

---

MAX School on Advanced Materials and Molecular Modelling  
with **QUANTUM ESPRESSO**

---

# **QE-2021: Hands-on session – Day-6**

## **TDDFPT**

**Iurii Timrov (EPFL, Switzerland)**

**Oscar Baseggio (SISSA, Italy)**

# Outline

1. **Part I** – Optical absorption in molecules
2. **Part II** – EELS in solids

# Part I – Optical absorption in molecules

1. **Exercise 1:** **turboDavidson** – from the Independent Particle Approximation (IPA) to interacting electrons
2. **Exercise 2:** **turboLanczos** – total spectra at low cost for interacting electrons

# turboDavidson program for calculations of absorption spectra

- The **turbo\_davidson.x** program allows us to calculate absorption spectra of molecules using time-dependent density functional perturbation theory (TDDFpT).
- The interactions of electrons (**Hartree** and **Exchange-Correlation effects**) are taken into account fully *ab initio* and self-consistently (or they can be neglected **if\_dft\_spectrum=.false.**).
- The electronic transitions from occupied to empty states can be analyzed by selecting a frequency range in which the transitions occur.
- However, calculations of overall absorption spectra in a wide frequency range are computationally rather expensive, because many eigenvalues of the Hamiltonian must be computed.

# turboDavidson program for calculations of absorption spectra

Coupled linear-response TDDFpT equations (**resonant** and **anti-resonant**):

$$\begin{aligned} (\hat{H}^o - \varepsilon_v - \hbar\omega) \tilde{\varphi}'_v(\mathbf{r}, \omega) + \hat{P}_c \hat{V}'_{\text{HXC}}[\tilde{\varphi}'_v(\mathbf{r}, \omega), \tilde{\varphi}'_v^*(\mathbf{r}, -\omega)] \varphi_v^o(\mathbf{r}) &= 0 \\ (\hat{H}^o - \varepsilon_v + \hbar\omega) \tilde{\varphi}'_v^*(\mathbf{r}, -\omega) + \hat{P}_c \hat{V}'_{\text{HXC}}[\tilde{\varphi}'_v(\mathbf{r}, \omega), \tilde{\varphi}'_v^*(\mathbf{r}, -\omega)] \varphi_v^o(\mathbf{r}) &= 0 \end{aligned}$$

These equations can be re-written as Casida's matrix equation:

$$\begin{pmatrix} 0 & \hat{D} \\ \hat{D} + \hat{K} & 0 \end{pmatrix} \begin{pmatrix} Q \\ P \end{pmatrix} = \omega \begin{pmatrix} Q \\ P \end{pmatrix}$$

**interaction terms**

where  $\hat{D} \longrightarrow (\hat{H}^o - \varepsilon_v)$  and  $\hat{K} \longrightarrow \hat{P}_c \hat{V}'_{\text{HXC}} \varphi_v^o(\mathbf{r})$

**Davidson algorithm** is used (the same algorithm as in the ground state SCF calculation) to solve the Casida's equations and to obtain the eigenvalues which are used for a calculation of the absorption coefficient.

# Independent Particle Approximation

**The simplest approximation:** Independent Particle Approximation (IPA) - a sum of independent excitations from occupied to empty states.

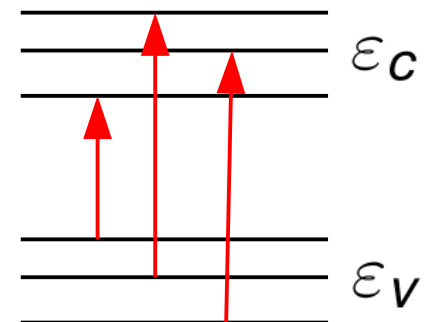
## Fermi's golden rule

The transition probability per unit time from occupied states  $|\varphi_v\rangle$  to empty states  $|\varphi_c\rangle$  reads:

$$T(\omega) = \frac{2\pi}{\hbar} \sum_{v,c} \delta(\varepsilon_c - \varepsilon_v - \hbar\omega) \left| \langle \varphi_c | V'_{ext} | \varphi_v \rangle \right|^2$$

$V'_{ext} = -e\mathbf{E} \cdot \mathbf{r}$  is the external potential induced by the electric field.

$\varepsilon_v, \varepsilon_c$  and  $|\varphi_v\rangle, |\varphi_c\rangle$  are the eigenvalues and the eigenfunctions of the ground-state Kohn-Sham equation

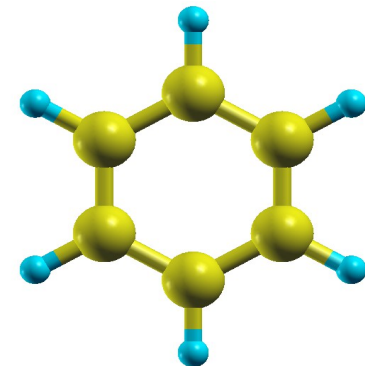
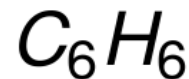


# Exercise 1: Input file for PWscf

**Step 1.** Perform a Self-Consistent Field ground-state calculation for benzene at the equilibrium structure using the **pw.x** program.

Specify the total number of Kohn-Sham bands (occupied + some empty states)

```
&CONTROL
  calculation = 'scf'
  restart_mode='from_scratch',
  prefix='Benzene'
  pseudo_dir = '../..pseudo'
  outdir='./tmp',
/
&SYSTEM
 ibrav = 6,
  celldm(1) = 32,
  celldm(3) = 0.83,
  nat = 12,
  ntyp = 2,
  ecutwfc = 30,
  nbnd = 20
/
&ELECTRONS
  conv_thr = 1.0d-8
  mixing_beta = 0.6,
/
ATOMIC_SPECIES
H 1.0 H.blyp-vbc.UPF
C 12.01 C.blyp-mt.UPF
ATOMIC_POSITIONS {Angstrom}
C 5.633200899 6.320861303 5.000000000
C 6.847051545 8.422621957 5.000000000
C 8.060751351 7.721904557 5.000000000
C 8.060707879 6.320636665 5.000000000
C 6.846898786 5.620067381 5.000000000
C 5.633279551 7.722134449 5.000000000
H 6.847254360 9.512254789 5.000000000
H 9.004364510 8.266639340 5.000000000
H 9.004297495 5.775895755 5.000000000
H 6.846845929 4.530522778 5.000000000
H 4.689556006 5.776237709 5.000000000
H 4.689791688 8.267023318 5.000000000
K_POINTS {gamma}
```



Input

Output

**pw.x < pw.benzene.scf.in > pw.benzene.scf.out**

# Exercise 1: Output file of PWscf

The output SCF file **pw.benzene.scf.out** :

```
End of self-consistent calculation
```

```
k = 0.0000 0.0000 0.0000 ( 37796 PWs) bands (ev):
```

```
-21.1410 -18.3897 -18.3884 -14.6742 -14.6729 -12.5121 -11.0334 -10.4379  
-9.8983 -9.8963 -8.5697 -7.8759 -7.8751 -5.8257 -5.8251 -0.7364  
-0.7362 -0.3332 0.2501 0.3207
```

```
highest occupied, lowest unoccupied level (ev): -5.8251 -0.7364
```

**HOMO** **LUMO**

**HOMO** – Highest Occupied Molecular Orbital

**LUMO** – Lowest Unoccupied Molecular Orbital

Energy gap = **LUMO** – **HOMO** = 5.1 (eV)



# Exercise 1: Calculation of absorption spectra using turboDavidson (IPA)

Perform the turboDavidson calculation without the electronic interaction using the **turbo\_davidson.x** program.

```
&lr_input
  prefix = 'Benzene',
  outdir = './tmp'
/
&lr_dav
  if_dft_spectrum = .true.
  num_init       = 30
/
```

Same as in PWscf

This means we are using IPA

Obsolete variable for the IPA case  
(will not be needed in future version of  
QE for the IPA case)

```
turbo_davidson.x < turbo_davidson.benzene.davidson.in >...
...turbo_davidson.benzene.davidson.out
```

# Exercise 1: Post-processing step

Perform a spectrum calculation using the **turbo\_spectrum.x** program and using the eigenvalues computed in the previous step.

