Implementation of the Strain Perturbation in ABINIT

D. R. Hamann,[1,2,3] Xifan Wu[1]

David Vanderbilt,[1] Karin Rabe,[1]

[1]Department of Physics and Astronomy, Rutgers University, Piscataway, NJ
[2]Bell Laboratories, Lucent Technologies, Murray Hill, NJ
[3]Mat-Sim Research LLC, Murray Hill, NJ

Supported by The Center for Piezoelectrics by Design, ONR grant N00014-01-1-0365 and Mat-Sim Research, *www.mat-simresearch.com*

Overview

- *Goal*: Direct calculation of the elastic and piezoelectric tensors and related quantities
- Strain and the Abinit reduced-coordinate formulation
- Brief review of density functional perturbation theory
- Response function (RF) code organization
- Development process design
- Special issues: nonlocal pseudopotentials, symmetry, non-linear core corrections, cutoff smoothing, metals
- Using: new input and output
- Atom coordinate relaxation and anaddb post processing
- Future development issues
- (Appendix)

Strain tensor $\eta_{\alpha\beta}$ as a perturbation

- Strain really only changes the positions of the atomic (pseudo)potentials,

$$V_{ext}(\mathbf{r}) = \sum_{\mathbf{R}} \sum_{\boldsymbol{\tau}}^{\text{cell}} V_{\boldsymbol{\tau}}(\mathbf{r}-\boldsymbol{\tau}-\mathbf{R}) \xrightarrow[\boldsymbol{\eta}]{} V_{ext}(\mathbf{r}) = \sum_{\mathbf{R}} \sum_{\boldsymbol{\tau}}^{\text{cell}} V_{\boldsymbol{\tau}}[\mathbf{r}-(\mathbf{1}+\boldsymbol{\eta})\cdot\boldsymbol{\tau}-(\mathbf{1}+\boldsymbol{\eta})\cdot\mathbf{R}].$$

- However, this causes unique problems for perturbation expansions
  - Infinite lattice view: the strain perturbation can never be small.
  - Single unit cell view: strain changes the periodic boundary conditions
  - Wave functions of the strained lattice cannot be expanded in terms of those of the unstrained lattice.

- The “canonical transformation” formulation changes structure of DFPT calculation from that for “ordinary” perturbations.[1]

[1] S. Baroni, P. Giannozzzi, and A. Testa, Phys. Rev. Lett. **59**, 2662 (1987).

Rutgers
Department of
Physics and Astronomy

Lucent Technologies
Bell Labs Innovations

Abinit reduced coordinate (~) formulation

- Every lattice, unstrained or strained, is a unit cube in reduced coordinates.
  - Primitive real and reciprocal lattice vectors define the transformations:

$$X_\alpha = \sum_i R^{\mathrm{P}}_{\alpha i}\tilde{X}_i\,, \quad K_\alpha \equiv (k_\alpha + G_\alpha) = \sum_i G^{\mathrm{P}}_{\alpha i}\tilde{K}_i\,, \quad \sum_\alpha R^{\mathrm{P}}_{\alpha i} G^{\mathrm{P}}_{\alpha j} = 2\pi\delta_{ij}$$

  - Cartesian indices $\alpha, \beta, \cdots = 1,3$ and reduced indices $i, j, \cdots = 1,3$
- **Every term in the DFT functional can be expressed in terms of dot products and the unit cell volume Ω.**
  - Dot products and Ω in reduced coordinates are computed with **metric tensors**,

$$\mathbf{X}'\cdot\mathbf{X} = \sum_{ij}\tilde{X}_i'\,\Xi_{ij}\,\tilde{X}_j\,, \quad \mathbf{K}'\cdot\mathbf{K} = \sum_{ij}\tilde{K}_i'\,\Upsilon_{ij}\,\tilde{K}_j\,, \quad \Omega = (\det[\Xi_{ij}])^{1/2}$$

- Strain is now a “simple” parameter of a density functional whose wave functions have invariant boundary conditions.

Abinit reduced coordinate (~) formulation

- Strain derivatives act **only** on the metric tensors,

$$\Xi_{ij}^{(\alpha\beta)} \equiv \frac{\partial \Xi_{ij}}{\partial \eta_{\alpha\beta}} = R^{\mathrm{P}}_{\alpha i} R^{\mathrm{P}}_{\beta j} + R^{\mathrm{P}}_{\beta i} R^{\mathrm{P}}_{\alpha j} , \qquad \Upsilon_{ij}^{(\alpha\beta)} \equiv \frac{\partial \Upsilon_{ij}}{\partial \eta_{\alpha\beta}} = -G^{\mathrm{P}}_{\alpha i} G^{\mathrm{P}}_{\beta j} - G^{\mathrm{P}}_{\beta i} G^{\mathrm{P}}_{\alpha j}$$

$$\begin{aligned} \Xi_{ij}^{(\alpha\beta\gamma\delta)} \equiv \frac{\partial^2 \Xi_{ij}}{\partial \eta_{\gamma\delta} \partial \eta_{\alpha\beta}} &= \delta_{\alpha\gamma}(R^{\mathrm{P}}_{\beta i} R^{\mathrm{P}}_{\delta j} + R^{\mathrm{P}}_{\delta i} R^{\mathrm{P}}_{\beta j}) + \delta_{\beta\gamma}(R^{\mathrm{P}}_{\alpha i} R^{\mathrm{P}}_{\delta j} + R^{\mathrm{P}}_{\delta i} R^{\mathrm{P}}_{\alpha j}) \\ &+ \delta_{\alpha\delta}(R^{\mathrm{P}}_{\beta i} R^{\mathrm{P}}_{\gamma j} + R^{\mathrm{P}}_{\gamma i} R^{\mathrm{P}}_{\beta j}) + \delta_{\beta\delta}(R^{\mathrm{P}}_{\alpha i} R^{\mathrm{P}}_{\gamma j} + R^{\mathrm{P}}_{\gamma i} R^{\mathrm{P}}_{\alpha j}), \end{aligned}$$

