

Basics of molecular modelling

M2 SERP-Chem
Fabien Cailliez
LCP – Bât 349
fabien.cailliez@u-psud.fr

What is molecular modelling?

Wikipedia:

Molecular modeling encompasses all theoretical methods and computational techniques used to model or mimic the behaviour of molecules.

In other words:

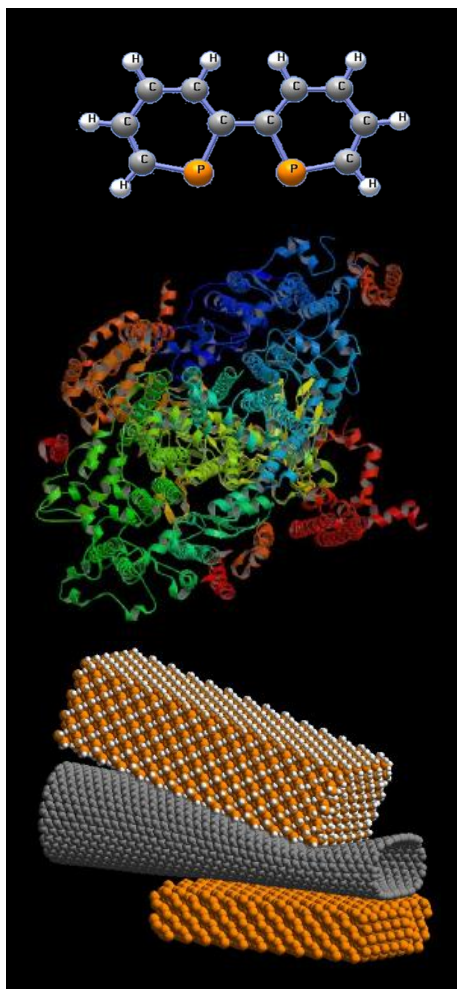
Theoretical methods that allows to describe **macroscopic** observations with the use of **microscopic** description of matter

What is molecular modelling?

Solving chemical problems with numerical experiments

OBJECTS

Molecular systems



TOOLS

MODELS

Physical laws

Atomic and
molecular interactions

Mathematical
models

GOALS

Properties
Comparison
Theory/experience

Geometries
Spectra
Thermodynamics
Activation energies
Rate constants

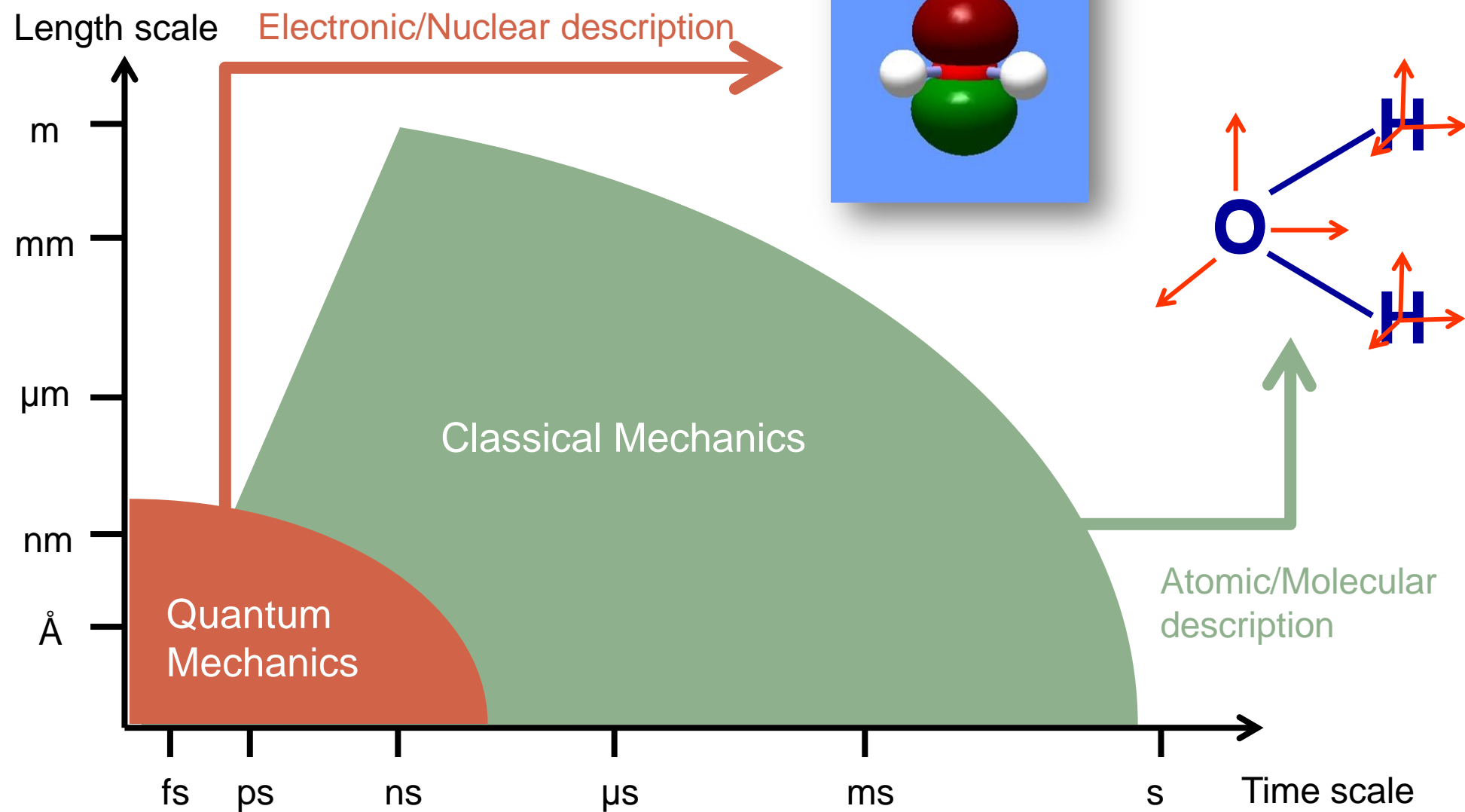
Understanding
Rationalization
Prediction

Outline

- I. Concepts in Molecular Simulations
- II. Molecular Dynamics simulations
- III. Monte Carlo simulations
- IV. Practical aspects of molecular simulations

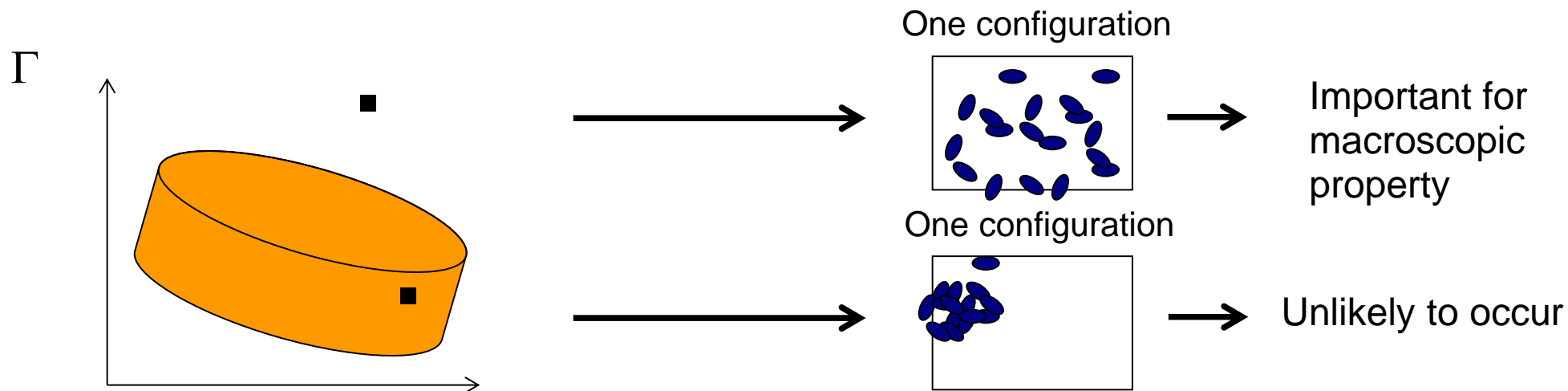
Concepts in Molecular Simulations

Molecular description and representation

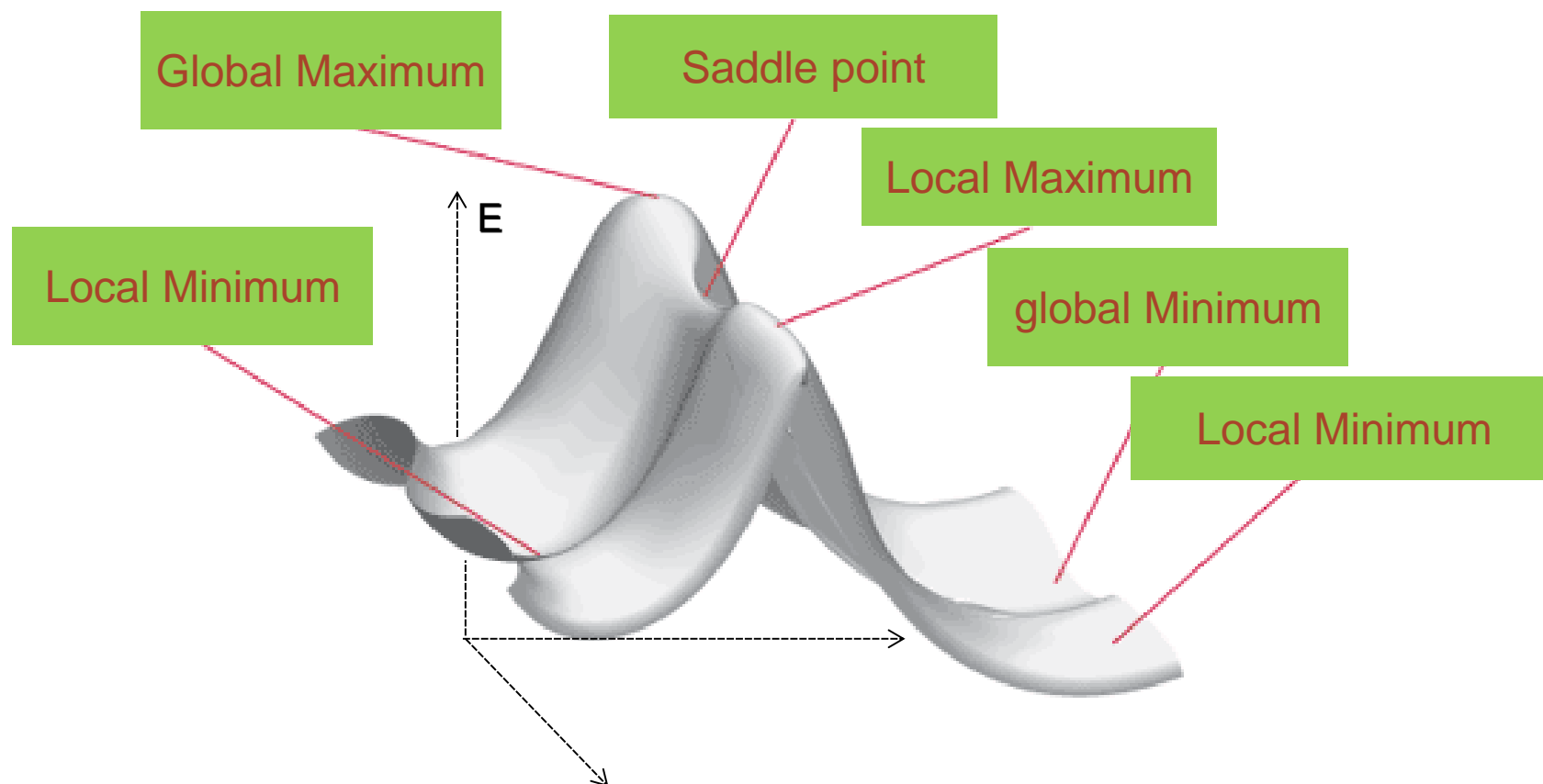


The phase space

- One microstate = a set of given coordinates and moments (r_N, p_N) of all the particles of the system
- N particles \rightarrow $6N$ -dimensional **phase space** Γ and $3N$ -dimensional **configuration space**
- All microstates do not have the same importance