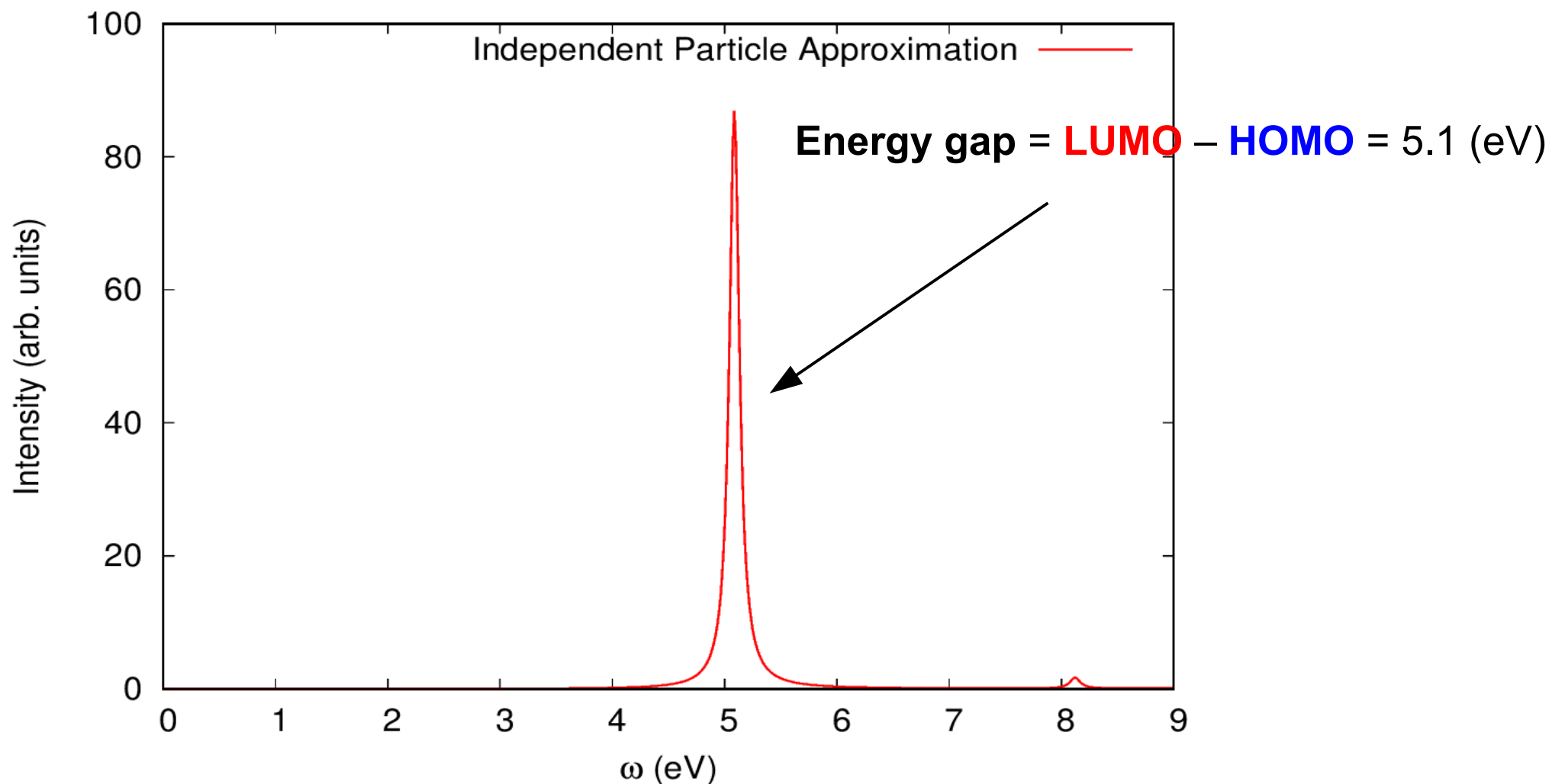
**turbo\_spectrum.Benzene.tddfpt\_pp.in :**

<pre>&amp;lr_input</pre>	
<pre>  prefix = 'Benzene',</pre>	← The same prefix as in the SCF calculation
<pre>  outdir = './tmp',</pre>	← Directory for temporary files
<pre>  td = 'davidson',</pre>	← Type of previous calculation
<pre>  epsil = 0.004,</pre>	← The value of Lorentzian smearing in Ry
<pre>  start = 0.0d0,</pre>	← Minimum value of frequencies for a plot in Ry
<pre>  end   = 1.0d0,</pre>	← Maximum value of frequencies for a plot in Ry
<pre>  increment = 0.0001d0,</pre>	← Frequency step in Ry
<pre>  eign_file = 'Benzene-dft.eigen'</pre>	← Frequency with Davidson eigenvalues
<pre>/</pre>	

**turbo\_spectrum.x < turbo\_spectrum.benzene.davidson.in >...**  
**...turbo\_spectrum.benzene.davidson.out**

# Exercise 1: Calculation of absorption spectra in IPA

Absorption spectrum of benzene in the Independent Particle Approximation  
(file **Benzene\_spectrum.eps**):



# Exercise 1: Spectra including interactions

Perform the turboDavidson calculation without the electronic interaction using the **turbo\_davidson.x** program.

```
&lr_input
  prefix = 'Benzene',
  outdir='./tmp'
/
&lr_dav
  if_dft_spectrum = .false.
  num_eign = 15
  num_init = 30
  num_basis_max = 90
  residue_conv_thr = 1.0E-4
  start = 0.0
  finish = 1.0
  step = 0.001
  broadening = 0.005
  reference = 0.3
/
```

← The same prefix as in the SCF calculation

← Directory for temporary files

← **Switch on the interactions**

← Number of eigenvalues to be calculated

← Number of initial vectors

← Maximum number of basis allowed for the sub-basis

← Convergence threshold

← Minimum value of frequencies for a plot in Ry

← Maximum value of frequencies for a plot in Ry

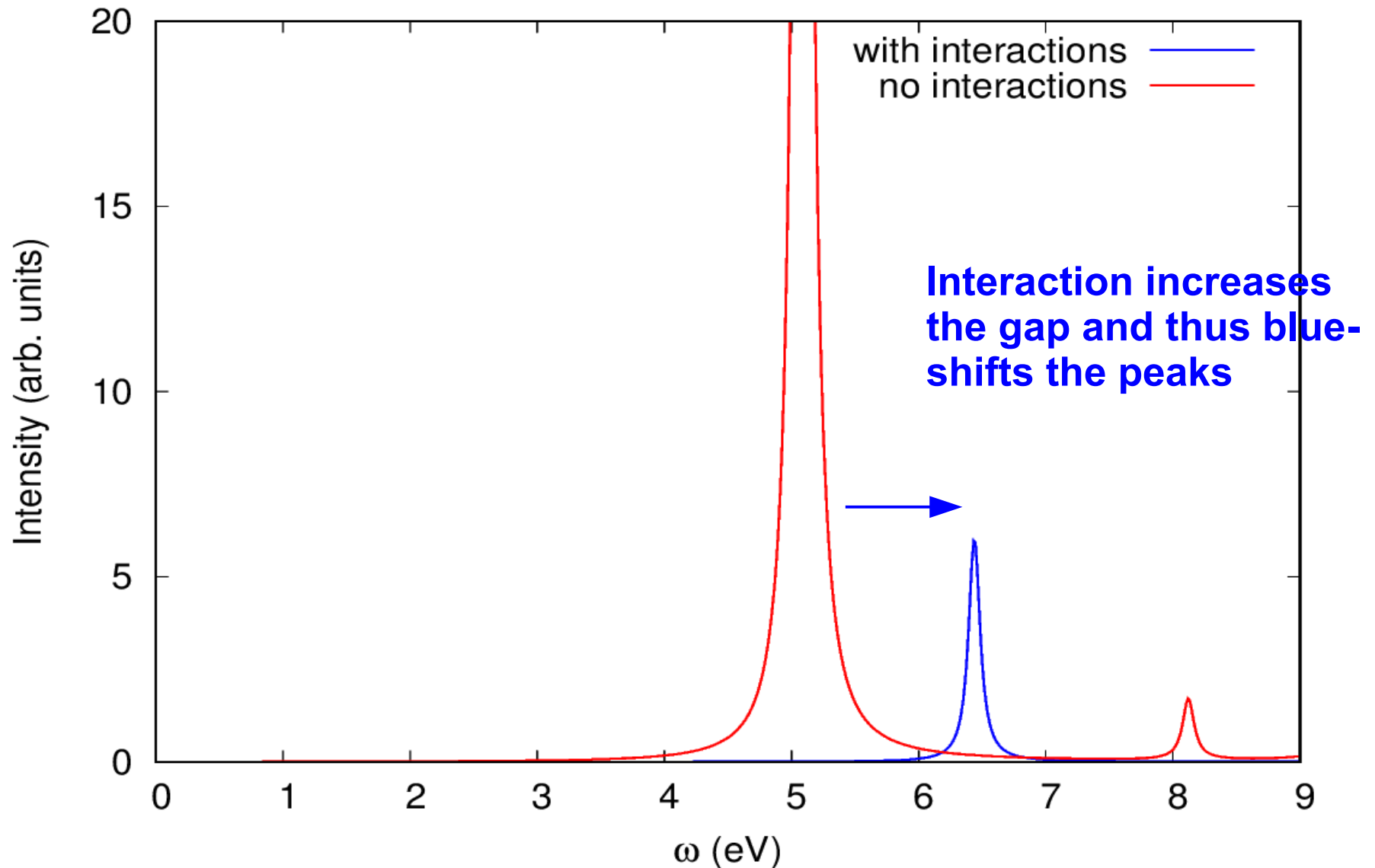
← Frequency step in Ry

← Lorentzian broadening parameter in Ry

← Reference frequency in Ry where the peak is expected

**turbo\_davidson.x < turbo\_davidson.benzene.davidson.in >...**  
**...turbo\_davidson.benzene.davidson.out**

# Exercise 1: Comparison of spectra with and without interactions



# turboLanczos program for calculation of absorption spectra

- The **turbo\_lanczos.x** program allows us to calculate absorption spectra of molecules using time-dependent density functional perturbation theory (TDDFpT) **without computing empty states!**
- The interactions of electrons (**Hartree** and **Exchange-Correlation effects**) are taken into account fully *ab initio* and self-consistently.
- The electronic transitions from occupied to empty states cannot be analyzed (use **turbo\_davidson.x** for this purpose).
- The overall absorption spectrum in a wide frequency range **can be calculated at once!**

# turboLanczos program for calculation of absorption spectra

Coupled linear-response TDDFpT equations (**resonant** and **anti-resonant**):

$$\begin{aligned}
 (\hat{H}^o - \varepsilon_v - \hbar\omega) \tilde{\varphi}'_v(\mathbf{r}, \omega) + \hat{P}_c \hat{V}'_{\text{HXC}}[\tilde{\varphi}'_v(\mathbf{r}, \omega), \tilde{\varphi}'_v^*(\mathbf{r}, -\omega)] \varphi_v^o(\mathbf{r}) &= -\hat{P}_c \hat{V}'_{\text{ext}} \varphi_v^o(\mathbf{r}) \\
 (\hat{H}^o - \varepsilon_v + \hbar\omega) \tilde{\varphi}'_v^*(\mathbf{r}, -\omega) + \hat{P}_c \hat{V}'_{\text{HXC}}[\tilde{\varphi}'_v(\mathbf{r}, \omega), \tilde{\varphi}'_v^*(\mathbf{r}, -\omega)] \varphi_v^o(\mathbf{r}) &= -\hat{P}_c \hat{V}'_{\text{ext}} \varphi_v^o(\mathbf{r})
 \end{aligned}$$

**perturbation**

These equations can be re-written as a quantum Liouville equation:

$$\begin{aligned}
 (\omega - \mathcal{L}) \cdot \rho'(\omega) &= [\hat{V}'_{\text{ext}}, \rho^o] \\
 \mathcal{L} \cdot \rho' &\equiv [\hat{H}^o, \rho'] + [\hat{V}'_{\text{HXC}}, \rho^o]
 \end{aligned}$$

**interaction terms**

**Absorption coefficient** is computed as:

$$\alpha(\omega) = -(\mathbf{r}, (\omega - \mathcal{L})^{-1} [\mathbf{r}, \rho^o])$$

**Lanczos algorithm** is used to solve recursively quantum Liouville equation in the standard batch representation. This allows us to avoid inversions and multiplications of large matrices.