- Ω has uniquely simple derivatives for **Cartesian Strains**

$$\frac{\partial \Omega}{\partial \eta_{\alpha\beta}} = \delta_{\alpha\beta} \Omega , \quad \frac{\partial^2 \Omega}{\partial \eta_{\alpha\beta} \partial \eta_{\gamma\delta}} = \delta_{\alpha\beta} \delta_{\gamma\delta} \Omega$$

- **Key decision**: strain will be Cartesian throughout the code
  - Existing perturbations will remain in reduced-coordinates

Stress and strain notation

Cartesian	xx	yy	zz	yz	xz	xy
Cartesian	1 1	2 2	3 3	2 3	1 3	1 2
Voigt	1	2	3	4	5	6
ipert, idir	natom+3, 1	natom+3, 2	natom+3, 3	natom+4, 1	natom+4, 2	natom+4, 3

- Only the symmetric part of the strain tensor matters
  - Antisymmetric strains are simply rotations
- All forms used at various places internally and in output

Density Functional Perturbation Theory

- All quantities are expanded in power series in a DF energy parameter λ,

$$X(\lambda)=X^{(0)}+\lambda X^{(1)}+\lambda^2 X^{(2)}+\cdots,\quad X=E_{el},T,V_{ext},\psi_\alpha(\mathbf{r}),n(\mathbf{r}),\varepsilon_\alpha,H$$

- Solutions $\psi^{(0)}$ of Kohn-Sham equation minimize the usual DFT functional $E^{(0)}$

$$H^{(0)}\left|\psi_\alpha^{(0)}\right\rangle=\varepsilon_\alpha^{(0)}\left|\psi_\alpha^{(0)}\right\rangle.$$

- The variational functional for $E^{(2)}$ is minimized by solutions $\psi^{(1)}$ of the self-consistent Sternheimer equation

$$P_c(H^{(0)}-\varepsilon_\alpha^{(0)})P_c\left|\psi_\alpha^{(1)}\right\rangle=-P_cH^{(1)}\left|\psi_\alpha^{(0)}\right\rangle,$$

  - where P_c is the projector on unoccupied states (conduction bands) and

$$H^{(1)}=T^{(1)}+V_{ext}^{(1)}+V_{Hxc}^{(1)},\quad V_{Hxc}^{(1)}=\frac{\partial}{\partial\lambda}\frac{\delta E_{Hxc}}{\delta n(\mathbf{r})}\bigg|_{n^{(0)}}+\int\frac{\delta^2E_{Hxc}}{\delta n(\mathbf{r})\delta n(\mathbf{r}')}n^{(1)}(\mathbf{r}')\,d\mathbf{r}',$$

$$n^{(1)}(\mathbf{r})=\sum_\alpha^{\mathrm{occ}}[\psi_\alpha^{*(1)}(\mathbf{r})\psi_\alpha^{(0)}(\mathbf{r})+\psi_\alpha^{*(0)}(\mathbf{r})\psi_\alpha^{(1)}(\mathbf{r})].$$

Rutgers
Department of
Physics and Astronomy

Lucent Technologies
Bell Labs Innovations

DFPT for elastic and piezoelectric tensors

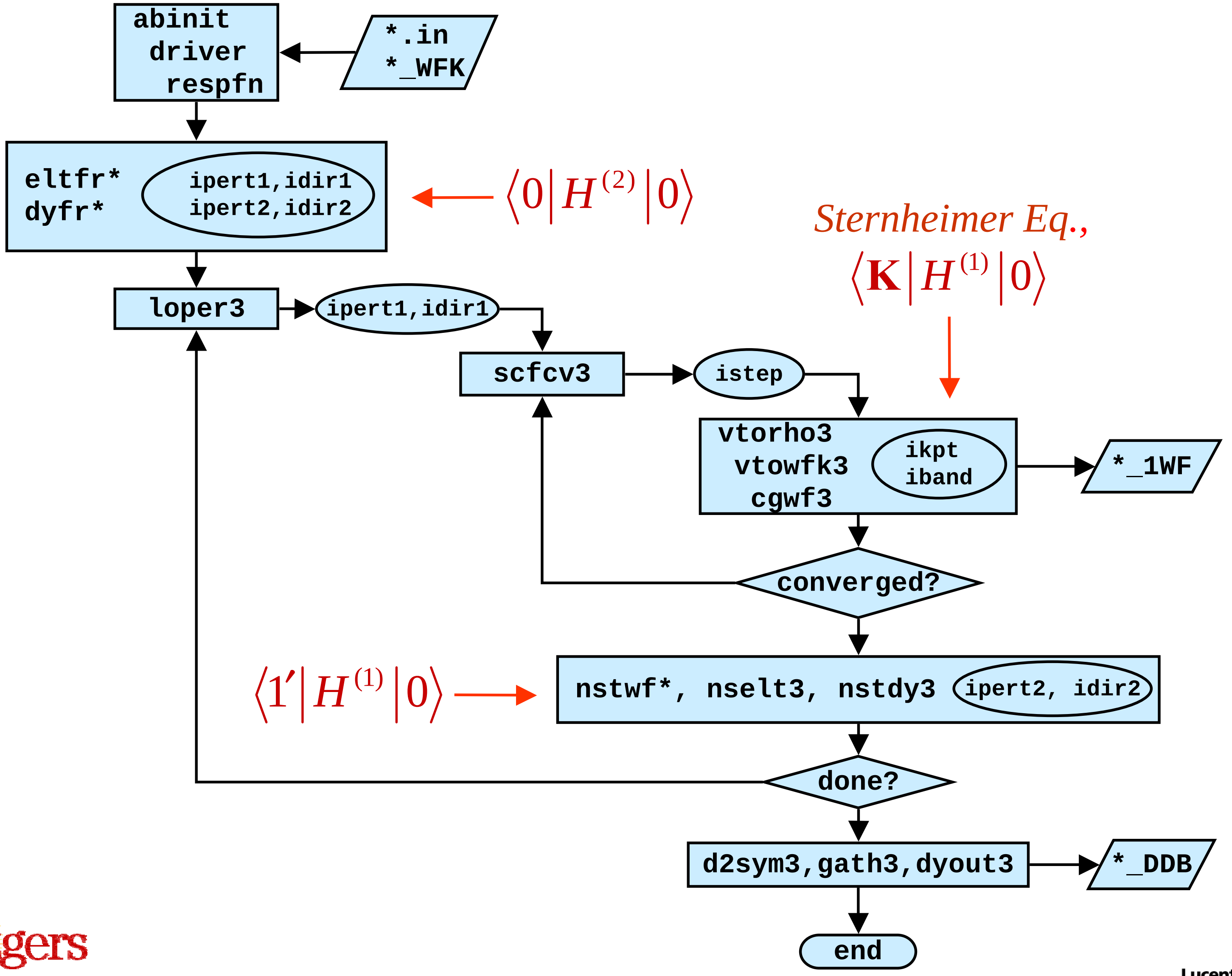
- Mixed 2nd derivatives of the energy with respect to two perturbations are needed.
  - By the “2n+1” theorem, these only require one set of 1st order wave functions,

