

Introduction to density functional theory

Mini course on DFT for PhD

LECTURE 4 - 11 March 2026

outline of today session

THEORY ABOUT :

- ENERGY BARRIERS (Nudged Elastic Band approach)
- SOME TECHNICALITIES that have to do with FORCES:
 - stress; condition of minimum energy vs. zero stress
- variable cell calculations
- CORE LEVEL SHIFTS

HANDS-ON:

- charge density distribution: total and charge rearrangements (e.g.: in small molecules by difference)
- energy barriers “by hand” in simple cases (e.g.: proton transfer)
- ...
- OTHER POSSIBLE POST-PROCESSING TOOLS

nudged elastic bands

nudged elastic bands

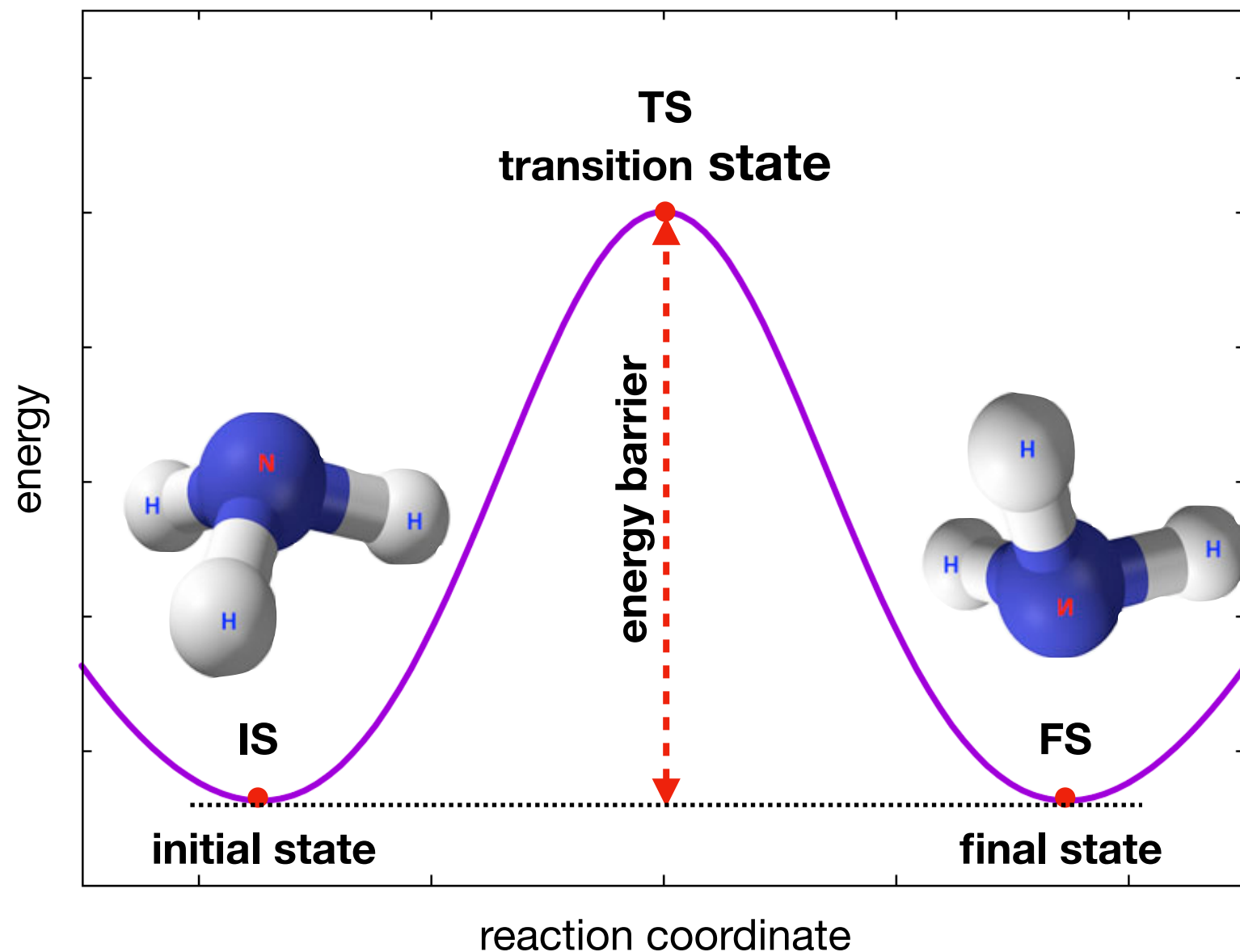
If we know: the **initial state** (first) and **final state** (last) corresponding to “**reactants**” and “**products**”

and we want to know the **reaction pathway**, i.e. what happens in between; even better, the **minimum energy path (MEP)**

or at least which is the **transition state (TS)**, that determines the **energy barrier** for the “reaction” to occur

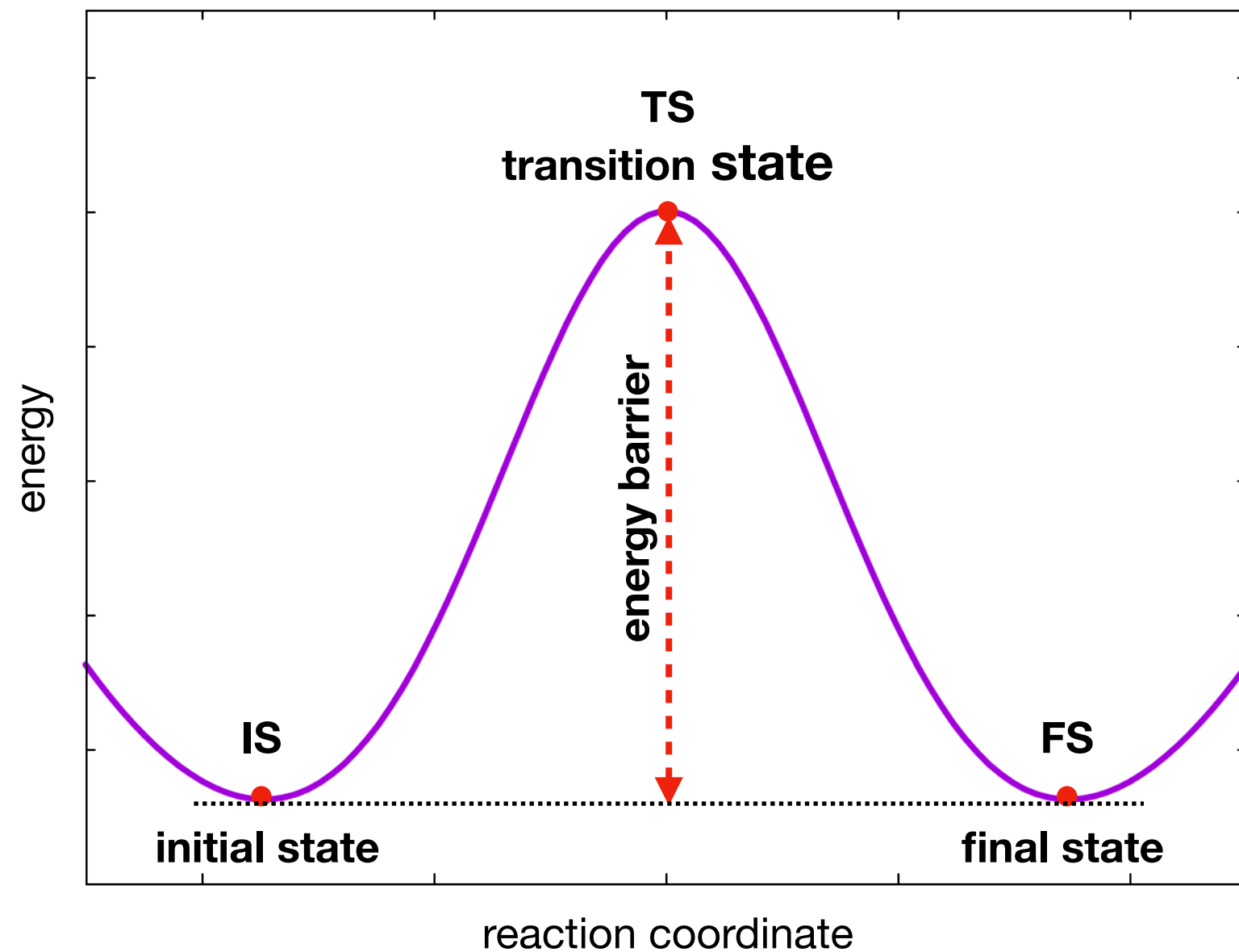
Energy barriers, transition states, reaction pathway “by hands”: NH_3 umbrella inversion barrier

In some cases the TS is well defined by **symmetry** and barriers can be calculated “by hands” following physical intuition and studying directly the TS configuration (e.g. in this example: TS corresponds to N in-plane with H3)



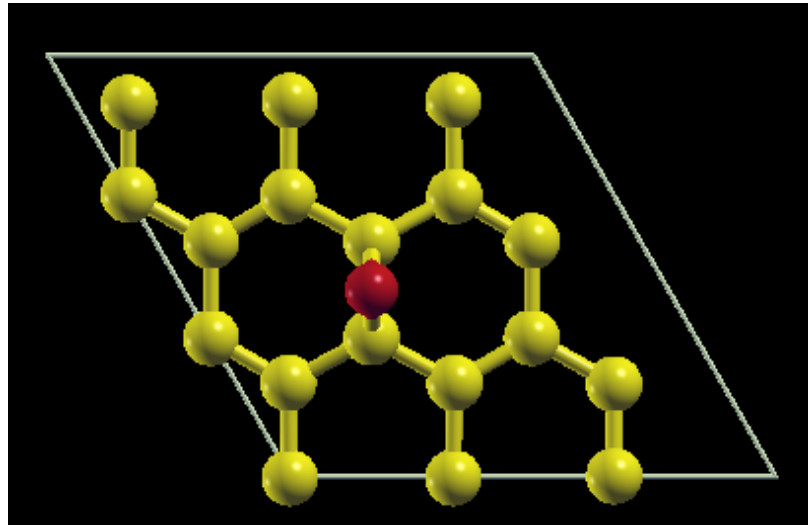
Energy barriers, transition states, reaction pathway “by hands”: collinear proton transfer

today

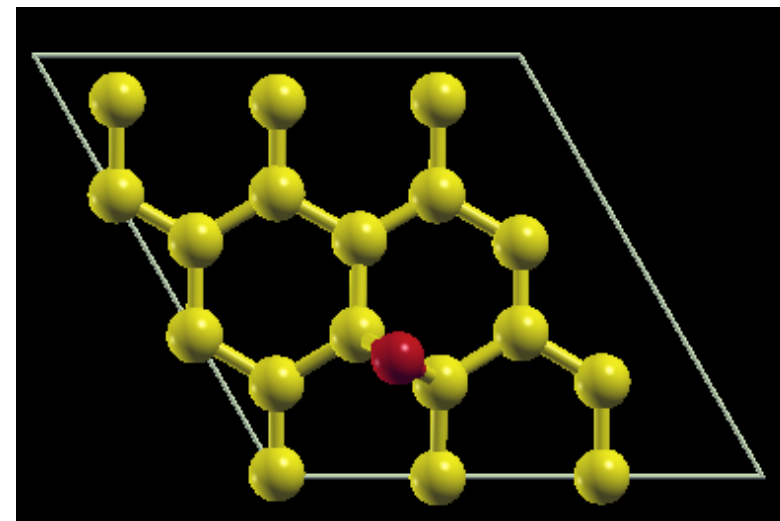


Energy barriers, transition states, reaction pathway “almost” “by hands”: epoxy diffusion on graphene