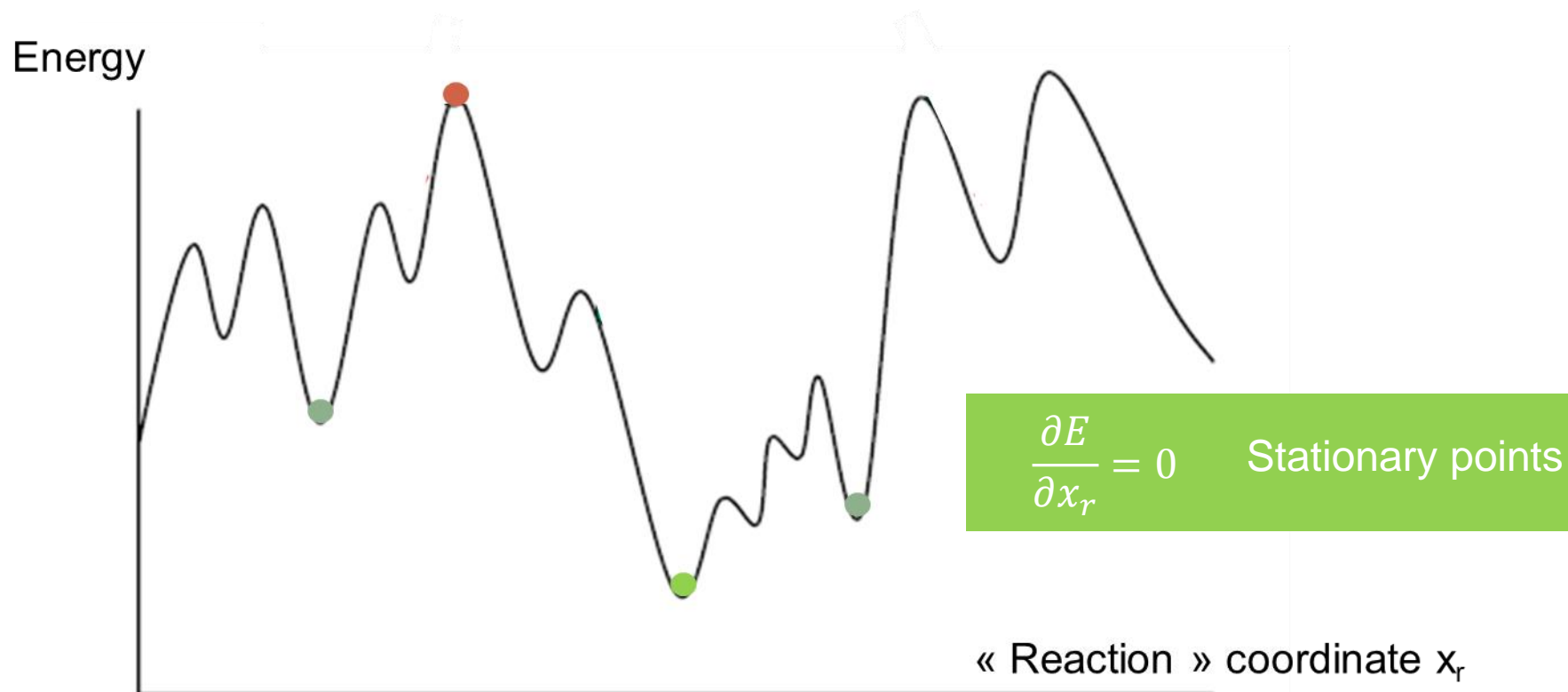


The potential energy surface (PES)



- **Potential energy surface** = energy as a function of nuclear coordinates (Born-Oppenheimer approximation)
- Each point of the configuration space corresponds to one point on the PES
- Various interesting points on the PES

Particular points of the PES



- Transition state: saddle point
- Equilibrium structures: local minima
- Equilibrium structure: global minimum

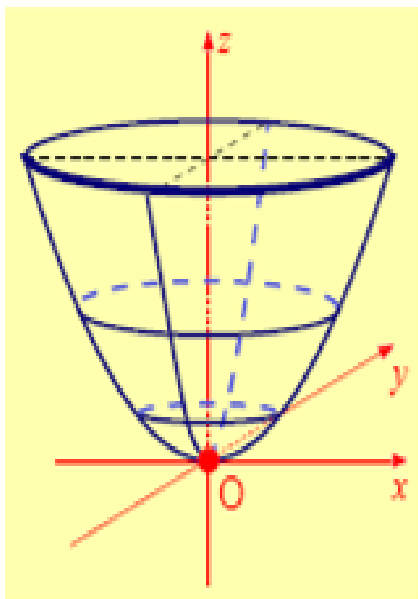
$$\frac{\partial^2 E}{\partial x_r^2} > 0$$

Maximum

$$\frac{\partial^2 E}{\partial x_r^2} < 0$$

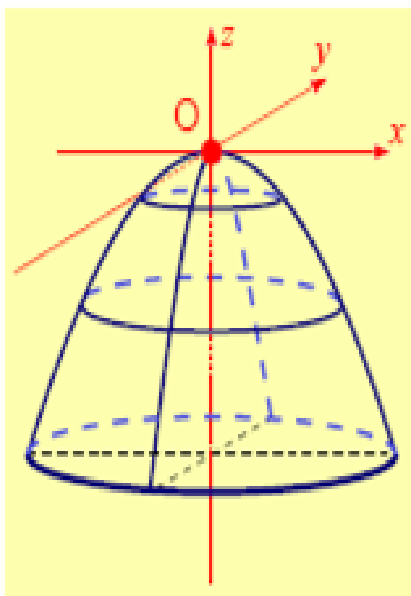
Minimum

The stationary points of the PES



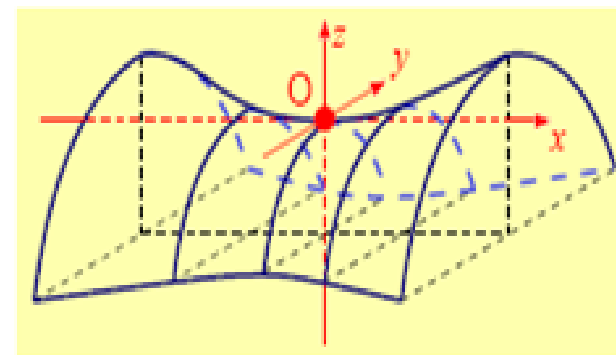
Minimum

$$\frac{\partial^2 f(x,y)}{\partial x^2} > 0; \frac{\partial^2 f(x,y)}{\partial y^2} > 0$$



Maximum

$$\frac{\partial^2 f(x,y)}{\partial x^2} < 0; \frac{\partial^2 f(x,y)}{\partial y^2} < 0$$

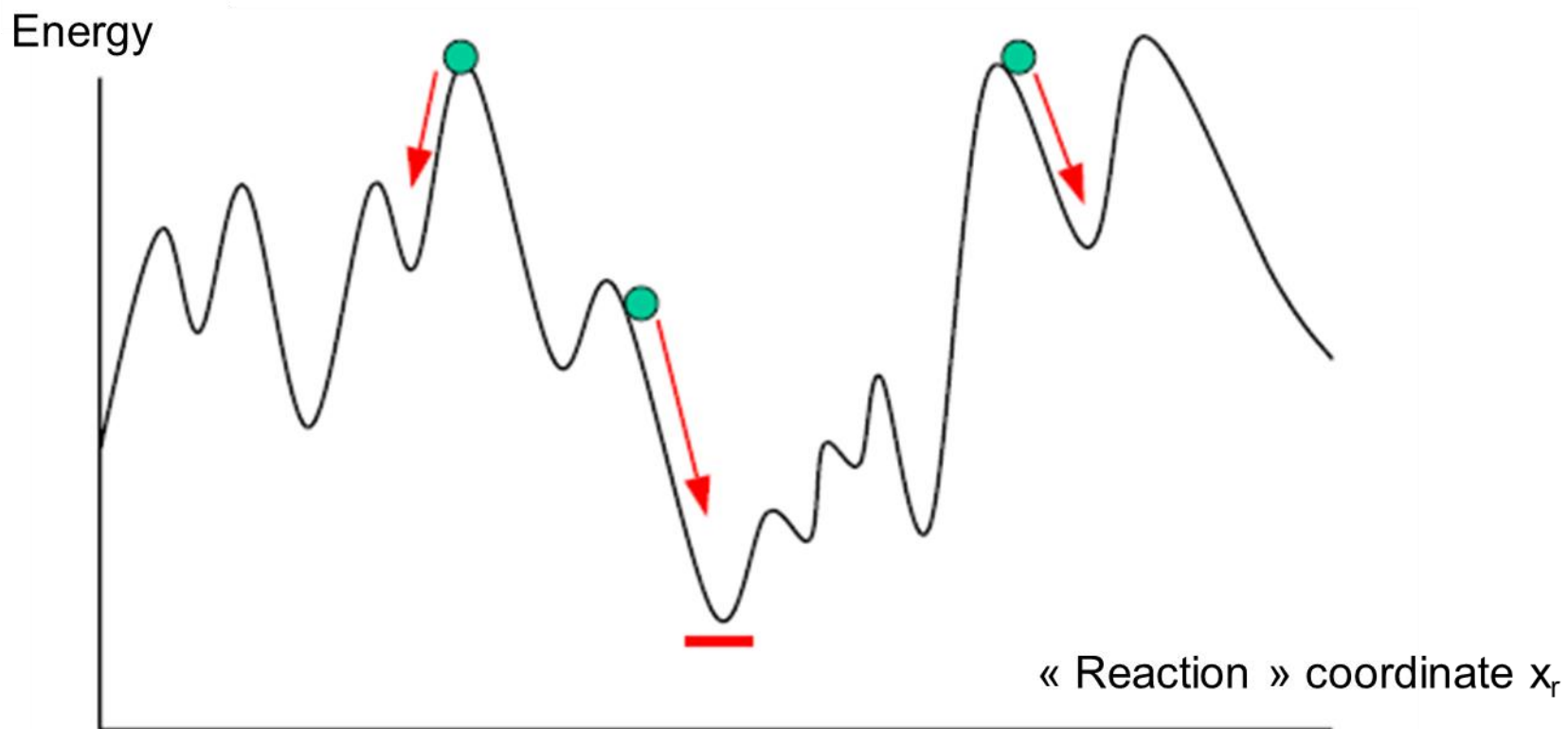


Saddle point

$$\frac{\partial^2 f(x,y)}{\partial x^2} > 0; \frac{\partial^2 f(x,y)}{\partial y^2} < 0$$

- **Stable states:** minimum of the PES with respect to every coordinates
- **Transition states:** minimum in every **normal** direction but one, for which it is maximum
- Stationary points thus correspond to important states of the system

Energy minimisation



- In practice, high dimensional surface
- Many local minima
- How to perform energy minimisation?

Classification of minimisation algorithms

- Goal of minimisation algorithms: find the coordinates that correspond to the minimum value of the energy:

$$\forall i, \frac{\partial E}{\partial x_i} = 0 \text{ and } \frac{\partial^2 E}{\partial x_i^2} > 0$$

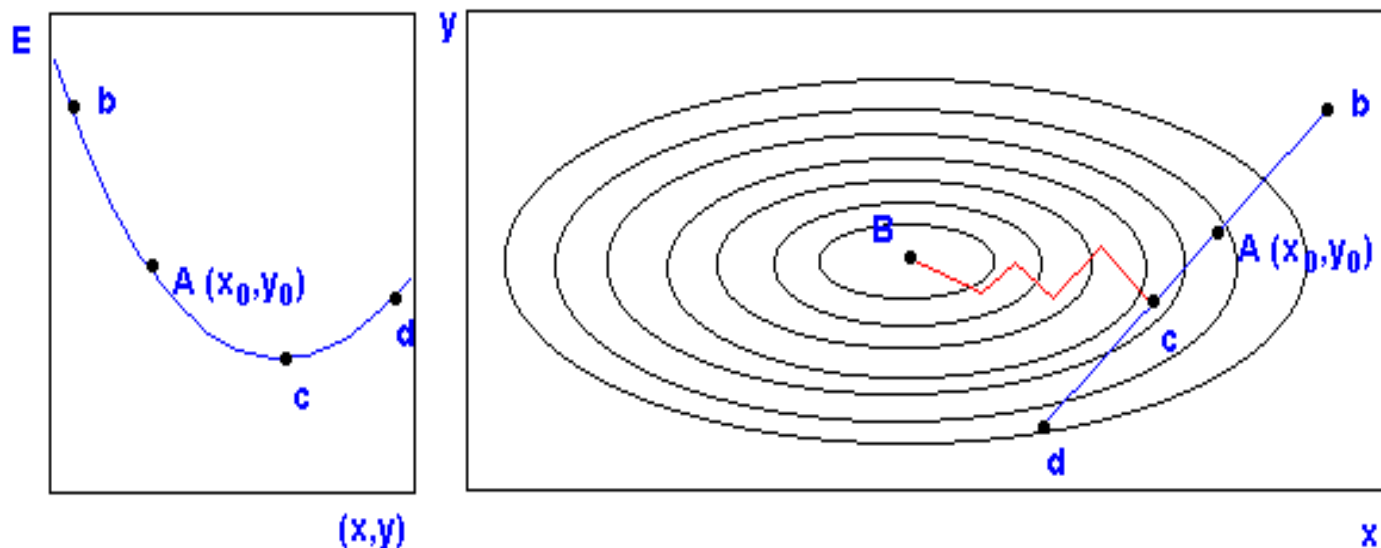
- Analytical search is impossible because of the complexity of the energy function



Use of numerical methods

- Two groups of numerical methods:
 - Methods using the derivatives of E:
 - First-order
 - Second-order
 - Methods that do not use the derivatives