$$E_{el}^{(\lambda_1\lambda_2)} = \sum_{\alpha}^{\mathrm{occ}} \left\langle \psi_\alpha^{(\lambda_2)} \right| \left(T^{(\lambda_1)} + V_{ext}^{(\lambda_1)} + H_{Hxc0}^{(\lambda_1)} \right) \left| \psi_\alpha^{(0)} \right\rangle$$

$$+ \sum_{\alpha}^{\mathrm{occ}} \left\langle \psi_\alpha^{(0)} \right| \left(T^{(\lambda_1\lambda_2)} + V_{ext}^{(\lambda_1\lambda_2)} \right) \left| \psi_\alpha^{(0)} \right\rangle + \frac{1}{2} \left. \frac{\partial^2 E_{Hxc}}{\partial\lambda_1 \partial\lambda_2} \right|_{n^{(0)}},$$

- Including atomic relaxation, we need
  - Clamped-atom elastic tensor ------------ $\partial^2 E_{el} / \partial\eta_{\alpha\beta}\partial\eta_{\gamma\delta}$
  - Internal strain tensor ----------------------- $\partial^2 E_{el} / \partial\eta_{\alpha\beta}\partial\tilde{\tau}_j$
  - Interatomic force constants -------------- $\partial^2 E_{el} / \partial\tilde{\tau}_i'\partial\tilde{\tau}_j$ ← Available
  - Clamped-atom piezoelectric tensor ---- $\partial^2 E_{el} / \partial\eta_{\alpha\beta}\partial\tilde{\mathcal{E}}_j$
  - Born effective charges --------------------- $\partial^2 E_{el} / \partial\tilde{\tau}_i\partial\tilde{\mathcal{E}}_j$ ← Available

Response function code organization



abinit driver respfn

*.in *_WFK

eltfr* dyfr*

ipert1,idir1 ipert2,idir2

$\langle 0 | H^{(2)} | 0 \rangle$

loper3

ipert1,idir1

scfcv3

istep

Sternheimer Eq.,

$\langle \mathbf{K} | H^{(1)} | 0 \rangle$

vtorho3 vtowfk3 cgwf3

ikpt iband

*_1WF

converged?

$\langle 1' | H^{(1)} | 0 \rangle$

nstwf*, nselt3, nstdy3

ipert2, idir2

done?

d2sym3,gath3,dyout3

*_DDB

end

Rutgers Department of Physics and Astronomy

Lucent Technologies Bell Labs Innovations

Design of development process

- Four stages based on RF code organization and degree of complexity
  - First $\langle \mathbf{K} | H^{(1)} | 0 \rangle$, second *Sternheimer*, third $\langle 0 | H^{(2)} | 0 \rangle$, and fourth $\langle 1' | H^{(1)} | 0 \rangle$
- **Stage-by-stage and term-by- term validation based on existing GS first derivatives of the total energy (1DTE's)**
- First stage
  - Re-compute stress as $\langle 0 | H^{(1)} | 0 \rangle = \sum_{\mathbf{K}} \langle 0 | \mathbf{K} \rangle \langle \mathbf{K} | H^{(1)} | 0 \rangle$
  - Compare term-by term to stress breakdown available in GS calculation:

```
stress: component    1 of hartree stress is   -8.625241635590E-04
stress: component    2 of hartree stress is   -7.368896556922E-04
stress: component    1 of loc psp stress is    2.656792257661E-03
stress: component    2 of loc psp stress is    2.166978270656E-03
stress: component    1 of xc stress is    4.902613744139E-03
stress: component    2 of xc stress is    4.902613744139E-03
stress: ii (diagonal) part is   -7.753477394392E-03
stress: component    1 of kinetic stress is   -5.477053634248E-03
stress: component    2 of kinetic stress is   -5.272489903492E-03
stress: component    1 of nonlocal ps stress is    2.396391029219E-03
stress: component    2 of nonlocal ps stress is    1.907769135572E-03
stress: component    1 of Ewald energ stress is    5.707853334522E-03
stress: component    2 of Ewald energ stress is    6.846039389167E-03
stress: component    1 of     core xc stress is   -1.789801472019E-03
stress: component    2 of     core xc stress is   -1.911589314962E-03
```

Design of development process (continued)

