

Octopus + BerkeleyGW

e-h matrix linear response

BuildBot, debugging



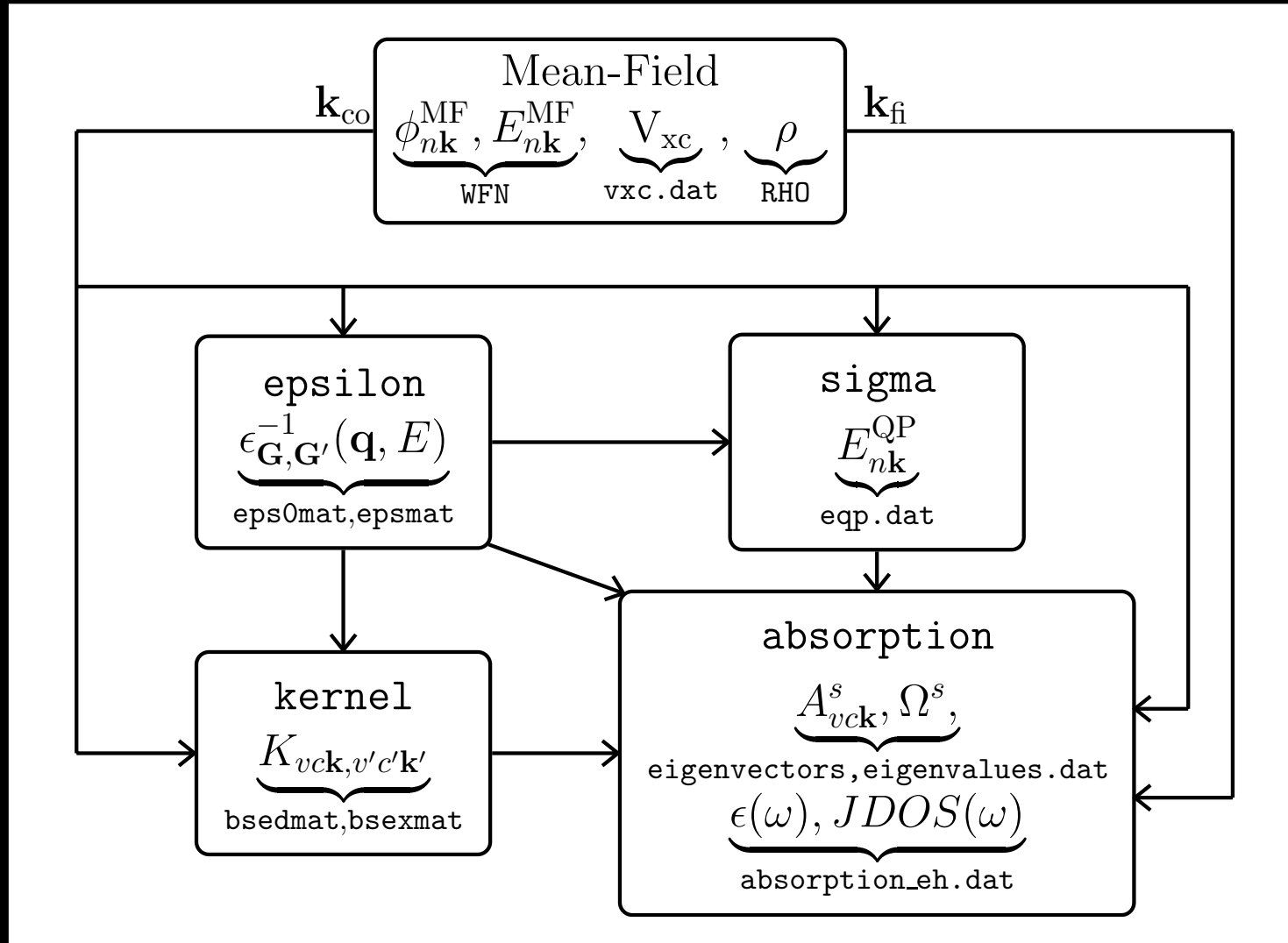
*J. Deslippe, G. Samsonidze, D. A. Strubbe, M. Jain, M. L. Cohen,
and S. G. Louie, Comput. Phys. Comm. **183**, 1269 (2012)*

