VASP exercises II

Outline

- Fcc Si
  - lattice relaxation

Fcc Si

- search for the optimal volume (lattice parameter)
  - use of equation of state (EOS) for solids, which relates the energy of the system to the its volume
    - e.g. harmonic solid

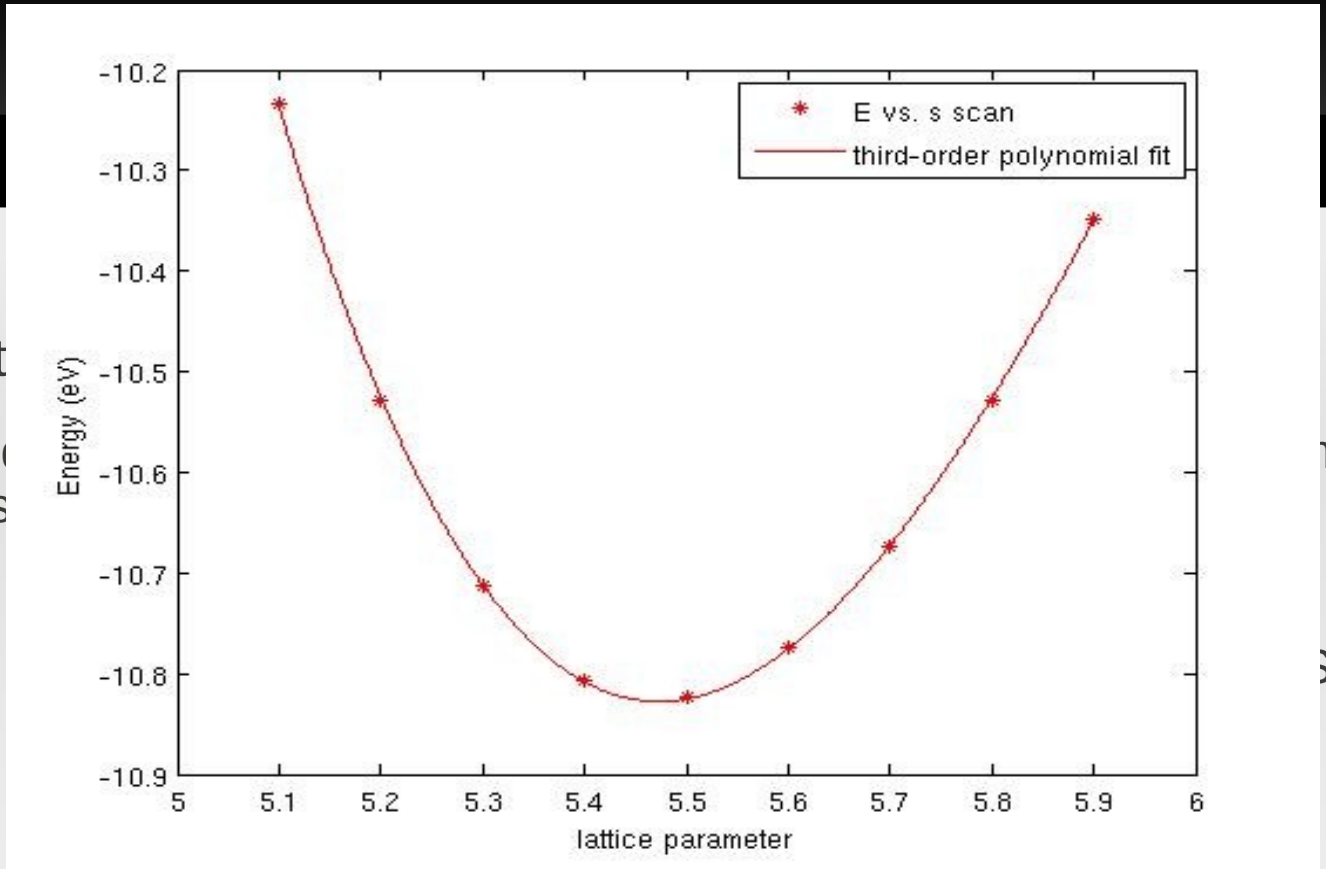
$$E = E_0 + \frac{1}{2} B_0 \frac{(V - V_0)^2}{V_0}.$$

    - or more sophisticated third-order Birch Murnaghan EOS

$$E(V) = E_0 + \frac{9 V_0 B_0}{16} \left\{ \left[\left(\frac{V_0}{V} \right)^{\frac{2}{3}} - 1 \right]^3 B_0' + \left[\left(\frac{V_0}{V} \right)^{\frac{2}{3}} - 1 \right]^2 \left[6 - 4 \left(\frac{V_0}{V} \right)^{\frac{2}{3}} \right] \right\}.$$

Fcc Si

- search for t
  - use of e ne system to the its s
  - generally - the polynomial fit of E over a certain volume range
    - if no internal degrees of freedom present (e.g. the non-cubic cell, or more atoms in the primitive cell) – possible to use E vs. lattice parameter fit
    - if internal degrees present – the optimization of atomic positions or/and shape needed – controlled by the ISIF keyword (ISIF = 4)
  - find the minimum => the optimal lattice parameter



Fcc Si

ISIF: - determines which degrees of freedom are allowed to change
=2 – (default) internal degrees of freedom (IDF) are relaxed
=3 – full relaxation (volume, cell shape, IDF)
=4 – IDF and cell shape is optimized

- search for the optimal lattice parameter
  - use of equation of state (EOS) for solids, which relates the energy of the system to the its volume
    - e.g. harmonic solid $$E = E_0 + \frac{1}{2} B_0 \frac{(V - V_0)^2}{V_0}.$$
    - or more sophisticated third-order Birch Murnaghan EOS

$$E(V) = E_0 + \frac{9 V_0 B_0}{16} \left\{ \left[\left(\frac{V_0}{V} \right)^{\frac{2}{3}} - 1 \right]^3 B_0' + \left[\left(\frac{V_0}{V} \right)^{\frac{2}{3}} - 1 \right]^2 \left[6 - 4 \left(\frac{V_0}{V} \right)^{\frac{2}{3}} \right] \right\}.$$

  - generally - the polynomial fit of E over a certain volume range
    - if no internal degrees of freedom present (e.g. the non-cubic cell, or more atom in the primitive cell) – possible to use E vs. lattice parameter fit
    - if internal degrees present – the optimization of atomic positions or/and shape needed – controlled by the ISIF keyword (ISIF = 4 or 2 for cubic cell)
  - find the minimum => the optimal lattice parameter
  - simpler “automatic” approach – full relaxation (internal degrees + volume) – ISIF = 3
    - but problematic due to basis set incompleteness – Pulay stress

Fcc Si – Pulay stress

$$\mathbf{b}_1 = 2\pi \frac{\mathbf{a}_2 \times \mathbf{a}_3}{\mathbf{a}_1 \cdot (\mathbf{a}_2 \times \mathbf{a}_3)}$$

$$|\mathbf{G} + \mathbf{k}| < G_{\text{cut}} \qquad \text{with} \qquad E_{\text{cut}} = \frac{\hbar^2}{2m} G_{\text{cut}}^2$$