- Second through fourth stage – numerical derivatives of GS quantities
  - “Five-point” strain first derivatives of GS quantities (symmetric shear strains)
  - Strain increments small enough to keep complete $\{\mathbf{K}\}$ set invariant
  - Second derivatives of total energy (2DTE’s) from numerical derivatives of 1DTE’s (eg., stress)
  - ***Extreme*** convergence of self-consistency required, but not of $\mathbf{k}$’s or cutoffs
- Second stage – *Sternheimer*
  - Convergence of self-consistency loop
  - Ensure variational part of RF 2DTE’s decreases with convergence
  - First-order density $n^{(1)}$ validated by comparison with numerical $n^{(0)}$ derivatives
- Second stage – validation of variational RF 2DTE’s
  - Non-variational part $\langle 0|H^{(2)}|0\rangle$ not yet available
  - Compute numerical 2DTE’s as above with converged GS wave functions for strained lattices
  - Subtract numerical 2DTE’s with strained lattices but unstrained (“frozen”) GS wave functions
- So far, we are only dealing with “diagonal” 2DTE’s

Rutgers Department of Physics and Astronomy

Lucent Technologies Bell Labs Innovations

Design of development process (continued)

- Third stage – $\langle 0 | H^{(2)} | 0 \rangle$:

$$E^{(\lambda_1\lambda_2)}_{\text{non-var}} = \sum_{\alpha}^{\text{occ}} \left\langle \psi_\alpha^{(0)} \right| \left(T^{(\lambda_1\lambda_2)} + V_{ext}^{(\lambda_1\lambda_2)}\right) \left| \psi_\alpha^{(0)} \right\rangle + \frac{1}{2}\frac{\partial^2 E_{Hxc}}{\partial\lambda_1\partial\lambda_2}\bigg|_{n^{(0)}}$$

  - Validate term-by-term with “frozen wave function” numerical strain derivatives of stress components
  - Diagonal and off-diagonal 2DTE’s
  - Numerical strain derivatives of forces for internal strain mixed 2DTE’s
  - Frozen wf strain derivatives of the electric polarization are zero, so there is no contribution to piezoelectric tensor from these terms

- Note that the numerical derivatives of stress σ need Ω factor for 2DTE comparison

$$\frac{\partial^2 E_{el}}{\partial\eta_{\alpha\beta}\partial\eta_{\gamma\delta}} = \frac{\partial}{\partial\eta_{\alpha\beta}}\left(\Omega\sigma_{\gamma\delta}\right)$$

Design of development process (continued)

- Fourth stage – $E^{(\lambda_1\lambda_2)}_{\text{non-stat}} = \sum_{\alpha}^{\text{occ}} \left\langle \psi_\alpha^{(\lambda_2)} \middle| \left(T^{(\lambda_1)} + V_{ext}^{(\lambda_1)} + H_{Hxc0}^{(\lambda_1)}\right) \middle| \psi_\alpha^{(0)} \right\rangle$

  Non-self-consistent

  – Use strain-perturbation wave functions $\psi_\alpha^{(\lambda_2)}$

  – λ_1 is Cartesian strain or reduced atomic displacement

- Electric field is special case – $\frac{\partial^2 E_{el}}{\partial \tilde{\mathcal{E}}_j \partial \eta_{\alpha\beta}} = \frac{2\Omega}{(2\pi)^3} \int_{\text{BZ}} \sum_m^{\text{occ}} \left\langle i\,\psi_{\mathbf{k}m}^{(\tilde{k}_j)} \middle| \psi_{\mathbf{k}m}^{(\eta_{\alpha\beta})} \right\rangle d\mathbf{k}$

  – Field and *d/dk* first-order wave function in reduced coordinates

- Validate using numerical strain derivatives of GS stresses, forces, and polarization

  – Use converged strained wave functions

  – No subtractions since non-variational contributions are already validated

  – For polarization, numerical derivatives have to be corrected to give “proper” piezoelectric tensor (see **`Infos/theory/lr.pdf`**)

  – Need to use finite-difference *d/dk* wf’s in RF calculation for accurate comparison

Subroutines for strain perturbation (53)

Modified routines

	K10	1'10	020	Utl
`cart29.f`				•
`cgwf3.f`	•			
`chkinp.f`				•
`dyout3.f`				•
`eneres3.f`	•			
`gath3.f`				•
`insy3.f`	•	•		
`loper3.f`	•	•		
`mkcor3.f`	•	•	•	
`mkffkg3.f`	•	•	•	
`mkffkg.f`	•	•	•	
`mkffnl.f`	•	•	•	
`nonlop.f`	•	•	•	
`nstdy3.f`		•		
`opernl2.f`	•	•	•	
`opernl3.f`	•	•	•	
`opernl4a.f`	•	•	•	
`opernl4b.f`	•	•	•	
`prtene3.f`	•			
`prtxf.f`				•
`respfn.f`	•	•	•	
`scfcv3.f`	•			
`vtorho3.f`	•			
`vtowfk3.f`	•			

New routines

	K10	1'10	020	Utl
`contistr01.f`			•	
`contistr03.f`			•	
`contistr10.f`			•	
`contistr12.f`			•	
`contistr21.f`			•	
`contistr30.f`			•	
`contstr21.f`			•	
`contstr22.f`			•	
`contstr23.f`			•	
`contstr24.f`			•	
`contstr25a.f`			•	
`contstr25.f`			•	
`contstr26.f`			•	

	K10	1'10	020	Utl
`d2kindstr2.f`			•	
`eltfrhar3.f`			•	
`eltfrkin3.f`			•	
`eltfrloc3.f`			•	
`eltfrnl3.f`			•	
`eltfrxc3.f`			•	
`ewald4.f`			•	
`hartrestr.f`	•	•		
`kpgstr.f`	•	•		
`metstr.f`	•	•		
`newfermie1.f`	•			
`nselt3.f`		•		
`nstwf4.f`		•		
`splfit2.f`			•	
`symkchk.f`				•
`vlocalstr.f`	•	•		