## Exercise 2: Calculation of absorption spectra using turboLanczos

**Step 1.** Perform a SCF ground-state calculation (no **nbnd** in the input):

```
pw.x < pw.benzene.scf.in > pw.benzene.scf.out
```

**Step 2.** Perform Lanczos recursions using the **turbo\_lanczos.x** program.

```
&lr_input  
  prefix = 'Benzene',  
  outdir='./tmp',  
  restart_step = 100,  
  restart = .false.  
/  
&lr_control  
  itermax = 500,  
  ipol = 1  
/
```

← The same prefix as in the SCF calculation

← Directory for temporary files

← The code writes restart files every restart\_step iterations

← Restart iterations after previous calculation

← Number of Lanczos iterations

← Polarization direction of incoming light, 1=x

```
turbo_lanczos.x < turbo_lanczos.benzene.lanczos.in >...  
...turbo_lanczos.benzene.lanczos.out
```



## Exercise 2: Post-processing step

Perform a spectrum calculation using the post-processing program **turbo\_spectrum.x** and using the Lanczos coefficients computed in the previous step.

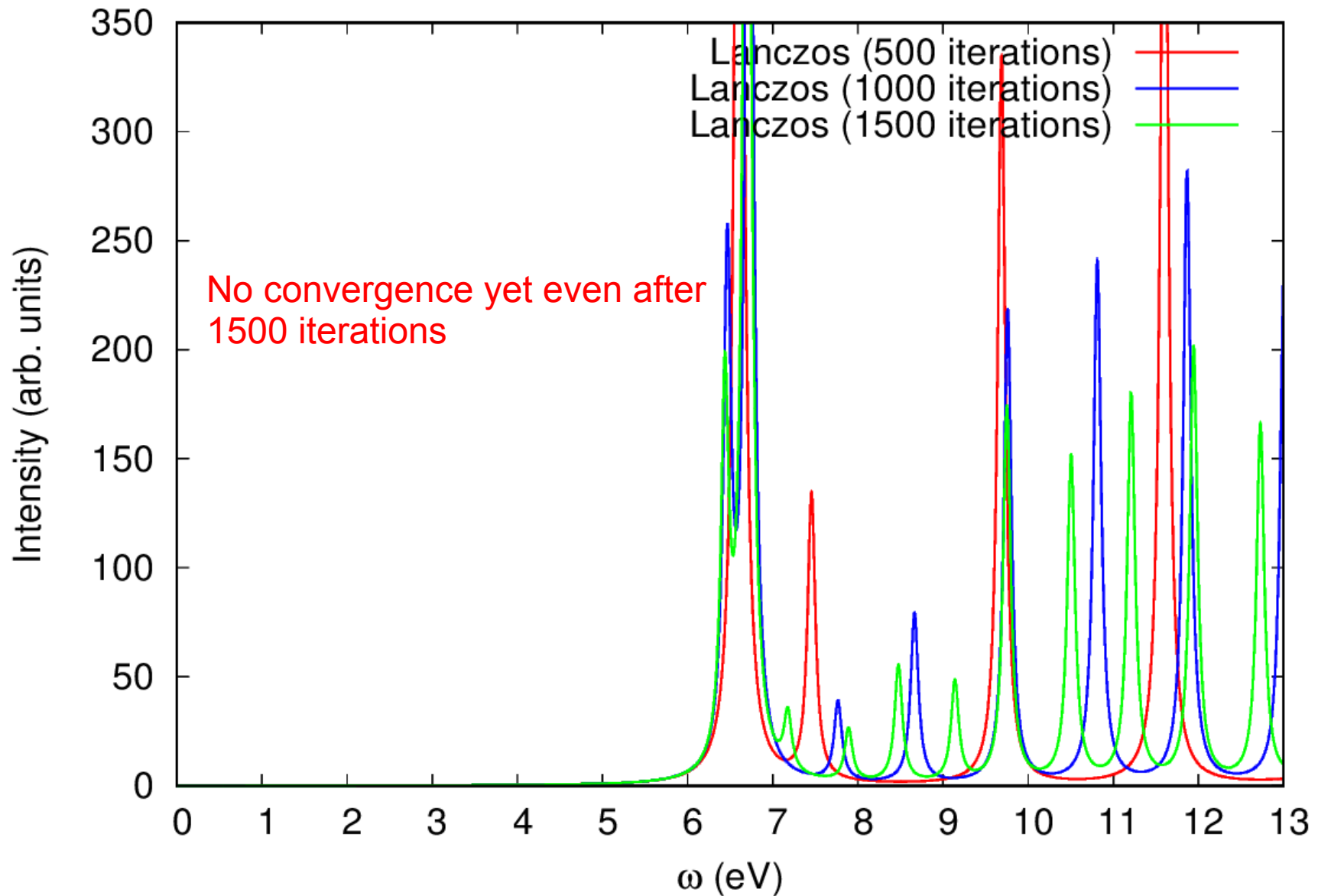
```
&lr_input
  prefix = 'Benzene',
  outdir = './tmp',
  itermax0 = 500,
  itermax = 500,
  extrapolation = 'no',
  epsil = 0.004,
  start = 0.0d0,
  end = 1.0d0,
  increment = 0.0001d0,
  ipol = 1
/
```

- ← The same prefix as in the SCF calculation
- ← Directory for temporary files
- ← Number of calculated Lanczos coefficient
- ← Number of extrapolated Lanczos coefficient
- ← Type of extrapolation
- ← The value of Lorentzian smearing in Ry
- ← Minimum value of frequencies for a plot in Ry
- ← Maximum value of frequencies for a plot in Ry
- ← Frequency step in Ry
- ← Polarization direction (same as in turbo\_lanczos.x)

```
turbo_spectrum.x < turbo_spectrum.benzene.in >...
...turbo_spectrum.benzene.out
```

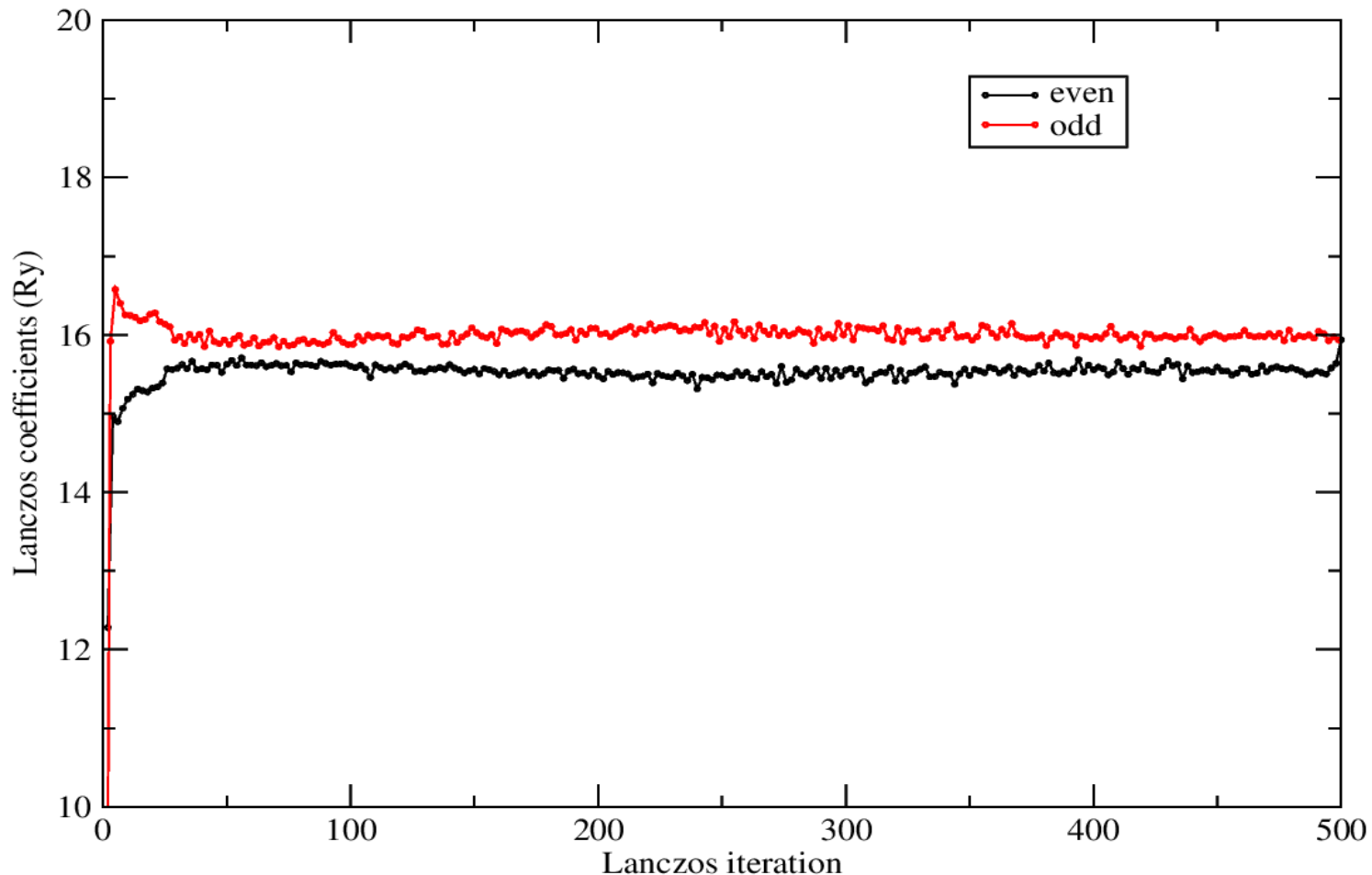
## Exercise 2: Post-processing step

### No extrapolation of Lanczos coefficients



## Exercise 2: Lanczos coefficients

Behavior of  $\beta$  Lanczos coefficients



Lanczos coefficients oscillate around half of the kinetic-energy cutoff ( $\sim 30/2 = 15$  Ry in this case). Hence, we can extrapolate these coefficients.

