siesta
TM

# Converge calculations:
## Mesh, k-points and SCF convergence

**Dr. Julio Gutiérrez Moreno**
Barcelona Supercomputing Center

19th November 2025

ICN2 10 YEARS
Institut Català de Nanociència i Nanotecnologia

BSC Barcelona Supercomputing Center
Centro Nacional de Supercomputación

CSIC
Consejo Superior de Investigaciones Científicas

MAX DRIVING THE EXASCALE TRANSITION

Ministerio de Ciencia, Innovación y Universidades

Financiado por la Unión Europea NextGenerationEU

Plan de Recuperación, Transformación y Resiliencia

AGENCIA ESTATAL DE INVESTIGACIÓN

# Convergence in DFT

- **Systematically reducing numerical errors** (by increasing computational sampling) **until they become "acceptable".**

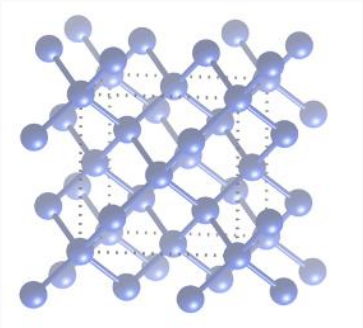
- Convergence is a critical step in computer simulations.
  Without proper convergence, results could lead to incorrect conclusions.
  Convergence is essential for publishing accurate and reproducible data.

# The Real Space Grid in SIESTA

# Sampling

## Real space

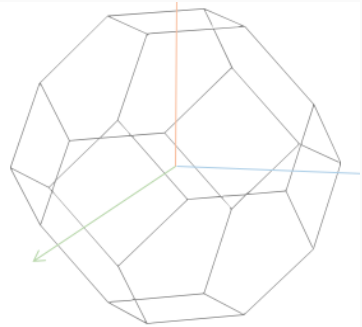
- **Potentials**
- **Densities**
- **Basis**



Real-space mesh for localized orbitals. The mesh density must be high enough to accurately represent the electronic density and potentials.

## Reciprocal space

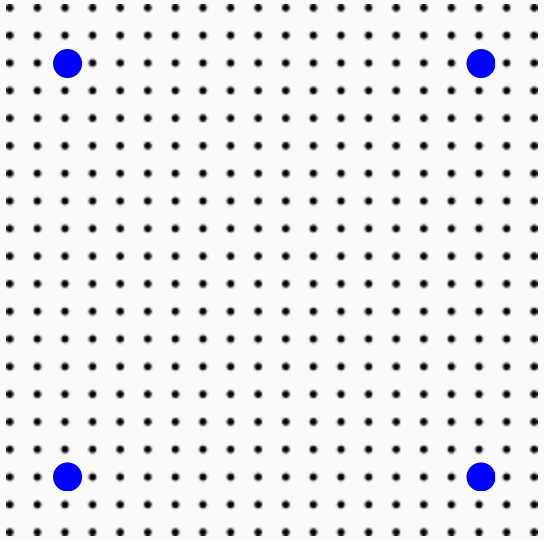
- **Density of states**
- **Band Structure**

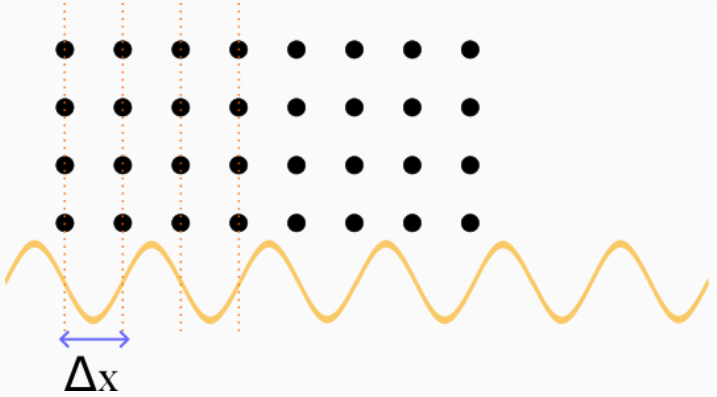


Reciprocal-space grid for sampling the Brillouin Zone of the periodic structure.

**Both grids need independent convergence tests.**

# The Real Space grid: mesh cutoff






$$\Delta x \rightarrow k_c = \frac{\pi}{\Delta x} \rightarrow E_c = \frac{\hbar^2 k_c^2}{2m_e}$$

Fineness ↔ Maximum energy avoiding aliasing

$$\Delta x \leftrightarrow E_{Cutoff}$$

MeshCutoff

Energy units (Ry)

# Real space grid: MeshCutoff

- What is it set by the user?

  Mesh.Cutoff  300 Ry  (default)

```
Mesh.Cutoff  100 Ry
```

- What is set by siesta?

  MESH = 18 x 18 x 30 = 9720

  Mesh cutoff (required, used) =  100.000  101.039 Ry

- How can one decide the good value?

  Minimize the total energy.

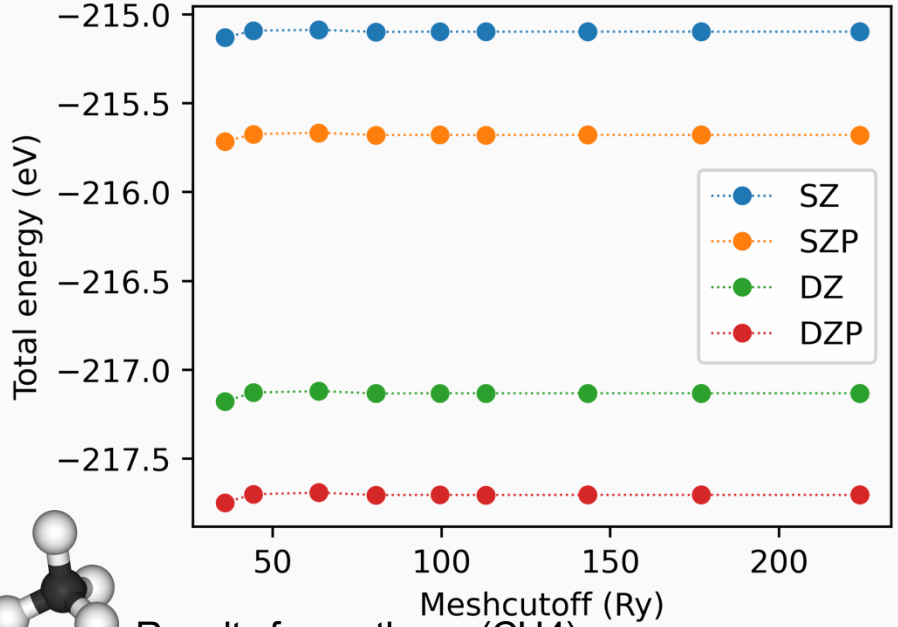
  Total (equilibrium) force to zero.

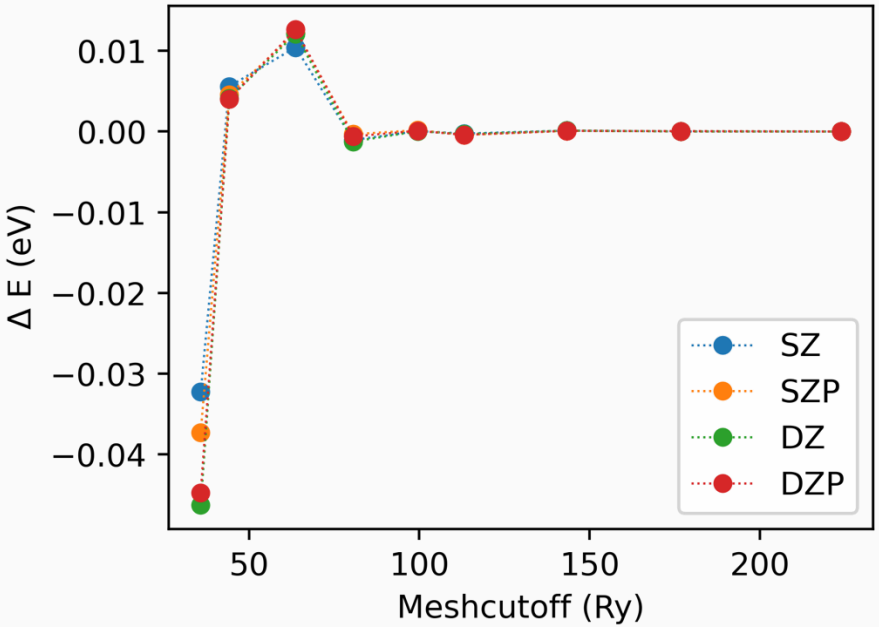
  Reasonable time (relatively small systems)