K10	1'10	020	Utl
$\langle \mathbf{K} \vert H^{(1)} \vert 0 \rangle$	$\langle 1' \vert H^{(1)} \vert 0 \rangle$	$\langle 0 \vert H^{(2)} \vert 0 \rangle$	*Utility*

Mnemonics

`str` – strain, **`eltfr`** – elastic tensor frozen

`istr` – internal strain, **`cont`** – contraction

Nonlocal pseudopotentials in Abinit[1]

- Most mathematically complex object for strain derivatives
- Reduced wave vector matrix elements have the form

$$\langle \tilde{\mathbf{K}}' | V_{NL} | \tilde{\mathbf{K}} \rangle = \frac{1}{\Omega} \sum_{\kappa\ell} e^{2\pi i \tilde{\mathbf{K}}' \cdot \tilde{\boldsymbol{\tau}}_\kappa} f_{\kappa\ell}\Big(\sum_{ij} \Upsilon_{ij} \tilde{K}'_i \tilde{K}'_j\Big) \times$$

$$\wp_\ell\Big(\sum_{ij} \Upsilon_{ij} \tilde{K}'_i \tilde{K}'_j, \sum_{ij} \Upsilon_{ij} \tilde{K}'_i \tilde{K}_j, \sum_{ij} \Upsilon_{ij} \tilde{K}_i \tilde{K}_j\Big) e^{-2\pi i \tilde{\mathbf{K}} \cdot \tilde{\boldsymbol{\tau}}_\kappa} f_{\kappa\ell}\Big(\sum_{ij} \Upsilon_{ij} \tilde{K}_i \tilde{K}_j\Big)$$

  - $\wp_\ell$ modified Legendre polynomials, $f_{\kappa\ell}$ psp form factors, $\tilde{\tau}_\kappa$ reduced atom coordinates
  - All arguments are dot products expressed with metric tensors

- Psp's act on wave functions (in **`opernl*.f`**) by summing wave function coefficients $c_{\alpha\tilde{\mathbf{K}}}$ times a set of tensor products of reduced $\tilde{\mathbf{K}}$ components,

$$T_{\ell m}(\tilde{\mathbf{K}}) = \tilde{K}_1^{I_T(1,\ell,m)} \tilde{K}_2^{I_T(2,\ell,m)} \tilde{K}_3^{\ell - I_T(1,\ell,m) - I_T(2,\ell,m)}$$

  - $I_T(i,\ell,m)$ are index arrays
  - $T_{\ell m}(\tilde{\mathbf{K}})$ (created in **`mkffkg.f`**) are *analogous* to spherical harmonics

[1] D. C. Allan (unpublished, ~ 1987)

Nonlocal pseudopotential strain derivatives

- *Mathematica* programs created to do symbolic differentiation and extract coefficients coupling pairs of "input" ($\tilde{\mathbf{K}}$) and "output" ($\tilde{\mathbf{K}}'$) tensors
- Coefficients are polynomials in $\Upsilon_{ij}, \Upsilon_{ij}^{(\alpha\beta)}, \Upsilon_{ij}^{(\gamma\delta)}, \Upsilon_{ij}^{(\alpha\beta\gamma\delta)}$
- SED and C programs turn *Mathematica* results into useful Fortran 90
  - Example from **`contstr24.f`**

```
 cm(6,10)=(gm(1,3)**2*(270*dgm01(2,3)*dgm10(1,1)+540*dgm01(1,3)&
&    *dgm10(1,2)+540*dgm01(1,2)*dgm10(1,3)+270*dgm01(1,1)*dgm10(2,3))&
&    +gm(3,3)*(-108*gm(1,2)*(dgm01(1,3)*dgm10(1,1)+dgm01(1,1)*dgm10(1,3))&
&    +gm(1,1)*(-54*dgm01(2,3)*dgm10(1,1)-108*dgm01(1,3)*dgm10(1,2)&
&    -108*dgm01(1,2)*dgm10(1,3)-54*dgm01(1,1)*dgm10(2,3)))+gm(2,3)&
&    *(540*gm(1,3)*(dgm01(1,3)*dgm10(1,1)+dgm01(1,1)*dgm10(1,3))-54*gm(1,1)&
&    *(dgm01(3,3)*dgm10(1,1)+dgm01(1,1)*dgm10(3,3))+270*gm(1,3)**2*d2gm(1,1)&
&    +gm(3,3)*(-108*dgm01(1,1)*dgm10(1,1)-54*gm(1,1)*d2gm(1,1)))+180*gm(1,3)&
&    **3*d2gm(1,2)+gm(1,3)*(-108*(gm(1,2)*(dgm01(3,3)*dgm10(1,1)+dgm01(1,1)&
&    *dgm10(3,3))+gm(1,1)*(dgm01(3,3)*dgm10(1,2)+dgm01(1,2)*dgm10(3,3)))&
&    +gm(3,3)*(-216*dgm01(1,2)*dgm10(1,1)-216*dgm01(1,1)*dgm10(1,2)&
&    -108*gm(1,2)*d2gm(1,1)-108*gm(1,1)*d2gm(1,2)))))/36.d0
```

  - Here **`gm(i,j), dgm*(i,j), d2gm(i,j)`** are Υ_{ij} , etc.
  - Many 1000's of lines of infrequently executed code in **`cont*str*.f and metstr.f`**
  - Evaluation of **`cm`**'s is not a major factor in execution time