## Exercise 2: Extrapolation of Lanczos coefficients

In the Liouville-Lanczos approach, the absorption coefficient is computed as:

$$\alpha(\omega) = (\zeta_N, (\omega - \hat{T}_N)^{-1} \cdot \mathbf{e}_1^N)$$

where  $\hat{T}_N$  is the tridiagonal matrix composed of the Lanczos coefficients.

$$\hat{T}_N = \begin{pmatrix} 0 & \gamma_2 & 0 & \cdots & 0 \\ \beta_2 & 0 & \gamma_3 & 0 & 0 \\ 0 & \beta_3 & 0 & \ddots & 0 \\ \vdots & 0 & \ddots & \ddots & \gamma_N \\ 0 & \cdots & 0 & \beta_N & 0 \end{pmatrix} \xrightarrow{\text{Extrapolation}} \hat{T}_{N'} = \begin{pmatrix} 0 & \gamma_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \beta_2 & 0 & \gamma_3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \beta_3 & 0 & \ddots & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \ddots & \ddots & \gamma_N & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \beta_N & 0 & \langle \gamma \rangle & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \langle \beta \rangle & 0 & \langle \gamma \rangle & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \ddots & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \langle \beta \rangle & 0 & \langle \gamma \rangle \end{pmatrix}$$

$$\langle \beta \rangle = \frac{\beta_1 + \beta_2 + \dots + \beta_N}{N}$$

$$\langle \gamma \rangle = \frac{\gamma_1 + \gamma_2 + \dots + \gamma_N}{N}$$

$$N = \text{itermax0}, \quad N' = \text{itermax}$$

## Exercise 2: Post-processing step

### No extrapolation of Lanczos coefficients

turbo\_lanczos.benzene.lanczos.in

turbo\_spectrum.benzene.lanczos.in

```
&lr_input
  prefix = 'Benzene',
  outdir = './tmp',
  restart_step = 100,
  restart = .false.
/
&lr_control
  itermax = 500,
  ipol = 1
/
```

```
&lr_input
  prefix = 'Benzene',
  outdir = './tmp',
  itermax0 = 500,
  itermax = 500,
  extrapolation = 'no',
  epsil = 0.004,
  start = 0.0d0,
  end = 1.0d0,
  increment = 0.0001d0,
  ipol = 1
/
```

Must be equal !

You can use `restart = .true.` option in order not to start from scratch, when you want to increase the number of Lanczos iterations.

## Exercise 2: Post-processing step

### With extrapolation of Lanczos coefficients

turbo\_lanczos.benzene.lanczos.in

turbo\_spectrum.benzene.lanczos.in

```
&lr_input
  prefix = 'Benzene',
  outdir = './tmp',
  restart_step = 100,
  restart = .false.
/
&lr_control
  itermax = 500,
  ipol = 1
/
```

```
&lr_input
  prefix = 'Benzene',
  outdir = './tmp',
  itermax0 = 500,
  itermax = 20000,
  extrapolation = 'osc',
  epsil = 0.004,
  start = 0.0d0,
  end = 1.0d0,
  increment = 0.0001d0,
  ipol = 
/
```

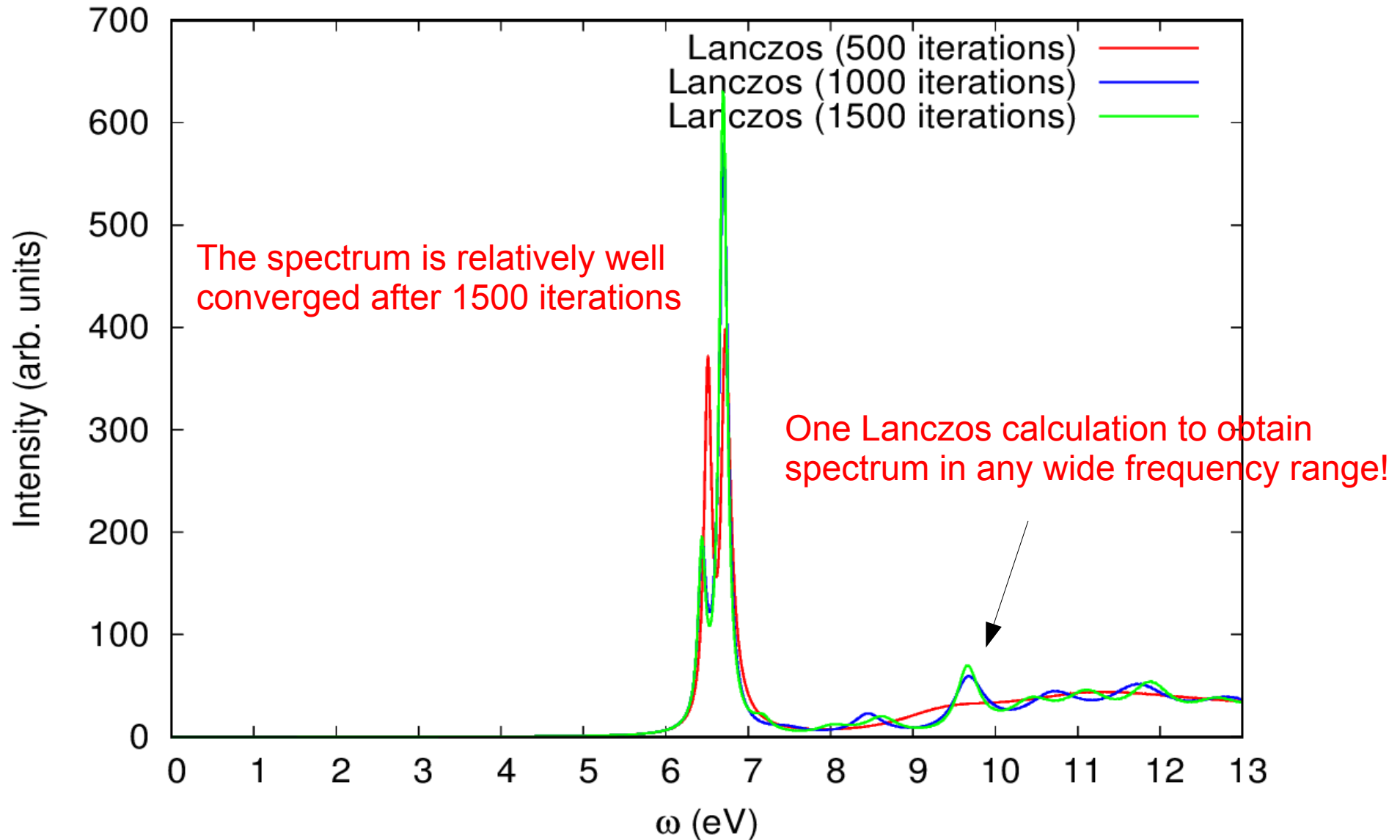
Must be equal !

Extrapolation until 20000

You can use `restart = .true.` option in order not to start from scratch, when you want to increase the number of Lanczos iterations.

## Exercise 2: Post-processing step

With extrapolation of Lanczos coefficients



# Part II – EELS in solids

1. **Exercise 1:** EELS spectrum of Si using Lanczos
2. **Exercise 2:** EELS spectrum of Si using Sternheimer



# Basic equations

Coupled (**resonant** and **anti-resonant**) frequency-dependent Kohn-Sham equations:

$$\begin{aligned}
 (\hat{H}_{\mathbf{k}+\mathbf{q}}^{\circ} - \varepsilon_{n,\mathbf{k}}^{\circ} - \omega) \tilde{u}'_{n,\mathbf{k}+\mathbf{q}}(\mathbf{r}, \omega) + \boxed{\hat{P}_c^{\mathbf{k}+\mathbf{q}} \tilde{v}'_{\text{HXC},\mathbf{q}}(\mathbf{r}, \omega) u_{n,\mathbf{k}}^{\circ}(\mathbf{r})} &= \boxed{-\hat{P}_c^{\mathbf{k}+\mathbf{q}} \tilde{v}'_{\text{ext},\mathbf{q}}(\mathbf{r}, \omega) u_{n,\mathbf{k}}^{\circ}(\mathbf{r})} \\
 (\hat{H}_{\mathbf{k}+\mathbf{q}}^{\circ} - \varepsilon_{n,\mathbf{k}}^{\circ} + \omega) \tilde{u}'_{n,-\mathbf{k}-\mathbf{q}}(\mathbf{r}, -\omega) + \boxed{\hat{P}_c^{\mathbf{k}+\mathbf{q}} \tilde{v}'_{\text{HXC},\mathbf{q}}(\mathbf{r}, \omega) u_{n,\mathbf{k}}^{\circ}(\mathbf{r})} &= \boxed{-\hat{P}_c^{\mathbf{k}+\mathbf{q}} \tilde{v}'_{\text{ext},\mathbf{q}}(\mathbf{r}, \omega) u_{n,\mathbf{k}}^{\circ}(\mathbf{r})}
 \end{aligned}$$

**interaction terms**
**perturbation**

These equations can be re-written as a quantum Liouville equation:

$$(\omega - \mathcal{L}) \cdot \hat{\rho}'_{\mathbf{q}}(\omega) = [\tilde{V}'_{\text{ext},\mathbf{q}}(\omega), \hat{\rho}^{\circ}]$$

$$\mathcal{L} \cdot \hat{\rho}'_{\mathbf{q}} \equiv [\hat{H}^{\circ}, \hat{\rho}'_{\mathbf{q}}] + [\tilde{V}'_{\text{HXC},\mathbf{q}}, \hat{\rho}^{\circ}]$$



$$\chi(\mathbf{q}, \mathbf{q}; \omega)$$

**Susceptibility**



$$-\text{Im}[\varepsilon^{-1}(\mathbf{q}, \omega)] = -\frac{4\pi e^2}{|\mathbf{q}|^2} \text{Im}[\chi(\mathbf{q}, \mathbf{q}; \omega)]$$

**Loss function**

# Exercise 1: Calculation of EELS of bulk silicon

**Step 1.** Perform a Self-Consistent Field ground-state calculation for silicon at the equilibrium structure using the **pw.x** program.