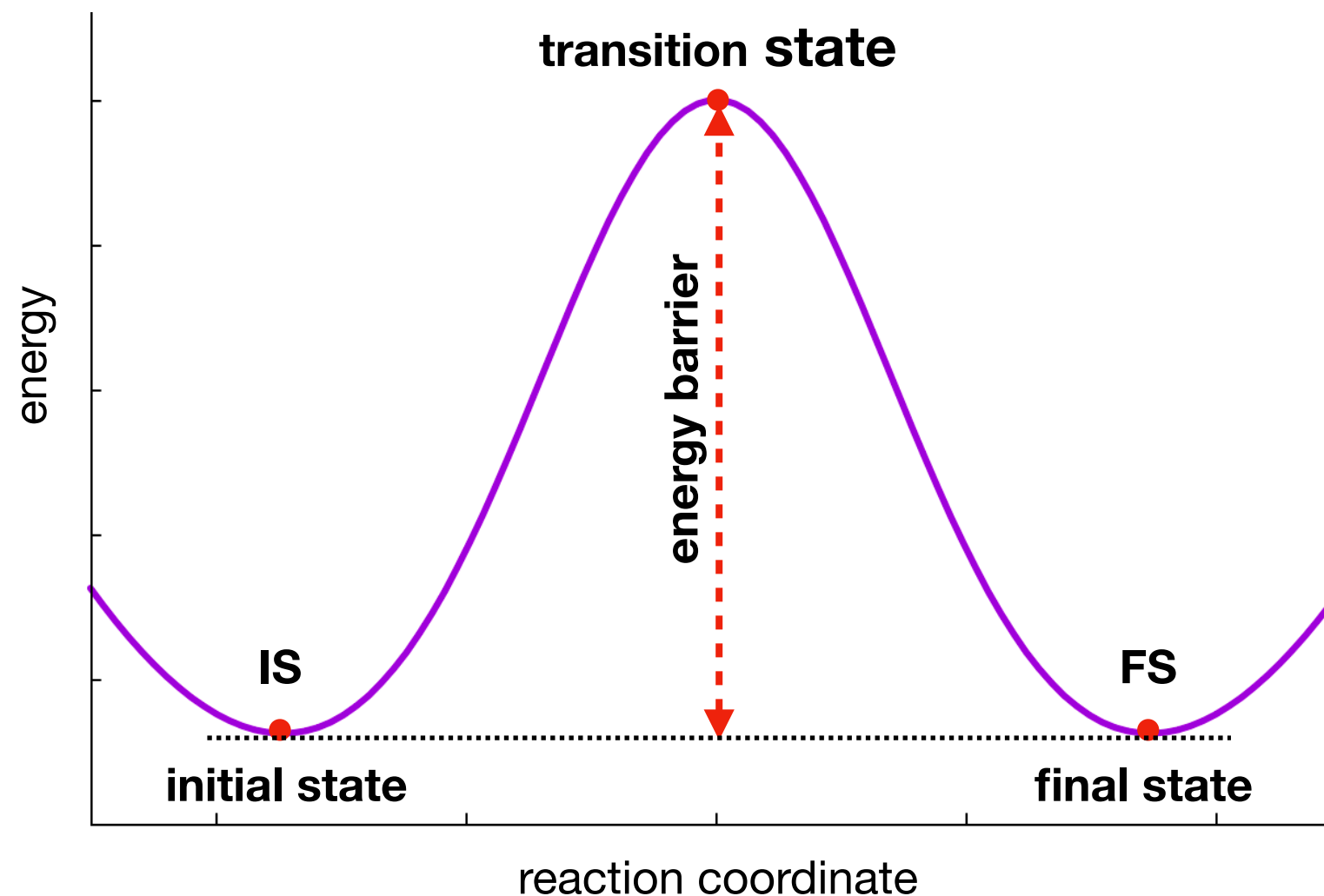
Initial State



Final State



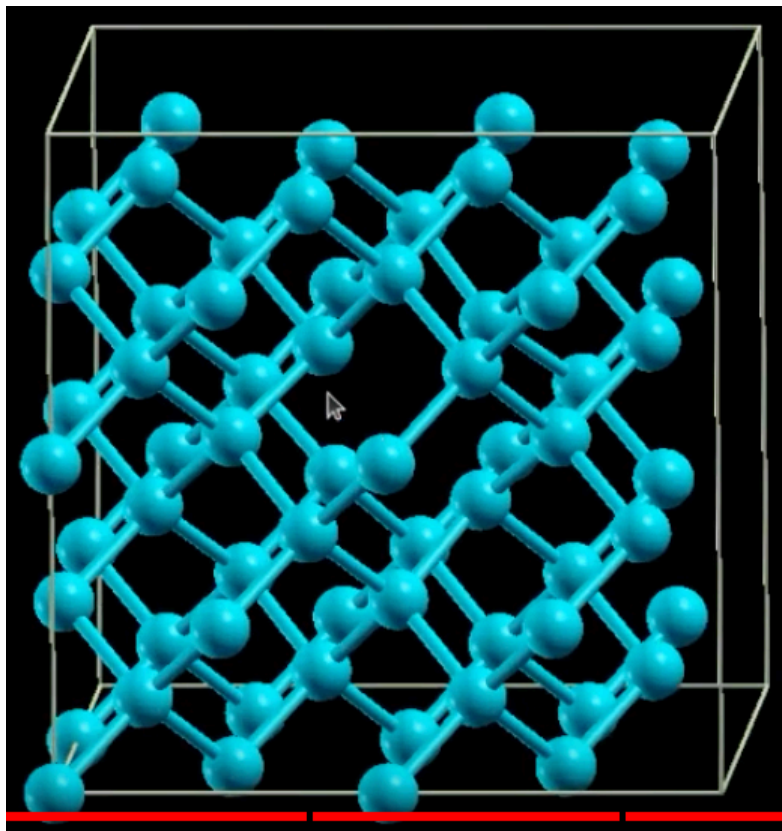
?



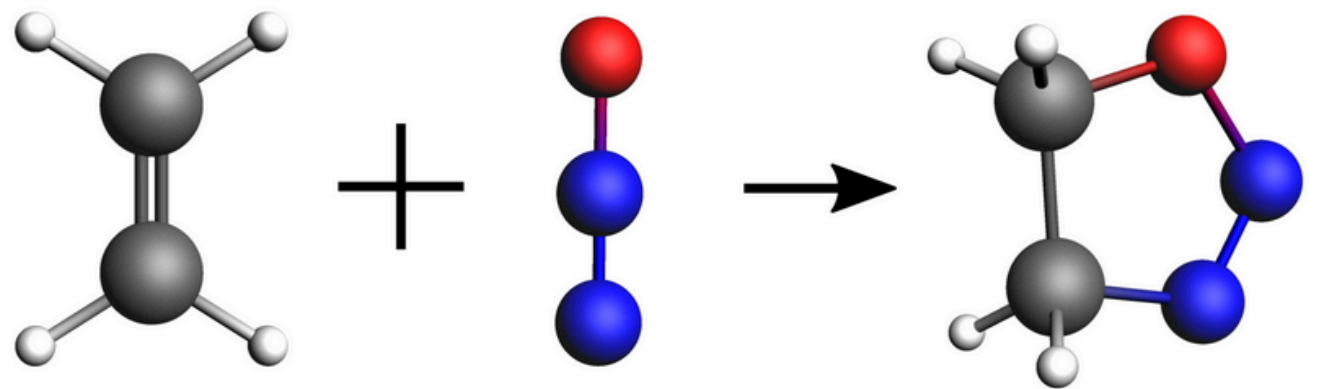
Energy barriers, transition states, reaction pathway

but in most other cases, even rather simple,
the TS is not at all obvious...

diffusion of vacancies in silicon



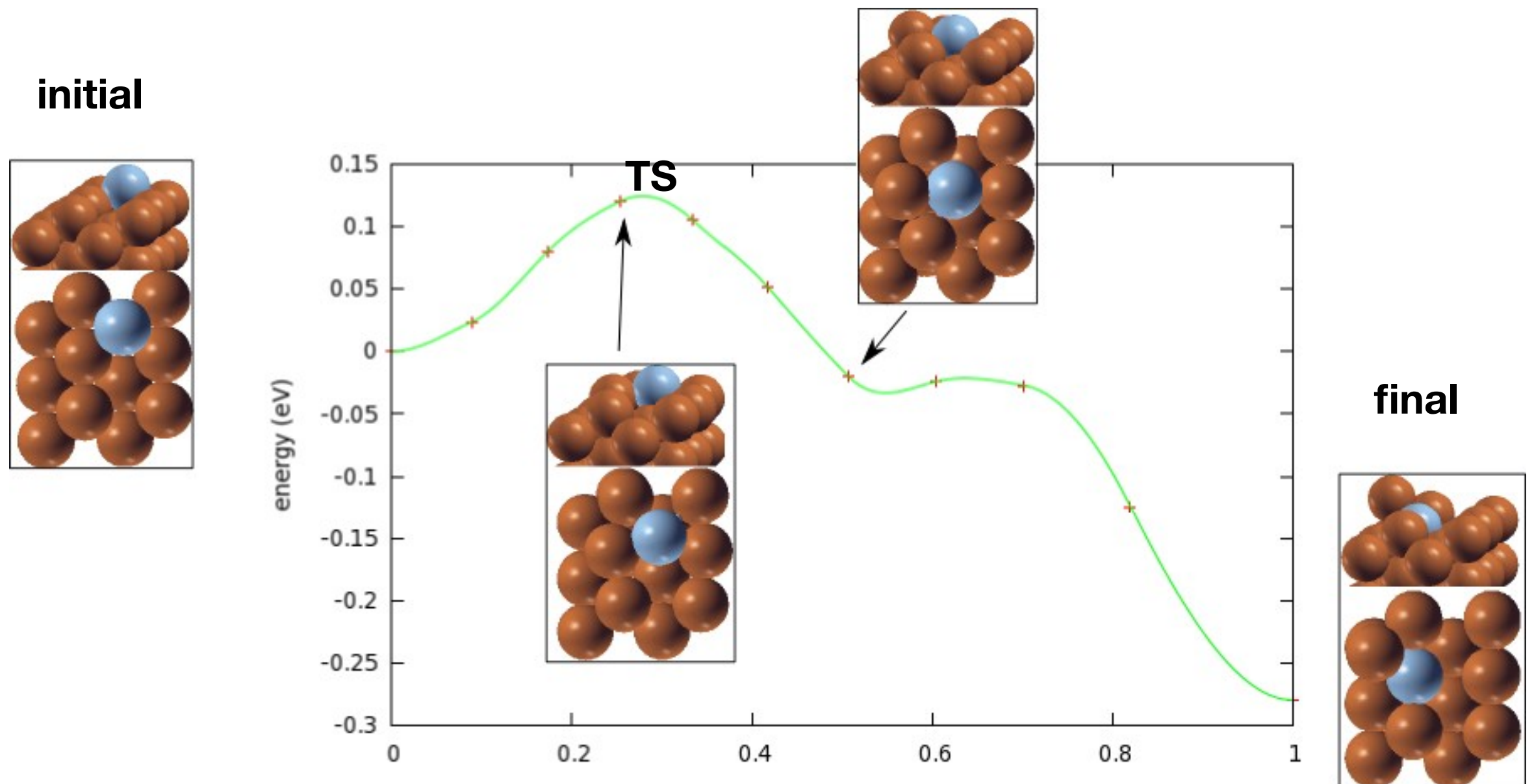
chemical reactions



Energy barriers, transition states, reaction pathway

a few examples

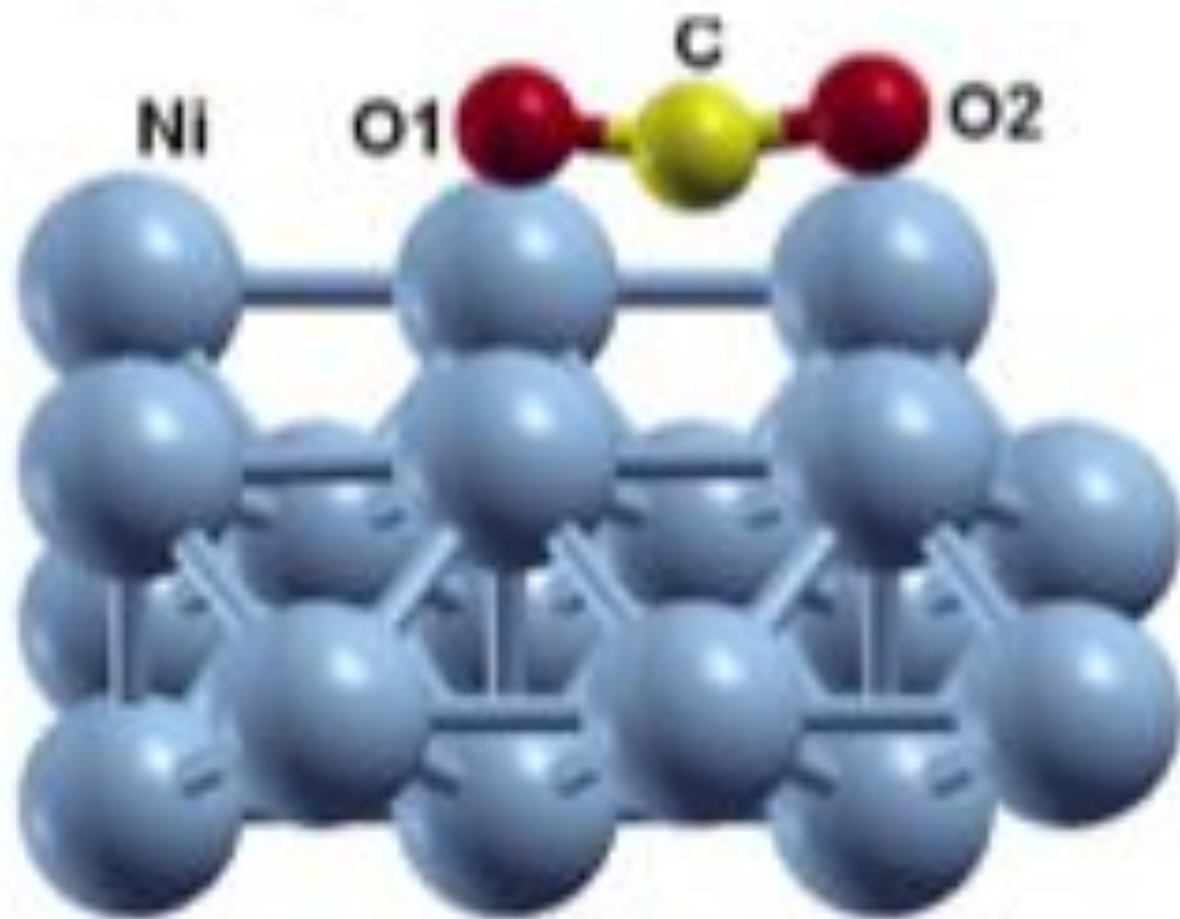
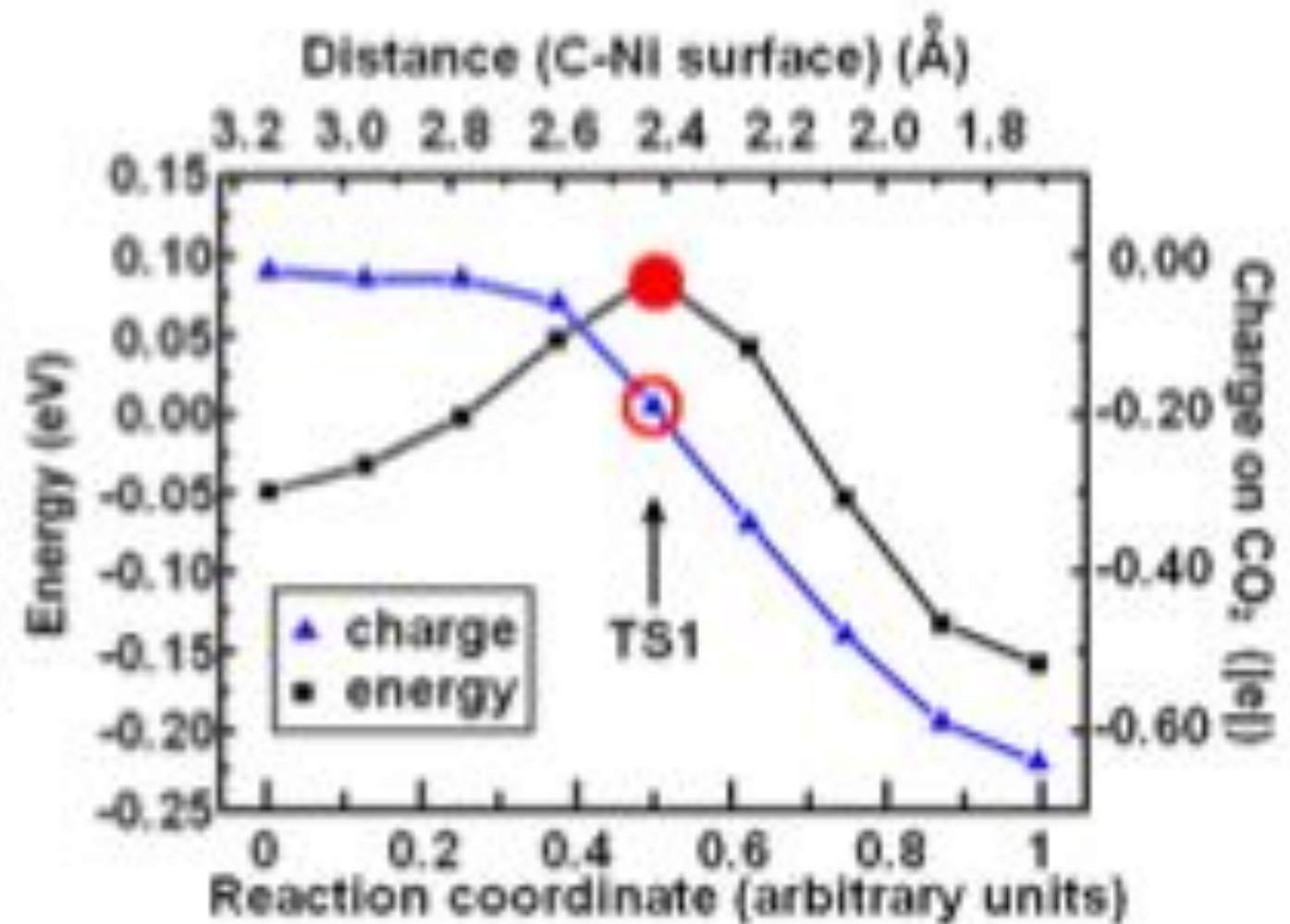
swap between Ni and Cu atoms on Cu surface



