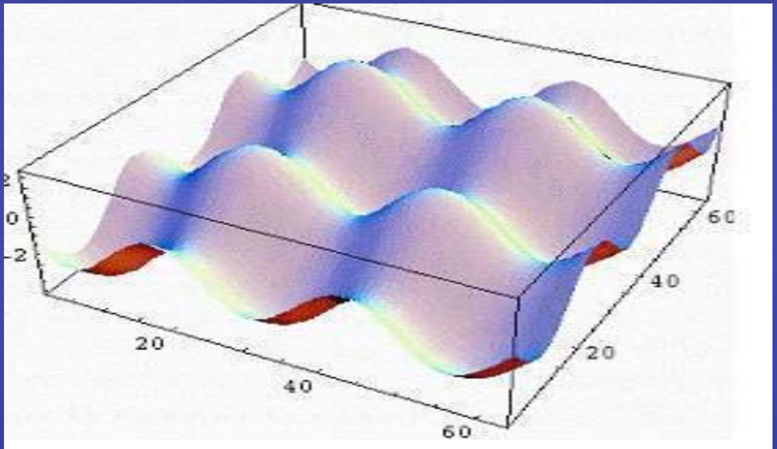
Molecular dynamics (MD) in different ensembles, geometry optimizations and calculation of vibrational spectrum