# Real space grid: MeshCutoff

## Energy

```
siesta: Final energy (eV):
```





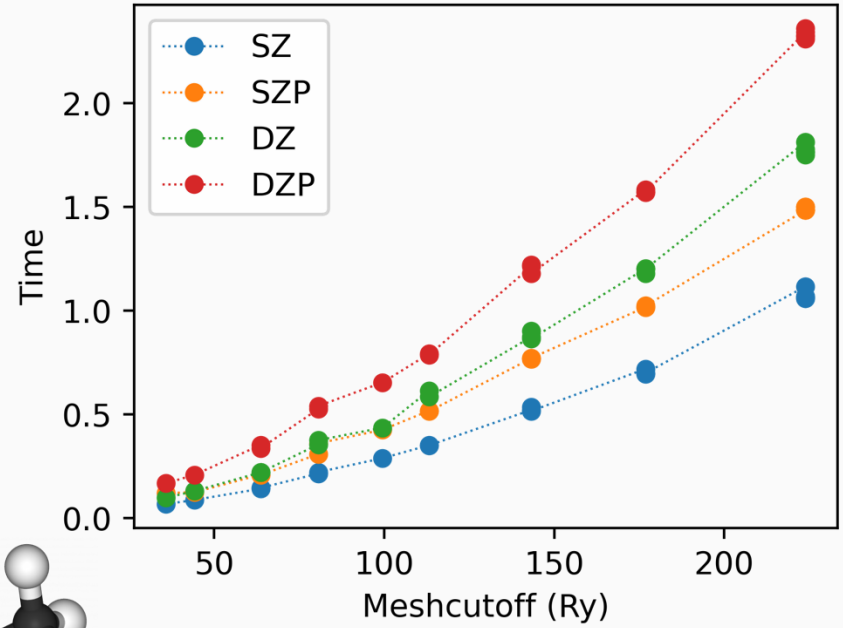

Results for methane (CH4)

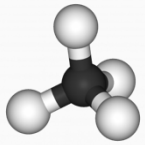
# Real space grid: MeshCutoff

## Time

`TIMES file`

**UseTreeTimer T**

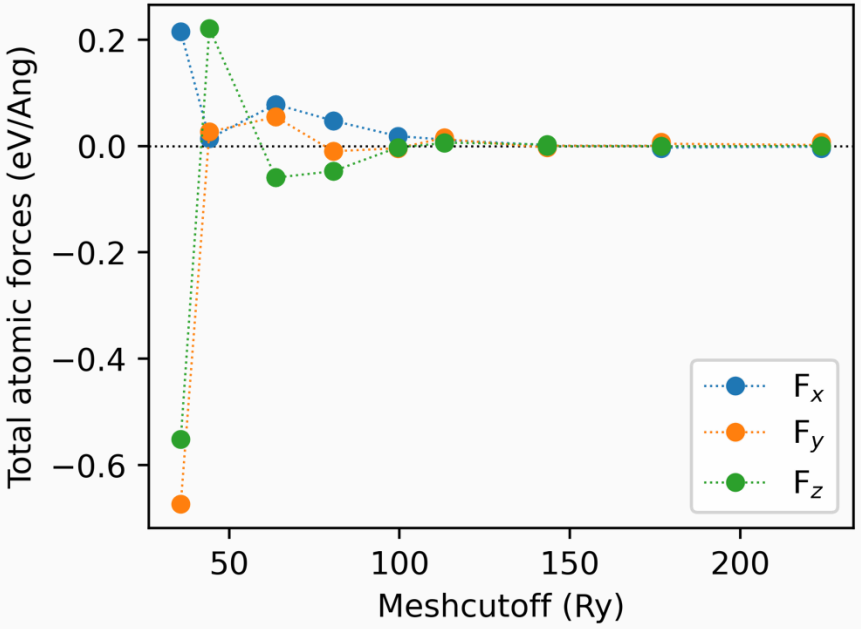


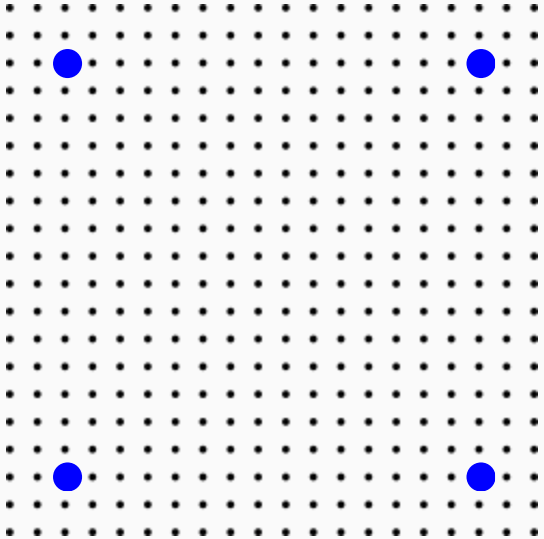
Results for methane (CH4)

## Force

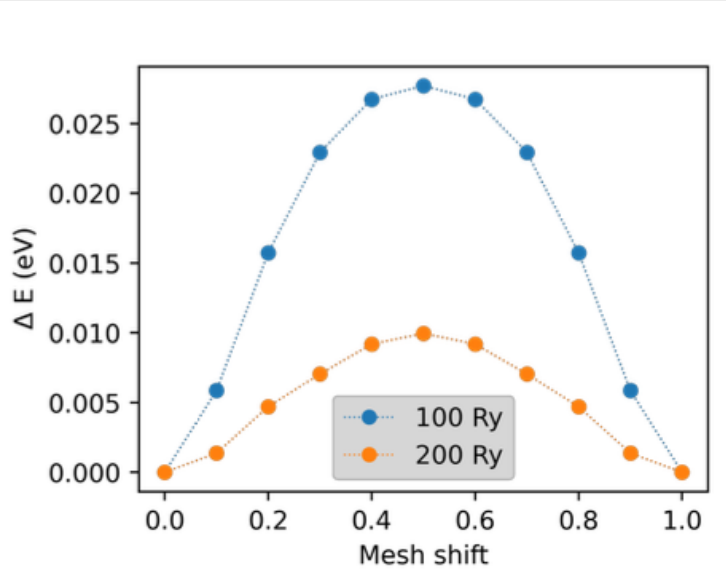
`siesta: Atomic forces (eV/Ang):`



# The Real Space grid: egg-box effect

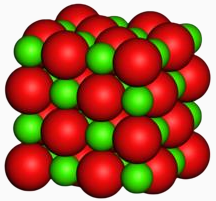


Invariant under any translation?




```
%block AtomicCoordinatesOrigin
 0.0  0.0  0.0
%endblock AtomicCoordinatesOrigin
```

$$\delta z_{shift} = \left(\frac{1}{M_z}\right)\frac{1}{10}$$



MgO

Solution:
- Increase Meshcutoff
- Use “grid-cell-sampling”

# Real space grid: Summary & friendly some recommendations

Mesh cutoff controls fineness of real-space grid: **must be tested for each system**

Increasing cutoff & monitor $E_{total}$ (and forces) to identify the converged values
Balanced **accuracy vs performance**: higher cutoff helps reducing numerical errors but increase computational cost

The egg-box effect occurs when the total energy oscillates due to the finite grid
Increase the mesh cutoff in steps

**Guidelines & Best Practices** (please take them carefully)
- Molecules: 100–150 Ry often sufficient
- Bulk solids: 150–250 Ry typical
- Heavy atoms, sharp densities: may require 300+ Ry
- For supercells: choose cutoffs that give grid sizes divisible by replication factors
- Converge energies (< 1 meV/atom) and forces (< 0.01 eV/Å)
- Only relax atoms (and move to more complex calculations) after cutoff is chosen

# Reciprocal Space

# Reciprocal Space

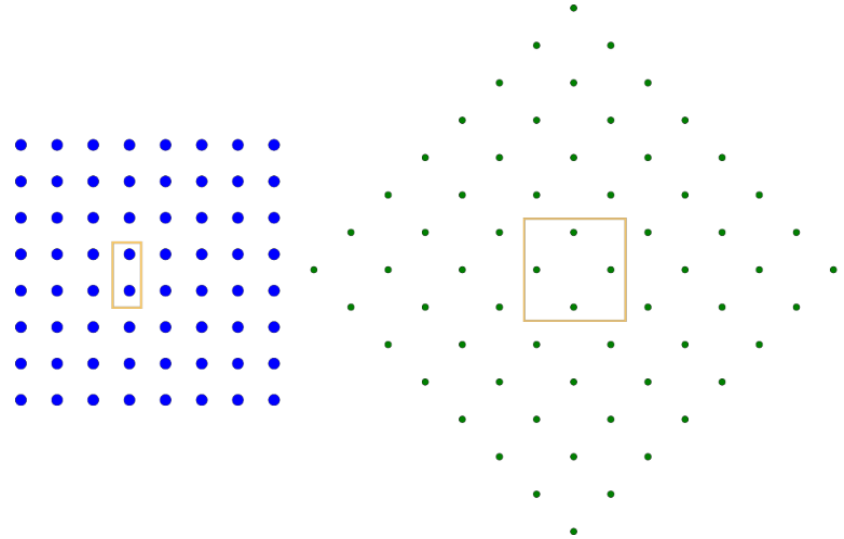
Crystals

$\psi(\mathbf{r})$
Infinite matrix

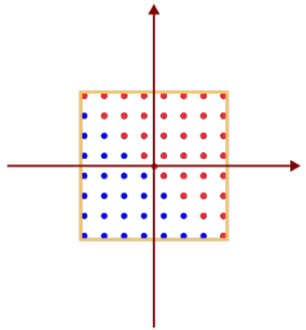
Periodicity

Reciprocal space
$\psi_{n,\mathbf{k}+\mathbf{G}}(\mathbf{r}) = \psi_{n,\mathbf{k}}(\mathbf{r})$

Finite matrix



$\psi_{up}(\mathbf{k}) = \psi_{down}(\mathbf{k})$



First Brillouin Zone

K-points are points in reciprocal space that represent electronic states in a periodic crystal. They are essential for accurate electronic structure calculations.