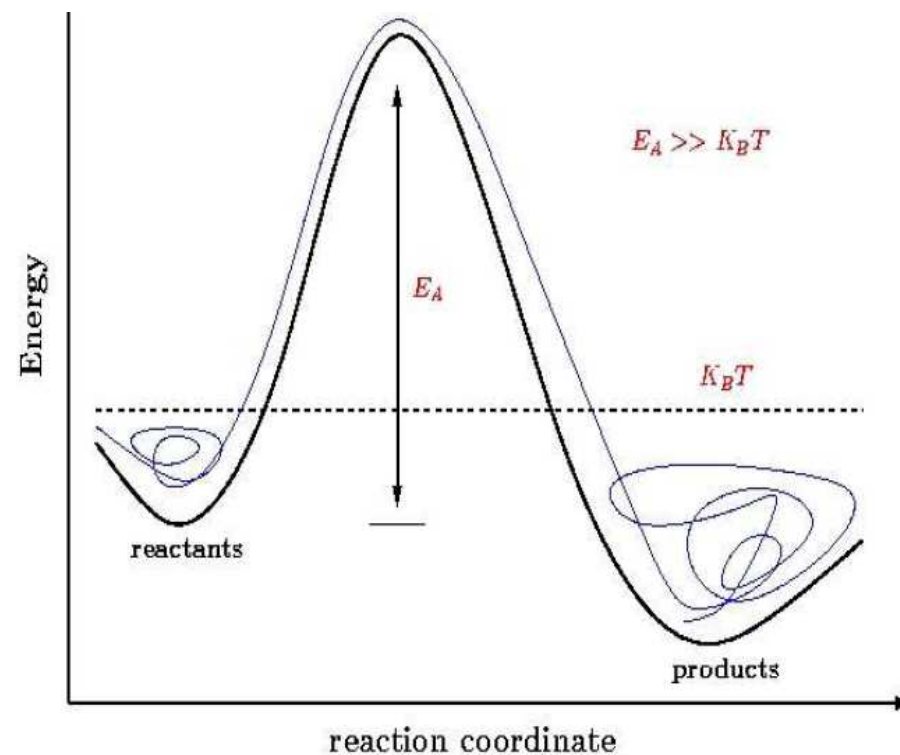
Energy barriers, transition states, reaction pathway

a few examples

activated adsorption of a molecule (CO_2) on a surface $\text{Ni}(110)$ from gas phase



Energy barriers, transition states, reaction pathway



The transition probability can be estimated using equilibrium statistical mechanics.

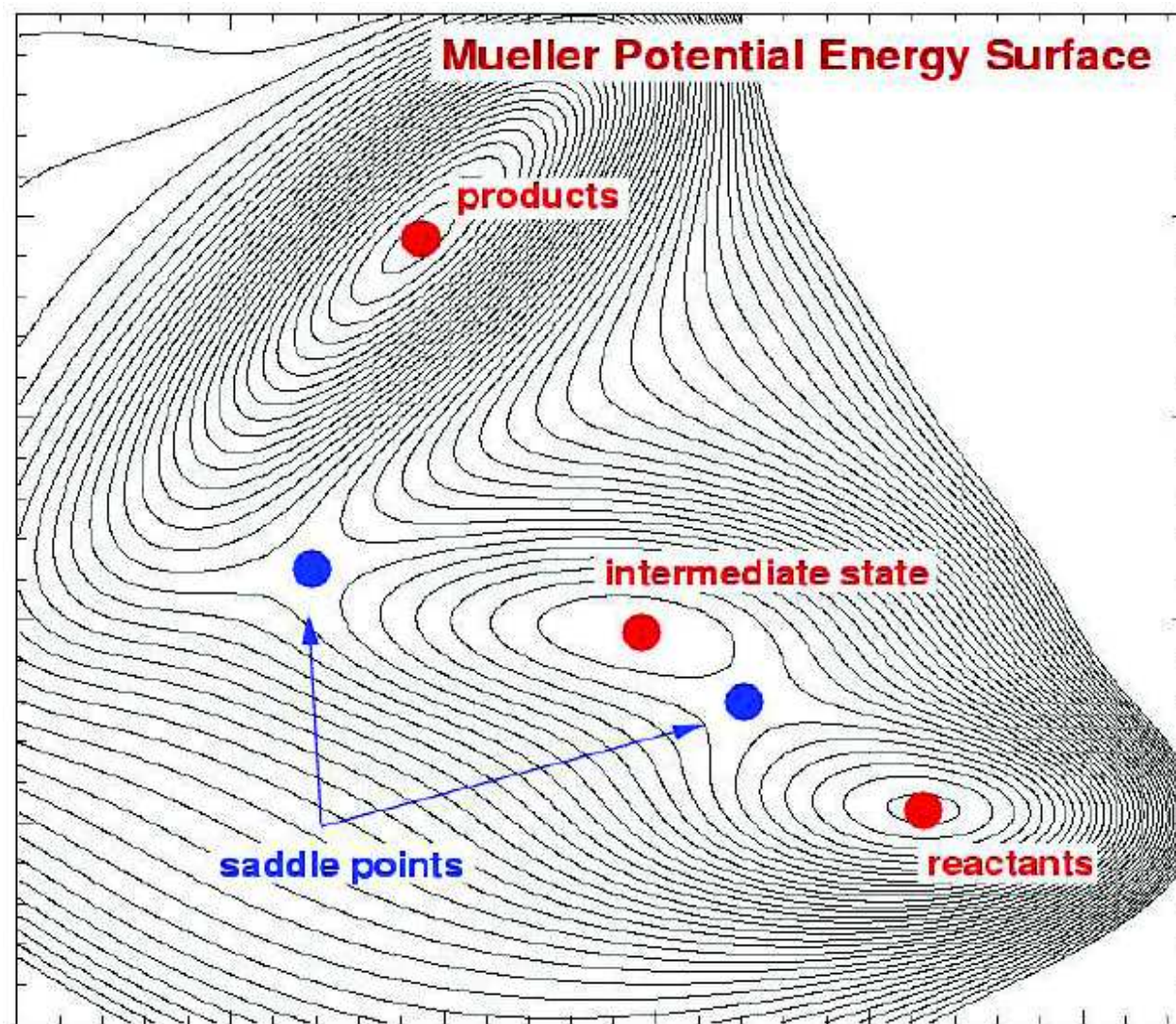
Once the saddle point has been located we can use **harmonic Transition State Theory** (hTST) to calculate the rate constants:

$$K_{\text{reactants} \rightarrow \text{products}} = \mathcal{A} \times e^{-\frac{E_A}{K_B T}}$$

Saddle points are unstable configurations and their location is a difficult task

Energy barriers, transition states, reaction pathway

Saddle points in multidimensions: the Mueller Potential



The path with the "highest" transition probability is the Minimum Energy Path.

MEP: the components of the force orthogonal to the path are zero.

Energy barriers, transition states, reaction pathway

Müller-Brown Potential

The Müller-Brown potential (Theoretica Chimica Acta, 53, 75-93, 1979) is a classic 2D benchmark for testing optimization algorithms and molecular dynamics methods, since it captures the essential topology of chemical reaction landscapes

It is a two-parametric model surface combining four two-dimensional Gaussian functions with : each Gaussian contributes a different “bump” or “well” to the landscape. The parameters control amplitude, width, orientation, and center position.

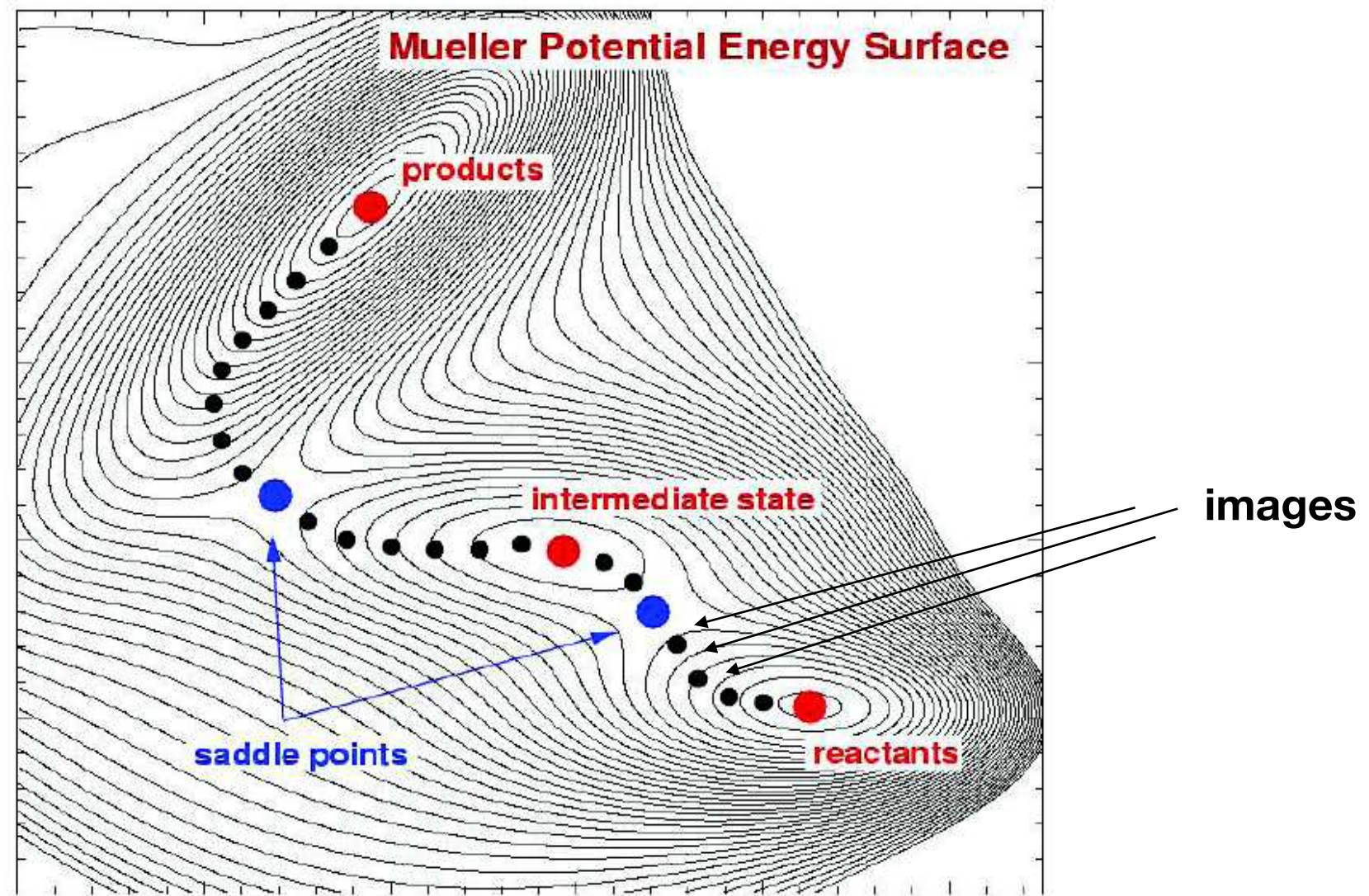
Three terms have negative amplitudes (creating energy wells), while the fourth has a positive amplitude (creating a barrier). A third Gaussian has a mixed term $\propto (x - x_0)(y - y_0)$ that creates a tilted orientation that gives the surface its characteristic curved pathways.

The simple formula creates a rich topography with exactly the features needed to challenge optimization algorithms:

Energy and gradients can be calculated analytically

Energy barriers, transition states, reaction pathway

Saddle points in multidimensions: the Mueller Potential



The path with the "highest" transition probability is the Minimum Energy Path.

