

2024 School on Electron-Phonon Physics, Many-Body Perturbation Theory, and Computational Workflows

Phonon-assisted optical absorption with EPW

Hands-on session (Thur.4)

Hands-on based on EPW-v5.9

Exercise

Before you start, please copy the tutorial tarball to your own scratch directory:

```
$ cp /work2/05193/sabyadk/stampede3/EPWSchool2024/tutorials/Thur.4.Zhang.tar $SCRATCH
```

In this example we are going to calculate the phonon-assisted optical absorption spectrum of silicon. Extract the tarball `Thur.4.Zhang.tar`:

```
$ tar -xvf Thur.4.Zhang.tar
$ cd Thur.4.Zhang
```

Under `Si` and `Ag` folder, you can find all inputs we will use in this tutorial as well as some output files for verification purposes in the `Si/results` folder. For a complete list of inputs please follow the link:

<https://epwdoc.gitlab.io/source/doc/Inputs.html>

You are advised to prepare the following queue submission script:

```
#!/bin/bash
#SBATCH -J myjob           # Job name
#SBATCH -o myjob.o\%j      # Name of stdout output file
#SBATCH -e myjob.e\%j     # Name of stderr error file
#SBATCH -N 1              # Total number of nodes
#SBATCH -t 00:10:00      # Run time (hh:mm:ss)
#SBATCH -A DMR23030
#SBATCH -p skx # partition
#SBATCH --reservation=NSF_Summer_School_Thu # reservation (must use current day)
QE=/work2/05193/sabyadk/stampede3/EPWSchool2024/q-e/bin
cd $PWD

#Copy the script to each working directory and put the commend for running the code below:
```

This script is also present inside the tarball and is named `submission`. You can copy the script to the working directory, for example:

```
$ cp submission Si/ph/
```

And modify the script to add the commend for running the calculations, then go to the working directory and submit the script by:

```
$ sbatch submission
```

Note: The path to the executables are set in the submission script. If you are using your own computer, remember to change the line to:

```
$ export QE='PATH-T0-QE/q-e/bin'
```

► Make a self-consistent calculation for Si.

The major part of this tutorial will be performed in the Si (Silicon) folder, with some exploratory calculations in the end that you can perform with Ag (Silver). First to into the Si directory, you will find two sub-directories. We will obtain phonon-related properties on the coarse grid using PH in Quantum Espresso in folder ph, and run EPW calculations in folder EPW.

To start with, we perform a self-consistent calculation in ph:

```
&control                                                                    scf.in
  calculation      = 'scf'
  prefix           = 'si'
  restart_mode     = 'from_scratch'
  pseudo_dir       = './'
  outdir           = './'
/
&system
  ibrav            = 2
  celldm(1)       = 10.262
  nat              = 2
  ntyp             = 1
  ecutwfc          = 20
/
&electrons
  diagonalization = 'david'
  mixing_beta     = 0.7
  conv_thr        = 1.0d-13
/
ATOMIC_SPECIES
Si 28.0855 Si_r.upf
ATOMIC_POSITIONS alat
Si 0.00 0.00 0.00
Si 0.25 0.25 0.25
K_POINTS automatic
12 12 12 0 0 0
```

Note: The k-point grid needs to be fairly dense in order to get converged dielectric function and Born effective charge during the following phonon calculation.

```
$ cd Si/ph
```

```
$ ibrun -np 8 $QE/pw.x -npool 8 < scf.in > scf.out
```

► Compute the vibrational properties of Si on a coarse 3x3x3 q-point grid.

```
$ ibrun -np 112 $QE/ph.x -npool 28 < ph.in > ph.out
```

During the run, notice the IBZ q-point grid. You can look at the output file or simply use the following command:

```
grep -6 Dynamical ph.out | tail -7
```

```
Dynamical matrices for ( 3, 3, 3) uniform grid of q-points
( 4 q-points):
  N      xq(1)      xq(2)      xq(3)
  1  0.000000000  0.000000000  0.000000000
  2 -0.333333333  0.333333333 -0.333333333
  3  0.000000000  0.666666667  0.000000000
  4  0.666666667 -0.000000000  0.666666667
```

We also obtain the dielectric function and Born effective charge tensor. You can look at the output file or simply use the following command

```
grep -15 'Dielectric constant in' ph.out | tail -17
```

```
Dielectric constant in cartesian axis

(      12.982914367      0.000000000      0.000000000 )
(      0.000000000     12.982914367     -0.000000000 )
(     -0.000000000     -0.000000000     12.982914367 )

Effective charges (d Force / dE) in cartesian axis

  atom      1      Si
Ex (      -0.00778      0.00000      0.00000 )
Ey (      0.00000     -0.00778     -0.00000 )
Ez (      0.00000     -0.00000     -0.00778 )
  atom      2      Si
Ex (      -0.00778     -0.00000     -0.00000 )
Ey (      -0.00000     -0.00778     -0.00000 )
Ez (      0.00000     -0.00000     -0.00778 )
```