# Reciprocal Space Grid: k-mesh

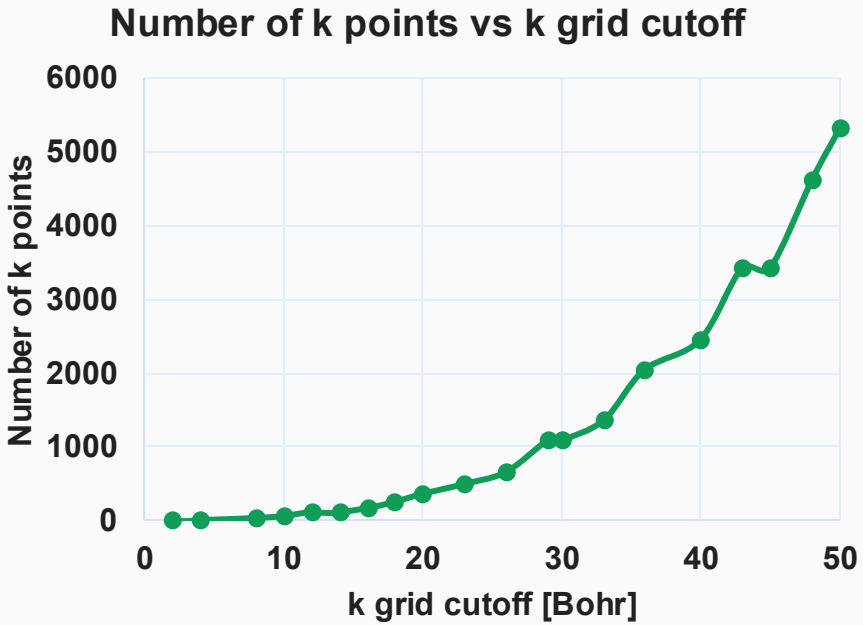
- What is it set by the user?
  - kgrid_cutoff
  - Monkhorst Pack grid
- What is set by siesta?
  - SystemLabel.KP
- How can one decide the good value?
  - Must consider the ratio between the lattice vectors.
  - Check:
    - DOS
    - Bandstructure
  - For metallic systems more k points will be needed.

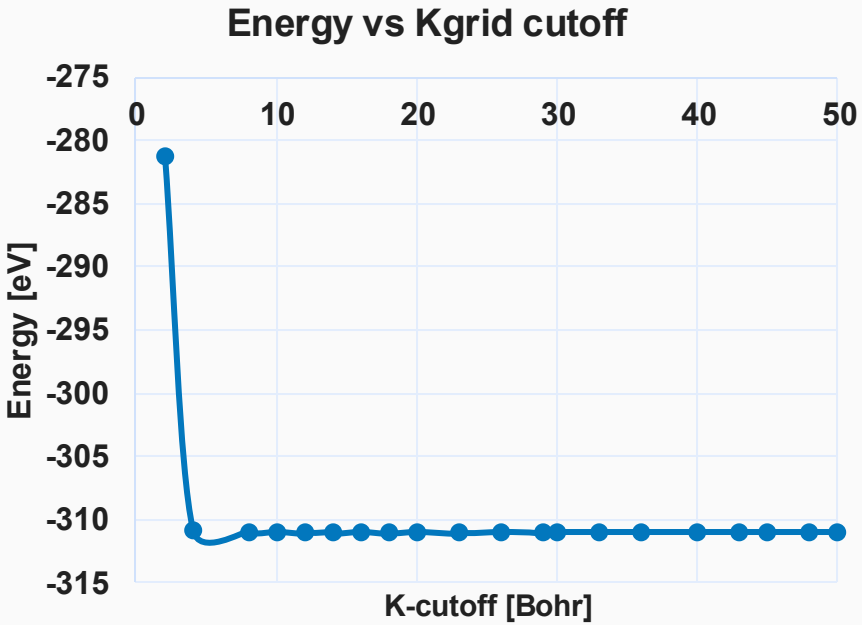
```
kgrid_cutoff          10.0 Ang

%block kgrid_Monkhorst_Pack
  6   0   0     0.0
  0   6   0     0.0
  0   0   1     0.0
%endblock kgrid_Monkhorst_Pack
```

SystemLabel.KP

```
 22
 1 -0.447497E+00 -0.258363E+00  0.000000E+00   0.555556E-01
 2 -0.223749E+00 -0.129181E+00  0.000000E+00   0.555556E-01
 3  0.000000E+00  0.000000E+00  0.000000E+00   0.277778E-01
 4  0.671246E+00  0.387544E+00  0.000000E+00   0.277778E-01
 5 -0.447497E+00  0.111022E-15  0.000000E+00   0.555556E-01
 6 -0.223749E+00  0.129181E+00  0.000000E+00   0.555556E-01
 7  0.000000E+00  0.258363E+00  0.000000E+00   0.555556E-01
 8  0.223749E+00  0.387544E+00  0.000000E+00   0.555556E-01
 9  0.447497E+00  0.516726E+00  0.000000E+00   0.555556E-01
10  0.671246E+00  0.645907E+00  0.000000E+00   0.555556E-01
11 -0.447497E+00  0.258363E+00  0.000000E+00   0.555556E-01
12 -0.223749E+00  0.387544E+00  0.000000E+00   0.555556E-01
13  0.000000E+00  0.516726E+00  0.000000E+00   0.555556E-01
14  0.223749E+00  0.645907E+00  0.000000E+00   0.555556E-01
15  0.447497E+00  0.775088E+00  0.000000E+00   0.555556E-01
16  0.671246E+00  0.904270E+00  0.000000E+00   0.555556E-01
17 -0.447497E+00  0.516726E+00  0.000000E+00   0.277778E-01
18 -0.223749E+00  0.645907E+00  0.000000E+00   0.277778E-01
19  0.000000E+00  0.775088E+00  0.000000E+00   0.277778E-01
20  0.223749E+00  0.904270E+00  0.000000E+00   0.277778E-01
21  0.447497E+00  0.103345E+01  0.000000E+00   0.277778E-01
22  0.671246E+00  0.116263E+01  0.000000E+00   0.277778E-01
```