First-order derivative method: Steepest descent



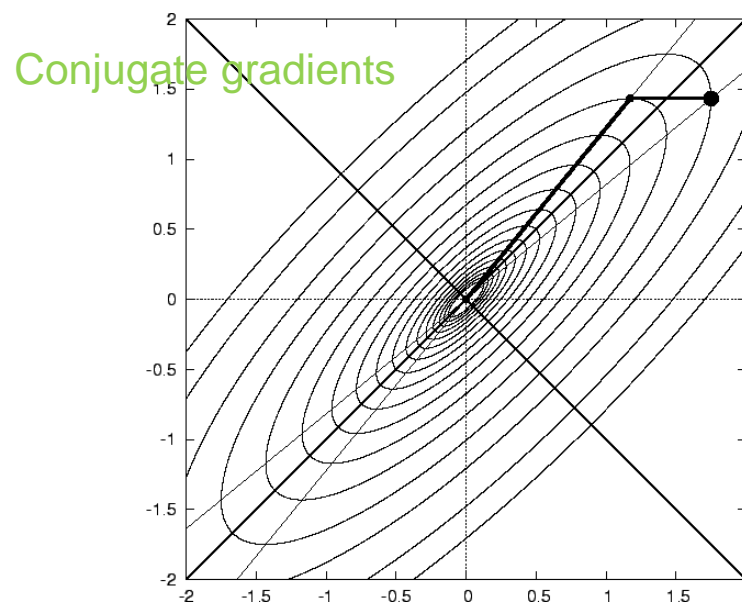
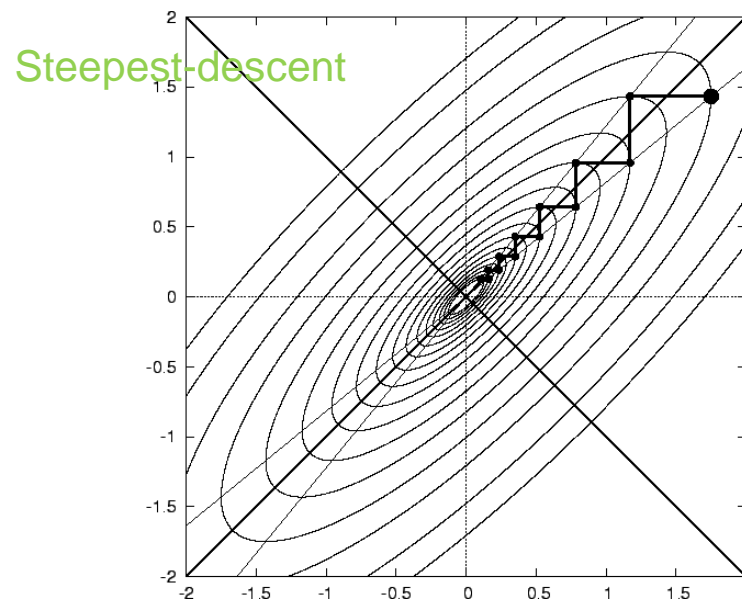
- The idea: the fastest decrease in energy at one point is seen along the gradient direction
- Algorithm steps:
 - The system is initially in A and the gradient \mathbf{g}_A of the energy is computed
 - The minimum along this gradient direction $\mathbf{v}_A = -\mathbf{g}_A$ is searched. The system reaches the point c
 - The gradient of the energy at c is computed (it is found perpendicular to the previous gradient) and the process iterates until one reaches B

First-order derivative method: Conjugate-gradients

- Drawbacks of the steepest-descent algorithm:
 - « very local »
 - May lead to oscillatory processes in narrow valleys
- The oscillatory behavior comes from the fact that the direction \mathbf{v}_i explored at step i is perpendicular to that \mathbf{v}_{i-1} at step $(i-1)$
- Conjugate-gradients: keeping memory of the previous direction explored

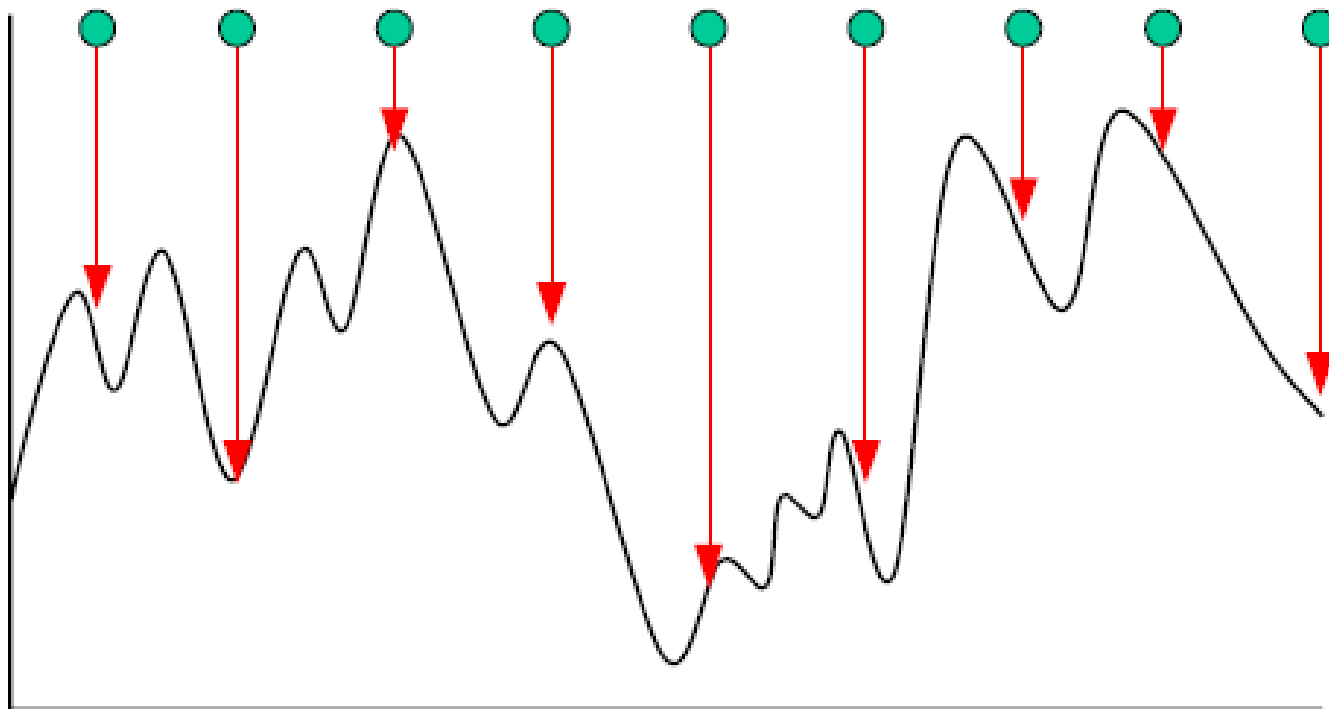
$$\mathbf{v}_i = -\mathbf{g}_i + \gamma_i \mathbf{v}_{i-1}$$

- Conjugate-gradients reaches minimum of a M-dimension quadratic function in M steps



Drawbacks of energy minimisation

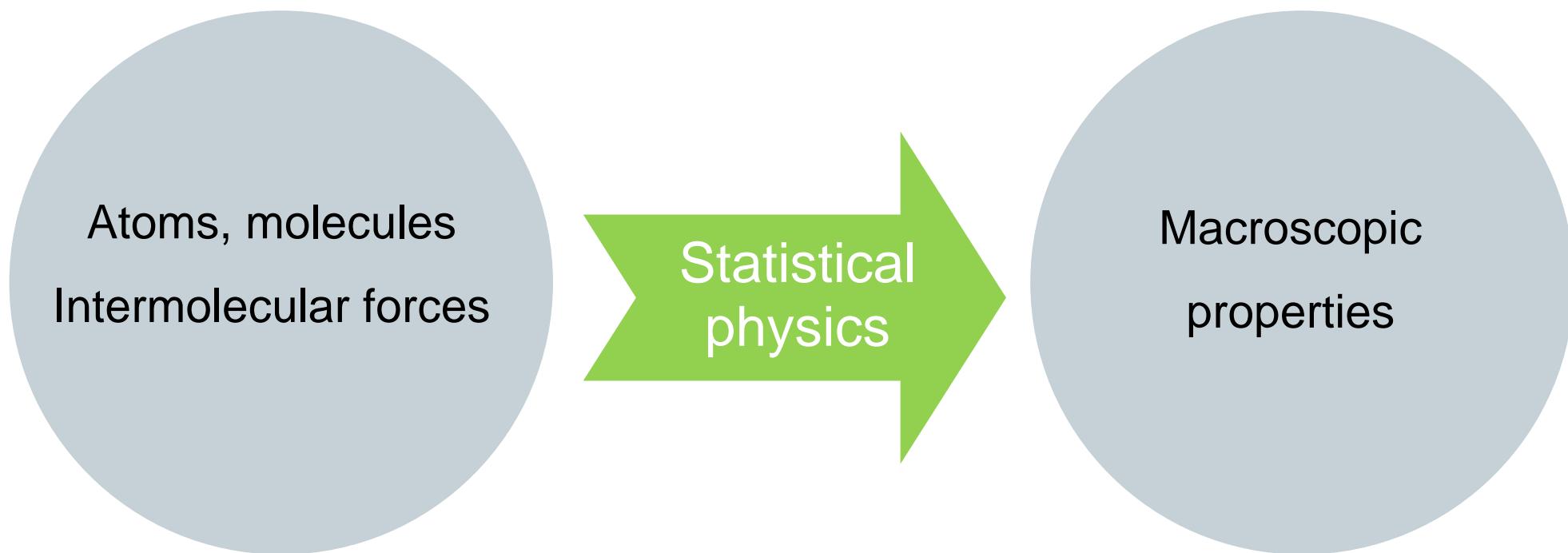
- Only the closest energy minimum is found



Energy minimisation from different configurations (generated by MD or MC simulations for example)

- Knowing minimum structures: is it enough?
 - Think of liquids and solids !!

How to compute macroscopic properties?



Principles of statistical physics

- **Ensemble averages**

Thermodynamic properties are averages of microscopic quantities over the **accessible microscopic** states of the system

Discrete phase space

$$X = \bar{X}_i = \sum_i P_i X_i$$

Continuous phase space

$$X = \overline{X(r_N, p_N)} = \int \dots \int a(r_N, p_N) P(r_N, p_N) dr_N dp_N$$

- **Ergodic principle**

It is equivalent to compute:

- The time average of the quantity

$$\langle X(t) \rangle = \frac{1}{\tau} \int_{t_0}^{t_0+\tau} X(t) dt$$

- The statistical average of the quantity

$$\overline{X(r_N, p_N)} = \int \dots \int X(r_N, p_N) P(r_N, p_N) dr_N dp_N$$

$$\overline{a(r_N, p_N)} = \langle a(t) \rangle$$

Principles of statistical physics: ensembles

- **Accessible microscopic states** depend on the external parameters applied
- **Statistical ensembles**
 - (N,V,E): microcanonical ensemble – isolated systems
 - (N,V,T): canonical ensemble – equilibrium with a thermostat
 - (N,P,T): isothermal-isobaric ensemble – equilibrium with a thermostat and a barostat
 - (μ ,V,T): grand-canonical ensemble – equilibrium with a thermostat and a molecular reservoir
- **Probabilities P_i of microstates depend on the ensemble**

Principles of statistical physics: partition function

- Canonical ensemble: (N, V, T)
 - Accessible microstates have the same volume and number of particles
 - **But** the energies E_i of the accessible microstates can vary.
- The probability of a microstate i depends on its energy:

$$P_i = \frac{\exp\left(\frac{-E_i(r_N, p_N)}{k_B T}\right)}{Q(N, V, T)}$$

- The partition function: normalisation factor of the probability

$$Q(N, V, T) = \frac{1}{h^{3N} N!} \int \dots \int \exp\left(\frac{-E_i(r_N, p_N)}{k_B T}\right) dr_N dp_N$$

The partition function contains all the information necessary to compute average quantities

Principles of statistical physics: partition function

$$E_i(r_N, p_N) = K_i(p_N) + U_i(r_N)$$

Kinetic energy

$$K_i(p_N) = \sum_j \frac{p_j^2}{2m_j}$$

Potential energy:
molecular interactions
Internal degrees of freedom

$$Q(N, V, T) = \left[\frac{1}{h^{3N} N!} \int \exp\left(\frac{-K_i(p_N)}{k_B T}\right) dp_N \right] \times \left[\int \exp\left(\frac{-U_i(r_N)}{k_B T}\right) dr_N \right]$$

“Ideal” partition function Q_{id}

Configurational partition
function: Q_c

Ideal and configurational partition functions

- Ideal partition function

- Easy to compute
- Analytical expression:

$$Q_{id} = \prod_j \left(\frac{2\pi m_j k_B T}{h^2} \right)^{3/2} = \prod_j \frac{1}{\Lambda_j^3}$$

- Configurational partition function

- Comes from intermolecular forces and internal degrees of freedom
- Analytical expression available only for few models: hard spheres, van der Waals fluids
- Need to be evaluated numerically