The experimental dielectric constant of silicon is about 11.9 so that we overestimate it a bit. A smaller value can be obtained with larger k-point grids but DFT will always overestimate experiment. Since Si is a non polar material, the Born effective charges should be 0. Indeed, one can see that we are close to zero. In a converged calculation you may want to get an even smaller value.

► Like for the previous exercises, run the python post-processing to create the save folder (**No job submission needed, can be ran simply on the login node.**)

```
$ python3 /work2/05193/sabyadk/stampede3/EPWSchool2024/q-e/EPW/bin/pp.py
```

A save folder will be generated which contains the potential variation. Copy this folder to the EPW folder, where subsequent calculations will be performed.

```
$ cp -r save ../EPW
```

Switch to the EPW folder, and run another self-consistent calculation.

```
$ cd ../EPW
$ ibrun -np 8 $QE/pw.x -npool 8 < scf.in > scf.out
```

► Do a non self-consistent calculation on a homogeneous 6x6x6 **positively defined between 0 and 1** k-point grid.

```
&control                                                                    nscf.in
  calculation      = 'nscf'
  prefix           = 'si'
  restart_mode     = 'from_scratch'
  pseudo_dir       = './'
  outdir           = './'
/
&system
 ibrav              = 2
cellldm(1)         = 10.262
nat                = 2
ntyp               = 1
```

```

    ecutwfc      = 20
    nbnd        = 12
  /
  &electrons
    diagonalization = 'david'
    mixing_beta    = 0.7
    conv_thr       = 1.0d-10
  /
  ATOMIC_SPECIES
    Si 28.0855 Si_r.upf
  ATOMIC_POSITIONS alat
    Si 0.00 0.00 0.00
    Si 0.25 0.25 0.25
  K_POINTS crystal
  216
    0.00000000 0.00000000 0.00000000 4.629630e-03
    0.00000000 0.00000000 0.16666667 4.629630e-03
  ...

```

```
$ ibrun -np 8 $QE/pw.x -npool 8 < nscf.in > nscf.out
```

► Perform an EPW calculation to interpolate the electron-phonon matrix element from a coarse $6 \times 6 \times 6$ k and $3 \times 3 \times 3$ q -point grids to real space and then interpolate the electronic and phononic bandstructure along the $L - \Gamma - X$ high symmetry line.

```

--
&inputepw
  prefix      = 'si'
  outdir      = './'

  elph        = .true.
  epbwrite    = .true.
  epbread     = .false.
  epwwrite    = .true.
  epwread     = .false.
  etf_mem     = 1

  nbndsub     = 8
  eig_read.   = .true.

  asr_typ     = 'crystal'
  use_ws.     = .true.

  wannierize  = .true.
  num_iter    = 1500
  iprint      = 2
  dis_win_max = 18
  dis_froz_max = 9
  proj(1)     = 'Si : sp3'
  wdata(1)    = 'bands_plot = .true.'
  wdata(2)    = 'begin kpoint_path'
  wdata(3)    = 'L 0.50 0.00 0.00 G 0.00 0.00 0.00'
  wdata(4)    = 'G 0.00 0.00 0.00 X 0.50 0.50 0.00'
  wdata(5)    = 'end kpoint_path'
  wdata(6)    = 'bands_plot_format = gnuplot'
  wdata(7)    = 'guiding_centres = .true.'
  wdata(8)    = 'dis_num_iter      = 500'
  wdata(9)    = 'num_print_cycles  = 10'
  wdata(10)   = 'dis_mix_ratio     = 1.0'
  wdata(11)   = 'use_ws_distance = T'

  elecselven = .false.
  phonselven = .false.
  a2f        = .false.

  fsthick    = 1.2 ! eV
  temps      = 1 ! K
  degaussw   = 0.005 ! eV

```

```

dvscf_dir = './save'

band_plot = .true.
filkf     = './LGX.txt'
filqf     = './LGX.txt'

nk1       = 6
nk2       = 6
nk3       = 6
nq1       = 3
nq2       = 3
nq3       = 3
/

```

```
$ ibrun -np 112 $QE/epw.x -npool 112 < epw1.in > epw1.out
```

The calculation should take about 1 min. In the output, notice the additional information related to the crystal ASR. At the end of the calculation, because of the keyword `band_plot = .true.`, the code will produce `band.eig` and `phband.freq` files that contains the electronic and phononic bandstructure along a path given by `filkf` and `filqf`. Note that the tag `eig_read = .true.` asks the code to read in an external set of eigenvalues (see `si.eig`) that needs to be on the same set of \mathbf{k} -points grid and number of bands as the `nscf` calculation. Here a set of GW eigenvalues calculated with BerkeleyGW are provided.