# Reciprocal Space Grid: k sampling



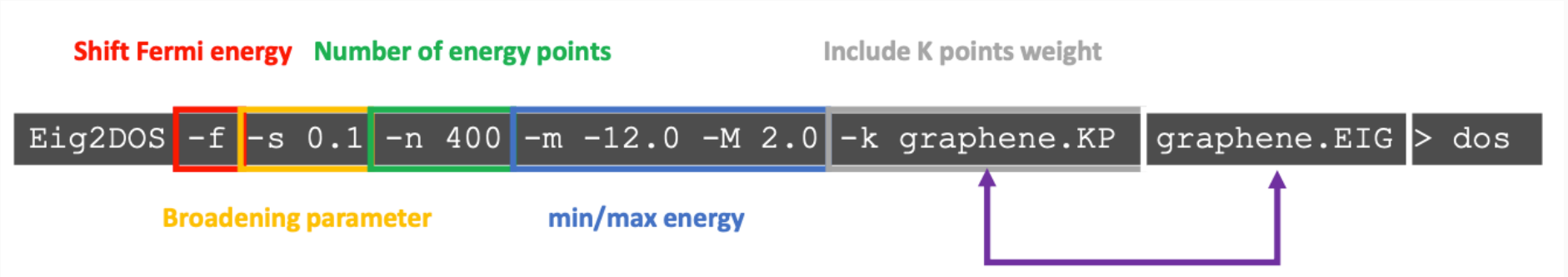




Results for diamond

# How do I plot PDOS & Band Structures

## A look at the SIESTA suite:

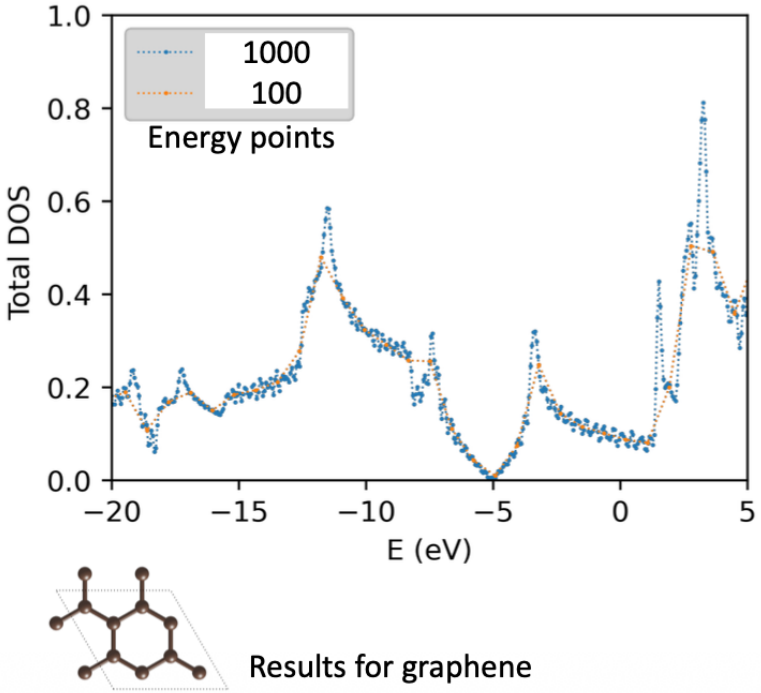
# Utilities & others: plotting DOS



# Utilities & others: plotting DOS

Eig2DOS output

```
 EIG2DOS: Utility for SIESTA to obtain the electronic density of states
#  E. Artacho, Apr 1999, A. Garcia, Apr 2012
#  Nick R. Papior, Feb 2017
# ----------------------------------------
# Eigenvalues calculated from a spin-polarized calculation
# Eigenvalues read from graphene.EIG
# Kpoint weights read from graphene.KP
# Using smearing parameter:    0.1000
# Using 400 points in the energy range
# Selected bands: 1 to: 26
# Emin, Emax in file for selected band(s):     -24.2236335     143.6658020
# Nbands, Nspin, Nk    =      26     2     5
# E_F                  =     -5.0301 eV --> (shifted to ZERO)
# Broadening           =      0.1000 eV
#
#         E             N(up)          N(down)         Ntot
    -12.000000       0.000000       0.000000       0.000000
    -11.964912       0.000000       0.000000       0.000000
    -11.929825       0.000000       0.000000       0.000000
    -11.894737       0.000000       0.000000       0.000000
    -11.859649       0.000000       0.000000       0.000000
    -11.824561       0.000000       0.000000       0.000000
    -11.789474       0.000000       0.000000       0.000000
    -11.754386       0.000000       0.000000       0.000000
    -11.719298       0.000000       0.000000       0.000000
    -11.684211       0.000000       0.000000       0.000000
    -11.649123       0.000000       0.000000       0.000000
    -11.614035       0.000000       0.000000       0.000000
    -11.578947       0.000000       0.000000       0.000000
    -11.543860       0.000000       0.000000       0.000000
    -11.508772       0.000000       0.000000       0.000000
    -11.473684       0.000000       0.000000       0.000000
    -11.438596       0.000000       0.000000       0.000000
    -11.403509       0.000000       0.000000       0.000000
    -11.368421       0.000000       0.000000       0.000000
    -11.333333       0.000000       0.000000       0.000000
    -11.298246       0.000000       0.000000       0.000000
    -11.263158       0.000000       0.000000       0.000000
    -11.228070       0.000000       0.000000       0.000000
    -11.192982       0.000000       0.000000       0.000000
    -11.157895       0.000000       0.000000       0.000000
    -11.122807       0.000000       0.000000       0.000000
    -11.087719       0.000000       0.000000       0.000000
    -11.052632       0.000000       0.000000       0.000000
    -11.017544       0.000000       0.000000       0.000000
    -10.982456       0.000000       0.000000       0.000000
    -10.947368       0.000000       0.000000       0.000000
    -10.912281       0.000000       0.000000       0.000001
```




Results for graphene

# Utilities & others: plotting band structure

Graphene bands

```
gnubands -F -G -o bands -E 10 -e -20 graphene.bands

gnuplot -e "set grid; set ylabel 'E - E_{F} (eV)'; set title 'Band structure of
graphene'; set terminal png; set output 'bandstructure.png'; set key noautotitle;
load 'bands.gplot'"
```

gnubands

```
$ gnubands -h
 Usage: gnubands [options] [bandsfile|PIPE]

   bandsfile   : SystemLabel.bands
        PIPE   : < SystemLabel.bands

 Options:

     -h         : print help
     -G         : print GNUplot commands for correct labels to stderr
                  Suggested usage: prog options 2> bands.gplot 1> bands.dat
                     gnubands [options] 1> bands.dat 2> bands.gplot
                  and then:
                     gnuplot -persist bands.gplot
     -s arg     : only plot selected spin bands [1,nspin]
     -F         : shift energy to Fermi-level
     -b arg     : first band to write
     -B arg     : last band to write
     -e arg     : minimum energy to write
                :   If -F set, will be with respect
                :   to Fermi level
     -E arg     : maximum energy to write
                :   Note, see -e
     -o file    : specify output file (instead of piping)
                : if used with -G a file name file.gplot will be created
```

# Utilities & others: plotting band structure

```
%block Bandlines
1    0.5000000000    0.000000000    0.0000    M
30    0.0000000000    0.000000000    0.0000    \Gamma
45    0.3333333333    0.333333333    0.0000    K
30    0.5000000000    0.500000000    0.0000    M
%endblock BandLines
```

→ SystemLabel.bands

```
-G      : print GNUplot commands for correct labels to stderr
          Suggested usage: prog options 2> bands.gplot 1> bands.dat
            gnubands [options] 1> bands.dat 2> bands.gplot
          and then:
            gnuplot -persist bands.gplot
  -s arg  : only plot selected spin bands [1,nspin]
  -F      : shift energy to Fermi-level
  -b arg  : first band to write
  -B arg  : last band to write
  -e arg  : minimum energy to write
          :  If -F set, will be with respect
          :  to Fermi level
  -E arg  : maximum energy to write
          :  Note, see -e
  -o file : specify output file (instead of piping)
          : if used with -G a file name file.gplot will be created
```

Shift Fermi energy

```
gnubands -F -G -o bandstructure -E 10 -e -20 SystemLabel.bands
```

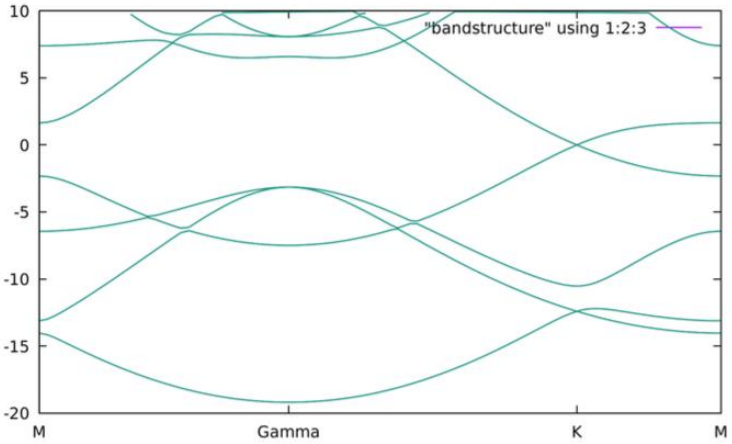
Shift Fermi energy

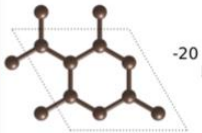
Min/max energy

# Utilities & others: plotting band structure

```
gnubands -F -G -o bandstructure -E 10 -e -20 *.bands

gnuplot -persist bandstructure.gplot
```

```
set xtics ("M"  0.000000, "Gamma"  0.775088, "K"  1.670083, "M"  2.117581)
plot "bandstructure" using 1:2:3 with lines lc variable
# -- Use line below for single-color#plot "bandstructure" with lines
```



Results for graphene

```
# GNUBANDS: Utility for SIESTA to transform bands output into
#Gnuplot format
#
# Emilio Artacho, Feb. 1999 # Alberto Garcia, May 2012
# Nick Papior, April 2013, July 2016
# ------------------------------------------------------------------------
# Bands for all spins
# E_F / orig = 0.0000 -5.0301
# k_min, k_max = 0.0000 2.1176
# E_min, E_max = -20.0000 10.0000
# Nbands, Nspin, Nk = 26 2 106
# Using min_band, max_band = 1 26
# Total number of bands = 26
#
# k E[eV]
# ------------------------------------------------------------------------
0.000000 -14.038730 1
0.025836 -14.158130 1
0.051673 -14.413430 1
0.077509 -14.707530 1
0.103345 -15.009230 1
0.129181 -15.308430 1
0.155018 -15.601030 1
0.180854 -15.885030 1
0.206690 -16.159330 1
0.232526 -16.423330 1
0.258363 -16.676430 1
0.284199 -16.918530 1
0.310035 -17.149230 1
```

# Reciprocal space: Summary & friendly some recommendations

Convergence k-point sampling is key for reliable periodic DFT results.