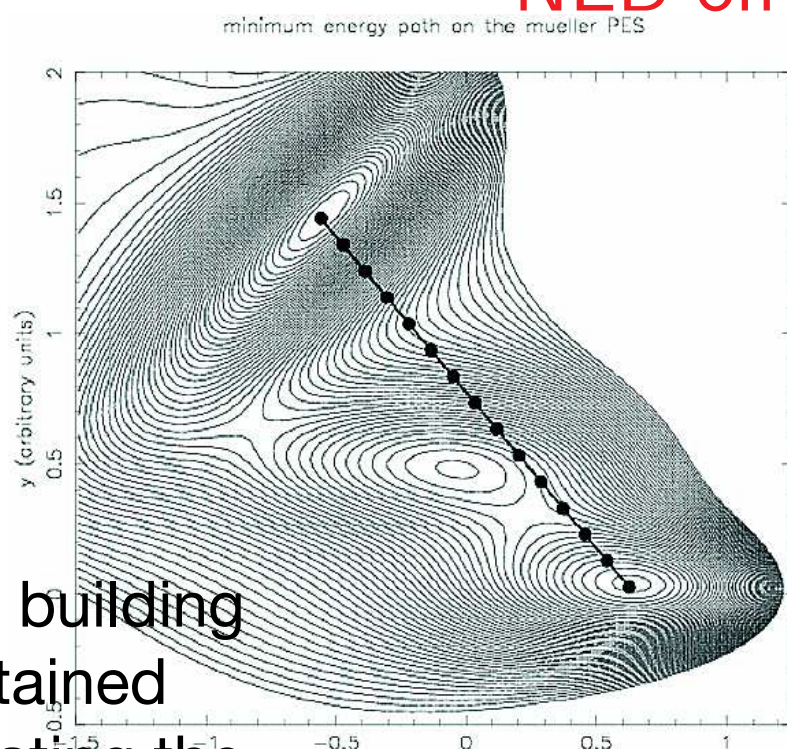
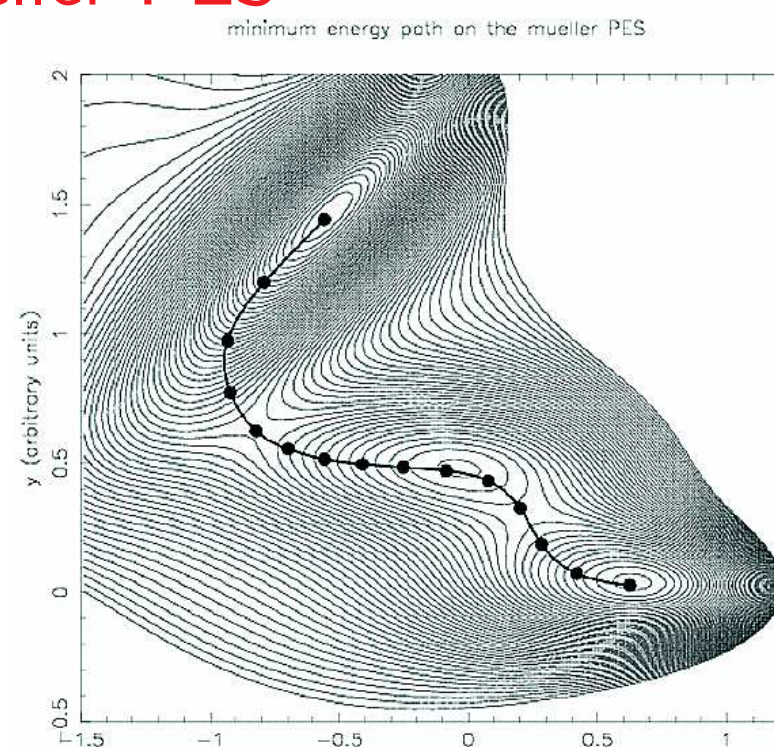
MEP: the components of the force orthogonal to the path are zero.

The MEP goes through the saddle point

Let us connect subsequent images by springs that only operate along the path

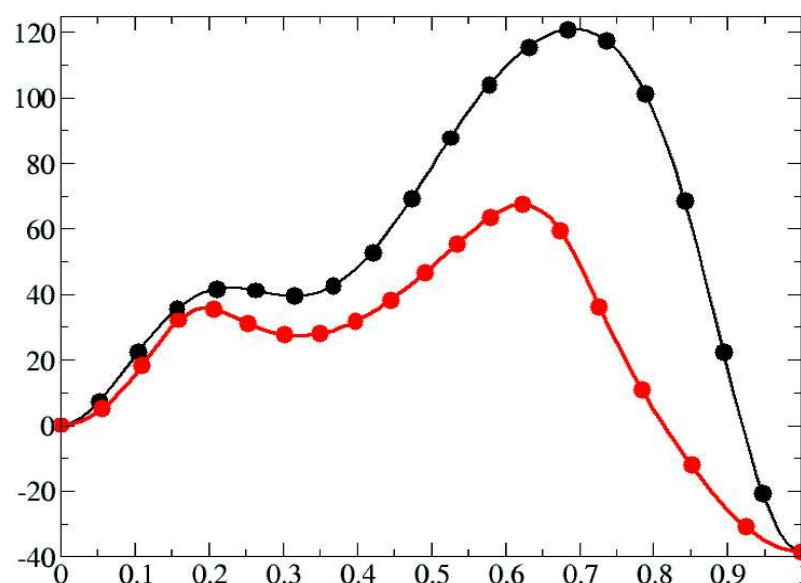
Energy barriers, transition states, reaction pathway

NEB on the Mueller PES



starts from building
images obtained
by interpolating the
first and the last

but we can supply
also some
intermediate images
to build a better initial
reaction path (if our
physical/chemical
intuition guides us...)



optimize the path on
the PES by using
“springs”

Energy barriers, transition states, reaction pathway

nudged elastic bands

PWneb is part of the *Quantum ESPRESSO* distribution and uses the *PWscf* package as electronic-structure computing tools (“engine”)

executable : **neb.x**

Running NEB

in this case, the input file contains a **NEB-specific part** with some keywords and a **part for the “engine”** (the PWscf electronic structure calculation):

```
BEGIN
```

```
BEGIN_PATH_INPUT
```

```
~... neb specific namelists and cards
```

```
END_PATH_INPUT
```

```
BEGIN_ENGINE_INPUT
```

```
~...pw specific namelists and cards
```

```
BEGIN_POSITIONS
```

```
FIRST_IMAGE
```

```
~...pw ATOMIC_POSITIONS card
```

```
INTERMEDIATE_IMAGE
```

```
~...pw ATOMIC_POSITIONS card
```

```
LAST_IMAGE
```

```
~...pw ATOMIC_POSITIONS card
```

```
END_POSITIONS
```

```
~... other pw specific cards
```

```
END_ENGINE_INPUT
```

```
END
```

<= optional

Running neb.x, first of all the **neb.dat** file and several **pw_X.in** input files are generated

NEB input variables

Detailed explanation of all keyword in:

http://www.quantum-espresso.org/Doc/INPUT_NEB.html

NEB-specific part

```
BEGIN
BEGIN_PATH_INPUT
&PATH
  restart_mode = 'from_scratch'
  string_method = 'neb',
  nstep_path = 20, ← Number of ionic+electronic steps
  ds = 2.D0,
  opt_scheme = "broyden",
  num_of_images = 7, ← Number of points used to discretize the path (>3)
  k_max = 0.3D0,
  k_min = 0.2D0,
  CI_scheme = "auto", ← Automatic choice of the climbing image
                        (alternative: "no-CI" [default] or "manual")
  path_thr = 0.1D0, ← Threshold for the norm of the force
                    orthogonal to the path, in eV/Å
/
END_PATH_INPUT
BEGIN_ENGINE_INPUT
.....
END_ENGINE_INPUT
END
```

NEB input variables

Detailed explanation of all keyword in:

http://www.quantum-espresso.org/Doc/INPUT_NEB.html

part for the “engine”

```
...
ATOMIC_SPECIES
H      1.01  H.pbe-rrkjus.UPF
BEGIN_POSITIONS
FIRST_IMAGE                                ← mandatory
ATOMIC_POSITIONS { angstrom }
H      -3.00  0.0  0.0  1  0  0
H      0.00  0.0  0.0  0  0  0
H      0.75  0.0  0.0  1  0  0
LAST_IMAGE                                ← optional intermediate images here
ATOMIC_POSITIONS { angstrom }
H      -3.00  0.0  0.0  1  0  0
H      0.00  0.0  0.0  0  0  0
H      0.75  0.0  0.0  1  0  0
END_POSITIONS
K_POINTS { gamma }

END_ENGINE_INPUT
```

Running NEB

Alternative way: specify the number N of images with the command line option `-input_images N`, and providing the input data for PWneb in a file named `neb.dat` and for the PWscf engine in the files `pw_X.in` ($X = 1, \dots, N$).

```
$[path]/neb.x -input_images [N]
```

neb.dat

```
&PATH
  restart_mode   = 'from_scratch'
  string_method = 'neb',
  nstep_path     = 20,
  ds             = 2.D0,
  opt_scheme     = "broyden",
  num_of_images  = 7,
  k_max          = 0.3D0,
  k_min          = 0.2D0,
  CI_scheme      = "auto",
  path_thr       = 0.1D0,
```

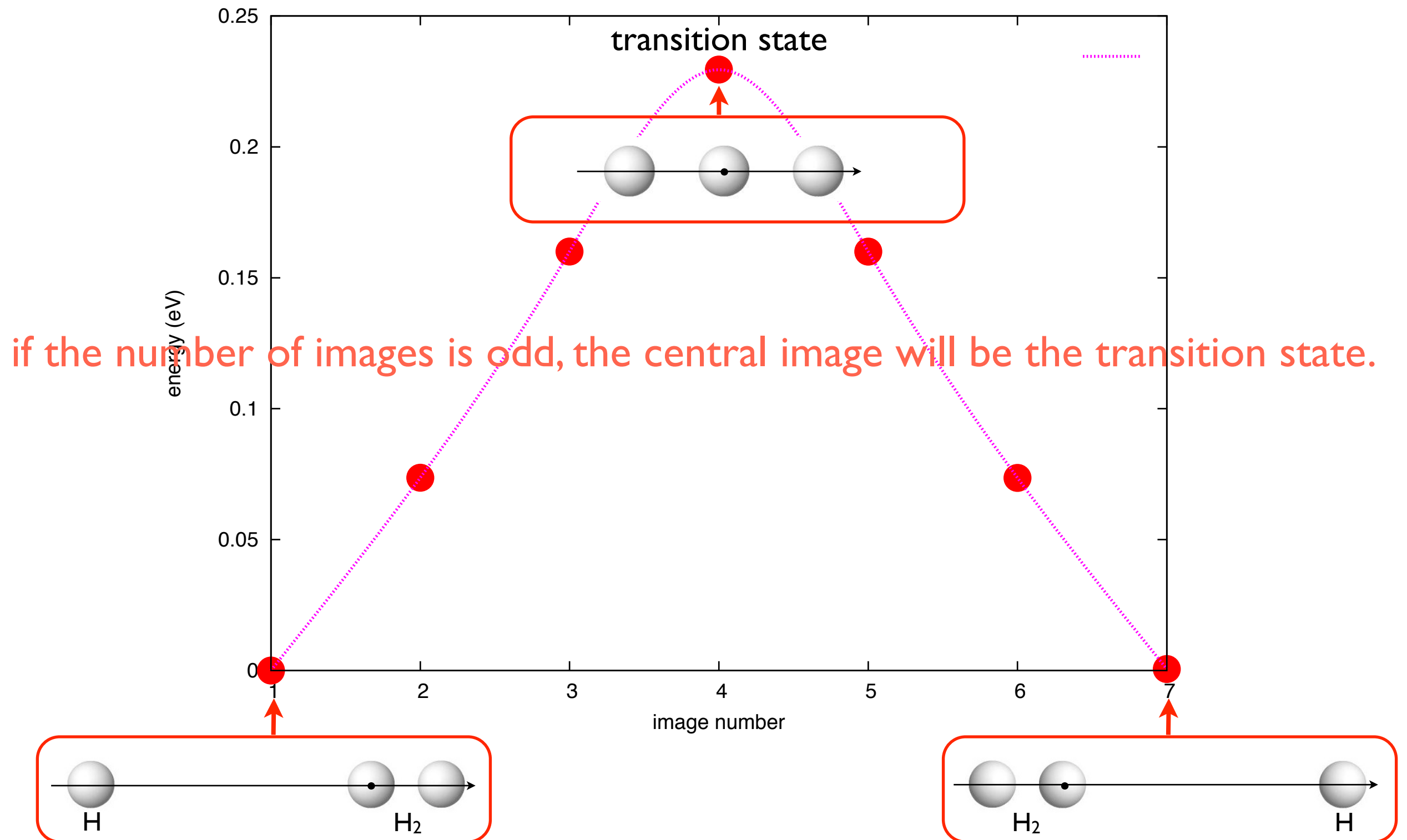
```
/
```

pw_X.in

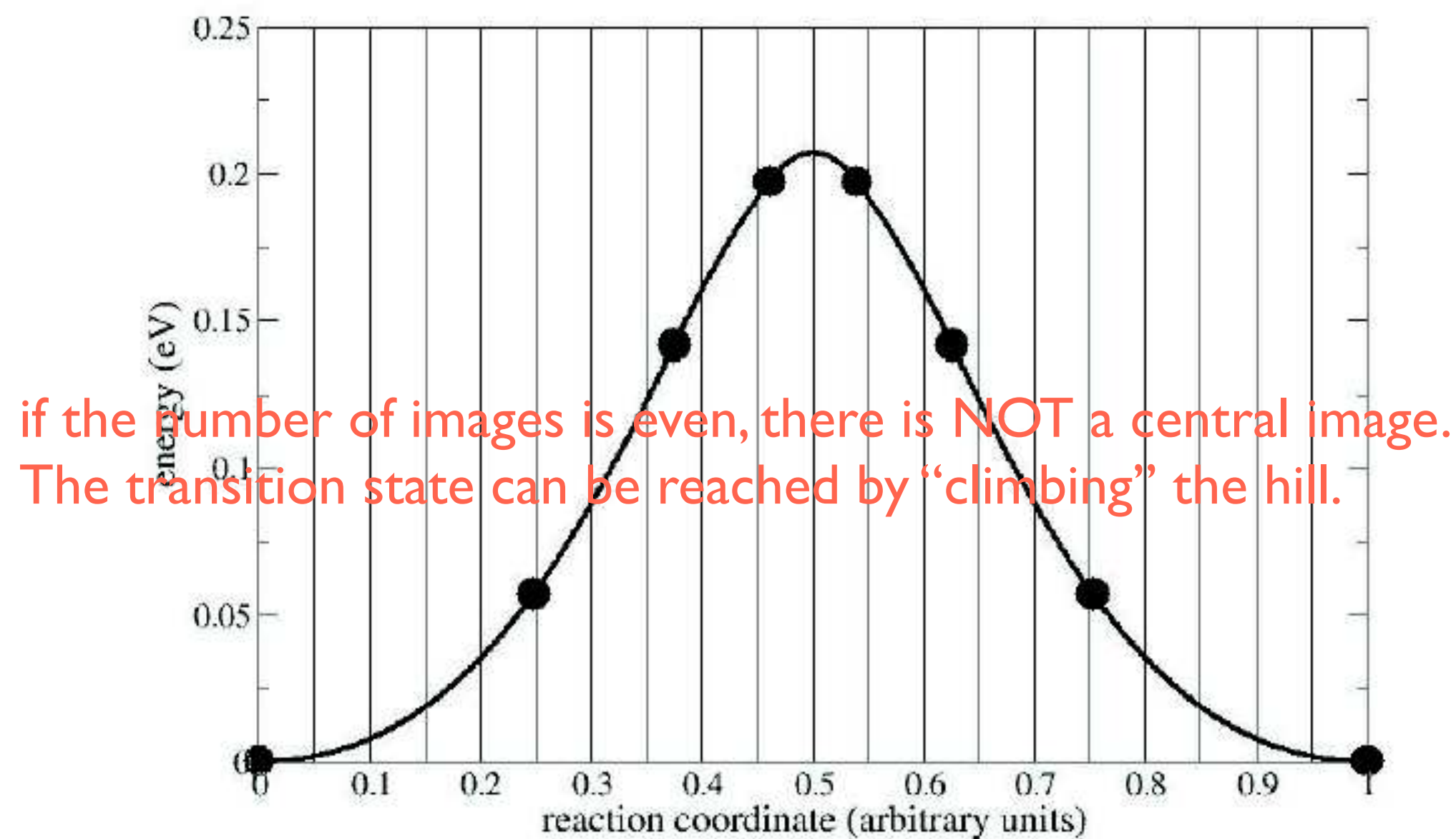
```
&control
  calculation = ...
/
&system
  ibrav = ...
/
&electrons
  ...
  ATOMIC_POSITIONS
  { angstrom }
  ...
```