Note: An option exists to use QE generated interatomic forces in EPW with setting `lifc = .true.`. This is not recommended, and instead, the combination of `lifc = .false.` (default) and `use_ws = .true.` to ensure construction of the correct Wigner-Seitz cell is always recommended for consistency.

If you want to convert the `band.eig` and `phband.freq` files into a format readable by gnuplot, you can use the `plotband.x` tool by doing: **(No job submission needed, can be ran simply on the login node.)**

```
$ /work2/05193/sabyadk/stampede3/EPWSchool2024/q-e/bin/plotband.x
```

to launch a command line interface for the file generation. Note that the second prompt will define the y-axis bounds. You should check that both bandstructure looks reasonable.

► Do a restart calculation (restarting from the `si.epmatwp` file) and compute the phonon-assisted absorption spectrum of silicon in the visible range.

The phonon-assisted optical absorption formalism is presented in J. Noffsinger *et al.*, *Phys. Rev Lett.* **108**, 167402 (2012), *Phys. Rev. B* **107**, 115207, (2023), and in the most recent EPW technical paper: H. Lee, et al., *npj Comput Mater* **9**, 156 (2023). The code calculates the imaginary part of the dielectric function due to phonon-assisted optical transitions in the spectral region between the indirect and direct band gaps:

$$\begin{aligned}
\text{Im}[\epsilon(\omega)] &= 2 \frac{4\pi e^2}{\Omega} \frac{1}{\omega^2} \sum_{mn\nu, \beta=\pm 1} \int \frac{d\mathbf{k}}{\Omega_{\text{BZ}}} \int \frac{d\mathbf{q}}{\Omega_{\text{BZ}}} \left| \mathbf{e} \cdot [\mathbf{S}_{1,m\nu}(\mathbf{k}, \mathbf{q}) + \mathbf{S}_{2,m\nu\beta}(\mathbf{k}, \mathbf{q})] \right|^2 \\
&\times P_{mn\nu\beta}(\mathbf{k}, \mathbf{q}) \delta(\epsilon_{m\mathbf{k}+\mathbf{q}} - \epsilon_{n\mathbf{k}} - \hbar\omega + \beta\hbar\omega_{\mathbf{q}\nu}),
\end{aligned} \tag{1}$$

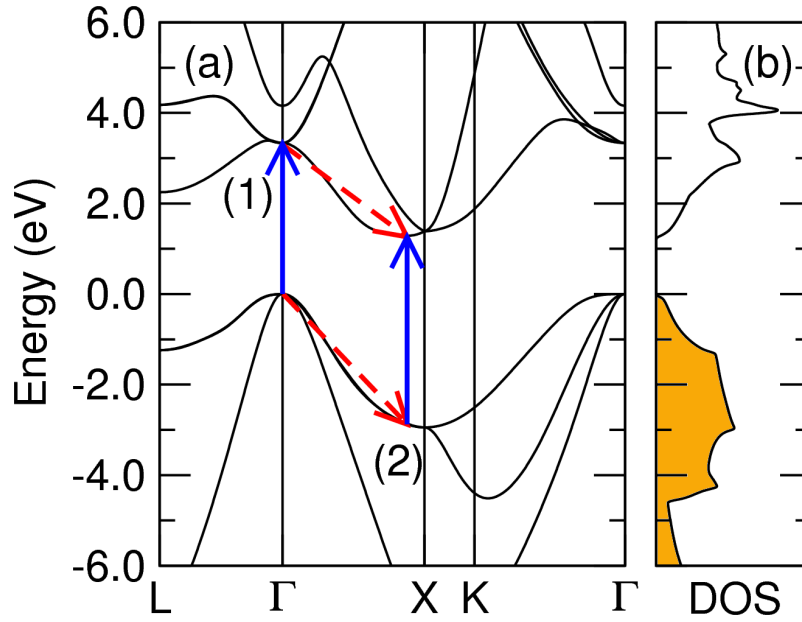
where $\hbar\omega$ and \mathbf{e} are the energy and polarization of the photon. β represent phonon absorption (−)

