VASP Tutorial: Dielectric properties and the Random-Phase-Approximation (RPA)

University of Vienna,
Faculty of Physics and Center for Computational Materials Science,
Vienna, Austria

Setting up a VASP calculation

VASP requires 4 input files to run a calculation:

- INCAR
- POSCAR
- KPOINTS
- POTCAR

I: The INCAR file

The INCAR file contains the input parameters that steer the calculation:

- The default values set by VASP itself are a clever choice for most standard calculations

- These standard settings may be modified to specify:

  - What kind of calculation you want to do:
    SCF calculation, DOS, dielectric properties, …

  - Basic inputs concering the required precision, the requested level of convergence, ...

For a list of all INCAR-tags have a look at:

- The VASP manual: http://cms.mpi.univie.ac.at/vasp/vasp/vasp.html
  Index: http://cms.mpi.univie.ac.at/vasp/vasp/Index.html
- The VASP wiki: http://cms.mpi.univie.ac.at/wiki/index.php/Main_page
  INCAR-tags: http://cms.mpi.univie.ac.at/wiki/index.php/Category:INCAR

II: The POSCAR file

In the POSCAR file you specify the structure (Bravais lattice and basis):

fcc: Ni	Header (comment)
3.53	Overall scaling constant
0.5 0.5 0.0 0.0 0.5 0.5 0.5 0.0 0.5	Bravais matrix
Ni	Name(s) of atomic type(s)
1	Number of atoms (of each type)
Selective Dynamics	(optional: selective dynamics)
Cartesian	Cartesian or Direct coordinates
0 0 0 (T T T)	positions of the atoms

III: The KPOINTS file

In the KPOINTS file you specify the points VASP will use to sample the first Brillouin zone in reciprocal space

Automatic mesh	Header (comment)
0	N_k=0: automatic mesh generation
G (M)	**Γ**-centered (G) mesh or Monkhorst-Pack (M) grid
4 4 4	# of subdivisions N_i along $\vec{b}_i$
0. 0. 0.	Optionally shift the mesh (s_i)

IV: The POTCAR file

The POTCAR file has to contain the PAW datasets for all atomic types you have specified in your POSCAR file:

VASP comes with a library of PAW datasets, (one or more) for most elements of the periodic table:

- Each individual PAW data set starts with a descriptive section, specifying amongst other things:
  - Parameters that were required to generate the dataset:
  - Number of valence electrons
  - Atomic mass
  - Default energy cutoffs

- When your unit cell contains more than one type of atom you have to concatenate the corresponding PAW datasets in the same order as you have specified the different atomic types in your POSCAR file.

- You should not mix PAW datasets generated with different exchange-correlation functionals.

OUTPUT files

OUTCAR

- detailed output of a VASP run, including:
  - a summary of the input parameters
  - information about the individual electronic steps: total energy, Kohn-Sham eigenvalues, Fermi-energy.
  - stress tensors
  - forces in the atoms
  - local charges, magnetic moments
  - dielectric properties
  - ... and a great many things more ...
- The amount of output written onto OUTCAR can be chosen by means of the NWRITE-tag in the INCAR file.

OSZICAR & stdout

- give a short summary of the self-consistency-cycle
  - chosen SCF algorithm
  - convergence of energy and charge density
  - free energies, total magnetic moment of the cell

OUTPUT files

CONTCAR & XDATCAR

- CONTCAR: updated geometry data *at the end of a run*
  - lattice parameter
  - Bravais matrix
  - ionic positions
  - velocities
- the format of the CONTCAR is the same as for POSCAR:
  It can be directly be used for a continuation run (copy CONTCAR to POSCAR)
- XDATCAR: updated ionic positions of each ionic step

DOSCAR, CHGCAR & WAVECAR

- DOSCAR: total DOS and integrated DOS, (local partial DOS)
- CHGCAR: the charge density
- WAVECAR: plane wave coefficients of the orbitals.
  Can be used to restart from a previous run

Documentation

- The VASP manual (http://cms.mpi.univie.ac.at/vasp/vasp/vasp.html)
  Index: http://cms.mpi.univie.ac.at/vasp/vasp/Index.html

- The VASP wiki (http://cms.mpi.univie.ac.at/wiki/index.php/Main_page)
  INCAR-tags: http://cms.mpi.univie.ac.at/wiki/index.php/Category:INCAR

Topics

Frequency dependent dielectric properties

RPA quasi-particle gaps (GW)

GW bandstructure

RPA total energies (ACFDT)

The Bethe-Salpeter Equation (BSE): excitonic effects

BSE: increase k-point sampling

Frequency dependent dielectric properties (ex.: SiC_dielectric)

Goal: calculate the frequency dependent dielectric function of SiC at two levels of aproximation:

- Independent-Particle-Approximation (IPA)
- Random-Phase-Approximation (RPA)

Minimal variety:

```
cd SiC_dielectric
./doall.sh
./plotall.sh
```

Frequency dependent dielectric properties (ex.: SiC dielectric)

- Step 1: a “standard” DFT (PBE) groundstate calculation

INCAR:

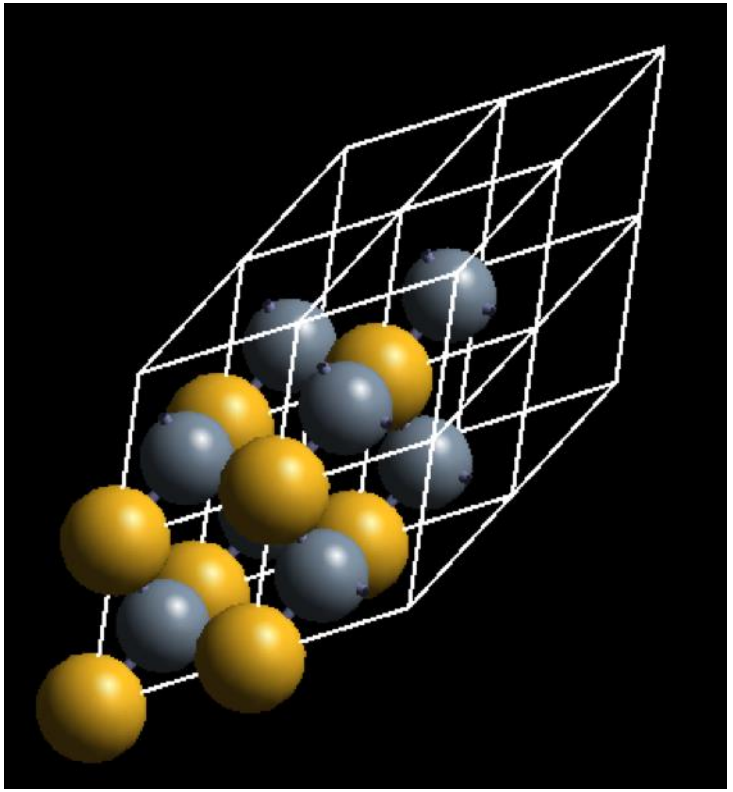
```
ISMEAR = 0      | Gaussian smearing
SIGMA = 0.01    | Set small smearing width
EDIFF = 1.E-8   | Set tight convergence criterium
```

KPOINTS:

```
6x6x6
 0
G
 6 6 6
 0 0 0
```

POSCAR:

```
system SiC
4.35
0.5 0.5 0.0
0.0 0.5 0.5
0.5 0.0 0.5
1 1
cart
0.00 0.00 0.00
0.25 0.25 0.25
```



Frequency dependent dielectric properties (ex.: SiC dielectric)

- Step 2: the Independent-Particle-Approximation (IPA), using `LOPTICS=.TRUE.`

INCAR:

`ALGO = Exact`	`Exact diagonalisation of Hamiltonian`
`NBANDS = 64`	`Keep 64 bands after diagonalisation`
`LOPTICS = .TRUE.`	`Compute frequency dependent dielectric function in the IPA, using a sum over unoccupied states (perturbation theory)`
`CSHIFT = 0.100`	`Complex shift used in Kramers-Kronig transformation`
`NEDOS = 2000`	`Number of bins in the DOS histogram`
`ISMEAR = 0` `SIGMA = 0.01`	`Gaussian smearing` `Set small smearing width`
`EDIFF = 1.E-8`	`Set tight convergence criterium`

In the OUTCAR (or OUTCAR.LOPTICS) file, you'll find the frequency dependent dielectric function, look for:

```
frequency dependent IMAGINARY DIELECTRIC FUNCTION (independent particle, no local field effects)
     E(ev)       X          Y          Z         XY         YZ         ZX
  -------------------------------------------------------------------------------
```

and

```
frequency dependent      REAL DIELECTRIC FUNCTION (independent particle, no local field effects)
     E(ev)       X          Y          Z         XY         YZ         ZX
  -------------------------------------------------------------------------------
```

N.B.: This calculation needs the orbitals (WAVECAR file) of Step 1.