- the volume relaxation algorithm (ISIF = 3) works with the fixed basis set size

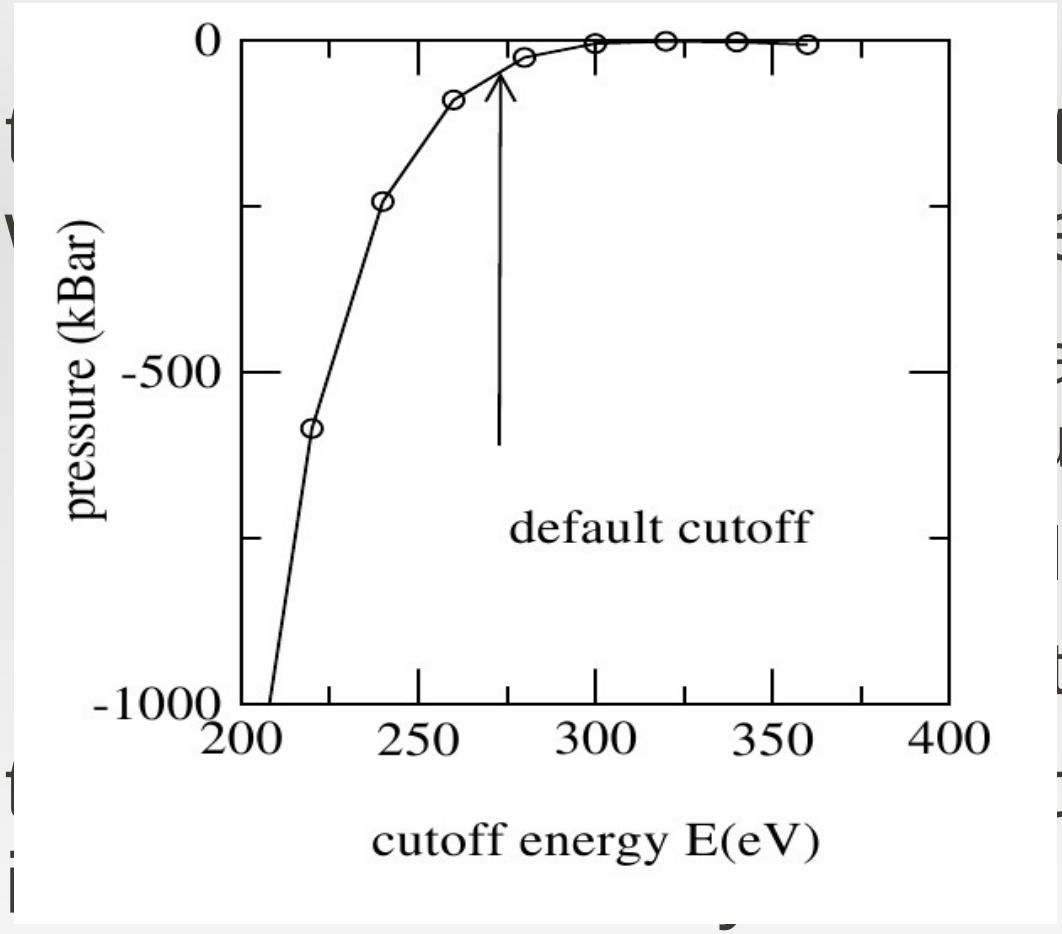
$$G = x_1 * b_1 + x_2 * b_2 + x_3 * b_3$$

  - the same number of plane waves for larger volume leads effectively to smaller cutoff
  - => the energy is too high for larger volumes
  - => the equilibrium volume is thus too small
- the effect is due to basis set incompleteness – i.e. the due to Pulay stress (force)
  - increase of the cutoff by 30% (compared to default one) and rerun the optimization

Fcc Si – Pulay stress

$$\mathbf{b_1} = 2\pi \frac{\mathbf{a_2} \times \mathbf{a_3}}{\mathbf{a_1} \cdot (\mathbf{a_2} \times \mathbf{a_3})}$$

$$|\mathbf{G} + \mathbf{k}| < G_{\text{cut}} \qquad \text{with} \qquad E_{\text{cut}} = \frac{\hbar^2}{2m} G_{\text{cut}}^2$$



- thm (ISIF = 3)
  et size

$$G = x_1 * b_1 + x_2 * b_2 + x_3 * b_3$$

  aves for larger volume
  utoff
  larger volumes
  thus too small
- : incompleteness –
  - increase of the cutoff by 30% (compared to default one) and rerun the optimization

Fcc Si - VASP run

- `cd cviceni_2/ex_1`
- POTCAR for si at `~grajciar/School/QCH../potpaw_PBE/Si`
- INCAR

```
SYSTEM = Si diamond            !title (optional)
ISMEAR = 0 ; SIGMA = 0.1
ENCUT = 240
```

- KPOINTS

```
Automatic mesh
0                      !automatic generation ('zero')
Monkhorst Pack         !method of automatic generatioin
11 11 11               !num. of subdivisions along reciprocal vectors
 0  0  0               !shift of the k-point mesh
```

- POSCAR

```
fcc Si
   3.9
    0.5   0.5   0.0
    0.0   0.5   0.5
    0.5   0.0   0.5
   Si
   1
Cartesian
    0    0    0
```

Fcc Si - VASP run

- we shall use simple bash script to obtain the E vs. lattice parameter scan – scan.sh
  - generates geom_scan.dat file

3.5	-.44256712E+01
3.6	-.46614699E+01
3.7	-.47979864E+01
3.8	-.48645041E+01
3.9	-.48773847E+01
4.0	-.48487437E+01
4.1	-.47852634E+01
4.2	-.46936947E+01
4.3	-.45831167E+01

- find the optimal lattice parameter by fitting to polynomial in MATLAB
- 3,874 Å – optimal value

```
#!/bin/bash
rm WAVECAR CONTCAR
for i in 3.5 3.6 3.7 3.8 3.9 4.0 4.1 4.2 4.3
#for i in 4.2 4.3
do
rm CONTCAR
cat > POSCAR <<EOF
fcc Si
   $i
    0.5    0.5    0.0
    0.0    0.5    0.5
    0.5    0.0    0.5
   Si
   1
Cartesian
    0    0    0
EOF
echo "parameter = $i"
VASP mq 8 1.3G
while [ ! -s CONTCAR ];do
sleep 5
done

E=`tail -1 OSZICAR|awk '{print $3}'`
echo $i $E >> geom_scan.dat
done
```

Fcc Si - full relaxation

- `cd ../ex_2`
- POTCAR, KPOINTS and POSCAR from previous run
- INCAR

```
SYSTEM = Si diamond              !title (optional)
ISMEAR = 0  ; SIGMA = 0.1
ENCUT = 240
IBRION = 2; ISIF = 3;NSW = 15
EDIFF = 0.1E-04
EDIFFG = -0.01                   !conv. criteria for ionic steps
```

EDIFFG: determines the convergence criteria for the ionic relaxation
=real number (positive if the change in the energies is smaller than EDIFFG
negative if the change of all forces is less then |EDIFFG|
Default = EDIFF*10

Fcc Si - full relaxation

EDIFFG: determines the convergence criteria for the ionic relaxation
=real number (positive if the change in the energies is smaller than EDIFFG
negative if the change of all forces is less then |EDIFFG|
Default = EDIFF*10

- `cd ../ex_2`
- POTCAR, KPOINTS and POSCAR from previous run
- INCAR

```
SYSTEM = Si diamond              !title (optional)
ISMEAR = 0  ; SIGMA = 0.1
ENCUT = 240
IBRION = 2; ISIF = 3;NSW = 15
EDIFF = 0.1E-04
EDIFFG = -0.01                   !conv. criteria for ionic steps
```