Use a single k-point (Γ-point) for isolated molecules, or very large systems.
Periodic systems require more k-points.
2D systems & slabs: sample in-plane only; kz=1 is fine if vacuum

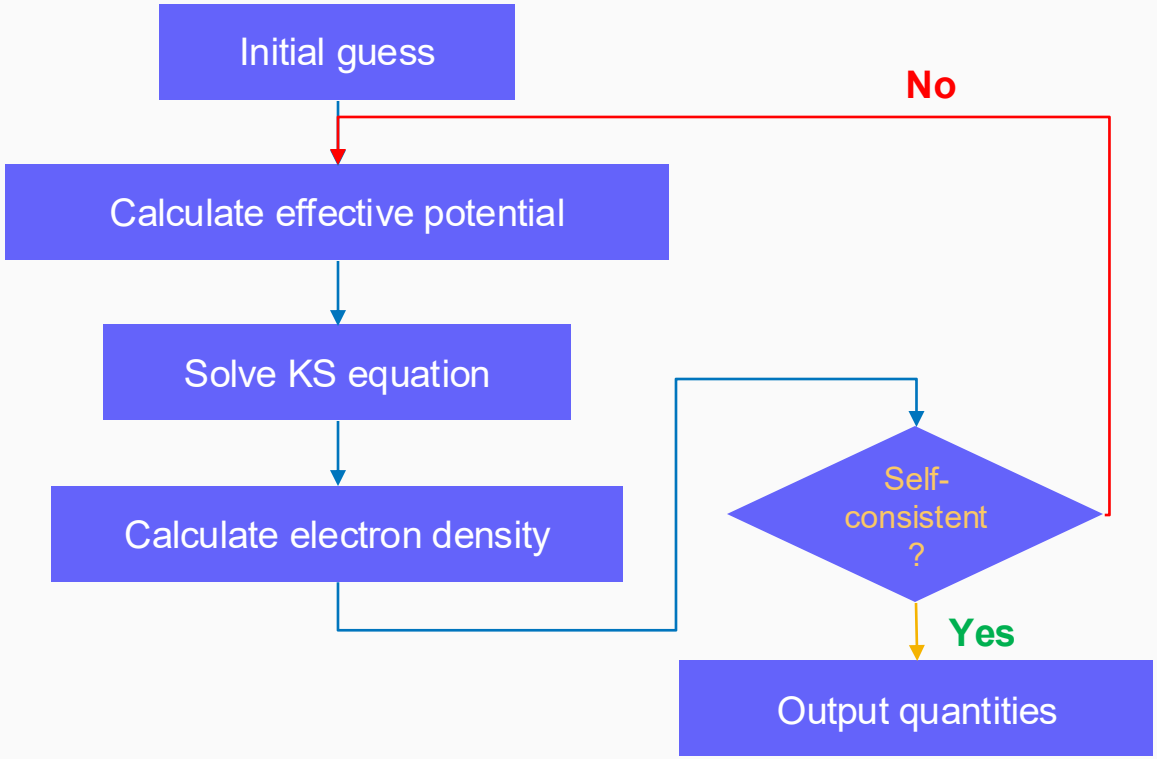
Always validate with energy, forces, DOS, and band structures.

**Guidelines and Best Practices** (once again, to be taken with extreme care)
- Insulators/semiconductors: start with 4×4×4 or 6×6×6 for a few-atom unit cell
- Metals: dense meshes needed (20×20×20 or higher) / electronic T can also help
- Always monitor Fermi level shifts.

- Spiky DOS: denser k-grid or larger broadening.
- Misaligned bands: check consistent k-paths.
- Fermi level jumps: increase grid, include high-symmetry points.
- Slow SCF: reuse density matrix, adjust mixing.
- High computational cost: test on smaller cells, transfer to full system.

# SCF Convergence

# SCF Convergence




**The physical quantity that is mixed:**

Density matrix

Hamiltonian matrix

**Mixing algorithm:**

Linear

Broyden } N previous steps

Pulay }

# SCF convergence

- SCF.Mix [default Hamiltonian]:
  - Density -> for systems hard to converge
  - Hamiltonian
- SCF.MixerMethod [default Pulay]
  - Linear
  - Pulay
  - Broyden
- SCF.Mixer.Weight [default 0.25]
  - 0.001 hard to converge systems ->lots of steps
  - 0.4 easier systems -> reduce steps
- SCF.Mixer.History [default 2]
- Max.SCF.Iterations [default 1000]
- SCF.DM.Converge F [default T]
- SCF.H.Converge F [default T]

**All of them strongly dependent on the system!**

SCF.Mix Hamiltonian
SCF.MixerMethod Pulay
SCF.Mixer.Weight 0.3

SCF.DM.Tolerance $10^{-4}$
SCF.H.Tolerance $10^{-3}$ eV

Max.SCF.Iterations 75

SCF.MixerMethod pulay
SCF.Mixer.Weight 0.2

SCF.Mixer.History 5

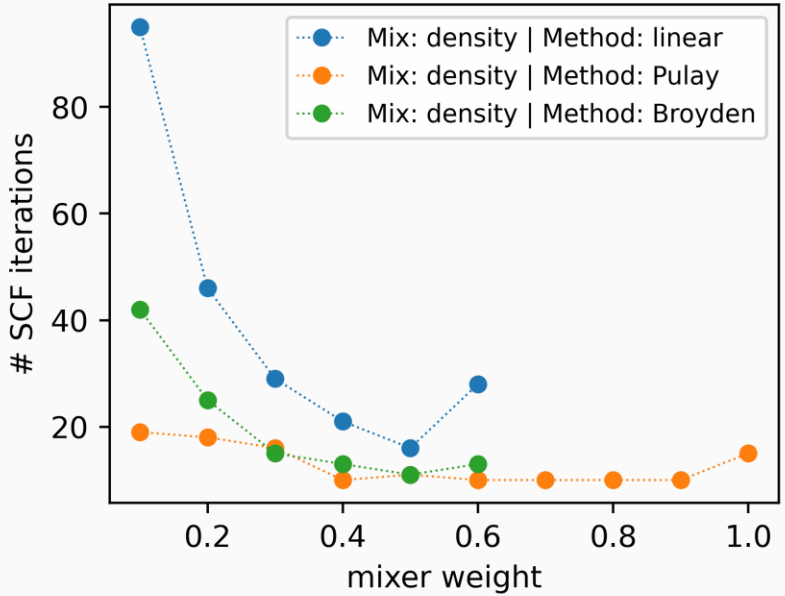
**More advanced options … (manual)**

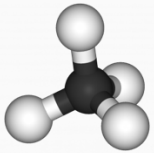
Oscillating $E_{total}$ could indicates poor mixing

# SCF convergence

```
SCF cycle converged
```

```
SCF_NOT_CONV:
```

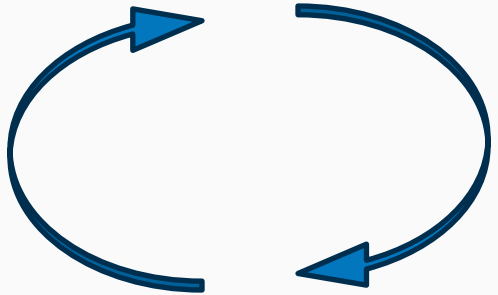


Results for methane (CH4)

# How do I converge the whole calculation?

1. Optimize the Basis set
2. Converge real space mesh: Energy
3. (Converge K grid: increase it for metallic systems)
4. SCF mixing



Let's play!

# Tutorials

This set of tutorials will guide you in the exploration of Siesta's features.

**Before you do anything else, start here.** You need to set up your local working environment to follow the tutorial.

- Setting up the local working environment for the tutorial exercises

## Basics of Siesta

This section is recommended for all beginners, and also as a refresher for more experienced users.

