}^{(0)}\rangle\right) \\
& - \sum_{\mathbf{k}\in\mathcal{B}}\sum_{n,m}^{\mathrm{occ}}\Lambda_{nm\mathbf{k}}^{(\lambda_1)}\left(\langle\psi_{n\mathbf{k}}^{(\lambda_2)}|S^{(0)}|\psi_{m\mathbf{k}}^{(\lambda_3)}\rangle + \langle\psi_{n\mathbf{k}}^{(\lambda_2)}|S^{(\lambda_3)}|\psi_{m\mathbf{k}}^{(0)}\rangle + \langle\psi_{n\mathbf{k}}^{(0)}|S^{(\lambda_2)}|\psi_{m\mathbf{k}}^{(\lambda_3)}\rangle\right) \\
& + \frac{1}{6}\int d\mathbf{r}\, E_{\mathrm{xc}}'''[\mathbf{r},\tilde{n}^{(0)}]\tilde{n}^{(\lambda_1)}(\mathbf{r})\tilde{n}^{(\lambda_2)}(\mathbf{r})\tilde{n}^{(\lambda_3)}(\mathbf{r}) \\
& + \frac{1}{6}\sum_a\int_{\Omega_a} d\mathbf{r}\left(E_{\mathrm{xc}}'''[\mathbf{r},n_a^{(0)}]n_a^{(\lambda_1)}(\mathbf{r})n_a^{(\lambda_2)}(\mathbf{r})n_a^{(\lambda_3)}(\mathbf{r}) - E_{\mathrm{xc}}'''[\mathbf{r},\tilde{n}_a^{(0)}]\tilde{n}_a^{(\lambda_1)}(\mathbf{r})\tilde{n}_a^{(\lambda_2)}(\mathbf{r})\tilde{n}_a^{(\lambda_3)}(\mathbf{r})\right) \\
& + \frac{1}{2}\int d\mathbf{r}\, E_{\mathrm{Hxc}}''[\mathbf{r},\tilde{n}^{(0)}]\tilde{n}^{(\lambda_1)}(\mathbf{r})\underline{\tilde{n}^{(\lambda_2\lambda_3)}}(\mathbf{r}) \\
& + \frac{1}{2}\sum_a\int_{\Omega_a} d\mathbf{r}\left(E_{\mathrm{Hxc}}''[\mathbf{r},n_a^{(0)}]n_a^{(\lambda_1)}(\mathbf{r})\underline{n_a^{(\lambda_2\lambda_3)}}(\mathbf{r}) - E_{\mathrm{Hxc}}''[\mathbf{r},\tilde{n}_a^{(0)}]\tilde{n}_a^{(\lambda_1)}(\mathbf{r})\underline{\tilde{n}_a^{(\lambda_2\lambda_3)}}(\mathbf{r})\right)
\end{aligned}$$

There are a lot of terms to implement, but only few terms need new routines or modifications of existing ones.

$$E^{(\lambda_1\lambda_2\lambda_3)} = \frac{1}{6}\left(\tilde{E}^{(\lambda_1\lambda_2\lambda_3)} + \tilde{E}^{(\lambda_1\lambda_3\lambda_2)} + \tilde{E}^{(\lambda_2\lambda_1\lambda_3)} + \tilde{E}^{(\lambda_2\lambda_3\lambda_1)} + \tilde{E}^{(\lambda_3\lambda_1\lambda_2)} + \tilde{E}^{(\lambda_3\lambda_2\lambda_1)}\right)$$

2. Implementation of 3rd order DFPT in ABINIT

Structure of the nonlinear routine

Third order derivatives are computed in the nonlinear routine, created by Marek Veithen :

call *nonlinear* ← modified routine, now compatible with PAW
 Read $\Psi^{(0)}_{n\mathbf{k}}$, $\epsilon^{(0)}_{n\mathbf{k}}$, $n^{(0)}(\mathbf{r})$ . . .
 if (*usepead* == 1) then ← Forbidden for PAW pseudopotentials (checked in the inputs)
  call pead_nl_loop ← old routine using PEAD formalism, remains untouched
   . . .
 else
  call dfptnl_loop ← new routine, compatible with PAW
   do *ipert*1 = 1, *mpert* ← loop on perturbations (only atomic displacement or electric field)
    Read $\Psi^{(\lambda_1)}_{n\mathbf{k}}$, $n^{(\lambda_1)}(\mathbf{r})$ . . .
    do *ipert*3 = 1, *mpert*
     Read $\Psi^{(\lambda_3)}_{n\mathbf{k}}$, $n^{(\lambda_3)}(\mathbf{r})$ . . .
     do *ipert*2 = 1, *mpert*
      Read $\Psi^{(\lambda_2)}_{n\mathbf{k}}$, $n^{(\lambda_2)}(\mathbf{r})$ . . .
      if (*ipert*2 == *natom* + 2) ← test if λ_2 is an electric field perturbation
       Read $\Psi^{(k_i\lambda_3)}_{n\mathbf{k}}$
      call dfptnl_pert ← Compute $\tilde{E}^{(\lambda_1\lambda_2\lambda_3)}$ (in one call, there is no self-consistent loop)
       . . .