Energy barriers, transition states, reaction pathway collinear proton transfer

num_of_images = 7

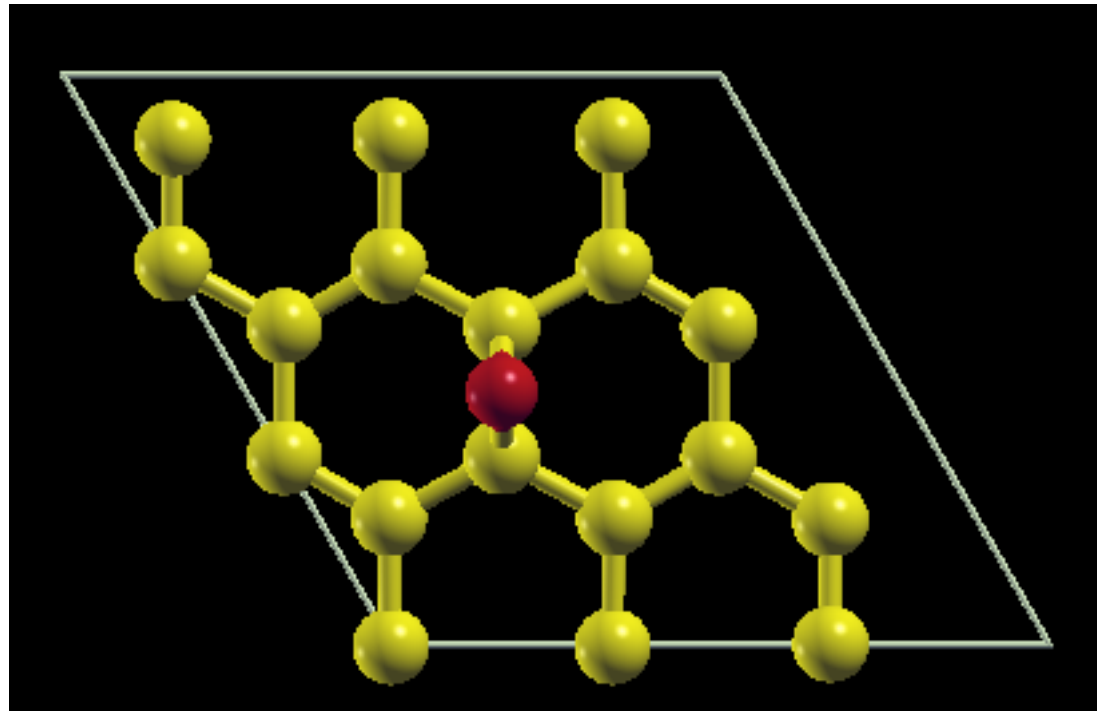


Energy barriers, transition states, reaction pathway collinear proton transfer



Example: epoxy diffusion on graphene

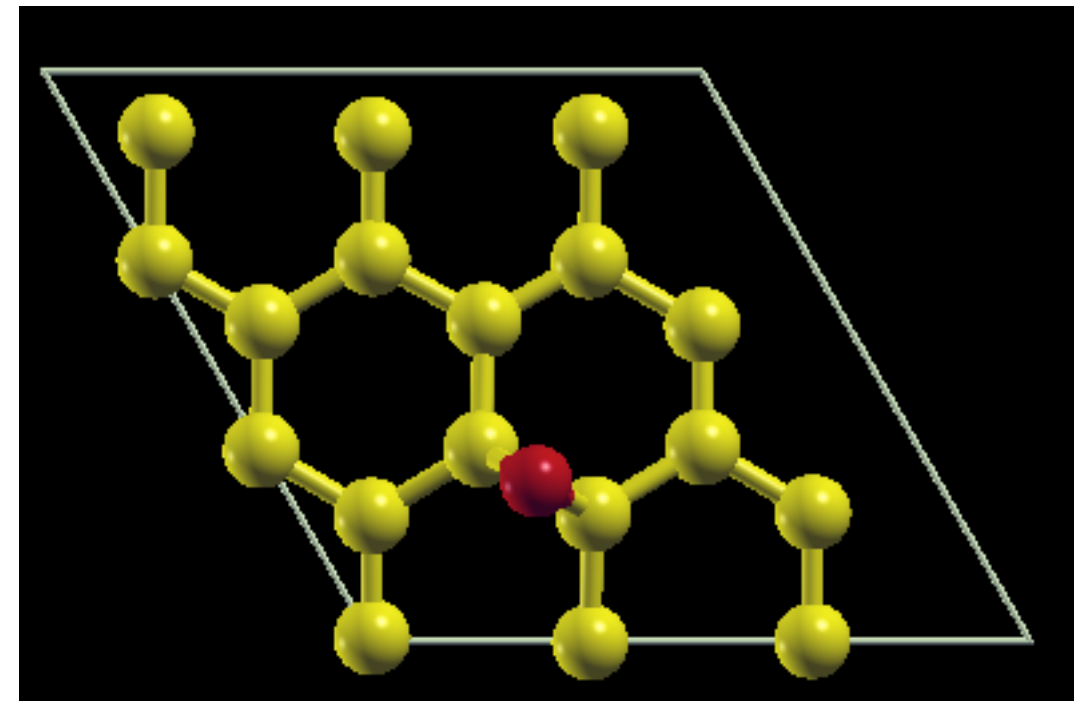
Initial State



(optimized)

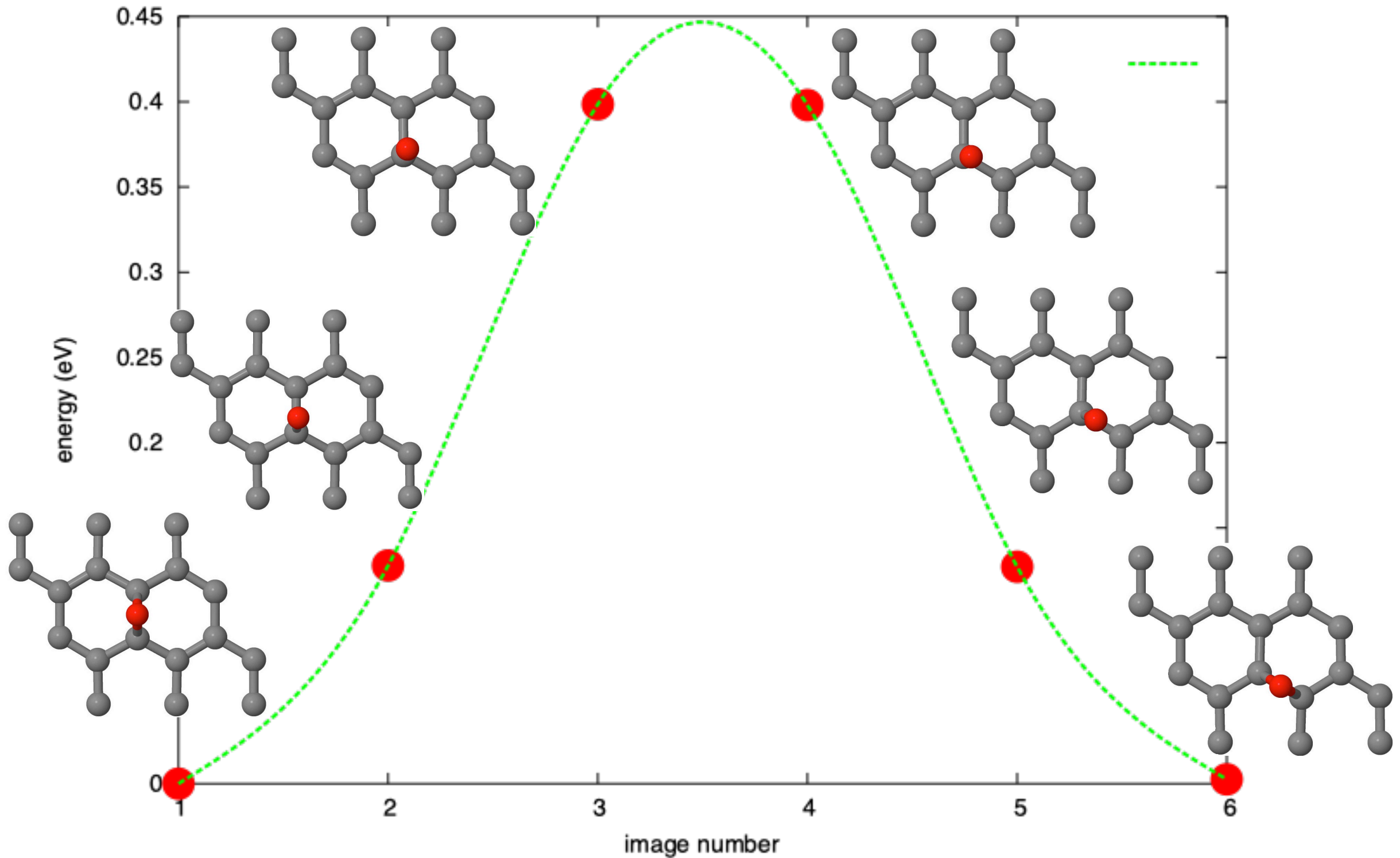


Final State



(optimized)

Example: epoxy diffusion on graphene



NEB: some advice

1) in the previous example, `CI_scheme="auto"` is fine.

Caution in other (non symmetric) cases! Difficult convergence if the CI image change during the optimization process. Better to start with `"no-CI"`, then restart with CI .

2) Carefully choose the initial path. If images are not explicitly provided, the code makes a linear interpolation of the atomic positions between consecutive input images (only initial and final are mandatory): troubles if some atoms overlap or get very close in some of the interpolated images!

3) For large system, start with low accuracy, and then increase it to refine the results. Keeping fixed some atoms and relaxing them later on, may also help.

other technicalities:

minimum of energy versus stress

STRAIN, STRESS (and PRESSURE)

homogeneous deformation

$$r \longrightarrow r' = (1 + \epsilon)r$$

stress

$$\sigma = -\frac{1}{\Omega} \frac{\partial E}{\partial \epsilon}$$

strain

It's a first order derivative (Hellman-Feynman)

-> NO NEED to know how electrons re-adjust

-> NO NEED to know how internal coordinates change

If we are interested just in homogeneous deformation (scalar ϵ), we can obtain σ directly from the ground state wavefunction (no need of doing $dE/d\epsilon$ by hand...)

Nielsen and Martin Phys.Rev.Lett. 50, 697 (1983),

Nielsen and Martin Phys.Rev. B 32, 3780 & 3792 (1985).

STRAIN, STRESS (and PRESSURE)

e.g.: Si (done)

... ..