- CONTCAR
  - optimal lattice from full relaxation is 3.867 Å
  - smaller then form E vs. lattice parameter fit 3.874 Å – due to Pulay stress

```
fcc Si
   3.90000000000000
     0.4957651233838283   0.4957651233838283   0.0000000000000000
     0.0000000000000000   0.4957651233838283   0.4957651233838283
     0.4957651233838283   0.0000000000000000   0.4957651233838283
   Si
   1
Direct
  0.0000000000000000  0.0000000000000000  0.0000000000000000

  0.00000000E+00  0.00000000E+00  0.00000000E+00
```

Outline

- Fcc Si
  - lattice relaxation
  - k-points study

Fcc Si - k-points study

```
Automatic mesh
0                !automatic generation ('zero')
Monkhorst Pack   !method of automatic generatioin
11 11 11         !num. of subdivisions along reciprocal vectors
  0   0  0       !shift of the k-point mesh
```

- what k-point mesh size to use?
- mind that number of k-points does not have to increase with mesh-size
  - e. g. for the FCC the even grids not centered at the Gamma(0,0,0) point are larger (e. g. 8x8x8=>60 k-points) than odd grids (9x9x9=>35 k-points)
- the grids centered at Gamma point (line 3 starting with 'G') might behave differently than e. g. Monkhorst Pack grid (line 3 startig with 'M') shifted away from Gamma point (see above)
- the actual k-points generated by automatic procedure are stored in IBZKPT file – this should be then directly related to the k-point sampling precision
- General recommendation:
  - use even meshes up to ~8 (8x8x8), use the odd meshes from there up (e. g. 11x11x11)
  - check the IBZKPT file to see how many k-points have been actually generated
  - use 'G' grid for hexagonal lattices – 'M' grids effectively lowers the cell symmetry leading to slower convergence

Fcc Si - k-points study

- `cd ../ex_3/`
- copy POSCAR, INCAR and POTCAR from ex_1
- use scan.sh script to generate k-point scan
- run by ./scan.sh
- various files generated
- scan.dat – mesh size vs. energy

```
1 0.64239277E+01
2 -.42083843E+01
3 -.47424589E+01
4 -.45562350E+01
5 -.45498122E+01
6 -.45747892E+01
7 -.45680113E+01
8 -.45738961E+01
9 -.45954740E+01
```

```
#!/bin/bash
for i in 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
do
if [ -a OUTCAR ]
then
        rm OUTCAR
fi
touch OUTCAR
cat > KPOINTS <<EOF
K - points
0
Monkhorst Pack
$i $i $i
0  0  0
EOF
echo "parameter = $i"
VASP mq 8 1.3G
while [ "General" != "$(grep 'General timing' OUTCAR|awk '{print $1}')" ];do
sleep 5
done

E=`tail -1 OSZICAR|awk '{print $3}'`
timing=`grep 'User time' OUTCAR|awk '{print $4}'`
kpoints=`head -2 IBZKPT|tail -1`
echo $i $E >> scan.dat
echo $kpoints $E >>  kpoints.dat
cp OUTCAR "OUTCAR_"$i
echo $i $timing >> timing.dat
echo $kpoints $timing >> kpoints_timing.dat
echo $i $kpoints >> kpoints_mesh.dat
done
```

Fcc Si - k-points study

- `cd ../ex_3/`
- copy POSCAR, INCAR and POTCAR from ex_1
- use scan.sh script to generate k-point scan
- run by ./scan.sh
- various files generated
- kpoints_mesh.dat – mesh size vs. actual number of k-points

```
1 1
2 2
3 4
4 10
5 10
6 28
7 20
8 60
9 35
```

```
#!/bin/bash
for i in 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
do
if [ -a OUTCAR ]
then
        rm OUTCAR
fi
touch OUTCAR
cat > KPOINTS <<EOF
K - points
0
Monkhorst Pack
$i $i $i
0  0  0
EOF
echo "parameter = $i"
VASP mq 8 1.3G
while [ "General" != "$(grep 'General timing' OUTCAR|awk '{print $1}')" ];do
sleep 5
done

E=`tail -1 OSZICAR|awk '{print $3}'`
timing=`grep 'User time' OUTCAR|awk '{print $4}'`
kpoints=`head -2 IBZKPT|tail -1`
echo $i $E >> scan.dat
echo $kpoints $E >>  kpoints.dat
cp OUTCAR "OUTCAR_"$i
echo $i $timing >> timing.dat
echo $kpoints $timing >> kpoints_timing.dat
echo $i $kpoints >> kpoints_mesh.dat
done
```

Fcc Si - k-points study

- `cd ../ex_3/`
- copy POSCAR, INCAR and POTCAR from ex_1
- use scan.sh script to generate k-point scan
- run by ./scan.sh
- various files generated
- kpoints_timing.dat – actual number of k-points vs. time

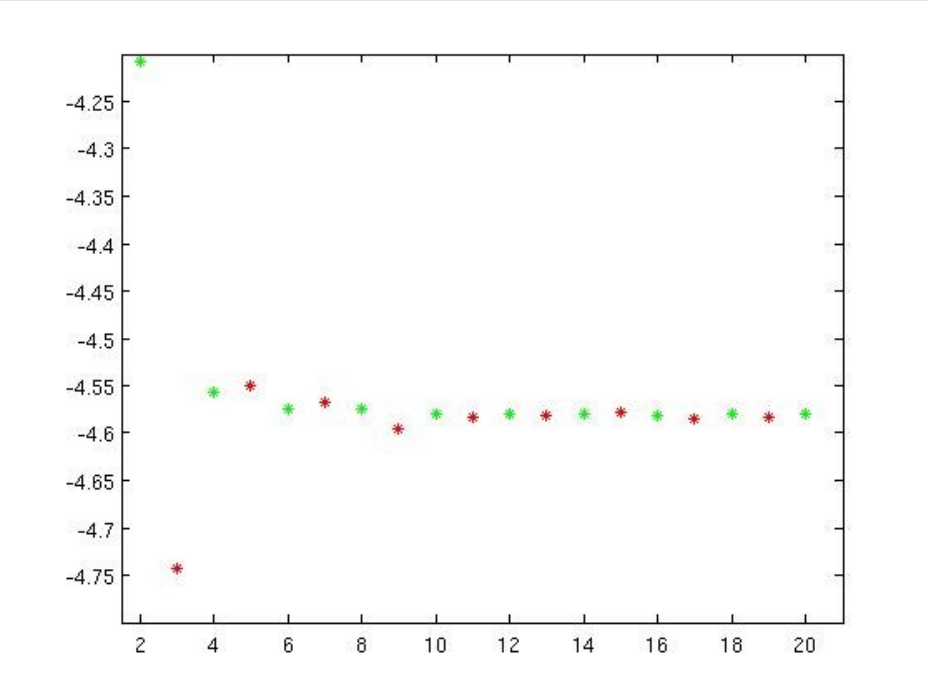
```
1 0.380
2 0.350
4 0.300
10 0.460
10 0.470
28 0.740
20 0.620
60 1.380
35 0.850
```

```
#!/bin/bash
for i in 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
do
if [ -a OUTCAR ]
then
        rm OUTCAR
fi
touch OUTCAR
cat > KPOINTS <<EOF
K - points
0
Monkhorst Pack
$i $i $i
0  0  0
EOF
echo "parameter = $i"
VASP mq 8 1.3G
while [ "General" != "$(grep 'General timing' OUTCAR|awk '{print $1}')" ];do
sleep 5
done

E=`tail -1 OSZICAR|awk '{print $3}'`
timing=`grep 'User time' OUTCAR|awk '{print $4}'`
kpoints=`head -2 IBZKPT|tail -1`
echo $i $E >> scan.dat
echo $kpoints $E >>  kpoints.dat
cp OUTCAR "OUTCAR_"$i
echo $i $timing >> timing.dat
echo $kpoints $timing >> kpoints_timing.dat
echo $i $kpoints >> kpoints_mesh.dat
done
```