**Si.scf.in**

Input file for the SCF calculation



**pw.x < Si.scf.in > Si.scf.out**



Input



Output

```
&CONTROL
  calculation = 'scf'
  restart_mode='from_scratch',
  pseudo_dir = './',
  outdir='./out',
  prefix='Si'
/
&SYSTEM
 ibrav = 2,
  celldm(1) = 10.26,
  nat = 2,
  ntyp = 1,
  ecutwfc = 16.0
/
&ELECTRONS
  diagonalization='david'
  mixing_mode = 'plain',
  mixing_beta = 0.7,
  conv_thr = 1.0d-8
/
ATOMIC_SPECIES
Si 28.08 Si.pz-vbc.UPF
ATOMIC_POSITIONS
Si 0.00 0.00 0.00
Si 0.25 0.25 0.25
K_POINTS {automatic}
10 10 10 1 1 1
```

# Exercise 1: Calculation of EELS of bulk silicon

**Step 2.** Perform Lanczos recursions using the **turbo\_eels.x** program in order to compute Lanczos coefficients, which will be needed in Step 3.

**Si.eels.in**

```
&lr_input
  prefix = 'Si',
  outdir='./out',
  restart_step = 100,
  restart = .false.
/
&lr_control
  itermax = 400,
  approximation = "TDDFT",
  q1 = 0.866
  q2 = 0.000
  q3 = 0.000
/
```

- ← The same prefix as in the SCF calculation
- ← Directory for temporary files
- ← The code writes restart files every restart\_step iterations
- ← Restart iterations after previous calculation
- ← Number of Lanczos iterations
- ← Level of approximation (IPA, RPA\_with\_CLFE, TDDFT)
- ← Three components of the transferred momentum (in units of  $2\pi/a$  in Cartesian coordinates)

How to specify q1, q2, q3 ? -> Next slide

**turbo\_eels.x < Si.eels.in > Si.eels.out**

# Exercise 1: Calculation of EELS of bulk silicon

**Step 2.** Perform Lanczos recursions using the **turbo\_eels.x** program in order to compute Lanczos coefficients, which will be needed in  $\mathcal{S}^{(n)}$

$$\mathbf{q} = \frac{2\pi}{a_0} (q_1, q_2, q_3)$$

$$a_0 = \text{celldm}(1) = 10.26 \text{ Bohr} \quad - \text{lattice parameter}$$

**Example:**  $|\mathbf{q}| = 0.53 \text{ Bohr}^{-1}$  and  $\mathbf{q} \parallel [100]$  .

Therefore, we obtain:

$$q_1 = \frac{|\mathbf{q}| a_0}{2\pi} = 0.866$$

$$q_2 = 0$$

$$q_3 = 0$$

# Exercise 1: Calculation of EELS of bulk silicon

In the output file **Si.eels.out** there is information about each Lanczos iteration :

```
LANCZOS LINEAR-RESPONSE SPECTRUM CALCULATION
```

```
Number of Lanczos iterations =    400
```

```
Lanczos iteration:      1
alpha(00000001)=  0.000000
beta (00000001)=  1.032661
gamma(00000001)=  1.032661
z1=      1  0.0000000000000000E+00  0.0000000000000000E+00
```

```
Lanczos iteration:      2
alpha(00000002)=  0.000000
beta (00000002)=  2.171113
gamma(00000002)=  2.171113
z1=      1  0.475636553664820E+00  0.224971950790742E-17
```

```
Lanczos iteration:      3
alpha(00000003)=  0.000000
beta (00000003)=  6.245202
gamma(00000003)=  6.245202
z1=      1  0.0000000000000000E+00  0.0000000000000000E+00
```

```
Lanczos iteration:      4
alpha(00000004)=  0.000000
beta (00000004)=  9.192672
gamma(00000004)=  9.192672
z1=      1 -0.323131985294841E+00 -0.134085315550356E-17
```

```
...
```

# Exercise 1: Calculation of EELS of bulk silicon

**Step 3.** Perform a spectrum calculation using the post-processing program **turbo\_spectrum.x** and using the Lanczos coefficients computed in the previous step.

**Si.tddfpt\_pp.in**

```
&lr_input
  prefix = 'Si',
  outdir = './out',
  eels = .true.
  itemax0 = 400,
  itemax = 10000,
  extrapolation = 'osc',
  epsil = 0.035,
  units = 1,
  start = 0.0d0,
  end = 50.0d0,
  increment = 0.01d0
/
```

- ← The same prefix as in the SCF calculation
- ← Directory for temporary files
- ← **Must be .true. for EELS, otherwise absorption**
- ← Number of calculated Lanczos coefficient
- ← Number of extrapolated Lanczos coefficient
- ← Type of extrapolation (bi-constant)
- ← The value of Lorentzian smearing in Ry
- ← The units for start, end and increment
- ← Minimum value of frequencies for a plot in eV
- ← Maximum value of frequencies for a plot in eV
- ← Frequency step in eV

**turbo\_spectrum.x < Si.tddfpt\_pp.in > Si.tddfpt\_pp.out**

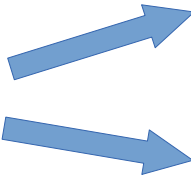
# Exercise 1: Calculation of EELS of bulk silicon

The code `turbo_spectrum.x` produces a file `Si.plot_chi.dat` which contains the real and imaginary parts of the polarizability  $\chi(\mathbf{q}, \mathbf{q}; \omega)$  :

```
#          \omega (eV)          Re(chi)          Im(chi)
chi_1_1=  0.0000000000000000E+00  -.170220600317043E+00  -.803614869812834E-17
chi_1_1=  0.1000000000000000E-01  -.170220704844176E+00  -.106347451821822E-04
chi_1_1=  0.2000000000000000E-01  -.170221018426756E+00  -.212702336077273E-04
chi_1_1=  0.3000000000000000E-01  -.170221541068313E+00  -.319072089820873E-04
chi_1_1=  0.4000000000000000E-01  -.170222272774703E+00  -.425464159351046E-04
chi_1_1=  0.5000000000000000E-01  -.170223213554069E+00  -.531886004838475E-04
chi_1_1=  0.6000000000000000E-01  -.170224363416783E+00  -.638345104961852E-04
chi_1_1=  0.7000000000000000E-01  -.170225722375369E+00  -.744848961553472E-04
chi_1_1=  0.8000000000000000E-01  -.170227290444417E+00  -.851405104256942E-04
chi_1_1=  0.9000000000000000E-01  -.170229067640472E+00  -.958021095199170E-04
chi_1_1=  0.1000000000000000E+00  -.170231053981908E+00  -.106470453367859E-03
chi_1_1=  0.1100000000000000E+00  -.170233249488786E+00  -.117146306087144E-03
chi_1_1=  0.1200000000000000E+00  -.170235654182687E+00  -.127830436455769E-03
chi_1_1=  0.1300000000000000E+00  -.170238268086534E+00  -.138523618386789E-03
chi_1_1=  0.1400000000000000E+00  -.170241091224386E+00  -.149226631405204E-03
chi_1_1=  0.1500000000000000E+00  -.170244123621218E+00  -.159940261127115E-03
chi_1_1=  0.1600000000000000E+00  -.170247365302674E+00  -.170665299741172E-03
...          ...          ...
```

# Exercise 1: Calculation of EELS of bulk silicon

The code `turbo_spectrum.x` also produces a file `Si.plot_eps.dat` which contains the real and imaginary parts of the dielectric function and its inverse:

$$\varepsilon^{-1}(\mathbf{q}, \omega) = 1 + \frac{4\pi e^2}{|\mathbf{q}|^2} \chi(\mathbf{q}, \mathbf{q}; \omega)$$

$$\begin{aligned}\operatorname{Re}[\varepsilon^{-1}(\mathbf{q}, \omega)] &= 1 + \frac{4\pi e^2}{|\mathbf{q}|^2} \operatorname{Re}[\chi(\mathbf{q}, \mathbf{q}; \omega)] \\ -\operatorname{Im}[\varepsilon^{-1}(\mathbf{q}, \omega)] &= -\frac{4\pi e^2}{|\mathbf{q}|^2} \operatorname{Im}[\chi(\mathbf{q}, \mathbf{q}; \omega)]\end{aligned}$$

$$\varepsilon(\mathbf{q}, \omega) = \frac{1}{\varepsilon^{-1}(\mathbf{q}, \omega)} = \underbrace{\frac{\operatorname{Re}[\varepsilon^{-1}]}{(\operatorname{Re}[\varepsilon^{-1}])^2 + (\operatorname{Im}[\varepsilon^{-1}])^2}}_{\operatorname{Re}[\varepsilon(\mathbf{q}, \omega)]} + i \underbrace{\frac{-\operatorname{Im}[\varepsilon^{-1}]}{(\operatorname{Re}[\varepsilon^{-1}])^2 + (\operatorname{Im}[\varepsilon^{-1}])^2}}_{\operatorname{Im}[\varepsilon(\mathbf{q}, \omega)]}$$