or emission process (+). The generalized optical matrix elements, S_1 and S_2 , are given by

$$S_{1,m\nu}(\mathbf{k}, \mathbf{q}) = \sum_j \frac{g_{mj\nu}(\mathbf{k}, \mathbf{q}) \mathbf{v}_{jn}(\mathbf{k})}{\epsilon_{j\mathbf{k}} - \epsilon_{n\mathbf{k}} - \hbar\omega + i\eta}, \quad (2)$$

$$S_{2,m\nu\beta}(\mathbf{k}, \mathbf{q}) = \sum_j \frac{\mathbf{v}_{mj}(\mathbf{k} + \mathbf{q}) g_{jn\nu}(\mathbf{k}, \mathbf{q})}{\epsilon_{j\mathbf{k}+\mathbf{q}} - \epsilon_{n\mathbf{k}} + \beta\hbar\omega_{\mathbf{q}\nu} + i\eta}, \quad (3)$$

and correspond to the two possible paths of the indirect absorption process labelled (1) and (2) in the figure below:



They are determined in terms of the velocity (\mathbf{v}) and electron-phonon coupling (g) matrix elements, as well as the quasiparticle energies ($\epsilon_{n\mathbf{k}}$) and an imaginary broadening parameter (η). The factor P accounts for the carrier and phonon statistics,

$$P_{m\nu\beta}(\mathbf{k}, \mathbf{q}) = \left(n_{\mathbf{q}\nu} + \frac{1+\beta}{2} \right) f_{n\mathbf{k}}(1 - f_{m\mathbf{k}+\mathbf{q}}) - \left(n_{\mathbf{q}\nu} + \frac{1-\beta}{2} \right) (1 - f_{n\mathbf{k}})f_{m\mathbf{k}+\mathbf{q}} \quad (4)$$

The P factors can be obtained by considering energy conservation of the absorption/emission process.

The input file is as follows:

```
--
&inputepw                                     epw2.in
  prefix      = 'si'
  outdir      = './'

  elph       = .true.
  epbwrite    = .false.
  epbread     = .false.
  epwwrite    = .false.
  epwread     = .true.
  etf_mem     = 1

! band_plot = .true.

mp_mesh_k = .true.
```

```

lindabs = .true.
efermi_read = .true.
fermi_energy = 5.57
eig_read = .true.

omegamin = 0.05
omegamax = 3.0
omegastep = 0.05

elecselphen = .false.
phonselfen = .false.
a2f = .false.

fsthick = 4.0 ! eV
temps = 300 ! K
degaussw = 0.05 ! eV

dvscf_dir = './save'

nkf1 = 12
nkf2 = 12
nkf3 = 12
nqf1 = 6
nqf2 = 6
nqf3 = 6

nk1 = 6
nk2 = 6
nk3 = 6
nq1 = 3
nq2 = 3
nq3 = 3

nbndsub = 8

use_ws = .true.
asr_typ = 'crystal'

wannierize = .false.
/

```

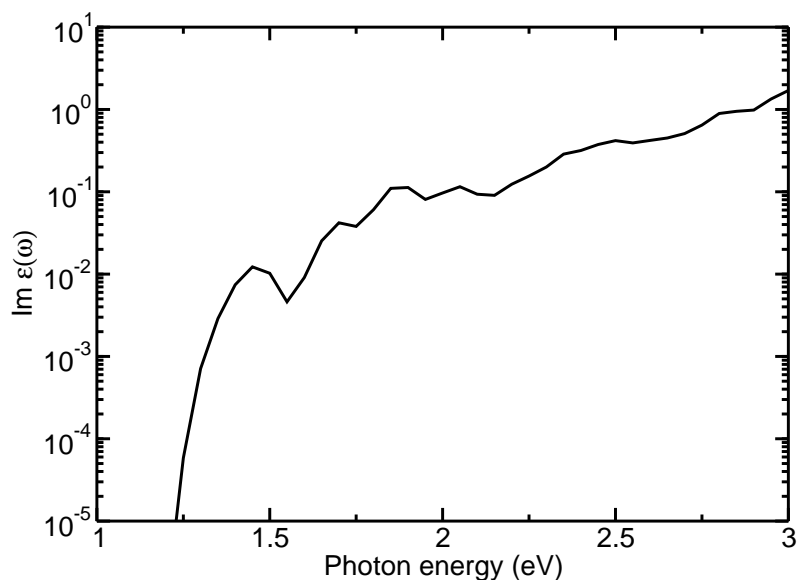
Notes:

- `epwread` allows for the restart from the `si.epmatwp` file.
- `lindabs` enables phonon-assisted optical absorption calculations.
- `mp_mesh_k` use only the `irr. wedge` for the fine electronic mesh. This reduces the computational cost significantly.
- `omegamin`, `omegamax`, `omegastep` are the minimum, maximum, and spacing of the photon energies for which we calculate the absorption coefficient.
- `fermi_energy = 5.57` sets the Fermi energy for silicon (insulator) in the middle of the gap, and `fsthick = 4.0` limits the absorption calculation to states within an energy window around the Fermi level (set to be larger than the maximum photon energy we want the absorption spectra for plus the maximum phonon energy of the material).
- The code calculates the spectra with nine different broadening parameter values (η in the S-parameters, where $\eta = 0.001, 0.002, 0.005, 0.01, 0.02, 0.05, 0.1, 0.2, 0.5$ eV). Note that within the indirect region of the spectra, the choice of the broadening of the denominator does not affect the spectra significantly.

```
$ ibrun -np 112 $QE/epw.x -npool 112 < epw2.in > epw2.out
```

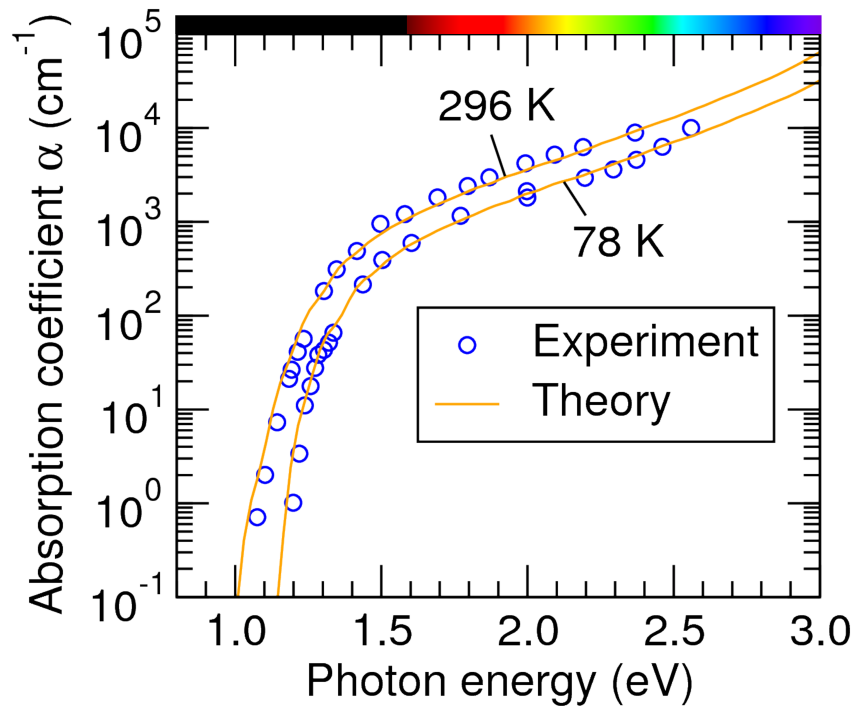
The run should take less than 1 min. You can find the directionally-averaged imaginary part of the phonon-assisted dielectric function for the different broadenings in `epsilon2_indabs_300.OK.dat` (Gaussian broadening for delta function) or `epsilon2_indabs_lorenz300.OK.dat` (Lorentzian broad-

ening for delta function).¹ The first column is photon energy, while the other columns are the corresponding $\text{Im } \epsilon(\omega)$ for different η s. The result should look like this (for the first η with Gaussian broadening for the delta function):



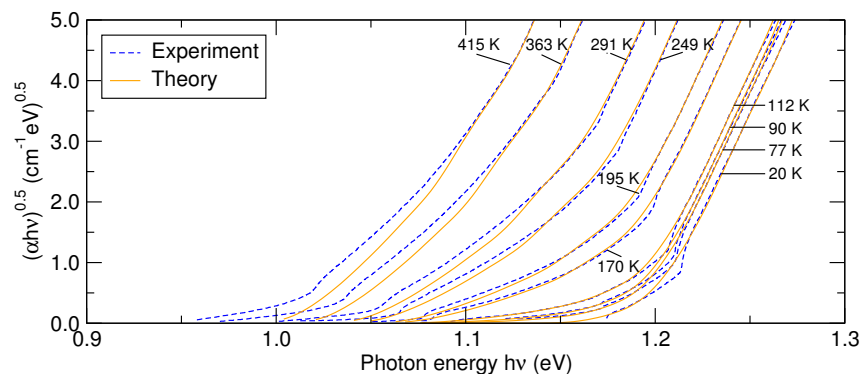
The fine \mathbf{k} and \mathbf{q} point grids need to be denser for production calculations. At convergence you should get a result similar to Fig. 3 of J. Noffsinger *et al.*, *Phys. Rev Lett.* **108**, 167402 (2012). Note that the relationship between $\text{Im } \epsilon(\omega)$ and the absorption coefficient is $\alpha(\omega) = \frac{\omega}{cn_r(\omega)} \text{Im } \epsilon(\omega)$. The paper assumes a constant refractive index of $n_r = 3.4$.