Frequency dependent dielectric properties (ex.: SiC dielectric)

- Step 3: the Random-Phase-Approximation (RPA), using `ALGO=CHI`

INCAR:

`ALGO = CHI`	`Compute dielectric function including local field effects in the RPA`
`NBANDS = 64`	`Use the same #-of-bands as in Step 2. (LOPTICS=.TRUE.) otherwise the WAVEDER file is not read correctly`
`ISMEAR = 0` `SIGMA = 0.01`	`Gaussian smearing` `Set small smearing width`
`EDIFF = 1.E-8`	`Set tight convergence criterium`
`LWAVE = .FALSE.` `LCHARG= .FALSE.`	`Skip writing WAVECAR and CHGCAR`

In the OUTCAR (or OUTCAR.CHI) file, you'll find the frequency dependent dielectric function in the RPA, look for:

```
INVERSE MACROSCOPIC DIELECTRIC TENSOR (including local field effects in RPA (Hartree))
------------------------------------
```

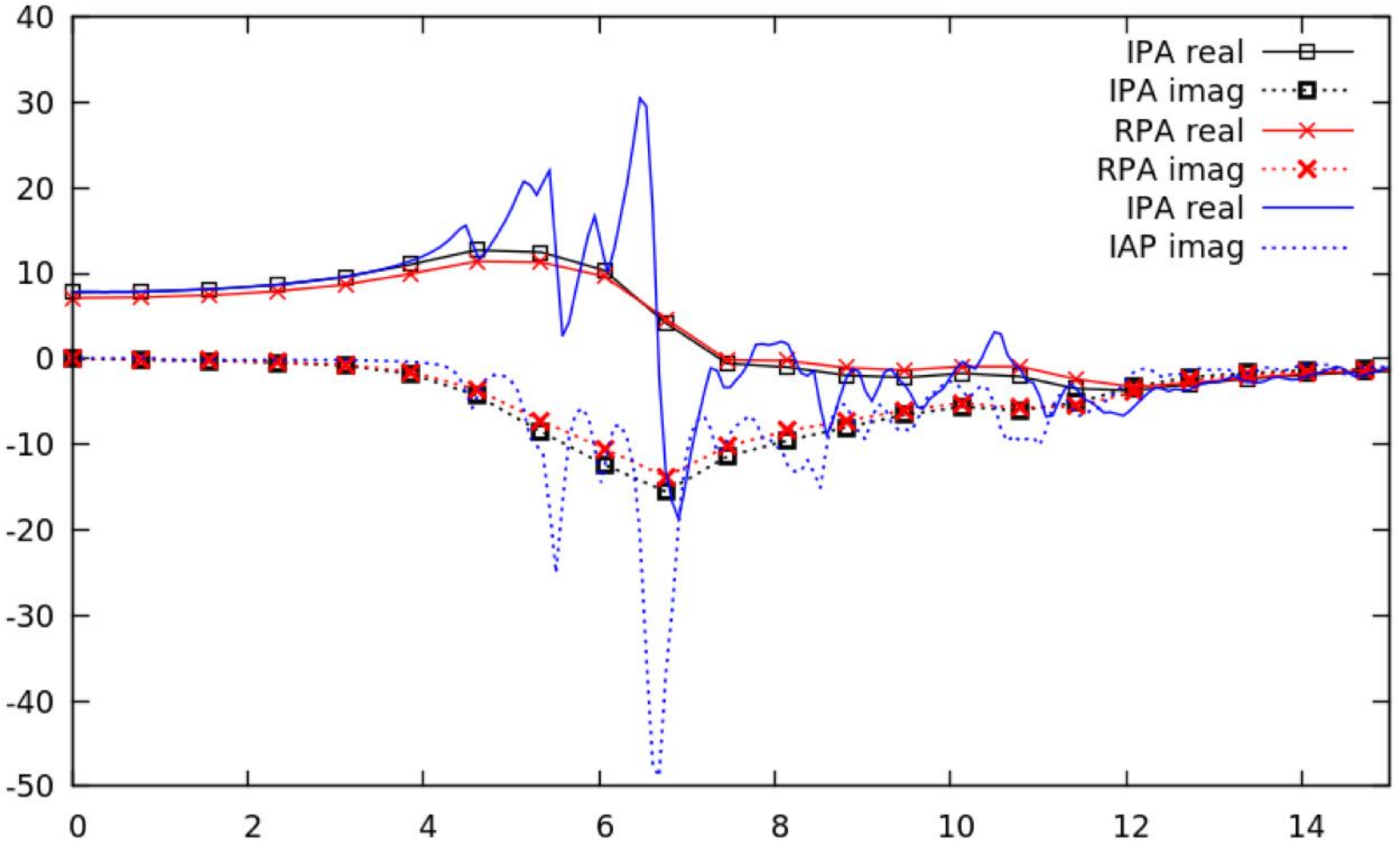
and in the IPA, after:

```
HEAD OF MICROSCOPIC DIELECTRIC TENSOR (INDEPENDENT PARTICLE)
------------------------------------
```

N.B.: This calculation needs the orbitals (WAVECAR file) , and the derivative of the orbitals w.r.t. the Bloch wave vectors (WAVEDER file) written in Step 2.

Frequency dependent dielectric properties (ex.: SiC dielectric)

./plotall.sh



RPA quasi-particle gaps (GW) (ex.: Si_GW_gap)

Goal: calculate quasi-particle (QP) bandgaps in the Random-Phase-Approximation

- Single-shot-GW (G_0W_0)
- Optional: partial self-consistency.
  Update the QP-energies in the Green's function (GW_0).

Minimal variety:

```
cd Si_GW_gap
./doall.sh
./gap_GW.sh OUTCAR.G0W0
```

RPA quasi-particle gaps (GW) (ex.: Si_GW_gap)

WORKFLOW of GW calculations

The workflow of GW calculations consists of three consecutive steps:

Step 1: a "standard DFT groundstate calculation

Step 2: compute additional DFT "virtual" orbitals (empty states):

- Needs the WAVECAR from Step 1.

Step 3: the actual GW calculation:

- Needs WAVECAR and WAVEDER files from Step 2.

N.B.: have a look at doall.sh

RPA quasi-particle gaps (GW) (ex.: Si_GW_gap)

- Step 1: a “standard” DFT (PBE) groundstate calculation

INCAR:

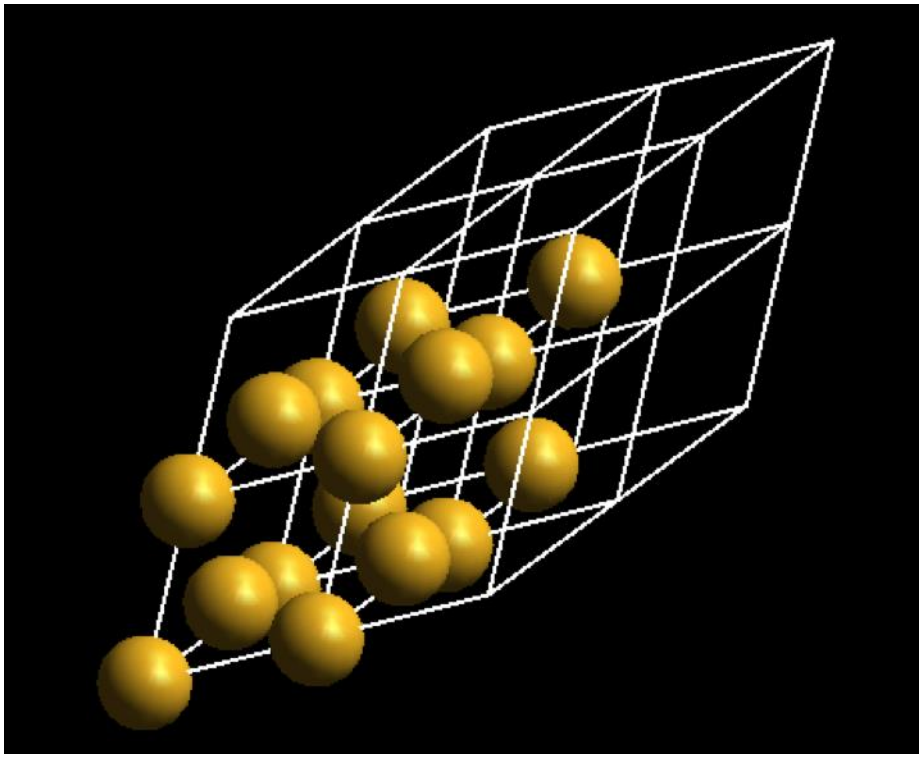
```
ISMEAR = 0      | Gaussian smearing
SIGMA = 0.05    | Set small smearing width
EDIFF = 1.E-8   | Set tight convergence criterium
```

KPOINTS:

```
6x6x6
 0
G
 6 6 6
 0 0 0
```

POSCAR:

```
system Si
5.430
0.5 0.5 0.0
0.0 0.5 0.5
0.5 0.0 0.5
2
cart
0.00 0.00 0.00
0.25 0.25 0.25
```



RPA quasi-particle gaps (GW) (ex.: Si_GW_gap)

- Step 2: compute additional DFT "virtual" orbitals (empty states):

  INCAR:

ALGO = Exact	Exact diagonalisation of Hamiltonian
NBANDS = 64	Keep 64 bands after diagonalisation
LOPTICS = .TRUE.	Compute derivative of the orbitals w.r.t. the Bloch wave vector (stored in the WAVEDER file)
CSHIFT = 0.100	Complex shift used in Kramers-Kronig transformation
ISMEAR = 0 SIGMA = 0.05	Gaussian smearing Set small smearing width
EDIFF = 1.E-8	Set tight convergence criterium

N.B.: In this step one needs to set LOPTICS=.TRUE. to have VASP calculate the derivative of the orbitals w.r.t. the Bloch wave vector (stored in the WAVEDER file). These are needed to correctly describe the long-wavelength limit of the dielectric screening.

N.B.: This calculation needs the orbitals (WAVECAR file) of Step 1.