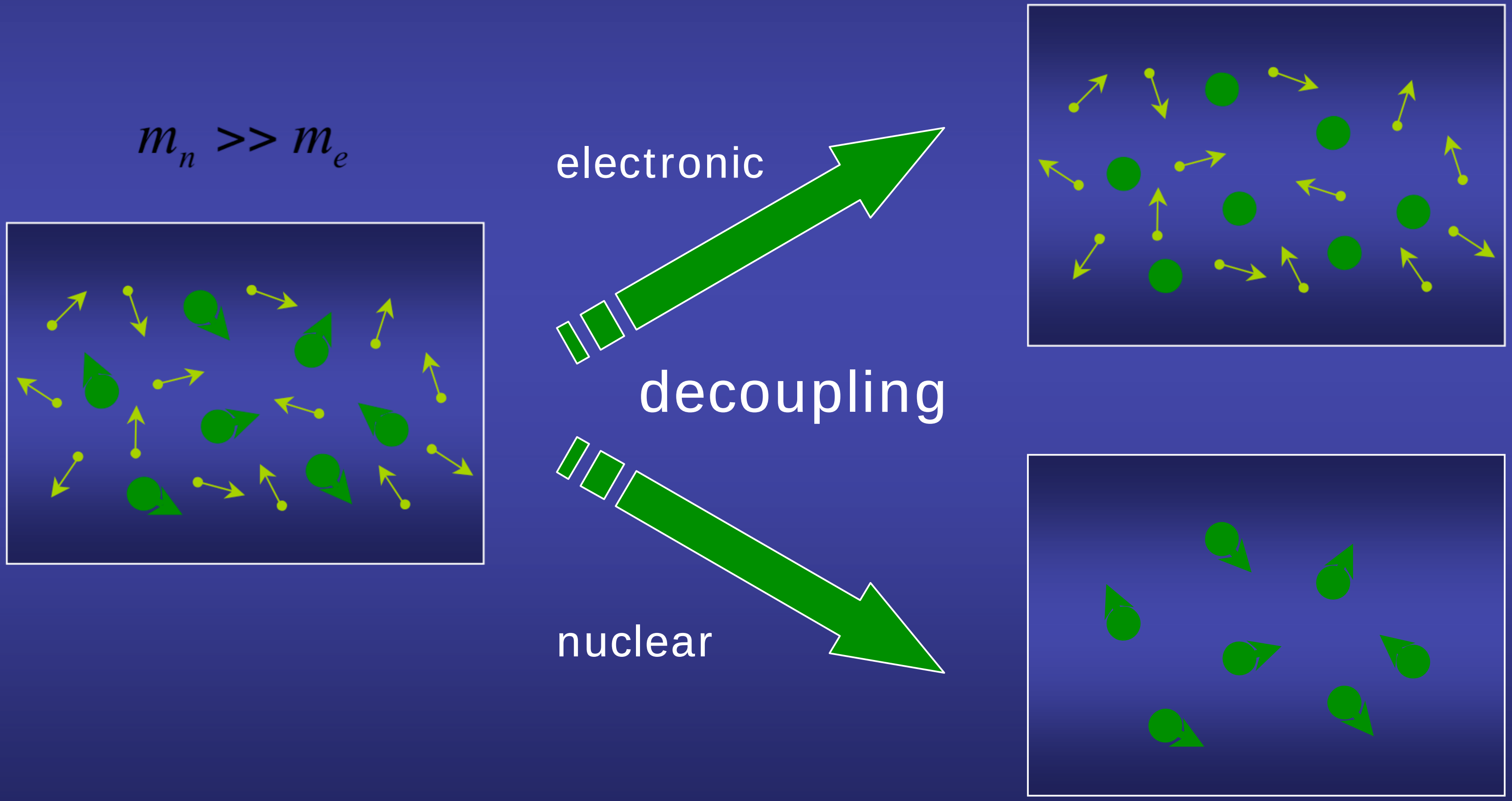
Marivi Fernandez-Serra

CECAM

Born-Oppenheimer dynamics

Nuclei are much slower than electrons

$$m_n >> m_e$$

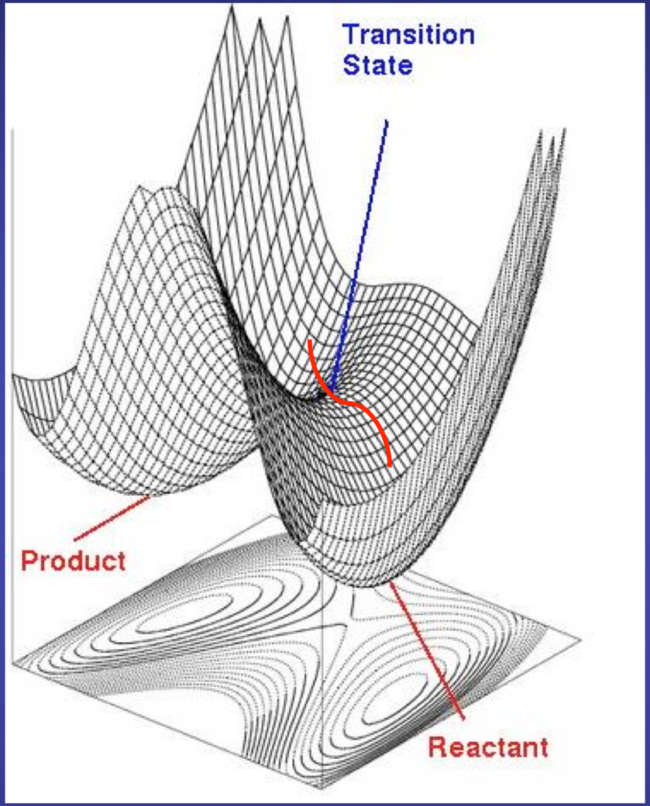


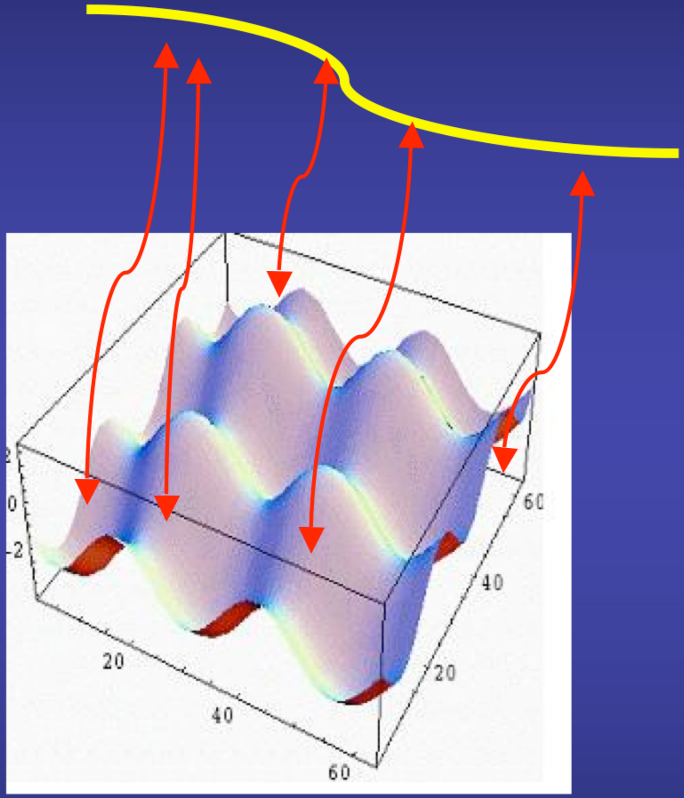
electronic

decoupling

nuclear

Extracting information from the Potential Energy Surface (PES)





-Optimizations and Phonons:
-We move on the PES
- Local vs global minima
- PES is harmonic close to minima

-MD
-We move over the PES
- Good Sampling is required!!

Theory of geometry optimization

$$E(x + \delta x) = E(x) + G(x)\delta x + \frac{1}{2}H(x)\delta x^2$$

Gradients

Hessian

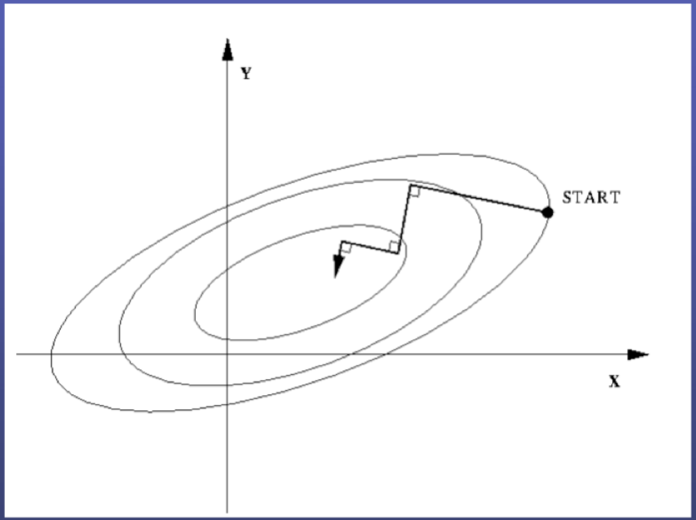
$$\delta x = -\alpha H(x)^{-1}G(x)$$

α=1 for quadratic region

Methods of optimization(I)

Steepest descent: Move in the direction of maximum incline.

Repeats search directions



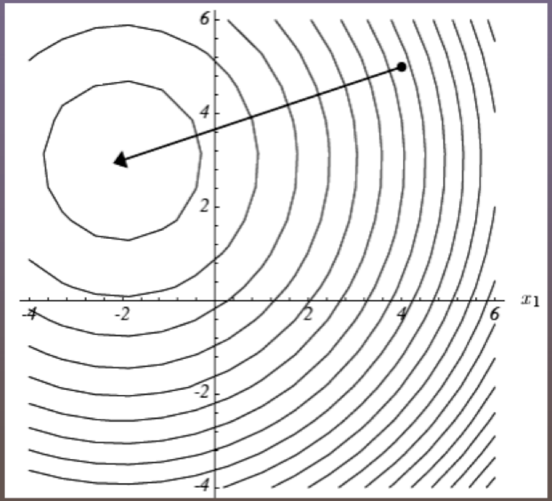
Ill conditioning:

$$H\vec{\chi}_i = \omega_i \vec{\chi}_i$$

Condition number: Determines convergence

$$\kappa \propto \frac{\omega_i^{max}}{\omega_i^{min}}$$

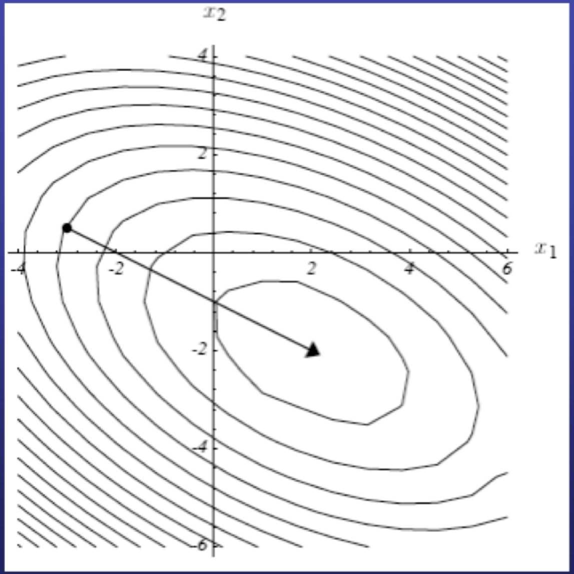
Converges to the first iteration
If all ω are equal