Numerical evaluation of Q_c

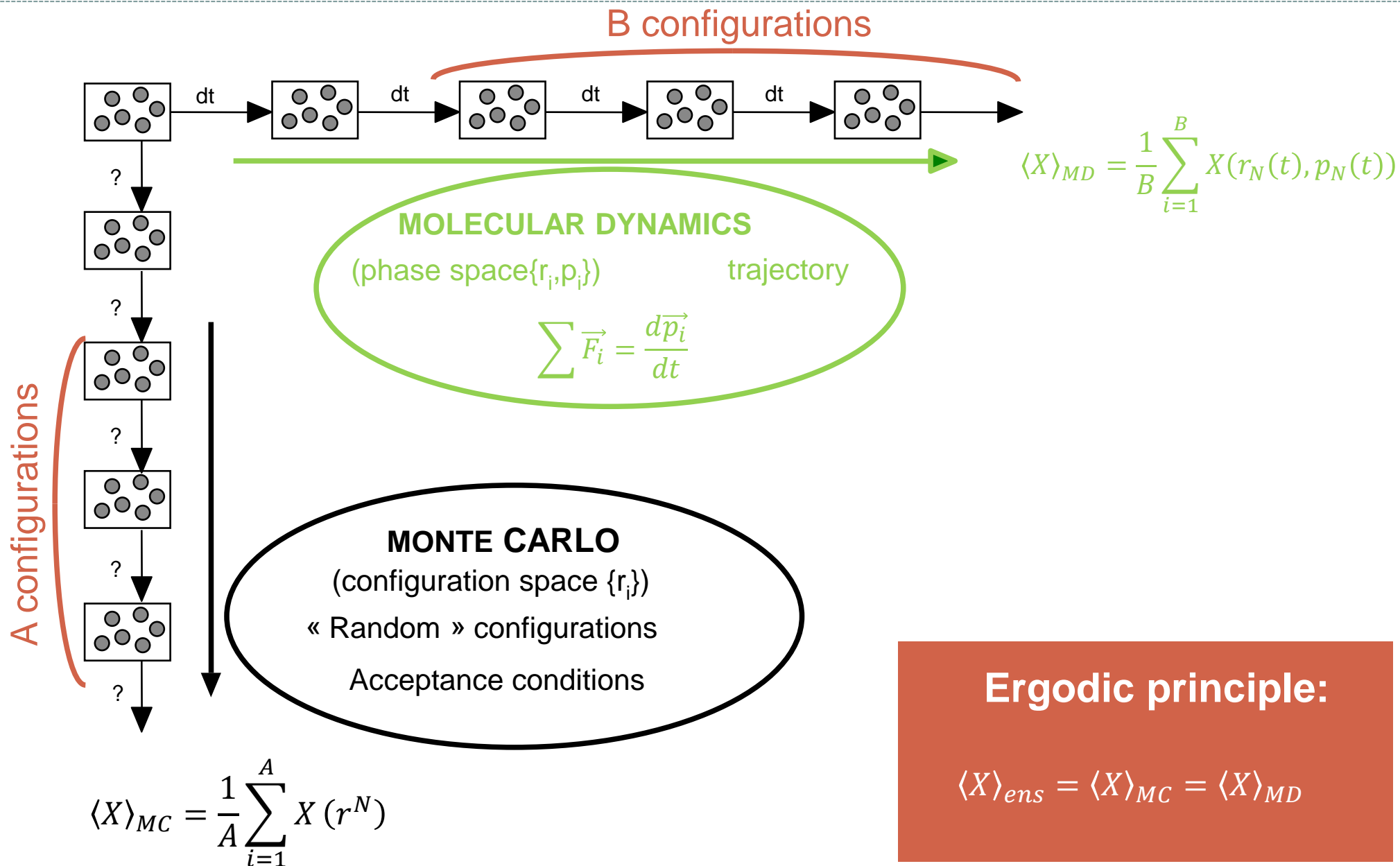
- Direct evaluation:
Generation of all the possible configurations of the system
- Over all the possible configurations, only few have a non-negligible contribution to Q_c

 Impossible because of high dimensionality

 Generation of **representative** configurations

Molecular simulation

Molecular simulation principles



Brief history of molecular simulations

- Around 1950** Use of the first computers for civil applications
- 1953** First simulation of a liquid by Monte Carlo (hard spheres)
Métropolis, Rosenbluth et Teller
- 1956** Molecular dynamics Simulation of hard disks
(Alder et Wainright)
- 1957** Monte Carlo Simulation of a Lennard-Jones liquid
- 1964** Molecular dynamics Simulation of condensed
Argon (liquid) (Rahman)

From then and until now

Strong and continuous evolution of algorithms
Simulation of complex mixtures, polar molecules, heterogenous systems

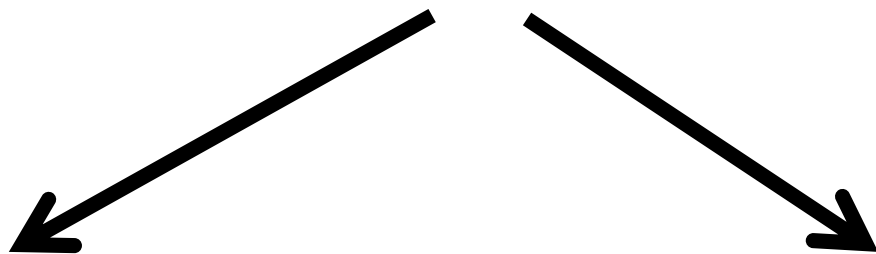
Molecular dynamics simulations

Basics of molecular dynamics simulations

- Particles are submitted to inter-atomic potentials U
- Framework of classical mechanics: Newton's laws for the nuclei

$$\vec{F}_i(t) = m_i \vec{a}_i(t) = -\vec{\nabla}_i U(\vec{r}(t))$$

Expression of the potential U :



Quantum mechanics:
Ab-initio MD

Classical mechanics:
(Classical) MD

Expression of U : forcefield

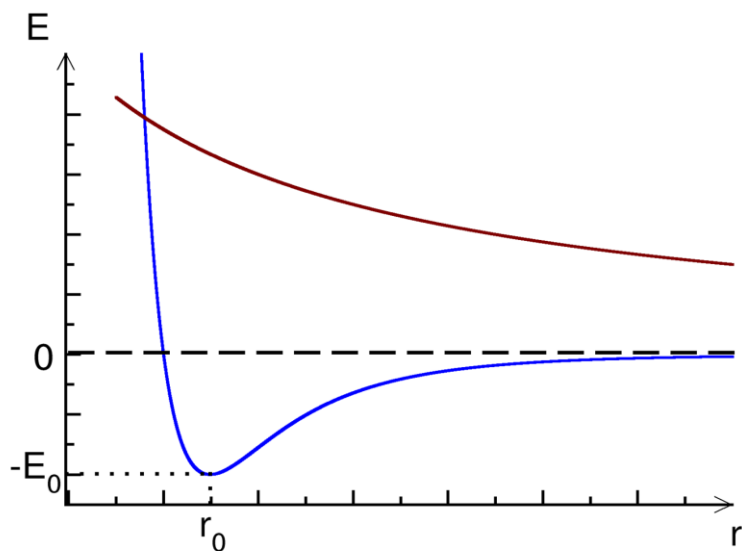
Classical forcefields

$$\vec{F}_i(t) = m_i \vec{a}_i(t) = -\vec{\nabla}_i U(\vec{r}(t))$$

Expression of the potential U:

- Two-body interactions
- Analytical expression
- Differentiable function

Forcefield



$$V_{coul}(r) = \frac{q_i q_j}{4\pi\epsilon_0 r}$$

Charge-charge interactions

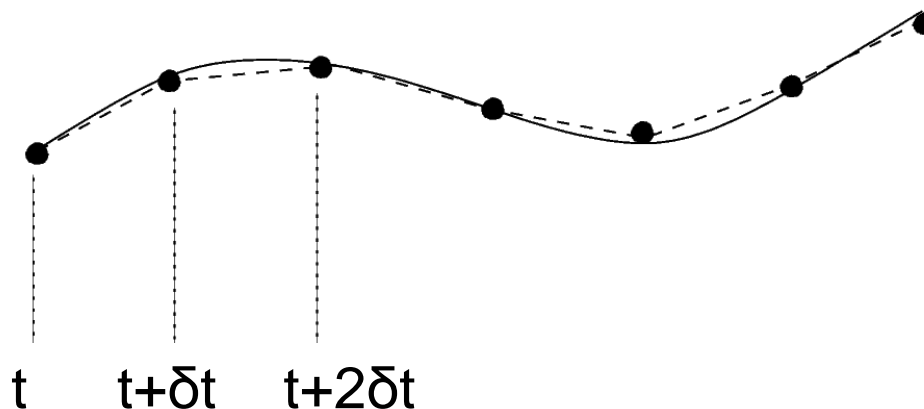
$$V_{LJ}(r) = 4\epsilon \left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^6 \right]$$

Dipole-dipole interactions

Integration of the equations of motion

Finite difference methods:

Discretization of the trajectory and numerical integration



$$\vec{r}(t + \delta t) = \vec{r}(t) + \delta t \vec{v}(t) + \frac{1}{2} (\delta t)^2 \vec{a}(t) + \dots$$

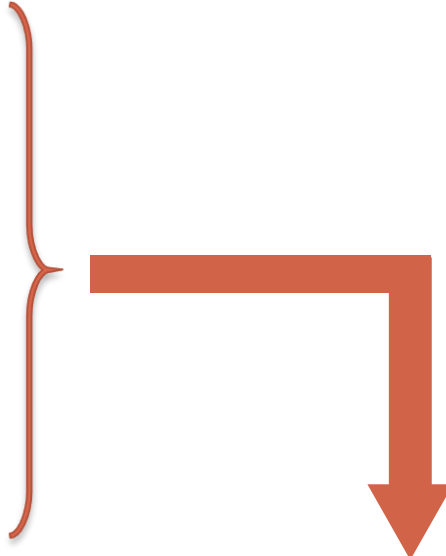
$$\vec{v}(t + \delta t) = \vec{v}(t) + \delta t \vec{a}(t) + \dots$$

Equations of motion: the Verlet algorithm (1967)

$$\vec{r}(t + \delta t) = \vec{r}(t) + \delta t \vec{v}(t) + \frac{1}{2} (\delta t)^2 \vec{a}(t)$$

$$\vec{r}(t - \delta t) = \vec{r}(t) - \delta t \vec{v}(t) + \frac{1}{2} (\delta t)^2 \vec{a}(t)$$

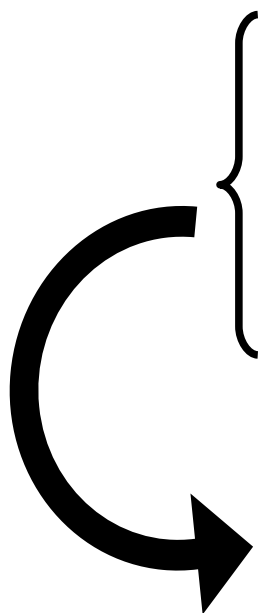
$$\vec{a}(t) = -\vec{\nabla}U(t)$$


$$\vec{r}(t + \delta t) = 2\vec{r}(t) - \vec{r}(t - \delta t) + (\delta t)^2 \vec{a}(t)$$

$$\vec{v}(t) = [\vec{r}(t + \delta t) - \vec{r}(t - \delta t)] / 2 \delta t$$

Equation of motions: other algorithms

Leap-frog algorithm



$$\vec{r}(t + \delta t) = \vec{r}(t) + \delta t \vec{v}\left(t + \frac{\delta t}{2}\right)$$

$$\vec{v}\left(t + \frac{\delta t}{2}\right) = \vec{v}\left(t - \frac{\delta t}{2}\right) + \delta t \vec{a}(t)$$

$$\vec{v}(t) = \frac{1}{2} \left[\vec{v}\left(t + \frac{\delta t}{2}\right) + \vec{v}\left(t - \frac{\delta t}{2}\right) \right]$$

Velocity-Verlet algorithm

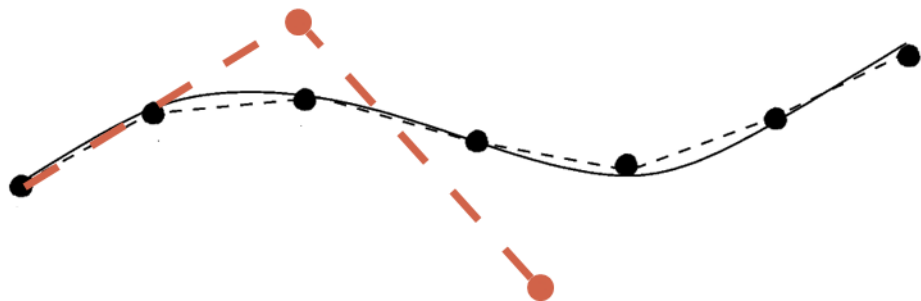
$$\vec{r}(t + \delta t) = \vec{r}(t) + \delta t \vec{v}(t) + \frac{1}{2} (\delta t)^2 \vec{a}(t)$$

$$\vec{v}(t + \delta t) = \vec{v}(t) + \frac{\delta t}{2} [\vec{a}(t) + \vec{a}(t + \delta t)]$$

And others....

Integration in practice

- Which integration scheme should be used?
 - Good energy conservation
 - Depends on the time step value δt
- Choice of the time step
 - Small: good accuracy but computationally expensive
 - Big: poor accuracy for the trajectory but cheap



Compromise between accuracy
and computation time

More about the time step...