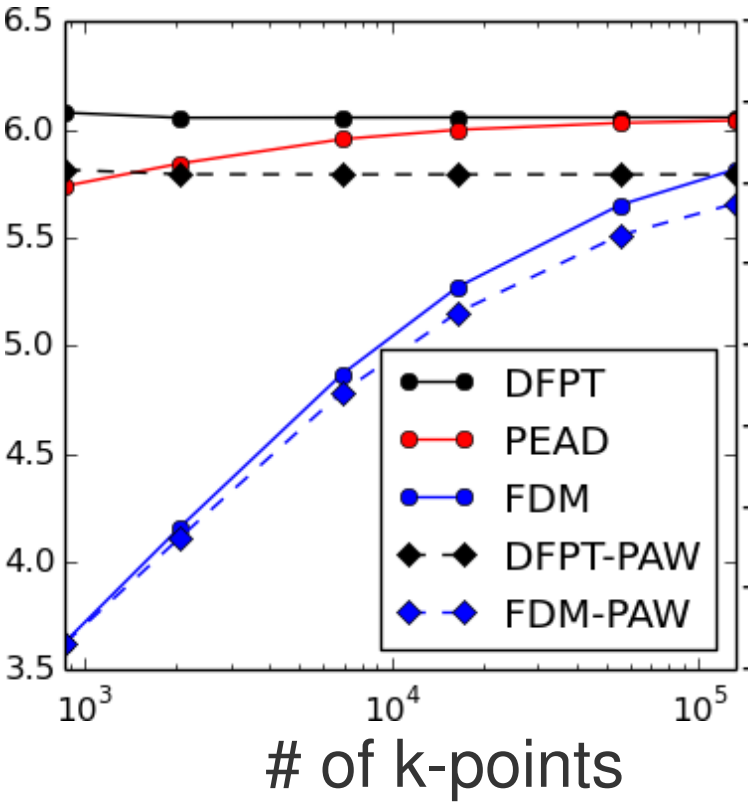
2. Implementation of 3rd order DFPT in ABINIT

Comparison of 3rd derivatives of E

System : AlAs

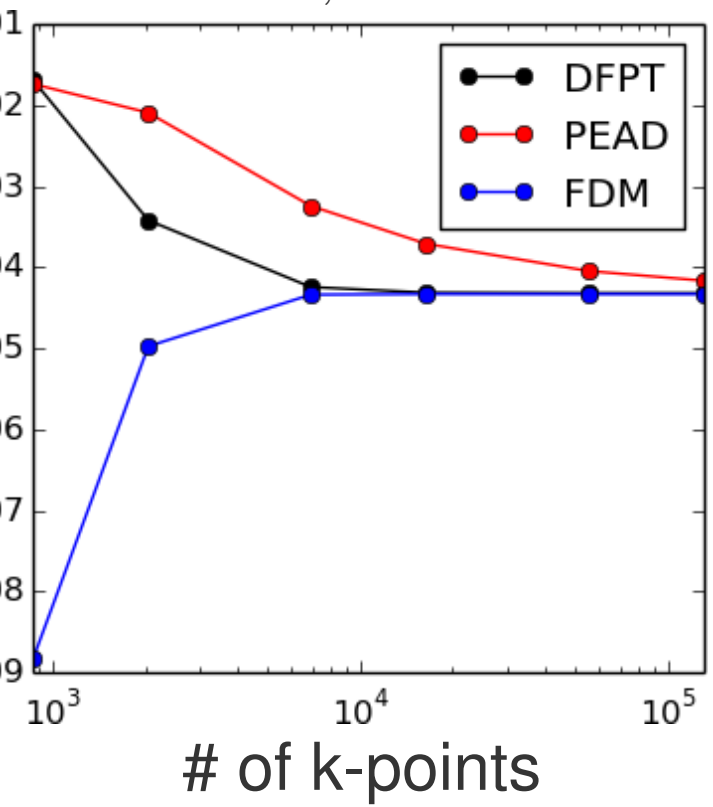
Due to symmetries : only 1 degree of liberty in the tensors

$\chi^{(2)}_{xyz}$



# of k-points

$\frac{\partial \chi^{(1)}_{xy}}{\partial \tau_{Al,z}}$



# of k-points

3. Conclusion

- The implementation of the 2^{nd} order Sternheimer equation has been done and checked for NC and PAW pseudopotentials.
- The implementation of 3^{rd} derivatives of E is operational for NC pseudopotentials, but still under debugging for PAW ones.
- The "full DFPT" method converges faster than PEAD one with respect to the number of **k** points.

$\Rightarrow$ It leads to a precise and efficient computation of the Raman tensor.