RPA quasi-particle gaps (GW) (ex.: Si_GW_gap)

- Step 3: RPA quasiparticles: single-shot GW (G_0W_0)

INCAR:

ALGO = GW0	Algorithm used for G0W0 and GW0 calculations: for G0W0 set NELM=1 (Default) for GW0 set NELM=n
LSPECTRAL = .TRUE. NOMEGA = 50	Number of points used in the frequency integration
NBANDS = 64	Use the same #-of-bands as in Step 2. (LOPTICS=.TRUE.) otherwise the WAVEDER file is not read correctly
ISMEAR = 0 SIGMA = 0.05	Gaussian smearing Set small smearing width
EDIFF = 1.E-8	Set tight convergence criterium

In the OUTCAR (or OUTCAR.G0W0) file, you'll find the RPA quasi-particle energies after:

```
QP shifts <psi_nk| G(iteration)W_0 |psi_nk>: iteration 1
for sc-GW calculations column KS-energies equals QP-energies in previous step
and V_xc(KS)=  KS-energies - (<T + V_ion + V_H > + <T+V_H+V_ion>^1  + <V_x>^1)

k-point   1 :       0.0000    0.0000    0.0000
 band No.  KS-energies  QP-energies   sigma(KS)   V_xc(KS)     V^pw_x(r,r')   Z            occupation

     1      -6.4888      -6.8443     -11.0050     -10.4570     -17.5189       0.6487       2.0000
    ..         ..           ..           ..           ..           ..           ..           ..
```

To quickly obtain an estimate of the QP bandgap, type: ./grepall.sh OUTCAR.G0W0

N.B.: This calculation needs the orbitals (WAVECAR file) , and the derivative of the orbitals w.r.t. the Bloch wave vectors (WAVEDER file) written in Step 2.

RPA quasi-particle gaps (GW) (ex.: Si_GW_gap)

- Step 4: RPA quasiparticles: partial self-consistency.
  Update the QP-energies in the Green's function (GW_0)

  INCAR: add the following line

`NELM = 4`	`for G0W0 set NELM=1 (Default)` `for GW0  set NELM=n (in this case 4)`

  Make sure the WAVECAR and WAVEDER files of Step 2. are present:

  ```
  cp WAVECAR.DIAG WAVECAR
  cp WAVEDER.DIAG WAVEDER
  ```

  and start the GW calculation:

  ```
  ./job.sh
  ```

  To obtain the GW0 quasi-particle gap inspect the QP-energies at the end of the OUTCAR file, or use:

  ```
  ./gap_GW.sh OUTCAR
  ```

GW bandstructure (ex.: SrVO3_GW_band)

Goal: calculate the DFT, RPA (G_0W_0), and HSE (hybrid functional) bandstructure of the Vanadium t_{2g}-manifold in $SrVO_3$, using:

- VASP for the DFT, GW, and hybrid function calculations.
- wannier90 to construct the bandstructure out of VASP output.

Minimal variety:

```
cd SrVO3_GW_band
./doall.sh
./plotall.sh
```

and maybe:

```
./plotdos.sh OUTCAR.DFT
./plotdos.sh OUTCAR.GW0
./plotdos.sh OUTCAR.HSE
```

GW bandstructure (ex.: SrVO3_GW_band)

This example consists of a series of consecutive calculations (see doall.sh):

Step 1: a "standard DFT groundstate calculation

Step 2: compute additional DFT "virtual" orbitals (empty states):

- Needs the WAVECAR from Step 1.

Step 3: the actual GW calculation:

- Needs WAVECAR and WAVEDER files from Step 2.

Step 4: obtain the *lm*-decomposed site resolved density-of-states of the GW calculation, and construct the bandstructure of the Vanadium t_{2g}-manifold in $SrVO_3$ using wannier90.

- Needs the WAVECAR file of Step 3.

Step 5: (optional) estimate what DOS and bandstructure a hybrid functional would yield.

- Needs the WAVECAR file of Step 2.

Steps 1, 4, 5 include a wannier90 run to construct Maximally-Localized-Wannier-Functions (MLWFs) for the Vanadium t_{2g} states. These are use to interpolate the bandstructure of the Vanadium t_{2g} manifold.

GW bandstructure (ex.: SrVO3_GW_band)

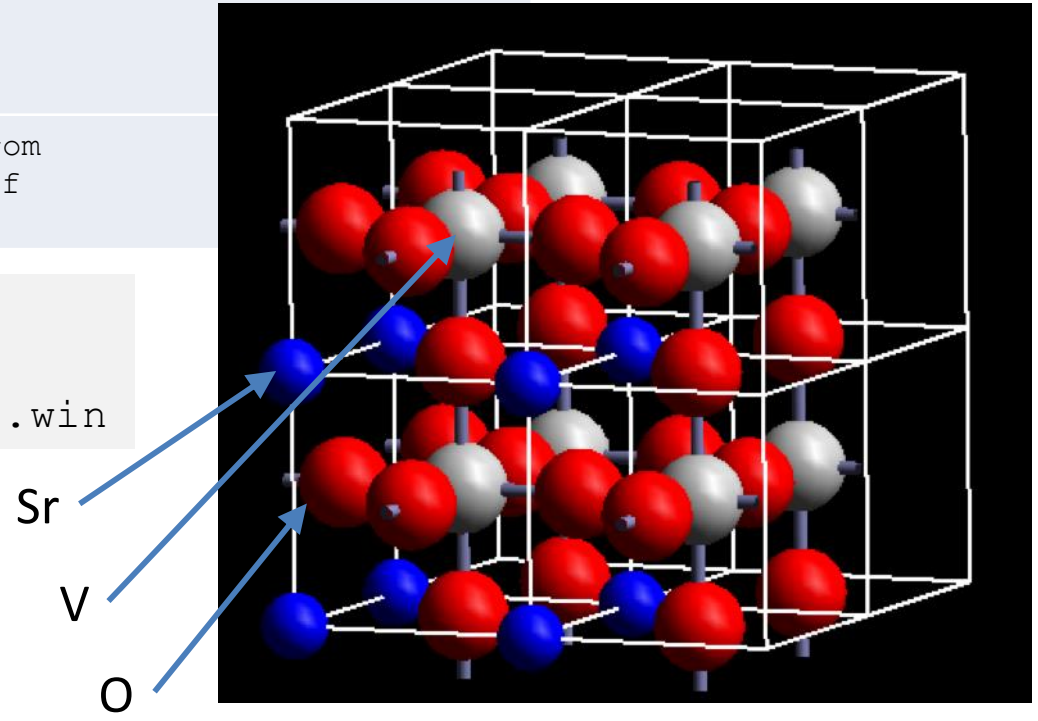
- Step 1: a “standard” DFT (PBE) groundstate calculation

INCAR (copy INCAR.DFT INCAR):

NBANDS = 36	
ISMEAR = -5	Tetrahedron method with Blöchl corrections
EMIN = -20 EMAX = 20 NEDOS = 1000	Energy range and number of bins in DOS
EDIFF = 1.E-8	Set tight convergence criterium
KPAR = 2	Additional parallelization over k-points
LORBIT = 11	lm-decomposed site resolved density-of-states
LWANNIER90_RUN=.TRUE.	Call wannier90 from VASP at the end of the run

- wannier90 takes its input from the file *wannier90.win.* To run this example:
  `cp wannier90.win.dft wannier90.win`

N.B.: KPOINTS file is missing:
VASP generates one automatically
(in this case $4 \times 4 \times 4$ Γ-centered)



GW bandstructure (ex.: SrVO3_GW_band)

- Step 2: compute additional DFT “virtual” orbitals (empty states):

  INCAR (copy INCAR.DIAG INCAR):

`ISMEAR = -5`	`Tetrahedron method with Blöchl corrections`
`ALGO = Exact` `NELM = 1`	`Exact diagonalisation of the Hamiltonian` `Only 1 electronic step`

`EMIN = -20`
`EMAX = 20`
`NEDOS = 1000`
	`Energy range and number of bins in DOS`

`EDIFF = 1.E-8`	`Set tight convergence criterium`
`NBANDS = 96`	`Keep 96 bands after diagonalisation`
`LOPTICS = .TRUE.`	`Compute derivative of the orbitals w.r.t. the` `Bloch wave vector (stored in the WAVEDER file)`
`KPAR = 2`	`Additional parallelization over k-points`

N.B.: In this step one needs to set LOPTICS=.TRUE. to have VASP calculate the derivative of the orbitals w.r.t. the Bloch wave vector (stored in the WAVEDER file). These are needed to correctly describe the long-wavelength limit of the dielectric screening.

N.B.: This calculation needs the orbitals (WAVECAR file) of Step 1.