- Not too small but not too big, but how much?
- Typically one order of magnitude lower than the period of the fastest motion in the system

Example 1: argon atoms

Characteristic distance: $d \approx 0.3\text{nm}$

Characteristic speed:

$$v \approx \sqrt{\frac{3k_B T}{m}} \approx 350\text{m}\cdot\text{s}^{-1}$$

Characteristic time: $\tau \approx \frac{d}{v} \approx 10^{-13}\text{s}$

➔ Time step $\delta t \approx 10^{-14}\text{s}$

Example 2: water molecules

Fastest motion:
vibration of O-H bond

Period of the vibration:

$$\nu \approx 4000\text{cm}^{-1} \Rightarrow T \approx 10^{-14}\text{s}$$

➔ Time step $\delta t \approx 10^{-15}\text{s}$

Statistical ensembles and molecular dynamics

All intermolecular forces are conservative



Total energy of the system constant:
(N,V,E) microcanonical ensemble



T is an output of the simulation: $T = \frac{2\langle K \rangle}{3Nk_B}$

Is it possible to fix T and to sample in the (N,V,T) or (N,P,T) ensemble ?

Constant temperature molecular thermodynamics

- Fixing temperature to $T_0 =$ Fixing kinetic energy

$$K = \sum_j \frac{1}{2} m_j v_j^2 \quad \longrightarrow \quad T = \frac{2}{3Nk_B} \sum_j \frac{1}{2} m_j v_j^2$$

- If $T \neq T_0$, modification of the velocities:

$$v_j \rightarrow (\lambda v_j) \quad \longrightarrow \quad K' = \sum_j \frac{1}{2} m_j (\lambda v_j)^2 \quad \longrightarrow \quad T' = \lambda^2 T$$

$$\lambda = \sqrt{\frac{T_0}{T}}$$

- In practice, this method is not used anymore and more sophisticated algorithm are used, but the idea of “modifying” the velocities of particles

How to run an MD simulation?

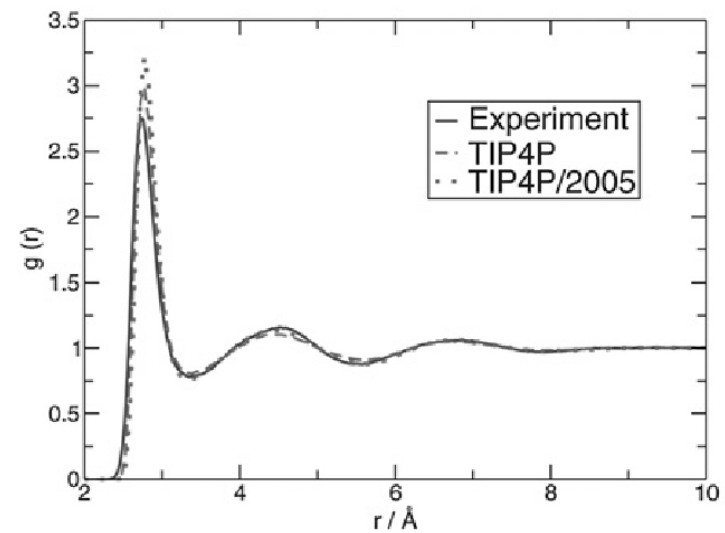
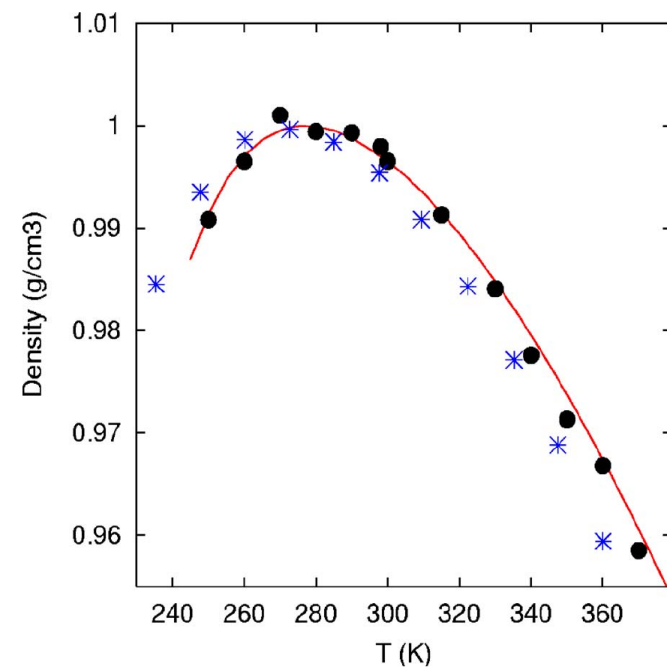
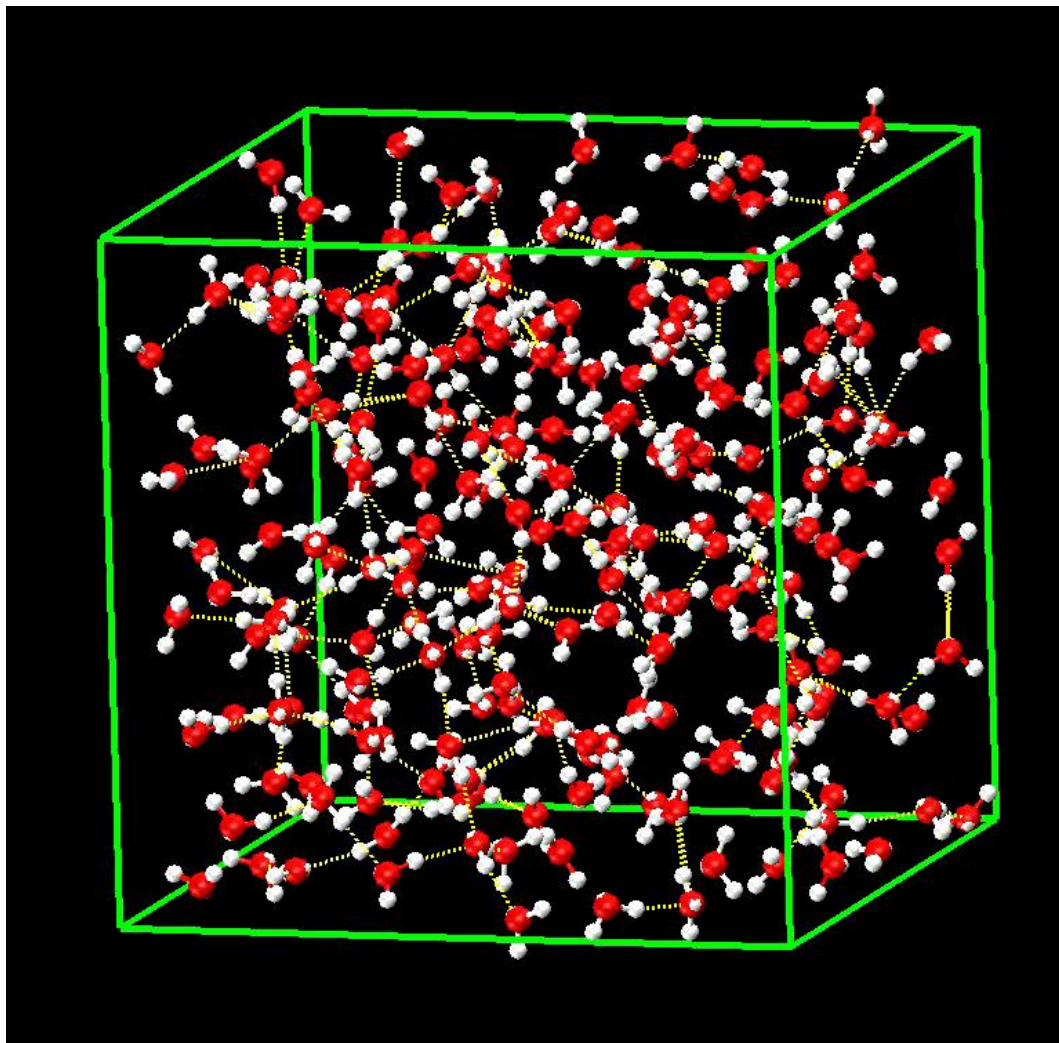
- System definition
 - Statistical ensemble
 - Choice of the forcefield
 - Temperature, density,...
- Setting up the initial “configuration”
 - Initial coordinates
 - Initial velocities (linked with temperature T)
- Calculation of forces applied to each particle
- Integration of the equations of motion



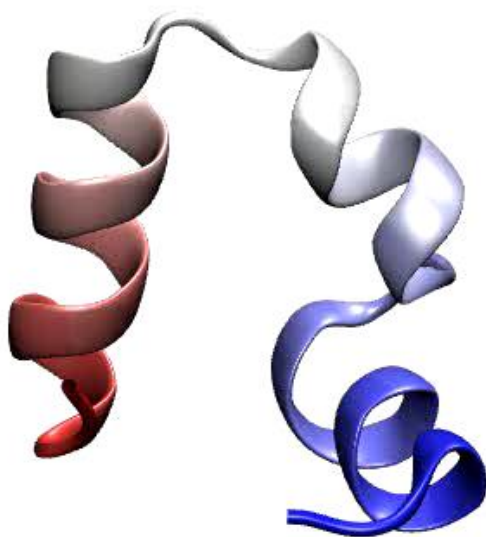
New coordinates and velocities



Few examples in motion: fluid simulation



Few examples in motion: protein folding simulation



0.0 ns

Monte Carlo simulations

Monte Carlo: the principles

- Calculation of the ensemble average :

- Random generation of M configurations i of the system
- Calculation of the probability P_i of each configuration
- Estimation of the ensemble average:

$$A = \sum_{i=1}^M P_i a_i$$

- Drawback: many configurations have a negligible weight in the average

➡ M should be very big

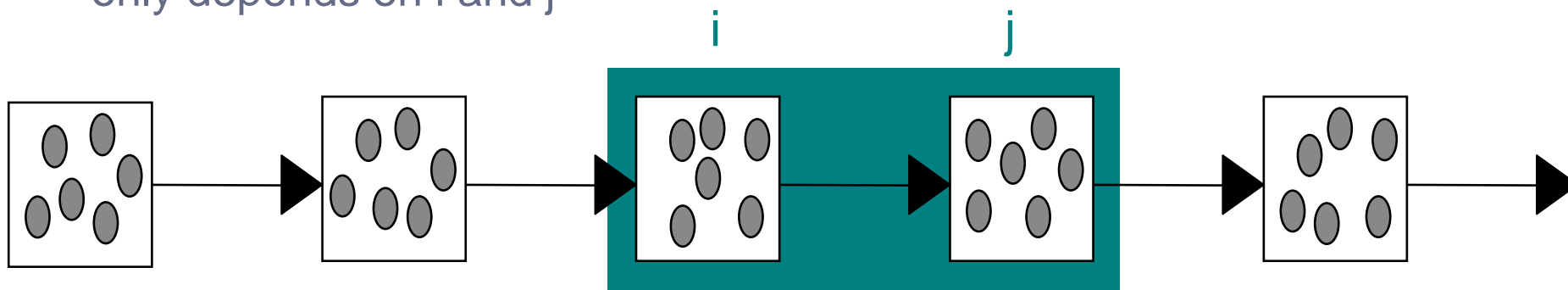
- Alternative: importance sampling

- Generation of M' configurations that follow the Boltzmann probability distribution $\{P_i\}$
- Direct calculation of the ensemble average:

$$A = \frac{1}{M'} \sum_{i=1}^{M'} a_i$$

Monte Carlo: Markov chain and microreversibility

- How to generate configurations following the probability distribution $\{P_i\}$?
- Markov chain:
 - Creation of successive configurations
 - **Random** creation of each configuration
 - The probability $W(i \rightarrow j)$ to go from configuration i to the next configuration j only depends on i and j



- If i is the configuration at step t , the configuration at step $(t+1)$ may then be:
 - Either the same configuration i
 - Or the new configuration j

Monte Carlo: Markov chain and microreversibility

- Probability to be in configuration i at step $(t+1)$:

$$P(i, t + 1) = \underbrace{\left[P(i, t) - \sum_{k \neq i} P(i, t) W(i \rightarrow k) \right]}_{\text{Probability to stay in configuration } i} + \underbrace{\sum_{k \neq i} P(k, t) W(k \rightarrow i)}_{\text{Probability to arrive in configuration } i \text{ from another configuration}}$$

- Equilibrium condition (stationnarity):
At equilibrium, the probability to be in configuration i does not depend on the step, but only on its Boltzmann probability: $P(i, t+1) = P(i, t) = P(i)$

$$\sum_{k \neq i} P(i) W(i \rightarrow k) = \sum_{k \neq i} P(k) W(k \rightarrow i)$$

Probability of i in the statistical ensemble considered

Transition probability to go from configuration i to configuration k in the Markov chain

- Sufficient condition: microreversibility

$$\forall (i, k), P(i) W(i \rightarrow k) = P(k) W(k \rightarrow i)$$

Monte Carlo: Metropolis algorithm

- **Aim of the algorithm:** determine the transition probabilities so that configurations are visited with the right probability

- **Transition probability $W(i \rightarrow k)$:**

$$W(i \rightarrow k) = P_{gen}(i \rightarrow k) \cdot P_{acc}(i \rightarrow k)$$

- **Micro-reversibility condition:**

$$\forall(i, k), P(i)P_{gen}(i \rightarrow k)P_{acc}(i \rightarrow k) = P(k)P_{gen}(k \rightarrow i)P_{acc}(k \rightarrow i)$$