Symmetry with the strain perturbation

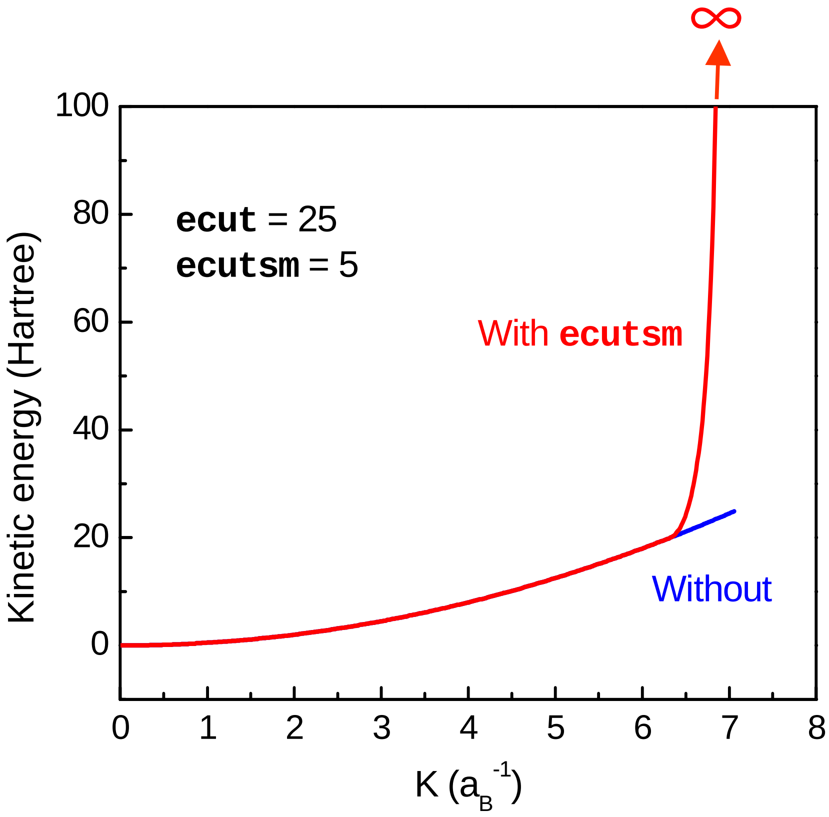
- The reduced-zone $\mathbf{k}$ sample determined for (space group / strain) is used for $\langle \mathbf{K} | H^{(1)} | 0 \rangle$, *Sternheimer*, and $\langle 1' | H^{(1)} | 0 \rangle$
  - The full-zone sample specified in input data must have the full space group symmetry (enforced by test).
- Loop on (**`ipert1, idir1`**) for 1st-order wave functions restricted by input variables (**`rfstrs, rfdir`**) but not by symmetry
  - This could be improved, but would have limited impact on performance
- Inner loop on (**`ipert2, idir2`**) in $\langle 1' | H^{(1)} | 0 \rangle$ calculations is carried over all strain and atomic displacement terms
  - piezoelectric contribution is computed if *d/dk* wf's are avaialble
- All $\langle 0 | H^{(2)} | 0 \rangle$ strain and internal-strain tensor elements are computed, using the full zone $\mathbf{k}$ sample
  - It is more efficient here to keep loops on strains and displacements inside routines like **`nonlop.f`**
  - The reduced zone for pairs of perturbations would seldom be reduced much anyway

XC non-linear core correction

- On the reduced real-space grid, electron charge depends only on Ω^{-1}
- Model core charge has a detailed dependence on Ξ_{ij}
  - Resulting analysis is rather complex
- Core charges must be ***extremely smooth*** functions to avoid significant convergence errors
  - Reason: Strain and atomic position derivatives of the xc self-interaction of a single core don't cancel point-by-point on the grid, but only in the integral
  - Inconsistencies in the treatment of the core charges and their derivatives in some **`Src_2psp/psp*cc.f`** routines makes matters worse

Kinetic energy cutoff “smoothing”

- Existing Abinit strategy to smooth energy dependence on lattice parameters in GS calculations
- RF strain derivative calculations do accurately reproduce GS numerical derivatives with non-zero **`ecutsm`**
- Divergence can produce large shifts in elastic tensor if calculation is not ***very well*** converged with respect to **`ecut`**
  - Remember, we take two derivatives
  - Perhaps the cutoff function could be improved



Strain perturbation for metals

- Thermal smearing of the Fermi surface must be introduced for stability
- In RF calculations, a band of partially-occupied states around ε_F is treated by finite-temperature perturbation theory in the Sternheimer solution, and only the completely unoccupied states are found by the conjugate-gradient method (1)
- For strain, a first-order Fermi energy $\varepsilon_F^{(1)}$ must be introduced(2)
- $\varepsilon_F^{(1)}$ enters into the Sternheimer self-consistency process
- Convergence can be rather slow
  - Only simple mixing is presently used to iterate $\varepsilon_F^{(1)}$
  - Coupling to the first-order potential iteration through Anderson or CG mixing may help
- Is $\varepsilon_F^{(1)}$ needed for the $\mathbf{Q} = 0$ interatomic force constant calculations needed to get the relaxed-atom elastic tensor for metals?

(1) S. de Gironcoli, Phys. Rev. B 51, 6773 (1995)
(2) S. Baroni, S. de Gironcoli, and A. Dal Corso, Rev. Mod. Phys. **73**, 515 (2001)

Rutgers
Department of Physics and Astronomy

Lucent Technologies
Bell Labs Innovations

Input file for RF run with strain

```
# First dataset  : Self-consistent run

# Second dataset : Non-self-consistent run
#                  for full k point set

# Third dataset  : d/dk response calculation
                   #this section is omitted if
 getwfk3  2        #only the elastic tensor is
 getden3  1        #wanted
   iscf3 -3
 rfelfd3  2
  rfdir3  1 1 1

# Fourth dataset : phonon, strain,and homogeneous
#                  electric field response

 diemix4  0.85
 diemac4  1.0
 getwfk4  2
 getddk4  3        #omitted for ELT only
   iscf4  3
 rfelfd4  3        #omitted for ELT only
rfatpol4  1 2
  rfdir4  1 1 1
 rfphon4  1
 rfstrs4  3        #only this is new for strain

# Common data      #stresses and forces should
    nqpt  1        #(in general) be relaxed
     qpt  0 0 0    #beforehand
```