GW bandstructure (ex.: SrVO3_GW_band)

- Step 3: RPA quasiparticles: single-shot GW (G_0W_0)

INCAR (copy INCAR.GW INCAR):

ISMEAR = -5	Tetrahedron method with Blöchl corrections
EMIN = -20 EMAX = 20 NEDOS = 1000	Energy range and number of bins in DOS
NBANDS = 96	Use the same #-of-bands as in Step 2., otherwise the WAVEDER file can not be read correctly
ALGO = GW0 NELM = 1 PRECFOCK = Fast	Algorithm used for G0W0 and GW0 calculations: for G0W0 set NELM=1 (Default) for GW0 set NELM=n
ECUTGW = 100 NOMEGA = 200	Energy cutoff for response functions Number of points used in the frequency integration
KPAR = 2	

In the OUTCAR (or OUTCAR.G0W0) file, you'll find the RPA quasi-particle energies after:

```
QP shifts <psi_nk| G(iteration)W_0 |psi_nk>: iteration 1
for sc-GW calculations column KS-energies equals QP-energies in previous step
and V_xc(KS)=  KS-energies - (<T + V_ion + V_H > + <T+V_H+V_ion>^1  + <V_x>^1)

k-point   1 :       0.0000    0.0000    0.0000
 band No.  KS-energies  QP-energies   sigma(KS)   V_xc(KS)     V^pw_x(r,r')   Z            occupation
```

N.B.: This calculation needs the orbitals (WAVECAR file) , and the derivative of the orbitals w.r.t. the Bloch wave vectors (WAVEDER file) written in Step 2.

GW bandstructure (ex.: SrVO3_GW_band)

- Step 4: obtain the *lm*-decomposed site resolved density-of-states of the G_0W_0 calculation, and construct the bandstructure using wannier90.

INCAR (copy INCAR.NONE to INCAR):

`ISMEAR = -5`	`Tetrahedron method with Blöchl corrections`

`EMIN = -20`
`EMAX = 20`
`NEDOS = 1000`
	`Energy range and number of bins in DOS`

`NBANDS = 96`	`Use the same #-of-bands as in Step 2., otherwise the WAVEDER file can not be read correctly`
`ALGO = None` `NELM = 1`	`Do nothing except read the WAVECAR and proceed to post-processing`
`LORBIT = 11`	`lm-decomposed site resolved density-of-states`
`LWANNIER90_RUN=.TRUE.`	`Call wannier90 from VASP at the end of the run`

- wannier90 takes its input from the file *wannier90.win*
  To run this example:

```
cp wannier90.win.gw wannier90.win
```

N.B.: This calculation needs the orbitals (WAVECAR file) written in Step 3.

GW bandstructure (ex.: SrVO3_GW_band)

- Step 5: Compute HSE "eigenvalues" from PBE orbitals:

INCAR (copy INCAR.HSE to INCAR):

ISMEAR = -5	Tetrahedron method with Blöchl corrections
EMIN = -20 EMAX = 20 NEDOS = 1000	Energy range and number of bins in DOS
EDIFF = 1E-8	
KPAR = 2	
NBANDS = 48	
LHFCALC = .TRUE. HFSCREEN = 0.2 PRECFOCK = Fast	Select the HSE range separated DFT/HF hybrid functional Range separation parameter for HSE06
ALGO = Eigenval NELM = 1	Only compute the eigenvalues, do not optimize the orbitals
LWAVE = .FALSE.	Do not write a WAVECAR file
LORBIT = 11	*lm*-decomposed site resolved density-of-states
LWANNIER90_RUN=.TRUE.	Call wannier90 from VASP at the end of the run

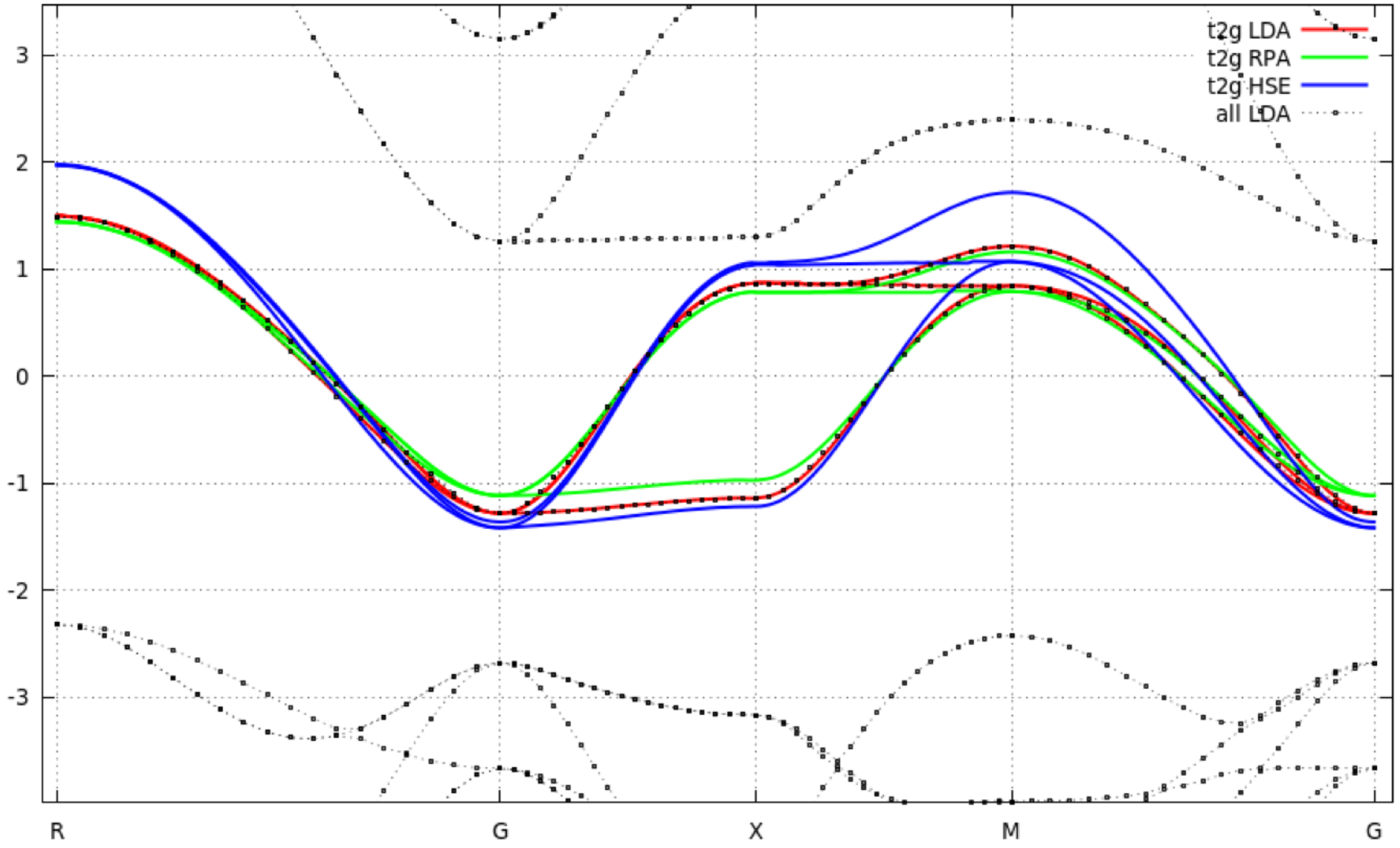
- wannier90 takes its input from the file *wannier90.win*
  To run this example:

```
cp wannier90.win.hse wannier90.win
```

N.B.: This calculation needs the orbitals (WAVECAR file) written in Step 2.

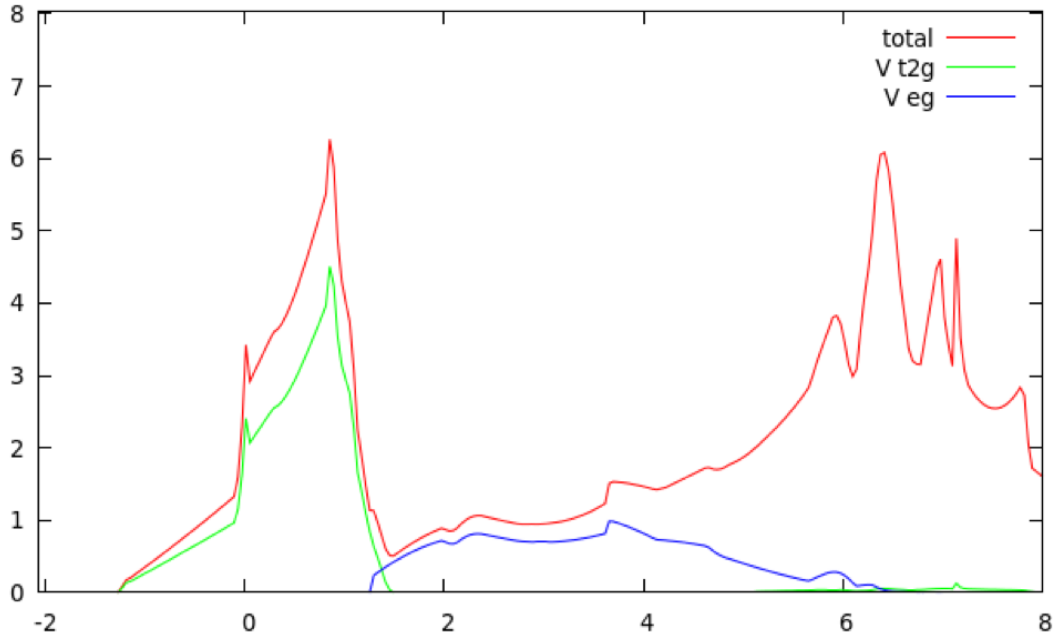
GW bandstructure (ex.: SrVO3_GW_band)