# Exercise 1: Calculation of EELS of bulk silicon

The code `turbo_spectrum.x` also produces a file `Si.plot_eps.dat` which contains the real and imaginary parts of the dielectric function and its inverse:

└

#

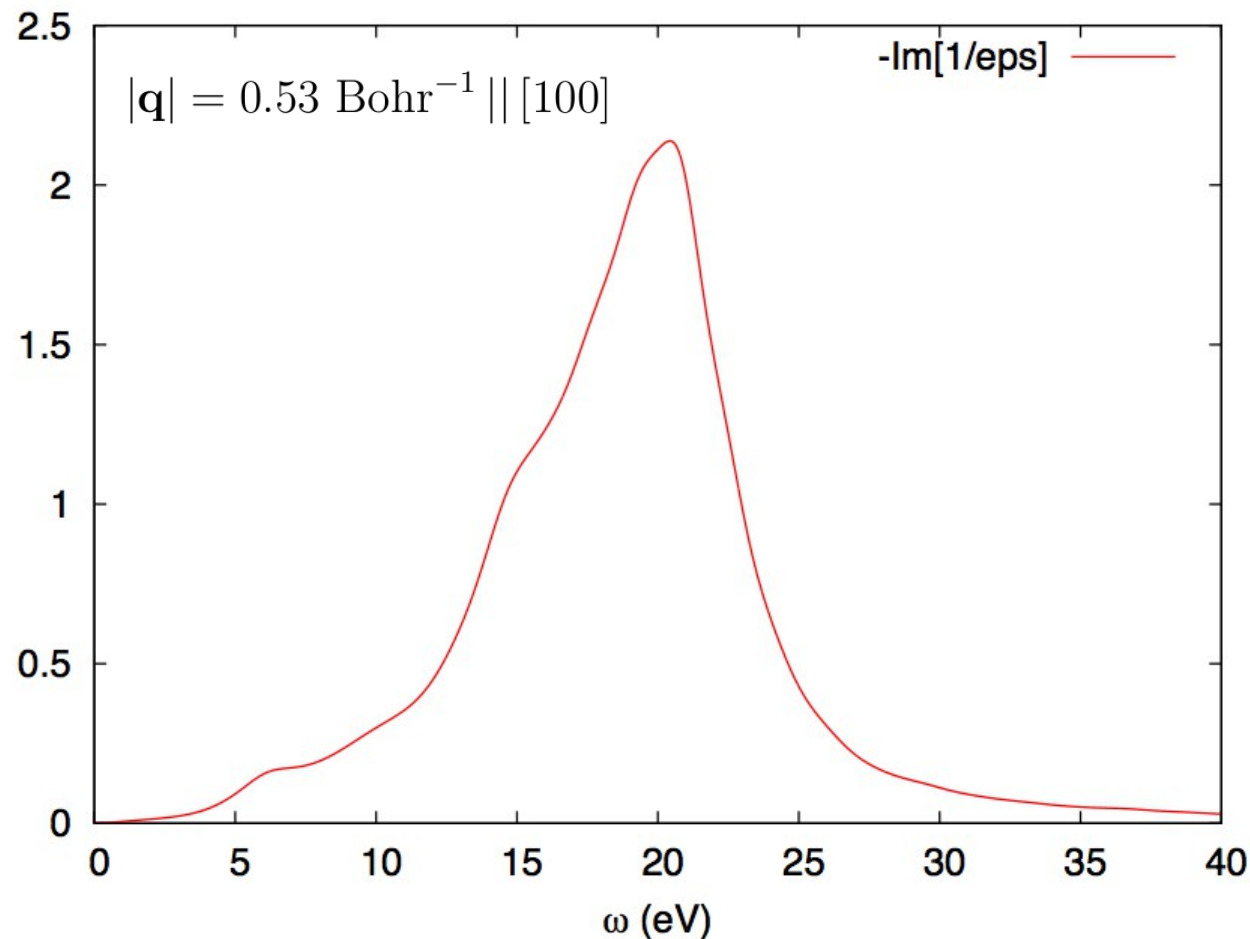
Frequency (eV)	Re(1/eps)	-Im(1/eps)	Re(eps)	Im(eps)
0.000000000000000E+00	0.233539601856329E+00	0.361847491974360E-16	0.428192902638924E+01	0.663444343783860E-15
0.100000000000000E-01	0.233539131197030E+00	0.478855732579376E-04	0.428193747588143E+01	0.877981474180613E-03
0.200000000000000E-01	0.233537719213824E+00	0.957744931531374E-04	0.428196282460077E+01	0.175604532153211E-02
0.300000000000000E-01	0.233535365890814E+00	0.143670108403533E-03	0.428200507327490E+01	0.263427395981432E-02
0.400000000000000E-01	0.233532071201626E+00	0.191575771889353E-03	0.428206422311414E+01	0.351274989598586E-02
0.500000000000000E-01	0.233527835109602E+00	0.239494842737156E-03	0.428214027580785E+01	0.439155577086439E-02
0.600000000000000E-01	0.233522657568039E+00	0.287430688407188E-03	0.428223323351964E+01	0.527077440385873E-02
0.700000000000000E-01	0.233516538520538E+00	0.335386686785180E-03	0.428234309888067E+01	0.615048883779445E-02
0.800000000000000E-01	0.233509477901409E+00	0.383366228279594E-03	0.428246987498165E+01	0.703078238385626E-02
...	...	...	...	...

# Exercise 1: Calculation of EELS of bulk silicon

Step 4. Plot the spectrum using **gnuplot** and the script

`plot_spectrum.gnu`

— Electron Energy Loss Spectrum of bulk silicon (file `Spectrum.eps`):



Is the spectrum converged? -> **exercise2**

What is the origin of peaks?

# Exercise 1: Calculation of EELS of bulk silicon

The code `turbo_spectrum.x` also produces a file `Si.plot_eps.dat` which contains the real and imaginary parts of the dielectric function and its inverse:

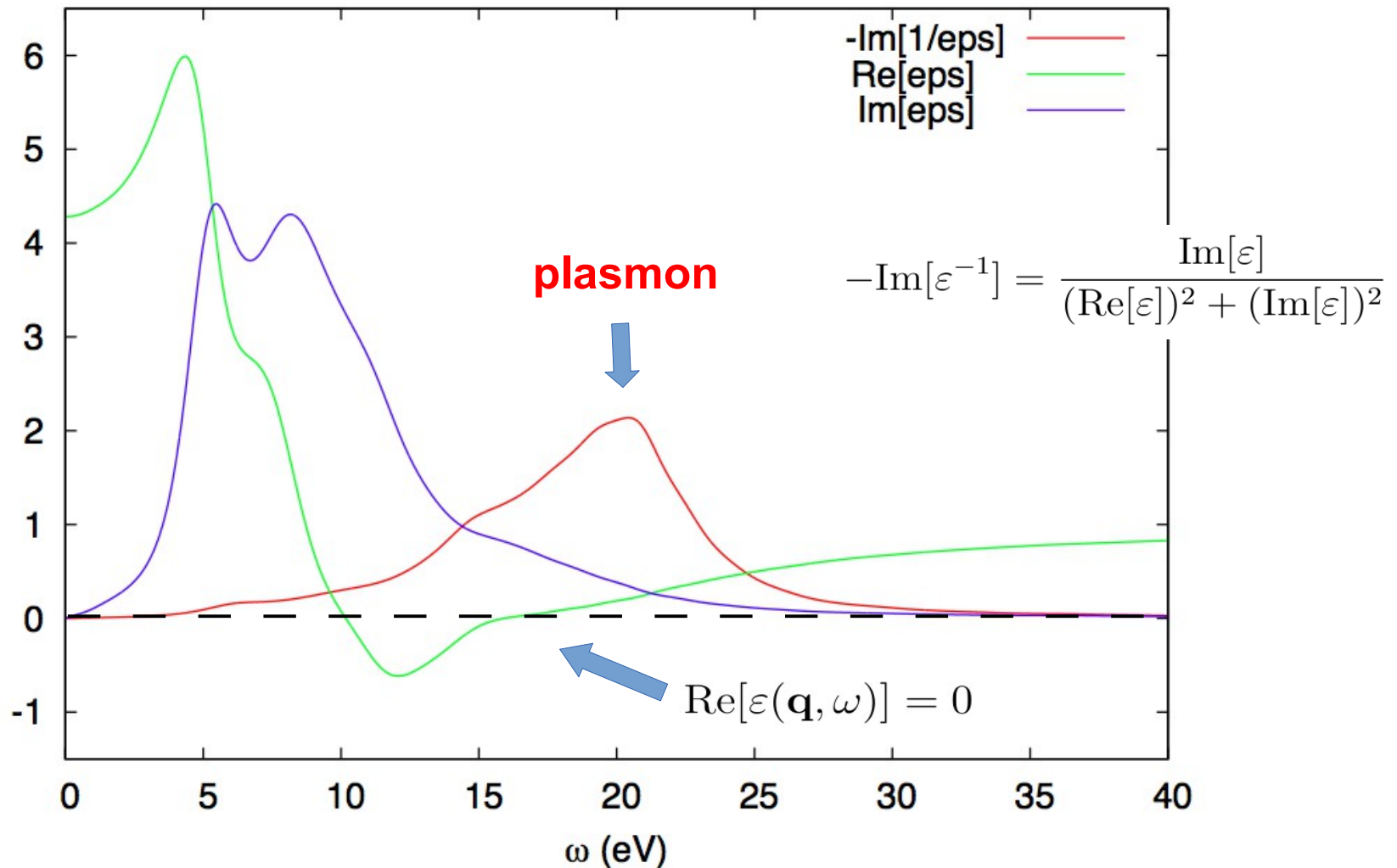
└

#

Frequency (eV)	Re(1/eps)	-Im(1/eps)	Re(eps)	Im(eps)
0.000000000000000E+00	0.233539601856329E+00	0.361847491974360E-16	0.428192902638924E+01	0.663444343783860E-15
0.100000000000000E-01	0.233539131197030E+00	0.478855732579376E-04	0.428193747588143E+01	0.877981474180613E-03
0.200000000000000E-01	0.233537719213824E+00	0.957744931531374E-04	0.428196282460077E+01	0.175604532153211E-02
0.300000000000000E-01	0.233535365890814E+00	0.143670108403533E-03	0.428200507327490E+01	0.263427395981432E-02
0.400000000000000E-01	0.233532071201626E+00	0.191575771889353E-03	0.428206422311414E+01	0.351274989598586E-02
0.500000000000000E-01	0.233527835109602E+00	0.239494842737156E-03	0.428214027580785E+01	0.439155577086439E-02
0.600000000000000E-01	0.233522657568039E+00	0.287430688407188E-03	0.428223323351964E+01	0.527077440385873E-02
0.700000000000000E-01	0.233516538520538E+00	0.335386686785180E-03	0.428234309888067E+01	0.615048883779445E-02
0.800000000000000E-01	0.233509477901409E+00	0.383366228279594E-03	0.428246987498165E+01	0.703078238385626E-02
...	...	...	...	...