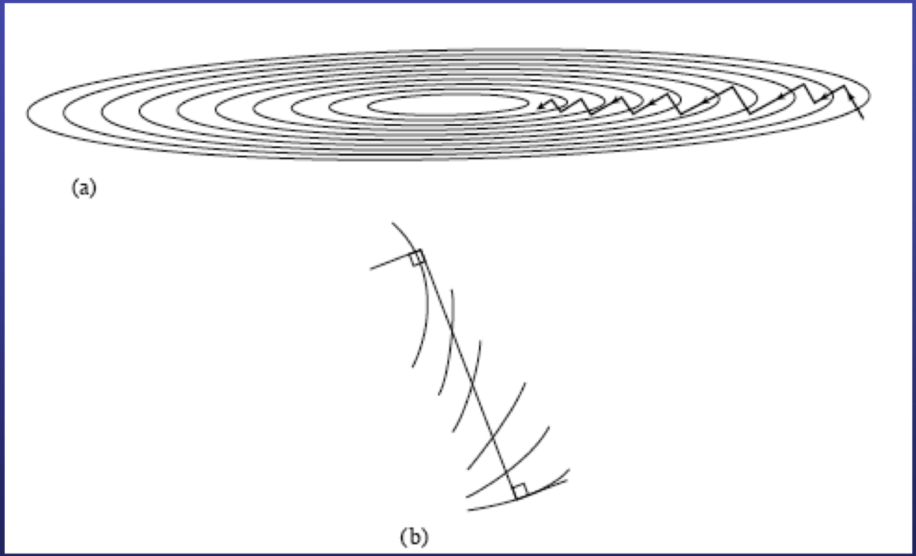


Methods of optimization(II)

- **Energy and first derivatives (forces):**
  - conjugate gradients (retains information)

Conjugate search directions: Make sure that new search directions are orthogonal to previous ones.





Methods of optimization(III)

- Energy, first and second derivatives:

  - Newton-Raphson: An approximation of H at a position (Xk) is calculated. Then finding the inverse of that Hessian (H^{-1}), and solving the equation $P = -H^{-1}*F(Xk)$ gives a good search direction P. Later, a line search procedure has to determine just how much to go in that direction (producing the scalar alpha). The new position is given by: Xk+1 = Xk + alpha*P

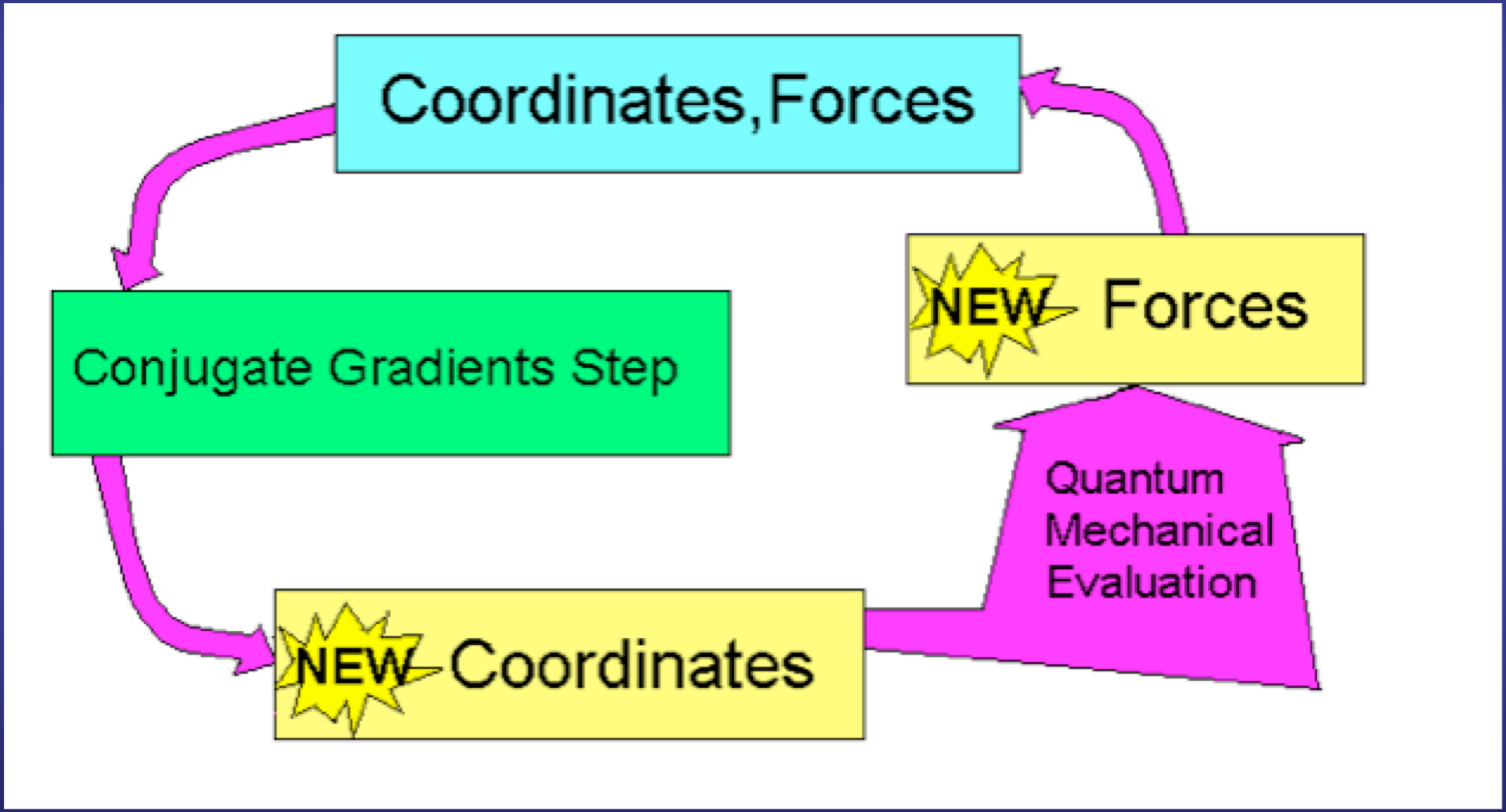
- BFGS updating of Hessian (reduces inversions)

  Basic idea, update the Hessian along the minimization to fit:

$$\delta F = H \delta x$$

SIESTA presently uses conjugate gradients and BFGS

Optimization (and MD) general basic Step.