www.berkeleygw.org

David Strubbe

23 October 2012

BerkeleyGW: workflow for calculations



Many-body perturbation theory in a plane-wave basis

What can the code do? Dielectric function

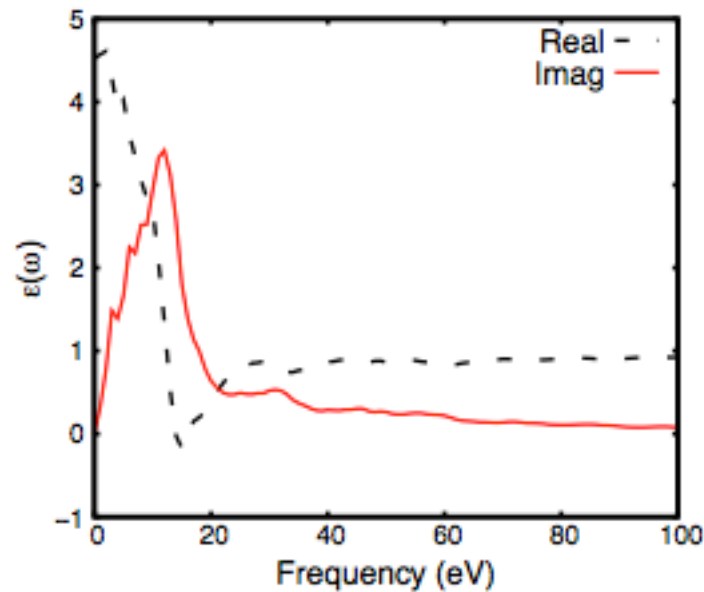


Figure 5: Example output plotted from EpsDyn showing the computed $\epsilon_{00}(\omega)$ in ZnO.

What can the code do? Band structure

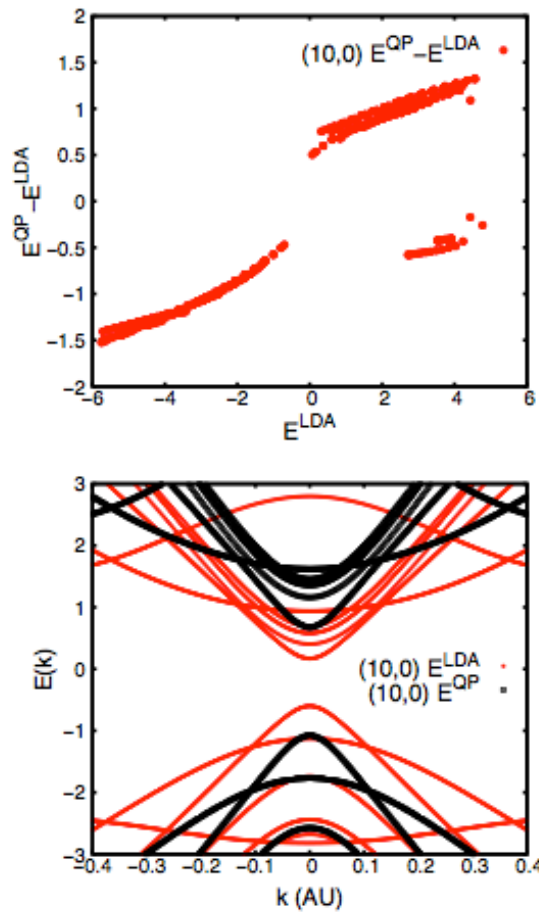


Figure 8: Top: GW quasiparticle self-energy corrections, $E^{\text{QP}} - E^{\text{LDA}}$ vs. the LDA energy for (10,0) SWCNT. Both a rigid opening of the band gap and a non-linear energy scaling are present. Bottom: The fine-grid quasiparticle bandstructure using the interpolated self-energy corrections (black open) and the LDA uninterpolated bandstructure (red). 256 points are used to sample the Brillouin zone.

