

DE LA RECHERCHE À L'INDUSTRIE



Structural Optimizations & Molecular Dynamics in ABINIT

ABINIT school 2019

Grégory GENESTE

Marc TORRENT

CEA, DAM, DIF, F-91297 Arpajon, France



1 – Introduction

2 – The keyword **ionmov**

- Structural Optimizations
- Molecular Dynamics

3 – The keyword **imgmov**

- Minimum Energy Paths
(=> *Advanced Feature on thursday/friday*)
- Path Integral Molecular Dynamics

4 – Conclusion

1 – Introduction

2 – The keyword **ionmov**

- Structural Optimizations
- Molecular Dynamics

3 – The keyword **imgmov**

- Minimum Energy Paths
(=> *Advanced Feature on thursday/friday*)
- Path Integral Molecular Dynamics


4 – Conclusion

General framework :

DFT formulated in the **Born-Oppenheimer (BO) Approximation**


Hamiltonian of the system of atoms :

$$H(\{\vec{R}_n\}, \{\vec{P}_n\}) = \sum_{n=1}^N \frac{\vec{P}_n^2}{2M_n} + \sum_{n < n'} \frac{Z_n Z_{n'} e^2}{4\pi\epsilon_0 \|\vec{R}_n - \vec{R}_{n'}\|} + E_e(\{\vec{R}_n\})$$


Kinetic energy
of the nuclei

Ion-ion (repulsive)
electrostatic energy

Energy of the **electron gas**
at FIXED positions of the nuclei
 $\{\vec{R}_n\}$


Potential energy
of the nuclei

$$V(\{\vec{R}_n\})$$

General framework :

DFT formulated in the **Born-Oppenheimer (BO) Approximation**

Hamiltonian of the system of atoms :

Electron gas : energy from DFT calculation

Which electronic state ?

$$E_e(\{\vec{R}_n\})$$

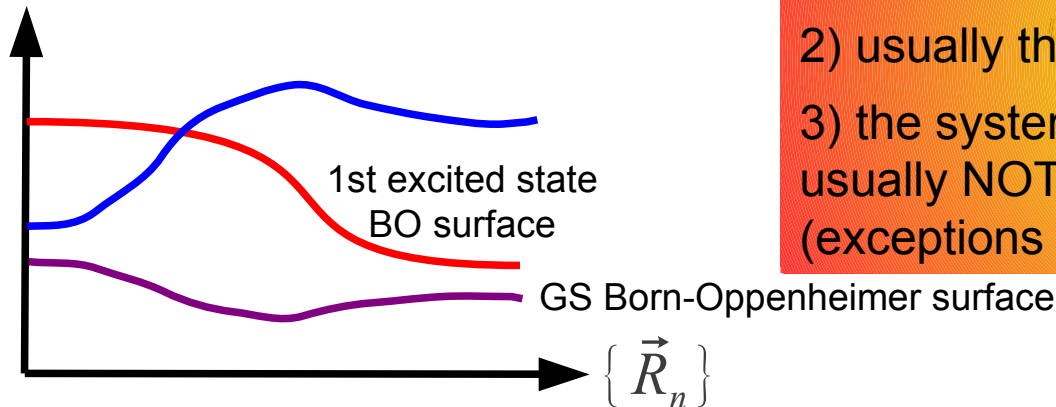
DFT : this is the **Ground State**, but

...
constrained electronic states possible (with **ocopt 2**)

Energy of the **electron gas**
at FIXED positions of the nuclei
 $\{\vec{R}_n\}$

of the nuclei

Electronic energy



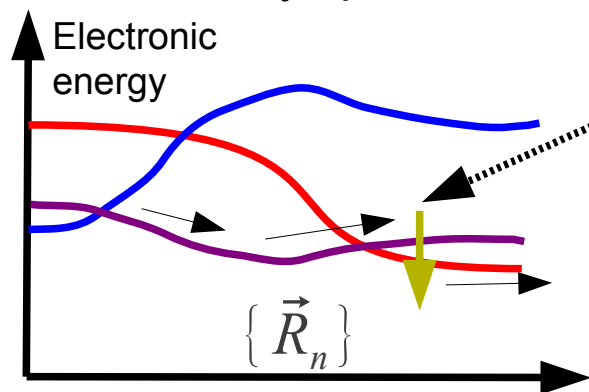
Algorithms that make atoms MOVE :

- 1) a priori, motion on a SINGLE BO surface
- 2) usually the **Ground State** surface
- 3) the system may change BO surface, but usually NOT wanted, (exceptions possible for some algos)

1 - Introduction

Example 1 : Molecular dynamics

=> MD always performed in a *STATISTICAL ENSEMBLE* (NVE, NVT, NPT, etc)



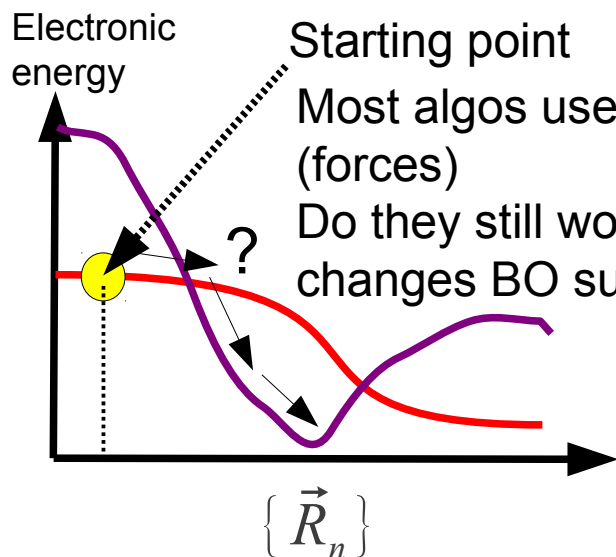
At that point, imagine the system is changing BO surf.

Energy not conserved, forces discontinuous !

=> The statistical ensemble is LOST !

=> necessity for MD to remain on the same BO surface

Example 2 : Structural optimization



Most algos use the GRADIENT (forces)

Do they still work if the system changes BO surf ?

Non spin-polarized system :

=> no problem for staying on the GS BO surface, a priori...

Spin-polarized system :

You can constrain the electronic state using **occopt 2** (& **occ**)

Rq : in DFT+U, this constraint can be achieved on the occupation matrices of correlated orbitals (**usedmatpu**, **dmatpawu**)

Calculation of the FORCES : the HELLMAN-FEYNMAN theorem

We suppose that $|\Psi(x)\rangle$ is an eigenvector of $H(x)$, both function of a parameter x
The corresponding eigenvalue is

$$E(x) = \langle \psi(x) | H(x) | \psi(x) \rangle$$

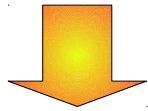
The we have :

$$\frac{dE}{dx} = \langle \psi(x) | \frac{dH}{dx} | \psi(x) \rangle$$

=> the force can be evaluated directly from the electronic calculation of $|\Psi(x)\rangle$,
No need to know the derivative of $|\Psi(x)\rangle$

ABINIT : many algorithms allowing ions to move
=> controlled by one among **TWO keywords** :

ionmov

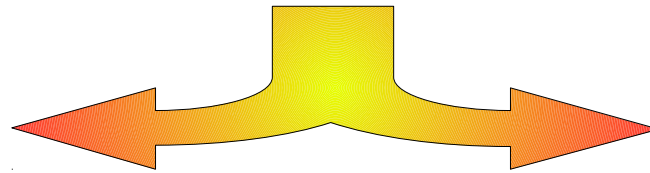


The algorithm **need not replicas**
of the system

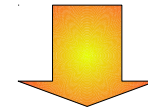
=> **Structural Optimizations**

=> **Molecular Dynamics**

=> ...



imgmov



The algorithm **involves replicas**
of the system

=> **Minimum Energy Paths**
(MEPs)

=> **Path Integral**
Molecular Dynamics (PIMD)

=> ...