Optimization in SIESTA(1)

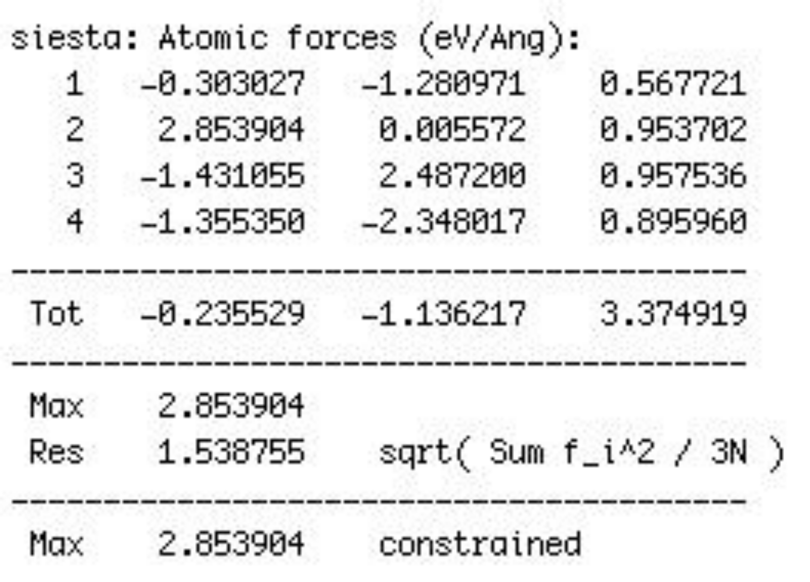
- Set runtype to conjugate gradients:
  MD.TypeOfRun CG, Broyden
- Set maximum number of iterative steps:
  MD.NumCGsteps 100
- Optionally set force tolerance:
  MD.MaxForceTol 0.04 eV/Ang
- Optionally set maximum displacement:
  MD.MaxCGDispl 0.2 Bohr

Optimizations in SIESTA(2)

- By default optimisations are for a fixed cell
- To allow unit cell to vary:
  MD.VariableCell true
- Optionally set stress tolerance:
  MD.MaxStressTol 1.0 Gpa
- Optionally set cell preconditioning:
  MD.PreconditionVariableCell 5.0 Ang
- Set an applied pressure:
  MD.TargetPressure 5.0 GPa

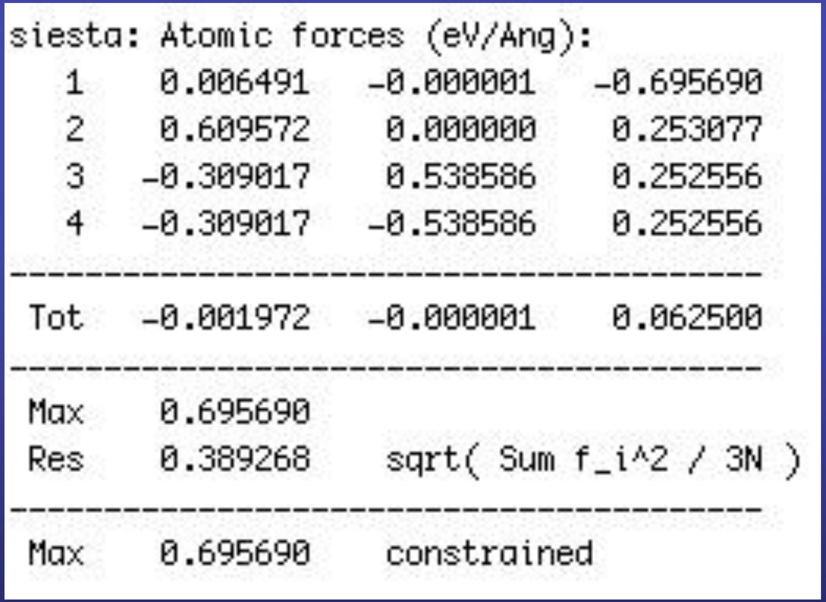
Advice on optimizations in SIESTA(I)

- **Make sure that your MeshCutoff is high enough:**
  - **- Mesh leads to space rippling**
  - **- If oscillations are large convergence is slow**
  - **- May get trapped in wrong local minimum**



```
siesta: Atomic forces (eV/Ang):
    1   -0.303027   -1.280971    0.567721
    2    2.853904    0.005572    0.953702
    3   -1.431055    2.487200    0.957536
    4   -1.355350   -2.348017    0.895960
----------------------------------------
  Tot   -0.235529   -1.136217    3.374919
----------------------------------------
  Max    2.853904
  Res    1.538755    sqrt( Sum f_i^2 / 3N )
----------------------------------------
  Max    2.853904    constrained
```

✗



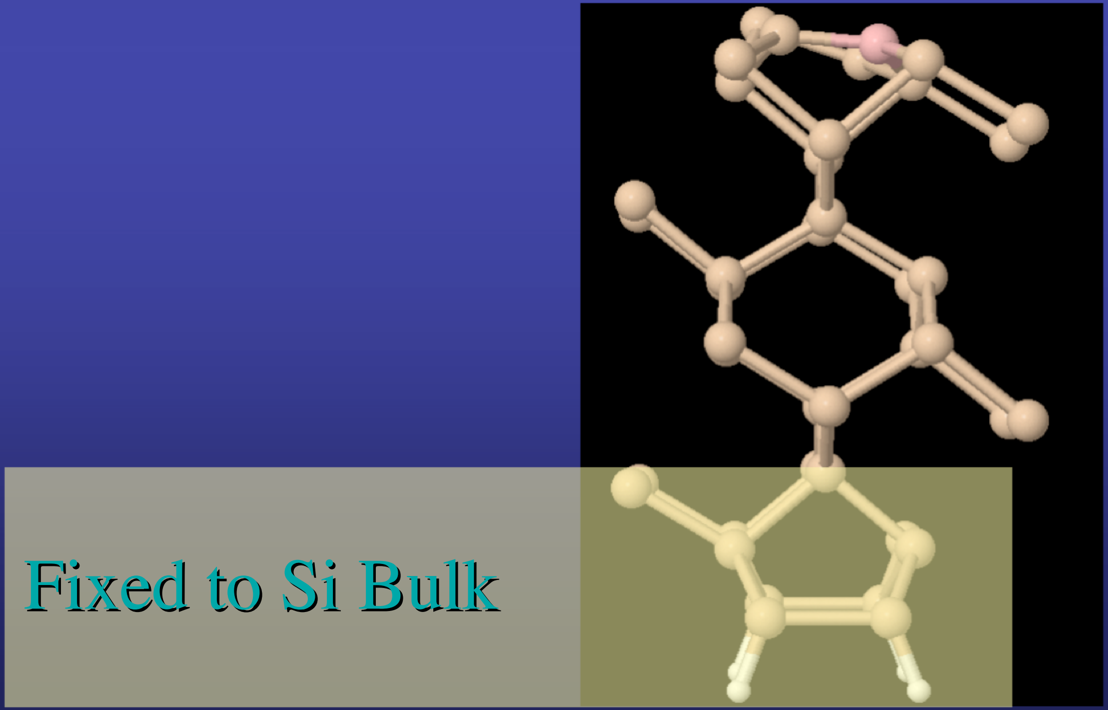
```
siesta: Atomic forces (eV/Ang):
    1    0.006491   -0.000001   -0.695690
    2    0.609572    0.000000    0.253077
    3   -0.309017    0.538586    0.252556
    4   -0.309017   -0.538586    0.252556
----------------------------------------
  Tot   -0.001972   -0.000001    0.062500
----------------------------------------
  Max    0.695690
  Res    0.389268    sqrt( Sum f_i^2 / 3N )
----------------------------------------
  Max    0.695690    constrained
```