What can the code do? Optical absorption

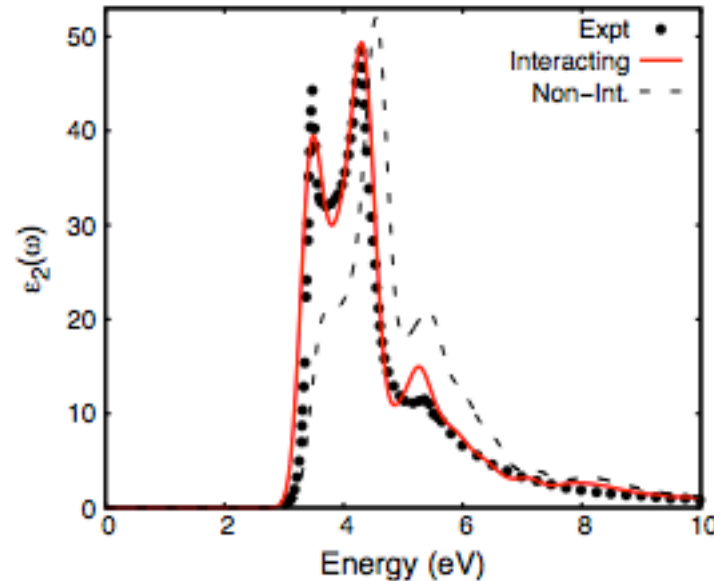


Figure 1: The absorption spectra for silicon calculated at the GW (black dashed) and GW-BSE (red solid) levels using the BerkeleyGW package. Experimental data from [22].