1 – Introduction

2 – The keyword **ionmov**

- Structural Optimizations
- Molecular Dynamics

3 – The keyword **imgmov**

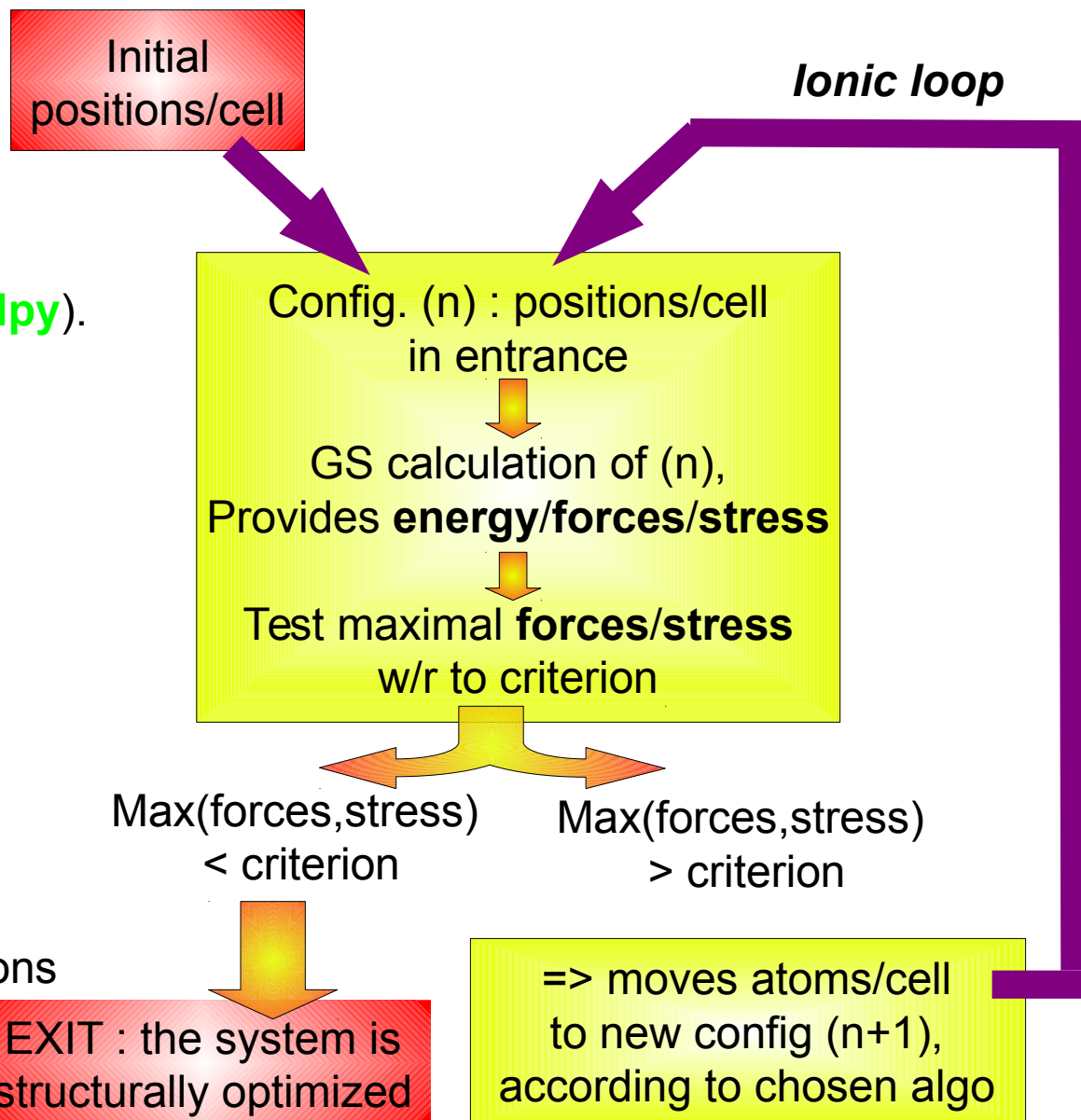
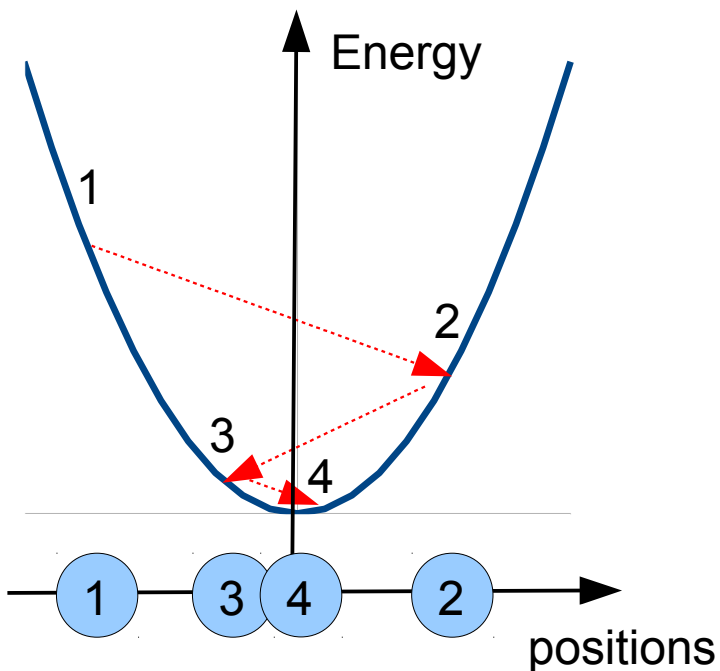
- Minimum Energy Paths
(=> *Advanced Feature on thursday/friday*)
- Path Integral Molecular Dynamics

4 – Conclusion

2 – ionmov : structural optimizations

Principle :

starting from an initial set of atomic positions, modifies these positions (**possibly also the cell**) to minimize the energy (**enthalpy**).



2 – ionmov : structural optimizations

Numerical parameters needed :

- initial positions/cell of the system : try to make it as close as possible to the solution !

- maximal number of steps (it may never converge!)

ntime

- Stopping criterion on Maximal forces/stress

tolmxf

- choose an algorithm

Value of **ionmov**

Initial positions/cell

Ionic loop

Config. (n) : positions/cell in entrance

GS calculation of (n). Provides **energy/forces/stress**

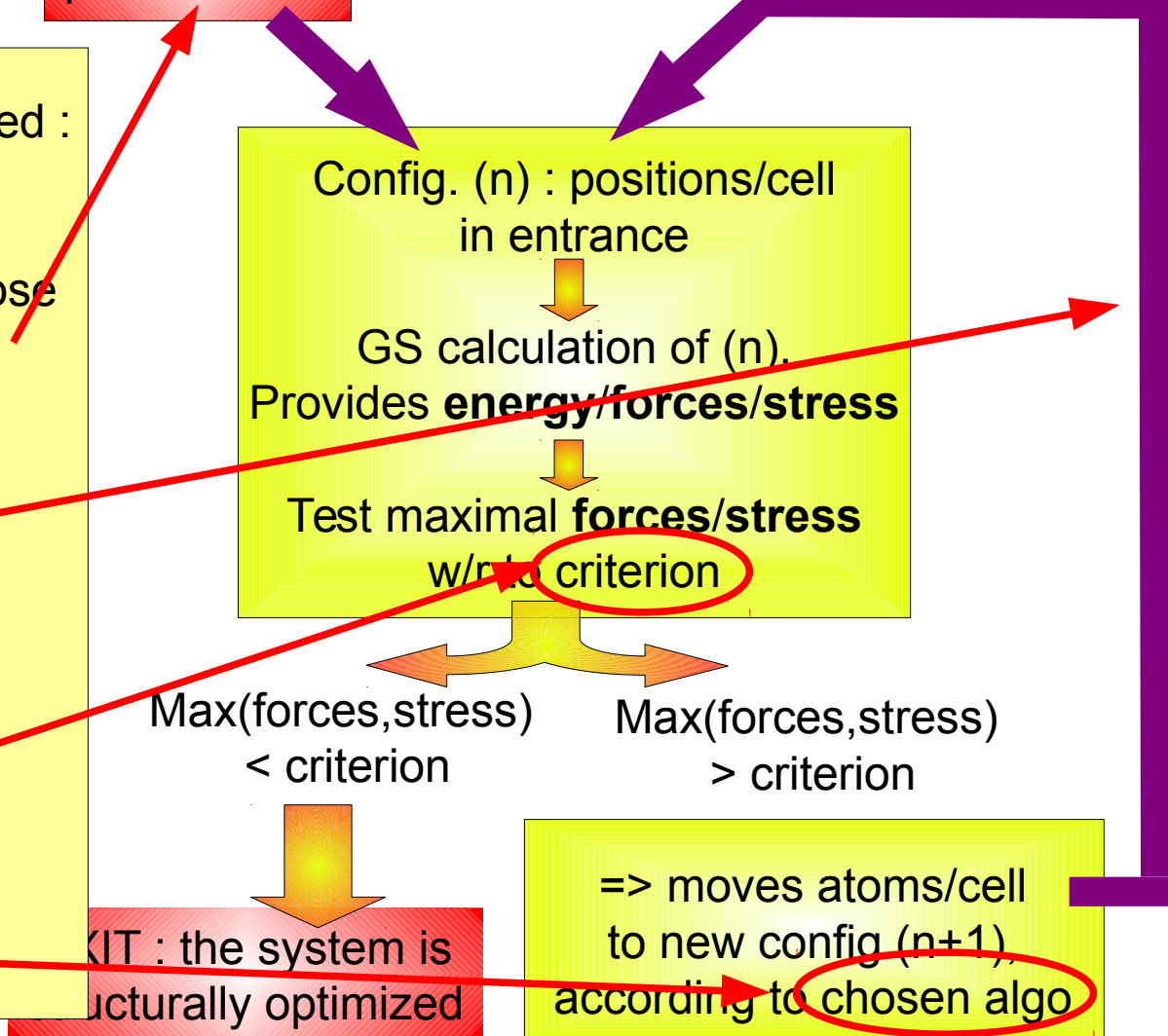
Test maximal **forces/stress** w/r to **crit**erion

Max(forces, stress) < crit

Max(forces, stress) > crit

IT : the system is structurally optimized

=> moves atoms/cell to new config (n+1), according to **chosen algo**



Algorithms :

Many algorithms do exist : **Speepest-Descent, Conjugate Gradient,**

In ABINIT : Broyden-Fletcher-Goldfarb-Shanno minimization (BFGS) algorithm

Very efficient minimization algorithm :

Uses not only the GRADIENT (forces) but also (approximations of) the HESSIAN B_k to predict the direction to follow.