✓

Advice on Optimizations in SIESTA(II)

- Ill conditioned systems (soft modes) can slow down optimizations, very sensitive to mesh cuttof.

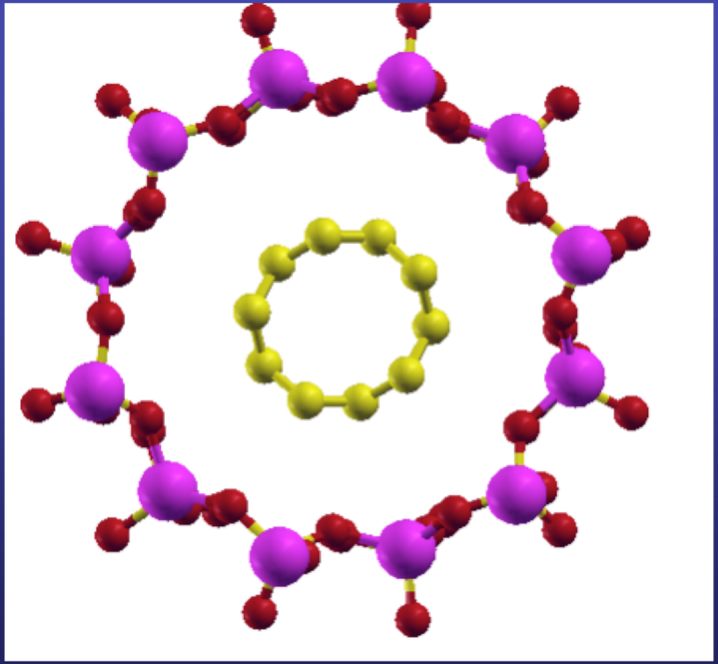
  - Use constraints when relevant.



Advice on Optimizations in SIESTA(III)

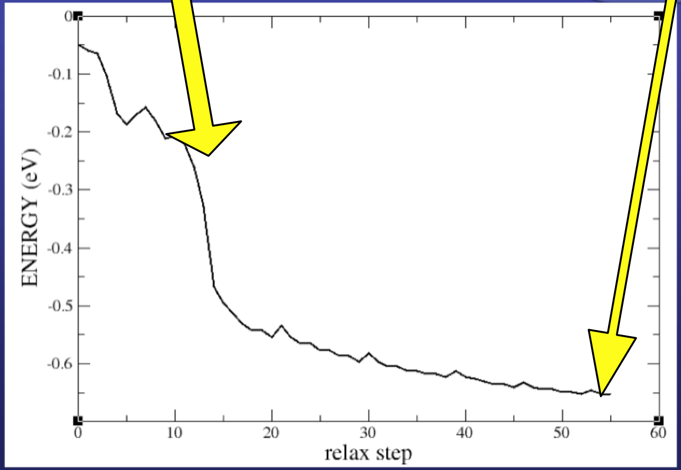
- Decouple Degrees of freedom (relax separately different parts of the system).

Look at the evolution of relevant physics quantities (band structure, Ef).



Fix the Zeolite,
Its relaxation is no
Longer relevant.
F_{tube}<0.04 eV/A
F_{zeol}>0.1 eV/A

No constraints



Molecular Dynamics

- Follows the time evolution of a system
- Solve Newton's equations of motion:

$$\vec{F}(t) = -\frac{dE}{d\vec{x}} = m\vec{a}(t) = m\frac{d^2\vec{x}(t)}{dt^2}$$

- Treats electrons quantum mechanically
- Treats nuclei *classically*
    - Hydrogen may raise issues:
        - tunnelling (overestimating Energy barriers)
- Allows study of dynamic processes
- Annealing of complex materials
- Examines the influence of temperature
- Time averages Vs Statistical averages

Ergodicity

- **In MD we want to replace a full sampling on the appropriate statistical ensemble by a SINGLE very long trajectory.**

- **This is OK only if system is** ergodic**.**

- ***Ergodic Hypothesis: a phase point for any isolated system passes in succession through every point compatible with the energy of the system before finally returning to its original position in phase space. This journey takes a Poincare cycle.***

- ***In other words, Ergodic hypothesis*****: each state consistent with our knowledge is equally “likely”.**
  - **Implies the average value does not depend on initial conditions.**
  - $\langle A \rangle_{time} = \langle A \rangle_{ensemble}$, **so** $\langle A_{time} \rangle = (1/N_{MD}) = \sum_{t=1,N} A_t$ **is good estimator.**

- **Are systems in nature really ergodic? Not always!**
  - **Non-ergodic examples are glasses, folding proteins (in practice) and harmonic crystals (in principle).**