What can the code do? Exciton wfns

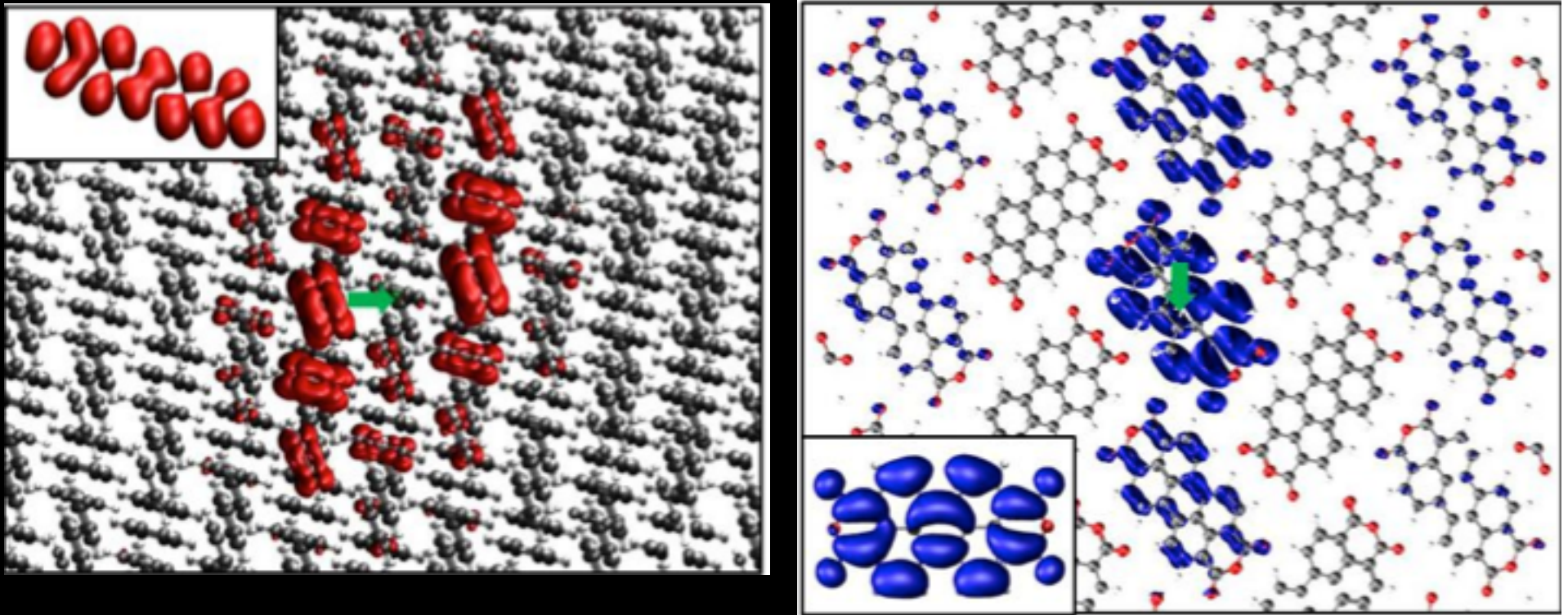


Figure 2: The electronic component of the excited state wavefunction (Eq. 1) for pentacene (red) and PTCDA (blue), with the hole located at the center of a molecule, as indicated by a green arrow. Insets show the same for the gas-phase molecule. For all cases, a common isosurface value is taken to be 1% of the maximum.

S Sharifzadeh, A Biller, L Kronik, and JB Neaton, *Phys. Rev. B* **85**, 125307 (2012)

Website: www.berkeleygw.org



BerkeleyGW

[home](#)[download](#)[forum](#)[literature](#)[about](#)

CAPTCHA

Only real scientists may create an account. If you aren't sure, read some more about GW!

What does the G stand for in 'GW'? (only give the first word) *

Fill in the blank.

Create new account

Theoretical framework: mean field

Mean field = One-electron energies and wavefunctions

(like mean-field theory for Ising model vs. correlated treatment)

Could be DFT (LDA, GGA, hybrids), Hartree-Fock, etc.

$$\left[-\frac{1}{2}\nabla^2 + V_{\text{ion}} + V_{\text{H}} + V_{\text{xc}}^{\text{DFT}} \right] \psi_{n\mathbf{k}}^{\text{DFT}} = E_{n\mathbf{k}}^{\text{DFT}} \psi_{n\mathbf{k}}^{\text{DFT}}$$

Supported codes: PARATEC, ESPRESSO, SIESTA, PARSEC, Octopus.
Must be norm-conserving pseudopotentials (no PAW, ultrasoft).

Need also charge density and XC matrix elements.

Empirical pseudopotential method (EPM) code based on TBPW from Richard Martin's *Electronic Structure* book, with extensive bugfixes and improvements

Binary input to BerkeleyGW

EPM, paratecSGL, ESPRESSO: write wavefunction, density, V_{xc}

Epsilon: read wfns on unshifted (and shifted) grid

Sigma: read wfns for constructing operator and for taking matrix elements, read density for GPP, read V_{xc} to subtract from self-energy

Kernel: read wfns on coarse unshifted (and shifted) grid

Diag/Haydock: read wfns on coarse and fine unshifted (and shifted) grid

Specification of header

WFN, RHO, VXC

WFN only

scalars [WFN/RHO/VXC]-[Real/Complex] date time
nspin ng ntran cell_symmetry nat ecutrho nk nbands ngkmax ecutwfc

arrays of fixed size kmax(1:3) kgrid(1:3) kshift(1:3)
celvol alat avec(1:3, 1:3) adot(1:3, 1:3)
recvol blat bvec(1:3, 1:3) bdot(1:3, 1:3)
mtrx(1:3, 1:3, 1:ntran)
tnp(1:3, 1:ntran)

real space
reciprocal space
symmetry operations
fractional translations

allocatable arrays apos(1:3, 1:nat) atyp(1:3, 1:nat)
ngk(1:nk)
kweights(1:nk)
kpt(1:3, 1:nk)
ifmin(1:nspin, 1:nk)
ifmax(1:nspin, 1:nk)
energies(1:nbands, 1:nk, 1:nspin)
occupations(1:nbands, 1:nk, 1:nspin)

atoms
G-vector for each k-point
k-weights
k-coordinates
lowest occupied band
highest occupied band
eigenvalues
occupations

Specification of body

WFN:

`gvec(1:3, 1:ng)` *global list*

`gvec(1:3, 1:ngk(ik))` *for each k-point*
`data(1:ngk(ik), 1:nspin)`

RHO/VXC:

`gvec(1:3, 1:ng)`
`data(1:ng, 1:nspin)`

Driver routines

[read/write]_[binary/format/]_header
[read/write]_[binary/format/]_gvector
[read/write]_[binary/format/]_[real/complex/]_data

Generated by preprocessor from only 3 underlying routines.