2 – ionmov : structural optimizations

ABINIT Keywords :

Broyden-Fletcher-Goldfarb-Shanno minimization (BFGS) algorithm :

ionmov 2 or **3** or **22**

Number of optimization steps (ionic loop) : **ntime**

Stopping criterion – on maximal forces : **tolmxf** (in Ha/bohr : default 5.0×10^{-5})

Optimize the atomic positions ONLY (**optcell 0**) or also the cell (**optcell \neq 0**)

optcell 1: homothetic dilatation of the cell to optimize volume only

optcell 2 : all degrees of freedom optimized

optcell 3 : optimize under constant volume

optcell 4 ... 9 : optimize one or two lattice vectors

You can continue a previous run using **restartxf**

If **optcell \neq 0**, criterion on stresses : **strfact** (default=100)

Target stress : **strtarget** (Voigt notations)

Other keywords : **ecutsm**, **dilatmx**

Rq : **initial SYMMETRIES are constrained along the optimizations**

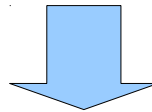
(one can release this constraint by setting **nsym 1**)

Performing properly a structural optimization :

Structural optimization AND electronic ground state calculation are in relation:

At each step,

Max of forces/stress is compared to stopping criterion



To make sense, the numerical precision on forces/stress must be \ll criterion = **tolmxf**

Precision : depends on convergence of the GS calculation

ABINIT possesses a GS convergence criterion related to forces : **toldff** (or **tolrff**)

Typically you may use,
(very safe value)

toldff = **tolmxf**/50
(or **tolrff** = 1/50 = 0.02)

Ex : **tolmxf** 5.0d-05 (default)
toldff 1.0d-06

GS criterion for structural optimization (and MD) :

Preferred : **toldff**, **tolrff**

Also possible : **tolvrs**

In some cases, forces are always zero by symmetry => **tolvrs** or **toldfe**

NOW CORRECTED !
toldff/tolrff also work

2 – ionmov : structural optimizations

Example of input file for structural optimization (ATOMIC POSITIONS ONLY)

```
## CaZrO3: structural optimization at fixed lattice vectors ##

##### The system #####
acell 5.583 8.007 5.759 Angstrom #expt latt con
natom 20
znucl 20 40 8
ntypat 3
typat 4*1 4*2 12*3

xred
..... # here follows the 20 atomic positions in reduced coord.

##### parameters for GS calculations #####
ixc 11 #GGA-PBE
ecut 18.0 #plane-wave cut-off
pawecutdg 30.0 #cut-off double grid
kptopt 1
ngkpt 4 3 4 #k-point mesh sampling the Brillouin Zone
toldff 4.0d-06 ## tolmxf/50
nstep 100 # maximal number of iterations for GS convergence
occopt 1 #default: insulator
nband 100

##### structural optimization #####
ionmov 2 #Broyden algo
optcell 0 #lattice vectors kept fixed (this is the default value)
ntime 200
tolmxf 2.0d-04 #stopping criterion for struct. opt. in Ha/bohr
```

tolmxf and toldff should be chosen in coherence with each other

**toldff =
tolmxf/50**



2 – ionmov : structural optimizations

Example of output :
H₂ molecule

iter	Etot (hartree)	deltaE(h)	residm	nres2	diffor	maxfor
ETOT 1	-1.1626798387780	-1.163E+00	3.879E-13	2.503E-04	1.160E-03	1.973E-04
ETOT 2	-1.1626810426665	-1.204E-06	1.070E-14	2.829E-05	1.446E-04	5.263E-05
ETOT 3	-1.1626809824101	6.026E-08	2.566E-11	4.586E-07	6.917E-05	1.653E-05
ETOT 4	-1.1626809602542	2.216E-08	9.907E-14	4.447E-08	3.945E-07	1.614E-05
ETOT 5	-1.1626809603685	-1.143E-10	1.098E-13	1.132E-08	9.116E-07	1.523E-05
ETOT 6	-1.1626809672017	-6.833E-09	6.748E-14	3.727E-09	5.323E-07	1.470E-05
ETOT 7	-1.1626809772975	-1.010E-08	8.807E-15	1.216E-09	4.866E-07	1.421E-05
ETOT 8	-1.1626809845912	-7.294E-09	4.658E-15	4.987E-10	2.656E-07	1.394E-05
ETOT 9	-1.1626809876569	-3.066E-09	1.527E-15	2.564E-10	1.694E-07	1.377E-05
ETOT 10	-1.1626809902953	-2.638E-09	1.839E-14	1.009E-10	1.393E-07	1.363E-05

At SCF step 10, forces are converged :
for the second time, max diff in force= 1.393E-07 < toldff= 2.000E-07

Final maximal force = 1.36d-05 Ha/bohr : this is a VERY LOW value !
does it make sense ?

Yes because the GS is converged using **toldff** = 2.0d-07 !

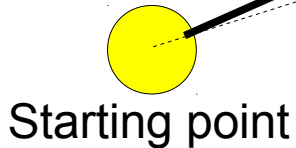
=> the numerical precision on forces at the end of the SCF
loop is ~ 2.0d-07 Ha/bohr

2 – ionmov : structural optimizations

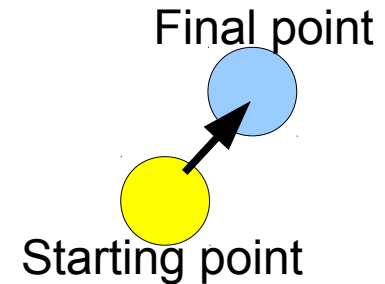
The keyword **tolrff** for GS convergence

The idea : if the initial geometry is FAR from the minimum, you need not converge the first SCF cycles with a high precision, but you need to refine convergence when you approach the minimum !

Starting point far from the minimum :
=> **rough** estimation of the force
is **sufficient** to go in the good direction
+ forces large !



Final point



Starting point close to the minimum :
=> **precise** estimation of the force
is **necessary** to go in the good direction
+ forces small !

Solution : **tolrff**

Stops the electronic loop when : $\text{diffor}/\text{maxfor} < \text{tolrff}$ (twice consecutively)

Suggested value = 0.02

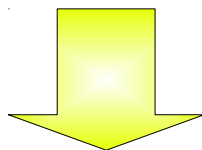
2 – ionmov : structural optimizations

Example :

Initial config :

iter	Etot(hartree)	deltaE(h)	residm	nres2	diffor	maxfor
ETOT 1	-660.31969297315	-6.603E+02	4.540E-01	1.799E+02	6.460E-01	6.460E-01
ETOT 2	-662.93461378251	-2.615E+00	1.436E-02	1.068E+02	4.883E-01	8.833E-01
ETOT 3	-662.84457843461	9.004E-02	3.872E-03	2.298E+01	2.150E-01	6.695E-01
ETOT 4	-662.85676465636	-1.219E-02	4.155E-03	1.258E+02	2.835E-01	5.390E-01
ETOT 5	-662.80945102193	4.731E-02	1.940E-03	1.907E+01	1.625E-01	6.201E-01
ETOT 6	-662.79564881801	1.380E-02	6.608E-04	2.164E+00	1.021E-01	5.902E-01
ETOT 7	-662.79164620269	4.003E-03	1.519E-04	2.598E-01	2.972E-02	5.725E-01
ETOT 8	-662.79180014724	-1.539E-04	8.317E-06	1.035E-01	9.571E-03	5.644E-01
ETOT 9	-662.79189607734	-9.593E-05	3.928E-06	1.063E-01	7.093E-03	5.601E-01

At SCF step 9, forces are sufficiently converged :
 for the second time, max diff in force= 7.093E-03 is less than < tolrff= 2.000E-02 times max force



Last config :

iter	Etot(hartree)	deltaE(h)	residm	nres2	diffor	maxfor
ETOT 1	-664.27218782796	-6.643E+02	2.233E-06	2.186E-04	4.815E-04	5.324E-04
ETOT 2	-664.27218776245	6.551E-08	2.412E-10	2.710E-05	5.763E-04	3.568E-04
ETOT 3	-664.27218770753	5.492E-08	3.384E-09	3.335E-06	3.378E-04	6.656E-05
ETOT 4	-664.27218770225	5.283E-09	4.361E-10	4.611E-07	3.349E-05	4.602E-05
ETOT 5	-664.27218770055	1.703E-09	4.847E-11	1.388E-07	1.558E-05	5.053E-05
ETOT 6	-664.27218769856	1.986E-09	1.680E-11	3.846E-08	6.561E-06	5.031E-05
ETOT 7	-664.27218769719	1.374E-09	1.210E-11	1.657E-08	6.143E-06	4.844E-05
ETOT 8	-664.27218769715	3.956E-11	9.212E-13	1.090E-08	1.853E-06	4.800E-05
ETOT 9	-664.27218769731	-1.612E-10	9.455E-13	6.016E-09	1.609E-06	4.709E-05
ETOT 10	-664.27218769833	-1.020E-09	6.241E-12	2.022E-09	2.365E-06	4.604E-05
ETOT 11	-664.27218769952	-1.191E-09	8.622E-12	1.466E-09	4.689E-07	4.587E-05
ETOT 12	-664.27218770090	-1.378E-09	7.484E-12	8.249E-10	4.205E-07	4.565E-05

At SCF step 12, forces are sufficiently converged :
 for the second time, max diff in force= 4.205E-07 is less than < tolrff= 2.000E-02 times max force

Here :
tolmxf 5.0d-05
tolrff 0.02

CONSTRAINTS :