2DTE terms in output file

- Mix of reduced and Cartesian coordinates, also in _DDB output file
  - With **`natom`** = 2, electric field **`pert`** = 4 and strain **`pert`** = 5, 6
  - Only a sample of the complete matrix shown

```
 2nd-order matrix (non-cartesian coordinates, masses not included,
   asr not included )
  cartesian coordinates for strain terms (1/ucvol factor
   for elastic tensor components not included)
     j1       j2             matrix element
  dir pert dir pert     real part     imaginary part


   1    1   2    2   -2.8200006186    0.0000000000

   1    1   3    2   -2.8654826400    interatomic force constant (red-red)

   1    1   1    4   -4.1367712586    Born effective charge (red-red)

   1    1   2    5   -0.0238530938    internal strain (red-cart)

   1    4   3    4   46.0269881204    dielectric tensor (red-red)

   1    4   3    5   -0.2214090328    piezoelectric tensor (red-cart)

   1    5   2    6   -0.0103809572    elastic tensor (cart-cart)
```

- Cartesian ELT, PZT, and internal strain are also included in the output
- Detailed breakdown of contributions is given for **`prtvol`** = 10

Incorporating atomic relaxation

- Implemented as post-processing procedure in anaddb
  - New and modified routines: **`dielmore9.f, elast9.f, piezo9.f, instr9.f, invars9.f, outvars9.f, diel9.f, anaddb.f, defs_common.f, defs_basis.f`**
- Full theoretical discussion in **`Infos/Theory/lr.pdf`**
- Basic results:

$$\tilde{C}_{\alpha\beta,\gamma\delta} = C_{\alpha\beta,\gamma\delta} + \Omega^{-1} \sum_{mn=1}^{natom} \sum_{ij=1}^{3} \Lambda_{mi,\alpha\beta} (K^{-1})_{mi,nj} \Lambda_{nj,\gamma\delta}$$

$$\tilde{e}_{\alpha\beta,\gamma} = e_{\alpha\beta,\gamma} + \Omega^{-1} \sum_{mn=1}^{natom} \sum_{ij=1}^{3} \Lambda_{mi,\alpha\beta} (K^{-1})_{mi,nj} Z_{nj,\gamma}$$

  - $\tilde{C}, C$ physical and clamped-atom elastic tensors
  - $\tilde{e}, e$ physical and clamped-atom piezoelectric tensors
  - K^{-1} pseudo-inverse Q=0 interatomic force constant matrix
  - Λ internal-strain “force response” tensor
  - Z Born effective charge matrix
  - All in Cartesian coordinates

Rutgers Department of Physics and Astronomy

Lucent Technologies Bell Labs Innovations

Input file for anaddb run

```
  dieflag  3  !flag for relaxed-ion dielectric tensor
  elaflag  3  !flag for the elastic tensor
piezoflag  3  !flag for the piezoelectric rensor
instrflag  1  !flag for the internal strain tensor

!the effective charge part
      asr  1
   chneut  1

!Wavevector list number 1

    nph1l  1
    qph1l  0.0 0.0 0.0 1.0

!Wave vector list no. 2

    nph2l  1
    qph2l  0.0 0.0 1.0 0.0
```

New flags and/or values in violet

New output from anaddb

```
     Elastic Tensor(relaxed ion)(Unit:10^2GP,VOIGT notation):

 1.2499151   0.6699976   0.6835944   0.0022847  -0.0113983  -0.0001512
 0.6699976   1.6217899   0.5566207   0.0194005  -0.0055653  -0.0055915
 0.6835944   0.5566207   1.5896839  -0.0207927   0.0107924   0.0080825
 0.0022847   0.0194005  -0.0207927   0.6659339   0.0077398  -0.0056845
-0.0113983  -0.0055653   0.0107924   0.0077398   0.7283916   0.0014049
-0.0001512  -0.0055915   0.0080825  -0.0056845   0.0014049   0.7222881

     proper piezoelectric constants(relaxed ion)(Unit:c/m^2)

               0.01714694      0.05107080     -0.00883676
               0.00828454      0.03716812     -0.00810176
               0.01882065      0.05180658     -0.00576393
              -0.03872154     -0.01245206      0.01902693
              -0.01424058      0.00757132     -0.00294782
               0.01566436     -0.00054740      0.00218470
```

- Also in output
  - Clamped-ion versions of the above in standard units
  - Clamped and relaxed compliance tensors
  - “Force-response” and “displacement response” internal strain tensors
  - More tensors corresponding to different boundary conditions to be added

Global comparison with numerical derivatives

- Zinc-blende AlP with random distortions so all tensor elements are non-zero.
  - Ground state calculations of stress and polarization with ***exquisitely*** relaxed atomic coordinates (but unrelaxed stress)
  - Finite-difference $d/dk\ \psi_\alpha^{(1)}$'s for best consistency with polarization calculations
  - Sample of complete set of tensor elements

Elastic Tensor (GPa)

		Numerical	DFPT	Diff
xx	xx	124.991500	124.991500	-1.1E-05
yy	xx	66.999750	66.999760	8.2E-06
zz	xx	68.359440	68.359440	7.0E-07
yz	xx	0.228447	0.228466	1.9E-05
xz	xx	-1.139838	-1.139828	9.6E-06
xy	xx	-0.015028	-0.015117	-9.0E-05
xx	yz	0.228471	0.228466	-4.4E-06
yy	yz	1.940050	1.940054	3.7E-06
zz	yz	-2.079264	-2.079275	-1.1E-05
yz	yz	66.593340	66.593390	5.2E-05
xz	yz	0.773972	0.773977	5.1E-06
xy	yz	-0.568446	-0.568449	-3.2E-06

Piezoelectric Tensor (C/m^2 x 10^{-2})

		Numerical	DFPT	Diff
x	xx	1.714769	1.714694	-7.5E-05
y	xx	5.107069	5.107080	1.1E-05
z	xx	-0.883962	-0.883676	2.9E-04
x	yy	0.828569	0.828454	-1.2E-04
y	yy	3.716843	3.716812	-3.2E-05
z	yy	-0.810201	-0.810176	2.5E-05
x	yz	-3.871980	-3.872154	-1.7E-04
y	yz	-1.245173	-1.245206	-3.3E-05
z	yz	1.902687	1.902693	5.6E-06

- RMS Errors 4.0X10^{-5}, ELT and 1.7X10^{-6}, PZT
  - One-two orders of magnitude smaller errors for clamped-atom quantities.