HDF5 in progress (Jamal Mustafa, UC Berkeley)

Interfaces to other codes welcome!
M4 configure script, Fortran90 modules
Sort of like ETSF_IO.

Current status: more or less works in Octopus.
Exchange for silane agrees between paratec, paratec -> BerkeleyGW, octopus -> BerkeleyGW. But not with octopus ...

Velocity matrix elements for optical spectrum

$$f_S = \frac{2 |\mathbf{e} \cdot \langle 0 | \mathbf{v} | S \rangle|^2}{\Omega^S} \quad (44)$$

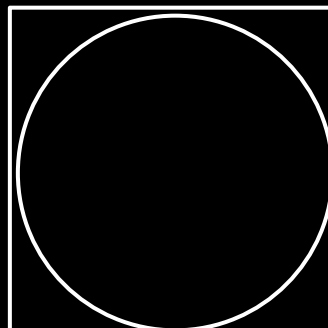
We compute the velocity matrix element via the commutator of the many-body Hamiltonian, as follows [43]:

$$\begin{aligned} \langle 0 | \mathbf{v} | S \rangle &= \langle 0 | i [H, \mathbf{r}] | S \rangle = i (E_0 - E_S) \langle 0 | \mathbf{r} | S \rangle \\ &= -i \Omega^S \sum_{v\mathbf{ck}} A_{v\mathbf{ck}}^S \langle v\mathbf{k} | \mathbf{r} | c\mathbf{k} \rangle \end{aligned} \quad (45)$$

In a periodic system, we cannot calculate matrix elements of the position operator, but we can use a $\mathbf{q} \rightarrow 0$ limit [8]:

$$\begin{aligned} \langle v\mathbf{k} | \mathbf{r} | c\mathbf{k} \rangle &= \lim_{\mathbf{q} \rightarrow 0} \frac{\langle v\mathbf{k} + \mathbf{q} | e^{i\mathbf{q} \cdot \mathbf{r}} - 1 | c\mathbf{k} \rangle}{iq} \\ &= -i \lim_{\mathbf{q} \rightarrow 0} \frac{\langle v\mathbf{k} + \mathbf{q} | e^{i\mathbf{q} \cdot \mathbf{r}} | c\mathbf{k} \rangle}{q} \end{aligned} \quad (46)$$

FFT sphere or box?



Density-matrix-based TDDFT approaches

CasidaTheoryLevel

Section: Linear Response::Casida

Type: flag

Default: `eps_diff + petersilka + lrtdfft_casida`

Choose which electron-hole matrix-based theory levels to use in calculating excitation energies. More than one may be used to take advantage of the significant commonality between the calculations. Only `eps_diff` is available for spinors. Note the restart data saved by each theory level is compatible with all the others.

Options:

- **lrtdfft_casida**: The full Casida method. Ref: C Jamorski, ME Casida, and DR Salahub, *J. Chem. Phys.* **104**, 5134 (1996) and ME Casida, "Time-dependent density functional response theory for molecules," in *Recent Advances in Density Functional Methods*, edited by DE Chong, vol. 1 of *Recent Advances in Computational Chemistry*, pp. 155-192 (World Scientific, Singapore, 1995).
- **eps_diff**: Difference of eigenvalues, *i.e.* independent-particle approximation.
- **petersilka**: The Petersilka approximation uses only the diagonal part of the Tamm-Dancoff matrix. This is acceptable if there is little mixing between single-particle transitions. Ref: M Petersilka, UJ Gossmann, and EKU Gross, *Phys. Rev. Lett.* **76**, 1212 (1996).
- **tamm_dancoff**: The Tamm-Dancoff approximation uses only occupied-unoccupied transitions and not unoccupied-occupied transitions. Ref: S Hirata and M Head-Gordon, *Chem. Phys. Lett.* **314**, 291 (1999).
- **variational**: Second-order constrained variational theory CV(2)-DFT. Ref: T Ziegler, M Seth, M Krykunov, J Autschbach, and F Wang, *J. Chem. Phys.* **130**, 154102 (2009).