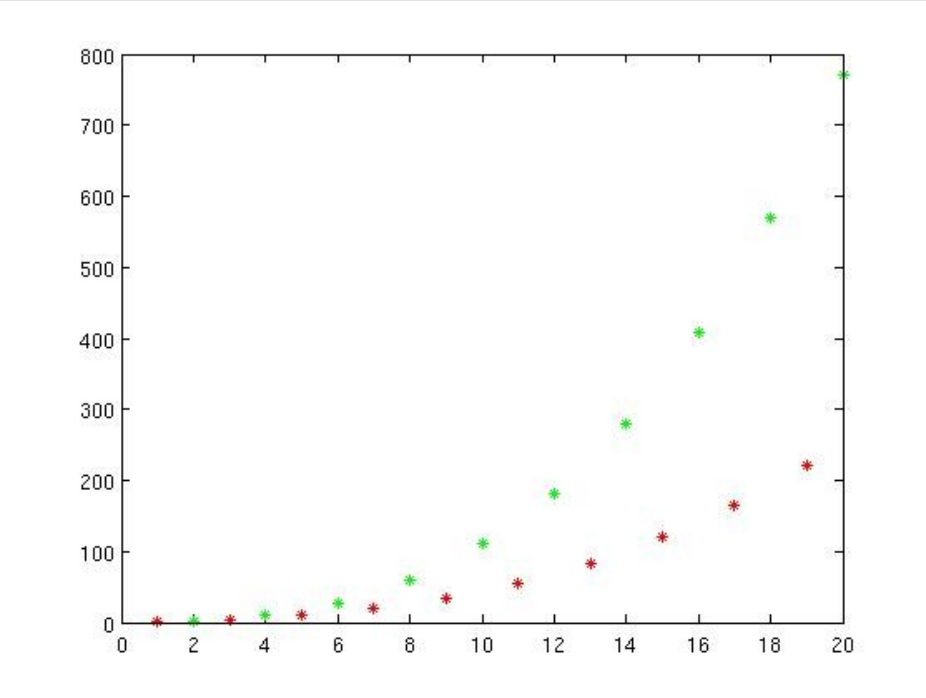
Fcc Si - k-points study

- load and visualize
  - the energy is converged for mesh size ~10 and more



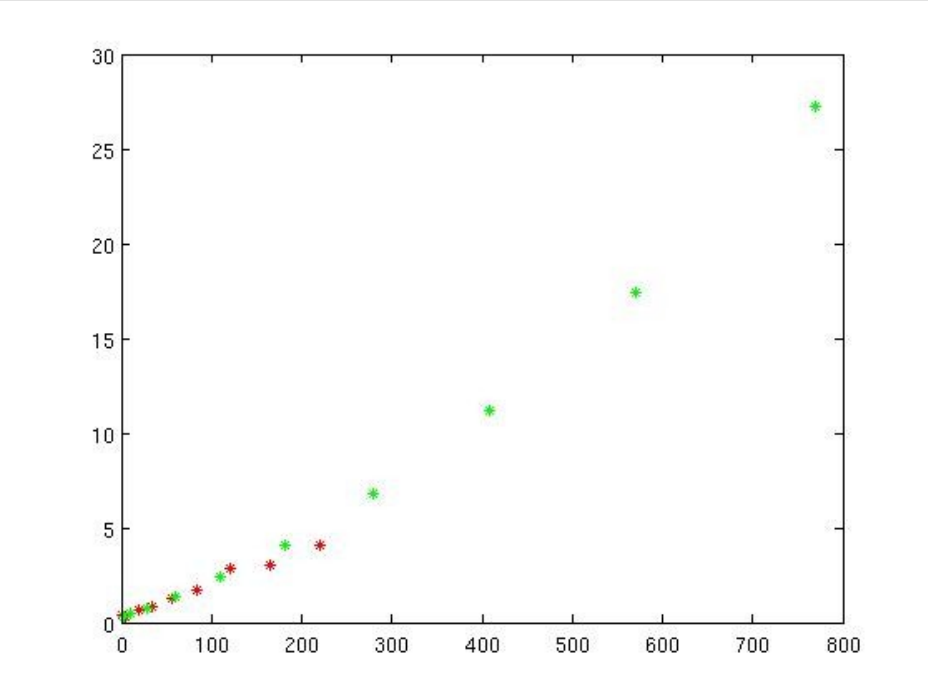
Fcc Si - k-points study

- load and visualize
  - the even k-point meshes generate in this case more k-points than comparable odd o-point meshes
  - e. g. the 8x8x8 mesh contains the same number of k-points as the 11x11x11 odd mesh => always check the IBZKPT file to see the actual k-points number



Fcc Si - k-points study

- load and visualize
  - the linear dependence until about 200 k-points observed
  - slightly different behavior of odd and even grids



Outline

- Fcc Si
  - lattice relaxation
  - k-points study
- Diamond Si
  - DOS and band-structure calculation – insulator/semiconductor/metal

Density of states - DOS

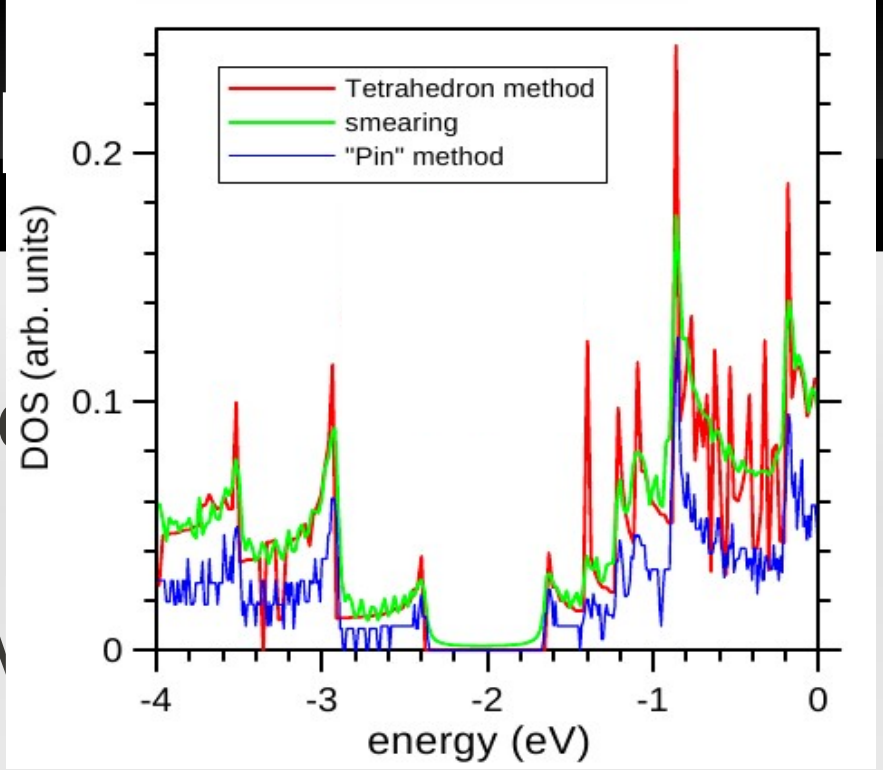
$$g(E) = \frac{1}{dE} \sum_{E}^{E+dE} 1$$