¹Note that direct absorption within the independent particle approximation is also calculated, see `epsilon2_dirabs_300.0K.dat` and `epsilon2_dirabs_lorenz300.0K.dat`. Careful convergence with respect to \mathbf{k} -grid is important.



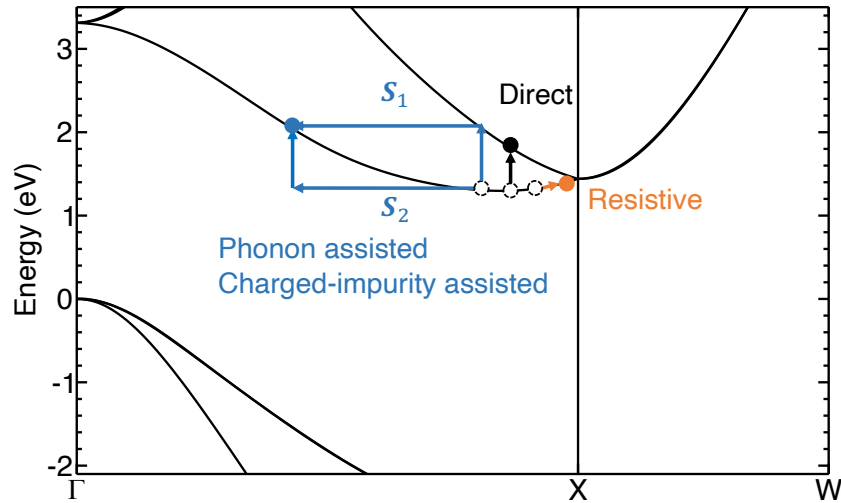
► Increase the fine grids to improve convergence.

You can also perform calculations to study the onset of phonon-assisted absorption and see the separate contributions from the phonon-absorption and the phonon-emission terms. These calculations need a small broadening parameter on the order of 10 meV (smaller than the phonon frequency) to distinguish the different onsets of the phonon-absorption and phonon-emission terms as well as fine \mathbf{k} and \mathbf{q} grids on the order of $40 \times 40 \times 40$. You can accelerate these calculations by selecting a narrow ω_{\min} and ω_{\max} range near the indirect absorption onset and adjusting the f_{sthick} window accordingly. The converged result is similar to the spectra below from J. Noffsinger *et al.*, *Phys. Rev. Lett.* **108**, 167402 (2012):



► Free-carrier absorption (FCA)

In addition to the phonon-assisted optical absorption, you can add the combination of n_{carrier} and p_{carrier} to specify the carrier density in n-type (positive n_{carrier}) or p-type (negative n_{carrier}) semiconductors to study phonon-assisted free-carrier absorption. Free-carrier absorption occurs in doped semiconductors/metals when the conduction band minimum or valence band maximum is partially occupied/empty. Various mechanisms need to be considered, as illustrated by Fig. 1 in *Phys. Rev. B* **106**, 205203:



The input for such a calculation is provided in `epw3.in`. Note that some of the previous output epsilon files in `epw2` will be overridden.

```

--
&inputepw
...

omegamin = 0.05
omegamax = 3.0
omegastep = 0.05
carrier = .true.
ncarrier = 1e19

ii_g = .true.
ii_n = 1e19
ii_lscreen = .true.

sigma_ref = 20000

...

nkf1      = 16
nkf2      = 16
nkf3      = 16
nqf1      = 16
nqf2      = 16
nqf3      = 16

...
/

```

The following tags are relevant for free-carrier absorption:

- `carrier`: enable calculation of free-carrier absorption. The Fermi level will be determined by the carrier density.
- `ncarrier`: free-carrier density in cm^{-3} , positive for free electrons and negative for free holes.
- `ii_g`, `ii_n`, `ii_lscreen`: optical tags, enables calculation of ionized-impurity scattering matrix element (See tutorial Wed.7.) to enable ionized-impurity assisted optics.
- `sigma_ref`: conductivity of the material in $\Omega^{-1}\cdot\text{m}^{-1}$. Used to determine the resistive contribution within a constant relaxation time approximation:

$$\text{Im } \epsilon^{\text{resis}}(\omega) = \frac{4\pi\sigma}{\omega(1 + \omega^2\tau^2)} \quad (5)$$