- **Metropolis criteria:**

- Random generation of the configurations: $P_{gen}(i \rightarrow k) = P_{gen}(k \rightarrow i)$
- Acceptation probability:

$$P_{acc}(i \rightarrow k) = 1 \quad \text{if} \quad P_k > P_i$$

$$P_{acc}(i \rightarrow k) = \frac{P_k}{P_i} \quad \text{if} \quad P_k < P_i$$



$$P_{acc}(i \rightarrow k) = \min\left(1, \frac{P_k}{P_i}\right)$$

The Metropolis algorithm in the canonical ensemble

- Probability of a configuration i :

$$P_i = \exp\left(-\frac{E_i}{k_B T}\right)$$

- Acceptance probability

$$P_{acc}(i \rightarrow j) = \min\left(1, \exp\left(-\frac{E_j - E_i}{k_B T}\right)\right)$$

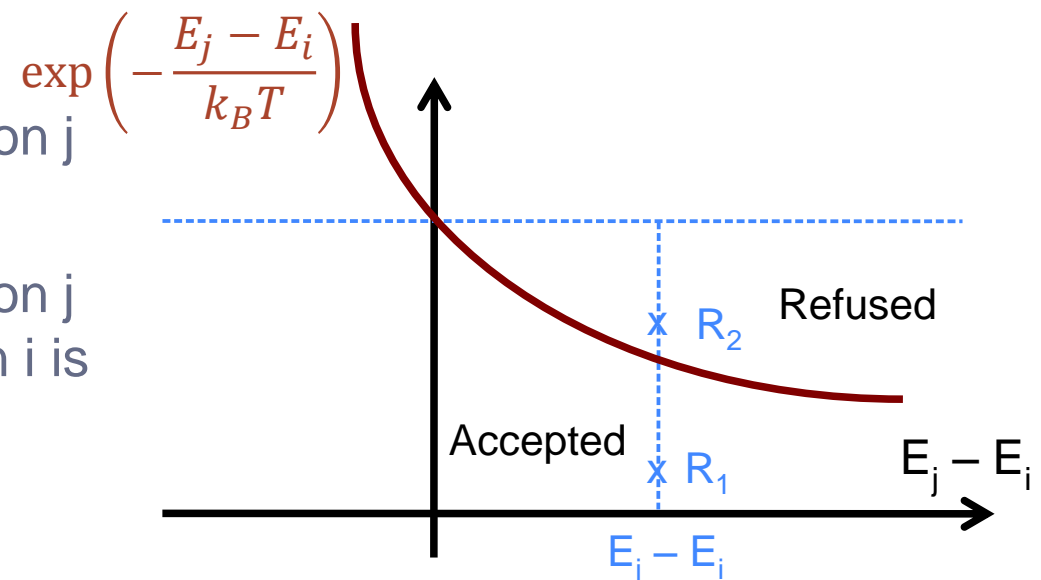
Monte Carlo: Metropolis algorithm in practice

- Calculation of the energy E_i of the current configuration i
- Construction of a new possible configuration j from configuration i
- Calculation of the energy E_j of the new configuration j
- Calculation of:

$$R(i \rightarrow j) = \exp\left(-\frac{E_j - E_i}{k_B T}\right)$$

- Generation of a random number R from a uniform distribution between 0 and 1

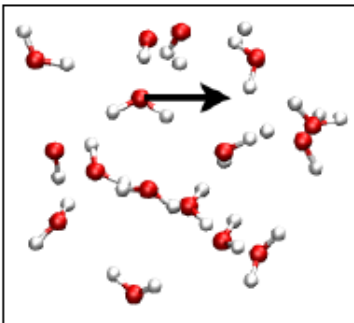
- $R < R(i \rightarrow j)$: the new configuration j is accepted
- $R > R(i \rightarrow j)$: the new configuration j is rejected and the configuration i is conserved



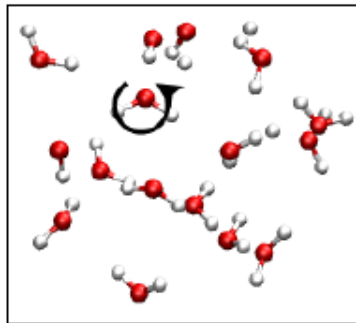
Creation of the new configuration

- If the configuration j is created randomly: few chances that $R(i \rightarrow j)$ will be high.
- Configuration j build from configuration i with “small” modifications (moves):

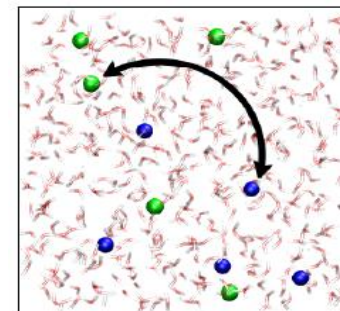
Translation



Rotation



Exchange



Moves can be « non-physical »

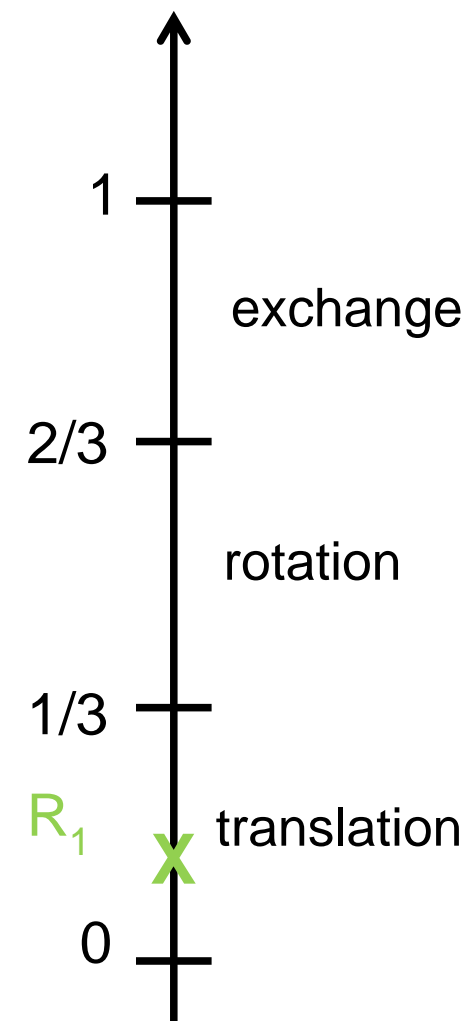
- Advantages:
 - Energy difference ($E_i - E_j$) easy to compute
 - Energy difference small \rightarrow “high” probability of accepting the move

Generating a new configuration in practice (1)

- Selection of a type of move:
 - Each type of move has a probability $P(M)$ (sum to 1)
 - Generation of a random number R_1 between 0 and 1
 - Move M selected if:

$$\sum_{j=1}^{M-1} P(j) < R_1 < \sum_{j=1}^M P(j)$$

 Translation

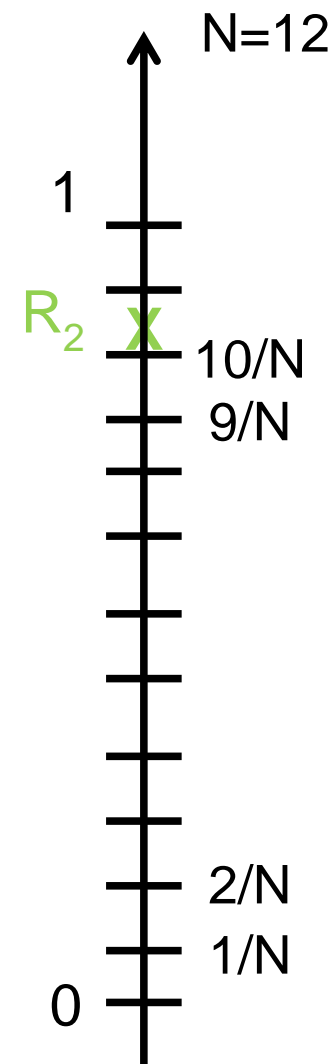


Generating a new configuration in practice (2)

- Selection of a type of move:
- Select a molecule to which the move is applied:
 - Generate a random number R_2 between 0 and 1
 - Chose molecule k so that:

$$\frac{k-1}{N} < R_2 < \frac{k}{N}$$

→ $k = 10$



Generating a new configuration in practice (3)

- Selection of a type of move:
- Select a molecule to which the move is applied:
- Apply the selected move to molecule k:
 - Generate random numbers (3 for a translation: ξ_x, ξ_y, ξ_z)
 - Move the molecule to create the trial configuration j

$$x_k(j) = x_k(i) + (2\xi_x - 1)\delta r$$

$$y_k(j) = y_k(i) + (2\xi_y - 1)\delta r$$

$$z_k(j) = z_k(i) + (2\xi_z - 1)\delta r$$

Monte Carlo simulations and random numbers

- Monte Carlo simulations make use of a large amount of random numbers
- Example: one trial configuration with a translation move
 - 1 number to “choose” the translation move
 - 1 number to “choose” the molecule to which the translation is applied
 - 3 random numbers to generate the amount of translation
 - 1 random number to determine the acceptance/rejection of the configuration
 - Total: 6 random numbers for only one trial move (10-1000 billions moves in one typical MC simulation)
- Need for “good” random number generators !

Monte Carlo simulations and statistical ensembles

- Monte Carlo simulations well suited for canonical ensembles:
 - Typical moves: translation, rotation
 - Biased moves for complex molecules (reptation, regrowth,...) to accelerate convergence
 - Acceptance probability: Metropolis criterion

$$P_{acc}(i \rightarrow j) = \min \left(1, \exp \left(\frac{E_i - E_j}{k_B T} \right) \right)$$

- But Monte Carlo algorithm can be adapted to other ensembles:
 - Isothermic-isobaric ensemble (N,P,T)
 - Grand-canonical ensemble (μ, V, T)
 - And others...



New moves
New Boltzmann
probabilities
New acceptance
probability

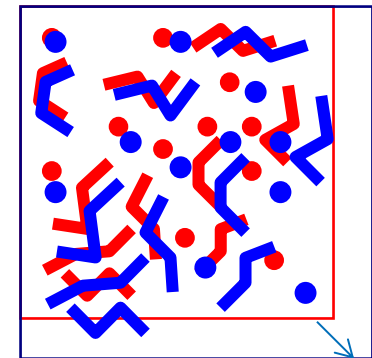
Monte Carlo simulations in (N,P,T) ensemble

- The volume of the system can fluctuate between two configurations

—————→ New move: volume change

$$V_j = V_i + (2\xi - 1)\delta V$$

—————→ Multiplication of every coordinate by a same factor



- Probability of one configuration:

$$P(i) \propto \frac{1}{N!} \left(\frac{V_i}{\Lambda}\right)^N \exp\left(\frac{-E_i + PV_i}{k_B T}\right)$$

- Probability of accepting the new configuration:

$$P_{acc}(i \rightarrow j) = \min\left(1, \left(\frac{V_k}{V_i}\right)^N \exp\left(\frac{E_i - E_k}{k_B T} + \frac{P(V_k - V_i)}{k_B T}\right)\right)$$

Monte Carlo simulations in (μ, V, T) ensemble

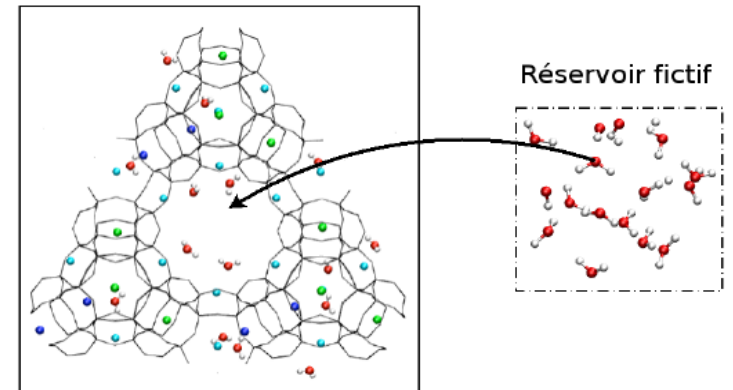
- Grand-canonical ensemble adapted to open systems, where the number of particles N can fluctuate:

→ Two new moves:

insertion ($N_k = N_i + 1$) and deletion ($N_k = N_i - 1$) of particles

- Probability of one configuration:

$$P(i) \propto \frac{1}{N_i!} \left(\frac{V}{\Lambda}\right)^{N_i} \exp\left(\frac{-E_i - \mu N_i}{k_B T}\right)$$



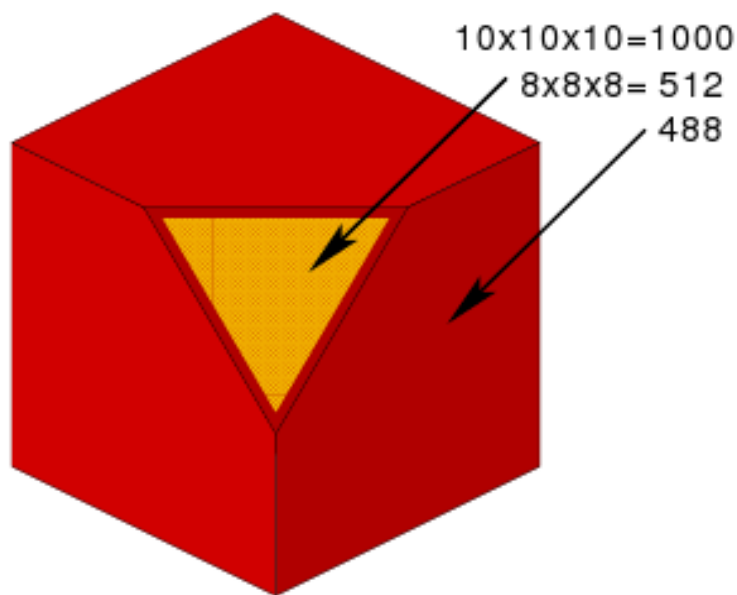
- Probabilités d'accepter la nouvelle configuration :