CV(2) theory solves for density of excited state. Approximations: kernel expanded around gs density, finite number of unoccupied states, energies only to second order. 4th-order theory has same eigenvectors.

Density-matrix-based TDDFT approaches

$$\mathbf{Q}\mathbf{F}_I = \Omega_I^2 \mathbf{F}_I, \quad (1)$$

given here in a spin-independent form for simplicity. The matrix elements of \mathbf{Q} are given by

$$\mathbf{Q}_{ij,kl} = \delta_{i,k} \delta_{j,l} \omega_{kl}^2 + 2\sqrt{\lambda_{ij} \omega_{ij}} \mathbf{K}_{ij,kl} \sqrt{\lambda_{kl} \omega_{kl}}, \quad (2)$$

$$\begin{aligned} \mathbf{K}_{ij,kl} = 2 \int \int \psi_i^*(\mathbf{r}) \psi_j^*(\mathbf{r}) & \left(\frac{1}{|\mathbf{r} - \mathbf{r}'|} + \frac{\delta^2 E_{xc}}{\delta \rho(\mathbf{r}) \delta \rho(\mathbf{r}')} \right) \\ & \times \psi_k(\mathbf{r}') \psi_l(\mathbf{r}') d\mathbf{r} d\mathbf{r}'. \end{aligned}$$

<- singlet kernel

where $\lambda_{ij} = n_j - n_i$ is the difference between the occupation numbers of the i and j single-particle states, $\omega_{ij} = \varepsilon_i - \varepsilon_j$ is the difference between the i and j eigenvalues, and \mathbf{K} is a coupling matrix whose elements in the adiabatic approximation

triplet kernel: $f_{xc}^{\uparrow\uparrow} - f_{xc}^{\uparrow\downarrow}$

Density-matrix-based TDDFT approaches

$$\mathbf{K}_{ij,kl} = 2 \int \int \psi_i^*(\mathbf{r}) \psi_j^*(\mathbf{r}) \left(\frac{1}{|\mathbf{r} - \mathbf{r}'|} + \frac{\delta^2 E_{xc}}{\delta \rho(\mathbf{r}) \delta \rho(\mathbf{r}')} \right) \times \psi_k(\mathbf{r}') \psi_l(\mathbf{r}') d\mathbf{r} d\mathbf{r}'.$$

$$\mathbf{Q}_{ij,kl} = \delta_{i,k} \delta_{j,l} \omega_{kl}^2 + 2 \sqrt{\lambda_{ij} \omega_{ij}} \mathbf{K}_{ij,kl} \sqrt{\lambda_{kl} \omega_{kl}},$$

$$\left[\begin{pmatrix} -\Delta E - \mathcal{K} & -\mathcal{K} \\ \mathcal{K} & \Delta E + \mathcal{K} \end{pmatrix} - \omega I \right] \begin{pmatrix} -B^{(+\omega, re)} \\ B^{(-\omega, re)} \end{pmatrix} = 0.$$