Note that the \mathbf{k} - and \mathbf{q} -points grid are finer in the input for this optional calculation. Since free-carrier absorption happens close to the conduction band minimum or valence band maximum, a much denser

grid is necessary to obtain converged result. In practice, similar to the case of mobility calculations, one can typically set `fermi_energy` close to the conduction band minimum (for free electrons) or valence band maximum (for free holes), and use a smaller value of `fsthick` to reduce the computational cost due to that free-carrier absorption is dominant only in sub-gap region where photon energy is relatively small compared to cross-gap indirect absorption. Note that in this case, `fermi_energy` is not used to set the Fermi-level used to calculate absorption, but only used to determine the states to consider according to `fsthick`.

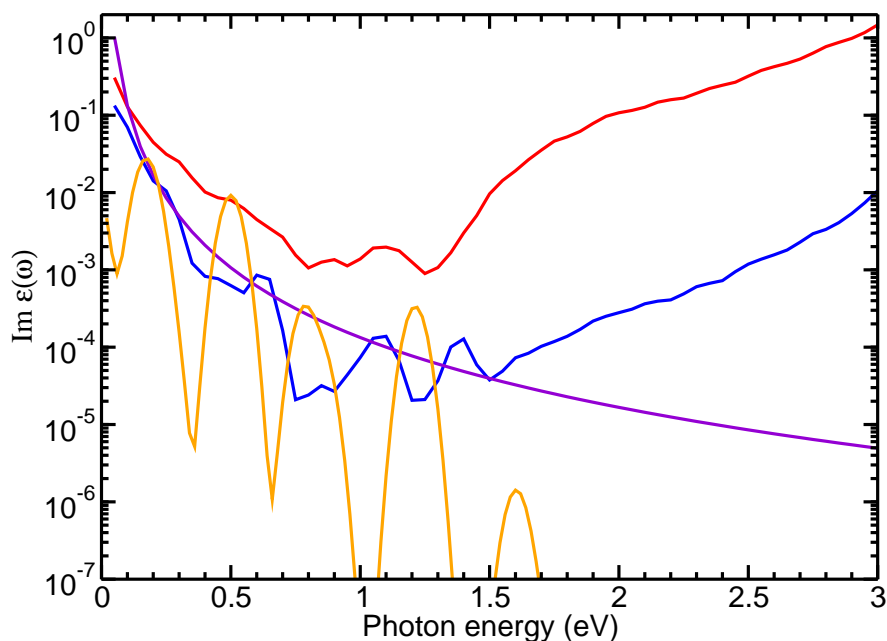
We evaluate the free-carrier absorption in silicon:

```
$ ibrun -np 112 $QE/epw.x -npool 112 < epw3.in > epw3.out
```

Several outputs will be generated when free-carrier absorption is enabled:

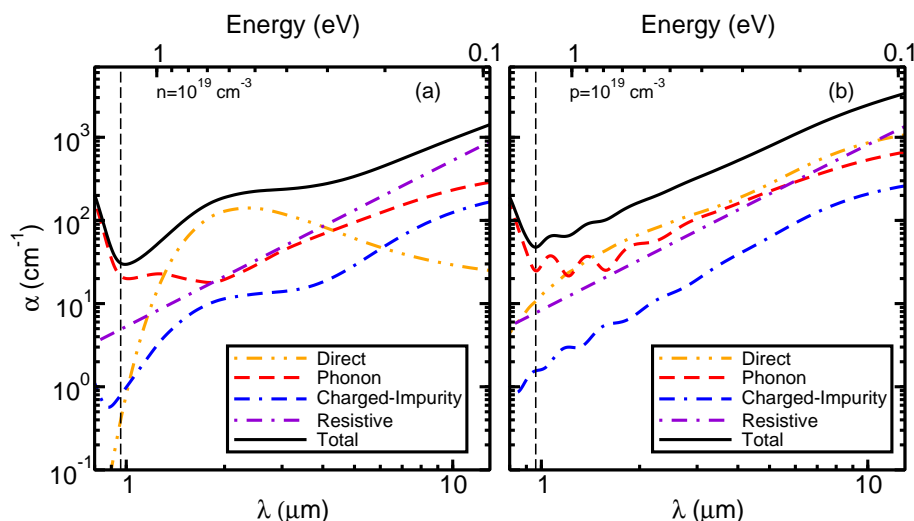
- `epsilon2_indabs_300.0K.dat`: Phonon-assisted contribution.
- `epsilon2_indabs_imp_300.0K.dat`: Ionized-impurity-assisted contribution.
- `epsilon2_dirabs_300.0K.dat`, Direct contribution.
- `epsilon2_indabs_resis_300.0K.dat`: Resistive contribution.