1) *Fixing atoms or coordinates*

All coordinates : **natfix** (number of fixed atoms), **iatfix** (indices of the fixed atoms)

Or just one coordinate :

natfixx, **iatfixx**

natfixy, **iatfixy**

natfixz, **iatfixz**

2) *Fixing linear relations between atomic coordinates*

Number of such relations : **nconeq**

Atoms involved : **iatcon**

Coeff. of the linear combination : **wtatcon**

2 – ionmov : structural optimizations, optcell ≠ 0

Fixed pressure structural optimizations : optcell=2
= all degrees of freedom relaxed under symmetry constraints (those of the initial configuration)

Ex : T=0 K equation of state of a solid

In conjunction with **ionmov=2**

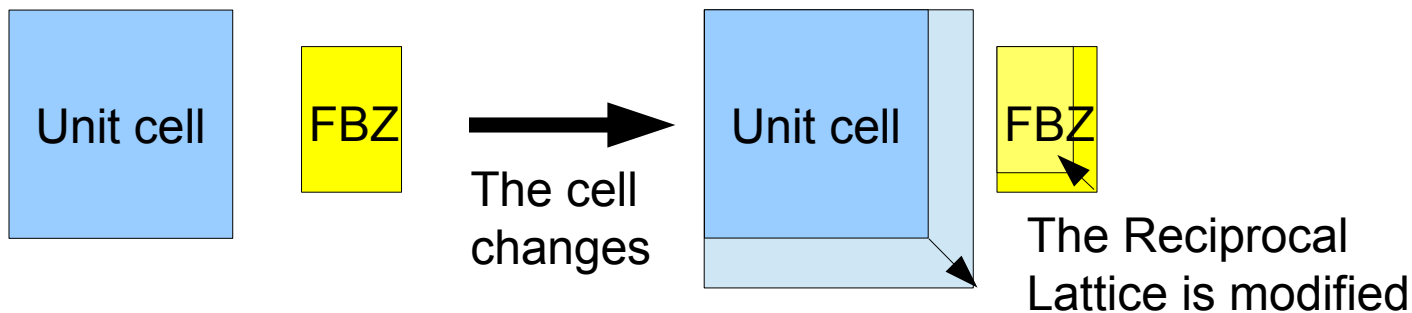
Stopping criterion on stresses : ABINIT uses **tolmxf/strfact**

Default : **strfact**=100 bohr²

For **tolmxf**=5.0d-05 Ha/bohr => max stress = 5.0d-07 Ha/bohr³

optcell=2 performs the optimization under SYMMETRY constraints, i.e. the space group of the starting configuration is maintained.

A few additional keywords have to be specified when **optcell** ≠ 0 :



2 – ionmov : structural optimizations, optcell ≠ 0

=> **G-vectors of the Reciprocal Lattice can get in or out the plane-wave sphere :**

1) may cause discontinuities of the energy :

These discontinuities are **SMOOTHED** by using **ecutsm** ≠ 0

Recommended value : **ecutsm** = 0.5 (Ha)

Rq: **ecutsm** ≠ 0 is compulsory to perform cell optimization

2) A maximal permitted dilatation of the lattice vectors has to be defined, thanks to **dilatmx**

Rq : a too large value of **dilatmx** causes waste of memory and CPU time !

3) the final stress tensor can be specified using **strtarget** (Voigt notations)

Ex : suppose you want to optimize a crystal under an hydrostatic pressure of 15 GPa

This corresponds to a stress tensor of

$$\sigma = \begin{bmatrix} -0.000509839 & 0.0 & 0.0 \\ 0.0 & -0.000509839 & 0.0 \\ 0.0 & 0.0 & -0.000509839 \end{bmatrix}$$

strtarget -0.000509839 -0.000509839 -0.000509839 0.0 0.0 0.0 (in Ha/bohr³)

2 – ionmov : structural optimizations, optcell ≠ 0

=> **G-vectors of the Reciprocal Lattice can get in or out the plane-wave sphere :**

1) may cause discontinuities of the energy :

These discontinuities are **SMOOTHED** by using **ecutsm** ≠ 0

Recommended value: **ecutsm = 0.5 (Ha)**

Rq : **ecutsm** ≠ 0 is compulsory to perform cell optimization

2) A maximal permitted dilatation of the lattice vectors has to be defined

thanks to **strtarget**

Rq : a too large value of **dilatmx** causes waste of memory and CPU time

3) the final stress tensor can be specified using **strtarget** (Voigt notations)

Ex : suppose a hydrostatic pressure of 15 GPa

This corresponds to a stress tensor of

$$\sigma = \begin{bmatrix} -0.000509839 & 0.0 & 0.0 \\ 0.0 & -0.000509839 & 0.0 \\ 0.0 & 0.0 & -0.000509839 \end{bmatrix}$$

strtarget -0.000509839 -0.000509839 -0.000509839 0.0 0.0 0.0 (in Ha/bohr³)

2 – ionmov : structural optimizations, optcell ≠ 0

1st example : optimize the lattice constants of CaZrO3 in GGA-PBE

```

#### CaZrO3: structural optimization at zero pressure ####

##### The system #####
acell 5.583 8.007 5.759 Angstrom #expt latt con
natom 20
znucl 20 40 8
ntypat 3
typat 4*1 4*2 12*3

xred
..... # here follows the 20 atomic positions in reduced coord.

##### parameters for GS calculations #####
ixc 11 #GGA-PBE
ecut 18.0 #plane-wave cut-off
pawecutdg 30.0 #cut-off double grid
kptopt 1
ngkpt 4 3 4 #k-point mesh sampling the Brillouin Zone
toldff 4.0d-06 ## tolmxf/50
nstep 100 # maximal number of iterations for GS convergence
occopt 1 #default: insulator
nband 100

##### structural optimization #####
ionmov 2 #Broyden algo
optcell 2 #lattice vectors kept fixed (this is the default value)
ntime 200
tolmxf 2.0d-04 #stopping criterion for struct. opt. in Ha/bohr
ecutsm 0.5
dilatmx 1.1
strtarget 0.0 0.0 0.0 0.0 0.0 0.0 ## optional because 0 = default value
    
```

Take previous input
Change **optcell** from 0 to 2
Add :
ecutsm 0.5
dilatmx 1.1 (allows max dilatation of 10%)
If dilatmx exceeds 1.15, add
chkdilatmx 0
Add optionnally :
strtarget (default= 0 0 0 0 0 0)

**toldff =
tolmxf/50**

2 – ionmov : structural optimizations, optcell ≠ 0

2nd example : T=0 K EOS of FeH₃

```
##### T=0K equation of state of FeH3 #####
chkysmbreak 0
```

```
##### The system ## ### FeH3 #####
```

```
acell 3*4.69
natom 4
ntypat 2
znucl 26 1
typat 1 2 2 2
xred
0.0 0.0 0.0
0.5 0.5 0.0
0.5 0.0 0.5
0.0 0.5 0.5
```

FeH₃=cubic crystal,
space group *Pm-3m*
Forces are ZERO
by symmetry !
Here **toldfe** used, but
now you can use
toldff or **tolrff**

```
##### parameters for GS calculations #####
```

```
ixc 11 #GGA-PBE
ecut 30.0 pawecutdg 40.0
kptopt 1 ngkpt 24 24 24
nstep 100
toldfe 1.0d-12
occup 3 #metallic occupation
tsmear 0.001
nband 60
```

=> use alternative
GS criterion
toldfe or **tolvrs**

```
##### structural optimization #####
```

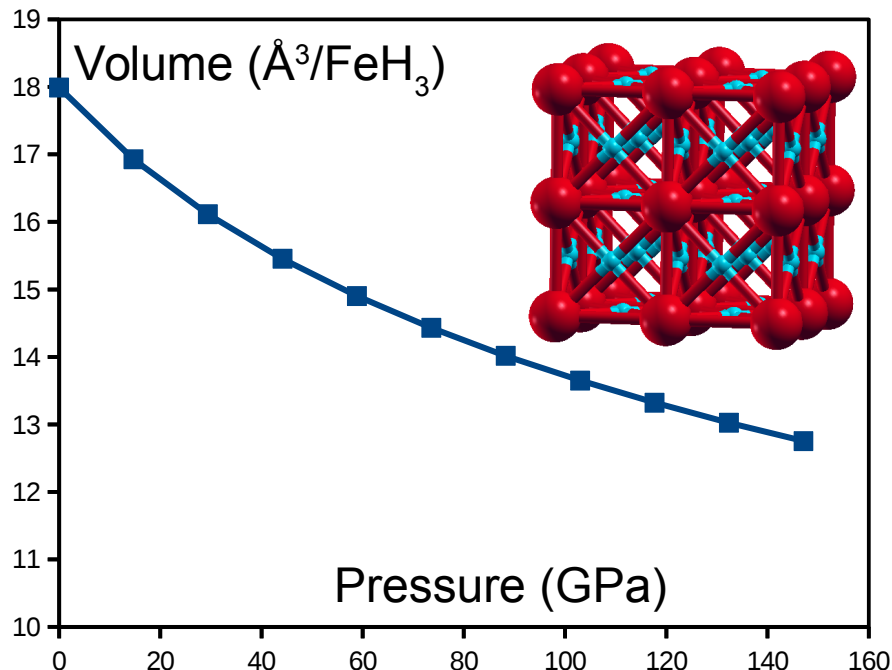
```
ionmov 2
optcell 2
ntime 200
tolmxf 1.0d-05
ecutsm 0.5
dilatmx 1.3 chkdilatmx 0
```

getcell -1 getxred -1

Takes final positions/cell of dataset n
to initialize dataset n+1

```
ndtset 11
strtarget1 -0.0050 -0.0050 -0.0050 0.0 0.0 0.0
strtarget2 -0.0045 -0.0045 -0.0045 0.0 0.0 0.0
strtarget3 -0.0040 -0.0040 -0.0040 0.0 0.0 0.0
strtarget4 -0.0035 -0.0035 -0.0035 0.0 0.0 0.0
strtarget5 -0.0030 -0.0030 -0.0030 0.0 0.0 0.0
strtarget6 -0.0025 -0.0025 -0.0025 0.0 0.0 0.0
strtarget7 -0.0020 -0.0020 -0.0020 0.0 0.0 0.0
strtarget8 -0.0015 -0.0015 -0.0015 0.0 0.0 0.0
strtarget9 -0.0010 -0.0010 -0.0010 0.0 0.0 0.0
strtarget10 -0.0005 -0.0005 -0.0005 0.0 0.0 0.0
strtarget11 0.0000 0.0000 0.0000 0.0 0.0 0.0
```