Eps_diff: $K = 0$

Petersilka: only diagonal part of K

Tamm-Dancoff: $B = 0$, no sqrts

CV(2): no sqrts

$$\begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^* & \mathbf{A}^* \end{pmatrix} \begin{pmatrix} \mathbf{X} \\ \mathbf{Y} \end{pmatrix} = \omega \begin{pmatrix} \mathbf{1} & \mathbf{0} \\ \mathbf{0} & -\mathbf{1} \end{pmatrix} \begin{pmatrix} \mathbf{X} \\ \mathbf{Y} \end{pmatrix}, \quad (21)$$

with

$$A_{ai\sigma, bj\tau} = \delta_{ij} \delta_{ab} \delta_{\sigma\tau} (\varepsilon_{a\sigma} - \varepsilon_{i\tau}) + (a_{\sigma} i_{\sigma} | j_{\tau} b_{\tau}) + (a_{\sigma} i_{\sigma} | w_{\sigma\tau} | j_{\tau} b_{\tau}), \quad (22)$$

$$B_{ai\sigma, bj\tau} = (a_{\sigma} i_{\sigma} | b_{\tau} j_{\tau}) + (a_{\sigma} i_{\sigma} | w_{\sigma\tau} | b_{\tau} j_{\tau}). \quad (23)$$

Results for N₂

Comparing theory levels for singlet excitations (eV)
9 unoccupied states

Index	IPA	Petersilka	Tamm-Dancoff	CV(2)	Casida	Exp't
1	8.147	9.336	9.271	10.265	9.181	9.31
2	9.670	10.371	9.670	9.670	9.670	9.92
3	9.670	11.568	10.232	10.793	10.216	10.27
4	9.670	11.568	10.253	10.834	10.236	

Tamm-Dancoff occupations: 8.5 eV (mix of two degenerate versions of #1)
1.997, 2.000, 1.999, 1.999, 1.005, 0.499, 0.499, 0.000, 0.002

Excited-state forces

PHYSICAL REVIEW A **76**, 054501 (2007)

Excited-state forces within time-dependent density-functional theory: A frequency-domain approach

Amit Sitt and Leeor Kronik*

Department of Materials and Interfaces, Weizmann Institute of Science, Rehovoth 76100, Israel

Sohrab Ismail-Beigi

Department of Applied Physics, Yale University, New Haven, Connecticut 06520, USA

James R. Chelikowsky

*Center for Computational Materials, Institute for Computational Engineering and Science, Departments of Physics
and Chemical Engineering, University of Texas, Austin, Texas 78712, USA*

(Received 28 May 2007; published 2 November 2007)

Excited-state forces

$$\Omega_I^2 = \mathbf{F}_I^\dagger \Omega_I^2 \mathbf{F}_I = \mathbf{F}_I^\dagger \mathbf{Q} \mathbf{F}_I,$$

$$\partial_{R_a} \Omega_I = \frac{\partial_{R_a} (\mathbf{F}_I^\dagger \mathbf{Q} \mathbf{F}_I)}{2\Omega_I}.$$

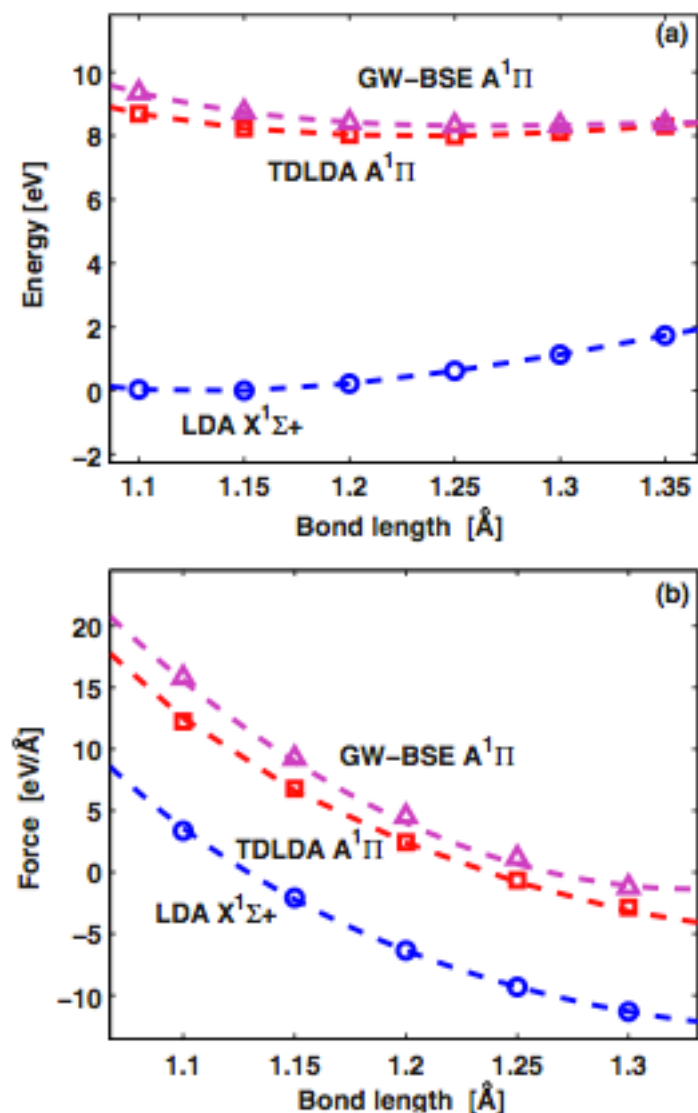
$$D_{\alpha\beta\gamma\delta} = \int \int \partial_{R_a} [\psi_\alpha^*(\mathbf{r})] \psi_\beta^*(\mathbf{r}) \left(\frac{1}{|\mathbf{r} - \mathbf{r}'|} \right) \psi_\gamma(\mathbf{r}') \psi_\delta(\mathbf{r}') d\mathbf{r} d\mathbf{r}'.$$