./plotall.sh

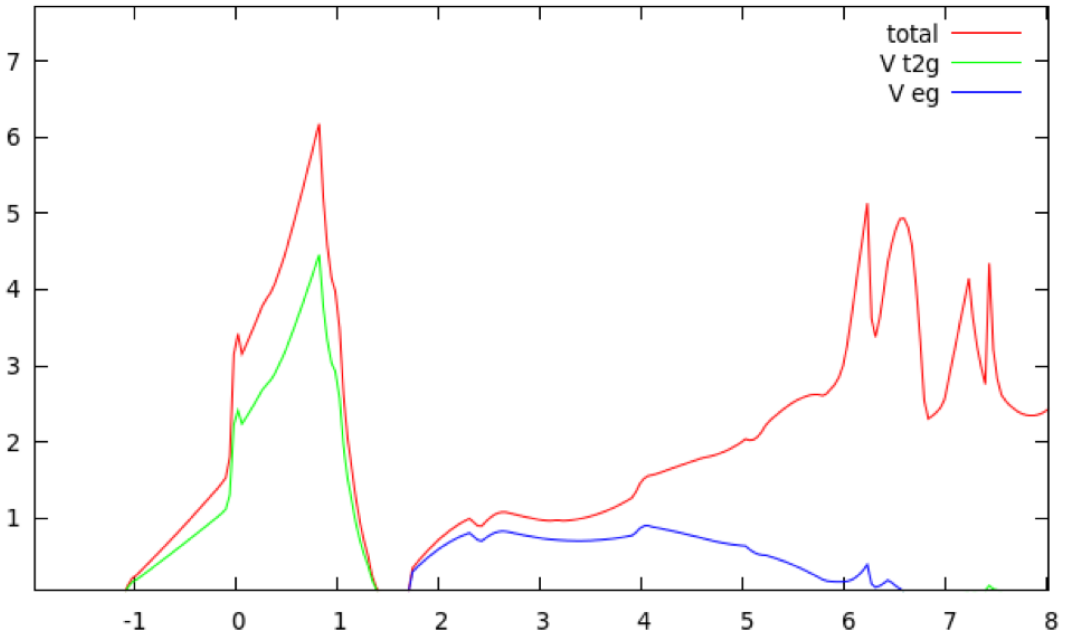


GW bandstructure (ex.: SrVO3_GW_band)

./plotdos.sh DOSCAR.DFT



./plotdos.sh DOSCAR.GW0



RPA total energies (ACFDT) (ex.: Si_ACFDT_vol)

Goal: determine the RPA total energy of cd-Si as a function of the lattice constant

Minimal variety:

```
cd Si_ACFDT_vol
./doall.sh
./plotall.sh
```

RPA total energies (ACFDT) (ex.: Si_ACFDT_vol)

WORKFLOW of RPA total energy (ACFDT) calculations

The workflow of "ACFDT" calculations consists of five consecutive steps:

Step 1: a "standard DFT groundstate calculation, with a "dense" mesh of k-points

Step 2: compute the Hartree-Fock energy with DFT orbitals of Step 1

- Needs the WAVECAR of Step 1.

Step 3: a "standard DFT groundstate calculation, with a "coarse" mesh of k-points

Step 4: compute additional DFT "virtual" orbitals (empty states):

- Needs the WAVECAR from Step 3.

Step 5: the actual ACFDT calculation:

- Needs WAVECAR and WAVEDER files from Step 4.

N.B.: have a look at doall.sh

In case of metallic systems one should use the same k-point grid throughout the calculation, i.e., skip Step 3, and neglect the long-wavelength contributions to the dielectric screening (delete WAVEDER before Step 5).

RPA total energies (ACFDT) (ex.: Si_ACFDT_vol)

- Step 1: a “standard” DFT groundstate calculation with a “dense” k-point mesh

INCAR (copy INCAR.DFT to INCAR):

ISMEAR = 0 SIGMA = 0.05	Gaussian smearing Set small smearing width
EDIFF = 1.E-8	Set tight convergence criterium

KPOINTS (e.g. KPOINTS.12):

```
12x12x12
 0
G
12  12  12
 0   0   0
```

- Step 2: compute the Hartree-Fock energy with the DFT orbitals of Step 1

INCAR (copy INCAR.EXX to INCAR):

ISMEAR = 0 SIGMA = 0.05	Gaussian smearing Set small smearing width
ALGO = Eigenval NELM = 1	Only compute the eigenvalues, do not optimize the orbitals
LWAVE = .FALSE.	Do not write WAVECAR
LHFCALC = .TRUE. AEXX = 1.0 ; ALDAC = 0.0 ; AGGAC = 0.0	Hartree-Fock
NKRED = 2	Downsample recip. space rep. of the Fock potential, to save time (optional)
KPAR = 8	
NBANDS =4	Occupied states only (to save time)

N.B.: This calculation needs the orbitals (WAVECAR file) written in Step 1.

RPA total energies (ACFDT) (ex.: Si_ACFDT_vol)

- Step 3: a “standard” DFT groundstate calculation with a “coarse” k-point mesh

INCAR (copy INCAR.DFT to INCAR):

ISMEAR = 0 SIGMA = 0.05	Gaussian smearing Set small smearing width
EDIFF = 1.E-8	Set tight convergence criterium

KPOINTS (e.g. KPOINTS.6):

```
6x6x6
 0
G
6 6 6
0 0 0
```

- Step 4: compute additional DFT “virtual” orbitals (empty states)

INCAR (copy INCAR.DIAG to INCAR):

ALGO = Exact NELM = 1	Exact diagonalisation of the Hamiltonian Only 1 electronic step
NBANDS = 64	Keep 64 bands after diagonalisation
LOPTICS = .TRUE.	Compute derivative of the orbitals w.r.t. the Bloch wave vector (stored in the WAVEDER file)
ISMEAR = 0 SIGMA = 0.05	

N.B.: In this step one needs to set LOPTICS=.TRUE. to have VASP calculate the derivative of the orbitals w.r.t. the Bloch wave vector (stored in the WAVEDER file). These are needed to correctly describe the long-wavelength limit of the dielectric screening.

N.B.: This calculation needs the orbitals (WAVECAR file) written in Step 3.

RPA total energies (ACFDT) (ex.: Si_ACFDT_vol)

- Step 5: the actual ACFDT calculation

INCAR (copy INCAR.ACFDT to INCAR):

`ALGO = ACFDT`	`Request ACFDT calculation`
`NBANDS = 64`	`Use the same #-of-bands as in Step 4., otherwise the WAVEDER file can not be read correctly`
`ISMEAR = 0` `SIGMA = 0.05`	`Gaussian smearing` `Set small smearing width`

In the OUTCAR (OUTCAR.ACFDT.X.X) you'll find the RPA correlation energy, *e.g.*:

```
 cutoff energy     smooth cutoff    RPA    correlation    Hartree contr. to MP2
-------------------------------------------------------------------------------
            163.563           130.851      -10.7869840331      -19.0268026572
            155.775           124.620      -10.7813600055      -19.0200457142
            148.357           118.685      -10.7744584182      -19.0118291822
            141.292           113.034      -10.7659931963      -19.0017871991
            134.564           107.651      -10.7555712745      -18.9894197881
            128.156           102.525      -10.7428704760      -18.9742991317
            122.054            97.643      -10.7273118140      -18.9556871679
            116.241            92.993      -10.7085991597      -18.9331679971
  linear regression
  converged value                          -10.9079580568      -19.1711146204
```

Take the "converged value", in this case: E_C^{RPA} = -10.9079580568 eV.
(an approximate "infinite basis set" limit)

N.B.: This calculation needs the orbitals (WAVECAR file) , and the derivative of the orbitals w.r.t. the Bloch wave vectors (WAVEDER file) written in Step 4.

RPA total energies (ACFDT) (ex.: Si_ACFDT_vol)

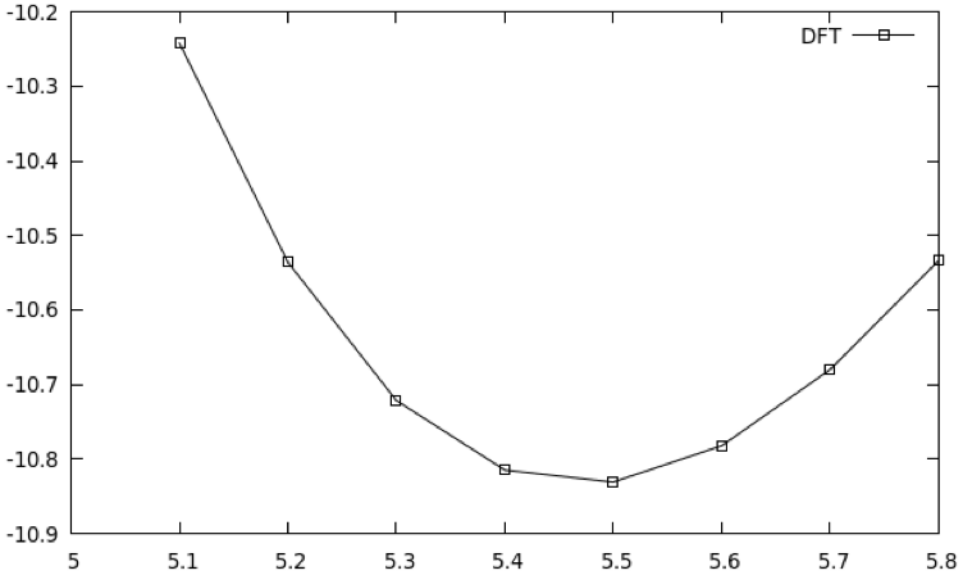
The RPA total energy is given by: $E_{\mathrm{RPA}} = E_{\mathrm{EXX}} + E_C^{\mathrm{RPA}}$