*Multidataset
mode to
compute
several
pressures*



1 – Introduction

2 – The keyword **ionmov**

- Structural Optimizations

- Molecular Dynamics

3 – The keyword **imgmov**

- Minimum Energy Paths

(=> *Advanced Feature on thursday/friday*)

- Path Integral Molecular Dynamics

4 – Conclusion

2 – ionmov : Molecular Dynamics

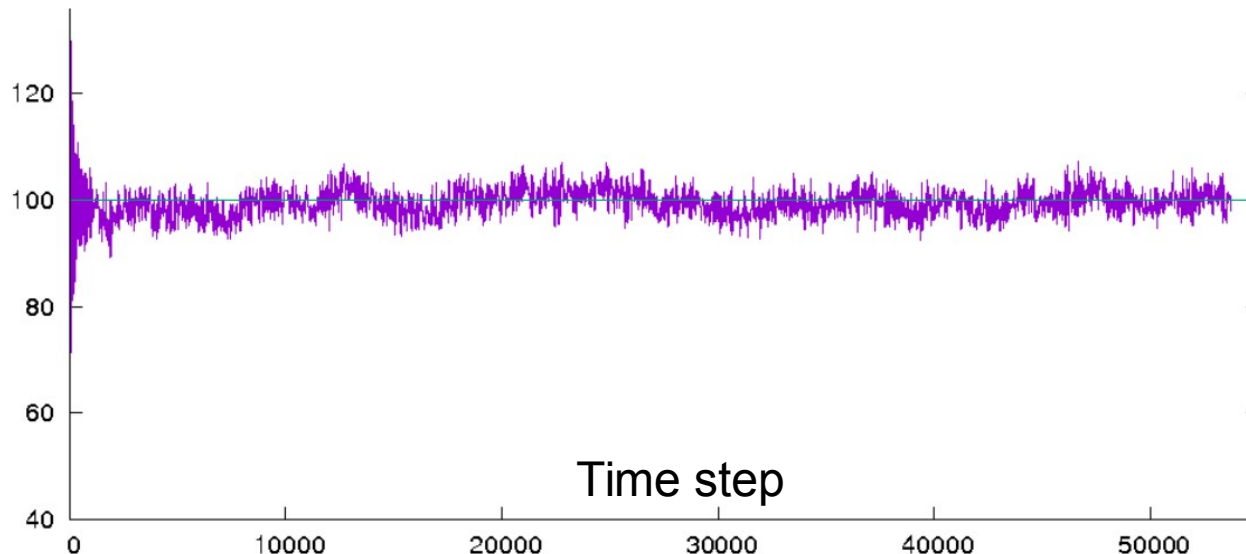
Molecular Dynamics : numerical technique in **Classical Statistical Physics**

Allows **exploring Phase Space**, produces a trajectory characteristic of a **Statistical Ensemble**.

After an **equilibration period**, the observables *fluctuate around a mean value*
= that of thermodynamic equilibrium

Hypothesis : the system must be **ERGODIC**, i.e. the time-averages quantities are equal to the ensemble-average ones.

Example :
instantaneous
temperature



2 – ionmov : Molecular Dynamics

Molecular dynamics : the ions move according to the **Newton equations of motion**

$$m_i \frac{d^2 \vec{r}_i}{dt^2} = \vec{f}_i = - \frac{\partial V}{\partial \vec{r}_i}$$

usually : GS BO surface

Force : computed by Hellman-Feynman Theorem, from DFT ground state calculation

**= AB INITIO Molecular Dynamics
(Quantum Molecular Dynamics)**

Rq : the nuclei remain **CLASSICAL particles**

To start a MD simulation, you **ABSOLUTELY** need to specify :

nsym 1

- initial positions and cell (**xred/acell/rprim**)
- initial velocities (**vel**), usually controlled by an initial temperature => **mdtemp(1)**
- the kind of MD (see below), controlled by the value of **ionmov**
- number of time steps **ntime**
- atomic masses **amu**

2 – ionmov : Molecular Dynamics

Molecular Dynamics at Thermodynamic Equilibrium (with classical atoms)

TWO goals :

THERMODYNAMICS

Compute **static thermodynamic quantities**

= ensemble averages of some microscopic quantities

Ex : energy, enthalpie, $g(r)$, density of proba of atomic positions, volume ...

TIME does not need to have physical meaning

You can use **any mass you want** !¹

MD here does same job as MC =
Explore configuration space according to the probability density of the chosen statistical ensemble : NVE, NVT, NPT ...

KINETICS

Compute **Time-correlation functions**
= ensemble averages of some microscopic quantities

Ex : diffusion coefficient, susceptibility as a function of frequency ...

TIME must have correct physical meaning

You must use **the true mass** !

NVE preferred

¹ see e.g. Hopkins et al, JCTC **11**, 1864 (2015) and references therein

2 – ionmov : Molecular Dynamics

To be integrated, the Newton equations of motion are **DISCRETIZED**

=> We need a **time step**, specified in ABINIT by the keyword **dtion**

(expressed in atomic times units, i.e. 1 a.u. of time $\sim 2.4 \times 10^{-17}$ s)

Typical value of dtion : 100 or below (determined by maximal phonon frequency)

VERLET algorithm :

$$m_i \frac{d^2 \vec{r}_i}{dt^2} = \vec{f}_i = -\frac{\partial E_0}{\partial \vec{r}_i} \quad \longrightarrow \quad \vec{r}_i(t + dtion) = 2\vec{r}_i(t) - \vec{r}_i(t - dtion) + \frac{dtion^2}{m_i} \vec{f}_i(t)$$

At time step 1 : **Taylor** algorithm

$$\vec{r}_i(dtion) = \vec{r}_i(0) + \vec{v}_i(0) \times dtion + \frac{1}{2} \frac{dtion^2}{m_i} \vec{f}_i(0)$$

Initial velocities (**vel** or **mdtemp(1)**)

At each time step : electronic GS recalculated ! => provides **FORCES & ENERGY**

Electronic loops : convergence using **toldff** preferred (also possible : **tolvrs**)

An alternative = « velocity Verlet »

Many other algorithms implemented, see the website at input variable **ionmov**

2 – ionmov : Molecular Dynamics

MD can be performed with ABINIT in several ways :

- **NVE ensemble (microcanonical)** : $m_i \frac{d^2 \vec{r}_i}{dt^2} = \vec{f}_i$ **ionmov 6**
the total energy is fixed

The volume (and more generally the supercell) is fixed

- **NVT ensemble (canonical)** : one uses a **THERMOSTAT**
(=additional degrees of freedom that exchange energy with the system to maintain the temperature fluctuating around a fixed value).

The rate of exchange is controlled by a parameter (« mass » or friction coeff) :

- **Nose-Hoover** thermostat **ionmov 8**

Nose « mass » : **noseinert**

- **Langevin** thermostat : **ionmov 9**

stochastic thermostat, uses random numbers

$$m \frac{d^2 \vec{r}_i}{dt^2} = \vec{f}_i - \gamma m \frac{d \vec{r}_i}{dt} + \vec{R}_i$$

friction

Langevin force,
drawn in a gaussian $\sqrt{\frac{2\gamma m k_B T}{\delta t}}$
with variance

mdtemp(1) = initial temperature (a random distribution of velocities is generated)

mdtemp(2) = final temperature (at time step *ntime*)

Thermostat temperature evolves linearly inbetween

- **Isokinetic ensemble** (rescaling of velocities) : **ionmov 12**

Temperature maintained strictly constant

MD can be used also to perform **STRUCTURAL optimizations** :

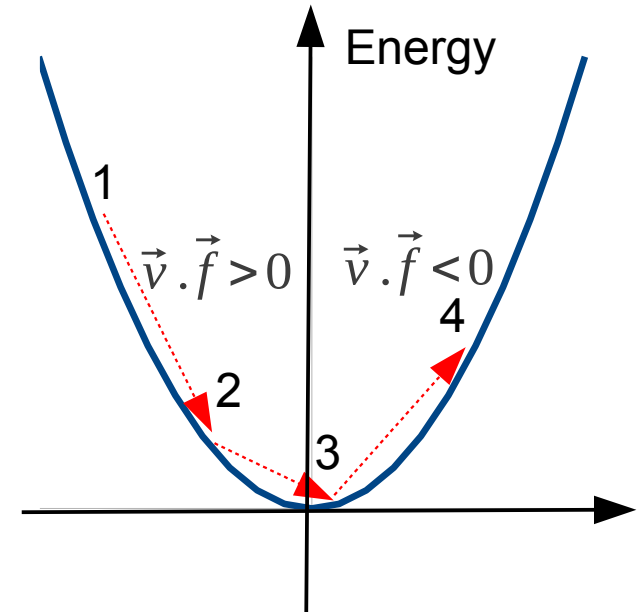
- « **Quenched** » molecular dynamics **ionmov 7**

The velocities are set to zero as soon as the force and the velocity have opposite direction (because it means that you have gone too far and are beyond the minimum...)