- the number of energy levels between E and E + dE divided by the dE

$$g(E) = \sum_{k} \delta(E - E(k))$$

- in periodic system sum over all possible k-point values

Density of states -



- the number of energy leve
  dE divided by the dE
- in periodic system sum ov
  values
- in practice finite k-point mesh and dE set used
  - DOS vs. E plot is “spiky”
  - smearing of the spikes into smooth functions like gaussian (see smearing methods)
  - tetrahedron method (ISMEAR = -5) – interpolation between k-points
  - finer k-point grid needed

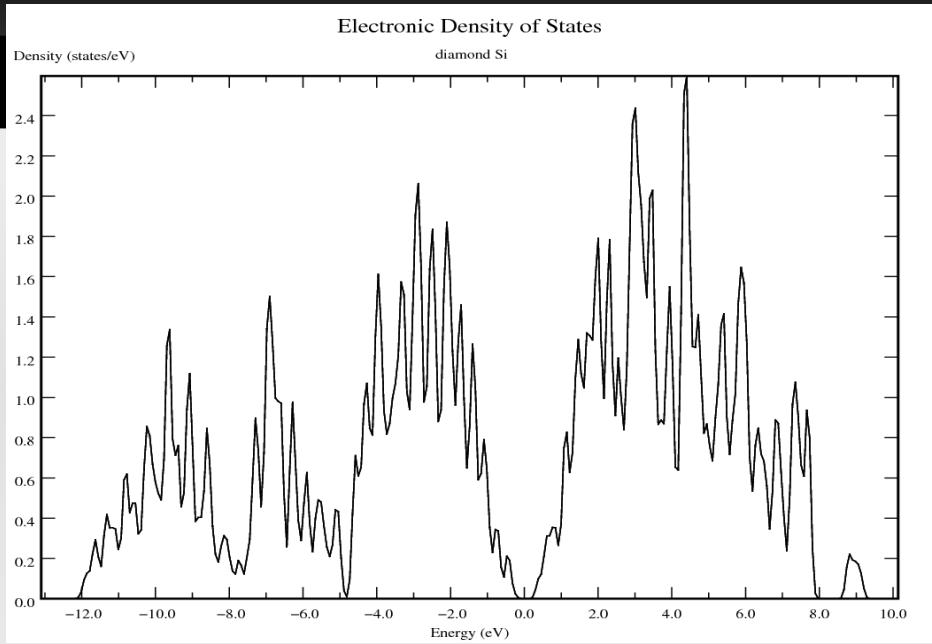
Diamond Si

- `cd cviceni_2/si_diamond/ex_1`
- POTCAR, INCAR and KPOINTS from si_fcc/ex_1
- POSCAR

```
diamond Si
   5.5
    0.0    0.5    0.5
    0.5    0.0    0.5
    0.5    0.5    0.0
   Si
   2
Direct
 -0.125 -0.125 -0.125
  0.125  0.125  0.125
```

- run VASP

Diamond Si - DOS



- DOS – directly from the last run
  - provided in DOSCAR file
  - use the visualization tool (p4vasp)
  - not accurate enough!
  - finer k-point mesh size needed => time consuming
    - two step calculation:
    - 1. self-consistent calculation with smaller number of k-points (already done)
    - 2a. increase of the number of k-points
    - 2b. while charge density converges fast with the number of k-points => non-self consistent calculation possible using the charge density from previous self-consistent run. The charge density is fixed. The k-points are treated independetly.
    - 2c. controlled by ICHARG keyword (ICHARG = 11)

Diamond Si - DOS

ICHARG: determines how to construct initial charge density
0=calculate from initial wave-function (default if WAVECAR present)
1=read the density from CHGCAR
2=superposition of atomic densities (default if no WAVECAR present)
11=non-selfconsistent calculation – read in CHGCAR and do not update

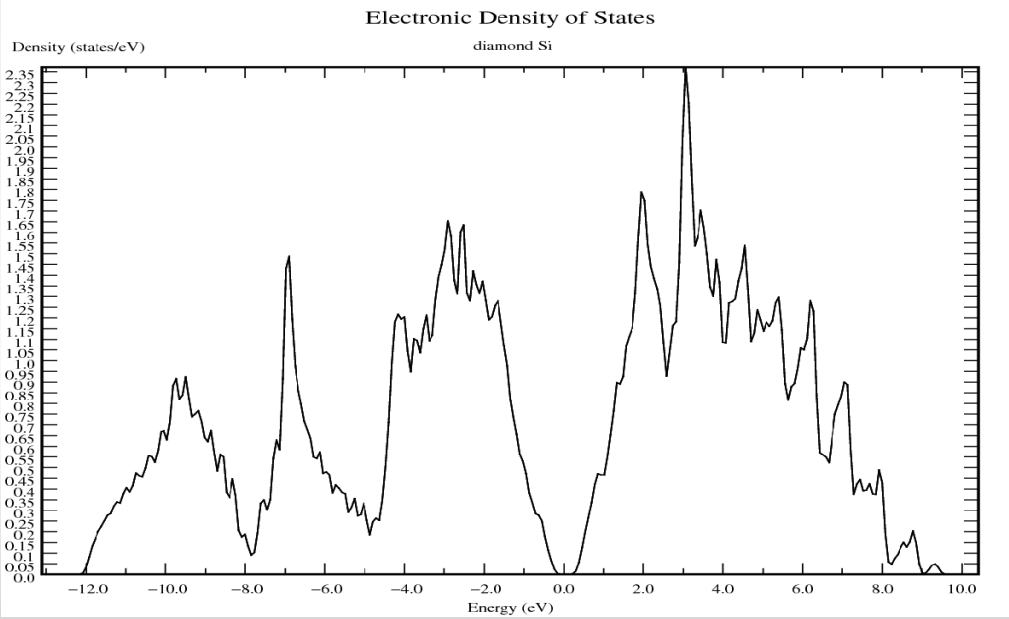
- DOS – directly from the last run
  - provided in DOSCAR file
  - use the visualization tool (p4vasp)
  - not accurate enough!
  - finer k-point mesh size needed => time consuming
    - two step calculation:
    - 1. self-consistent calculation with smaller number of k-points (already done)
    - 2a. increase of the number of k-points
    - 2b. while charge density converges fast with the number of k-points => non-self consistent calculation possible using the charge density from previous self-consistent run. The charge density is fixed. The k-points are treated independetly.
    - 2c. controlled by ICHARG keyword (ICHARG = 11)

Diamond Si - DOS

- `cd cviceni_2/si_diamond/ex_2`
- POTCAR, POSCAR from si_diamond/ex_1
- INCAR

```
SYSTEM = Si diamond          !title (optional)
ISMEAR = -5                  !linear tetrahedron method
SIGMA = 0.1
ENCUT = 240
ICHARG = 11                  !read charge file and do not update
```

  - besides finer k-point mesh also usage of linear tetrahedron method to smooth the DOS



Diamond Si - DOS

- `cd cviceni_2/si_diamond/ex_2`
- POTCAR, POSCAR from si_diamond/ex_1
- INCAR

```
SYSTEM = Si diamond          !title (optional)
ISMEAR = -5                  !linear tetrahedron method
SIGMA = 0.1
ENCUT = 240
ICHARG = 11                  !read charge file and do not update
```