$$P_{\alpha\beta\gamma\delta} = \int \int \partial_{R_a} [\psi_\alpha^*(\mathbf{r})] \psi_\beta^*(\mathbf{r}) \\ \times \left(\frac{\delta^2 E_{xc}}{\delta \rho(\mathbf{r}) \delta \rho(\mathbf{r}')} \right) \psi_\gamma(\mathbf{r}') \psi_\delta(\mathbf{r}') d\mathbf{r} d\mathbf{r}'.$$

$$= \int \psi_i^*(\mathbf{r}) \psi_j^*(\mathbf{r}) \left(\frac{\delta^3 E_{xc}}{\delta \rho^3(\mathbf{r})} \right) \frac{\partial \rho}{\partial R_a} \psi_k(\mathbf{r}) \psi_l(\mathbf{r}) d\mathbf{r}.$$

Excited-state forces

For the proof-of-concept calculations presented below, all pertinent derivatives were calculated by explicit small nucleus displacements with all derivatives of ground-state quantities calculated by finite difference. While this approach is straightforward and relatively easy to implement, it is clearly computationally inefficient. However, as shown below, it suffices for testing the above formalism. We expect the use of density-functional perturbation theory [23] to increase the efficiency substantially.



Debugging

Here are several strategies that can be used.

compile with `-g -C, -Mbounds, -Warn` all etc. depending on compiler to check array bounds etc.

`valgrind` CPU emulator to find memory leaks (available as Ubuntu package) (<http://valgrind.org>). Compile with `-g`, then run executable in `valgrind`.

`gdb` GNU debugger (available as Ubuntu package) (<http://sourceware.org/gdb>). On supercomputers, packages such as Totalview or DDT are available for parallel debugging. Compile with `-g`, then run executable in debugger.

`electric-fence` (available as Ubuntu package). Link code with `-lefence` and code will die with segmentation fault at the origin of memory corruption errors. Be careful, it uses a ton of extra memory.

(<http://manpages.ubuntu.com/manpages/intrepid/man3/libefence.3.html>) written at Pixar!

link code with `-lmcheck` to investigate `glibc` memory corruption errors (<http://www.gnu.org/s/hello/manual/libc/Heap-Consistency-Checking.html>)

link code with `-ffpe-trap=invalid,zero,overflow,underflow` (gcc only) to catch floating-point exceptions (<http://gcc.gnu.org/onlinedocs/gcc-4.5.3/gfortran/Debugging-Options.html>) so you can see where NaN's are created.

General reference: <http://flylib.com/books/en/1.381.1.37/1/>

Optimization: erroneous results at `-O3` probably indicate latent problems at `-O0`.