Very efficient way to pump kinetic energy and go to the minimum, but can be long if your starting point is far from the minimum.

- « **Damped** » molecular dynamics **ionmov 1**

Molecular Dynamics with a viscous damping (**vis**).



1 – Introduction

2 – The keyword **ionmov**

- Structural Optimizations
- Molecular Dynamics

3 – The keyword **imgmov**

- Minimum Energy Paths

(=> Advanced Feature on thursday/friday)

- Path Integral Molecular Dynamics

4 – Conclusion

Algorithms for the determination of a Minimum Energy Path (MEP)
= NEB or « string » method

Not treated in detail this part :
see special session on thursday/friday

Only a few words here :

String method : **imgmov** 2

NEB : **imgmov** 5

Number of images along the path : **nimage**

Maximal number of steps : **ntimimage**

Tolerance criterion for convergence : **tolimg**

Parallelization over images : **npimage** (recommended = **nimage**)

1 – Introduction

2 – The keyword **ionmov**

- Structural Optimizations
- Molecular Dynamics

3 – The keyword **imgmov**

- Minimum Energy Paths
(=> *Advanced Feature on thursday/friday*)

- Path Integral Molecular Dynamics

4 – Conclusion

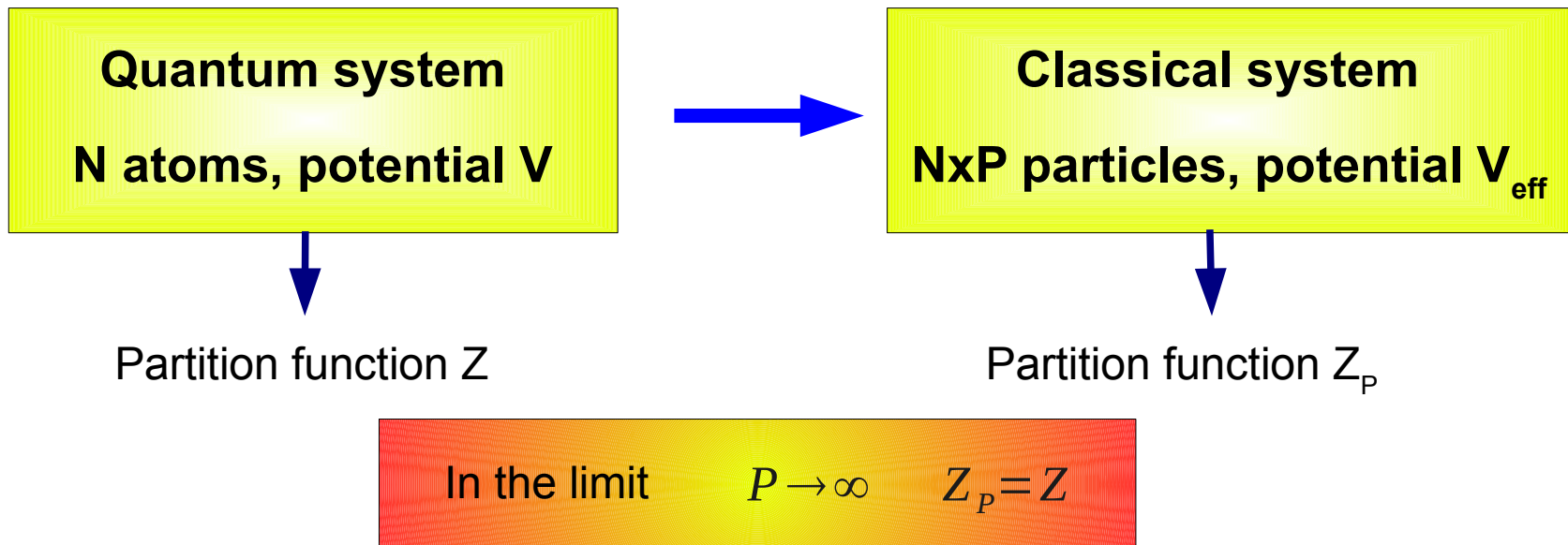
3 – imgmov : Path Integral Molecular Dynamics

Describes the **quantum effects** associated with the **nuclei motions**

With following assumptions :

- **thermal equilibrium** (described a statistical mixing – quantum canonical ensemble)
- **discernible** nuclei (*Boltzmann* statistics)
- no information on **time correlation functions**

Based on an analogy between a quantum system of nuclei and a fictitious system of classical particles in which **each nucleus is replicated P times** :



3 – imgmov : Path Integral Molecular Dynamics

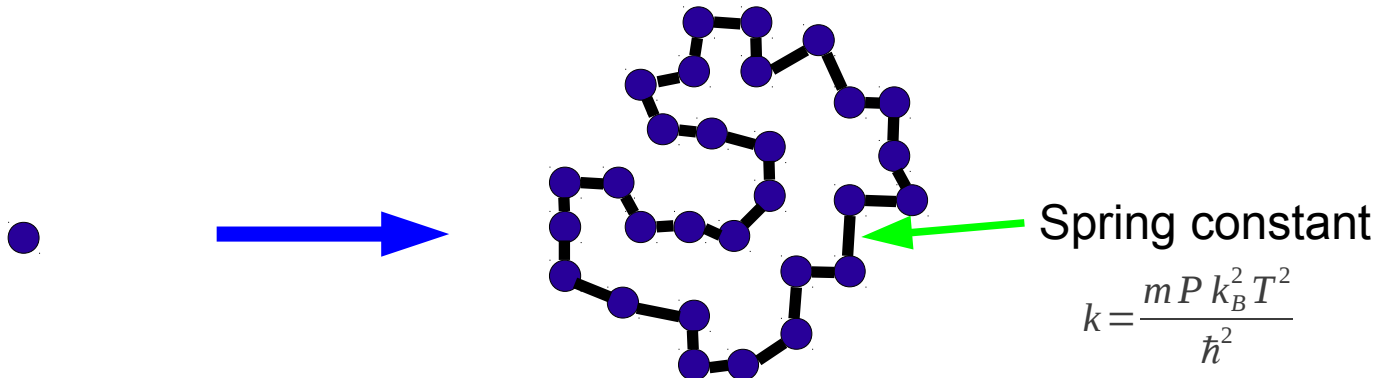
$$Z = \lim_{P \rightarrow \infty} \left[\frac{2\pi m P k_B T}{h^2} \right]^{3NP/2} \int_{\vec{R}_1} \dots \int_{\vec{R}_P} e^{-\beta V_{eff}(\vec{R}_1 \dots \vec{R}_P)} d\vec{R}_1 \dots d\vec{R}_P$$

$$\vec{R}_s = (\vec{r}_1^{(s)} \dots \vec{r}_N^{(s)}) \quad 3N\text{-vector}$$

$$V_{eff}(\vec{R}_1 \dots \vec{R}_P) = V_{eff}(\vec{r}_1^{(1)} \dots \vec{r}_N^{(1)} \dots \vec{r}_1^{(P)} \dots \vec{r}_N^{(P)})$$

$$V_{eff}(\vec{R}_1 \dots \vec{R}_P) = \sum_{s=1}^P \left[\sum_{i=1}^N \frac{1}{2} k(P, \beta) (\vec{r}_i^{(s)} - \vec{r}_i^{(s+1)})^2 + \frac{1}{P} V(\vec{r}_1^{(s)} \dots \vec{r}_N^{(s)}) \right]$$

1 – replace each (quantum) nucleus by a closed ring of P replicas (beads, imaginary time slices) bonded to each other by a harmonic attraction :



2 – In each slice, the images interact through the physical potential (divided by P)

3 – The system of the NxP replicas is treated **classically**, by MD = **PIMD**

3 – imgmov : Path Integral Molecular Dynamics

Time has no physical meaning in standard PIMD !

It just allows to system to explore phase space and sample it according to the probability distribution of the desired statistical ensemble (here canonical).

=> Time-correlation functions are not accessible.

In ABINIT, PIMD equations are integrated using the Verlet algorithm.