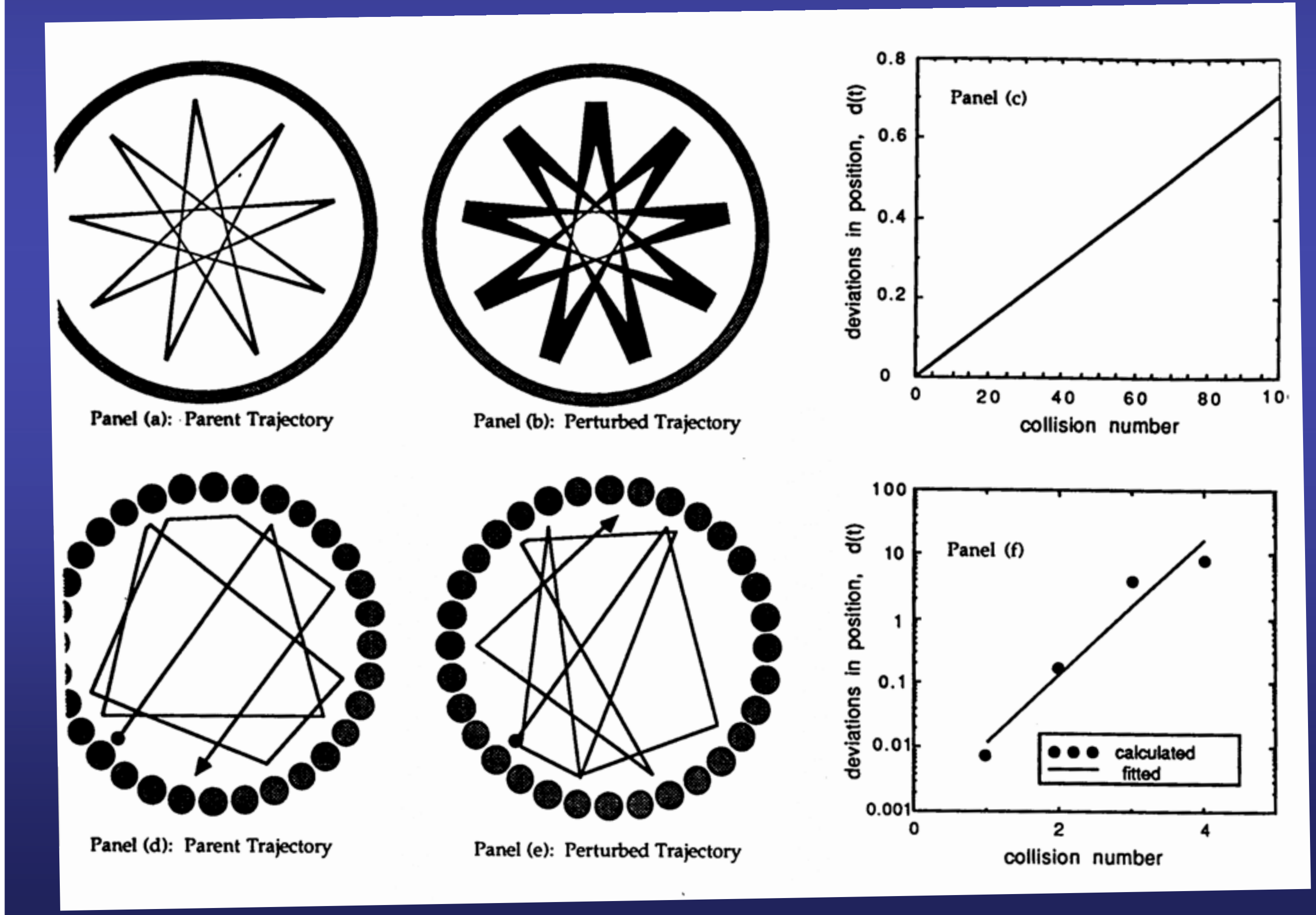
Different aspects of ergodicity

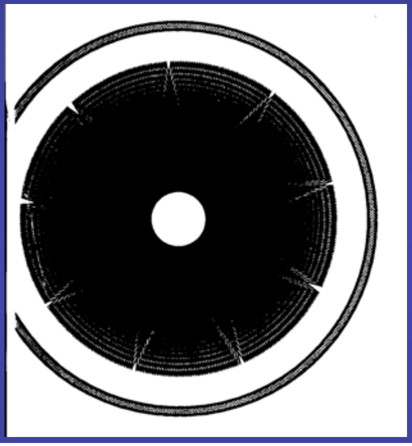
- The system relaxes on a “reasonable” time scale towards a unique equilibrium state (microcanonical state)
- Trajectories wander irregularly through the energy surface eventually sampling all of accesible phase space.
- Trajectories initially close together separate rapidily.(sensitivity to initial conditions). Lyapunov exponent.

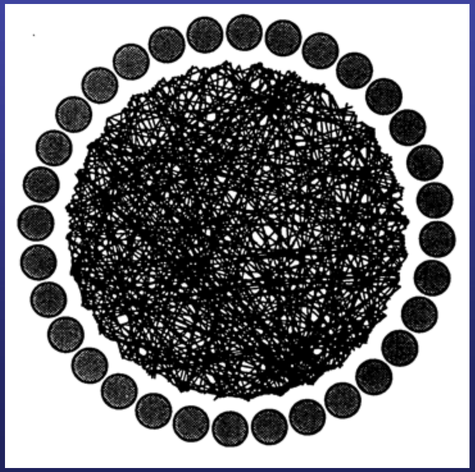
Ergodic behavior makes possible the use of statistical methods on MD of small system. Small round-off errors and other mathematical approximations may not matter.

Particle in a smooth/rough circle

From J.M. Haile: MD Simulations







Molecular Dynamics(I)

In Molecular Dynamics simulations, one computes the evolution of the positions and velocities with time, solving Newton’s equations.

$$\vec{F}(t) = -\frac{dE}{d\vec{x}} = m\vec{a}(t) = m\frac{d^2\vec{x}(t)}{dt^2} \quad \Rightarrow$$

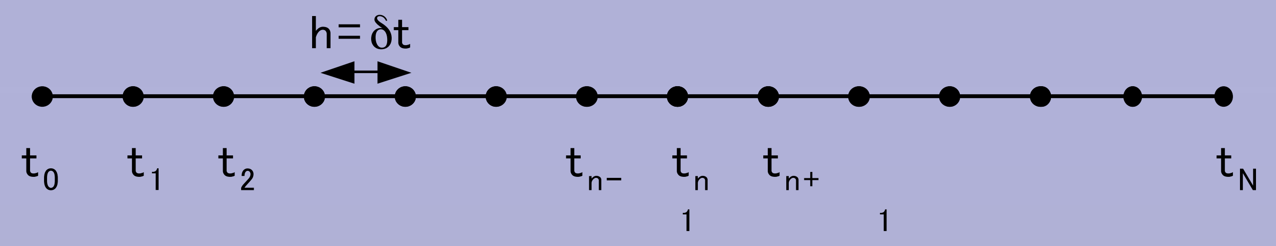
$$\vec{x}(t) = \vec{x}(t_0) + \vec{v}(t_0)(t - t_0) + \int_{t_0}^{t} dt' \int_{t_0}^{t'} \tfrac{1}{m}\vec{F}(t'')dt''$$

•Algorithm to integrate Newton’s equations: “Verlet”

- Initial conditions in space and time.