N_i	→	{	$N_k = N_i + 1$ insertion	$P_{acc}(i \rightarrow k) = \min\left(1, \frac{V}{\Lambda^3(N_i + 1)} \exp\left(\frac{E_i - E_k}{k_B T} + \frac{\mu}{k_B T}\right)\right)$
			$N_k = N_i$ rotation, translation, ...	$P_{acc}(i \rightarrow k) = \min\left(1, \exp\left(\frac{E_i - E_k}{k_B T}\right)\right)$
			$N_k = N_i - 1$ deletion	$P_{acc}(i \rightarrow k) = \min\left(1, \frac{N_i \Lambda^3}{V} \exp\left(\frac{E_i - E_k}{k_B T} - \frac{\mu}{k_B T}\right)\right)$

Practical aspects of molecular simulations

Periodic boundary conditions

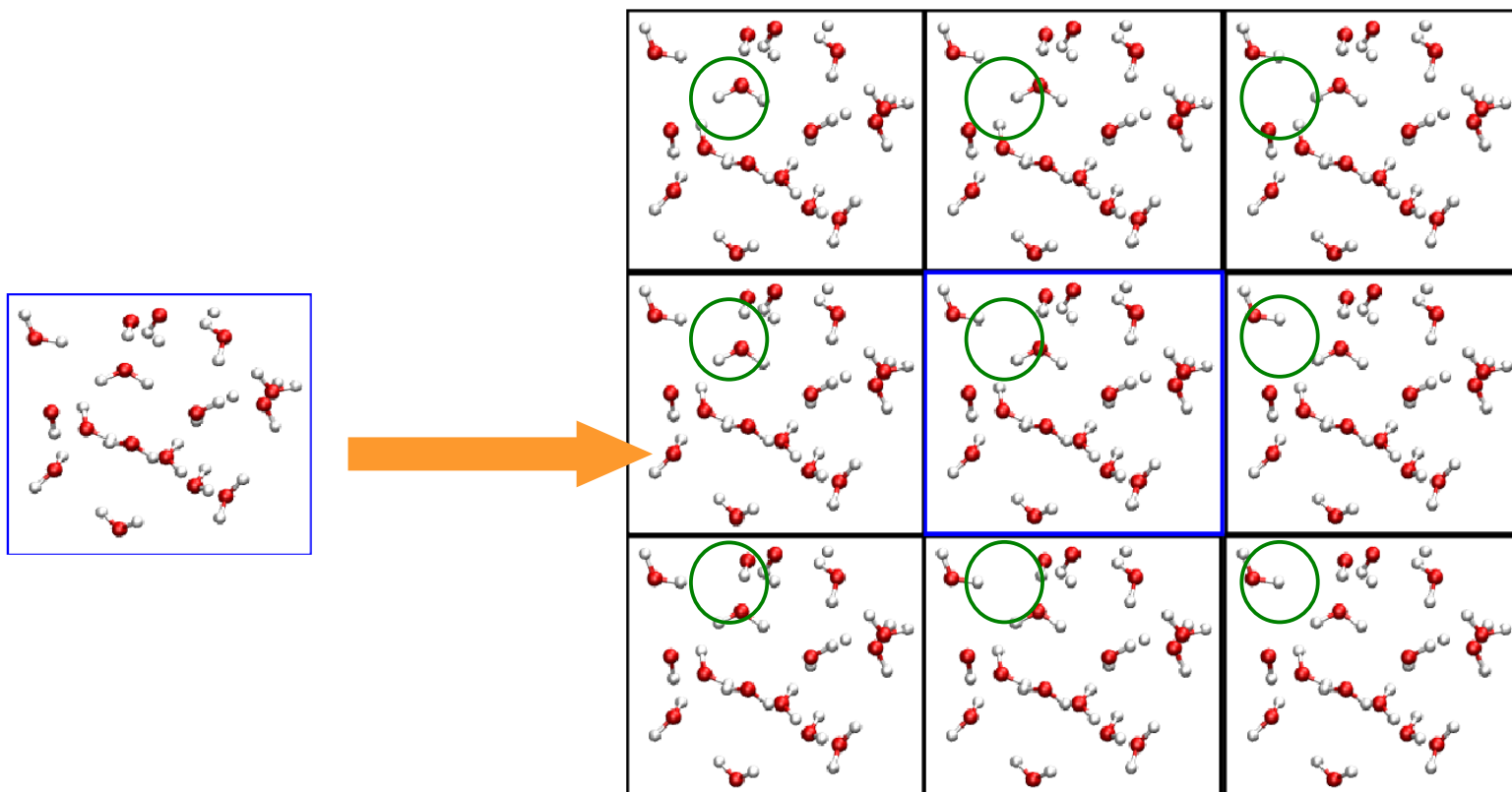
- Computational time for energy proportional to N^2
 - Limitation of system size typically to 10^3 - 10^6 atoms
 - Surface effect problems



- Example: regular cubic network of 1000 atoms ($10 \times 10 \times 10$)
 - ~ 50% of the atoms on the surface !
- Even with 10^6 atoms ($100 \times 100 \times 100$):
 - $98 \times 98 \times 98 = 941192$ atoms « in the heart » of the system
 - ~6% of the atoms on the surface

Periodic boundary conditions

- **Periodic conditions:**
Replication of the cell in every direction of space:

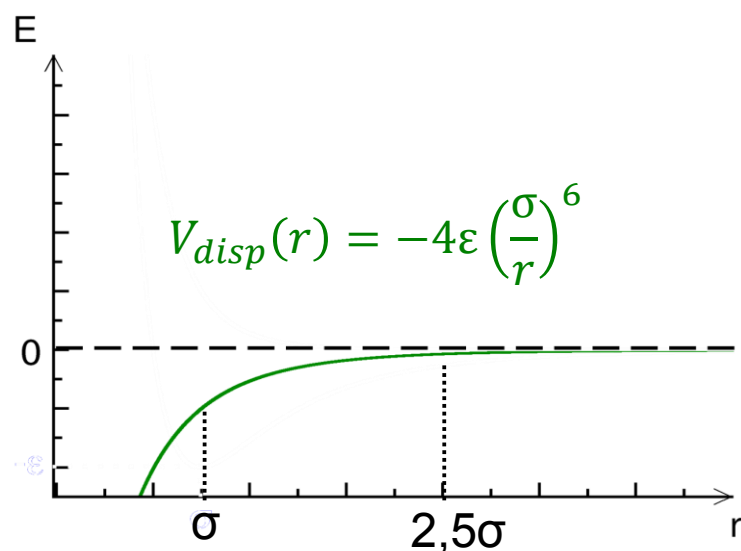


- **Creation of a crystalline network:**
 - Suppression of surface effects
 - Artefacts due to the artificial periodicity?

Potential truncation and minimum image convention

- Intermolecular potentials: mathematical function in $(1/r)^n$
- If $n > 2$, short-range interactions

Example: dispersion interactions



$$V_{disp}(\sigma) = -4\epsilon$$

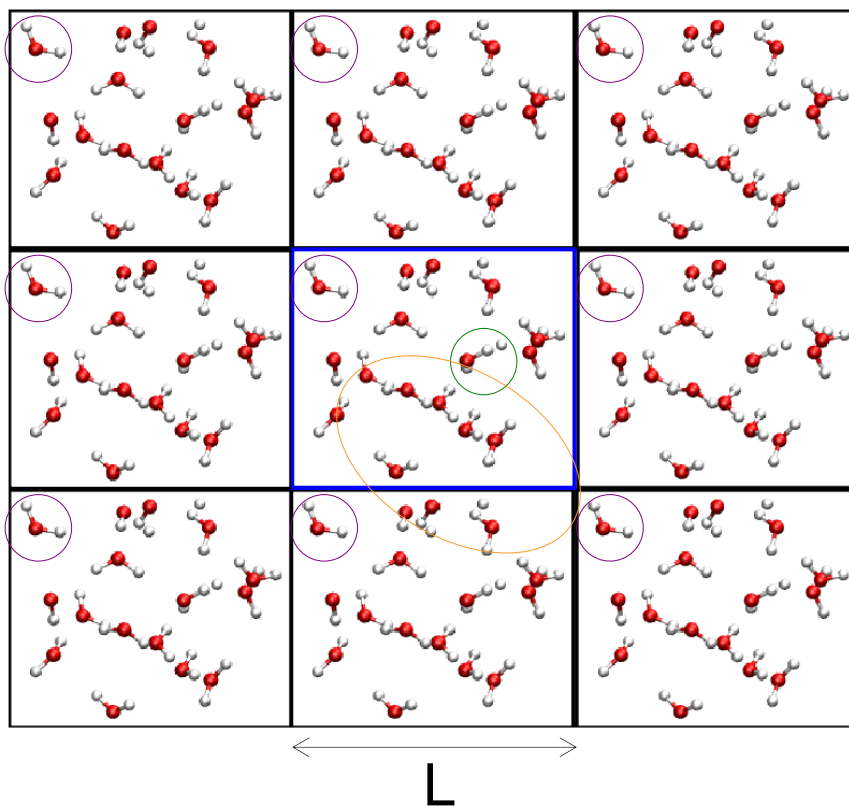
$$V_{disp}(2.5\sigma) = \frac{V_{disp}(\sigma)}{2.5^6} \approx 0.005V_{disp}(\sigma)$$

σ : few Å

- Introduction of a cutoff radius r_c in the calculation of interactions
- Which value for r_c ?

Potential truncation and minimum image convention

- **Minimum image convention:**
One molecule should feel the influence of only one image of each molecule of the system (the closest one)
- Restriction on the value of r_c



$$r_c < \frac{L}{2}$$

- Lower the periodicity effects

Potential truncation in practice

- Chose the cutoff radius r_c
- Build a neighbour-list for each atom i that includes all the atoms lying at a distance less than r_c from i .

- Compute the interaction for atom i :

- For each atom j in the neighbour-list:
$$V_{ij} = 4\epsilon \left[\left(\frac{\sigma}{r_{ij}} \right)^{12} - \left(\frac{\sigma}{r_{ij}} \right)^6 \right]$$

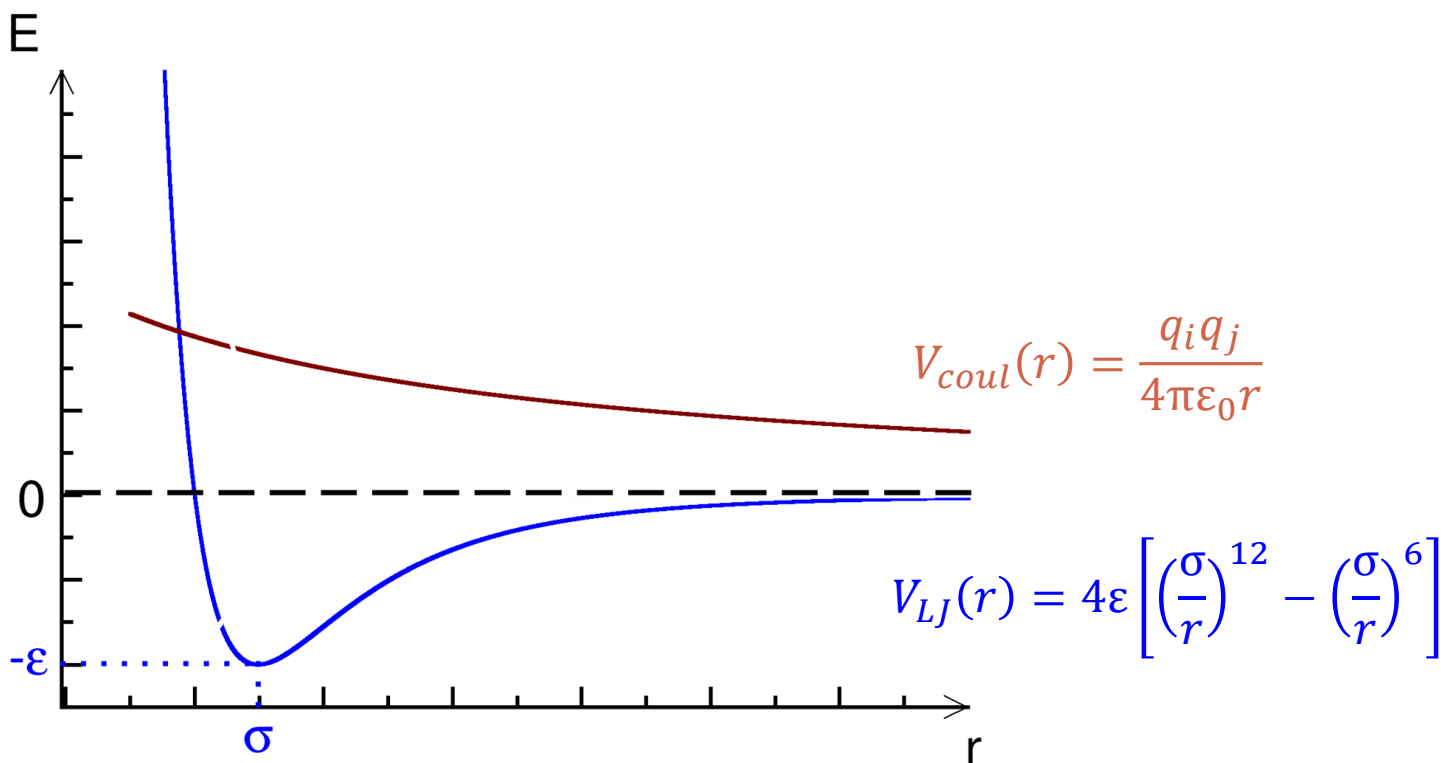
- For any other atom k :
$$V_{ik} = 0$$

- Add a long-range correction:
$$V_{corr} = \rho \int_{r_c}^{\infty} V(r) 4\pi r^2 dr = 8\pi\rho \left[\frac{\sigma^{12}}{6r_c^9} - \frac{\sigma^6}{3r_c^3} \right]$$