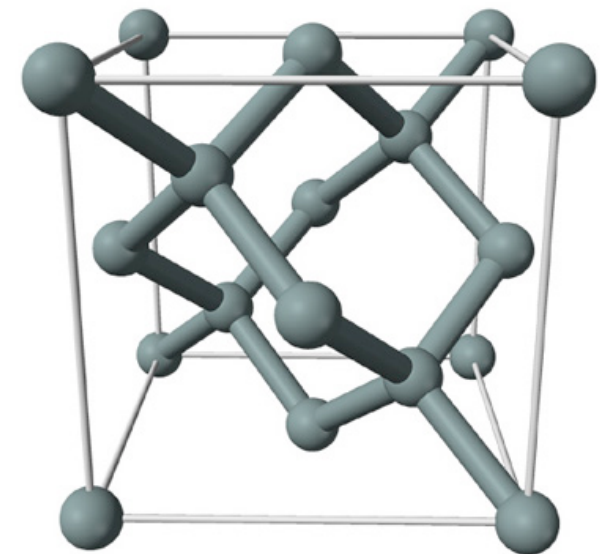
```
!   total energy           =      -15.82676466 Ry
    Harris-Foulkes estimate =      -15.82676466 Ry
    estimated scf accuracy  <       4.5E-09 Ry
```



The total energy is the sum of the following terms:

```
one-electron contribution =      4.80030732 Ry
hartree contribution      =      1.09369711 Ry
xc contribution           =     -4.82101051 Ry
ewald contribution        =    -16.89975858 Ry
```

convergence has been achieved in 6 iterations

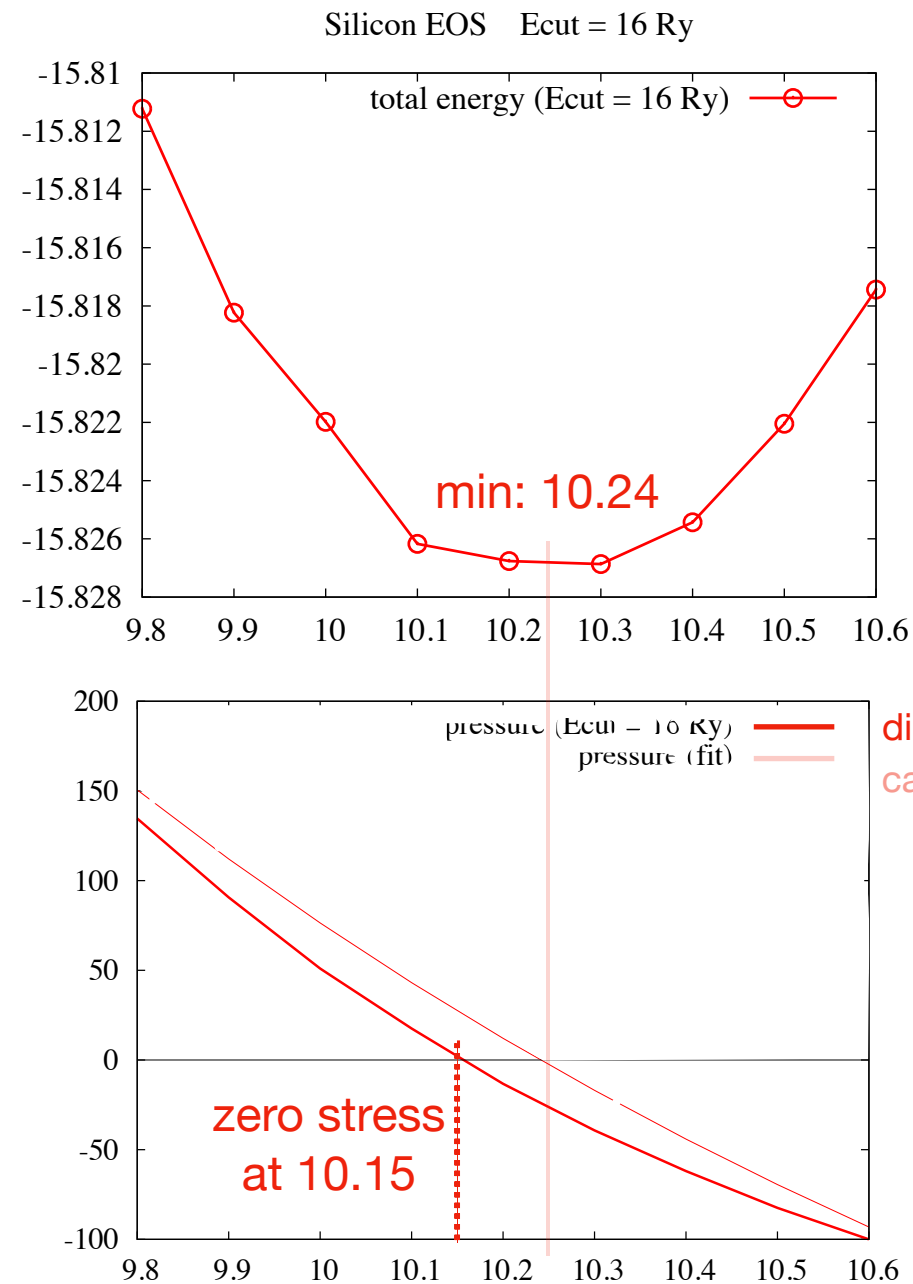


entering subroutine stress ...

total	stress	(Ry/bohr**3)	(kbar)	P=
-0.00009036	0.00000000	0.00000000	-13.29	-13.29
0.00000000	-0.00009036	0.00000000	0.00	0.00
0.00000000	0.00000000	-0.00009036	0.00	0.00

STRESS (and PRESSURE)

Minimum energy is not exactly zero pressure!!!

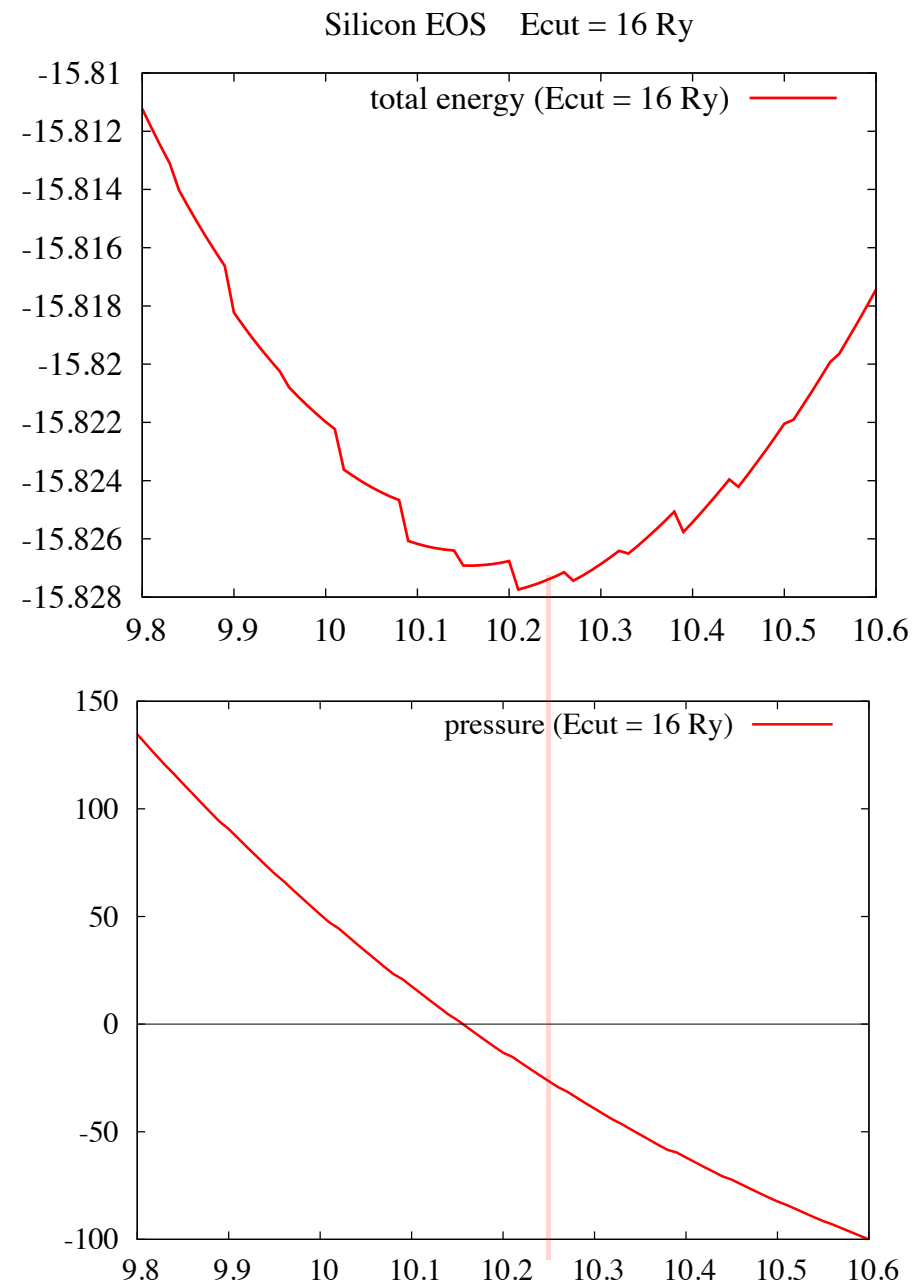


directly calculated from QE (Hellman-Feynman)
calculated as derivative of the energy
calculated from QE

Is there something wrong ?

STRESS (and PRESSURE)

... and constant energy cutoff does not mean constant NPW

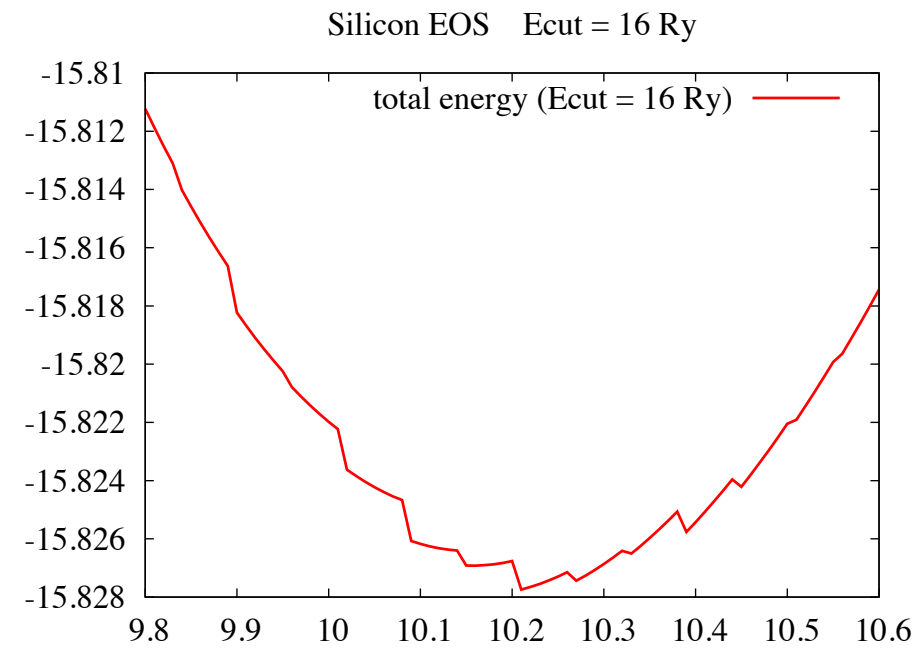
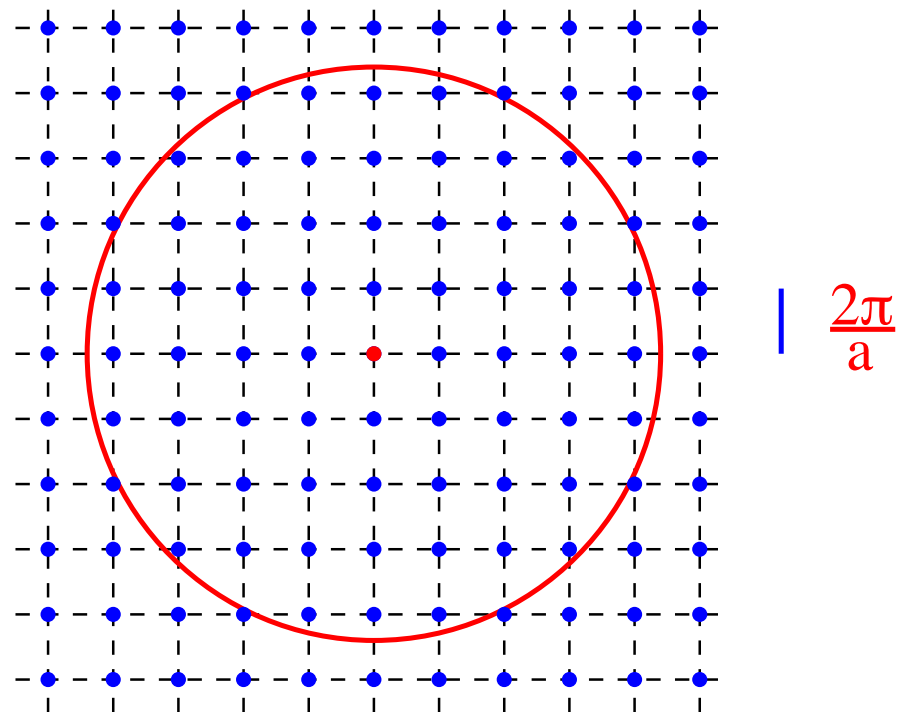


finer sampling in the lattice parameter range
emphasizes discontinuities
due to different size of the basis set
(NPW: number of plane waves)

results for a denser grid

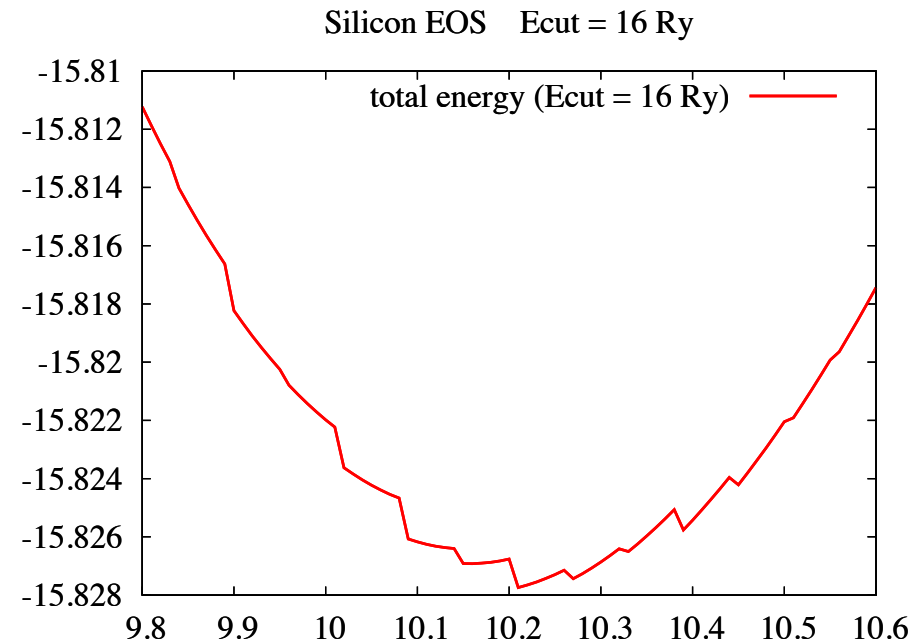
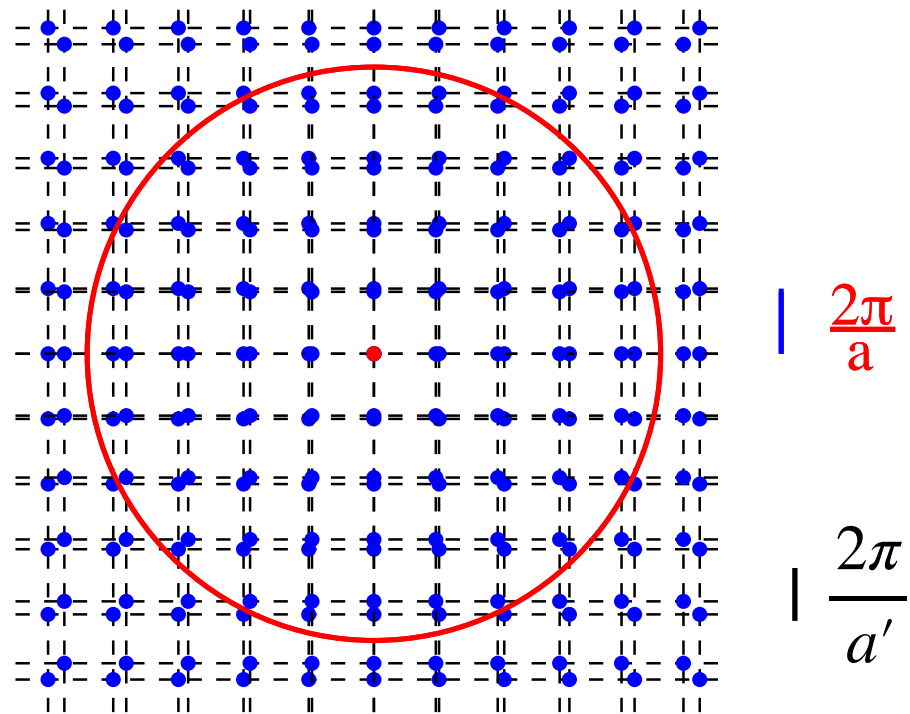
STRESS (and PRESSURE)