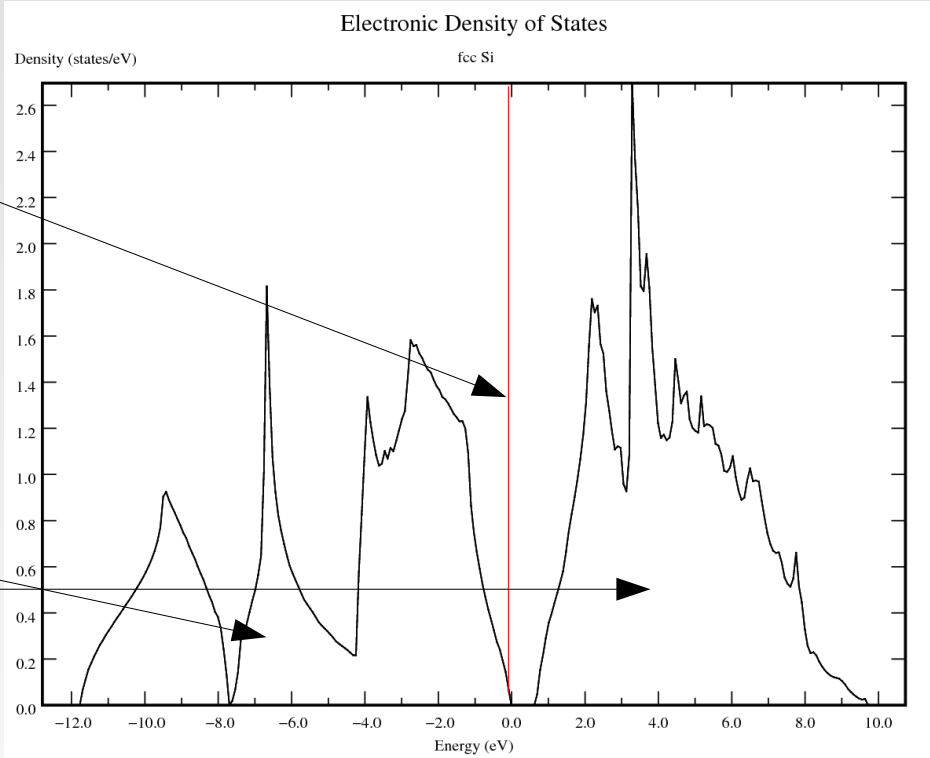
  - besides finer k-point mesh also usage of linear tetrehedron method to smooth the DOS
- KPOINTS

```
Automatic mesh
0                   !automatic generation ('zero')
Monkhorst Pack      !method of automatic generatioin
21 21 21            !num. of subdivisions along reciprocal vectors
  0   0  0          !shift of the k-point mesh
```

- run VASP

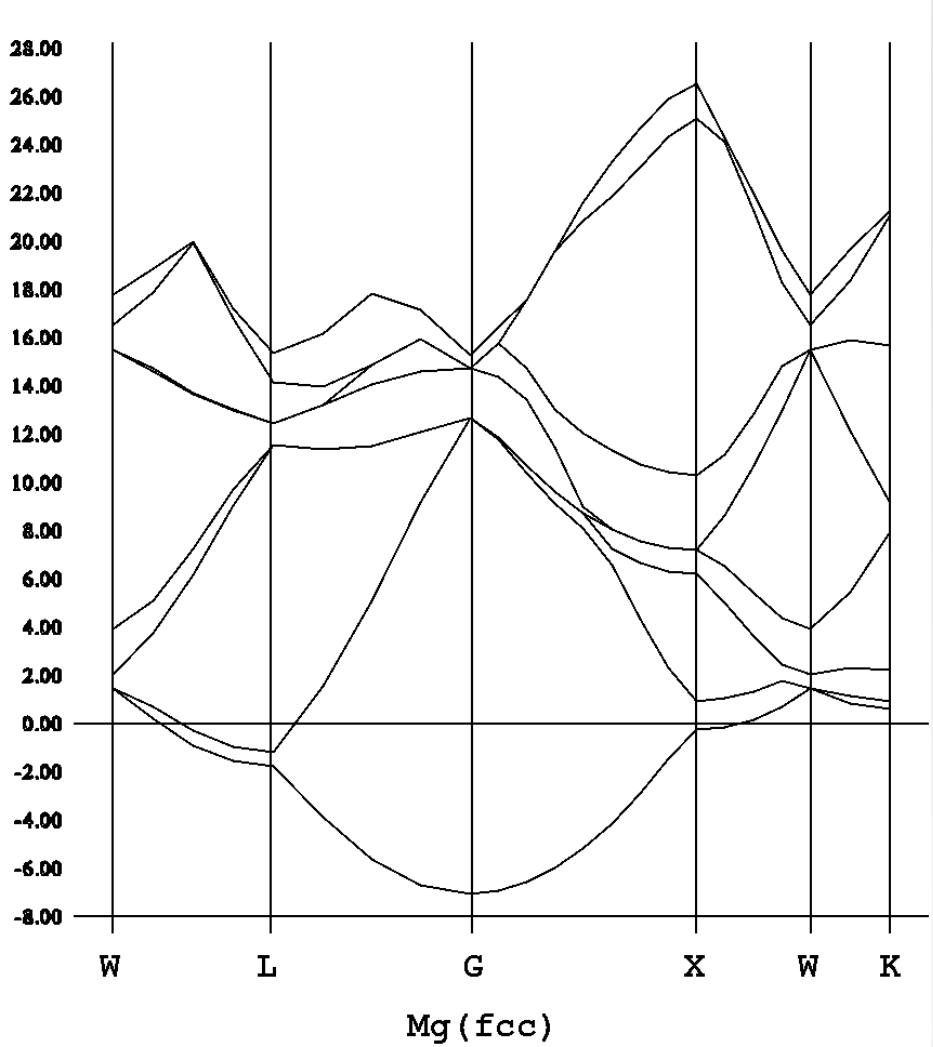
Diamond Si - DOS

- visualize with p4vasp
- smooth DOS observed
- a small gap (~0.6 eV) present on the Fermi level – semiconductor
- experimental band gap ~1.12
- valence bands
- conduction bands



Diamond Si – Band structure

- Band structure
  - energy vs. k-point plot
  - 3D data (k is a vector) – need for a simplification