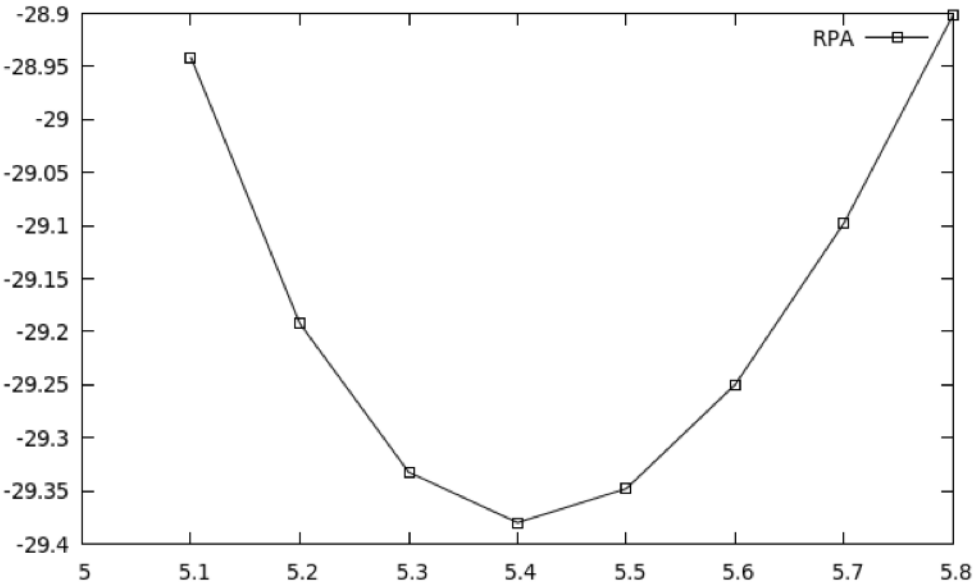
- The Hartree-Fock energy (using DFT orbitals) of Step 2: E_{EXX}
  type: grep ”free energy“ OUTCAR.EXX

  (mind there are two spaces between „free“ and „energy“)

- RPA correlation energy of Step 5: E_C^{RPA}

./plotall.sh





The Bethe-Salpeter-Equation (BSE): excitonic effects (ex.: Si_BSE)

Goal: account for excitonic effects in the frequency dependent dielectric function

- GW0+BSE: Solve the Bethe-Salpeter-Equation on top of GW0

Minimal variety:

```
cd Si_BSE
./doall.sh
./plotall.sh
```

The Bethe-Salpeter-Equation (BSE) (ex.: Si_BSE)

WORKFLOW of a GW0+BSE calculation

The workflow of the GW0+BSE calculation in *job.sh* consists of five consecutive steps:

Step 1: a "standard DFT groundstate calculation

Step 2: compute additional DFT "virtual" orbitals (empty states):

- Needs the WAVECAR from Step 1.

Step 3: the GW0 calculation:

- Needs the WAVECAR and WAVEDER files from Step 2.

Step 4: (optional) use LOPTICS=.TRUE. to plot the IPA dielectric function using the GW0-QP energies instead of DFT eigenenergies

- Needs the WAVECAR file from Step 3.

Step 5: the BSE calculation:

- Needs WAVECAR file from Step 3 and WAVEDER file from Step 2.

N.B.: have a look at doall.sh

The Bethe-Salpeter-Equation (BSE) (ex.: Si_BSE)

- Step 1: a “standard” DFT groundstate calculation

INCAR (copy INCAR.DFT to INCAR):

PREC = Normal ENCUT = 250	Default Set cutoff energy to 250 eV
ISMEAR = 0 SIGMA = 0.01	Gaussian smearing Set small smearing width
KPAR = 2	Additional parallelization
EDIFF = 1.E-8	Set tight convergence criterium

KPOINTS (e.g. KPOINTS.6):

```
6x6x6
 0
G
6 6 6
0 0 0
```

- Step 2: compute additional DFT “virtual” orbitals (empty states)

INCAR (copy INCAR.DIAG to INCAR):

PREC = Normal ; ENCUT = 250	
ALGO = Exact NELM = 1	Exact diagonalisation of the Hamiltonian Only 1 electronic step
ISMEAR = 0 ; SIGMA = 0.01	
NBANDS = 128	Keep 128 bands after diagonalisation
LOPTICS = .TRUE. LPEAD = .TRUE.	Compute derivative of the orbitals w.r.t. the Bloch wave vector (stored in the WAVEDER file) using the PEAD formalism
OMEGAMAX = 40	

N.B.: This calculation needs the orbitals (WAVECAR file) written in Step 1.

The Bethe-Salpeter-Equation (BSE) (ex.: Si_BSE)

- Step 3: RPA quasiparticles: single-shot GW (G_0W_0)

INCAR (copy INCAR.GW0 INCAR):

PREC = Normal ; ENCUT = 250	
ALGO = GW0 ; NELM = 1	Request G0W0 calculation
ISMEAR = 0 ; SIGMA = 0.01	
ENCUTGW = 150 NOMEGA = 50 OMEAGTL = 280 PRECFOCK = Normal	Eenergy cutoff for response functions Number of points for frequency integration
NBANDS = 128	Use the same #-of-bands as in Step 2., otherwise the WAVEDER file can not be read correctly
NBANDSGW = 12	Compute QP energies only for the first 12 bands
KPAR = 2	
LWAVE = .TRUE.	Write WAVECAR (contains QP-energies)

In the OUTCAR (or OUTCAR.GW0) file, you'll find the RPA quasi-particle energies after:

```
QP shifts <psi_nk| G(iteration)W_0 |psi_nk>: iteration 1
for sc-GW calculations column KS-energies equals QP-energies in previous step
and V_xc(KS)=  KS-energies - (<T + V_ion + V_H > + <T+V_H+V_ion>^1  + <V_x>^1)

k-point   1 :       0.0000    0.0000    0.0000
 band No.  KS-energies  QP-energies   sigma(KS)    V_xc(KS)      V^pw_x(r,r')    Z              occupation
```

N.B.: This calculation needs the orbitals (WAVECAR file) , and the derivative of the orbitals w.r.t. the Bloch wave vectors (WAVEDER file) written in Step 2.

The Bethe-Salpeter-Equation (BSE) (ex.: Si_BSE)

- Step 4: (optional) plot IPA dielectric function using G_0W_0 QP-energies

INCAR (copy INCAR.NONE INCAR):

ALGO = Nothing ; NELM = 1	Do nothing except read WAVECAR and proceed to post-processing
NBANDS = 128	
LWAVE = .FALSE.	Do not write WAVECAR
LOPTICS = .TRUE. ; LPEAD = .TRUE.	Compute IPA dielectric function

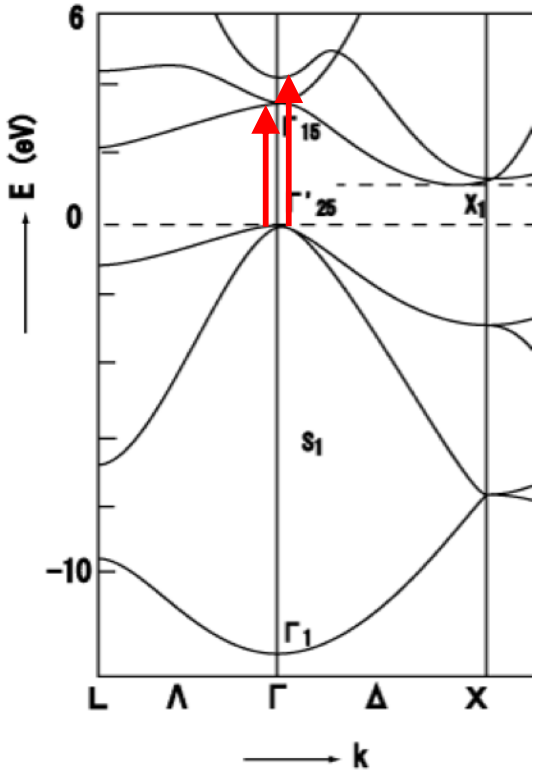
- Step 5: the BSE calculation

INCAR (copy INCAR.BSE INCAR):

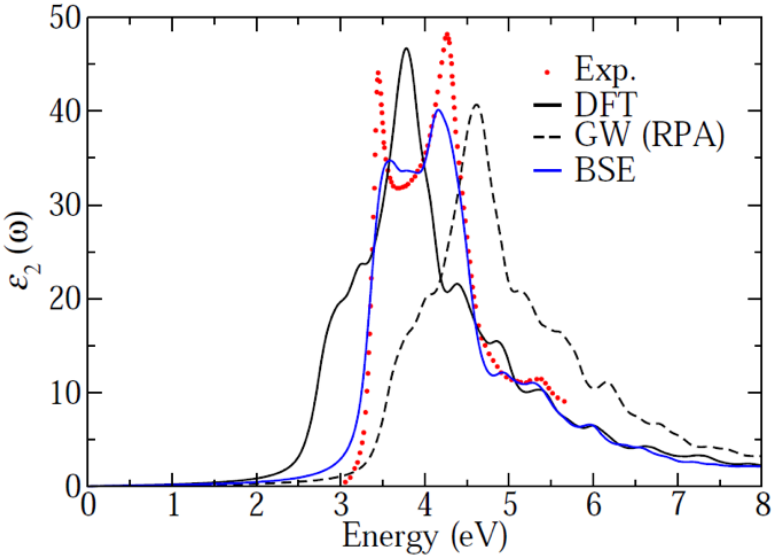
PREC = Normal ; ENCUT = 250	
ALGO = BSE ANTIRES = 0	Request BSE calculation Use the Tamm-Dancoff approximation
ISMEAR = 0 ; SIGMA = 0.01	
ENCUTGW = 150	Energy cutoff for response functions
NBANDS = 128	Use the same #-of-bands as in Step 2., otherwise the WAVEDER file can not be read correctly
NBANDSO = 4 NBANDSV = 8	Setup BSE matrix for NBANDSO HOMOs and NBANDSV LUMOs
OMEGAMAX = 20	
PRECFOCK = Normal	

N.B.: This calculation needs the orbitals (WAVECAR file) of Step 3, and the derivative of the orbitals w.r.t. the Bloch wave vectors (WAVEDER file) written in Step 2.