- A first encounter with Siesta
- First crystals
- Pseudopotentials
- Basis sets
- Basis set optimization
- The real-space grid
- Sampling of the BZ with k-points
- The self-consistent-field cycle
- Structural optimization using forces and stresses
- Vibration modes and phonons
- Magnetism

# How is the material organised?

my_workspace/siesta-docs/work-files/tutorials/basic/

**tutorials/basic/**

- grid_convergence/
  - 01-CH4
  - 02-MgO
- kpoint_convergence/
  - 01-graphene
  - 02-diamond
  - 03-graphite
- scf_convergence/
  - 01-CH4
  - 02-Fe_cluster

# Tutorial #1: **Real Space Grid convergence**

**OBJECTIVES**

Compare input vs generated mesh

Progressively increase the Cutoff in the $CH_4$ example, and plot:

- Total Energy (and ΔE, using highest cutoff as reference)
- Forces
- Computational time

Test egg-box effect in MgO unit cell

- Total Energy vs mesh shift
- Forces
- Computational time

# Tutorial #1: **Real Space Grid convergence**

## INPUTS FOLDER

**basic/grid_convergence**
├── 01-CH4/

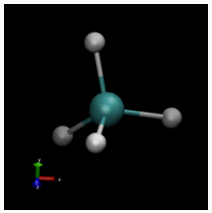
## OUTPUTS FOLDER

**siesta/**
├── exercise-day3-grid_convergence/
│ ├── create_your_own_folders

1. Copy all files from the input folder to the exercise folder.
2. In this exercise you will do a scan modifying the mesh cutoff value.

```
#Real space grid
MeshCutoff 125.0 Ry
```

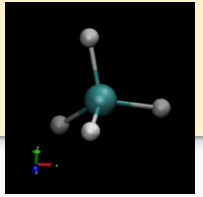
Mesh cutoff: 30 60 90 120 150 180 210 240 270 300

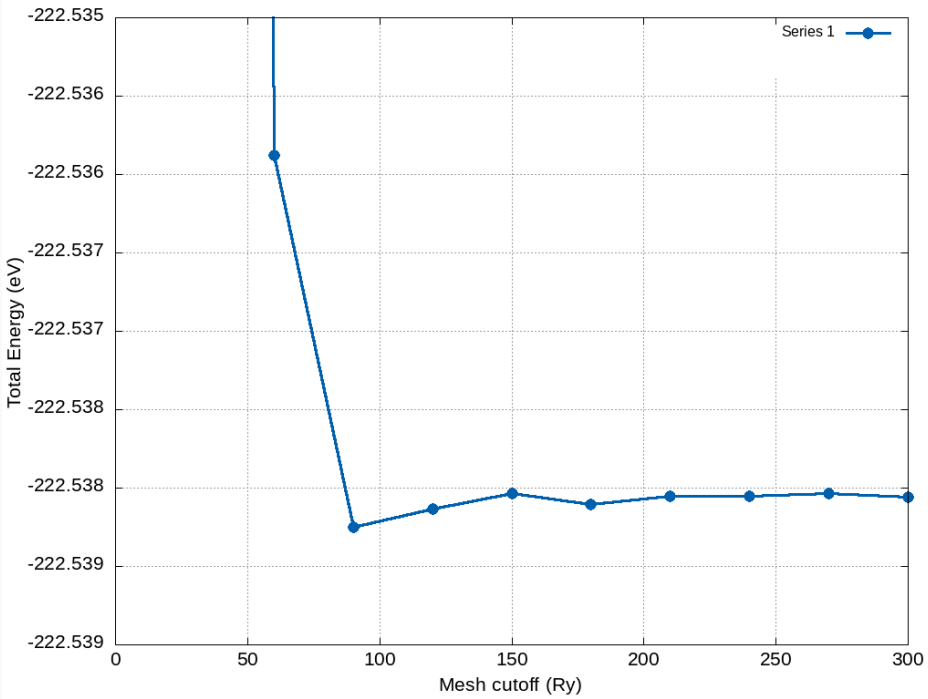


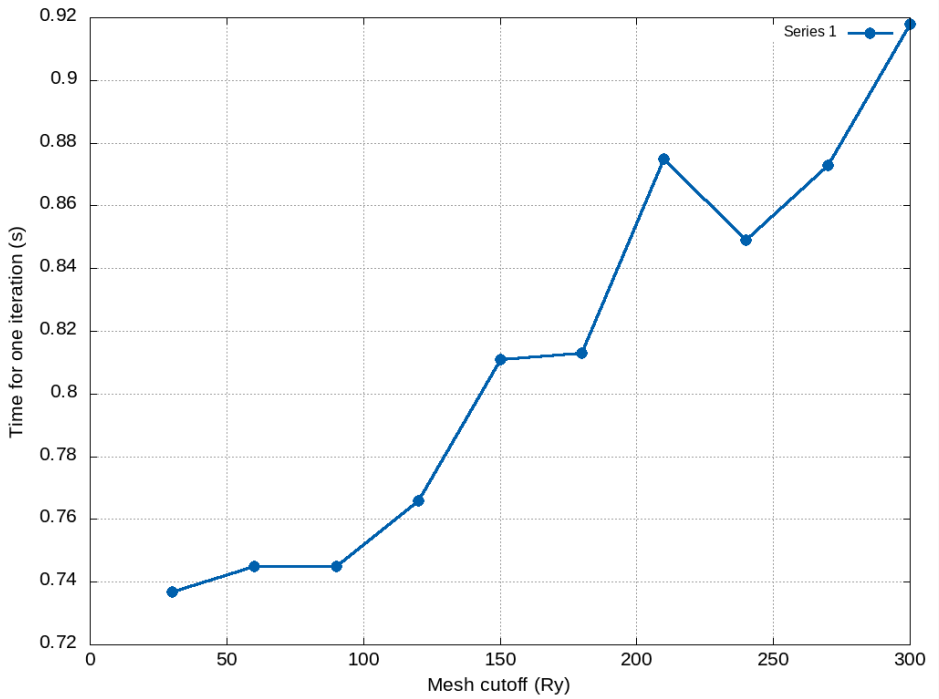
Everything depends on how you organize it

**siesta: Total = -221.815381**

# Tutorial #1: **Real Space Grid convergence**





# Tutorial #1: **Real Space Grid convergence**

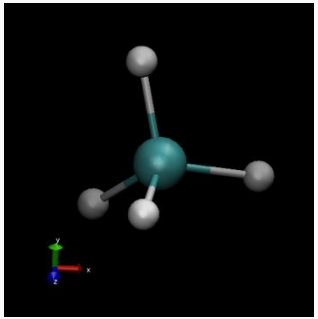
```
# Basis set definition
PAO.BasisSize    SZ
PAO.EnergyShift 250 meV
```

DZP

50-100 meV

```
#Real space grid
MeshCutoff 125.0 Ry
```

150 Ry



# Tutorial #2: **Reciprocal Space k-grid convergence**

**OBJECTIVES**

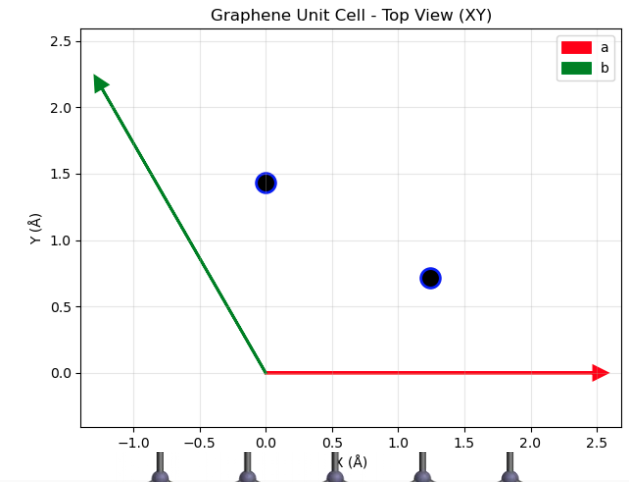
Check k-point convergence

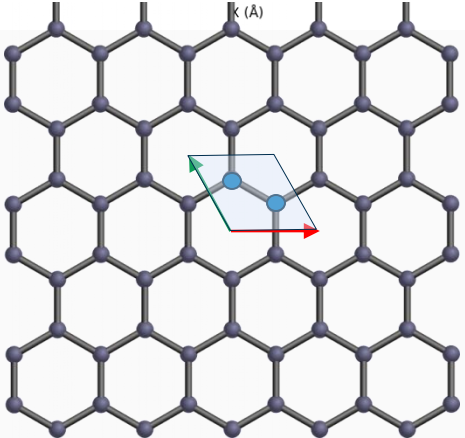
Plotting electronic structure features

- PDOS
- band structures

Try different materials: graphene, diamond, graphite

# Tutorial #2: **Reciprocal Space k-grid convergence**



```
#Atomic coordinates
AtomicCoordinatesFormat fractional

%block AtomicCoordinatesAndAtomicSpecies
  0.33333333  0.66666667  0.5        1     # C
  0.66666667  0.33333333  0.5        1     # C
%endblock AtomicCoordinatesAndAtomicSpecies

#Unit cell for the calculation
LatticeConstant 2.476675 Ang
%block LatticeParameters
1.000  1.000  10.0  90.0  90.0  120.0
%endblock LatticeParameters
```