Molecular Dynamics(II)

- Choosing particles, masses and interatomic forces (model of interactions)
- Initialize positions and momenta at t=0 (initial conditions in space and time)
- Solve F = m a to determine r(t), v(t). (integrator)
  - We need to make time discrete, instead of continuous!!!



- Calculate the properties of interest along the trajectory
- Estimate errors
- Use the results of the simulation to answer physical questions!!.

Molecular Dynamics III

- **Timestep must be small enough to accurately sample highest frequency motion**
- **Typical timestep is 1 fs (1 x 10^{-15} s)**
- **Typical simulation length = Depends on the system of study!!**

  (the more complex the PES the longer the simulation time)
- ***Is this timescale relevant to your process?***
- **Simulation has two parts:**
  - **equilibration (redistribute energy)**

  System is equilibrated if averages of dynamical and structural quantities do not change with time.
  - **production (record data)**
- **Results:**
  - **diffusion coefficients**
  - **Structural information (RDF's,)**
  - **free energies / phase transformations (very hard!)**
- ***Is your result statistically significant?***

Choosing the integrator

- The interatomic potentials are highly non-linear, often with discontinuous high derivatives, or are evaluated with limited precision.
- Small errors (precision) or minimal differences in the initial conditions lead to completely different trajectories (Ergodicity!). Statistical averages are the relevant quantities; they do not depend on the details of the trajectories (IF the simulation is long enough!!!!).
- Because of this, and since potentials are not perfect (all potential models are approximations to the real ones), one does not need too much accuracy in the integration of the equations of motion (as long as errors are not too large, and they do not affect fundamental properties such as conserved quantities).
- Conservation of energy IS important!!. We can allow errors in the total energy conservation of the order of 0.01 kT.
- CPU time is completely dominated by the calculation of the forces. Therefore, it is preferable to choose algorithms that require few evaluations of the forces, and do not need higher derivatives of the potential.

Verlet algorithm

The most commonly used algorithm:

$r(t+h) = r(t) + v(t)\,h + 1/2\,a(t)\,h^2 + b(t)\,h^3 + O(h^4)$ (Taylor series expansion)
$r(t-h) = r(t) - v(t)\,h + 1/2\,a(t)\,h^2 - b(t)\,h^3 + O(h^4)$

$r(t+h) = 2\,r(t) - r(t-h) + a(t)\,h^2 + O(h^4)$ Sum

$v(t) = (r(t+h) - r(t-h))/(2h) + O(h^2)$ Difference (*estimated velocity*)

- Trajectories are obtained from the first equation. Velocities are not necessary.
- Errors in trajectory: $O(h^4)$
- Preserves time reversal symmetry.
- Excellent energy conservation.
- Modifications and alternative schemes exist (leapfrog, velocity Verlet), always within the second order approximation
- Higher order algorithms: Gear

When do we use MD?

- Amorphous systems:
  - Molecular Liquids ($H2O$,$CO2$)
  - Glasses (Si, $SiO2$)
- Displacive Phase transitions (P and T relevant).
- Study of kinetic effects.
  - Diffusion at surfaces
  - Thermal stability

Different ensembles, different Lagrangians, different Conserved magnitudes.

- **NVE (Verlet): Microcanonical.**
- Integrates Newtons equations of motion, for N particles, in a fixed volume V.
- Natural time evolution of the system: E is a constant of motion

- **NVT (Nose): Canonical**
- System in thermal contact with a heat bath.
- Extended Lagrangian:
- N particles + Thermostat, mass Q.

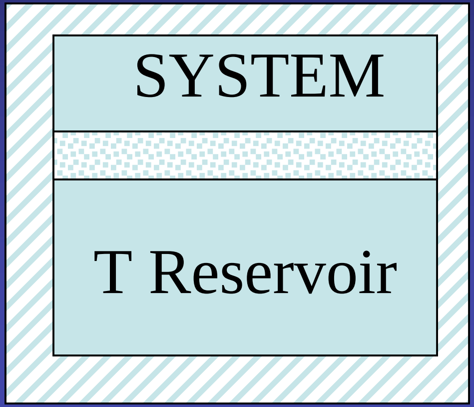
- **NPE (Parrinello-Rahman) (isobarical)**
- Extended Lagrangian
- Cell vectors are dynamical variables with an associated mass.

- **NPT (Nose-Parrinello-Rahman)**
- 2 Extended Lagrangians
- NVT+NPE.

Nose-Hoover thermostat

- **MD in canonical distribution (TVN)**
- **Introduce a friction force** ζ(t)

$$\frac{dp}{dt} = F(q,t) - \zeta(t)\,p(t)$$



SYSTEM

T Reservoir

Dynamics of friction coefficient to get canonical ensemble.

$$Q\frac{d\zeta}{dt} = \sum \frac{1}{2} m_i v_i^2 - \frac{3N}{2} k_B T$$

Feedback makes
K.E.=3/2kT

Q= fictitious “heat bath mass”. Large Q is weak coupling

Hints

- **Nose Mass: Match a vibrational frequency of the system, better high energy frequency**

$$Q \approx \frac{KT}{\omega^2}$$

Which Ensemble should we use?

- **NVE (Verlet): Microcanonical**
    - Good trajectories.
    - Time reversible (up to numerical error)
    - Dynamical variables are well defined.
    - Initial X and V are relevant: necessity of equilibration.

Same sampling
In the
thermodynamic limit

- **NVT (Nose): Canonical**
    - Good T control
    - Equilibrates the system.
    - Choice for Structural sampling.
    - Sensitive to Nose mass.