# Exercise 1: Calculation of EELS of bulk silicon

What is the origin of peaks?



# Outline

1. Introduction
2. **Exercise 1:** Calculation of EELS of bulk silicon
3. **Exercise 2:** Convergence of EELS of bulk silicon
4. **Exercise 3:** Calculation of EELS of Al(111)

## Exercise 2: Convergence of EELS of bulk silicon

Go to the directory with the input files:

```
cd Hands-on_TDDFPT_EELS/exercise2
```

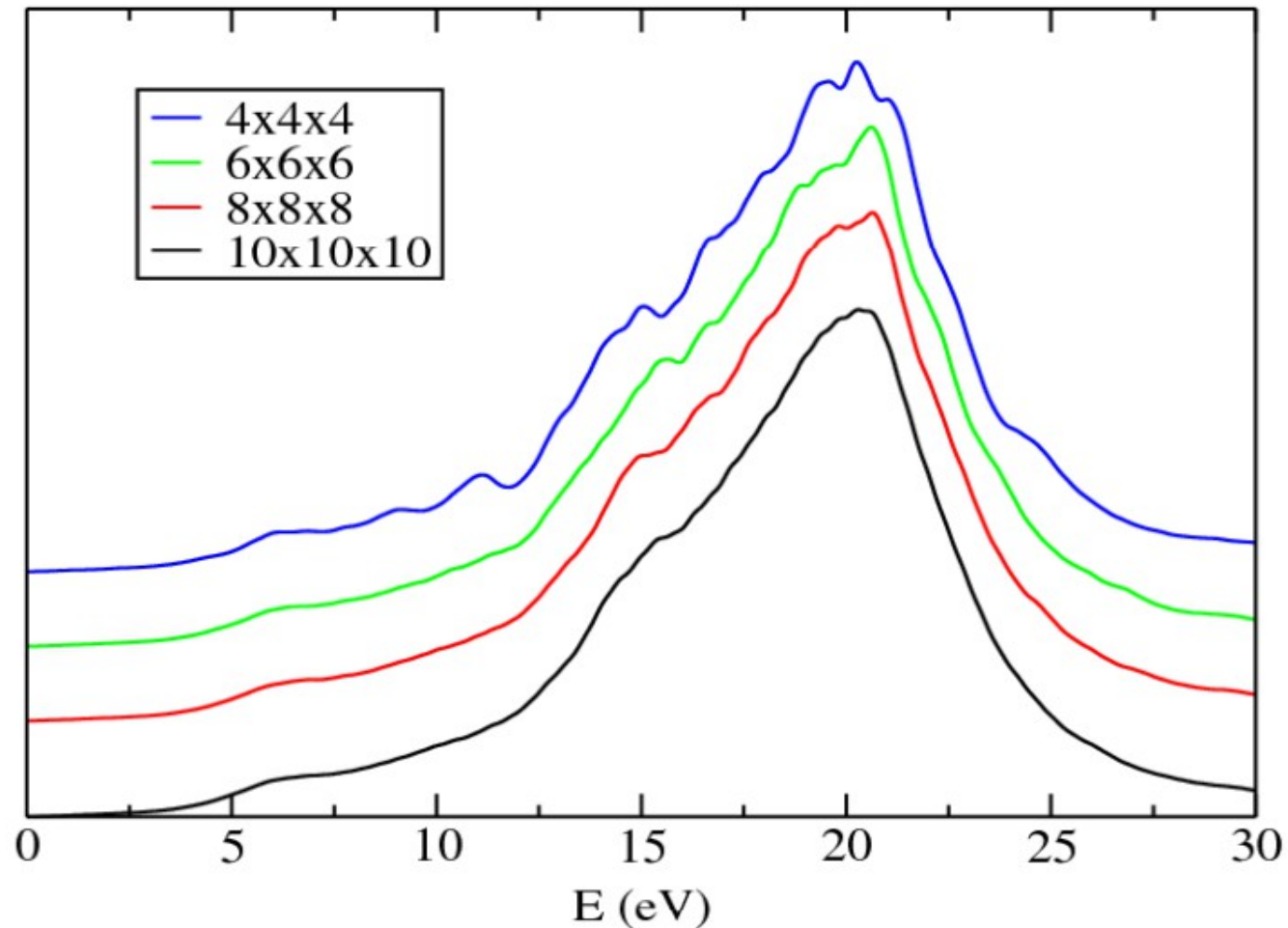
1. Study the convergence of the loss spectrum of bulk silicon with respect to:
  - **k** points mesh
  - Number of Lanczos iterations

2. Compute the **plasmon dispersion**, i.e. change in the position of the plasmon peak when changing the value of the transferred momentum:

$|\mathbf{q}| = 0.1, 0.2, 0.3, \dots \text{Bohr}^{-1}$  along the  $[100]$  direction.

## Exercise 2: Convergence of EELS of bulk silicon

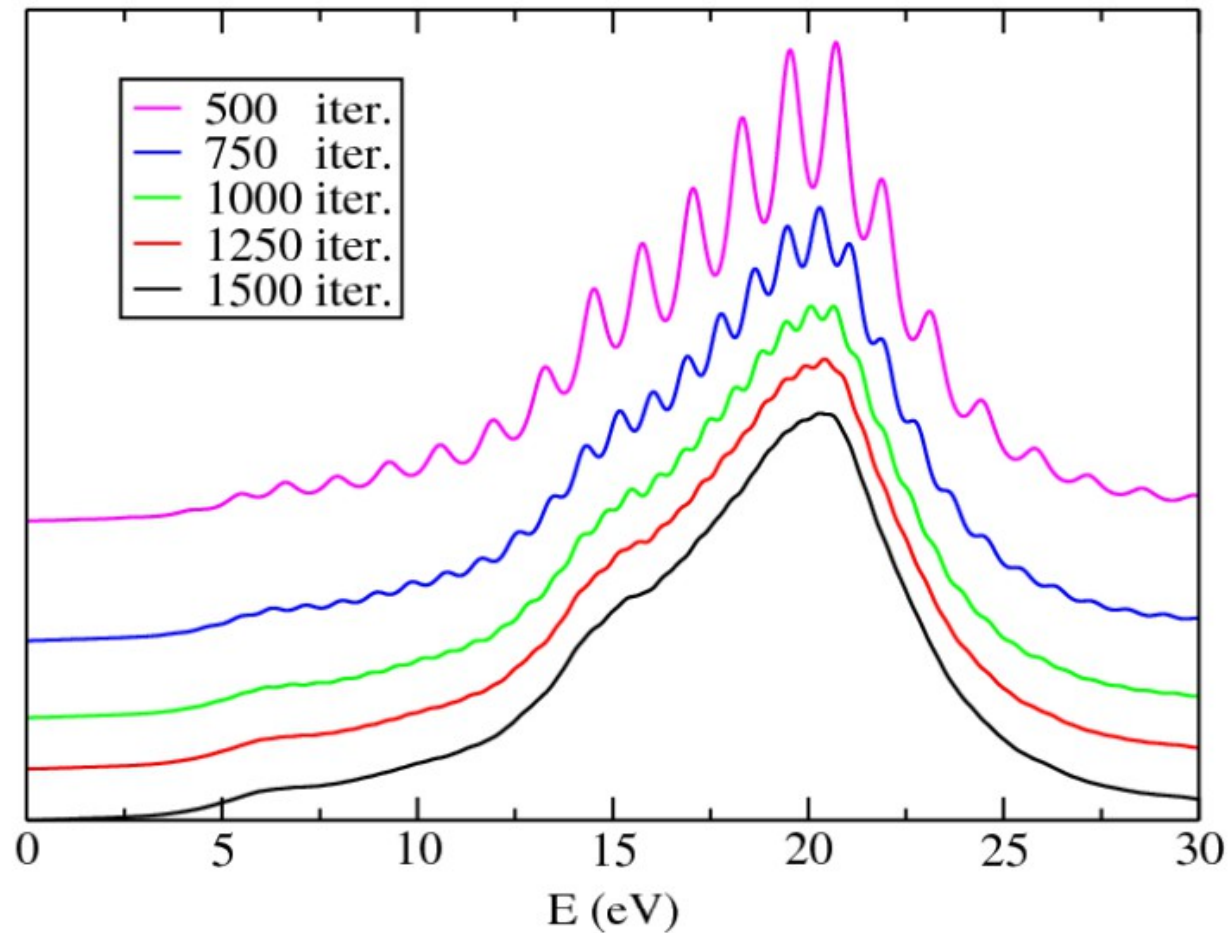
Convergence of EELS with respect to the  $\mathbf{k}$  points mesh



Here, 400 Lanczos iterations + extrapolation was used.