... and constant energy cutoff does not mean constant NPW



STRESS (and PRESSURE)

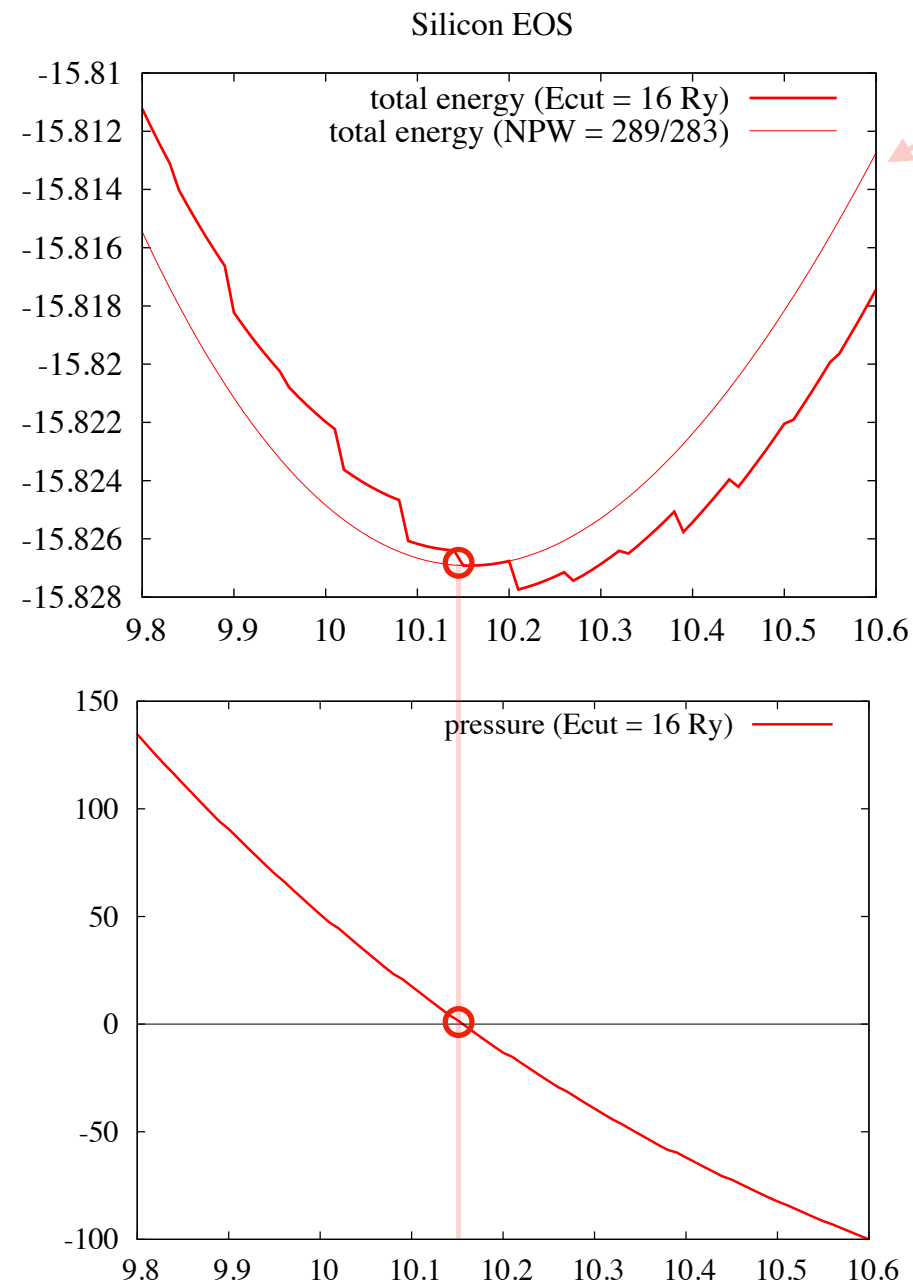
... and constant energy cutoff does not mean constant NPW



(larger a' => smaller k spacing => more NPW)

STRESS (and PRESSURE)

... and constant energy cutoff does not mean constant NPW



3) keep NPW (almost) fixed when changing celldm(1)

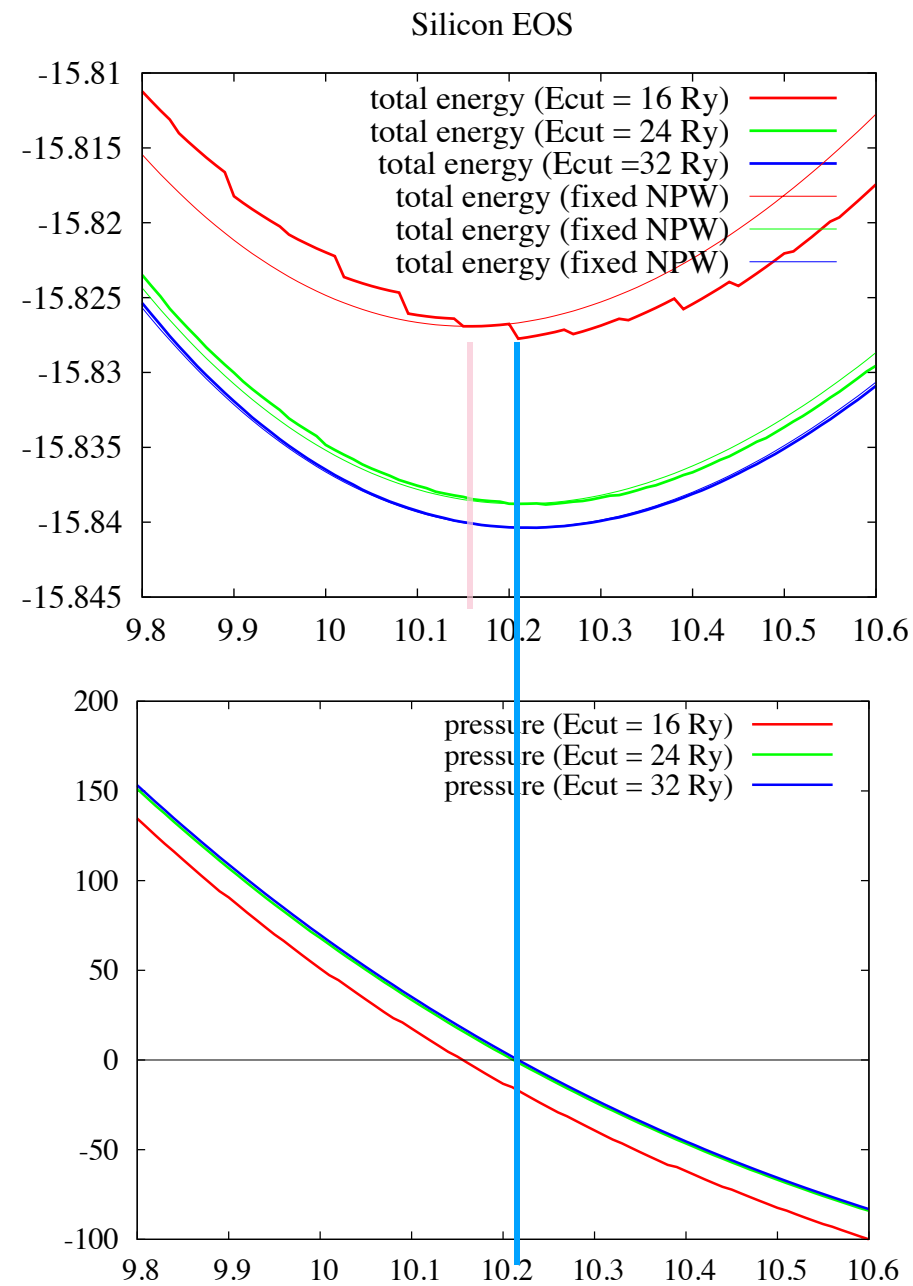
2) get NPW

1) find a (celldm(1)) corresponding to zero pressure

fixed Cutoff vs fixed Number of Plane waves

STRESS (and PRESSURE)

increasing the energy cutoff is better



fixed Ecut $\approx$ fixed NPW

$\min(E_{\text{tot}}) \approx \text{pressure}=0$

fixed Cutoff vs fixed Number of Plane waves

but don't worry too much....

The incompleteness of basis set is not as serious as it may look: energy *differences* between different structures, structural parameters such as lattice parameters and bond lengths, converge much quicker than absolute energies.

**other tools (not post-processing,
but nested with the scf cycle):**

Variable cell

How to perform variable-cell relaxation?

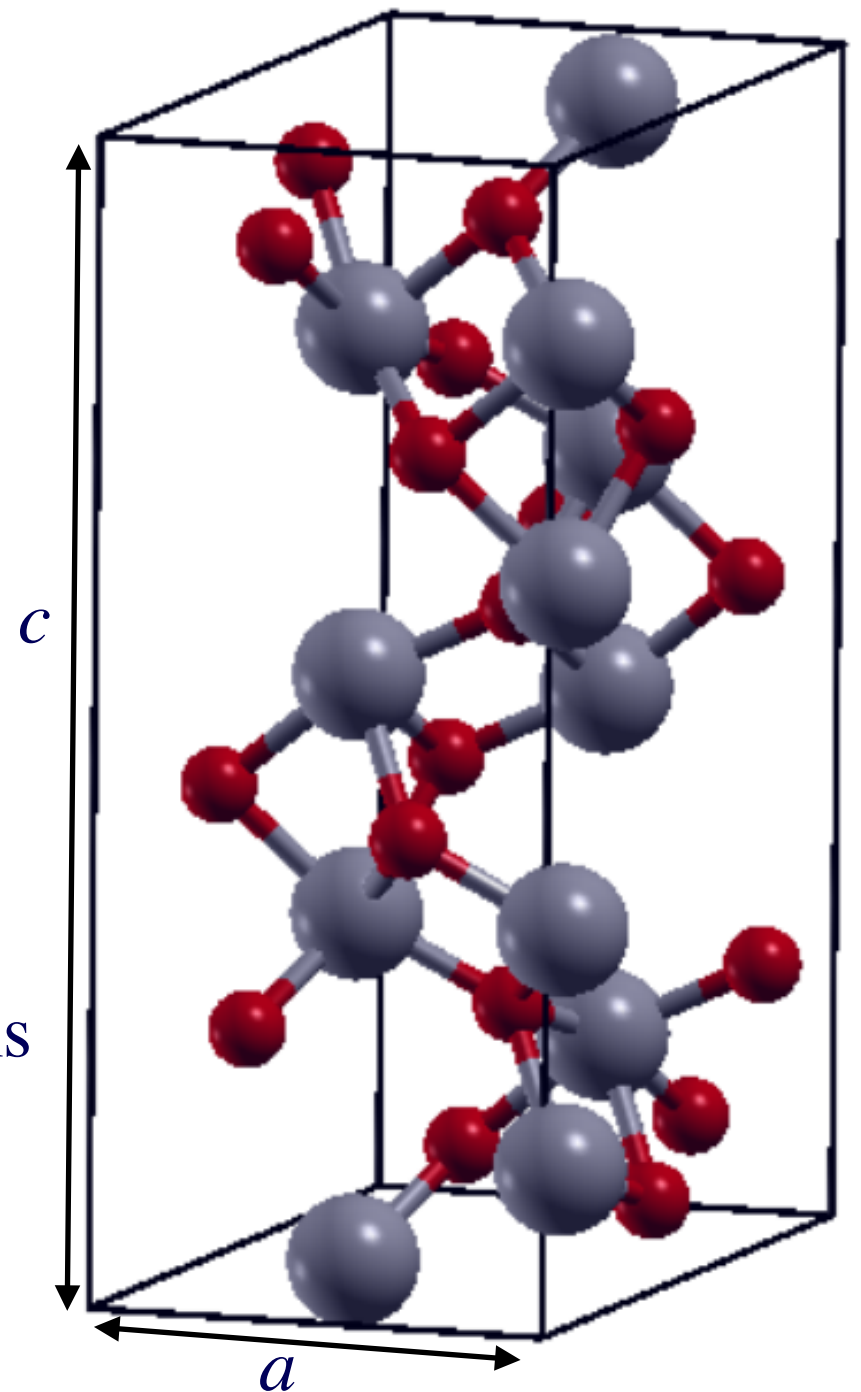
“by hands”: suppose we have a cell with 2 or more parameters (e.g., a and c) and we need to optimize them
 $\Rightarrow$

perform multiple scans over the parameters?