Warning: temperature is controlled by the keyword **mdtemp**

mdtemp(1) = initial temperature

mdtemp(2) = thermostat temperature

There is no linear evolution between the two (in contrast to ionmov 9)

To perform PIMD, do not forget to set **nsym** 1 (like in MD)

3 – imgmov : Path Integral Molecular Dynamics

Main keywords for PIMD in ABINIT :

imgmov=9 (PIMD with Langevin thermostat), 13 (Nose-Hoover chains)

nimage : Trotter number = number of slices

ntimimage : number of time steps

dtion : time steps in atomic units (1 a.u. of time $\sim 2.4 \times 10^{-17}$ s)

npimage : keyword to parallelize over images

pitransform : controls the coordinate transformation

(0 : primitive ; 1 : normal mode ; 2 : staging)

amu(ntypat) : real masses in atomic mass units (proton =1)

pimass(ntypat) : fictitious masses (same unit as amu)

dynimage(nimage) : 1 if the image is free to move, 0 if it is fixed

(for PIMD, use nimage x 1, or let to the default value)

prtvoling : controls the volume of printed information

mdtemp(2) : array of two values to control the temperature

mdtemp(1) = initial temperature ; mdtemp(2) = thermostat temperature

restartxf : -1 to restart automatically from `_HIST` file written on disk

Langevin thermostat :

vis (friction coeff for Langevin thermostat in a.u.) : 5.0d-05 is usually OK

irandom : choice of the random number generator (use 3 for non-deterministic

Random number generator)

3 – imgmov : Path Integral Molecular Dynamics

Performing PIMD in the canonical ensemble :

*Preferred way at the moment : **USE LANGEVIN THERMOSTAT (imgmov 9)***

Example of input file : H2 molecule at 200 K, canonical ensemble

```
##### H2 molecule in a box : example of PIMD input #####
##### The system #####
acell 20.0 20.0 20.0 Angstrom
natom 2 znuc1 1 ntypat 1 typtat 1 1

xangst_1img
0.0 0.0 0.0
0.75 0.0 0.0

xangst_lastimg #all images at the same position initially
0.0 0.0 0.0
0.75 0.0 0.0

##### electronic GS parameters #####
ixc 11 #GGA-PBE
ecut 20.0 #plane-wave cut-off
pawecutdg 30.0 #cut-off double grid
kptopt 1
ngkpt 1 1 1 nshiftk 1 shiftk 0 0 0 # Gamma point only
toldff 1.0d-06
nstep 100 # maximal number of iterations for GS convergence
occopt 1 #default: insulator
nband 1

nsym 1 ##### VERY IMPORTANT

##### PIMD #####
imgmov 9 #Langevin thermostat
nimage 64 #nb of beads
dynimage 64*1 #this is the default value for PIMD
vis 5.0d-05 #friction Langevin : recommended value
dtion 20 #time step
ntimimage 10000
mdtemp 400 200 #initial T=400K, thermostat T=200 K
amu 1.0 #physical mass (here hydrogen, e.g. use 2 for deuterium)
irandom 3 #non-deterministic random number generator, advised
optcell 0 #no cell evolution, this is the default

pitransform 0 #no coord transformation
                #1 for normal mode ; 2 for staging
pimass 1.0 #fictitious mass
                #automatically fixed to the correct value if pitransform=1 or 2

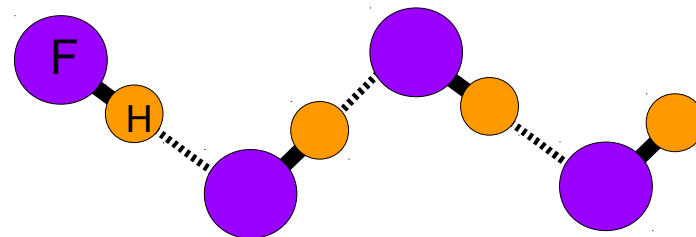
restartxf -1 #restart from _HIST file written on disk (see below)

##### parallelization (if any), here over 64 cores, one per bead
paral_kgb 1
npkpt 1
npband 1
npfft 1
bandpp 1
npimage 64
```

3 – imgmov : Path Integral Molecular Dynamics

Example : symmetrization of hydrogen bond in HF at low temperature ($T=100$ K)
(*H. Dammak, F. Briec, G. Geneste, M. Torrent, M. Hayoun, PCCP accepted 2019*)

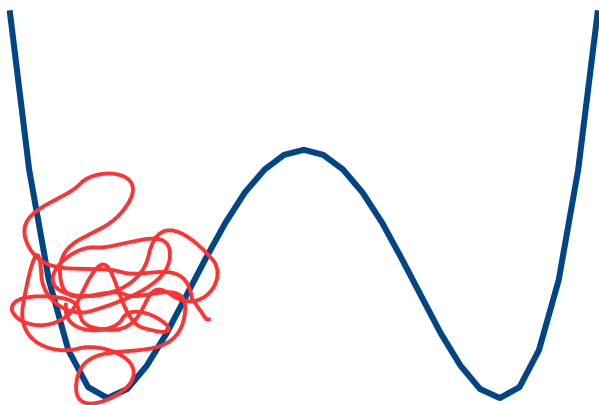
Barrier for hopping ~ 30 meV.



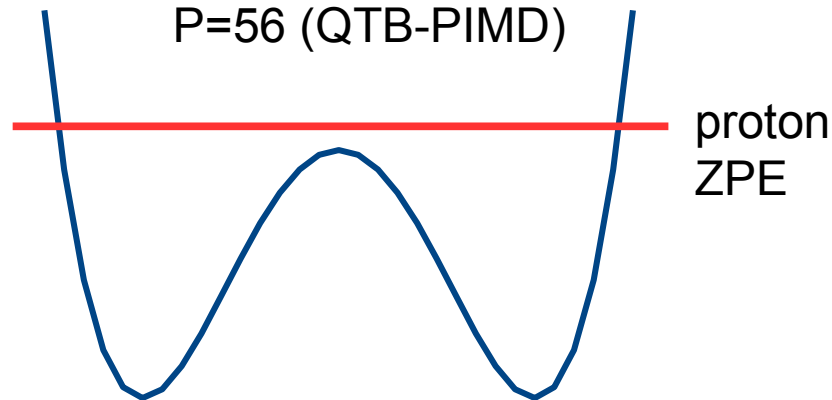
Langevin thermostat (**imgmov** 9)

- Classical simulation $T=100$ K : no proton jump
- QTB-PIMD/PIMD simulation $T=100$ K : the ZPE lies above the proton barrier
=> H-bond symmetrization !

Classical atoms
P=1



Quantum atoms
P=64, 128 (PIMD)
P=56 (QTB-PIMD)



3 – imgmov : Path Integral Molecular Dynamics

imgmov 9 can be used also to perform **CLASSICAL MD** by simply setting **nimage** to 1
=> Advised way to make Classical MD with Langevin thermostat

At each PIMD time step, ABINIT prints (after GS calculation of all images) :

PATH-INTEGRAL MOLECULAR DYNAMICS (LANGEVIN):
Moving images of the cell...

At PIMD time step 10, the **temperature** is 387.66856 K

Energy:

Internal energy (PRIMITIVE estimator) = -1.156702170 Ha

Internal energy (VIRIAL estimator) = -1.158625897 Ha

Stress tensor from PRIMITIVE estimator (Ha/Bohr³):

-0.000014616	-0.000005422	0.000003090
-0.000005422	0.000025695	0.000003793
0.000003090	0.000003793	0.000005807

Pressure (primitive estimator) = -0.165607799 GPa

Center of mass:

0.7086472998	0.0000000000	0.0000000000
--------------	--------------	--------------

Atomic positions:

xred			
-0.0009056794	-0.0021311176	0.0010474810	
0.1830400967	0.0026994680	0.0107059589	
xred_2img			
0.0123610044	0.0097888890	-0.0009567539	
0.1755687527	0.0236438047	-0.0045914081	
xred_3img			
0.0001715794	-0.0473470595	-0.0041760251	
0.1701962959	-0.0434734851	-0.0232401971	
xred_4img			
-0.0033910323	0.0572334113	0.0092800439	
0.1716062825	-0.0004139108	0.0119309005	

Velocities:

vel			
0.0001563291	0.0001049462	0.0000971409	
0.0000297633	0.0001299715	0.0003940452	
vel_2img			
-0.0001516894	0.0002730760	-0.0000649509	
0.0005404062	0.0008878152	-0.0002122851	
vel_3img			
-0.0002223423	-0.0018146927	-0.0001225370	
-0.0001028884	-0.0016686690	-0.0009454436	
vel_4img			
0.0000179793	0.0022523886	0.0003877888	
-0.0002675578	-0.0001648357	0.0004662416	

Atomic positions and velocities for all beads

(here P=4 and natom=2)

Can be used to restart the dynamics by putting xred_1img, xred_2img... vel_1img, vel_2img ... in input file

3 – imgmov : Path Integral Molecular Dynamics

imgmov 9 can be used also to perform **CLASSICAL MD** by simply setting **nimage** to 1
=> Advised way to make Classical MD with Langevin thermostat

At each PIMD time step, ABINIT prints (after GS calculation of all images) :

PATH-INTEGRAL MOLECULAR DYNAMICS (LANGEVIN):
Moving images of the cell...