Present status, future development

- Examples of strain RF and anaddb calculations are `Test_v4/t61-70`
- RF strain is fully parallelized
  - Parallel version was developed simultaneously with sequential
- Present limitations
  - Norm-conserving psp's
  - Non-spin polarized (this is about to be relaxed, testing is nearly complete)
  - LDA only
  - No spin-orbit
- GGA prospects
  - Probably straightforward but complicated by "two kinds of charge" problem with model cores
  - Model core smoothness problem is undoubtedly worse
- Spin-orbit coupling
  - This has all the nonlocal psp complexity, probably significantly worse judging by the existing spin-orbit code for stress
  - *Mathematica* code will eventually be added to the documentation and may help a future developer with this

Future development, continued

- PAW
  - Far beyond norm-conserving psp non-local complexity
  - Needs spherical harmonics with off-diagonal coupling which cannot be turned into dot products with simple metric-tensor dependencies
  - Has “two kinds of charge” problem like model core but much worse, because augmentation charge has non-spherical components

On the upbeat side, however

- 3rd-order response functions involving strain via “2n+1” theorem
  - Require two $\psi^{(1)}$ and one $H^{(1)}$, all available
  - Eg., electrostriction, non-linear elastic constants, Grüneisen parameters
- It’s time for feedback – let’s see what the users want and what trouble they get into
  - If a user wants a strain feature badly enough *we’ll have a new developer* !
  - Isn’t that the ABINIT philosophy?

Appendix : *Mathematica* for nonlocal psp

$$\langle \mathbf{K}' | V_{NL} | \mathbf{K} \rangle = \frac{1}{\Omega} \sum_{\kappa \ell m} e^{i\mathbf{K}'\cdot\boldsymbol{\tau}_\kappa} f_{\kappa\ell}(\mathbf{K}'\cdot\mathbf{K}') \wp_\ell(\mathbf{K}'\cdot\mathbf{K}', \mathbf{K}'\cdot\mathbf{K}, \mathbf{K}\cdot\mathbf{K}) e^{-i\mathbf{K}\cdot\boldsymbol{\tau}_\kappa} f_{\kappa\ell}(\mathbf{K}\cdot\mathbf{K}).$$

- Define tensor products $T_{\ell m}(\tilde{\mathbf{K}})$ and $T_{\ell m}(\tilde{\mathbf{K}}')$
  - Follows D. C. Allan

```
tnk = {
{1},   (* 0, 1 *)

{k1,   (* 1, 1 *)
 k2,   (* 1, 2 *)
 k3},  (* 1, 3 *)

{k1 k1,   (* 2, 1 *)
 k2 k2,   (* 2, 2 *)
 k3 k3,   (* 2, 3 *)
 k3 k2,   (* 2, 4 *)

   etc. to rank 7
```

- Define K's, metric tensor functions, dot products, and Legengre's
- `s1` and `s2` are strain variables

```
k = {k1, k2, k3};  kp = {kp1, kp2, kp3};

m = {{m11[s1,s2], m12[s1,s2], m13[s1,s2]},
{m12[s1,s2], m22[s1,s2], m23[s1,s2]},
{m13[s1,s2], m23[s1,s2], m33[s1,s2]}};

dt = kp.m.k;  ks = k.m.k;  kps = kp.m.kp;

Plegendre={1, dt, 1.5 dt^2 - 0.5 kps ks,
2.5 dt^3 - 1.5 kps dt ks};
```

Mathematica for nonlocal psp, continued

- Strain derivatives of form factors f_ℓ “bring out” derivatives of dot products
- Define 6 combinations of dot product derivatives and Legendre derivatives that have given offsets between “input” and “output” rank

```
poly = {D[kps,s2] D[kps,s1] Plegendre[[rank+1]],
        D[ks, s2] D[ks, s1] Plegendre[[rank+1]],
       (D[D[kps,s2],s1] Plegendre[[rank+1]]
      + D[kps,s1] D[Plegendre[[rank+1]],s2]
      + D[kps,s2] D[Plegendre[[rank+1]],s1]),
       (D[D[ks, s2],s1] Plegendre[[rank+1]]
      + D[ks, s1] D[Plegendre[[rank+1]],s2]
      + D[ks, s2] D[Plegendre[[rank+1]],s1]),
       (D[kps,s1] D[ks,s2] + D[kps,s2] D[ks,s1])
         Plegendre[[rank+1]],
        D[D[Plegendre[[rank+1]],s2],s1]};
```

$\ell \rightarrow \ell+4$	$f_\ell,\ f_\ell''$
$\ell+4 \rightarrow \ell$	$f_\ell'',\ f_\ell$
$\ell \rightarrow \ell+2$	$f_\ell,\ f_\ell'$
$\ell+2 \rightarrow \ell$	$f_\ell',\ f_\ell$
$\ell+2 \rightarrow \ell+2$	$f_\ell',\ f_\ell'$
$\ell \rightarrow \ell$	$f_\ell,\ f_\ell$

  - In *Mathematica* df/dx is **`D[f,x]`**

- Now, do the work – extract the coefficients of each pair of input and output tensors

```
Do[term = Simplify[Coefficient[poly[[iterm]], (tnkp[[rankout+1]][[jj]] *
        tnk[[rankin+1]][[ii]])]];
```