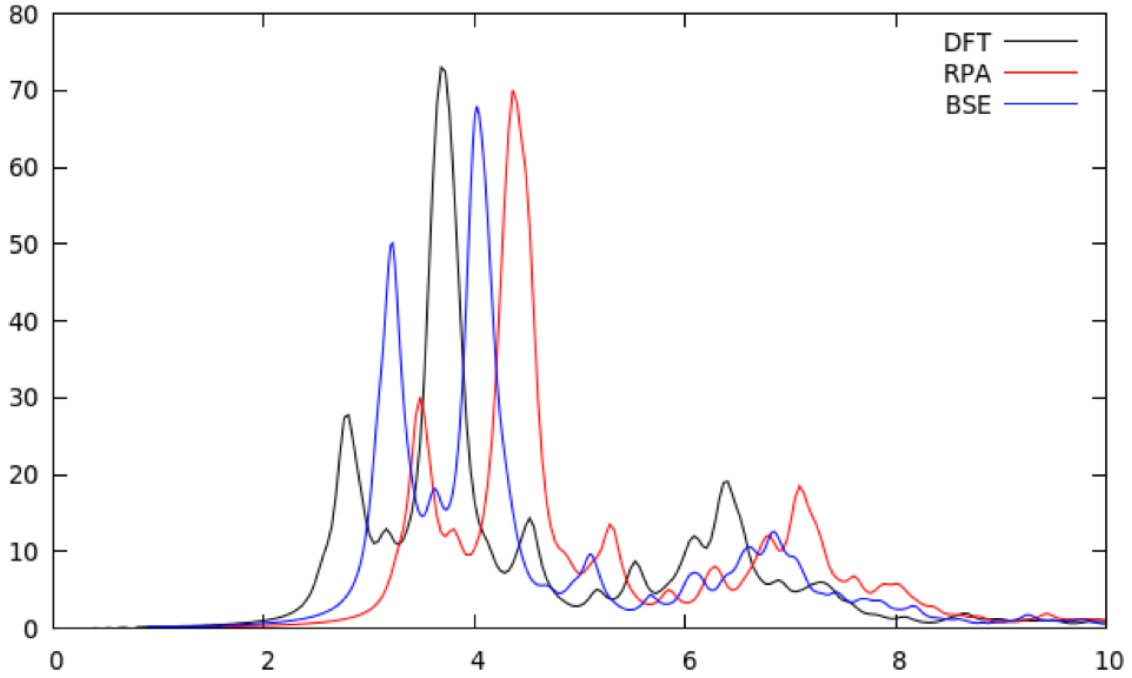
The Bethe-Salpeter-Equation (BSE) (ex.: Si BSE)



Converged w.r.t. **k**-point sampling



./plotall.sh →



BSE: increase k-point sampling (ex.: Si_improve_eps)

Goal: explore two approximate ways to increase the k-point sampling in BSE calculations

- Method I: Averaging over multiple shifted k-point grids

  Compute N independent dielectric functions using different shifted **k**-point grids, and average over them.

- Method II: Model-BSE

  Use a dense **k**-point grid, but with a model dielectric screening function and DFT single-particle energies shifted by a scissor-operator instead of GW QP-energies.

BSE: increase k-point sampling (ex.: Si_improve_eps)

Method I: Averaging over multiple shifted k-point grids

- Construct "shifted" **k**-point grids with the same sampling density:

  Take a $(n \times n \times n)$ **k**-point grid $\rightarrow X_n$ irreducible k-points $\{\boldsymbol{K}_n\}$ with weights W_n and do X_n calculations on $(m \times m \times m)$ **k**-point grids, shifted away from Γ by $\boldsymbol{K}_n$.

- Extract the dielectric function of each of the X_n calculations and average over them w.r.t. to the weights W_n.

We then have effectively constructed the result for a $(nm) \times (nm) \times (nm)$ grid.
BUT: interactions with a range longer than *m* times the cell size are not taken into account.

BSE: increase k-point sampling (ex.: Si_improve_eps)

Method I: Averaging over multiple shifted k-point grids

Example: $n = m = 4 \longrightarrow 16 \times 16 \times 16$ effectively

IBZ of the $(n \times n \times n)$ **k**-points grid for *n*=4:

```
4x4x4 Gamma centered grid:
Found      8 irreducible k-points:

 Following reciprocal coordinates:
            Coordinates                 Weight
  0.000000  0.000000  0.000000       1.000000
  0.250000 -0.000000  0.000000       8.000000
  0.500000 -0.000000  0.000000       4.000000
  0.250000  0.250000  0.000000       6.000000
  0.500000  0.250000  0.000000      24.000000
 -0.250000  0.250000  0.000000      12.000000
  0.500000  0.500000  0.000000       3.000000
 -0.250000  0.500000  0.250000       6.000000
```

K_n W_n

Do 8 calculations with different shifted $(m \times m \times m)$ KPOINTS files for *m*=4:

```
4x4x4
 0
G
4   4   4
Kx  Ky  Kz
```

In *doall-average.sh* this scheme is implemented for n = 4 and m= $NKPT.
The dielectric functions are extracted and averaged accordingly. You can choose up to which level of theory (DFT, RPA, BSE) the dielectric function is computed by commenting out the corresponding lines in the script. (*default:* all the way up to BSE!)

BSE: increase k-point sampling (ex.: Si_improve_eps)

Method I: Averaging over multiple shifted k-point grids

A few things have to be done slightly differently compared to Si_BSE.

INCAR.DIAG / INCAR.NONE:

PREC = Normal ; ENCUT = 250	
ALGO = Exact NELM = 1	Exact diagonalisation of the Hamiltonian Only 1 electronic step
ISMEAR = 0 ; SIGMA = 0.01	
NBANDS = 128	Keep 128 bands after diagonalisation
LOPTICS = .TRUE. LPEAD = .FALSE.	Compute derivative of the orbitals w.r.t. the Bloch wave vector (stored in the WAVEDER file) Do *not* use the PEAD formalism
ISYM = -1	Switch off symmetry
OMEGAMAX = 40	

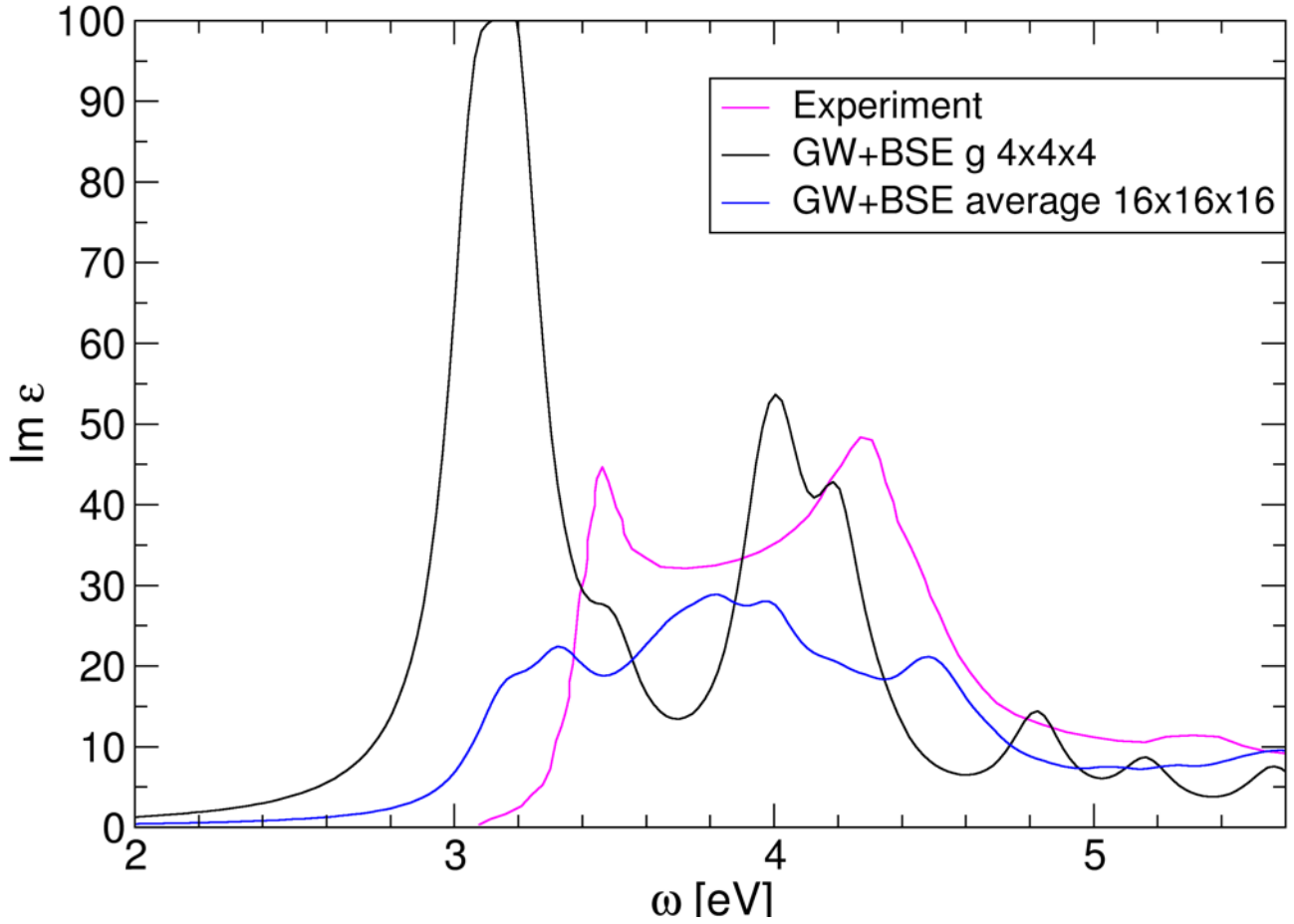
Symmetry has to be switched off in all calculations!