- The neighbour-list is rebuilt periodically

Long range forces

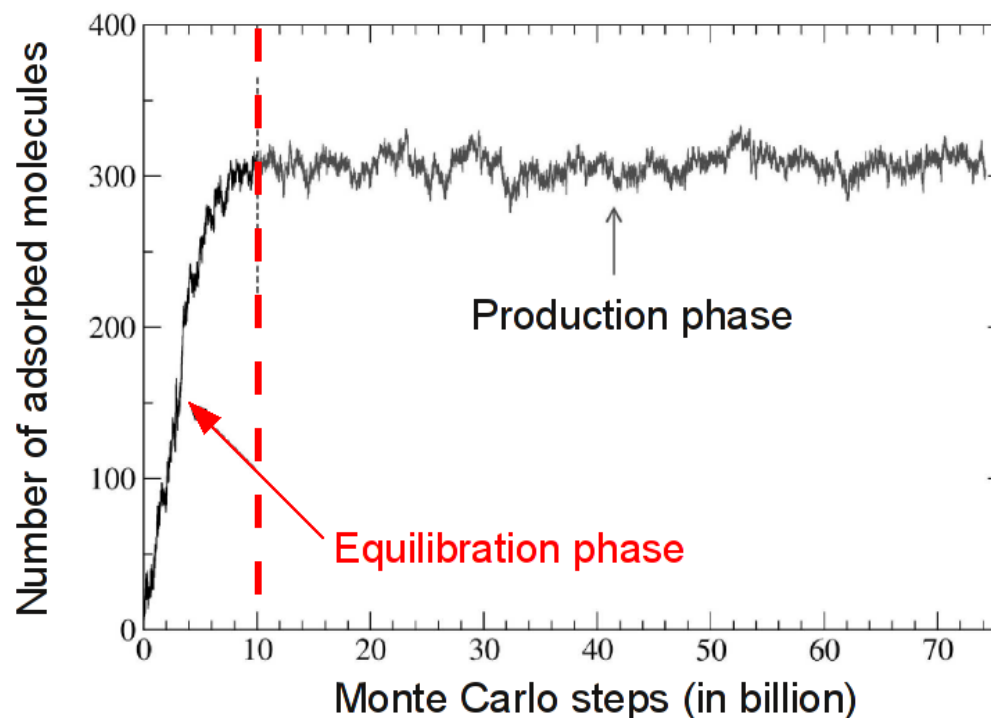
- Electrostatic interactions in $(1/r)$ are long-range interactions



- Using a cutoff is inappropriate – more sophisticated methods are used (Ewald summations)

Equilibration and production phases

- Initial configuration is most of the time not at equilibrium
- The first stage of the simulation is to reach equilibrium: this is the equilibration stage.
- Once equilibrium is reached, the production phase can begin, during which properties can be calculated



Which information is accessible with MC or MD?

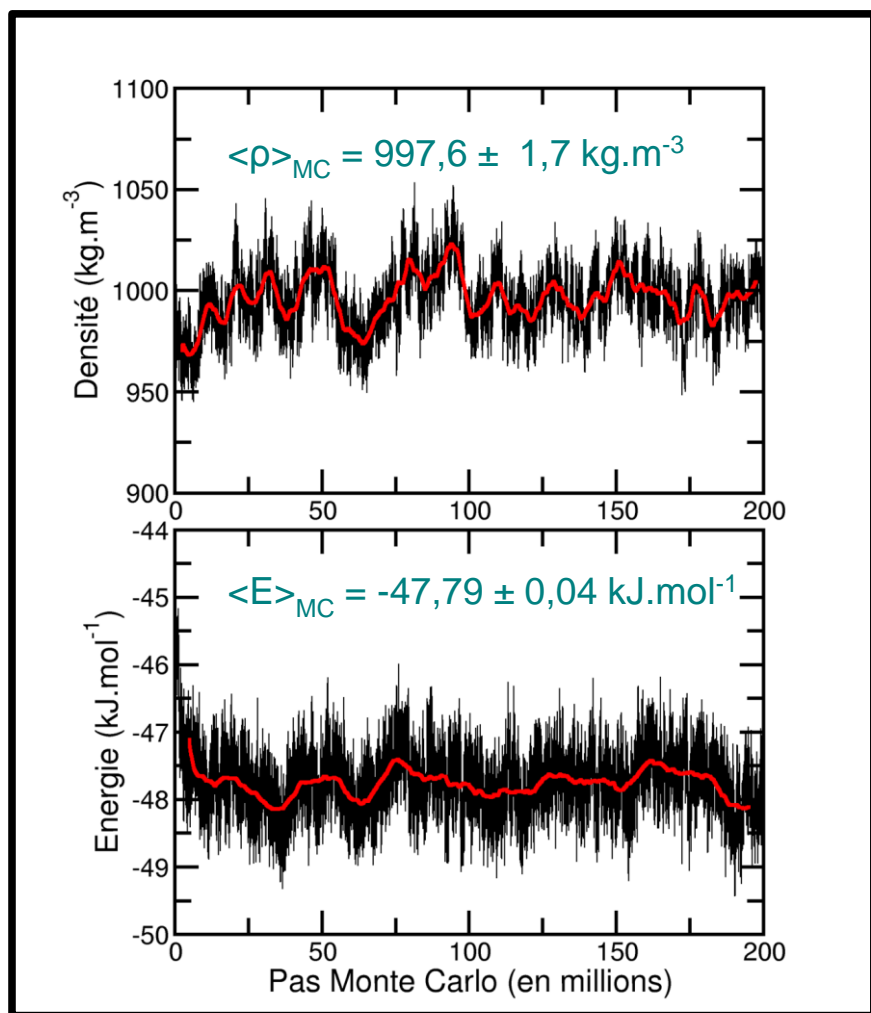
- **Structural data**
 - Equilibrium structure of a macromolecule
 - Fluid density in various (P,T) conditions
 - Structure of a fluid (radial distribution function $g(r)$)
- **Energetic data**
 - Binding affinities
 - Heat capacities
 - Thermoelastic coefficients (compressibility, Joule-Thomson coefficients)
- **Transport properties**
 - Diffusion coefficients
 - Viscosity
 - Thermal conductivity

.... and many others

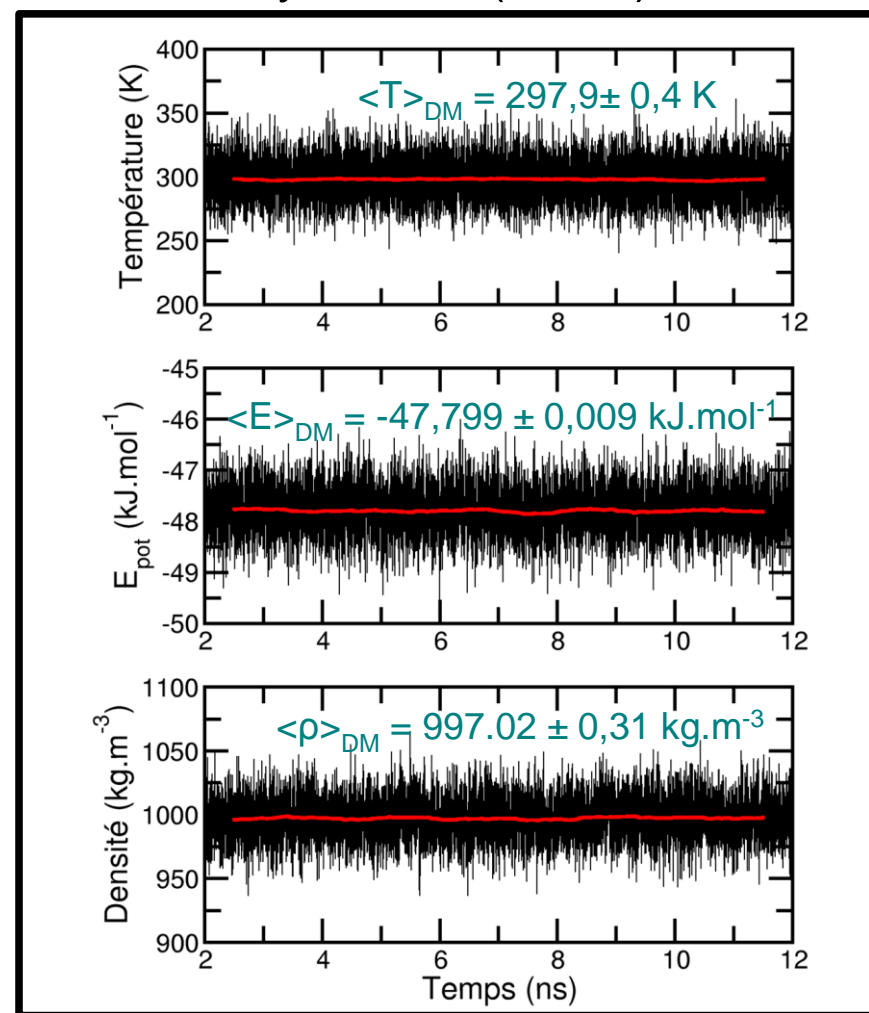
Monte Carlo or Molecular dynamics?

Water simulation at 298K and under a pressure of 1bar

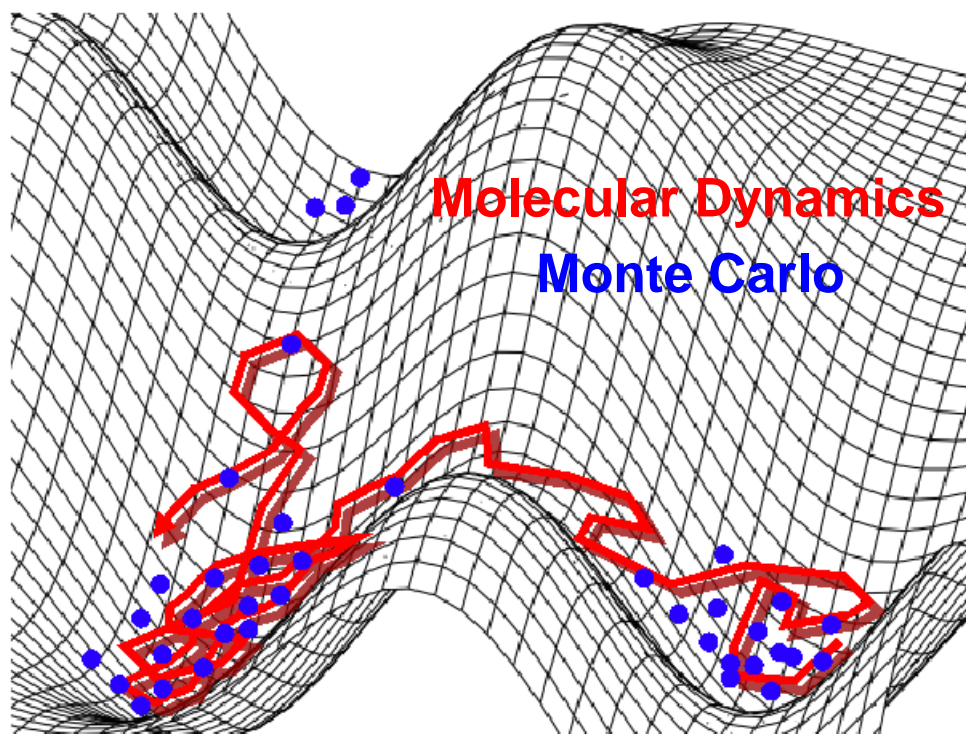
Monte Carlo (N,P,T) simulation



Molecular Dynamics (N,P,T) simulation



Monte Carlo or Molecular dynamics?



$$A = \langle a \rangle = \frac{\sum a_i}{N} = \lim_{t \rightarrow \infty} \frac{\int a(\tau) d\tau}{t}$$

- **Molecular dynamics:** temporal information, collective motions
- **Monte Carlo:** may be easier to overcome energy barriers, no need of thermostat or barostat (“exact” T or P)
- Hybrid MD/MC simulations are in development