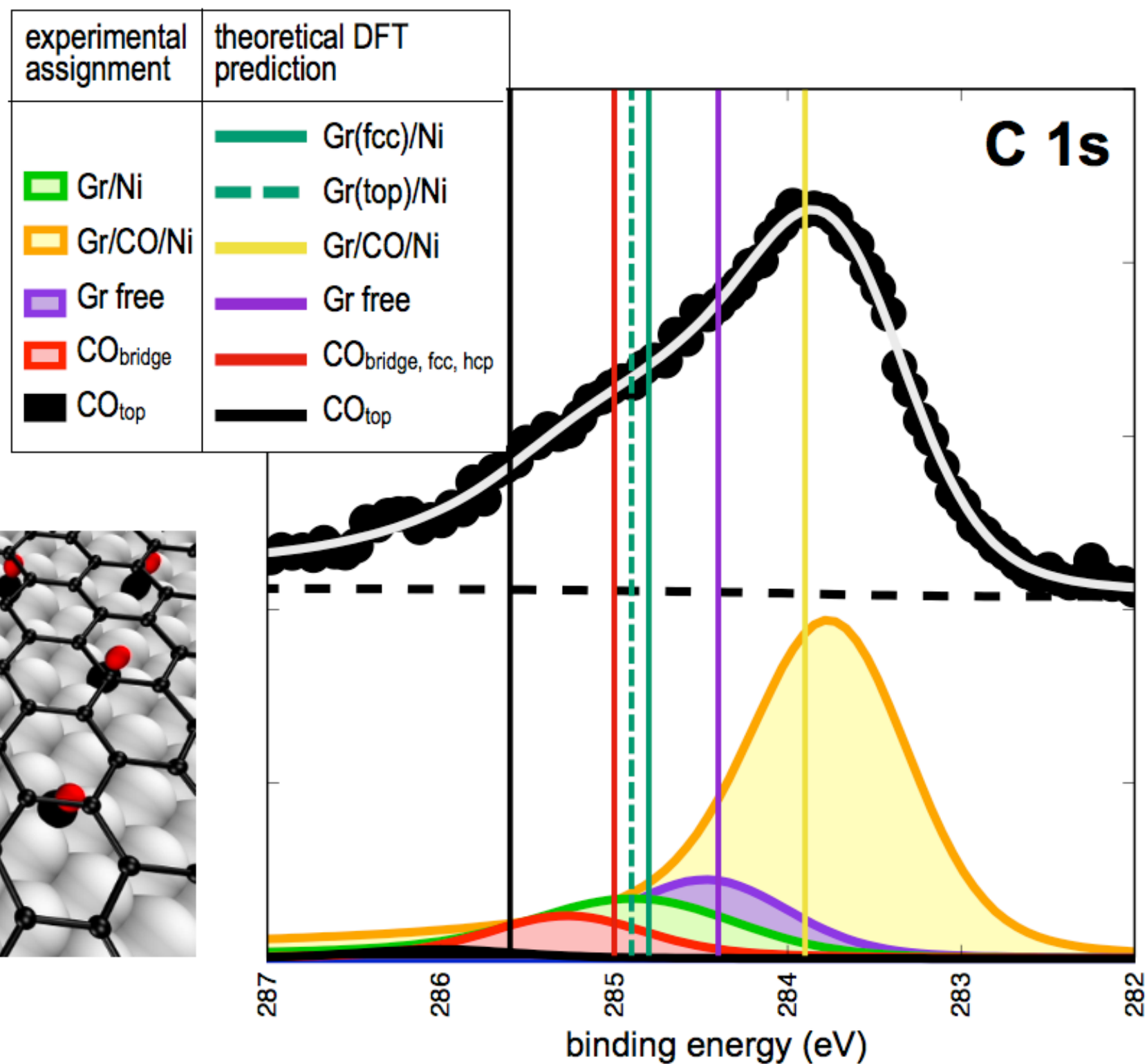
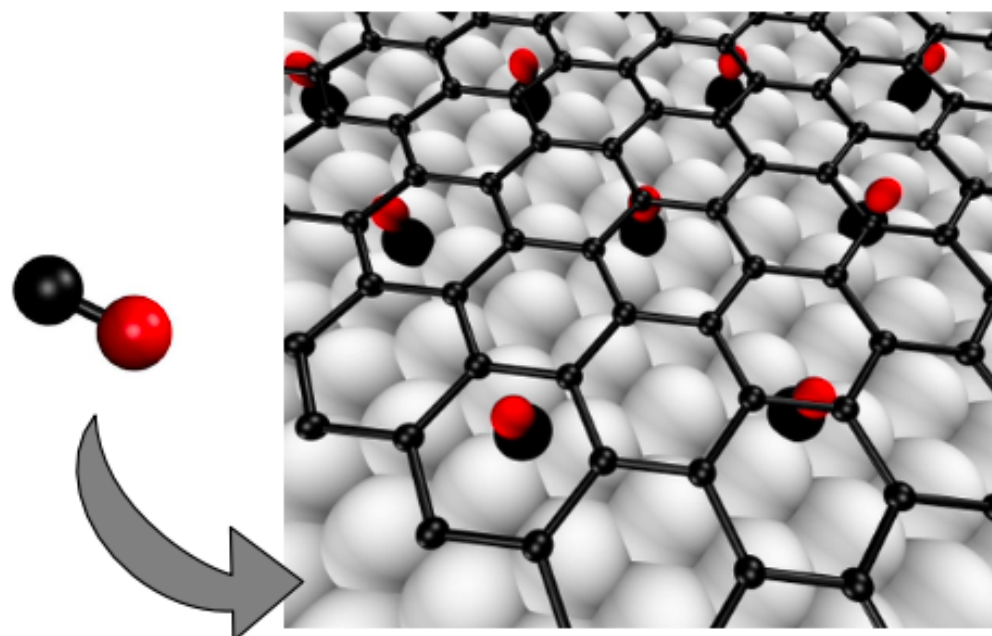
Variable-cell relaxation: in some cases this can be a more convenient option

- calculation = 'vc-relax'

Alumina (Al_2O_3): 30 atoms
(12 Al - grey ;
18 O - red)
hexagonal cell



core level shifts



core level shifts

calculated as the **binding energy difference** (=total energy differences for ground and excited states) between atoms in different chemical environments, using a core-excited pseudopotential to model the final state.

Final state approximation (FS or Δ SCF): uses a “core-hole” pseudop. where one electron is removed from the core (e.g. $1s$) of the target atom to calculate the total energy differences.

1. **Generate Core-Hole Potential:** Use specialized, core-excited pseudopotentials.
2. **SCF Calculation:** Run *pw.x* to obtain total energy with the core hole.
3. **Calculate Shifts:** Subtract the total energy of the reference system (e.g., bulk) from the target system (e.g., surface).

core level shifts

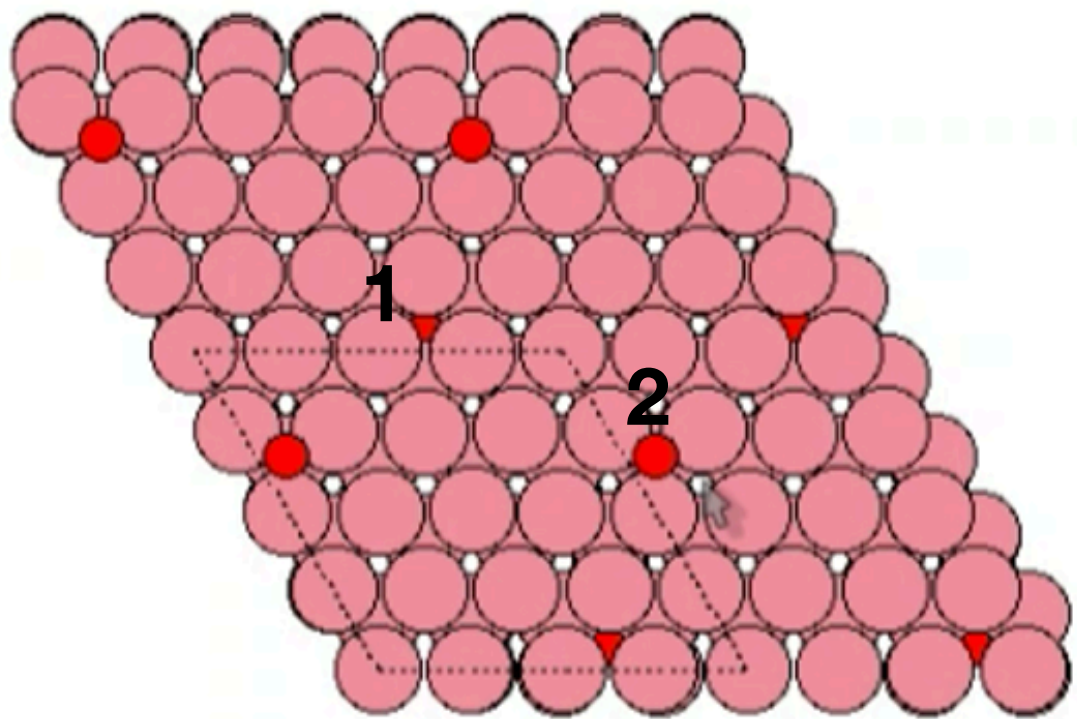
PSEUDOPOTENTIALS:

necessary to define a **normal pseudopotential** and a **core-excited one** for the atom of interest, **consistent with each other** (functional, parameters, ecc..), since the two pseudopotentials describe the **same atomic type**, only with a different, core-excited, electronic configuration.

E. Pehlke and M. Scheffler - <http://link.aps.org/doi/10.1103/PhysRevLett.71.2338>

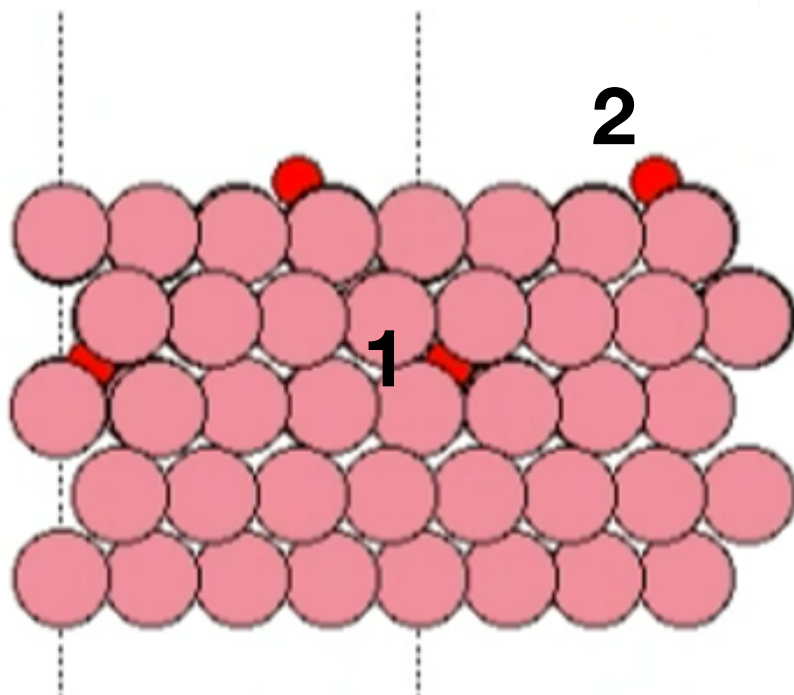
J. N. Andersen, D. Hennig, E. Lundgren, M. Methfessel, R. Nyholm, and M. Scheffler - <http://link.aps.org/doi/10.1103/PhysRevB.50.17525>

e.g. surface core level shifts (SCLS)



1=> in bulk

2=> on surf



1) Make a regular SCF calculation of the slab where the **core-excited** pseudopotential is used for the reference atom (e.g., **1**)

2) Make other SCF calculation for the selected atom (e.g. **2**) in which only this one is described by the **core-excited** pseudopotential (and 1 by normal pseudop.)

3) Calculate the ground state (GS) energy difference between the last SCF calculation (2) and the one for the reference atom (1).

This difference is the FS core-level shift.

core level shifts

We can doubt that differences in total energies using pseudopotentials are not correct...
In the following we justify the procedure, comparing pseudopotentials (PS) energies with all electron (AE) energies

core level shifts

Definizione di CLS tra configurazioni j e i : differenza tra *binding energies* "all electron" (AE) :

$$CLS(j - i) = \Delta BE(j - i) = BE^{AE}(j) - BE^{AE}(i) \quad (1)$$

con

$$BE^{AE}(i) = E_{tot}^{AE}(i^*) - E_{tot}^{AE}(i)$$

dove i e i^* indicano lo stesso sistema in configurazione "normale" (i) o "core excited" (i^*).

Introducendo quantità calcolate con pseudopotenziali (PS) possiamo riscrivere:

$$BE^{AE}(i) = BE^{PS}(i) + \Delta E_{at}^{AE-PS}(C1s^* - C1s)$$

con

$$BE^{PS}(i) = E_{tot}^{PS}(i^*) - E_{tot}^{PS}(i)$$

e

$$\Delta E_{at}^{AE-PS}(C1s^* - C1s) = E_{at}^{AE}(C1s^*) - E_{at}^{PS}(C1s^*) - [E_{at}^{AE}(C1s) - E_{at}^{PS}(C1s)]$$

che sono i conti atomici (nei nostri casi, sempre atomo di C isolato). Allora:

$$CLS(j - i) = BE^{PS}(j) - BE^{PS}(i) = E_{tot}^{PS}(j^*) - E_{tot}^{PS}(j) - [E_{tot}^{PS}(i^*) - E_{tot}^{PS}(i)] \quad (2)$$

perchè i termini di correzione atomica si annullano.

core level shifts

Riarrangiando i termini, scriviamo:

$$CLS(j - i) = E_{tot}^{PS}(j^*) - E_{tot}^{PS}(i^*) - [E_{tot}^{PS}(j) - E_{tot}^{PS}(i)]$$

da cui si vede che nel caso specifico in cui il sistema abbia la stessa configurazione "normale", non eccitata, l'ultimo termine si annulla e vale anche:

$$\boxed{CLS(j - i) = E_{tot}^{PS}(j^*) - E_{tot}^{PS}(i^*)} \quad (3)$$

The end