## Exercise 2: Convergence of EELS of bulk silicon

Convergence of EELS with respect to the number of Lanczos iterations  
(no extrapolation)

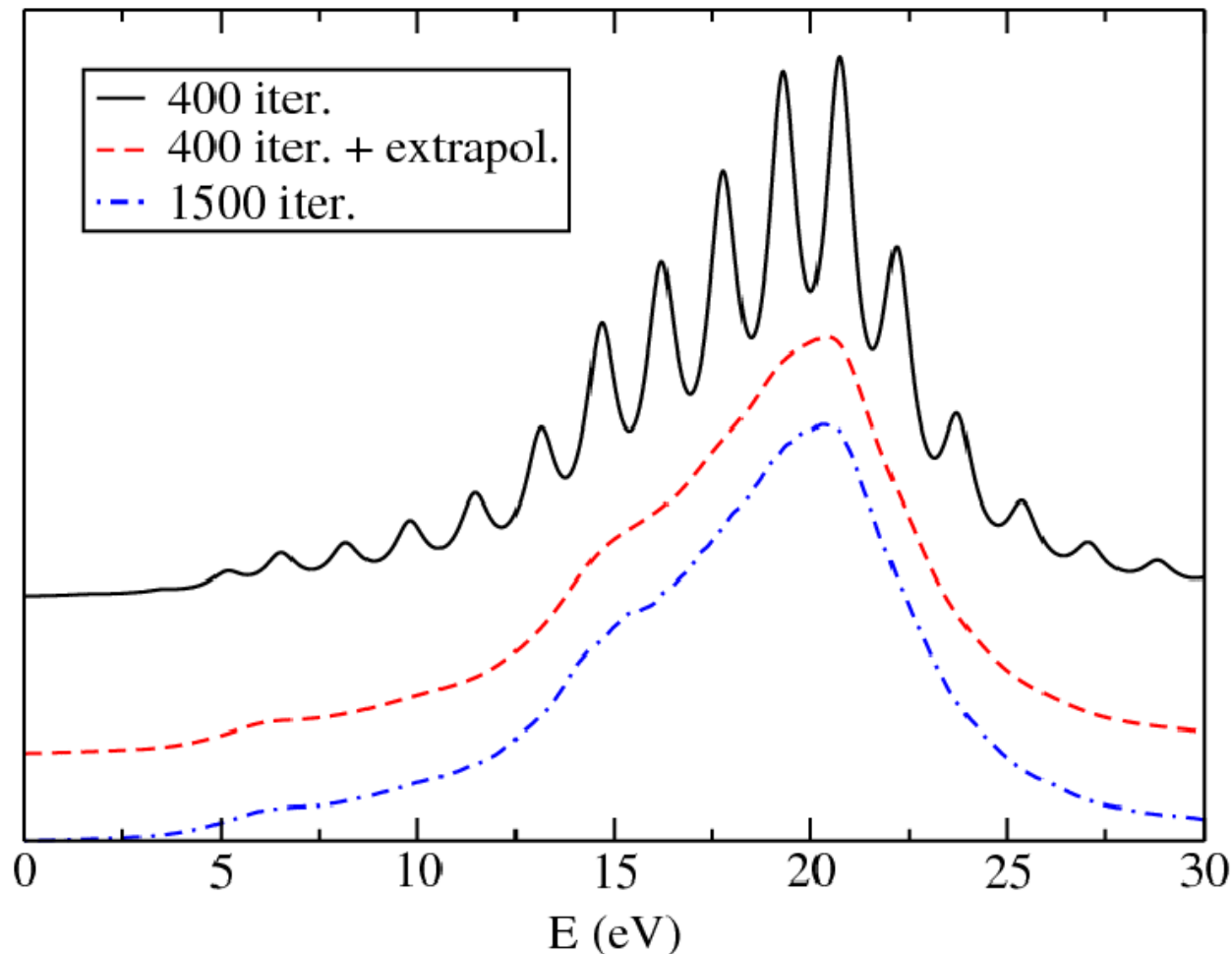


Here, k points mesh 10x10x10 1 1 1 was used.



## Exercise 2: Convergence of EELS of bulk silicon

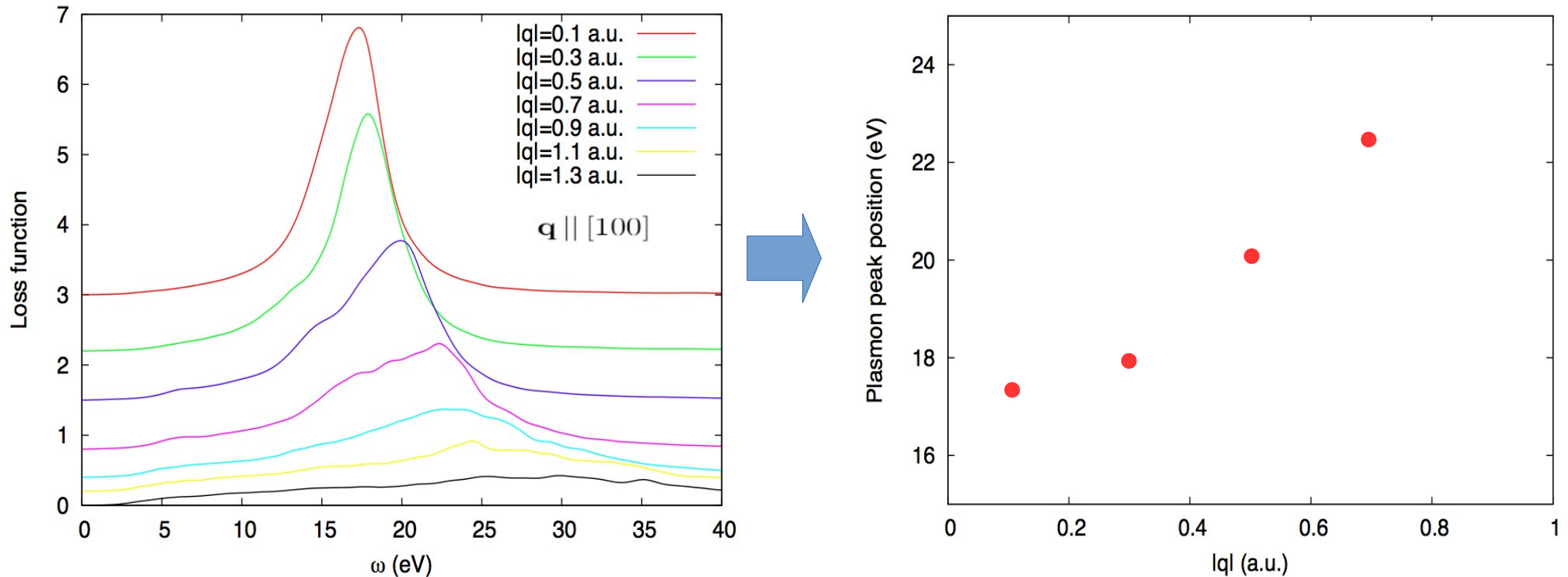
Convergence of EELS with respect to the number of Lanczos iterations  
(with and without extrapolation)



Here, k points mesh 10x10x10 1 1 1 was used.

# Exercise 2: Convergence of EELS of bulk silicon

## Plasmon dispersion in bulk silicon

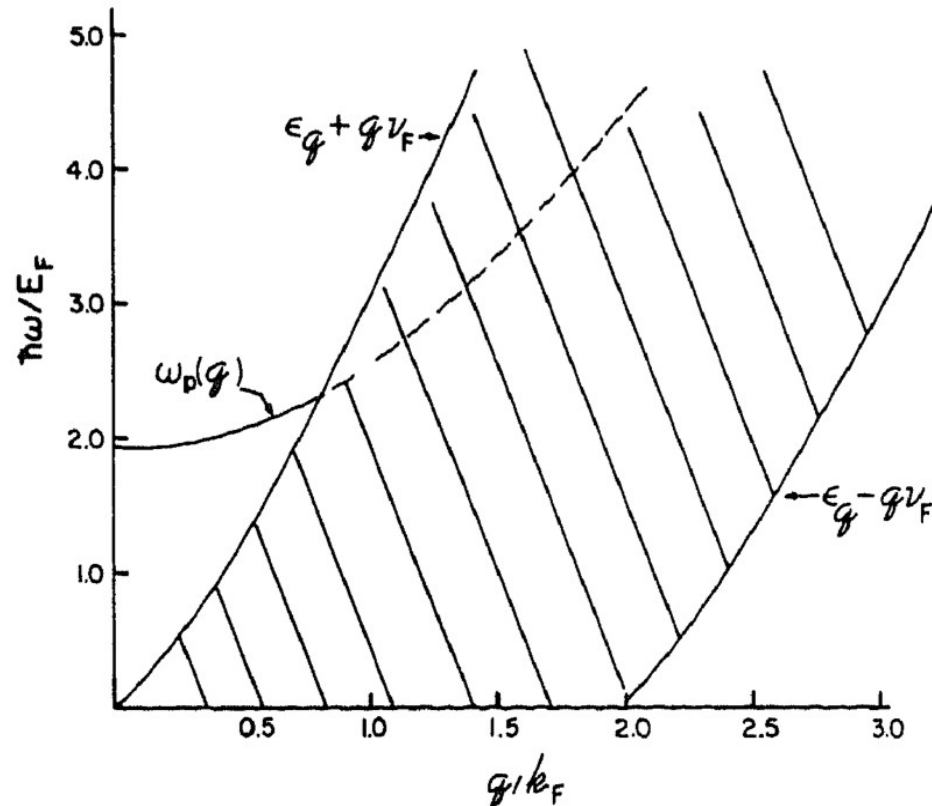


Here, k points mesh 10x10x10 1 1 1 was used and 400+extrapol. Lanczos iterations.

By increasing  $q$ , the plasmon peak blue-shifts and broadens, because it enters in the electron-hole continuum.

## Exercise 2: Convergence of EELS of bulk silicon

The excitation region of the electron gas in  $(\mathbf{q}, \omega)$  space



The plasmon line becomes highly damped in the region of electron-hole pairs, which is shown hatched.