# Tutorial #2: **Reciprocal Space k-grid convergence**

## INPUTS FOLDER

**siesta/kpoint-convergence**

```
├── 01-graphene/
```

## OUTPUTS FOLDER

**siesta/**

```
├── exercise_day3/
│   ├── create_your_own_folders
```

1. Copy all files from the input folder to the exercise folder.
2. In this exercise you will do a scan modifying the Monkhorst_pack block.

```
# K-sampling
%block kgrid_Monkhorst_Pack
   0  0    0.0
   0  0    0.0
   0  0  1  0.0
%endblock kgrid_Monkhorst_Pack
```

Monkhorst block: 1, 2, 3, 4, …

**siesta: k-grid: Number of k-points = 22**

| | | |
|---|---|---|
| Y | 0 | 0 |
| 0 | Y | 0 |
| 0 | 0 | 1 |

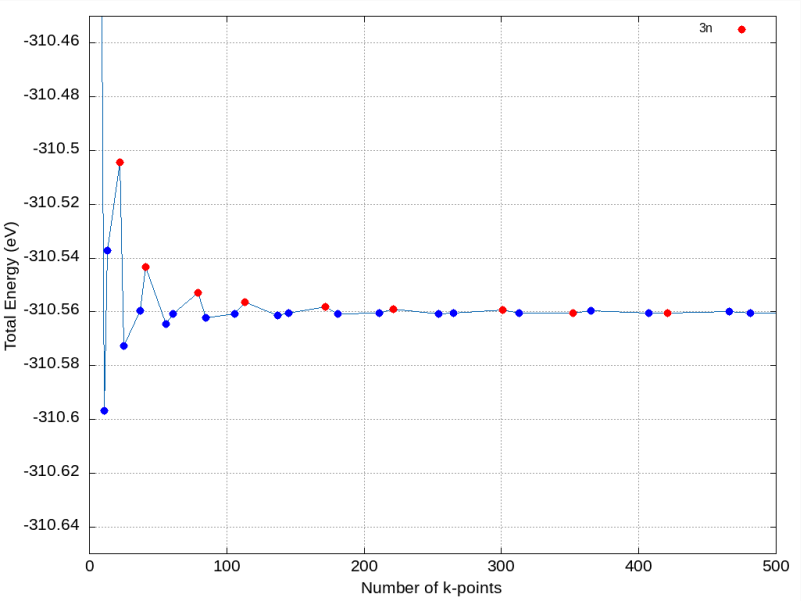
Using: Y=1

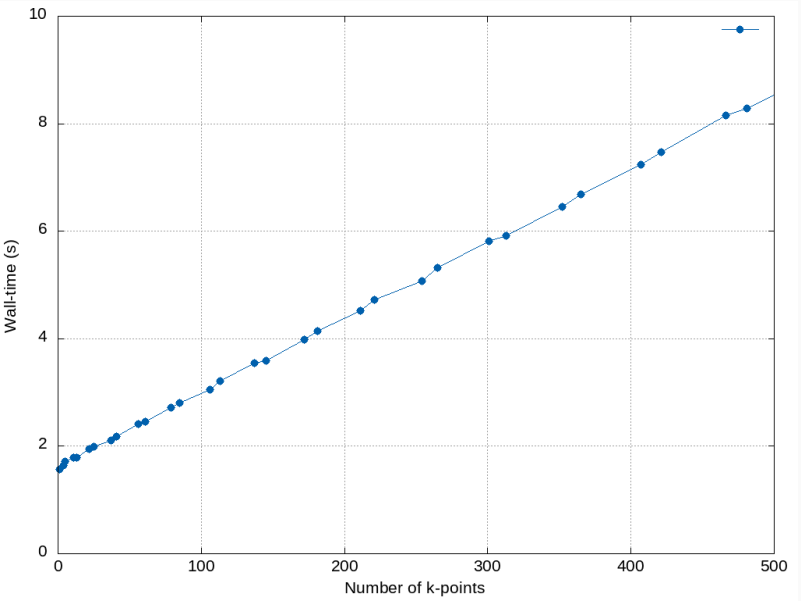
| | | |
|---|---|---|
| 1 | 0 | 0 |
| 0 | 1 | 0 |
| 0 | 0 | 1 |

Using: Y=2

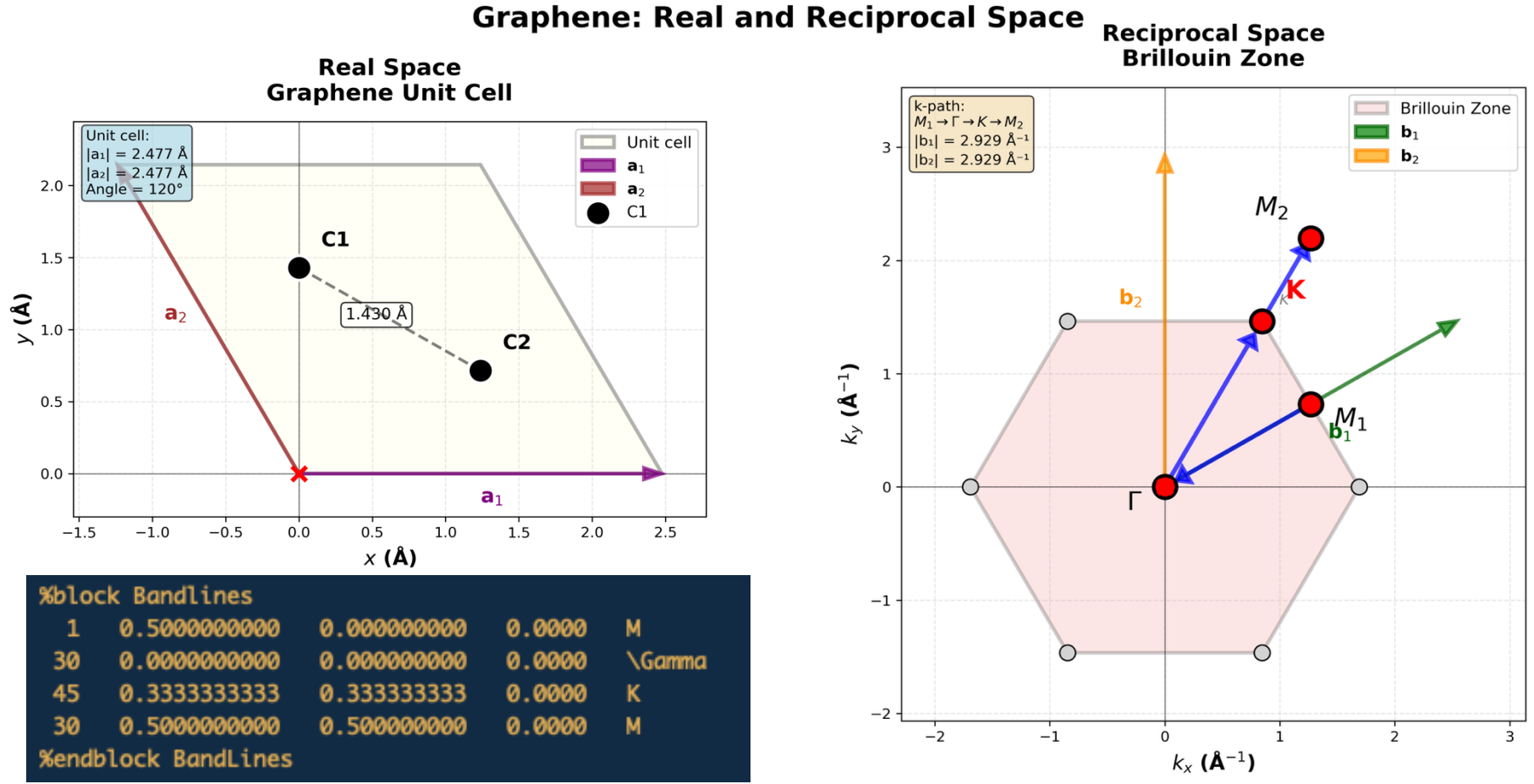
| | | |
|---|---|---|
| 2 | 0 | 0 |
| 0 | 2 | 0 |
| 0 | 0 | 1 |

# Tutorial #2: **Reciprocal Space k-grid convergence**



# Tutorial #2: **Reciprocal Space k-grid convergence**




```
%block Bandlines
  1    0.5000000000    0.000000000    0.0000    M
 30    0.0000000000    0.000000000    0.0000    \Gamma
 45    0.3333333333    0.333333333    0.0000    K
 30    0.5000000000    0.500000000    0.0000    M
%endblock BandLines
```