- **NPE (Parrinello-Rahman)**
    - Phase transitions systems under pressure.
    - 1 mass parameter (barostat)

- **NPT (Nose-Parrinello-Rahman)**
    - Phase transitions under P and T
    - 2 mass parameters, barostat and thermostat. (Fluctuations!!

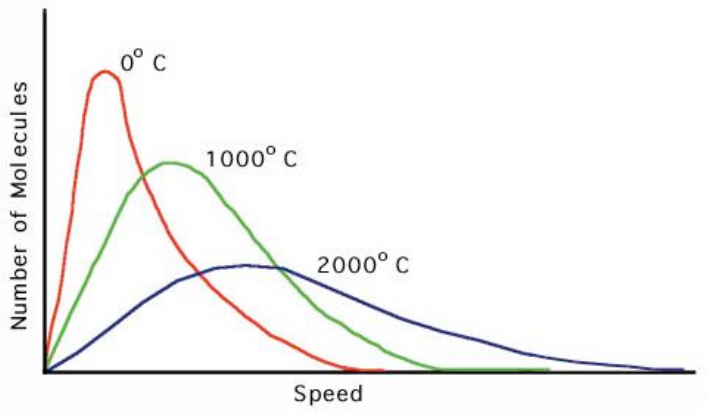
Molecular Dynamics in SIESTA(1)

- **MD.TypeOfRun Verlet**
  NVE ensemble dynamics
- **MD.TypeOfRun Nose**
  NVT dynamics with Nose thermostat
- **MD.TypeOfRun ParrinelloRahman**
  NPE dynamics with P-R barostat
- **MD.TypeOfRun NoseParrinelloRahman**
  NPT dynamics with thermostat/barostat
- **MD.TypeOfRun Anneal**
  Anneals to specified p and T

Variable
Cell

Molecular Dynamics in SIESTA(2)

- **Setting the length of the run:**
  MD.InitialTimeStep 1
  MD.FinalTimeStep 2000
- **Setting the timestep:**
  MD.LengthTimeStep 1.0 fs
- **Setting the temperature:**
  MD.InitialTemperature 298 K
  MD.TargetTemperature 298 K
- **Setting the pressure:**
  MD.TargetPressure 3.0 Gpa
- **Thermostat / barostat parameters:**
  MD.NoseMass / MD.ParrinelloRahmanMass



Maxwell-Boltzmann

Annealing in SIESTA

- **MD can be used to optimize structures:**
  MD.Quench true
  - zeros velocity when opposite to force
- **MD annealing:**
  MD.AnnealOption Pressure
  MD.AnnealOption Temperature
  MD.AnnealOption TemperatureAndPressure
- **Timescale for achieving target**
  MD.TauRelax 100.0 f

Vibrational spectrum: Phonons

- **Calculating Dynamical Matrix: Mass weighted Hessian Matrix (Harmonic approximation).**

$$H_{i,j}^{\alpha,\beta} = \frac{\delta^2 E}{\delta x_i^\alpha \delta x_j^\beta} = \frac{\delta F_i^\alpha}{\delta x_j^\beta}$$

Harmonic Approx.

- **Frozen Phonon approximation:**
  - **Numerical evaluation of the second derivatives. (finite differences).**
- **Density Functional Perturbation Theory (Linear Response):**
  - **Perturbation theory used to obtain analytically the Energy second derivatives within a self consistent procedure.**

Beyond Harmonic Approx.

- **Molecular dynamics: Green-Kubo linear response.**
  - **Link between time correlation functions and the response of the system to weak perturbations.**

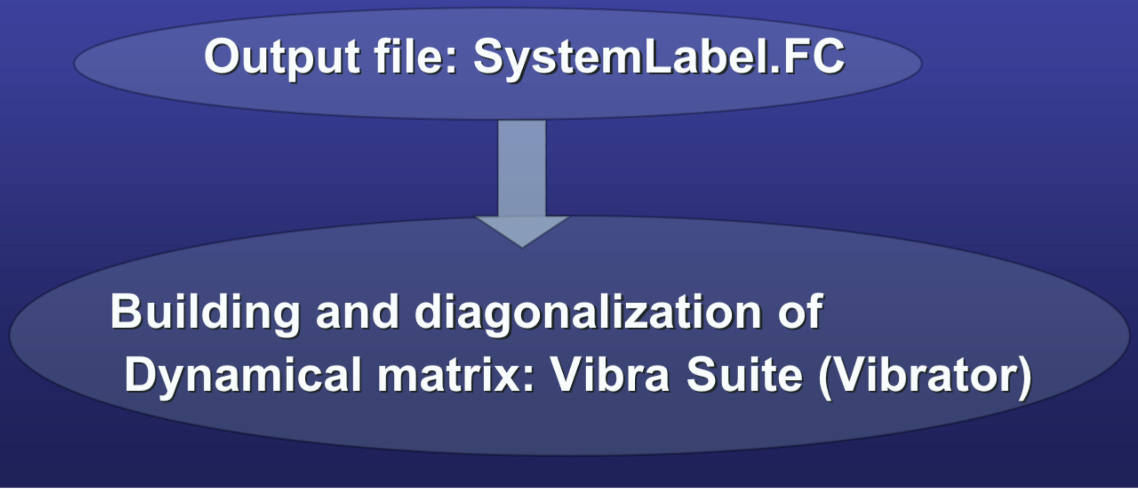
Phonons in Siesta (I)

- **Frozen Phonon approximation:**

MD.TypeOfRun FC

MD.FCDispl 0.04 Bohr (default)

Total number of SCF cycles: $\underset{\underset{(x,y,z)}{\uparrow}}{3} \; \mathrm{X} \; \underset{\underset{(+,-)}{\uparrow}}{2} \; \mathrm{X} \; \underset{\underset{\mathrm{Nat}}{\uparrow}}{N} = 6N$



Output file: SystemLabel.FC

Building and diagonalization of Dynamical matrix: Vibra Suite (Vibrator)

Phonons in Siesta (II)

1. Relax the system: Max F<0.02 eV/Ang

2. Increase MeshCutof, and run FC.

3. If possible test the effect of MaxFCDispl.

Phonons and MD

1. MD simulations (NVE)
2. Fourier transform of
   Velocity-Velocity autocorrelation function.
1. Anharmonic effects: ω(T)
2. Expensive, but information available for MD
   simulations.

$$Ph(\omega) = \int_0^\infty dt e^{2\pi i\omega t} \frac{\langle \vec{v}(0)\vec{v}(t)\rangle}{\langle \vec{v}(0)^2\rangle}$$