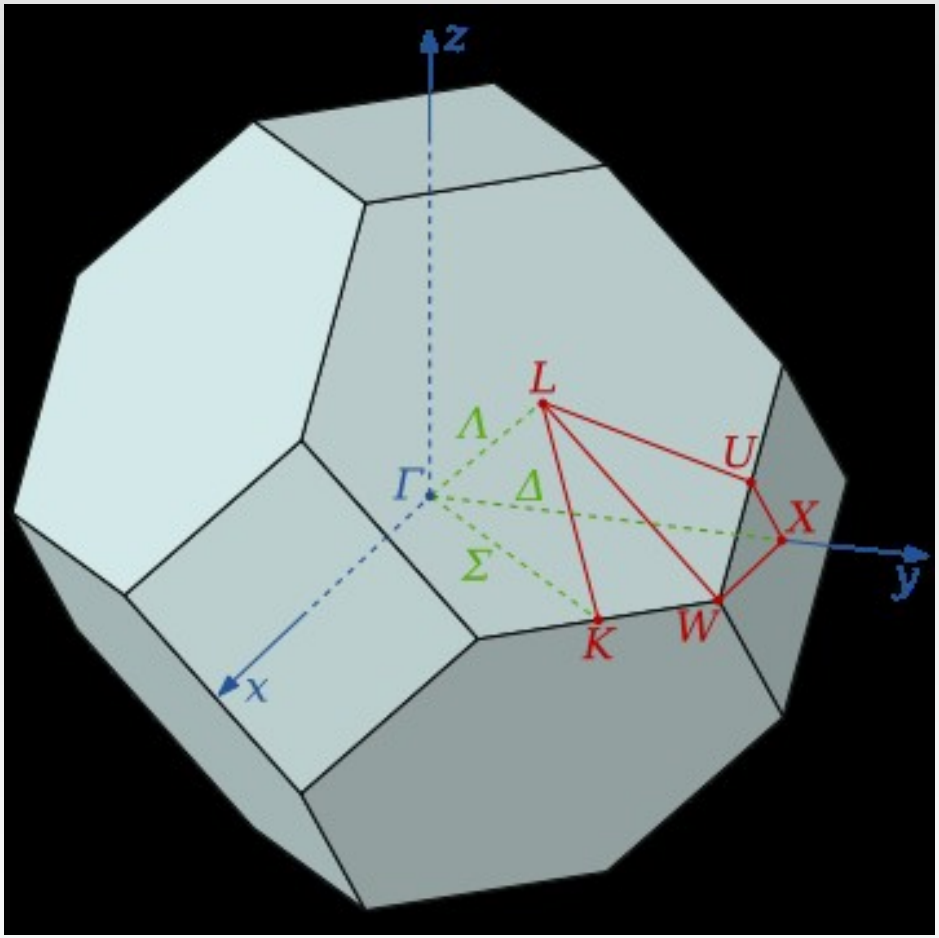
Diamond Si – Band structure

- Band structure
  - energy vs. k-point plot
  - 3D data (k is a vector) – need for a simplification
  - the plot along specific lines within Brillouin zone(BZ)

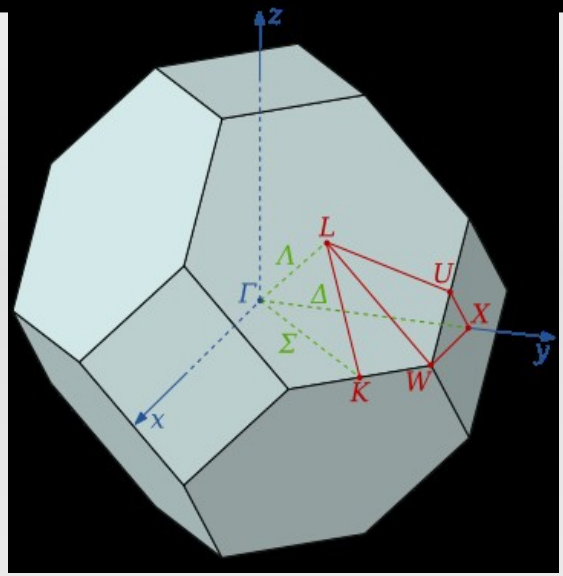
    Γ – gamma point – center of BZ

    X – center of face

    L – center of hexagonal face ...
  - energies(eigenvalues) for various k-points stored in EIGENVAL file



Diamond Si – Band structure



- `cd cviceni_2/si_diamond/ex_3`
- POTCAR, POSCAR from si_diamond/ex_1
- INCAR

```
SYSTEM = fcc Si
ICHARG = 11
ENCUT = 240
ISMEAR = 0; SIGMA=0.1
```

  - use ICHARG = 11 – the k-point mesh is not regular – need of self-consistent density with regular mesh beforehand
- KPOINTS
  - 3rd line with tag 'line'
- Run VASP

```
k-points for bandstructure L-G-X-U K-G
10                      !number of points between k-point couples
line
reciprocal
0.5 0.5 0.5 1           !L
0.0 0.0 0.0 1           !G

0.0 0.0 0.0 1           !G
0.0 0.5 0.5 1           !X

0.0 0.5 0.5 1           !X
0.25 0.625 0.625 1      !U

0.375 0.75 0.375 1      !K
0.0 0.0 0.0 1           !G
```

Diamond Si – Band structure

Energies(eigenvalues) for various k-points are stored in EIGENVAL file

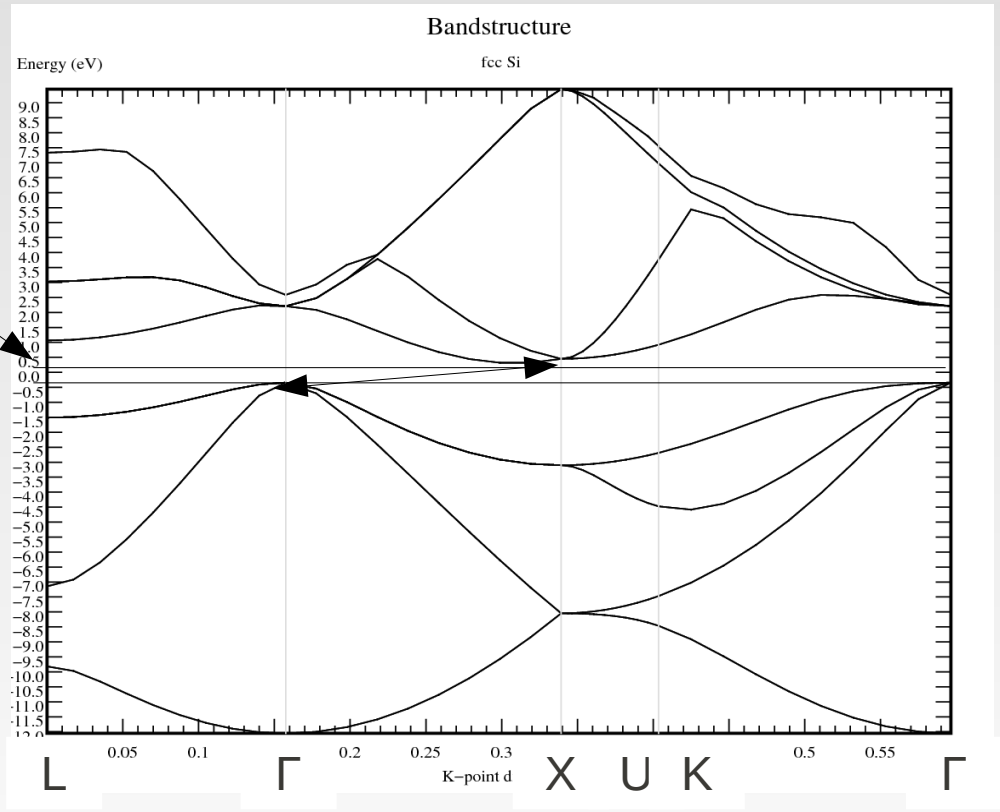
- Band structure
  - energy vs. k-point plot
  - forbidden band
  - indirect gap
  - conduction and valence s(p) bands
  - the plot along specific lines within Brillouin zone(BZ)

    Γ – gamma point – center of BZ

    X – center of face

    L – center of hexagonal face

    etc.