For an insulator, the density matrix writes :

$$\rho_{\mathbf{k}} = \sum_{n}^{\mathrm{occ}} |\psi_{n\mathbf{k}}^{(0)}\rangle\langle\psi_{n\mathbf{k}}^{(0)}|$$

$$\rho_{\mathbf{k}}(\mathbf{G},\mathbf{G}') \equiv \langle \mathbf{k}+\mathbf{G}|\rho_{\mathbf{k}}|\mathbf{k}+\mathbf{G}'\rangle = \sum_{n}^{\mathrm{occ}} \left(c_{n\mathbf{k}\mathbf{G}}^{(0)}\right)^{*} c_{n\mathbf{k}\mathbf{G}'}^{(0)}$$

where $c_{n\mathbf{k}\mathbf{G}}^{(0)}$ are the coefficients of ground state Bloch waves :

$$|\psi_{n\mathbf{k}}^{(0)}\rangle = \sum_{\mathbf{G}} c_{n\mathbf{k}\mathbf{G}}^{(0)} |\mathbf{k}+\mathbf{G}\rangle$$

Derivatives of the density matrix write :

$$\rho_{\mathbf{k}}^{(\lambda_1)} = \sum_{n}^{\mathrm{occ}} \left(|\psi_{n\mathbf{k}}^{(\lambda_1)}\rangle\langle\psi_{n\mathbf{k}}^{(0)}| + |\psi_{n\mathbf{k}}^{(0)}\rangle\langle\psi_{n\mathbf{k}}^{(\lambda_1)}| \right)$$

$$\rho_{\mathbf{k}}^{(\lambda_1\lambda_2)} = \sum_{n}^{\mathrm{occ}} \left(|\psi_{n\mathbf{k}}^{(\lambda_1\lambda_2)}\rangle\langle\psi_{n\mathbf{k}}^{(0)}| + |\psi_{n\mathbf{k}}^{(\lambda_1)}\rangle\langle\psi_{n\mathbf{k}}^{(\lambda_2)}| + |\psi_{n\mathbf{k}}^{(\lambda_2)}\rangle\langle\psi_{n\mathbf{k}}^{(\lambda_1)}| + |\psi_{n\mathbf{k}}^{(0)}\rangle\langle\psi_{n\mathbf{k}}^{(\lambda_1\lambda_2)}| \right)$$

So :

$$\rho_{\mathbf{k}}^{(\lambda_1\lambda_2)}(\mathbf{G},\mathbf{G}') = \sum_{n}^{\mathrm{occ}} \left(\left(c_{n\mathbf{k}\mathbf{G}}^{(\lambda_1\lambda_2)}\right)^{*} c_{n\mathbf{k}\mathbf{G}'}^{(0)} + \left(c_{n\mathbf{k}\mathbf{G}}^{(\lambda_1)}\right)^{*} c_{n\mathbf{k}\mathbf{G}'}^{(\lambda_2)} + \left(c_{n\mathbf{k}\mathbf{G}}^{(\lambda_2)}\right)^{*} c_{n\mathbf{k}\mathbf{G}'}^{(\lambda_1)} + \left(c_{n\mathbf{k}\mathbf{G}}^{(0)}\right)^{*} c_{n\mathbf{k}\mathbf{G}'}^{(\lambda_1\lambda_2)} \right)$$

4. Appendix
Gauge invariance of the density matrix

We define :

$$|\Phi_{n\mathbf{k}}\rangle = \sum_m U_{nm\mathbf{k}} |\Psi_{m\mathbf{k}}\rangle$$

where $U_{nm\mathbf{k}}$ is a unitary matrix :

$$\sum_{m'} U^*_{nm'\mathbf{k}} U_{mm'\mathbf{k}} = \delta_{nm}$$

For any operator A, one gets :

$$\sum_n^{\mathrm{occ}} \langle \Phi_{n\mathbf{k}}|A|\Phi_{n\mathbf{k}}\rangle = \sum_{nmm'}^{\mathrm{occ}} U^*_{nm\mathbf{k}} U_{nm'\mathbf{k}} \langle \Psi_{m\mathbf{k}}|A|\Psi_{m'\mathbf{k}}\rangle = \sum_n^{\mathrm{occ}} \langle \Psi_{n\mathbf{k}}|A|\Psi_{n\mathbf{k}}\rangle$$

So for *any observable* A :

$$\langle A\rangle_\Phi = \langle A\rangle_\Psi$$

In a same way :

$$\rho_{\mathbf{k}} = \sum_n^{\mathrm{occ}} |\Phi_{n\mathbf{k}}\rangle\langle\Phi_{n\mathbf{k}}| = \sum_{nmm'}^{\mathrm{occ}} U_{nm\mathbf{k}} U^*_{nm'\mathbf{k}} |\Psi_{m\mathbf{k}}\rangle\langle\Psi_{m'\mathbf{k}}| = \sum_n^{\mathrm{occ}} |\Psi_{n\mathbf{k}}\rangle\langle\Psi_{n\mathbf{k}}|$$