Because of the ”shifted” **k**-point grids we can not use the PEAD method to compute the derivatives of the orbitals w.r.t. the Bloch wave vectors.

BSE: increase k-point sampling (ex.: Si_improve_eps)

Method I: Averaging over multiple shifted k-point grids

$n = m = 4 \longrightarrow 16 \times 16 \times 16$ effectively



BSE: increase k-point sampling (ex.: Si_improve_eps)

Method II: model-BSE

- The dielectric function $\epsilon^{-1}_{\mathbf{G},\mathbf{G}'}(\mathbf{k})$ is replaced by a local model function

$$\epsilon^{-1}(\mathbf{k}+\mathbf{G}) = 1 - (1-\epsilon_\infty^{-1}) e^{\frac{-(2\pi|\mathbf{k}+\mathbf{G}|)^2}{4\lambda^2}}$$

With this model dielectric screening the Coulomb kernel is diagonal w.r.t. **G** and **G'**:

Type equation here.

$$W^{cv\mathbf{k}}_{c'v'\mathbf{k}} = \frac{4\pi e^2}{\Omega} \sum_{\mathbf{G}} B^{c\mathbf{k}}_{c'\mathbf{k}}(\mathbf{G}) \frac{\epsilon^{-1}(\mathbf{k}+\mathbf{G})}{|\mathbf{k}+\mathbf{G}|^2} B^{v\mathbf{k}}_{v'\mathbf{k}}(\mathbf{G})$$

where the $B^{n\mathbf{k}}_{n'\mathbf{k}}(\mathbf{G})$ denote so-called overlap charge densities of the cell-periodic parts of Bloch functions *n***k** and *n*'**k**.

- Besides a model dielectric function we use approximate "quasiparticle" energies:

$$\varepsilon_i^{\text{app.}} = \varepsilon_i^{\text{DFT}} + E_{\text{scissor}} \quad \text{and} \quad E_{\text{scissor}} = E_g^{GW} - E_g^{\text{DFT}}$$

i.e., single particle eigenenergies shifted by a scissor operator

BSE: increase k-point sampling (ex.: Si_improve_eps)

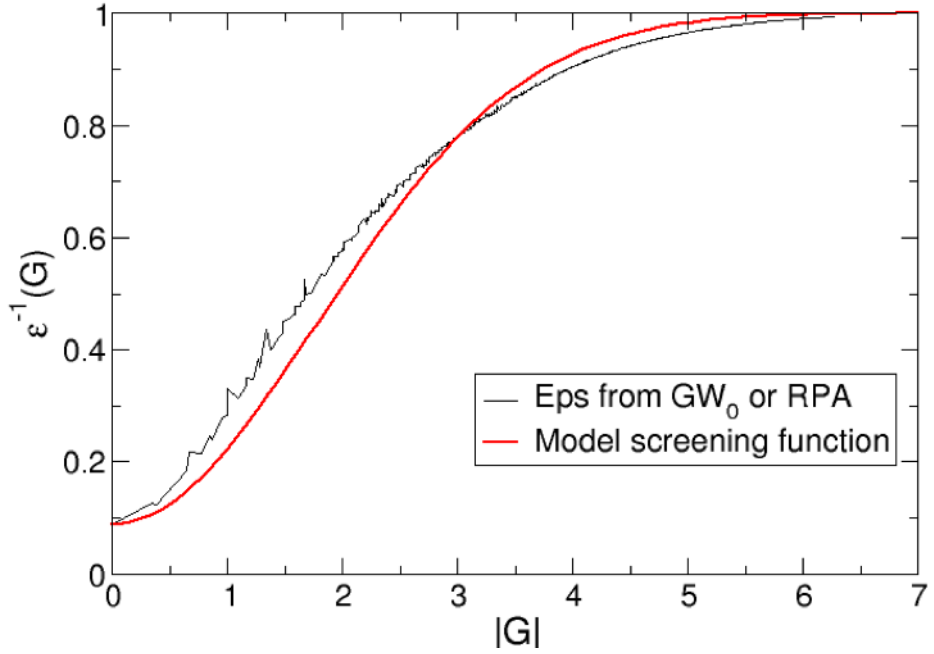
Method II: model-BSE

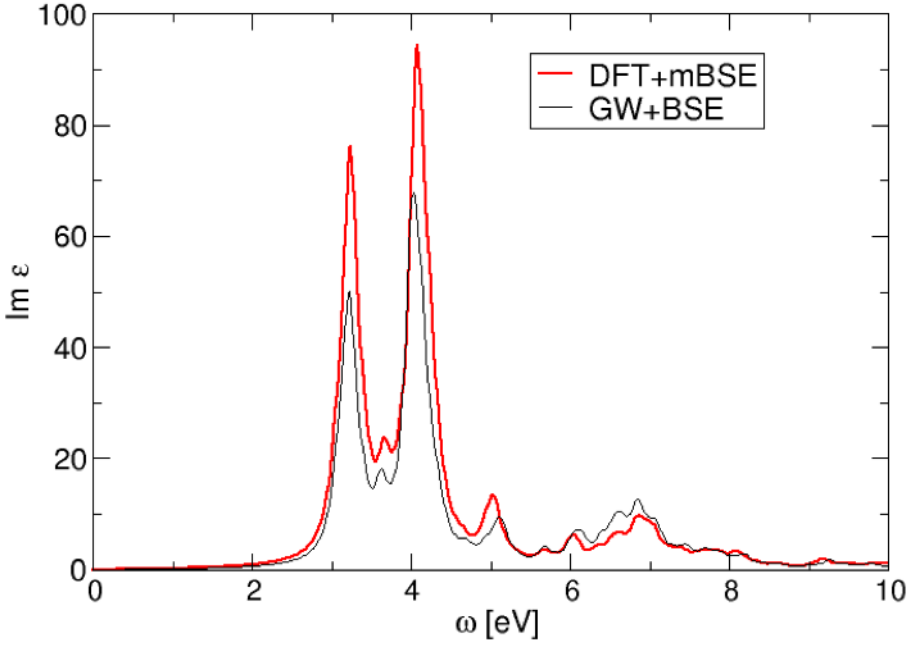
Extract **G=G'** dielectric function from the vasprun.xml file from the previous GW calculation with the *extract_die_G* script (or view *dieG_g6x6x6-GW0.dat*).
Then fit ϵ_∞^{-1} and λ in the model function:

$$\epsilon^{-1}(\mathbf{k}+\mathbf{G}) = 1 - (1-\epsilon_\infty^{-1}) e^{\frac{-(2\pi|\mathbf{k}+\mathbf{G}|)^2}{4\lambda^2}}$$

INCAR:

AEXX = 0.088	ϵ_∞^{-1}
HFSCREEN = 1.26	λ





Check the GW+BSE and DFT+mBSE spectra for consistency!

BSE: increase k-point sampling (ex.: Si_improve_eps)

Method II: model-BSE

- Step 1: standard DFT run
  INCAR (copy INCAR.DFT to INCAR):

```
PREC = Normal ; ENCUT = 250.0
ISMEAR = 0 ; SIGMA = 0.01
EDIFF = 1.E-8
NBANDS = 16
PRECFOCK = Normal

#WAVEDER file must be made:
LOPTICS = .TRUE.
LPEAD = .TRUE.
OMEGAMAX = 40
```

Both Steps use the same KPOINTS, POSCAR and POTCAR files

N.B.: have a look at doall-model.sh

- Step 2: model-BSE calculation
  INCAR (copy INCAR.mBSE to INCAR):

```
PREC = Normal ; ENCUT = 250.0

ALGO = TDHF
ANTIRES = 0
ISMEAR = 0 ; SIGMA = 0.01
ENCUTGW = 150

EDIFF = 1.E-8
NBANDS = 16
NBANDSO = 4
NBANDSV = 8
OMEGAMAX = 20

PRECFOCK = Normal

LMODELHF = .TRUE.
HFSCREEN = 1.26
AEXX = 0.088
SCISSOR = 0.69
```

BSE: increase k-point sampling (ex.: Si_improve_eps)

Method II: model-BSE