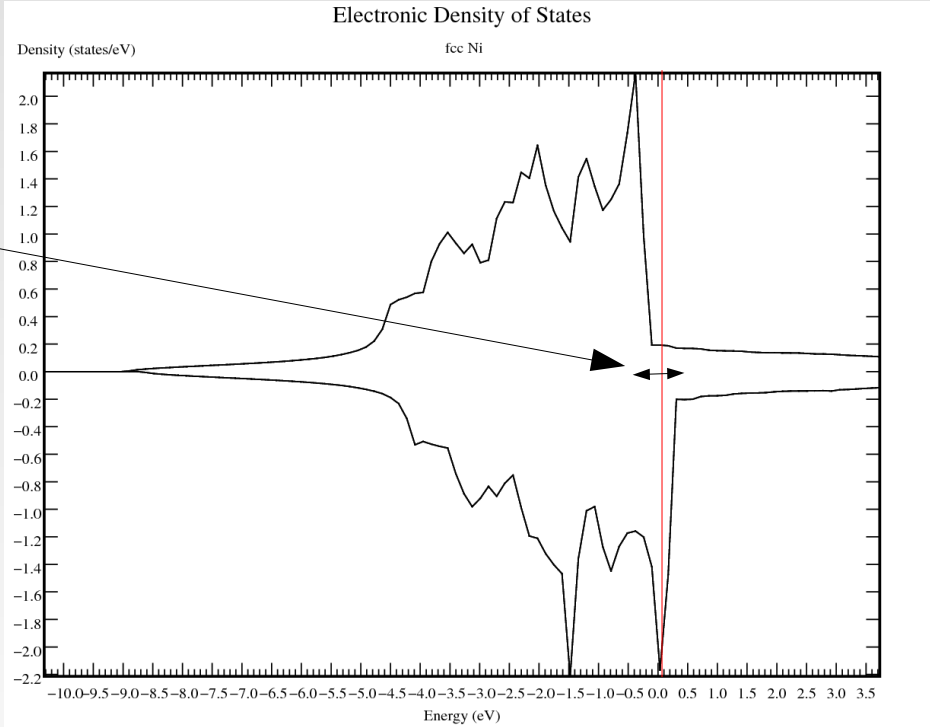


Outline

- Lattice parameter
  - fcc Si
- DOS and band-structure calcualtion – insulator/semiconductor/metal
  - Diamond Si
  - fcc Ni - metal

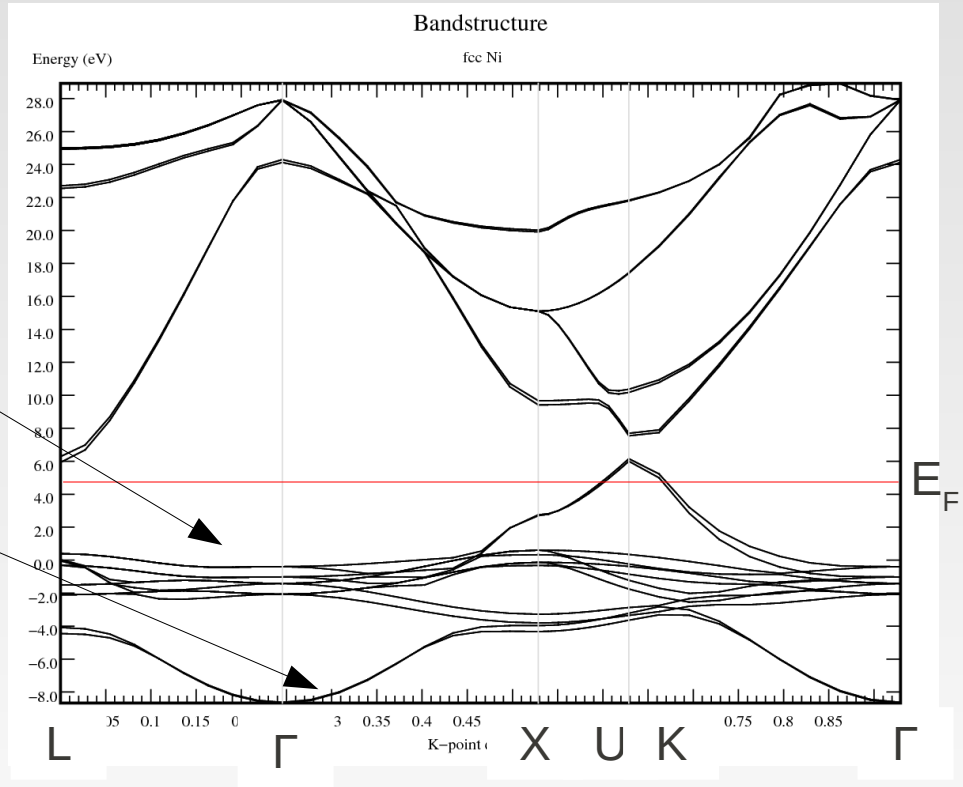
fcc Ni - DOS

- calculate like in diamond Si case (optional)
- DOS for the spin up and spin down
- exchange splitting on the Fermi level
- no band gap present on the Fermi level – metal



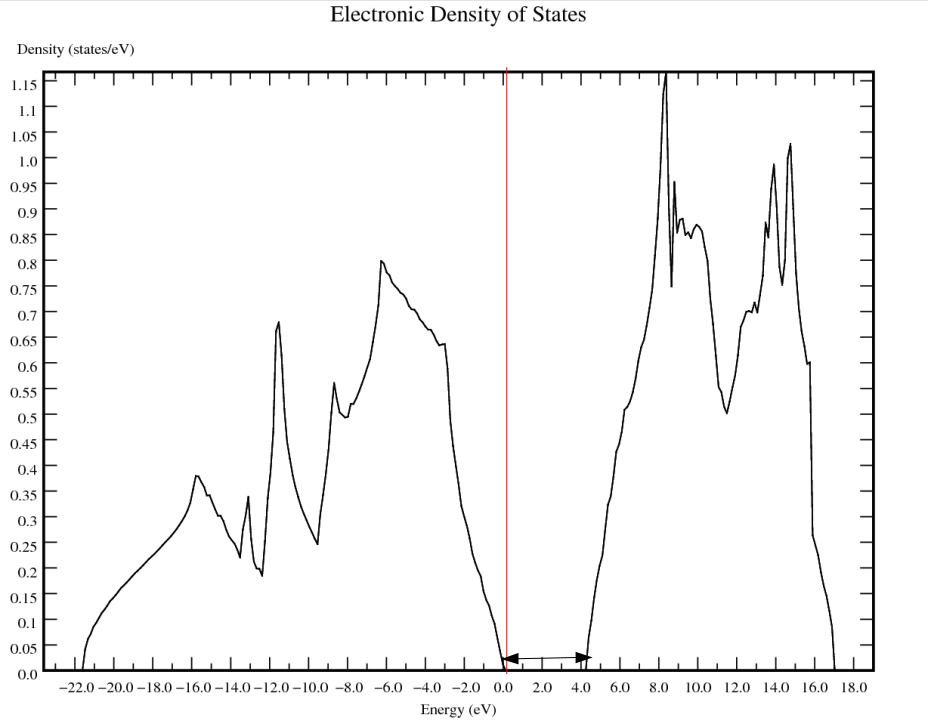
Fcc Ni – Band structure

- Band structure
  - energy vs. k-point plot
  - no gap at Fermi level
  - s(p) and d bands
  - Fermi level – 4,67 eV



Diamond C – insulator - DOS

- calculate like in diamond Si case (optional)
- large band gap (~4 eV) – exp. ~5.5 eV



Diamond Si – Band structure

- Band structure
  - energy vs. k-point plot
  - large band gap
  - s(p) bands

Energies(eigenvalues) for various k-points are stored in EIGENVAL file