At PIMD time step 10, the **temperature** is 387.66856 K

Energy:

Internal energy (PRIMITIVE estimator) = -1.156702170 Ha

Internal energy (VIRIAL estimator) = -1.158625897 Ha

Atomic positions:

xred

-0.0009056794 -0.0021311176 0.0010474810

0.1830400967 0.0026994680 0.0107059589

xred_2img

0.0123610044 0.0097888890 -0.0009567539

0.1755687527 0.0236438047 -0.0045914081

xred_3img

0.0001715704 0.0172470595 0.0041760251

Stress tensor from PRIMITIVE

-0.000014616 -0.000000000

-0.000005422 0.000000000

0.000003090 0.000000000

Pressure (primitive estimator)

Center of mass:

0.7086472998 0.000000000

About energies in PIMD :

Two values are given, using two estimators :

$$E^{(prim)} = \frac{3}{2} N P k_B T - \left\langle \sum_{s=1}^P \sum_{i=1}^N \frac{1}{2} m_i \omega_P^2 (\vec{r}_i^{(s+1)} - \vec{r}_i^{(s)})^2 \right\rangle + \frac{1}{P} \left\langle \sum_{s=1}^P V(\vec{r}_1^{(s)} \dots \vec{r}_N^{(s)}) \right\rangle$$

$$E^{(vir)} = \frac{3}{2} N k_B T + \frac{1}{P} \left\langle \sum_{s=1}^P \sum_{i=1}^N \frac{1}{2} (\vec{r}_i^{(s)} - \vec{r}_i^c) \cdot (-\vec{F}_i^{(s)}) \right\rangle + \frac{1}{P} \left\langle \sum_{s=1}^P V(\vec{r}_1^{(s)} \dots \vec{r}_N^{(s)}) \right\rangle$$

$$\vec{r}_i^c = \frac{1}{P} \sum_{s=1}^P \vec{r}_i^{(s)}$$

Important point : the kinetic part is computed using the **thermostat temperature**, not the instantaneous one ;
thus do not be surprised if, using NVE (**vis**=0), the energy does not seem to be conserved...

3 – imgmov : Path Integral Molecular Dynamics

restartxf -1 : the next run automatically starts from the last configuration of the previous one, and continues Verlet integration, by reading the `_HIST` file

One known (minor) problem : with **restartxf** -1 and **irandom** 3, ABINIT rewrites at each restart incorrect velocities (and thus incorrect temperature), but the trajectory is correct. Positions, forces, stress in `_HIST` are OK. If you use this option, rather takes instantaneous temperature in ABINIT output files (as written when calculated at the first time).

Alternatively for restart : take manually positions and velocities of previous run.

Warning : internal energy computed using thermostat temperature, not instantaneous one

Alternatively, restarts can be made by using last positions and velocities :

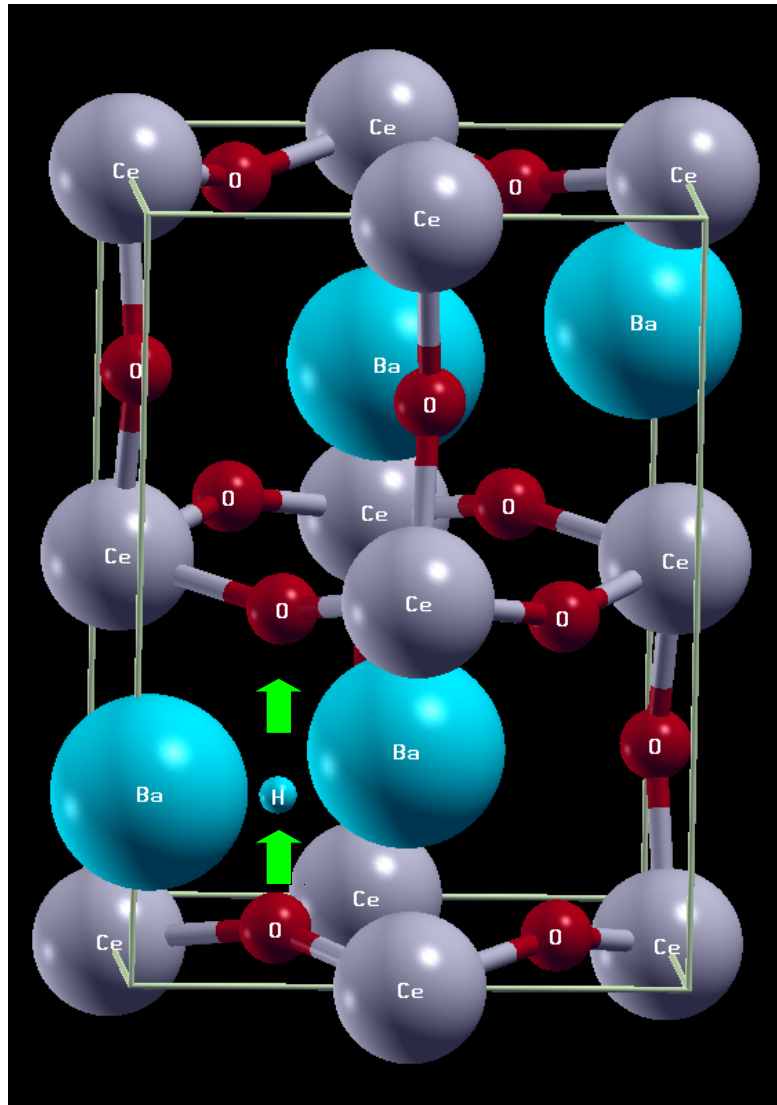
xred_1img ... **xred_lastimg**, **vel_1img** ... **vel_lastimg**

NPT ensemble: coming soon !

Constraints :

PIMD has been made compatible with constraints, using the keyword **pimd_constraint**
For the moment, only one constraint which is a linear combination of atomic positions is possible (using **natcon**, **iatcon**, **wtatcon**, **nconeq**)

Example of input file of constrained PIMD



BaCeO₃ : orthorhombic perovskite
Pnma space group

Excellent protonic conductor

Inter-octahedral transfer of H⁺

Cell = 20 atoms

Reaction coordinate : $(O_1H)_y - (O_2H)_y$
 $= y(H) - y(O_1) - [y(O_2) - y(H)]$

PIMD, T=200 K

P=32 beads

3 – imgmov : Path Integral Molecular Dynamics

Example of input file of constrained PIMD**# PARALLELIZATION**

=> 864 procs

paral_kgb 1
 npkpt 9
 npband 3
 npfft 1
 npimage 32

PIMD/MD

optcell 0
 irandom 3
 restartxf -1
imgmov 9 # langevin PIMD
 ntimimage 10000
nimage 32
mdtemp 400 200
vis 5d-05
 dynimage 32*1
nsym 1
 pitransform 0
 amu 140 16 138 1 # ce o ba h
 pimass 140 16 138 1
 dtion 10

CONSTRAINTS

pimd_constraint 1
nconeq 1
natcon 3
iatcon 5 20 1 #ox1 ox2 hydrogen
wtatcon
0.0 -1.0 0.0 0.0 -1.0 0.0 0.0 2.0 0.0

ground state

nband 111
 occopt 3
 tsmear 0.001
 pawovlp -1
 ixc 11
 ecut 18.0
 pawecutdg 25.0
 kptopt 1
 ngkpt 3 2 3
 nstep 100
 toldff 1.0d-05
 prtdden 0
 prtww 0

 charge 1.0

atoms

znuc1 58 8 56 1
 ntypat 4
 natom 21
 typat
 4 3 1 2 2 3 2 2 3 3 1 1 1 2 2 2 2 2 2 2

cell and atomic positions

acell 11.941471 16.813056 11.913748

xred

0.279	0.216	0.723 #yH to be varied
2.5108646807E-02	2.5000000000E-01	-8.0469316958E-03
-1.3877787808E-17	-9.7144514655E-17	5.0000000000E-01
4.8011508619E-01	2.5000000000E-01	7.9658009587E-02
2.7974388262E-01	4.3160947462E-02	7.2102708776E-01 #ox1
9.7489135319E-01	7.5000000000E-01	1.0080469317E+00
5.1988491381E-01	7.5000000000E-01	9.2034199041E-01
7.2025611738E-01	9.5683905254E-01	2.7897291224E-01
5.2510864681E-01	2.5000000000E-01	5.0804693170E-01
4.7489135319E-01	7.5000000000E-01	4.9195306830E-01
5.0000000000E-01	5.0000000000E-01	-6.9388939039E-17
-1.3877787808E-17	5.0000000000E-01	5.0000000000E-01
5.0000000000E-01	-9.7144514655E-17	-6.9388939039E-17
9.8011508619E-01	2.5000000000E-01	4.2034199041E-01
1.9884913809E-02	7.5000000000E-01	5.7965800959E-01
7.7974388262E-01	4.5683905254E-01	7.7897291224E-01
7.2025611738E-01	5.4316094746E-01	2.7897291224E-01
2.2025611738E-01	9.5683905254E-01	2.2102708776E-01
2.2025611738E-01	5.4316094746E-01	2.2102708776E-01
2.7974388262E-01	4.5683905254E-01	7.2102708776E-01 #ox2
7.7974388262E-01	4.3160947462E-02	7.7897291224E-01

1 – Introduction

2 – The keyword **ionmov**

- Structural Optimizations
- Molecular Dynamics

3 – The keyword **imgmov**

- Minimum Energy Paths
(=> *Advanced Feature on thursday/friday*)
- Path Integral Molecular Dynamics

4 – Conclusion

4 – Conclusion

With ABINIT, you can

- **structurally optimize** a system,
 - * atomic positions only (cell fixed)
 - * atomic positions + cell
 - * atomic positions + cell under non zero pressure/stress
- perform **Molecular Dynamics** in various statistical ensembles
- compute the **Minimum Energy Path** between two optimized configurations (useful to study, e.g. diffusion of defects in solids)
 - => **come to optional lesson on thursday/friday**
- perform **Path-Integral Molecular Dynamics** to simulate the quantum effects associated to nuclear motions.
- many other things ! See the website at input variables **ionmov**, **imgmov**.



Thanks for your attention !