You may notice that the spectrum now has a strong dependence on the broadening η in Eq.(2) and Eq.(3), which is due to the divergence of the denominator. This divergence is present when direct transition (in this case, inter-conduction bands) is possible, and is a limitation of the second-order perturbation theory itself. It can be reduced by numerical approaches to cancel the diverging term, see e.g. Brown *et al.*, *ACS Nano*, **10.1**, 957-966 (2016). Plotting the phonon-assisted contribution (red) and ionized-impurity-assisted contribution (blue) with $\eta = 0.1$ eV (column #8 in the files), direct contribution (orange) averaged over light polarization (last column), and the resistive contribution (purple), they should look like this:



You will notice the $\text{Im } \epsilon(\omega)$ from all the contributions now contain sub-gap contributions that results from free-carrier absorption. In addition, one can see a severe underconvergence for all the contribution due to Brillouin zone sampling. In our previous study, a grid of $48 \times 48 \times 48$ was required for

converged free-carrier absorption spectra of silicon. More details about free-carrier absorption can be found at: *Phys. Rev. B* **106**, 205203. The converged result e.g. using a grid of $48 \times 48 \times 48$, is similar to Fig. 2 in the paper (note that the x-axis in the paper is wavelength, i.e. $\lambda_{nm} \approx 1.24/(\hbar\omega)_{eV}$):



► Metallic materials

In the case of a metallic material, free-carrier absorption becomes a natural and intrinsic contribution that needs to be considered when considering optical response.

As an exploratory task, once you finish the example of silicon, you can take a look at the other directory, which contains the input for evaluating optical absorption in noble metal silver (Ag). The steps of running the calculation is very similar. One key difference you would notice is that instead of specifying a carrier density, now the code automatically finds the Fermi level of the metallic system, triggered by the combination of `carrier` and `assume_metal` with `lindabs`.

Try to run the calculation following the same procedure for free-carrier absorption in silicon. Take a look at the output, and think about the following questions:

- In this case, do we have the same contributions as in the case of doped silicon? If not, what are we missing, and why?
- In what region does each different contribution matter? Is this what you expect and why? (look at the band structure, and the Fermi level determined by the code in the EPW output.)
- In practice, how do you obtain the electrical conductivity that is used as an input to determine the resistive contribution?

► Some tips for plotting

The output of the indirect optics module contains many multi-column data, with the first column being energy, the the other column being $\text{Im } \epsilon(\omega)$ with different parameters. A few tips for visulizing the data using gnuplot:

- If you want to use gnuplot, you need to enable X-forwarding when logging in:

```
$ ssh your_account@stampede3.tacc.utexas.edu -XC
```

- In the directory of your calculation, you can open gnuplot and plot a certain column of the data. For example, to plot the phonon-assisted absorption, i.e. output from epw2.in:

```
$ gnuplot
gnuplot> set logscale y
gnuplot> set yrange [1e-5:10]
gnuplot> plot "epsilon2_indabs_300.0K.dat" using 1:8 with lines
```

These commands set the y-axis to be log scale, and set the range to from 10^{-5} to 10, and plot the 8th column ($\eta = 0.1$ eV) against the 1st column (Photon energy) with lines connecting the scattered data points.

- Similarly, you can plot the phonon-assisted absorption in the case of free-carrier absorption (epw3.in) with plot, change the file name and use replot and plot other contributions on top of the current plot:

```
gnuplot> set logscale y
gnuplot> set yrange [1e-7:1]
gnuplot> plot "epsilon2_indabs_300.0K.dat" using 1:8 with lines
gnuplot> replot "epsilon2_dirabs_300.0K.dat" using 1:5 with lines
gnuplot> replot "epsilon2_indabs_imp_300.0K.dat" using 1:8 with lines
gnuplot> replot "epsilon2_indabs_resis_300.0K.dat" using 1:2 with lines
```

- The above comment plots the average of the direct absorption (5th column), the ionized-impurity-assisted absorption with $\eta = 0.1$ eV (8th column) and the resistive contribution (2nd column). After you are done, you can exit gnuplot by typing:

```
gnuplot> exit
```

Various other packages can be used to plot the data such as the matplotlib library in python, xmgrace. You can choose whichever you are more familiar with to visulize the data.