# Tutorial #2: **Reciprocal Space k-grid convergence**

## INPUTS FOLDER

**siesta/kpoint-convergence /**
├── 01_graphene/

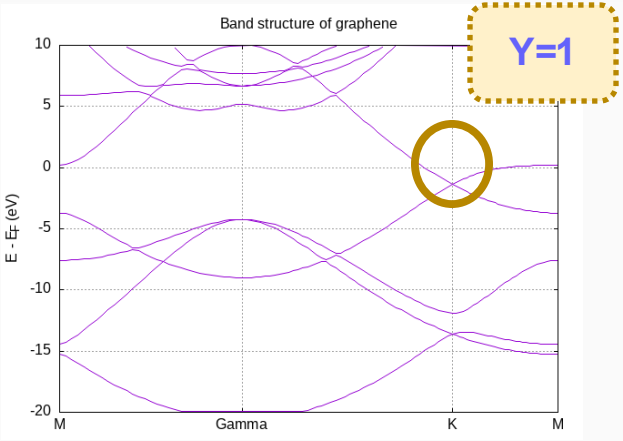
## OUTPUTS FOLDER

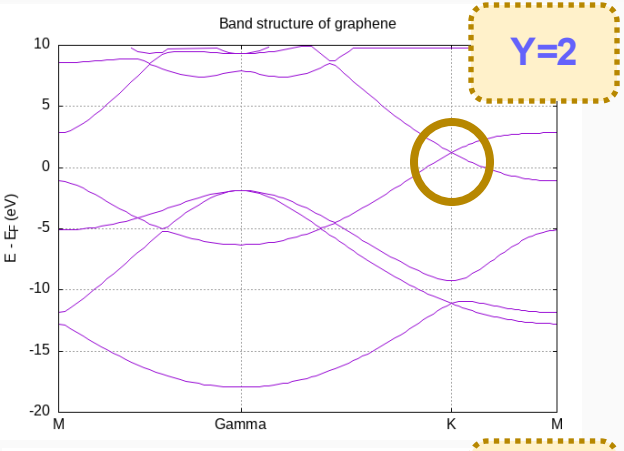
**siesta/**
├── exercise_day3_kpoints2/
│ ├── create_your_own_folders

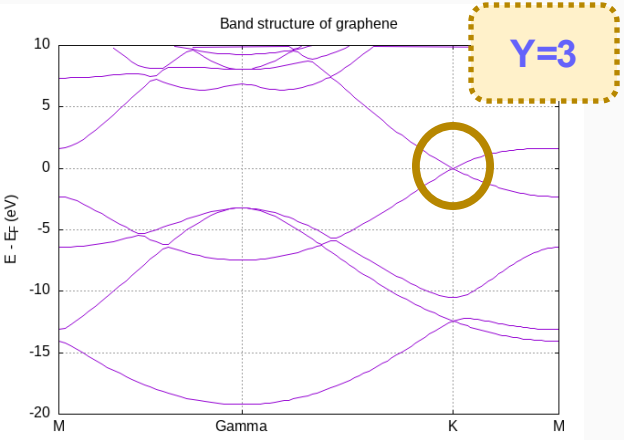
1. Copy all files from previous exercise folder to new exercise.
2. In this exercise you will redo the Graphene exercise 4 but this time uncomment the following options:

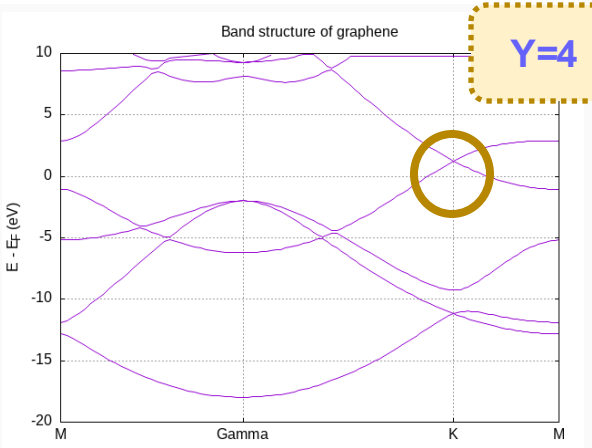
```
#WriteKbands        T
#WriteBands         T
#BandLinesScale     ReciprocalLatticeVectors

#%block Bandlines
#  1    0.5000000000   0.000000000   0.0000   M
# 30    0.0000000000   0.000000000   0.0000   \Gamma
# 45    0.3333333333   0.333333333   0.0000   K
# 30    0.5000000000   0.500000000   0.0000   M
#%endblock BandLines
```

# Tutorial #2: **Reciprocal Space k-grid convergence**




Y 0 0
0 Y 0
0 0 1

# Tutorial #3: **SCF convergence**

**OBJECTIVES**

Tests different mixing strategies

- Mixing methods
- Adjust weight
- History

Check different systems

- Simple molecule
- More challenging metal cluster

# Tutorial #3: **SCF convergence**

## INPUTS FOLDER

**siesta/scf_convergence**
├── 01-CH4/

## OUTPUTS FOLDER

**siesta/**
├── exercise_day3_scf/
│ ├── create_your_own_folders

1. Copy all files from input folder to the exercise folder.
2. In this exercise you will scan three keywords:

Add these lines in the input file

- **SCF.mix** **density**
- **SCF.mixer.method** **linear**
- **SCF.mixer.weight** **0.6**

{ density | hamiltonian }

{ linear | Pulay | Broyden }

{ 0 | 0.1 | 0.2 | … | 1 }

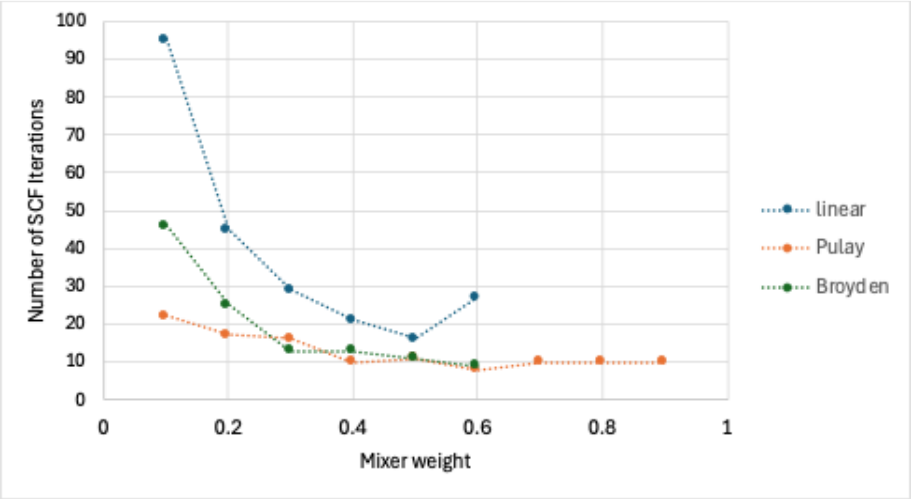
0: New DM is the same as in the previous step
1: New DM is totally different from the previos step

**comment:**

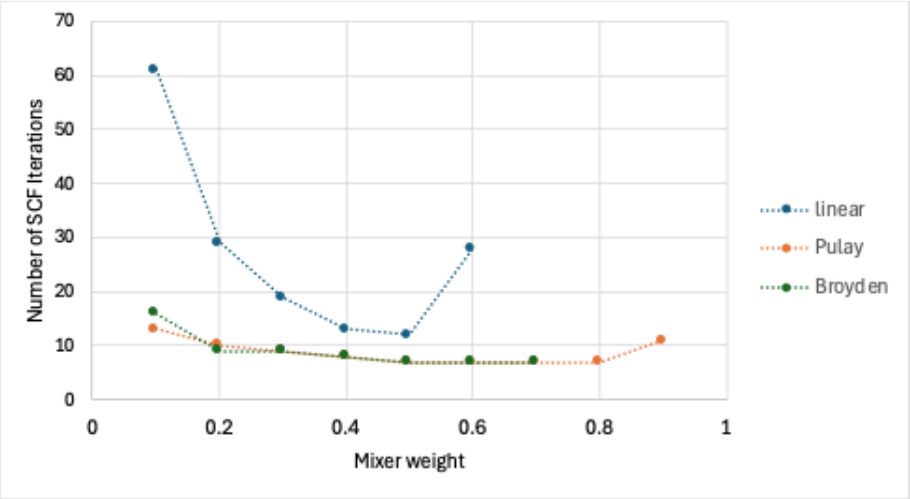
**#MaxSCFIterations 50**

# Tutorial #3: **SCF convergence**

Mixing Density Matrix




Mixing Hamiltonian



siesta
TM

# Thank you

# **Questions?**



Barcelona
Supercomputing
Center
Centro Nacional de Supercomputación

CSIC
CONSEJO SUPERIOR DE INVESTIGACIONES CIENTÍFICAS